ระบบวิเคราะห์และแสดงผลการทำงานเครื่องปรับอากาศ
Monitor & Analyze Air conditioner system

รัฐกูล อักษรกลาง
อลงกรณ์ ภูสิริพัฒนานนท์
ศุภกร ขึงโพธิ์

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต
สาขาวิศวกรรมการวัดคุม
คณะวิศกรรมศาตร์
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
ปีการศึกษา 2557

Monitor & Analyze Air conditioner system

Rattakoon Aksornklang
Alongkorn Poosiripattananon
Supakorn khungpo

A THESIS SUBMITTED IN PARTIAL FULFILLMENT
OF THE REQUIREMENT FOR THE DEGREE OF
BACHELOR OF ENGINEERING IN INSTRUMENTTATION ENGINEERING
FACUKTY OF ENGINEERING
KING MONGKUT'S INSTITUE OF TECHNOLOGY LARDKRABANG
ACADEMIC YEAR 2014

ปริญญานิพนธ์ปีการศึกษา 2557
คณะวิศวกรรมศาสตร์
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
ใบรับรองปริญญานิพนธ์

.....

หัวข้อปริญญานิพนธ์	ระบบวิเคราะห์และแสดงผลการทำงานของเครื่องปรับอากาศ Monitor & Analyze Air conditioner system			
นักศึกษาผู้จัดทำ	นายรัฐกูล	อักษรกลาง	**รหัสนักศึกษา**	54011078
	นายอลงกรณ์	ภูสิริพัฒนานนท์	**รหัสนักศึกษา**	54011528
	นายศุภกร	ขึงโพธิ์	**รหัสนักศึกษา**	54011284
ปริญญา	วิศวกรรมศาวตร์บัณฑิต			
สาขาวิชา	วิศวกรรมวัดคุม			
ปีการศึกษา	2557			

อาจารย์ผู้ควบคุมปริญญานิพนธ์	**ลายมือชื่อ**
อาจารย์เชื้อ นกอยู่	

หัวข้อปริญญานิพนธ์	ระบบวิเคราะห์และแสดงผลการทำงานของเครื่องปรับอากาศ Monitor & Analyze Air conditioner system			
นักศึกษาผู้จัดทำ	นายรัฐกูล	อักษรกลาง	**รหัสนักศึกษา**	54011078
	นายอลงกรณ์	ภูสิริพัฒนานนท์	**รหัสนักศึกษา**	54011528
	นายศุภกร	ขึงโพธิ์	**รหัสนักศึกษา**	54011284
อาจารย์ที่ปรึกษา	อาจารย์เชื้อ	นกอยู่		
ปีการศึกษา	2557			

บทคัดย่อ

ปริญญานิพนธ์ฉบับนี้เป็นการศึกษาเกี่ยวกับการวัดและแสดงผลสถานภาพการทำงานของเครื่องปรับอากาศ ณ เวลาขณะนั้นได้ เพื่อที่จะได้ทราบการทำงานของเครื่องปรับอากาศว่ามีการทำงาoเป็นไปตามค่ามาตรฐานอุปกรณ์ตามแต่ละยี่ห้อ,บริษัทกำหนดไว้หรือเมื่อ เมื่อเครื่องปรับอากาศทำงานได้ไม่เต็มประสิทธิภาพหรือมีข้อผิดพลาดที่ไม่เป็นไปตามค่ามาตรฐานที่ตั้งไว้ก็สามารถที่จะระบุได้ว่าเครื่องปรับอากาศตัวใดเกิดการทำงานที่ผิดพลาดและสามารบอกได้ในขึ้นต้นว่าความเสียหายเกิดขึ้นที่จุดใด ซึ่งสามรถลดขึ้นตอนการตรวจสอบและตรวจหาอุปกรณ์ที่เสียหายได้ ซึ่งจุดประสงค์นี้นำไปสู้การศึกษาการทำงานของอุปกรณ์และระบบการทำงานของเครื่องปรับอากาศ ศึกษาอุปกรณ์ที่จะใช้วัดตรวจสอบการทำงานของเครื่องปรับอากาศในจุด

Thesis Title	Monitor & Analyze Air conditioner system
Authors	Mr. Rallakoon Aksornklang
	Mr .Alongkorn Poosiripattananon
	Mr. Supakron Khungpo
Thesis Advisor	Asst.Prof. Chuae Nokyoo
Year	2015

Abstract

The project was created to reduce duration of Inspection the operation of the air conditioning system. Because of the air conditioning system cannot know the status of work at the moment that according to standard. Which to know late. May cause damage to the air conditioning system. Make cost and time than necessary. This project will build monitoring the air conditioner system for show the system status, presented by Real-time such at this moment electrical current have lower or higher usage and monitoring the condensing unit. This element could monitoring value and standard value to operator can see also and operating the system, make it saving cost and time.

กิตติกรรมประกาศ

ปริญญานิพนธ์ฉบับนี้สำเร็จลุล่วงได้ด้วยดีด้วยความอนุเคราะห์ของ บุคคลหลายท่าน ซึ่งไม่อาจจะนำมากล่าวได้ทั้งหมด ซึ่งผู้มีพระคุณท่านแรกที่ผู้ศึกษาใคร่ขอกราบพระคุณคือ ท่านอาจารย์ เชื้อ นกอยู่ อาจารย์ผู้สอนที่ได้ให้ความรู้ คำแนะนำตรวจทาน และแก้ไขข้อบกพร่องต่าง ๆ ด้วยความเอาใจใส่ทุกขั้นตอน เพื่อให้การเขียนรายงานค้นคว้าอย่างอิสระฉบับนี้สมบูรณ์ที่สุด เพื่อนๆพี่ๆน้องๆ ชาววิศวกรรมการวัดและควบคุม สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง ที่ช่วยในการสืบค้นข้อมูลแลกเปลี่ยนความรู้ความคิดและให้กำลังใจในการศึกษาค้นคว้าตลอดมา

คณะผู้จัดทำ

สารบัญ

หน้า

บทคัดย่อภาษาไทย I
บทคัดย่อภาษาอังกฤษ II
กิตติกรรมประกาศ III
สารบัญ IV
สารบัญตาราง VI
สารบัญภาพ VII

บทที่ 1 บทนำ 1
1.1 ความสำคัญของปริญญานิพนธ์ 1
1.2 วัตถุประสงค์ของปริญญานิพนธ์ 2
1.3 ขอบเขตของปริญญานิพนธ์ 2
1.4 ขั้นตอนการศึกษา 2
1.5 ประโยชน์ที่คาดว่าจะได้รับ 3

บทที่ 2 ทฤษฎี 4
2.1 เครื่องปรับอากาศ 4
2.1.1 ส่วนประกอบของเครื่องปรับอากาศ 4
2.1.2 ข้อมูลจำเพาะของเครื่องปรับอากาศ (specification) 6
2.2 Arduino Uno R3 7
2.2.1 คุณสมบัติของ Arduino Uno R3 8
2.3 เซ็นเซอร์วัดอุณหภูมิ DS18B20 9
2.3.1 คุณสมบัติของ DS18B20 10
2.4 Split-core Current Transformer SCT–013-030 12
2.4.1 คุณสมบัติของ SCT-013-030 12
2.5 LCD Character Display 14
2.4.1 โครงสร้างของ LCD 14

บทที่ 3 การออกแบบและการทำงานของโปรแกรม 16
3.1 บล็อกไดอะแกรมของระบบแสดงผล การทำงานเครื่องปรับอากาศ 16
3.1.1 โครงสร้างของระบบ 16

สารบัญ (ต่อ)

หน้า

3.1.2 การควบคุมการแสดงผลของ LCD....17
3.1.3 การเชื่อมต่อสัญญาณขาข้อมูลระหว่าง Microcontroller กับ LCD Controller....19
3.1.4 รายละเอียดคำสั่งในการสั่งงานระหว่าง Arduino กับ จอ LCD....20
3.2 วงจรวัดอุณหภูมิไอซีวัดอุณหภูมิ DS18B20....21
3.2.1 ข้อมูลเชิงเทคนิคเกี่ยวกับไอซี DS18B20....21
3.3 วงจรวัดกระแสไฟฟ้า....24

บทที่ 4 การทดลองและบันทึกผล....26
4.1 การทดลองวัดค่ากระแสไฟฟ้าขณะที่เครื่องปรับอากาศทำงาน....26
4.1.1 วัตถุประสงค์....26
4.1.2 วิธีการทดลอง....26
4.1.3 ผลการทดลอง....28
4.2 การทดลองวัดค่าอุณหภูมิ....29
4.2.1 วัตถุประสงค์....29
4.2.2 วิธีการทดลอง....29
4.2.3 ผลการทดลอง....31
4.3 การติดตั้งอุปกรณตรวจวัด....31
4.3.1 เซ็นเซอร์วัดกระแสไฟฟ้า....31
4.3.2 เซ็นเซอร์วัดอุณหภูมิ....32
บทที่ 5 สรุปผล....34
5.2 สรุปผลงานวิจัย....35
5.3 ข้อเสนอแนะ....35

ภาคผนวก....36

สารบัญตาราง

ตารางที่ **หน้าที่**

3.1 ตารางแสดง Pinout ของจอ LCD....17
3.2 แสดงตารางขาที่ใช้ในการเชื่อมต่อกับ Arduino UNO R3....19
4.1 แสดงการสอบเทียบเซ็นเซอร์วัดกระแสไฟฟ้า....28
4.2 แสดงการสอบเทียบเซ็นเซอร์วัดกระแสไฟฟ้า....28
4.3 แสดงการสอบเทียบเซ็นเซอร์วัดอุณหภูมิ....31
5.1 Error Code....34

สารบัญรูป

รูปที่	หน้าที่
2.1 แสดงส่วนประกอบของเครื่องปรับอากาศ	4
2.2 รูปแบบทั่วไปของวงจรการถ่ายเทความร้อนในระบบทำความเย็น	5
2.3 ข้อมูลจำเพาะของเครื่องปรับอากาศบริษัท Mitsubishi electric	6
2.4 แสดงค่ามาตรฐานของเครื่องปรับอากาศที่ใช้น้ำหล่อเย็นประเภท R-22	7
2.5 แสดงลักษณะของบอร์ด Arduino Uno R3	7
2.6 บล๊อกไดอะแกรมของบอร์ด Arduino Uno R3	8
2.7 Pinout Diagram ของบอร์ด Arduino Uno R3	9
2.8 แสดงลักษณะของเซ็นเซอร์วัดอุณภูมิ DS18B20	9
2.9 แสดงการใช้งานของเซ็นเซอร์วัดอุณหภูมิ DS18B20	10
2.10 แสดงบล๊อกไดอะแกรมของเซ็นเซอร์วัดอุณภูมิ DS18B20	10
2.11 แสดงโครงสร้างภายในของ DS18B20	11
2.12 โครงสร้างภายในรีจิสเตอร์ Temperature LSB และ MSB	11
2.13 แสดงลักษณะของ Split-core Current Transformer SCT-013-030	12
2.14 บล๊อกไดอะแกรมของ split-core current transformer SCT -013-030	13
2.15 แสดงจอ LCD แบบ Parallel 16 ขา	14
2.16 โครงสร้างทั่วไปของ LCD	15
3.1 จอ LCD แบบ Parallel 16 ขา	16
3.2 ขาที่ใช้ในการเชื่อมต่อกับ Arduino UNO R3	19
3.3 แสดงโปรแกรมแสดงผล LCD	20
3.4 แสดงผลเซ็นเซอร์วัดอุณหภูมิ DS18B20	22
3.5 แสดงการต่อวงของเซ็นเซอร์วัดอุณหภูมิ	23
3.6 แสดงรายละเอียดของ SCT-013-030	24
3.7 แสดงวงจรของ SCT-013-030	25
3.8 แสดงโปรแกรมการวัดกระแสไฟฟ้าของ SCT-013-030	25
4.1 แสดงการสอบเทียบเซ็นเซอร์วัดกระแสไฟฟ้า	27
4.2 แสดงการสอบเทียบเซ็นเซอร์วัดกระแสไฟฟ้า	27
4.3 ผลการวัดค่ากระแสไฟฟ้า	29
4.4 แสดงการสอบเทียบเซ็นเซอร์วัดอุณหภูมิ	30
4.5 แสดงการสอบเทียบเซ็นเซอร์วัดอุณหภูมิ	30

สารบัญรูป (ต่อ)

หน้าที่

4.6 แสดงการการติดตั้งเซ็นเซอร์วัดกระแสไฟฟ้า.....32
4.7 แสดงการติดตั้งเซ็นเซอร์วัดอุณหภูมิตัวที 1.....32
4.8 แสดงการติดตั้งเซ็นเซอร์วัดอุณหภูมิตัวที 2.....33
4.9 แสดงการติดตั้งเซ็นเซอร์วัดอุณหภูมิตัวที 3.....33

บทที่1

บทนำ

1.1 ความสำคัญของปริญญานิพนธ์

ในปัจจุบันเครื่องปรับอากาศกลายเป็นอุปกรณ์อำนวยความสะดวกที่เป็นที่ต้องการของมนุษย์และมีการเพิ่มปริมาณการใช้เครื่องปรับอากาศขึ้นอย่างมากตั้งแต่อดีตถึงปัจจุบันสัดส่วนการใช้เครื่องปรับอากาศในหลายๆสถานที่เพิ่มมากขึ้น เช่น บ้านเรือนที่อยู่อาศัย อาคารสำนักงานตามโรงเรียน, มหาวิทยาลัย หน่วยงานของรัฐและเอกชน ห้างสรรพสินค้า และอื่นๆอีกมากมาย

การใช้งานเครื่องปรับอากาศในการใช้งานปกติ เมื่อมีการใช้งานไปอย่างต่อเนื่อง ก็จะมีการเสื่อมสภาพของอุปกรณ์เครื่องปรับอากาศ ทำให้เครื่องปรับอากาศทำงานได้ไม่เต็มประสิทธิภาพหรือผิดพลาด ซึ่งในปกติแล้วเมื่อมีการทำงานที่ผิดพลาดหรืออุปกรณ์เกิดเสียขึ้นมา เครื่องปรับอากาศบางตัวอาจจะทำงานได้ไม่เต็มประสิทธิภาพ แต่ก็ยังไม่ถึงขั้นเสียหาย เราไม่สามารถรู้สถานภาพได้ว่าเครื่องปรับอากาศทำงานได้เต็มประสิทธิภาพหรือไม่ อาจจะทำให้เกิดการละเลยในการตรวจสอบได้ ซึ่งเมื่อเกิดการเสียหายต้องมีการซ่อมแซม ต้องมีการตรวจเช็คสภาพว่ามีการผิดพลาดที่จุดใด ต้องมีการประเมินความเสียหาย ซึ่งอาจจะทำให้ต้องใช้เวลาพอสมควรในการจะตรวจสอบเมื่อยู่ในอาคารหรือตึกสำหนักงานที่มีการติดตั้งเครื่องปรับอากาศจำนวนมากๆ

ด้วยเหตุนี้จึงนำมาสู่การสร้างโครงการนี้ขึ้น เพื่อสร้างอุปกรณ์ที่สามารถแสดงผลสถานภาพการทำงานของเครื่องปรับอากาศ ณ เวลาขณะนั้นได้ เพื่อที่จะได้ทราบการทำงานของเครื่องปรับอากาศว่ามีการทำงานเป็นไปตามค่ามาตรฐานอุปกรณ์ตามแต่ละยี่ห้อ,บริษัทกำหนดไว้หรือเมื่อ เมื่อเครื่องปรับอากาศทำงานได้ไม่เต็มประสิทธิภาพหรือมีข้อผิดพลาดที่ไม่เป็นไปตามค่ามาตรฐานที่ตั้งไว้ก็สามารถที่จะระบุได้ว่าเครื่องปรับอากาศตัวใดเกิดการทำงานที่ผิดพลาด และสามารบอกได้ในขั้นต้นว่าความเสียหายเกิดขึ้นที่จุดใด ซึ่งสามรถลดขั้นตอนการตรวจสอบและตรวจหาอุปกรณ์ที่เสียหายได้ ซึ่งจุดประสงค์นี้นำไปสู้การศึกษาการทำงานของอุปกรณ์และระบบการทำงานของเครื่องปรับอากาศ ศึกษาอุปกรณ์ที่จะใช้วัดตรวจสอบการทำงานของเครื่องปรับอากาศในจุดต่างๆ ศึกษาการสร้างอุปกรณ์ควบคุมการทำงานของอุปกรณ์ที่จะใช้เพื่อนำองค์ความรู้นี้ไปต่อยอดเพื่อทำให้เป็นไปตามวัตถุประสงค์

1.2.วัตถุประสงค์ของปริญญานิพนธ์

1. ทราบข้อมูลและการทำงานของระบบปรับอากาศ

2. เลือกอุปกรณ์ที่จะมาใช้วัดค่าเพื่อแสดงผลการทำงานของระบบเครื่องปรับอากาศได้

3. ทราบข้อมูลและรายละเอียดของอุปกรณ์ที่จะนำมาใช้ได้

4.นำความรู้และข้อมูลที่มีไปพัฒนาต่อยอดและสร้างเป็นอุปกรณ์แสดงสถานภาพของระบบเครื่องปรับอากาศที่สมบูรณ์ได้

1.3.ขอบเขตของปริญญานิพนธ์

1. ข้อมูลการทำงานของระบบปรับอากาศ

2. ข้อมูลของอุปกรณ์ Arduino

3. ข้อมูลของอุปกรณ์การวัด Temperature และ Current

4. สร้างอุปกรณ์วัดและแสดงสถานภาพของระบบเครื่องปรับอากาศ

1.4.ขั้นตอนการศึกษา

1.ศึกษาการทำงานของระบบปรับอากาศและอุปกรณ์ภายในเพื่อที่จะได้ข้อมูลและนำไปใช้ในการติดตั้งอุปกรณ์

2.ศึกษาและหาข้อมูลของบอร์ด Arduino ที่จะนำมาใช้ในการสร้างชุดแสดงสถานภาพของระบบเครื่องปรับอากาศ

3.ศึกษาและหาข้อมูลของอุปกณ์ที่ใช้ในการวัด ในที่นี้มี Temperature Sensor และ Current Sensor

4.นำความรู้มาสร้างอุปกรณ์ชุดแสดงสถานภาพของระบบเครื่องปรับอากาศ

5 สรุปผลของการทำงานและสรุปปัญหาของการทำงาน

1.5 ประโยชน์ที่คาดว่าจะได้รับ

1. ได้ทราบและเข้าใจถึงการทำงานของระบบเครื่องปรับอากาศ

2. ได้ทราบและเข้าใจการทำงานและใช้งานบอร์ด Arduino

3. ได้ทราบและเข้าใจถึงการใช้งานอุปกรณ์ที่จะใช้ในโครงงานนี้ทั้ง Temperature Sensor และ Current Sensor

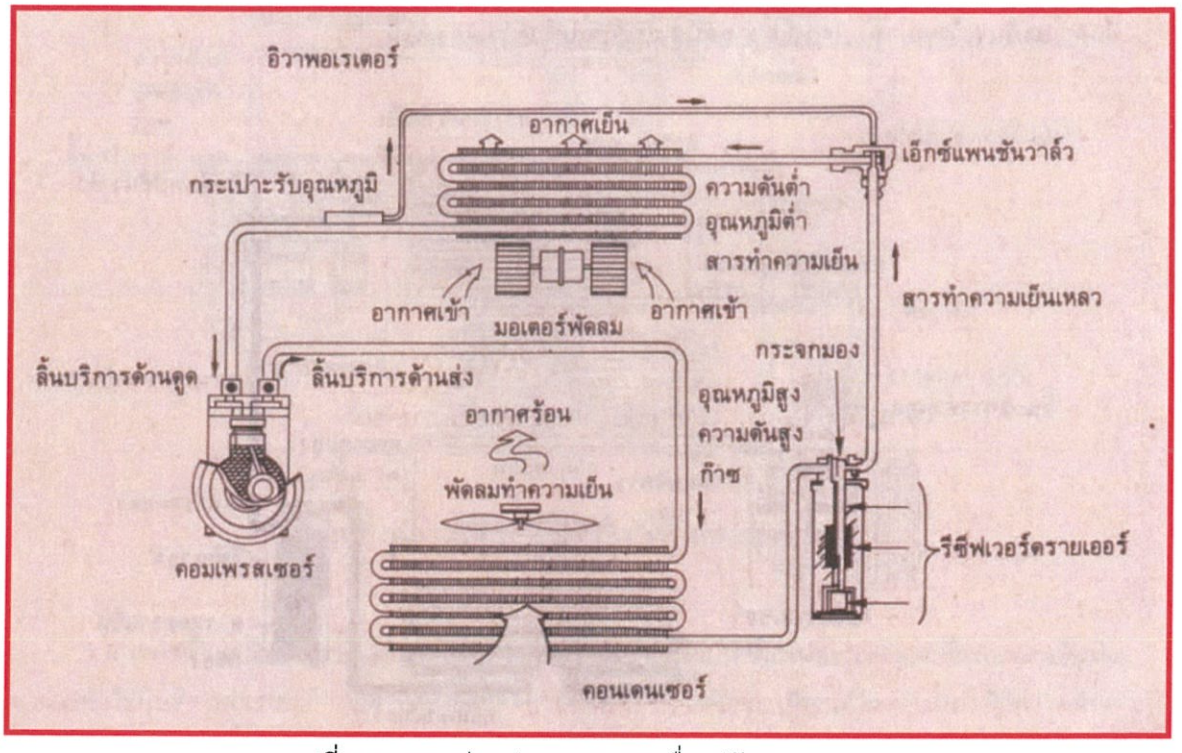
4. สามารถวิเคราะห์ปัญหาที่เกิดขึ้นกับเครื่องปรับอากาศได้

5. สามมรถสร้างอุปกรณ์ชุดแสดงสถานภาพการทำงานของระบบเครื่องปรับอากาศได้

บทที่ 2

ทฤษฎี

2.1. เครื่องปรับอากาศ



รูปที่ 2.1 แสดงส่วนประกอบของเครื่องปรับอากาศ

2.1.1 ส่วนประกอบของเครื่องปรับอากาศ

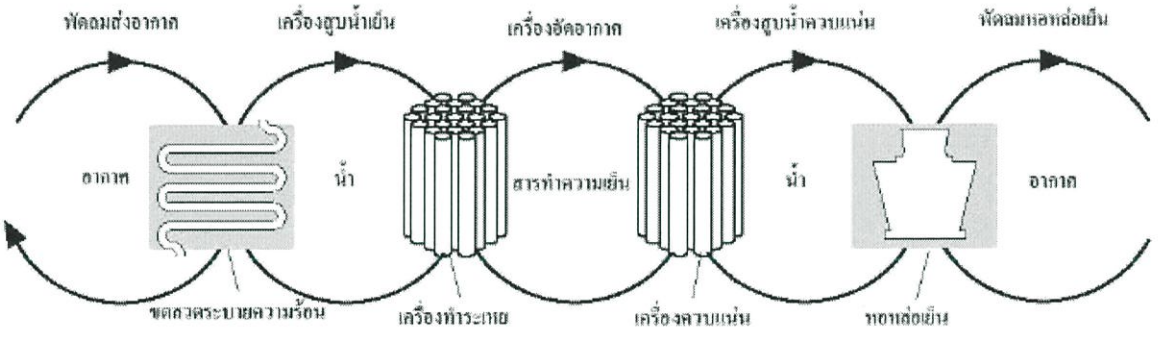
1.COMPRESSOR คือเครื่องอัดไอ การทำงานหรือหน้าที่ของมันคือ ดูดไอ(แรงดันต่ำ) ซึ่งเกิดจากการระเหยภายในคอยล์เย็น ทำการอัดให้เป็นไอ(แรงดันสูง) อุณหภูมิสูง เพื่อส่งไประบายความร้อนต่อไป

2.CONDENSER คือเครื่องควบแน่น หรือ คอยล์ร้อน หน้าที่ของมันคือรับไอร้อนที่ถูก COMPRESSOR อัดจนร้อนและมีอุณหภูมิสูง เข้ามาในแผงพื้นที่ของมัน จากไอที่มีอุณหภูมิสูง เมื่อมาเจอ

กับอากาศภายในห้อง ซึ่งมีอุณหภูมิต่ำกว่า ความร้อนจึงถูกถ่ายเทออกไปได้โดยไอร้อนนั้น จะควบแน่นกลายเป็นของเหลว(แรงดันสูง-อุณหภูมิสูง)แต่มีมอเตอร์พัดลมเป็นตัวช่วยระบายความร้อนออกไปให้เร็วขึ้น เมื่อเป็นของเหลวแล้วก็สามารถกลับมารับความร้อนภายในห้องได้อีก แต่ของเหลวนั้นยังมีอุณหภูมิสูงอยู่ จึงต้องทำให้อุณหภูมินั้นลดลงก่อน

3.CAPILLARY TUBE คือท่อลดแรงดันหรือท่อรูเข็ม หน้าที่ของมันคือลดแรงดันของน้ำยาแอร์(ของเหลว)จากที่ถูกระบายความร้อนแล้ว ยังมีอุณหภูมิสูง-แรงดันสูง เมื่อมาเจอท่อรูเข็ม ทำให้ของเหลวอั้น ผ่านได้น้อย ทำให้ของเหลวนั้น มีอุณหภูมิลดลง และแรงดันลดลง น้ำยาแอร์(ของเหลว)และไหลพอดีเหมาะสมกับพื้นที่ของคอล์ยเย็น เพื่อที่จะมารับความร้อน ในห้องได้อีกครั้ง

4.EVAPPORATOR คือเครื่องระเหยหรือคอล์ยเย็น การทำงานของมันคือ ดูดความร้อนจากภายในห้อง โดยมีมอเตอร์พัดลมเป็นตัวดูดเข้ามา ผ่านช่องที่เรียกว่า RetumAirซึ่งมี Filter เป็นตัวกรองฝุ่นให้ก่อน แล้วความร้อนที่ถูกดูดเข้ามานั้น จะมาสัมผัสกับคอล์ยเย็นซึ่งมีน้ำยาแอร์(ของเหลว) ซึ่งอุณหภูมิติดลบ วิ่งอยู่ในท่อนั้น จะเกิดการระเหยเป็นไอ(แรงดันต่ำ)



รูปที่ 2.2 รูปแบบทั่วไปของวงจรการถ่ายเทความร้อนในระบบทำความเย็น

ในระบบทำความเย็นจะมีวงจรการถ่ายเทความร้อนอยู่หลายรูปแบบ ดังแสดงในรูปที่ 2 พลังงานความร้อนจะเคลื่อนที่จากซ้ายไปขวาในขณะที่มันถูกดึงออกมาจากช่องว่างและถูกขับดันไปสู่ภายนอกโดยผ่านวงจรการถ่ายเทความร้อน 5 รูปแบบ

1) วงจรอากาศภายใน ซึ่งในวงจรทางซ้ายมืออากาศภายในจะถูกขับเคลื่อนโดยพัดลมอากาศผ่านมาทางขดลวดทำความเย็นซึ่งเป็นที่ที่มีการถ่ายเทความร้อนไปสู่น้ำเย็น อากาศที่เย็นแล้วจึงไปทำให้พื้นที่ภายในอาคารเย็นลง

2) วงจรน้ำเย็น ซึ่งจะถูกขับเคลื่อนโดยเครื่องสูบน้ำน้ำที่ไหลกลับมาจากขดลวดทำความเย็นไปยังเครื่องทำระเหยเพื่อถูกทำให้เย็นอีกครั้ง

3) วงจรสารทำความเย็น ซึ่งจะใช้สารทำความเย็นแบบเปลี่ยนเฟส เครื่องอัดอากาศของเครื่องทำความเย็นจะสูบความร้อนจากน้ำเย็นไปยังน้ำควบแน่น

4) วงจรน้ำควบแน่น ซึ่งน้ำจะดูดความร้อนจากเครื่องควบแน่นของเครื่องทำความเย็นและเครื่องสูบน้ำควบแน่นจะส่งมันไปที่หอหล่อเย็น

5) วงจรหอหล่อเย็น ซึ่งพัดลมของหอหล่อเย็นจะขับเคลื่อนอากาศผ่านช่องการไหลแบบเปิดของน้ำควบแน่นร้อน แล้วถ่ายเทความร้อนไปสู่ภายนอก

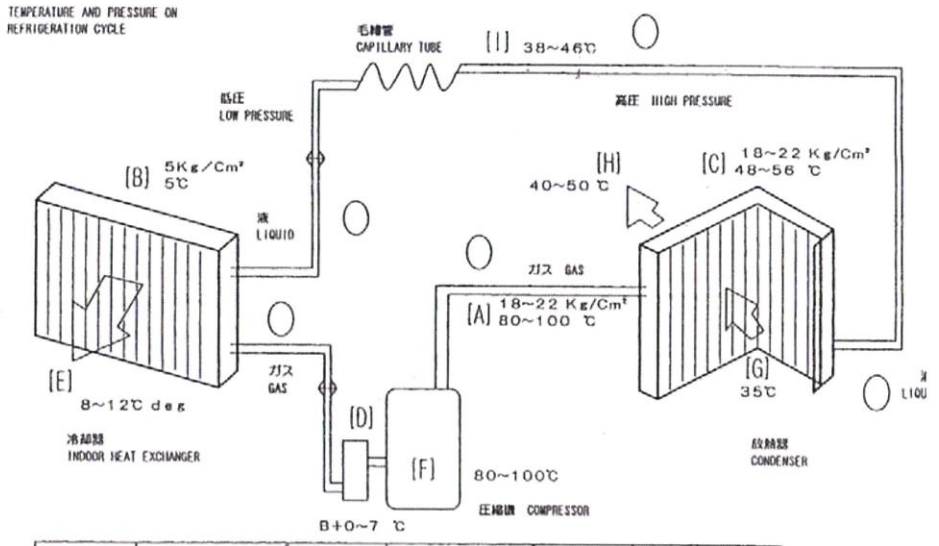
2.1.2 ข้อมูลจำเพาะของเครื่องปรับอากาศ (specification)

เครื่องปรับอากาศของแต่ละตัวจะมีคุณสมบัติและข้อมูลจำเพาะซึ่งเป็นการแสดงรายละเอียดสถานะการทำงานว่ามีคุณสมบัติอย่างไร

PC-KAKLT-5, PC-KAKLT เครื่องปรับอากาศชนิดฝังฝ้าเพดาน CEILING SUSPENDED

รุ่น Models				PC-2.5KAKLT-5	PC-3KAKLT-5	PC-3KAKLT	PC-4KAKLT	PC-5KAKLT	PC-6KAKLT
EGAT Level 5						-	-	-	-
ความสามารถในการทำความเย็น Cooling capacity	50Hz		W	7,200	8,800	8,800	10,600	12,300	14,100
			BTU/Hr	23,936.2	29,922.9	30,000	36,000	42,000	48,000
กำลังไฟฟ้าที่ใช้ Total input			kW	2.05	2.5	2.95	3.47/3.32	4.2	5.22
อัตราส่วนประสิทธิภาพของเครื่องปรับอากาศ EER			BTU/Hr/W	12.03	12.01	10.2	10.4/10.8	10	9.2
ชุดภายในอาคาร Indoor unit	รุ่น Model name			PC-2.5KAKLT-5	PC-3KAKLT-5	PC-3KAKLT	PC-4KAKLT	PC-5KAKLT	PC-6KAKLT
	กระแสไฟฟ้า Power Supply	50Hz		1ph 220V					
	วัสดุในการทำตัวเครื่อง External finish			Munsell 6.4Y 8.9/0.4					
	กำลังขับมอเตอร์ Fan motor output		kW	0.16		0.095	0.16		
	ระดับความเร็วลมเป่า (ต่ำ-สูง) Airflow (low-high)	50Hz	m³/min	24 - 30	25 - 32	16 - 22	24 - 30	25 - 32	27 - 34
			CFM	850 - 1,060	885 - 1,130	565 - 777	847 - 1,059	883 - 1,130	953 - 1,200
	แรงดันลม External static pressure		Pa (mmAq)	0 (direct blow)					
	การควบคุมการทำงาน & ตัวควบคุมอุณหภูมิ Operation control & thermostat			Remote control & Built-in					
	ระดับเสียง (ต่ำ-สูง)* Sound level (low-high)*	50Hz	dB(A)	39 - 45	40 - 46	34 - 42	39 - 45	40 - 46	42 - 48
	ขนาดท่อน้ำทิ้ง (วัดด้านนอก) Unit drain pipe O.D.		mm	26					
	ขนาดตัวเครื่อง Dimensions	กว้าง W	mm	1,600		1,280	1,600		
		ลึก D	mm	680					
		สูง H	mm	230					
	น้ำหนัก Weight		kg	36	38	32	36	38	39
ชุดภายนอกอาคาร Outdoor unit	รุ่น Model name			PU-2.5VAKD-5	PU-3VAKD-5	PU-3VAKD	PU-4V/YAKD	PU-5YAKD	PU-6YAKD
	กระแสไฟฟ้า Power Supply	50Hz		1ph 220V			1ph 220V/3ph 380V	3ph 380V	
	วัสดุในการทำตัวเครื่อง External finish			Munsell 3.0Y 7.8/1.1					
	วาล์วควบคุมสารทำความเย็น (R22) Refrigerant (R22) control			Capillary Tube					
	ขนาดมอเตอร์คอมเพรสเซอร์ Compressor output	50Hz	kW	1.6	2.0	2.5	2.7	3.5	4.2
	อุปกรณ์ป้องกัน Protection devices			Direct cut		Inner protector (Compressor)	(V): Inner protector (Compressor) (Y): Inner protector (Compressor) Thermal relay, Anti-phase protector	Thermal switch, HP switch, LP switch, Anti-phase protector, Thermal relay	
	กำลังขับมอเตอร์ Fan motor output		kW	0.075	0.065 + 0.065			0.10 + 0.10	
	ระดับความเร็วลมเป่า Airflow	50Hz	CFM	1,800	3,350	3,354		3,530	
	ระดับเสียง Sound level	50Hz	dB(A)	56	54			55	56
	ขนาดตัวเครื่อง Dimensions	กว้าง W	mm	840	870			970	
		ลึก D	mm	330	295 + 24			345 + 24	
		สูง H	mm	880	1,258				
	น้ำหนัก Weight		kg	65	84	85		111	112
	ระดับความต่างสูงสุด MAX.height difference		m	Max. 5m	Max. 20m		Max. 30 m	Max. 50m	
	ระดับความยาวท่อสูงสุด MAX.piping length		m	Max. 15m	Max. 30m		Max. 40m	Max. 50m	
	ขนาดท่อน้ำยา Outer Diameter	Liquid	Ø mm(inch)	9.52 (3/8)					
		Gas	Ø mm(inch)	15.88 (5/8)			19.05 (3/4)		
	ไล่ท่อน้ำยาสำหรับความยาวท่อไม่เกิน		m	7			20		

รูปที่ 2.3 ข้อมูลจำเพาะของเครื่องปรับอากาศบริษัท Mitsubishi electric

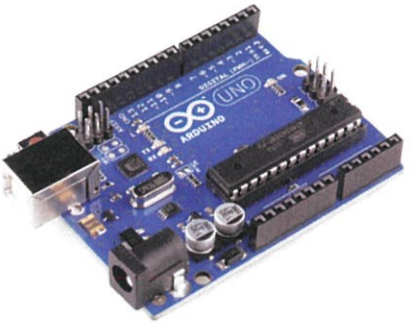


A 20 A 90	18~22 Kg/Cm² 80~100 ℃	B 5 B 5	5~6.5 Kg/Cm² 5~12 ℃	C 20 C 50	18~22 Kg/Cm² 48~56 ℃
SH=D-B D 5	0~7 ℃ deg B+0~7 ℃	E 10	8~12 ℃ deg	F 90	80~100℃
G 35	35℃	H 45	40~50 ℃	I 40	38~46℃

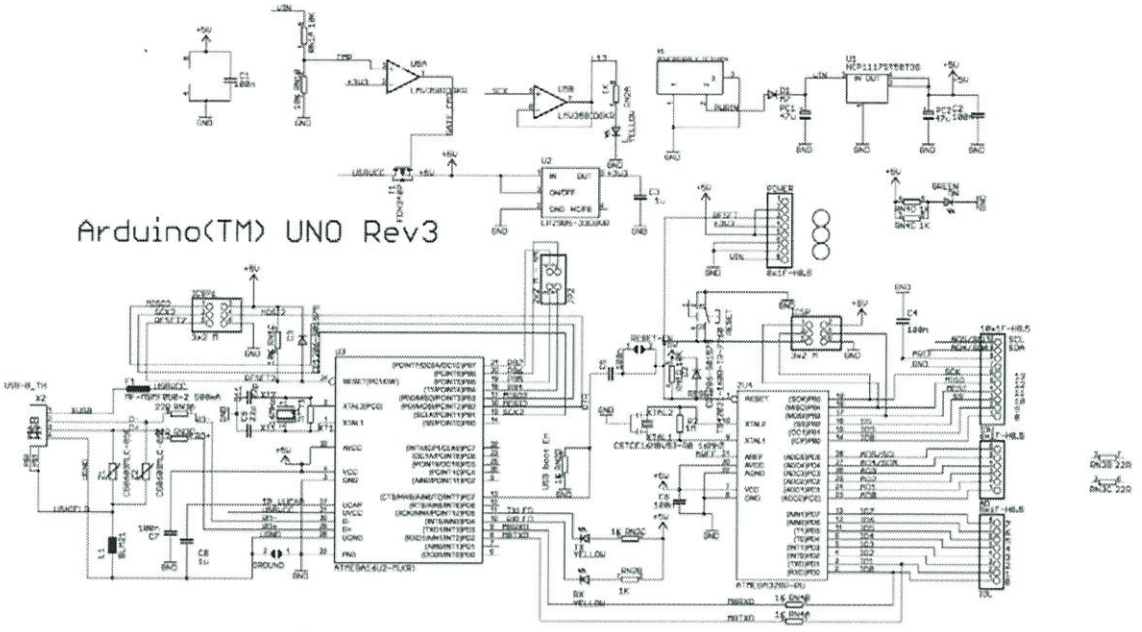
รูปที่ 2.4 แสดงค่ามาตรฐานของเครื่องปรับอากาศที่ใช้น้ำหล่อเย็นประเภท R-22

2.2 Arduino Uno R3

Arduino Uno R3 เป็น Microcontroller board ที่ใช้ ATmega328 เป็น MCU หลัก ซึ่งตัวนี้จะมีขา Digital 14 ขา อินพุท/เอ้าพุท (สามารถทำเป็น PWM ได้ถึง 6 ขา) และมีขา Analog อินพุทได้อีก 6 ขา, รันที่ความถี่ 16 MHz มี USB Connector และ Power Jack DC ซึ่ง Concept ของ Arduino Board นี้ทำมาเพื่อความสะดวก ง่ายในการเชื่อมต่อเข้ากับคอมพิวเตอร์ สามารถต่อ USB เข้ากับช่องคอมพอร์ต ก็สามารถ Run โปรแกรมที่ Board ได้



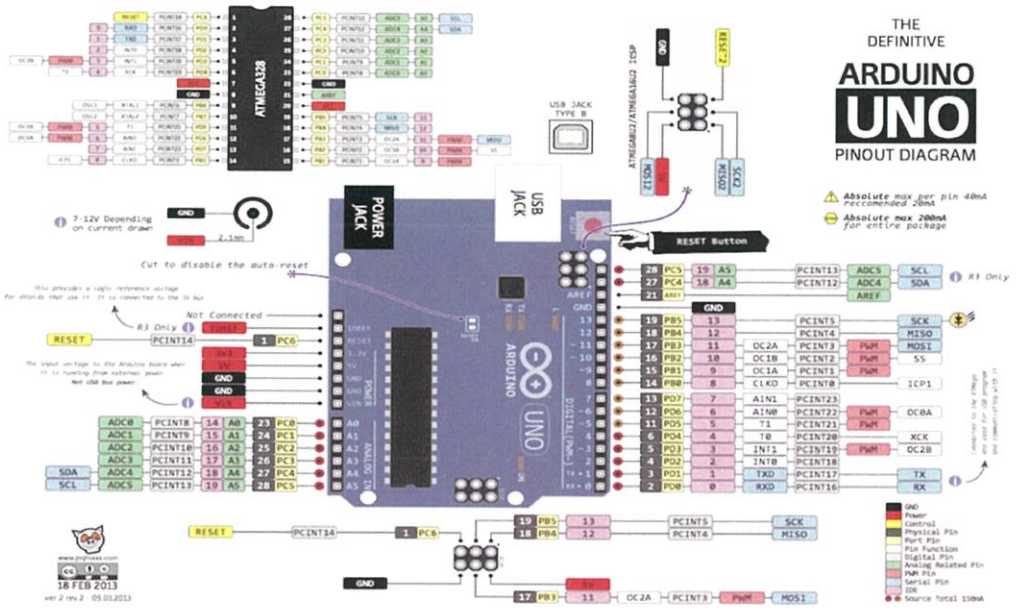
รูปที่ 2.5 แสดงลักษณะของบอร์ด Arduino Uno R3



รูปที่ 2.6 บล๊อกไดอะแกรมของบอร์ด Arduino Uno R3

2.2.1 คุณสมบัติของ Arduino Uno R3

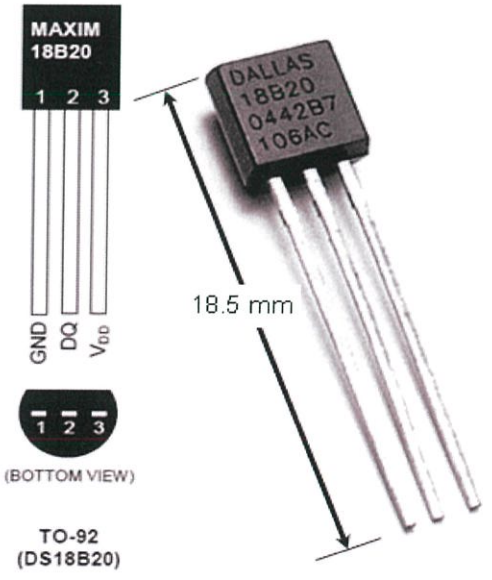
- Microcontroller	ATmega328
- Operating Voltage	5V
- Input Voltage (recommended)	7-12V
- Input Voltage (limits)	6-20V
- Digital I/O Pins	14 (of which 6 provide PWM output)
- Analog Input Pins	6
- DC Current per I/O Pin	40 mA
- DC Current for 3.3V Pin	50 mA
- Flash Memory	32 KB (ATmega328)
- SRAM	2 KB (ATmega328)
- EEPROM	1 KB (ATmega328)
- Clock Speed	16 MHz



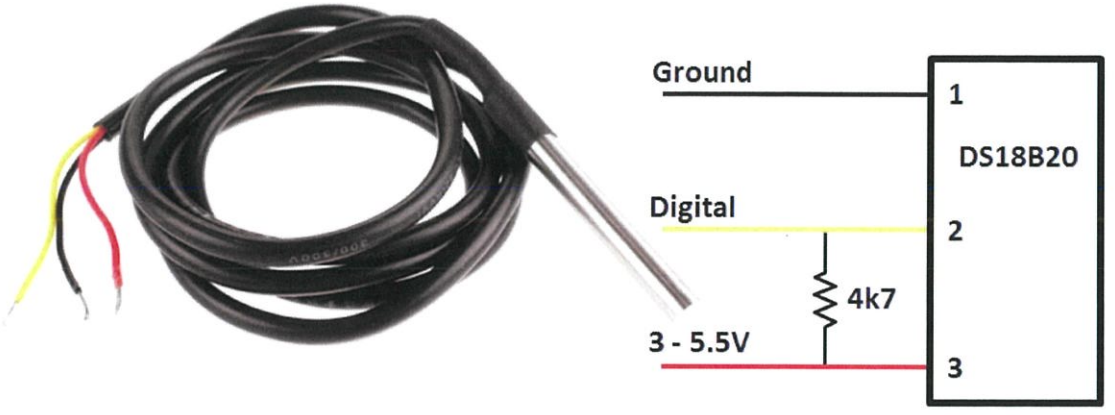
รูปที่ 2.7 Pinout Diagram ของบอร์ด Arduino Uno R3

2.3. เซ็นเซอร์วัดอุณหภูมิ (DS18B20)

Digital Thermometer เป็นไอซี Digital temperature sensors ของค่าย Dallas สามารถที่จะวัดค่าอุณหภูมิแล้วถอดรหัสออกมาเป็นดิจิตอลซึ่งในโครงงานนี้เรามาอ่านค่าอุณหภูมิของเครื่องปรับอากาศแล้วนำค่าที่อ่านนี้ไปประมวลผลเพื่อให้ไมโครคอนโทรลเลอร์เป็นตัวตัดสินใจในกาสั่งงาน



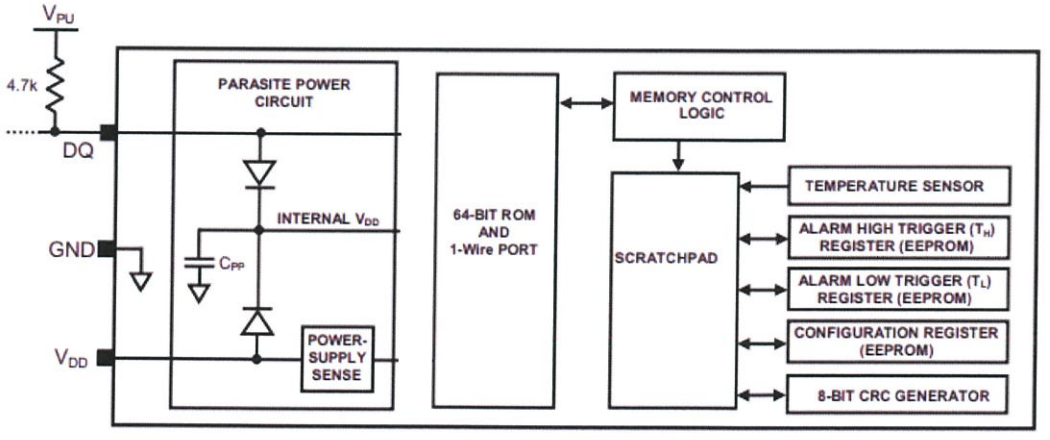
รูปที่ 2.8 แสดงลักษณะของเซ็นเซอร์วัดอุณภูมิ DS18B20



รูปที่ 2.9 แสดงการใช้งานของเซ็นเซอร์วัดอุณหภูมิ DS18B20

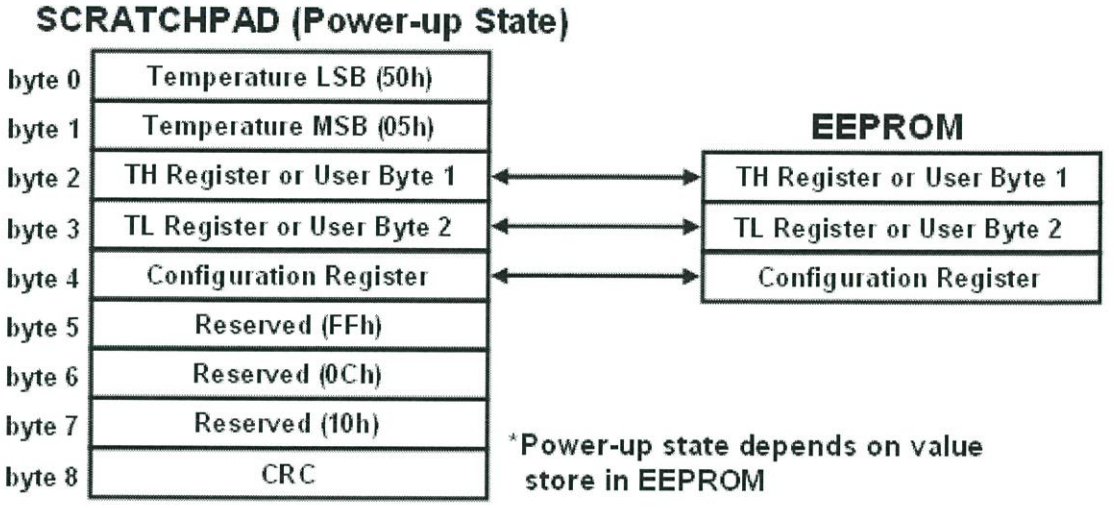
2.3.1 คุณสมบัติของDS18B20

- ใช้แรงดันไฟเลี้ยงในช่วง 3.0V ถึง 5.5V
- มี 3 ขา (ตามรูปที่ 1)
- ใช้งานได้สองแบบ: normal mode (ใช้ทั้ง 3 ขา) และ parasite power mode (ใช้เพียง 2 ขา คือ DQ และ GND ในขณะที่ขา Vddจะต่อกับขา Gnd)
- สามารถนำไอซีมาพ่วงต่อกันในบัสได้หลายอุปกรณ์
- วัดอุณหภูมิได้ในช่วง -55 °C ถึง +125 °C
- มีความแม่นยำ ±0.5 °C สำหรับอุณหภูมิในช่วง -10°C ถึง +85°C
- มีความละเอียดของค่าที่อ่านได้ 12 บิต
- ใช้เวลาในการแปลงข้อมูลสำหรับ ADC ไม่เกิน 750 msec สำหรับข้อมูล 12 บิต
- ในการใช้งานจะต้องต่อ pull-up 4.7kΩ ที่ขา DQ กับแรงดันไฟเลี้ยง



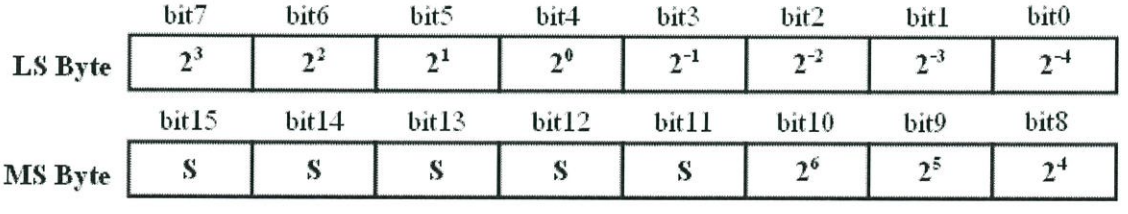
รูปที่ 2.10 แสดงบล๊อกไดอะแกรมของเซ็นเซอร์วัดอุณภูมิ DS18B20

โครงสร้างรีจิสเตอร์ภายในของ DS18B20 มีลักษณะดังแสดงในรูปที่ 2 จะเห็นได้ว่าประกอบไปด้วย SRAM Scratchpad ขนาด 9 ไบต์ และ EEPROM ขนาด 3 ไบต์ ซึ่งใช้เก็บค่าอุณหภูมิสูงสุด (TH) ต่ำสุด(TL)สำหรับเปรียบเทียบการเกิดสัญญาณเตือนและรีจิสเตอร์ควบคุม



รูปที่ 2.11 แสดงโครงสร้างภายในของ DS18B20

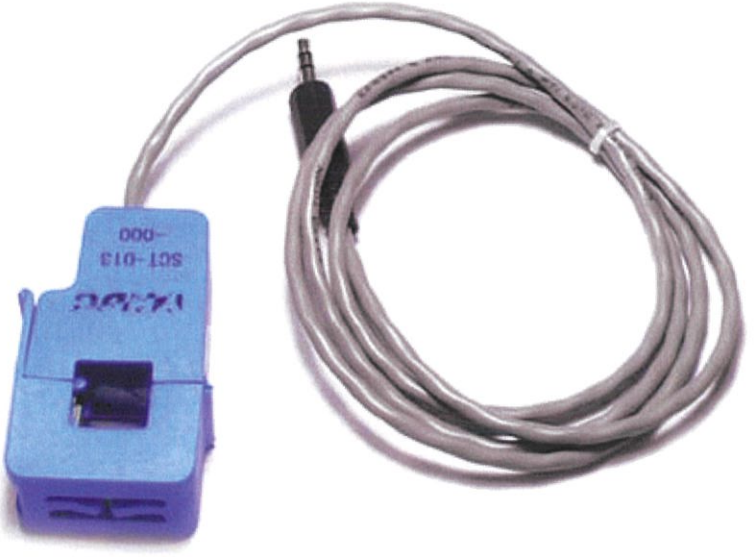
ข้อมูลอุณหภูมิที่วัดได้จะถูกเก็บอยู่ในรีจิสเตอร์ Temperature ซึ่งมีขนาด 16 บิต ดังแสดงในรูปที่ 3 ถ้าข้อมูลอุณหภูมิเป็นบวก S จะเป็น “1” แต่ถ้าข้อมูลอุณหภูมิเป็นลบ S จะเป็น “0” ในกรณีที่ DS18B20 ทำงานในโหมดความละเอียด 12 บิต บิตทุกบิตในรีจิสเตอร์ Temperature จะถูกใช้หมด แต่ในกรณีที่ทำงานในโหมด 9-11 บิต บิตล่าง (บิต 0 – บิต 2) จะไม่ถูกใช้งาน ซึ่งในการกำหนดโหมดความละเอียดการทำงานของ DS18B20 นั้นสามารถกำหนดได้ที่รีจิสเตอร์ Configuration ซึ่งโดยปกติเริ่มต้น DS18B20 จะทำงานในโหมด 12 บิต



รูปที่ 2.12 โครงสร้างภายในรีจิสเตอร์ Temperature LSB และ MSB

2.4 Split-core Current Transformer SCT–013-030

Non-invasive current sensor เป็นเซ็นเซอร์วัดกระแสสำหรับวัดไฟฟ้ากระแสสลับ (AC) แบบแคลมป์คล้องสาย สามารถวัดแรงดันไฟบ้านขนาด 220 V ได้ สามารถวัดกระแสได้สูงสุด 30 A ให้เอาต์พุตเป็นแรงดันผ่านคอนเน็ตเตอร์แจ็คแบบหูฟังขนาด 3.5 mm โดยให้ค่าแรงดันสูงสุดที่ 1 V เมื่อวัดกระแสได้ 30A ในการใช้งานกับไมโครคอนโทรลเลอร์ต้องต่อผ่านวงจรขยายแรงดันก่อนต่อเข้าอ่านค่าด้วย Analog Input สามารถใช้งานได้ในสภาพแวดล้อมที่มีอุณหภูมิ -25 ถึง 70 องศาเซลเซียส เหมาะกับการนำไปใช้ตรวจสอบป้องกันมอเตอร์กระแสสลับ อุปกรณ์ที่ใช้ในระบบส่องสว่างหรือเครื่องปรับอากาศ



รูปที่ 2.13 แสดงลักษณะของ Split-core Current Transformer SCT-013-030

2.4.1 คุณสมบัติของ SCT-013-030

- Core Material	Ferrite
- External Material	Plastic
- Input Current	0A – 30A
- Opening Size	13 X 13mm
- Output Plug	3.5mm
- Dielectric Strength	6000V AC / 1min

- Working Temperature -25℃ - 70℃

- Anti-Flaming rate UL 94 – V0

- Mechanical >500 (20℃)

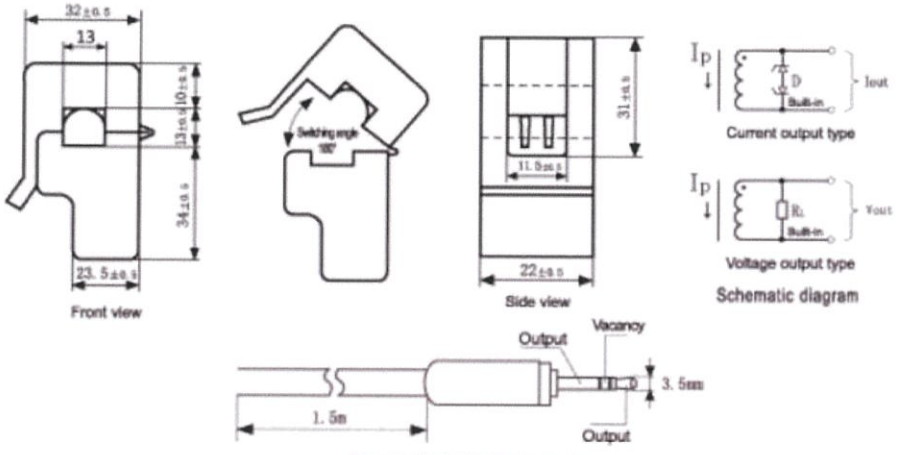


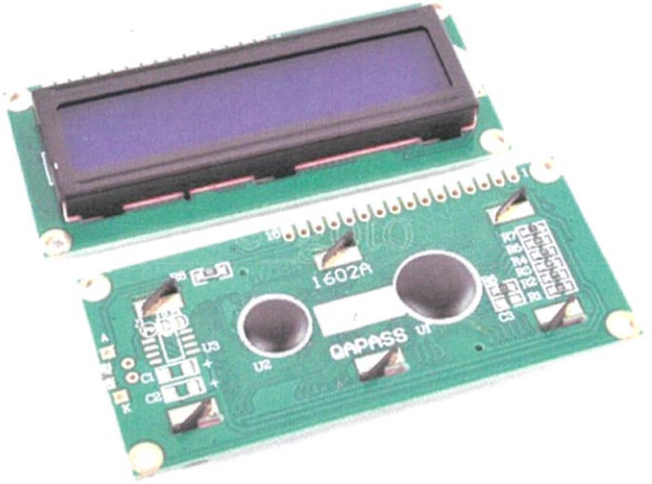
Table of technical parameters:

Model	SCT-013-000	SCT-013-005	SCT-013-010	SCT-013-015	SCT-013-020
Input current	0-100A	0-5A	0-10A	0-15A	0-20A
Output mode	Current/33mA	Voltage/1V	Voltage/1V	Voltage/1V	Voltage/1V
Model	SCT-013-025	SCT-013-030	SCT-013-050	SCT-013-060	SCT-013-070
Input current	0-25A	0-30A	0-50A	0-60A	
Output mode	Voltage/1V	Voltage/1V	Voltage/1V	Voltage/1V	

รูปที่ 2.14 บล๊อกไดอะแกรมของ split-core current transformer SCT -013-030

2.5 LCD Character Display

จอ Liquid Crystal Display (LCD) เป็นจอแสดงผลรูปแบบหนึ่งที่นิยมนำมาใช้งานกันกับระบบสมองกลฝังตัวอย่างแพร่หลาย จอ LCD มีทั้งแบบแสดงผลเป็นตัวอักขระเรียกว่า Character LCD ซึ่งมีการกำหนดตัวอักษรหรืออักขระที่สามารถแสดงผลไว้ได้อยู่แล้ว และแบบที่สามารถแสดงผลเป็นรูปภาพหรือสัญลักษณ์ได้ตามความต้องการของผู้ใช้งานเรียกว่า Graphic LCD

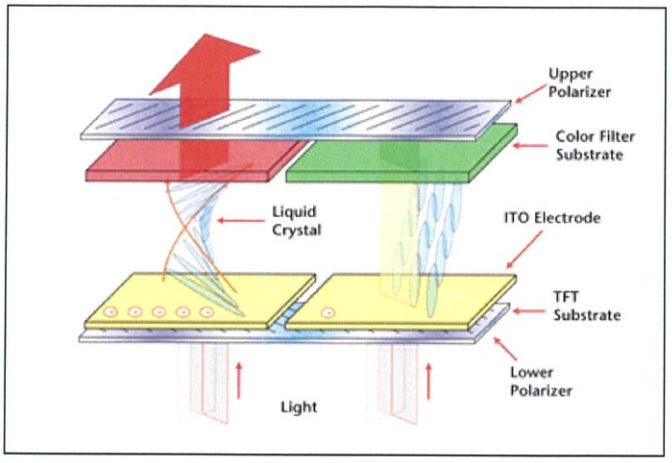


รูปที่ 2.15 แสดงจอ LCD แบบ Parallel 16 ขา

2.4.1 โครงสร้างของ LCD

โครงสร้างของ LCD ทั่วไปจะประกอบขึ้นด้วยแผ่นแก้ว 2 แผ่นประกบกันอยู่ โดยเว้นช่องว่างตรงกลางไว้ 6-10 ไมโครเมตร ผิวด้านในของแผ่นแก้วจะเคลือบด้วยตัวนำไฟฟ้าแบบใสเพื่อใช้แสดงตัวอักษร ตรงกลางระหว่างตัวนำไฟฟ้าแบบใสกับผลึกเหลวจะมีชั้นของสารที่ทำให้โมเลกุลของผลึกรวมตัวกันในทิศทางที่แสงส่องมากระทบเรียกว่า Alignment Layer และผลึกเหลวที่ใช้โดยทั่วไปจะเป็นแบบ Magnetic โดย LCD สามารถแสดงผลให้เรามองเห็นได้ทั้งหมด 3 แบบด้วยกันคือ

- แบบใช้การสะท้อนแสง (Reflective Mode) LCD แบบนี้ใช้สารประเภทโลหะเคลือบอยู่ที่แผ่นหลังของ LCD ซึ่ง LCD ประเภทนี้เหมาะกับการนำมาใช้งานในที่ที่มีแสงสว่างเพียงพอ
- แบบใช้การส่งผ่าน (Transitive Mode) LCD แบบนี้วางหลอดไฟไว้ด้านหลังจอ เพื่อทำให้การอ่านค่าที่แสดงผลทำได้ชัดเจน
- แบบส่งผ่านและสะท้อน (Transflective Mode) LCD แบบนี้เป็นการนำเอาข้อดีของจอแสดงผล LCD ทั้ง 2 แบบมารวมกัน



รูปที่ 2.16 โครงสร้างทั่วไปของ LCD

บทที่3

การออกแบบและการทำงานของโปรแกรม

3.1 บล็อกไดอะแกรมของระบบแสดงผล การทำงานเครื่องปรับอากาศ

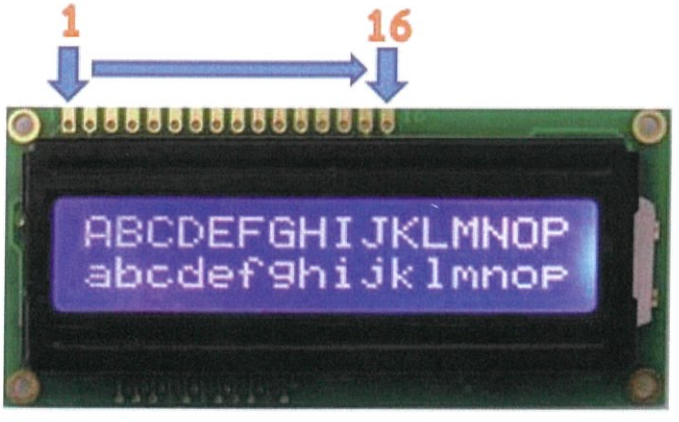
3.1.1 โครงสร้างของระบบ

1.จอLCD

2.วงจรวัดอุณหภูมิ

3.วงจรวัดกระแสไฟฟ้า

โครงสร้างของ LCD ทั่วไปจะประกอบขึ้นด้วยแผ่นแก้ว 2 แผ่นประกบกันอยู่ โดยเว้นช่องว่างตรงกลางไว้ 6-10 ไมโครเมตร ผิวด้านในของแผ่นแก้วจะเคลือบด้วยตัวนำไฟฟ้าแบบใสเพื่อใช้แสดงตัวอักษร ตรงกลางระหว่างตัวนำไฟฟ้าแบบใสกับผลึกเหลวจะมีชั้นของสารที่ทำให้โมเลกุลของผลึกรวมตัวกันในทิศทางที่แสงส่องมากระทบเรียกว่า Alignment Layer และผลึกเหลวที่ใช้โดยทั่วไปจะเป็นแบบ Magnetic



รูปที่ 3.1 จอ LCD แบบ Parallel 16 ขา

ตารางที่ 3.1 ตารางแสดง Pinout ของจอ LCD

Pin No.	Symbol	Description
1	VSS/GND	Ground
2	VDD	+5VDC
3	VO/VEE	LCD Control สำหรับปรับความเข้มของตัวอักษร
4	RS	Register Select เป็นขาอินพุตสำหรับเลือกเขียนอ่านข้อมูลในรีจิสเตอร์
5	RW	Read/Write เป็นขาอินพุตสำหรับเลือกโหมดเขียนหรืออ่านข้อมูล
6	E/EN	Enable เป็นขาอินพุตสำหรับสัญญาณ Pulse เมื่อต้องการเขียนหรืออ่านข้อมูล
7	DB0	Data Pins 8-Bit
8	DB1	
9	DB2	
10	DB3	
11	DB4	
12	DB5	
13	DB6	
14	DB7	
15	A	(LED+) เป็นขา Vcc สำหรับ LED backlight (5V)
16	K	(LED-) เป็นขา Gnd สำหรับ LED backlight (Gnd)

3.1.2 การควบคุมการแสดงผลของ LCD

ในการควบคุมหรือสั่งงาน ตัวจอ LCD นั้นมีส่วนควบคุม (Controller) รวมไว้ในตัวแล้ว ผู้ใช้สามารถส่งรหัสคำสั่งควบคุมการทำงานของจอ LCD ผ่าน Controller ว่าต้องการใช้แสดงผลอย่างไร โดย LCD Controller

ในการเชื่อมต่อระหว่าง LCD กับ Microcontroller มีดังนี้

1. GND เป็นกราวด์ใช้ต่อระหว่าง Ground ของระบบ Microcontroller กับ LCD

2. VCC เป็นไฟเลี้ยงวงจรที่ป้อนให้กับ LCD ขนาด +5VDC

3. VO ใช้ปรับความสว่างของหน้าจอ LCD

4. RS ใช้บอกให้ LCD Controller ทราบว่า Code ที่ส่งมาทางขา Data เป็นคำสั่งหรือข้อมูล

5. R/W ใช้กำหนดว่าจะอ่านหรือเขียนข้อมูลกับ LCD Controller

6. E เป็นขา Enable หรือ Chips Select เพื่อกำหนดการทำงานให้กับ LCD Controller

7-14. DB0-DB7 เป็นขาสัญญาณ Data ใช้สำหรับเขียนหรืออ่านข้อมูล/คำสั่ง กับ LCD Controller

วิธีการสั่งงานจะแตกต่างกันไป โดย LCD Controller สามารถรับรหัสคำสั่งจาก Microcontroller ได้จากสัญญาณ RS R/W และ DB0-DB7 ในขณะที่สัญญาณ E มีค่า Logic เป็น “1” ซึ่งสัญญาณเหล่านี้จะใช้ร่วมกันเพื่อกำหนดเป็นรหัสคำสั่งสำหรับสั่งงาน LCD โดยหน้าที่ของแต่ละสัญญาณพอสรุปได้ดังนี้

- E เป็นสัญญาณ Enable เมื่อมีค่าเป็น

“1” เป็นการบอกให้ LCD ทราบว่าอุปกรณ์ภายนอกต้องการติดต่ออ่านหรือเขียนข้อมูล

“0” ให้ LCD ไม่สนใจสัญญาณ RS R/W และ DB7-DB0

- RS เป็นสัญญาณสำหรับกำหนดให้ LCD ทราบว่าอุปกรณ์ภายนอกต้องการติดต่อกับ LCD ในขณะนั้นเป็นรหัสคำสั่งหรือข้อมูล โดยถ้า

RS = “0” หมายถึง คำสั่ง

RS = “1” หมายถึง ข้อมูล

- R/W เป็นสัญญาณสำหรับบอกให้ LCD ทราบว่าอุปกรณ์ภายนอกต้องการอ่านหรือเขียนกับ LCD โดยถ้า

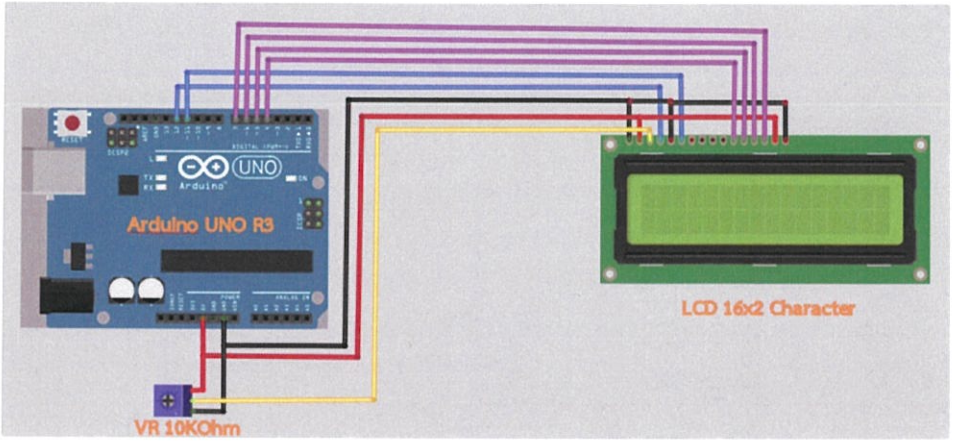
R/W = “0” หมายถึง เขียน

R/W = “1” หมายถึง อ่าน

- DB0-DB7 เป็นสัญญาณแบบ 2 ทิศทาง โดยจะสัมพันธ์กับสัญญาณ R/W ใช้สำหรับรับสั่งคำสั่งและข้อมูลระหว่าง LCD กับอุปกรณ์ภายนอก โดยถ้า R/W = “0” สัญญาณ DB7-DB0 จะส่งจากอุปกรณ์ภายนอกมาที่ LCD แต่ถ้า R/W = “1” สัญญาณ DB7-DB0 จะส่งจาก LCD ไปยังอุปกรณ์ภายนอก

3.1.3 การเชื่อมต่อสัญญาณขาข้อมูลระหว่าง Microcontroller กับ LCD Controller

การเชื่อมต่อสัญญาณขาข้อมูลระหว่าง Microcontroller กับ LCD Controller สามารถทำได้ 2 ลักษณะ คือ การเชื่อมต่อแบบ 8 บิต (DB0-DB7)



รูปที่ 3.2 ขาที่ใช้ในการเชื่อมต่อกับ Arduino UNO R3

ตารางที่ 3.2 แสดงตารางขาที่ใช้ในการเชื่อมต่อกับ Arduino UNO R3

VR 10 KOhm	LCD	Arduino
GND	VSS/GND	Ground
VCC	VDD	+5VDC
Signal	VO/VEE	-
-	RS	Digital Pin 12
-	RW	Ground (เพราะเราต้องการเขียน)
-	E/EN	Digital Pin 11
-	DB4	Digital Pin 4
-	DB5	Digital Pin 5
-	DB6	Digital Pin 6
-	DB7	Digital Pin 7
-	A	+5VDC
-	K	Ground

3.1.4 รายละเอียดคำสั่งในการสั่งงานระหว่าง Arduino กับ จอ LCD

คำสั่งในการควบคุมจอ LCD ของ Arduino นั้น ทาง Arduino.cc เขียนเป็น Library มาให้เพื่อสะดวกในการนำไปใช้งาน หลังจากต่อสายเสร็จเรียบร้อย ขั้นตอนแรกในการเริ่มเขียนโปรแกรมคือการเรียกใช้ Library ของ LCD จากไฟล์ชื่อ LiquidCrystal.h หลังจากนั้นมาดูกันว่ามีฟังก์ชั่นที่สำคัญอะไรบ้างที่ใช้สั่งงานให้จอ LCDฟังก์ชั่น LiquidCrystal(); ใช้ประกาศขาที่ต้องการส่งข้อมูลไปยังจอ LCD รูปแบบในการสั่งงานคือLiquidCrystal lcd(rs, enable, d0, d1, d2, d3, d4, d5, d6, d7) ฟังก์ชั่น begin(); ใช้กำหนดขนาดของจอ ในบทความนี้เราใช้ขนาด 16 ตัวอักษร 2 บรรทัด จึงประกาศเป็น lcd.begin(16, 2); ฟังก์ชั่น setCursor(); ใช้กำหนดตำแหน่งและบรรทัดของ Cursor เช่น lcd.setCursor(0, 1); คือ ให้เคอร์เซอร์ไปที่ตำแหน่งที่ 0 บรรทัดที่ 1 การนับตำแหน่งเริ่มจาก 0 ดังนั้น LCD 16x2 มีตำแหน่ง 0 – 15 บรรทัด คือ 0 กับ 1 ฟังก์ชั่น print(); ใช้กำหนดข้อความที่ต้องการแสดง เช่น lcd.print("Temp"); คือ ให้แสดงข้อความ “Temp” ออกทางหน้าจอ LCD

```
#include <DallasTemperature.h>
#include <LiquidCrystal.h>

// initialize the library with the numbers of the interface pins
LiquidCrystal lcd(3, 4, 5, 6, 7, 8);


OneWire oneWire(2);
DallasTemperature sensors(&oneWire);
void setup()
{
  lcd.begin(16, 2);
  sensors.begin();
  Serial.begin(9600);
}
void loop()
{
  sensors.requestTemperatures();
  float currentTemp0;
  currentTemp0 = sensors.getTempCByIndex(0);
  float currentTemp1;
  currentTemp1 = sensors.getTempCByIndex(1);
  float currentTemp2;
  currentTemp2 = sensors.getTempCByIndex(2);
Serial.print("Temp0 = ");
Serial.print(currentTemp0,2);
Serial.print("\t");
Serial.print("Temp1 = ");
Serial.print(currentTemp1,2);
Serial.print("\t");
Serial.print("Temp2 = ");
Serial.print(currentTemp2,2);
Serial.print("\n");
```

รูปที่ 3.3 แสดงโปรแกรมแสดงผล LCD

3.2 วงจรวัดอุณหภูมิ ไอซีวัดอุณหภูมิ DS18B20

เป็นไอซีที่วัดอุณหภูมิและให้ค่าแบบดิจิทัล เชื่อมต่อในรูปแบบของบัสที่เรียกว่า 1-Wire

3.2.1 ข้อมูลเชิงเทคนิคเกี่ยวกับไอซี DS18B20

- ใช้แรงดันไฟเลี้ยง Vdd (หรือ Vcc) ได้ในช่วง 3.0V ถึง 5.5V
- มี 3 ขา (สำหรับตัวถัง TO-92) คือ Gnd (Pin 1), DQ (Pin 2), Vdd (Pin 3)
- ใช้งานได้สองแบบ: normal mode (ใช้ทั้ง 3 ขา) และ parasite power mode (ใช้เพียง 2 ขา คือ DQ และ GND ในขณะที่ขา Vdd จะต่อกับขา Gnd)
- สามารถนำไอซีมาพ่วงต่อกันในบัสเดียว (เส้นสัญญาณ DQ) ได้หลายอุปกรณ์
- ในการใช้งาน จะต้องต่อ pull-up 4.7kΩ (หรือน้อยกว่าได้เล็กน้อย) ที่ขา DQ กับแรงดันไฟเลี้ยง
- วัดอุณหภูมิได้ในช่วง -55 °C ถึง +125 °C
- มีความแม่นยำ ±0.5 °C สำหรับอุณหภูมิในช่วง -10°C ถึง +85°C
- มีความละเอียดของค่าที่อ่านได้ 12 บิต (Resolution)
- ใช้เวลาในการแปลงข้อมูลสำหรับ ADC ไม่เกิน 750 msec (มิลลิวินาที) สำหรับข้อมูล 12 บิต
- ไอซีแต่ละตัวมีหมายเลขเฉพาะตัว ขนาด 64 บิต (64-bit serial code)
- สำหรับตระกูล DS18B20 มีค่าไบต์สำหรับ 8-bit family code ตรงกับ 28h (0x28) เป็นไบต์แรกของหมายเลขอุปกรณ์

ภายในไอซี DS18B20 มีหน่วยความจำแบบ SRAM ขนาดความจุ 9 ไบต์ (Byte 0 ถึง Byte 9) และเรียกว่า Scratchpad ส่วนหนึ่งของหน่วยความจำนี้ จะใช้สำหรับเก็บค่าอุณหภูมิที่ได้จากการอ่านและแปลงเป็นข้อมูลดิจิทัลในแต่ ละครั้ง (ใช้ 2 ไบต์ และเก็บไว้ใน Byte 0 และ Byte 1) และยังมีการคำนวณค่า CRC (checksum) ขนาดหนึ่งไบต์ด้วย (เก็บไว้ใน Byte 8)

ในการเขียนโปรแกรมสำหรับ Arduino เพื่อเชื่อมต่อกับไอซี DS18B20 สามารถใช้ไลบรารี่ "OneWire.h" ของ Arduino ซึ่งทำให้สะดวกในการเขียนโปรแกรม เช่น เพื่อการค้นหาหลายเลข (64-bit code) ของอุปกรณ์ที่ต่ออยู่กับบัส 1-Wire การสั่งให้ไอซีตามหมายเลขอุปกรณ์อ่านและแปลงค่าอุณหภูมิ จากนั้นเว้นระยะเวลาอย่างน้อย 750msec แล้วจึงอ่านค่าจากหน่วยความจำ Scratchpad ของไอซี

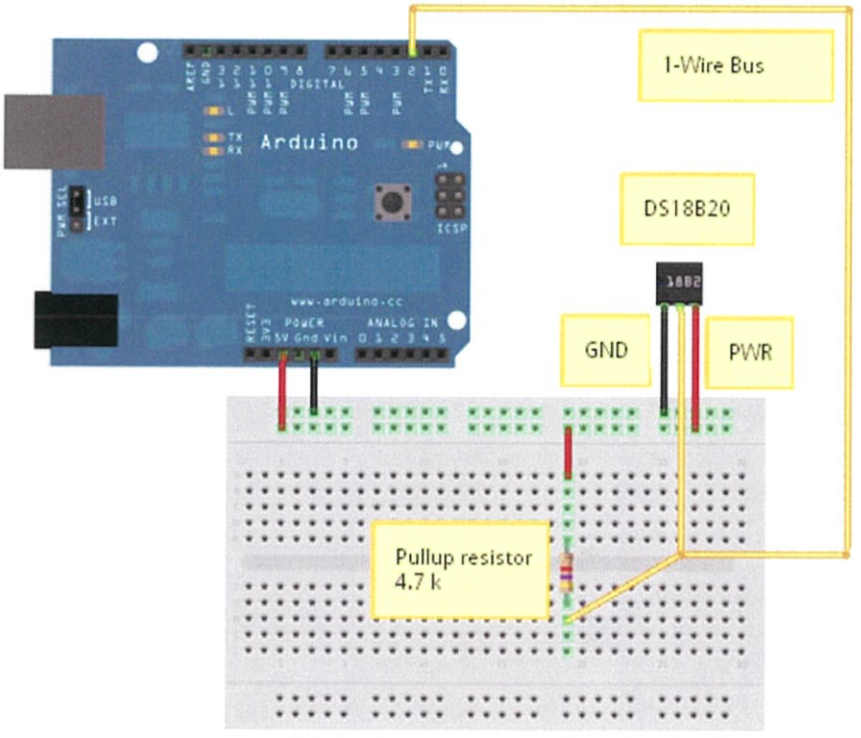
โค้ดตัวอย่างต่อไปนี้ ทำหน้าที่ค้นหาไอซี DS18B20 ที่ต่ออยู่กับบัส 1-Wire ซึ่งในตัวอย่างนี้ได้ใช้อุปกรณ์ 2 ชุด ขา DQ ของไอซีแต่ละตัว จะนำไปต่อกับขา D2 ของ Arduino และที่ขานี้จะต้องต่อ 4.7kΩ แบบ pull-up กับ +5V อยู่ด้วย ข้อมูลที่ได้จากอุปกรณ์วัดอุณหภูมิทั้งสองชุดจะปรากฏเป็นข้อความที่ถูกส่งผ่านทาง Serial (Baudrate 115200)

```
#include <OneWire.h>
#include <DallasTemperature.h>
#define ONE_WIRE_BUS 2
OneWire oneWire(ONE_WIRE_BUS);
DallasTemperature sensors(&oneWire);
void setup(void)
{

  Serial.begin(9600);
  Serial.println("Dallas Temperature IC Control Library Demo");
  sensors.begin();
}


void loop(void)
{
  float average = 0;
  for(int i = 0; i < 1000; i++)
{
  average = average + (.0264 * analogRead(A0) -13.51) / 1000;
  Serial.println("current done");
  Serial.print(" Requesting temperatures...");
  sensors.requestTemperatures();
  Serial.println("DONE");
  Serial.print("Temperature for Device 1 is: ");
  Serial.print(sensors.getTempCByIndex(0));
 delay(1000);
}
}
```

รูปที่ 3.4 แสดงผลเซ็นเซอร์วัดอุณหภูมิ DS18B20



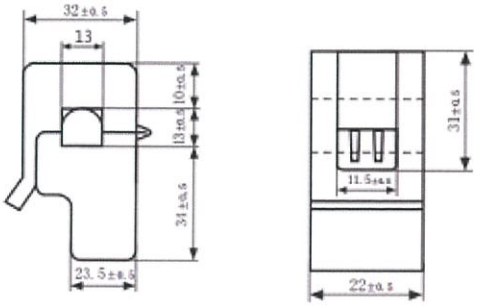
รูปที่ 3.5 แสดงการต่อวงของเซ็นเซอร์วัดอุณหภูมิ

3.3 วงจรวัดกระแสไฟฟ้า

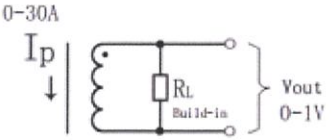
รุ่นที่ใช้ มี R-Burden อยู่ภายใน(Build-in) ให้แล้วเมื่อมีการใช้งาน Primary กระแสไหลผ่าน 0-30A ทาง Secondary ก็จะมีกระแสไหลผ่าน R ทำให้เกิดแรงดัน 0-1V ให้เลย

Core material:Ferrite
Fire resistance property:in accordance with UL 94-V0
Dielectric strength: 1500V AC/1min 5mA (between shell and output)

Outline size diagram:(in mm)



Front View　　Side View



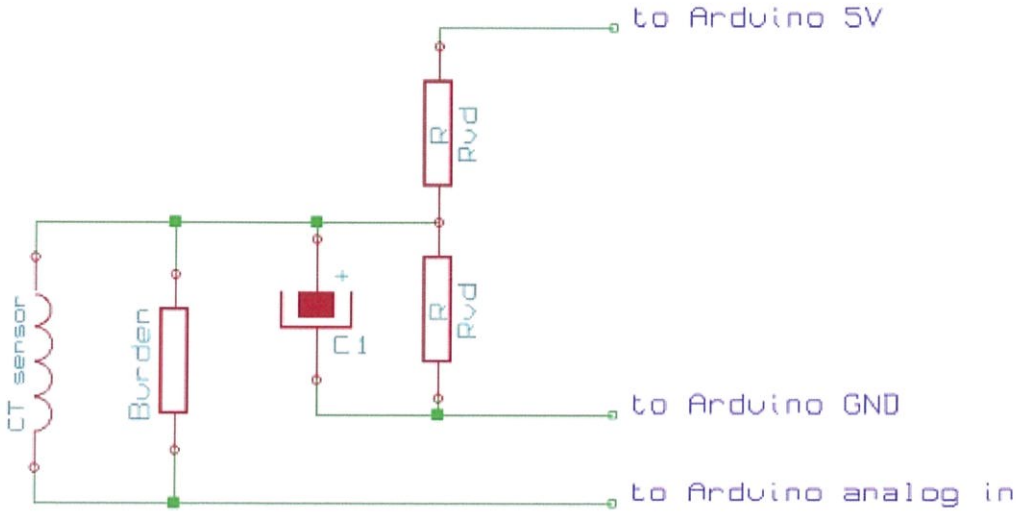
Voltage output type　　Schematic Diagram

Typical table of technical parameters:

input current	output voltage	non-linearity	build-in sampling resistance (RL)
0-30A	0-1V	±1%	62Ω
turn ratio	resistance grade	work temperature	dielectric strength(between shell and output)
1800:1	Grade B	-25℃~+70℃	1500V AC/1min 5mA

รูปที่ 3.6 แสดงรายละเอียดของ SCT-013-030

เพราะแรงดันออกมาเป็น AC +-1V การต่อเข้า ไมโครคอนโทรลเลอร์ โดยตรง ที่รับได้แต่ DC ก็ใช้ตามนี้ก็ได้ โดยใช้ R แบ่งแรงดัน เป็น 2.5V และใช้ C ช่วยรักษาระดับแรงดันไว้ต่อ Out ของ CT ด้านหนึ่งเข้าจุด แรงดัน 2.5V อีกด้านต่อเข้า ADC ของ ไมโครฯ ได้แรงดันที่อ่านได้ ก็จะเป็น 2.5V สวิง +-1V ถ้าเซ็ต ADC เป็น 10bit ค่าที่อ่านได้ ก็จะเป็น 511 +-ค่า ADC ที่เพิ่มขึ้นหรือลดลง ประมาณ +-204



รูปที่ 3.7 แสดงวงจรของ SCT-013-030

```
// EmonLibrary examples openenergymonitor.org, Licence GNU GPL V3

#include "EmonLib.h"                   // Include Emon Library
EnergyMonitor emon1;                   // Create an instance

void setup()
{
  Serial.begin(9600);

  emon1.current(0, 29);             // Current: input pin, calibration.
}

void loop()
{
  double Irms = emon1.calcIrms(1480);  // Calculate Irms only

  Serial.print(Irms*230.0);            // Apparent power
  Serial.print(" ");
  Serial.println(Irms*);
  delay(1000);                  // Irms
}
```

รูปที่ 3.8 แสดงโปรแกรมการวัดกระแสไฟฟ้าของ SCT-013-030

บทที่ 4

การทดลองและบันทึกผล

การทดลองแบ่งออกได้เป็น 2 การทดลองคือ

1. การทดลองวัดค่ากระแสไฟฟ้าขณะที่แอร์มีการทำงาน

2. การทดลองวัดค่าอุณหภูมิ

4.1 การทดลองวัดค่ากระแสไฟฟ้าขณะที่เครื่องปรับอากาศทำงาน

4.1.1 วัตถุประสงค์

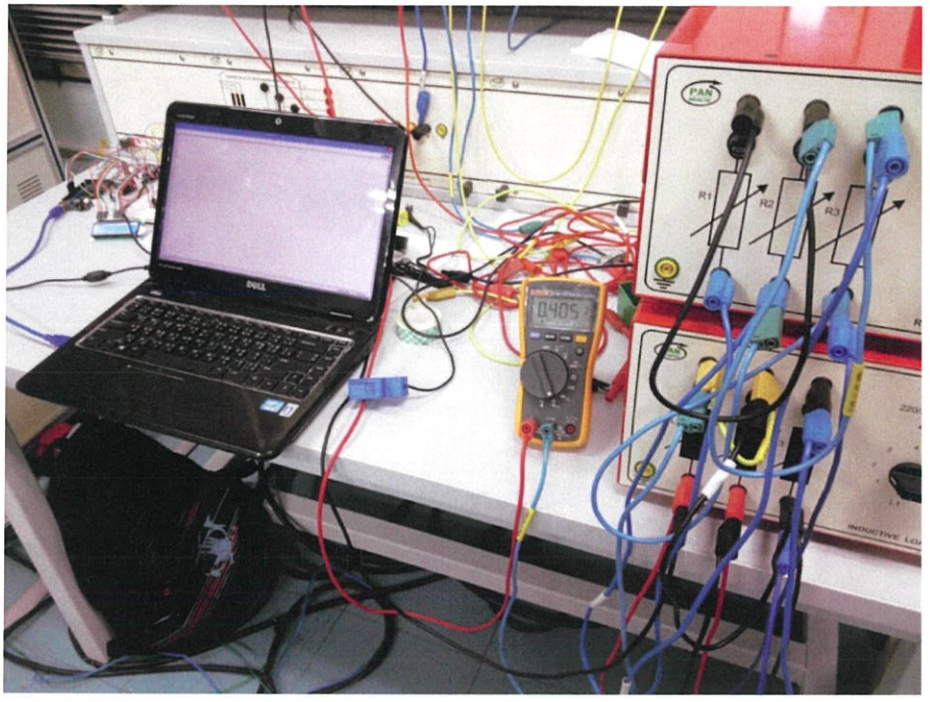
1) สามารถทราบค่ากระแสไฟฟ้า ของเครื่องปรับอากาศขณะที่มีการใช้งานได้

2) สามารถนำค่าที่ได้มาเทียบกับค่ามาตรฐาน เพื่อหาค่าความผิดพลาด ของการทำงานเครื่องปรับอากาศได้

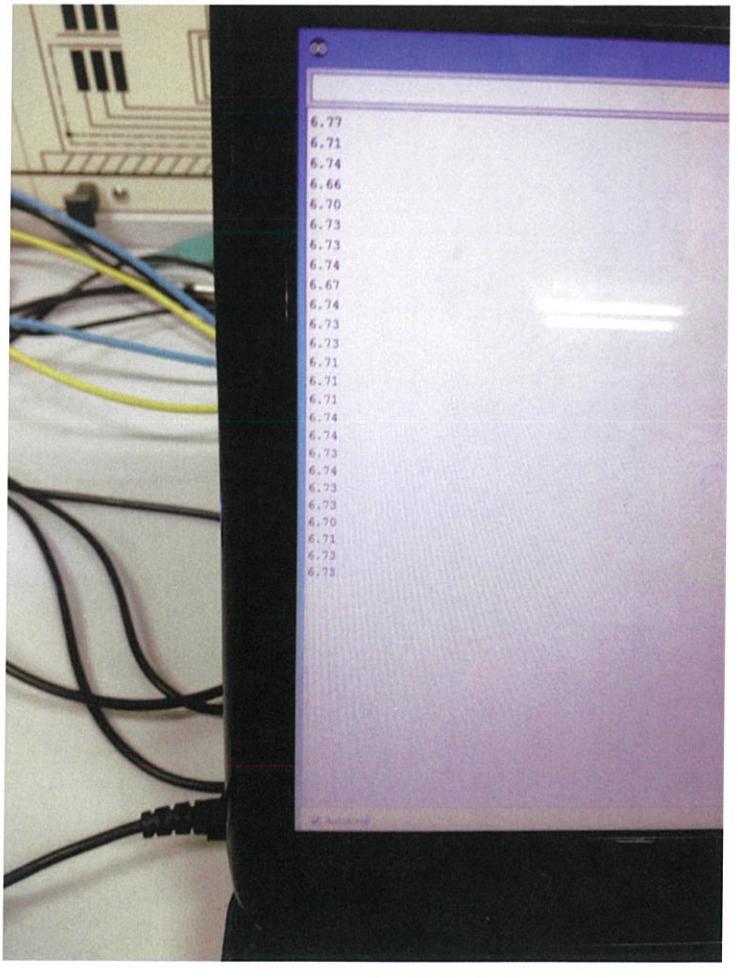
4.1.2 วิธีการทดลอง

1) นำชุดอุปกรณ์วัดกระแสไฟฟ้า ที่ได้สร้างขึ้นมาสอบเทียบกับ ชุดเครื่องมือวัด เพื่อหาค่าความคลาดเคลื่อน และทำการปรับแก้

2) เมื่อทำการปรับแก้ชุดอุปกรณ์วัดกระแสไฟฟ้าแล้ว ก็นำชุดอุปกรณ์วัดกระแสไฟฟ้าไปติดตั้งกับสายไฟของ ชุดอัดความดันเครื่องปรับอากาศ



รูปที่ 4.1 แสดงการสอบเทียบเซ็นเซอร์วัดกระแสไฟฟ้า



รูปที่ 4.2 แสดงการสอบเทียบเซ็นเซอร์วัดกระแสไฟฟ้า

4.1.3 ผลการทดลอง

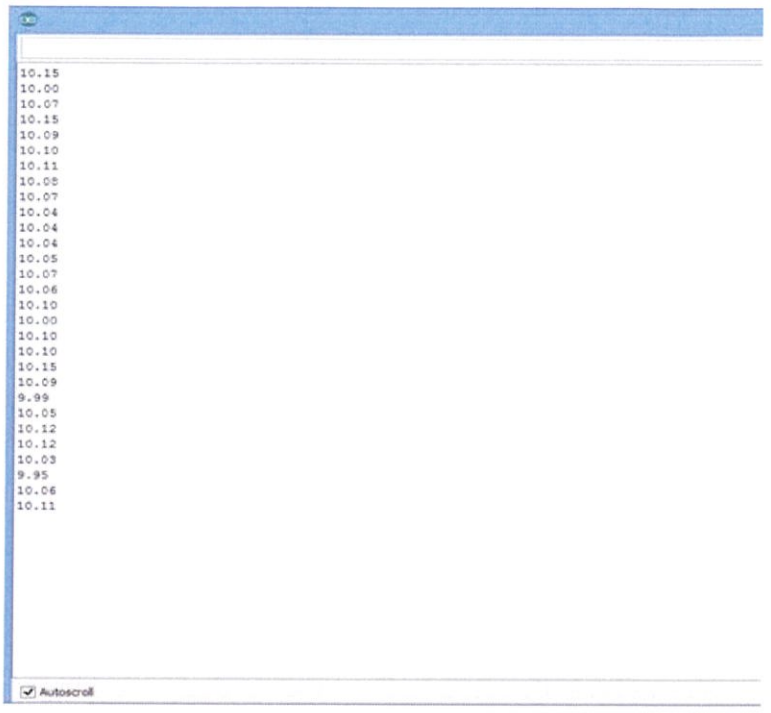
1) ที่จอแสดงผลจะสามารถแสดงค่าของกระแสไฟฟ้า ที่เครื่องปรับอากาศขณะกำลังทำงานอยู่ได้

ตารางที่ 4.1 แสดงการสอบเทียบเซ็นเซอร์วัดกระแสไฟฟ้า

			ครั้งที่ 1						
fluke			stc-030			รวม	จำนวนค่า	เฉลี่ย	ผลต่าง	%คาดเคลื่อน
0.134	0.14					0.14	1	0.14	-0.006	4.48
0.291	0.3					0.3	1	0.3	-0.009	3.09
0.454	0.45	0.46				0.91	2	0.455	-0.001	0.22
0.725	0.71	0.72				1.43	2	0.715	0.01	1.38
0.982	0.96	0.95				1.91	2	0.955	0.027	2.75
1.189	1.16	1.17				2.33	2	1.165	0.024	2.02
1.325	1.29	1.3				2.59	2	1.295	0.03	2.26
1.479	1.44	1.45				2.89	2	1.445	0.034	2.30
1.641	1.61	1.62	1.6			4.83	3	1.61	0.031	1.89
1.871	1.83	1.84	1.82			5.49	3	1.83	0.041	2.19
2.059	2.01	2.02	2.03	2.04		8.1	4	2.025	0.034	1.65
2.29	2.24	2.25	2.26	2.27		9.02	4	2.255	0.035	1.53
2.423	2.38	2.39	2.4			7.17	3	2.39	0.033	1.36
2.583	2.53	2.54	2.55	2.56		10.18	4	2.545	0.038	1.47
2.743	2.69	2.71	2.72			8.12	3	2.706667	0.036333	1.32
3.013	2.94	2.95	2.96	2.97	2.98	14.8	5	2.96	0.053	1.76
3.265	3.18	3.19	3.2	3.21		12.78	4	3.195	0.07	2.14
3.475	3.39	3.4	3.41			10.2	3	3.4	0.075	2.16
3.603	3.51	3.52	3.53	3.54		14.1	4	3.525	0.078	2.16
3.76	3.69	3.7	3.68	3.71		14.78	4	3.695	0.065	1.73
3.92	3.85	3.86	3.82	3.83	3.84	19.2	5	3.84	0.08	2.04
4.02	3.94	3.95	3.96			11.85	3	3.95	0.07	1.74
4.304	4.22	4.24	4.19	4.2		16.85	4	4.2125	0.0915	2.13
4.463	4.38	4.35	4.36	4.37		17.46	4	4.365	0.098	2.20
4.766	4.68	4.67	4.69			14.04	3	4.68	0.086	1.80
5.107	5	4.99	5.02	5.06	5.08	25.15	5	5.03	0.077	1.51
5.404	5.29	5.3	5.28	5.32	5.27	26.46	5	5.292	0.112	2.07
5.667	5.55	5.56				11.11	2	5.555	0.112	1.98
7.2	7.08	7.07	7.06	7.11		28.32	4	7.08	0.12	1.67

ตารางที่ 4.2 แสดงการสอบเทียบเซ็นเซอร์วัดกระแสไฟฟ้า

			ครั้งที่ 2						
fluke			stc-030			รวม	จำนวนค่า	เฉลี่ย	ผลต่าง	%คาดเคลื่อน
0.137	0.14					0.14	1	0.14	-0.003	2.19
0.415	0.42					0.42	1	0.42	-0.005	1.20
0.553	0.56	0.57				1.13	2	0.565	-0.012	2.17
0.786	0.81	0.82				1.63	2	0.815	-0.029	3.69
0.924	0.95	0.94				1.89	2	0.945	-0.021	2.27
1.204	1.21	1.22	1.23			3.66	3	1.22	-0.016	1.33
1.34	1.35	1.36	1.37			4.08	3	1.36	-0.02	1.49
1.5	1.51	1.52	1.53	1.54		6.1	4	1.525	-0.025	1.67
1.764	1.77	1.78	1.79			5.34	3	1.78	-0.016	0.91
1.921	1.92	1.93	1.94	1.95		7.74	4	1.935	-0.014	0.73
2.112	2.12	2.13	2.14	2.15		8.54	4	2.135	-0.023	1.09
2.276	2.3	2.31	2.32			6.93	3	2.31	-0.034	1.49
2.54	2.55	2.56	2.57			7.68	3	2.56	-0.02	0.79
2.7	2.71	2.72	2.73	2.74		10.9	4	2.725	-0.025	0.93
2.867	2.88	2.89	2.9	2.91		11.58	4	2.895	-0.028	0.98
3.075	3.08	3.1	3.12			9.3	3	3.1	-0.025	0.81
3.238	3.25	3.26	3.28			9.79	3	3.263333	-0.02533	0.78
3.474	3.49	3.5	3.51			10.5	3	3.5	-0.026	0.75
3.642	3.64	3.65	3.67			10.96	3	3.653333	-0.01133	0.31
3.88	3.9	3.91	3.92			11.73	3	3.91	-0.03	0.77
4.046	4.05	4.06	4.07			12.18	3	4.06	-0.014	0.35
4.112	4.12	4.13	4.14	4.15		16.54	4	4.135	-0.023	0.56
4.276	4.27	4.29	4.31			12.87	3	4.29	-0.014	0.33
4.36	4.35	4.36	4.37	4.38		17.46	4	4.365	-0.005	0.11
4.517	4.53	4.52				9.05	2	4.525	-0.008	0.18
4.706	4.7	4.71	4.72			14.13	3	4.71	-0.004	0.08
4.91	4.92	4.93	4.91			14.76	3	4.92	-0.01	0.20
5	4.99	5	5.01			15	3	5	0	-
5.254	5.24	5.25	5.26			15.75	3	5.25	0.004	0.08
5.375	5.35	5.36	5.37			16.08	3	5.36	0.015	0.28



รูปที่ 4.3 ผลการวัดค่ากระแสไฟฟ้า

4.2 การทดลองวัดค่าอุณหภูมิ

4.2.1 วัตถุประสงค์

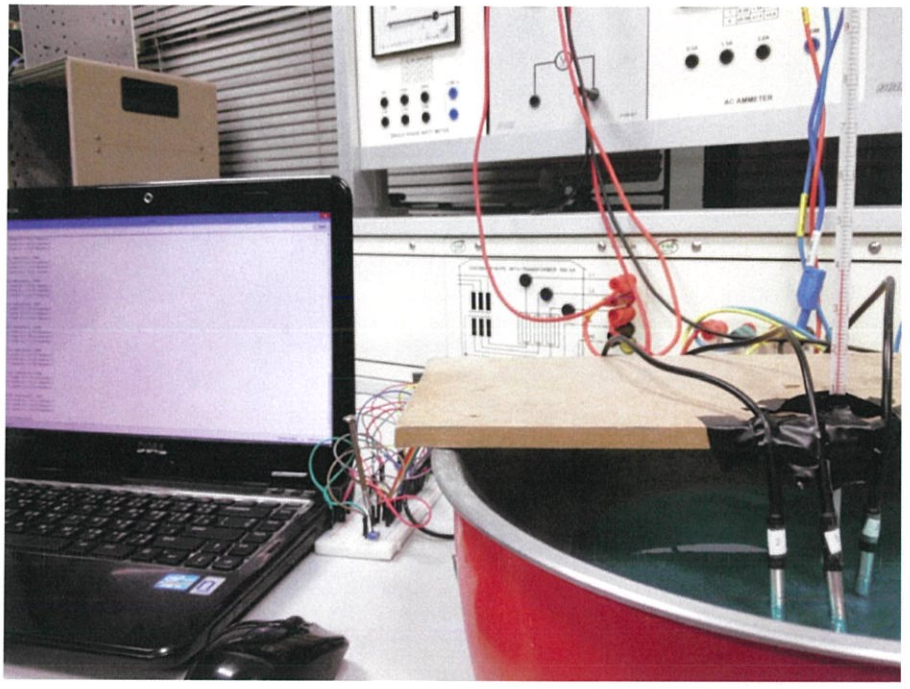
1) สามารถทราบค่าอุณหภูมิ ของจุดต่างๆของระบบเครื่องปรับอากาศในขณะที่เครื่องปรับอากาศกำลังทำงานได้

2) สามารถนำค่าที่ได้มาเทียบกับค่ามาตรฐาน เพื่อหาค่าความผิดพลาด ของการทำงานเครื่องปรับอากาศได้

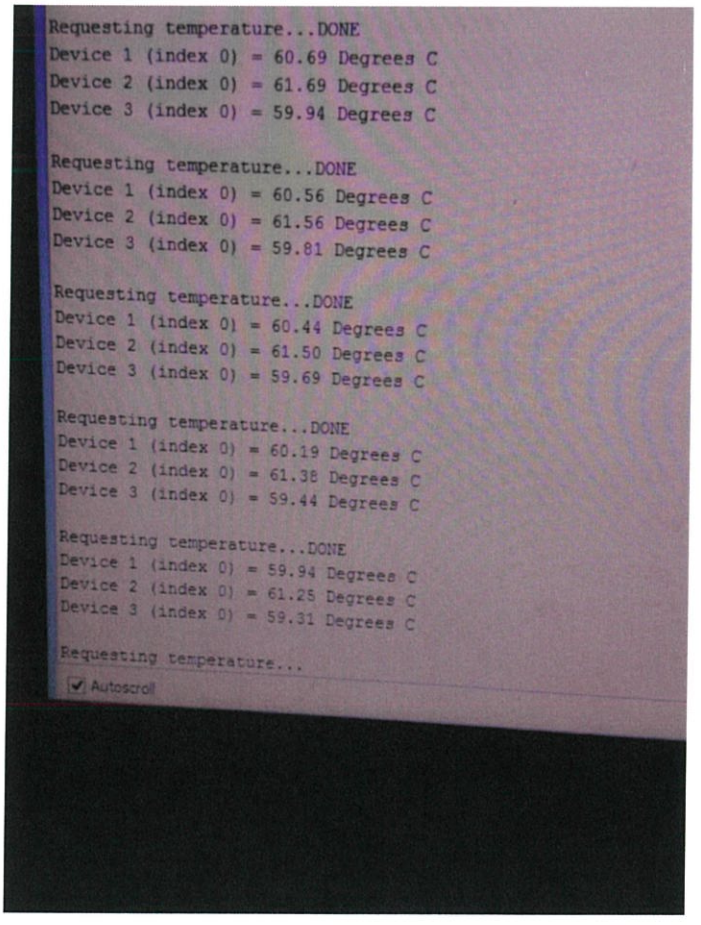
4.2.2 วิธีการทดลอง

1) นำชุดอุปกรณ์วัดอุณหภูมิ ที่ได้สร้างขึ้นมาสอบเทียบกับ น้ำเดือด และน้ำแข็งที่วัดค่าโดยเทอร์โมมิเตอร์ เพื่อหาค่าความคลาดเคลื่อนและทำการปรับแก้

2) เมื่อทำการปรับแก้ชุดอุปกรณ์วัดอุณหภูมิแล้ว ก็นำชุดอุปกรณ์วัดวัดอุณหภูมิไปติดตั้งกับจุดต่างๆของ ชุดอัดความดัน และบริเวณแผงทำความเย็นเครื่องปรับอากาศ



รูปที่ 4.4 แสดงการสอบเทียบเซ็นเซอร์วัดอุณหภูมิ



รูปที่ 4.5 แสดงการสอบเทียบเซ็นเซอร์วัดอุณหภูมิ

4.2.3 ผลการทดลอง

1) ที่จอแสดงผลจะสามารถแสดงค่าของอุณหภูมิ ณ จุดต่างๆ ขณะที่เครื่องปรับอากาศขณะกำลังทำงานอยู่ได้

ตารางที่ 4.3 แสดงการสอบเทียบเซ็นเซอร์วัดอุณหภูมิ

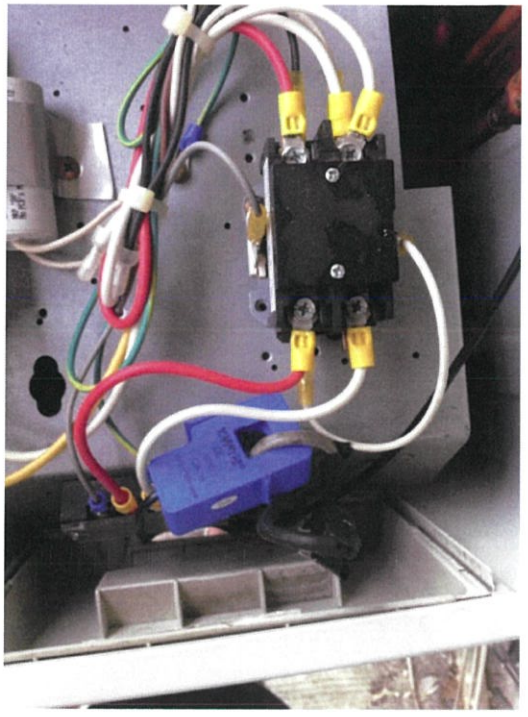
สอบเทียบ			
อุณหภูมิ	T1	T2	T3
30	30.61	30.58	30.81
35	35.00	34.25	35.00
40	40.63	41.00	40.94
45	44.69	46.25	44.56
50	50.69	52.25	50.38
55	51.69	50.79	49.61
60	59.31	61.25	59.31

4.3 การติดตั้งอุปกรณตรวจวัด

4.3.1 เซ็นเซอร์วัดกระแสไฟฟ้า

1) ติดตั้งเพื่อบอกสถานการทำงานของคอมเพรสเซอร์ ว่ากระแสที่เข้ามาในคอมเพรสเซอร์เป็นอย่างไรในขณะนั้น

2) ติดตั้งบริเวณแหล่งจ่ายที่เข้าคอมเพรสเซอร์



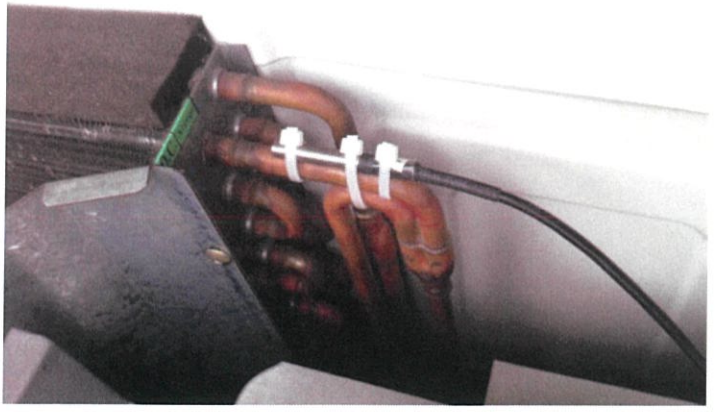
รูปที่ 4.6 แสดงการการติดตั้งเซ็นเซอร์วัดกระแสไฟฟ้า

4.3.2 เซ็นเซอร์วัดอุณหภูมิ

4.3.2.1 เซ็นเซอร์วัดอุณหภูมิตัวที่ 1 (T_1)

1) ติดตั้งเพื่อบอกคุณภาพการทำงานของคอมเพรสเซอร์ว่ามีประสิทธิภาพดีหรือไม่

2) ติดตั้งบริเวณทางเข้าหน้าคอยล์ร้อน

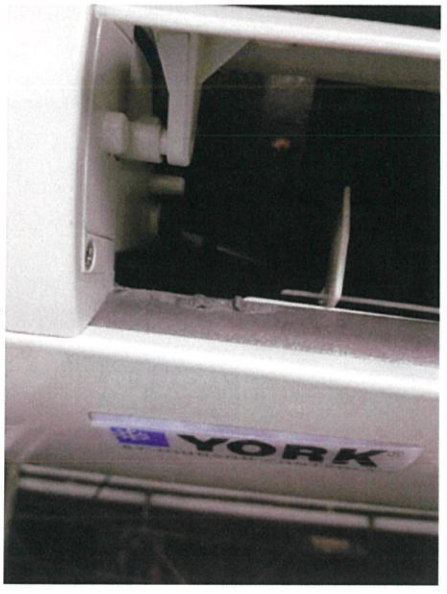


รูปที่ 4.7 แสดงการติดตั้งเซ็นเซอร์วัดอุณหภูมิตัวที่ 1

4.3.2.2 เซ็นเซอร์วัดอุณหภูมิตัวที่ 2 (T_2)

1) ติดตั้งเพื่อบอกประสิทธิภาพการทำงานของพัดลมหน้าคอยล์เย็น

2) ติดตั้งบริเวณหน้าพัดลมทางออกของช่องแอร์หน้าคอยล์เย็น

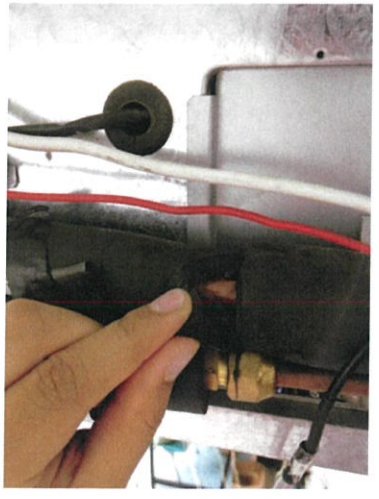


รูปที่ 4.8 แสดงการติดตั้งเซ็นเซอร์วัดอุณหภูมิตัวที่ 2

4.3.2.3 เซ็นเซอร์วัดอุณหภูมิตัวที่ 3 (T_3)

1) ติดตั้งเพื่อบอกคุณภาพการทำงานของคอยล์เย็นและบอกปริมาณของน้ำยา

2) ติดตั้งบริเวณคอยล์เย็น



รูปที่ 4.9 แสดงการติดตั้งเซ็นเซอร์วัดอุณหภูมิตัวที่ 3

บทที่ 5

สรุปผล

5.1 สรุปผลการทดลอง

ตารางที่ 5.1 Error Code

ค่าที่แสดง	จุดที่เกิด	สาเหตุ
N	ไม่มีปัญหาใดๆ เป็นปกติ	- ไม่มี
C	กระแส Compressor ผิดปกติ	- น้ำยาของระบบน้อยหรือเกิดการรั่ว - คอมเพรสเซอร์เสีย - เกิดการชำรุดที่สายไฟหรือสายไฟหลุด - แผงควบคุมหรืออุปกรณ์อิเล็กทรอนิกส์เสีย - อุปกรณ์หรือสวิทช์เกิดการค้างค่าสถานะ
FC	อุณหภูมิ Fan Coil Indoor ผิดปกติ	- มีฝุ่นเกาะที่แผ่นกรองมาก - น้ำยาของระบบมีน้อยหรือมากเกินไป - มอเตอร์พัดลมไม่ทำงานหรือขัดข้อง
CT	อุณหภูมิ Cool Coil Tubeผิดปกติ	- น้ำยาในระบบมากหรือน้อยกว่าปกติ
CO	การทำงานในโหมด Cut-Off	- เครื่องหยุดการทำงานหรืออยู่ในช่วงตัดไฟ
N/A	ยังไม่ทราบที่เกิด	- ค่าที่ทำการวัดยังไม่คงที่ต้องใช้เวลาในการวัด

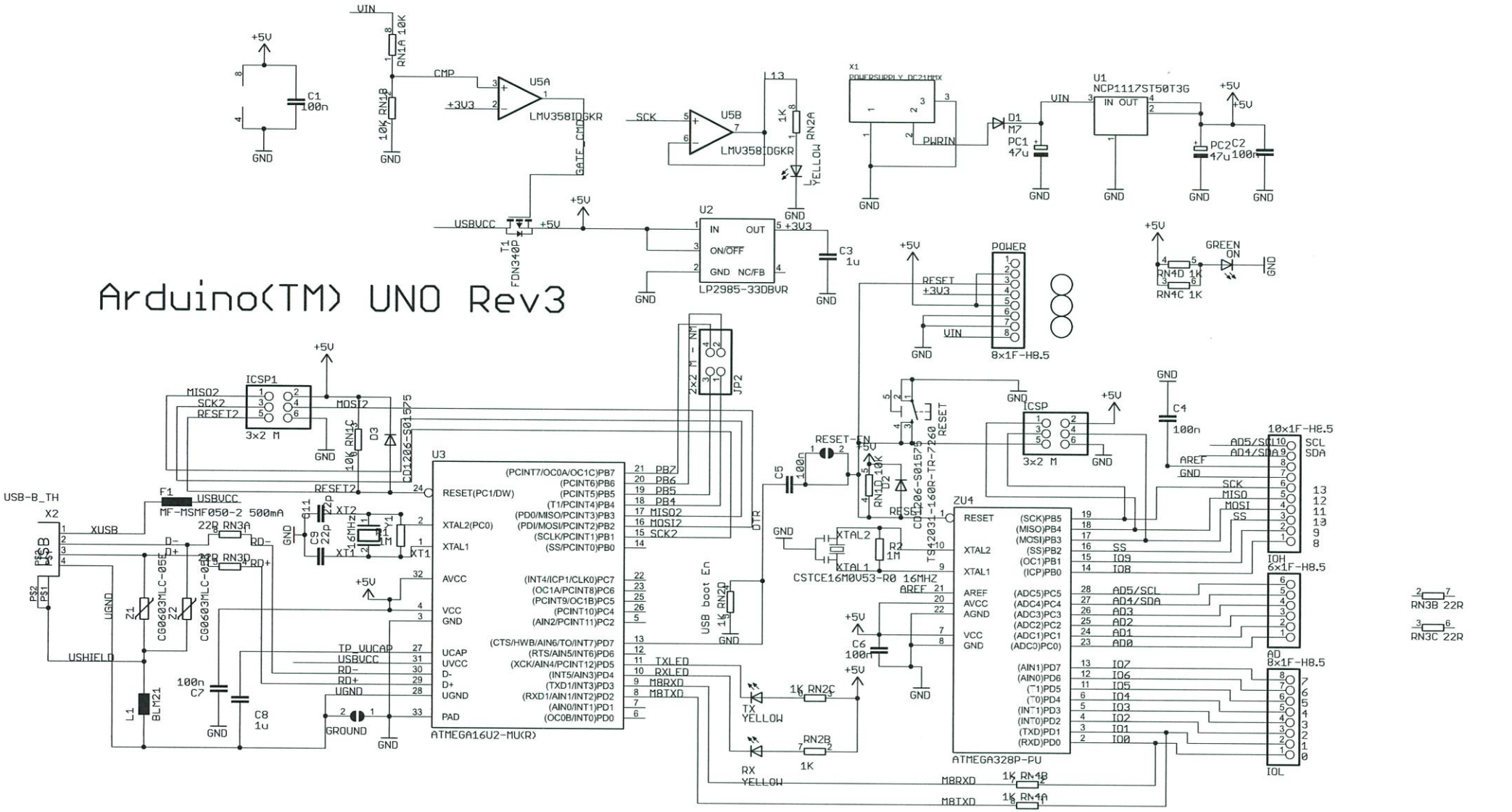
5.2 สรุปผลงานวิจัย

เนื่องด้วยโครงงานนี้เป็นโครงงานที่เกี่ยวระบบเครื่องปรับอากาศ โดยที่เป็นการสร้างอุปกรณ์แสดงสถานภาพการทำงานของเครื่องปรับอากาศ เพื่อทำให้สามารถวิเคราะห์ปัญหาที่จะเกิดขึ้นของเครื่องปรับอากาศ อันเนื่องมาจากการทำงานของอุปกรณ์ภายในเครื่องปรับอากาศ นำมาซึ่งการเข้าไปแก้ปัญหาของเครื่องปรับอากาศได้อย่างรวดเร็ว ตรงจุด และถูกต้อง เป็นการลดขั้นตอนการตรวจสอบหาจุดที่เป็นผิดปกติภายในการทำงานของระบบเครื่องปรับกาศได้

5.3 ข้อเสนอแนะ

โครงงานนี้ได้ใช้บอร์ด Arduino ที่เป็นสำเร็จรูปที่ใช้งานกันอย่างแพร่หลาย และมีซอฟแวร์ใช้ร่วมกันเป็นซอฟแวร์ประเภทที่สามารถนำไปใช้ได้อย่างเสรี แต่การใช้บอร์ดสำเร็จรูปประเภทนี้ก็จะมีต้นทุนค่อนข้างสูงในการทำไปสร้างโครงงาน และยิ่งเมื่อสร้างในจำนวนมากๆยิ่งจะทำให้เกิดต้นทุนจำนวนมาก ถ้าสามารถสร้างบอร์ดพัฒนาโปรแกรมขึ้นมาได้เอง และซอฟแวร์ที่ใช้ก็สามารถใช้ซอฟแวร์เดียวกันกับของ Arduino ได้เพราะเป็นซอฟแวร์เสรีสามารถนำไปพัฒนาต่อยอดได้ ถ้าสามารถพัฒนาได้สำเร็จโดยที่ไม่จำเป็นต้องใช้บอร์ดสำเร็จรูปได้ ก็จะเป็นการลดต้นทุนในการใช้บอร์ดสำเร็จรูปลงได้อย่างมาก และยังทำให้การพัฒนาต่อยอดโครงงานนี้ไปติดตั้งใช้งานกับเครื่องปรับอากาศในอาคารหรือสถานที่ต่างๆ ซึ่งจะทำให้ต้นทุนต่อหนึ่งหน่วยในการสร้างนั้นก็จะลดลงเป็นประโยชน์ต่อการพัฒนาอย่างมาก

ภาคผนวก



Reference Designs ARE PROVIDED "AS IS" AND "WITH ALL FAULTS. Arduino DISCLAIMS ALL OTHER WARRANTIES, EXPRESS OR IMPLIED, REGARDING PRODUCTS, INCLUDING BUT NOT LIMITED TO, ANY IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE
Arduino may make changes to specifications and product descriptions at any time, without notice. The Customer must not rely on the absence or characteristics of any features or instructions marked "reserved" or "undefined." Arduino reserves these for future definition and shall have no responsibility whatsoever for conflicts or incompatibilities arising from future changes to them. The product information on the Web Site or Materials is subject to change without notice. Do not finalize a design with this information.

ARDUINO is a registered trademark.

Use of the ARDUINO name must be compliant with http://www.arduino.cc/en/Main/Policy

DS18B20
Programmable Resolution 1-Wire Digital Thermometer

DESCRIPTION

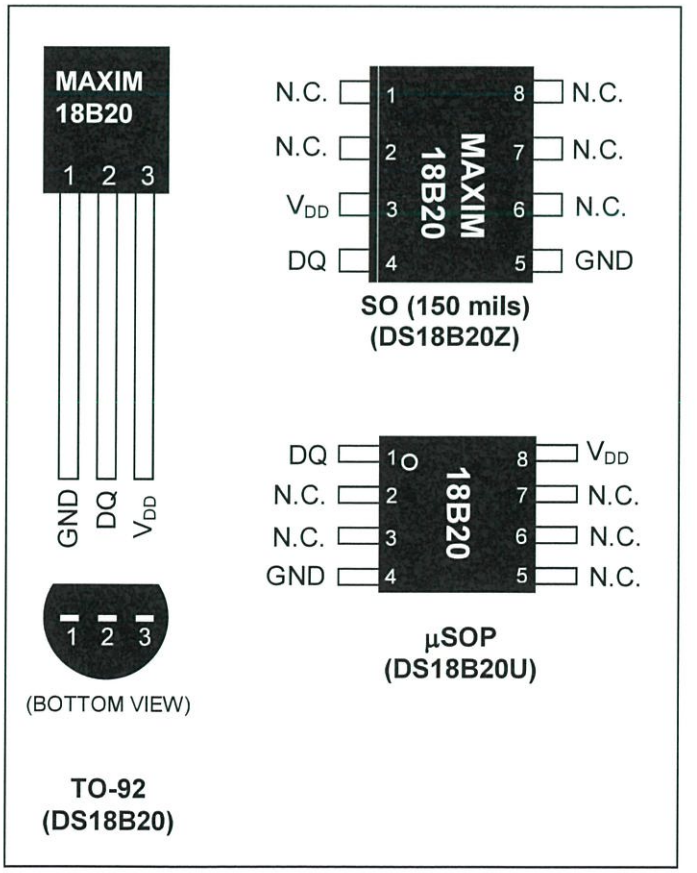
The DS18B20 digital thermometer provides 9-bit to 12-bit Celsius temperature measurements and has an alarm function with nonvolatile user-programmable upper and lower trigger points. The DS18B20 communicates over a 1-Wire bus that by definition requires only one data line (and ground) for communication 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 DS18B20 can derive power directly from the data line ("parasite power"), eliminating the need for an external power supply.

Each DS18B20 has a unique 64-bit serial code, which allows multiple DS18B20s to function on the same 1-Wire bus. Thus, it is simple to use one microprocessor to control many DS18B20s 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.

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
- Thermometer Resolution is User Selectable from 9 to 12 Bits
- Converts Temperature to 12-Bit Digital Word 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)
- Available in 8-Pin SO (150 mils), 8-Pin µSOP, and 3-Pin TO-92 Packages
- Software Compatible with the DS1822
- Applications Include Thermostatic Controls, Industrial Systems, Consumer Products, Thermometers, or Any Thermally Sensitive System

PIN CONFIGURATIONS



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

For pricing, delivery, and ordering information, please contact Maxim Direct at 1-888-629-4642, or visit Maxim's website at www.maximintegrated.com.

REV: 042208

ORDERING INFORMATION

PART	TEMP RANGE	PIN-PACKAGE	TOP MARK
DS18B20	-55°C to +125°C	3 TO-92	18B20
DS18B20+	-55°C to +125°C	3 TO-92	18B20
DS18B20/T&R	-55°C to +125°C	3 TO-92 (2000 Piece)	18B20
DS18B20+T&R	-55°C to +125°C	3 TO-92 (2000 Piece)	18B20
DS18B20-SL/T&R	-55°C to +125°C	3 TO-92 (2000 Piece)*	18B20
DS18B20-SL+T&R	-55°C to +125°C	3 TO-92 (2000 Piece)*	18B20
DS18B20U	-55°C to +125°C	8 µSOP	18B20
DS18B20U+	-55°C to +125°C	8 µSOP	18B20
DS18B20U/T&R	-55°C to +125°C	8 µSOP (3000 Piece)	18B20
DS18B20U+T&R	-55°C to +125°C	8 µSOP (3000 Piece)	18B20
DS18B20Z	-55°C to +125°C	8 SO	DS18B20
DS18B20Z+	-55°C to +125°C	8 SO	DS18B20
DS18B20Z/T&R	-55°C to +125°C	8 SO (2500 Piece)	DS18B20
DS18B20Z+T&R	-55°C to +125°C	8 SO (2500 Piece)	DS18B20

+Denotes a lead-free package. A "+" will appear on the top mark of lead-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
SO	µSOP	TO-92		
1, 2, 6, 7, 8	2, 3, 5, 6, 7	—	N.C.	No Connection
3	8	3	V_{DD}	Optional V_{DD}. V_{DD} must be grounded for operation in parasite power mode.
4	1	2	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 DS18B20* section.)
5	4	1	GND	Ground

OVERVIEW

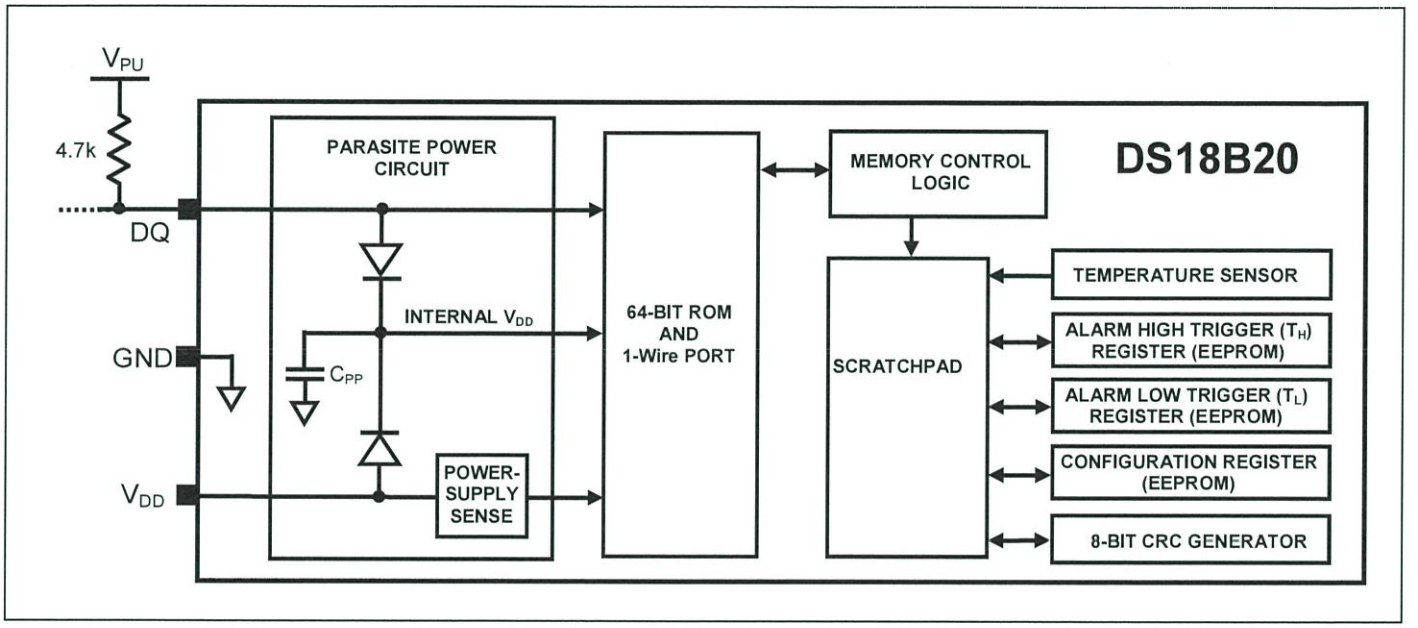
Figure 1 shows a block diagram of the DS18B20, and pin descriptions are given in the *Pin Description* table. The 64-bit ROM stores the device's unique serial code. The scratchpad memory contains the 2-byte temperature register that stores the digital output from the temperature sensor. In addition, the scratchpad provides access to the 1-byte upper and lower alarm trigger registers (T_H and T_L) and the 1-byte configuration register. The configuration register allows the user to set the resolution of the temperature-to-digital conversion to 9, 10, 11, or 12 bits. The T_H, T_L, and configuration registers are nonvolatile (EEPROM), so they will retain data when the device is powered down.

The DS18B20 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 DS18B20). 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 DS18B20 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 DS18B20 may also be powered by an external supply on V_{DD}.

Figure 1. DS18B20 Block Diagram



OPERATION—MEASURING TEMPERATURE

The core functionality of the DS18B20 is its direct-to-digital temperature sensor. The resolution of the temperature sensor is user-configurable to 9, 10, 11, or 12 bits, corresponding to increments of 0.5°C, 0.25°C, 0.125°C, and 0.0625°C, respectively. The default resolution at power-up is 12-bit. The DS18B20 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 DS18B20 returns to its idle state. If the DS18B20 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 DS18B20 will respond by transmitting 0 while the temperature conversion is in progress and 1 when the conversion is done. If the DS18B20 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 DS18B20* section.

The DS18B20 output temperature data is calibrated in degrees Celsius; 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. If the DS18B20 is configured for 12-bit resolution, all bits in the temperature register will contain valid data. For 11-bit resolution, bit 0 is undefined. For 10-bit resolution, bits 1 and 0 are undefined, and for 9-bit resolution bits 2, 1, and 0 are undefined. Table 1 gives examples of digital output data and the corresponding temperature reading for 12-bit resolution conversions.

Figure 2. Temperature Register Format

	BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
LS BYTE	2^3	2^2	2^1	2^0	2^{-1}	2^{-2}	2^{-3}	2^{-4}
	BIT 15	BIT 14	BIT 13	BIT 12	BIT 11	BIT 10	BIT 9	BIT 8
MS BYTE	S	S	S	S	S	2^6	2^5	2^4

S = SIGN

Table 1. Temperature/Data Relationship

TEMPERATURE (°C)	DIGITAL OUTPUT (BINARY)	DIGITAL OUTPUT (HEX)
+125	0000 0111 1101 0000	07D0h
+85*	0000 0101 0101 0000	0550h
+25.0625	0000 0001 1001 0001	0191h
+10.125	0000 0000 1010 0010	00A2h
+0.5	0000 0000 0000 1000	0008h
0	0000 0000 0000 0000	0000h
-0.5	1111 1111 1111 1000	FFF8h
-10.125	1111 1111 0101 1110	FF5Eh
-25.0625	1111 1110 0110 1111	FE6Fh
-55	1111 1100 1001 0000	FC90h

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

OPERATION—ALARM SIGNALING

After the DS18B20 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	2^4	2^3	2^2	2^1	2^0

Only bits 11 through 4 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 or equal to T_H, an alarm condition exists and an alarm flag is set inside the DS18B20. 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 DS18B20s on the bus by issuing an Alarm Search [ECh] command. Any DS18B20s with a set alarm flag will respond to the command, so the master can determine exactly which DS18B20s 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 DS18B20

The DS18B20 can be powered by an external supply on the V_{DD} pin, or it can operate in "parasite power" mode, which allows the DS18B20 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 DS18B20'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 DS18B20 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 DS18B20 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 DS18B20 for most operations as long as the specified timing and voltage requirements are met (see the *DC Electrical Characteristics* and *AC Electrical Characteristics*). However, when the DS18B20 is performing temperature conversions or copying data from the scratchpad memory to EEPROM, the operating current can be as high as 1.5mA. 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 DS18B20 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 10µs (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} = 10ms). No other activity can take place on the 1-Wire bus while the pullup is enabled.

The DS18B20 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 DS18B20 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 DS18B20 be powered by an external power supply.

In some situations the bus master may not know whether the DS18B20s 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 DS18B20s will pull the bus low, and externally powered DS18B20s 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.

Figure 4. Supplying the Parasite-Powered DS18B20 During Temperature Conversions

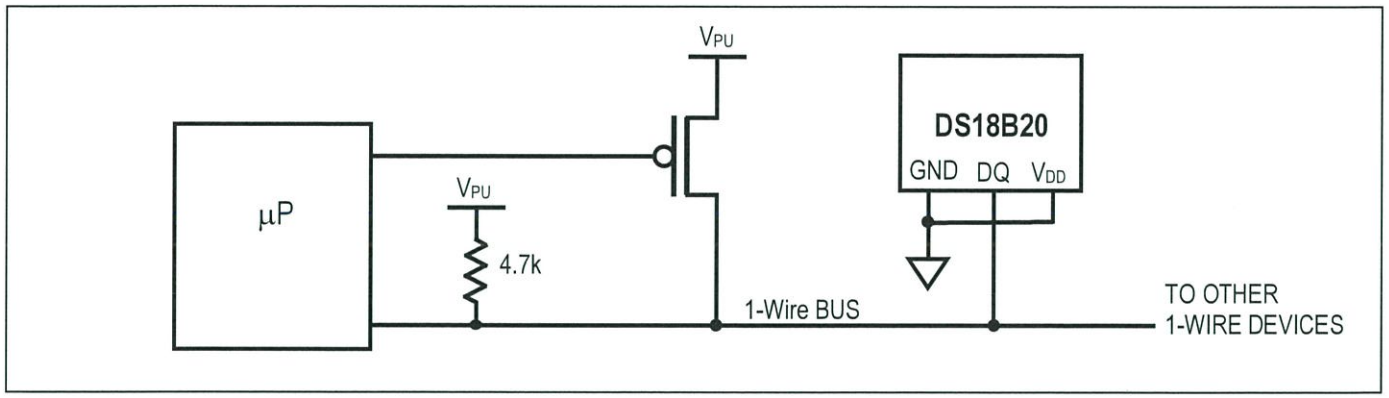
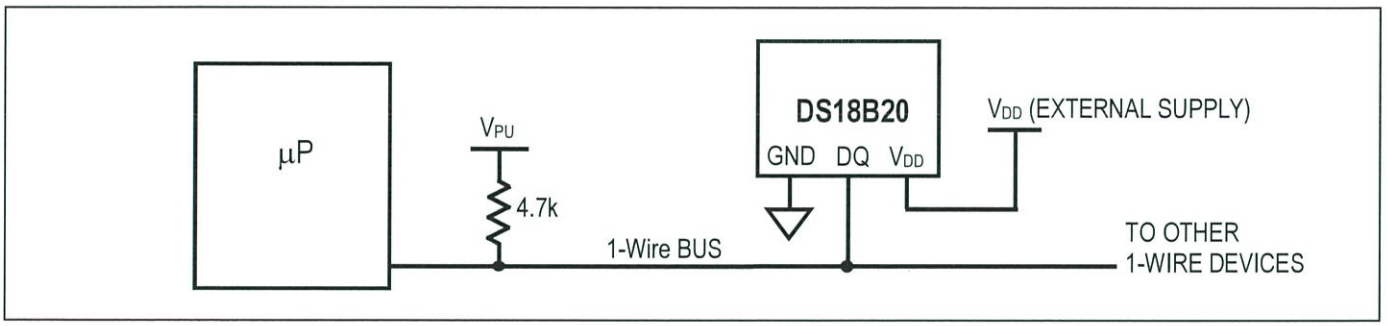


Figure 5. Powering the DS18B20 with an External Supply



64-BIT LASERED ROM CODE

Each DS18B20 contains a unique 64–bit code (see Figure 6) stored in ROM. The least significant 8 bits of the ROM code contain the DS18B20's 1-Wire family code: 28h. 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 DS18B20 to operate as a 1-Wire device using the protocol detailed in the *1-Wire Bus System* section.

Figure 6. 64-Bit Lasered ROM Code

8-BIT CRC		48-BIT SERIAL NUMBER		8-BIT FAMILY CODE (28h)	
MSB	LSB	MSB	LSB	MSB	LSB

MEMORY

The DS18B20'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) and configuration register. Note that if the DS18B20 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 *DS18B20 Function Commands* section.

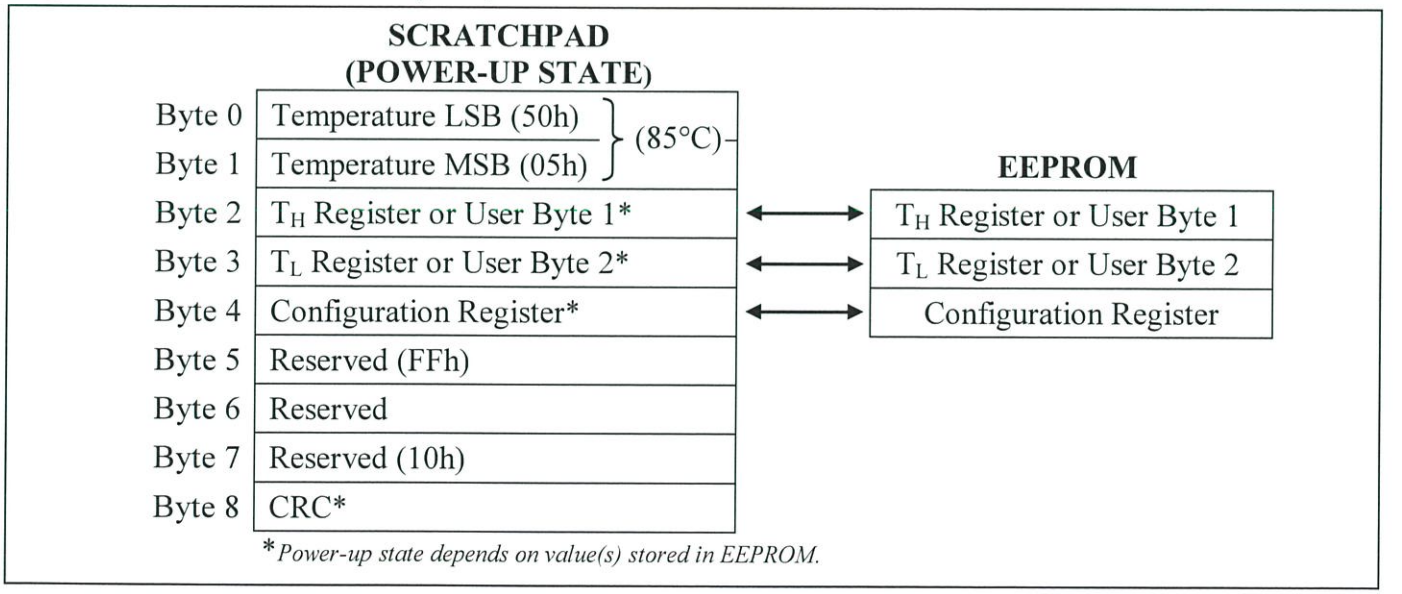
Byte 0 and byte 1 of the scratchpad contain the LSB and the MSB of the temperature register, respectively. These bytes are read-only. Bytes 2 and 3 provide access to T_H and T_L registers. Byte 4 contains the configuration register data, which is explained in detail in the *Configuration Register* section. Bytes 5, 6, and 7 are reserved for internal use by the device and cannot be overwritten.

Byte 8 of the scratchpad is read-only and contains the CRC code for bytes 0 through 7 of the scratchpad. The DS18B20 generates this CRC using the method described in the *CRC Generation* section.

Data is written to bytes 2, 3, and 4 of the scratchpad using the Write Scratchpad [4Eh] command; the data must be transmitted to the DS18B20 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, T_L and configuration 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 E^2 [B8h] command. The master can issue read time slots following the Recall E^2 command and the DS18B20 will indicate the status of the recall by transmitting 0 while the recall is in progress and 1 when the recall is done.

Figure 7. DS18B20 Memory Map



CONFIGURATION REGISTER

Byte 4 of the scratchpad memory contains the configuration register, which is organized as illustrated in Figure 8. The user can set the conversion resolution of the DS18B20 using the R0 and R1 bits in this register as shown in Table 2. The power-up default of these bits is R0 = 1 and R1 = 1 (12-bit resolution). Note that there is a direct tradeoff between resolution and conversion time. Bit 7 and bits 0 to 4 in the configuration register are reserved for internal use by the device and cannot be overwritten.

Figure 8. Configuration Register

BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
0	R1	R0	1	1	1	1	1

Table 2. Thermometer Resolution Configuration

R1	R0	RESOLUTION (BITS)	MAX CONVERSION TIME	
0	0	9	93.75ms	($t_{CONV}/8$)
0	1	10	187.5ms	($t_{CONV}/4$)
1	0	11	375ms	($t_{CONV}/2$)
1	1	12	750ms	(t_{CONV})

CRC GENERATION

CRC bytes are provided as part of the DS18B20'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 DS18B20. 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 DS18B20 that prevents a command sequence from proceeding if the DS18B20 CRC (ROM or scratchpad) does not match the value generated by the bus master.

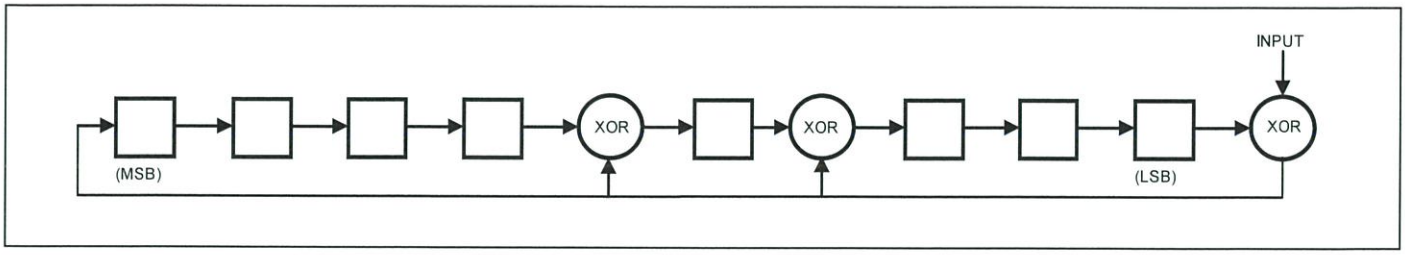
The equivalent polynomial function of the CRC (ROM or scratchpad) is:

$$CRC = X^8 + X^5 + X^4 + 1$$

The bus master can re-calculate the CRC and compare it to the CRC values from the DS18B20 using the polynomial generator shown in Figure 9. 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 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 DS18B20 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 Maxim 1-Wire cyclic redundancy check

is available in *Application Note 27: Understanding and Using Cyclic Redundancy Checks with Maxim iButton Products*.

Figure 9. CRC Generator



1-WIRE BUS SYSTEM

The 1-Wire bus system uses a single bus master to control one or more slave devices. The DS18B20 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 "multidrop" if there are multiple slaves on the bus.

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

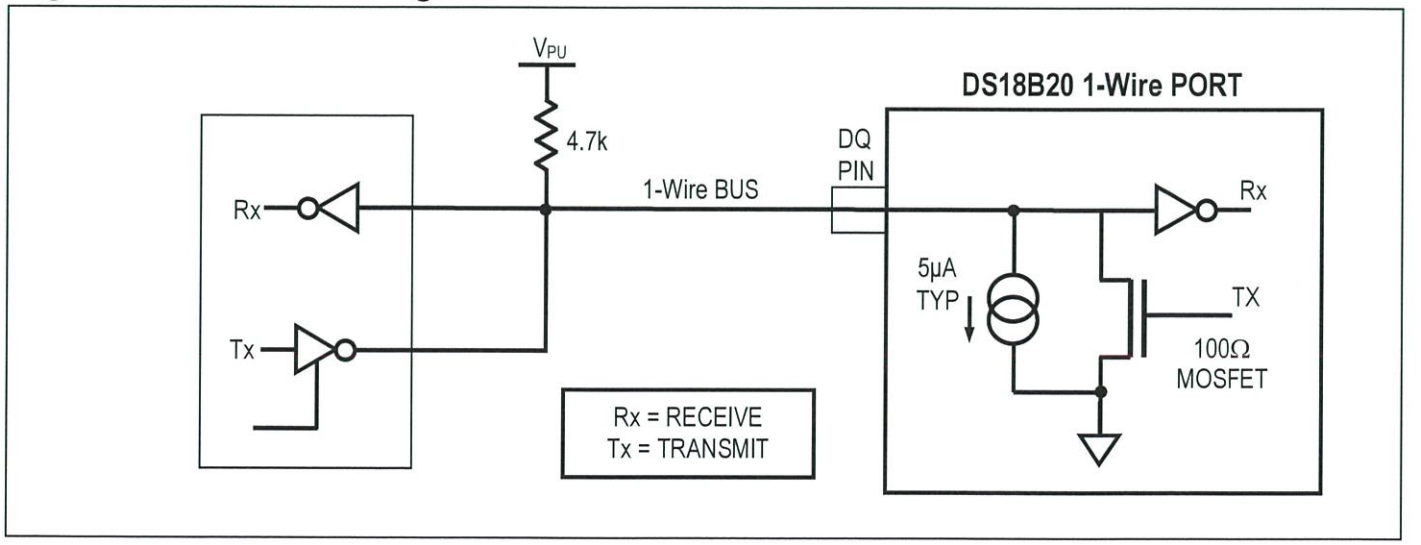
The following discussion of the 1-Wire bus system is broken down into three topics: hardware configuration, transaction sequence, and 1-Wire signaling (signal types and timing).

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 DS18B20 (the DQ pin) is open drain with an internal circuit equivalent to that shown in Figure 10.

The 1-Wire bus requires an external pullup resistor of approximately 5kΩ; 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 480µs, all components on the bus will be reset.

Figure 10. Hardware Configuration



TRANSACTION SEQUENCE

The transaction sequence for accessing the DS18B20 is as follows:

Step 1. Initialization

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

Step 3. DS18B20 Function Command (followed by any required data exchange)

It is very important to follow this sequence every time the DS18B20 is accessed, as the DS18B20 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 DS18B20) 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 DS18B20 function command. A flowchart for operation of the ROM commands is shown in Figure 11.

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.maxim-ic.com/ibuttonbook. 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 multidrop 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.

iButton is a registered trademark of Maxim Integrated Products, Inc.

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 DS18B20s on the bus perform simultaneous temperature conversions by issuing a Skip ROM command followed by a Convert T [44h] command.

Note that the Read Scratchpad [BEh] command can follow the Skip ROM command only if there is a single slave device on the bus. In this case, time is saved by allowing the master to read from the slave without sending the device's 64-bit ROM code. A Skip ROM command followed by a Read Scratchpad command 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 DS18B20s 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. See the *Operation—Alarm Signaling* section for an explanation of alarm flag operation.

DS18B20 FUNCTION COMMANDS

After the bus master has used a ROM command to address the DS18B20 with which it wishes to communicate, the master can issue one of the DS18B20 function commands. These commands allow the master to write to and read from the DS18B20's scratchpad memory, initiate temperature conversions and determine the power supply mode. The DS18B20 function commands, which are described below, are summarized in Table 3 and illustrated by the flowchart in Figure 12.

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 DS18B20 returns to its low-power idle state. If the device is being used in parasite power mode, within 10μs (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 DS18B20* section. If the DS18B20 is powered by an external supply, the master can issue read time slots after the Convert T command and the DS18B20 will respond by transmitting a 0 while the temperature conversion is in progress and a 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 3 bytes of data to the DS18B20's scratchpad. The first data byte is written into the T_H register (byte 2 of the scratchpad), the second byte is written into the T_L register (byte 3), and the third byte is written into the configuration register (byte 4). Data must be transmitted least significant bit first. All three 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, T_L and configuration registers (bytes 2, 3 and 4) to EEPROM. If the device is being used in parasite power mode, within 10µs (max) after this command is issued the master must enable a strong pullup on the 1-Wire bus for at least 10ms as described in the *Powering the DS18B20* section.

RECALL E^2 [B8h]

This command recalls the alarm trigger values (T_H and T_L) and configuration data from EEPROM and places the data in bytes 2, 3, and 4, respectively, in the scratchpad memory. The master device can issue read time slots following the Recall E^2 command and the DS18B20 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 DS18B20s on the bus are using parasite power. During the read time slot, parasite powered DS18B20s will pull the bus low, and externally powered DS18B20s will let the bus remain high. See the *Powering the DS18B20* section for usage information for this command.

Table 3. DS18B20 Function Command Set

COMMAND	DESCRIPTION	PROTOCOL	1-Wire BUS ACTIVITYAFTER COMMAND IS ISSUED	NOTES
TEMPERATURE CONVERSION COMMANDS				
Convert T	Initiates temperature conversion.	44h	DS18B20 transmits conversion status to master (not applicable for parasite-powered DS18B20s).	1
MEMORY COMMANDS				
Read Scratchpad	Reads the entire scratchpad including the CRC byte.	BEh	DS18B20 transmits up to 9 data bytes to master.	2
Write Scratchpad	Writes data into scratchpad bytes 2, 3, and 4 (T_H, T_L, and configuration registers).	4Eh	Master transmits 3 data bytes to DS18B20.	3
Copy Scratchpad	Copies T_H, T_L, and configuration register data from the scratchpad to EEPROM.	48h	None	1
Recall E^2	Recalls T_H, T_L, and configuration register data from EEPROM to the scratchpad.	B8h	DS18B20 transmits recall status to master.	
Read Power Supply	Signals DS18B20 power supply mode to the master.	B4h	DS18B20 transmits supply status to master.	

Note 1: For parasite-powered DS18B20s, 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.
Note 2: The master can interrupt the transmission of data at any time by issuing a reset.
Note 3: All three bytes must be written before a reset is issued.

Figure 11. ROM Commands Flowchart

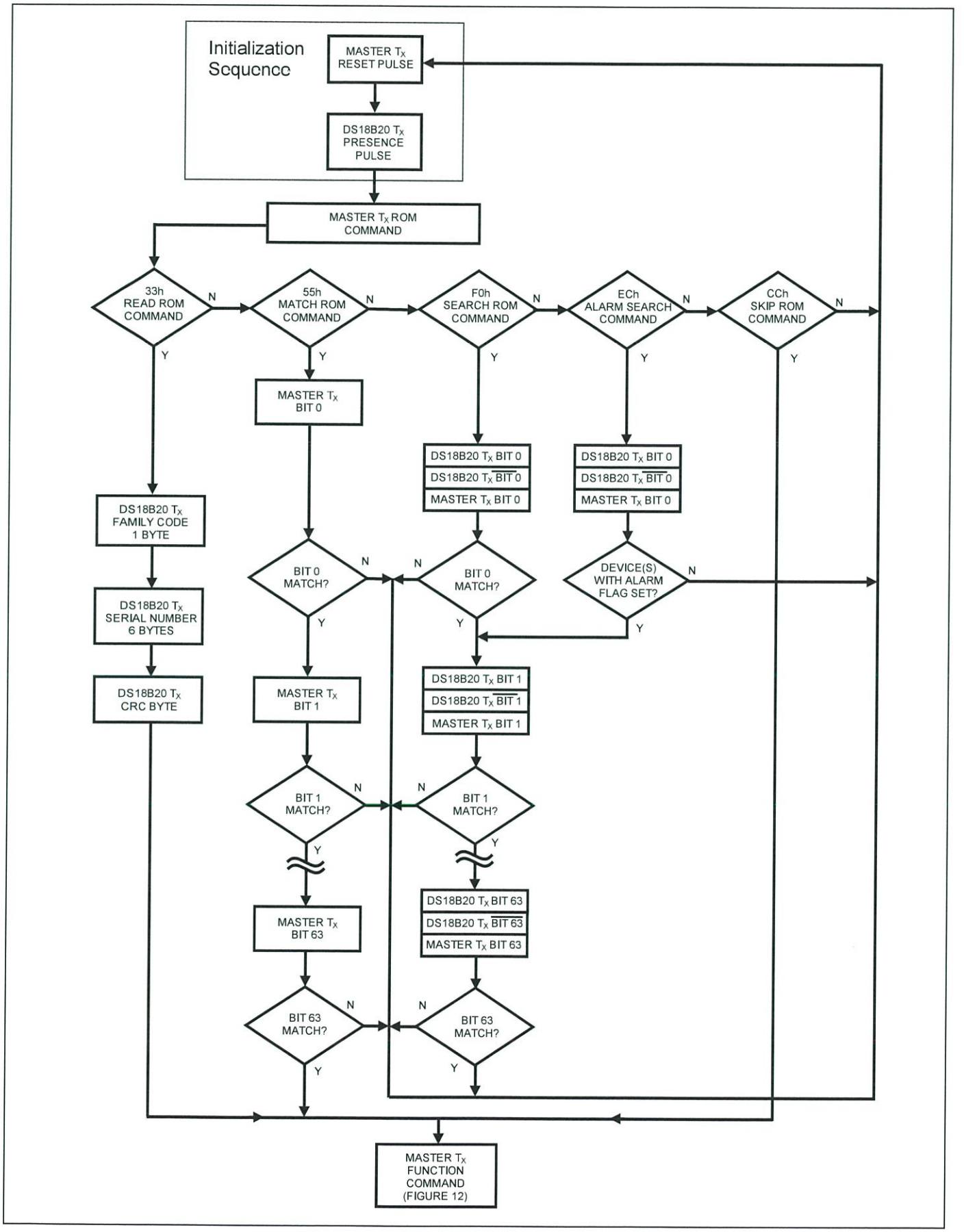
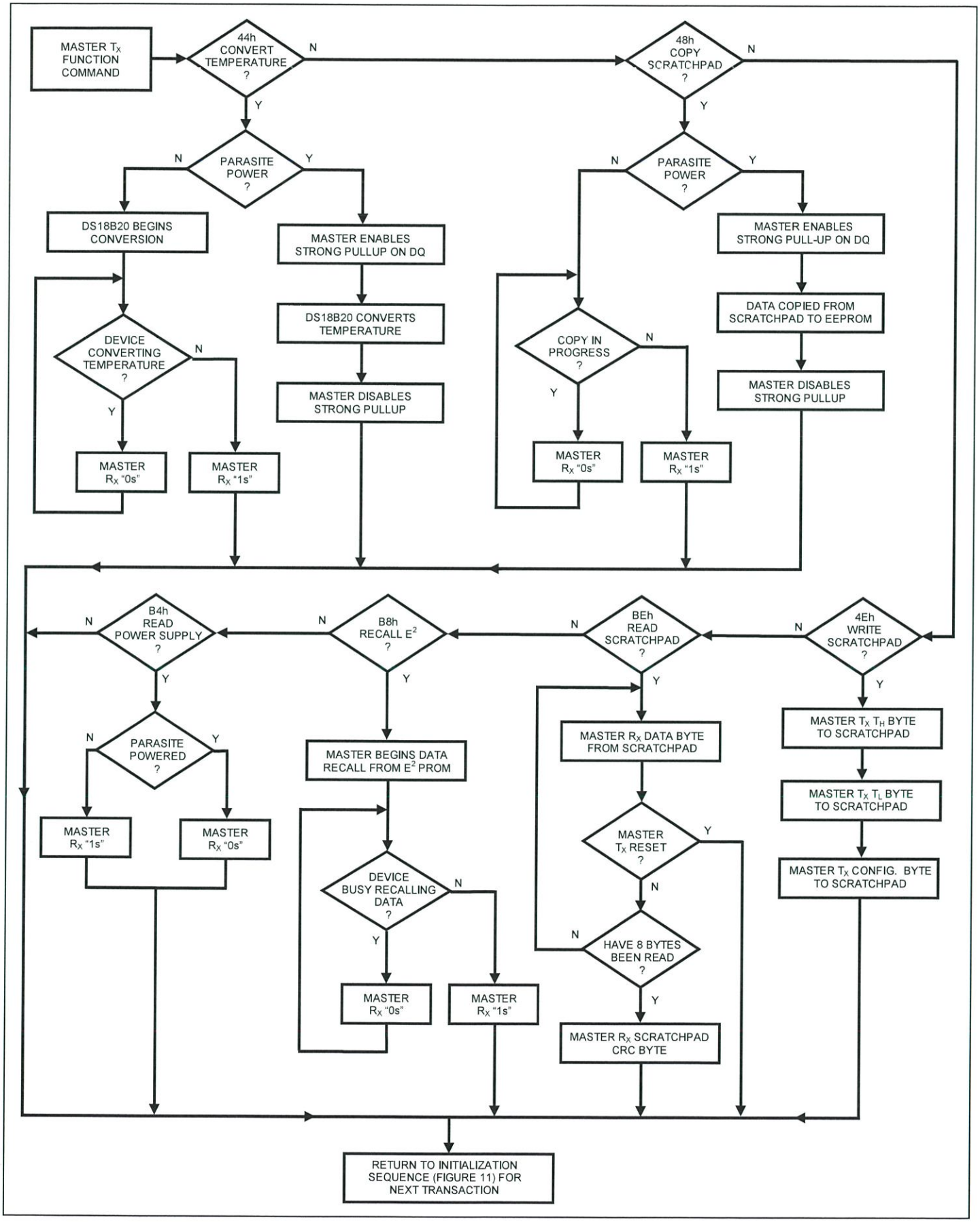


Figure 12. DS18B20 Function Commands Flowchart



1-WIRE SIGNALING

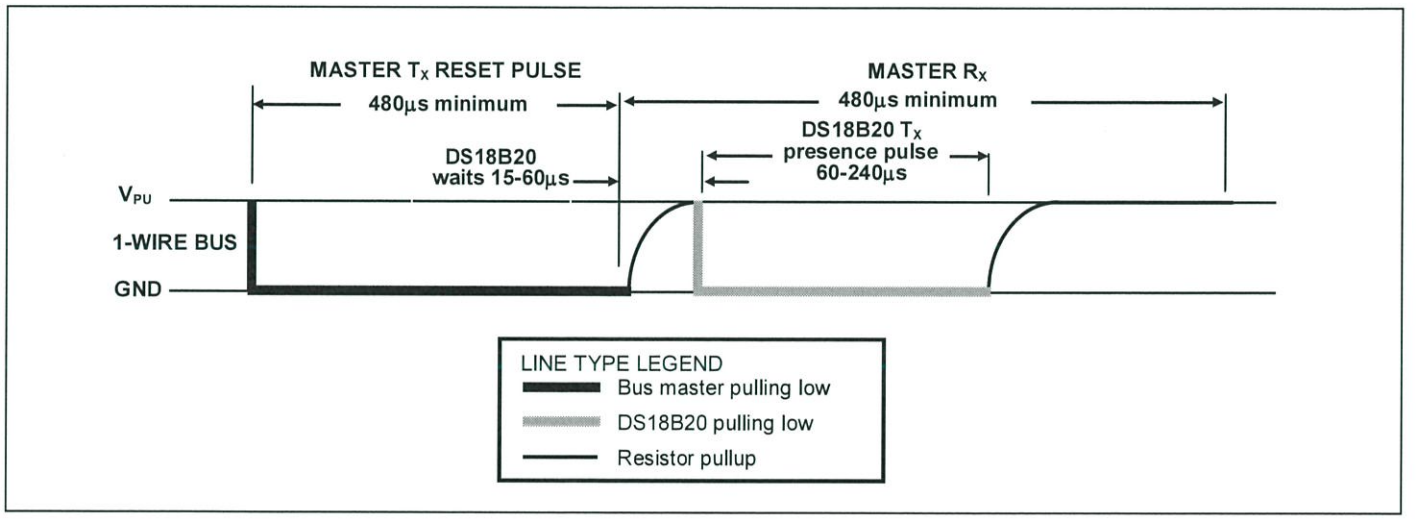
The DS18B20 uses a strict 1-Wire communication protocol to ensure data integrity. Several signal types are defined by this protocol: reset pulse, presence pulse, write 0, write 1, read 0, and read 1. The bus master initiates all these signals, with the exception of the presence pulse.

INITIALIZATION PROCEDURE—RESET AND PRESENCE PULSES

All communication with the DS18B20 begins with an initialization sequence that consists of a reset pulse from the master followed by a presence pulse from the DS18B20. This is illustrated in Figure 13. When the DS18B20 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 480µs. The bus master then releases the bus and goes into receive mode (R_X). When the bus is released, the 5kΩ pullup resistor pulls the 1-Wire bus high. When the DS18B20 detects this rising edge, it waits 15µs to 60µs and then transmits a presence pulse by pulling the 1-Wire bus low for 60µs to 240µs.

Figure 13. Initialization Timing



READ/WRITE TIME SLOTS

The bus master writes data to the DS18B20 during write time slots and reads data from the DS18B20 during read time slots. One bit of data is transmitted over the 1-Wire bus per time slot.

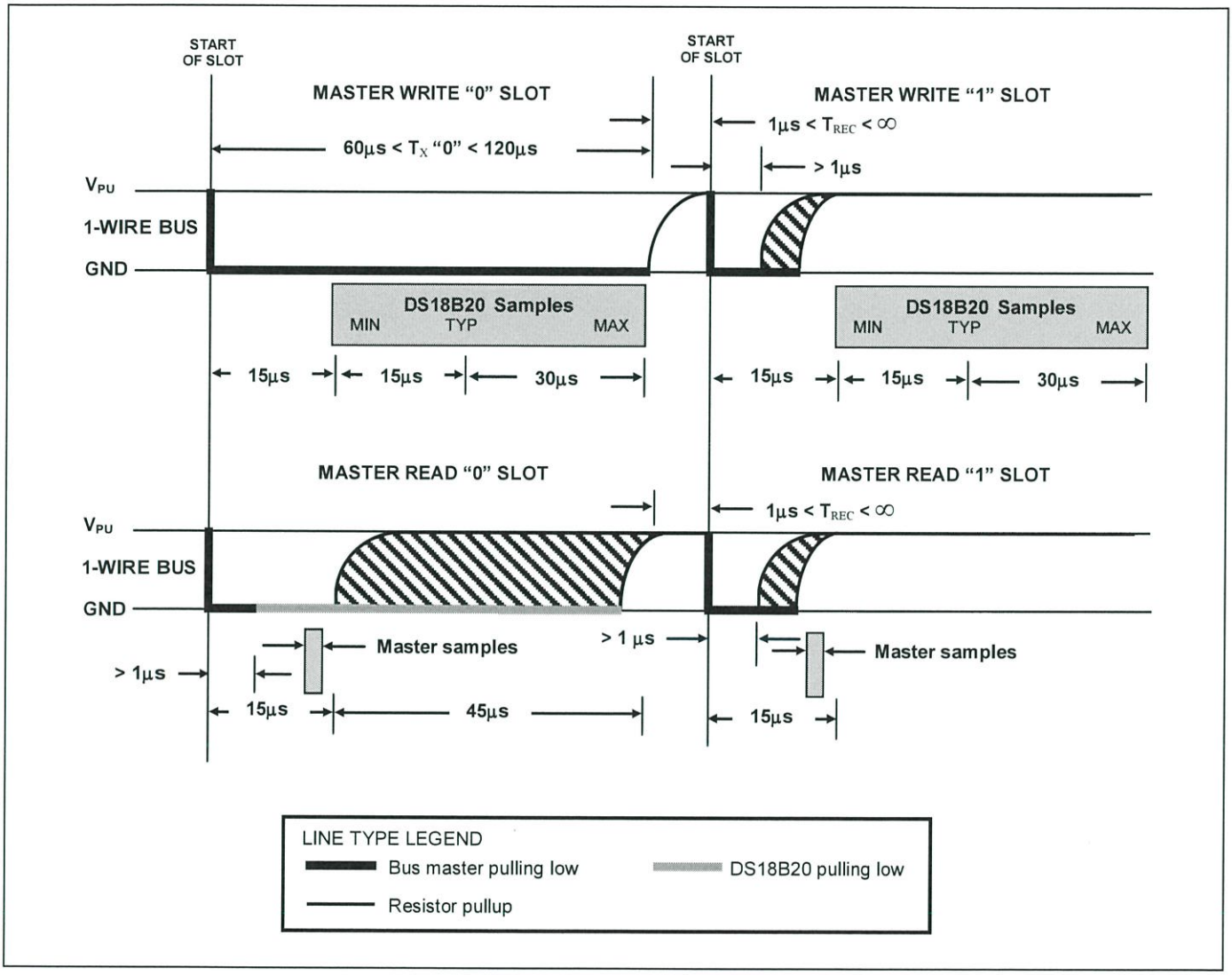
WRITE TIME SLOTS

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 DS18B20 and a Write 0 time slot to write a logic 0 to the DS18B20. All write time slots must be a minimum of 60µs in duration with a minimum of a 1µs 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 14).

To generate a Write 1 time slot, after pulling the 1-Wire bus low, the bus master must release the 1-Wire bus within 15µs. When the bus is released, the 5kΩ 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 60µs).

The DS18B20 samples the 1-Wire bus during a window that lasts from 15µs to 60µs after the master initiates the write time slot. If the bus is high during the sampling window, a 1 is written to the DS18B20. If the line is low, a 0 is written to the DS18B20.

Figure 14. Read/Write TIme Slot TImIng DIagram



READ TIME SLOTS

The DS18B20 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 DS18B20 can provide the requested data. In addition, the master can generate read time slots after issuing Convert T [44h] or Recall E^2 [B8h] commands to find out the status of the operation as explained in the *DS18B20 Function Commands* section.

All read time slots must be a minimum of 60µs in duration with a minimum of a 1µs recovery time between slots. A read time slot is initiated by the master device pulling the 1-Wire bus low for a minimum of 1µs and then releasing the bus (see Figure 14). After the master initiates the read time slot, the DS18B20 will begin transmitting a 1 or 0 on bus. The DS18B20 transmits a 1 by leaving the bus high and transmits a 0 by pulling the bus low. When transmitting a 0, the DS18B20 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 resister. Output

data from the DS18B20 is valid for 15µs after the falling edge that initiated the read time slot. Therefore, the master must release the bus and then sample the bus state within 15µs from the start of the slot.

Figure 15 illustrates that the sum of T_{INIT}, T_{RC}, and T_{SAMPLE} must be less than 15µs for a read time slot. Figure 16 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 15µs period.

Figure 15. Detailed Master Read 1 Timing

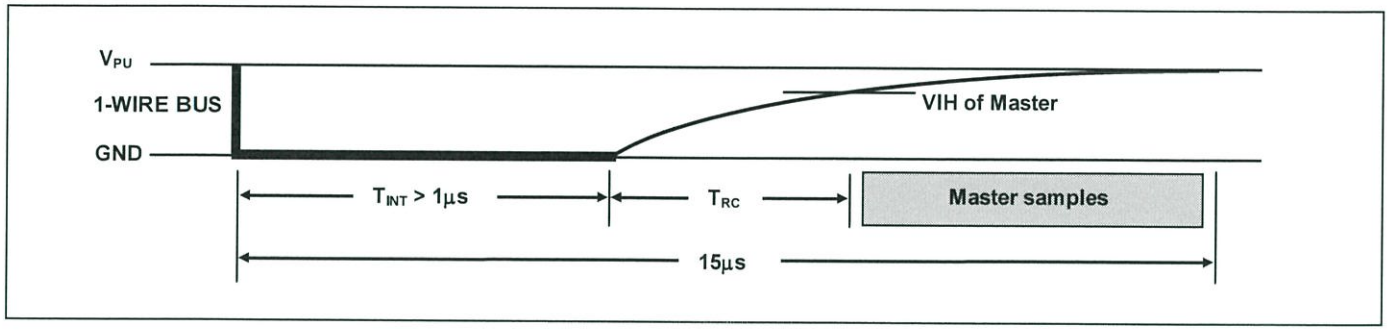
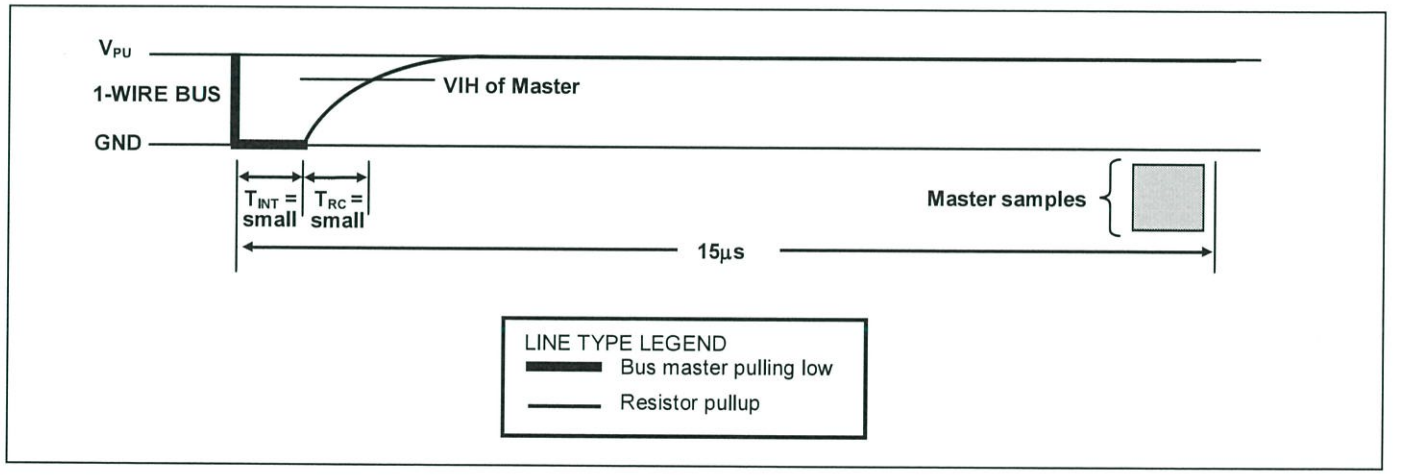


Figure 16. Recommended Master Read 1 Timing



RELATED APPLICATION NOTES

The following application notes can be applied to the DS18B20 and are available on our website at www.maxim-ic.com.

Application Note 27: Understanding and Using Cyclic Redundancy Checks with Maxim iButton Products
Application Note 122: Using Dallas' 1-Wire ICs in 1-Cell Li-Ion Battery Packs with Low-Side N-Channel Safety FETs Master
Application Note 126: 1-Wire Communication Through Software
Application Note 162: Interfacing the DS18x20/DS1822 1-Wire Temperature Sensor in a Microcontroller Environment
Application Note 208: Curve Fitting the Error of a Bandgap-Based Digital Temperature Sensor
Application Note 2420: 1-Wire Communication with a Microchip PICmicro Microcontroller
Application Note 3754: Single-Wire Serial Bus Carries Isolated Power and Data

Sample 1-Wire subroutines that can be used in conjunction with *Application Note 74: Reading and Writing iButtons via Serial Interfaces* can be downloaded from the Maxim website.

DS18B20 OPERATION EXAMPLE 1

In this example there are multiple DS18B20s on the bus and they are using parasite power. The bus master initiates a temperature conversion in a specific DS18B20 and then reads its scratchpad and recalculates the CRC to verify the data.

MASTER MODE	DATA (LSB FIRST)	COMMENTS
Tx	Reset	Master issues reset pulse.
Rx	Presence	DS18B20s respond with presence pulse.
Tx	55h	Master issues Match ROM command.
Tx	64-bit ROM code	Master sends DS18B20 ROM code.
Tx	44h	Master issues Convert T command.
Tx	DQ line held high by strong pullup	Master applies strong pullup to DQ for the duration of the conversion (t_{CONV}).
Tx	Reset	Master issues reset pulse.
Rx	Presence	DS18B20s respond with presence pulse.
Tx	55h	Master issues Match ROM command.
Tx	64-bit ROM code	Master sends DS18B20 ROM code.
Tx	BEh	Master issues Read Scratchpad command.
Rx	9 data bytes	Master reads entire scratchpad including CRC. The master then recalculates the CRC of the first eight data bytes from the scratchpad and compares the calculated CRC with the read CRC (byte 9). If they match, the master continues; if not, the read operation is repeated.

DS18B20 OPERATION EXAMPLE 2

In this example there is only one DS18B20 on the bus and it is using parasite power. The master writes to the T_H, T_L, and configuration registers in the DS18B20 scratchpad and then reads the scratchpad and recalculates the CRC to verify the data. The master then copies the scratchpad contents to EEPROM.

MASTER MODE	DATA (LSB FIRST)	COMMENTS
Tx	Reset	Master issues reset pulse.
Rx	Presence	DS18B20 responds with presence pulse.
Tx	CCh	Master issues Skip ROM command.
Tx	4Eh	Master issues Write Scratchpad command.
Tx	3 data bytes	Master sends three data bytes to scratchpad (T_H, T_L, and config).
Tx	Reset	Master issues reset pulse.
Rx	Presence	DS18B20 responds with presence pulse.
Tx	CCh	Master issues Skip ROM command.
Tx	BEh	Master issues Read Scratchpad command.
Rx	9 data bytes	Master reads entire scratchpad including CRC. The master then recalculates the CRC of the first eight data bytes from the scratchpad and compares the calculated CRC with the read CRC (byte 9). If they match, the master continues; if not, the read operation is repeated.
Tx	Reset	Master issues reset pulse.
Rx	Presence	DS18B20 responds with presence pulse.
Tx	CCh	Master issues Skip ROM command.
Tx	48h	Master issues Copy Scratchpad command.
Tx	DQ line held high by strong pullup	Master applies strong pullup to DQ for at least 10ms while copy operation is in progress.

ABSOLUTE MAXIMUM RATINGS

Voltage Range on Any Pin Relative to Ground .....-0.5V to +6.0V
Operating Temperature Range .....-55°C to +125°C
Storage Temperature Range .....-55°C to +125°C
Solder Temperature .....Refer to the IPC/JEDEC J-STD-020 Specification.

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

DC ELECTRICAL CHARACTERISTICS (-55°C to +125°C; V_{DD}=3.0V to 5.5V)

PARAMETER	SYMBOL	CONDITIONS	MIN	TYP	MAX	UNITS	NOTES
Supply Voltage	V_{DD}	Local Power	+3.0		+5.5	V	1
Pullup Supply Voltage	V_{PU}	Parasite Power	+3.0		+5.5	V	1,2
		Local Power	+3.0		V_{DD}		
Thermometer Error	t_{ERR}	-10°C to +85°C			±0.5	°C	3
		-55°C to +125°C			±2		
Input Logic-Low	V_{IL}		-0.3		+0.8	V	1,4,5
Input Logic-High	V_{IH}	Local Power	+2.2		The lower of 5.5 or V_{DD} + 0.3	V	1, 6
		Parasite Power	+3.0				
Sink Current	I_L	$V_{I/O}$ = 0.4V	4.0			mA	1
Standby Current	I_{DDS}			750	1000	nA	7,8
Active Current	I_{DD}	V_{DD} = 5V		1	1.5	mA	9
DQ Input Current	I_{DQ}			5		µA	10
Drift				±0.2		°C	11

NOTES:

1) All voltages are referenced to ground.
2) The Pullup Supply Voltage specification assumes that the pullup device is ideal, and therefore the high level of the pullup is equal to V_{PU}. In order to meet the V_{IH} spec of the DS18B20, the actual supply rail for the strong pullup transistor must include margin for the voltage drop across the transistor when it is turned on; thus: $V_{PU_ACTUAL} = V_{PU_IDEAL} + V_{TRANSISTOR}$.
3) See typical performance curve in Figure 17.
4) Logic-low voltages are specified at a sink current of 4mA.
5) To guarantee a presence pulse under low voltage parasite power conditions, V_{ILMAX} may have to be reduced to as low as 0.5V.
6) Logic-high voltages are specified at a source current of 1mA.
7) Standby current specified up to +70°C. Standby current typically is 3µA at +125°C.
8) To minimize I_{DDS}, DQ should be within the following ranges: GND ≤ DQ ≤ GND + 0.3V or V_{DD} – 0.3V ≤ DQ ≤ V_{DD}.
9) Active current refers to supply current during active temperature conversions or EEPROM writes.
10) DQ line is high ("high-Z" state).
11) Drift data is based on a 1000-hour stress test at +125°C with V_{DD} = 5.5V.

AC ELECTRICAL CHARACTERISTICS—NV MEMORY

(-55°C to +100°C; V_{DD} = 3.0V to 5.5V)

PARAMETER	SYMBOL	CONDITIONS	MIN	TYP	MAX	UNITS
NV Write Cycle Time	t_{WR}			2	10	ms
EEPROM Writes	N_{EEWR}	-55°C to +55°C	50k			writes
EEPROM Data Retention	t_{EEDR}	-55°C to +55°C	10			years

AC ELECTRICAL CHARACTERISTICS (-55°C to +125°C; V_{DD} = 3.0V to 5.5V)

PARAMETER	SYMBOL	CONDITIONS	MIN	TYP	MAX	UNITS	NOTES
Temperature Conversion Time	t_{CONV}	9-bit resolution			93.75	ms	1
		10-bit resolution			187.5		
		11-bit resolution			375		
		12-bit resolution			750		
Time to Strong Pullup On	t_{SPON}	Start Convert T Command Issued			10	µs	
Time Slot	t_{SLOT}		60		120	µs	1
Recovery Time	t_{REC}		1			µs	1
Write 0 Low Time	t_{LOW0}		60		120	µs	1
Write 1 Low Time	t_{LOW1}		1		15	µs	1
Read Data Valid	t_{RDV}				15	µs	1
Reset Time High	t_{RSTH}		480			µs	1
Reset Time Low	t_{RSTL}		480			µs	1,2
Presence-Detect High	t_{PDHIGH}		15		60	µs	1
Presence-Detect Low	t_{PDLOW}		60		240	µs	1
Capacitance	$C_{IN/OUT}$				25	pF	

NOTES:

1) See the timing diagrams in Figure 18.
2) Under parasite power, if t_{RSTL} > 960µs, a power-on reset may occur.

Figure 17. Typical Performance Curve

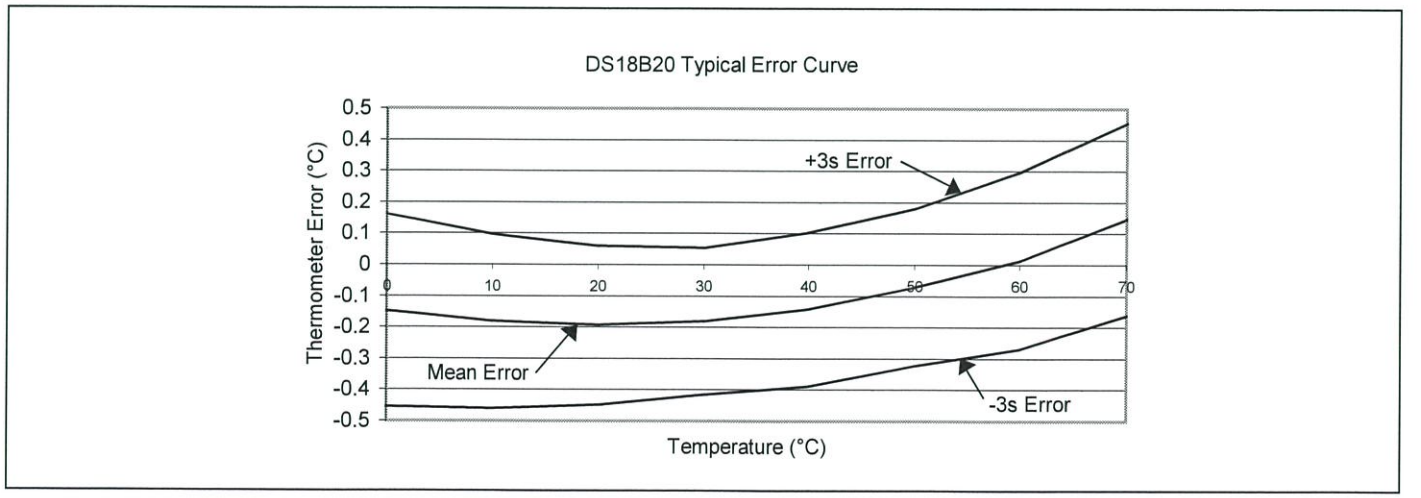
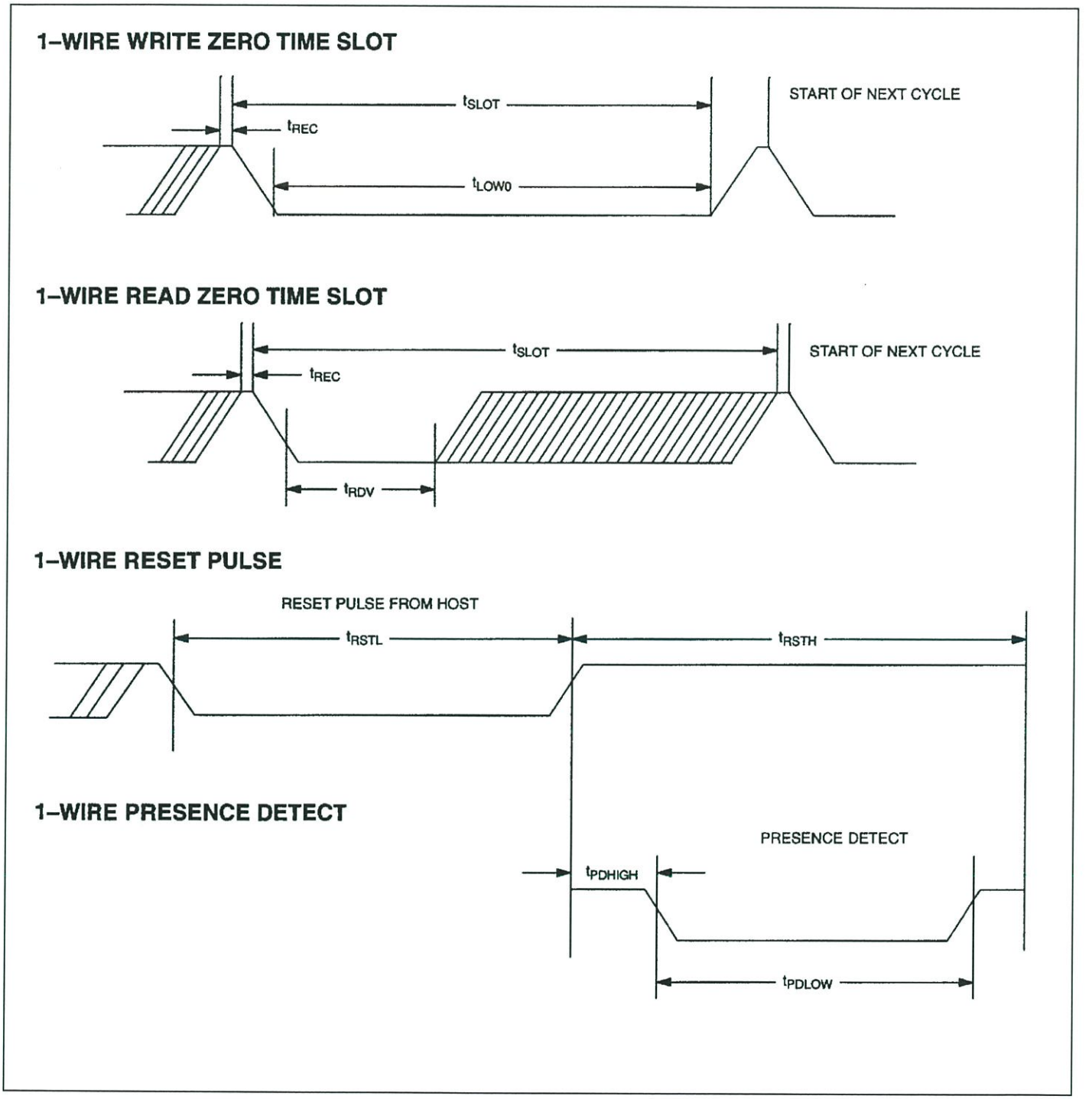


Figure 18. Timing Diagrams



REVISION HISTORY

REVISION DATE	DESCRIPTION	PAGES CHANGED
030107	In the *Absolute Maximum Ratings* section, removed the reflow oven temperature value of +220°C. Reference to JEDEC specification for reflow remains.	19
101207	In the *Operation—Alarm Signaling* section, added "or equal to" in the desciption for a TH alarm condition	5
	In the *Memory* section, removed incorrect text describing memory.	7
	In the *Configuration Register* section, removed incorrect text describing configuration register.	8
042208	In the *Ordering Information* table, added TO-92 straight-lead packages and included a note that the TO-92 package in tape and reel can be ordered with either formed or straight leads.	2

maxim integrated™

Maxim cannot assume responsibility for use of any circuitry other than circuitry entirely embodied in a Maxim product. No circuit patent licenses are implied. Maxim reserves the right to change the circuitry and specifications without notice at any time. The parametric values (min and max limits) shown in the Electrical Characteristics table are guaranteed. Other parametric values quoted in this data sheet are provided for guidance.

© 2008 Maxim Integrated The Maxim logo and Maxim Integrated are trademarks of Maxim Integrated Products, Inc.

SPECIFICATION

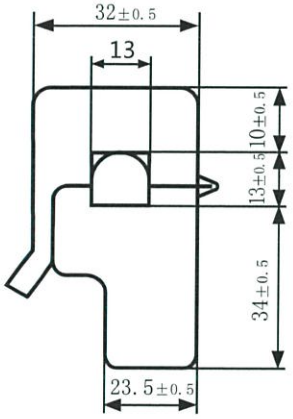
Customer Title : XiDi Technology

Product Name: Splilt-core current transformer

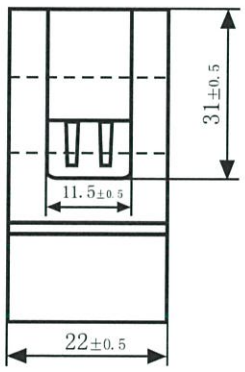
Manufacture Model : SCT-013-030

Charateristics: open size:13mm×13mm
1m leading wire
Core material:Ferrite
Fire resistance property:in accordance with UL 94-V0
Dielectric strength: 1500V AC/1min 5mA (between shell and output)

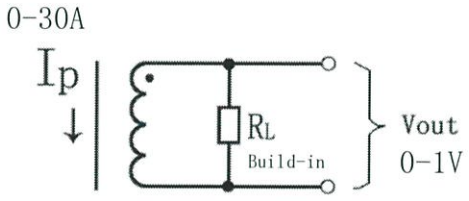
Outline size diagram:(in mm)



Front View



Side View



Voltage output type

Schematic Diagram

Typical table of technical parameters:

input current	output voltage	non-linearity	build-in sampling resistance (R_L)
0-30A	0-1V	±1%	62Ω
turn ratio	resistance grade	work temperature	dielectric strength(between shell and output)
1800:1	Grade B	-25℃～+70℃	1500V AC/1min 5mA

Customer Sign:

Beijing YaoHuadechang Electronic Co.,Ltd
Phone: 0355-7929499-803
Cell: 13693334514
Contact Name: Engineer Chen

Approve Sign: Chenjianping

2011-7-21