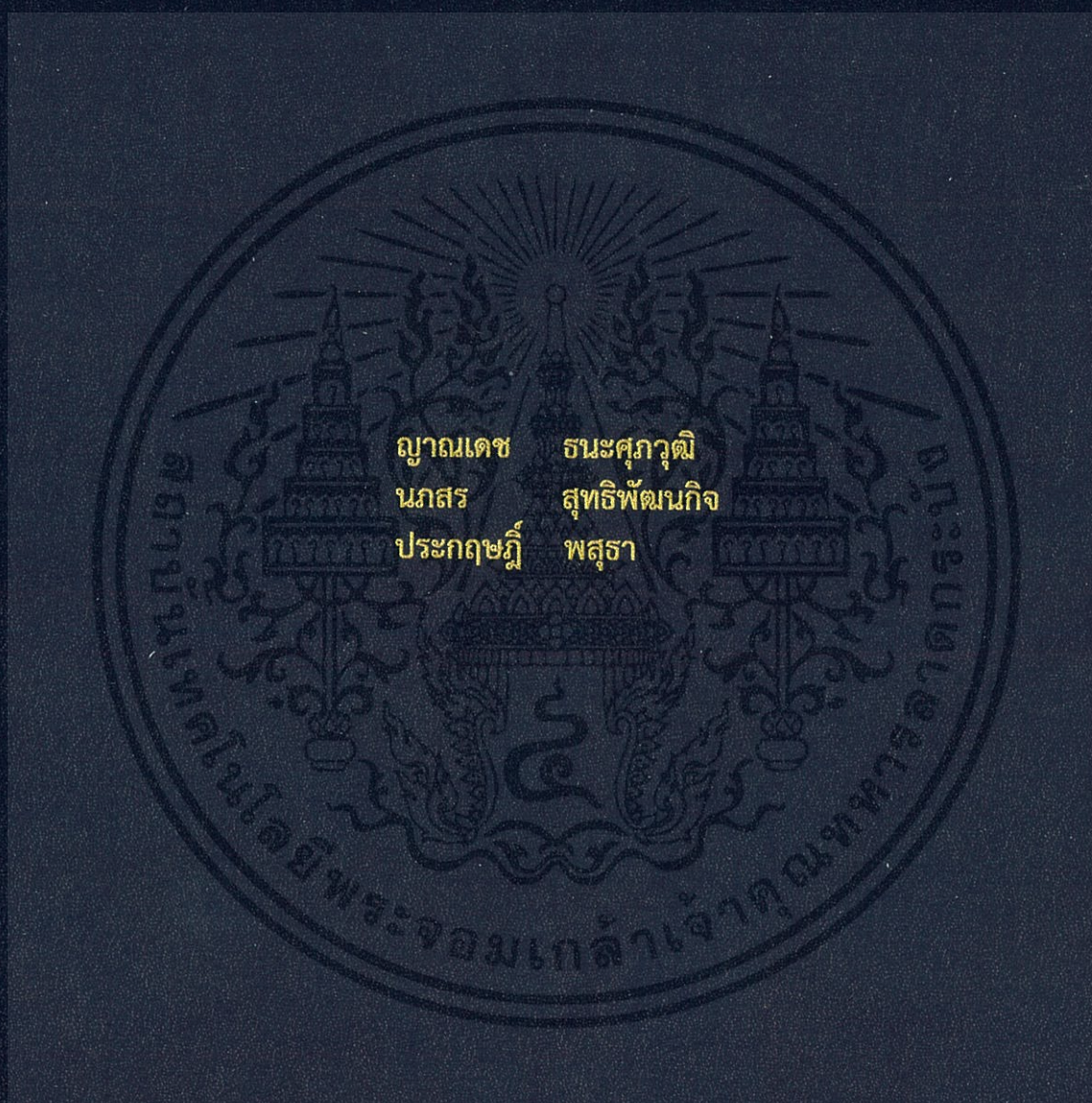


ตู้ควบคุมอุณหภูมิที่สามารถปรับค่าได้
PROGRAMMABLE TEMPERATURE CONTROL CHAMBER



ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต
สาขาวิชาวิศวกรรมระบบควบคุม
คณะวิศวกรรมศาสตร์
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
ปีการศึกษา 2559

ตู้ควบคุมอุณหภูมิที่สามารถปรับค่าได้
PROGRAMMABLE TEMPERATURE CONTROL CHAMBER



ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต

สาขาวิชาวิศวกรรมระบบควบคุม

คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2559

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

PROGRAMMABLE TEMPERATURE CONTROL CHAMBER



THIS THESIS IS SUBMITTED IN PARTIAL FULFILLMENT
OF THE REQUIREMENTS FOR THE DEGREE OF
BECHELOR OF ENGINEERING IN CONTROL ENGINEERING
FACULTY OF ENGINEERING
KING MONKUT'S INSTITUTE OF TECHNOLOGY LADKRABANG
ACADEMIC YEAR 2016

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ปริญญาานิพนธ์ปีการศึกษา 2559

ภาควิชาวิศวกรรมการวัดและควบคุม คณะวิศวกรรมศาสตร์
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เรื่อง ตู้ควบคุมอุณหภูมิที่สามารถปรับค่าได้
PROGRAMMABLE TEMPERATURE CONTROL CHAMBER

ผู้จัดทำ นายญาณเดช ธนะศุภวุฒิ 56010325
 นางสาวนภสร สุทธิพัฒน์กิจ 56010643
 นายประภฤษฎ์ พสุธา 56010716

.....อาจารย์ที่ปรึกษา
(ผู้ช่วยศาสตราจารย์ ดร.วรรณดี เพชรมณีล้ำค่า)

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ผู้ควบคุมอุณหภูมิที่สามารถปรับค่าได้

โดย

นายญาณเดช ณะศุภวุฒิ 56010325

นางสาวนภสร สุทธิพัฒน์กิจ 56010643

นายประภษุณี พสุธา 56010716

อาจารย์ที่ปรึกษา

ผู้ช่วยศาสตราจารย์ ดร.วรรณดี เพชรรมณีล้ำค่า

ปีการศึกษา 2559

บทคัดย่อ

ปริญญานิพนธ์ฉบับนี้นำเสนอการออกแบบและสร้างผู้ควบคุมอุณหภูมิที่สามารถโปรแกรมได้ โดยการควบคุมอุณหภูมิภายในตู้ใช้การควบคุมแบบ PID (Proportional-Integral-Derivative) เพื่อให้ระบบมีเสถียรภาพและเข้าสู่ค่าเป้าหมายได้อย่างรวดเร็ว ผู้ควบคุมอุณหภูมิที่สร้างขึ้นประกอบด้วย โครงสร้างไม้อัดพ่นสีน้ำเงิน ขดลวดความร้อนแบบครีป พัดลมกวนอากาศ พัดลมระบายอากาศ และวงจรรีเลย์ทรานซิสต์ที่ใช้ในการควบคุมอุณหภูมิ โดยการควบคุมขดลวดความร้อนจะใช้การควบคุมแบบ PID ส่วนการควบคุมพัดลมระบายอากาศจะใช้การควบคุมแบบเปิดปิด ผู้ใช้งานสามารถควบคุมการทำงานได้โดยการตั้งค่าเป้าหมายและการตั้งค่าพารามิเตอร์ของ PID ตัวผู้สามารถแสดงค่าอุณหภูมิปัจจุบันภายในตู้ผ่านหน้าจอแอลซีดี

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

PROGRAMMABLE TEMPERATURE CONTROL CHAMBER

By

Mr.Yannadej Thanasupawut 56010325

Ms.Napasorn Suttipattanakit 56010643

Mr.Pragrid Phasutha 56010716

Advisor

Asst.Prof. Dr. Wandee Petchmaneelumka

Academic Year 2016

ABSTRACT

This project presents a design and realization of programmable temperature control chamber with using PID (Proportional-Integral-Derivative) control to maintain stability and fast reach to set point. The proposed chamber consists of plywood structure sprayed blue and padded foil inside, fin heater, air circulation and vent fans, and electronic circuit used to control temperature. Fin heater and vent fan are controlled using PID and on/off controller, respectively. End user can set the desired temperature value and PID parameters. Moreover, the target and actual temperature values can monitor through LCD display.

กิตติกรรมประกาศ

ปริญญานิพนธ์สำเร็จลุล่วงไปได้ด้วยดีเนื่องจากได้รับความอนุเคราะห์จากอาจารย์ประจำภาควิชาวิศวกรรมการวัดและควบคุม ที่ถ่ายทอดความรู้ทั้งในภาคทฤษฎี และภาคปฏิบัติให้ผู้จัดทำนำมาปรับใช้ในการทำงาน และเพื่อให้การเขียนรายงานค้นคว้าจนออกมาเป็นฉบับนี้สมบูรณ์ที่สุด ซึ่งสิ่งที่ทางอาจารย์ได้ให้กับผู้จัดทำนั้นมีประโยชน์ ทั้งการให้คำแนะนำและให้คำปรึกษา รวมทั้งถ่ายทอดความรู้ ประสบการณ์ให้แก่ผู้จัดทำตลอดระยะเวลาที่ผู้จัดทำได้มีโอกาสจัดทำโครงการนี้ และสามารถนำไปใช้ในการศึกษาต่อในอนาคต โดยเฉพาะอย่างยิ่งผู้จัดทำอยากจะขอขอบคุณ ผู้ช่วยศาสตราจารย์ ดร.วรรณดี เพชรมณีล้ำค่า ซึ่งเป็นอาจารย์ที่ปรึกษาของโครงการนี้ ที่คอยให้คำแนะนำต่างๆ ทั้งในด้านการทำงาน และการแก้ไขปัญหาที่เกิดจากการทำโครงการนี้ สามารถนำมาประยุกต์ใช้ในการทำโครงการครั้งนี้ได้

ขอขอบคุณผู้สร้างเว็บไซต์และเอกสารอ้างอิงต่างๆ ที่ผู้จัดทำนำมาใช้เป็นข้อมูลในการทำโครงการและเป็นแหล่งอ้างอิง

สุดท้ายนี้ผู้จัดทำขอขอบพระคุณครอบครัวที่คอยให้การสนับสนุนและคอยให้กำลังใจแก่ผู้จัดทำมาโดยตลอด

ผู้จัดทำ

นายญาณเดช ณะศุภวุฒิ

นางสาวนภสร สุทธิพัฒน์กิจ

นายประภฤษฎี พสุธา

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ

หน้า

บทคัดย่อภาษาไทย	I
บทคัดย่อภาษาอังกฤษ	II
กิตติกรรมประกาศ	III
สารบัญ	IV
สารบัญรูป	VI
สารบัญตาราง	IX
บทที่ 1 บทนำ	1
1.1 ที่มาและความสำคัญ	1
1.2 วัตถุประสงค์	2
1.3 ขอบเขตของโครงการ	2
1.4 ขั้นตอนการดำเนินงาน	2
1.5 ประโยชน์ที่คาดว่าจะได้รับ	2
1.6 รายละเอียดปริญญานิพนธ์	3
บทที่ 2 แนวคิด ทฤษฎีและงานวิจัยที่เกี่ยวข้อง	4
2.1 ตัวควบคุมแบบ PID	4
2.1.1 ทฤษฎีตัวควบคุมแบบ PID	4
2.1.2 การปรับแต่งค่าพารามิเตอร์	7
2.2 เทอโมคัปเปิล (Thermocouple)	8
2.3 กลไกการถ่ายเทความร้อน	12
2.4 ขดลวดทำความร้อน	15
บทที่ 3 วิธีการดำเนินงาน	20
3.1 ขั้นตอนการดำเนินงาน	20
3.2 อุปกรณ์ที่เกี่ยวข้อง	21
3.3 ซอฟต์แวร์ที่เกี่ยวข้อง	25
3.3.1 โปรแกรมเขียนและสั่งการทำงาน : Arduino	25
3.3.2 โปรแกรมออกแบบอุปกรณ์และตัวควบคุมอุณหภูมิ : SolidWorks	26

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ(ต่อ)

	หน้า
3.3.3 โปรแกรมออกแบบ Schematic และ PCB : Altium Designer2016	27
3.4 การออกแบบและวางแผนการทำงาน	28
3.5 วิธีการดำเนินงาน	29
3.5.1 กระบวนการออกแบบการทำงานของตู้ควบคุมอุณหภูมิ	29
3.5.2 กระบวนการออกแบบการทำงานของโปรแกรมควบคุม	30
3.5.3 การออกแบบวงจร Interface	36
3.5.4 กระบวนการออกแบบตู้ควบคุม โดยโปรแกรม SolidWorks	37
3.5.5 กระบวนการประกอบและติดตั้งเพื่อทดสอบการทำงาน	40
บทที่ 4 ผลการดำเนินงาน	43
4.1 การออกแบบตู้ควบคุมอุณหภูมิที่สามารถโปรแกรมได้	43
4.2 การออกแบบโปรแกรมควบคุม PID	44
4.3 ผลการทดสอบการทำงานของตู้ควบคุมอุณหภูมิที่สามารถโปรแกรมได้ โดยใช้ตัวควบคุมแบบ On-Off Controller	47
4.4 ผลการทดสอบการทำงานของตู้ควบคุมอุณหภูมิที่สามารถโปรแกรมได้ โดยใช้ตัวควบคุมแบบ PID Controller	49
บทที่ 5 สรุปผลการดำเนินงานและข้อเสนอแนะ	53
5.1 สรุปผลการดำเนินงาน	53
5.2 ปัญหาและอุปสรรคในการดำเนินงาน	53
5.3 แนวทางในการแก้ปัญหา	54
เอกสารอ้างอิง	55
ภาคผนวก	56
ภาคผนวก ก Datasheet ของ DS18B20	57
ภาคผนวก ข Datasheet ของ Power Supply 24V AC-DC PMC-24V050W1AA	62
ภาคผนวก ค โปสเตอร์	67

สารบัญรูป

รูปที่	หน้า
2.1 ผลของเพลเทียร์	9
2.2 วงจรพื้นฐานในการวัดอุณหภูมิโดยใช้ Thermocouple	9
2.3 วงจรชดเชยผลของอุณหภูมิที่ Reference Junction	10
2.4 โครงสร้างของ Thermocouple	10
2.5 กราฟแสดงแรงดันและอุณหภูมิของ Thermocouple ชนิดต่างๆ	12
2.6 การถ่ายเทความร้อน	13
2.7 ความสัมพันธ์ระหว่างอุณหภูมิและการแผ่รังสีของโลหะ	15
2.8 ฮีตเตอร์แบบครีป	16
2.9 ฮีตเตอร์แบบคาร์ทริดจ์	17
2.10 ฮีตเตอร์รีดท่อ	17
2.11 ฮีตเตอร์สแตนเลสตัว U	18
2.12 ฮีตเตอร์อินฟราเรด	29
3.1 Arduino UNO R3	21
3.2 Digital Temperature Temp Sensor Probe DS18B20	21
3.3 1602 LCD (Blue Screen)	22
3.4 Power Supply 24V AC-DC PMC-24V050W1AA	22
3.5 DC Tangential Blower	23
3.6 Finned Tubular Heating	23
3.7 Cooling fan ORIX MU9255-41X	24
3.8 Solid State Relay SSR – 15DA	24
3.9 Panasonic Circuit Breaker 30A 2P	25
3.10 ไอคอนโปรแกรม Arduino	25
3.11 หน้าต่างโปรแกรม Arduino	26
3.12 ไอคอนโปรแกรม SolidWorks	26
3.13 หน้าต่างโปรแกรม SolidWorks	27
3.14 ไอคอนโปรแกรม Altium Designer2016	27

สารบัญรูป(ต่อ)

รูปที่	หน้า
3.15 หน้าต่างโปรแกรม Altium Designer2016	28
3.16 Block Diagram ของตู้ควบคุมอุณหภูมิแบบปรับค่าได้	30
3.17 Data Flow Diagram ของตู้ควบคุมอุณหภูมิแบบปรับค่าได้	30
3.18 โค้ดการทำงานของโปรแกรม PID	31
3.19 Temperature Sensor	31
3.20 คำสั่งในการทำงานของ Temperature Sensor	32
3.21 การทำงานของ PWM	33
3.22 คำสั่งค่า Carrier Frequency ของ PWM ให้มีค่า 10 Hz	34
3.23 ฟังก์ชันการทำงานของ PWM	34
3.24 สัญญาณ PWM Carrier Frequency	35
3.25 การทำงานของ PULL-UP	35
3.26 ทดลองแสดงผลการทำงานผ่านหน้าจอ LCD (1)	36
3.27 ทดลองแสดงผลการทำงานผ่านหน้าจอ LCD (2)	36
3.28 Schematic ของวงจร Interface ที่เขียนโดยโปรแกรม Altium Designer2016	37
3.29 ลายวงจร ของวงจร Interface ที่เขียนโดยโปรแกรม Altium Designer2016	37
3.30 หน้าจอการทำงานของโปรแกรม SolidWorks (1)	38
3.31 หน้าจอการทำงานของโปรแกรม SolidWorks (2)	38
3.32 หน้าจอการทำงานของโปรแกรม SolidWorks (3)	39
3.33 หน้าจอการทำงานของโปรแกรม SolidWorks (4)	39
3.34 เครื่องมือที่ใช้ในการออกแบบในโปรแกรม SolidWorks (1)	39
3.35 เครื่องมือที่ใช้ในการออกแบบในโปรแกรม SolidWorks (2)	40
3.36 แบบ 2D ของขดลวดความร้อนแบบครีป	40
3.37 ประกอบและติดตั้งอุปกรณ์ภายในตู้	41
3.38 ทดสอบการทำงาน	41
3.39 วงจร Power สำหรับตู้ควบคุมอุณหภูมิ	42
3.40 ตู้ควบคุมอุณหภูมิแบบที่สามารถโปรแกรมได้เมื่อประกอบเสร็จสิ้น	42

สารบัญรูป(ต่อ)

รูปที่	หน้า
4.1 การออกแบบของตัวควบคุมอุณหภูมิที่สามารถโปรแกรมได้ด้วยโปรแกรม SolidWorks	43
4.2 ผลการทดสอบการทำงานของตัวควบคุมอุณหภูมิที่สามารถโปรแกรมได้ โดยใช้ตัวควบคุมแบบ On-Off Controller ที่อุณหภูมิเป้าหมาย 40°C (ที่อุณหภูมิห้อง 25°C)	47
4.3 ผลการทดสอบการทำงานของตัวควบคุมอุณหภูมิที่สามารถโปรแกรมได้ โดยใช้ตัวควบคุมแบบ On-Off Controller ที่อุณหภูมิเป้าหมาย 50°C (ที่อุณหภูมิห้อง 25°C)	48
4.4 ผลการทดสอบการทำงานของตัวควบคุมอุณหภูมิที่สามารถโปรแกรมได้ โดยใช้ตัวควบคุมแบบ On-Off Controller ที่อุณหภูมิเป้าหมาย 60°C (ที่อุณหภูมิห้อง 25°C)	48
4.5 ผลการทดสอบการทำงานของตัวควบคุมอุณหภูมิที่สามารถโปรแกรมได้ โดยใช้ตัวควบคุมแบบ PID Controller ที่อุณหภูมิเป้าหมาย 40°C (ที่อุณหภูมิห้อง 25°C , $K_p = 80$, $K_i = 15$, $K_d = 0$)	49
4.6 ผลการทดสอบการทำงานของตัวควบคุมอุณหภูมิที่สามารถโปรแกรมได้ โดยใช้ตัวควบคุมแบบ PID Controller ที่อุณหภูมิเป้าหมาย 50°C (ที่อุณหภูมิห้อง 25°C , $K_p = 80$, $K_i = 15$, $K_d = 0$)	50
4.7 ผลการทดสอบการทำงานของตัวควบคุมอุณหภูมิที่สามารถโปรแกรมได้ โดยใช้ตัวควบคุมแบบ PID Controller ที่อุณหภูมิเป้าหมาย 60°C (ที่อุณหภูมิห้อง 25°C , $K_p = 80$, $K_i = 15$, $K_d = 0$)	50
4.8 ผลการทดสอบการทำงานของตัวควบคุมอุณหภูมิที่สามารถโปรแกรมได้ โดยใช้ตัวควบคุมแบบ PID Controller ที่อุณหภูมิเป้าหมาย 40°C (ที่อุณหภูมิห้อง 25°C , $K_p = 100$, $K_i = 15$, $K_d = 1$)	51
4.9 ผลการทดสอบการทำงานของตัวควบคุมอุณหภูมิที่สามารถโปรแกรมได้ โดยใช้ตัวควบคุมแบบ PID Controller ที่อุณหภูมิเป้าหมาย 50°C (ที่อุณหภูมิห้อง 25°C , $K_p = 100$, $K_i = 15$, $K_d = 1$)	51
4.10 ผลการทดสอบการทำงานของตัวควบคุมอุณหภูมิที่สามารถโปรแกรมได้ โดยใช้ตัวควบคุมแบบ PID Controller ที่อุณหภูมิเป้าหมาย 60°C (ที่อุณหภูมิห้อง 25°C , $K_p = 100$, $K_i = 15$, $K_d = 1$)	52

สารบัญตาราง

ตารางที่	หน้า
2.1 ตารางแสดงผลการปรับเปลี่ยนค่าพารามิเตอร์ของคอนโทรลเลอร์ PID	6
2.2 ตารางแสดงการคำนวณค่า PID ด้วยวิธี Ziegler–Nichols	7
2.3 ตารางแสดงคุณสมบัติของ Thermocouple แต่ละ Type	11
3.1 ตารางแสดงแผนการดำเนินงาน	20



บทที่ 1

บทนำ

1.1 ที่มาและความสำคัญ

การควบคุมนั้นเป็นการทำงานทั่วไปอย่างหนึ่งซึ่งมนุษย์หรือสิ่งมีชีวิตใดๆ จำเป็นต้องทำเสมอ ในการทำกิจกรรมต่างๆ ในชีวิตประจำวัน อาทิ การควบคุมร่างกายมนุษย์เองให้เดินตรงไปยังจุดเป้าหมาย ซึ่งเห็นว่าสมองจะคิดตัดสินใจเพื่อสั่งการควบคุมกล้ามเนื้อขาให้เคลื่อนที่ไป โดยอาศัยการเปรียบเทียบข้อมูลที่ตั้งใจไว้กับข้อมูลที่ลูกตามองเห็น นอกจากนั้นจะเห็นว่าคำสั่งการร่างกายให้ทำงานอื่นๆ ไม่ว่าจะงานอะไรนั้นล้วนต้องมีการควบคุมทั้งสิ้น ซึ่งพื้นฐานแนวคิดทางวิศวกรรมระบบควบคุม (Control Engineering) นั้นก็มีแนวคิดพื้นฐานจากการควบคุมของมนุษย์นั่นเอง โดยการควบคุมจะประกอบไปด้วย กระบวนการ (Process) ที่สามารถถูกขับโดยอินพุตสัญญาณควบคุม (Control Signal หรือ Command Signal) ผ่านอุปกรณ์ขับเคลื่อน (Actuator) เพื่อให้ได้ผลลัพธ์หรือเอาต์พุตที่ต้องการควบคุมเป็นไปตามต้องการ ซึ่งทราบค่าได้โดยอาศัยการวัด (Measurement) ผ่านอุปกรณ์หรือเซนเซอร์ (Sensor) ซึ่งทำหน้าที่แปลงสัญญาณ เพื่อให้ได้เอาต์พุตค่าวัด (Measure Output) ในลักษณะที่สามารถนำมาประมวลผลเทียบกับค่าเป้าหมาย (Set Point) หรืออินพุตอ้างอิง (Reference Input) ที่ต้องการ โดยอาศัยตัวควบคุม (Controller) เพื่อตัดสินใจสั่งสัญญาณควบคุมที่ควรจะเป็น

ซึ่งในรายงานนี้ได้ประยุกต์ใช้แนวคิดข้างต้นนี้ในการควบคุมกระบวนการ เพื่อให้การควบคุมเป็นไปโดยอัตโนมัติ โดยจะนำมาประยุกต์ใช้งานในการทำตัวควบคุมอุณหภูมิที่สามารถโปรแกรมได้ โดยนำความรู้ทางด้านวิศวกรรมระบบควบคุม ในการควบคุมอุณหภูมิภายในตู้โดยใช้ตัวควบคุมแบบ PID (Proportional-Integral-Derivative) ในการควบคุมระบบ เพื่อให้การควบคุมมีประสิทธิภาพมากที่สุด และสามารถแสดงผลได้แบบเรียลไทม์ รวมถึงการควบคุมโดยการปรับแฟคเตอร์ต่างๆ

ซึ่งเป้าหมายสำหรับโครงการตัวควบคุมอุณหภูมิที่สามารถโปรแกรมได้ เพื่อควบคุมอุณหภูมิอย่างแม่นยำและมีความรวดเร็วในการเข้าสู่ Set Point ไวที่สุด ดังนั้นโครงการนี้จะมุ่งศึกษาในการควบคุมอุณหภูมิภายในตู้โดยใช้ตัวควบคุมแบบ PID และสามารถปรับค่าพารามิเตอร์ต่างๆ ได้ผ่านหน้าจอแสดงผล (Display) และเพื่อนำความรู้และความสามารถที่ได้เรียนมานำมาประยุกต์ และเตรียมพร้อมสำหรับการทำงานทางวิศวกรรมต่อไป

1.2 วัตถุประสงค์

1. เพื่อศึกษาการทำงานของตัวควบคุมอุณหภูมิที่สามารถโปรแกรมได้
2. เพื่อควบคุมอุณหภูมิภายในตัวควบคุมอุณหภูมิสามารถโปรแกรมได้อย่างแม่นยำ
3. เพื่อทดสอบการทำงานของวงจรที่นำไปทดสอบภายในตัวควบคุมอุณหภูมิสามารถโปรแกรมได้
4. เพื่อทดสอบประสิทธิภาพในการควบคุมอุณหภูมิของตัวควบคุมอุณหภูมิสามารถโปรแกรมได้

1.3 ขอบเขตของโครงการ

1. ศึกษาการทำงานของตัวควบคุมอุณหภูมิแบบปรับค่าอุณหภูมิได้
2. วิเคราะห์การทำงานของระบบควบคุมกระบวนการ
3. สามารถอธิบายหลักการทำงานของตัวควบคุมอุณหภูมิแบบปรับค่าได้
4. สามารถปรับค่าเพคเตอร์ของ PID เพื่อควบคุมการทำงานของตัวควบคุมอุณหภูมิได้

1.4 ขั้นตอนการดำเนินงาน

1. กำหนดรายละเอียดการทำงานของตัวควบคุมอุณหภูมิสามารถโปรแกรมได้
2. ออกแบบตัวควบคุมอุณหภูมิสามารถโปรแกรมได้ด้วยโปรแกรม SolidWorks
3. จัดหาอุปกรณ์
4. ประกอบตัวควบคุมอุณหภูมิสามารถโปรแกรมได้
5. ติดตั้งระบบสายไฟภายในตัวควบคุมอุณหภูมิสามารถโปรแกรมได้
6. ออกแบบโปรแกรมสำหรับควบคุมอุณหภูมิ
7. ทดสอบการทำงานและแก้ไขข้อผิดพลาด
8. สรุปผลการทดลอง
9. นำเสนอโครงการ

1.5 ประโยชน์ที่คาดว่าจะได้รับ

1. สามารถรักษาและควบคุมอุณหภูมิภายในตัวควบคุมอุณหภูมิได้ตามที่ต้องการ
2. สะดวกและง่ายต่อการใช้งาน
3. สามารถใช้งานได้จริง
4. สามารถพัฒนาต่อยอดได้

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1.6 รายละเอียดปฏิญญานิพนธ์

เนื้อหาที่จะกล่าวในปฏิญญานิพนธ์ฉบับนี้ประกอบด้วย

บทที่ 1 บทนำ กล่าวถึงความเป็นมาของโครงการ วัตถุประสงค์ของโครงการ ขอบเขตของโครงการ ขั้นตอนการดำเนินงาน ประโยชน์ที่คาดว่าจะได้รับ พร้อมทั้งรายละเอียดของโครงการแต่ละบท

บทที่ 2 งานวิจัยและทฤษฎีที่เกี่ยวข้อง นำเสนอถึงหลักการและทฤษฎีที่เกี่ยวข้องในการจัดทำโครงการ

บทที่ 3 การออกแบบและโครงสร้างของระบบ อธิบายภาพรวมของระบบ อธิบายขั้นตอนการทำงานในการออกแบบและประมวลผล

บทที่ 4 การทดลอง ผลการทดลอง และการประยุกต์ใช้งานอุปกรณ์

บทที่ 5 บทวิจารณ์และสรุป เป็นการสรุปผลการดำเนินงาน ปัญหาที่เกิดขึ้น และแนวทางการปรับปรุงพัฒนาโครงการนี้ต่อไป



บทที่ 2

แนวคิด ทฤษฎีและงานวิจัยที่เกี่ยวข้อง

2.1 ตัวควบคุมแบบ PID

คอนโทรลเลอร์แบบ PID (Proportional Integral Derivative Controller) คือ อุปกรณ์ในการควบคุมระบบตามที่ต้องการ โดยจะมีการตั้ง SP (Set Point หรือค่าที่ต้องการไว้) และนำมาเปรียบเทียบกับค่า PV (Process Variable หรือตัวแปรกระบวนการหรือค่าจริงที่เกิดจากผลการทำงานของระบบ) เพื่อให้ได้ค่า Error แล้วคอนโทรลเลอร์จะนำค่า Error นั้นมาทำการปรับแต่งค่าเอาต์พุตหรือ MV (Manipulated Variable หรือตัวแปรที่ถูกควบคุม) เพื่อลดค่า Error ให้ได้ต่ำที่สุด ส่วน PID ย่อมาจาก Proportional-Integral-Derivative โดยที่

1. Proportional เป็นส่วนปฏิกิริยาต่อ Error ณ ขณะนั้น (Current Error)
2. Integral เป็นส่วนปฏิกิริยาที่เกิดจากผลรวมของค่า Error ที่ผ่านมามีค่าล่าสุด (Sum of Recent Errors)
3. Derivative เป็นส่วนปฏิกิริยาที่เกิดจากอัตราการเปลี่ยนแปลงของค่า Error (Rate at Which the Error has Been Changing)

ผลรวมตามน้ำหนักของทั้งสาม ซึ่งปรับแต่งโดยค่าคงที่ K_p , K_i และ K_d เพื่อให้ได้การตอบสนองที่ต้องการ จะถูกนำไปใช้ในการควบคุมกระบวนการต่างๆ เช่น ครอบเครื่อง/เพลลา ตำแหน่งวาล์ว หรืออุปกรณ์ทำความร้อน โดยทั่วไปค่า Gain บางค่า อาจเป็นศูนย์ก็ได้ซึ่งคอนโทรลเลอร์แบบ PI, PD, P หรือ I ขึ้นอยู่กับว่าเทอมไหนหายไป

2.1.1 ทฤษฎีตัวควบคุมแบบ PID

การควบคุมแบบ PID นั้นประกอบด้วยเทอมสำหรับการปรับแต่ง 3 เทอมรวมกันเป็น MV ดังสมการที่ (2.1)

$$MV(t) = P_{out} + I_{out} + D_{out} \quad (2.1)$$

โดยที่ P, I และ D เป็นเอาต์พุตจากเทอม Proportional, Integral และ Derivative ตามลำดับ

เทอม Proportional

เทอม Proportional เปลี่ยนแปลงตามอัตราส่วนของค่า Error ปัจจุบัน ซึ่งค่า P สามารถคำนวณได้โดยการนำค่า Error มาคูณกับค่าคงที่ K_p ดังสมการที่ (2.2)

$$P_{out} = K_p e(t) \quad (2.2)$$

ค่า K_p ที่สูง จะเป็นผลให้ค่าเอาต์พุตมากขึ้นตาม หากค่า K_p มากเกินไป ระบบอาจจะไม่เสถียรได้ ในทางตรงข้ามหากค่า K_p น้อยเกินไปอาจทำให้ระบบตอบสนองช้าเกินไป ซึ่งในกรณีนี้เป็นไปได้ว่าการตอบสนองดังกล่าว อาจจะถูกสิ่งรบกวน (Disturbance) ทำให้ระบบไม่สามารถทำงานได้ทันการ เช่น ในการเร่งรอบเครื่องยนต์ เมื่อมีโหลดกระทำกับเครื่อง หากคอนโทรลเลอร์เร่งรอบเครื่องรอบเครื่องช้าไม่ทันกับโหลดที่เพิ่มขึ้นมาทันที รอบเครื่องจะค่อยๆ ตกลงและเครื่องจะดับในที่สุด ซึ่งในกรณีระบบที่ไม่มี Disturbance การใช้เทอม Proportional อย่างเดียวจะทำให้ระบบเกิดการ Oscillate รอบๆ SP จะไม่อยู่ที่ SP พอดี

เทอม Integral

เทอม Integral เป็นอัตราส่วนของค่า Error สะสมในหนึ่งช่วงเวลา (ปัจจุบันย้อนไปในอดีต) ค่า I เกิดจากผลคูณของค่าคงที่ K_i กับผลรวมของ $e(t)$ ซึ่งเป็นค่าสะสมของ Error ที่ควรจะต้องถูกแก้ไขมาก่อนหน้านี้ ดังสมการที่ (2.3)

$$I_{out} = K_i \int_0^t e(t) dt \quad (2.3)$$

เทอม Integral (เมื่อใช้ร่วมกับเทอม Proportional) จะช่วยเร่งให้ระบบวิ่งเข้าหา SP เร็วขึ้น และช่วยลด Error ที่เกิดจากการใช้เทอม Proportional อย่างเดียว อย่างไรก็ตามเนื่องจากว่าเทอม Integral นั้นเกิดจากการคำนวณโดยรวม Error ที่เกิดขึ้นในอดีตด้วยอาจทำให้เกิด Overshoot เกิน SP ในค่าปัจจุบันด้วย

เทอม Derivative

อัตราการเปลี่ยนแปลงของค่า Error ในกระบวนการนั้นคำนวณได้โดยความชัน (Slope) ของกราฟ Error และ Time ซึ่งก็คืออนุพันธ์แรกเทียบกับเวลา และคูณค่าความชันนี้กับค่าคงที่ K_d ก็จะได้เทอม Derivative ดังสมการที่ (2.3)

$$D_{out} = K_d \frac{de}{dt} e(t) \quad (2.3)$$

เทอม Derivative จะช่วยดึงเอาต์พุตจากคอนโทรลเลอร์ให้ช้าลง ซึ่งจะเห็นผลได้ชัดเมื่อโปรเซสเข้าใกล้ Set Point ฉะนั้นเทอม Derivative จึงช่วยลดอาการ Overshoot ซึ่งเกิดจากเทอม Integral และช่วยปรับปรุงเสถียรภาพของระบบโดยรวม อย่างไรก็ตามค่า Derivative นั้นค่อนข้างไวต่อ Noise ซึ่งอาจทำให้ระบบไม่เสถียรได้หาก Noise และค่า K_p สูงเกินไป

เมื่อรวมเทอม Proportional, Integral และ Derivative เข้าด้วยกันก็จะได้เอาต์พุตจากคอนโทรลเลอร์ PID ดังสมการที่ (2.5)

$$MV(t) = K_p e(t) + K_i \int_0^t e(t) dt + K_d \frac{de}{dt} e(t) \quad (2.5)$$

โดยค่าพารามิเตอร์ต่างๆ คือ

ค่า Proportional Gain, K_p : ค่าที่มากขึ้นหมายถึงการตอบสนองที่เร็วขึ้นเพราะค่า Error ยิ่งมาก ค่าชดเชยจากเทอมนี้ก็จะมากขึ้นตามค่า Gain ที่มากขึ้นจะไปสู่โปรเซสที่ไม่เสถียรการแกว่ง (Oscillation)

ค่า Integral Gain, K_i : ค่าที่มากขึ้นหมายถึงค่า Error แบบ Steady-state จะถูกกำจัดได้เร็วขึ้น ข้อเสียก็คือ Overshoot ค่า Error ที่เป็นลบจะต้องถูกแก้ด้วย Error ที่เป็นบวก ก่อนที่ระบบจะเข้าสู่ Steady-state

ค่า Derivative Gain, K_d : ค่าที่มากขึ้นหมายถึงขนาด Overshoot ที่ลดลง แต่อาจทำให้การตอบสนองช้าลงบ้างและอาจนำไปสู่ความไม่เสถียรของระบบเนื่องจาก Noise ได้

ตารางที่ 2.1 ตารางแสดงผลการเพิ่มค่าพารามิเตอร์ของคอนโทรลเลอร์ PID

พารามิเตอร์	Rise time	Overshoot	Settling time	Error at equilibrium
K _p	ลดลง	เพิ่มขึ้น	เปลี่ยนเล็กน้อย	ลดลง
K _i	ลดลง	เพิ่มขึ้น	เพิ่มขึ้น	ถูกกำจัด
K _d	นานมาก (ลดลงหรือเพิ่มขึ้นน้อยมาก)	ลดลง	ลดลง	ไม่มี

2.1.2 การปรับแต่งค่าพารามิเตอร์

ในการปรับแต่งค่าพารามิเตอร์มี 3 วิธีหลักดังนี้

1. การปรับแต่งด้วยมือ (Manual Tuning)

จะเริ่มโดยการเซตค่า K_i และ K_d เป็นศูนย์ และค่อยๆ เพื่อค่า K_p เรื่อยๆ จนกระทั่งระบบเริ่มเกิดการแกว่ง จากนั้นให้เซตค่า K_p เป็นครึ่งหนึ่งค่านั้น จากนั้นให้เริ่มเพิ่มค่า K_i จนได้เวลาการตอบสนองของระบบที่ต้องการ จากนั้นหากจำเป็นเพิ่มค่า K_d จนกระทั่งการควบคุมความเร็วพอที่ยอมรับได้โดยเทียบกับเมื่อระบบมี Disturbance อ่างอิง

2. การปรับแต่งด้วยวิธี Ziegler-Nichols

จะเริ่มด้วยการเซตค่า K_i และ K_d เป็นศูนย์จากนั้นจะเพิ่มค่า K_p ไปจนถึงค่า K_c (Critical Gain) ระบบจะเริ่มแกว่งให้วัดคาบของการแกว่ง P_c และให้ใช้ตารางข้างล่างเพื่อหาค่า Gain อื่นๆ

ตารางที่ 2.2 ตารางแสดงการคำนวณค่า PID ด้วยวิธี Ziegler-Nichols

วิธี Ziegler - Nichols			
Control Type	K_p	K_i	K_d
P	$0.50K_c$	-	-
PI	$0.45K_c$	$1.2K_p/P_c$	-
PID	$0.60K_c$	$2K_p/P_c$	$K_p P_c/8$

3. การปรับแต่งด้วยซอฟต์แวร์

ในอุตสาหกรรมส่วนใหญ่ในปัจจุบัน จะใช้ซอฟต์แวร์ในการคำนวณและปรับแต่งค่า Gain โดยซอฟต์แวร์จะทำการเก็บข้อมูลสร้างโมเดลและคำนวณค่า Gain ให้โดยอัตโนมัติ

ข้อจำกัดของคอนโทรลเลอร์แบบ PID

ในกรณีที่คอนโทรลเลอร์ทำงานไม่ดีเกิดอาการฮันต์ (Hunt) อาจจะต้องใช้รวมคอนโทรลเลอร์แบบ Feed Forward ด้วยโดยการบวกค่า Bias เพิ่มเข้าไปในสมการ $MV(t)$

เนื่องจากคอนโทรลเลอร์ PID เป็นเชิงเส้น (Linear) ในกรณีที่ใช้กับระบบที่ไม่เป็นเชิงเส้น (Non-Linear) อาจจะต้องใช้การปรับค่า Gain ตามตารางเวลา หรือใช้คอนโทรลเลอร์แบบ Fuzzy Logic แทน

2.2 เทอร์โมคัปเปิล (Thermocouple)

เทอร์โมคัปเปิลคือ อุปกรณ์วัดอุณหภูมิโดยใช้หลักการเปลี่ยนแปลงอุณหภูมิ หรือความร้อน เปลี่ยนมาเป็นแรงเคลื่อนไฟฟ้า (EMF) เทอร์โมคัปเปิลทำมาจากโลหะตัวนำที่ต่างชนิดกัน 2 ตัว (แตกต่างกันทางโครงสร้างของอะตอม) นำมาเชื่อมต่อปลายทั้งสองเข้าด้วยกันที่ปลายด้านหนึ่งเรียกว่า จุดวัดอุณหภูมิ ส่วนปลายอีกด้านหนึ่งปล่อยให้เรียกว่าจุดอ้างอิง หากจุดวัดอุณหภูมิและจุดอ้างอิง มีอุณหภูมิต่างกันก็จะทำให้การนำกระแสในวงจรเทอร์โมคัปเปิลทั้งสองข้าง ดังนั้นสิ่งที่เทอร์โมคัปเปิล วัดได้จึงเป็นความแตกต่างระหว่างอุณหภูมิ 2 จุด ทฤษฎีพื้นฐานของผลจากเทอร์โมอิเล็กทริก เกิดจากการส่งผ่านทางไฟฟ้าและทางความร้อนของโลหะที่ต่างกัน จึงทำให้เกิดความต่างศักย์ทางไฟฟ้าตกคร่อมที่โลหะนั้น ความต่างศักย์นี้จะสัมพันธ์กับความจริงที่ว่า อิเล็กตรอนในปลายด้านร้อนของโลหะ จะมีพลังงานความร้อนมากกว่าปลายทางด้านเย็น จึงทำให้อิเล็กตรอนมีความเร็วไปหาปลายด้านเย็น ที่อุณหภูมิเดียวกันนี้การเคลื่อนที่ของอิเล็กตรอนจะแปรเปลี่ยนไปตามโลหะต่างชนิดกันด้วย ที่เป็นเช่นนี้เพราะว่า โลหะที่ต่างกันจะมีการนำความร้อนที่ต่างกันนั่นเอง ผลของซีแบ็ค (Seebeck Effect) โดยใช้ทฤษฎีโซลิตสเตต สามารถวิเคราะห์ค่าได้จากสมการอินทิเกรตค่าจากย่านของอุณหภูมิดังกล่าวนั้นดังสมการที่ (2.6)

$$\mathcal{E} = \int_{T_1}^{T_2} Q_A - Q_B dT \quad (2.6)$$

สมการนี้จะอธิบายผลของซีแบ็คซึ่งพบว่าค่า EMF ที่เกิดจะเป็นสัดส่วนกับความแตกต่างของอุณหภูมิ จึงเกิดความแตกต่างของค่าคงที่ในการส่งผ่านความร้อนของโลหะ ถ้าใช้โลหะชนิดเดียวกัน มาทำเทอร์โมคัปเปิลค่า EMF ที่ได้จะมีค่าเป็นศูนย์ ถ้าอุณหภูมิทั้งสองจุดวัดและจุดอ้างอิงเหมือนกัน ค่า EMF ก็จะเป็นศูนย์ดังสมการที่ (2.7)

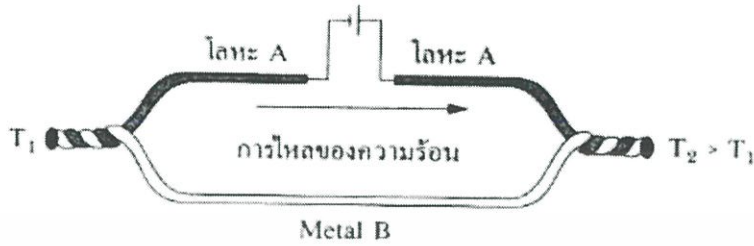
$$\mathcal{E} = \alpha(T_1 - T_2) \quad (2.7)$$

เมื่อ α = ค่าคงที่หรือเรียกว่าสัมประสิทธิ์ของซีแบ็ค ; Volts/K

T_1, T_2 = อุณหภูมิที่จุดต่อ ; K

ผลของเพลเทียร์ (Peltier Effect) หากคิดย้อนกลับจากผลของซีแบ็ค นั่นคือ ใช้โลหะที่แตกต่างกันสองชนิดมาเชื่อมต่อทั้งสองเข้าด้วยกัน แล้วจ่ายพลังงานจากภายนอกเข้าไปก็จะเป็นเหตุให้กระแสไหลในวงจร เพราะจากคุณสมบัติในการส่งไฟฟ้าและความร้อนของโลหะ พบว่าขั้วหนึ่งจะเกิดความร้อน (T_2) และอีกขั้วหนึ่งจะเกิดความเย็น (T_1) ขึ้น โดยผลดังกล่าวเรียกว่า “ผลของเพล

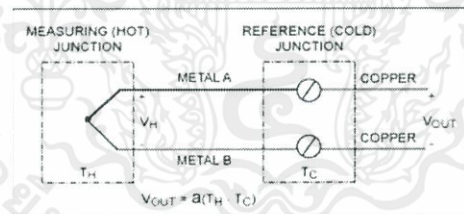
เทียร์” (Peltier Effect) และถูกนำไปใช้งานพิเศษสำหรับการทำความเย็นกับส่วนของระบบ อิเล็กทรอนิกส์ หรือแม้กระทั่งเครื่องทำความเย็นขนาดเล็ก ดังรูปที่ 2.1



รูปที่ 2.1 ผลของเพลเทียร์

จุดอ้างอิงในการวัด (Reference Junction)

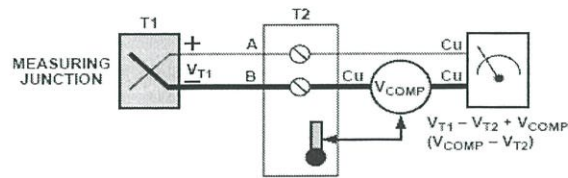
ในวงจรทั่วไปซึ่งใช้เทอร์โมคัปเปิล เพื่อทำการวัดอุณหภูมินั้น สายทองแดงจะถูกนำมาต่อเข้ากับ Reference Junction เพื่อทำการวัดแรงเคลื่อนไฟฟ้าที่เกิดขึ้น ตรงจุดนี้จะเกิดจุดเชื่อมต่อระหว่าง โลหะเพิ่มขึ้นอีก 2 จุด ซึ่งจะส่งผลให้แรงเคลื่อนไฟฟ้าที่ได้เปลี่ยนแปลงไป แต่ถ้าอุณหภูมิ ณ จุดที่ เชื่อมต่อกับสายทองแดงทั้ง 2 จุดนั้นที่อุณหภูมิที่เท่ากันแรงเคลื่อนไฟฟ้าจาก 2 จุดนี้จะหักล้างกันไป และไม่ส่งผลกระทบต่อการวัดแรงเคลื่อนไฟฟ้า (V_{out}) ดังนั้นจึงจำเป็นต้องรักษาอุณหภูมิทั้งสองจุด ให้มีค่าเท่ากันและมักเรียกว่าจุดต่อนี้ว่า Isothermal Block ดังรูปที่ 2.2



รูปที่ 2.2 วงจรพื้นฐานในการวัดอุณหภูมิโดยใช้ Thermocouple

ในสมัยก่อนนิยมใช้ Ice Bath ในการรักษาอุณหภูมิของจุด Reference Point ให้มีค่าคงที่ 0 องศาเซลเซียส และกำหนดเป็นตารางมาตรฐาน (Standard Table) ระหว่างอุณหภูมิที่ทำการวัดและแรงเคลื่อนไฟฟ้าที่เกิดขึ้นเมื่อวัดแรงเคลื่อนไฟฟ้า ที่เกิดขึ้นนำไปเปรียบเทียบกับตารางมาตรฐานก็จะทราบว่าอุณหภูมิที่ ณ จุดวัดนั้นมีค่าเท่าไร แต่การใช้งานจริงนั้น Ice Bath มีข้อยุ่งยากหลายประการ ในภายหลังจึงได้ทำการออกแบบวงจรสำหรับอุณหภูมิที่ Reference Junction แล้วนำไปสร้างแรงดัน เพื่อ Compensate กับผลของอุณหภูมิที่เกิดขึ้น Reference Junction ดังรูปที่ 2.3

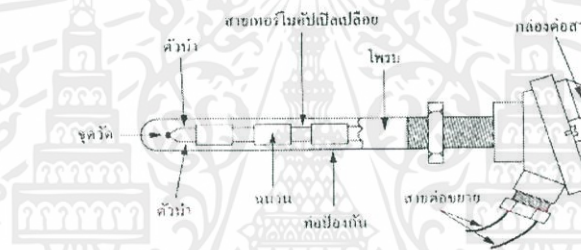
เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.3 วงจรชดเชยผลของอุณหภูมิที่ Reference Junction

คุณสมบัติของเทอร์โมคัปเปิลมาตรฐาน (Characteristic of Standard Thermocouple)

1. ความไว (Sensitivity) จากตารางแรงเคลื่อนของ NBS แสดงว่าย่านของแรงเคลื่อนจากเทอร์โมคัปเปิลจะมีค่าน้อยกว่า 100 mV แต่ความไวที่แท้จริงในการใช้งานจะขึ้นอยู่กับการใช้วงจรปรับสภาพสัญญาณและตัวเทอร์โมคัปเปิลเอง ดังรูปที่ 2.4



รูปที่ 2.4 โครงสร้างของ Thermocouple

2. โครงสร้าง (Construction) โครงสร้างของเทอร์โมคัปเปิลมีลักษณะดังรูปโดยต้องมีลักษณะคือ มีความต้านทานต่ำให้สัมประสิทธิ์อุณหภูมิสูง ต้านทานต่อการเกิดออกไซด์ที่อุณหภูมิสูง ทนต่อสภาวะแวดล้อมที่นำไปใช้วัดค่า และเป็นเชิงเส้นสูงที่สุดเท่าที่จะเป็นไปได้ตัวผู้ผลิตหรือท่อป้องกันส่วนมากจะทำจากสแตนเลส ความไวของเทอร์โมคัปเปิลขึ้นอยู่กับความหนาของท่อป้องกันทั้งเยอร์มาเนียมและซิลิคอน จะทำให้คุณสมบัติการเกิดเทอร์โมอิเล็กทริกจึงใช้กันมากในอุปกรณ์ทำความเย็น (Peltier Element) มากกว่าที่จะใช้เป็นเทอร์โมคัปเปิลวัดอุณหภูมิขนาดของสายเทอร์โมคัปเปิลกำหนดได้จากการใช้งานแต่ละอย่าง และมีขนาดที่ใช้ในสภาวะแวดล้อมที่ไม่คงที่หรือแม้กระทั่งสายแบบ ไมโครไวร์ (Micro Wire) ที่ใช้กับการวัดอุณหภูมิการกลั่นในงานทางชีววิทยา

3. ย่านการใช้งาน (Range) ย่านการวัดอุณหภูมิการใช้งานและความไวในการวัดของเทอร์โมคัปเปิลแต่ละตัว จะแตกต่างกันตามแต่ละสมาคมจะกำหนด ในส่วนที่สำคัญคือ ค่าแรงเคลื่อนที่ออกมาจากแต่ละอุณหภูมิจะต้องอ้างอิงกับตารางค่ามาตรฐานของแต่ละสมาคม ที่ใช้ให้ถูกต้องเป็นเอกภาพเดียวกันหมดทั้งระบบ ดังตารางที่ 2.3

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 2.3 ตารางแสดงคุณสมบัติของ Thermocouple แต่ละ Type

Type of Wire (USA and Canada colors)	Temperature Range	Limit of Error (Select whichever is greater)	
		Standard Grade	Premium Grade
Type T Copper-Constantan (Blue or brown sheath; blue+,red-)	-200 to 0°C	±1°C or ±1.5%	---
	0 to 350°C	±1°C or ±0.75%	±0.5°C or ±0.4%
	-300 to 32°F	±1.5°F or ±2%	±0.75°F or
	32 to 700°F	±1.5°F or ±0.75%	±1% ±0.75°F or ±0.38%
Type J Iron-Constantan (Black or brown sheath; white+,red-)	0 to 750°C	±2.2°C or	±1.1°C or
	32 to 1400°F	±0.5% ±4°F or ±0.5%	±0.4% ±2°F or ±0.38%
Type K Chromel-Alumel (Yellow Sheath; purple+,red-)	0 to 1250°C	±2.2°C or	±1.1°C or
	32 to 2300°C	±0.75% ±4°F or	±0.4% ±2°F or
		±0.75%	±0.38%
Type R or S Platinum-Rhodium/Platinum (Green sheath; black+,red-)	0 to 1450°C	0 to 1450°C	---
	32 to 2700°F	32 to 2700°F	---
Type B Platinum30% Rhodium/Platinum6%Rhodium (Grey Sheath; grey+,red-)	800 to 1700°C	800 to 1700°C	---
	1600 to 3100 °F	1600 to 3100°F	---
Type E Chormel-Constantan (Purpleorbrown sheath;purple+,red-)	0 to 900°C	0 to 900°C	±1°C or ±0.4%
	32 to 1600°F	32 to 1600°F	±2°F or ±0.38%

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ซึ่งการใช้งานจะต่างกัน

Type S : เหมาะกับการใช้งานในสภาวะที่เกิดปฏิกิริยาเคมี

Type R : เป็นแบบที่เหมาะสมกับการวัดอุณหภูมิสูงๆ

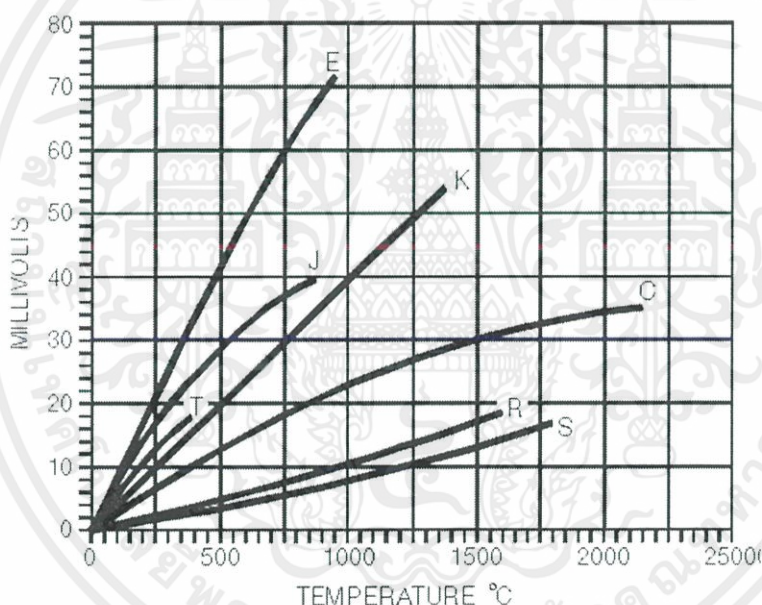
Type B : เหมาะกับการใช้งานในสภาวะที่มีปฏิกิริยาแบบออกซิไดซิงและสภาวะเฉื่อย

Type J : ใช้แพลททินัมมาทำเป็นเทอร์โมคัปเปิล

Type K : เป็นแบบที่นิยมใช้แพร่หลายมากที่สุด

Type T : เหมาะกับการวัดอุณหภูมิต่ำกว่าจุดเยือกแข็งของน้ำ

Type E : ให้แรงเคลื่อนไฟฟ้าสูงสุดเมื่อวัดอุณหภูมิกับแบบอื่นๆ ในสภาวะเดียวกัน



รูปที่ 2.5 กราฟแสดงแรงดันและอุณหภูมิของ Thermocouple ชนิดต่างๆ

2.3 กลไกการถ่ายเทความร้อน

ความร้อนคือ พลังงานที่ถ่ายเทระหว่างจุดสองจุดในจักรวาล ที่เกิดขึ้นจากความไม่สมดุลทางความร้อนของจุดสองจุดนั้น การถ่ายเทความร้อนคือ การถ่ายเทพลังงานที่เกิดเนื่องจากความไม่สมดุลทางความร้อนระหว่างจุดสองจุดในจักรวาล โดยความร้อนจะถ่ายเทจากจุดที่มีอุณหภูมิสูงกว่าไปยังจุดที่มีอุณหภูมิต่ำกว่ากลไกการถ่ายเทความร้อนคือ วิธีการที่ความร้อนเคลื่อนที่จากจุดหนึ่งในจักรวาลไปยังอีกจุดหนึ่งในจักรวาล โดยกลไกการถ่ายเทความร้อนที่มนุษย์ค้นพบแล้วมี 3 วิธีคือ การนำความร้อน การพาความร้อน และการแผ่รังสี

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การนำความร้อน

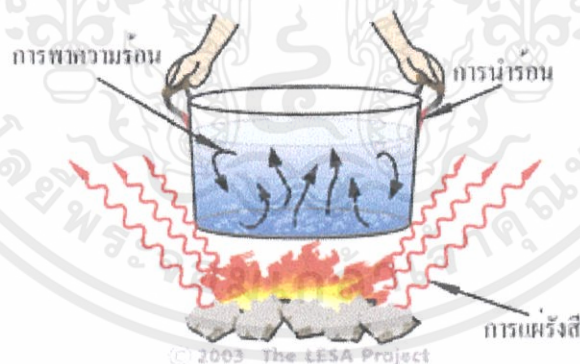
การนำความร้อน เป็นการถ่ายเทความร้อนจากโมเลกุลไปสู่อีกโมเลกุลหนึ่งที่อยู่ติดกันไปเรื่อยๆ จากอุณหภูมิสูงไปยังอุณหภูมิต่ำ ยกตัวอย่างเช่น หากจับทัพพีในหม้อหุงข้าว ความร้อนจะเคลื่อนที่ผ่านทัพพีมายังมือ ทำให้รู้สึกร้อน โดยโลหะเป็นตัวนำความร้อนที่ดี อโลหะและอากาศเป็นตัวนำความร้อนที่เลว

การพาความร้อน

การพาความร้อน เป็นการถ่ายเทความร้อนด้วยการเคลื่อนที่ของโมเลกุลของสสาร ซึ่งมีสถานะเป็นของเหลวและแก๊ส ส่วนของแข็งมีการถ่ายเทความร้อนด้วยการนำความร้อนและการแผ่รังสีเท่านั้น การพาความร้อนส่วนมากมักเกิดขึ้นในบรรยากาศและมหาสมุทร รวมทั้งแม็กมาและโลหะเหลวภายในโลก และแก๊สร้อนในดวงอาทิตย์

การแผ่รังสี

การแผ่รังสี เป็นการถ่ายเทความร้อนออกรอบตัวในทุกทิศทาง โดยไม่อาศัยตัวกลางในการส่งถ่ายพลังงาน ดังเช่น การนำความร้อนและการพาความร้อน การแผ่รังสีจึงสามารถถ่ายเทความร้อนผ่านอวกาศได้ วัตถุทุกชนิดที่มีอุณหภูมิสูงกว่า -273 องศาเซลเซียส หรือ 0 เคลวิน ย่อมมีการแผ่รังสี โดยวัตถุที่มีอุณหภูมิสูงแผ่รังสีคลื่นสั้น วัตถุที่มีอุณหภูมิต่ำแผ่รังสีคลื่นยาวตามกฎของวิน



รูปที่ 2.6 การถ่ายเทความร้อน

กฎของวิน

การอธิบายในเชิงฟิสิกส์ อุณหภูมิหมายถึงระดับพลังงานภายในอะตอม สสารทุกชนิดในจักรวาลมีอุณหภูมิสูงกว่า 0 เคลวิน (-273 องศาเซลเซียส) เนื่องจากอะตอมมีพลังงาน อะตอมประกอบด้วยอิเล็กตรอนหมุนรอบนิวเคลียส สภาวะที่อุณหภูมิ 0 เคลวิน เป็นสภาวะไร้พลังงานและ

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ทุกอย่างหยุดนิ่ง ซึ่งเป็นสภาวะที่ยังไม่มีการค้นพบ ในปี ค.ศ.1893 วิลเฮล์ม วิน นักฟิสิกส์ชาวเยอรมัน ค้นพบว่า อุณหภูมิของสสารแปรผกผันกับการแผ่คลื่นแม่เหล็กไฟฟ้า ดังสมการที่ (2.8)

$$\lambda_{\max} = 0.0029/T \quad (2.8)$$

โดยที่ λ_{\max} หมายถึง ความยาวคลื่นเข้มสุดของการแผ่รังสี มีหน่วยเป็นเมตร

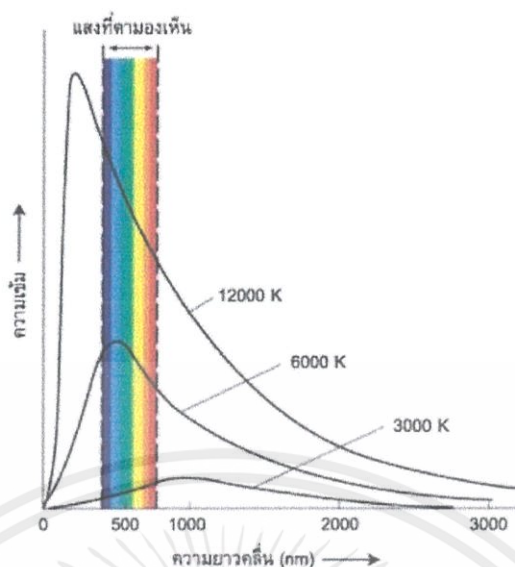
T หมายถึง อุณหภูมิของวัตถุ มีหน่วยเป็นเคลวิน

สามารถสรุปกฎของวินได้ว่า วัตถุอุณหภูมิสูงแผ่รังสีคลื่นสั้น วัตถุอุณหภูมิต่ำแผ่รังสีคลื่นยาว รังสีที่มีความยาวคลื่นน้อยกว่าแสงที่ตามองเห็น เรียกว่า คลื่นสั้น เป็นอันตรายต่อมนุษย์ รังสีที่มีความยาวคลื่นมากกว่าแสงที่ตามองเห็นเรียกว่า คลื่นยาว ไม่เป็นอันตรายต่อมนุษย์ อย่างไรก็ตามไม่ว่ารังสีจะมีความยาวคลื่นเท่าไร หากมีความเข้มสูงก็จะทำให้วัตถุที่ดูดกลืนรังสีนั้นๆ มีอุณหภูมิสูงขึ้นได้ เช่น รังสีอินฟราเรดทำให้โลกร้อน เตาไมโครเวฟสามารถทำให้น้ำเดือด

ตัวอย่างแหล่งกำเนิดคลื่นแม่เหล็กไฟฟ้า ทั้งที่เกิดขึ้นโดยธรรมชาติและเกิดขึ้นโดยฝีมือมนุษย์ ได้แก่

- รังสีแกมมา เกิดจากการระเบิดของดาวมวลมาก (ซูเปอร์โนวา) หรือระเบิดปรมาณู
- รังสีเอ็กซ์ เกิดจากการยุบตัวของดาวมวลมาก (หลุมดำ) หรือรังสีที่ใช้ในวงการแพทย์
- รังสีอัลตราไวโอเล็ต เกิดจากการแผ่รังสีของดวงอาทิตย์
- แสงที่ตามองเห็น เช่น แสงแดด หลอดไฟฟ้า
- รังสีอินฟราเรด เกิดจากการแผ่รังสีของโลก แก๊สเรือนกระจก แก๊สในอากาศ และสิ่งมีชีวิต
- คลื่นไมโครเวฟและคลื่นวิทยุ เกิดจากธรรมชาติ และมนุษย์สร้างขึ้นเพื่อการสื่อสาร

โทรคมนาคม



รูปที่ 2.7 ความสัมพันธ์ระหว่างอุณหภูมิและการแผ่รังสีของโลหะ

2.4 ขดลวดทำความร้อน

ฮีตเตอร์ (Heater) เป็นอุปกรณ์ให้ความร้อนแก่ชิ้นงานในโรงงานอุตสาหกรรม โดยใช้หลักการจ่ายกระแสไฟฟ้าไหลผ่านลวดตัวนำ (ตัวความต้านทาน R) ซึ่งส่งผลให้ลวดตัวนำมีความร้อนเกิดขึ้น โดยแหล่งจ่ายไฟสามารถใช้ได้กับแรงดัน 220VAC และ 380VAC ทำให้ผู้ใช้สามารถใช้งานฮีตเตอร์ได้ง่ายและสะดวก เนื่องจากการใช้งานฮีตเตอร์นั้นสามารถเข้าใจหลักการทำงานได้ง่าย ซึ่งในปัจจุบันโรงงานอุตสาหกรรมส่วนใหญ่นิยมหันมาใช้ฮีตเตอร์มากยิ่งขึ้น เนื่องจากราคาถูก และสั่งขนาดรูปทรง และวัตต์ได้ตามความต้องการ โดยส่วนประกอบส่วนใหญ่ของฮีตเตอร์มีดังนี้

1. ฉนวนแมกนีเซียมออกไซด์ (MgO) คุณสมบัติมีค่าความนำทางไฟฟ้าต่ำ แต่นำความร้อนได้ดีมาก ซึ่งทำหน้าที่กั้นกลางระหว่างลวดตัวนำฮีตเตอร์กับปลอกโลหะ เพื่อป้องกันไม่ให้มีกระแสรั่ว (Leak Current) จากลวดฮีตเตอร์ออกไปยังผิวโลหะ จุดสำคัญคือ ห้ามมีความชื้นในฉนวนเด็ดขาด เนื่องจากจะทำให้ค่าความนำไฟฟ้าสูงขึ้น วิธีการแก้ไขคือการนำฮีตเตอร์ไปอบในเตาอบเพื่อไล่ความชื้น

2. สแตนเลส (Stainless) ที่นำมาใช้ในการผลิตฮีตเตอร์มีอยู่หลายชนิด ซึ่งแต่ละชนิดจะมีคุณสมบัติแตกต่างกันไป ดังนี้

+Stainless 304 : ใช้ในงานที่ต้องการขึ้นรูปเพื่อการตกแต่งให้สวยงาม สามารถป้องกันสนิมได้เป็นอย่างดี

+Stainless 316 : ถูกออกแบบให้ป้องกันสนิมได้เป็นอย่างดี สามารถใช้ในงานอุตสาหกรรมหนัก และสถานที่ใกล้ทะเลที่มีความเป็นกรด-ด่างสูง

+Stainless 430 : เป็นสแตนเลสที่ใช้โครเมียมเป็นส่วนประกอบ 100% และมีโอกาสเกิดสนิมน้อยกว่าเบอร์ 300 ซึ่งสแตนเลสแบบนี้นิยมนำมาตกแต่งภายใน

3. ลวดฮีตเตอร์ซึ่งเรียกว่า ลวด Nikrothal 80 หรือ R80 โดยมีส่วนผสมของนิกเกิล 80% และโครเมียม 20% ซึ่งสามารถทนอุณหภูมิสูงสุดได้ถึง 1400 องศาเซลเซียส โดยมีคุณสมบัติเหนียว และทนความร้อนได้สูงถึง 1400 องศาเซลเซียส

ฮีตเตอร์ถูกแบ่งออกเป็นชนิดต่างๆ ตามลักษณะการใช้งานที่แตกต่างกันดังนี้

ฮีตเตอร์ครีป (Finned Heater)

ฮีตเตอร์ครีปคือ อุปกรณ์ให้ความร้อนกับชิ้นงานโดยใช้หลักการนำพาความร้อน มีครีปช่วยระบายความร้อนออกจากตัวฮีตเตอร์ ครีปมีส่วนทำให้ฮีตเตอร์สามารถใช้วัตต์ได้สูงกว่าฮีตเตอร์แบบไม่มีครีปในพื้นที่ที่เท่ากัน นอกจากนี้ครีปยังทำให้ฮีตเตอร์สามารถถ่ายเทความร้อนได้ไวขึ้นด้วย ฮีตเตอร์แบบครีป เหมาะสำหรับงานที่ต้องการให้ความร้อนในอากาศ มักใช้กับงานอบที่ต้องการความร้อนในการอบชิ้นงานสม่ำเสมอ คงที่ เช่น งานอบสี, อบชิ้นส่วนรถยนต์, อบไล่ความชื้น, อบใยผ้า, อบแม่พิมพ์, อบไม้, อบพลาสติก, อบอาหาร และลดความชื้นในระบบทำความเย็น เป็นต้น ดังแสดงในรูปที่ 2.8



รูปที่ 2.8 ฮีตเตอร์แบบครีป

ฮีตเตอร์คาร์ทริดจ์ (Cartridge Heater)

ฮีตเตอร์คาร์ทริดจ์หรือฮีตเตอร์แท่งคือ อุปกรณ์ให้ความร้อนแก่ชิ้นงานโดยใช้หลักการนำพาความร้อน โดยให้ความร้อนคงที่และสม่ำเสมอ เหมาะสำหรับใช้ในงานอุ่น หรือให้ความร้อนแก่แม่พิมพ์ที่เป็นโลหะ, ชิ้นงานเหล็ก, เครื่องปั๊มทอง, เครื่องรีดถุงพลาสติก, งานขึ้นรูป, งานต้มไม้ และ

สารเคมี เป็นต้น ฮีตเตอร์คาร์ทริดจ์ (Cartridge Heater) ถูกออกแบบมาในรูปแบบแท่ง การนำไปใช้งานจะต้องเจาะช่องเหนือวัตถุ ก่อนใส่ฮีตเตอร์คาร์ทริดจ์เข้าไปเพื่อทำความร้อน ข้อควรระวัง ในการติดตั้งฮีตเตอร์คาร์ทริดจ์ ขนาดของช่องใส่ควรมีขนาดใหญ่กว่าฮีตเตอร์เพียง 0.02-0.04 มม. เพื่อให้การถ่ายเทความร้อนเป็นไปอย่างมีประสิทธิภาพ และช่วยยืดอายุการใช้งาน ดังแสดงในรูปที่ 2.9



รูปที่ 2.9 ฮีตเตอร์แบบคาร์ทริดจ์

ฮีตเตอร์รัดท่อ (Band Heater/Nozzel Heater)

ฮีตเตอร์รัดท่อคือ อุปกรณ์ให้ความร้อนกับชิ้นงานโดยหลักการนำความร้อน ให้ความร้อนสม่ำเสมอ ออกแบบเป็นรูปทรงกระบอก เพื่อใช้สำหรับรัดท่อหรือถังทรงกระบอก เหมาะสำหรับงานที่ไม่ต้องการให้ความร้อนกับวัตถุโดยตรง หรือในสายงานกระบวนการผลิตที่ต้องมีการใช้ความร้อน เช่น เครื่องฉีดพลาสติก, เครื่องเป่าถุง, ท่อลำเลียงขาว, ท่อส่งน้ำมัน, ท่อส่งน้ำร้อน, กระบวนการทำละลายเม็ดพลาสติก หรือ ใช้เป็นส่วนประกอบในการผลิตเครื่องใช้ไฟฟ้า เช่น หม้อหุงข้าว, กระจกน้ำร้อน เป็นต้น ดังแสดงในรูปที่ 2.10



รูปที่ 2.10 ฮีตเตอร์รัดท่อ

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ฮีตเตอร์สแตนเลสตัว I, ตัว U (I-U Stainless Heater)

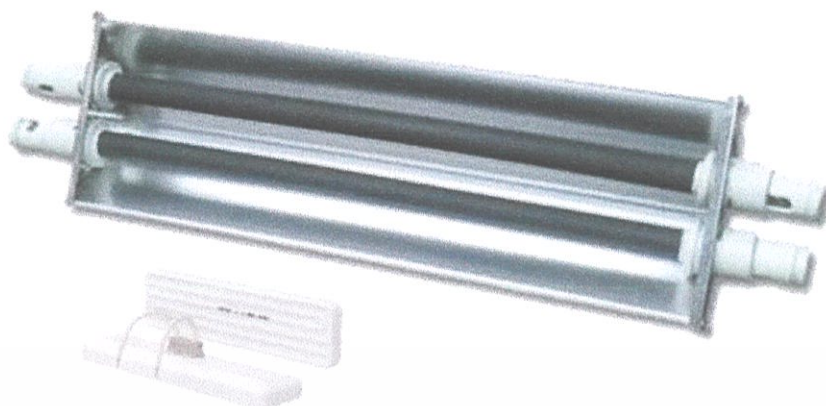
ฮีตเตอร์สแตนเลสตัว I, ตัว U คือ อุปกรณ์ให้ความร้อนกับชิ้นงานทั่วไป ซึ่งเหมาะสำหรับงานอบแห้งทั่วไป เช่น อบสีรถยนต์, พืชผลทางเกษตร, อบเส้นใยผ้า, อบชิ้นส่วนอะไหล่รถยนต์, แม่พิมพ์พลาสติก, แม่พิมพ์โลหะ, ห้องควบคุมความชื้น ที่มีต้องการให้ความร้อนในการอบชิ้นงานอย่างสม่ำเสมอ ฮีตเตอร์สแตนเลสตัว I, ตัว U ที่ถูกคัดสรรมาเป็นอย่างดีของบริษัท แสงชัยมิเตอร์ จำกัด สามารถช่วยให้การให้ความร้อนในกระบวนการอย่างสม่ำเสมอ ลดการเกิดความเสียหายของชิ้นงาน สร้างมาตรฐานคุณภาพให้สินค้า ซึ่งแตกต่างจากฮีตเตอร์สแตนเลสตัว I, ตัว U แบบอื่นในท้องตลาดตรงที่มีขนาด-กำลังวัตต์ให้เลือกใช้หลากหลาย ราคาประหยัด อายุการใช้งานยาวนาน ดังในรูปที่ 2.11



รูปที่ 2.11 ฮีตเตอร์สแตนเลสตัว U

ฮีตเตอร์อินฟราเรด (Infrared Heater)

ฮีตเตอร์อินฟราเรดคือ อุปกรณ์ให้ความร้อนแก่ชิ้นงานโดยวิธีการแผ่รังสีความร้อน ซึ่งแตกต่างจากฮีตเตอร์รูปแบบอื่นๆ เหมาะสำหรับงานให้ความร้อนกับแม่พิมพ์ หรืองานอบไล่ความชื้น เช่น อบสี, อบฟิล์มหด, งานบรรจุหีบห่อ, ชิ้นรูปพลาสติก, อบให้ความร้อน, ซิลฟาคอบ, ซิลคอบขวด, ใช้ในกระบวนการถนอมอาหาร เป็นต้น ฮีตเตอร์อินฟราเรด (Infrared Heater) มีข้อดีคือ วิธีการแผ่รังสีความร้อนควบคุมได้ง่าย ให้ความร้อนคงที่และสม่ำเสมอ ไม่ก่อให้เกิดฝุ่นละอองและมลภาวะ อีกทั้งประหยัดพลังงานไฟฟ้าที่เป็นต้นทุนการผลิตได้สูงถึง 30-50% ดังแสดงในรูปที่ 2.12



รูปที่ 2.12 ฮีตเตอร์อินฟาเรด



เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 3

วิธีการดำเนินงาน

3.1 ขั้นตอนการดำเนินงาน

ขั้นตอนของการดำเนินงานได้วางแผนไว้ดังนี้

1. ศึกษาและทำความเข้าใจถึงหัวข้อของโครงการที่ได้รับมอบหมาย เพื่อให้ทราบถึงที่มา และลักษณะในการทำงาน

2. กำหนดขนาดและอุปกรณ์ที่ต้องใช้ในการทำงานของผู้ควบคุมอุณหภูมิ

3. ออกแบบผู้ควบคุมอุณหภูมิด้วยโปรแกรม SolidWorks

4. ออกแบบตัวควบคุมแบบ PID โดยใช้โปรแกรม Arduino และทดสอบการทำงานของ PID ก่อนใช้งานจริง

5. จัดหาอุปกรณ์ตามขนาดที่กำหนดไว้เพื่อทำการติดตั้งอุปกรณ์

6. ทำการประกอบผู้ควบคุมอุณหภูมิและติดตั้งอุปกรณ์

7. ทดสอบการทำงานของผู้ควบคุมอุณหภูมิและแก้ไขข้อผิดพลาดที่เกิดขึ้น

โดยแผนการดำเนินงานตามทั้ง 7 ข้อจะเป็นไปตาม ตารางที่ 3.1

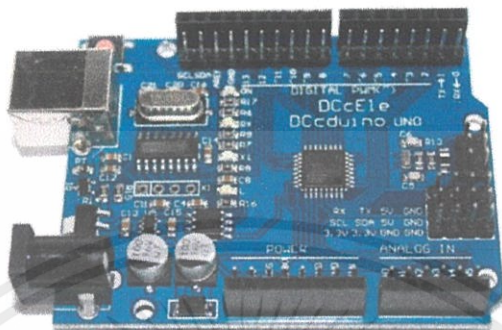
ตารางที่ 3.1 ตารางแสดงแผนการดำเนินงาน

แผนการดำเนินงาน	เดือน							
	สิงหาคม	กันยายน	ตุลาคม	พฤศจิกายน	มกราคม	กุมภาพันธ์	มีนาคม	เมษายน
ข้อที่ 1	■	■						
ข้อที่ 2		■						
ข้อที่ 3			■	■				
ข้อที่ 4				■	■			
ข้อที่ 5					■	■	■	
ข้อที่ 6						■	■	■
ข้อที่ 7							■	■

3.2 อุปกรณ์ที่เกี่ยวข้อง

อุปกรณ์ที่ใช้ในการดำเนินงานมีดังนี้

1. Arduino UNO R3



รูปที่ 3.1 Arduino UNO R3

มีคุณสมบัติดังนี้

- Microcontroller : ATmega328
- VCC : 5V
- Digital I/O : 14 I/O with 6 PWM
- Analog I/O : 6 I/O
- Clock Speed : 16 MHz

2. Digital Temperature Temp Sensor Probe DS18B20



รูปที่ 3.2 Digital Temperature Temp Sensor Probe DS18B20

มีคุณสมบัติดังนี้

- Power Supply Range : 3V to 5.5V
- Adjustable Resolution : 9-12
- Operating Temperature Range : -55°C to +125°C
- Cable Length : 100 cm
- Stainless Steel Tube Size : 6 X 45 mm

3. 1602 LCD (Blue Screen) 16 x 2 LCD with Backlight of The LCD Screen I2C Interface

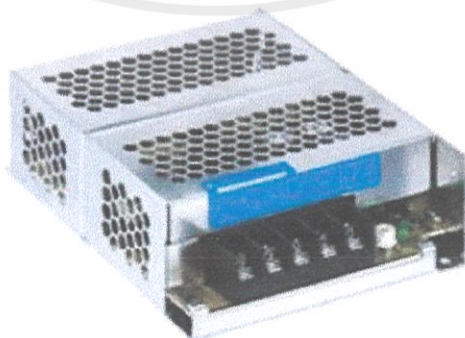


รูปที่ 3.3 1602 LCD (Blue Screen)

มีคุณสมบัติดังนี้

- Supply Voltage : 3.3V-5V
- Characters Displayed : 16 x 2
- Weight : 5g

4. Power Supply 24V AC-DC PMC-24V050W1AA



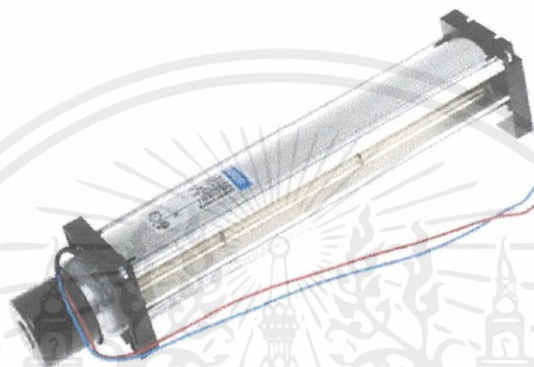
รูปที่ 3.4 Power Supply 24V AC-DC PMC-24V050W1AA

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

มีคุณสมบัติดังนี้

- Output Voltage : 24V
- Output Current : 2.1A
- Input Voltage : 85-264VAC
- Power : 50W

5. DC Tangential Blower

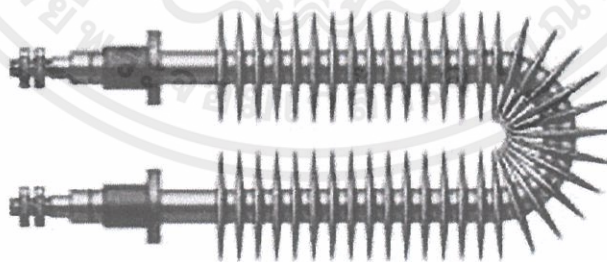


รูปที่ 3.5 DC Tangential Blower

มีคุณสมบัติดังนี้

- Input Voltage : 24V
- Speed : 1600 RPM

6. Finned Tubular Heating



รูปที่ 3.6 Finned Tubular Heating

มีคุณสมบัติดังนี้

- Input Voltage : 220 VAC
- Power : 400W

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

7. Cooling Fan ORIX MU925S-41X



รูปที่ 3.7 Cooling Fan ORIX MU925S-41X

มีคุณสมบัติดังนี้

- Input Voltage : 220VAC
- Ball Bearings
- Impedance Protected

8. Solid State Relay SSR-15DA



รูปที่ 3.8 Solid State Relay SSR-15DA

มีคุณสมบัติดังนี้

- Type : DC to AC
- Input Voltage : 3-32 VDC
- Output Voltage : 24-380 VAC
- Response Time : ≤ 10 ms
- Leakage Current : ≤ 2 mA

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

9. Panasonic Circuit Breaker 30A 2P



รูปที่ 3.9 Panasonic Circuit Breaker 30A 2P

มีคุณสมบัติดังนี้

- Range : 30A 240VAC
- I_c : 1.5kA
- Time Tripped : 0.01s

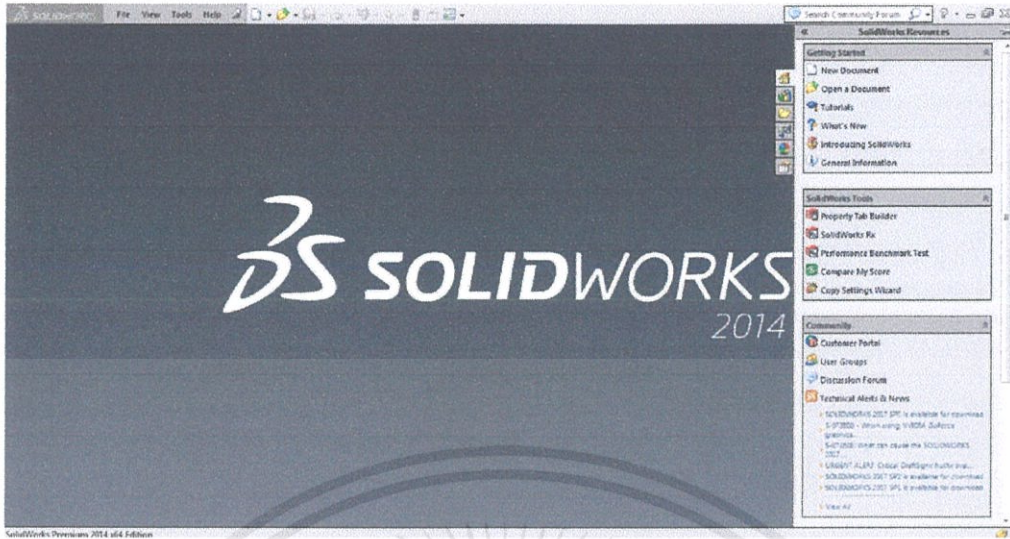
3.3 ซอฟต์แวร์ที่เกี่ยวข้อง

3.3.1 โปรแกรมเขียนและสั่งการทำงาน : Arduino



Arduino

รูปที่ 3.10 ไอคอนโปรแกรม Arduino



รูปที่ 3.13 หน้าต่างโปรแกรม SolidWorks

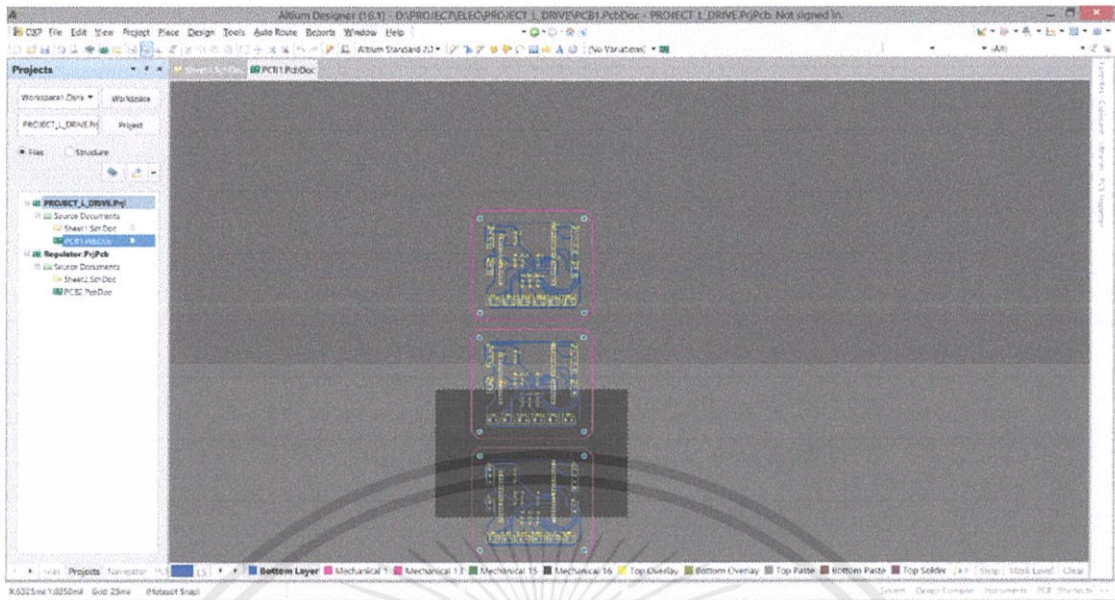
มีคุณสมบัติดังนี้

- ออกแบบอุปกรณ์ทั้ง 3D และ 2D
- รองรับระบบปฏิบัติการ Window7 (32-bits/64-bits) 8/10(64-bits)

3.3.3 โปรแกรมออกแบบ Schematic และ PCB : Altium Designer 2016



รูปที่ 3.14 ไอคอนโปรแกรม Altium Designer 2016



รูปที่ 3.15 หน้าต่างโปรแกรม Altium Designer 2016

มีคุณสมบัติดังนี้

- ซอฟต์แวร์ออกแบบวงจรอิเล็กทรอนิกส์
- รองรับระบบปฏิบัติการ Window7 (32-bits/64-bits) 8/10(64-bits)

3.4 การออกแบบและวางแผนการทำงาน

การออกแบบการทำงานจะแบ่งเป็น 4 อย่างคือ

1. การวางแผนและออกแบบโปรแกรมเพื่อควบคุมลำดับการทำงานของ Arduino โดยมี การวางแผนการทำงานดังนี้
 - กำหนด Input และ Output ต่างๆ เพื่อให้ง่ายและสะดวกในการแก้ไขลำดับการทำงาน ขั้นตอนของ โปรแกรม
 - เลือกใช้คำสั่งบนโปรแกรมที่เหมาะสมและทดสอบการทำงานของโปรแกรม
 - แก้ไขปัญหาของโปรแกรมที่เกิดขึ้นและทำการทดสอบใหม่
2. การวางแผนการออกแบบอุปกรณ์และตู้ควบคุมอุณหภูมิด้วยโปรแกรม SolidWorks
 - กำหนดขนาดและเลือกอุปกรณ์ที่ต้องการ
 - ทำการออกแบบตู้ควบคุมอุณหภูมิ
 - จัดหาอุปกรณ์ตามคุณสมบัติที่กำหนดไว้

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3. ออกแบบบอร์ดวงจรอิเล็กทรอนิกส์สำหรับการติดตั้ง Arduino ด้วยโปรแกรม Altium Designer และการออกแบบลายวงจรอิเล็กทรอนิกส์ลงบนแผ่น PCB

- กำหนดอุปกรณ์ในการออกแบบ
- ทำการออกแบบ Schematic เพื่อออกแบบการเชื่อมต่อระหว่างอุปกรณ์ภายในวงจร
- กำหนดขนาดของแผ่น PCB เพื่อทำการวางอุปกรณ์ลงบนบอร์ดวงจรอิเล็กทรอนิกส์
- วาดลายวงจรลงบนแผ่น PCB ทำการลงอุปกรณ์อิเล็กทรอนิกส์และบอร์ด

Microcontroller

- ทดสอบการทำงานของบอร์ดวงจรอิเล็กทรอนิกส์

4. ประกอบและติดตั้งอุปกรณ์ในแต่ละส่วนเข้าด้วยกันและทดสอบควบคุมอุณหภูมิภายในตู้และแก้ไขปัญหาที่เกิดขึ้น

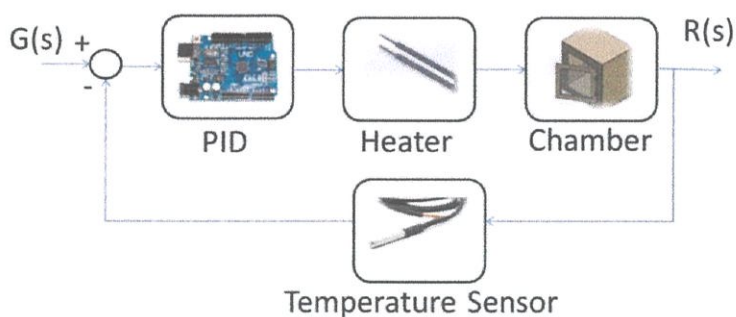
- ประกอบตู้ควบคุมอุณหภูมิ
- ติดตั้งอุปกรณ์ลงบนตู้ควบคุมอุณหภูมิ
- ทดสอบการทำงานของโปรแกรม PID ในการควบคุมอุณหภูมิ
- ปรับจูนค่าพารามิเตอร์
- ทำการทดลองซ้ำจากนั้นทำการเก็บค่าและบันทึกผล

3.5 วิธีการดำเนินงาน

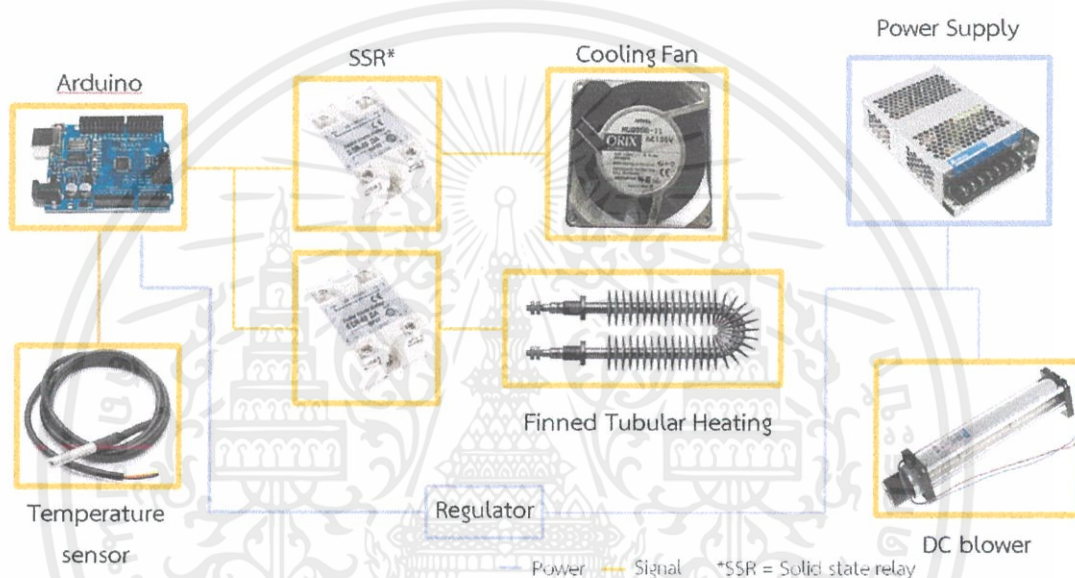
วิธีการดำเนินงานแบ่งออกเป็น 5 กระบวนการ คือ

3.5.1 กระบวนการออกแบบการทำงานของตู้ควบคุมอุณหภูมิ

ในกระบวนการออกแบบและควบคุมการทำงานของตู้ควบคุมอุณหภูมิที่สามารถโปรแกรมได้ ก่อนจะทำการออกแบบตัวควบคุม PID จะต้องทำการกำหนดอุปกรณ์ที่ใช้ในการควบคุมในการรับค่า Input และ Output รวมถึงประเภทของข้อมูลที่นำมาประมวลผลที่เป็นทั้ง Analog และ Digital เช่นกัน ซึ่งการกำหนดการทำงานจำเป็นต้องมีการเขียน Block Diagram ในการดูการทำงานของกระบวนการและคำนวณค่าต่างๆ ส่วนการทำงานการสั่งการของระบบ สามารถนำมาเขียน Data Flow Diagram เพื่อดูการทำงานและประเภทของการรับข้อมูลและส่งข้อมูลในการประมวลผล ดังรูปที่ 3.16 และรูปที่ 3.17



รูปที่ 3.16 Block Diagram ของตู้ควบคุมอุณหภูมิแบบปรับค่าได้



รูปที่ 3.17 Data Flow Diagram ของตู้ควบคุมอุณหภูมิแบบปรับค่าได้

3.5.2 กระบวนการออกแบบการทำงานของโปรแกรมควบคุม

ในส่วนของการออกแบบโปรแกรมเพื่อนำมาควบคุมอุณหภูมิภายในตู้ควบคุมอุณหภูมิที่สามารถปรับค่าได้จะใช้โปรแกรม Arduino ในการเขียนโปรแกรมเพื่อส่งการทำงานไปยังบอร์ดควบคุม Arduino UNO R3 โดยเชื่อมต่อกันด้วย USB port ซึ่งในการออกแบบการทำงานของ Arduino จะแบ่งการออกแบบที่ละส่วนแล้วจึงนำทุกส่วนมาประกอบเข้าด้วยกันซึ่งจะประกอบด้วย 4 ส่วนได้แก่

1. PID Controller

ในส่วนของตัวควบคุมแบบ PID จะใช้การส่งการทำงานออกมาเป็นสมการในการคำนวณหาค่าปรับแก้ ซึ่งในการคำนวณจะอาศัยค่าพารามิเตอร์ในแต่ละเทอมนำมาคำนวณผ่านสมการที่ได้ทำการกำหนดตัวแปรไว้ ซึ่งในการเลือกใช้ตัวแปรจะใช้ตัวแปร Float ซึ่งสามารถเก็บค่าเป็นทศนิยมและ

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

นำมาคำนวณได้ละเอียด ดังรูปที่ 3.18 ส่วนค่าปรับแก้ที่ได้จะถูกนำไปคำนวณในส่วนของฟังก์ชัน PWM

```

SW_CODE_SENSOR_PID_3 | Arduino 1.6.12
File Edit Sketch Tools Help
SW_CODE_SENSOR_PID_3
// PID Controller for a temperature sensor
// PID constants
double Setpoint = 100; // Target temperature
double Error = (Setpoint - T1) - previous; // Error
double Integral = 0; // Integral of error
double Derivative = 0; // Derivative of error
// PID gains
double Kp = 1; // Proportional gain
double Ki = 0.01; // Integral gain
double Kd = 0.1; // Derivative gain
// PID output
double Output = 0; // PWM value
// PID loop
void loop() {
  double Temp = analogRead(A0); // Read temperature sensor
  double Error = Setpoint - Temp; // Calculate error
  Integral = Integral + Error; // Update integral
  Derivative = Error - previous; // Update derivative
  Output = Kp * Error + Ki * Integral + Kd * Derivative; // Calculate output
  previous = Error; // Update previous error
  digitalWrite(LED_BUILTIN, Output > 0); // Turn LED on if output is positive
  delay(100); // Delay 100ms
}
// PID constants
double Setpoint = 100; // Target temperature
double Error = (Setpoint - T1) - previous; // Error
double Integral = 0; // Integral of error
double Derivative = 0; // Derivative of error
// PID gains
double Kp = 1; // Proportional gain
double Ki = 0.01; // Integral gain
double Kd = 0.1; // Derivative gain
// PID output
double Output = 0; // PWM value
// PID loop
void loop() {
  double Temp = analogRead(A0); // Read temperature sensor
  double Error = Setpoint - Temp; // Calculate error
  Integral = Integral + Error; // Update integral
  Derivative = Error - previous; // Update derivative
  Output = Kp * Error + Ki * Integral + Kd * Derivative; // Calculate output
  previous = Error; // Update previous error
  digitalWrite(LED_BUILTIN, Output > 0); // Turn LED on if output is positive
  delay(100); // Delay 100ms
}

```

รูปที่ 3.18 โค้ดการทำงานของโปรแกรม PID

2. Sensor

การทำงานของ Sensor จะถูกควบคุมโดยผ่าน Port Digital Input ของบอร์ด Arduino UNO R3 เพื่อรับค่าของอุณหภูมิภายในตู้ควบคุมอุณหภูมิและนำไปประมวลผล ซึ่ง Sensor ที่ใช้เป็นชนิด One-wire Sensor ในการสั่งการทำงานจึงต้องมีการเรียกใช้ Library โดยใช้คำสั่ง “#include <OneWire.h>” ในการเรียก Library เพื่อทำการรับค่าอุณหภูมิ ส่วนคำสั่งในการทำงาน ถูกแสดงในรูปที่ 3.19



รูปที่ 3.19 Temperature Sensor

```

SW_CODE_SENSOR_PID_3 | Arduino 1.6.12
File Edit Sketch Tools Help
SW_CODE_SENSOR_PID_3
{
  ///////////////SENSOR/////////////////
  sensors.requestTemperatures();

  temp[0] = sensors.getTempC(T_addr[0]);
  temp[1] = sensors.getTempC(T_addr[1]);
  float room = (temp[0] + temp[1]) / 2 ;
  ///////////////PID CONTROL/////////////////

  float position = room ;
  float Error = (Setpoint_VALUE+0.1) - position ;
  if (abs(Error) < IntThresh)
  {
    IntThresh = Integral + Error ;
  }
  else
  {
    Integral = 0;
  }
  double SampleTimeInSec = ((double)SampleTime)/1000 ;
  P = Error * P_VALUE ;
  I = Integral * (I_VALUE*SampleTimeInSec) ;
  D = (LastError - position)*(D_VALUE/SampleTimeInSec) ;
  S_Temp = P + I + D ;

  LastError = position ;
  delay(80);
  if(S_Temp >= 0)
  PWM();

  if (room > Setpoint_VALUE+0.5)
  digitalWrite(3,HIGH);
  if (room < Setpoint_VALUE-0.4)
  digitalWrite(3,LOW);

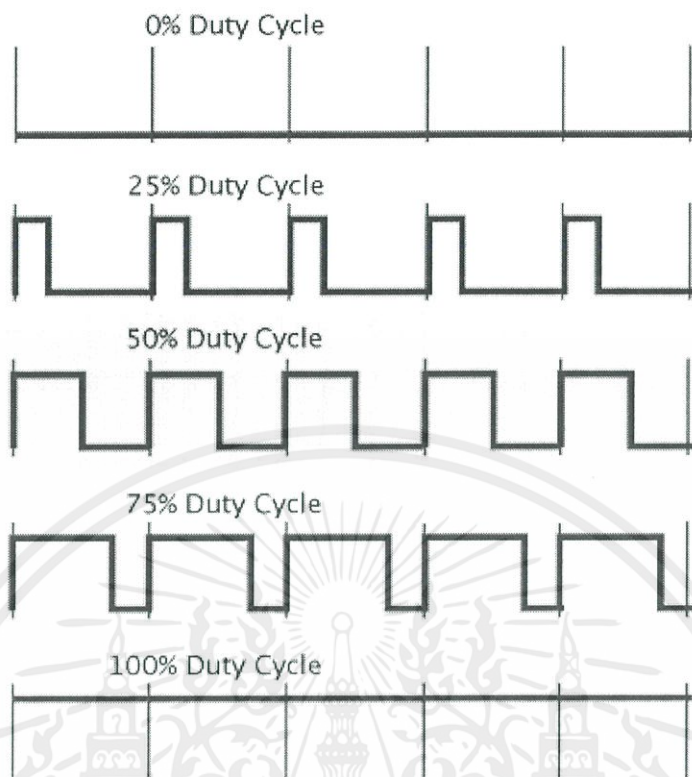
  ///////////////DISPLAY/////////////////
  Serial.println(S_Temp);
  lcd.clear();
  lcd.setCursor(0,0);
  lcd.print("TEMP : ");
  lcd.print(room);
}
120
Arduino-Genuino Ltd. by CC BY

```

รูปที่ 3.20 คำสั่งในการทำงานของ Temperature Sensor

3. PWM (Pulse Width Modulation)

ในการทำงานของส่วนคำสั่ง หรือฟังก์ชัน PWM เพื่อนำไปควบคุม Port Digital Output ในการควบคุมการทำงานของ Solid State Relay เพื่อควบคุมอุณหภูมิของขดลวดความร้อน ซึ่งเป็นอุปกรณ์ให้ความร้อนแก่ตู้ควบคุมอุณหภูมิ การทำงานของฟังก์ชัน PWM ใน Arduino โดยปกติแล้ว Digital Port จะสามารถมีได้แค่ 2 สถานะ คือ HIGH (5V) กับ LOW (0V) เท่านั้น จึงทำให้สร้างสัญญาณลอจิกได้เพียง เปิดหรือปิด ซึ่ง PWM จะถูกนำมาใช้เพื่อให้ Digital Port สามารถเขียนค่าได้มากกว่า HIGH หรือ LOW โดยวิธีการจะใช้การปรับสถานะของสัญญาณลอจิก HIGH/LOW สลับกันไปมาด้วยคาบเวลาหนึ่ง โดยค่าที่ได้นั้นจะขึ้นอยู่กับสัดส่วนเวลาของสัญญาณในช่วงเวลาที่มีสถานะเป็น HIGH และช่วงที่มีสถานะเป็น LOW โดยช่วงเวลาที่แสดงสถานะเป็น HIGH นั้นจะเรียกว่า Pulse Width โดยสัญญาณพัลส์เมื่อเทียบ % ของช่วงเวลาที่ เป็น HIGH (หรือก็คือ % ของ Pulse Width) กับ % ของคาบเวลา (Peroid) ของพัลส์สัญญาณนั้นๆ จะเรียกว่า Duty Cycle ดังรูปที่ 3.21



รูปที่ 3.21 การทำงานของ PWM

ส่วนการใช้งาน PWM เพื่อนำมาควบคุมการเปิด-ปิดของ Solid State Relay กำหนดให้ใช้ค่า Carrier Frequency 10 Hz เพื่อให้ Solid State Relay ตอบสนองทันต่อสัญญาณ PWM แต่เนื่องจากค่าของ Carrier Frequency ที่ได้จาก Arduino มีค่า 490 Hz ซึ่งมากเกินไปที่ต้องการต้องทำการตั้งค่าการทำงานของ ATmega328 ตามคำสั่งการทำงานที่แสดงในรูปที่ 3.22 ถึงรูปที่ 3.23 และรูปที่ 3.24 แสดงสัญญาณ PWM ที่ได้จาก Output ของ Controller

```

SW_CODE_SENSOR_PID_3

void setup()
{
  TCCR1A = (1<<CCM1B1)|(0<<CCM1B0)|(1<<CRM11)|(0<<CRM10);
  TCCR1B = (1<<CRM13)|(1<<CRM12)|(1<<CS12)|(0<<CS11)|(1<<CS10);
  TCCR1C = 0x00;
  TCM1H = TCM1L = 0x00;
  TCR1H = 0x00;
  ICR1L = 0x00;
  //pinMode(12, OUTPUT);
  pinMode(1);
}

WINDOW_STATE = Rp;
TEMP = WINDOW_T;
Serial.begin(9600);
pinMode(0, INPUT_PULLUP);
pinMode(9, INPUT_PULLUP);
pinMode(4, INPUT_PULLUP);
pinMode(5, INPUT_PULLUP);
pinMode(6, INPUT_PULLUP);
pinMode(7, INPUT_PULLUP);
pinMode(8, OUTPUT);
//attachInterrupt(digitalPinToInterrupt(2),sv,FALLING);
//attachInterrupt(digitalPinToInterrupt(3),p,FALLING);

Serial.println("Dallas Temperature Control");
sensors.begin();
sensors.getDeviceCount();

sensors.getAddress(T_addr[0], 0);
sensors.getAddress(T_addr[1], 1);

sensors.getResolution(T_addr[0], 11);
sensors.getResolution(T_addr[1], 11);
// initialize the LCD
LCD.begin();
LCD.setCursor(0, 1);
LCD.println("Hello World!");
LCD.begin();

TCCR1A = (1<<CCM1B1)|(0<<CCM1B0)|(1<<CRM11)|(0<<CRM10);
TCCR1B = (1<<CRM13)|(1<<CRM12)|(1<<CS12)|(0<<CS11)|(1<<CS10);
TCCR1C = 0x00;
TCM1H = TCM1L = 0x00;
TCR1H = 0x00;
ICR1L = 0x00;
}

```

รูปที่ 3.22 คำสั่งตั้งค่า Carrier Frequency ของ PWM ให้มีค่า 10 Hz

```

SW_CODE_SENSOR_PID_3

}

void sv()
{
  if(WINDOW_STATE == Rp)
  {
    WINDOW_STATE = K1;
  }
  else if(WINDOW_STATE == K1)
  {
    WINDOW_STATE = M1;
  }
  else if(WINDOW_STATE == M1)
  {
    WINDOW_STATE = Setpoint;
  }
  else if(WINDOW_STATE == Setpoint)
  {
    WINDOW_STATE = Rp;
  }
  Serial.println("123");
}

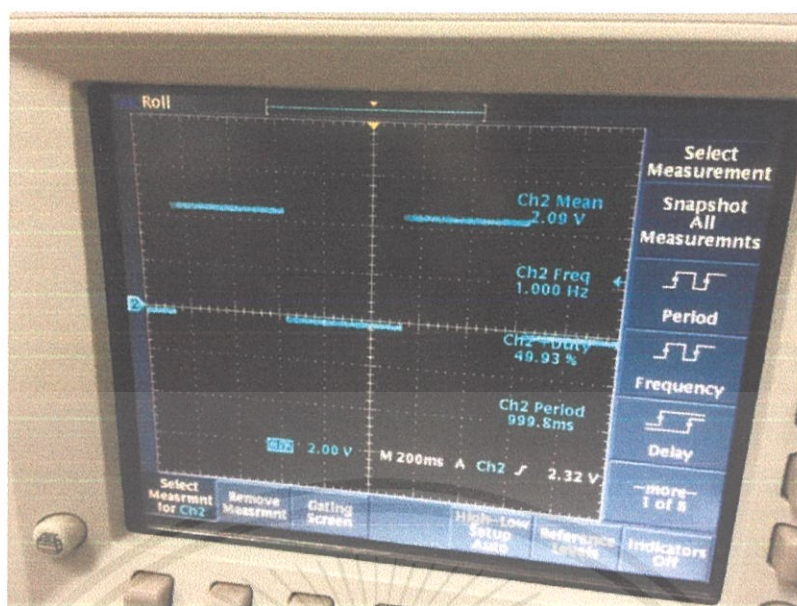
void p1()
{
  if(TEMP == WINDOW_TEMP)
  {
    TEMP = WINDOW_T;
  }
  else if(TEMP == WINDOW_R)
  {
    TEMP = WINDOW_TEMP;
  }
}

void PWM()
{
  unsigned int j = (15625/100)*S_Temp;
  OCR1EL = (j >> 8);
  OCR1L = (j & 0xFF);
}

```

รูปที่ 3.23 ฟังก์ชันการทำงานของ PWM

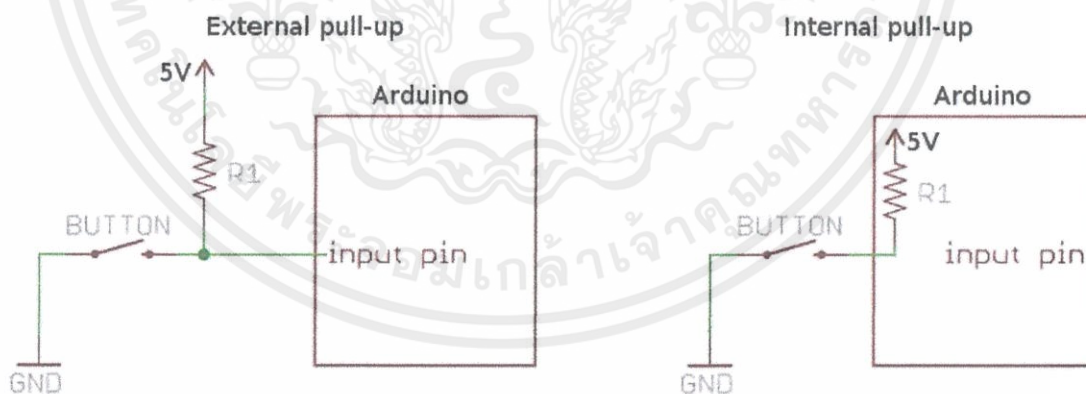
เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.24 สัญญาณ PWM Carrier Frequency 10 Hz

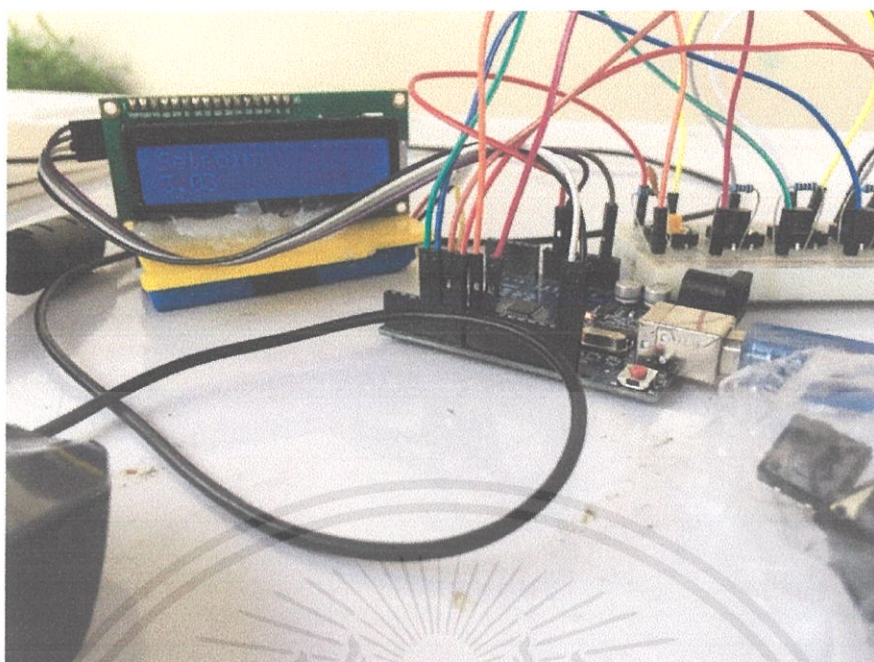
4. Interface

ในส่วนคำสั่ง Interface จะเป็นส่วนคำสั่งในการรับค่าการใช้งานจากผู้ใช้ (User) ซึ่งผู้ใช้งานผู้ควบคุมอุณหภูมิสามารถปรับค่าพารามิเตอร์ผ่านวงจร Interface และสามารถดูค่าสถานะต่างๆ ได้ผ่านหน้าจอ LCD โดยการรับค่าต่างๆ โดยจะใช้คำสั่งเป็น Input PULL-UP เพื่อนำ Push Button Switch มาทำการ Interrupt ในการประมวลคำสั่งไปยัง Arduino ดังรูปที่ 3.25 ถึงรูปที่ 3.27

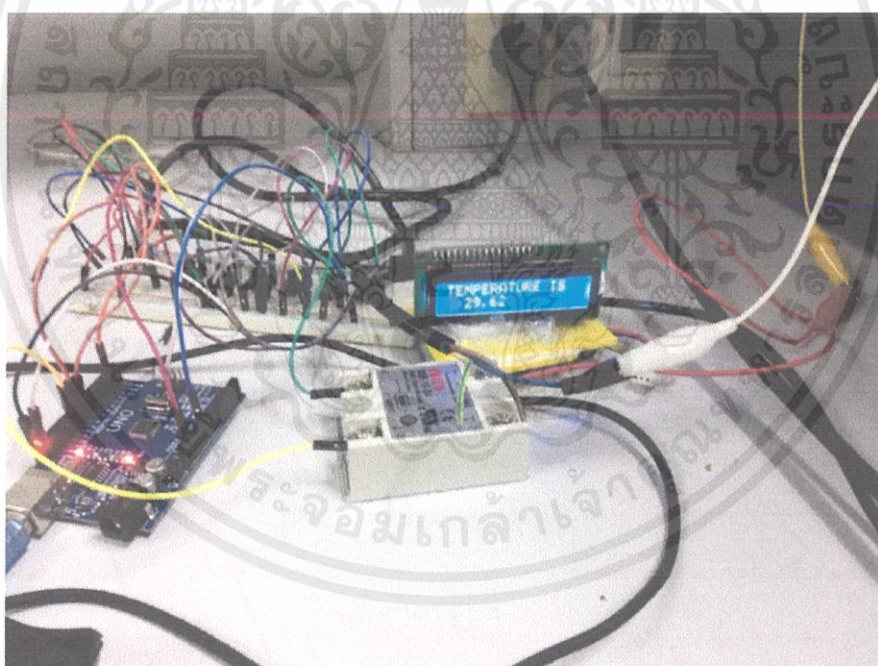


รูปที่ 3.25 การทำงานของ PULL-UP

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.26 ทดลองแสดงผลการทำงานผ่านหน้าจอ LCD (1)

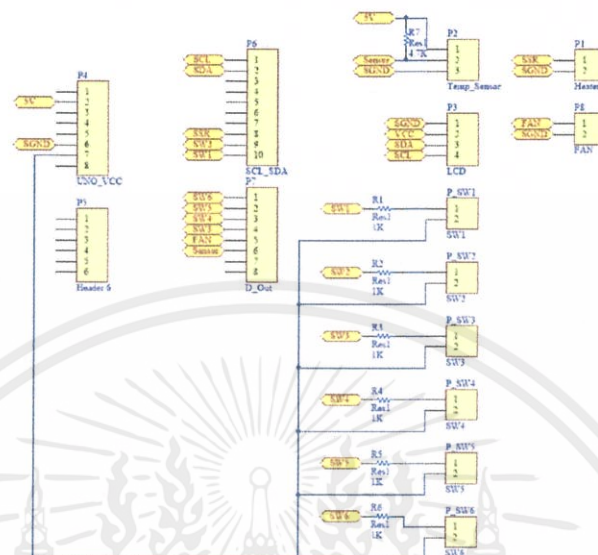


รูปที่ 3.27 ทดลองแสดงผลการทำงานผ่านหน้าจอ LCD (2)

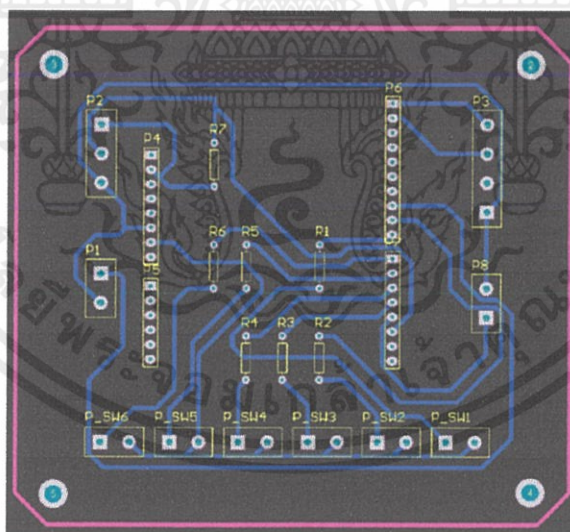
3.5.3 การออกแบบวงจร Interface

ในการออกแบบวงจร Interface สำหรับการนำบอร์ดวงจรอิเล็กทรอนิกส์ไปติดตั้งจะใช้โปรแกรม Altium Designer 2016 ในการออกแบบ โดยโปรแกรมจะสามารถเขียนแบบ Schematic ดังรูปที่ 3.28 เพื่อใช้ในการเชื่อมต่ออุปกรณ์ต่างๆ ภายในบอร์ดวงจรอิเล็กทรอนิกส์ และวาดลายวงจร เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่นอนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ลงบนแผ่นทองแดง PCB ดังรูปที่ 3.29 เพื่อนำไปทำแผ่นวงจร PCB ในการติดตั้งอุปกรณ์อิเล็กทรอนิกส์ได้



รูปที่ 3.28 Schematic ของวงจร Interface ที่เขียนโดยโปรแกรม Altium Designer 2016

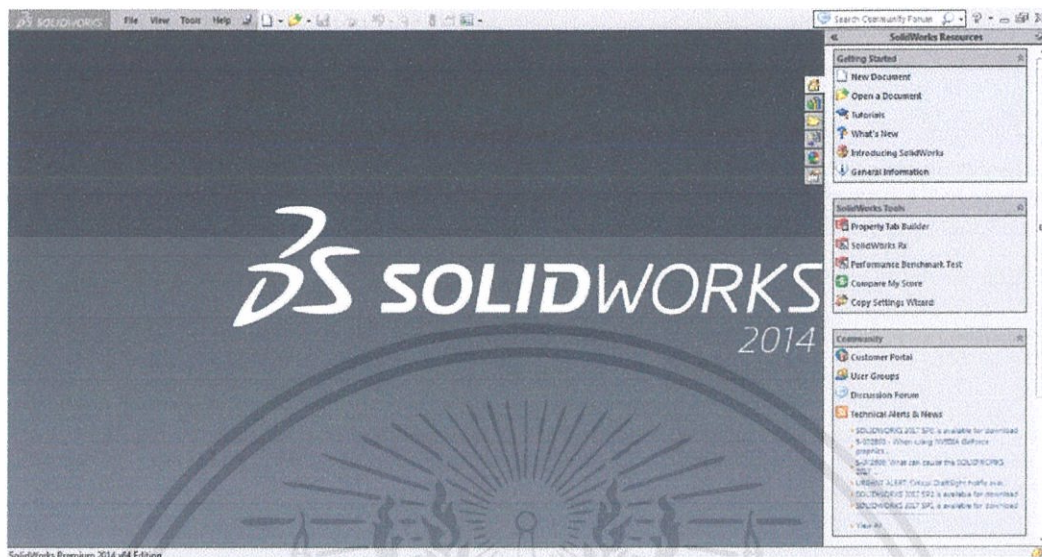


รูปที่ 3.29 ลายวงจร ของวงจร Interface ที่เขียนโดยโปรแกรม Altium Designer 2016

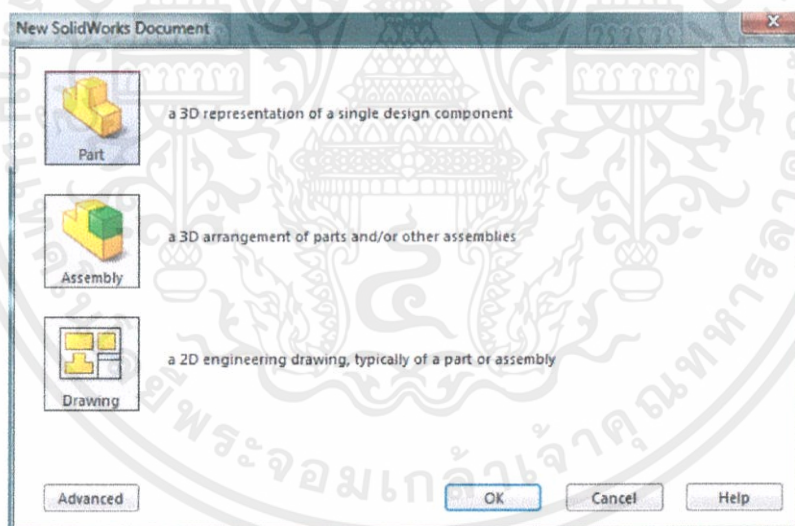
3.5.4 กระบวนการออกแบบตู้ควบคุมอุณหภูมิ โดยโปรแกรม SolidWorks

ใช้โปรแกรม SolidWorks ในการออกแบบตู้ควบคุมอุณหภูมิ และใช้ในการจัดวางตำแหน่งอุปกรณ์ รวมถึงการกำหนดขนาดของตู้ควบคุมอุณหภูมิและอุปกรณ์ทั้งหมดที่ใช้ภายในตู้ โดยเครื่องมือในหมวด Sketch และ Feature ในการออกแบบ โดยมีอุปกรณ์บางตัวที่ต้องใช้แบบ Sketch เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ในการระบุประเภทให้ได้ตามที่เราร้องการ เช่น ขดลวดความร้อนแบบครีบ แสดงดังรูปที่ 3.30 ถึงรูปที่ 3.36

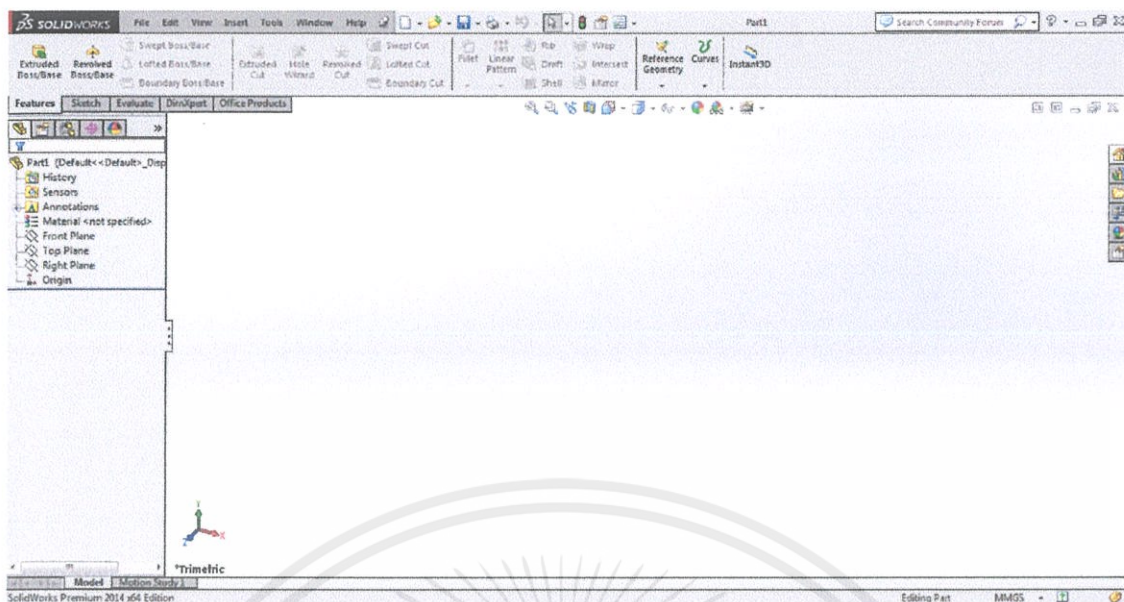


รูปที่ 3.30 หน้าจอการทำงานของโปรแกรม SolidWorks (1)

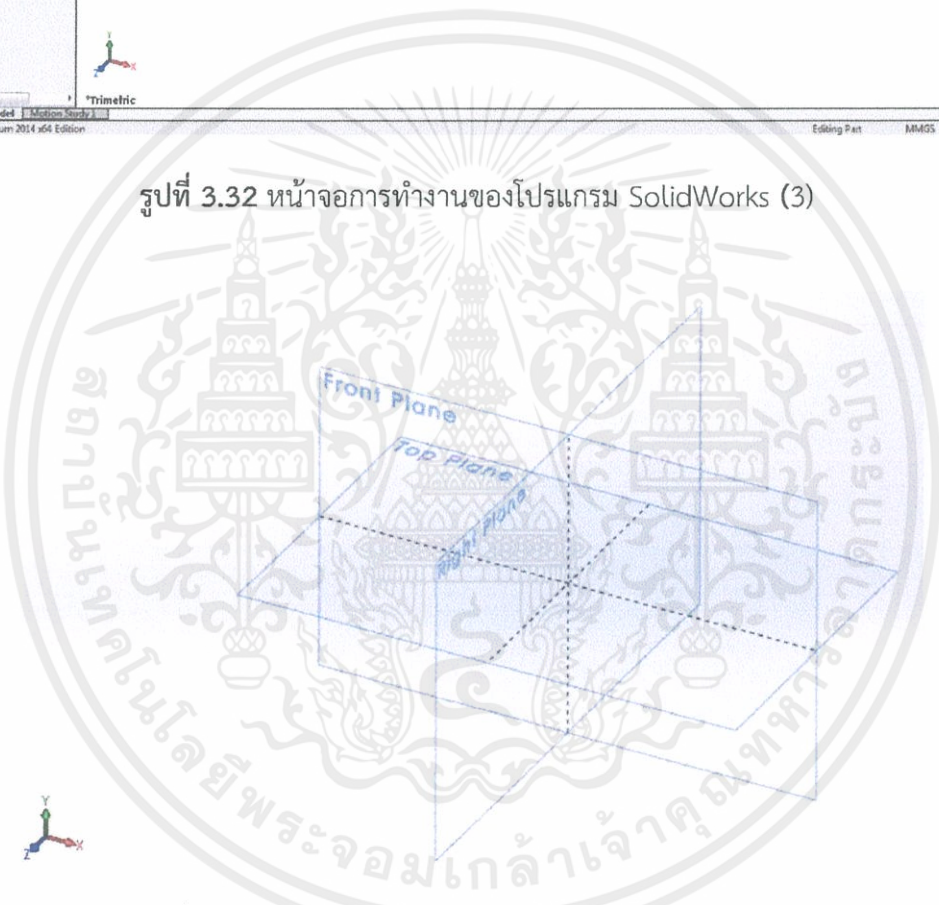


รูปที่ 3.31 หน้าจอการทำงานของโปรแกรม SolidWorks (2)

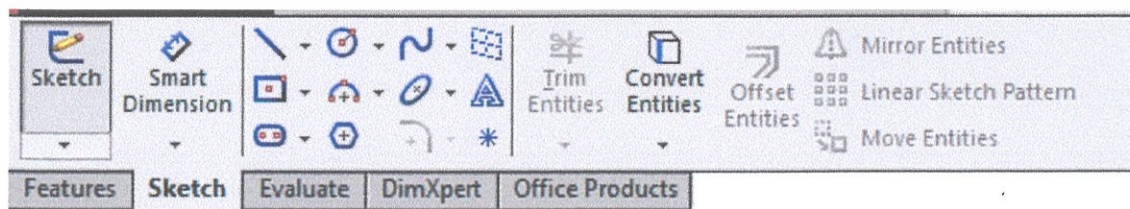
เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.32 หน้าจอการทำงานของโปรแกรม SolidWorks (3)

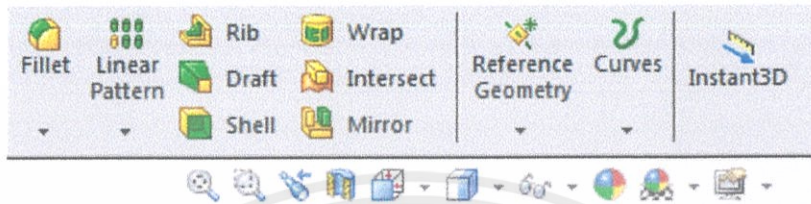
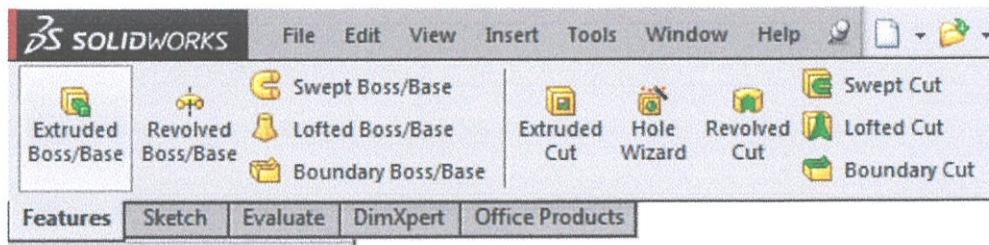


รูปที่ 3.33 หน้าจอการทำงานของโปรแกรม SolidWorks (4)

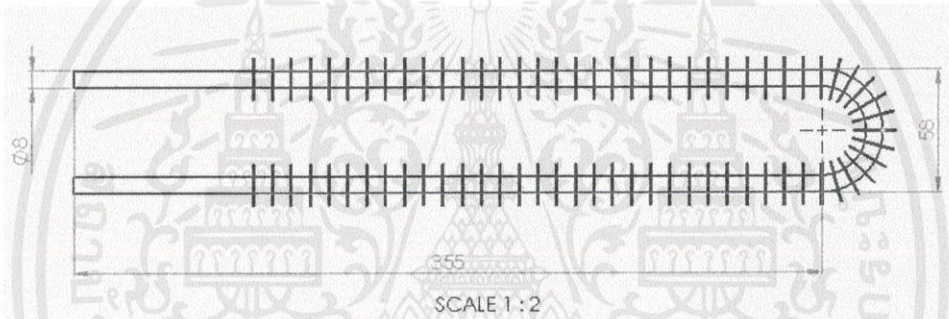


รูปที่ 3.34 เครื่องมือที่ใช้ในการออกแบบในโปรแกรม SolidWorks (1)

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



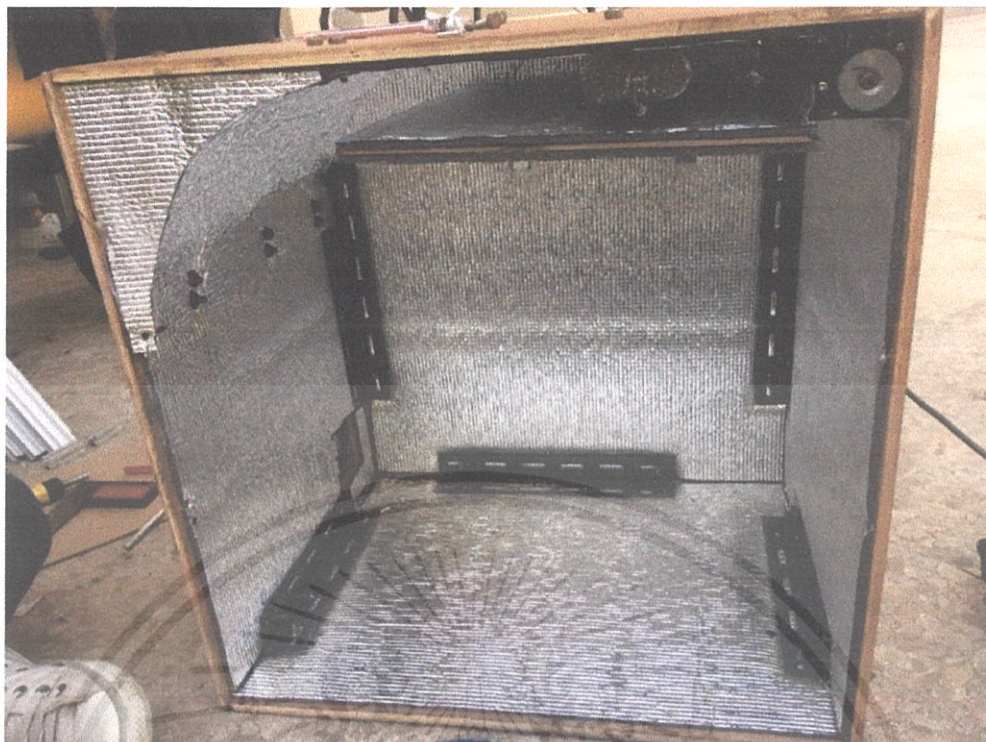
รูปที่ 3.35 เครื่องมือที่ใช้ในการออกแบบในโปรแกรม Solidworks (2)



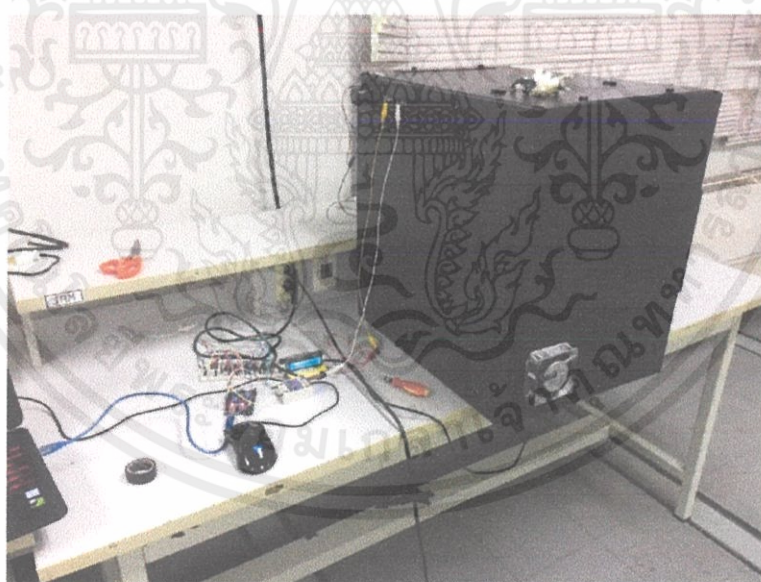
รูปที่ 3.36 แบบ 2D ของขดลวดความร้อนแบบครีป

3.5.5 กระบวนการประกอบและติดตั้งเพื่อทดสอบการทำงาน

กระบวนการประกอบเพื่อทดสอบการทำงานนั้นจะทำให้สามารถรู้ถึงข้อผิดพลาดที่จะเกิดขึ้นได้ในการออกแบบ และควบคุมการทำงานของตัวควบคุมอุณหภูมิที่สามารถโปรแกรมได้ เพื่อที่จะสามารถแก้ไขปัญหาได้อย่างเหมาะสม ดังรูปที่ 3.37 ถึงรูปที่ 3.40

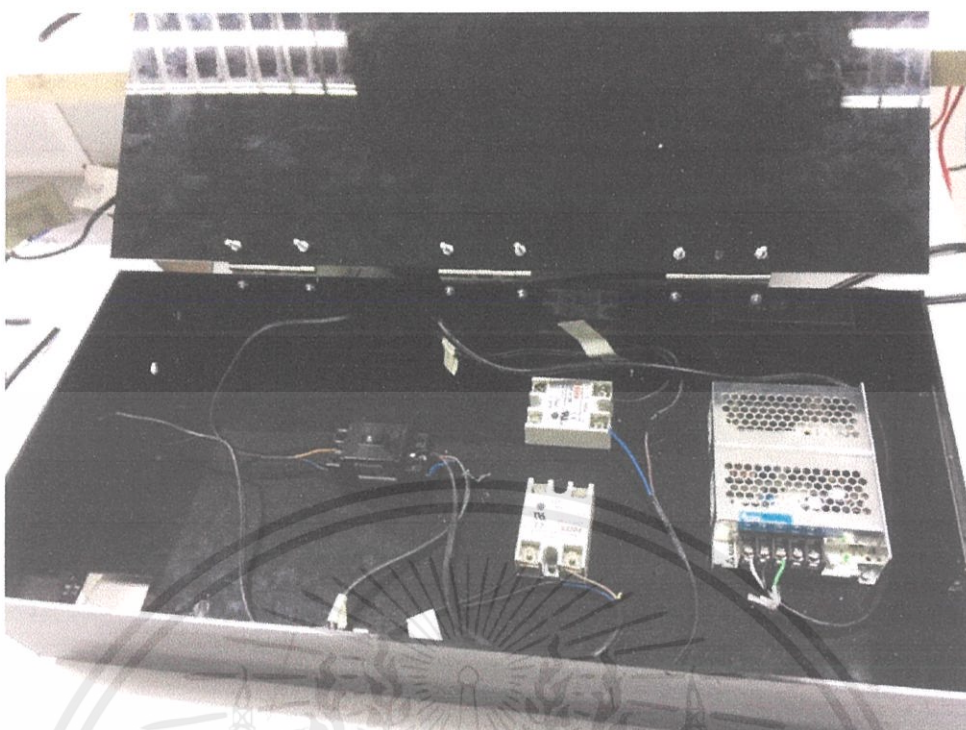


รูปที่ 3.37 ประกอบและติดตั้งอุปกรณ์ภายในตู้



รูปที่ 3.38 ทดสอบการทำงาน

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.39 วงจร Power สำหรับตู้ควบคุมอุณหภูมิ



รูปที่ 3.40 ตู้ควบคุมอุณหภูมิแบบที่สามารถโปรแกรมได้เมื่อประกอบเสร็จสิ้น

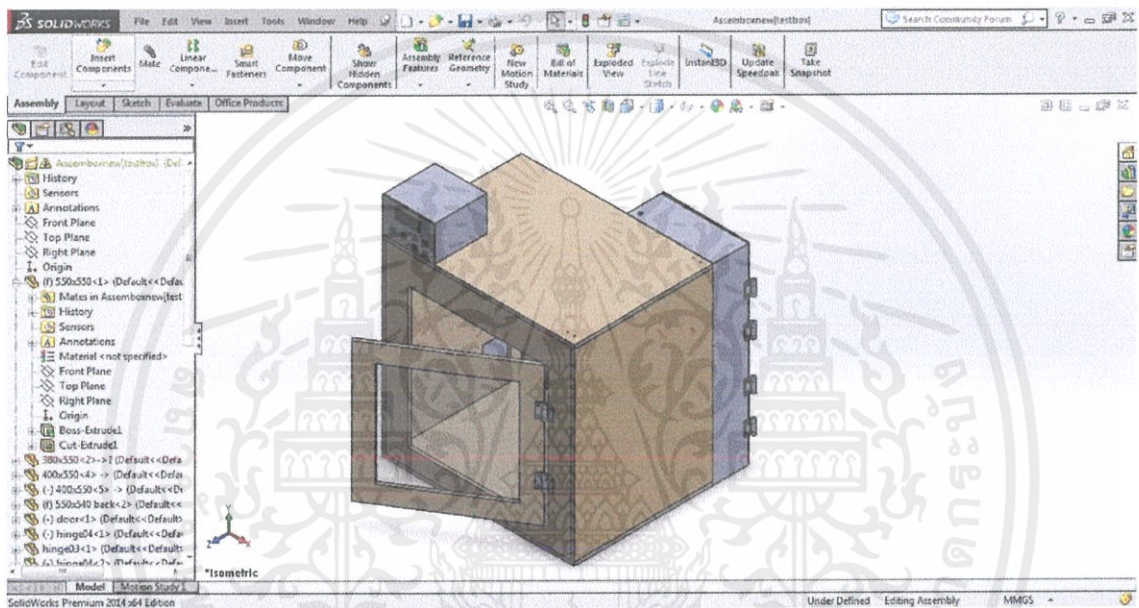
เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 4

ผลการดำเนินงาน

4.1 การออกแบบตู้ควบคุมอุณหภูมิที่สามารถโปรแกรมได้

จากการออกแบบตู้ควบคุมอุณหภูมิที่สามารถโปรแกรมได้โดยโปรแกรม SolidWorks ได้ผลลัพธ์รูปแบบของตู้ควบคุมอุณหภูมิเป็นดังที่แสดงในรูปที่ 4.1



รูปที่ 4.1 การออกแบบของตู้ควบคุมอุณหภูมิที่สามารถโปรแกรมได้ด้วยโปรแกรม SolidWorks

4.2 การออกแบบโปรแกรมควบคุม PID

จากการออกแบบโปรแกรมควบคุมแบบ PID ในโปรแกรม Arduino ได้ผลลัพธ์ของโปรแกรมเป็นดังนี้

```

void setup()
{
  TCCR1A = (1<<COM1B1) | (0<<COM1B0) | (1<<WGM11) | (0<<WGM10);
  TCCR1B = (1<<WGM13) | (1<<WGM12) | (1<<CS12) | (0<<CS11) | (1<<CS10);
  TCCR1C = 0x00;
  ICR1H = ICR1L = 0x00;
  ICR1H = 0x3D;
  ICR1L = 0x08;
  //pinMode(10 ,OUTPUT);
  DDRB |= (1<<FB2);

  WINDOW_STATE = Rp;
  TEMP = WINDOW_K;
  Serial.begin(9600);
  pinMode(8, INPUT_PULLUP);
  pinMode(9, INPUT_PULLUP);
  pinMode(4, INPUT_PULLUP);
  pinMode(5, INPUT_PULLUP);
  pinMode(6, INPUT_PULLUP);
  pinMode(7, INPUT_PULLUP);
  pinMode(3, OUTPUT);
  //attachInterrupt(digitalPinToInterrupt(2),sw,FALLING);
  //attachInterrupt(digitalPinToInterrupt(3),p,FALLING);

  Serial.println("Dallas Temperature Control");
  sensors.begin();
  sensors.getAddress(I_addr[2]);

  sensors.getAddress(I_addr[0], 0);
  sensors.getAddress(I_addr[1], 1);

  sensors.setResolution(I_addr[0], 11);
  sensors.setResolution(I_addr[1], 11);
  // initialize the LCD
  lcd.begin();
  lcd.backlight();
  lcd.print("Hello World");
  Wire.begin();

  lcd(TEMP == WINDOW_IEMP)
  {
    ///////////SENSOR//////////
    sensors.requestTemperatures();

    temp[0] = sensors.getTemp(I_addr[0]);
    temp[1] = sensors.getTemp(I_addr[1]);
    float room = (temp[0] + temp[1]) / 2;

    ///////////PID CONTROL//////////

    float position = room;
    float Error = (Setpoint_VALUE+0.1) - position;
    if (abs(Error) < IntThresh)
    {
      IntThresh = Integral + Error;
    }
    else
    {
      Integral = 0;
    }
    double SampleTimeInSec = ((double)SampleTime)/1000;
    P = Error * P_VALUE;
    I = Integral * (I_VALUE*SampleTimeInSec);
    D = (LastError - position)*(D_VALUE/SampleTimeInSec);
    S_Temp = P + I + D;

    LastError = position;
    delay(50);
    if(S_Temp >= 0)
      PWM();

    if (room > Setpoint_VALUE+0.5)
      digitalWrite(3,HIGH);
    if (room < Setpoint_VALUE-0.4)
      digitalWrite(3,LOW);
  }
}

void loop()
{
  //P_VALUE = EEPROM.read(0);
  //I_VALUE = EEPROM.read(1);
  //D_VALUE = EEPROM.read(2);
  //Setpoint_VALUE = EEPROM.read(3);
  //if(digitalRead(2)==0)
  {
    // lcd.clear();
    // lcd.print("Hello");
  }
  delay(10);
  if(digitalRead(9)==0)
  {
    p();
    delay(100);
  }
  if(digitalRead(8)==0)
  {
    sw();

    delay(100);
  }

  if(TEMP == WINDOW_IEMP)
  {

```

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้


```

void sw()
{
  if(WINDOW_STATE == Kp)
  {
    WINDOW_STATE = Ki ;
  }
  else if(WINDOW_STATE == Ki)
  {
    WINDOW_STATE = Kd ;
  }
  else if(WINDOW_STATE == Kd)
  {
    WINDOW_STATE = Setpoint ;
  }
  else if(WINDOW_STATE == Setpoint)
  {
    WINDOW_STATE = Kp ;
  }
  Serial.println("123");
}

void p()
{
  if(TEMP == WINDOW_TEMP)
  {
    TEMP = WINDOW_K ;
  }
  else if(TEMP = WINDOW_K)
  {
    TEMP = WINDOW_TEMP ;
  }
}

void PWM()
{
  unsigned int j = (15624/100)*S_Temp ;

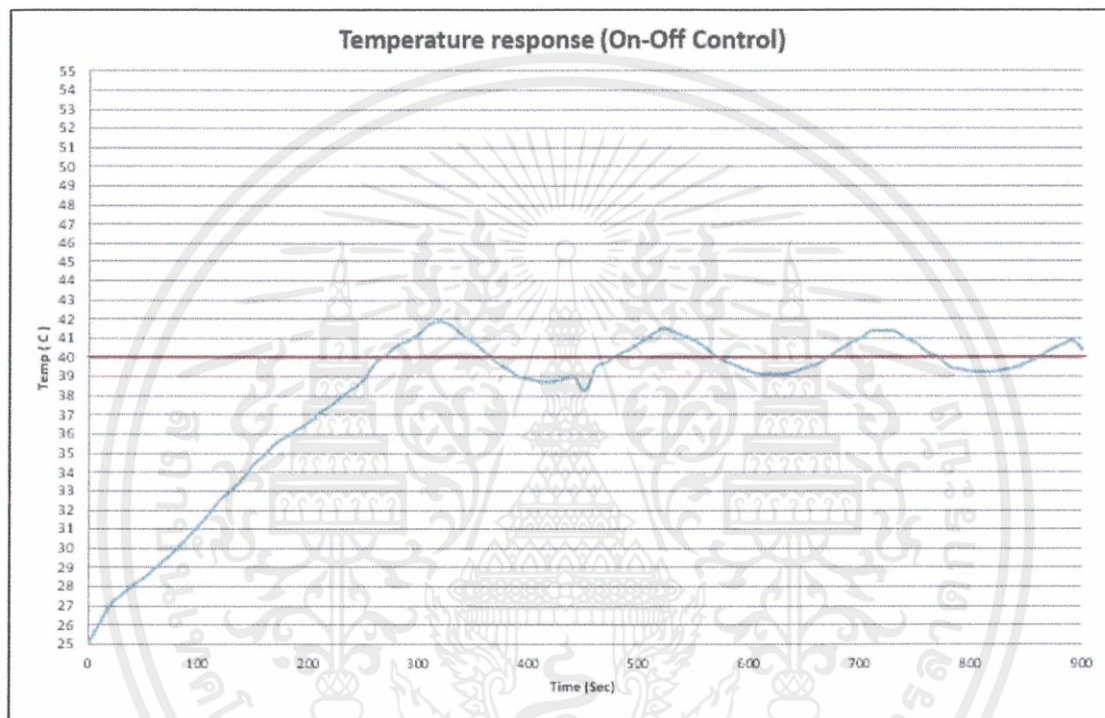
  OCR1BH = (j >> 2);
  OCR1BL = (j & 0xFF);
}

```

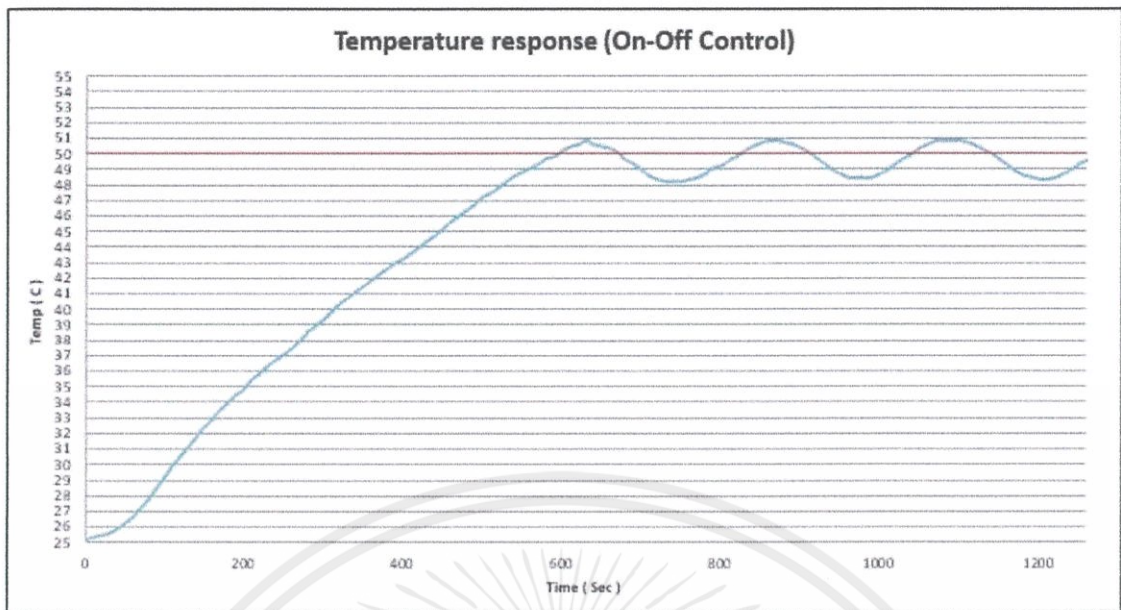
เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.3 ผลการทดสอบการทำงานของตัวควบคุมอุณหภูมิที่สามารถโปรแกรมได้โดยใช้ตัวควบคุมแบบ On-Off Controller

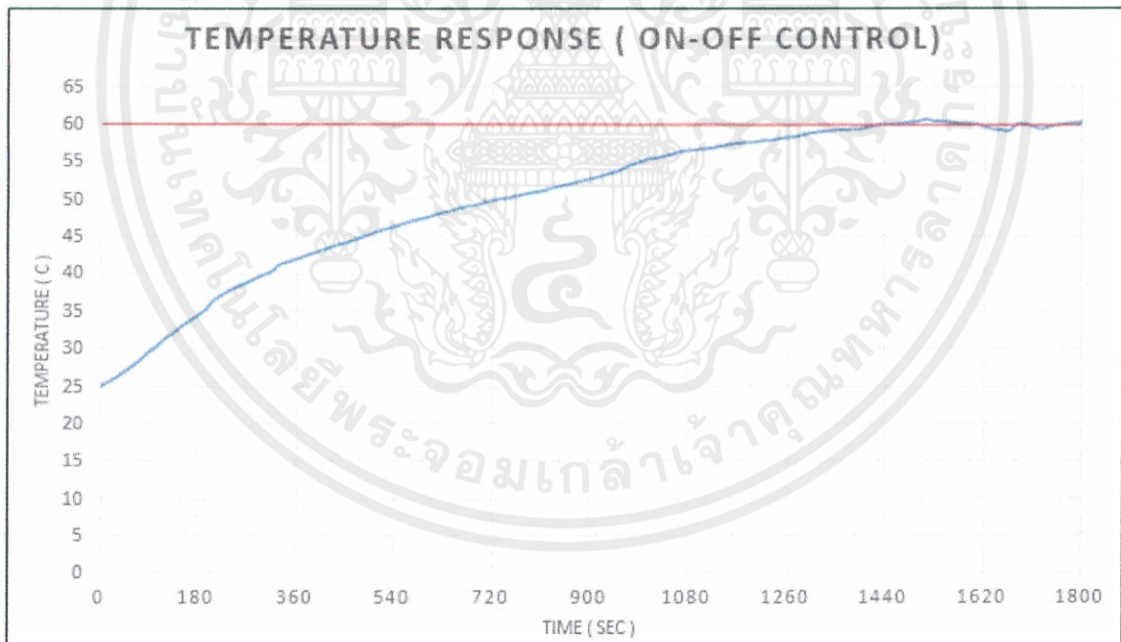
จากการทดสอบการทำงานของตัวควบคุมอุณหภูมิที่สามารถโปรแกรมได้โดยใช้ตัวควบคุมแบบ On-Off Controller ทำให้เห็นถึงประสิทธิภาพของการควบคุมอุณหภูมิภายในตู้ โดยการกำหนดค่าอุณหภูมิห้องที่ 25°C และทำการตั้งค่าเป้าหมายที่ 40°C , 50°C และ 60°C ซึ่งผลทดสอบแสดงดังรูปที่ 4.2 ถึงรูปที่ 4.4 ตามลำดับ



รูปที่ 4.2 ผลการทดสอบการทำงานของตัวควบคุมอุณหภูมิที่สามารถโปรแกรมได้โดยใช้ตัวควบคุมแบบ On-Off Controller ที่อุณหภูมิเป้าหมาย 40°C (อุณหภูมิห้อง 25°C)



รูปที่ 4.3 ผลการทดสอบการทำงานของตัวควบคุมอุณหภูมิที่สามารถโปรแกรมได้ โดยใช้ตัวควบคุมแบบ On-Off Controller ที่อุณหภูมิเป้าหมาย 50°C (อุณหภูมิห้อง 25°C)

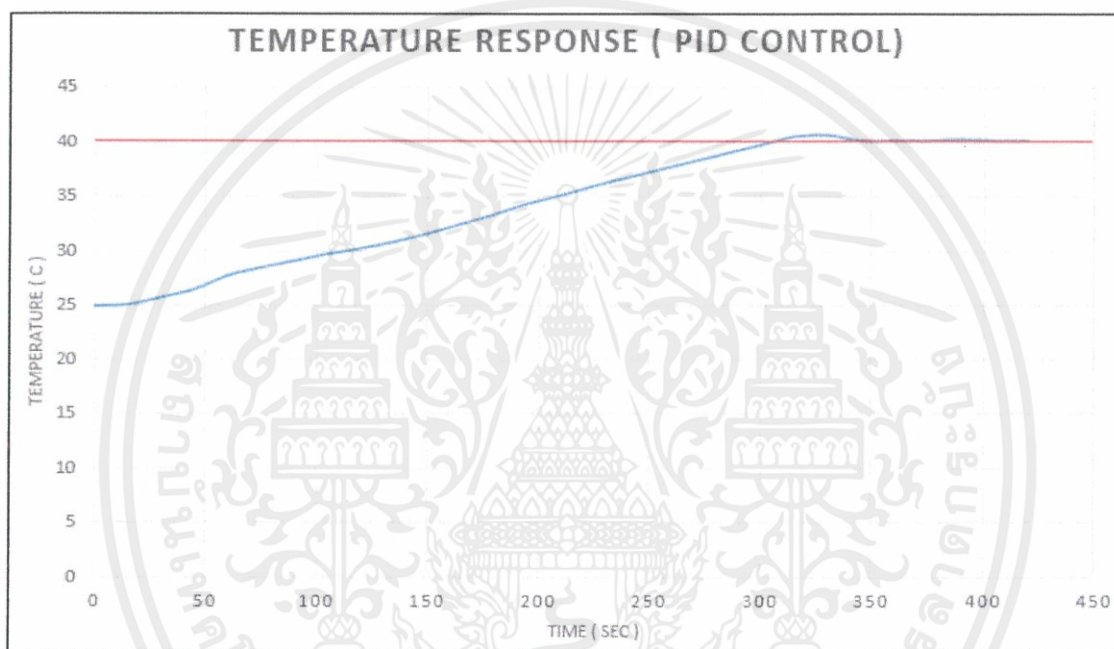


รูปที่ 4.4 ผลการทดสอบการทำงานของตัวควบคุมอุณหภูมิที่สามารถโปรแกรมได้ โดยใช้ตัวควบคุมแบบ On-Off Controller ที่อุณหภูมิเป้าหมาย 60°C (อุณหภูมิห้อง 25°C)

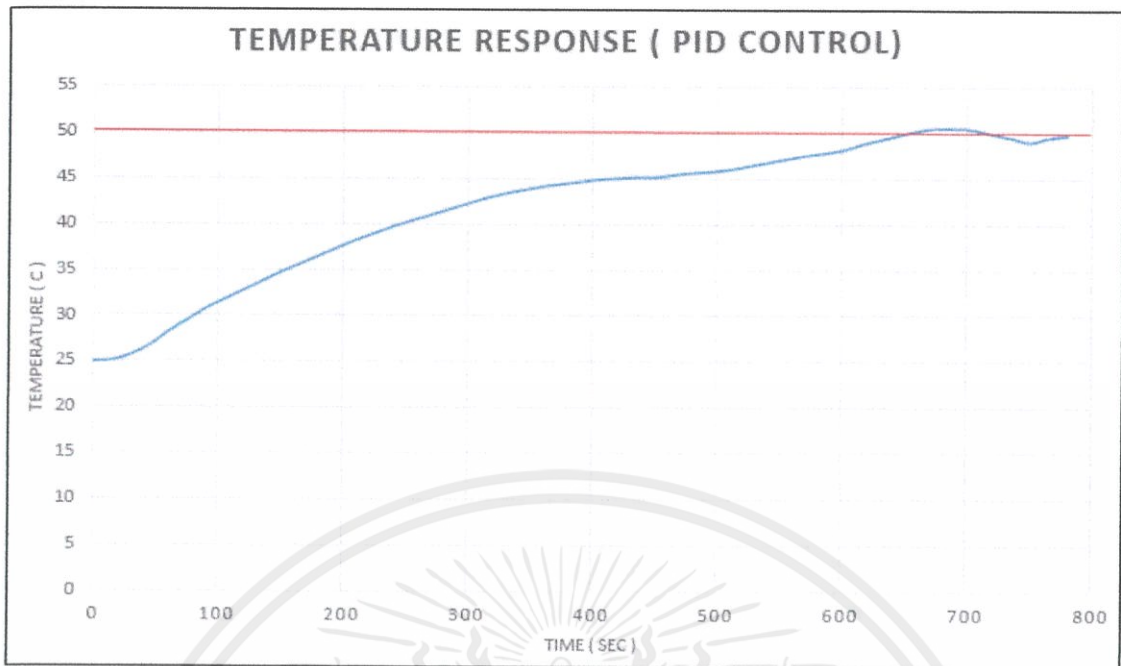
เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.4 ผลการทดสอบการทำงานของตัวควบคุมอุณหภูมิที่สามารถโปรแกรมได้โดยใช้ตัวควบคุมแบบ PID Controller

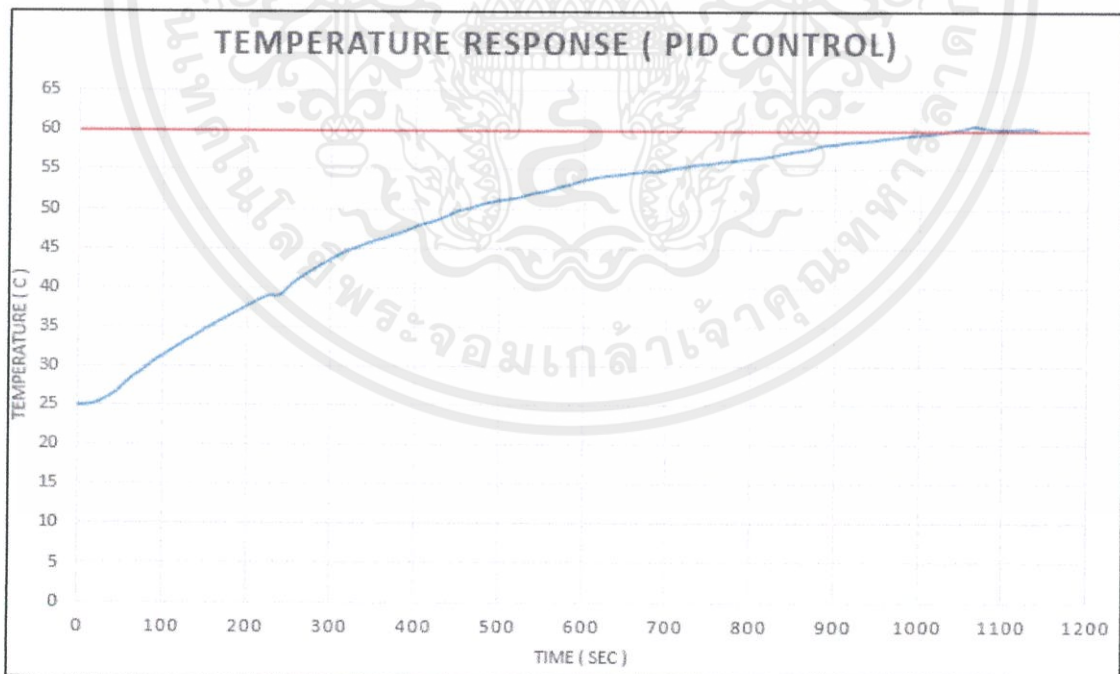
จากการทดสอบการทำงานของตัวควบคุมอุณหภูมิที่สามารถโปรแกรมได้โดยใช้ตัวควบคุมแบบ PID Controller ทำให้เห็นถึงประสิทธิภาพของการควบคุมอุณหภูมิภายในตู้โดยกำหนดให้อุณหภูมิห้องที่ 25°C และกำหนดค่าเป้าหมาย 3 ค่า คือ 40°C , 50°C และ 60°C ในการทดสอบตัวควบคุมแบบ PID ได้ทำการทดสอบโดยกำหนดค่าพารามิเตอร์เป็น 2 ชุด คือ ($K_p = 80$, $K_i = 15$, $K_d = 0$) และ ($K_p = 100$, $K_i = 15$, $K_d = 1$) โดยผลการทดสอบแสดงดังรูปที่ 4.5 ถึงรูปที่ 4.10



รูปที่ 4.5 ผลการทดสอบการทำงานของตัวควบคุมอุณหภูมิที่สามารถโปรแกรมได้ โดยใช้ตัวควบคุมแบบ PID Controller ที่อุณหภูมิเป้าหมาย 40°C (ที่อุณหภูมิห้อง 25°C , $K_p = 80$, $K_i = 15$, $K_d = 0$)

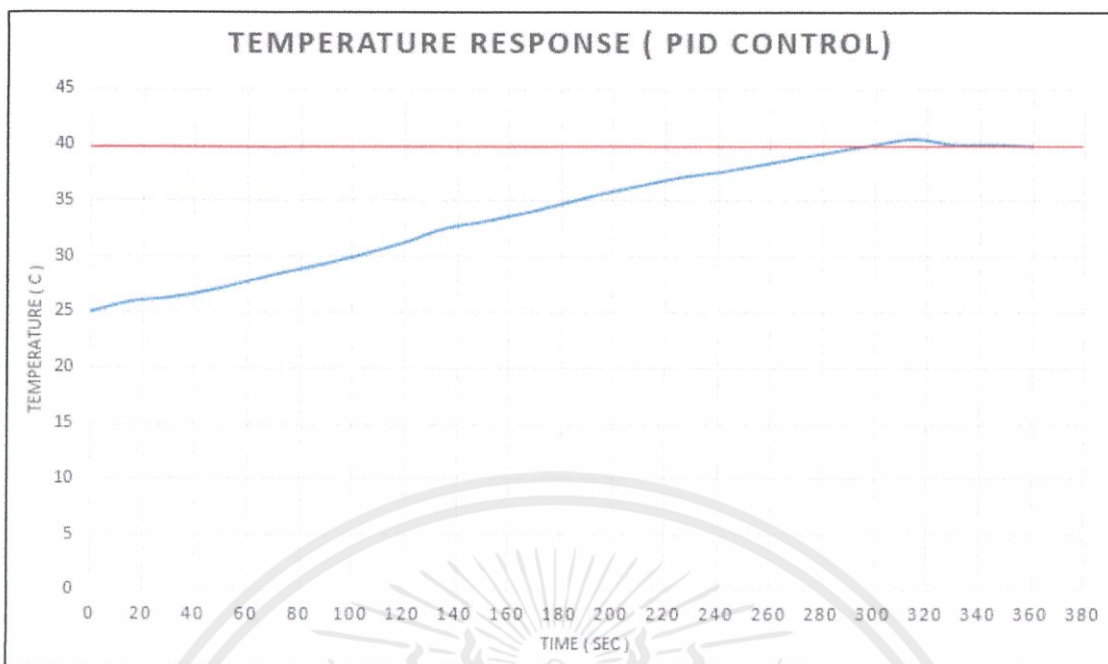


รูปที่ 4.6 ผลการทดสอบการทำงานของตัวควบคุมอุณหภูมิที่สามารถโปรแกรมได้ โดยใช้ตัวควบคุมแบบ PID Controller ที่อุณหภูมิเป้าหมาย 50 °C (ที่อุณหภูมิห้อง 25 °C , $K_p = 80$, $K_i = 15$, $K_d = 0$)

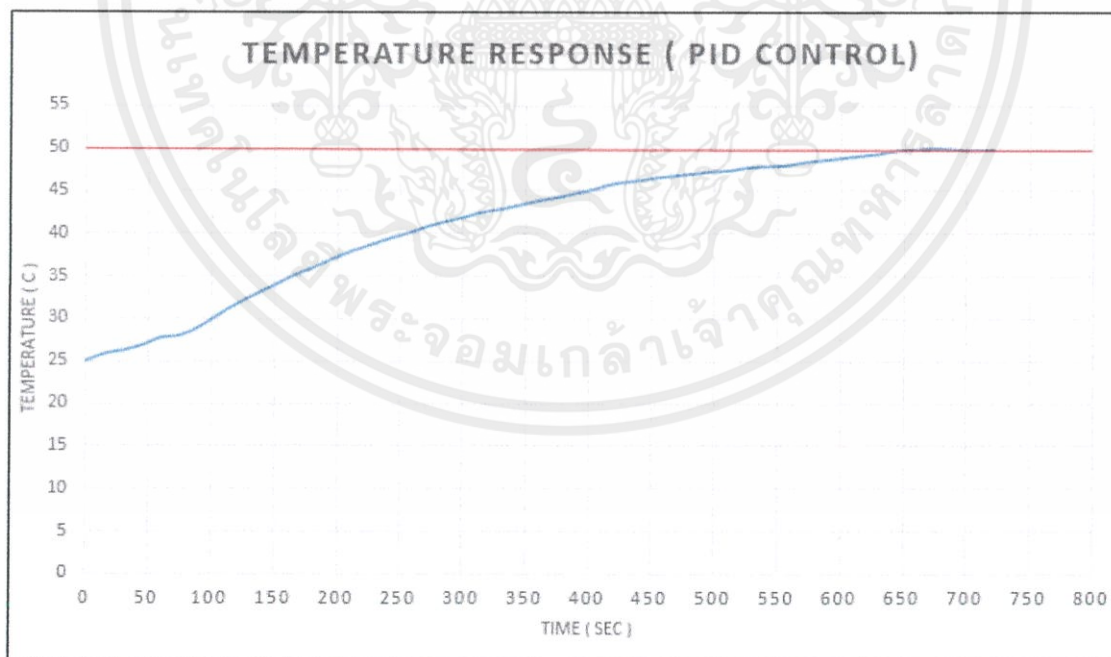


รูปที่ 4.7 ผลการทดสอบการทำงานของตัวควบคุมอุณหภูมิที่สามารถโปรแกรมได้ โดยใช้ตัวควบคุมแบบ PID Controller ที่อุณหภูมิเป้าหมาย 60 °C (ที่อุณหภูมิห้อง 25 °C , $K_p = 80$, $K_i = 15$, $K_d = 0$)

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

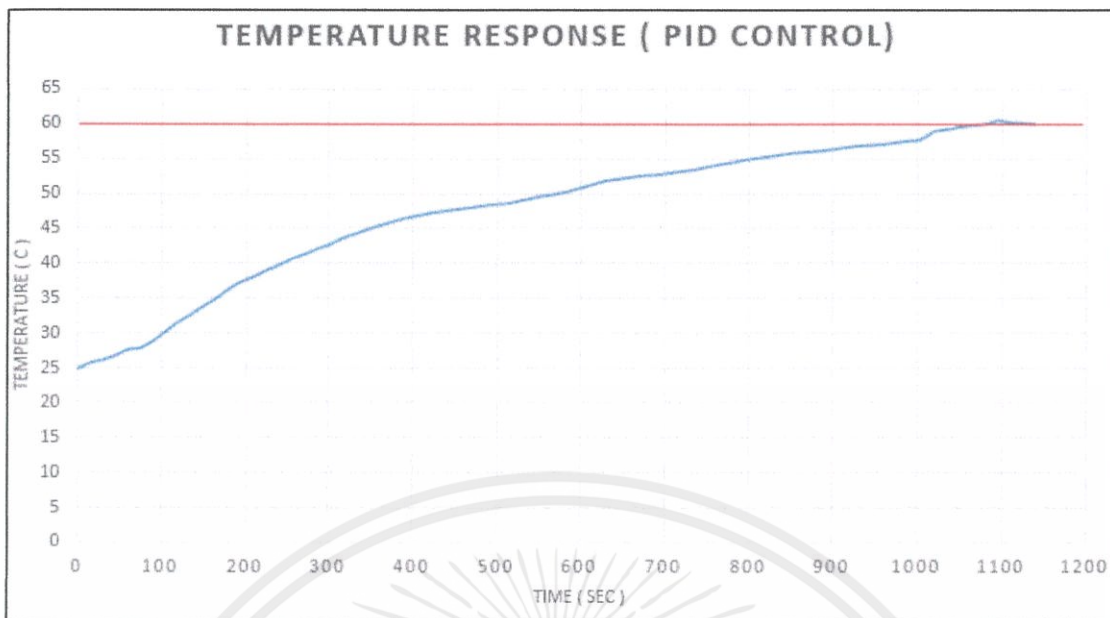


รูปที่ 4.8 ผลการทดสอบการทำงานของตัวควบคุมอุณหภูมิที่สามารถโปรแกรมได้ โดยใช้ตัวควบคุมแบบ PID Controller ที่อุณหภูมิเป้าหมาย 40°C (ที่อุณหภูมิห้อง 25°C , $K_p = 100$, $K_i = 15$, $K_d = 1$)



รูปที่ 4.9 ผลการทดสอบการทำงานของตัวควบคุมอุณหภูมิที่สามารถโปรแกรมได้ โดยใช้ตัวควบคุมแบบ PID Controller ที่อุณหภูมิเป้าหมาย 50°C (ที่อุณหภูมิห้อง 25°C , $K_p = 100$, $K_i = 15$, $K_d = 1$)

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.10 ผลการทดสอบการทำงานของตัวควบคุมอุณหภูมิที่สามารถโปรแกรมได้ โดยใช้ตัวควบคุมแบบ PID Controller ที่อุณหภูมิเป้าหมาย 60°C (ที่อุณหภูมิห้อง 25°C , $K_p = 100$, $K_i = 15$, $K_d = 1$)

สรุปผลการดำเนินงานและข้อเสนอแนะ

5.1 สรุปผลการดำเนินงาน

จากการดำเนินงานตามขั้นตอนที่ได้กล่าวมา ในขั้นตอนแรกการศึกษาและทำความเข้าใจถึงโครงการที่ได้รับมอบหมาย ทางผู้จัดทำจึงได้แบ่งส่วนการทำงานเป็น 3 ส่วน คือ ส่วนของกลไกอิเล็กทรอนิกส์ และโปรแกรม ทำให้สามารถศึกษาความรู้ในทางทฤษฎีมาประยุกต์ใช้ในการออกแบบตัวควบคุมอุณหภูมิได้อย่างครอบคลุมทั้ง 3 ส่วน จากนั้นกำหนดสเปคของอุปกรณ์ที่ต้องการ นำสเปคของอุปกรณ์ที่ต้องการไปค้นคว้าศึกษาข้อมูลของอุปกรณ์เหล่านั้นผ่านเอกสารทางเทคนิค เพื่อนำมาใช้ในการจัดวางอุปกรณ์เหล่านั้นภายในตู้ แต่เนื่องจากอุปกรณ์ทางกลไกบางอย่างนั้นไม่ได้สามารถหาได้ทั่วไปในตลาด ดังนั้นทำให้ขั้นตอนการจัดหาอุปกรณ์ทางกลไกค่อนข้างเสียเวลา แต่ระหว่างการจัดหาอุปกรณ์ทางกลไกนั้นทางผู้จัดทำได้ศึกษาโปรแกรม และวงจรอิเล็กทรอนิกส์ที่จะนำมาใช้กับตัวควบคุมอุณหภูมิ จึงทำให้เมื่อได้อุปกรณ์ทางกลไกครบทั้งหมดแล้วสามารถนำโปรแกรม และวงจรอิเล็กทรอนิกส์ที่ได้เตรียมไว้มาทดสอบได้อย่างรวดเร็วและทดแทนเวลาที่เสียไปได้

ในขั้นตอนการโปรแกรมควบคุมการทำงานของขดลวดความร้อน มีการออกแบบและแก้ไขอยู่หลายครั้งเพื่อทำให้ได้ซึ่งเป้าหมายตามที่ได้กำหนดไว้ แต่เนื่องจากการศึกษาที่ยังไม่เพียงพอและแก้ไขปัญหายังไม่ถูกต้อง ทำให้ต้องปรึกษาและขอคำแนะนำจากอาจารย์ที่ปรึกษาและรุ่นพี่ปริญญาโท ภาควิชาวิศวกรรมระบบควบคุม คณะวิศวกรรมศาสตร์ จนสามารถออกแบบและแก้ปัญหาได้อย่างถูกต้องและเหมาะสม จึงทำให้ผลการควบคุมการทำงานเป็นไปด้วยดี

ในขั้นตอนการดำเนินงาน ทำให้ทราบถึงปัญหาต่างๆ จริง ไม่ว่าจะเป็นการออกแบบ การทดสอบ และการปรับแก้ค่าต่างๆ และมีข้อผิดพลาดอยู่บ้าง ทำให้ต้องปรึกษาและขอคำแนะนำจากรุ่นพี่และอาจารย์เพื่อให้ได้วิธีการแก้ไขปัญหาย่างถูกต้องและเหมาะสม จนทำให้ผลการทดลองเป็นไปในทางที่ดี และสำเร็จตามเป้าหมายที่ได้วางเอาไว้

5.2 ปัญหาและอุปสรรคในการดำเนินงาน

1. เนื่องจากในตอนต้น นักศึกษายังไม่เข้าใจถึงหลักการและจุดประสงค์ในการดำเนินงาน ทำให้การดำเนินงานในช่วงต้นนั้น มีความล่าช้า
2. ยังไม่มีความชำนาญในการออกแบบและควบคุมการทำงานของโปรแกรม PID จึงทำให้ใช้เวลาในการศึกษา

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3. เมื่อทดสอบการทำงานจริงกับอุปกรณ์ อาจเกิดความคลาดเคลื่อนเนื่องจากอุปกรณ์ที่จัดหา
4. มีการเปลี่ยนแปลงและแก้ไขโปรแกรมในการควบคุมการทำงาน Arduino ทำให้การวางแผนตัวแปรในตอนต้นไม่ตรงตามจุดประสงค์ที่วางไว้

5.3 แนวทางในการแก้ไขปัญหา

1. ทำความเข้าใจและศึกษาขอบเขตการทำงานของโครงการที่ได้รับมอบหมาย รวมถึงให้ข้อคำปรึกษาอย่างละเอียดจากอาจารย์และรุ่นพี่
2. ทำการศึกษาและฝึกทักษะของนักศึกษาเกี่ยวกับทักษะเบื้องต้นและทักษะที่เกี่ยวข้องกับอุปกรณ์ที่เกี่ยวข้อง รวมถึงทำการศึกษาการใช้งานอย่างละเอียด เพื่อให้เข้าใจและเกิดข้อผิดพลาดในการทำงานน้อยที่สุด
3. หาอุปกรณ์ที่จำเป็นต้องใช้งานในโครงการที่เกี่ยวข้อง เพื่อให้สามารถดำเนินการได้อย่างต่อเนื่อง
4. ตรวจสอบขีดจำกัดของฟังก์ชันและอุปกรณ์บางอย่าง ซึ่งอาจจะทำให้เกิดข้อผิดพลาดในภายหลังได้ หากไม่ได้ตรวจสอบว่าสามารถทำงานตามเป้าหมายที่วางเอาไว้ได้

เอกสารอ้างอิง

- [1] CHIANG MAI MAKERCLUB. (2016). Easy DIY PID Temperature Control Oven เปิดประสบการณ์การทำตู้อบแบบง่ายๆ, 1 เมษายน 2560. <http://cmmakerclub.com/2016/06/show-tell/easy-diy-pid-temperature-control-oven-เปิดประสบการณ์การทำตู้อบแบบง่ายๆ>
- [2] Riewruja, V.(2557). วิศวกรรมการควบคุมแบบอิเล็กทรอนิกส์. Bangkok: มินิ เซอร์วิส ซัพพลาย
- [3] ยุทธนา ลีลาศวัฒน์กุล(2547). เริ่มต้นการเขียนโปรแกรมด้วยภาษา C++. Bangkok: หจก.ไทยเจริญการพิมพ์





เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ก

Datasheet ของ DS18B20

DS18B20

Programmable Resolution 1-Wire Digital Thermometer

General 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. 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.

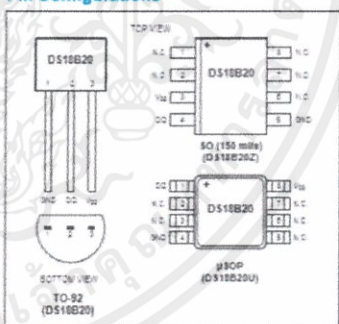
Applications

- Thermostatic Controls
- Industrial Systems
- Consumer Products
- Thermometers
- Thermally Sensitive Systems

Benefits and Features

- Unique 1-Wire[®] Interface Requires Only One Port Pin for Communication
- Reduce Component Count with Integrated Temperature Sensor and EEPROM
 - Measures Temperatures from -55°C to $+125^{\circ}\text{C}$ (-67°F to $+257^{\circ}\text{F}$)
 - $\pm 0.5^{\circ}\text{C}$ Accuracy from -10°C to $+85^{\circ}\text{C}$
 - Programmable Resolution from 9 Bits to 12 Bits
 - No External Components Required
- Parasite Power Mode Requires Only 2 Pins for Operation (DQ and GND)
- Simplifies Distributed Temperature-Sensing Applications with Multidrop Capability
 - Each Device Has a Unique 64-Bit Serial Code Stored in On-Board ROM
- Flexible User-Definable Nonvolatile (N.V.) Alarm Settings with Alarm Search Command Identifies Devices with Temperatures Outside Programmed Limits
- Available in 8-Pin SO (150 mils), 8-Pin μSOP and 3-Pin TO-92 Packages

Pin Configurations




The diagram shows three pin configurations for the DS18B20:

- TO-92 (DS18B20):** A 3-pin package with pins labeled 1 (DQ), 2 (GND), and 3 (VCC).
- SO (150 mils) (DS18B20Z):** An 8-pin package with pins: 1 (DQ), 2 (GND), 3 (VCC), 4 (NC), 5 (NC), 6 (NC), 7 (NC), and 8 (NC).
- μSOP (DS18B20U):** An 8-pin package with pins: 1 (DQ), 2 (GND), 3 (VCC), 4 (NC), 5 (NC), 6 (NC), 7 (NC), and 8 (NC).

Ordering Information appears at end of data sheet.

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



12-7427 Rev. 4, 1/15

รูปที่ ก.1 ข้อมูลทั่วไปของ DS18B20 (1)

DS18B20

Programmable Resolution
1-Wire Digital Thermometer**Absolute Maximum Ratings**

Voltage Range on Any Pin Relative to Ground: -0.5V to +6.0V Storage Temperature Range: -55°C to +125°C
 Operating 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 ratings 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
Supply Voltage	V _{DD}	Local power (Note 1)	+3.0		+5.5	V
Pullup Supply Voltage	V _{PU}	Parasite power	+3.0		+5.5	V
		Local power	+3.0		V _{DD}	
Thermometer Error	T _{ERR}	-10°C to +85°C			±0.5	°C
		-55°C to +125°C			±2	
Input Logic-Low	V _{IL}	(Notes 1, 4, 5)	-0.3		+0.8	V
Input Logic-High	V _{IH}	Local power	+2.2		The lower of 5.5 or V _{DD} + 0.3	V
		Parasite power	+3.0		V _{DD} + 0.3	
Sink Current	I _L	V _{I/O} = 0.4V	4.0			mA
Standby Current	I _{DDS}	(Notes 7, 8)		750	1000	nA
Active Current	I _{DD}	V _{DD} = 5V (Note 9)		1	1.5	mA
DQ Input Current	I _{DQ}	(Note 10)		5		μA
Drift		(Note 11)		±0.2		°C

Note 1: All voltages are referenced to ground.

Note 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_{DD}. 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}

Note 3: See typical performance curve in Figure 1.

Note 4: Logic-low voltages are specified at a sink current of 4mA.

Note 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.

Note 6: Logic-high voltages are specified at a source current of 1mA.

Note 7: Standby current specified up to +70°C. Standby current typically is 3μA at +125°C.

Note 8: To minimize I_{DD}, DQ should be within the following ranges: GND ≤ DQ ≤ GND + 0.3V or V_{DD} - 0.3V ≤ DQ ≤ V_{DD}.

Note 9: Active current refers to supply current during active temperature conversions or EEPROM writes.

Note 10: DQ line is high ("high-Z" state).

Note 11: Drift data is based on a 1000-hour stress test at +125°C with V_{DD} = 5.5V.

www.maximintegrated.com

Maxim Integrated | 2

รูปที่ ก.2 ข้อมูลทั่วไปของ DS18B20 (2)

DS18B20

Programmable Resolution
1-Wire Digital Thermometer**AC Electrical Characteristics—NV Memory**(-55°C to +125°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
Temperature Conversion Time	t _{CONV}	9-bit resolution			93.75	ms
		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}	(Note 12)	60		120	μs
Recovery Time	t _{REC}	(Note 12)	1			μs
Write 0 Low Time	t _{LOW0}	(Note 12)	60		120	μs
Write 1 Low Time	t _{LOW1}	(Note 12)	1		15	μs
Read Data Valid	t _{RDV}	(Note 12)			15	μs
Reset Time High	t _{RSTH}	(Note 12)	480			μs
Reset Time Low	t _{RSTL}	(Notes 12, 13)	480			μs
Presence-Detect High	t _{PDHIGH}	(Note 12)	15		60	μs
Presence-Detect Low	t _{PDLOW}	(Note 12)	60		240	μs
Capacitance	C _{IN/OUT}				25	pF

Note 12: See the timing diagrams in Figure 2.

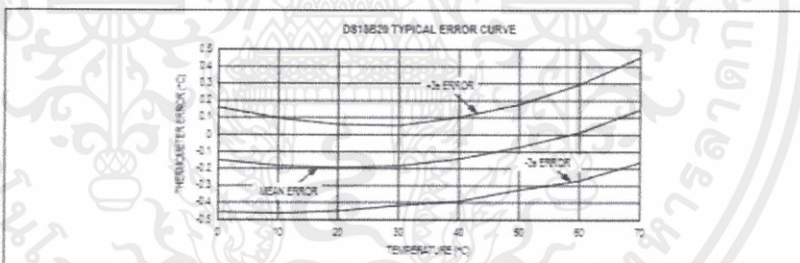
Note 13: Under parasite power, if t_{RSTL} > 900μs, a power-on reset can occur.

Figure 1. Typical Performance Curve

www.maximintegrated.com

Maxim Integrated | 3

รูปที่ ก.3 ข้อมูลทั่วไปของ DS18B20 (3)

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

DS18B20

Programmable Resolution
1-Wire Digital Thermometer

Overview

Figure 3 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 through 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} .

Operation—Measuring Temperature

The core functionality of the DS18B20 is its direct-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.

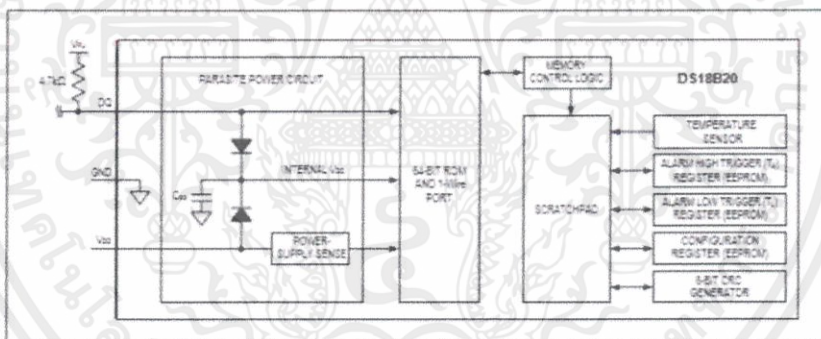


Figure 3 DS18B20 Block Diagram

รูปที่ ก.4 ข้อมูลทั่วไปของ DS18B20 (4)

DS18B20

Programmable Resolution
1-Wire Digital Thermometer

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 4). 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.

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 5). 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.

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

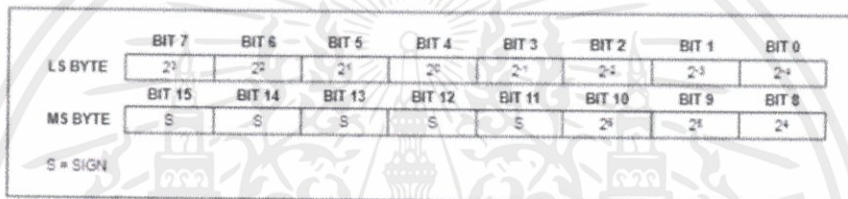


Figure 4. Temperature Register Format

Table 1. Temperature/Data Relationship

TEMPERATURE (°C)	DIGITAL OUTPUT (BINARY)	DIGITAL OUTPUT (HEX)
+125	0000 0111 1101 0000	07D5h
+85*	0000 0101 0101 0000	0690h
+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	FF2Eh
-25.0625	1111 1110 0110 1111	FE8Fh
-55	1111 1100 1001 0000	FC90h

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

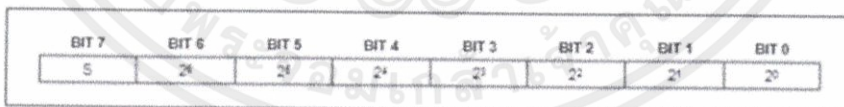


Figure 5. T_H and T_L Register Format

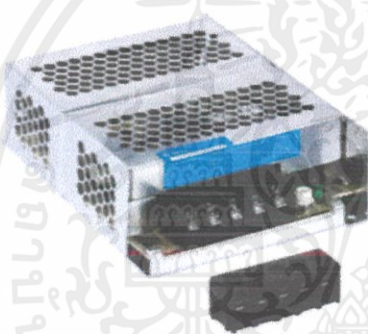
รูปที่ ก.5 ข้อมูลทั่วไปของ DS18B20 (5)

ภาคผนวก ข

Datasheet ของ Power Supply 24V AC-DC PMC-24V050W1AA

TECHNICAL DATASHEET

PMC Panel Mount Power Supply 24V 50W 1 Phase / PMC-24V050W1A□



PMC

Highlights & Features

- Universal AC input range from 85Vac to 264Vac without power de-rating
- Full Aluminium casing for lightweight and corrosion resistant handling
- Conform to harmonic current IEC/EN 61000-3-2, Class A
- High MTBF > 700 000 hrs. as per Telcordia SR-332
- Overvoltage / Overcurrent / Over Temperature / Short Circuit Protections
- Also available IP20 connector

Safety Standards

CE cULus CCC N
CB Certified for worldwide use

Model Number: PMC-24V050W1A□
Unit Weight: 0.26 kg
Dimensions (H x W x D): 128 x 97 x 38 mm

General Description

The new Panel Mount Power Supply is the latest offering from one of the world's largest power supply manufacturers and solution providers - Delta. The product range offers a nominal output voltage of 24V, a wide temperature range from -10°C to +70°C and a highly dependable minimum holdup time. The state-of-the-art design is made to withstand harsh industrial environments. What makes the product stand out from the crowd is its lightweight full aluminium body design which can withstand shock and vibration according to IEC 60068-2-6. Delta's Panel Mount Power Supply also offers overvoltage and overload protection. Using a wide input voltage range design, it is compatible worldwide. The input also includes DC operating voltage from 125-375Vdc. Best of all, this excellent design and quality does not come with a big price tag.

Model Information

PMC Panel Mount Power Supply


Model Number	Input Voltage Range	Output Voltage	Output Current
PMC-24V050W1A□	85-264Vac (125-375Vdc)	24Vdc	2.10A

Model Numbering

PM	C-	24V	050W	1	A	□
Panel Mount	Product Type C - Enclosed	Output Voltage	Output Power	Single Phase	No PFC	Connector Type A - Terminal Block J - IP20 Terminal Block*

*Options

1 | All parameters are specified at 25°C ambient unless otherwise noted
www.DeltaPSU.com (September 2013, Rev. 01)



รูปที่ ข.1 ข้อมูลทั่วไปของ Power Supply 24V AC-DC PMC-24V050W1AA (1)

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

TECHNICAL DATASHEET

PMC Panel Mount Power Supply

24V 50W 1 Phase / PMC-24V050W1A□

Specifications

Input Ratings / Characteristics

Nominal Input Voltage	100-240Vac
Input Voltage Range	85-264Vac
Nominal Input Frequency	50-60Hz
Input Frequency Range	47-63Hz
Nominal DC Input Voltage	125-250Vdc
DC Input Voltage Range	125-375Vdc
Input Current	< 1.10A @ 115Vac, < 0.50A @ 230Vac
Efficiency at 100% Load	> 86% @ 115Vac & 230Vac
Max Inrush Current	< 30A @ 115Vac, < 60A @ 230Vac
Power Factor	Conform to EN 61000-3-2
Leakage Current	< 1mA @ 240Vac

Output Ratings / Characteristics

Nominal Output Voltage	24Vdc
Output Voltage Tolerance	± 2% (initial set point tolerance from factory)
Output Voltage Adjustment Range	22-28Vdc
Output Current	2.10A
Output Power	50W
Line Regulation	< 0.5% typ. (@ 85-264Vac input, 100% load)
Load Regulation	< 1% typ. (@ 85-264Vac input, 0-100% load)
PARV (20MHz)	< 150mV/pp
Rise Time	< 30ms @ nominal input (100% load)
Start-up Time	< 2500ms @ nominal input (100% load)
Hold-up Time	> 15ms @ 115Vac, > 90ms @ 230Vac (100% load)
Dynamic Response (Overshoot & Undershoot O/P Voltage)	± 5% @ 0-100% load
Start-up with Capacitive Loads	8,000µF Max

Mechanical

Case Cover	Aluminium
Dimensions (L x W x D)	128 x 97 x 38 mm
Unit Weight	0.26 kg
Indicator	Green LED (DC OK)
Cooling System	Convection
Terminal	PMC-24V050W1AA M3.5 x 5 Pins (Rated 300V/15A)
	PMC-24V050W1AJ M3.5 x 5 Pins (Rated 300V/20A)
Wire	AWG 22-12
Noise (1 Meter from power supply)	Sound Pressure Level (SPL) < 40dBA


2

All parameters are specified at 25°C ambient unless otherwise noted.
www.DeltaPSU.com (September 2013, Rev. 01)



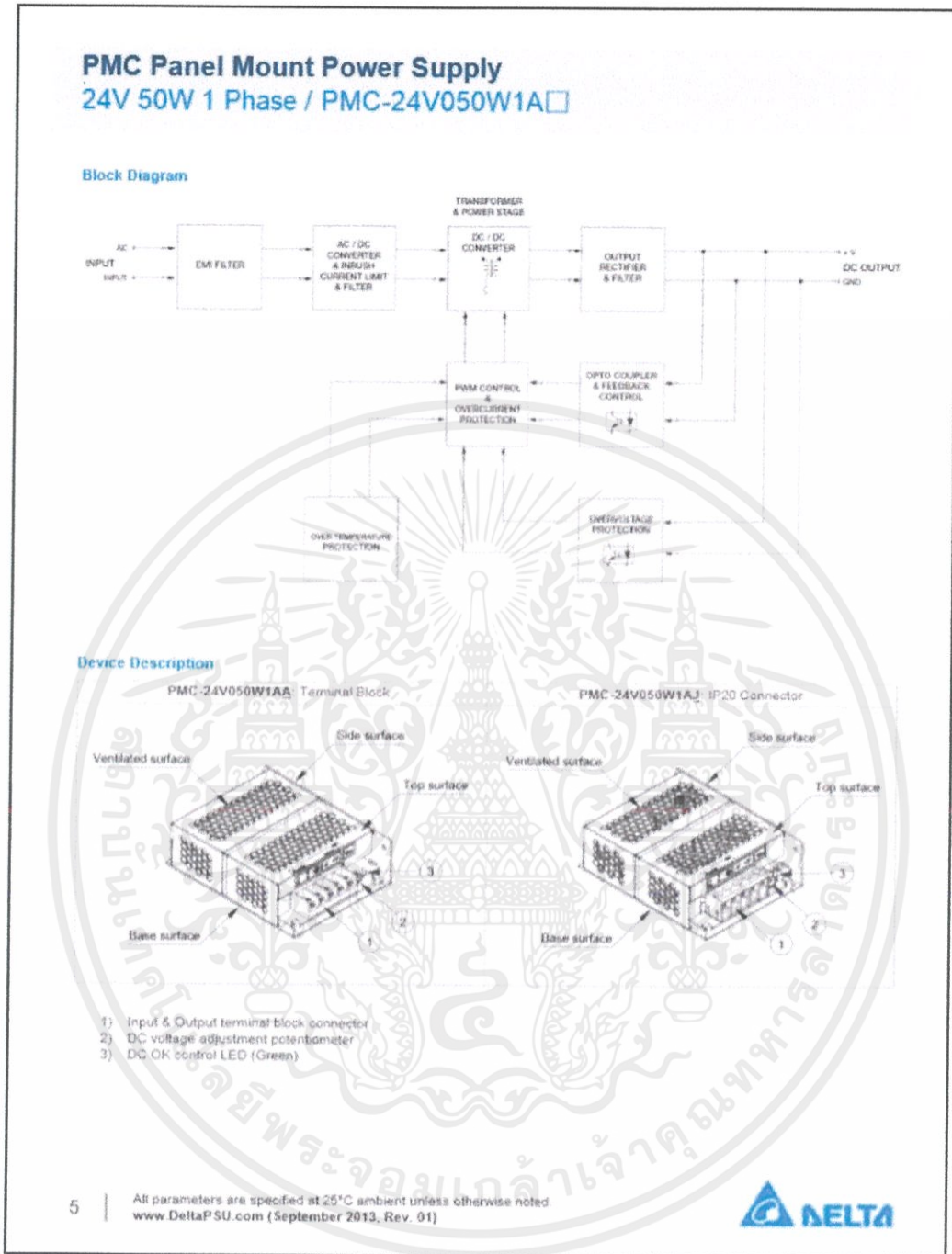
รูปที่ ข.2 ข้อมูลทั่วไปของ Power Supply 24V AC-DC PMC-24V050W1AA (2)

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

TECHNICAL DATASHEET	
PMC Panel Mount Power Supply	
24V 50W 1 Phase / PMC-24V050W1A□	
Environment	
Surrounding Air Temperature	Operating: -10°C to +70°C Storage: -25°C to +85°C
Power De-rating	> 50°C de-rate power by 2.5% / °C
Operating Humidity	< 95% RH (Non-Condensing)
Operating Altitude	3,000 Meters
Shock Test (Non-Operating)	IEC 60068-2-27, 30G (300m/S ²) for a duration of 18ms, 3 times per direction, 18 times in total
Vibration (Non-Operating)	IEC 60068-2-6, 10Hz to 150Hz @ 50m/S ² (5G peak), 90 min per axis for all X, Y, Z direction
Pollution Degree	2
Protections	
Overvoltage	< 32V ±10% SELV Output, Hicc-up Mode, Non-Latching (Auto-Recovery)
Overload / Overcurrent	> 120% of rated load current, Hicc-up Mode, Non-Latching (Auto-recovery)
Over Temperature	< 75°C Ambient Temp @ 100% load, Non-Latching (Auto-recovery)
Short Circuit	Hicc-up Mode, Non-Latching (Auto-recovery when the fault is removed)
Protection Against Shock PE Primary Earth	Class I with PE ¹ connection
Reliability Data	
MTBF	> 700,000 hrs. as per per Telcordia SR-332
Expected Cap Life Time	10 years (115Vac & 230Vac, 90% load @ 40°C)
3	All parameters are specified at 25°C ambient unless otherwise noted www.DeltaPSU.com (September 2013, Rev. 01)
	

รูปที่ ข.3 ข้อมูลทั่วไปของ Power Supply 24V AC-DC PMC-24V050W1AA (3)

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ ข.4 ข้อมูลทั่วไปของ Power Supply 24V AC-DC PMC-24V050W1AA (4)

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

TECHNICAL DATASHEET

PMC Panel Mount Power Supply 24V 50W 1 Phase / PMC-24V050W1A□

Dimensions
H x W x D: 128 x 97 x 38 mm

Engineering Data

De-rating

Note

- Power supply components may degrade or be damaged when the power supply is continuously used outside the shaded region, refer to the graph shown in Fig. 1
- If the output capacity is not reduced when surrounding air temperature >50°C, the device will run into Over Temperature Protection. When activated, the output voltage will go into bouncing mode and will recover when the surrounding air temperature is lowered or the load is reduced as far as necessary to keep the device in working condition.
- In order for the device to function in the manner intended, it is also necessary to keep a safety distance of 20mm with adjacent units while the device is in operation.
- Depending on the surrounding air temperature and output load delivered by the power supply, the device can be very hot!
- If the device has to be mounted in any other orientation, please do not hesitate to contact info@deltapsu.com for more details.

Fig. 1 De-rating for Vertical and Horizontal Mounting Orientation
> 50°C de-rate power by 2.5% / °C

6 | All parameters are specified at 25°C ambient unless otherwise noted
www.DeltaPSU.com (September 2013, Rev. 01)

รูปที่ ข.5 ข้อมูลทั่วไปของ Power Supply 24V AC-DC PMC-24V050W1AA (5)

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ค

โปสเตอร์

Department of Instrumentation and Control Engineering
(Control Engineering)

KMITL ENGINEERING
PROJECT
DAY 2017

Programmable Temperature Control Chamber

Napassorn Suttipattanakit, Pragrid Phasutha and Yannadej Thanasupawut

Abstract

This project presents a design and realization of programmable temperature control chamber. The proposed temperature control chamber is stability and efficiency controlled using proportional-integral-derivative(PID) controller. User can press the button to set the desired temperature value. Moreover, the target and actual temperature values can monitor through LCD display.

Introduction

The purpose of this project is to implement the temperature control chamber with lower cost than commercial chamber. By using inexpensive component such as plywood, acrylic, slotted angle steel to build chamber. And by using basic electrical component that easy to buy such as arduino board, fin type heater, metal cooling fan in electrical system and control system.

Methodology

Step1: Chamber design
Step2: Electrical system and control system design
Step3: Build chamber
Step4: Install electrical system and control system in chamber
Step5: Programming
Step6: PID tuning and testing
Step7: User interface design



Figure1:Chamber Model

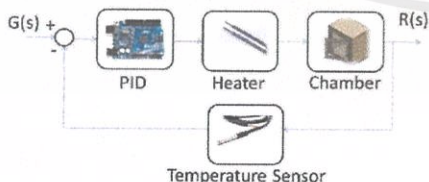


Figure2: Block diagram control system

Results

- Solidwork program is used to design the chamber frame
- Chamber made from plywood and slotted angle steel
- Chamber door made from acrylic and wooden drawing board
- Insulation is used to store heat in the chamber
- Squirrel cage fan is used to compress air thru heater
- Proportional-integral-derivative(PID) and pulse width modulation (PWM) are used to control power that supply to fin type heater



Figure3:Control system with chamber



Figure4:Control system and output signal

Conclusion

Programmable temperature control chamber can be built with lower cost and still work for the purpose like commercial chamber. The chamber can raise temperature up to 80 degree Celsius. This project can develop to lower the cost and increase the performance of chamber in the future by redesign the chamber and improve control system.

References

- [1] <http://cmmakerclub.com/2016/06/show-tell/easy-diy-pid-temperature-control-oven-เปิดประสบการณ์การ/>
- [2] Riewruja, V.(2557). วิศวกรรมควบคุมแบบอิเล็กทรอนิกส์. Bangkok: มินเซอร์วิส ซัพพลาย



**ENGINEERS
LADKRABANG**
วิศวกรรมควบคุมแบบอิเล็กทรอนิกส์ และอิเล็กทรอนิกส์

E-mail: nps.suttipattanakit@gmail.com, pragrid.p@gmail.com, yannadej.t@hotmail.com

รูปที่ ค.1 โปสเตอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้ สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใด ๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้