สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

เตาอบอินเตอร์เน็ต

INTERNET OVEN

โดย

นางสาวพรพันธ์ งามประเสริฐ

นายพิเชษฐ์ สุนทรวรเชษฐ์

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต

สาขาวิชาอิเล็กทรอนิกส์

คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกกระบัง

ปีการศึกษา 2542

เลขหมู่.....
เลขทะเบียน 36942
วัน, เดือน, ปี 30 ส.ค. 2543

ลิขสิทธิ์ของสถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่อนุญาตให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เตาอบอินเตอร์เน็ต

INTERNET OVEN

โดย

นางสาวพรพันธ์ งามประเสริฐ รหัส 39014352

นายพิเชษฐ์ สุนทรวรเชษฐ์ รหัส 39014367

อาจารย์ที่ปรึกษา

ดร.กิติพล ชิตสกุล

ปริญญานิพนธ์สำหรับปริญญา วิศวกรรมศาสตร์บัณฑิต

สาขาวิชาอิเล็กทรอนิกส์

คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2542

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ปริญญานิพนธ์ ปีการศึกษา 2542

ภาควิชาอิเล็กทรอนิกส์

คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เรื่อง เตาอบอินเตอร์เน็ต (INTERNET OVEN)

ผู้จัดทำ

1.นางสาวพรพันธ์ งามประเสริฐ

2.นายพิเชษฐ์ สุนทรวรเชษฐ์

.....อาจารย์ที่ปรึกษา

(ดร.กิติพล ชิตสกุล)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เตาอบอินเตอร์เน็ต

INTERNET OVEN

นางสาวพรพันธ์ งามประเสริฐ เลขประจำตัว 39014352

นายพิเชษฐ์ สุนทรวรเชษฐ์ เลขประจำตัว 39014367

โครงงานผ่านการตรวจสอบแล้ว พร้อมที่จะทำการสอบได้

.....

(ดร.กิติพล ชิตสกุล)

อาจารย์ที่ปรึกษา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เตาอบอินเตอร์เน็ต

นางสาวพรพันธ์ งามประเสริฐ
นายพิเชษฐ์ สุนทรวรเชษฐ์
ดร. กิติพล ชิตสกุล (อาจารย์ที่ปรึกษา)
ปีการศึกษา 2542

บทคัดย่อ

เตาอบอินเตอร์เนต เป็นโครงงานเพื่อศึกษาการนำข้อมูลจากอินเตอร์เนต เพื่อควบคุมอุปกรณ์ไฟฟ้า ระบบที่ใช้ศึกษาเป็นการจำลองว่า บริษัทผู้ผลิตเตาอบได้จัดหาเมนูอาหารสำหรับใช้กับเตาอบที่ตนผลิตขึ้น ผู้ใช้สามารถต่อเตาอบกับไมโครคอมพิวเตอร์ที่ต่ออยู่กับเครือข่ายอินเตอร์เนตและโหลดข้อมูลเมนูอาหารและสั่งการให้เตาอบทำงานตามโปรแกรมในการควบคุมเวลาและอุณหภูมิผ่านจอไมโครคอมพิวเตอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

INTERNET OVEN

Miss Pornpun Ngamprasert
Mr. Pichaith Soonthornworachaith
Dr. Kitiphol Chitsakul (Advisor)
Academic year 1999

Abstract

The Internet Oven is a simulation for a study of using the information of the internet to operate an electrical apparatus. We suppose that an oven manufacturer provides the cooking information (menu, setting time and temperature) used for its product. The customer can load the information to automatically operate the oven via a microcomputer.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กิตติกรรมประกาศ

ปริญญานิพนธ์ฉบับนี้สำเร็จลุล่วงไปเป็นอย่างดีเนื่องด้วยความช่วยเหลือจาก

- บิดามารดาที่คอยอบรมสั่งสอนพร้อมทั้งให้โอกาสข้าพเจ้า และคอยให้กำลังใจอยู่เสมอ
- อาจารย์กิติพล ชิตสกุล อาจารย์ที่ปรึกษาให้ความรู้คำชี้แนะ พร้อมให้คำปรึกษาแก้ไขปัญหาต่าง ๆ พี่ ๆ และเพื่อน ๆ ที่คอยให้กำลังใจและคำปรึกษา

ทางผู้จัดทำขอกราบขอบพระคุณและขอบคุณท่านที่ได้กล่าวนามมาข้างต้น

สุดท้ายขอขอบคุณภาควิชาอิเล็กทรอนิกส์ที่สนับสนุนสถานที่และเครื่องมือต่าง ๆ คุณค่าและประโยชน์อันเกิดมีจากปริญญานิพนธ์ฉบับนี้ผู้จัดขอมอบแด่ผู้มีพระคุณทุกท่าน

ผู้จัดทำ

นางสาวพรพันธ์ งามประเสริฐ

นายพิเชษฐ์ สุนทรวรเชษฐ์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ

	หน้า
บทคัดย่อ	I
Abstract	II
กิตติกรรมประกาศ	III
สารบัญ	IV
สารบัญรูป	VII
บทที่ 1 บทนำ	1
1.1 วัตถุประสงค์ของโครงงาน	1
1.2 ขอบเขตของโครงงาน	1
1.3 โครสร้างของปริญญานิพนธ์	2
บทที่ 2 ทฤษฎีและหลักการเบื้องต้น	3
2.1 ส่วนโครงสร้าง	3
2.2 ฮาร์ดแวร์	3
2.3 ไทริสเตอร์	4
2.3.1 การเลือกใช้ไทริสเตอร์และการอ่านสเปค	5
2.3.2 ไตรแอค	6
2.4 ไอซี TCA 785	7
2.5 ออฟโต้ไอโซเลเตอร์	8
2.5.1 บทนำ	8
2.5.2 รายละเอียดเบื้องต้นทางไฟฟ้า	9
2.6 หลักการพื้นฐานของ DAC	9
2.6.1 ไอซี DAC	11
2.7 อุปกรณ์ตรวจจับ RTD (Resistance Temperature Detector)	12
2.7.1 การหาคุณสมบัติของอาร์ทีดี	13
2.7.2 วงจรวิธสโตนบริดจ์	14

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.8 วงจรขยายอินสตรูเมนเตชัน 15
2.8.1 การทำงานของวงจรขยายอินสตรูเมนเตชัน 15
2.9 ซอฟท์แวร์ 16
2.10 โปรแกรมภาษา Delphi 16
2.10.1 โครงสร้างทั่วไปของเดลไฟ 17
2.11 World Wide Web 17
2.11.1 ส่วนประกอบของ เวิลด์ไวด์เว็บ 18
2.11.2 โฮมเพจ (Home page) 18
2.12 โปรโตคอล HTTP 21
2.12.1 วิธีการติดต่อของโพร โตคอล HTTP 22
2.12.2 โครงสร้างโพรโตคอล HTTP 23
2.10.3 ข้อความร้องขอ (request) 23
2.13 CGI (Common Gateway Interface) 24
2.13.1 การใช้งาน CGI ที่เป็นที่นิยมในปัจจุบัน 25
2.13.2 การส่งข้อมูลของ CGI มี 2 วิธีที่เป็นที่นิยมในปัจจุบัน คือ 25
2.13.3 การทำงานของ CGI 27
2.14 เพิร์ล 29
บทที่ 3 หลักการออกแบบส่วนควบคุม 32
3.1การควบคุมกำลังงานไฟฟ้าด้วย TCA785 32
3.2 ไตรแอค 34
3.3 การเปลี่ยนสัญญาณดิจิตอลเป็นอนาลอก 35
3.4 การเปลี่ยนสัญญาณอนาล็อกเป็นดิจิตอล 35
3.5 การออกแบบส่วนการตรวจวัดอุณหภูมิ 36
3.6 วงจรขยายสัญญาณอินสตรูเมนเตชัน 37
3.7 ส่วนของไมโครคอนโทรลเลอร์ MCS-51 37
3.7.1 การควบคุมการทำงานของ 8255 37
3.7.2 การทำงานของคีย์บอร์ด 4*3 39
3.7.3 การทำงานของจอแสดงผล LCD 41

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.7.4 การรับค่าจากคอมพิวเตอร์ 42
3.7.5 การส่งค่าไปควบคุมอุณหภูมิ และ การรับค่าอุณหภูมิที่ฟีดแบกกลับมา 42
3.8 การติดตั้ง และ เซตโปรแกรมสำรับ ใช้เป็นเซิฟเวอร์ 42
3.9 ส่วนของโปรแกรม HTML 46
3.10 ส่วนของโปรแกรมภาษาเพิร์ล 49
บทที่ 4 การทดลอง
4.1 ผลการทดลองต่อ D/A Converter 53
4.2 ผลการทดลองต่อ A/D Converter 54
4.3 ผลการทดลองต่อ INSTRUMENT เทียบกับอุณหภูมิ 55
4.4 ผลการทดลองของวงจร Control Heater 56
4.5 การทดลองหาความสัมพันธ์ของอุณหภูมิกับเวลาที่จ่ายให้เตาอบ 60
บทที่ 5 สรุปและวิจารณ์ผลการทดลอง 65
หนังสืออ้างอิง
ภาคผนวก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป

รูป	หน้า
1.1 แสดงโครงสร้างการทำงานของระบบ	2
2.1 แสดงส่วนประกอบของอุปกรณ์ที่ใช้ควบคุมเตาอบ	3
2.2 โครงสร้างไตรแอค	4
2.3 การทำงานของไตรแอคในสภาวะต่างๆ	5
2.4 Characterristic curve	6
2.5 วงจรกำเนิดสัญญาณพัลซ์โดยใช้ ไอซี TCA 785	7
2.6 วงจรขยายและแยกระบบกราวด์ของสัญญาณกระตุ้น	8
2.7 โครงสร้างของออฟโต้ไอโซเลเตอร์	9
2.8 สัญญาณทางไฟฟ้าของออฟโตไอโซเลเตอร์	9
2.9 แสดงวงจร R-2R Ladder	10
2.10 แสดงค่า Setting Time ของ D/A	11
2.11 ไอซี DAC 0800	12
2.12 ตารางแสดงจุดสมดุลย์ที่ใช้ในการทดสอบ	14
2.13 แสดงวงจรวิธสโตนบริดจ์	15
2.14 วงจรขยายอินสตรูเมนเตชั่น	16
2.15 แสดงหน้าโฮมเพจ	19
2.16 แสดงการติดต่อระหว่างไคลเอนต์กับเซิร์ฟเวอร์	22
3.1 แสดงลักษณะภายในของ TCA 785	32
3.2 แสดงลักษณะสัญญาณที่ขา TCA 785	33
3.3 ส่วนวงจรต่อออฟโต้คัปเปลอร์	34
3.4 แสดงการออกแบบวงจรตรวจจับอุณหภูมิ	36
3.5 วงจรอินสตรูเมนแอม	37
3.6 แสดงโครงสร้างภายในและการจัดเรียงขาของ 8255	38
3.7 การสร้างลอจิกคีย์สวิตซ์อย่างง่าย	39
3.8 แสดงการต่อแบบเมตริกซ์	40
3.9 แสดงการสแกนค่าคีย์บอร์ด	40

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.4 แสดงการออกแบบวงจรตรวจจับอุณหภูมิ 36
3.5 วงจรอินสตรูเมนแอม 37
3.6 แสดงโครงสร้างภายในและการจัดเรียงขาของ 8255 38
3.7 การสร้างลอจิกคีย์สวิตซ์อย่างง่าย 39
3.8 แสดงการต่อแบบเมตริกซ์ 40
3.9 แสดงการสแกนค่าคีย์บอร์ด 40

3.10 แสดงโครงสร้างของ LCD โมดูล 41
3.11 แสดงลำดับการควบคุมการทำงาน LCD โมดูล 41
3.12 แสดงลักษณะของโปรแกรม Sambar 42 43
3.13 แสดงลักษณะของ System Administration 44
3.14 แสดงลักษณะการเซต Aliases 44
3.15 แสดงการ Add User 45
3.16 แสดงลักษณะการเซต Security Restriction 45
3.17 แสดงการเชื่อมโยงของส่วนต่างๆ 46
3.18 แสดงลักษณะเพจของ Index.html 47
3.19 แสดงลักษณะเพจของ Register.html 47
3.20 แสดงลักษณะเพจของ guestbook.html 48
3.21 แสดงลักษณะเพจของ addguest.html 48
3.22 แสดงลักษณะเพจของ Menu.html 49
3.23 แสดงลักษณะความผิดพลาดที่เกิดจากข้อมูลว่างเปล่า 50
3.24 แสดงการควบคุมการทำงานของ MCS-51 51
3.25 แสดงการควบคุมอุณหภูมิ 52
4.1 กราฟแสดงเอาท์พุทจากอินสตรูเสนเตชั่นเทียบกับอุณหภูมิ 55
4.2-4.12 แสดงผลการทดลองคอนโทรล Heater 56
4.13 กราฟแสดงอุณหภูมิกับเวลาตั้งค่าอุณหภูมิที่ 100 ^{0}C 60
4.14 กราฟแสดงอุณหภูมิกับเวลาตั้งค่าอุณหภูมิที่ 200 ^{0}C 61
4.15 กราฟแสดงอุณหภูมิกับเวลาตั้งค่าอุณหภูมิที่ 300 ^{0}C 62

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 1
บทนำ

ในปัจจุบันเทคโนโลยีมีความก้าวหน้าและมีบทบาทในชีวิตประจำวันของมนุษย์เราเป็นอย่างมาก และในปัจจุบันอินเตอร์เนตเข้ามามีส่วนในการทำงานเพิ่มมากขึ้นซึ่งทำให้สามารถใช้งานและแบ่งปันฐานข้อมูลต่าง ๆ ได้อย่างรวดเร็ว เช่นการค้นคว้าข้อมูลในการทำวิจัยทางวิชาการ การค้นคว้าข้อมูลทางการศึกษาหรือแม้แต่กระทั่งเพื่อความบันเทิงของผู้ใช้อินเตอร์เนตเอง ดังนั้นจึงมีผู้ใช้งานมากมีแนวโน้มที่จะที่ผู้ใช้งานเพิ่มขึ้นอย่างต่อเนื่อง ด้วยเหตุนี้จึงมีแนวความคิดในการนำข้อมูลมาใช้ในการควบคุมการทำงานอุปกรณ์ทางไฟฟ้า โดยเริ่มจากดัดแปลงทำให้อุปกรณ์ไฟฟ้าชิ้นนั้นสามารถเชื่อมต่อกับระบบอินเตอร์เน็ตได้และสร้าง Home page กำหนดข้อมูลควบคุมอุปกรณ์ทำให้สามารถใช้ข้อมูลนี้ที่ใดก็ได้ที่สามารถเชื่อมต่อกับอินเตอร์เนต

ในการศึกษานี้เลือกใช้เตาอบไฟฟ้าโดยเสมือนว่าเป็นผู้ผลิตเตาอบขึ้นมาให้สามารถเชื่อมต่อกับอินเตอร์เน็ตได้ พร้อมทั้งสร้างฐานข้อมูลบนอินเตอร์เน็ต เพื่อที่ผู้ใช้จะนำข้อมูลการควบคุมการทำงานของเตาอบได้แก่ เวลาและอุณหภูมิมาใช้กับเตาอบได้

1.1 วัตถุประสงค์ของโครงงาน

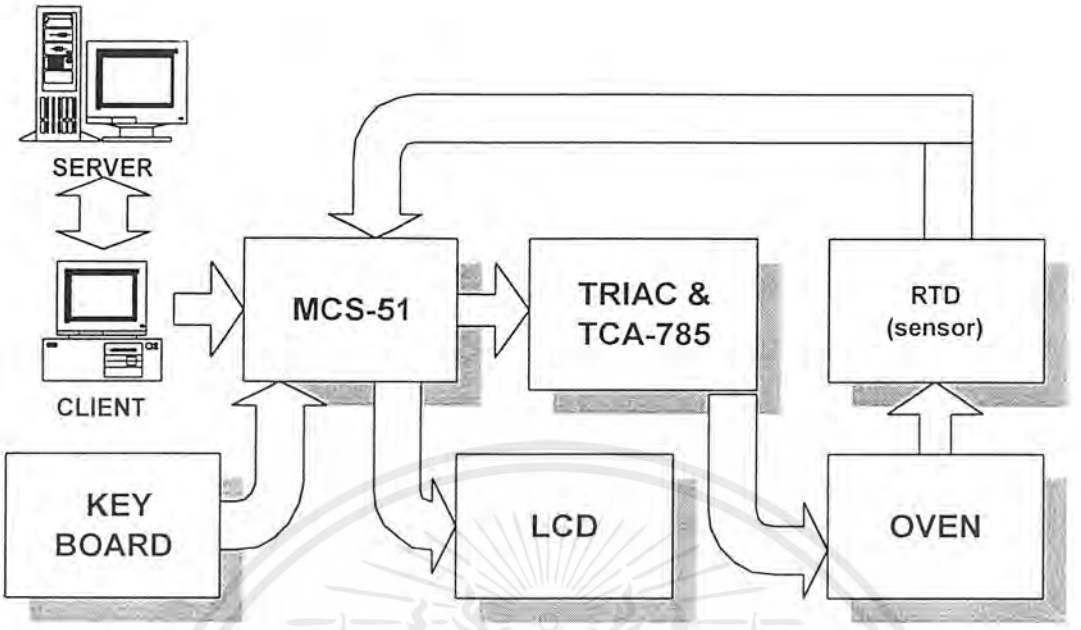
จุดประสงค์หลักของการทำโครงงานนี้เป็นการการศึกษาประยุกต์ใช้งานอินเตอร์เนต เพื่อการการควบคุมอุปกรณ์ไฟฟ้าโดยใช้เตาอบเป็นกรณีศึกษา

1.2 ขอบเขตของโครงงาน

ตามแนวคิดของโครงงานเตาอบจะต้องถูกออกแบบใหม่เพื่อให้สามารถเชื่อมต่อกับระบบอินเตอร์เนตได้โดยตรง แต่ในการศึกษานี้จะใช้เตาอบไฟฟ้าแบบที่มีขายในตลาดมาดัดแปลงเพิ่มเติมโมดูลควบคุมการทำงานด้วยไมโครคอนโทรเลอร์ให้สามารถติดต่อกับไมโครคอมพิวเตอร์ซึ่งเชื่อมต่อกับระบบอินเตอร์เนต จากนั้นจำลองเว็บเซิร์ฟเวอร์ของผู้ผลิตที่กำหนดค่าอุณหภูมิ เวลา สำหรับรายการอาหารและวิธีการทำอาหารต่าง ๆ สำหรับให้ผู้ใช้งานสามารถดึงข้อมูล อุณหภูมิและเวลามาควบคุมเตาอบได้โดยผ่านโปรแกรมควบคุมอีกทีหนึ่ง

เตาอบที่ได้ดัดแปลงนี้แม้ไม่ได้ต่อกับคอมพิวเตอร์ ก็ยังสามารถทำงานได้ผ่านคีย์บอร์ดของโมดูลที่เพิ่มเติมเข้าไป ซึ่งโครงสร้างของโครงงานแสดงดังรูป 1.1

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 1.1 แสดงโครงสร้างของระบบ

1.3 โครสร้างของปริญญานิพนธ์

ปริญญานิพนธ์ฉบับนี้เป็นการรายงานผลจากการศึกษาและทดลองตลอดสองภาคการศึกษา เพื่อออกแบบสร้าง เตาอบอิเตอร์เนต ตามแนวคิดที่กล่าวมาข้างต้น ซึ่งเนื้อหานั้นจะประกอบด้วยส่วนต่าง ๆ แยกเป็นบท ๆ ไปดังนี้

บทที่ 1 บทนำ กล่าวถึงความเป็นมา แนวคิดของโครงงาน

บทที่ 2 จะกล่าวถึงทฤษฎี คุณสมบัติของส่วนอุปกรณ์ต่างๆ เพื่อเป็นความรู้พื้นฐานในการออกแบบส่วนของระบบฮาร์ดแวร์ และ ซอฟแวร์

บทที่ 3 เป็นหลักการในการออกแบบ หลักการทำงานและการหาค่าต่างๆ ของตัวอุปกรณ์ และการเขียนโปรแกรมติดต่อกับอุปกรณ์ เพื่อเชื่อมโยงกับอินเตอร์เน็ตได้

บทที่ 4 การทดลอง

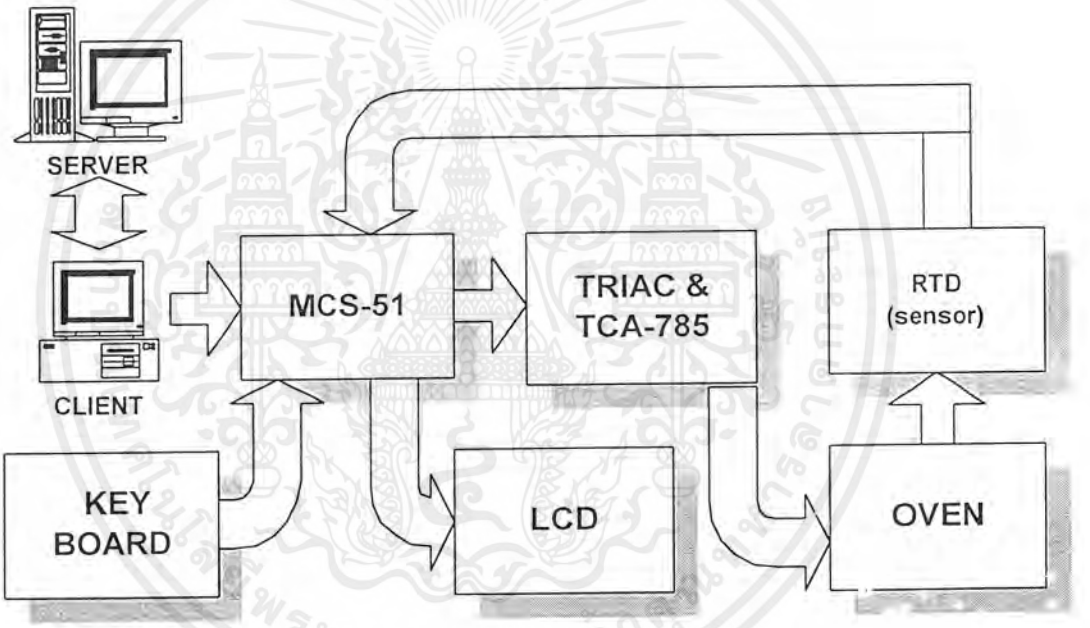
บทที่ 5 สรุปและวิเคราะห์ผล รวมถึงปัญหาที่เกิดขึ้นในระหว่างการทำงาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 2
ทฤษฎีและหลักการเบื้องต้น

2.1 ส่วนโครงสร้าง

เราสามารถแบ่งการทำงานออกเป็นส่วนของ ฮาร์ดแวร์ และซอร์ฟแวร์ ในรูปที่ 2.1 แสดงโมเดลเบื้องต้นของเตาอบควบคุมด้วยไมโครคอนโทรเลอร์ จะเห็นได้ว่าประกอบด้วย เตาอบและมีส่วนของฮาร์ดแวร์ในการควบคุมการทำงาน ซึ่งจะได้รับค่าอินพุทมาจากส่วนของคอมพิวเตอร์ซึ่งสามารถดึงข้อมูลมาจากอินเตอร์เน็ต หรือจากคีบอร์ดได้ด้วย



รูปที่ 2.1 แสดงส่วนประกอบของอุปกรณ์ที่ใช้ควบคุมเตาอบ

2.2 ฮาร์ดแวร์

ในส่วนของการควบคุมนี้จะใช้ MCS-51 เป็นหลักในการทำงาน ซึ่งควบคุมการทำงานของคีย์บอร์ด แอลซีดี ไตรแอค TCA-785 ส่วนที่รับค่าจากคอมพิวเตอร์ และส่วนเซ็นเซอร์ที่ทำหน้าที่ฟีดแบคอุณหภูมิกลับมา

การควบคุมการทำงานของ ไตรแอค นั้นจะใช้ TCA-785 เป็นตัวควบคุมโดยอาศัยหลักการของการควบคุมเฟส ในการควบคุมกำลังงาน และส่วนที่ทำหน้าที่ฟีดแบคอุณหภูมิกลับมาโดยผ่านอินสตรูเมนท์แอมป์ และ ADC กลับมายัง MCS-51 ดังแสดงในรูปที่ 2.1

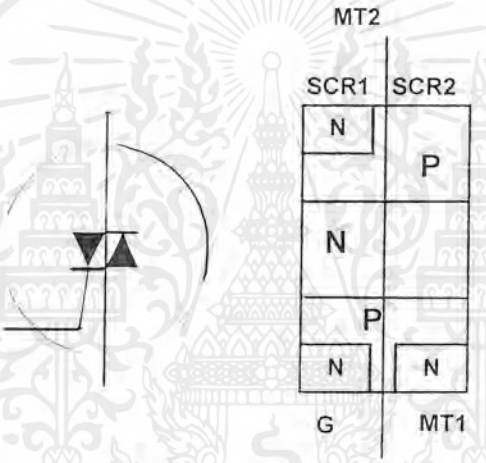
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.3 ไทริสเตอร์

ไทริสเตอร์เป็นอุปกรณ์กึ่งตัวนำที่มีคุณสมบัติคล้ายหลอดไทราทอน โครงสร้างของไทริสเตอร์ประกอบด้วยสารกึ่งตัวนำอย่างน้อย 4 ชั้น คือ p n p n เรามักใช้ไทริสเตอร์ใน วงจรควบคุมกำลังงาน วงจรจ่ายไฟ วงจรควบคุมการทำงานของมอเตอร์ อินเวอร์เตอร์ และคอนเวอร์เตอร์ เป็นต้น

ไทริสเตอร์เป็นกลุ่มของอุปกรณ์อิเลคทรอนิคส์กำลัง เอสซีอาร์และไตรแอค เป็นอุปกรณ์ที่ค่อนข้างจะเป็นที่รู้จักกันดีในกลุ่มไทริสเตอร์ ในที่นี้เราเลือกใช้ ไตรแอคเป็นตัวควบคุมวงจร

ไตรแอคเป็นไทริสเตอร์ที่นำกระแสได้สองทิศทาง กล่าวคือ ไตรแอคจะเสมือนเอสซีอาร์สองตัวที่ต่อหันหัวตรงข้ามกัน แต่การต่อเช่นนี้จะใช้หลักการของชิ้นสารกึ่งตัวนำที่เป็น n p n p n ซึ่งมีทั้งสิ้น 5 ชั้น โดยมีลักษณะโครงสร้างที่เขียนเป็นไดอะแกรมได้ดังรูปที่ 2.2



รูปที่ 2.2 โครงสร้างไตรแอค

ส่วนการทริกให้ไตรแอคทำงานสามารถทำได้ทั้งกระแสบวก (กระแสไหลเข้าเกท) และกระแสลบ (กระแสไหลออกจากเกท) ลักษณะของการทริกจึงแบ่งแยกออกเป็นควอแดรนท์ได้ 4 แบบ คือ

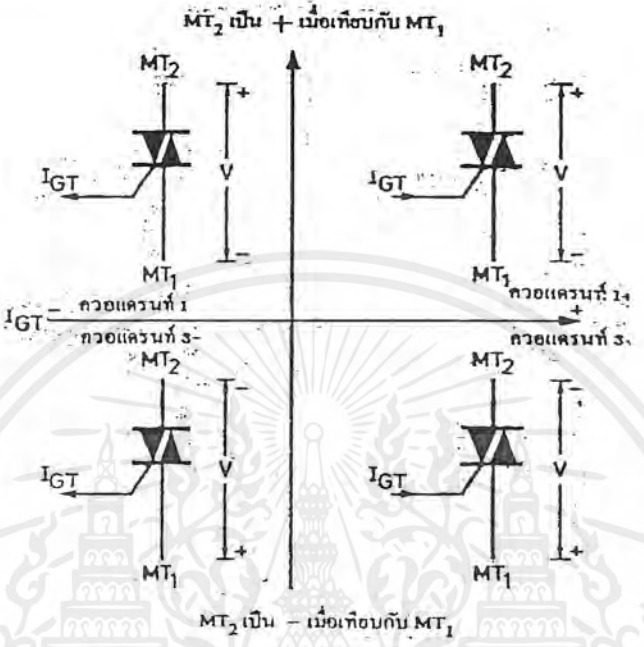
1. ควอแดรนท์ 1+ เป็นลักษณะที่แรงดัน MT2 เป็นบวกเทียบกับ MT1 กระแสทริกเกทเป็นค่าบวกเมื่อไตรแอคนำกระแสจะไหลจาก MT2ไป MT1 เป็นลักษณะที่ทริกได้ง่ายที่สุด คือใช้กระแสทริกไตรแอคน้อยที่สุด

2. ควอแดรนท์ 1- เป็นลักษณะที่แรงดัน MT2 เป็นบวกเทียบกับ MT1 กระแสทริกเกทเป็นค่าลบเมื่อไตรแอคนำกระแสจะไหลจาก MT2ไป MT1 เป็นลักษณะที่ทริกได้ยากต้องใช้กระแสในการทริกไตรแอคสูง

3. ควอแดรนท์ 3+ เป็นลักษณะที่แรงดัน MT2 เป็นลบเทียบกับ MT1 กระแสทริกเกทเป็นค่าบวกเมื่อไตรแอคนำกระแสจะไหลจาก MT1ไป MT2 เป็นลักษณะที่ทริกได้ยากที่สุดต้องใช้กระแสในการทริกไตรแอคสูงมาก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4. ควอแดรนท์ 3- เป็นลักษณะที่แรงดัน MT2 เป็นลบเทียบกับ MT1 กระแสทริกเกทเป็นค่าลบเมื่อไตรแอคนำกระแสจะไหลจาก MT1ไป MT2 เป็นลักษณะที่ทริกได้ง่ายเป็นอันดับสอง ใช้กระแสในการทริกไตรแอคน้อย



รูปที่ 2.3 การทำงานของไตรแอคในสภาวะต่างๆ

2.3.1 การเลือกใช้ไทริสเตอร์และการอ่านสเปค

การพิจารณารายละเอียดของไทริสเตอร์มีหลักการกว้างๆ ดังนี้

1. ขีดจำกัดแรงดัน ต้องพิจารณาค่าแรงดันพังทลายด้านฟอร์เวิร์ดและยังต้องพิจารณาค่าแรงดันพังทลายขณะรีเวิร์สไบแอสด้วย ค่าแรงดันเหล่านี้ต้องมีค่ามากกว่าค่าแรงดันที่ป้อนให้คร่อมตัวมัน

2. ขีดจำกัดทางด้านกระแส ไทริสเตอร์ที่ใช้ต้องทนกระแสได้ไม่ต่ำกว่ากระแสโหลดแบบต่อเนื่อง ส่วนค่ากระแสกระชากสูงสุด (peak surge current) ควรมีค่าประมาณ 15 เท่าของกระแสโหลด

3. ความไวในการจุดชนวนเกท ไทริสเตอร์ที่ใช้ต้องมีความไวในการจุดชนวนเกทสูงพอ คือ กระแสจุดชนวนเกท I_{GT} ในสเปคไม่สูงเกินกว่ากระแสที่วงจรจุดชนวนเกทจะจ่ายให้ได้ และแรงดันจุดชนวนเกท V_{GT} ก็ไม่สูงเกินกว่าค่าแรงดันเอาท์พุทที่วงจรจุดชนวนเกท

4. ความเร็วในการใช้งาน ไทริสเตอร์ที่ใช้ต้องมีเวลาในการกลับสู่สภาพออฟ t_{off} ไม่เกิน1/2f ของการใช้งาน (f คือความถี่ในการใช้งานไทริสเตอร์เป็นสวิทช์)

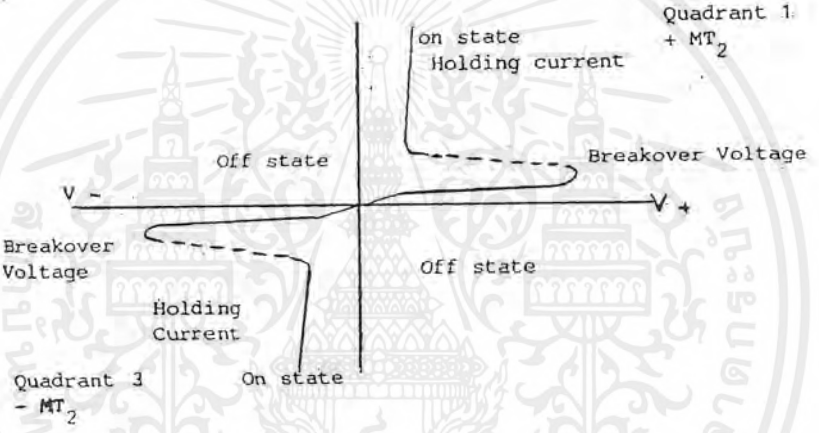
5. กระแสรั่วไหล ไทริสเตอร์ที่ใช้ต้องมีค่ากระแสรั่วไหลทั้งทางด้านไบแอสตรงและไบแอสกลับน้อยพอที่จะไม่ทำให้โหลดทำงานได้ ซึ่งค่ากระแสรั่วไหลนี้จะเป็นตัวจำกัดการใช้งานของไทริสเตอร์กับโหลดน้อยๆ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

6. แรงดันคร่อมตัวไทริสเตอร์ขณะนำกระแส ต้องไม่มากจนเกิดผลต่อวงจรด้านโหลด เช่น โหลดที่ต้องทำงานแรงดันไม่ต่ำกว่า 6 โวลท์โดยใช้แหล่งจ่ายไฟตรง 7 โวลท์ ไทริสเตอร์ที่ใช้ต้องมีแรงดันคร่อมตัวเอง ขณะนำกระแสไม่เกิน 7 – 6 = 1 โวลท์ มิฉะนั้นโหลดจะไม่ทำงาน เป็นต้น

2.3.2 ไตรแอค

ไตรแอค เป็นอุปกรณ์ที่ยอมให้กระแสไหลกลับ (Bidirectional triode) ได้ขณะที่ เอสซีอาร์นั้นยินยอมให้กระแสไหลผ่านตัวมันได้เพียง 1 ทิศทางเท่านั้น โดยมันจะไม่ยอมให้กระแสไหลย้อนกลับตามเส้นทางเดิม ดังนั้นจึงดูเหมือนว่า ไตรแอค นั้นคือ เอสซีอาร์ ที่นำมาขนานกันแต่กลับหัวกลับหางกันอยู่ ไตรแอคมีความสัมพันธ์ระหว่างศักดาและกระแสระหว่างขา MT1 ,MT2ดังรูปที่ 2.4



รูปที่ 2.4 Characteristic curve

จากรูปกราฟความสัมพันธ์ระหว่างกระแสและศักดาของ ไตรแอค เป็นการวัดที่ขา MT_2 เทียบกับขา MT_1 จะเห็นว่าในควอแดรนท์ที่1 รูปกราฟมีลักษณะเช่นเดียวกับ เอสซีอาร์ กล่าวคือหากที่ขา MT_2 ได้รับการป้อนศักดาบวกเทียบกับขา MT_1 จะสังเกตุเห็นว่าจากจุดกำเนิดนั้นตัวไตรแอค จะยังไม่นำกระแสคือยังอยู่ในช่วง off state ไตรแอคจะต้องได้รับการกระตุ้นที่ขาเกท ให้ศักดาคร่อมตัวไตรแอคมีค่าเพิ่มขึ้นโดยวิธีการป้อนกระแสกระตุ้นที่ขาเกท จะทำให้มีกระแสไหลระหว่างขา MT_2 และ MT_1มากขึ้นเรื่อยๆ ซึ่งจะทำให้ศักดาตกคร่อมขา MT_2 และ MT_1 ค่อยๆมีค่ามากขึ้นเรื่อยๆ โดยในสภาวะนี้ ไตรแอคก็ยังไม่นำกระแสยังคงอยู่ในสภาวะ off state ต่อเมื่อศักดาตกคร่อม MT_2 และ MT_1 มีค่ามากขึ้นจนถึงศักดาที่เรียกว่า Break over voltage จุดนี้จะเป็นจุดที่ ไตรแอคเกิดกระแสไหลมากขึ้นโดยฉับพลัน ก็จะทำให้ศักดาตกคร่อม MT_2 และ MT_1 มีค่าลดลงอย่างรวดเร็ว กระแสที่เพิ่มขึ้นอย่างฉับพลันนี้มีค่าถึงขนาดหนึ่งที่เรียกว่า Holding current ณ จุดนี้จะถือเป็นจุดเริ่มต้นของการนำกระแสของไตรแอค ซึ่งกระแสจะไหลได้เป็นจำนวนมากในขณะที่ศักดาตกคร่อมตัวไตรแอคมีค่าต่ำมากๆ ซึ่ง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

มีสภาพเหมือนสวิทซ์ ปิด-เปิดไฟนั่นเอง และในทางกลับกันในควอแดรนท์ที่ 3 นั้นเป็นการป้อนศักดาบวกให้กับ MT_1 หรือเมื่อวัดที่ MT_2 เทียบกับ MT_1 และศักดาตกคร่อม MT_2 และ MT_1 จะมีค่าเป็นลบ ซึ่งเป็นการกระตุ้นให้เกิดการนำกระแสดังเช่นที่กล่าวไว้ข้างต้น ความสามารถในการนำกระแสของไตรแอค (รวมทั้งเอสซีอาร์ด้วย) นั้นจะนำกระแสได้มากน้อยเท่าใดขึ้นอยู่กับความทนต่ออุณหภูมิที่เพิ่มขึ้นอันเนื่องมาจากพลังงานสูญเสียที่แผ่กระจายออกมาสู่บรรยากาศในรูปของความร้อน โดยกำลังงานที่สูญเสียออกมาในรูปของความร้อนจะมีค่าเป็น

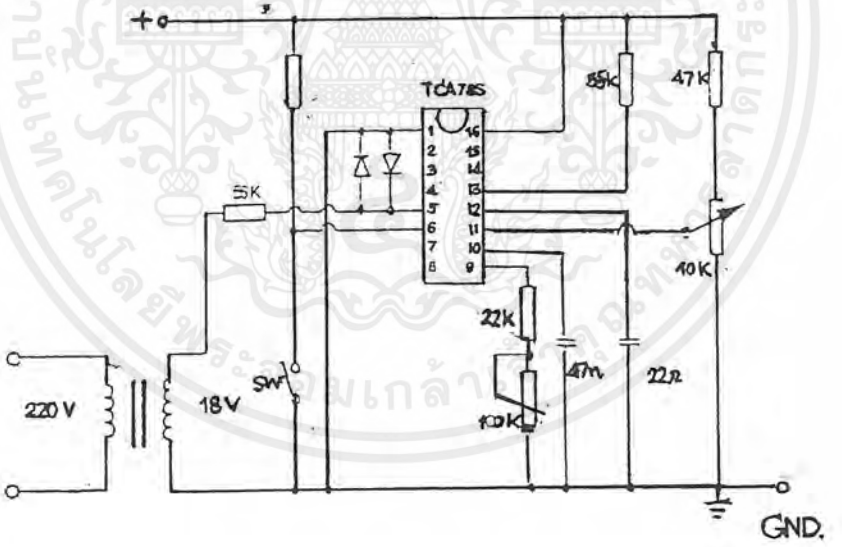
$$P = I_T V_T \quad \text{Watt}$$

I_T = กระแสที่ไหลผ่าน ไตรแอค ขณะนำกระแส

V_T = ศักดาตกคร่อม ไตรแอค ขณะนำกระแส

ดังนั้นการเลือกใช้ ไตรแอคหรือเอสซีอาร์ หากต้องใช้นำกระแสจำนวนมากก็จำเป็นต้องนำตัวระบายความร้อน (Heat Sink) มาช่วยในการระบายความร้อนจากตัวไตรแอคหรือเอสซีอาร์ออกสู่บรรยากาศด้วย มิฉะนั้นความร้อนที่เพิ่มขึ้นและสะสมอยู่จะทำให้ ไตรแอค ไหม้ชำรุดเสียหายได้

2.4 ไอซี TCA 785



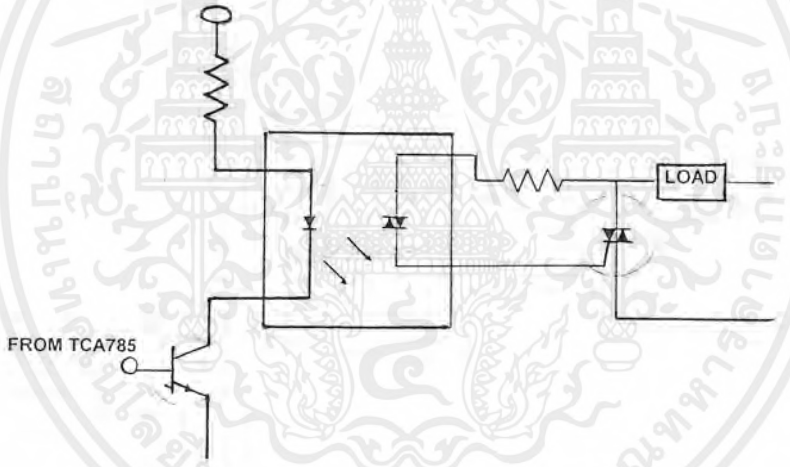
รูปที่ 2.5 วงจรกำเนิดสัญญาณพัลซ์โดยใช้ไอซี TCA785

ไอซีเบอร์นี้มีหน้าที่กำเนิดสัญญาณพัลซ์ที่จะไปกระตุ้นไทริสเตอร์ ซึ่งวงจรการกำเนิดสัญญาณเป็นดังรูป 2.6 มีการทำงานดังนี้คือ สัญญาณไฟฟ้ากระแสสลับซึ่งได้จากการแปลงลงหม้อแปลง 220 V/18V จะถูกป้อนเข้าที่ขา (5) ของไอซี แซะไอซีจะนำสัญญาณนี้มาสร้างสัญญาณแรมพ์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

(Ramp signal) ซึ่งสัญญาณนี้จะแสดงออกมาที่ขา (10) ต่มมาสัญญาณนี้จะถูกเรียกว่า ศักดาไฟฟ้าควบคุมและวัดที่ศักดาไฟฟ้าควบคุมและสัญญาณแรมพ์ตัดกันจะทำให้เกิดสัญญาณกระตุ้นออกมาที่ขา (15) และ (14) โดยที่ขา (15) นั้นสัญญาณกระตุ้นจะมีเฟสเดียวกับศักดาไฟฟ้าสลับที่ขา (5) และขาที่ (14) จะมีเฟสเลื่อนไปข้างหลัง 180 องศา ดังนั้น ถ้าศักดาไฟฟ้าควบคุมเปลี่ยนแปลง และความสูงของสัญญาณแรมพ์เป็น 10 โวลท์ ดังนั้น เมื่อศักดาไฟฟ้าควบคุมมีพิสัย 0 ถึง 10 โวลท์ พิสัยมุมกระตุ้นก็จะเป็น 0-180 องศาด้วย

หลังจากไอซี TCA 785 สร้างสัญญาณกระตุ้นออกมาทางขา (15) และ (14) แล้วไปต่อกับไทริสเตอร์โดยตรงจะมีปัญหาเกี่ยวกับระบบกราวด์ และปัญหาเนื่องจากวงจรสร้างสัญญาณกระตุ้นอาจเสียหายได้เมื่อวงจรกำลังทำงานผิดพลาด ดังนั้นจึงใช้วงจรต่อไปนี้ขยายสัญญาณกระตุ้นและแยกระบบกราวด์ โดยการขยายสัญญาณกระตุ้นนั้นจะใช้ทรานซสเตอร์ ส่วนการแยกระบบกราวด์เราจะได้ออพโต้ไอโซเลเตอร์



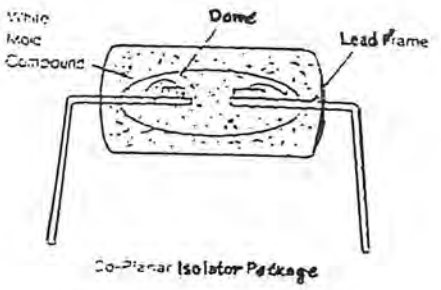
รูปที่ 2.6 วงจรขยายและแยกระบบกราวด์ของสัญญาณกระตุ้น

2.5 ออฟโต้ไอโซเลเตอร์

2.5.1 บทนำ

MOC 3011 จะเคลื่อนผ่านไตรแอคโดยไม่ผ่านศูนย์ ซึ่งประกอบด้วยธาตุแกลเลียมอินฟาเรด LED จะถูกกระตุ้นให้มองเห็นด้วยตัวค้นพบซิลิคอน ซึ่งถูกออกแบบมาเป็นพิเศษเพื่อควบคุมการขับเคลื่อนของไตรแอคบนกระแสไฟฟ้าสลับขนาด 115 V ตัวค้นพบเป็นเครื่องมือที่ซับซ้อน ซึ่งมีหน้าที่การทำงานคล้ายกับไตรแอคตัวเล็ก คือการผลิตสัซซาณที่จำเป็นเพื่อขับเกตของไตรแอคที่ใหญ่กว่า

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



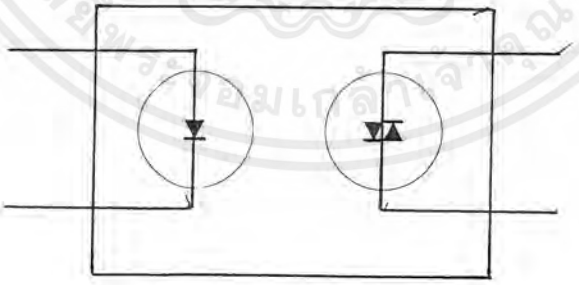
รูปที่ 2.7 โครงสร้างของออฟโต้ไอโซเลเตอร์

MOC 3011 จะแสดงการกระตุ้นสัญญาณกำวังที่ต่ำเพื่อขับไปยังโหลดกำลังที่สูงด้วยส่วนประกอบเล็กๆ จำนวนมากและในขณะเดียวกับผลที่ได้ในทางปฏิบัติก็จะมีการแยกกันของวงจรสายไฟ

2.5.2 รายละเอียดเบื้องต้นทางไฟฟ้า

Ga As LED โดยทั่วไปกำลัง 1.3 V จะตกคร่อมที่ 10 mA และ เมื่อกำลังมากกว่า 3 V จะมีการย้อนกลับกระแสสูงสุดผ่าน LED คือ 50 mA

ตัวค้นพบจะหยุดนำไฟกระแสตรงได้อย่างน้อย 250 V ไม่ว่าจะในทิศทางใดเมื่ออยู่ในสภาวะหยุดทำงานในกรณีที่อยู่ในสภาวะที่ทำงานอยู่ตัวค้นพบจะมีกระแสไหลผ่าน 100 mA ไม่ว่าจะในทิศทางใดแรงดันเท่ากับ 3 V เมื่อหยุดอุปกรณ์ จะมีเหตุการณ์หนึ่งที่ต้องเกิดขึ้นในขณะมีการทำงานอยู่ คือตัวค้นพบจะควบคุมกระแสไฟภายใต้กระแสโฮลดิ้ง



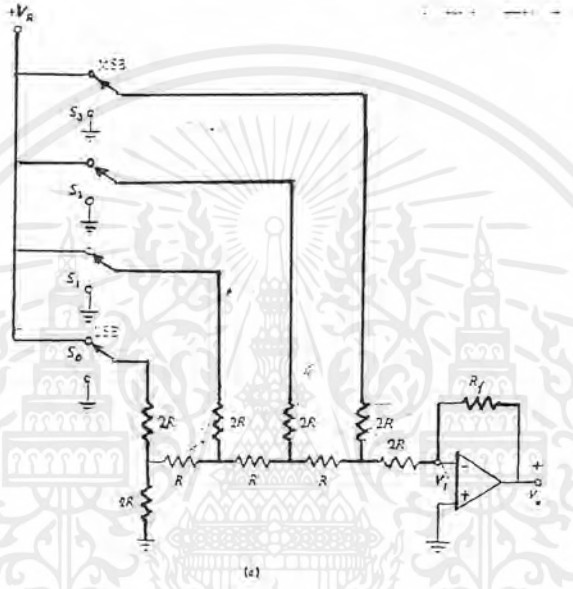
รูปที่ 2.8 สัญลักษณ์ทางไฟฟ้าของออฟโต้ไอโซเลเตอร์

2.6 หลักการพื้นฐานของ DAC

หลักการทำงานของ DAC คือ การนำเอากลุ่มของบิทจากคอมพิวเตอร์ หรืออุปกรณ์ดิจิตอลมาเปลี่ยนเป็นระดับแรงดันอนาลอกเอาท์พุทของ DAC เป็นความแตกต่างของแต่ละบิทของดิจิตอลอิน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

พุท เอาท์พุทชนิดใดก็ตามของ DAC ที่ผลิตขึ้นมาได้จากวงจรที่นำมาใช้ในการเปลี่ยนดิจิตอลเป็น อนาลอก จำนวนความแตกต่างของระดับแรงดันและกระแสที่สร้างขึ้นที่เอาพุทของ DAC จะสัมพันธ์กับจำนวนบิทที่มา เปลี่ยนจากสมการ N คือจำนวนของระดับแรงดันและกระแสที่สร้างขึ้นที่เอาพุท ของ DAC จะขึ้นอยู่กับขอบเขตของการจำแนกของอุปกรณ์ที่ใช้งาน จำนวนบิทของอินพุทจะใช้บิทสูงสุดที่ใช้ในการคำนวณ เช่น อินพุท DAC Resolution เป็นค่าที่เล็กน้อยที่สุดของศักดาไฟฟ้าของวงจร และขึ้นอยู่กับจำนวนบิทของดิจิตอล ยกตัวอย่างเช่น ถ้าอินพุทเป็น 4 บิทดิจิตอล ของวงจรดังรูป



รูปที 2.9 แสดงวงจร R-2R Ladder

มีสมการ Vo ดังนี้

$$V_O = (R_F V_R/3R)\ (S_3/2^1 + S_2/2^2 + S_1/2^3 + S_0/2^4)$$

สังเกต LSB มี weight เท่ากับ 1/16 หมายความว่า ค่าที่จะเพิ่มขึ้นใน Vo ที่เล็กที่สุดเท่ากับ 1/16 ของศักดาไฟฟ้าอ้างอิง Vr = 3R เพราะว่า LSB มี weight = 1/16 ดังนั้นถ้ามีการเปลี่ยนค่าในอินพุทดิจิตอล 1 ค่า จะทำให้ค่าเอาท์พุทจะเป็นแบบขั้นบันได

เมื่อต่อกับวงจรค่าศักดาไฟฟ้าที่น้อยที่สุดที่สามารถเพิ่มขึ้นได้ มีค่าเท่ากับ 1V นี้เรียกว่า Voltage Resolution และค่า Percentage Resolution สามารถหาได้จาก

Percent Resolution $= (1/2^n)*100$: n = จำนวนบิทสูงสุด

$= (1/2^4)*100 = 6.25\%$

ความผิดพลาดอาจเกิดขึ้นได้ใน DACเช่น

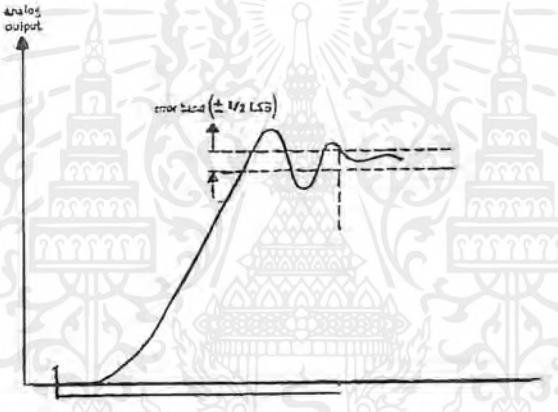
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

OFFSET ERROR เป็นผลที่เกิดขึ้นที่เอาท์พุทของ DAC ไม่เป็น 0 เมื่ออินพุทไบนารี่เป็น 0 ทำให้เกิดค่าคงที่เลื่อน Vo ให้เกิดย่านไบนารี่อินพุท

GAIN ERROR จะสร้างขึ้นขนาดใหญ่กว่า หรือเล็กกว่าขนาดปกติ ซึ่งเป็นสาเหตุให้ค่า Vo เบี่ยงเบนจากค่าความจริงของไบนารี่เอาท์พุท

LINEARITY ERROR เป็นความคลาดเคลื่อนอีกชนิดหนึ่งที่เป็นสาเหตุทำให้ DAC ไม่เป็นเชิงเส้นตัวอย่างเช่น ถ้าอัตราการขยายของ DAC ไม่คงที่สำหรับไบนารี่เอาท์พุท จะเกิดการเปลี่ยนแปลงที่ขนาดของขั้นที่สร้างขึ้น

คุณลักษณะที่สำคัญอีกอย่างหนึ่งคือ ความสัมพันธ์เกี่ยวกับเวลาที่ใช้ในการเปลี่ยนแปลงคุณสมบัตินี้เรียกว่า Setting Time เป็นการวัดความเร็วการตอบสนองทางด้านความเร็วของ DAC



รูปที่ 2.10 แสดงค่า Setting Time (T_S) ของ D/A

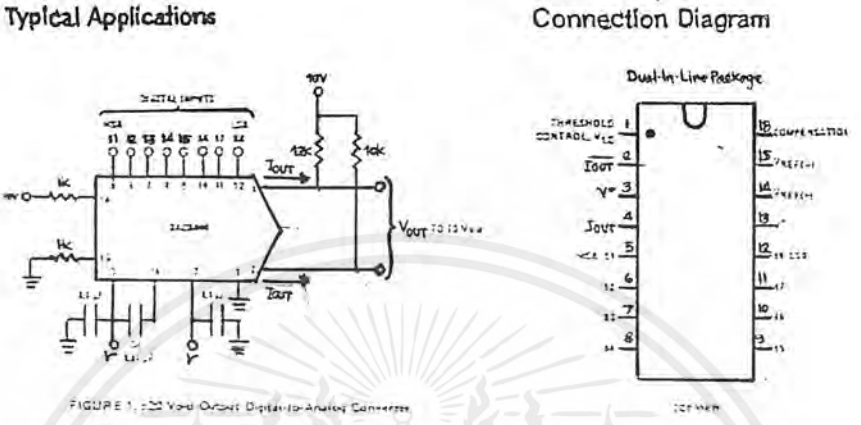
2.6.1 ไอซี DAC

วงจร DAC ทั้งสองที่กล่าวมาเป็นวงจรที่นิยมและเป็นการออกแบบง่ายๆ เมื่อต้องการความสามารถสูงขึ้น จำนวนบิทมากขึ้นก็จะต้องใช้ความต้านทานมากขึ้น เช่น 7 บิท R-2R DAC จะต้องใช้ความต้านทาน 16 ตัว สำหรับวงจร LADDER ทำให้ความต้านทานที่ต้องใช้มากขึ้น เมื่อจำนวนบิทมากขึ้นเกินความสามารถที่จะทดสอบต่อบน Circuit Board และรวมทั้งขั้นตอนการผลิตความต้านทานของโรงงาน ฉะนั้นเราจึงต้องผลิตไอซี ซึ่งจะเหมาะสมในเทอมของค่าต่างๆ และคุณสมบัติความทนทานต่ออุณหภูมิ

ไอซี DAC มีหลายแบบและแตกต่างกัน ตัวอย่างของไอซี DAC ที่ผลิตขึ้นโดยบริษัทเนชั่นแนลเซมิคอนดักเตอร์ คือ DAC0800 เป็น DAC ชนิด 8 บิท ให้เอาท์พุทเป็นกระแสซึ่งมีความสัมพันธ์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กับค่าของ ไบนารีอินพุทที่ป้อนเข้ามา DAC0800 มีทั้งหมด 16 ขา และมีtsetting time 135 ns ดังแสดงในรูปที่ 2.11



รูปที่ 2.11 ไอซี DAC 0800

$$I_{FS} = I_0 + I_o$$

$$V_{REF} = 10\ V$$

$R_{REF} = 5\ k$ $\quad C_C = 0.01\ uF$ $\quad V_{LC} = 0\ V$

$$I_0 = (v_{REF}/R)(B_7/2 + B_6/4 + B_5/8 + \ldots\ldots B_0/256)$$

เมื่อ $B_0 = 0.1\ V$

อินพุท B จะเปลี่ยนแปลงไปมาระหว่าง 0 กับ 1 ซึ่งอินพุทของDAC สามารถต่อเข้าคาต้าบัสของ CPU ได้ขาแรงดันอ้างอิงลง V_{REF}-จะต่อลงกราวน์ผ่านความต้านทาน R_2 ซึ่งมีค่าความต้านทานเท่ากับ R_1 จะช่วยป้องกัน offset errer ขา 16 จะต่อไฟลบ –V โดยมีค่าตัวเก็บประจุคั่นไว้ (มีค่าประมาณ 0.01 uf) .ซึ่งจะช่วยป้องกัน Ringing ,Over shoot เราสามารถผ่านเอาท์พุทของ DAC ที่เป็นกระแสให้เป็นแรงดันได้ โดยใช้อ๊อปแอมป์เปลี่ยนกระแสเป็นแรงดัน

2.7 อุปกรณ์ตรวจจับ RTD (Resistance Temperature Detector)

อาร์ทีดีเป็นอุปกรณ์ที่ใช้ในการตรวจจับอุณหภูมิโดยอาศัยการเปลี่ยนแปลงความต้านทานของโลหะบริสุทธิ์ เมื่ออุณหภูมิเปลี่ยนมีทั้งแบบเส้นลวด และแบบฟิลล์มบาง ในที่นี้จะกล่าวถึงแบบเส้นลวดเท่านั้น

ทฤษฎีความสัมพันธ์ของอุณหภูมิและความต้านทานของ อาร์ทีดี คือ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$$R_T = R_0 \{1+\alpha(\Delta T)+\beta(\Delta T)^2+\gamma(\Delta T)^2+....\}$$

โดยที่

R_0 = ความต้านทาน ณ อุณหภูมิอ้างอิง ($0^{\circ}C$) (ohm)

R_T = ความต้านทาน ณ อุณหภูมิที่สังเกตุ (ohm)

α = สัมประสิทธ์ของการเปลี่ยนแปลงค่าความต้านทานต่ออุณหภูมิ 1 $^{\circ}C$

β และ γ = ค่า สัมประสิทธ์ ที่ใช้คำนวณค่าความต้านทานต่ออุณหภูมิได้ละเอียดขึ้น

โดยทั่วไป สมการ ที่ได้จะเท่ากับ $R_T = R_0 [1+\alpha(\Delta T)]$ โดยเลือกค่าตามชนิดของโลหะ ที่นำมาผลิต ซึ่งมีหลายชนิด แต่ที่นิยมจะมี พลาตินัม ทองแดง และนิเกิล

พลาตินัมอาร์ทีดี เป็นที่นิยมใช้กันอย่างแพร่หลาย ไม่เป็นตัวออกซิไดซ์ และใช้งานได้ในช่วงกว้าง มีความเป็นเชิงเส้นสูง ส่วนทองแดงอาร์ทีดี จะมีความเป็นเชิงเส้นในช่วงแถบประมาณ –200 $^{\circ}C$ ถึง 150 $^{\circ}C$ แต่นิเกิลอาร์ทีดี จะมีความเป็นเชิงเส้นต่ำกว่าชนิดอื่นๆ

โดยทั่วไปมาตรฐานของ พลาตินัมอาร์ทีดีได้กำหนดค่ามาตรฐานดังนี้

มาตรฐานยุโรป α = 0.00385 ohm/$^{\circ}C$

มาตรฐานอเมริกา α = 0.00392 ohm/$^{\circ}C$

ค่าความต้านทานมาตรฐาน 100 ohm ที่ 0 $^{\circ}C$ และ 10 ohm ที่ 0 $^{\circ}C$

มาตรฐานยุโรปพลาตินัมอาร์ทีดี แบบ 100 ohm จะเปลี่ยนความต้านทานโดยเฉลี่ย 0.385 ohm ถ้าในการใช้งานปกติมีแหล่งจ่ายกระแสที่ 1 mA เลี้ยงอาร์ทีดีอยู่ ดังนั้นทุกๆ 1 $^{\circ}C$ จะมีการเปลี่ยนแปลง 0.385 mV ซึ่งมากกว่า เทอร์โมคัปเปิ้ลแบบ K ถึง 10 เท่า ดังนั้นสัญญาณรบกวนค่าเดียวกันจะมีผลต่ออาร์ทีดีน้อยกว่าเทอร์โมคัปเปิ้ล

2.7.1 การหาคุณสมบัติของอาร์ทีดี

ในทางอุตสาหกรรมค่าความถูกต้องของอาร์ทีดีถูกกำหนดในช่วง 0.1 % ถึง 0.5 % ซึ่งสามารถหาค่าความถูกต้องได้ 2 วิธี คือ

1.การเปรียบเทียบ (Comparison Method) เป็นการวัดค่าความต้านทานของอาร์ทีดีเทียบกับการใช้เทอร์โมมิเตอร์วัดอุณหภูมิ

2. อุณหภูมิมาตรฐาน (Fixed Point) โดยการนำเทอร์โมมิเตอร์ที่ต้องการทดสอบไปทำการวัดในอุณหภูมิมาตรฐาน โดยจะมีจุดสมดุลที่ใช้ดังแสดงตามตาราง รูปที่ 2.12

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Fixed Point	Temperature (°C)
Tripple Point of Water	0.01
Boiling Point of Water	1.00
Freezing Point of Tin	231.9681
Freezing Point of Zinc	419.98
Freezing Point of Gold	1065.43

รูปที่ 2.12 ตาราง แสดงจุดสมดุลย์ที่ใช้ในการทดสอบ

ความเสถียรถูกนิยามว่าเป็นความสามารถของอาร์ทีดี ที่รักษาอุณหภูมิไว้เมื่อเวลาผ่านไป โดยค่าความเสถียรจะถูกแสดงในเทอมของดริฟท์คือการเปลี่ยนแปลงค่าความต้านทานที่ไม่พึงประสงค์ในระยะเวลาการใช้งานช่วงค่าดริฟท์สูงสุดจะหาได้จากการทดลองโดยทดสอบจากจุดเยือกแข็งจนเลยจุดจำกัดของอุณหภูมิที่ อาร์ทีดีจะรับได้

ความแม่นยำ คือ ค่าความแม่นยำของการทำซ้ำ โดยวัดค่าความต้านทานที่อุณหภูมิเดิมหลายๆครั้งและสังเกตุค่าความแตกต่างของความต้านทานโดยทั่วไปทางอุตสาหกรรมจะให้ค่าผิดพลาดในการทำซ้ำเป็น0.2%

ความร้อนในตัวอาร์ทีดี (Self Heating) เป็นความผิดพลาดที่เกิดจากกระแสเลี้ยงอาร์ทีดี โดยในวงจรอาร์ทีดี โดยในวงจรอาร์ทีดีต้องมีกระแสเลี้ยงตัวมันเองอยู่ ถ้ากระแสนี้น้อยมาก เอาท์พุทจะสูงแต่จะเกิดความร้อนขึ้นด้วยและจะทำให้เกิดค่าความผิดพลาดในการวัด

ในการสร้างต้องคำนึงถึงความร้อนที่ถ่ายเทจากตัวอาร์ทีดีไปสู่ตัวกลางด้วย และผลลัพธ์ที่ได้ต้องมีค่าผิดพลาดน้อยที่สุด การหาค่าความผิดพลาดที่เกิดจากความร้อนในตัวอาร์ทีดีทำได้โดยการวัดกระแสที่ใช้เลี้ยงอาร์ทีดีและความต้านทานของอาร์ทีดีในช่วงอุณหภูมิใช้งานที่กำหนดโดยการคำนวณทำได้ ดังนี้

$$H = \frac{S.P_2 - P_1}{R_2 - R_1}$$

โดย H = Self Heating Error (mW/°C)

S = การเปลี่ยนแปลงความต้านทานต่ออุณหภูมิ 1 °C โดยเฉลี่ย (ohm/°C)

P_2 = พลังงานสูญเสียที่กระแสสูงสุด (mW)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

P_1 = พลังงานสูญเสียที่กระแสต่ำสุด (mW)

R_2 = ค่าความต้านทานที่กระแสสูงสุด (ohm)

R_1 = ค่าความต้านทานที่กระแสต่ำสุด (ohm)

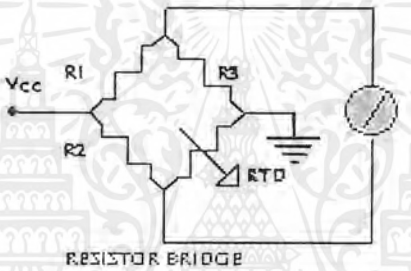
2.7.2 วงจรวิธสโตนบริดจ์

วงจรบริดจ์ประกอบดด้วยตัวต้านทาน 4 ตัว ต่อเป็นรูปแต่ละตัวมีค่าคววามต้านทานที่เป็นสัดส่วนกันซึ่งทำให้แรงดันเกิดการสมดุล

จากวงจร $R_1R_x = R_2R_3$

ถ้าให้ $R_1 = R_2$

จะได้ $R_x = R_3$



รูปที่ 2.13 แสดงวงจรวิธสโตนบริดจ์

2.8 วงจรขยายอินสตรูเมนเตชัน

การพัฒนาวงจรดิฟเฟอเรนเชียลทั่วไปให้ดีขึ้นทำได้โดย

1. เพิ่มความต้านทานที่อินพุท เพราะวงจรขยายแบบดิฟเฟอเรนเชียลทั่วไปมีความต้านทานด้านอินพุทต่ำแก้ไขโดยการใส่บัพเฟอร์เข้าที่อินพุทของวงจร ดังรูปที่ 2.14 เอาต์พุทของออปแอมป์ A_1 จะมีค่าเป็น E_1 และเอาต์พุทของออปแอมป์ A_2 จะมีค่าเป็น E_2 เมื่อเทียบกับกราวด์

2. ค่าอัตราขยายที่ปรับค่าได้ เพราะวงจรดิฟเฟอเรนเชียลคือการที่วงจรชนิดนี้ ไม่สามารถปรับเปลี่ยนอัตราขยายได้ง่าย เราสามารถแก้ไขได้โดยการใช้ตัวต้านทานเพียง 3 ตัว มาต่อเพิ่มเติม ได้ดังรูปที่ 2.14 แรงดันที่จุด 1 ,2 จะมีค่าเป็น E_1 และ E_2 และนั่นทำให้แรงดันตกคร่อมตัวต้านทาน aR มีค่าเท่ากับ $E_1 - E_2$ ดังนั้นกระแสที่ไหลผ่านจะมีค่าดังสมการ

$$I = (E_1-E_2)/aR$$

เมื่อ E_1 มีค่ามากกว่า E_2 ทิศทางไหลของกระแสจะเป็นดังรูปที่ 2.15 ดังนั้นแรงดันที่ตกคร่อมที่ตัวต้านทานทั้งสามจะมีค่าเท่ากับ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$V_0 = (E_1-E_2)(1+ 2/a)$ เมื่อ a= aR/R

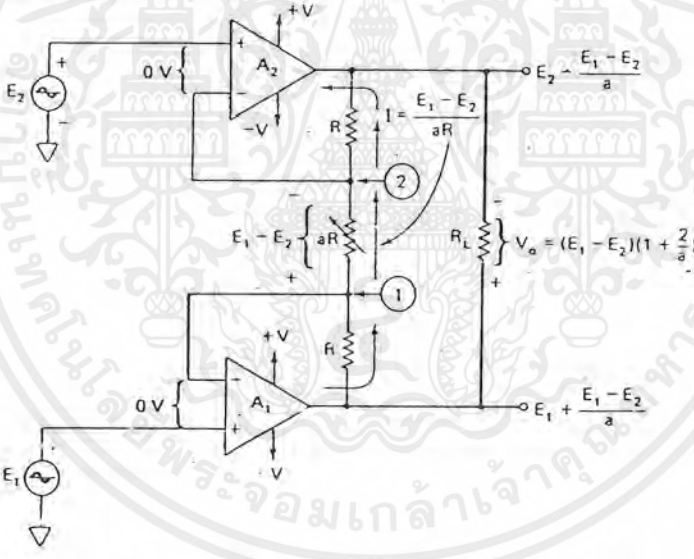
2.8.1 การทำงานของวงจรขยายอินสตรูเมนเตชัน

ส่วนของวงจรอินสตรูเมนเตชั่นเราจะเพิ่มวงจรอินเวอร์ติ้งแอมป์เข้าไปอีกวงจรหนึ่งสำหรับสมการที่ xx เราอาจนำมาเขียนใหม่ให้สะดวกขึ้นดังนี้

$V_O / E_1-E_2 = 1+2/a$ เมื่อ a= aR/R

และเมื่อ E_1 ถูกป้อนเข้าที่ขาบวกของออปแอมป์ ในขณะที่ E_2ป้อนเข้าที่ขาลบเราจะได้คุณสมบัติโดยย่อของวงจรขยายอินสตรูเมนเตชั่นดังนี้

1. อัตราขยายแรงดันซึ่งจะนำมาคูณกับค่า E_1-E_2 จะกำหนดได้โดยตัวต้านทานเพียงตัวเดียว
2. ความต้านทานทางด้านอินพุทจะมีค่าสูงมาก และไม่เปลี่ยนแปลงตามอัตราขยายในขณะที่ออปแอมป์ธรรมดาทั่วไปจะมีการเปลี่ยนแปลงในกรณีเสมอ
3. V_o ไม่ขึ้นอยู่กับแรงดันคอมมอนที่มาจากทั้ง E_1และ E_2 เลย แต่จะเป็นผลที่ขึ้นกับค่าของแรงดันแตกต่างหรือแรงดันดิฟเฟอเรนเชียลเท่านั้น



รูปที่ 2.14 วงจรขยายอินสตรูเมนเตชั่น

2.9 ซอฟท์แวร์

ปัจจุบันการเขียนโปรแกรม สามารถทำให้ง่ายยิ่งขึ้น คือการเขียนโปรแกรมในลักษณะ ของ Visuarl ที่ผู้ออกแบบโปรแกรมเพียงแต่นำเอาส่วนประกอบย่อยๆมาประกอบกันใหม่ให้มองเป็นหน้าตาของโปรแกรมที่ต้องการก่อน แล้วจึงเขียนโปรแกรมสั้นๆควบคุม เพียงเท่านี้ดังนั้นเดลไฟล์จึงเป็นทางเลือกหนึ่ง ที่ผู้จัดทำเลือกใช้ในการเขียนโปรแกรมเพราะเป็นโปรแกรมที่เหมาะกับวินโดส์ 95 และสามารถพัฒนา แอบพลิเคชันได้อย่างหลากหลายเช่น Database Data Communication,Editor, Graphic

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

2.10 โปรแกรมภาษา Delphi

เดลไฟเป็นคอมไพเลอร์ที่มีจุดเริ่มต้นมาจากภาษาปาสคาลและพัฒนามาเป็นออบเจคปาสคาล และสุดท้ายเป็นเดลไฟการเขียนแบบออบเจคจะช่วยลดระยะเวลาในการเขียนโปรแกรมเพราะสามารถดึงโมดูลหรือฟังก์ชันกลับมาใช้ได้เรื่อยๆซึ่งออบเจคต่างๆจะถูกเก็บไว้เป็นระดับที่เรียกว่า Class ซึ่งจะเก็บฟังก์ชันและโมดูลต่างๆเอาไว้เมื่อต้องการนำมาใช้จะต้องสร้างออบเจคขึ้นมา

2.10.1 โครงสร้างทั่วไปของเดลไฟ

หน้าจอเดลไฟแบ่งออกเป็น 4 ส่วนใหญ่คือ

1) หน้าต่างหลัก

ประกอบด้วย 1. ไตเติลบาร์ จะแสดงรายชื่อของชิ้นงาน(Application) แต่ละชิ้นงาน

2. เมนูบาร์ แสดงรายการตั้งแต่ File ถึง Help

3. กล่องคอมโพเนนท์เก็บคอมโพเนนท์ต่างๆที่จะว่างบนฟอร์ม

4. สปีดบาร์

2) หน้าต่างฟอร์มเป็นหน้าต่างที่มีไว้สำหรับแสดงผลในการทำงานของตัวโปรเจคเวลาใช้ให้นำคอมโพเนนท์วงไปวางแล้วกำหนด Properties ,Event ในออบเจคอินสเป็คเตอร์

3) หน้าต่างยูนิต (Unit)มีไว้สำหรับใส่เงื่อนไขต่างๆในการทำงานของโปรแกรมโดยผ่านทางออบเจคอินสเปคเตอร์หรือจะใส่เงื่อนไขโดยตรง

4) หน้าต่างออบเจคอินสเปคเตอร์

ประกอบด้วย Properties เป็นตัวกำหนดการทำงานของคอมโพเนนท์ต่างๆ

Event จะเป็นตัวผ่านจากคอมโพเนนท์ไปกำหนดเงื่อนไขต่างๆในยูนิต

2.11 World Wide Web

เมื่อไม่กี่ปีก่อนหน้านี้มีผู้ที่สนใจใช้งานอินเตอร์เน็ตไม่มากนัก เนื่องจากการใช้บริการอินเตอร์เน็ตไม่ว่าจะเป็นการค้นหาข้อมูลข่าวสารข้อมูล การส่ง E-mail การสำเนาไฟล์ฯลฯ จะอยู่ในรูปแบบ Text Mode เท่านั้น ไม่มีการแสดงที่เป็นรูปภาพ เสียง และไม่มีตัวอักษร แบบต่างๆให้ปรากฏเห็นแต่อย่างใด นอกจากนี้ผู้ใช้ต้องเรียนรู้ และจดจำคำสั่งคอมพิวเตอร์มากมาย เช่น ต้องเรียนรู้คำสั่งเบื้องต้น UNIX เนื่องจากเมื่อมีการเรียกใช้งานอินเตอร์เน็ต เครื่องคอมพิวเตอร์ของผู้ถูกใช้จะเปลี่ยนเป็นเทอร์มินัลของโฮสต์คอมพิวเตอร์ ที่ให้บริการอินเตอร์เน็ต และโฮสต์ส่วนมากจะทำงานอยู่ภายใต้ระบบ UNIX ดังนั้นผู้ใช้จึงต้องเรียนรู้คำสั่งเบื้องต้นของ UNIX เพื่อทำการป้อนคำสั่งที่เป็นตัวอักษรด้วยตนเองให้โฮสต์คอมพิวเตอร์ทำงานตามที่เราต้องการได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้
36942

จนกระทั่งมีบริการที่เรียกว่า เวิร์ดไวด์เวบ เกิดขึ้น ทำให้ความนิยมในการใช้อินเตอร์เน็ตสูงขึ้นเป็นทวีคูณ เนื่องจากเวิร์ดไวด์เวบ เป็นบริการอันหนึ่งที่มีอยู่ในอินเตอร์เน็ตทำให้การใช้งานอินเตอร์เน็ตเป็นสิ่งที่ง่ายขึ้น ผู้ใช้ไม่ต้องจดจำคำสั่งของ UNIX อีกต่อไป การอ่าน และค้นหาข้อมูลข่าวสาร ทำได้โดยการกดปุ่มเมาส์เพียงอย่างเดียว นอกจากนี้ข้อมูลข่าวสาร หรือเอกสารอันประกอบไปด้วยตัวอักษรข้อความที่มีรูปแบบต่างๆ รูปภาพ เสียง ภาพเคลื่อนไหว วีดีโอ และไฮเปอร์เท็กซ์ ซึ่งเป็นการเชื่อมโยงถึงกัน ได้เหมือนกับการใช้ Help ในโปรแกรม Windows โดยผู้ใช้สามารถเรียกดูเอกสารหนึ่งจากเอกสารหนึ่งได้ ซึ่งเป็นลักษณะการเชื่อมต่อที่คล้ายกับใยแมงมุม

2.11.1 ส่วนประกอบของ เวิลด์ไวด์เว็บ

การจะใช้บริการ WWW ได้นั้นจำเป็นต้องมีส่วนประกอบสองส่วนดังนี้

- แหล่งข้อมูล หรือ Web Site
- โปรแกรม Browser

1) แหล่งข้อมูล หรือ Web Site

Web Site หรือ Web Server คือระบบคอมพิวเตอร์ที่เป็นแหล่งเก็บเว็บเพจที่ผู้ใช้บริการสามารถเรียกดูเว็บเพจที่เก็บอยู่ในเว็บไซต์นั้นได้ ซึ่งเครื่องคอมพิวเตอร์ที่ทำหน้าที่เป็นเว็บไซต์อาจจะใช้ระบบ UNIX หรือ Windows NT ก็ได้ และจะต้องมีการติดตั้งโปรแกรมจัดการที่จะทำให้เครื่องคอมพิวเตอร์เครื่องนั้นทำหน้าที่เว็บไซต์ ตัวอย่างเช่น ในกรณีที่เครื่องคอมพิวเตอร์นั้นใช้ระบบ Windows NT จะมีซอฟต์แวร์ให้เลือกใช้ เช่น Purveyor HTTP Server ของสถาบัน European Microsoft Windows NT Academic Center(EMWAC) เป็นต้น ผู้ที่เป็นเจ้าของเว็บไซต์จะจัดสร้างเว็บเพจของตนเก็บไว้ที่เว็บไซต์นั้น เพื่อให้ผู้ใช้คนอื่นๆ ทั่วโลกสามารถเข้ามาดูเว็บไซต์นั้น เช่น เว็บเพจของ CNN จะเก็บอยู่ที่เว็บไซต์ www.cnn.com

2) โปรแกรม เว็บเบราเซอร์

เป็นโปรแกรมหรือซอฟต์แวร์ที่ใช้ในการเข้าสู่ เวิล์ดไวด์เว็บ และเปิดดูเว็บเพจที่เก็บในเว็บไซต์ใดๆ เช่น Netscape Navigator ของ Netscape Communication ซึ่งโปรแกรม Web Browser ส่วนใหญ่นั้นจะทำงานร่วมกับโปรแกรม Windows และนอกจาจะใช้เพื่อดูเว็บไซต์ใดๆ แล้ว หลายโปรแกรมยังมีความสามารอื่นๆ เช่น E-mail การค้นหาข้อมูล การโอนถ่ายโปรแกรมด้วย FTP เป็นต้น

2.11.2 โฮมเพจ (Home page)

โฮมเพจ หมายถึงหน้าแรกของ เว็บเพจ ทั้งหมดที่ผู้ใช้บริการบนอินเตอร์เน็ตจะพบเมื่อมีการเข้าไปยังเว็บไซต์ใดๆโฮมเพจ เปรียบเสมือนกับสารบัญ และคำนำ ที่เจ้าของเว็บไซต์สร้างขึ้นเพื่อ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ประชาสัมพันธ์องค์กรของตนว่าให้บริการในสิ่งใดบ้าง นอกจากนี้ภายใน Home Page ก็อาจมีเอกสารข้อความอื่นๆเชื่อมโยงต่อจาก Home Page นั้นได้อีกที่เรียกว่า Web Page ซึ่ง Home Page หนึ่งๆ อาจมีการเชื่อมต่อ Web Page ไปอีกจำนวนมากได้

รูปที่ 2.15 แสดงหน้าโฮมเพจ

1) การสร้างโฮมเพจด้วย HTML

HTML ย่อมาจากคำว่า Hyper Text Markup Language เป็นรูปแบบหนึ่งของภาษา SGML (Standard Generalized Markup Language) นิยมใช้กันทั่วไปบนอินเตอร์เนต เหมือนกับที่เราใช้โปรแกรมระบบปฏิบัติการ DOS ซึ่งถูกตัดแยกออกมาจากโปรแกรมระบบปฏิบัติการ UNIX เช่นเดียวกับ HTML ซึ่งเป็นภาษาหลักสำหรับการสร้างโฮมเพจ แฟ้มเอกสารที่สร้างขึ้นจะนำไปแสดงผลบนเวบเบราเซอร์เช่น โปรแกรม Netscape Navigator หรือ Internet Expolror

HTML เป็นภาษาที่ง่ายต่อการเรียนรู้และการเขียน ซึ่งจัดได้ว่าง่ายที่สุด เอกสาร HTML สามารถสร้างจากโปรแกรม Text Editor ทั่วไป เช่น Notepad,Wordpd,Microsoft Word, Editorของ DOS ในกรณีที่ใช้โปรแกรม Word ในการสร้างจำเป็นต้องบันทึกแฟ้มข้อมูลด้วยวคำสั่ง “Save as text”เอกสาร HTML โดยทั่วไปจะมีนามสกุลเป็น “.htm” (สำหรับ Windows 95 or UNIX)

2) การทำงานของ HTML

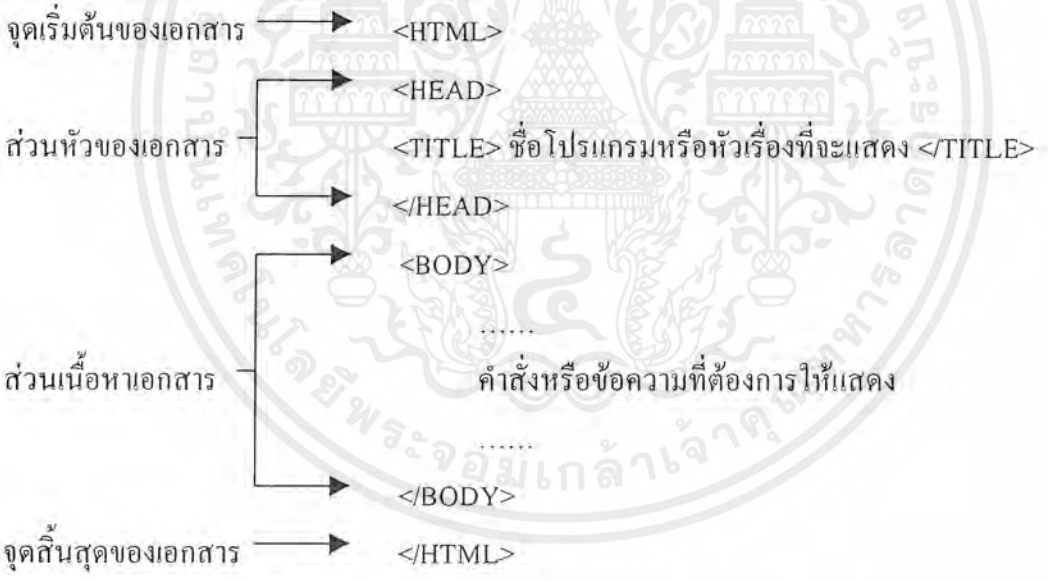
การให้บริการบนอินเตอร์เนตไม่ว่าจะเป็น E-Mail ,FTP,Gopher,Telnet ต้องใช้อุปกรณ์เชื่อมต่อภายในอันซับซ้อนของฮาร์ดแวร์ที่สามารถทำงาน ได้ด้วยโปรแกรมเฉพาะที่ทำงานบนอินเตอร์เน็ตเท่านั้น เวิลด์ไวด์เว็บ แบ่งออกเป็น 2 ส่วนคือส่วนที่เป็น ไคลเอนต์ และส่วนที่เป็น เซิร์ฟเวอร์ เหมือนกับที่มีแบ่งกันในระบบเครือข่ายทั่วไป ทั้งสองจะถูกเชื่อมโยงถึงกันผ่านทางอินเตอร์เน็ต โดยมี

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

HTML เป็นฐานข้อมูลสำคัญ เมื่อ เว็บเบราเซอร์ส่งข้อความร้องขอข้อมูลที่อยู่ในรูปแบบของไฟล์ HTML จากเครื่องคอมพิวเตอร์ที่เราใช้งานอยู่ผ่านโมเด็มหรืออุปกรณ์สื่อสารข้อมูลผ่านไปยังศูนย์บริการอินเตอร์เนต ตามโปรโตคอล ที่กำหนดไว้ผ่านทาง URL (Uniform Resource Locators) เมื่อข้อมูลเดินทางมาถึงเว็บเซิร์ฟเวอร์ (web server) ศูนย์บริการปลายทางที่ผู้ใช้ต้องการ ณ.ที่นี้เครื่องเว็บเซิร์ฟเวอร์ของศูนย์ฯจะทำการอ่านข้อมูลที่ถูกส่งมาและจะทำงานตามคำสั่งที่กำหนดโดยอาจมีการเชื่อมโยงไปยังที่อื่นอีกหลังจากจบสิ้นกระบวนการแล้วจะทำการจัดส่งข้อมูลคำตอบย้อนกลับมายังเครื่องคอมพิวเตอร์ที่เราใช้งานอยู่โปรแกรมเว็บเบราเซอร์ที่เครื่องของเราก็จะแปลงสัญญาณคำสั่งและแสดงผลเป็นข้อความ รูปภาพ เสียง ให้เราได้ใช้งานกันต่อไป

3) โครงสร้างของ HTML

เอกสาร HTML จะประกอบไปด้วยคำสั่งหรือรหัสที่เรียกว่า "Tags"ที่อยู่ภายในเครื่องหมาย <และ> และมีข้อความที่เป็น Text ทั่วไปเพื่อบอกให้บราวเซอร์ทราบว่าจะต้องแสดงผลออกมาเป็นอย่างไร โครงสร้างของ HTML จะประกอบไปด้วยส่วนของคำสั่ง 2 ส่วน ได้แก่ส่วนที่เป็นหัว (HEAD)และส่วนที่เป็นเนื้อหา (Body) ดังนี้



Tag ส่วนใหญ่จะประกอบด้วย Tag เริ่มต้น และ Tag สุดท้ายโดย Tag สุดท้ายจะเขียนเหมือนกับ Tag เริ่มต้นแต่จะต้องมีเครื่องหมาย "/" อยู่ข้างหน้า Tag นั้นด้วย

- ส่วนหัวของเอกสาร <HEAD>...</HEAD>

Tag ที่นิยมใช้อยู่ภายในส่วนหัวของเอกสารคือ <TITLE>..</TITLE>เพื่อที่ใช้แสดงหัวเรื่องของเอกสาร ณ.ตำแหน่งส่วนบนสุด <TITLE BAR> ของเบราเซอร์

- ส่วนเนื้อหาของเอกสาร<BODY>....</BODY>

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เป็นส่วนที่สามารถใส่ข้อความเอกสารได้โดยตรงแต่เบราเซอร์จะไม่มีการจัดรูปแบบของเอกสารให้

ส่วนของโปรแกรมภาษา HTML มีคำสั่งเรียกใช้งานมากมาย เช่นการเชื่อมโยงไฟล์ การทำงานระบบเสียงและภาพวีดีโอ การแมปภาพ การใช้สีตกแต่งเอกสารการนำเสนอในรูปแบบตาราง เป็นต้น

2.12 โปรโตคอล HTTP

สำหรับโปรแกรมเมอร์หรือผู้ที่เขียนโปรแกรมเพื่อติดต่อสื่อสารผ่านเว็บหรืออินเตอร์เน็ตแล้วสิ่งหนึ่งที่จะต้องเรียนรู้และเข้าใจคือเรื่องของ **โปรโตคอล (protocol)** เช่นเว็บเบราเซอร์จะใช้โปรโตคอล HTTP (Hyper Text Transfer Protocol) ในการรับส่งข้อมูล ในความเป็นจริงโปรโตคอล เป็นเพียงข้อตกลงระหว่างสองฝ่ายที่จะติดต่อพูดคุยกัน เพื่อให้การติดต่อระหว่างสองฝ่ายเป็นไปอย่างราบรื่นและถูกต้องมากที่สุดเพราะมีกฎเกณฑ์หรือโปรโตคอลควบคุมอยู่ สิ่งที่พอจะเรียกได้ว่าเป็นโปรโตคอลในชีวิตประจำวันก็คือไฟจราจรทุกคนใช้สิ่งนั้นตกลงกันปัญหาการจราจรจึงลดลง

ในทางอินเตอร์เน็ตและคอมพิวเตอร์ก็เช่นเดียวกัน เมื่อต้องมีการสื่อสารระหว่างคอมพิวเตอร์ ก็จำเป็นต้องกำหนดกฎเกณฑ์อะไรบางอย่างให้ตรงกันสมมุติว่าเราเขียนโปแกรมเพื่อรับส่งข้อมูลระหว่างคอมพิวเตอร์ 2 เครื่องผ่านทางพอร์ตขนาน (parallel port) แล้วตั้งกฎเกณฑ์ง่ายๆนี้เป็นโปรโตคอล

1. เครื่องไหนที่เป็นเครื่องแม่ (server)ต้องเปิดเครื่องรอ
2. ตัวลูก (client)จะส่งข้อมูลขนาด 5 ไบต์ไปถามตัวแม่
3. ถ้าตัวแม่พร้อมที่จะทำงาน จะนำข้อมูลขนาด 5 ไบต์นั้นส่งกลับให้ตัวลูก
4. ถ้าตัวลูกได้รับข้อมูล 5 ไบต์กลับมาจากตัวแม่ แสดงว่าตัวแม่พร้อมแล้ว ตัวลูกจะส่งชื่อไฟล์ที่ต้องการไปให้ตัวแม่
5. เมื่อตัวแม่ได้รับชื่อไฟล์แล้ว จะส่งไฟล์นั้นไปยังเครื่องตัวลูกที่รอรับโดยส่งครั้งละ 512 ไบต์
6. ตัวลูกอ่านข้อมูลจากพอร์ตที่ละ 512 ไบต์แล้วนำมาต่อกันจนได้เป็นไฟล์ที่สมบูรณ์นี่เป็นกฎเกณฑ์ง่ายๆ เพื่อให้มองเห็นภาพโปรโตคอลเท่านั้นถ้าจะเขียนโปรแกรมกันจริงๆต้องมีรายละเอียดให้มากกว่านี้ต้องกำหนดรายละเอียดของโปรโตคอลในระดับบิตเลยทีเดียว

สำหรับโปรแกรมเมอร์หรือผู้ที่อยากจะเขียนโปรแกรมเพื่อติดต่อสื่อสารผ่านเว็บหรืออินเตอร์เนตแล้ว สิ่งหนึ่งที่จะต้องรู้และเข้าใจคือเรื่องของโพรโตคอล **HTTP (Hypertext Transfer Protocol)** ในการรับส่งข้อมูลการรับส่งไฟล์อาจจะใช้ HTTP หรือ FTP (FIle Transfer Protocol) ซึ่งในความเป็น

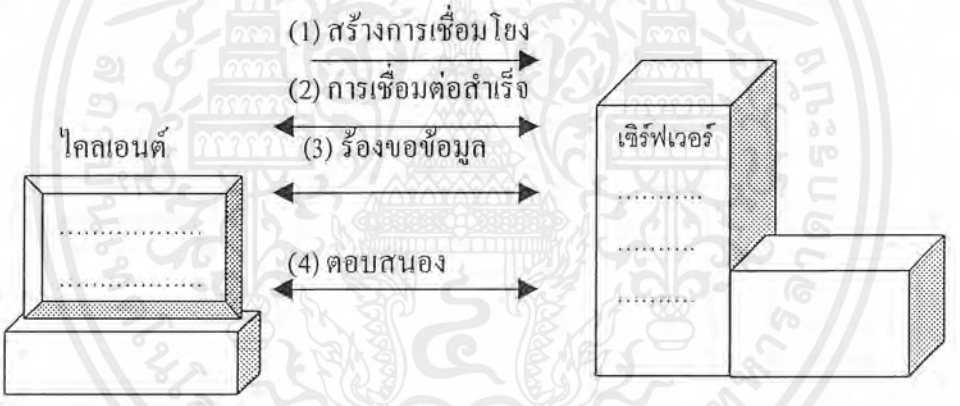
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จริงแล้วโพรโตคอลเป็นเพียงข้อตกลงระหว่าง2ฝ่ายที่จะพูดคุยกันไม่ได้เป็นโปรแกรมหรืออุปกรณ์ฮาร์ดแวร์ใดๆ เรามีโพรโตคอลเพื่อให้การติดต่อระหว่างสองฝ่ายเป็นไปอย่างราบเรียบและถูกต้องมากที่สุด เพราะมีกฎเกณฑ์หรือโพรโตคอลควบคุมอยู่สิ่งที่เรียกว่าโพรโตคอลในชีวิตประจำวันก็คือไฟจราจร ปัญหาการจราจรจึงลดลงไปบ้าง

ในทางอินเตอร์เนตและคอมพิวเตอร์ก็เช่นเดียวกันเมื่อต้องมีการสื่อสารระหว่างเครื่องคอมพิวเตอร์ก็จำเป็นต้องกำหนดกฎเกณฑ์อะไรบางอย่างให้ตรงกันก่อนโพรโตคอลนี้จึงเป็นที่นิยมใช้ทางอินเตอร์เนตระบบบริการที่เรียกว่า เวิล์ดไวด์เว็บในเครือข่ายอินเตอร์เนตโดยเฉพาะควบคุมการรับส่งข้อมูลได้ทั้งภาพและเสียงจนเกือบจะเป็นมัลติมีเดีย จากเดิมแลกเปลี่ยนข้อความได้เพียงอย่างเดียว

2.12.1 วิธีการติดต่อของโพรโตคอล HTTP

การทำงานของโพรโตคอลเป็นแบบไคลเอนต์ และเซิร์ฟเวอร์ (cllent/server)ดังนั้นการติดต่อสื่อสารใดๆระหว่างโพรโตคอลนี้จำเป็นต้องมีคอมพิวเตอร์ระหว่างตัวลูกกับตัวแม่



รูปที่ 2.16 แสดงการติดต่อระหว่างไคลเอนต์กับเซิร์ฟเวอร์

จาก รูปที่ 2.16 ขั้นแรกคือ ไคลเอนต์ (ในตอนนี้คือเว็บเบราเซอร์)จะสร้าง การเชื่อมต่อ (connection) กับเซิร์ฟเวอร์ผ่านสิ่งที่เรียกว่า "ซ็อกเก็ต" (socket) เมื่อซ็อกเก็ตทั้งสองฝั่งเสียบต่อกันได้สำเร็จ ไคไคลเอนต์จะส่ง คำร้องขอข้อมูล (request) ไปยังเซิร์ฟเวอร์จากนั้นเซิร์ฟเวอร์จะไปหาข้อมูลที่ไคลเอนต์ต้องการ ไม่ว่าจะมีข้อมูลที่ร้องขอหรือไม่ก็ตาม เซิร์ฟเวอร์ก็จะต้องส่งข้อมูลตอบสนอง(response) กลับมายังไคลเอนต์เสมอ สุดท้ายการเชื่อมต่อจะถูกตัดขาดหรือปลดการเชื่อมต่อของซ็อกเก็ตทั้งสองฝั่งออกนั่นเอง

ด้วยการทำงานของโพรโตคอล HTTP ที่มีการเชื่อมต่อในระยะเวลาเพียงสั้นๆ หรือที่เรียกว่าเป็น โพรโตคอลแบบ connectionless ทำให้ในช่วงเวลาหนึ่งๆ เซิร์ฟเวอร์ที่ให้บริการ เวิล์ดไวด์เว็บสามารถรองรับไคลเอนต์ได้จำนวนมากพร้อมๆกัน เพราะไม่มีใครได้ทำการเชื่อมต่ออย่างถาวร

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ถ้าเราสั่งให้เว็บเบราเซอร์ติดต่อไปยัง sanook.com เว็บเบราเซอร์จะต้องค้นหาเซิร์ฟเวอร์ sanook.com ให้ได้เสียก่อนเพื่อทำการเชื่อมต่อโดยจะปรากฏข้อความแจ้งให้ทราบที่แถบสถานะ (status bar) เมื่อเซิร์ฟเวอร์รับทราบการร้องขอแล้วพบว่าไม่มีการบอกระบุชื่อไฟล์ ดังนั้นเว็บเบราเซอร์จะส่งไฟล์ที่เป็นไฟล์โฮมเพจกลับมาให้แทน ซึ่งก็คือไฟล์ index.html เมื่อเว็บเบราเซอร์ได้รับไฟล์มาแล้ว เซิร์ฟเวอร์จะตัดการเชื่อมต่อทันที เมื่อเวบเบราเซอร์เริ่มอ่านและตีความหมายของแท็กในไฟล์ index.html จะพบว่ารูปพื้นแบ็กกราวด์ของเว็บเพจระบุให้ใช้ไฟล์ linethai.gif เว็บเบราเซอร์จะสร้างการเชื่อมต่อไปยัง ventula.lanna.com เป็นครั้งที่สองเพื่อขอไฟล์ linethai.gif จากเซิร์ฟเวอร์อีก เมื่อเซิร์ฟเวอร์ส่งไฟล์ linethai.gif ให้การเชื่อมต่อก็จะถูกตัดขาดลงอีกกครั้ง การทำงานของเว็บเบราเซอร์จะเป็นอย่างนี้ไปเรื่อยๆ

สังเกตการทำงานของเว็บเบราเซอร์ได้ที่เปอร์เซ็นต์การโหลดข้อมูลที่แถบสถานะ ซึ่งจะเริ่มการโหลดจาก0แสดงว่ามีการเชื่อมต่อได้และเซิร์ฟเวอร์จะทยอยส่งข้อมูลรูปกราฟิกมาให้เว็บเบราเซอร์

2.12.2 โครงสร้างโพรโตคอล HTTP

โพรโตคอล HTTP อยู่บนพื้นฐานของไคลเอนต์/เซิร์ฟเวอร์ที่ต้องมีการร้องขอและการตอบสนอง โพรโตคอลนี้อาศัยการเชื่อมต่อผ่านทางโพรโตคอล TCP/IP อีกทีหนึ่งโดยใช้พอร์ตหมายเลข 80 เป็นมาตรฐานในการติดต่อ แต่จะทำให้เกิดความลำบากต่อผู้ใช้ที่ต้องระบุหมายเลขพอร์ตลงใน URL ด้วย

ข้อมูลที่รับส่งกันในแต่ละครั้งไม่ได้มีเฉพาะข้อมูลเพียงอย่างเดียว แต่ละฝ่ายจะต้องเพิ่มส่วนที่เรียกว่า เฮดเดอร์ HTTP(HTTP Header) เข้าไปในส่วนหัวของข้อมูลด้วย โดยจะใช้เป็นตัวบ่งบอกว่าข้อมูลที่ส่งหลังจากนี้เป็นอะไรเป็นการร้องขอจากไคลเอนต์หรือเป็นข้อมูลตอบสนองจากเซอร์เวอร์ ในบางครั้งจะเรียกส่วนนี้ว่า Meta Information

2.10.3 ข้อความร้องขอ (request)

การส่งข้อมูลไปยังเซิร์ฟเวอร์เพื่อบอกการขอข้อมูล สามารถทำได้หลายแบบ ดังนี้

- **GET** เป็นการขอให้เซิร์ฟเวอร์ส่งไฟล์มาให้
- **HEAD** เป็นการร้องขอเพื่อถามว่า มีไฟล์ที่ต้องการอยู่ในเซิร์ฟเวอร์หรือไม่แต่ไม่ต้องการให้ส่งไฟล์จริงมาให้
- **POST** เป็นการร้องขอให้เซิร์ฟเวอร์รับข้อมูลจากไคลเอนต์

1) การร้องขอแบบ GET

กรณีที่เว็บเบราเซอร์ขอเปิดเอกสารแบบ GET เมื่อเว็บเบราเซอร์สร้างซ็อกเก็ตเชื่อมต่อกับเซิร์ฟเวอร์ได้แล้วจะสร้างข้อความร้องขอขึ้นมาเพื่อส่งไปยังเซิร์ฟเวอร์ ข้อความในบรรทัดแรกสุดของ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เฮดเดอร์จะประกอบด้วย 3 ส่วน คือ วิธีการร้องขอ (method) ,ไดเรกทอรีกับชื่อไฟล์ที่ร้องขอและเวอร์ชันของ HTTPที่ผู้ร้องขอใช้อยู่โดยแต่ละส่วนจะถูกคั่นด้วยช่องว่าง มีรูปแบบดังนี้

GET /path/file HTTP/x.x

สังเกตได้ว่าการขอข้อมูลจากเซิร์ฟเวอร์จะระบุเฉพาะไดเรกทอรีและชื่อไฟล์ที่ต้องการ ไม่มีการบอกชื่อโฮสต์และโดเมนเพราะเว็บเบราเซอร์ได้สร้างการเชื่อมต่อกับโฮสต์เอาไว้แล้ว เราเรียกส่วนที่ระบุ /path/file นี้ว่า **URI** (Uniform Resource Identifier)

2) การร้องขอแบบ HEAD

การร้องขอแบบนี้มีประโยชน์ในการตรวจสอบว่า มีไฟล์ที่ต้องการอยู่ในเซิร์ฟเวอร์หรือไม่ หรือใช้ตรวจสอบความสมบูรณ์ของลิงก์ก็ได้และอาจมีข้อมูลอื่นส่งเข้ามาในเฮดเดอร์ด้วยเช่น วันแก้ไขไฟล์ครั้งสุดท้าย แต่การถามว่ามีไฟล์ที่ต้องการอยู่ในเซิร์ฟเวอร์หรือไม่นั้น เราสามารถใช้การร้องขอแบบ GET ก็ได้ แล้วเช็กรหัสตอบสนองจากบรรทัดสถานะถ้ามีไฟล์อยู่จะได้รหัสเป็น 200 ถ้าไม่มีจะได้รหัสตอบสนองเป็น 404 แต่การทำงานของการร้องขอแบบ HEAD จะทำงานได้คำตอบเร็วกว่า

โปรแกรมจะทำงานโดยการกำหนดชื่อไฟล์ที่ต้องการจากเซิร์ฟเวอร์ จากนั้นโปรแกรมจะวิ่งไปเช็กกับเซิร์ฟเวอร์เป็นช่วงๆ ตามระยะเวลาที่ตั้งไว้ หากพบว่ามีการปรับปรุงคือวันที่ของไฟล์ใหม่กว่าที่เก็บอยู่ในเครื่อง โปรแกรมก็จะดาวน์โหลดไฟล์นั้นมาทับในไฟล์เดิมในเครื่อง(ร้องขอไฟล์ด้วยการร้องขอแบบ GET)

3) การร้องขอแบบ POST

จะใช้ในการส่งข้อมูลจากไคลเอนต์ไปยังเซิร์ฟเวอร์คือจะส่งข้อมูลให้เซิร์ฟเวอร์รับไปทำงาน มีรูปแบบดังนี้ POST /path/file HTTP/1.0

การร้องขอแบบ POST จะมีเงื่อนไขดังนี้

- ข้อมูลที่จะส่งให้เซิร์ฟเวอร์จะอยู่ในบล็อกข้อมูล ดังนั้นจึงต้องมีเฮดเดอร์เพื่อบอกรายละเอียดของข้อมูลในบล็อกข้อมูลแนบไปด้วย
- /path/file คือชื่อโปรแกรม CGI ในเซิร์ฟเวอร์ที่จะทำหน้าที่รับข้อมูลไปประมวลผล
- ข้อความตอบสนองที่เซิร์ฟเวอร์ส่งกลับไปให้ไคลเอนต์ จะได้มาจากการทำงานของโปรแกรมหรือCGI ในเซิร์ฟเวอร์นั้น ดังนั้น CGI ที่รับข้อมูลไปจึงต้องทำหน้าที่ส่งข้อความตอบกลับไปให้ไคลเอนต์นอกจากนี้การส่งข้อมูลไปยังเซิร์ฟเวอร์ก็สามารถใช้การส่งแบบ GET ได้แต่มีข้อจำกัดในเรื่องความยาวของข้อมูลที่จะส่งไปให้เซิร์ฟเวอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.13 CGI (Common Gateway Interface)

CGI คือรูปแบบของการประมวลผล Application บนฝั่ง เซิร์ฟเวอร์ โดยผู้ใช้สามารถเรียกใช้ได้จากฝั่ง ไคลเอนต์ ได้ จากนั้นผลลัพธ์ที่ได้จากการประมวลผลจะถูกส่งกลับไปให้กับผู้ใช้ที่ฝั่ง ไคลเอนต์ โดยการสื่อสารข้อมูลระหว่างฝั่งไคลเอนต์ และเซิร์ฟเวอร์ นี้จะอาศัย โปรโตคอล HTTP

ซี จี ไอ สามารถเขียนขึ้นได้โดยภาษาโปรแกรมมิ่ง [Programming Language] หลายชนิดเช่น Visual Basic, C, C++, Perl หรือ UNIX Shell Script เป็นต้น เราสามารถแบ่งภาษาโปรแกรมมิ่งเหล่านี้ ได้เป็น 2 ประเภท คือ

1. ภาษาโปรแกรมมิ่งที่ไม่ต้อง Compile
2. ภาษาโปรแกรมมิ่งที่ต้อง Compile

ข้อแตกต่างของภาษาทั้ง 2 ชนิดนี้ก็คือ ภาษาที่ไม่ต้อง Compile จะสามารถทำการปรับปรุงเปลี่ยนแปลงแก้ไขได้ง่าย เมื่อทำการแก้ไขแล้วก็ไม่ต้อง Compile ใหม่ แต่การทำงานจะช้ากว่าภาษาที่ต้อง Compile ดังนั้นการเลือกใช้ภาษาใดนั้น ก็ขึ้นอยู่กับความสะดวก ของผู้ใช้งาน และความเหมาะสมกับลักษณะงานที่ต้องการ ในการใช้ CGI โปรแกรมโดยทั่วไป นิยม เก็บโปรแกรมที่ Compile แล้ว หรือ Script ไว้ที่ Directory ที่มีชื่อว่า cgi-bin ส่วนในกรณีที่ใช้ภาษาที่ต้อง Compile ก็นิยมเก็บ Source-code ไว้ที่ Directory ที่มีชื่อว่า cgi-src

ในการทำงานของ CGI จะมีสื่อสารระหว่าง ไคลเอนต์ และฝั่ง เซิร์ฟเวอร์ โดยจะมีตัวแปรสำหรับเก็บข้อมูลการติดต่อต่างๆ เรียกว่า Environment Variable ซึ่งตัวแปรแต่ละตัว นี้ก็จะ มีหน้าที่สำหรับเก็บข้อมูลต่างๆ แตกต่างกันไป ตัวแปรแวดล้อมในแต่ละชุด จะแบ่งได้เป็น 3 กลุ่มตามข้อมูลที่เก็บคือ กลุ่มข้อมูลที่เก็บข้อมูลของเซิฟเวอร์ กลุ่มข้อมูลของไคลเอนต์ และรายละเอียดการร้องขอของไคลเอนต์

1) กลุ่มที่ใช้เก็บข้อมูลของเซิฟเวอร์

SERVER_SOFTWARE : ชื่อโปรแกรมเว็บเซิฟเวอร์ เช่น Netscape-Enterprise/2.01

SERVER_NAME : ชื่อของเซิฟเวอร์ เช่น Sambar42

SERVER_PROTOCAL : บอกชื่อและเวอร์ชันโปรโตคอลที่เซิฟเวอร์ ใช้ เช่น HTTP/1.0

SERVER_PORT : หมายเลขพอร์ตของเซิฟเวอร์ เช่น 80

2) ที่ใช้เก็บข้อมูลของไคลเอนต์

HTTP_ REFERER : URL ของเว็บเพจที่ข้อมูลถูกส่งมา

HTTP_USER_ AGENT : ชื่อโปรแกรมที่ไคลเอนต์ใช้ร้องขอมายังเซิฟเวอร์ เช่น Mozilla/4.03

HTTP_ACCEPT : รายการประเภทข้อมูลที่ไคลเอนต์สามารถรองรับและเข้าใจเช่น

Image/gif ,image/x

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

HTTP_ACCEPT_LANGUAGE : บอกรหัสภาษาตามมาตรฐาน ISO ที่ไคลเอนต์ ใช้และเข้าใจ

3) กลุ่มที่ใช้เก็บข้อมูลการร้องขอของไคลเอนต์

REMOTE_HOST : ชื่อโฮสต์และโดเมนเนมของเครื่องที่ร้องขอ หากไม่มีตัวแปรนี้ จะเก็บหมายเลข IP แอดเดรสแทน

REMOTE_ADDR : หมายเลข IP แอดเดรสของเครื่องที่ร้องขอ

REQUEST_METHOD : เมธอดการร้องของขอที่เรียกมายังเซิฟเวอร์

SCRIPT_NAME : ชื่อสคริปต์ CGI ในเซิฟเวอร์ที่ถูกเรียกให้ทำงาน

QUERY_STRING : ข้อมูลที่ผู้ร้องขอส่งมาให้(ในกรณีร้องขอด้วยเมธด GET)

ซึ่งค่าตัวแปรต่างๆ เหล่านี้ ผู้ใช้งานสามารถนำไปใช้สำหรับ Application ต่างๆ ได้ตามต้องการ

2.13.1 การใช้งาน CGI ที่เป็นที่นิยมในปัจจุบัน

Imagemap คือการรับค่าพิกัด x,y บนภาพที่ผู้ใช้กด เมาส์ เพื่อส่งต่อให้กับ Application เพื่อทำงานตามที่ได้กำหนดไว้โดยมี File ที่เรียกว่า Map อยู่ที่ เซิร์ฟเวอร์เพื่อรับค่าพิกัด x,y แล้วพิจารณาค่า Resource ต่างๆ ตามที่ได้กำหนดไว้ แต่ในปัจจุบัน สามารถทำ Imagemap ได้โดยใช้ Tag ของ HTML ซึ่งสามารถรับค่าพิกัด และพิจารณาเรียกใช้ Resource ที่กำหนดได้ที่ฝั่ง ไคลเอนต์ ไม่ต้องมีการประมวลผลที่ CGI Application คือการสร้าง Application เพื่อทำงานอย่างหนึ่งอย่างใด แล้วส่งผลลัพธ์ ในรูปแบบต่างๆ ให้กับไคลเอนต์ ที่ได้ร้องขอมา เช่นการสร้างภาพเพื่อแสดงจำนวนผู้เข้า ใช้งาน หรือที่เรียกว่า Counter เป็นต้น

Form คือการสร้างแบบฟอร์มขึ้นมาโดยใช้ HTML จากนั้นเมื่อผู้ใช้ ได้กรอกข้อมูล แล้วส่งกลับมาที่ HTP-Server แล้ว Application ที่ฝั่ง Server ก็จะรับข้อมูลดังกล่าวเพื่อทำการประมวลต่อไป

2.13.2 การส่งข้อมูลของ CGI มี 2 วิธีที่เป็นที่นิยมในปัจจุบัน คือ

การส่งโดยวิธี GET [GET Method]

การส่งโดยวิธี POST [POST Method]

ในการส่งข้อมูลจาก Client มายัง Server ในแต่ละครั้ง โปรโตคอล HTTP จะระบุวิธี การที่ส่งในตัวแปรที่มีชื่อว่า REQUEST_METHOD ซึ่งนอกจาก Method GET และ Method POST นี้แล้วยังอาจมี Method อื่นๆ สำหรับการใช้งานเฉพาะอย่างใดอย่างหนึ่ง ซึ่งจะไม่พูดถึงในที่นี้ Method GET และ Method POSTนั้นแต่ละวิธีนั้นมีข้อดี ข้อเสีย และวิธีการส่งข้อมูลแตกต่างกันไป ดังนี้

1)การส่งข้อมูลโดยวิธี GET

ในการเรียกใช้ URL โดยปกติก็เป็นการใช้วิธี GET อยู่แล้ว การส่งข้อมูลของวิธี GET จะส่งข้อมูลรวบไปกับ URL โดยจะตามหลัง URL คั่นด้วยเครื่องหมายปรัศนี (?) จากนั้นจะตามด้วยชุดของข้อมูลเป็นคู่ๆ แต่ละคู่ประกอบด้วยชื่อคีย์ [Key name] และค่าของคีย์ [Key Value] ข้อมูลแต่ละคู่จะถูกคั่นแยกจากกันด้วยเครื่องหมายปละ [&] ข้อมูลต่างๆ ในส่วนของ Key Value จะถูกทำการแปลงเล็ก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

น้อยเรียกว่าการทำ URL Encode จากนั้นข้อมูลจะถูกส่งไปอยู่ในตัวแปรชื่อ QUERY_STRING ซึ่งการแปลงข้อมูลต่างๆทำได้ดังนี้

- เครื่องหมายเว้นวรรค จะถูกเปลี่ยนเป็นเครื่องหมาย +
- ตัวอักษรพิเศษต่างๆ จะถูกเปลี่ยนให้เป็นเครื่องหมาย % ตามด้วยเลขฐาน 16ซึ่งเป็นรหัสแอสกี [ASCII] ของอักษรพิเศษนั้นๆ เช่นเครื่องหมาย ! จะถูกแปลงเป็น %23 เป็นต้น

ข้อดีของการส่งข้อมูลโดยวิธี GET คือ สามารถสร้าง Application ให้รับข้อมูลได้ง่าย ทั้งนี้สามารถรับข้อมูลได้ทั้งตัวแปร QUERY_STRING ได้โดยตรง

เนื่องจากการส่งข้อมูลด้วยวิธี GET จะทำการส่งข้อมูลรวมไปกับ URL และข้อมูลจะถูกนำไปเก็บในตัวแปรที่มีชื่อว่า QUERY_STRING ดังนั้นข้อมูลที่สามารถส่งได้จึงมีขนาดจำกัด โดยมีขนาดไม่เกิน 256 ตัวอักษร ดังนั้นสำหรับแบบฟอร์มขนาดใหญ่ ที่ต้องส่งข้อมูลจำนวนมากกลับไปให้ Server จึงไม่สามารถใช้การส่งข้อมูลด้วยวิธี GET นี้ได้

2) การส่งข้อมูลด้วยวิธี POST

การส่งข้อมูลด้วยวิธี POST นี้สามารถแก้ปัญหาข้อมูลที่มีขนาดใหญ่ ไม่สามารถส่งโดยวิธี GET ได้ โดยการส่งข้อมูลด้วยวิธี POST นี้จะทำการส่งจำนวนตัวอักษรของผู้ที่จะส่งไปให้กับ Server โดยใช้ตัวแปรที่มีชื่อว่า COUNTENT_LENGTH หลังจากนั้น Application ที่ฝั่ง Server ก็จะสามารรับข้อมูลได้จาก STDIN [Standard Input] ที่ HTTP Server กำหนดขึ้นมาใหม่เป็นการชั่วคราว โดยรับข้อมูลในปริมาณเท่ากับที่ได้กำหนดไว้ในตัวแปรชื่อ COUNTENT_LENGTH นั่นเอง ดังนั้นการส่งข้อมูลโดยวิธี POST นี้จึงสามารถส่งข้อมูลเป็นจำนวนได้ไม่จำกัด

ในการเขียน Application สำหรับข้อมูลซึ่งส่งมาจากฝั่ง Client โดยวิธี GET และ POST จึงมีความแตกต่างกันซึ่งในที่นี้จะขอยกตัวอย่าง วิธีการเขียนโปรแกรมสำหรับการรับข้อมูลด้วยภาษา C บน Unix

2.13.3 การทำงานของ CGI

เมื่อ Client มีการร้องขอ URI มาที่ ซีจีไอ โปรแกรมที่ฝั่ง เซิร์ฟเวร์ แล้วเซิร์ฟเวร์ จะทำการประมวลผลโปรแกรมดังกล่าวแบบ Realtime โดยที่ ซีจีไอ โปรแกรมสามารถรับข้อมูล ต่างๆจากฝั่งไคลเอนต์ ได้จากตัวแปรที่เรียกว่า Environment Variable ดังที่ได้กล่าวไว้ในตอนต้น จากนั้นก็จะส่งผลลัพธ์ที่ได้ กลับไปที่ฝั่ง Client โดยผลลัพธ์ที่ส่งกลับไปนี้ มีได้หลายรูปแบบ ไม่ว่าจะเป็นเอกสาร HTML, รูปภาพ, ตัวอักษร, เสียง, ภาพเคลื่อนไหวเป็นต้น การที่ฝั่ง Client จะแสดงผลลัพธ์ประเภทต่างๆ ได้อย่างถูกต้องนั้น CGI โปรแกรมจึงต้องมีการกำหนดชนิดของผลลัพธ์ที่ได้ ไว้ในส่วนที่เรียกว่า ส่วนหัว [Header] ซึ่งข้อมูลในส่วนนี้จะมีรูปแบบดังนี้

- เป็นตัวอักษรในรหัส ASCII

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- เก็บข้อมูลบรรทัดละ 1 ข้อมูล
- ส่วนหัวจะถูกแยกออกจาก ส่วนผลลัพธ์ โดยใช้บรรทัดว่างคั่น 1 บรรทัด

ข้อมูลในส่วนหัวมีหลายข้อมูล เช่น

1) Content-Type : ใช้สำหรับระบุชนิดของผลลัพธ์ที่ได้ด้วย MIME Type [Multipurpose Internet Mail Extension] ทั้งนี้เพื่อเป็นการบอกให้ส่วนของBrowser รับทราบชนิดของข้อมูลว่าข้อมูลที่ได้รับจาก Server นั้นเป็นข้อมูลประเภทใด

Location : ใช้ระบุที่อยู่ของผลลัพธ์ ที่จะอ้างถึงโดยใช้ URL

Status : ใช้สำหรับบอกให้ Server ส่งข้อมูลสถานะไปให้ Client ซึ่งตามปกติ ในส่วนนี้ตัว Server จะเป็นตัวจัดการสร้างขึ้นให้เองโดยอัตโนมัติ เช่นในกรณีที่หา Resource ที่อ้างถึงไม่พบ จะส่งรหัส 404 หรือผู้เรียกใช้ ไม่มีสิทธิจะส่งรหัส 403 เป็นต้น

2) Content-Length : ใช้ระบุความยาวของไบต์ข้อมูลในบล็อกข้อมูลมีหน่วยเป็นไบต์ เพื่อที่ผู้รับจะได้ทราบว่ามีข้อมูลส่งมาให้กี่ไบต์

Windows ซึ่งการทำงานของ Windows ในCGI จะแตกต่างจาก CGI เดิมเล็กน้อยซึ่งสามารถอธิบายได้ดังนี้

ในระบบปฏิบัติการ Windows นั้นไม่มีตัว Interpreter เหมือนใน Unix ดังนั้นจึงไม่สามารถเขียน CGI เป็น Script ได้ การเขียน CGI บน Windows จึงต้องเป็น Executable Program คือต้องมีการคอมไพล์ให้เป็นไฟล์ .EXE ก่อน อย่างไรก็ตาม เราก็สามารถที่จะเรียกใช้ ขึ้นมา 3 ไฟล์ คือ CGI ที่เป็น Bacth File ซึ่งสามารถเอ็กซีคิวต์ได้โดยอาศัย Shell ของ DOS เช่น 4DOS,NDOS ก็ได้

การทำงานของ Windows CGI เมื่อมีการเรียกใช้ Executable Program ใดๆ จะมีการสร้างไฟล์ชั่วคราว

1) CGI-DATA-FILE

เป็นไฟล์ชั่วคราว ที่ใช้เก็บข้อมูลต่างๆ ที่ใช้ในการติดต่อสื่อสาร ระหว่างฝั่ง Client และServer ภายในไฟล์นี้จะประกอบด้วยส่วนต่างๆ ดังนี้

- CGI เป็นส่วนที่เก็บค่า Environment Variable ต่างๆ
- Accept แสดงชนิดของข้อมูลที่ Client สามารถรับได้
- System เก็บ Full Path/name ของ CONTENT-FILE และ OUTPUT-FILE
- Extra Headers เก็บข้อมูลพิเศษที่มาพร้อมกับ URL เช่นใน Method Get จะมีข้อมูลในส่วนนี้
- Form Literal ข้อมูลจาก FORM ที่ส่งมาโดย Method POST จะถูก Decode และเก็บเป็นคู่ๆ ของ Key=Value ไว้ที่ส่วนนี้
- Form External เก็บชื่อของ CONTENT-FILE และความยาวของข้อมูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- Form Huge เก็บตำแหน่งของข้อมูลใน CONTENT-FILE และความยาวของข้อมูล

2)CONTENT-FILE

เป็นไฟล์ชั่วคราวที่เก็บข้อมูลที่ส่งมาจากฝั่ง Client ในการประมวลผลของโปรแกรมที่ฝั่ง Server สามารถที่นำข้อมูลนี้ไปใช้งานได้

3) OUTPUT-FILE

ในการประมวลผลของโปรแกรมที่ฝั่ง Server ข้อมูลส่วนใดที่ต้องการเป็นผลลัพธ์ส่งกลับไปที่ฝั่ง Client จะถูกนำมาเขียนใส่ OUTPUT-FILE ก่อน และเมื่อโปรแกรมทำงานเสร็จสิ้นแล้ว HTTP Server ก็จะส่ง OUTPUT-FILE นี้ไปให้แก่ฝั่ง Client

ไฟล์ชั่วคราวทั้ง 3 ไฟล์นี้จะถูก HTTP Server ส่งผลลัพธ์กลับไปให้ Client แล้ว ดังนั้น ในการสร้าง ซีจีไอ แอพพลิเคชั่น ขึ้นบนวินโดว์ จะต้องทำการอ่านข้อมูล และเขียนผลลัพธ์ลงที่ไฟล์ที่กล่าวถึงนี้ ไม่สามารถจะรับค่าตัวแปรต่างๆ โดยตรงได้ อย่างไรก็ตาม ได้มีการสร้างเฟรมเวิร์ก สำหรับรับข้อมูลจากไฟล์ชั่วคราวนี้ โดยมีทั้งที่เป็นวิชวลเบสิค และเดลไฟ ซึ่งเฟรมเวิร์กนี้จะทำการอ่านข้อมูลจาก ซีจีไอ ดาต้าไฟล์ เพื่อกำหนดตัวแปรต่างๆ และกำหนดว่าจะต้องส่งผลลัพธ์ไปยังไฟล์ใด นอกจากนี้ยังมีส่วนของการรับข้อมูล จากแบบฟอร์มให้ใช้งานได้ด้วย เราสามารถใช้งานเฟรมเวิร์กนี้ ได้โดยการนำไปรวมเข้ากับส่วนของซีจีไอ แอพพลิเคชั่น ที่สร้างขึ้นได้ทันที

2.14 เพิร์ล

perl มาจากคำเต็มว่า Practical Extraction and Report Language เป็นภาษาโปรแกรมมิ่งที่สร้างโดยนาย Larry Wall จุดประสงค์ในการสร้างครั้งแรกก็เพื่อเป็นเครื่องมือ สำหรับการเขียนโปรแกรมในสภาวะแวดล้อมของระบบ ยูนิกส์ โดยเฉพาะ เป็นภาษาที่มีความสามารถและมีความยืดหยุ่นเหมือนภาษาชั้นสูงทั่วๆไป ความสามารถของ เพิร์ล บางอย่างก็นำมาจากภาษา ซี มาใช้ ทำให้ผู้ที่มีพื้นฐานทางภาษาซี สามารถเขียน เพิร์ลได้ง่ายยิ่งขึ้น

เพิร์ล เป็นภาษาแบบ สคริปต์ การรันสคริปต์จะต้องเรียก อินเทอร์พรีเตอร์ ของ เพิร์ล มาอ่านสคริปต์เพื่อตีความหมายและทำงานตามที่เขียนไว้ในสคริปต์ ไม่ต้องมีการคอมไพล์ ซอร์ซโค้ดของสคริปต์ให้เป็นโปรแกรมก่อนการรัน ดังนั้น เพิร์ล จึงมีความสะดวกและความรวดเร็วในการแก้ไขสคริปต์

ภาษา เพิร์ล ได้รับความนิยมอย่างรวดเร็วเพราะกระแสของอินเทอร์เน็ต ได้เข้าไปสู่ผู้ใช้เกือบทุกระดับ เซิร์ฟเวอร์ ยูนิกส์จำนวนมากต้องแปลงสภาพให้เป็นเซิฟเวอร์สำหรับให้บริการอินเทอร์เน็ตไปด้วย การที่ผู้ใช้จะติดต่อกับฐานข้อมูลหรือทำการใดๆ ในเซิฟเวอร์ได้ จะต้องผ่านการทำงานของ CGI (Common Gateway Interface) ที่จัดเตรียมไว้ในเซิฟเวอร์ ตัว ซีจีไอ จะถูกเขียนจากภาษา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โปรแกรมมิ่งอย่างซี หรือเขียนจาก เชลล์ของยูนิกส์ ก็ได้ ขึ้นอยู่กับว่า เซิฟร์เวอร์ สนับสนุนให้ใช้งานอะไรได้บ้าง แต่เนื่องจากภาษาซี และ เพิร์ล มักจะถูกติดตั้งมาพร้อมกับเซิร์ฟเวอร์ ยูนิกส์ อยู่แล้ว โปรแกรมเมอร์ส่วนใหญ่จึงนิยมใช้ 2 ภาษานี้ในการเขียนโปรแกรมหรือสคริปต์ให้ทำงานเป็น ซีจีไอ เพิร์ลในยูนิกส์ และดอส

ปัจจุบัน เพิร์ลได้มีการพัฒนาให้รันในระบบการปฏิบัติการอื่นๆ อีกด้วยเช่น เพิร์ล มีเวอร์ชันที่เรียกว่า Win32 สำหรับในระบบการปฏิบัติการ วินโดวส์ หรือ ดอส แต่ไม่ว่าจะรันในระบบปฏิบัติการใดก็ตามโครงสร้างการเขียนสคริปต์โดยรวมล้วนเหมือนกันหมด จะต่างกันก็ตรงที่ว่าเราไม่สามารถใช้คำสั่งบางคำสั่งของ เพิร์ลได้ในทุกระบบเพราะเพิร์ลเกิดมากับยูนิกส์จึงมีคำสั่งที่จำนวนหนึ่งที่เกี่ยวโยงกับยูนิกส์ คำสั่งกลุ่มนั้นจึงไม่สามารถนำมาใช้ใน เพิร์ล ที่รันในวินโดวส์ หรือ ดอสได้

สำหรับเพิร์ล เวอร์ชันในดอส นั้นปัจจุบันมีการสร้างโปรแกรมสำหรับคอมไพล์สคริปต์ เพิร์ล ให้เป็นไฟล์นามสกุล .exe ชื่อว่า Perl2Exe จึงเป็นอีกทางเลือกหนึ่งสำหรับ เพิร์ลเวอร์ชันในดอส ที่สามารถแจกจ่ายโปรแกรมที่เขียนขึ้นไปให้ผู้อื่นใช้งานได้โดยไม่ต้องให้ซอร์ซโค้ด หรือเนื้อหาสคริปต์ต้นฉบับ

อีกอย่างที่ควรรู้คือ นามสกุลของไฟล์สคริปต์เพิร์ล จะเป็นอะไรก็ได้ เพราะนามสกุลของไฟล์สคริปต์ เพิร์ลไม่มีความหมายใดๆต่อการรัน (ไม่ว่าจะเป็นในยูนิกส์หรือในดอส) แต่ที่เห็นใช้กันบ่อยๆ คือ .pl หรือ .cgi ซึ่งนามสกุลเป็นเพียงส่วนขยายให้เห็น และรู้ว่าเป็นสคริปต์ของไฟล์เพิร์ลเท่านั้นเอง

โครงสร้างพื้นฐานของสคริปต์เพิร์ล

โครงสร้างการเขียนและวางคำสั่งของ เพิร์ลนั้น บางส่วนจะคล้ายกับภาษาซี ซึ่งมีส่วนสำคัญดังนี้

- เมื่อจบประโยคคำสั่งหรือจบ statement จะต้องมีเครื่องหมาย ;ต่อท้าย
- ข้อความที่อยู่หลังเครื่องหมาย # ในบรรทัดแรกจนหมดบรรทัดถือว่าเป็นหมายเหตุหรือ comment
- ไม่สามารถเขียนเครื่องหมาย # เพื่อกำหนดข้อความหมายเหตุในสคริปต์คลุมเป็นช่วงเหมือนภาษาซี หรือ ปาสคาลได้ หากมีข้อความหมายเหตุหลายบรรทัดต้องเขียนเครื่องหมาย # กำกับหน้าข้อความเหล่านั้นทุกบรรทัด

การรันสคริปต์เพิร์ล ต้องเรียกเอาอินเทอร์พรีเตอร์ของเพิร์ลมาตีความหมายสคริปต์ที่เราเขียน สมมุติว่าเราเก็บไว้ในไฟล์ชื่อ hello.pl

การรันสคิปต์วิธีแรกนี้ใช้ได้ทั้งยูนิกส์และดอส คือใช้คำสั่ง perl <ชื่อสคริปต์ที่ต้องการรัน>

วิธีที่สอง ใช้ได้กับระบบยูนิกส์เท่านั้น คือเวลารันสคริปต์ให้ระบุเพียงชื่อสคริปต์ที่ต้องการรันได้เลยโดยไม่ต้องระบุอินเทอร์รีเตอร์ perl หน้าชื่อสคริปต์ แต่ก่อนจะรันสคริปต์ด้วยวิธีนี้ต้องทำ2 อย่างต่อไปนี้คือ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- ในบรรทัดแรกสุดของสคริปต์ให้เพิ่มข้อความ #! และตามด้วยไดเรกทอรีที่เก็บอินเทอร์พรีเตอร์ของเพิร์ล เช่น #!/usr/local/bin/perl
- เปลี่ยนแอตทริบิวต์ ของไฟล์สคริปต์ ให้สามารถรันได้ ด้วยคำสั่ง chmod 755<ชื่อไฟล์สคริปต์>

ในการตัดสินของยูนิกส์ ว่าไฟล์ไหนรันได้หรือไม่ได้ในเบื้องต้นนั้นไม่ได้ดูจากนามสุล ซึ่งต่างกับดอสที่มองว่าไฟล์นามสกุล .exe หรือ .com เป็นไฟล์ที่รันได้ แต่ยูนิกส์มองจากแอตทริบิวต์ของไฟล์แทน โดยแอตทริบิวต์ของไฟล์จะดูได้ด้วยคำสั่ง ls –l แอตทริบิวต์ของไฟล์แต่ละไฟล์จะอยู่ตรงคอลัมน์แรกสุด ถ้าตัวอักษรแรกเป็นตัว d หมายความว่าเป็นไดเรกทอรี ตัวอักษรอีก 9 ตัวที่เหลือจะแบ่งออกเป็น 3 กลุ่ม เพื่อบอกสิทธิในการทำงานกับไฟล์หรือไดเรกทอรีนั้นๆจากผู้ใช้ 3 กลุ่มเรียงกันไปตามลำดับคือ เจ้าของไฟล์ ,ผู้ใช้ในกลุ่มเดียวกันกับเจ้าของไฟล์ และผู้ใช้รายอื่นๆ สิทฑอในการทำงานเขียนเป็นสัญลักษณ์คือ r (readable= อ่านได้), w (writable = เขียนได้) และ x (executable = รันได้)

การเปลี่ยนแอตทริบิวต์ของไฟล์สคริบ์ต เพิร์ลให้ทำงานได้ ต้องกำหนดด้วยตัวเลขใน คำสั่ง chmod เป็นการนำค่าของตัวอักษร r ,w และ x ที่บอกสิทธิของของผู้ใช้มาบวกกัน (r = 4 , w = 2 , x = 1) เช่นถ้าต้องการให้เจ้าของไฟล์และผู้ใช้กลุ่มเท่านั้นที่สามารถอ่าน เขียน และรันไฟล์ hello.pl ได้ แอตทริบิวต์ของไฟล์จะเป็น rwxrwx--- เมื่อแปลงเป็นตัวเลขจะได้ 770 ก็ใช้คำสั่ง chmod 770 hello.pl

ดังนั้นคำสั่ง chmod 755 จึงหมายถึง เจ้าของไฟล์มีสิทธิอ่าน,เขียน และรันได้ ส่วนผู้ใช้ภายในกลุ่มและผู้ใช้อื่นๆ มีสิทธิเพียงแค่อ่านกับรันไฟล์เท่านั้น ในการทดลองเขียน และรันสคริปต์เพิร์ลเองโดยยังไม่นำสคริปต์นั้นไปทำงานเป็น CGI แค่กำหนดแอตทริบิวต์ของไฟล์เป็น 700 ก็สามารถทำงานได้แล้วซึ่งจะได้ว่า chmod นั้นเป็นคอมมานด์ไลน์ของ ยูนิกซ์

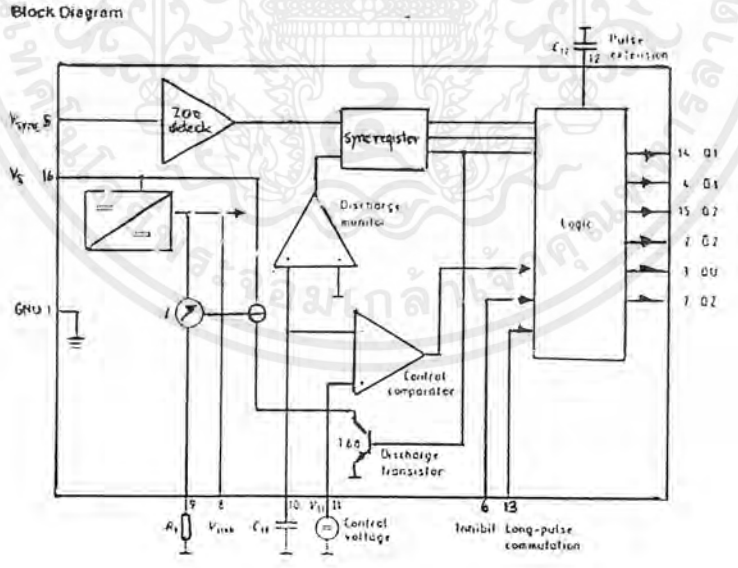
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 3
หลักการออกแบบส่วนควบคุม

3.1การควบคุมกำลังงานไฟฟ้าด้วย TCA785

ในส่วนควบคุมกำลังงานนี้ เราใช้ IC TCA785 เป็นตัวกำเนิดสัญญาณทริกไตรแอค (triac) เพื่อควบคุมเฟสในการทริก ไตรแอค การทำงาน TCA785 เริ่มด้วยการนำสัญญาณซิงค์จากแหล่งจ่ายสัญญาณความถี่เดียวกัน มาผ่านการกำจัดกระแสให้พอดีพอเหมาะก่อนเข้าขา 5 จากนั้นผ่านวงจรตรวจจับแรงดันศูนย์ (zero voltage detector) ไปยังส่วนของวงจรซิงโครไนซ์รีจิสเตอร์ (SYNC.register) ซึ่งควบคุมแรงดันแรมป์ที่เกิดจากการเก็บประจุของ C_{10}ด้วยกระแสที่คงที่ ซึ่งกำหนดความชันของสัญญาณแรมป์ด้วย R_9ถ้าสัญญาณแรมป์ที่ 10 มีระดับ แรงดันต่างไปจากแรงดันไฟตรงควบคุมที่ขา 11

ตัวควบคุมการเปรียบเทียบ (control comparator) จะผลิตสัญญาณเชิงตรรกไปยังวงจรลอจิกต่อไป ซึ่งสัญญาณลอจิกนี้สามารถควบคุมให้เลื่อนไปมาจาก 0-180 องศา ได้ในทุกๆครึ่งคลื่นจะปรากฏพัลส์ต่างๆที่เอาท์พุท ดังรูป 3.1



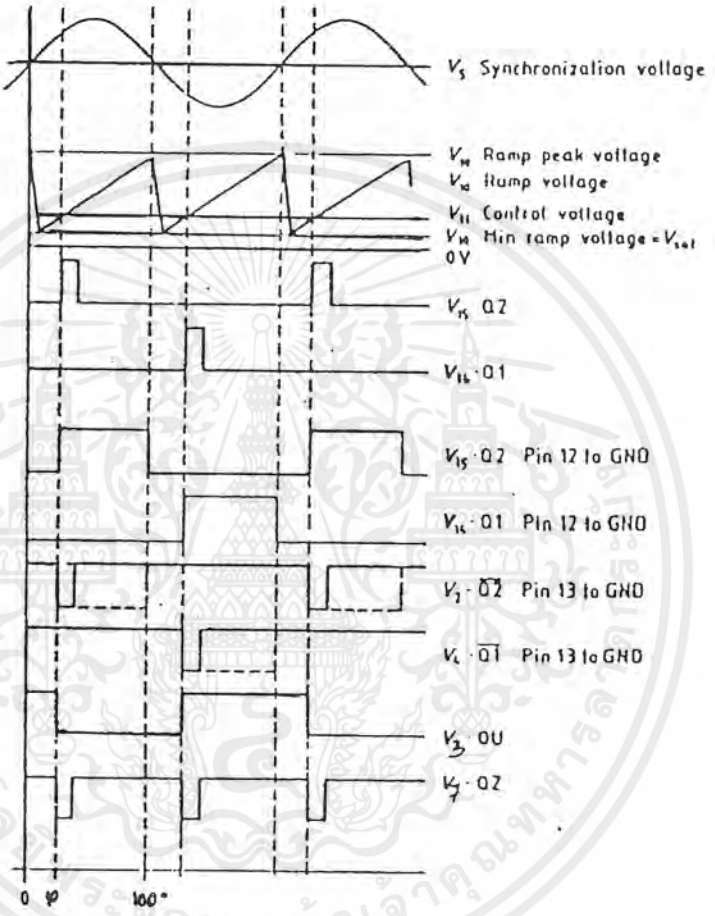
รูปที่ 3.1 แสดงลักษณะภายในของ TCA785

พัลส์ต่างๆ เหล่านี้ปรกติจะมีความกว้างพัลส์ประมาณ 30 ไมโครวินาที สามารถจะขยายความกว้างพัลส์ต่างๆ ออกไปได้อีกถึง 180องศา โดยการเปลี่ยนค่า C_{12} หรือจะลัด C_{12} ลงกราวด์ก็จะได้ความกว้างพัลส์ ตั้งแต่มุมทริกไปถึง 180 องศา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สำหรับขา 6 จะเป็นตัวปิดเปิดพัลล์ทั้งหมด หากขา 6 มีลอจิกเป็น “1”ก็จะมีพัลล์ออกตามปกติ ถ้าขา 6 มีลอจิกเป็น”0” ก็จะไม่มีพัลล์ใดๆ ออกมา

Pulse Diagram



รูปที่ 3.2 แสดงลักษณะสัญญาณที่ขาต่างๆของ TCA785

ส่วนการออกแบบเราจะทำการจ่ายไฟเลี้ยงให้ IC TCA785 12V ทำให้ได้ V_{pulse} output 12 V_p โดยความกว้างพัลล์จะกำหนดด้วย ค่า C_{12} โดยให้ค่า C =150 pF แล้วจะได้ค่าความกว้างของพัลล์ประมาณ 0.1 ms (จากการทดลอง)

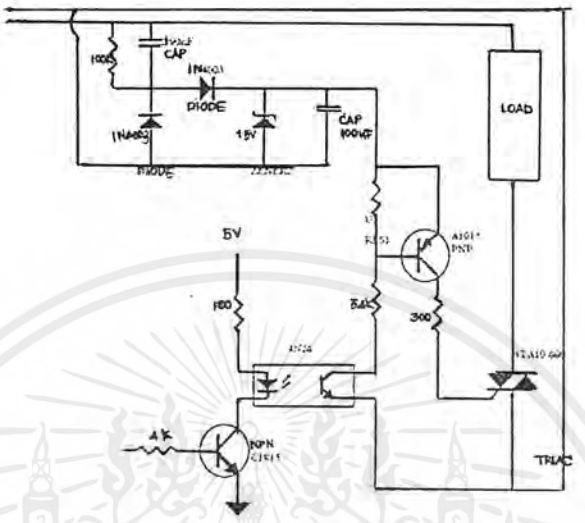
โหลดของเราจะใช้ไฟบ้านเป็น 220 V_{RMS} 50 Hz

$\therefore$ คาบเวลาของไฟบ้าน ซึ่งจะมีทั้งซีกบวกและลบ

$\therefore$ T = 1/50 Hz =20 ms

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ค่าของเอาท์พุทที่จะนำไปใช้จะต้องต่อผ่าน Opto Couple เพื่อที่ไฟบ้านซึ่งเป็นไฟสลับจะไม่สามารถไปรบกวน สัญญาณพัลส์ที่เป็น DC ได้โดยใช้วงจรดังรูปที่3.3



รูปที่ 3.3 ส่วนวงจรต่อ อ๊อปโต้คัปเปลอร์

3.2 ไตรแอค

หลักการคำนวณคุณสมบัติไตรแอคโดยพิจารณาพารามิเตอร์ต่างๆ ต้องการกระแสทริกอย่างต่ำ I_{GS} =50 mA จากรูปวงจรที่ 3.4 จะได้ว่า กระแสที่ไหลในวงจรเป็นกระแสพัลส์ซึ่งจะเกิดจาก

$$I_{รวม} = I_{ZENER} + I_{R3\,(750\,\Omega)} + I_{GT}$$

โดย $I_{ZENER} \approx 20$ mA

$I_{R3} \approx 0.536/750 = 0.8$ mA

$I_{GT} \approx 50$ mA

$\therefore \quad I_{รวม} = 71$ mA

จากที่กล่าวมาแล้วข้างต้นกระแสที่ไหลในวงจร เป็นกระแสพัลส์ซึ่งคิดค่าเป็นกระแสเฉลี่ยจะได้ว่า

$$I_{dc} = I_{รวม} * 0.1ms/10ms = 0.71mA$$

เราจึงต้องกำหนดค่า R_I เพื่อที่กระแสไฟตรงที่ไหลเข้ามาจะไม่มากจนเกินไป

$$R_I = V_{dc}/I_{dc} = (V_P\text{-}15\text{-}V_{diode}) / I_{dc}$$

$$= 132k$$

เราจะให้ค่า R_I ประมาณ 100K เพื่อที่ I_{GT} ไหลเพียงพอกับไตรแอค ซึ่งกระแสที่คำนวณได้อาจคลาดเคลื่อนจากความเป็นจริง จึงเผื่อค่าให้ I_{GT} minประมาณ 70mA คำนวณจาก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

I_{dc} = (220 ♦ 2 – 15-0.536)/100k

= $0.941*10^{-3}$

I_{PULSE} = 94.1 mA

3.3 การเปลี่ยนสัญญาณดิจิตอลเป็นอนาลอก

วงจรเปลี่ยนสัญญาณดิจิตอลเป็นสัญญาณอนาลอก (Digital to Analog Converter)เป็นวงจรรับค่าที่ส่งออกมาจาก หน่วยประมวลผลกลาง ซึ่งเป็นค่า ไบนารี 8 บิต แล้วเปลี่ยนเป็นค่าโวลท์เตจที่ส่งต่อไปยังส่วน เฟสคอนโทรล โดยผ่านวงจร Summing ซึ่งทำให้เมื่อส่งค่าจากหน่วยประมวลผลกลาง ออกมาน้อยพลังงานที่โหลดได้รับก็น้อยด้วยเมื่อส่งค่ามากพลังงานก็มาก

สำหรับวงจรที่ใช้ IC เบอร์ DAC0808 ซึ่งจะได้ค่า เอาท์พุทเป็นกระแสจะมากหรือน้อยขึ้นกับอินพุทโดยค่าที่มากที่สุดจะเกิดเมื่อส่ง อินพุทเป็น FFHค่าเอาท์พุท สูงสุดเท่ากับกระแสอ้างอิงที่ขา 14 ซึ่ง ตามที่วงจรมีค่า 2 mA เนื่องจากเอาท์พุทที่ขา 4 จาก IC เป็นกระแส ดังนั้นจึงต่อออปแอมป์เพื่อเปลี่ยนกระแสเป็นแรงดันให้ได้ค่ามากที่สุด 10V เป็นค่าแรงดันเอาท์พุท ออปแอมป์ที่นำมาต่อใช้เป็น Inverting Amp.ต่อเข้ากับเอาท์พุท

กระแสเอาท์พุทสูงสุดจะได้จากกระแสอ้างอิงของวงจรซึ่งจะให้กระแสอ้างอิงสูงสุดเป็น

I_{REF} = (5.26-0.56)/ 2.2k= 2.13mA

เมื่อทำการต่ออินเวอร์ติงแอมป์แล้วจะได้แรงดันเอาท์พุทเป็น

V_{OUT} = (5k)(2.13mA) = 10.65 V

เพราะฉะนั้น เอาท์พุทของ DAC 0808 สามารถจ่ายแรงดันได้ตั้งแต่ประมาณ 0-10 V

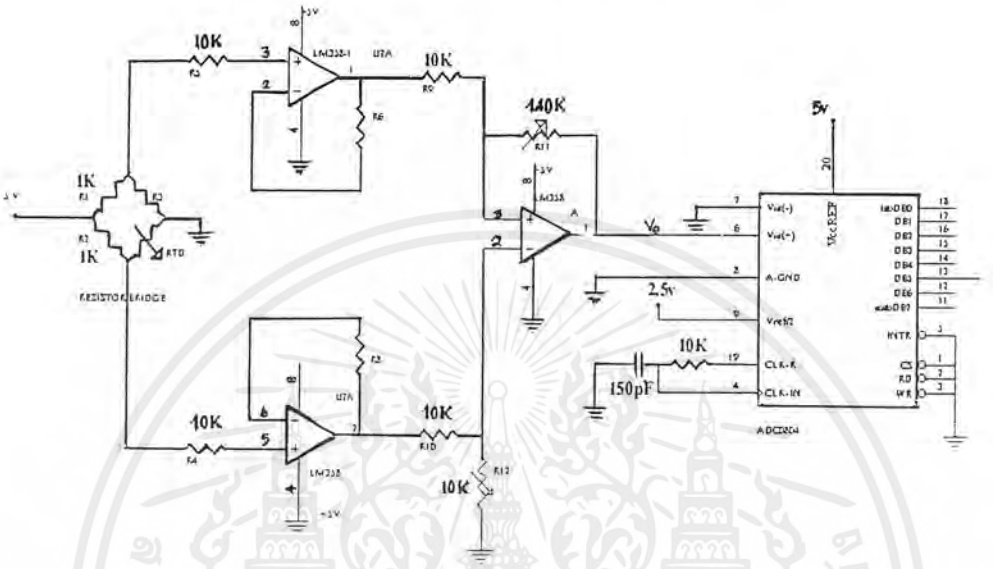
3.4 การเปลี่ยนสัญญาณอนาล็อกเป็นดิจิตอล

การทำงานของวงจร อนาล็อกทูดิจิตอลตามหลักการใหญ่ๆคือการแปลงสัญญาณ อนาล็อกเป็นสัญญาณดิจิตอล เลือกใช้ IC# ADC0804 มีการสร้างสัญญาณเวลาขึ้นเองได้ภายในตัวโดยใช้วงจร RC ต่อร่วมภายนอกที่ขา 4 กับขา 19 ทำการป้อนแรงดัน อินพุทเข้าที่ ขา 6 ซึ่งเป็นสัญญาณที่มาจากวงจรขยายอินสตรูเมนเตชั่น IC# ADC0804 ทำงานเป็นแบบ FREE RUNNING และที่ขา 20 คือขาป้อนไฟเลี้ยง จะมีค่า C 10 uFต่อลงกราวด์เพื่อกันสัญญาณรบกวนและที่ขา 8 เป็นขา ANALOG GROUND และที่ขา 10 เป็นขา DIGITAL GROUND ซึ่งจะถูกแยกออกจากัน ป้องกันสัญญาณรบกวน อินพุทเข้าขา 6 ซึ่งเป็นสัญญาณอนาล็อก0-5 โวลต์ เอาท์พุท ออกที่ขา 11-18 ขนาด 8 บิตแรงดันเปลี่ยนแปลง ทุก 20mV และถูกส่งต่อไปยัง MCS-51

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.5 การออกแบบส่วนการตรวจวัดอุณหภูมิ

การตรวจวัดอุณหภูมิโดยใช้อาร์ทีดีในโครงงานนี้เลือกใช้พลาตินัมอาร์ทีดีแบบ 2 สายเป็นตัวเซ็นเซอร์โดยใช้วงจรวิธสโตนบริดจ์เป็นวงจรตรวจวัดอุณหภูมิดังรูปที่ 3.4



รูปที่ 3.4 แสดงการออกแบบวงจรตรวจจับอุณหภูมิ

จากรูปใช้หลัก KVL จะได้

$$R_1 + R_3 = R_2 + RTD$$

และ $R_1 = R_2$ ถ้า $R_3 = RTD$

ดังนี้คือ จุดสมดุล

เพื่อให้ค่าผิดพลาดที่เกิดขึ้นจากความร้อนในตัวอาร์ทีดีมีค่าน้อย เนื่องจสกกระแสที่ไหลผ่านอาร์ทีดี โดยกำหนดค่าความต้านทานในวงจรให้เหมาะสมและให้ค่าเอาท์พุทสูงพอที่จะนำไปใช้งานได้เนื่องจากค่าของกระแสแปรผกผันกับความต้านทานเมื่อแรงดันตกคร่อมมีค่าคงที

ถ้าให้ $R_1 = R_2 = 1K$, $R_3 = 127$ ohm

$I_{MAX} = V_S / (R_2 + R_{RTD(MIN)})$

$I_{MIN} = V_S / (R_2 + R_{RTD(MAX)})$

$R_{RTD(MIN)}$ = ความต้านทานของอาร์ทีดีที่อุณหภูมิต่ำสุด (70^0 C)

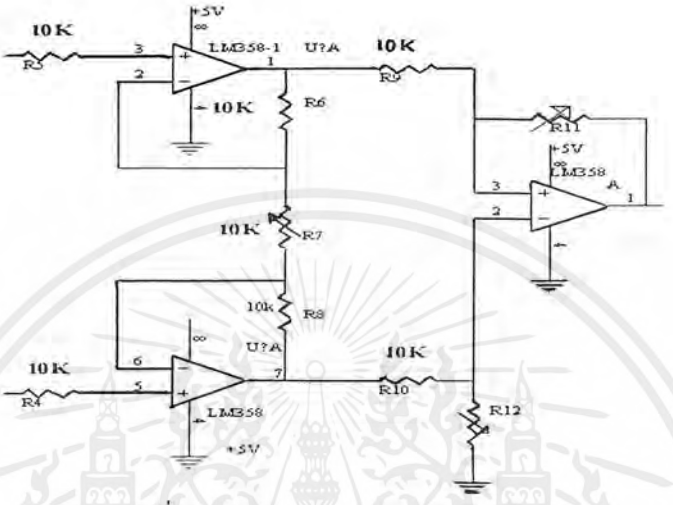
$R_{RTD(MAX)}$ = ความต้านทานของอาร์ทีดีที่อุณหภูมิสูงสุด (300^0 C) $V_{0ช}$

$= V_2 - V_1$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.6 วงจรขยายสัญญาณอินสตรูเมนเตชัน

เนื่องจากสัญญาณ V_0 ที่ได้มีค่าเล็กมากและส่วนแสดงผลเราต้องการสัญญาณที่มีขนาดใหญ่ จึงต้องทำการขยายสัญญาณโดยวงจรขยายผลต่างแบบอินสตรูเมนเตชัน ดังรูปที่ 3.5



รูปที่ 3.5 วงจร Instrument Amplifier

จากสมการ

$$V_0 / E_1 - E_2 = 1 + 2/a$$

เมื่อ a= aR/R

กำหนดให้ $R_4 = R_5 = R_6 = R_8 = R_9 = R_{10} = 10K(\text{ ohm})$

โครงงานนี้ต้องการให้วงจรขยายการวัดอุณหภูมิมีเกณฑ์ เท่ากับ 15เท่า จะได้ค่า $R_7 \approx 5K$ แบบตัวต้านทานบ่รับค่าได้เป็นตัวปรับอัตราขยาย และเลือกใช้ R_{12} เป็นตัวต้านทานปรับค่าได้ขนาด 500K ohm เป็นตัวปรับแรงดันแบบ คอมมอนโหมด

3.7 ส่วนของไมโครคอนโทรลเลอร์ MCS-51

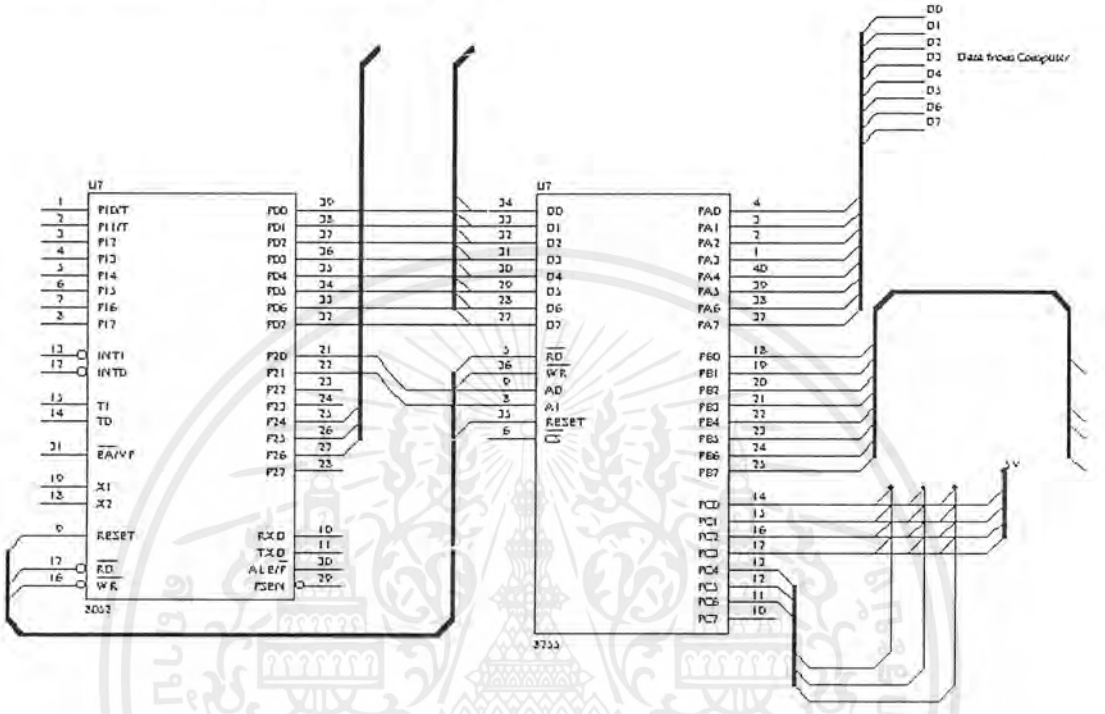
ในส่วนของการใช้ไมโครคอนโทรลเลอร์มาควบคุมการทำงานของเตาอบ การอินเตอร์เฟสกับคอมพิวเตอร์ และการสั่งงานจากคีบอร์ดเมื่อไม่มีการเชื่อมต่อกับคอมพิวเตอร์ได้ ซึ่งทำให้พิจารณาการใช้งานที่ครอบคลุมยิ่งขึ้น ดังนั้นการเชื่อมต่อในพอร์ตต่างๆของ ไมโครคอนโทรลเลอร์ จะแสดงรายละเอียดการทำงานของโปรแกรมดังนี้

3.7.1 การควบคุมการทำงานของ 8255

การควบคุมการทำงานของ 8255 ซึ่งเป็นตัวอุปกรณ์ที่ใช้ขยายพอร์ต อินพุต เอาท์พุต เพื่อให้สามารถควบคุมการทำงานของอุปกรณ์ได้มากขึ้น และใช้งานได้หลากหลายยิ่งขึ้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

พอร์ตของ 8255 จะมีอยู่ 3 พอร์ต คือ พอร์ตA พอร์ตB โดย พอร์ตC แยกเป็น 2 ส่วน คือ พอร์ต C ล่าง และพอร์ตC บน ซึ่งหน้าที่พิเศษของพอร์ตเหล่านี้คือ เป็นได้ทั้งพอร์ตอินพุต และพอร์ตเอาท์พุต โครงสร้างการทำงานของ 8255แสดงดังรูป



รูปที่ 3.6 แสดงโครงสร้างภายในและการจัดวางขาของ 8255

การใช้งาน 8255 ในที่นี้เราจะพิจารณาการใช้งานที่โหมด 0 หรืออินพุต เอาท์พุตแบบพื้นฐาน เราจึงต้องกำหนดก่อนว่าจะให้ พอร์ตA(0000H) พอร์ตB(0100H) พอร์ตC(1000H) บน และ พอร์ตC ล่าง เป็นอินพุตหรือ เอาท์พุต โดยผ่านรหัสควบคุมทางพอร์ตข้อมูล ซึ่งกำหนดให้ พอร์ตA เป็น พอร์ตอินพุต พอร์ตB เป็น พอร์ตเอาท์พุต พอร์ตC บนเป็น พอร์ตเอาท์พุต และพอร์ตC ล่างเป็น พอร์ตอินพุต เพราะฉะนั้นต้องกำหนดรหัสควบคุมเป็น (10010001H) จึงจะได้พอร์ตอินพุต เอาท์พุต ตามที่ต้องการ

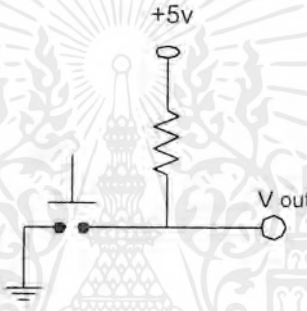
ถ้าเราต้องการใช้งาน พอร์ตA พอร์ตB และ พอร์ตC ก็สามารถ ส่ง และรับข้อมูลผ่านการอ้างอิงแอดเดรสแบบ Memory Map ได้โดยส่งผ่านรีจิสเตอร์ DPTR พอร์ตA(0000H) พอร์ตB(0100H) พอร์ตC(1000H)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.7.2 การทำงานของคีย์บอร์ด 4*3

คีย์บอร์ดจัดว่าเป็นอุปกรณ์พื้นฐานสำหรับการควบคุมการสั่งงานเตาอบ ซึ่งสามารถใช้งานได้ง่ายที่สุด การใช้คีย์บอร์ดประเภทต่างๆกับ ไมโครคอนโทรลเลอร์ ทีแตกต่างกันหลายประเภท เช่น การต่อแบบที่สวิตซ์กับพอร์ตโดยตรง แต่จะทำให้จำนวนคียบอร์ดที่มีได้สูงสุดเท่ากับจำนวนขาของพอร์ตที่มีใน ไมโครคอนโทรลเลอร์ และอีกแบบหนึ่งคือ การต่อแบบเมตริกซ์ วิธีนี้จะมีการเขียนโปรแกรมที่ซับซ้อน และยุ่งยากกว่าแบบแรกมาก

การต่อคีย์บอร์ดกับไมโครคอนโทรลเลอร์ โดยคีย์บอร์ดแต่ละตัวจะสร้างลอจิก "0" หรือ ลอจิก "1" และจะถูกนำไปต่อกับบิตข้อมูลของพอร์ตอินพุต การสร้างสัญญาณลอจิกจากคีย์สวิตซ์อย่างง่าย แสดงดังรูปที่

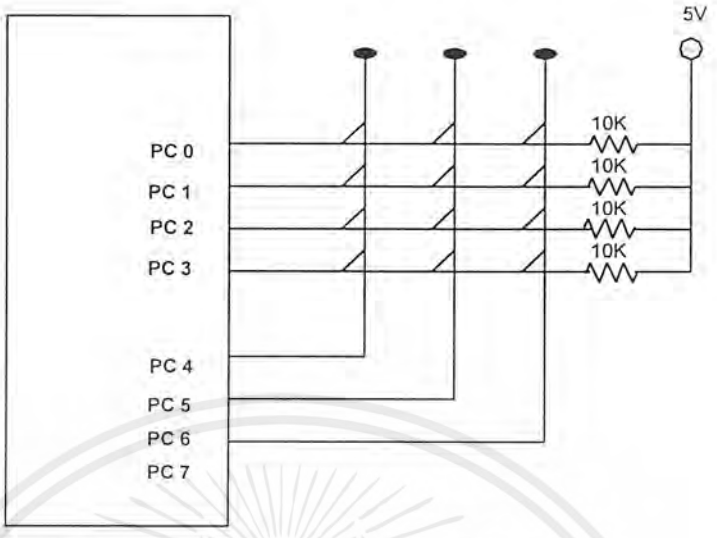


รูปที่ 3.7 การสร้างลอจิกคีย์สวิตซ์อย่างง่าย

จากรูปถ้าหากมีการกดคีย์ค่าเอาต์พุตจะเป็นลอจิก "0" ถ้าหากไม่มีการกดคีย์เอาต์พุตจะเป็นลอจิก "1" การต่อคีย์สวิตซ์ดังกล่าวจะต้องใช้จำนวนบิตของพอร์ตอินพุตเท่ากับจำนวนคีย์สวิตซ์ แต่ถ้าต้องการต่อคีย์สวิตซ์จำนวนมากเราสามารถนำวงจรเข้ารหัสมาช่วยได้ วิธีนี้จะทำให้ลดจำนวนพอร์ตอินพุตของระบบไมโครคอนโทรลเลอร์ได้ถ้าหารต้องการพอร์ตอินพุตจำนวนมาก

การต่อคีย์สวิตซ์อีกวิธีหนึ่งที่เป็นที่นิยมนำมาใช้เป็นอุปกรณ์อินพุตสำหรับระบบไมโครคอนโทรลเลอร์คือการจัดวางแบบเมตริกซ์ และการต่อคีย์สวิตซ์ ในวงจรจึงเลือกใช้วิธีนี้ โดยคีย์สวิตซ์แต่ละตัวที่ประกอบอยู่ในเมตริกซ์นั้น ซึ่งสามารถอ้างตำแหน่งของคีย์ได้ในแนวแกน X-Y หรือค่าแนวแถว-หลัก การต่อคีย์สวิตซ์แบบนี้จะทำให้ต่อคีย์ได้จำนวนมาก โดยใช้สัญญาณควบคุมพอร์ตน้อยกว่า และไม่ต้องใช้วงจรเข้ารหัสอีกด้วยแต่ต้องใช้เทคนิคการเขียนโปรแกรมเข้ามาช่วยในการอ่านค่าสวิตซ์แทน การต่อคีย์แบบเมตริกซ์แสดงดังรูปที่ 3.8

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.8 แสดงการต่อคีย์แบบเมตริกซ์

การต่อคีย์บอร์ดในแนวแกน X จะมีตัวต้านทานต่อกับไฟเลี้ยง +5V ในลักษณะการต่อ พูลอัพ สัญญาณ และต่อเข้ากับพอร์ต พีซีของ 8255 ทำให้พอร์ตพีซี 4 บิตต่ำมีค่าเป็นลอจิก "1" อยู่ตลอดเวลา ในการตรวจสอบการตรวจสอบการกดคีย์บอร์ด นั้นไมโครคอนโทรลเลอร์จะส่งค่าลอจิก "0" ออกมาในแนวแกน Y ทีละเส้น จากนั้นจะอ่านค่าเข้าไปในแนวแกน X ถ้าคีย์ใดถูกกด จะทำให้บิตในแนวแกน X ที่คีย์นั้นต่ออยู่มีค่าลอจิกเป็น "0" การเขียนโปรแกรมอ่านค่าจะใช้เทคนิคการเขียนโปรแกรมแบบสแกนคีย์ โดยจะส่งค่าออกมาในแนวแกน Y ให้เป็นลอจิก "0" ทีละหลักอ่านค่าเข้าไปในแนวแกน X ต่อมาจะให้ค่าในแนวแกน Y หลักต่อไปเป็น "0" และอ่านค่าเข้าไปในแนวแกน X อีก โดยทำอย่างเร็ว ถ้าหากคีย์ใดถูกกดเราจะทราบค่าได้โดยการพิจารณาจากความสัมพันธ์ระหว่างค่าในแนวแกน X และแนวแกน Y ดังรูปที่ 3.9

ค่าสแกน->	110	101	011
1110	1	2	3
1101	4	5	6
1011	7	8	9
0111	*	0	#

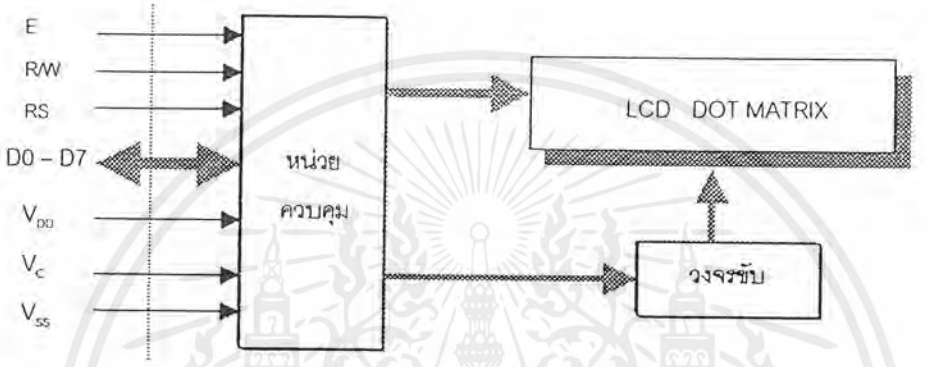
ค่าที่อ่านได้(PC)

รูปที่ 3.9 แสดงการสแกนค่าคีย์บอร์ด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.7.3 การทำงานของจอแสดงผล LCD

การแสดงผลการควบคุมการทำงานนี้จะแสดงผ่านทาง จอแสดงผล LCD ซึ่ง LCD นี้เป็นอุปกรณ์ที่นิยมนำมาใช้ เนื่องจากเป็นหน่วยแสดงผลที่กินพลังงานต่ำ เป็นอุปกรณ์ที่มีขนาดเล็ก และมีความละเอียดสูง การที่จะทำให้ LCD สว่างเป็นรูปร่างหรือตัวอักษรต่างๆ นั้นจะต้องใช้วงจรขับที่มีความซับซ้อนมาก แต่ในปัจจุบันจึงมีการสร้างเป็น LCD สำเร็จรูปเพื่อให้สะดวกต่อการใช้งาน โดยโครงสร้างทั่วไปของ LCD โมดูลแสดง ดังรูปที่ 3.10



รูปที่ 3.10 แสดงโครงสร้างของ LCD โมดูล

การใช้งาน LCD จะทำการต่อ LCD กับพอร์ต 0 ซึ่งพอร์ตนี้จะใช้งานร่วมกับ 8255 และขาที่คอนโทรล LCD คือขา RS(P2.4) ขา R/W(P2.5) ขา E(P2.6) เพื่อทำให้การควบคุมการทำงานของ LCD เป็นไปได้ง่ายยิ่งขึ้นจะต้องทำความเข้าใจคำสั่งต่างๆของมันด้วย โดยการเริ่มต้นใช้งานจะต้องส่งรหัสควบคุมให้กับรีจิสเตอร์คำสั่ง และส่งข้อมูลให้กับรีจิสเตอร์ข้อมูล ก่อนเริ่มต้นการใช้งาน LCD โดยมีลำดับการทำงานของข้อมูล ดังรูปที่ 3.11

รูปที่ 3.11 แสดงลำดับการควบคุมการทำงาน LCD โมดูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.7.4 การรับค่าจากคอมพิวเตอร์

การรับค่าจากคอมพิวเตอร์จะทำการส่งข้อมูลผ่านพอร์ตขนาน หรือเรียกอีกอย่างหนึ่งว่าพอร์ตพรินเตอร์ ข้อมูลจากคอมพิวเตอร์ที่ส่งมาจะเป็นค่าอุณหภูมิ และเวลาของอาหารในแต่ละประเภทที่เราสามารถโหลดมาจากอินเตอร์เน็ต หรือจะตั้งค่าจากโปรแกรมเดลไฟล์ที่เขียนขึ้นมาก็ได้ โดยเมื่อข้อมูลถูกส่งมาเรียบร้อยแล้ว ไมโครคอนโทรลเลอร์จะเป็นตัวอุปกรณ์ที่ทำหน้าที่รับข้อมูลนั้นมาเก็บไว้ในหน่วยความจำข้อมูลชั่วคราว และข้อมูลนั้นจะถูกส่งไปควบคุมการทำงานของไตรแอคทันที ทำให้เราสามารถตัดการเชื่อมต่อกับคอมพิวเตอร์ได้เพราะว่าข้อมูลได้ถูกส่งออกมาเรียบร้อยแล้ว

ข้อมูลที่ถูกส่งมาจะส่งมาให้ พอร์ตA ของ 8255 ซึ่งจะใช้เป็นพอร์ตอินพุตรับค่าจากคอมพิวเตอร์อย่างเดียว โครงสร้างการเชื่อมต่อกับคอมพิวเตอร์แสดงได้ดังรูปที่ 3.6 โดยสัญญาณที่ทำให้ไมโครคอนโทรลเลอร์ รู้ว่าตอนนี้กำลังเชื่อมต่อกับคอมพิวเตอร์อยู่คือสัญญาณอินเตอร์รัปต์ ซึ่งเป็นสัญญาณที่ติดต่อ และเลือกใช้สัญญาณอินเตอร์รัปต์ที่ 0 เป็นตัวติดต่อกับคอมพิวเตอร์

3.7.5 การส่งค่าไปควบคุมอุณหภูมิ และ การรับค่าอุณหภูมิที่ฟีดแบกกลับมา

การส่งค่าไปควบคุมอุณหภูมิ เมื่อรับค่าอุณหภูมิและ ส่งค่าไปควบคุมอุณหภูมินั้นให้มีค่าคงที่มากที่สุด ต้องมีการฟีดแบกค่าอุณหภูมิกลับมา โดยการส่งค่าไปควบคุมอุณหภูมิจะส่งผ่านพอร์ตB ของ 8255 และการรับค่าอุณหภูมิที่ฟีดแบกกลับมา ผ่านทาง พอร์ต1 ซึ่งรับค่ามาจากวงจร A to D converter

ในส่วนการเปรียบเทียบค่าที่ต้องการ กับค่าที่ฟีดแบกกลับมา จะใช้ไมโครคอนโทรลเลอร์เขียนโปรแกรมเปรียบเทียบในส่วนนี้ โดยถ้าส่วนที่ฟีดแบกกลับมา ที่มีค่าเป็นเลขฐานสิบหกน้อยกว่าค่าที่ต้องการที่ได้จากการตั้งค่า แสดงว่าอุณหภูมิที่ได้ตอนนี้มีค่าน้อยกว่าอุณหภูมิที่ต้องการ ทำให้ต้องเพิ่มค่าบิตที่ส่งไปควบคุมอุณหภูมิให้มากขึ้น โดยจะเพิ่มค่าครั้งละ 10 แต่ถ้าค่าที่ ฟีดแบกกลับมามีค่ามากกว่าค่าที่ต้องการ แสดงว่าอุณหภูมิที่ได้ตอนนี้มีค่าสูงกว่าอุณหภูมิที่ต้องการ ทำให้ต้องลดค่าบิตที่ส่งไปควบคุมอุณหภูมิให้ลดลง โดยจะลดค่าครั้งละ 10 และ การตรวจสอบค่าอุณหภูมิที่ฟีดแบกกลับมา จะตรวจสอบทุกๆ 10 วินาที ทำให้ค่าอุณหภูมิที่ควบคุมได้มีค่าใกล้เคียงกับอุณหภูมิที่ต้องการ

3.8 การดิดตั้ง และ เซตโปรแกรมสำรับ ใช้เป็นเซิฟเวอร์

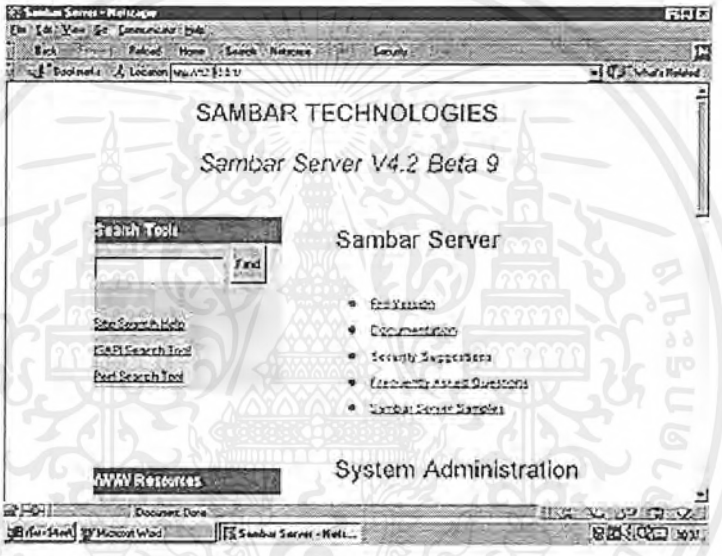
การเขียนโปรแกรมควบคุมการทำงาน บนอินเตอร์เน็ทซึ่งต้องใช้หลักการเขียนโปรแกรมหลายภาษาเข้ามาช่วยในการควบคุมการทำงาน จะต้องทำการตั้ง เซิฟเวอร์ขึ้นมา เพื่อใช้เป็นฐานข้อมูล เขียนเว็บเพจเพื่อแสดงหน้าจอบนอินเตอร์เน็ต และ โปรแกรมเดลไฟล์ ซึ่งสามารถดาวน์โหลดโปรแกรมได้จากฐานข้อมูลบนอินเตอร์เน็ต มาใช้ควบคุมฮาร์ดแวร์ได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ในการทำงาน จะสมมติว่าบริษัทที่ผลิตเตาอบ จะต้องสร้างเว็บเพจขึ้นมาเว็บหนึ่ง เว็บเพจนั้นจะประกอบด้วยฐานข้อมูล ที่ใช้สำหรับอ้างอิงเกี่ยวกับอาหารที่ทำด้วยเตาอบ รายการ วิธีการปรุง และอุณหภูมิที่เหมาะสมสำหรับอาหารแต่ละประเภท พร้อมทั้งการทำงานบนเว็บเพจ ผู้ที่จะใช้งานได้จะต้องเป็นผู้ที่ซื้อเตาอบไปเท่านั้น โดยจะมี หมายเลขเครื่องให้สำหรับลูกค้า

ดังนั้น จากการสมมติข้างต้น เราจะต้องทำการสร้างดังนี้

โปรแกรมที่ทำงานเป็นเซิร์ฟเวอร์นี้จะใช้โปรแกรม Sambar42 ติดตั้งบนคอมพิวเตอร์ของผู้ผลิตเตาอบ ซึ่งคอมพิวเตอร์ตัวนี้จะเป็นฐานข้อมูลของระบบได้รูปร่างหน้าตาของโปรแกรม Sambar42 ได้แสดงดังรูป



รูปที่ 3.12 แสดงลักษณะของโปรแกรม Sambar42

เมื่อติดตั้งเซิร์ฟเวอร์ เรียบร้อยแล้ว ต้องทำการเซต Document Aliases และ CGI Aliases เพื่อใช้เป็นไดเรกทอรีเทียม ทำให้ผู้ใช้ไม่สามารถรู้ถึงไดเรกทอรีที่เก็บข้อมูลจริงๆได้ เป็นวิธีการป้องกันการขโมยข้อมูลได้ และเมื่อทำการเซต Aliasaes แล้วยังสามารถกำหนดผู้ใช้ ได้ ว่าให้คนใด บุคคลใดที่สามารถเข้าไปยังเว็บเพจหน้าต่างๆ ได้ วิธีการเซตทำดังนี้

วิธีการเซต Aliases1) เข้าไปที่ข้อความส่วนเชื่อมโยง System Administration ที่อยู่ในรูปที่ 4.1 โดยใส่ user name ว่า admin ไม่ต้องใส่ password จะได้หน้าจอดังรูปที่ 4.2

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 3.13 แสดงลักษณะของ System Administration

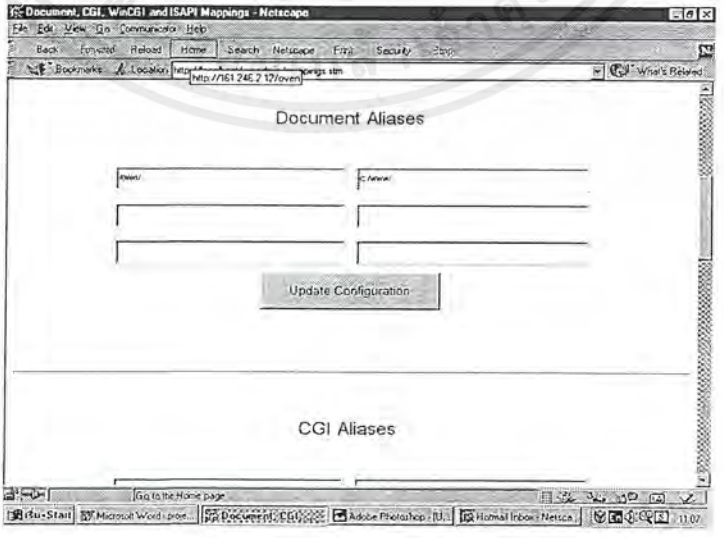
2) เลือก Document, CGI , Win CGI & ISAPI Mappings

3) กำหนดค่า และรีสตาร์เซิร์ฟเวอร์ โดยป้อนค่าเป็นดังนี้

Document Aliases จาก /oven/ เป็น /c:/www/

CGI Aliases จาก /CGI/ เป็น /c:/www/cgi/

จะได้ดังรูปที่ 4.3 โดยผลที่ได้คือเมื่อเรียกไปที่เว็บนี้ ที่ตามด้วย /oven/ จะทำการดึงข้อมูลจากเพจมาใช้งาน เป็น HTML จะไปดึงข้อมูลจากไดเรกทอรี c:\www\ ที่เครื่องเซิร์ฟเวอร์ และเมื่อมีการใช้ CGI ประมวลผล จะไปดึงโปรแกรม CGI จากไดเรกทอรี c:\www\cgi\ ที่เครื่องเซิร์ฟเวอร์

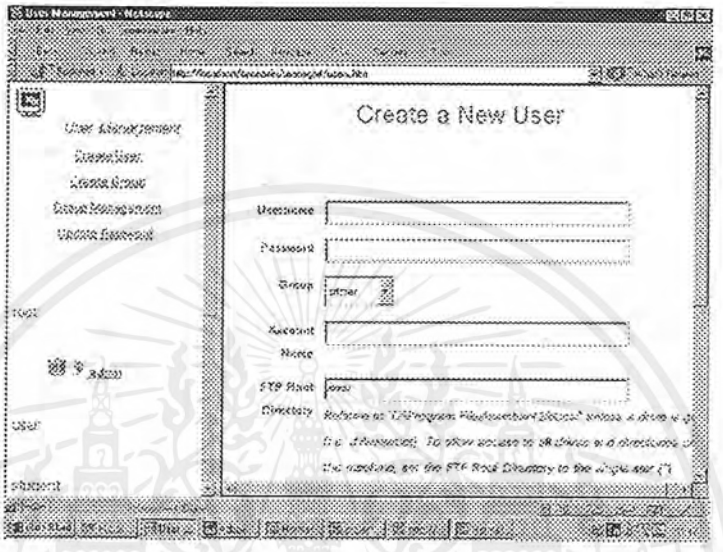


รูปที่ 3.14 แสดงลักษณะการเซต Aliases

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

วิธีการเซต Add user และ โฮมเพจที่ user นั้นๆสามารถใช้งานได้

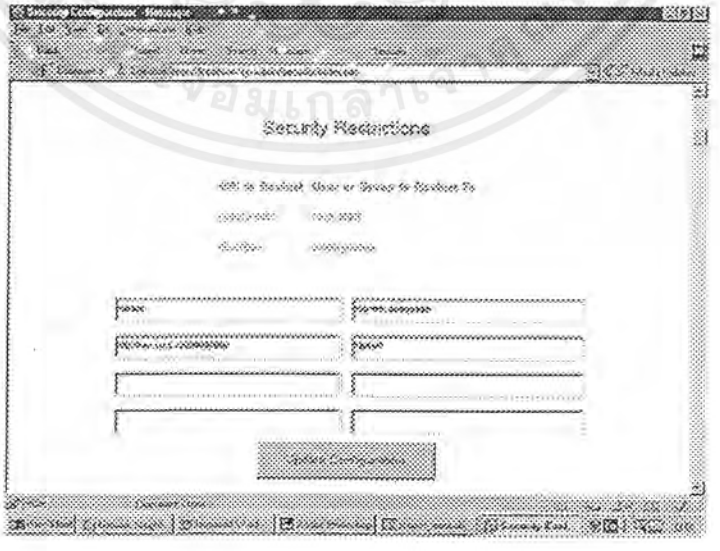
1) จากหน้าจอรูปที่ 3.2 ให้เลือก User Management จะได้หน้าจอดังรูปที่ 4.4 การ Add User นั้นสามารถกำหนด กรุ๊ปให้ user นั้นๆได้เพื่อความสะดวกในการจำกัดกลุ่ม



รูปที่ 3.15 แสดงการ Add User

2) หลังจากนั้น ให้กำหนด เพจที่กลุ่มผู้ใช้สามารถเข้าถึงได้โดยเซตค่าใน Security Configuration เลือกได้ในรูปที่ 4.5

3) เซตค่าใน security restriction โดยให้ส่วนที่สามารถเข้าไปดูได้ เป็นส่วนที่อยู่ใน www/menu/ และกลุ่มที่เข้าได้ให้เป็นกลุ่มของ student



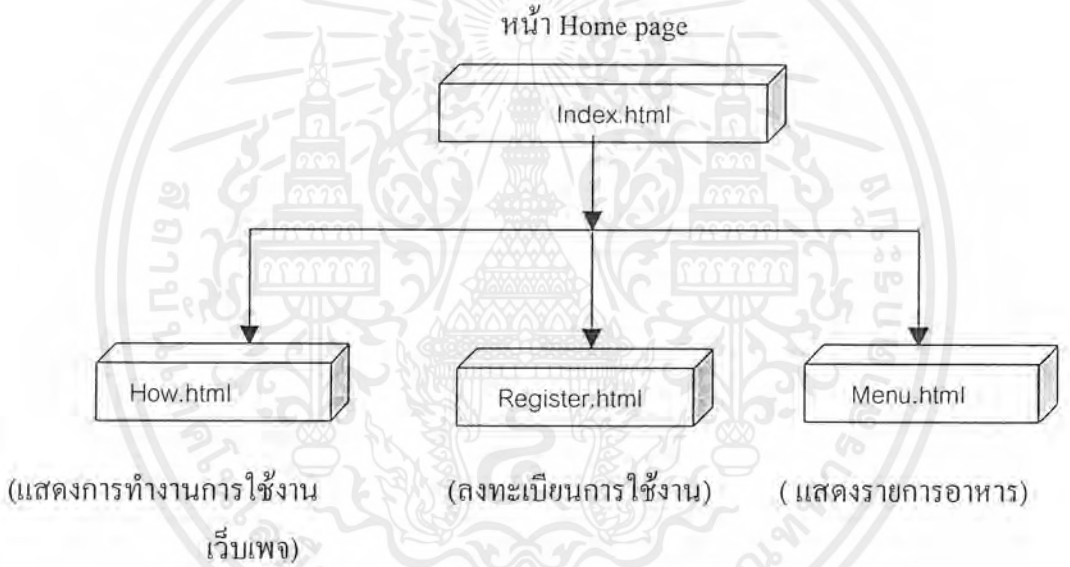
รูปที่ 3.16 แสดงลักษณะการเซต Security Restriction

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ขั้นตอนดังกล่าวนี้เป็นการเซตค่าต่างๆ ของโปรแกรมเซิฟเวอร์ สำหรับโปรเจต์นี้ เพื่อให้ใช้งานได้อย่างเหมาะสม และเป็นการสมมติการทำงานของเซิฟเวอร์ขึ้นมา ดังนั้น ในการทำงานจริง สามารถ ดัดแปลงให้ใช้กับสภาพงานนั้นๆ ให้เหมาะสมได้

3.9 ส่วนของโปรแกรม HTML

โปรแกรม HTML นี้จะเป็นส่วนแสดงผลบนหน้าจอ ผู้ใช้ หรือที่เรียกอีกอย่างหนึ่งว่าไคลเอนต์ การทำงานของโปรแกรมจะเป็นส่วนที่ทำหน้าที่ ตอบโต้ ระหว่างไคลเอนต์และ เซิฟเวอร์ ซึ่งโปรแกรม HTML ที่เขียนขึ้นมาจะมีการเชื่อมโยงระหว่างเพจ เชื่อมโยงกันโดยจะแสดงแผนภาพการเชื่อมโยงได้ดังรูปที่ 3.17



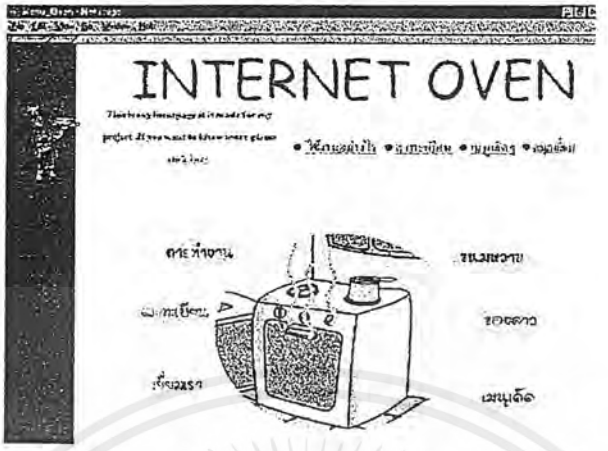
รูปที่ 3.17 แสดงการเชื่อมโยงของส่วนต่างๆ

รายละเอียดของไฟล์HTML ดังต่อไปนี้

1) index.html

เป็นเพจที่แสดงหน้าแรกของเว็บเพจนี้ ซึ่งไคลเอนต์จะต้องลงทะเบียนเพื่อขอรหัสผ่านก่อนเข้าใช้งาน และสามารถดูรายละเอียดการใช้งานได้ โดยเพจนี้จะแสดงได้ดัง รูปที่ 3.18

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.18 แสดงลักษณะเพจของ index.html

2) regiter.html

เป็นเพจที่ทำหน้าที่ ให้ผู้ใช้ลงทะเบียน โดยนำหมายเลขเครื่องที่ให้ไป ใส่ลงในฟอร์มที่กำหนด และทำการขอพาสเวิร์ดได้ แสดงได้ดังรูปที่ 3.19 ซึ่งเซิฟเวอร์จะทำการรับข้อมูลจากฟอร์มของHTML ด้วยเมธอด GETผ่านไฟล์ที่ชื่อ register.pl ซึ่งจะกล่าวรายละเอียดถึงในหัวข้อถัดไป

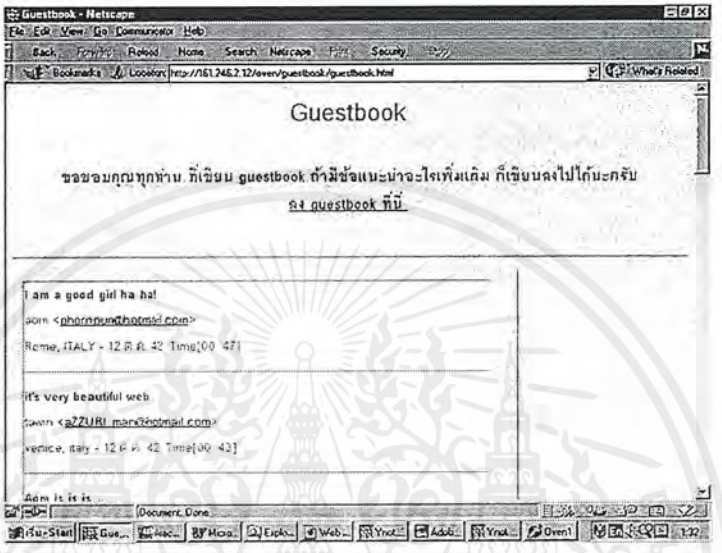
รูปที่ 3.19 แสดงลักษณะเพจของ register.html

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3) guestbook.html

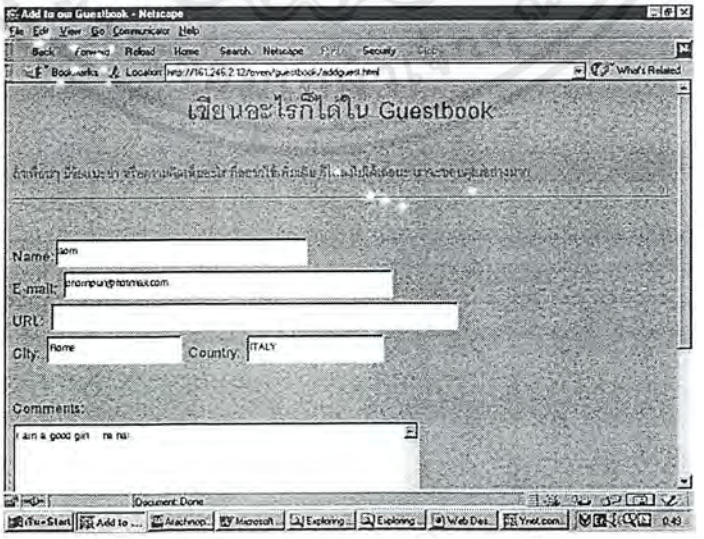
เพจนี้แสดงการทำงาน ของการรับข้อมูล จากเพจ addguest.html โดยใช้ โครงสร้างของเพิร์ล เพื่อเปิดไฟล์และทำการเขียนข้อมูล ในเพจนี้ เมื่อทำการ reload อีกครั้งหนึ่ง ซึ่งได้แสดงดังรูปที่ 3.20

รูปที่ 3.20 แสดงลักษณะเพจของ guestbook.html



4) addguest.html

โครงสร้างของเพจนี้แสดงการทำงาน ของการรับข้อมูลเป็น ข้อความที่มีความยาวมากได้ จึงใช้การส่งข้อมูลแบบ POST ซึ่งได้กล่าวตามทฤษฎีในบทที่ 2 และมีการส่งผ่านตัวแปรที่ชื่อ realname, username ,url ,city ,country และ comments ลักษณะเพจได้แสดงดังรูป



รูปที่ 3.21 แสดงลักษณะเพจของ addguest.html

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5) menu.html

โฮมเพจนี้ได้ใช้เฟรมแบ่ง ออกเป็น 3 ส่วน ซึ่งประกอบด้วยหน้าจอดังรูปที่4.10แบ่งแบ่งรายการอาหารออกเป็นประเภทต่างๆ ดังรูป

รูปที่ 3.22 Menu.html

3.10 ส่วนของโปรแกรมภาษาเพิร์ล

ในส่วนโปรแกรมที่เป็นภาษาเพิร์ล จะเป็นตัวที่ทำหน้าที่รับค่าตัวแปรจากฟอร์มของ HTML ที่มีการส่งข้อมูลด้วยวิธี GET หรือ POST ได้ อีกทั้งยังเป็นตัวตรวจสอบข้อมูลที่เข้ามาได้ ว่าเป็นข้อมูลที่ต้องการหรือ ไม่ โดยตัวแปรที่รับมา และนำมาตรวจสอบจะรับมาจากค่า name ใน ฟอร์มของ HTML ไฟล์ของเพิร์ลที่ใช้ในการรับการร้องขอ ของข้อมูล จะประกอบด้วยไฟล์ต่างๆ ดังนี้

1) guestbook.pl

เป็นไฟล์ cgi ที่ทำการรับตัวแปรที่ ผู้ใช้ได้ป้อนข้อมูลลงในฟอร์ม โดยมีการรับตัวแปรแวดล้อมทีชื่อ $ENV{'CONTENT_LENGTH'} แล้วนำมาแยกตัวแปร โดยใช้ ชื่อเหล่านี้เป็นตัวรับค่า คือ

$realname	รับค่าจากฟอร์ม name
$username	รับค่าจากฟอร์ม e-mail
$url	รับค่าจากฟอร์ม url
$city	รับค่าจากฟอร์ม city
$country	รับค่าจากฟอร์ม country

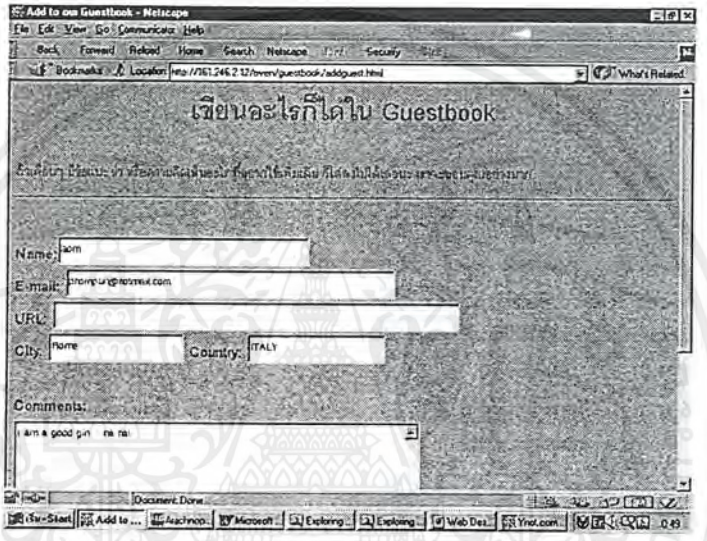
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$comment รับค่าจากฟอร์ม comment

แล้ว นำตัวแปรดังกล่าว ไปเก็บไว้ในไฟล์ guestbook.html ทำให้ข้อความที่ได้ ไปปรากฏอยู่ใน เพจ ของ guestbook.html โดยอัตโนมัติ

แต่การรับข้อมูลโปรแกรมจะมีการตรวจสอบว่า ข้อมูลที่รับมา มีช่องใดที่ไม่ได้มีการใส่ข้อมูลลงไป โปรแกรมจะทำการสร้างเพจที่มีความผิดพลาดของการใส่ข้อมูลโดยจะแสดงดังนี้

ถ้าผู้ใช้ไม่ได้กรอกชื่อ โปรแกรมจะทำการตรวจสอบข้อความที่ว่าง แล้วจะทำการรันส่วนของ HTML ที่อยู่ใน guestbook.pl แสดงข้อความที่ผิดพลาดดังรูปที่ 3.23



รูปที่ 3.23 แสดงลักษณะความผิดพลาดที่เกิดจากข้อมูลว่างเปล่า

2) register.pl

เป็นโปรแกรมที่ใช้ตรวจสอบข้อมูล ของระบบการใช้งานบนเว็บนี้โดยผู้ที่เข้ามาใช้งานต้องลงทะเบียนกรอกหมายเลขเครื่องที่ถูกต้องเพื่อขอรหัสผ่าน จากเซิฟเวอร์ และสามารถใช้งานเว็บเพจนี้ได้การออกแบบการทำงานในส่วนนี้ จะมีการรับค่าตัวแปรจากผู้ใช้ 5 ตัว

$fname :

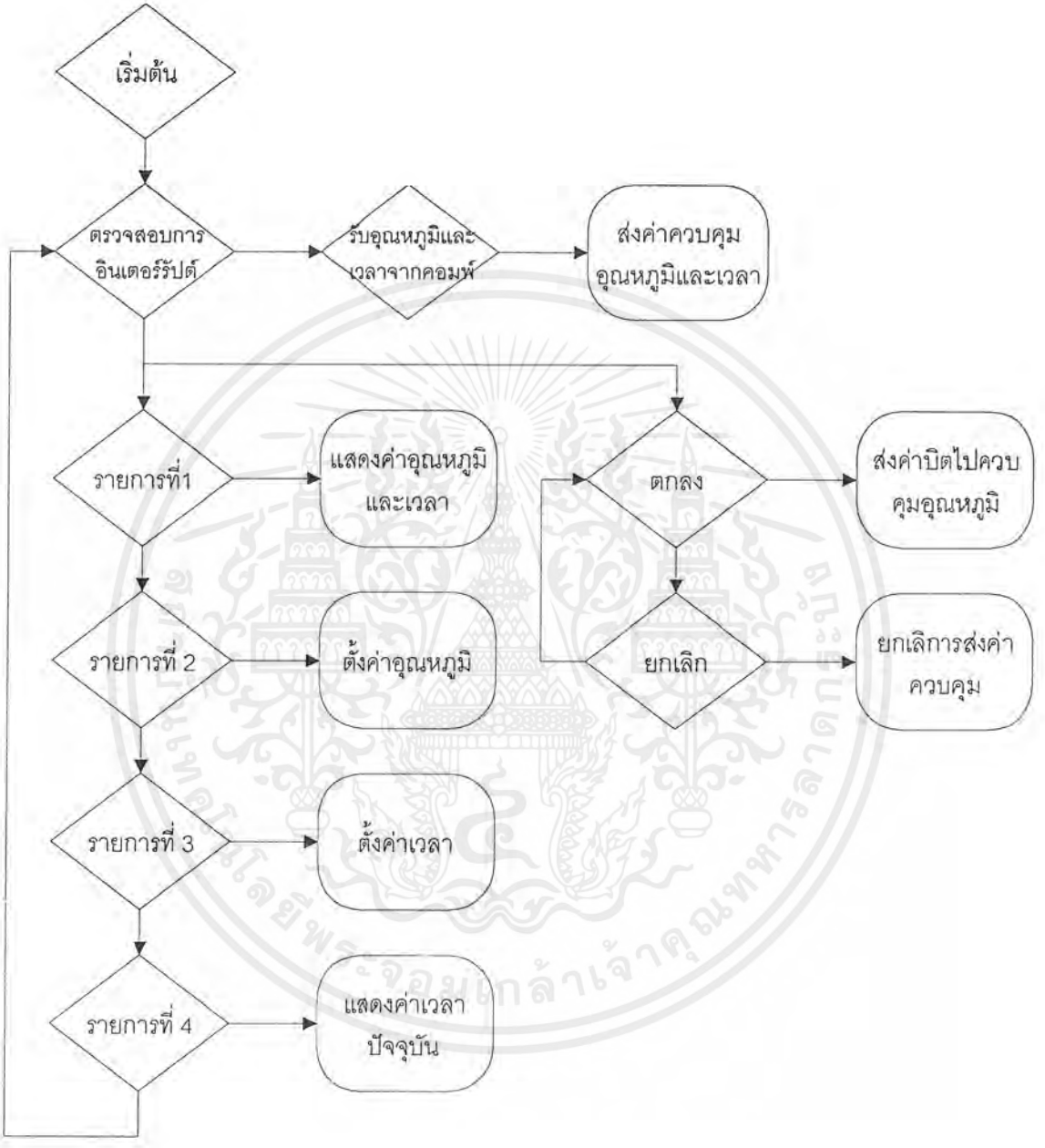
$surname:

$number:

$code :

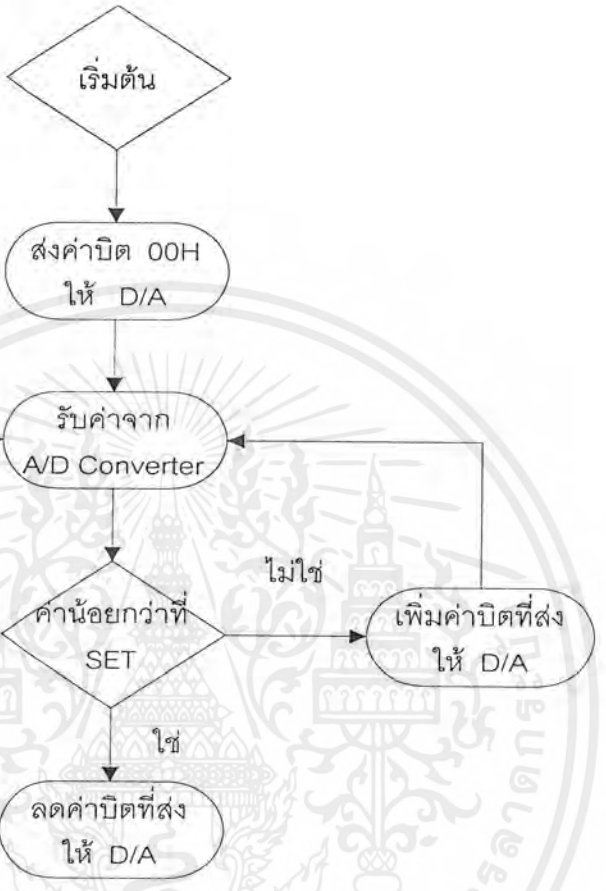
$recode :

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.24 แสดงการควบคุมการทำงานของ MCS-51

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.25 แสดงการควบคุมอุณหภูมิ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 4
การทดลอง

4.1 ผลการทดลองต่อ D/A CONVERTER

การทดลองส่วนนี้แสดงค่าที่ส่งไปให้กับ D/A ซึ่งต่ออยู่กับ คอนโทรลเลอร์เอาท์พุทที่ออกมาส่งต่อไปให้กับ ส่วนการคอนโทรลกำลังงาน ค่าเปลี่ยนจากดิจิตอลเป็นอนาลอก

BIT INPUT	Vout (volt)
00000000	-0.029
00000101	0.190
00001111	0.628
00011001	1.066
00100011	1.505
00101101	1.943
00110111	2.386
01000001	2.85
01001011	3.29
01010101	3.73
01011111	4.17
01101001	4.61
01110011	5.04
01111101	5.48
10000111	5.92
10010001	6.36
10011011	6.8
10100101	7.24
10101111	7.68
10111001	8.12
11000011	8.57
11001101	9.01
11010111	9.45

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

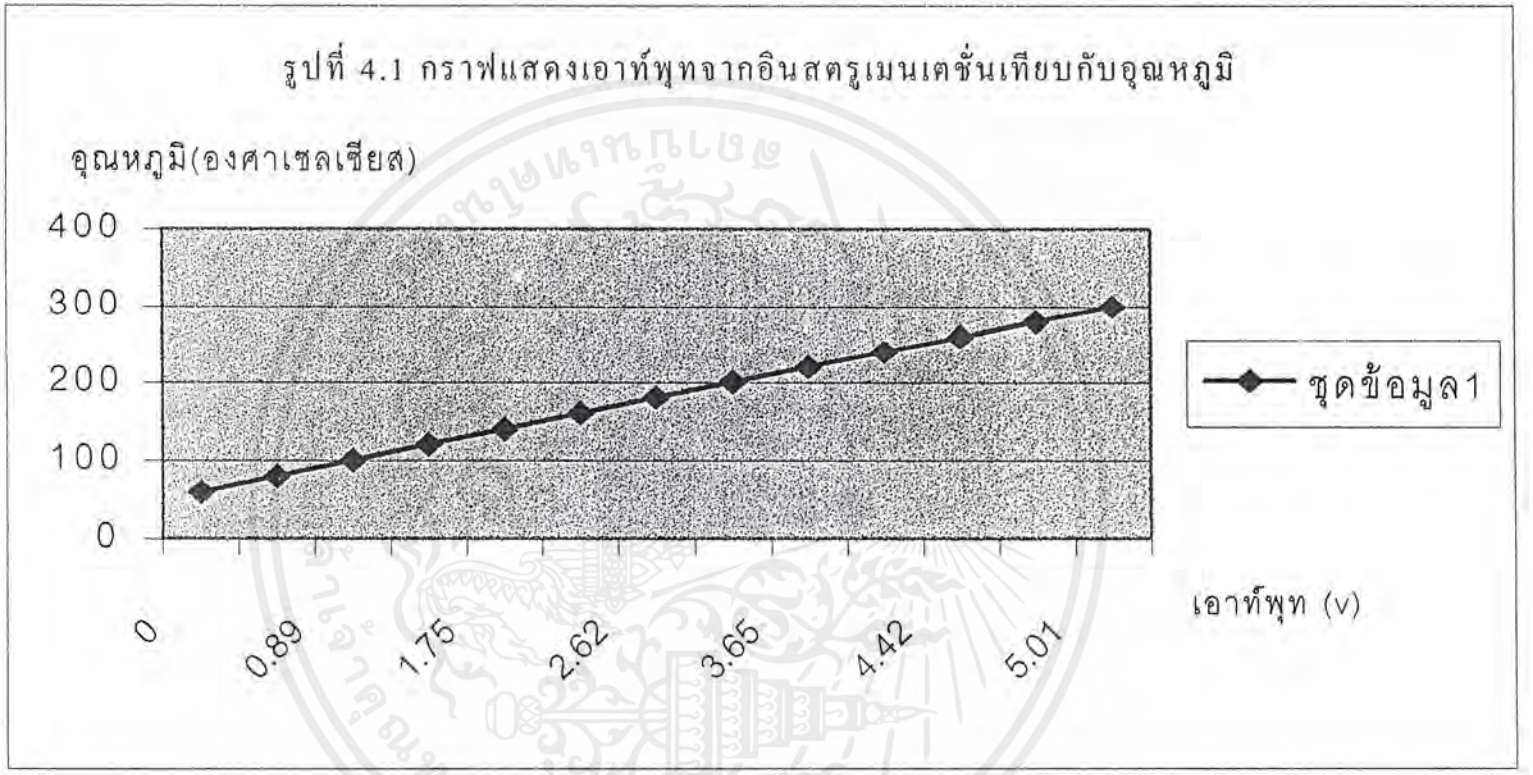
11100001	9.89
11101011	10.33
11110101	10.76
11111111	11.23

4 .2ผลการทดลอง A/D CONVERTER

Vin (Volt)	BIT (OUTPUT)
0.022	00000001
0.040	00000010
0.058	00000011
0.110	00000110
0.571	00011101
1.012	00110011
1.99	01100100
3.0	10011000
4.0	11001000
4.2	11010100
4.6	11101000
4.93	11111000
5.1	11111111

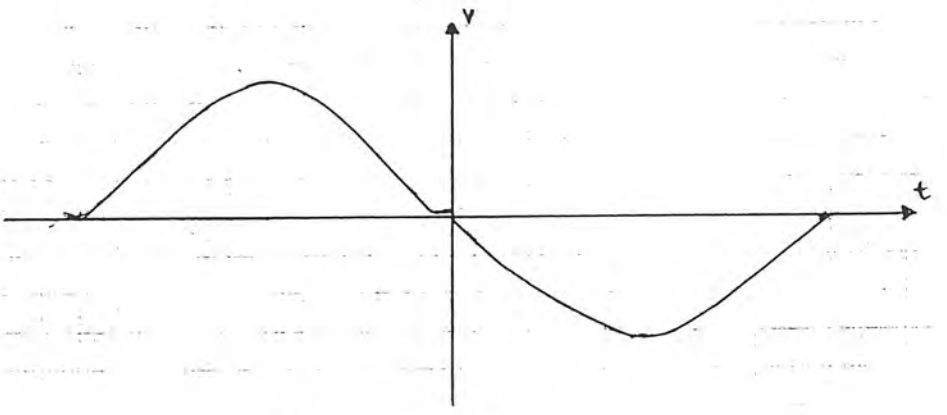
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.3 ผลการทดลองต่อ Instrument Amp เทียบกับอุณหภูมิ

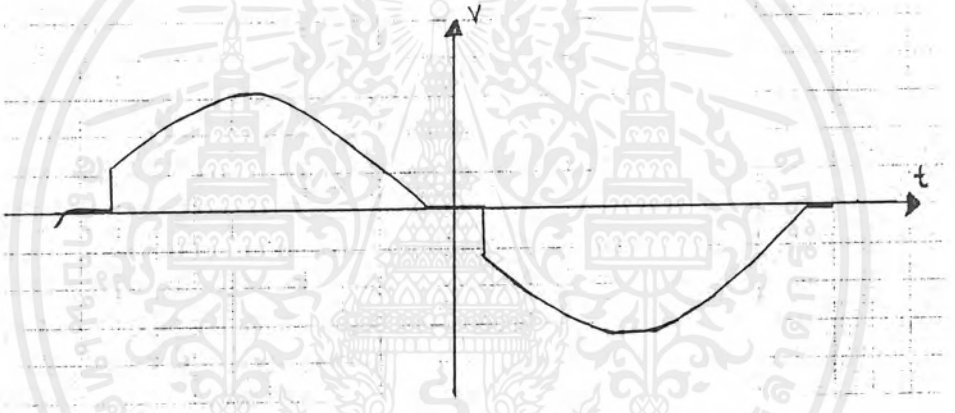


รูปที่ 4.1 กราฟแสดงเอาท์พุทจากอินสตรูเมนเตชั่นเทียบกับอุณหภูมิ

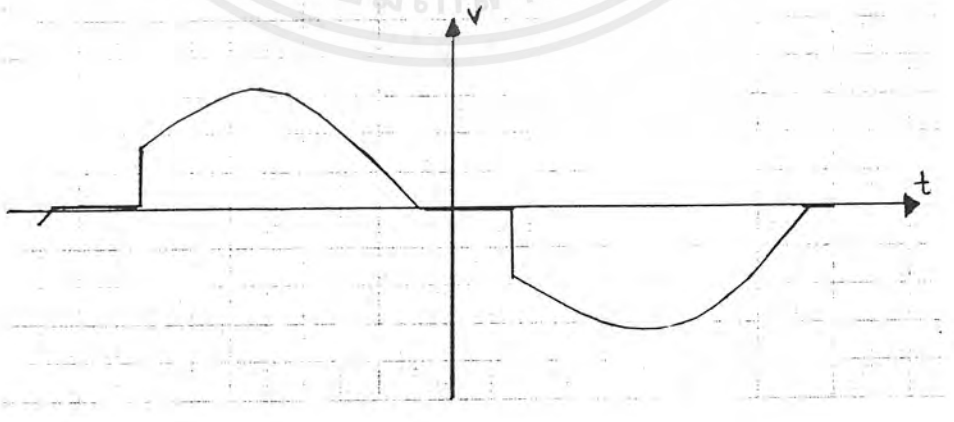
4.4 การทดลอง control heater



รูปที่4.2 บิทอินพุท 0000 0000 (0)

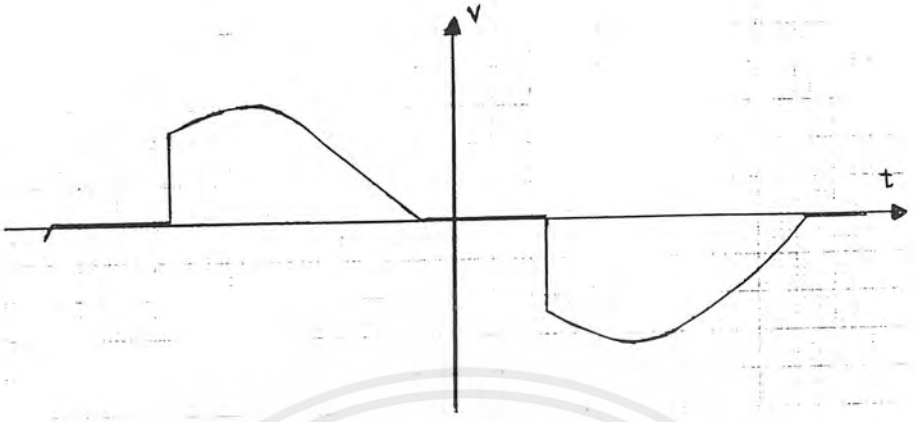


รูปที่ 4.3 บิทอินพุท 0001 1001 (25)

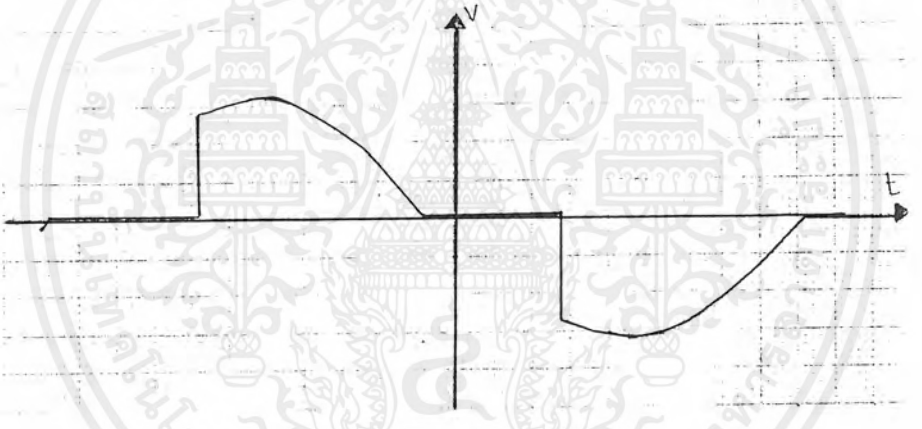


รูปที่ 4.4 บิทอินพุท 0001 1001 (50)

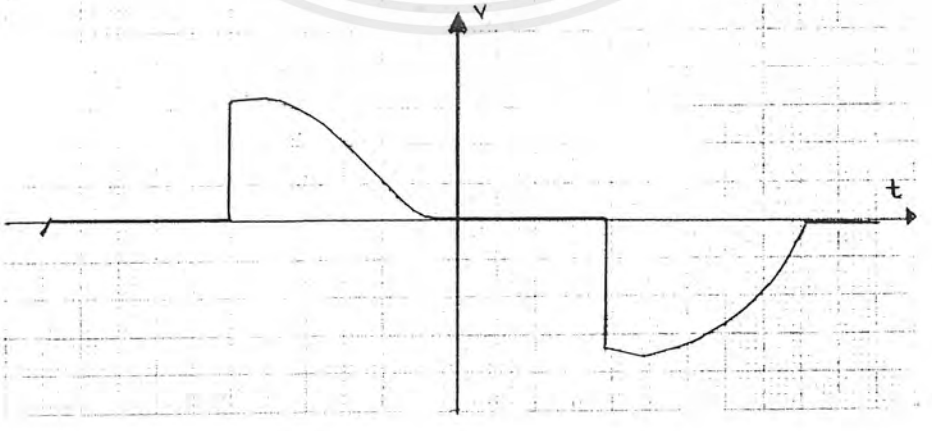
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่4.5 บิทอินพุท 0100 1011 (75)

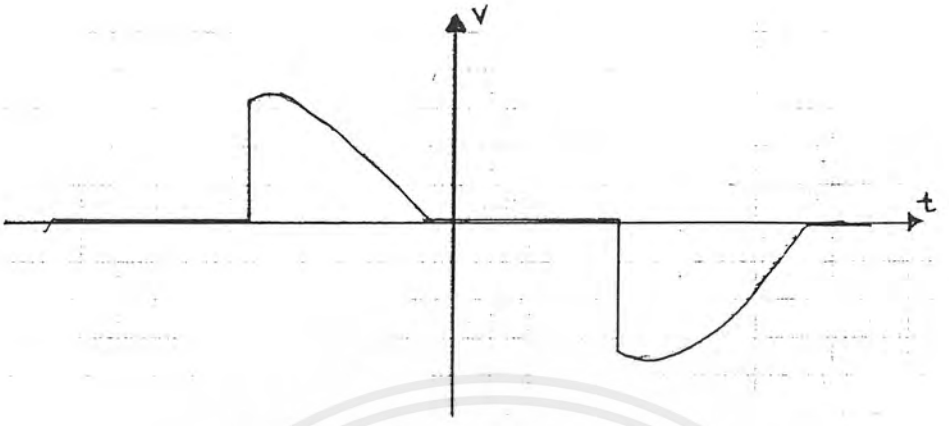


รูปที่ 4.6 บิทอินพุท 0110 0100 (100)

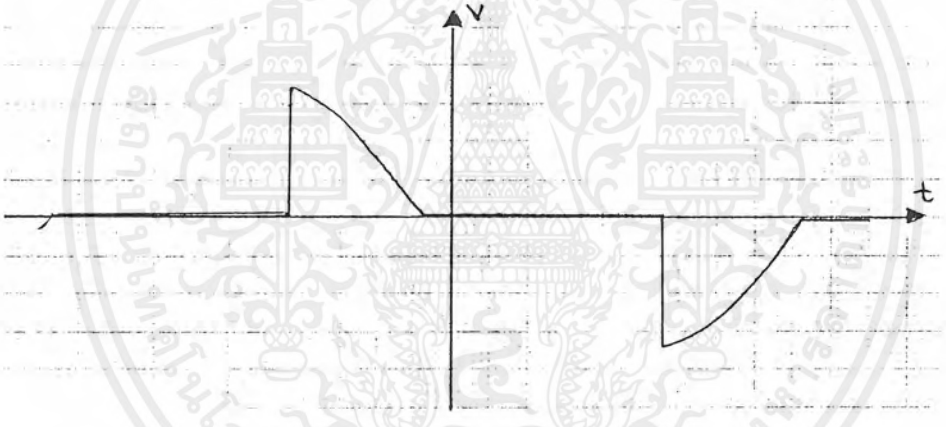


รูปที่4.7 บิทอินพุท 0111 1101 (125)

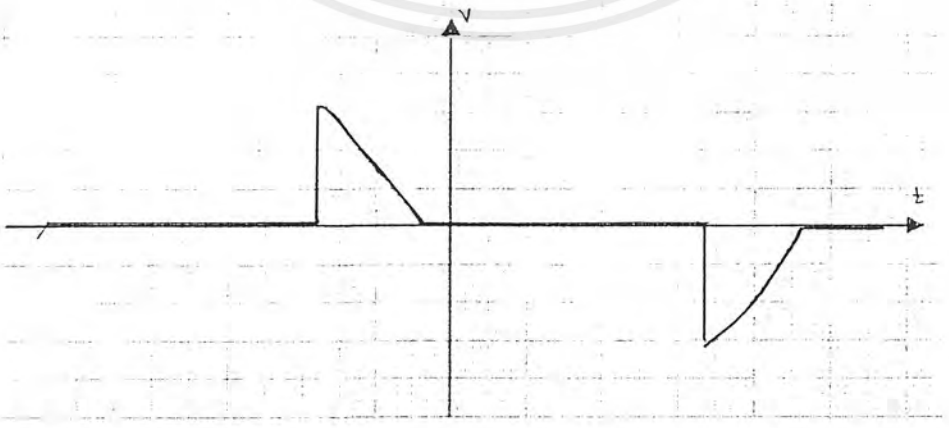
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่4.8 บิทอินพุท 1001 0110 (150)

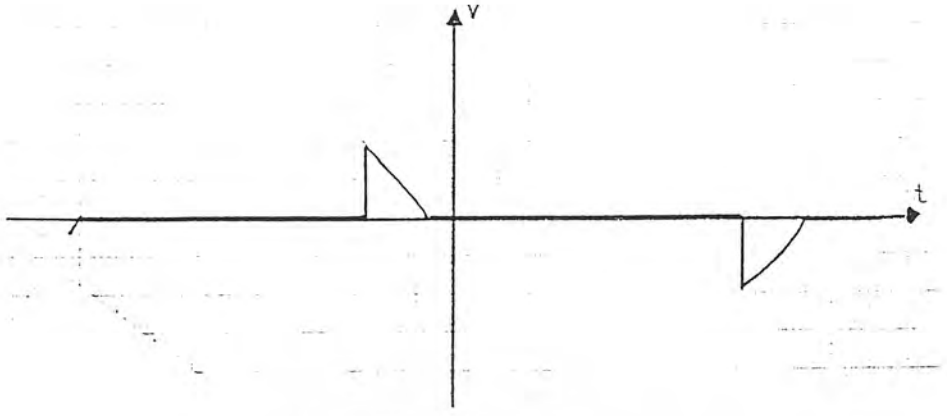


รูปที่4.9 บิทอินพุท 1010 1111 (175)

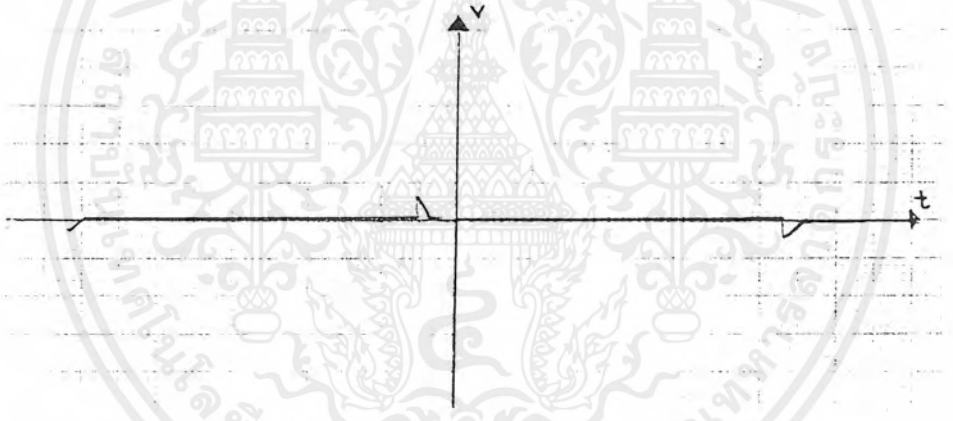


รูปที่4.10 บิทอินพุท 1100 1000 (200)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่4.11 บิทอินพุท 1110 0001 (225)



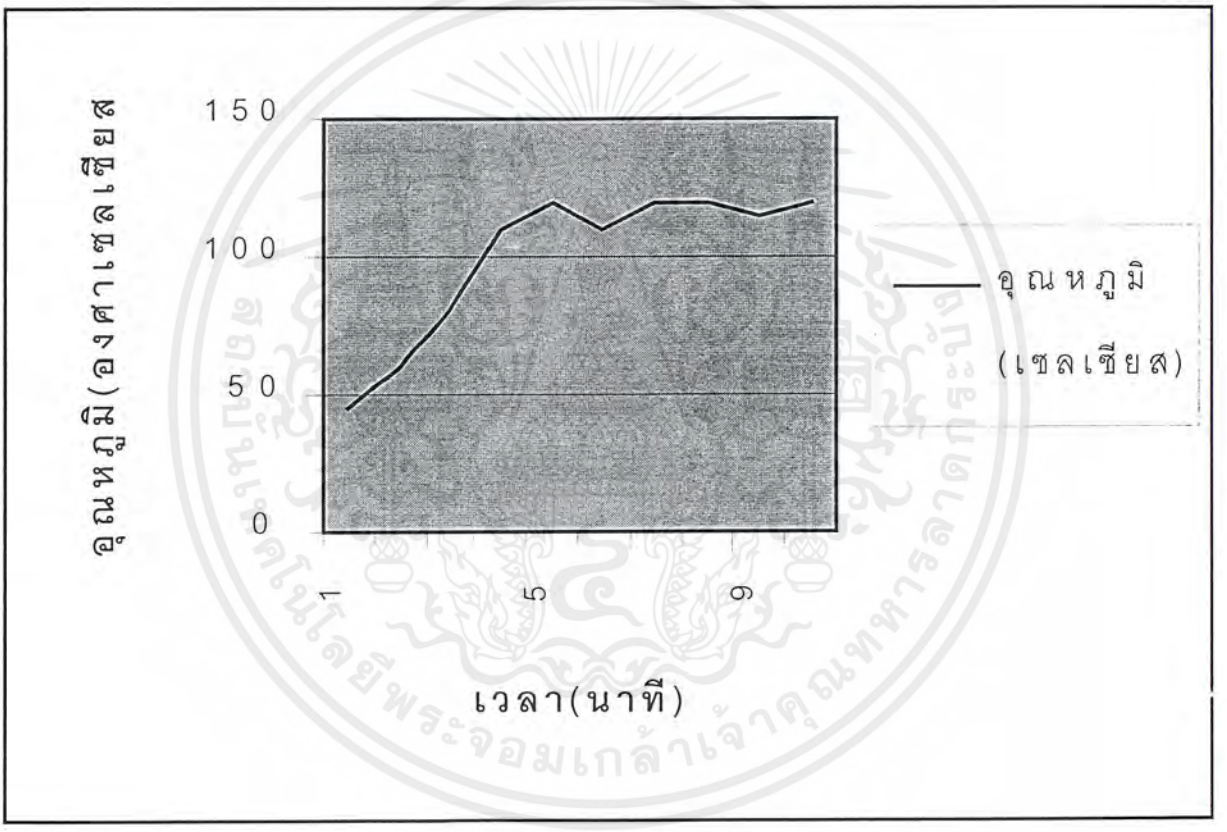
รูปที่4.12 บิทอินพุท 1111 1010 (250)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.5 การทดลอง หาความสัมพันธ์ของอุณหภูมิกับเวลา ที่จ่ายให้กับเตาอบ

4.5.1 เมื่อตั้งค่าอุณหภูมิ ที่100^{0}C เวลา 10 นาที

เวลา (นาที)	1	2	3	4	5	6	7	8	9	10
อุณหภูมิ (เซลเซียส)	45	60	80	110	120	110	120	110	125	110

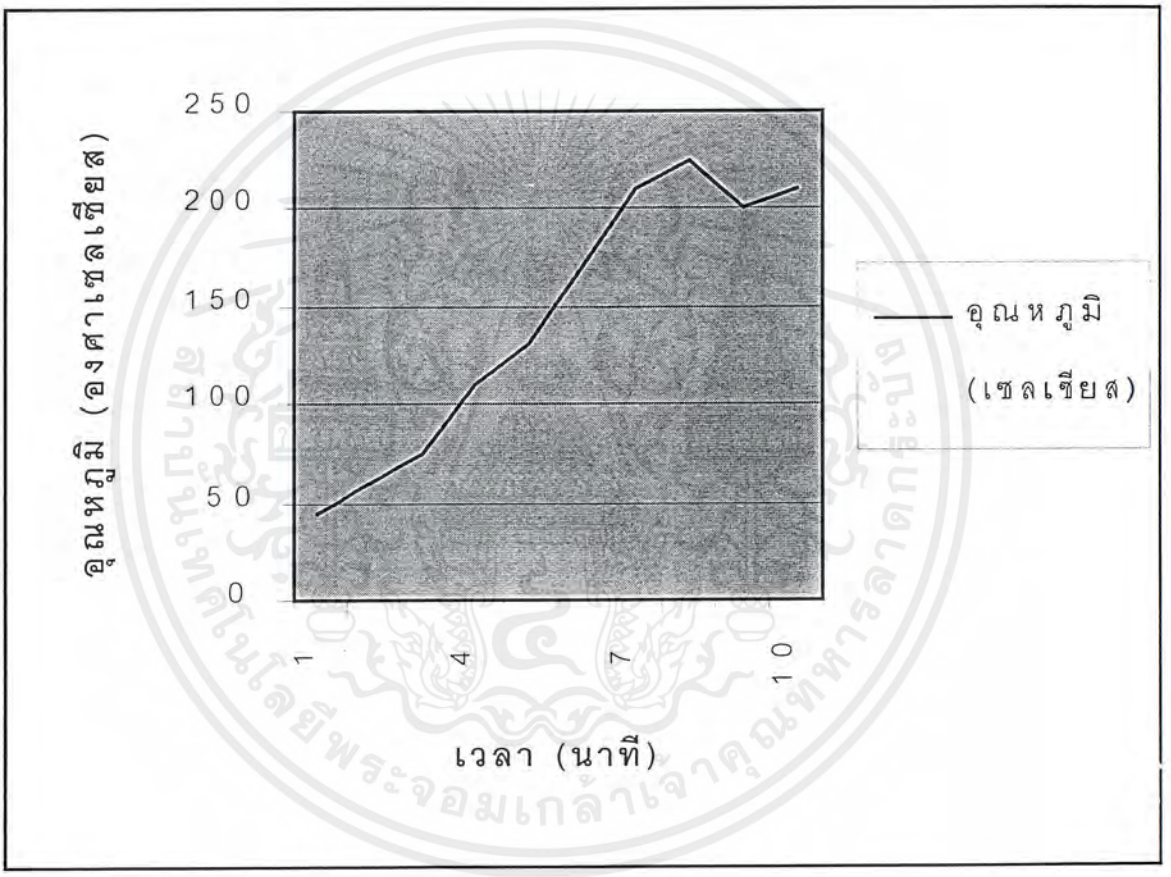


รูปที่ 4.13 กราฟแสดงอุณหภูมิกับเวลาตั้งค่าอุณหภูมิที่ 100 ^{0}C

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.5.2 เมื่อตั้งค่าอุณหภูมิ ที่ 200 ^{0}C เวลา 10 นาที

เวลา (นาที)	1	2	3	4	5	6	7	8	9	10
อุณหภูมิ(เซลเซียส)	45	60	75	110	130	170	210	225	200	210

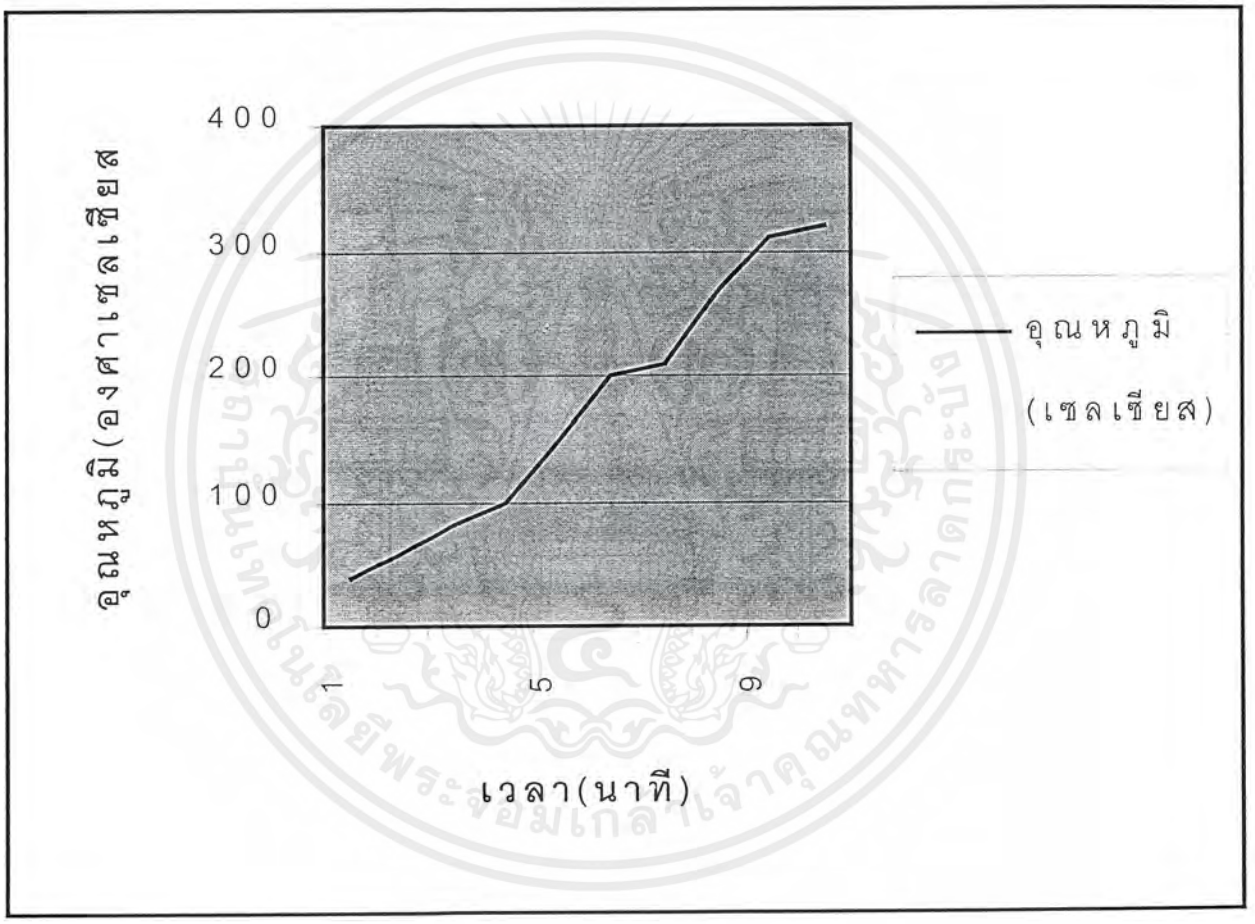


รูปที่ 4.14 กราฟแสดงอุณหภูมิกับเวลาตั้งค่าอุณหภูมิที่ 200 ^{0}C

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.5.3 เมื่อตั้งค่าอุณหภูมิ ที่ 300^{0}C เวลา 10 นาที

เวลา (นาที)	1	2	3	4	5	6	7	8	9	10
อุณหภูมิ (เซลเซียส)	40	60	80	100	150	200	210	270	310	320



รูปที่ 4.15 กราฟแสดงอุณหภูมิกับเวลาตั้งค่าอุณหภูมิที่ 300 ^{0}C

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 5
สรุปผลการทดลองและวิจารณ์

จากผลการทดลองในเรื่องของวงจร D/A Converter 8 Bit จะพบว่าค่าที่เปลี่ยนแปลง ในแต่ละบิต จากวงจรจะรับค่า I/P และทำให้ค่า O/P มีค่าเปลี่ยนแปลงได้ 255 ค่า เมื่อนำค่ามาบวกกันครบ 255 ค่าจะได้ค่าประมาณ 11.2 โวลต์ แต่ในทฤษฎีจะต้องได้ค่าประมาณ 10.2 โวลต์ ซึ่งค่าที่คลาดเคลื่อนไปน่าจะมีสาเหตุมาจากการป้อนค่ากระแสให้ D/A มีค่าคลาดเคลื่อนนั่นเอง

จากผลการทดลองในเรื่องของวงจร Control Phase จะได้ว่าเราสามารถนำค่า O/P ที่ได้จาก D/A มาควบคุมการที่เฟสที่ ทำการทริกไตรแอคได้โดยใช้สัญญาณพัลส์ ไปเปรียบเทียบเฟสกับไฟ AC ซึ่งสามารถควบคุมเฟสได้ ตั้งแต่ 0 องศา ถึง 180 องศา เพื่อให้สามารถควบคุมกำลังงานที่จ่ายให้กับโหลดได้ตามต้องการ

จากผลการทดลองในเรื่องในเรื่องของวงจร Instrument Amplifier เมื่อทำการต่อให้อัตราการขยายแรงดันในส่วนแรกของ Instrument Amplifier ให้สามารถปรับค่าได้ จะทำให้วงจรไม่เสถียร เพราะฉะนั้นจึงทำการกำหนด อัตราการขยายแรงดันมีค่าคงที่ คือ 1 เท่า และทำการกำหนดค่าอัตราการขยายแรงดันในส่วน Common Mode แทน ซึ่งจะทำให้วงจรมีอัตราการขยายแรงดันคงที่ และได้ O/P ตามที่ต้องการ

จากผลการทดลองของวงจร A/D Converter 8 Bit จะพบ ค่าที่เปลี่ยนแปลงในแต่ละ Bit O/P จากวงจรจะทำให้มีค่าเปลี่ยนแปลงได้ 255 ค่า โดยค่า I/P ที่ทำให้ O/P เปลี่ยนแปลงได้ 1 Bit จะมีค่าแรงดันประมาณ 0.02 โวลต์ ซึ่งทำให้ค่า I/P ที่เข้ามามีค่าได้ตั้งแต่ 0-5 โวลต์ จึงจะทำให้ค่า O/P มีค่าเปลี่ยนแปลงได้ ทั้ง 255 ค่า แต่จากการที่ RTD เปลี่ยนแปลงค่าซึ่งค่อยๆ เปลี่ยนแปลงค่าทีละน้อยๆ จึงทำให้แรงดัน O/P ไม่ค่อยเปลี่ยนแปลงให้อย่างชัดเจน

จากการทดลองการดึงข้อมูลจากอินเตอร์เน็ต โดยสามารถโหลดข้อมูลจากอินเตอร์เน็ต มาควบคุมการทำงานของเตา ซึ่งโปรแกรม Delphi สามารถรับค่าข้อมูลจาก อินเตอร์เน็ตมาใช้ และส่งผ่านพอร์ตขนาน มาสั่งงานเตาอบได้ แต่ในการทำงานบนหน้าจออินเตอร์เน็ต ไม่สามารถทำการคลิกบนหน้าจอเพื่อให้ส่งค่าออกทางพอร์ตขนานได้ ต้องทำการโหลด Text Files มาเก็บไว้ใน Dephi ก่อน เพราะว่าไม่สามารถใช้ Perl ให้สั่งงานทางไคลเอนต์ได้ โดย Perl จะทำการ Run ที่เซิฟเวอร์เท่านั้น

จากการทดลองการควบคุมอุณหภูมิของเตาอบที่ได้ อุณหภูมิที่ได้จะมีค่าไม่คงที่มากนักเนื่องมาจากเมื่อฟีดแบกกลับมาให้หยุดจ่ายไฟให้เตาอบ เตาอบจะมีอุณหภูมิสูงขึ้นไปเรื่อยๆ อยู่ช่วงหนึ่งน่าจะเป็นเพราะว่าเป็นระบบปิดทำให้ความร้อนเพิ่มขึ้นในช่วงขณะนั้น อุณหภูมิจึงแกว่งอยู่ในช่วงที่ ตั้งค่าไว้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สรุปผลการทดลองเมื่อนำวงจรในแต่ละภาคมาต่อรวมกันเพื่อให้เป็นวงจรที่สมบูรณ์และสมบูรณ์ และสามารถนำมาใช้งานได้ จะได้ว่าการควบคุมกำลังงานไฟฟ้าที่จ่ายให้เตาอบได้โดยใช้ ไมโครคอนโทรลเลอร์ควบคุมไตรแอค ให้ ON-OFF ได้ตามต้องการ และเมื่อส่งข้อมูลจากอินเตอร์เน็ตมาควบคุม อุณหภูมิและเวลาของเตาอบ สามารถทำได้ แต่ในการรรักษาระดับอุณหภูมิที่ตั้งไว้ให้มีค่าคงที่หรือมีค่าที่ใกล้เคียงกับค่าที่ตั้งไว้ จะมีความคลาดเคลื่อนเกิดขึ้นน่าจะมีสาเหตุเนื่องมาจาก RTD มีการเปลี่ยนแปลงค่าไปทีละน้อยทำให้วงจร Instrument Amp มีการเลี่ยนแปลงค่า O/P ช้าทำให้เทอร์โมมิเตอร์ วัดค่าอุณหภูมิที่ได้มีค่ามากกว่า อุณหภูมิที่ RTD ตรวจจับได้ในขณะนั้น และปัญหาที่เกิดขึ้นอีกอย่างหนึ่งก็คืออุณหภูมิที่เพิ่มขึ้นสูงมากทำให้การทำงานของอุปกรณ์ต่างๆในวงจรมีความคลาดเคลื่อนเกิดขึ้นได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เอกสารอ้างอิง

1. ทรงเกียรติ ภาวดี ,"เริ่มเขียนสคริปด้วยภาษา Perl ","แกะรอย CGI",บริษัท ซีเอ็ดยูเคชั่นจำกัด (มหาชน) พิมพ์ครั้งที่1 มิถุนายน 2542
2. งามนิจ อาจอินทร์ ,"การเขียนโปรแกรมบนเว็บ (Web Programming)",บริษัท ดวงกมลสมัย จำกัด พิมพ์ครั้งที่1 มกราคม 2542
3. กิตติ ภักดีวัฒนกุล " สร้าง Web Page แบบมืออาชีพด้วย HTML" ,บริษัทดวงกมลสมัย จำกัด พิมพ์ครั้งที่ 3 สิงหาคม2541
4. จักรพงษ์ สุขประเสริฐ ,กนกพร ภาวศุทธิกุล,สัจจะ จรัสรุ่งรวีวร," คู่มือการเขียนโปรแกรมด้วย Delphi" ,สำนักพิมพ์อินโฟเพรส พิมพ์ครั้งที่ 1 เมษายน 2542
5. สุนทร วิทูสุรพจน์ ," การโปรแกรมภาษาแอสแซมบลี ของ ไมโครคอนโทรลเลอร์ตระกูล 8051", บริษัท ซีเอ็ดยูเคชั่นจำกัด
6. วันชัย แซ่เตีย , สิทธิชัย ประสานวงศ์," สร้างเว็บเพจด้วย HTML4", บริษัทซีเอ็ดยูเคชั่น จำกัด พิมพ์ครั้งที่ 1 กุมพาพันธ์ 2542
7. ปรเมษฐ์ ประนยานันทน์ ,ปิยพงศ์ เผ่าวานิช," ไมโครคอนโทรลเลอร์ MCS-51"บริษัท ซีเอ็ดยูเคชั่น จำกัด พิมพ์ครั้งที่ 1 2540

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

พอร์ตขนาน

พอร์ตขนานหรือพอร์ตเครื่องพิมพ์มีขั้วต่อเป็นคอนเนคเตอร์แบบD ขนาด25ขา (DB-25)มีการจัดการสัญญาณดังรูปที่1 และมีหน้าที่การทำงานแสดงในตตารางที่1 ภายในพอร์ตขนานประกอบด้วยรีจิสเตอร์พื้นฐานที่ใช้ในการรับและส่งข้อมูล 2ตัวคือ รีจิสเตอร์ DATA, STATUS ,CONTROL โดยแอดแดรสของรีจิสเตอร์ทั้งสามนี้มีตำแหน่งไล่เรียงกันไปตามลำดับขึ้นอยู่กับแอสเดรสของพอร์ตขนาน เช่นที่พอร์ต LPT1 แอดเดรสของรีจิสเตอร์ DATA อยู่ที่ &H378 (Hเป็นตัวอักษรที่แสดงว่าข้อมูลนี้เป็นข้อมูลเลขฐานสิบหก เมื่อทำการเขียนโปรแกรมด้วย
QBASIC,VISUAL BASIC) ในขณะที่แอดเดรสของรีจิสเตอร์ STATUS จะอยู่ที่แอดเดรสของรีจิสเตอร์ CONTROL จะอยู่ที่ &H37A โดยความสัมพันธ์ระหว่างแอดเดรสพอร์คขนานกับแอดเดรสของรีจิสเตอร์พื้นฐานทั้งสามตัวแสดงในตารางที่2

ตารางที่ 1 หน้าที่และการทำงานของขาต่างๆ ของพอร์ตขนาน

ขา	หน้าที่	พอร์ต		ทิศทาง
		ชื่อ	บิต	
1	Strobe	Control	C0	เอาต์พุท
2	บิต 0	Data	D0	เอาต์พุท
3	Data บิต 1	Data	D1	เอาต์พุท
4	Data บิต 2	Data	D2	เอาต์พุท
5	Data บิต 3	Data	D3	เอาต์พุท
6	Data บิต 4	Data	D4	เอาต์พุท
7	Data บิต 5	Data	D5	เอาต์พุท
8	Data บิต 6	Data	D6	เอาต์พุท
9	Data บิต 7	Data	D7	เอาต์พุท
10	Acknowledge	Status	S6	อินพุท
11	Busy	Status	S7	อินพุท

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
;PROJECT MICROCONTROLLER CONTROL TEMPERATURE
;#################################################
;

        ORG   0000H
    RS    EQU   P2.4
    RW    EQU   P2.5
    EN    EQU   P2.6
    KEYBUF  EQU   30H   ;1 Byte Keep key is pressed
    LCDADD  EQU   31H   ;1 Byte Keep start address LCD
    LCDKEY  EQU   32H   ;1 Byte Keep key is take it
    KEYDEC  EQU   33H   ;1 Byte Keep key in decimal
    KEYASC  EQU   34H   ;1 Byte Keep key in asci
    LCDNUM  EQU   35H   ;1 Byte Keep number take in LCD
    LCDNUM1 EQU   36H
    LCDRADD EQU   37H   ;1 Byte Address LCD read
    HTEMP   EQU   38H   ;Keep TEMP in HEX
    HTIME   EQU   39H   ;Keep TIME in HEX
    PRESS   EQU   20H   ;1 Bit flag show press
    MENUS1  EQU   21H   ;1 Bit flag show menu1
    MENUS2  EQU   22H   ;1 Bit flag show menu2
    MENUS3  EQU   23H   ;1 Bit flag show menu3
    MENUS4  EQU   24H   ;1 Bit flag show menu4
    LETKEY  EQU   25H   ;1 Bit flag show let key
    SUBMIT  EQU   26H   ;1 Bit flag show operate
    CANCEL  EQU   27H   ;1 Bit flag show cancel

    PA_8255     EQU   0000H
    PB_8255     EQU   0100H
    PC_8255     EQU   0200H
    PD_8255     EQU   0300H

;Port interface
        RTCSCL  EQU   P2.2
        RTCSDA  EQU   P2.3
;Variables
 DISBUF  EQU   48H   ;8 Byte DS 1307
 HEXBUF  EQU   40H   ;8 Byte
 TIMBUF  EQU   50H   ;8 Byte
 LCDRDA  EQU   58H   ;3 Byte Keep temp in ASC
 LCDRDA1 EQU   5BH   ;3 Byte Keep time in ASC
 LCDRDD  EQU   5EH   ;3 Byte Keep temp in DEC
 LCDRDD1 EQU   61H   ;3 Byte Keep time in DEC
 TSTART  EQU   64H   ;Time start send output
 TCA     EQU   65H   ;Port start to compare temperature
 TCAS    EQU   66H   ;Port to send output
 COMTEMP EQU   67H   ;Temperature transfer from computer
 COMTIME EQU   68H   ;Time transfer from computer
 RTCERR  EQU   79H   ;1 Bit

        JMP   LCDSET
        ORG   0003H
INTA:   ACALL LCD_ADD_LINE1
        MOV   DPTR,#COM
        ACALL WRITE_DATA
        ACALL LCD_ADD_LINE2
        MOV   DPTR,#COM+8
        ACALL WRITE_DATA
        RETI

        ORG   0030H
CONTACT:    CALL  PORTA
        MOV   A,#0FFH
        MOVX  @DPTR,A
        CJNE  A,#00H,$+2
        SJMP  CONTACT
        CJNE  A,#0FFH,$+2
        SJMP  CONTACT
        MOVX  A,@DPTR
        MOV   COMTEMP,A
        MOVX  A,@DPTR
        MOV   COMTIME,A
        MOV   HTEMP,COMTEMP
        MOV   HTIME,COMTIME
        CALL  DIVTEMP
        CALL  DIVTIME
        SETB  SUBMIT
        CALL  MAIN1

LCDSET:     MOV   R5,#15      ;wait more than 15ms
        ACALL DELAY
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        CLR   RS          ;select command register
        CLR   RW          ;select write level
        SETB  EN
        MOV   P0,#38H
        CLR   EN
        MOV   R5,#5       ;wait more than 4.1ms
        ACALL DELAY
        SETB  EN
        MOV   P0,#38H
        CLR   EN
        MOV   R5,#1       ;wait more than 100us
        ACALL DELAY
        SETB  EN
        MOV   P0,#38H     ;function set
        CLR   EN
        MOV   A,#38H      ;function set
        ACALL STROBE
        MOV   A,#08H      ;display off
        ACALL STROBE
        MOV   A,#0CH      ;display on cursor on blink off
        ACALL STROBE
        MOV   A,#06H      ;entry mode set
        ACALL STROBE
        MOV   A,#01H      ;clear display
        ACALL STROBE
        MOV   A,#02H      ;return home
        ACALL STROBE

;*****************************************
;Set time in RTC

RTCTIME:  MOV   R5,#2
          ACALL DELAY
          MOV   TIMBUF,#50H    ;SECOND
          MOV   TIMBUF+1,#59H  ;MINUTE
          MOV   TIMBUF+2,#23H  ;HOUR
          MOV   TIMBUF+3,#01H  ;DAY
          MOV   TIMBUF+4,#31H  ;DATE
          MOV   TIMBUF+5,#12H  ;MONTH
          MOV   TIMBUF+6,#99H  ;YEAR
          CALL  RTSET
          CALL  RTWR
;*****************************************
MAIN:     ACALL LCD_ADD_LINE1
          MOV   DPTR,#TESTLCD
          ACALL WRITE_DATA
          ACALL LCD_ADD_LINE2
          MOV   DPTR,#TESTLCD+8
          ACALL WRITE_DATA
          ACALL P8255
          SJMP  $

TESTLCD:  DB    'TEST LCD XXXXXXX'
;*****************************************
LCD_ADD_LINE1: CLR  RS
          MOV   A,#080H
          ACALL STROBE
          RET

LCD_ADD_LINE2: CLR  RS
          MOV   A,#0C0H
          ACALL STROBE
          RET

LCD_ADD:  CLR   RS        ;Write any address ,Keep in Acc
          ACALL STROBE
          RET

WRITE_DATA: SETB  RS
          CLR   RW
          MOV   R2,#0FH   ;R2 keep number address
          MOV   R0,#0H
WRITE_DATA1: MOV  A,R0
          MOVC  A,@A+DPTR
          ACALL STROBE
          INC   R0
          DJNZ  R2,WRITE_DATA1
          CLR   EN
          RET
;********Write strobe**************
STROBE:   MOV   P0,#0FFH  ;configure P0 is input
          MOV   C,RS      ;save rs (instruction/data)
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        CLR   RS          ;read busy flag
        SETB  RW          ;set read level
WAIT:   SETB  EN          ;generate read strobe
        JNB   P0.7,WAIT_END  ;check for busy when BF=1
        CLR   EN          ;disable read strobe
        SJMP  WAIT

WAIT_END:  CLR  EN
        SETB  EN          ;disable read strobe
        MOV   RS,C        ;restore RS
        CLR   RW          ;write(instruction/data)
        MOV   P0,A        ;instruction/data
        CLR   EN          ;generate write strobe
        RET

CLEAR_DISPLAY: CLR  RS
        MOV   A,#01H      ;clear display
        ACALL STROBE
        MOV   A,#02H      ;curser at home
        ACALL STROBE
        MOV   A,#0FH
        ACALL STROBE
        RET

;*************************************
;SCAN KEYBROAD 3*4

P8255:  MOV   DPTR,#PD_8255
        MOV   A,#91H ;Set PA-INPUT PB-OUTPUT PC-INPUT-OUTPUT
        MOVX  @DPTR,A
        MOV   LCDRDA,#30H
        MOV   LCDRDA+1,#35H
        MOV   LCDRDA+2,#30H
        MOV   LCDRDA1,#30H
        MOV   LCDRDA1+1,#30H
        MOV   LCDRDA1+2,#30H
        MOV   LCDRDD1,#30H
        MOV   LCDRDD1+1,#30H
        MOV   LCDRDD1+2,#30H
        MOV   R4,#0FH
        CALL  CLEAR_DISPLAY
        ACALL LCD_ADD_LINE1
        MOV   DPTR,#MENU0
        ACALL WRITE_DATA
        ACALL LCD_ADD_LINE2
        MOV   DPTR,#MENU0+8
        ACALL WRITE_DATA
        SETB  MENUS1
        MOV   LCDADD,#80H
        MOV   HTEMP,#01H
        CLR   SUBMIT
        CLR   CANCEL
        SETB  EA
        SETB  EX0
        SETB  EX1
;*************************************
;Var KEYBUF
;R0-Round of scan,R1-Number of scan ,R2-Get data of scan

MAIN1:  CALL  GETKEY
        MOV   R2,KEYDEC
        CALL  MENUD
        CALL  WAITKEY
        SETB  T1
        CALL  LCD_WRITE_M
        SJMP  MAIN1
;*************************************
;Var KEYBUF
;R0-Round of scan,R1-Number of scan ,R2-Get data of scan
;R4-Keep number key in KEYBUF,

GETKEY: CALL  SCAN
        CALL  CHECK
        CALL  CONDEC
        MOV   KEYBUF,KEYDEC
        MOV   R5,#10      ;R5 keep round of delay
        CALL  DELAY
        CALL  SCAN
        CALL  CHECK
        CALL  CONDEC
        CALL  CONASC
        MOV   A,KEYDEC
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        CJNE   A,KEYBUF,GETKEY
        CALL   KEEPKEY1
        RET

KEEPKEY:    MOV   R4,#0FH
        MOV   DPTR,#KEYBUF
KEEPKEY1:   MOV   A,R2
        MOVX  @DPTR,A
        INC   DPTR
        DEC   R4
        MOV   A,R4
        JZ    KEEPKEY
        RET
;*************************************
SCAN:       SETB  C
        CALL  TEST1
        CALL  TEST2
        MOV   R0,=04H      ;Number column of keyboard+1
        MOV   R1,=11101111B  ;Scan column of keyboard
SCAN1:      CALL  KEYDOWN
        MOV   A,R1
        RLC   A
        MOV   R1,A
        MOV   A,R0
        DEC   A
        MOV   R0,A
        JZ    SCAN
        MOV   A,R3
        JZ    SCAN1
        RET

KEYDOWN:    CALL  PORTC
        MOV   A,R1
        MOVX  @DPTR,A
        NOP
        MOVX  A,@DPTR
        MOV   R2,A
        MOV   A,#0F0H
        ORL   A,R2         ;If no press A is 0FFH
        CPL   A
        MOV   R3,A         ;If no press R3 is 00H
        RET
;*************************************
WAITKEY:    CALL  SCANW
WAITKEY1:   NOP
        JB    LETKEY,WAITKEY
        MOV   R5,#1
        CALL  DELAY
        MOV   A,KEYBUF
        RET

SCANW:      SETB  C
        CLR   LETKEY
        MOV   R0,#04H      ;Number column of keyboard+1
        MOV   R1,#11101111B  ;Scan column of keyboard
SCANW1:     CALL  KEYDOWN
        MOV   A,R1
        RLC   A
        MOV   R1,A
        MOV   A,R0
        DEC   A
        MOV   R0,A
        JZ    WAITKEY1
        MOV   A,R3
        JZ    SCANW1
        SETB  LETKEY
        RET
;*************************************
PORTA:      MOV   DPTR,#PA_8255
        MOV   A,#0FFH
        MOVX  @DPTR,A
        RET

PORTB:      MOV   DPTR,#PB_8255
        MOV   A,#0FFH
        MOVX  @DPTR,A
        RET

PORTC:      MOV   DPTR,#PC_8255  ; Set Port C Of 8255
        MOV   A,#0FFH
        MOVX  @DPTR,A      ; Set PC is input
        RET
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
;**************************************
NOMENU:     NOP
            RET

MENUD:      CJNE   R2,#0AH,NOMENU
            MOV    LCDNUM1,#0

MENUD1:     JNB    MENUS1,MENUD2
            ACALL  LCD_ADD_LINE1
            MOV    DPTR,#MENU1
            ACALL  WRITE_DATA
            ACALL  LCD_ADD_LINE2
            MOV    DPTR,#MENU1+8
            ACALL  WRITE_DATA
            MOV    LCDADD,#085H
            MOV    A,LCDADD
            ACALL  LCD_ADD
            MOV    R2,#3
            MOV    A,#LCDRDA
            ACALL  WRITE_NUMBER
            MOV    LCDADD,#0C5H
            MOV    A,LCDADD
            ACALL  LCD_ADD
            MOV    R2,#3
            MOV    A,#LCDRDD1
            ACALL  WRITE_NUMBER
            SETB   MENUS2
            CLR    MENUS1
            RET
MENUDD1:    JMP    MENUD1

MENUD2:     JNB    MENUS2,MENUD3
            ACALL  LCD_ADD_LINE1
            MOV    DPTR,#MENU2
            ACALL  WRITE_DATA
            ACALL  LCD_ADD_LINE2
            MOV    DPTR,#MENU2+8
            ACALL  WRITE_DATA
            MOV    LCDADD,#0C4H
            MOV    A,LCDADD
            ACALL  LCD_ADD
            MOV    A,#LCDRDA
            MOV    R2,#3
            ACALL  WRITE_NUMBER
            MOV    LCDNUM,#03H
            MOV    LCDNUM1,#00H
            SETB   MENUS3
            CLR    MENUS2
            RET

MENUD3:     JNB    MENUS3,MENUD4
            ACALL  LCD_ADD_LINE1
            MOV    DPTR,#MENU3
            ACALL  WRITE_DATA
            ACALL  LCD_ADD_LINE2
            MOV    DPTR,#MENU3+8
            ACALL  WRITE_DATA
            MOV    LCDADD,#0C0H
            MOV    A,LCDADD
            ACALL  LCD_ADD
            MOV    A,#LCDRDA+3
            MOV    R2,#3
            ACALL  WRITE_NUMBER
            MOV    LCDNUM,#03H
            MOV    LCDNUM1,#00H
            CLR    MENUS3
            SETB   MENUS4
            RET

MENUD4:     JNB    MENUS4,MENUDD1
            CALL   CLEAR_DISPLAY      ;Clear display
            ACALL  LCD_ADD_LINE1
            MOV    DPTR,#MENU4
            ACALL  WRITE_DATA

;**************************************
READTIME:   CALL   LOOP
            MOV    R2,#8
            ACALL  SCANDIS
            CALL   WAITKEY
            ACALL  RTCGETKEY
            ACALL  TEST1
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        MOV   A,KEYDEC
        CJNE  A,#0AH,READTIME
        CLR   MENUS4
        SETB  MENUS1
        AJMP  MENUD1

LOOP:   CALL  RTRD
        MOV   HEXBUF+2,TIMBUF     ;keep second
        MOV   HEXBUF+1,TIMBUF+1   ;keep minute
        MOV   HEXBUF,TIMBUF+2     ;keep hour
        MOV   DISBUF+2,#3AH
        MOV   DISBUF+5,#3AH
        MOV   A,HEXBUF+2
        ACALL HTOA
        MOV   DISBUF+7,R3
        MOV   DISBUF+6,R2
        MOV   A,HEXBUF+1
        ACALL HTOA
        MOV   DISBUF+4,R3
        MOV   DISBUF+3,R2
        MOV   A,HEXBUF
        ACALL HTOA
        MOV   DISBUF+1,R3
        MOV   DISBUF+0,R2
        RET

SCANDIS: ACALL LCD_ADD_LINE2
        MOV   A,#DISBUF
        ACALL WRITE_NUMBER
        RET

RTCGETKEY: CALL RTCSCAN
        CALL  CHECK
        CALL  CONDEC
        MOV   KEYBUF,KEYDEC
        RET

RTCSCAN: SETB C
        CALL  LOOP
        MOV   R2,#8
        ACALL SCANDIS
        MOV   R0,#04H      ;Number column of keyboard+1
        MOV   R1,#11101111B ;Scan column of keyboard
RTCSCAN1: CALL KEYDOWN
        MOV   A,R1
        RLC   A
        MOV   R1,A
        MOV   A,R0
        DEC   A
        MOV   R0,A
        JZ    RTCSCAN
        MOV   A,R3
        JZ    RTCSCAN1
        RET
;.....
TEST1:  JNB   SUBMIT,NOMENU3
        JB    CANCEL,NOMENU3
        CALL  LOOP
        MOV   A,TIMBUF
        ; MOV  B,TIMBUF
        ; ANL  A,#00001111B
        ; CJNE A,#00H,$+2
        ; CALL COMPARE
        ; MOV  A,B
        CJNE  A,TSTART,NOMENU1
        ACALL OPERATE
        CALL  PORTB
        MOV   A,TCA
        MOVX  @DPTR,A
        SETB  CANCEL
        JNB   MENUS2,NOMENU1
        ACALL LCD_ADD_LINE1
        MOV   DPTR,#MENU1
        ACALL WRITE_DATA
        ACALL LCD_ADD_LINE2
        MOV   DPTR,#MENU1+8
        ACALL WRITE_DATA
        MOV   LCDADD,#085H
        MOV   A,LCDADD
        ACALL LCD_ADD
        MOV   R2,#3
        MOV   A,#LCDRDA
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        ACALL   WRITE_NUMBER
        MOV     LCDADD,#0C5H
        MOV     A,LCDADD
        ACALL   LCD_ADD
        MOV     R2,#3
        MOV     A,#LCDRDD1
        ACALL   WRITE_NUMBER
        RET

TEST2:  JNB     CANCEL,NOMENU1
        CALL    LOOP
        MOV     A,TIMBUF
        MOV     B,TSTART
        INC     B
        CJNE    A,B,NOMENU1
        JNB     MENUS2,NOMENU1
        CLR     CANCEL
        RET

NOMENU3:    NOP
        RET

;*************************************
;Compare Temperature Feedback To Port1 As TCA

COMPARE:    MOV     P1,#0FFH
        MOV     A,P1
        CJNE    A,TCA,$+1   ;Compare between ADC(A) and Tset(TCA)

INCREASE:   JNC     DECREASE
        MOV     A,TCAS
        CJNE    A,#255,$+2
        JMP     NOMENU3
        ADDC    A,#10
        MOV     TCAS,A
        RET

DECREASE:   MOV     A,TCAS
        CJNE    A,#50,$+1
        JC      NOMENU3
        SUBB    A,#10
        MOV     TCAS,A
        RET

;*************************************
NOMENU1:    NOP
        RET

LCD_WRITE_M:    NOP
        MOV     A,#0AH
        CJNE    A,KEYBUF,LCD_WRITE_M1
        RET

M1:     NOP
LCD_WRITE_M1:   JNB     MENUS2,LCD_WRITE_M2
        ACALL   LCD_ADD_LINE1
        MOV     DPTR,#MENU1
        ACALL   WRITE_DATA
        ACALL   LCD_ADD_LINE2
        MOV     DPTR,#MENU1+8
        ACALL   WRITE_DATA
        MOV     LCDADD,#085H
        MOV     A,LCDADD
        ACALL   LCD_ADD
        MOV     R2,#3
        MOV     A,#LCDRDA
        ACALL   WRITE_NUMBER
        MOV     LCDADD,#0C5H
        MOV     A,LCDADD
        ACALL   LCD_ADD
        MOV     R2,#3
        MOV     A,#LCDRDD1
        ACALL   WRITE_NUMBER
        MOV     A,KEYDEC
        CJNE    A,#0BH,NOMENU1

SUBMIT1:    JB      SUBMIT,CANCEL1
        CALL    ADDTIME
        MOV     A,HTIME
        JZ      NOMENU1
        SETB    SUBMIT
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        CALL   PORTB
        MOV    A,TCA
        MOVX   @DPTR,A
        CALL   LOOP
        MOV    A,TIMBUF
        DEC    A
        MOV    TSTART,A
        CLR    P2.7
        CLR    CANCEL
        RET

CANCEL1:    JNB    SUBMIT,SUBMIT1
OVERTIME:   CLR    SUBMIT
        CALL   PORTB
        MOV    A,#00H
        MOVX   @DPTR,A
        SETB   P2.7
        SETB   CANCEL
        RET

M2:         MOV    LCDNUM,#03H
        MOV    LCDNUM1,#00H
        RET
LCD_WRITE_M2:  JNB    MENUS3,LCD_WRITE_M3
        MOV    A,LCDADD
        ADD    A,LCDNUM1
        MOV    LCDRADD,A
        ACALL  LCD_ADD
        MOV    A,#LCDRDD
        ADD    A,LCDNUM1
        MOV    R0,A
        MOV    @R0,KEYDEC
        MOV    A,#LCDRDA
        ADD    A,LCDNUM1
        MOV    R0,A
        MOV    @R0,KEYASC
        MOV    A,R0
        MOV    R2,#1
        CALL   WRITE_NUMBER
        INC    LCDNUM1
        DJNZ   LCDNUM,NOMENU2
        DJNZ   LCDNUM,NOMENU2
        SJMP   M2
M11:        AJMP   LCD_WRITE_M1

M3:         MOV    LCDNUM,#03H
        MOV    LCDNUM1,#00H
        RET
LCD_WRITE_M3:  JNB    MENUS4,M11
        MOV    A,LCDADD
        ADD    A,LCDNUM1
        MOV    LCDRADD,A
        ACALL  LCD_ADD
        MOV    A,#KEYASC
        MOV    R2,#1
        ACALL  WRITE_NUMBER
        MOV    A,#LCDRDA1
        ADD    A,LCDNUM1
        MOV    R0,A
        MOV    @R0,KEYASC
        MOV    A,#LCDRDD1
        ADD    A,LCDNUM1
        MOV    R0,A
        MOV    @R0,KEYASC
        INC    LCDNUM1
        DJNZ   LCDNUM,NOMENU2
        SJMP   M3

NOMENU2:    NOP
        RET
;*****************************************
OPERATE:    CALL   TIMEHEX
        CALL   COMPUTE
        CALL   TIMEASC
        CALL   ADDTIME
        MOV    A,HTIME
        JZ     OVERTIME
        RET

TIMEHEX:    MOV    R3,LCDRDD1+2
        ACALL  ATOH
        ANL    A,#0FH
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        MOV   R3,LCDRDD1+1
        ACALL ATOH
        ANL   A,#0FH
        MOV   LCDRDD1+1,A
        MOV   R3,LCDRDD1
        ACALL ATOH
        ANL   A,#0FH
        MOV   LCDRDD1,A
        RET

COMPUTE:  MOV   A,LCDRDD1+2
        JZ    $+3
        DEC   LCDRDD1+2
        JMP   NOMENU2
        MOV   LCDRDD1+2,#9
        MOV   A,LCDRDD1+1
        JZ    $+3
        DEC   LCDRDD1+1
        JMP   NOMENU2
        MOV   LCDRDD1+1,#9
        MOV   A,LCDRDD1
        JNZ   $+5
        MOV   LCDRDD1,#00H
        MOV   LCDRDD1+1,#00H
        MOV   LCDRDD1+2,#00H
        JMP   NOMENU2
        DEC   LCDRDD1
        RET

TIMEASC:  MOV   A,LCDRDD1+2
        ACALL HTOA
        MOV   LCDRDD1+2,R3
        MOV   A,LCDRDD1+1
        ACALL HTOA
        MOV   LCDRDD1+1,R3
        MOV   A,LCDRDD1
        ACALL HTOA
        MOV   LCDRDD1,R3
        RET
;*****************************************
ADDTEMP:  MOV   HTEMP,#00
        MOV   A,LCDRDD
        MOV   B,#100
        MUL   AB
        ADD   A,HTEMP
        MOV   HTEMP,A
        MOV   A,LCDRDD+1
        MOV   B,#10
        MUL   AB
        ADD   A,HTEMP
        MOV   HTEMP,A
        MOV   A,LCDRDD+2
        MOV   B,#1
        MUL   AB
        ADD   A,HTEMP
        MOV   HTEMP,A
        SUBB  A,#50
        MOV   TCA,A
        RET

ADDTIME:  MOV   HTIME,#00
        CALL  TIMEHEX
        MOV   A,LCDRDD1
        MOV   B,#100
        MUL   AB
        ADD   A,HTIME
        MOV   HTIME,A
        MOV   A,LCDRDD1+1
        MOV   B,#10
        MUL   AB
        ADD   A,HTIME
        MOV   HTIME,A
        MOV   A,LCDRDD1+2
        ADD   A,HTIME
        MOV   HTIME,A
        CALL  TIMEASC
        RET

;*****************************************
DIVTIME:  MOV   A,HTIME
        MOV   B,#100
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        CALL   HTOA
        MOV    LCDRDD1,R3
        MOV    LCDRDA1,R3
        MOV    A,B
        MOV    B,#10
        DIV    AB
        CALL   HTOA
        MOV    LCDRDD1+1,R3
        MOV    LCDRDA1+1,R3
        MOV    A,B
        CALL   HTOA
        MOV    LCDRDD1+2,R3
        MOV    LCDRDA1+2,R3
        RET

DIVTEMP:    MOV    A,HTEMP
        MOV    B,#100
        DIV    AB
        CALL   HTOA
        MOV    LCDRDA,R3
        MOV    A,B
        MOV    B,#10
        DIV    AB
        CALL   HTOA
        MOV    LCDRDA+1,R3
        MOV    A,B
        CALL   HTOA
        MOV    LCDRDA+2,R3
        RET

;***********************************************

; ********** VS82_5 **********
; TEST RTC (DS1307) (TIMBUF = SS MM HH WW DD MM YY xx)

; ********** RTSET SUB **********
; SET RTC STATUS BIT (V-S82)
; IN  = TIMBUF  (8 BYTE)
; OUT = TIMBUF
; REG = A

RTSET:  MOV    A,TIMBUF      ;CLEAR CH (CLOCK HALT)
        CLR    ACC.7
        MOV    TIMBUF,A
        MOV    A,TIMBUF+2    ;SET 24 HOUR
        CLR    ACC.6
        MOV    TIMBUF+2,A
        MOV    A,TIMBUF+7    ;CONTROL REGISTER
        ANL    A,#11101100B  ;B7=OUT B4=SQWE B1,B0=RS
        ORL    A,#00000000B  ;DISABLE SQW, SET 1 Hz
        MOV    TIMBUF+7,A
        RET

; ********** RTWR SUB **********
; DS1307 RTC WRITE (V-S82)
; IN  = TIMBUF
; OUT = RTCERR  0=OK 1=ERROR
; REG = A,R0,R2,R3

RTWR:   CLR    RTCSDA        ;START
        CALL   CDEL
        CALL   CLOW

        MOV    A,#11010000B  ;DS1307 ADDRESS & R/W BIT
        CALL   RTWRB
        JB     RTCERR,RTWR9
        MOV    A,#0          ;START REGISTER ADDRESS=0
        CALL   RTWRB
        JB     RTCERR,RTWR9

        MOV    R0,#TIMBUF    ;8 BYTE
        MOV    R3,#8
RTWR2:  MOV    A,@R0
        CALL   RTWRB
        JB     RTCERR,RTWR9
        INC    R0
        DJNZ   R3,RTWR2

RTWR9:  CLR    RTCSDA        ;STOP
        CALL   CHIGH
        CALL   CDEL
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        RET

; ********** RTWRB SUB **********
; DS1307 RTC WRITE BYTE (V-S82)
; IN  = A
; OUT = RTCERR  0=OK 1=ERROR
; REG = A,R2

RTWRB:  CLR   RTCERR        ;CLEAR RTCERR FIRST
        MOV   R2,#8
RTWRB1: RLC   A
        MOV   RTCSDA,C
        CALL  CHIGH         ;CLOCK
        CALL  CLOW
        DJNZ  R2,RTWRB1

        SETB  RTCSDA        ;ACKNOWLEDGE BIT
        CALL  CHIGH
        JNB   RTCSDA,RTWRB2
        SETB  RTCERR
RTWRB2: CALL  CLOW
        RET

CHIGH:  SETB  RTCSCL        ;**** SCL HIGH + DELAY ****
        NOP
        NOP
        NOP
        RET

CLOW:   CLR   RTCSCL        ;**** SCL LOW + DELAY ****
        NOP
        NOP
        NOP
        RET

CDEL:   NOP                 ;**** DELAY ****
        NOP
        NOP
        RET

; ********** RTRD SUB **********
; DS1307 RTC READ (V-S82)
; OUT = TIMBUF
;       RTCERR  0=OK 1=ERROR
; REG = A,R0,R2,R3

RTRD:   CLR   RTCSDA        ;START
        CALL  CDEL
        CALL  CLOW

        MOV   A,#11010000B  ;DS1307 ADDRESS & R/W BIT
        CALL  RTWRB
        JB    RTCERR,RTRD9
        MOV   A,#0          ;START REGISTER ADDRESS=0
        CALL  RTWRB
        JB    RTCERR,RTRD9

        CLR   RTCSDA        ;STOP
        CALL  CHIGH
        CALL  CDEL
        SETB  RTCSDA
        CALL  CDEL
        CALL  CDEL

        CLR   RTCSDA        ;START (AGAIN)
        CALL  CDEL
        CALL  CLOW

        MOV   A,#11010001B  ;DS1307 ADDRESS & R/W BIT
        CALL  RTWRB
        JB    RTCERR,RTRD9

        MOV   R0,#TIMBUF    ;7 BYTE
        MOV   R3,#7
RTRD2:  CALL  RTRDB
        MOV   @R0,A
        INC   R0
        DJNZ  R3,RTRD2
        CALL  RTRDE         ;END BYTE
        MOV   @R0,A

; ********** RTRD SUB **********
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        CALL  CHIGH
        CALL  CDEL
        SETB  RTCSDA
        RET

; ********** RTRDB SUB **********
; DS1307 RTC READ BYTE (V-S82)
; OUT = A
; REG = A,R2

RTRDB:  MOV   R2,#8
RTRDB1: CALL  CHIGH
        MOV   C,RTCSDA
        RLC   A
        CALL  CLOW
        DJNZ  R2,RTRDB1

        CLR   RTCSDA        ;ACKNOWLEDGE BIT (0)
        CALL  CHIGH
        CALL  CLOW
        SETB  RTCSDA
        RET

RTRDE:  MOV   R2,#8         ;**** END BYTE ****
RTRDE1: CALL  CHIGH
        MOV   C,RTCSDA
        RLC   A
        CALL  CLOW
        DJNZ  R2,RTRDE1

        SETB  RTCSDA        ;ACKNOWLEDGE BIT (1)
        CALL  CHIGH
        CALL  CLOW
        RET

; ********** HTOA SUB **********
; CONVERT HEX TO ASCII
; IN  = A
; OUT = R2,R3
; REG = A,R2,R3

HTOA:   PUSH  ACC
        SWAP  A
        CALL  HTOAS
        MOV   R2,A
        POP   ACC
        CALL  HTOAS
        MOV   R3,A
        RET

HTOAS:  ANL   A,#0FH
        CJNE  A,#0AH,$+3
        JNC   HTOAS1
        ORL   A,#30H
        RET
HTOAS1: SUBB  A,#9
        ORL   A,#40H
        RET
; ********** ATOH SUB **********
; ASCII TO HEX CONVERT
; IN  = R2,R3  30H,41H
; OUT = A      0AH
; REG = A,R2
ATOH:   MOV   A,R2
        CALL  ATOHS
        SWAP  A
        MOV   R2,A
        MOV   A,R3
        CALL  ATOHS
        ORL   A,R2
        RET

ATOHS:  CJNE  A,#'A',$+3
        JC    ATOHS1
        ADD   A,#9
ATOHS1: ANL   A,#0FH
        RET

;**************************************

CHECK:      MOV   A,#00H
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
NUM1:   CJNE  R2,#0EEH,NUM2
        MOV   P1,#06H
        RET
NUM2:   INC   A
        CJNE  R2,#0DEH,NUM3
        MOV   P1,#05BH
        RET
NUM3:   INC   A
        CJNE  R2,#0BEH,NUM4
        MOV   P1,#04FH
        RET
NUM4:   INC   A
        CJNE  R2,#0EDH,NUM5
        MOV   P1,#066H
        RET
NUM5:   INC   A
        CJNE  R2,#0DDH,NUM6
        MOV   P1,#06DH
        RET
NUM6:   INC   A
        CJNE  R2,#0BDH,NUM7
        MOV   P1,#07DH
        RET
NUM7:   INC   A
        CJNE  R2,#0EBH,NUM8
        MOV   P1,#07H
        RET
NUM8:   INC   A
        CJNE  R2,#0DBH,NUM9
        MOV   P1,#07FH
        RET
NUM9:   INC   A
        CJNE  R2,#0BBH,NUMA
        MOV   P1,#06FH
        RET
NUMA:   INC   A
        CJNE  R2,#0E7H,NUM0
        MOV   P1,#0CH
        RET
NUM0:   INC   A
        CJNE  R2,#0D7H,NUMB
        MOV   P1,#03FH
        RET
NUMB:   INC   A
        CJNE  R2,#0B7H,NUMA
        MOV   P1,#0EH
        RET

CONDEC: MOV   DPTR,#BCD
        MOVC  A,@A+DPTR
        MOV   KEYDEC,A
        MOV   R2,A
        RET

CONASC: MOV   DPTR,#ASCI
        MOV   A,R2
        MOVC  A,@A+DPTR
        MOV   KEYASC,A
        RET

ASCI:   DB    30H,31H,32H,33H,34H
        DB    35H,36H,37H,38H,39H

BCD:    DB    01H,02H,03H,04H,05H,06H
        DB    07H,08H,09H,0AH,00H,0BH

LCD:    DB    80H,81H,82H,83H,84H,85H,86H,87H
        DB    0C0H,0C1H,0C2H,0C3H,0C4H,0C5H,0C6H,0C7H

WRITE_NUMBER: SETB  RS
        CLR   RW
        MOV   R0,A
WRITE_NUMBER1: MOV  A,@R0
        ACALL STROBE
        CLR   EN
        INC   R0
        DJNZ  R2,WRITE_NUMBER1
        RET

KEY:    DB    '01234567890AB'
MENU0:  DB    ' <MAIN MENU> '
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
MENU2:      DB    'TEMPERATURE:  C'
MENU3:      DB    'TIME:  000 Min '
MENU4:      DB    'TIMER: 00:00:00'
COM:        DB    'CONTACT COMPUTER'

DELAY:      MOV   R7,#2    ; 1 DELAY is 1ms
DELAY1:     MOV   R6,#20
            DJNZ  R6,$
            DJNZ  R7,DELAY1
            DJNZ  R5,DELAY
            RET
            END
```


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Guestbook.pl

```
#!/c:/program files/sambar42/perl
#########################################################################
#########
#Guestbook
# Set Variables

$guestbookurl = "http://161.246.2.12/oven/guestbook/guestbook.html";
$guestbookreal = "/www/guestbook/guestbook.html";
$guestlog = "/www/guestbook/guestlog.html";
$cgiurl = "http://161.246.2.12/cgi/guestbook.pl";
#$date_command = "/usr/bin/date";

# Set Your Options:
$mail = 0;            # 1 = Yes; 0 = No
$uselog = 1;          # 1 = Yes; 0 = No
$linkmail = 1;        # 1 = Yes; 0 = No
$separator = 1;       # 1 = <hr>; 0 = <p>
$redirection =0;      # 1 = Yes; 0 = No
$entry_order = 1;     # 1 = Newest entries added first;
                      # 0 = Newest Entries added last.
$remote_mail = 0;     # 1 = Yes; 0 = No
$allow_html = 1;      # 1 = Yes; 0 = No
$line_breaks = 0;           # 1 = Yes; 0 = No

# If you answered 1 to $mail or $remote_mail you will need to fill out
# these variables below:
$mailprog = '/usr/lib/sendmail';
$recipient = 'you@your.com';

# Done
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
##########################################################################

# Get the Date for Entry
#$date = `$date_command +"%A, %B %d, %Y at %T (%Z)"`; chop($date);
#$shortdate = `$date_command +"%D %T %Z"`; chop($shortdate);
{
($sec,$min,$hour,$mday,$mon,$year,$wday,$yday,$isdst) = localtime(time+
($time_miss*3600));
if ($sec < 10) { $sec = "0$sec"; }
if ($min < 10) { $min = "0$min"; }
if ($hour < 10) { $hour = "0$hour"; }
if ($mon < 10) { $mon = "0$mon"; }
if ($mday < 10) { $mday = "0$mday"; }
$mon++;
@month = ("ม.ค.","ก.พ.","มี.ค.","เม.ย.","พ.ค.","มิ.ย.","ก.ค.","ส.ค.","ก.ย.","ต.ค.","พ.
ย.", "ธ.ค.");
$year = $year +543 - 600;
$long_date = "$mday $month[$mon-1] 25$year  ($hour\:$min\:$sec)";
chop($long_date) if ($long_date =~ /\n$/);
$daten = $long_date;
$short_date = "$mday/$month[$mon-1]/$year";
$userip= $ENV{'REMOTE_ADDR'};
$date = "$mday $month[$mon-1] $year";
}
# Get the input
read(STDIN, $buffer, $ENV{'CONTENT_LENGTH'});

# Split the name-value pairs
@pairs = split(/&/, $buffer);

foreach $pair (@pairs) {
  ($name, $value) = split(/=/, $pair);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    # Un-Webify plus signs and %-encoding
    $value =~ tr/+/ /;
    $value =~ s/%([a-fA-F0-9][a-fA-F0-9])/pack("C", hex($1))/eg;
    $value =~ s/<!--(.|\n)*-->//g;

    if ($allow_html != 1) {
      $value =~ s/<([^>]|\n)*>//g;
    }
    $FORM{$name} = $value;
  }
  # Print the Blank Response Subroutines
  &no_comments unless $FORM{'comments'};
  &no_name unless $FORM{'realname'};

  # Begin the Editing of the Guestbook File
  open (FILE,"$guestbookreal") || die "Can't Open $guestbookreal: $!\n";
  @LINES=<FILE>;
  close(FILE);
  $SIZE=@LINES;

  # Open Link File to Output
  open (GUEST,">$guestbookreal") || die "Can't Open $guestbookreal: $!\n";

  for ($i=0;$i<=$SIZE;$i++) {
    $_=$LINES[$i];
    if (/<!--begin-->/) {
      if ($entry_order eq '1') {
        print GUEST "<!--begin-->\n";
      }
        if ($line_breaks == 1) {
        $FORM{'comments'} =~ s/\cM\n/<br>\n/g;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
}
print GUEST "<b>$FORM{'comments'}</b><br>\n";

if ($FORM{'url'}) {
  print GUEST "<a href=\"$FORM{'url'}\">$FORM{'realname'}</a>";
}
else {
  print GUEST "$FORM{'realname'}";
}

if ( $FORM{'username'} ){
  if ($linkmail eq '1') {
    print GUEST " \&lt;<a href=\"mailto:$FORM{'username'}\">";
    print GUEST "$FORM{'username'}</a>\&gt;";
  }
  else {
    print GUEST " &lt;$FORM{'username'}&gt;";
  }
}
print GUEST "<br>\n";

if ( $FORM{'city'} ){
  print GUEST "$FORM{'city'},";
}

if ( $FORM{'state'} ){
  print GUEST " $FORM{'state'}";
}

if ( $FORM{'country'} ){
  print GUEST " $FORM{'country'}";
}
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
      if ($separator eq '1') {
        print GUEST " - $date  :Time[$hour :$min]<hr>\n\n";
      }
      else {
        print GUEST " - $date  :Time[$hour :$min]<p>\n\n";
      }
      if ($entry_order eq '0') {
        print GUEST "<!--begin-->\n";
      }
    }
    else {
      print GUEST $_;
    }
  }
  close (GUEST);

  # Log The Entry

  if ($uselog eq '1') {
    &log('entry');
  }
  ########
  # Options

  # Mail Option
  if ($mail eq '1') {
    open (MAIL, "|$mailprog $recipient") || die "Can't open $mailprog!\n";
    print MAIL "Reply-to: $FORM{'username'} ($FORM{'realname'})\n";
    print MAIL "From: $FORM{'username'} ($FORM{'realname'})\n";
    print MAIL "Subject: Entry to Guestbook\n\n";
    print MAIL "You have a new entry in your guestbook:\n\n";
    print MAIL "-------------------------------------------------\n";
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    print MAIL "$FORM{'comments'}\n";
    print MAIL "$FORM{'realname'}";

    if ( $FORM{'username'} ){
      print MAIL " <$FORM{'username'}>";
    }
    print MAIL "\n";

    if ( $FORM{'city'} ){
      print MAIL "$FORM{'city'},";
    }

    if ( $FORM{'state'} ){
      print MAIL " $FORM{'state'}";
    }

    if ( $FORM{'country'} ){
      print MAIL " $FORM{'country'}";
    }

    print MAIL " - $date\n";
    print MAIL "-------------------------------------------------\n";

    close (MAIL);
  }

  if ($remote_mail eq '1' && $FORM{'username'}) {
    open (MAIL, "|$mailprog -t") || die "Can't open $mailprog!\n";

    print MAIL "To: $FORM{'username'}\n";
    print MAIL "From: $recipient\n";
    print MAIL "Subject: Entry to Guestbook\n\n";
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    print MAIL "Thank you for adding to my guestbook.\n\n";
    print MAIL "-------------------------------------------------\n";
    print MAIL "$FORM{'comments'}\n";
    print MAIL "$FORM{'realname'}";

    if ( $FORM{'username'} ){
      print MAIL " <$FORM{'username'}>";
    }
    print MAIL "\n";

    if ( $FORM{'city'} ){
      print MAIL "$FORM{'city'},";
    }
    if ( $FORM{'state'} ){
      print MAIL " $FORM{'state'}";
    }
    if ( $FORM{'country'} ){
     print MAIL " $FORM{'country'}";
    }
    print MAIL " - $date\n";
    print MAIL "-------------------------------------------------\n";

    close (MAIL);
  }

  # Print Out Initial Output Location Heading
  if ($redirection eq '1') {
    print "Location: $guestbookurl\n\n";
  }
  else {
    &no_redirection;
  }
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
#######################
# Subroutines

sub no_comments {
  print "Content-type: text/html\n\n";
  print "<html><head><title>No Comments</title></head>\n";
  print "<body background = \" /oven/img/rgree112.jpg\"><h1>กรุณาใส่ข้อความใน
ช่องแสดงความคิดเห็นด้วย</h1>\n";
  print "<form method=POST action=\"$cgiurl\">\n";
  print "Your Name:<input type=text name=\"realname\" size=30 ";
  print "value=\"$FORM{'realname'}\"><br>\n";
  print "E-Mail: <input type=text name=\"username\"";
  print "value=\"$FORM{'username'}\" size=40><br>\n";
  print "City: <input type=text name=\"city\" value=\"$FORM{'city'}\" ";
  print "size=15>  Country: <input type=text ";
  print "name=\"country\" value=\"$FORM{'country'}\" size=15><p>\n";
  print "Comments:<br>\n";
  print "<textarea name=\"comments\" COLS=60 ROWS=4></textarea><p>\n";
  print "<input type= submit  value = Submit> <input type=reset  value=  Reset>
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
#######################
# Subroutines

sub no_comments {
  print "Content-type: text/html\n\n";
  print "<html><head><title>No Comments</title></head>\n";
  print "<body background = \" /oven/img/rgree112.jpg\"><h1>กรุณาใส่ข้อความใน
ช่องแสดงความคิดเห็นด้วย</h1>\n";
  print "<form method=POST action=\"$cgiurl\">\n";
  print "Your Name:<input type=text name=\"realname\" size=30 ";
  print "value=\"$FORM{'realname'}\"><br>\n";
  print "E-Mail: <input type=text name=\"username\"";
  print "value=\"$FORM{'username'}\" size=40><br>\n";
  print "City: <input type=text name=\"city\" value=\"$FORM{'city'}\" ";
  print "size=15>  Country: <input type=text ";
  print "name=\"country\" value=\"$FORM{'country'}\" size=15><p>\n";
  print "Comments:<br>\n";
  print "<textarea name=\"comments\" COLS=60 ROWS=4></textarea><p>\n";
  print "<input type= submit  value = Submit> <input type=reset  value=  Reset>
</form><hr>\n";
  print "กลับไปดูที่<a href=\"$guestbookurl\">Guestbook</a>.";
  print "\n</body></html>\n";

  # Log The Error
  if ($uselog eq '1') {
    &log('no_comments');
  }
  exit;
}
sub no_name {
  print "Content-type: text/html\n\n";
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    print "<html><head><title>No Name</title></head>\n";
    print "<body  background =\"/oven/img/rgree112.jpg\" >\n";
    print "<h1>กรุณาใส่ชื่อของท่านด้วย</h1>\n";
    print "<form method=POST action=\"$cgiurl\">\n";
    print "Your Name:<input type=text name=\"realname\" size=30><br>\n";
    print "E-Mail: <input type=text name=\"username\"";
    print " value=\"$FORM{'username'}\" size=40><br>\n";
    print "City: <input type=text name=\"city\" value=\"$FORM{'city'}\" ";
    print "size=15>     Country: <input type=text ";
    print "value= Thailand  name=\"country\" value=\"$FORM{'country'}\" ";
    print "size=15><p>\n";
    print "Comments have been retained.<p>\n";
    print "<input type=hidden name=\"comments\" ";
    print "value=\"$FORM{'comments'}\">\n";
    print "<input type=submit  value=Submit> <input type=reset  value=Reset><hr>\n";
    print "กลับไปที่ <a href=\"$guestbookurl\">Guestbook</a>.";
    print "\n</body></html>\n";

    # Log The Error
    if ($uselog eq '1') {
      &log('no_name');
    }
    exit;
  }
  # Log the Entry or Error
  sub log {
    $log_type = $_[0];
    open (LOG, ">>$guestlog");
          if ($log_type eq 'entry') {
              print LOG "$ENV{'REMOTE_HOST'}  $FORM{'realname'}  -
[$short_date]  : Time [$hour:$min]<br>\n";
              close (LOG);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        }
        elsif ($log_type eq 'no_name') {
          print LOG "$ENV{'REMOTE_HOST'}  FORM{'realname'}  -
[$short_date] : Time [$hour:$min] - ERR: No Name<br>\n";
          close (LOG);
        }
        elsif ($log_type eq 'no_comments') {
          print LOG "$ENV{'REMOTE_HOST'}   FORM{'realname'}  -
[$short_date]  : Time [$hour:$min]- ERR: No ";
          print LOG "Comments<br>\n";
           close (LOG);
        }
      }
# Redirection Option
sub no_redirection {
  # Print Beginning of HTML
  print "Content-Type: text/html\n\n";
  print "<html><head><title>Thank You</title></head>\n";
  print "<body  background =\" /oven/img/rgree112.jpg\"><h1>Thank You For
Signing The Guestbook</h1>\n";

  # Print Response
  print "ขอขอบคุณทุกท่านที่ กรุณาแสดงความคิดเห็นข้อแนะนำ ให้กับเรา<br>
<br>\n";
  print "<b>แสดงความคิดเห็นที่ได้แนะนำ:</b><p>\n";
  print "<b>$FORM{'comments'}</b><br>\n";

  if ($FORM{'url'}) {
    print "<a href=\"$FORM{'url'}\">$FORM{'realname'}</a>";
  }
  else {
    print "$FORM{'realname'}";
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
}
if ( $FORM{'username'} ){
  if ($linkmail eq '1') {
    print " &lt;<a href=\"mailto:$FORM{'username'}\">";
    print "$FORM{'username'}</a>&gt;";
  }
  else {
    print " &lt;$FORM{'username'}&gt;";
  }
}
print "<br>\n";

if ( $FORM{'city'} ){
  print "$FORM{'city'},";
}

if ( $FORM{'state'} ){
  print " $FORM{'state'}";
}

if ( $FORM{'country'} ){
  print " $FORM{'country'}";
}

print " - $date :";
print " Time   $hour: $min: $sec \n";

# Print End of HTML
print "<hr>\n";
print "<a href=\"$guestbookurl\">กลับไปที่ Guestbook</a>\n";
print "</body></html>\n";
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
  exit;
}
```

Register.pl

```
#!c:/program files/sambar42/perl/perl.exe

$datafiles = "data.txt";
$recept = recept.htm;
$title = "REGISTER  FOR  OVEN";
$number ="number.txt";

##########################################################################
#######

#Date_variable
{
($sec,$min,$hour,$mday,$mon,$year,$wday,$yday,$isdst) = localtime(time+
($time_miss*3600));
if ($sec < 10) { $sec = "0$sec"; }
if ($min < 10) { $min = "0$min"; }
if ($hour < 10) { $hour = "0$hour"; }
if ($mon < 10) { $mon = "0$mon"; }
if ($mday < 10) { $mday = "0$mday"; }
$mon++;
@month = ("ม.ค.","ก.พ.","มี.ค.","เม.ย.","พ.ค.","มิ.ย.","ก.ค.","ส.ค.","ก.ย.","ต.ค.","พ.
ย.", "ธ.ค.");
$year = $year +543 - 600;
$long_date = "$mday $month[$mon-1] 25$year  ($hour\:$min\:$sec)";
chop($long_date) if ($long_date =~ /\n$/);
$daten = $long_date;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
$short_date = "$mday/$month[$mon-1]/$year";
$userip= $ENV{'REMOTE_ADDR'};
$date = "$mday $month[$mon-1] $year";
}
# Get data  and separate variable
&parse_form;
#Check  data  at  form
&check_data ;
#Check  confirm  password
&confirm_password;
#Check number to oven
 &check_number;
#Save  data  to  data.txt
&save_data;
##############################################
sub parse_form {
read(STDIN, $buffer, $ENV{'CONTENT_LENGTH'});
# Split the name-value pairs
@pairs = split(/&/, $buffer);
foreach $pair (@pairs) {
  ($name, $value) = split(/=/, $pair);
  $value =~ tr/+/ /;
  $value =~ s/%([a-fA-F0-9][a-fA-F0-9])/pack("C", hex($1))/eg;
  $value =~ s/<!--(.|\n)*-->//g;
  $FORM{$name} = $value;
        }
}
##############################################
sub  save_data {
open(DATA,">>$datafiles");
          print DATA   " Date[ $date]  : Time [$hour:$min] ";
         print DATA   " First name =  $FORM{'fname'} :";
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        print DATA  " Last name =  $FORM{'surname'} :";
        print DATA  " Number    = $FORM{'number'} :";
        print DATA  "Password  = $FORM{'code'} \n";
close(DATA);
print "Content-type: text/html\n\n";
print " <HTML><HEAD><TITLE>Test  Perl</TITLE></HEAD>\n ";
print "<BODY  Background=\" /oven/img/bgrnd151.gif\">\n ";
print "<Hr><center><h1><b>ขอขอบคุณที่กรุณาใช้บริการเว็บของเรา</b></h1>
</center>\n" ;
print "<br><hr><center><h1><b>ตอนนี้ท่านได้ผ่านการลงทะเบียนเรียบร้อยแล้ว และ
จะใช้งานได้ภายใน 12 ช.ม.</b></h1></center>\n" ;
print "<h2><TABLE    Align=center  HEIGHT= 0  BORDER= 1 >\n";
print "<TR>\n";
print "<TD>Name: $FORM{'fname'} </TD>\n";
print "<TD  >Number : $FORM{'number'} </TD>\n";
print "</TR>\n";
print "</TABLE></h2>\n";
print "</BODY></HTML>";
exit;
}
##########################################
sub rest_of_form  {
print "<form action=\"/cgi/register.pl\" method=\"post\">\n";
print " ชื่อ : <INPUT Type=\"text\"   Name=  fname    value = \"$FORM{'fname'}\"
Size=\"15\"  >  <Br>\n";
print "นามสกุล :<INPUT Type=\"text\" Name= surname    value = \"$FORM
{'surname'}\"Size=\"15\" Maxlength=\"15\"  ><Br>\n";
print "หมายเลขเครื่อง: <INPUT Type= \"text\"  Name= number   value = \"$FORM
{'number'}\"Size=\"15\"  Maxlength=\"6\"><Br>\n";
print "รหัส:<INPUT  Type =\"password\"  Name = code    Size=\"15\"  Maxlength
=\"10\"> <Br>\n";
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
print " ใส่รหัสซ้ำ:<INPUT Type =\"password\" Name = recode Size=\"15\"
Maxlength =\"10\"> <Br>\n";
print "<INPUT Type=\"submit\" Value=\"Submit\" >\n";
print "$time_miss";
print "</FORM></BODY></HTML>";
}
##############################################
sub check_data {
if ($FORM{'fname'}) {
  $name = "$FORM{'fname'}";
  $name =~ s/"//g;
  $name =~ s/<//g;
  $name =~ s/>//g;
  $name =~ s/\&//g;

 }
 else {
   &error(no_fname);
      }
if ($FORM{'surname'}) {
  $name = "$FORM{'surname'}";
  $name =~ s/"//g;
  $name =~ s/<//g;
  $name =~ s/>//g;
  $name =~ s/\&//g;

 }
 else {
   &error(no_surname);
      }
if ($FORM{'number'}) {
  $name = "$FORM{'number'}";
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    $name =~ s/"//g;
    $name =~ s/</g;
    $name =~ s/>//g;
    $name =~ s/\&//g;

  }
  else {
    &error(no_number);
      }
 if ($FORM{'code'}) {
  $name = "$FORM{'code'}";
  $name =~ s/"//g;
  $name =~ s/</g;
  $name =~ s/>//g;
  $name =~ s/\&//g;

  }
  else {
    &error(no_code);
      }
 if ($FORM{'recode'}) {
  $name = "$FORM{'recode'}";
  $name =~ s/"//g;
  $name =~ s/</g;
  $name =~ s/>//g;
  $name =~ s/\&//g;

  }
  else {
    &error(no_recode);
      }
}
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
##################################################
sub error {
  $error = $_[0];

  print "Content-type: text/html\n\n";

  if ($error eq 'no_fname') {
    print "<html><head><title>$title ERROR: No Name</title></head>\n";
    print "<body  Background=\" /oven/img/bgrnd151.gif\" bgcolor=pink><center>
<h1>ERROR: No Name</h1></center>\n";
    print " กรุณาใส่ชื่อ ของคุณในช่องด้วย<p><hr size=7 width=75%><p>\n";
    &rest_of_form;
        exit;
  }
   elsif ($error eq 'no_surname') {
    print "<html><head><title>$title ERROR: No Surname</title></head>\n";
    print "<body   Background= \" /oven/img/bgrnd151.gif\"  bgcolor=pink><center>
<h1>ERROR: No Surname</h1></center>\n";
    print " กรุณาใส่นามสกุลของคุณในช่องด้วย<p><hr size=7 width=75%><p>\n";
    &rest_of_form;
    exit;
  }
     elsif ($error eq 'no_number') {
    print "<html><head><title>$title ERROR: No Number</title></head>\n";
    print "<body   Background= \" /oven/img/bgrnd151.gif\"  bgcolor=pink><center>
<h1>ERROR: No Number</h1></center>\n";
    print " กรุณาใส่หมายเลขเครื่อง ของคุณในช่องด้วย<p><hr size=7 width=75%>
<p>\n";
    &rest_of_form;
    exit;
  }
     elsif ($error eq 'no_code') {
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        print "<html><head><title>$title ERROR: No Password</title></head>\n";
        print "<body    Background= \"/oven/img/bgrnd151.gif\" bgcolor=pink><center>
    <h1>ERROR: No  Password</h1></center>\n";
        print " กรุณาใส่รหัสผ่านของคุณที่ต้องการ ในช่องด้วย<p><hr size=7 width=75%>
    <p>\n";
        &rest_of_form;
        exit;
      }
         elsif ($error eq 'no_recode') {
        print "<html><head><title>$title ERROR: No Confirm  password</title>
    </head>\n";
        print "<body    Background= \"/oven/img/bgrnd151.gif\"  bgcolor=pink><center>
    <h1>ERROR: No  Confirm  password</h1></center>\n";
        print "กรุณาใส่รหัสผ่านที่ต้องการ อีกครั้ง ในช่องด้วย<p><hr size=7 width=75%>
    <p>\n";
        &rest_of_form;
        exit;
      }
    }
    ##########################################################
    sub confirm_password {
            if ($FORM{'code'}  ne  $FORM{'recode'} ) {
            print "Content-type: text/html\n\n";
            print "<html><head><title>$title Password  confirm  is  not  correct</title>
    </head>\n";
            print "<body    Background= \"/oven/img/bgrnd151.gif\"  bgcolor=pink>
    <center><h1>Error    Password</h1></center>\n";
            print "<h1> Password  confirm  is  not  correct </h1>";
        &rest_of_form;
            exit;
            }
    }
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
##############################################################
sub check_number {
        if ( open ( NUMBER,"$number") ) {
        @data  = <NUMBER> ;
        close(NUMBER);
                $sum = 0;
                $num  = ($FORM{'number'}) ;
                        $SIZE = @data;
                        for ($i=0;$i<=($SIZE-1);$i++) {
                        $_=$data[$i];
                        chop($_);
                        if ($_ eq $num) {
                        &save_data;
                        exit;
                        }
                }
        print "Content-type: text/html\n\n";
        print "<html><head><title>$title  Number is not correct.</title></head>\n";
        print "<body     Background= \"/oven/img/bgrnd151.gif\"   bgcolor=pink>
<center><h1>ERROR: Number is not correct</h1></center>\n";
        print "<h3>You  fill  number wrong .</h3><p><hr size=7 width=75%>
<p>\n";
         &rest_of_form;
        exit;
        close (NUMBER);
        }
        else {
        print "Content-type: text/html\n\n";
        print "File number.txt  does  not  open "; }

}
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Oven2.exe

```
unit oven2;
interface
uses
  Windows, Messages, SysUtils, Classes, Graphics, Controls, Forms, Dialogs,
  ExtCtrls, ComCtrls, StdCtrls;
type
  TForm1 = class(TForm)
    Edit1: TEdit;
    Edit2: TEdit;
    Edit3: TEdit;
    Label1: TLabel;
    Label2: TLabel;
    Label3: TLabel;
    Label4: TLabel;
    Label5: TLabel;
    ProgressBar1: TProgressBar;
    Timer1: TTimer;
    Memo1: TMemo;
    Edit4: TEdit;
    Memo2: TMemo;
    Label6: TLabel;
    Memo3: TMemo;
    Memo4: TMemo;
    Edit5: TEdit;
    Edit6: TEdit;
    procedure FormCreate(Sender: TObject);
    procedure Timer1Timer(Sender: TObject);
  private
    { Private declarations }
  public
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    { Public declarations }
  end;
var
  Form1: TForm1;
  IOport,i,num: integer;
  time1,time2,times :integer;

const x= 255;y = 888;
implementation

{$R *.DFM}
procedure portout(IOport:word;value:byte);stdcall;
 asm
   mov dx,[IOport]
   mov al,[value]
   out dx,al

end;

procedure TForm1.FormCreate(Sender: TObject);
begin

 Memo4.Lines.loadFromFile('temp.txt');
 Edit5.text := IntToStr(Memo4.ComponentIndex );
 Edit1.text := Memo4.Lines[1];
 Label3.Caption := Memo4.Lines[2];
 Label5.Caption := Memo4.Lines[1];
 Label6.Caption := Memo4.Lines[0];
 times := 1;

end;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
procedure TForm1.Timer1Timer(Sender: TObject);
var time1,time2,Temp,Temps :integer;
begin
    time1 := strtoint(Label5.caption);
    time2 := StrToInt(Edit1.text);
    ProgressBar1.Max := time1*60 ;
    Temps  := StrtoInt(Memo4.Lines[times+1]);
   if (Edit3.Text='0')and  (Edit1.text='0') then
   begin
      Timer1.Enabled:=False;
      portout( 888,255);
   end
   else
     begin
       if  (time1-time2) > 1*times then
       begin
       times := times+1;
       end;
       Temp := abs(Temps-256) ;
       Edit6.text := IntToStr(Temp);
       Timer1.enabled:=true;
       portout(888,Temp);
       if Edit3.Text='0' then
       begin
       Edit3.text:='60';
       Edit1.Text:=inttostr(strtoint(Edit1.Text)-1);
       end;
       Edit3.Text:=inttostr(strtoint(Edit3.text)-1) ;
       ProgressBar1.Position := ProgressBar1.Position+1  ;
    end;
end;
end.
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Bipolar IC

Type	Ordering code	Package
TCA 785	Q67000-A2321	P-DIP 16

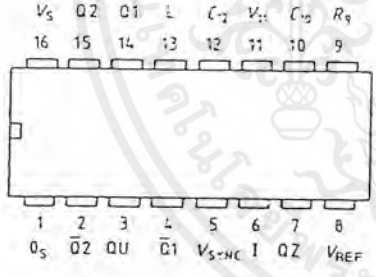
This phase control IC is intended to control thyristors, triacs, and transistors. The trigger pulses can be shifted within a phase angle between 0° and 180°. Typical applications include converter circuits, AC controllers and three-phase current controllers.

This IC replaces the previous types TCA 780 and TCA 780 D

Features

- Reliable recognition of zero passage
- Large application scope
- May be used as zero point switch
- LSL compatible
- Three-phase operation possible (3 ICs)
- Output current 250 mA
- Large ramp current range
- Large temperature range

Pin configuration
top view

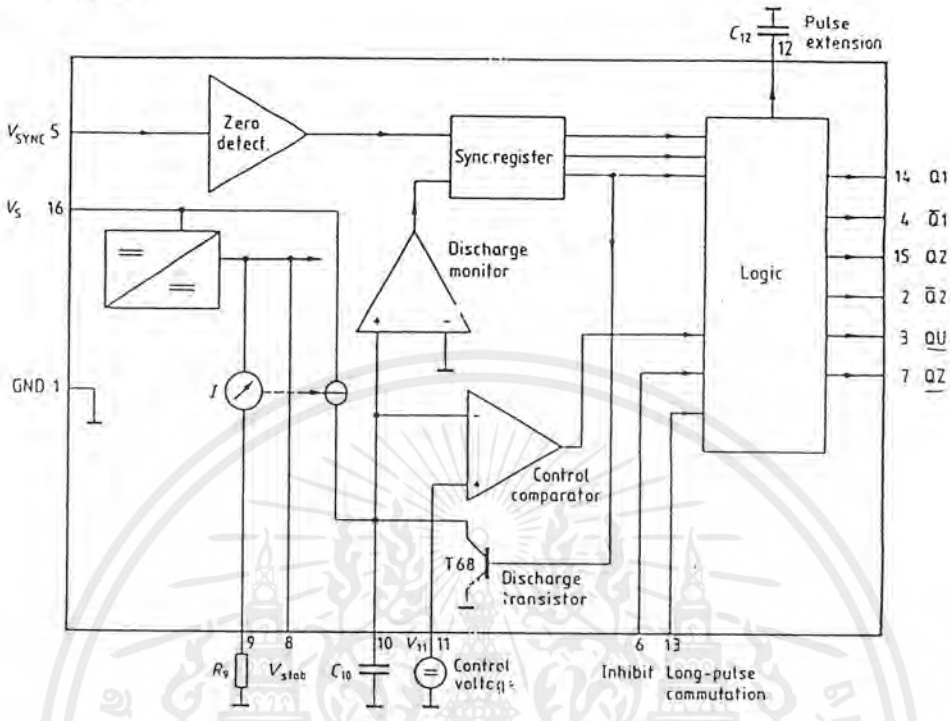


Pin description

Pin	Symbol	Function
1	0_S	Ground
2	$\overline{Q}2$	Output 2 inverted
3	$\overline{Q}U$	Output U
4	$\overline{Q}1$	Output 1 inverted
5	V_{SYNC}	Synchronous voltage
6	I	Inhibit
7	QZ	Output Z
8	V_{REF}	Stabilized voltage
9	R_9	Ramp resistance
10	C_{10}	Ramp capacitance
11	V_{11}	Control voltage
12	C_{12}	Pulse extension
13	L	Long pulse
14	Q1	Output 1
15	Q2	Output 2
16	V_S	Supply voltage

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Block diagram



Functional description

The synchronization signal is obtained via a high-ohmic resistance from the line voltage (voltage V_S). A zero voltage detector evaluates the zero passages and transfers them to the synchronization register.

This synchronization register controls a ramp generator, the capacitor C_{10} of which is charged by a constant current (determined by R_9). If the ramp voltage V_{10} exceeds the control voltage V_{11} (triggering angle φ), a signal is processed to the logic. Dependent on the magnitude of the control voltage V_{11}, the triggering angle φ can be shifted within a phase angle of 0° to 180°.

For every half wave, a positive pulse of approx. 30 µs duration appears at the outputs Q1 and Q2. The pulse duration can be prolonged up to 180° via a capacitor C_{12}. If pin 12 is connected to ground, pulses with a duration between φ and 180° will result.

Outputs $\overline{Q}1$ and $\overline{Q}2$ supply the inverse signals of Q1 and Q2.

A signal of $\varphi + 180°$ which can be used for controlling an external logic, is available at pin 3.

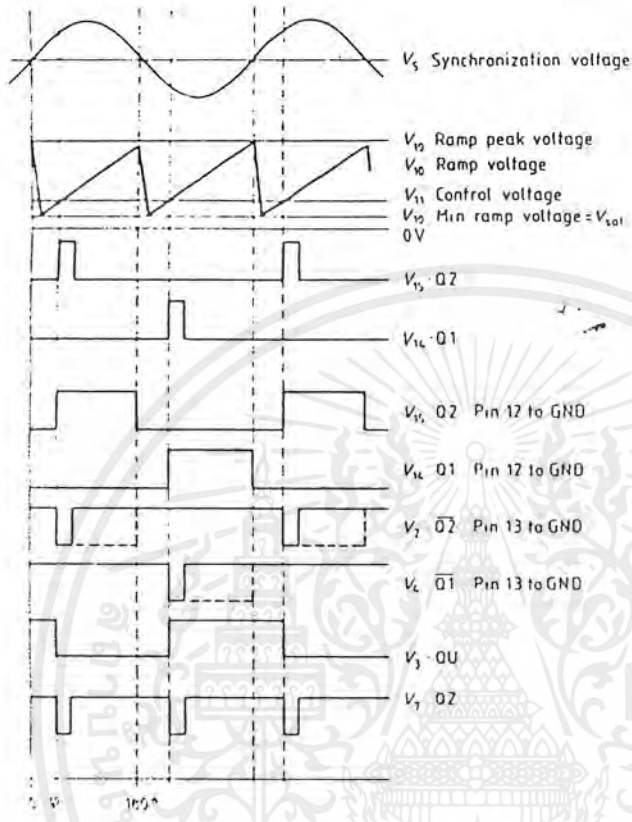
A signal which corresponds to the NOR link of Q1 and Q2 is available at output QZ (pin 7).

The inhibit input can be used to disable outputs Q1, Q2, $\overline{Q}1$, $\overline{Q}2$, QU.

Pin 13 can be used to extend the outputs $\overline{Q}1$ and $\overline{Q}2$ to full pulse length (180° – φ).

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Pulse diagram



Maximum ratings

		Lower limit B	Upper limit A	
Supply voltage	V_S	−0.5	18	V
Output current at pin 14, 15	I_Q	−10	400	mA
Inhibit voltage	V_5	−0.5	V_S	V
Control voltage	V_{11}	−0.5	V_S	V
Voltage short-pulse circuit	V_{12}	−0.5	V_S	V
Synchronization input current	I_5	−200	± 200	µA
Output voltage at pin 14, 15	V_Q		V_S	V
Output current at pin 2, 3, 4, 7	I_Q		10	mA
Output voltage at pin 2, 3, 4, 7	V_Q		V_S	V
Junction temperature	T		125	°C
Storage temperature	T_{stg}	−55	125	°C
Thermal resistance (system-air)	$R_{th\,SA}$		80	K/W

Operating range

Supply voltage	V_S	8	18	V
Operating frequency	f	10	500	Hz
Ambient temperature	T_A	−25	85	°C

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Characteristics

$8 \le V_S \le 18$ V; $-25\,°C \le T_A \le 85\,°C$; $f = 50$ Hz

		Test circuit	Lower limit B	$f = 50$ Hz $V_S = 15$ V typ	Upper limit A	
Supply current consumption S 1 ... S 6 open $V_{11} = 0$ V $C_{10} = 47$ nF, $R_9 = 100$ kΩ	I_S	1	4.5	6.5	10	mA
Synchronization pin 5 Input current R_9 varied	$I_{5\,rms}$	1	30		200	µA
Offset voltage	ΔV_5	4		30	75	mV
Control input pin 11 Control voltage range	V_{11}	1	0.2		$V_{10\,peak}$	V
Input resistance	R_{11}	5		15		kΩ
Ramp generator Load current	I_{10}		10		1000	µA
Max. ramp voltage	V_{10}	1			V_S-2	V
Saturation volt. at capacitor	V_{10}	1,6	100	225	350	mV
Ramp resistance	R_9	1	3		300	kΩ
Sawtooth return time	t_r	1		80		µs
Inhibit pin 6 switch-over of pin 7						
Outputs disabled	V_{6L}	1		3.3	2.5	V
Outputs enabled	V_{6H}	1	4	3.3		V
Signal transition time	t_r	1	1		5	µs
Input current $V_6 = 8$ V	I_{6H}	1		500	800	µA
Input current $V_6 = 1.7$ V	$-I_{6L}$	1	80	150	200	µA
Deviation of I_{10} R_9 = const $V_S = 12$ V, $C_{10} = 47$ nF	I_{10}	1	−5		5	%
Deviation of I_{10} R_9 = const $V_S = 8$ to 18 V	I_{10}	1	−20		20	%
Deviation of the ramp voltage between 2 following sawteeth, V_S = const	$\Delta V_{10\,max}$			± 1		%

Characteristics

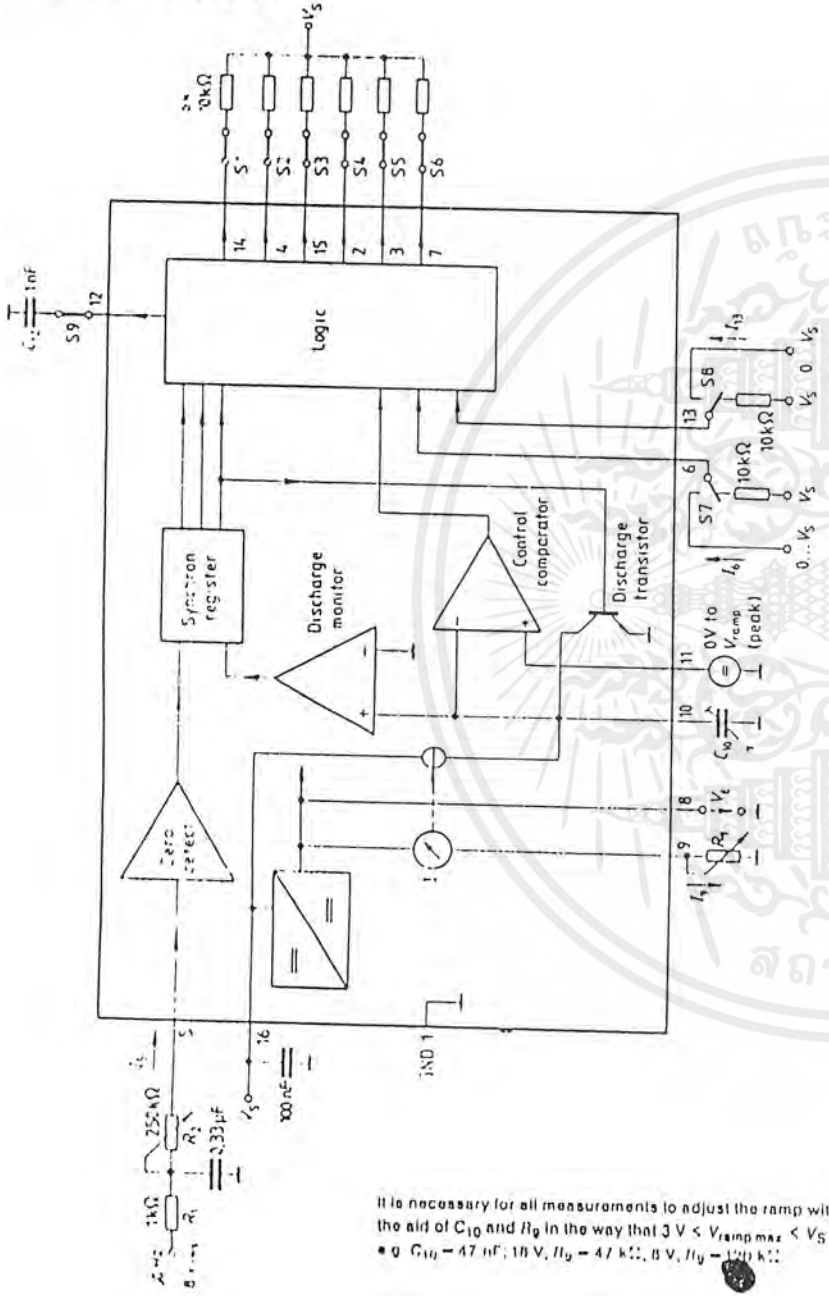
$8 \le V_S \le 18$ V; $-25\,°C \le T_A \le 85\,°C$; $f = 50$ Hz

		Test circuit	Lower limit B	$f = 50$ Hz $V_S = 15$ V typ	Upper limit A	
Long pulse switch-over pin 13 switch-over of S 8 Short pulse at output Long pulse at output Input current $V_{13} = 8$ V Input current $V_{13} = 1.7$ V	V_{13H} V_{13L} I_{13H} $-I_{13L}$	1 1 1 1	3.5 45	2.5 2.5 65	 2 10 100	V V µA µA
Outputs pin 2, 3, 4, 7 Reverse current $V_O = V_S$	I_{CEO}	2.6			10	µA
Saturation voltage $I_O = 2$ mA	V_{sat}	2.6	0.1	0.4	2	V
Outputs pin 14, 15 H output voltage $-I_O = 250$ mA	$V_{14/15\,H}$	3.6	V_S-3	$V_S-2.5$	$V_S-1.0$	V
L output voltage $I_O = 2$ mA	$V_{14/15\,L}$	2.6	0.3	0.8	2	V
Pulse width (short pulse) S 9 open	t_p	1	20	30	40	µs
Pulse width (short pulse) with C_{12}	t_p	1	530	620	760	µs/nF
Internal voltage control Reference voltage Parallel connection of 10 ICs possible	V_{ref}	1	2.8	3.1	3.4	V
TC of reference voltage	α_{ref}	1		2×10^{-4}	5×10^{-4}	1/K

Application hints for external components

		min	max	
Ramp capacitance	C_{10}	500 pF	1 µF[1]	The minimum and maximum values of I_{10} are to be observed
Triggering point	$t_{Tr} = \frac{V_{11} \times R_9 \times C_{10}}{V_{ref} \times K}$		[2]	
Charging current	$I_{10} = \frac{V_{ref} \times K}{R_9}$		[2]	Ramp voltage $V_{10\,max} = V_S - 2$ V $\quad V_{10} = \frac{V_{ref} \times K \times t}{R_9 \times C_{10}}$ [2]

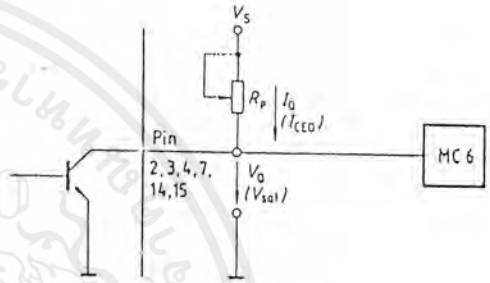
Test and measurement circuit 1



It is necessary for all measurements to adjust the ramp with the aid of C_{10} and R_9 in the way that $3\ V < V_{ramp\ MAX} < V_S - 2\ V$
e.g. $C_{10} = 47$ nF; 10 V, $R_9 = 47$ kΩ; 8 V, $R_9 = 100$ kΩ

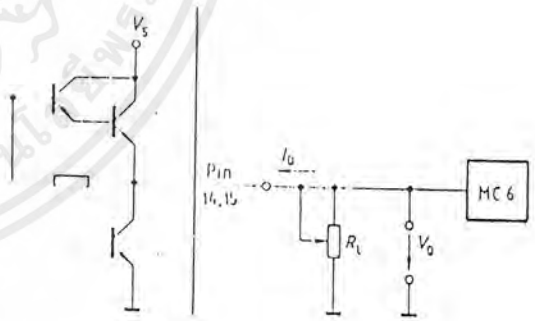
Test and measurement circuits

Measurement circuit 2



The residual pins are connected as in measurement circuit 1

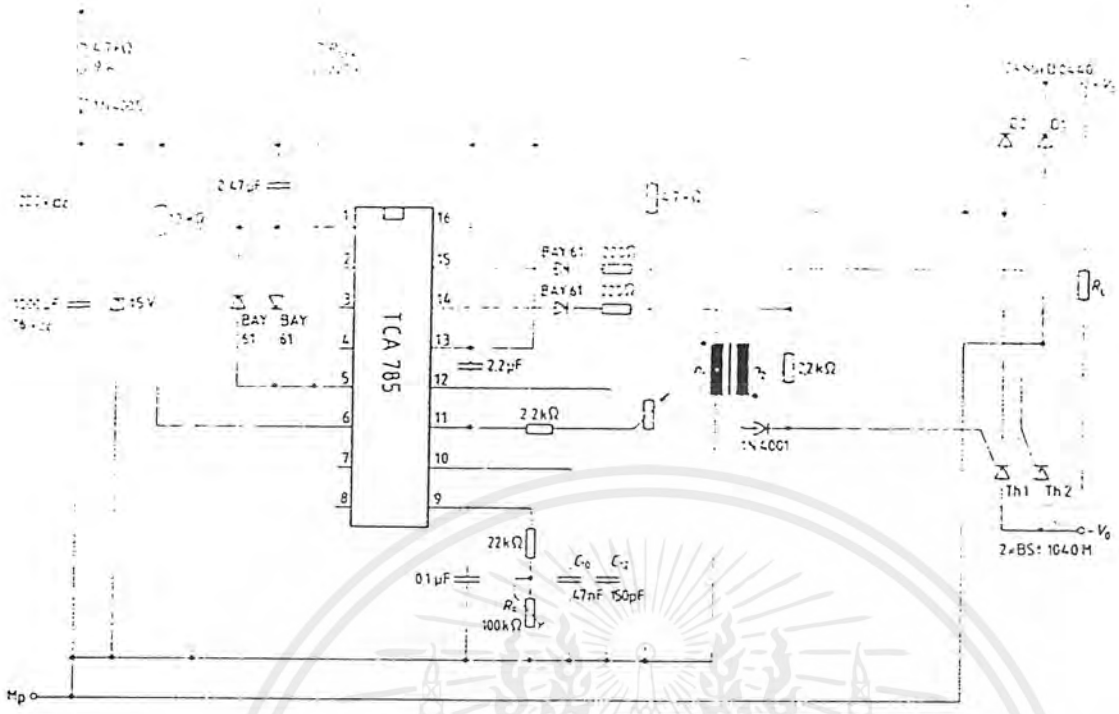
Measurement circuit 3



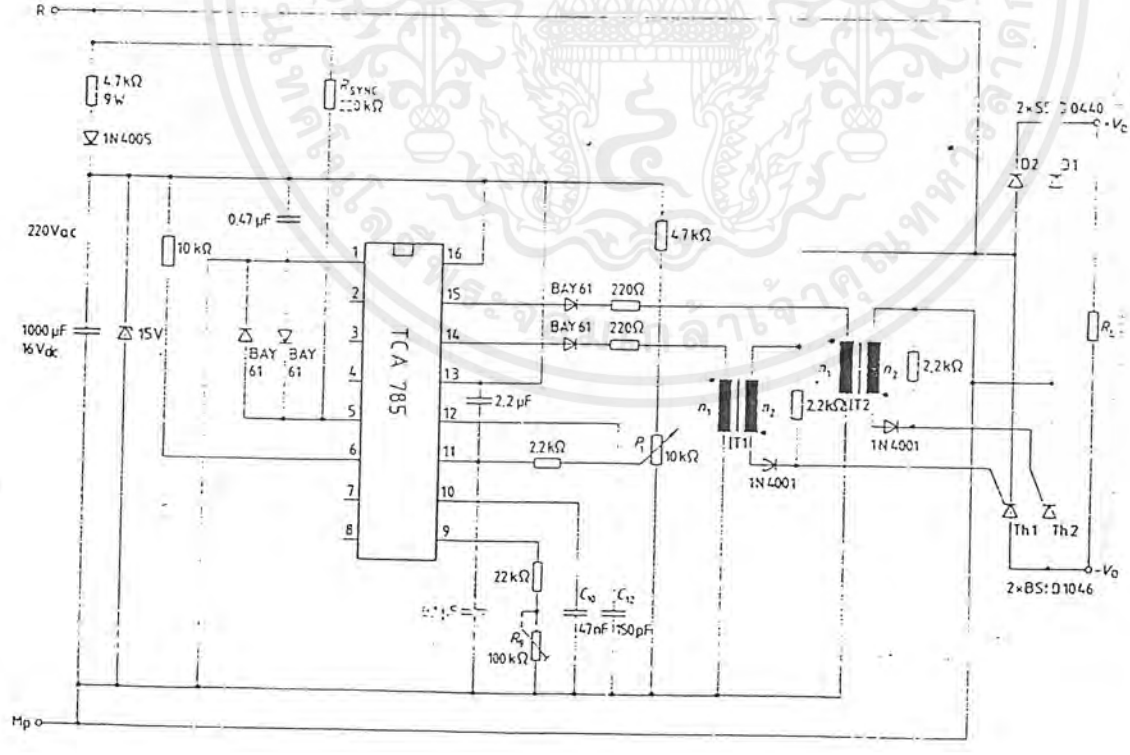
The residual pins are connected as in measurement circuit 1

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งไม่อนุญาตให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงลิขสิทธิ์ของเอกสารทุกครั้งที่มีการนำไปใช้

Half-controlled single-phase bridge circuit with trigger pulse transformer and direct control for low-power thyristors



Half-controlled single-phase bridge circuit with two trigger pulse transformers for low-power thyristors



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้