ปริญญานิพนธ์

ระบบตรวจจับและควบคุมด้วยเครื่องไมโครคอมพิวเตอร์
PC-SENSORS AND CONTROLLER

นายกิตติพล เครือวัลย์

นายนิด อ่อนอุทัย

นายพิสุทธิ์ อัมพุช

นายวีรพร ปานโชติ

A021050

เลขหมู่.....

เลขทะเบียน 1282 021050

วัน เดือน ปี -3 พ.ย. 2538

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรครุศาสตร์อุตสาหกรรมบัณฑิต

สาขาอิเล็กทรอนิกส์และคอมพิวเตอร์

ภาควิชาครุศาสตร์วิศวกรรม

คณะครุศาสตร์อุสาหกรรม

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2537

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ภาควิชาครุศาสตร์วิศวกรรม

คณะครุศาสตร์อุตสาหกรรม

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ใบรับรองปริญญานิพนธ์

ชื่อหัวข้อปริญญานิพนธ์ ระบบตรวจจับและควบคุมด้วยเครื่องไมโครคอมพิวเตอร์

– PC-SENSORS AND CONTROLLER

ชื่อนักศึกษา

1. นายกิตติพล เครือวัลย์ รหัสประจำตัว 36031403
2. นายนิด อ่อนอุทัย รหัสประจำตัว 36031416
3. นายพิสุทธิ์ อัมพุช รหัสประจำตัว 36031421
4. นายวีรพร ปานโชติ รหัสประจำตัว 36031427

อาจารย์ผู้ควบคุมปริญญานิพนธ์

1. อาจารย์กิติพงศ์ มะโน
2. อาจารย์วรวิทย์ สมหา
3. อาจารย์สันติ ตันตระกูล

คณะกรรมการสอบปริญญานิพนธ์		ลายมือ
1. อาจารย์กิติพงศ์	มะโน	.....
2. อาจารย์วิสุทธิ์	อธิพรธรรม	.....
3. อาจารย์วรวิทย์	สมหา	.....
4. อาจารย์สุชิน	อาจหาญ	.....
5. อาจารย์สันติ	ตันตระกูล	.....

วัน/เดือน/ปี ที่สอบ 25 ธันวาคม 2537 เวลา 16.00 น.

สถานที่สอบ ห้อง ค.301 คณะครุศาสตร์อุตสาหกรรม

.....

(ผศ.ดร.ธีรยุทธ [illegible] ยุธยา)

หัวหน้าภาควิชาครุศาสตร์วิศวกรรม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ปริญญานิพนธ์

เรื่อง ระบบตรวจจับและควบคุมด้วยเครื่องไมโครคอมพิวเตอร์
PC-SENSORS AND CONTROLLER

ผู้จัดทำ

1. นายกิตติพล เครือวัลย์ รหัสประจำตัว 36031403
2. นายนิด อ่อนอุทัย รหัสประจำตัว 36031416
3. นายพิสุทธิ์ อัมพุช รหัสประจำตัว 36031421
4. นายวีรพร ปานโชติ รหัสประจำตัว 36031427

อาจารย์ที่ปรึกษา

ลงนาม_____

(อาจารย์กิตินงค์ มะโน)

ลงนาม_____

(อาจารย์ารวิทย์ สมหา)

ลงนาม_____

(อาจารย์สันติ ตันตระกูล)

หัวหน้าภาควิชาครุศาสตร์วิศวกรรม

ลงนาม_____

(ผศ.ดร.ธีระพล เทพหัสดิน ณ อยุธยา)

ปีการศึกษา 2537

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ปริญญานิพนธ์

เรื่อง ระบบตรวจจับและควบคุมด้วยเครื่องไมโครคอมพิวเตอร์

PC-SENSORS AND CONTROLLER

จุดประสงค์

1. เพื่อศึกษาและพัฒนาโปรแกรมที่ใช้งานในระบบควบคุม บนไมโครซอฟท์วินด์โดวส์
2. เพื่อออกแบบและใช้งานฮาร์ดแวร์โดยใช้บอร์ดควบคุม CP-68HC11 ซึ่งใช้เชื่อมต่อระหว่าง PC และอุปกรณ์ต่างๆ จากภายนอก
3. เพื่อศึกษาและพัฒนาโปรแกรม ที่ใช้งานสำหรับบอร์ดควบคุม CP-68HC11 ซึ่งใช้ไมโครคอนโทรลเลอร์ 68HC11 เป็นซีพียู ด้วยภาษาแอสแซมบลี้
4. เพื่อสร้างระบบตรวจจับและควบคุมด้วยเครื่องไมโครคอมพิวเตอร์
5. เพื่อแสดงผลการตรวจจับและควบคุมต่างๆ ของระบบบนไมโครซอฟท์วินด์โดวส์
6. เพื่อสามารถนำไปใช้เป็นชุดสาธิตระบบควบคุมในสาขาเทคโนโลยีการวัดคุมทางอุตสาหกรรมที่กำลังจะเปิดใหม่ ในภาควิชาครุศาสตร์วิศวกรรม คณะครุศาสตร์อุตสาหกรรม สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ประโยชน์ที่คาดว่าจะได้รับ

1. ได้เข้าใจและพัฒนาโปรแกรมคอมพิวเตอร์ ที่ใช้ในงานระบบตรวจจับและควบคุมบนไมโครซอฟท์วินด์โดวส์และภาษาแอสแซมบลี้ของไมโครคอนโทรลเลอร์ เบอร์ 68HC11 มากยิ่งขึ้น
2. นำไมโครคอนโทรลเลอร์ เบอร์ 68HC11 ไปออกแบบฮาร์ดแวร์ ประยุกต์ใช้ในงานควบคุมต่างๆ ได้
3. ใช้เป็นชุดสาธิต ระบบตรวจจับและควบคุมด้วยไมโครคอมพิวเตอร์ ซึ่งเกี่ยวข้องกับสาขาเทคโนโลยีการวัดคุมทางอุตสาหกรรมได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ระบบตรวจจับและควบคุมด้วยเครื่องไมโครคอมพิวเตอร์

นายกิตติพล เครือวัลย์

นายนิด อ่อนอุทัย

นายพิสุทธิ์ อัมพุช

นายวีรพร ปานโชติ

อาจารย์ที่ปรึกษา

อ.กิติพงค์ มะโน

อ.วรวิทย์ สมหา

อ.สันติ ตันตระกูล

ปีการศึกษา 2537

บทคัดย่อ

ปริญญานิพนธ์ฉบับนี้เสนอระบบตรวจจับและควบคุมด้วยเครื่องไมโครคอมพิวเตอร์ ซึ่งสร้างเป็นชุดสาธิตการทำงานของระบบ โดยนำไมโครคอนโทรลเลอร์ เบอร์ 68HC11 มาประยุกต์ใช้งานในระบบควบคุมอุปกรณ์ไฟฟ้าต่างๆ ในลักษณะหรี่ไฟ, เปิดวงจรและปิดวงจรโซลิดสเตทรีเลย์ 220 โวลท์ ซึ่งสามารถกำหนดเวลาการทำงานอ้างอิงกับเวลาในปัจจุบัน และตรวจจับสัญญาณจากภายนอก โดยผ่านวงจรแปลงสัญญาณอนาลอกเป็นสัญญาณดิจิตอล

การใช้งานระบบควบคุมแบ่งเป็น 2 ลักษณะ คือ

1. ใช้งานในลักษณะควบคุมการทำงานที่ตัวเครื่อง โดยใช้เมตริกซ์คีย์บอร์ด ขนาด 4x4 และแสดงผลการใช้งานบนจอ แอลซีดี ขนาด 20 ตัวอักษร x 2 แถว

2. ใช้ไมโครคอมพิวเตอร์เป็นตัวควบคุม การทำงานต่างๆ ของระบบ โดยการส่งสัญญาณผ่านพอร์ทอนุกรม RS-232C ของไมโครคอมพิวเตอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

PC-SENSORS AND CONTROLLER

MR. KITTIPON KRUAWAN
MR. NID AEUNUTAI
MR. PISUT UMPUCH
MR. VEERAPORN PANCHOT

ADVISOR
MR. KITIPONG MANO
MR. WORAWIT SOMHA
MR. SUNTI TUNTRAKOOL
1994

ABSTRACT

THIS THESIS PRESENTS A PROJECT SENSORS AND CONTROLLER SYSTEM BY MICROCOMPUTER CREATED FOR OPERATING SYSTEM DEMONSTRATION PROJECT BY USED 68HC11 MICROCONTROLLER TO APPLIED FOR CONTROL ELECTRIC EQUIPMENTS IN DIMMER CIRCUIT, ON/OFF SOLIDSTATE RELAY BY 220 VOLTS CIRCUIT. IF CAN DEFIND OPERATING TIME WITH PRESENT REFERENCE TIME (REAL TIME CLOCK) AND DETECT SIGNAL BY USE ANALOG TO DIGITAL CONVERTER CIRCUIT.

CONTROLLING SYSTEM CAN PROVIDE 2 MODE

MODE 1 : IT SELF CONTROLLING USE MATRIX KEYBOARD 4 x 4 DIMENTION AND SHOW WORKING STATUS ON 20 CHARACTOR 2 ROW LCD DISPLAY.

MODE 2 : USED MICROCOMPUTER TO CONTROL REMOTE OPERATING SYSTEM AND USE MICROCOMPUTER'S RS-232C SERIAL PORT FOR COMMUNICATION.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กิตติกรรมประกาศ

ปริญญานิพนธ์ฉบับนี้สำเร็จลุล่วงได้ด้วยดี เพราะคำแนะนำจากอาจารย์ที่ปรึกษาและอาจารย์ทุกๆ ท่าน ในภาควิชาครุศาสตร์วิศวกรรม ตลอดจนได้รับความช่วยเหลือในด้านงบประมาณในการสร้างชิ้นงานและอุปกรณ์การทดลอง จากภาควิชา ครุศาสตร์วิศวกรรม คณะครุศาสตร์อุตสาหกรรม สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง ซึ่งทางคณะผู้จัดทำขอขอบพระคุณเป็นอย่างสูง

อนึ่ง ทางคณะผู้จัดทำรู้สึกซาบซึ้งและขอขอบคุณเพื่อนๆ ในสาขาอิเล็กทรอนิกส์และคอมพิวเตอร์ ที่ได้ให้ความช่วยเหลือทางด้านซอฟท์แวร์และข้อมูลต่างๆ รวมทั้งอุปกรณ์ในการจัดทำปริญญานิพนธ์นี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ

เรื่อง	หน้า
บทคัดย่อภาษาไทย	
บทคัดย่อภาษาอังกฤษ	
กิตติกรรมประกาศ	
สารบัญ	
สารบัญตาราง	
สารบัญรูปภาพ	
บทที่ 1 บทนำ	1
บทที่ 2 ทฤษฎี และหลักการ	2
2.1 68HC11ไมโครคอนโทรลเลอร์ขนาด 8 บิท ที่ใช้เทคโนโลยีไฮสปีดซีมอส	2
2.1.1 โครงสร้างของ 68HC11	2
2.1.2 คุณสมบัติของ 68HC11	4
2.1.3 ตระกูลของ 68HC11	4
2.1.4 โหมดการทำงานของ 68HC11	6
2.1.5 ลักษณะของสัญญาณและการจัดขา	11
2.1.6 การจัดหน่วยความจำ	16
2.1.7 หน่วยความจำแรมภายในซีพียู	16
2.1.8 วงจรกำเนิดสัญญาณนาฬิกา	19
2.1.9 รีจิสเตอร์ของซีพียู	19
2.1.10 วงจรวอตช์ด็อก	23
2.1.11 วงจรเอทูดี	23
2.1.12 ไทม์เมอร์	24
2.1.13 เรียลไทม์	25
2.1.14 พัลส์แอกคิวมูเลเตอร์	25
2.1.15 รีเซ็ต	25
2.1.16 อินเตอร์รัพต์	26
2.1.17 พอร์ทอนุกรม	26
2.1.18 มัลติโปรเซสเซอร์	26

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เรื่อง หน้า

2.2 การอ้างแอดเดรสและชุดคำสั่งของ 68HC11 27

2.2.1 การอ้างแอดเดรสของ 68HC11 27

2.2.2 พรีไบต์ (Prebyte) 29

2.2.3 ชุดคำสั่งของ 68HC11 29

2.2.4 ความแตกต่างของการ JUMP และ BRANCH 35

2.3 การเขียนโปรแกรมเพื่ออ้างแอดเดรส 36

2.3.1 โปรแกรมสำหรับการอ้างแอดเดรสแบบทันทีทันใด 36

2.3.2 โปรแกรมสำหรับการอ้างแอดเดรสแบบโดยตรง 36

2.3.3 โปรแกรมสำหรับการอ้างแอดเดรสแบบขยาย 37

2.3.4 โปรแกรมสำหรับการอ้างแอดเดรสแบบอินเด็กซ์ 37

2.3.5 โปรแกรมสำหรับการอ้างแอดเดรสแบบอินเฮียเรนต์ 37

2.3.6 โปรแกรมสำหรับการอ้างแอดเดรสแบบสัมพัทธ์ 37

2.4 รูปแบบการเขียนโปรแกรมทั่วไป 38

2.5 โปรแกรมคณิตศาสตร์ 40

2.5.1 โปรแกรมบวกลบเลข 40

2.5.2 โปรแกรมคูณเลข 40

2.5.3 โปรแกรมหารเลข 40

2.6 อินเตอร์เฟส RS-232 C 41

2.6.1 แบบจำลองของวงจรการสื่อสารแบบ RS-232 C 41

2.6.2 ข้อกำหนดทางไฟฟ้า 43

2.6.3 ขาขั้วต่อ RS-232 C และหน้าที่ 45

2.6.4 การควบคุมคลื่นพาห์และการส่งข้อมูล 47

2.6.5 ช่องสัญญาณรอง 50

2.6.6 การตอบรับอย่างอัตโนมัติ 51

2.6.7 การให้จังหวะเวลาแก่วงจรชิงโครนัส 51

2.6.8 ข้อจำกัดของ RS-232 C 54

2.7 นาฬิกาบอกเวลาที่แท้จริง (REAL TIME CLOCK) 55

2.7.1 การจัดขาและหน้าที่ของขาต่าง ๆ 56

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เรื่อง	หน้า
2.8 วงจรโซลิดสเตทรีเลย์ (SOLID STATE RELAY)	57
2.8.1 ZERO SWITCHING	57
2.8.2 ข้อดีของโซลิดสเตทรีเลย์	58
2.8.3 ข้อเสียของโซลิดสเตทรีเลย์	58
2.9 วงจรควบคุมความเข้มของแสงไฟด้วยสัญญาณดิจิตอล	59
2.9.1 บล็อกไดอะแกรมของวงจรควบคุมความเข้มของแสงไฟด้วยสัญญาณดิจิตอล	59
2.9.2 การทำงานของวงจรควบคุมความเข้มของแสงไฟด้วยสัญญาณดิจิตอล	59
บทที่ 3 การออกแบบและการสร้าง	60
3.1 บล็อกไดอะแกรมของระบบตรวจจับและควบคุมด้วยเครื่องไมโครคอมพิวเตอร์	60
3.1.1 ใช้งานบนเครื่องไมโครคอมพิวเตอร์ (PC)	60
3.1.2 ใช้งานบนชุดสาธิตระบบตรวจจับและควบคุม	61
3.2 ส่วนประกอบโดยทั่วไป	61
3.2.1 ส่วนชุดควบคุม (CP-68HC11 BOARD)	61
3.2.2 ส่วนวงจรนาฬิกาบอกเวลาที่แท้จริง (REAL TIME CLOCK)	61
3.2.3 ส่วนเมตริกซ์คีย์บอร์ด (MATRIX KEYBOARD)	61
3.2.4 ส่วนการแสดงผลการใช้งานบนจอแอลซีดี (LCD DISPLAY)	61
3.2.5 ส่วนวงจรถอดรหัสเอาท์พุทภายนอก (DECODER CIRCUIT)	61
3.2.6 ส่วนวงจรควบคุมความเข้มของแสงไฟ	62
3.2.7 ส่วนวงจรโซลิดสเตทรีเลย์ (SOLID STATE RELAY)	62
3.2.8 ส่วนวงจรดีซีโวลท์มิเตอร์ (DC VOLTMETER CIRCUIT)	62
3.2.9 ส่วนของวงจรตรวจจับ (SENSORS CIRCUIT)	62
3.2.10 ส่วนของแหล่งจ่ายไฟ (POWER SUPPLY)	62
3.3 การออกแบบและการทำงานในส่วนต่างๆ	62
3.3.1 ส่วนชุดควบคุม	62
3.3.2 ส่วนวงจรนาฬิกาบอกเวลาที่แท้จริง (REAL TIME CLOCK)	73
3.3.3 ส่วนของเมตริกซ์คีย์บอร์ด (MATRIX KEYBOARD)	74

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เรื่อง หน้า

3.3.4 ส่วนการแสดงผลการใช้งานบนจอแอลซีดี (LCD DISPLAY) 75
3.3.5 ส่วนวงจรถอดรหัสพอร์ทภายนอก (DECODER CIRCUIT) 76
3.3.6 ส่วนวงจรควบคุมความเข้มของแสงไฟ 77
3.3.7 ส่วนนวงจรโซลิดสเตทรีเลย์ (SOLID STATE RELAY) 78
3.3.8 ส่วนวงจรดีซีโวลท์มิเตอร์ (DC VOLTMETER CITCUIT) 79
3.3.9 ส่วนของวงจรตรวจจับ (SENSORS CIRCUIT) 80
3.3.10 ส่วนของแหล่งจ่ายไฟ (POWER SUPPLY) 81
บทที่ 4 การทดลองและผลการทดลอง 89
4.1 การทดลองการทำงานของโซลิดสเตทรีเลย์ 89
4.1.1 ลำดับขั้นการทดลอง 89
4.1.2 ผลการทดลอง 89
4.2 การทดลองการทำงานของวงจรถอดรหัสพอร์ทภายนอก 89
4.2.1 ลำดับขั้นการทดลอง 89
4.2.2 ผลการทดลอง 89
4.3 การทดลองวงจรควบคุมความเข้มของแสงไฟ 90
4.3.1 ลำดับขั้นการทดลอง 90
4.3.2 ผลการทดลอง 90
4.4 การทดลองการทำงานของเมตริกซ์คีย์บอร์ด 91
4.4.1 ลำดับขั้นการทดลอง 91
4.4.2 ผลการทดลอง 91
4.5 การทดลองการทำงานชุดควบคุม (CP-68HC11 BOARD) 91
4.5.1 ลำดับขั้นการทดลอง 91
4.5.2 ผลการทดลอง 91
4.6 การทดลองโปรแกรมมอนิเตอร์ของบอร์ด CP-68HC11 91
4.6.1 ลำดับขั้นการทดลอง 91
4.6.2 ผลการทดลอง 92
4.7 การทดลองวงจรนาฬิกาบอกเวลาจริง (REAL TIME CLOCK) 92
4.7.1 การทดลอง 92
4.7.2 ผลการทดลอง 92

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์ไว้สำหรับเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เรื่อง	หน้า
4.8 การทดลองวงจรตรวจจับแรงดันไฟตรง 0-500 โวลท์	92
4.8.1 ลำดับขั้นการทดลอง	92
4.8.2 ผลการทดลอง	93
4.9 สรุปผลการทดลอง	93
บทที่ 5 บทวิจารณ์ สรุป และแนวทางในการพัฒนา	94
5.1 บทสรุป	94
5.1.1 ควบคุมการทำงานของระบบบนตัวเครื่องโดยตรง	94
5.1.2 ควบคุมการทำงานของระบบด้วยไมโครคอมพิวเตอร์ (PC)	94
5.2 ปัญหาและแนวทางการแก้ไข	95
5.3 แนวทางในการพัฒนา	96
ภาคผนวก ก วงจรส่วนต่างๆของระบบตรวจจับและควบคุม	97
ภาคผนวก ข ลักษณะการจัดวางอุปกรณ์และการจัดวางขาคอนเนคเตอร์	105
ภาคผนวก ค โฟลว์ชาร์ตระบบตรวจจับและควบคุมด้วยไมโครคอมพิวเตอร์	108
ภาคผนวก ง โปรแกรมมอเตอร์บนบอร์ด CP-68HC11	116
ภาคผนวก จ โปรแกรมการทำงานบนไมโครซอฟท์วินโดวส์	195
ภาคผนวก ฉ การใช้งานระบบตรวจจับและควบคุมด้วยเครื่องไมโครคอมพิวเตอร์	235
บรรณานุกรม	

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญตาราง

ตาราง หน้า

ตารางที่ 2.1 ตระกูลของ 68HC11 5

ตารางที่ 2.2 การเลือกโหมดการทำงานของ 68HC11 7

ตารางที่ 2.3 อินเตอร์รัพต์เวคเตอร์ในโหมดการทำงานแบบบูตสแตร็ป 12

ตารางที่ 2.4 สรุปการใช้ขาของ 68HC11 ในโหมดการทำงานต่างๆ 15

ตารางที่ 2.5 รีจิสเตอร์ควบคุมและพอร์ทต่าง ๆ 19

ตารางที่ 2.6 การกำหนดช่องการทำงานและการเก็บข้อมูลของวงจร A/D ใน 68HC11 24

ตารางที่ 2.7 กลุ่มคำสั่งเคลื่อนย้ายข้อมูล 30

ตารางที่ 2.8 กลุ่มคำสั่งเปลี่ยนแปลงและแก้ไขข้อมูล 31

ตารางที่ 2.9 กลุ่มคำสั่งการเลื่อนและหมุนข้อมูล 31

ตารางที่ 2.10 กลุ่มคำสั่งทางคณิตศาสตร์ 32

ตารางที่ 2.11 กลุ่มคำสั่งทางลอจิก 32

ตารางที่ 2.12 กลุ่มคำสั่งการกระโดดแบบไม่มีเงื่อนไข 33

ตารางที่ 2.13 กลุ่มคำสั่งการกระโดดแบบมีเงื่อนไข 33

ตารางที่ 2.14 กลุ่มคำสั่งเปรียบเทียบและตรวจสอบข้อมูล 34

ตารางที่ 2.15 กลุ่มคำสั่งจัดการกับรีจิสเตอร์รหัสเงื่อนไข 34

ตารางที่ 2.16 กลุ่มคำสั่งควบคุม 35

ตารางที่ 2.17 หน้าที่ของสายและความหมายของโวลท์เตจที่กำหนดให้ 44

ตารางที่ 2.18 การกำหนดขาขั้วต่อ RS-232 C 46

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูปภาพ

รูปภาพ หน้า

รูปที่ 2.1 บล็อกไดอะแกรมของ 68HC11F1 3
รูปที่ 2.2 แบบตัวถังและการจัดขา 6
รูปที่ 2.3 วงจรการต่อ 68HC11 เมื่อทำงานในโหมดซิงเกิลชิพ 8
รูปที่ 2.4 วงจรดีมัลติเพล็กซ์บัสแอดเดรสและบัสข้อมูลเมื่อกำหนดการทำงานในโหมดมัลติเพล็กซ์ขยาย 9
รูปที่ 2.5 ตัวอย่างของวงจรการต่อใช้งาน 68HC11 เมื่อต้องการให้ทำงานในโหมดมัลติเพล็กซ์ขยาย 10
รูปที่ 2.6 ไดอะแกรมการทำงานของ 68HC11 ในโหมดบูตสแตร็ป 13
รูปที่ 2.7 การจัดหน่วยความจำในโหมดต่างๆ 16
รูปที่ 2.8 การต่อสัญญาณนาฬิกาให้กับ 68HC11 20
รูปที่ 2.9 การต่อสัญญาณนาฬิกาไมโครคอนโทรลเลอร์ตัวหนึ่งไปยังอีกตัวหนึ่ง 20
รูปที่ 2.10 รูปแบบการจัดบิทและความหมายรีจิสเตอร์รหัสเงื่อนไขของซีพียู 21
รูปที่ 2.11 ไมโครคอนโทรลเลอร์ 68HC11 ที่ทำงานแบบมัลติโปรเซสเซอร์ 26
รูปที่ 2.12 แบบจำลองของวงจรการสื่อสารที่ใช้ RS-232 C 41
รูปที่ 2.13 ขั้วต่อตัวเมียของ RS-232 C 42
รูปที่ 2.14 ระดับแรงดันของสัญญาณ RS-232 C ที่กำหนดให้ใช้งาน 43
รูปที่ 2.15 วงจรอินเตอร์เฟส RS-232 C 44
รูปที่ 2.16 การควบคุมคลื่นพาห์ในระบบฮาล์ฟดูเพล็กซ์ 48
รูปที่ 2.17 การควบคุมคลื่นพาห์ของระบบฟูลดูเพล็กซ์ 50
รูปที่ 2.18 การสลับสถานะแบบที่มีช่องย้อนกลับ 51
รูปที่ 2.19 DTE การควบคุมการตอบรับและหยุดการติดต่อการเรียกทางโทรศัพท์ 52
รูปที่ 2.20 รูปแบบข้อมูลเนื้อความแอสกี้แบบซิงโครนัส 53
รูปที่ 2.21 บล็อกไดอะแกรมของการส่งระบบซิงโครนัส 53
รูปที่ 2.22 การเลือกขา trans clock ของ ซิงโครนัสโมเด็ม 54
รูปที่ 2.23 บล็อกไดอะแกรมของวงจรควบคุมความเข้มของแสงไฟด้วยสัญญาณดิจิตอล 59
รูปที่ 3.1 บล็อกไดอะแกรมของระบบตรวจจับและควบคุมด้วยเครื่องไมโครคอมพิวเตอร์ 60

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปภาพ หน้า

รูปที่ 3.2 วงจรเชื่อมต่อ 68HC11 กับหน่วยความจำภายนอก 64
รูปที่ 3.3 การจัดหน่วยความจำของบอร์ด 65
รูปที่ 3.4 การเลือกจั๊มเปอร์ของแรม 65
รูปที่ 3.5 การเลือกจั๊มเปอร์ของอีพรอม 66
รูปที่ 3.6 แอดเดรสของพอร์ทอินพุทเอาท์พุท 8255 66
รูปที่ 3.7 การเลือกลักษณะของคีย์บอร์ดโดยใช้ J5 67
รูปที่ 3.8 ลักษณะการเชื่อมต่อ คอนเนคเตอร์ 20 ขา เข้ากับ คอนเนคเตอร์ เซนทรอนิกส์ขนาด 36 ขา ของเครื่องพิมพ์ 67
รูปที่ 3.9 การเลือก J6 เพื่อเลือกชนิดของโมดูลแอลซีดี 68
รูปที่ 3.10 การต่อแรงดันอ้างอิงโดย A/D โดย J9 69
รูปที่ 3.11 การเลือกโหมดการทำงานของ 68HC11 69
รูปที่ 3.12 การเลือกจั๊มเปอร์ J10 70
รูปที่ 3.13 การเลือกจั๊มเปอร์ J11 70
รูปที่ 3.14 การเชื่อมต่อ 68HC11 เพื่อส่งข้อมูลอนุกรม (SCI) โดยการต่อพอร์ทอนุกรมกับคอนเนคเตอร์ DB25 71
รูปที่ 3.15 กราฟสำหรับการเลือกค่าความต้านทาน Rz 71
รูปที่ 3.16 จั๊มเปอร์ J7 72
รูปที่ 3.17 การขยายพอร์ทเพิ่มเติมบนชุดควบคุม 72
รูปที่ 3.18 แสดงวงจรนาฬิกาบอกเวลาที่แท้จริง REAL TIME CLOCK 73
รูปที่ 3.19 การต่อวงจรส่วนคีย์บอร์ด 74
รูปที่ 3.20 วงจรเชื่อมต่อกับแอลซีดีขนาด 20 ตัวอักษร x 2 แถว 75
รูปที่ 3.21 วงจรถอดรหัสพอร์ทภายนอก 76
รูปที่ 3.22 วงจรควบคุมความเข้มของแสงไฟด้วยไมโครคอนโทรลเลอร์ 77
รูปที่ 3.23 วงจรโซลิดสเตทรีเลย์ 78
รูปที่ 3.24 วงจรวัดแรงดันไฟตรง 0-500 โวลท์ เปลี่ยนย่านการวัดอัตโนมัติ 79
รูปที่ 3.25 วงจรวัดแรงดันไฟตรง 0-5 โวลท์ 80
รูปที่ 3.26 วงจรรักษาระดับแรงดันไฟตรงในบอร์ดควบคุม 81
รูปที่ 3.27 วงจรรักษาระดับแรงดันไฟตรงในบอร์ดโซลิดสเตทรีเลย์ (SSRAC) และวงจรรักษาระดับแรงดันไฟตรงในบอร์ดวงจรถอดรหัสพอร์ทจากภายนอก 82

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปภาพ หน้า

รูปที่ 3.28 วงจรแปลงไฟสลับเป็นไฟตรง 82
รูปที่ 3.29 ชุดควบคุม (CP-68HC11) 83
รูปที่ 3.30 เมตริกซ์คีย์บอร์ดขนาด 4x4 83
รูปที่ 3.31 การแสดงผลบนจอแอลซีดีขนาด 20 ตัวอักษร 2 แถว 84
รูปที่ 3.32 การ์ดควบคุมความเข้มแสงสว่างของหลอดไฟ 84
รูปที่ 3.33 บอร์ดโซลิดสเตทรีเลย์ 220 โวลท์เอซี 85
รูปที่ 3.34 บอร์ดตรวจจับสัญญาณแรงดันจากภายนอก 85
รูปที่ 3.35 แหล่งจ่ายไฟ 86
รูปที่ 3.36 จุดต่อรับสัญญาณตรวจจับแรงดันจากภายนอก 86
รูปที่ 3.37 จุดต่ออุปกรณ์ไฟฟ้า 220 โวลท์เอซี 87
รูปที่ 3.38 ลักษณะสมบูรณ์ของระบบตรวจจับและควบคุม 87
รูปที่ 3.39 การควบคุมระบบโดยใช้โปรแกรมบนไมโครซอฟท์วินด์โดวส์ 88
รูปที่ 3.40 แสดงการอินเตอร์เฟสกับไมโครคอมพิวเตอร์ผ่านพอร์ท RS-232C 88

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 1
บทนำ

ปัจจุบันเทคโนโลยีทางด้านอิเล็กทรอนิกส์และคอมพิวเตอร์ เป็นส่วนประกอบสำคัญของการประยุกต์สร้างสิ่งอำนวยความสะดวกต่างๆ ในปริญญานิพนธ์นี้จะนำเอาเทคโนโลยีทางอิเล็กทรอนิกส์และคอมพิวเตอร์มาประยุกต์ใช้งาน ในลักษณะเชิงระบบ หรือควบคุมงานด้านอุตสาหกรรม คณะผู้จัดทำเห็นว่า การนำไมโครคอมพิวเตอร์เชื่อมต่อเข้ากับระบบเพื่อควบคุมการทำงาน จะช่วยให้สะดวกต่อการใช้งาน หรือในกรณีที่ผู้ใช้ไม่ต้องการใช้เครื่องไมโครคอมพิวเตอร์ในการควบคุม ผู้ใช้สามารถใช้งานได้จาก คีย์บอร์ดขนาด 4x4 ของระบบในการควบคุมการทำงาน ซึ่งจะแสดงผลทางจอแอลซีดี

เนื่องจากระบบควบคุมงานด้านอุตสาหกรรมนั้น มีการทำงานที่ยุ่งยาก และซับซ้อน ดังนั้น การใช้ไมโครคอนโทรลเลอร์เป็นตัวควบคุมการทำงานต่างๆ ของระบบและตัดสินใจการทำงานในสภาวะต่างๆ ของวงจรอิเล็กทรอนิกส์ จะสามารถทำได้สะดวก มีขนาดเล็กและสามารถเปลี่ยนแปลงการทำงานหรือควบคุมระบบได้โดยการเปลี่ยนแปลงทางซอฟต์แวร์ นอกจากนี้ ยังช่วยลดความยุ่งยากในการออกแบบวงจรอิเล็กทรอนิกส์ ให้มีการเปลี่ยนแปลงการทำงานได้สะดวกยิ่งขึ้น ในปริญญานิพนธ์นี้ ผู้จัดทำเลือกใช้ไมโครคอนโทรลเลอร์เบอร์ 68HC11 ซึ่งเป็นชีพยูที่มีประสิทธิภาพในการทำงานสูง

จากข้อความที่กล่าวมาข้างต้น เราจะต้องมีความเข้าใจการทำงาน และซอฟต์แวร์ของไมโครคอนโทรลเลอร์ เบอร์ 68HC11 เพื่อที่จะนำไปประยุกต์ใช้งานในระบบควบคุม รวมถึงซอฟต์แวร์ของไมโครคอมพิวเตอร์ที่ใช้เชื่อมต่อเข้ากับระบบ โดยจะศึกษาจากรายละเอียดในปริญญานิพนธ์ต่อไป

ปริญญานิพนธ์นี้มีเนื้อหาทั้งหมด 5 บท ดังต่อไปนี้

บทที่ 1 บทนำ กล่าวถึงเนื้อหาของปริญญานิพนธ์ในแต่ละบทพอสังเขป

บทที่ 2 ทฤษฎีและหลักการ นำเสนอเนื้อหาต่างๆ ที่นำมาอ้างอิงและใช้เป็นแนวทางในการออกแบบระบบตรวจจับและควบคุม ทางด้านฮาร์ดแวร์และซอฟต์แวร์

บทที่ 3 การออกแบบและการสร้าง อธิบายถึงขั้นตอนในการออกแบบของวงจรในแต่ละส่วนของของระบบตรวจจับและควบคุม

บทที่ 4 การทดลองและผลการทดลอง ในบทนี้เป็นการทดลองการทำงานต่างๆ ของฮาร์ดแวร์แต่ละวงจร และนำเสนอผลการทดลองต่างๆ ของระบบตรวจจับและควบคุม

บทที่ 5 บทวิจารณ์ สรุป และแนวทางในการพัฒนา ในบทนี้จะกล่าวถึงการสรุปผลการทำงาน มุ่งชี้แนะแนวทางในการแก้ปัญหาพัฒนาระบบให้มีประสิทธิภาพสูงต่อไป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 2
ทฤษฎี และหลักการ

2.1 68HC11ไมโครคอนโทรลเลอร์ขนาด 8 บิท ที่ใช้เทคโนโลยีไฮสปีดซีมอส

ในการออกแบบไมโครคอมพิวเตอร์เพื่อใช้ในงานควบคุม หรือใช้ในงานเฉพาะอย่าง โดยส่วนใหญ่จะใช้ไอซีประเภทไมโครคอนโทรลเลอร์ เพราะจะมีการทำงานที่รวดเร็วและเที่ยงตรงแม่นยำ นอกจากนั้นยังสามารถเปลี่ยนเงื่อนไขของงาน โดยเราเพียงแต่เปลี่ยนซอฟต์แวร์เท่านั้น เหตุผลอีกประการหนึ่งที่นิยมใช้ไอซีไมโครคอนโทรลเลอร์ คือ มีการรวมอุปกรณ์สนับสนุนทางด้านอินพุท เอาท์พุทและอินเตอร์เฟสต่าง ๆ ไว้ในตัวอย่างพร้อมมูล

68HC11 เป็นไมโครคอนโทรลเลอร์อีกตัวหนึ่งของบริษัทโมโตโรล่า ที่ถูกออกแบบมาโดยรวบรวมอุปกรณ์ที่จำเป็นในการใช้งานไว้ภายในตัวของมันเอง ได้แก่ อินพุท,เอาท์พุท, วงจรตั้งเวลา นอกจากนี้ยังมีวงจรนับ วงจรอ่านสัญญาณแบบอนาล็อก ระบบอินเตอร์รัพต์แบบเวคเตอร์อินเตอร์รัพต์ และระบบป้องกันความผิดพลาดที่จะเกิดขึ้นจากโปรแกรม

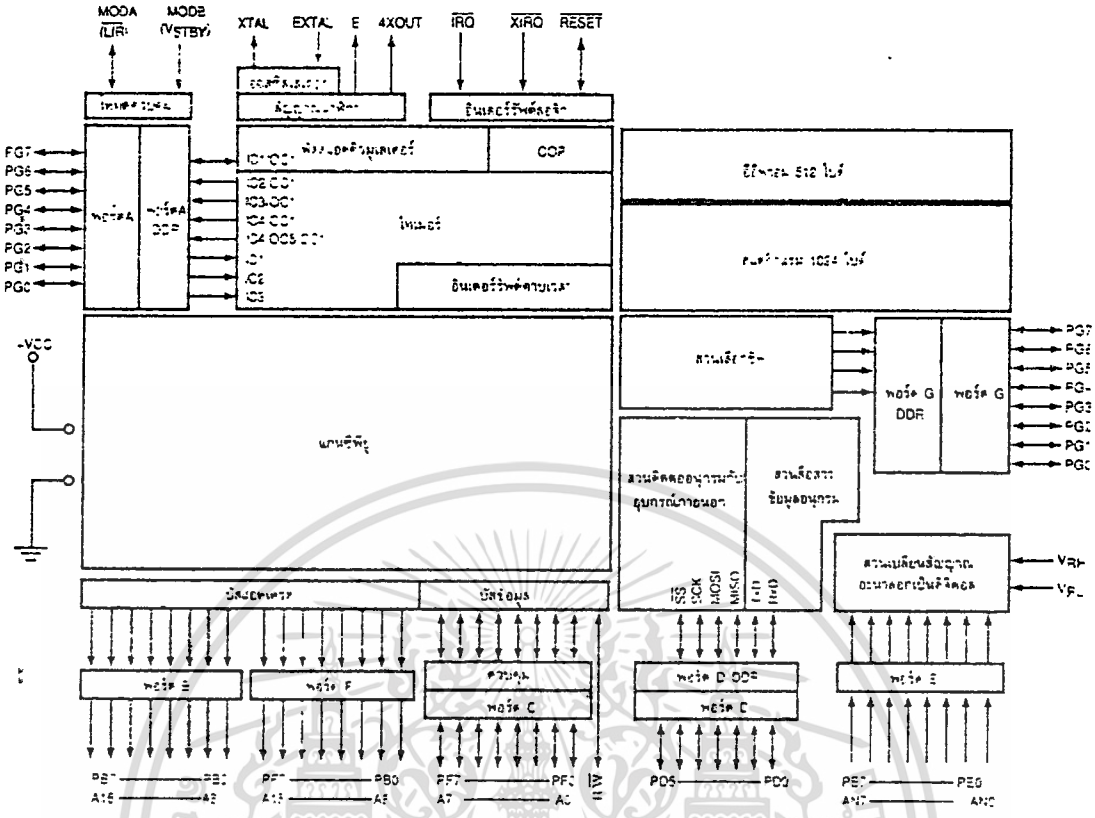
2.1.1 โครงสร้างของ 68HC11

68HC11 เป็นชิพที่บริษัทโมโตโรล่าใช้เทคโนโลยีใหม่ผลิตขึ้นมา เพื่อลดการใช้กำลังไฟฟ้าและเพิ่มความหนาแน่นของจำนวนทรานซิสเตอร์ภายในพื้นที่ชิพสูงกว่าชิพเบอร์ 68000 ของโมโตโรล่าเดิม ไมโครคอนโทรลเลอร์ 68HC11 เป็นแบบ VLSI CMOS จัดอยู่ในกลุ่มซีพียูแบบ 8 บิท รุ่นแรก ๆ ของ 68HC11 พัฒนาในรูปแบบของไมโครคอมพิวเตอร์ชิพเดี่ยว ที่มีจำนวนขาเพียง 40 ขา โดยมีขาที่เป็นอินพุทเอาท์พุท 18 ขา การจัดสัญญาณที่ขายังเป็นระบบมัลติเพล็กซ์เพื่อให้สามารถเชื่อมต่อขยายระบบไปได้อีก โดยจะมีบัสความสารมารถในการอ้างแอดเดรส 64 กิโลไบต์ หรือให้แอดเดรสบัสมีขนาด 16 เส้น บัสข้อมูลมีขนาด 8 เส้น

ซีพียูของ 68HC11 พัฒนามาจาก 6801 จึงใช้รหัสคำสั่งทั้งหมดของ 6801 ได้โดยตรง สัญญาณต่าง ๆ ของระบบก็เหมือนของ MC6801 เมื่อนำ 6801 มาใช้ร่วมกับอุปกรณ์อื่น ๆ ภายในชิพก็ได้มีการพัฒนาให้สามารถใช้คำสั่งใหม่ ๆ เพิ่มขึ้นได้อีก 91 คำสั่ง การทำงานจะทำได้กับความเร็วของสัญญาณนาฬิกาตั้งแต่ 0 คือ ไฟตรงไปจนถึงสูงสุด 4 MHz และในปัจจุบันนี้ยังมีรุ่นที่สามารถทำงานได้สูงกว่านี้อีก โดยมีลักษณะโครงสร้างภายในของไมโครคอนโทรลเลอร์ 68HC11 แสดงได้เป็นบล็อกไดอะแกรมดังรูปที่ 2.1

การที่โมโตโรล่าทำการพัฒนาจาก 6801 โดยมีการเพิ่มอินเด็กซ์รีจิสเตอร์ Y ขนาด 16 บิทเข้าไป ทำให้ได้คำสั่งใหม่เพิ่มขึ้นจากเดิมถึง 65 คำสั่ง และเพื่อให้ได้ลักษณะการ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.1 บล็อกไดอะแกรมของ 68HC11F1

ทำงานที่เป็นไมโครคอนโทรลเลอร์ที่สมบูรณ์นั้น จึงมีการเพิ่มคำสั่งที่กระทำต่อบิท โดยให้มีความสามารถเซ็ตหรือเคลียร์บิทใด ๆ ในหน่วยความจำขนาด 64 กิโลไบต์ที่แอดเดรสใดก็ได้ และยังจะสามารถตรวจสอบบิทใด ๆ ใน 64 กิโลไบต์นี้ เพื่อกำหนดเป็นเงื่อนไขในการกระโดดไปยังตำแหน่งต่าง ๆ

ขีดความสามารถของคำสั่งที่เพิ่มเติมจาก 6801 นั้นยังมีอีกหลายจุด เช่น การเพิ่มคำสั่งหารตัวเลข 16 บิท กับ 16 บิทเพื่อให้ได้ผลลัพธ์เป็นตัวเลขขนาด 16 บิท และเศษเหลือขนาด 16 บิทด้วย คำสั่งเหล่านี้กระทำการหารตัวเลขแบบจำนวนเต็ม คำสั่งหารมีขีดความสามารถพิเศษอยู่ บางคำสั่งใช้สำหรับการหารเลขเศษส่วน โดยตั้งข้อสมมติว่าตัวตั้งมีค่าน้อยกว่าตัวหาร ซึ่งผลลัพธ์ที่ได้จะมีค่าน้อยกว่า 1 ซึ่งนำมาใช้ในลักษณะเลขจุด ซึ่งมีค่าจาก 0.0 ถึง 0.99998 ตัวเลขที่เป็นจุดทศนิยมนี้เหมาะที่จะนำมาใช้กับวงจร A/D และ D/A ซึ่งได้ตัวเลขในรูปแบบที่เป็นเปอร์เซนต์นั่นเอง

คำสั่งใหม่ใน 68HC11 ยังมีคำสั่งเกี่ยวกับการเปรียบเทียบตัวเลขขนาด 16 บิท ที่ใช้สำหรับการทดสอบและแลกเปลี่ยนข้อมูลในอินเด็กซ์รีจิสเตอร์ ซึ่งเป็นสิ่งที่มีบทบาทสำหรับผู้เขียนโปรแกรมควบคุมอุปกรณ์ต่าง ๆ มาก โดยจากสถาปัตยกรรมที่ก้าวหน้าขึ้นทำให้ซีพียู 68HC11 สามารถทำงานได้เร็วกว่า 6801 ถึงเท่าตัวในกรณีที่ใช้สัญญาณนาฬิกาเท่ากัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โดยเราสามารถสรุปคุณสมบัติต่าง ๆ ของ 68HC11 ได้ดังต่อไปนี้

2.1.2 คุณสมบัติของ 68HC11

- เป็นซีพียูขนาด 8 บิท
- มีหน่วยความจำภายในที่ใช้สำหรับเก็บโปรแกรมชนิดรอมขนาดความจุ 4, 8 หรือ 12 กิโลไบต์
- มีหน่วยความจำภายในที่มีไว้สำหรับใช้เก็บโปรแกรมชนิดอีพีรอมขนาด 512 ไบต์ หรือ 2 กิโลไบต์
- มีหน่วยความจำภายในที่มีไว้สำหรับเก็บโปรแกรมชนิดแรมมีขนาด 192,256 หรือ 512 ไบต์
- มีวงจรตั้งเวลา/วงจรนับขนาด 16 บิท
- มีวงจรนับพัลซ์ขนาด 8 บิท
- มีวงจรสำหรับการรับส่งข้อมูลอนุกรมได้สองทิศทาง(Universal Synchronous Receiver Transmitter : USRT)
- ขนาดที่ใช้ในการติดต่อสำหรับวงจรการส่งข้อมูลอนุกรมเป็นแบบมัลติโปรเซสเซอร์
- มีวงจรเปลี่ยนสัญญาณจากอนาล็อกเป็นดิจิตอล 8 ช่อง
- มีวงจรไทม์อินเตอร์รัพต์
- มีระบบวอตช์ด็อกที่ตั้งเวลาได้
- มีวงจรตรวจสอบสัญญาณนาฬิกาให้กับซีพียู
- มีระบบอินเตอร์รัพต์ 2 ระดับจาก 21 แหล่ง
- สามารถติดต่อกับหน่วยความจำภายนอกได้ 64 กิโลไบต์

2.1.3 ตระกูลของ 68HC11

68HC11 เป็นไมโครคอนโทรลเลอร์ที่แบ่งออกเป็นหลายรุ่น โดยแบ่งตามลักษณะของหน่วยความจำที่อยู่ภายในซีพียู ดังตารางที่ 2.1

รูปแบบของชิพตระกูลนี้มีด้วยกัน 3 แบบ คือ

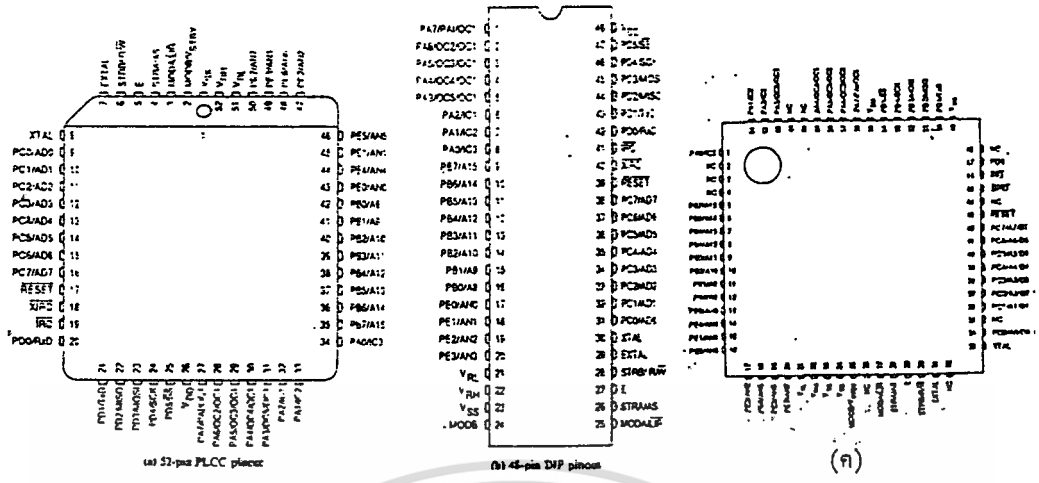
- PLCC (Plastic Leaded Chip Carrier) เป็นตัวถังรูปสี่เหลี่ยม 52 ขา
- DIP (Dual In-line Package) จะมีขนาด 48 ขา ในตัวถังแบบนี้จะไม่มีขาสัญญาณของพอร์ท PE4-PE7
- QFP (Quad Flat Pack) เป็นตัวถังรูปสี่เหลี่ยมขนาด 64 ขา ดังรูปที่ 2.2

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เบอร์ชิพ	รอม	อีอีพรอม	แรม	รีจีสเตอร์ CONFIG	หมายเหตุ
MC68HC11A8	8K	512	256	$0F	เป็นตัวมาตรฐานของชิพนี้
MC68HC11A1	0	512	256	$0D	เหมือนMC68HC11A8แต่ไม่มีรอม
MC68HC11A0	0	0	256	$0C	ไม่มีรอมและอีอีพรอม
XC68HC11B8	8K	512*	256	$0F	เป็นรุ่นทดลองรุ่นแรก
XC68HC11B1	0	512*	256	$0D	เหมือนXC68HC11B8แต่ไม่มีรอม
XC68HC11B0	0	0	256	$0C	ไม่มีรอมและแรม
XC68HC11E9	12K	512	512	$0F	มี 4 แคปเจอร์อินพุท แรมขนาดใหญ่และรอม 12 KByte
XC68HC11E1	0	512	512	$0D	เหมือนXC68HC11E9แต่ไม่มีรอม
XC68HC11E0	0	0	512	$0C	เหมือน XC6811E9 แต่ไม่มีรอมและอีอีพรอม
XC68HC11E2	0	2K**	256	$FF	ไม่สามารถเพิ่มรอมได้
XC68HC11D3	4K	0	192	N/A	เป็นรุ่นประหยัดไม่มี ADC และหน่วยความจำขนาดเล็ก

ตารางที่ 2.1 ตระกูลของ 68HC11

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



(ค)

รูปที่ 2.2 แบบตัวถังและการจัดขา

ก) ตัวถังแบบ PLCC

ข) ตัวถังแบบ DIP

ค) ตัวถังแบบ GFP

2.1.4 โหมดการทำงานของ 68HC11

68HC11 มีโหมดการทำงาน 4 โหมด ดังนี้

1. โหมดชิงเกิลชิพ (Single chip operating mode)
2. โหมดมัลติเพล็กซ์ขยาย (Expanded multiplexed operating mode)
3. โหมดบูตสแตร็ปพิเศษ (Special bootstrap operating mode)
4. โหมดทดสอบพิเศษ (Special test operating mode)

สามารถเลือกโหมดโดยการกำหนดสถานะที่ขา MODA และ MODB ดังตารางที่ 2.2

2.1.4.1 โหมดชิงเกิลชิพ

ในโหมดนี้ 68HC11 จะทำงานโดยลำพังตัวเดียวจากที่ได้กล่าวมาแล้วที่ว่าภายใน 68HC11 มีหน่วยความจำและพอร์ทถึง 5 พอร์ท ดังนั้นเมื่อทำงานโหมดชิงเกิลชิพตัวซีพียูก็จะทำการเรียกข้อมูลที่อยู่ในหน่วยความจำภายในออกมาใช้งาน และใช้พอร์ทที่มีอยู่รับหรือส่งค่าออกไป

ดังนั้นในโหมดนี้ 68HC11 จะไม่ต้องการหน่วยความจำภายนอก หรือชิพพอร์ทอินพุทเอาท์พุทภายนอก เมื่อใช้งานในโหมดนี้จะช่วยลดค่าใช้จ่ายลงได้มาก เพราะไม่ต้องมีชิพอื่น ๆ ภายนอกสนับสนุนเลย และนอกจากนี้ยังมีความเชื่อถือสูง เพราะแทบจะไม่มีจุดต่อ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

MODB	MODA	โหมดการทำงาน
1	0	ซิงเกิลชิพ (Single chip)
1	1	มัลติเพล็กขยาย (Extended multiplexed)
0	0	บูตสแตร็ปพิเศษ (Special bootrap)
0	1	ทดสอบพิเศษ (Special test)

ตารางที่ 2.2 การเลือกโหมดการทำงานของ 68HC11

สัญญาณภายนอกเลย ทำให้โอกาสที่จะเกิดความผิดพลาดลดลง แต่อย่างไรก็ตามในโหมดนี้ 68HC11 จะใช้หน่วยความจำและพอร์ทภายในชิพ ทำให้ไม่สามารถเพิ่มหน่วยความจำหรือพอร์ทเพื่อขยายระบบได้ การต่อวงจรของ 68HC11เมื่อทำงานในโหมดนี้แสดงดังรูปที่ 2.3

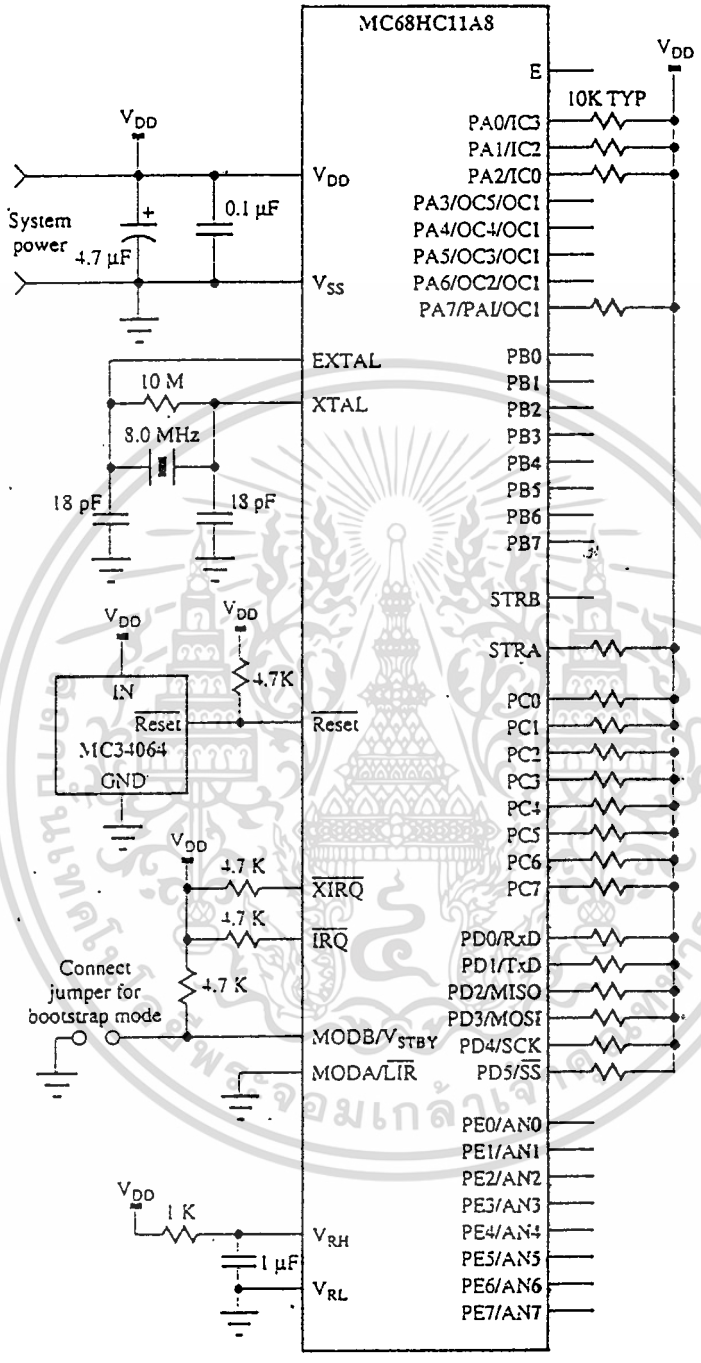
2.1.4.2 โหมดมัลติเพล็กซ์ขยาย

การทำงานของ 68HC11 ในโหมดนี้จะแตกต่างกับโหมดซิงเกิลชิพ โดยโปรแกรมการควบคุมการทำงานของซีพียูจะอยู่ในหน่วยความจำภายนอก โดยที่พอร์ท B และ C ของ 68HC11 จะทำหน้าที่เป็นบัสแอดเดรสและข้อมูล สังเกตได้จากบล็อกไดอะแกรมภายในของ 68HC11 ในรูปที่ 2.4 พอร์ท B จะทำหน้าที่เป็นบัสแอดเดรสตำแหน่งสูง และพอร์ท C ทำหน้าที่เป็นบัสแอดเดรสต่ำและบัสข้อมูล

การที่พอร์ท C จะสามารถเป็นได้ทั้งบัสแอดเดรสและเป็นบัสข้อมูลได้นั้น จะต้องใช้กระบวนการที่เรียกว่า การมัลติเพล็กซ์ (Multiplexed) การแยกสัญญาณแอดเดรสและข้อมูลมาใช้งานจะต้องต่อวงจรเพื่อทำการดีมัลติเพล็กซ์ไว้ที่ภายนอก ดังวงจรในรูปที่ 2.4

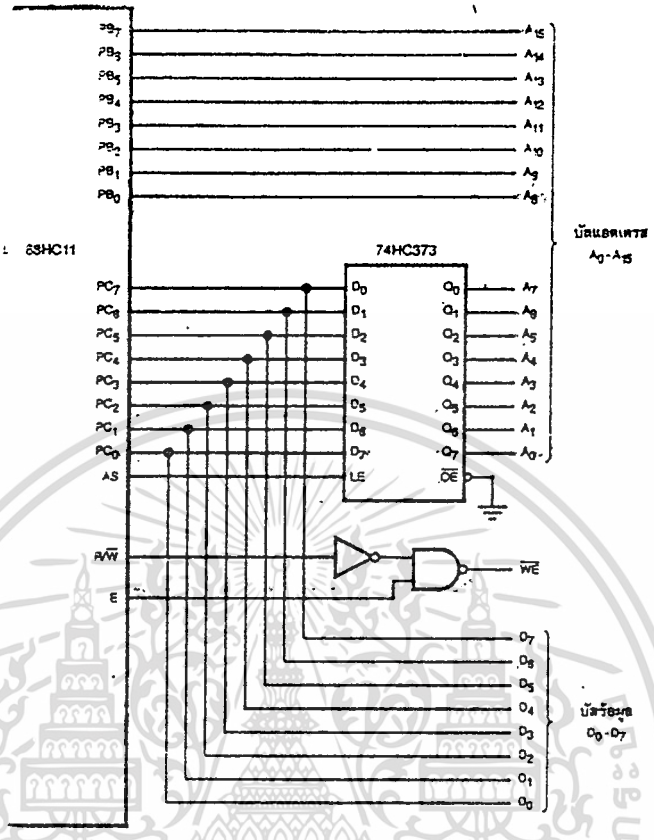
สัญญาณควบคุมที่ใช้ดีมัลติเพล็กซ์สัญญาณแอดเดรสและข้อมูล ได้แก่สัญญาณ STROBE B/R/W และสัญญาณ STROBE A/ADDRESS STROBE (STRA/AS) ในโหมดนี้ชิพ 68HC11 จะสามารถติดต่อกับหน่วยความจำภายนอกเพิ่มได้สูงสุดถึง 64 กิโลไบต์ แต่ก็มีข้อเสีย คือ เราจะต้องต่ออุปกรณ์เพิ่มอีก เช่น หน่วยความจำ ชิพพอร์ทอินพุทเอาท์พุท ทำให้ค่าใช้จ่ายของระบบสูงขึ้น แต่อย่างไรก็ตามในโหมดนี้เป็นโหมดที่นิยมใช้งานมากที่สุด เพราะสามารถต่อขยายหน่วยความจำและพอร์ทได้มากมายนั่นเอง กรณีที่ต้องการต่อ 68HC11 ให้ทำงานในโหมดมัลติเพล็กซ์ขยายแสดงในรูปที่ 2.5

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.3 วงจรการต่อ 68HC11 เมื่อทำงานในโหมดซิงเกิลชิพ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



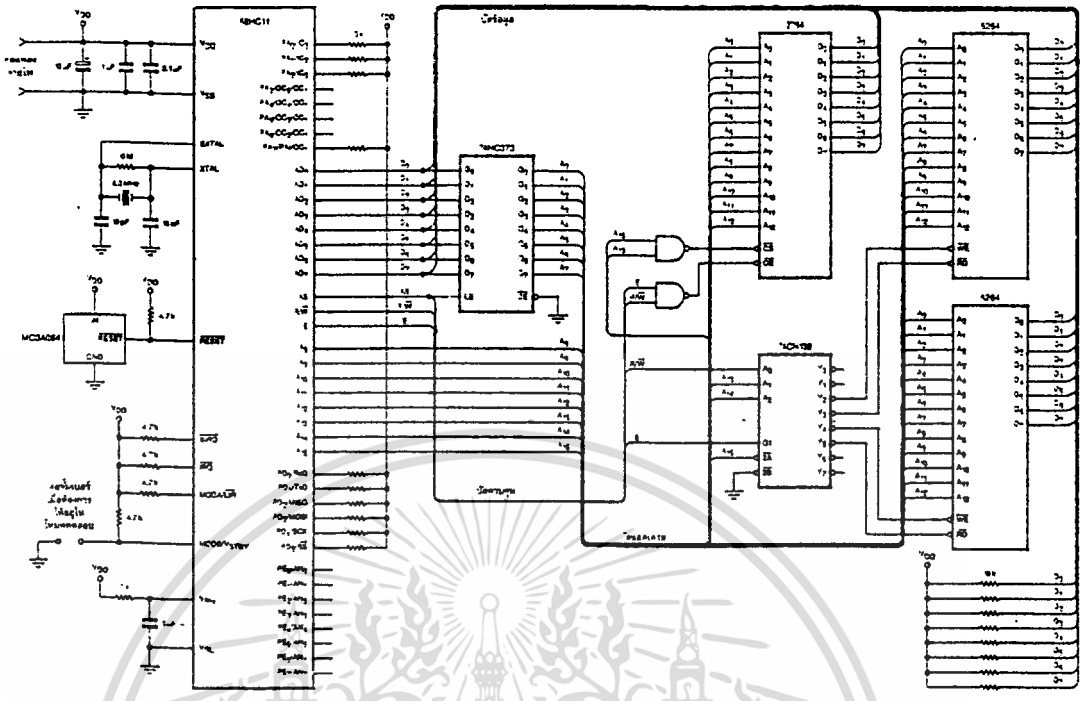
รูปที่ 2.4 วงจรดีมัลติเพล็กซ์บัสแอดเดรสและบัสข้อมูลเมื่อกำหนดการทำงานในโหมดมัลติเพล็กซ์ขยาย

2.1.4.3 โหมดบูตสแตร์ปพิเศษ

เป็นโหมดการทำงานที่คล้ายกับโหมดซิงเกิลชิพ คือตัวไมโครคอนโทรลเลอร์สามารถทำงานได้โดยลำพัง แต่จะมีข้อพิเศษกว่าแบบซิงเกิลชิพอยู่หลายประการ โดยสามารถเขียนโปรแกรมแล้วโหลดเข้าไปในแรมภายในชิพได้เลย โปรแกรมที่ใช้ในการบูตจะถูกเก็บไว้ในบูตสแตร์ปรอมจำนวน 192 ไบต์ รอมนี้จะสามารถเรียกออกมาใช้งานได้ก็ต่อเมื่อ 68HC11 ทำงานในโหมดบูตสแตร์ปนี้เท่านั้น โดยจะโปรแกรมบูตนี้เก็บไว้ที่แอดเดรส $BF40-$BFFF เมื่อเรียกใช้งานโปรแกรมบูต ส่วน SCI (Serial Communication Interface) จะส่งข้อมูลโปรแกรมไปยังแรมภายในชิพที่แอดเดรส $0000 - $00FF หลังจากส่งข้อมูลที่แอดเดรส $00FF เรียบร้อยแล้วซีพียูจะเริ่มทำงานที่แอดเดรส $0000 ต่อไป

ในโหมดการทำงานแบบนี้ 68HC11 จะส่งผ่านข้อมูลผ่านพอร์ท SCI โดยหลังจากที่ทำการเซ็ตซีพียูแล้ว ส่วน SCI จะทำงานที่สัญญาณนาฬิกา E/16 (หรือมีอัตราบอดเท่ากับ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.5 ตัวอย่างของวงจรการต่อใช้งาน 68HC11 เมื่อต้องการให้ทำงานในโหมดมัลติเพล็กซ์ขยาย

7,812 สำหรับสัญญาณนาฬิกาที่มีความถี่ 2 MHz) โหมดนี้ยังมีลักษณะการทำงานที่ค่อนข้างพิเศษอีกประการหนึ่งคือ การป้องกันการคัดลอก โดยจะมีบิทป้องกัน (Security bit) เข้ามาเกี่ยวข้อง ถ้าหากบิทป้องกันนี้ถูกเซ็ตเป็น "1" ที่เอาท์พุทของส่วน SCI จะส่งค่า $FF ออกไป จะทำให้ข้อมูลในอีอีพรอมถูกลบ และจะทำเช่นนี้จนกว่าจะทำการลบข้อมูลในอีอีพรอมภายในชิพหมด แต่ในทางตรงข้ามถ้าไม่มีการเซ็ตบิทป้องกันข้อมูลที่เป็นการหยุดการทำงาน วงจรชิพจะถูกส่งออกไปภายนอกโดยจะผ่านส่วน SCI

ผู้ใช้งานต้องทำการดาวน์โหลดข้อมูลโปรแกรม 256 ไบต์ ไปยังแรมภายในโดยให้เริ่มทำงานที่แอดเดรส $0000 ถ้าทุกอย่างเรียบร้อยก็จะไปเริ่มทำงานที่ตำแหน่ง $0000 เพื่อเตรียมที่จะทำงาน

อินเตอร์รัพต์เวคเตอร์ของ 68HC11 เมื่อถูกใช้งานในโหมดนี้ จะสามารถเข้าถึงหน่วยความจำแรมได้เร็วขึ้น ซึ่งในการใช้อินเตอร์รัพต์จะทำให้ผู้ใช้งานสามารถกระโดดไปทำงานที่ส่วนอื่นได้ อินเตอร์รัพต์เวคเตอร์ที่มีใน 68HC11 สรุปได้ดังตารางที่ 2.3

ถ้าหากต้องการใช้งานอินเตอร์รัพต์ SWI ต้องระบุคำสั่งกระโดดเพื่อให้ซีพียูสามารถเรียกใช้งานได้ถูกลงในแรมที่ตำแหน่ง $00F4,$00F5 และ $00F6 เมื่อเกิดอินเตอร์รัพต์ในแรมและที่นั่นจะมีคำสั่ง JUMP เพื่อให้ซีพียูสามารถตอบสนองการอินเตอร์รัพต์ได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

แบบ SWI ขึ้น ค่าเวคเตอร์นี้เป็นตัวบอกให้ซีพียูทำการอ่านข้อมูลเพื่อประมวลผลที่ $00F4

ลักษณะพิเศษอีกอย่างหนึ่งที่ไม่ควรมองข้าม คือ สามารถที่จะเข้าไปทำงานในรอมที่แอดเดรสต่ำแหน่ง $0000 ได้โดยไม่ต้องทำการดาวน์โหลดก่อน โดยวิธีการนี้จะใช้ได้กับเฉพาะเมื่อเราใช้สัญญาณนาฬิกา E/16 เริ่มด้วยการส่ง $55 ไปแทนค่า $FF ด้วยคำสั่งนี้จะทำให้ซีพียูกระโดดไปทำงานที่แอดเดรส $0000 โดยที่ไม่ต้องผ่านการกราวด์โหลดได้ ลักษณะการทำงานโหมดนี้แสดงในรูปที่ 2.6 แต่เนื่องจากในเบอร์ MC68HC11A1 ไม่มีรอมจึงไม่สามารถใช้งานในโหมดนี้ได้ สมบูรณ์ตามฟังก์ชั่นที่ระบุไว้ข้างต้น

2.1.4.4 โหมดทดสอบพิเศษ

เป็นโหมดการทำงานที่โรงงานผู้ผลิตเป็นผู้กำหนด ซึ่งการทำงานในโหมดนี้จะคล้ายกับโหมดมัลติเพล็กซ์ขยาย ซึ่งในการเซ็ตและอินเตอร์รัพต์เวคเตอร์จะได้รับการเฟตช์จากหน่วยความจำภายนอกที่แอดเดรส $BFC0-$BFFF มากกว่าที่แอดเดรส $FFC0-$FFFF รีจิสเตอร์ TEST1 จะได้รับการอีนาเบิล เพื่อเป็นการบ่งบอกให้ทราบว่าขณะนี้ซีพียูพร้อมรับการตรวจสอบแล้ว โหมดการทำงานนี้เป็นโหมดที่ผู้ใช้งานปกติไม่ควรที่จะเลือกใช้เพราะว่าจะทำให้ระบบป้องกันข้อมูลภายในชิพด้อยลงไป

2.1.5 ลักษณะของสัญญาณและการจัดขา

RESET จะเป็นขาที่ทำงานสองทิศทางคือ เป็นอินพุทสำหรับการทำให้ซีพียูเริ่มต้นการทำงาน และเป็นเอาท์พุทแสดงถึงการทำงานภายในผิดพลาด ซึ่งจะเกิดขึ้นได้ 3 กรณี คือ ความผิดพลาดจากสัญญาณนาฬิกา, ความผิดพลาดที่จากโปรแกรมในการใช้ระบบว็อตช์ด็อก และการเอ็กซ์คิวต์คำสั่งผิดพลาด

XTAL, EXTAL เป็นขาอินพุทสำหรับต่อคริสตอลเพื่อผลิตสัญญาณนาฬิกาให้กับซีพียูหรืออาจจะใช้สัญญาณนาฬิกาจากภายนอกป้อนเข้าที่ขา EXTAL โดยตรงก็ได้ โดยที่มีตัวต้านทาน 10-100 กิโลโอห์ม ต่อลงกราวด์เพื่อป้องกันสัญญาณรบกวน

E เป็นเอาท์พุท โดยจะมีความถี่สัญญาณนาฬิกาน้อยกว่าความถี่ของสัญญาณนาฬิกาที่ขา EXTAL และขา XTAL 4 เท่า และยังเป็นขาที่แสดงการทำงานของซีพียูด้วย คือ ถ้ามีสัญญาณเป็นลอจิก "0" แสดงว่าซีพียูกำลังทำการประมวลผลภายในและหากเป็นลอจิก "1" หมายถึงขณะนี้ซีพียูทำกับข้อมูล เมื่ออยู่ใน STOP MODE จะไม่มีสัญญาณ E ออกมา

IRQ เป็นขาอินพุททำงานที่ลอจิก "0" หรือขอบขาลง เป็นการขออินเตอร์รัพต์ซีพียู โดยที่ขานี้ต้องต่อตัวต้านทานจากภายนอกมีค่า 4.7 กิโลโอห์ม กับไฟเลี้ยง +5 โวลท์ เพื่อทำให้สภาวะปกติมีค่าเป็นลอจิก "1"

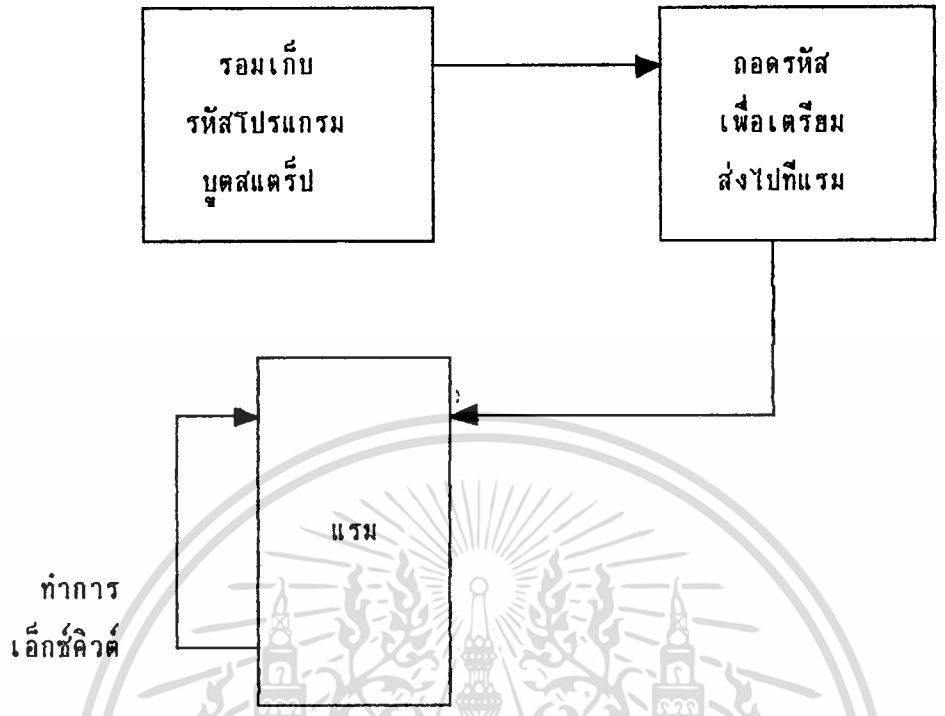
เอกสารนี้ใช้เพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

แอดเดรส	เวคเตอร์
$00C4	SCI
$00C7	SPI
$00CA	Pulse Accumulator Input Edge
$00CD	Pulse Accumulator Overflow
$00D0	Timer Overflow
$00D3	Timer Output compare 5
$00D6	Timer Output compare 4
$00D9	Timer Output compare 3
$00DC	Timer Output compare 2
$00DF	Timer Output compare 1
$00E2	Timer Input compare 3
$00E5	Timer Input compare 2
$00E8	Timer Input compare 1
$00EB	Real Time Interrupt
$00EE	IRQ
$00F1	XIRQ
$00F4	SWI
$00F7	Illegal Opcode
$00FA	COP Fail
$00FD	Clock Monitor
$BF40	Reset

ตารางที่ 2.3 อินเตอร์รัพต์เวคเตอร์ในโหมดการทำงานแบบบูตสแตร็ป

XIRQ เป็นขาอินพุทจะมีระดับการทำงานที่ลอจิก "0" ซึ่งจะเป็นการขออินเตอร์รัพต์แบบนอนมาสเคเบิลอินเตอร์รัพต์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.6 ไดอะแกรมการทำงานของ 68HC11 ในโหมดบูตสแตร็ป

MODA, MODB เป็นขาที่ใช้เลือกโหมดการทำงานของ CPU ซึ่งแสดงการเลือกโหมดดังตารางที่ 2.2 หลังจากที่ซีพียูเลือกโหมดการทำงานแล้ว ขา MODA จะเป็นขาเอาท์พุททำงานที่ลอจิก "0"เพื่อแสดงถึงการเฟตช์ของซีพียูที่จะใช้ในการทำให้ซีพียูทำงานทีละคำสั่ง และขา MODB เป็นขาอินพุทสำหรับป้อนไฟไปเลี้ยงหน่วยความจำภายใน เพื่อป้องกันไม่ให้ข้อมูลสูญหายหลังจากที่ไม่มีไฟเลี้ยงซีพียู

VRL, VRH เป็นขาอินพุทสำหรับป้อนแรงดันอ้างอิงในวงจรแปลงสัญญาณอนาล็อกเป็นดิจิตอล

STROB/R/W เป็นขาเอาท์พุทที่แสดงผลการทำงานในโหมด SINGLE CHIP ขานี้เป็นเอาท์พุทแสดงการแฮนด์เชคข้อมูล และเมื่อทำงานโหมด EXPANDED MULTIPLEXED ขานี้จะเป็นขาเอาท์พุทแสดงการอ่านเขียนข้อมูลของซีพียู โดยถ้าเป็นลอจิก "0" แสดงว่าเป็นการเขียนข้อมูล

STRA/AS เป็นทั้งขาอินพุทและเอาท์พุท ถ้าทำงานในโหมด SINGLE CHIP ขานี้จะเป็นขาอินพุทแสดงการแฮนด์เชคข้อมูล และเมื่อทำงานในโหมด EXPLANDED MULTIPLEXED จะเป็นการแยกสัญญาณระหว่างแอดเดรส 8 บิทล่างกับสัญญาณข้อมูล

PORT A หน้าที่ของพอร์ทนี้จะมีหลายแบบด้วยกัน ได้แก่ เป็นขาอินพุทเอาท์พุทปกติ,

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เป็นขาสัญญาณในระบบตัวตั้งเวลาหลักและแอกคิวมูเลเตอร์ เป็นต้น ที่พอร์ท A จะมีอยู่ 3 ขาสัญญาณเท่านั้นที่จะถูกกำหนดให้เป็นขาอินพุทได้ ส่วนที่ถูกกำหนดให้เป็นขาเอาท์พุทจะมีอยู่ 4 ขา ที่เหลืออีก 1 ขาเป็นขาสัญญาณที่เป็นได้ทั้งอินพุทและเอาท์พุท ขาที่ว่านั้นคือ ขา PA7 ทิศทางการส่งผ่านข้อมูลของ PA7 จะถูกควบคุมโดยข้อมูลในบิทที่ 7 ของรีจิสเตอร์ควบคุมพัลซ์แอกคิวมูเลเตอร์ (PACTL) โดยบิทที่ 7 นั้นมีชื่อว่า DDRA7 ข้อมูลที่ต้องการอ่านและเขียนของพอร์ท A จะถูกเก็บไว้ที่รีจิสเตอร์ PORTA

PORT B ถูกกำหนดให้เป็นพอร์ทเอาท์พุทเมื่อทำงานในโหมดชิงเกิลชิพและบูตสแตร็ปพิเศษ ข้อมูลที่ต้องการส่งออกมาที่พอร์ทนี้ ต้องเขียนเข้าไปเก็บไว้ที่รีจิสเตอร์ PROTB และจะส่งออกไปได้ก็ต่อเมื่อมีสัญญาณ STRB เกิดขึ้น นั่นคือในการส่งข้อมูลพอร์ท B นั้นจะมีการแฮนด์เชคระหว่างอุปกรณ์ที่ต่ออยู่กับพอร์ทและตัวไมโครคอนโทรลเลอร์ แต่ถ้าหาก 68HC11 ทำงานในโหมดมัลติเพล็กซ์ขยายหรือโหมดทดสอบพิเศษ พอร์ท B ทำหน้าที่เป็นบัสแอดเดรสในไบต์สูง (A8-A15) เพื่อติดต่อกับหน่วยความจำภายนอกชิพ

PORT C เมื่อทำงานในโหมดชิงเกิลชิพและโหมดบูตสแตร็ปพิเศษ พอร์ท C จะเป็นพอร์ทอินพุทเอาท์พุท สามารถส่งผ่านข้อมูลได้ 2 ทิศทาง โดยแต่ละขาของพอร์ท (หรือจะเรียกว่าแต่ละบิทก็ได้) จะเป็นอิสระต่อกันในด้านการกำหนดทิศทาง ในการกำหนดทิศทางของขาพอร์ท C แต่ละขานั้นเราสามารถกำหนดได้ที่รีจิสเตอร์ควบคุมทิศทางข้อมูลพอร์ท C หรือที่เรียกว่า (Data Direction for Port C: DDRC) ถ้าบิทใดในรีจิสเตอร์ DDRC ถูกกำหนดให้เป็น "1" ขาของพอร์ท C ที่ตำแหน่งเดียวกันนั้นจะเป็นขาพอร์ทเอาท์พุท และถ้าต้องการให้เป็นอินพุทก็เขียนข้อมูล "0" ไปยังบิทที่ต้องการในรีจิสเตอร์ DDRC ถ้าหาก 68HC11 ทำงานในโหมดมัลติเพล็กซ์ขยายหรือที่โหมดทดสอบพิเศษพอร์ท C จะถูกปรับให้ทำหน้าที่เป็นขาสัญญาณแอดเดรสไบต์ต่ำ (A0 - A7) และขาบัสข้อมูล (D0-D7) โดยจะสามารถแยกสัญญาณทั้งสองนี้ด้วยการมัลติเพล็กซ์

PORT D เป็นพอร์ทอินพุทเอาท์พุทขนาด 6 บิท PD0 และ PD1 จะถูกกำหนดให้ทำหน้าที่เป็นขารับส่งข้อมูลอนุกรมเมื่อให้ 68HC11 ทำการสื่อสารข้อมูลอนุกรม (SCI) ส่วนอีก 4 ขาที่เหลือจะนำมาใช้ในการเชื่อมต่ออุปกรณ์อนุกรม (SPI) การกำหนดทิศทางการส่งผ่านข้อมูลของพอร์ท ทำได้โดยการเขียนข้อมูลไปรีจิสเตอร์ควบคุมทิศทางข้อมูลพอร์ท D (DDRD) ส่วนข้อมูลที่ต้องการจะส่งเข้ามาหรือส่งออกไปทางพอร์ท D ให้เขียนมาเก็บไว้ที่รีจิสเตอร์ PORTD

PORT E ถูกกำหนดให้เป็นพอร์ทอินพุทขนาด 8 บิท และใช้เป็นพอร์ทอินพุทของวงจรแปลงสัญญาณอะนาล็อกเป็นสัญญาณดิจิตอล (ถ้าเป็นรุ่น 48 ขา พอร์ท E จะมีเพียง 4 ขา)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ก674ร 2537
ห้องสมุด
คณะครุศาสตร์อุตสาหกรรม

Port-Bit	Single-Chip & Bootstrap Mode	Expanded Multiplexed and Special Test Mode
A-0	PA0 IC3	PA0 IC3
A-1	PA1 IC2	PA1 IC2
A-2	PA2 IC1	PA2 IC1
A-3	PA3 OC5 and-or OC1	PA3 OC5 and-or OC1
A-4	PA4 OC4 and-or OC1	PA4 OC4 and-or OC1
A-5	PA5 OC3 and-or OC1	PA5 OC3 and-or OC1
A-6	PA6 OC2 and-or OC1	PA6 OC2 and-or OC1
A-7	PA7 PAI and-or OC1	PA7 PAI and-or OC1
B-0	PB0	A8
B-1	PB1	A9
B-2	PB2	A10
B-3	PB3	A11
B-4	PB4	A12
B-5	PB5	A13
B-6	PB6	A14
B-7	PB7	A15
C-0	PC0	A0 D0
C-1	PC1	A1 D1
C-2	PC2	A2 D2
C-3	PC3	A3 D3
C-4	PC4	A4 D4
C-5	PC5	A5 D5
C-6	PC6	A6 D6
C-7	PC7	A7 D7
D-0	PD0 RxD	PD0 RxD
D-1	PD1 TxD	PD1 TxD
D-2	PD2 MISO	PD2 MISO
D-3	PD3 MOSI	PD3 MOSI
D-4	PD4 SCK	PD4 SCK
D-5	PD5 SS	PD5 SS
	STRA	AS
	STRB	R W
E-0	PE0 AN0	PE0 AN0
E-1	PE1 AN1	PE1 AN1
E-2	PE2 AN2	PE2 AN2
E-3	PE3 AN3	PE3 AN3
E-4	PE4 AN4 **	PE4 AN4 **
E-5	PE5 AN5 **	PE5 AN5 **
E-6	PE6 AN6 **	PE6 AN6 **
E-7	PE7 AN7 **	PE7 AN7 **

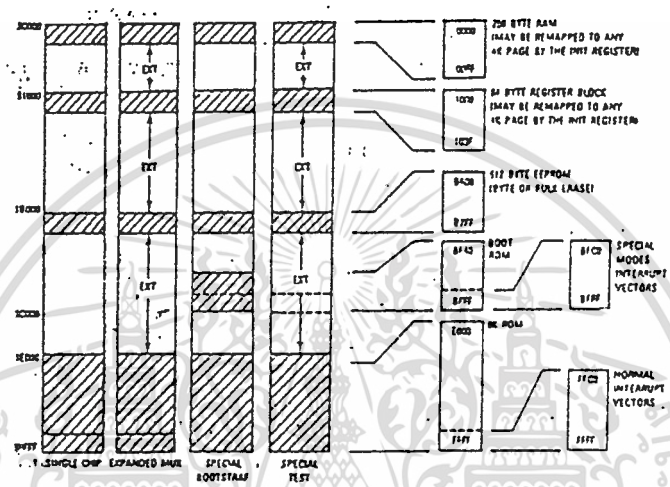
ตารางที่ 2.4 สรุปการใช้ขาของ 68HC11 ในโหมดการทำงานต่างๆ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1282

2.1.6 การจัดหน่วยความจำ

68HC11 เป็นซีพียูที่มีการจัดหน่วยความจำได้อย่างมีประสิทธิภาพ โดยสามารถติดต่อกับหน่วยความจำได้ถึง 64 กิโลไบต์ รวมทั้งหน่วยความจำที่อยู่ภายในตัวซีพียูด้วย การจัดหน่วยความจำแบ่งออกได้เป็น 4 แบบตามโหมดการทำงาน ดังรูปที่ 2.7



รูปที่ 2.7 การจัดหน่วยความจำในโหมดต่าง ๆ

พื้นที่หน่วยความจำแอดเดรส 0000H - 00FFH เป็นพื้นที่ของหน่วยความจำแรมที่อยู่ภายในชิพของทุกโหมดการทำงาน แอดเดรส 0100H - 0FFFH จะเป็นพื้นที่หน่วยความจำภายนอกในโหมดการทำงาน EXPANDED MUX กับ SPECIAL TEST ที่ตำแหน่งแอดเดรส 1000H-103FH จะเป็นพื้นที่ของรีจิสเตอร์ภายในชิพของทุกโหมดการทำงาน โดยแอดเดรส 1040H-B5FFH เป็นพื้นที่หน่วยความจำภายนอกใน EXPANDED MUX กับ SPECIAL TEST ถัดมาที่แอดเดรส B600H-B7FFH เป็นพื้นที่ของหน่วยความจำอีอีพรอมภายในชิพ แอดเดรส BF40H-BFFFH เป็นพื้นที่บูตรอมในโหมด SPECIAL BOOT STRAP ส่วนตำแหน่งแอดเดรส B800H-FFFFH เป็นพื้นที่สำหรับหน่วยความจำภายนอกและที่แอดเดรส FFC0H ถึง FFFFH เป็นพื้นที่สำหรับเก็บเวคเตอร์การอินเตอร์รัพต์ของอินเตอร์รัพต์ทุกประเภท

2.1.7 หน่วยความจำแรมภายในซีพียู

หน่วยความจำแรมภายในซีพียูแบ่งออกเป็น 2 ประเภท คือ

1. แรมที่ใช้ในการอ่านเขียนข้อมูลและรีจิสเตอร์ โดยตั้งแต่ที่แอดเดรส 0000H ถึง 00FFH (256 ไบต์) เป็นหน่วยความจำที่ใช้ในการเขียนอ่านข้อมูลทั่วไป ซึ่งหน่วยความจำ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บริเวณนี้สามารถเก็บข้อมูลได้แม้ไฟเลี้ยงของวงจรจะหายไป ข้อมูลแรมในส่วนนี้ในสภาวะปกติจะได้ไฟเลี้ยงจาก VDD แต่เมื่อ VDD หายไปจะได้ไฟเลี้ยงจากอีกทางหนึ่ง โดยก่อนที่ VDD จะหายไปต้องมีสัญญาณรีเซ็ต เพื่อให้ซีพียูหยุดการทำงานเพื่อป้องกันไม่ให้มีการเขียนข้อมูลในหน่วยความจำ และเมื่อ VDD เข้ามาก็จะเข้าไปจ่ายให้กับแรมส่วนนี้ตามเดิมทันที โดยที่ข้อมูลที่อยู่ในแรมส่วนนี้จะไม่มีการเปลี่ยนแปลง กระแสไฟฟ้าที่ใช้เลี้ยงแรมส่วนนี้จะใช้กระแสไฟน้อยมากจึงสามารถใช้แบตเตอร์รี่เล็ก ๆ เป็นไฟเลี้ยงให้กับแรมนี้ได้

2. แรมอีกส่วนที่ใช้เป็นรีจิสเตอร์จะถูกใช้ในการควบคุมหารทำงานต่าง ๆ ของซีพียู โดยมีหน้าที่แตกต่างกันออกไปดังนี้

DDRD รีจิสเตอร์ควบคุมทิศทางการติดต่อของพอร์ท D

PORTE รีจิสเตอร์พอร์ท E ใช้สำหรับอ่านข้อมูลของพอร์ท E

CFORC รีจิสเตอร์บอกผลของการเปรียบเทียบของไทม์เมอร์ต่าง ๆ

OC1M รีจิสเตอร์ที่ใช้กำหนดบิทของข้อมูลในพอร์ท A ที่มีการเปรียบเทียบกับ OC1

OC1D รีจิสเตอร์สำหรับเก็บข้อมูลของการเปรียบเทียบระหว่างพอร์ท A กับ OC1

TOC1M รีจิสเตอร์เซ็ตผลการเปรียบเทียบบิทต่อบิทระหว่างพอร์ท A กับ TOC1

TCNT รีจิสเตอร์ 16 บิทใช้เป็นตัวฟรีเคาน์เตอร์ของระบบไทม์เมอร์ โดยจะทำการนับตั้งแต่ 0000H - FFFFH แล้วกลับมาเริ่มต้นใหม่

TIC1 รีจิสเตอร์ขนาด 16 บิทเป็นอินพุทของไทม์เมอร์ 1

TIC2 รีจิสเตอร์ขนาด 16 บิทเป็นอินพุทของไทม์เมอร์ 2

TIC3 รีจิสเตอร์ขนาด 16 บิทเป็นอินพุทของไทม์เมอร์ 3

TOC1 รีจิสเตอร์ขนาด 16 บิท ใช้ในการเปรียบเทียบกับไทม์เมอร์เคาน์เตอร์ 1

TOC2 รีจิสเตอร์ขนาด 16 บิท ใช้ในการเปรียบเทียบกับไทม์เมอร์เคาน์เตอร์ 2

TOC3 รีจิสเตอร์ขนาด 16 บิท ใช้ในการเปรียบเทียบกับไทม์เมอร์เคาน์เตอร์ 3

TOC4 รีจิสเตอร์ขนาด 16 บิท ใช้ในการเปรียบเทียบกับไทม์เมอร์เคาน์เตอร์ 4

TOC5 รีจิสเตอร์ขนาด 16 บิท ใช้ในการเปรียบเทียบกับไทม์เมอร์เคาน์เตอร์ 5

TCTL1 เป็นรีจิสเตอร์ที่ใช้ในการเลือกลักษณะเอาท์พุทจากการใช้ฟังก์ชันการเปรียบเทียบเอาท์พุท

TCTL2 เป็นรีจิสเตอร์ที่ใช้ในการเลือกลักษณะอินพุท เพื่อใช้ฟังก์ชันอินพุทแคปเจอร์

TMSK1 เป็นรีจิสเตอร์ที่ใช้สำหรับอีนาเบิล หรือดิสเอเบิลอินเตอร์รัพต์ของตัวเปรียบเทียบเอาท์พุทกับตัวเปรียบเทียบอินพุท

TFLG1 เป็นรีจิสเตอร์ที่แสดงถึงการเท่ากันของไทม์เมอร์เคาท์เตอร์ กับ ตัวเปรียบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เทียบเอาท์พุท กับ ตัวเปรียบเทียบอินพุท

TMSK2 เป็นรีจิสเตอร์ที่ใช้อีนาเบิลหรือดิสเอเบิลการอินเตอร์รัพต์ของเรียลไทม์และเป็นตัวกำหนดตัวหารของเวลาเรียลไทม์

TFLG2 เป็นรีจิสเตอร์ใช้แสดงการทำงานต่าง ๆเป็นไทม์อินเตอร์รัพต์โอเวอร์โฟลว์

ACTL เป็นรีจิสเตอร์ที่ใช้ในการควบคุมพัลซ์แอกคิวมูเลเตอร์

PACNT เป็นรีจิสเตอร์เคาน์เตอร์ของพัลซ์แอกคิวมูเลเตอร์

AFCR เป็นรีจิสเตอร์ที่ใช้ในการควบคุมการทำงานแบบ SPI (Serial Peripheral Interface)

SPSR เป็นรีจิสเตอร์ที่ใช้ในการบอกลักษณะการติดต่อแบบ SPI

BACD เป็นรีจิสเตอร์ที่ใช้กำหนดความเร็วในการส่งข้อมูลแบบ SCI (Serial - Communication Interface)

SCCR1 เป็นรีจิสเตอร์ที่ใช้ในการกำหนดรูปแบบในการส่งข้อมูลแบบ SCIS, CCR2 เป็นรีจิสเตอร์ที่ใช้ในการควบคุมการอินเตอร์รัพต์

SCCR เป็นรีจิสเตอร์ที่ใช้ในการบ่งสถานะการรับส่งข้อมูลแบบ SCI

SCDR เป็นรีจิสเตอร์ที่ใช้ในการเก็บข้อมูลจากการรับหรือส่งข้อมูลแบบ SCI

ADCT1 เป็นรีจิสเตอร์ที่ใช้ในการเลือกการอ่านข้อมูลแบบอะนาล็อก และแสดงสถานะการอ่านข้อมูล

ADR1 เป็นรีจิสเตอร์ที่ใช้ในการเก็บข้อมูลที่ได้จากวงจร A/D 1

ADR2 เป็นรีจิสเตอร์ที่ใช้ในการเก็บข้อมูลที่ได้จากวงจร A/D 2

ADR3 เป็นรีจิสเตอร์ที่ใช้ในการเก็บข้อมูลที่ได้จากวงจร A/D 3

ADR4 เป็นรีจิสเตอร์ที่ใช้ในการเก็บข้อมูลที่ได้จากวงจร A/D 4

OPTION เป็นรีจิเตอร์ที่ใช้ในการกำหนดแบบของการใช้ระบบ COP (Computer Operating Próperly) การเปิดสัญญาณนาฬิกาสำหรับการแซมปลิ้งในวงจร A/D

COPRST การเปิดสัญญาณนาฬิกาเลือกรีเซ็ตค่าเวลาในระบบ COP

PPROG เป็นรีจิสเตอร์ที่ใช้ในการกำหนดรูปแบบของการติดต่อกับอีพรอม

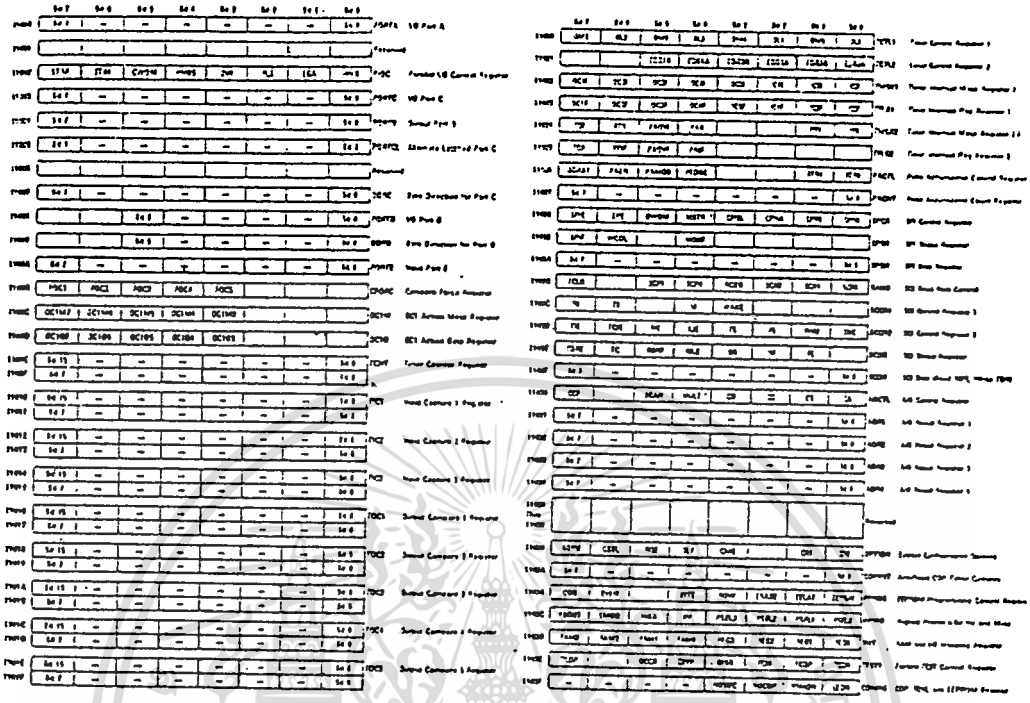
HPIRO เป็นรีจิสเตอร์ที่ใช้ในการจัดลำดับการอินเตอร์รัพต์

INIT เป็นรีจิสเตอร์ที่ใช้ในการกำหนดตำแหน่งของแรมและรีจิสเตอร์

TEST1 เป็นรีจิสเตอร์ที่ใช้ในการทดสอบของโรงงานผู้ผลิต

CONFIG เป็นรีจิสเตอร์ที่ใช้ในการบ่งบอกอุปกรณ์ภายในตัวซีพียู

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



ตารางที่ 2.5 รีจิสเตอร์ควบคุมและพอร์ทต่าง ๆ

2.1.8 วงจรกำเนิดสัญญาณนาฬิกา

68HC11 สามารถต่อสัญญาณนาฬิกาได้สองรูปแบบคือ

1. ใช้วงจรกำเนิดสัญญาณนาฬิกาภายใน โดยการต่อคริสตอลภายนอกกับตัวต้านทานและตัวเก็บประจุอีกสองตัว ดังรูปที่ 2.8 (ก)

2. การต่อสัญญาณนาฬิกาจากภายนอกเข้าที่ขา EXTAL และต่อตัวต้านทานเข้าที่ขา XTAL ลงกราวด์ ดังรูปที่ 2.8 (ข) นอกจากนั้นสัญญาณนาฬิกาจากการที่ต่อ EXTAL จากไมโครคอนโทรลเลอร์ตัวที่หนึ่งยังสามารถใช้เป็นสัญญาณนาฬิกาให้กับไมโครคอนโทรลเลอร์ตัวต่อไปได้อีก ดังรูปที่ 2.9 โดยสัญญาณนาฬิกาที่ให้กับซีพียูต้องมีความถี่สูงกว่าสัญญาณที่ขา E ประมาณ 4 เท่า

2.1.9 รีจิสเตอร์ของซีพียู

ซีพียูของไมโครคอนโทรลเลอร์ 68HC11 จะมีรีจิสเตอร์ใช้งานอยู่ 7 ตัวอันได้แก่

1. แอกคิวมูเลเตอร์ A และ B
2. แอกคิวมูเลเตอร์ D
3. รีจิสเตอร์อินเด็กซ์ IX

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

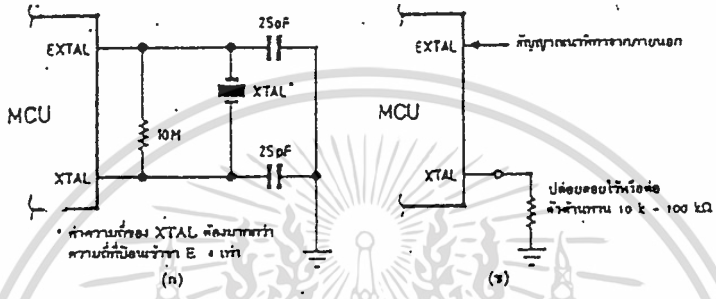
4. รีจิสเตอร์อินเด็กซ์ IY

5. รีจิสเตอร์ตัวชี้สแต็ค (Stack Pointer : SP)

6. รีจิสเตอร์โปรแกรมเคาน์เตอร์ (Prigram Counter: PC)

7. รีจิสเตอร์รหัสเงื่อนไข (Condition Code Register : CCR)

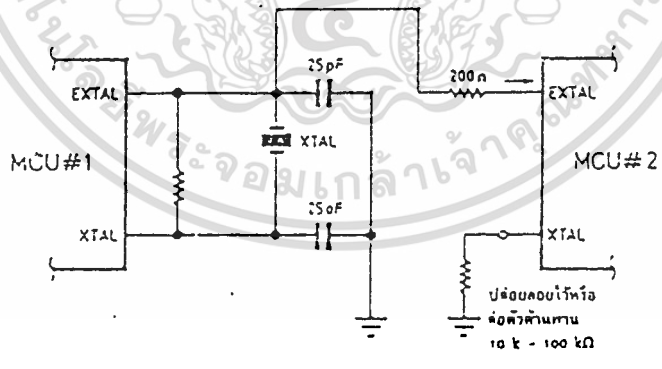
โดยมีลักษณะของรูปแบบการจัดจำนวนบิทและความหมายดังรูปที่ 2.10



รูปที่ 2.8 การต่อสัญญาณนาฬิกาให้กับ 68HC11

ก. สัญญาณนาฬิกาภายใน

ข. สัญญาณานาฬิกาจากภายนอก

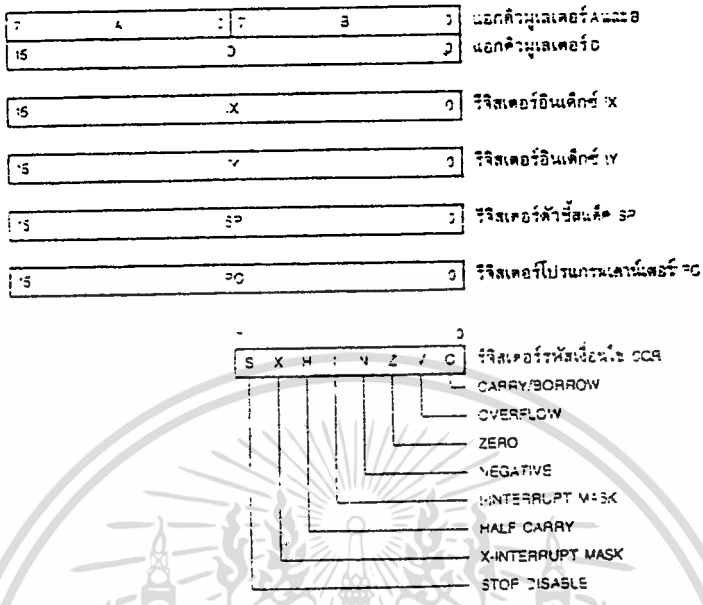


รูปที่ 2.9 การต่อสัญญาณนาฬิกาไมโครคอนโทรลเลอร์ตัวหนึ่งไปยังอีกตัวหนึ่ง

2.1.9.1 แอกคิวมูเลเตอร์ A และ B

เป็นรีจิสเตอร์ขนาด 8 บิท จะใช้ในการเก็บค่าโอเปอร์แรนด์และผลของการคำนวณทางคณิตศาสตร์ หรือผลของการจัดข้อมูลโดยซีพียู ในการประมวลผลทางคณิตศาสตร์หรือลอจิก จะต้องนำข้อมูลเหล่านั้นมาเก็บในรีจิสเตอร์ทั้งสองนี้จึงจะสามารถประมวลผลได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.10 รูปแบบการจัดบิทและความหมายรีจิสเตอร์รหัสเงื่อนไขของซีพียู

2.1.9.2 แอกคิวมูเลเตอร์ D

เป็นรีจิสเตอร์ขนาด 16 บิท ที่ใช้ในการประมวลผลและใช้สำหรับเก็บค่าการคำนวณทางคณิตศาสตร์และลอจิก แอกคิวมูเลเตอร์ D เกิดจากการรวมกันของแอกคิวมูเลเตอร์ A และ B จึงทำให้มีขนาด 16 บิท

2.1.9.3 รีจิสเตอร์อินเด็กซ์ IX

เป็นรีจิสเตอร์ขนาด 16 บิทสำหรับใช้ในการชี้ตำแหน่งแอดเดรสเพื่อเข้าไปจัดการประมวลผลกับข้อมูลในแอดเดรสนั้น ๆ นอกจากนั้นรีจิสเตอร์ IX ยังสามารถใช้เป็นตัวนับหรือรีจิสเตอร์เก็บข้อมูลชั่วคราวได้ด้วย

2.1.9.4 รีจิสเตอร์อินเด็กซ์ IY

เป็นรีจิสเตอร์ขนาด 16 บิท มีหน้าที่เหมือนกับ IX แต่แตกต่างกันที่ในทุกคำสั่งต้องเกี่ยวข้องกับ IY จะต้องมีข้อมูลไบต์พิเศษของรหัสแมชชีน และไซเคิลพิเศษของช่วงเวลาการเอ็กซ์คิวต์ด้วย จึงทำให้คำสั่งที่มี IY ไปเกี่ยวข้องมีขนาดอย่างน้อย 2 ไบต์ขึ้นไปหรือที่เรียกว่า พรีไบต์ (Prebyte)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.1.9.5 รีจิสเตอร์ตัวชี้สแต็ค SP

เป็นรีจิสเตอร์ขนาด 16 บิท ใช้เก็บแอดเดรสบนสแต็ค (Stack) โดยสแต็คบน 68HC11 นี้จะมีลักษณะการเก็บข้อมูลเข้าและการนำข้อมูลออกมาเป็นแบบ LIFO (Last In First Out) หรือข้อมูลที่นำเข้าไปเก็บในสแต็คหลังสุดเมื่อจะเรียกออกมาจะถูกเรียกออกมาก่อน สแต็คจะใช้เก็บข้อมูลของรีจิสเตอร์เมื่อต้องการนำรีจิสเตอร์ตัวนั้นไปทำงานในโปรแกรมย่อยอื่น ๆ หรือต้องไปใช้ในการตอบสนองการอินเตอร์รัพต์ เพื่อเป็นการป้องกันข้อมูลเดิมสูญหายต้องเก็บข้อมูลนั้นไว้ในหน่วยความจำสำรองก่อนก็คือ สแต็คนั่นเอง

ทุกครั้งที่มีการเก็บข้อมูลลงบนสแต็ค ค่าของ SP จะลดลง ในทางตรงข้ามถ้าเรียกข้อมูลออกมาจากสแต็ค ค่าของ SP ก็จะเพิ่มขึ้น

2.1.9.6 รีจิสเตอร์โปรแกรมเคาน์เตอร์ PC

เป็นรีจิสเตอร์ขนาด 16 บิทใช้เก็บค่าแอดเดรสของคำสั่งถัดไปที่ซีพียูจะไปทำการเอ็กซ์คิวต์

2.1.9.7 รีจิสเตอร์รหัสเงื่อนไข CCR

เป็นรีจิสเตอร์ขนาด 8 บิท ในแต่ละบิทจะแสดงความหมายของผลจากการกระทำคำสั่งที่เพิ่งจะเอ็กซคิวต์ไปของซีพียู ดังแสดงในรูปที่ 2.10 แต่ละบิทของ CCR เป็นอิสระต่อกัน จึงสามารถตรวจสอบสถานะได้โดยโปรแกรม และสามารถนำผลของการตรวจสอบนั้นไปดำเนินการต่อได้ด้วย รายละเอียดแต่ละบิทใน CCR มีดังนี้

CARRY/BORROW (C) : เรียกว่า "บิททด" บิทนี้จะเซ็ตเมื่อซีพียูทำการประมวลผลทางคณิตศาสตร์แล้วเกิดการทดค่า (Carry) หรือยืมค่า (Borrow) บิทนี้สามารถที่จะใช้งานร่วมกับคำสั่งการหมุน (Rotate) และเลื่อนข้อมูล (Shift) ได้

OVER FLOW (V) : บิทนี้จะเซ็ตเป็น "1" เมื่อซีพียูได้ทำคำสั่งทางคณิตศาสตร์แล้วเกิดค่าที่เกินออกมา นอกเหนือจากเงื่อนไขดังกล่าวบิทนี้จะเป็น "0"

ZERO (Z) : บิทนี้จะเซ็ตเมื่อผลของการกระทำทางคณิตศาสตร์และลอจิกหรือการประมวลผลข้อมูลแล้วทำให้เกิดเป็นค่าศูนย์ขึ้นมา

NEGATIVE (N) : ถ้าหากผลของการกระทำทางคณิตศาสตร์และลอจิก หรือการประมวลผลข้อมูลแล้วทำให้เกิดค่าเป็นลบ บิทนี้จะเซ็ตผลลัพธ์ที่มีค่าเป็นลบสามารถสังเกตุได้จากบิทที่มีนัยสำคัญสูงสุด (MSB) มีค่าเป็นลอจิก "1"

I-INTERRUPT MASK (I) : เราสามารถเซ็ตบิทนี้ให้เป็น "1" ได้ 2 วิธี คือ โดยวิธีทางฮาร์ดแวร์ และโดยการใช้คำสั่งที่ใช้ในการดิสเอเบิล การกำเนิดอินเตอร์รัพต์

เอกสารนี้สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

แบบมาสเคเบิลทั้งภายในและภายนอก

HALF CARRY (H) : จะเซ็ตเป็น "1" เมื่อซีพียูกระทำคำสั่งทางคณิตศาสตร์ เช่น ADD, ABA หรือ ADC แล้วเกิดการทดเข้ามาจากบิทที่ 3 มายังบิทที่ 4

X-INTERRUPT MASK (X) : บิทนี้จะถูกเซ็ตด้วยวิธีการทางฮาร์ดแวร์เท่านั้นโดยการป้อนสัญญาณเข้าที่ขา RESET และขา XIRQ จะรีเซ็ตบิทนี้ด้วยการใช้คำสั่ง TAP หรือ RTI เท่านั้น

STOP DISABLE (S) : บิทนี้จะถูกเซ็ตเมื่อต้องการดิสเอเบิลคำสั่ง STOP และถ้ารีเซ็ตบิทนี้จะเป็นการอีนาเบิลคำสั่ง STOP บิทนี้ถูกควบคุมโดยโปรแกรมและเมื่อบิทนี้ถูกเซ็ตจะทำให้คำสั่ง STOP มีผลเช่นเดียวกับคำสั่ง NOP

2.1.10 วงจรวอตช์ด็อก

ในไมโครคอนโทรลเลอร์สมัยใหม่มักมีวงจรวอตช์ด็อกอยู่ในชิพ โดยหน้าที่ของวงจรวอตช์ด็อก คือ ทำตัวคอยตรวจสอบการทำงานของซีพียูว่ายังทำงานอยู่ในโปรแกรม หรือมีปัญหาเกิดขึ้นกับซอฟต์แวร์จนไม่สามารถทำงานต่อไปได้ถูกต้อง หลักการของวอตช์ด็อกจะใช้หลักการง่าย ๆ ด้วยการเขียนรหัสอักขระมายังรีจิสเตอร์ตัวใดตัวหนึ่ง ในลักษณะลำดับช่วงเวลาที่กำหนด หากไม่เขียนภายในเวลาที่กำหนด วงจรจะทำการกระตุ้นบอกซีพียูให้กลับมาเริ่มต้นโปรแกรมที่กำหนดใหม่เพื่อตั้งต้นการทำงานให้ถูกต้องตามเดิม

สำหรับ 68HC11ใช้วิธีการเขียนรหัส $55 ลงแอดเดรสที่กำหนดเพื่อบอกการทำงานและส่งรหัส $AA มาเขียนลงในแอดเดรสที่กำหนดเพื่อรีเซ็ตวงจรตั้งเวลา ดังนั้นเมื่อวงจรตั้งเวลาไม่ได้รับการรีเซ็ตภายในเวลาที่กำหนดไว้ก็จะส่งสัญญาณไปรีเซ็ตซีพียูให้เริ่มต้นใหม่

2.1.11 วงจรเอทูดี

เป็นวงจรแปลงสัญญาณอะนาล็อกให้เป็นสัญญาณดิจิตอลขนาด 8 บิท โดยสามารถรับอินพุทได้ถึง 8 ช่อง จะทำการรับครั้งละช่องโดยการโปรแกรมเพื่อเลือกว่าจะทำการแปลงสัญญาณที่ช่องใด ผลลัพธ์ที่ได้จากการแปลงเป็นอัตราส่วนระหว่างสัญญาณอินพุทกับแหล่งจ่ายที่เป็นไฟอ้างอิงของวงจร A/D โดยแรงดันที่ใช้จะอยู่ในช่วง 2.5 - 5 โวลท์ เวลาที่ใช้ในการแปลงสัญญาณนั้นจะใช้สัญญาณนาฬิกา 32 ลูกต่อหนึ่งครั้งหรือที่เวลา 16 ไมโครวินาที ที่ความถี่ 8 MHz ผลลัพธ์ที่อ่านเข้ามาได้แล้วจะถูกเก็บอยู่ในรีจิสเตอร์ ADDR1 ถึง ADR4 ดังตารางที่ 2.6

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

CD	CC	CB	CA	Channel Signal	Result in ADRx if MULLTL = 1
0	0	0	0	AN0	ADR1
0	0	0	1	AN1	ADR2
0	0	1	0	AN2	ADR3
0	0	1	1	AN3	ADR4
0	1	0	0	AN4	ADR1
0	1	0	1	AN5	ADR2
0	1	1	0	AN6	ADR3
0	1	1	1	AN7	ADR4
1	0	0	0	Reserved	ADR1
1	0	0	1	Reserved	ADR2
1	0	1	0	Reserved	ADR3
1	0	1	1	Reserved	ADR4
1	1	0	0	V_{RH} Pin	ADR1
1	1	0	1	V_{RL} Pin	ADR2
1	1	1	0	$(V_{RH})/2$	ADR3
1	1	1	1	Reserved	ADR4

ตารางที่ 2.6 การกำหนดช่องการทำงานและการเก็บข้อมูลของวงจร A/D ใน 68HC11

2.1.12 ไทม์เมอร์

68HC11 มีไทม์เมอร์ขนาด 16 บิท ที่ทำงานในลักษณะฟรีรันนิ่ง คือ จะทำการนับตั้งแต่ 0000H-FFFFH และกลับมาเริ่มต้นนับใหม่ โดยใช้สัญญาณนาฬิกาจากภายในตัวซีพียู โดยไทม์เมอร์นี้จะเป็นตัวเปรียบเทียบกับรีจิสเตอร์ไทม์เมอร์อินพุทและค่าไทม์เมอร์เอาท์พุท รีจิสเตอร์ไทม์เมอร์อินพุท (input captuer) เป็นรีจิสเตอร์ที่อ่านได้อย่างเดียวมีทั้งหมด 3 ตัว ถ้าหากมีสัญญาณลอจิก "0" ที่ขาอินพุทของแต่ละตัวหลังจากที่ได้ทำการโหลดค่าจากฟรีรันไทม์เมอร์แล้ว สามารถที่จะทำการอินเตอร์รัพต์ซีพียูได้และยังสามารถนำค่าที่โหลดมา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ใช้ในการวิเคราะห์คาบเวลาได้ว่าเป็นความถี่เท่าไร

รีจิสเตอร์ไทม์เมอร์เอาท์พุท (output compare) เป็นรีจิสเตอร์ที่เขียนและอ่านได้มีทั้งหมด 5 ตัว อยู่ที่ตำแหน่งตั้งแต่แอดเดรส 1016H-101FH เราสามารถเขียนค่าลงไปยังรีจิสเตอร์ไทม์เมอร์เอาท์พุท เพื่อที่จะนำไปเปรียบเทียบกับฟรีรันไทม์เมอร์ เมื่อค่าทั้งสองเท่ากันก็จะทำการอินเตอร์รัพต์ซีพียู หรือจะทำการหารความถี่จากสัญญาณนาฬิกาไปยังเอาท์พุทได้ทั้ง 5 ตัว ซึ่งสามารถควบคุมโดยกำหนดค่าในรีจิสเตอร์ TMSK1 และ TCTL1

2.1.13 เรียลไทม์

เป็นวงจรส่วนหนึ่งทำหน้าที่เป็นฐานเวลาให้โปรแกรม โดยสามารถโปรแกรมเลือกเวลาไปยังรีจิสเตอร์ PACTL ที่บิท RTR0 และRTR1 บิท RTIF ในรีจิสเตอร์ TFLG2 จะเป็นตัวกำหนดว่าจะให้ทำการอินเตอร์รัพต์หรือไม่ ถ้าต้องการให้บิท RTOF มีค่าเป็น "1"

2.1.14 พัลส์แอกคิวมูเลเตอร์

68HC11 มีรีจิสเตอร์สำหรับนับพัลส์จากภายนอกขนาด 8 บิท จึงยความถี่ที่สามารถนับได้สูงสุดจะเท่ากับความถี่ของสัญญาณนาฬิกาภายในตัวซีพียูหารด้วย 64 ลักษณะของการนับจะเป็นการนับขึ้นและถ้านับจนเกินค่าของรีจิสเตอร์แล้วจะทำการอินเตอร์รัพซีพียูได้ โดยการเซ็ตที่บิท PAOVI ของรีจิสเตอร์ TMSK2

2.1.15 รีเซ็ต

การรีเซ็ตของ 68HC11 จะแบ่งออกเป็น 3 ชนิด คือ

1. การรีเซ็ตจากภายนอกซึ่งจะเกิดจากการเปิดเครื่องครั้งในแรกหรือสัญญาณรีเซ็ตจากภายนอก

2. การรีเซ็ตจากวงจรวอตช์ด็อก

3. การรีเซ็ตเนื่องจากการผิดพลาดของสัญญาณนาฬิกา

การรีเซ็ตจากการเปิดเครื่องจะใช้จำนวนพัลส์ของสัญญาณนาฬิกา 4064 ลูกหลังจากที่สัญญาณนาฬิกาลูกแรกทำงาน ถ้าใช้คริสตอลภายนอกที่ 8 MHz จะใช้เวลา 2 มิลลิวินาที ถ้าซีพียูทำงานในโหมดบูตสแตร์ปหรือโหมดทดสอบพิเศษ หลังจากซีพียูรีเซ็ตจะไปเฟตช์ข้อมูลจากเวคเตอร์แอดเดรสที่ FFFCH-FFFFH หรือ BFFEH-BFFFH ในกรณีที่มีการรีเซ็ตจากวงจรวอตช์ด็อกจะทำให้สัญญาณที่จะถูกรีเซ็ตเป็นลอจิก "0" ด้วยแล้วจะไปเฟตช์เอาข้อมูลที่เวคเตอร์แอดเดรส FFFAH-FFFBH และกรณีการรีเซ็ตจากสัญญาณนาฬิกาจะไปเฟตช์ข้อมูลที่เวคเตอร์แอดเดรส FFFCH-FFFDH

เอกสารนี้ใช้เพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.1.16 อินเตอร์รัพต์

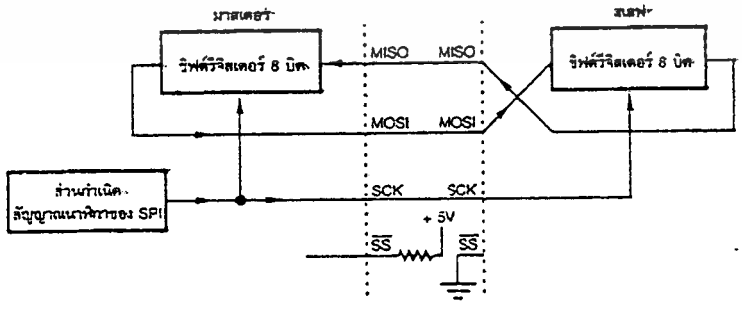
ลักษณะการอินเตอร์รัพต์นี้แบ่งเป็น 2 แบบ คือ ฮาร์ดแวร์และซอฟต์แวร์อินเตอร์รัพต์ ฮาร์ดแวร์อินเตอร์รัพต์ ซึ่งยังแบ่งออกได้เป็นการอินเตอร์รัพต์จากภายในและจากภายนอก สามารถที่จะลำดับความสำคัญของการขออินเตอร์รัพต์ได้ด้วยการควบคุมรีจิสเตอร์ HPRIO การอินเตอร์รัพต์จะทำงานเป็นแบบเวคเตอร์อินเตอร์รัพต์

2.1.17 พอร์ทอนุกรม

พอร์ทอนุกรมของ 68HC11 มีลักษณะเหมือนกับพอร์ทอนุกรมในไมโครคอนโทรลเลอร์โดยทั่วไป การรับส่งข้อมูลเป็นแบบฟูลดูเพล็กซ์ คือ สามารถรับส่งข้อมูลได้พร้อมกันในเวลาเดียวกันก่อนที่จะรับข้อมูลมายังรีจิสเตอร์จะต้องอ่านข้อมูลเดิมออกไปก่อน มิฉะนั้นข้อมูลที่ส่งมาใหม่จะทับข้อมูลเดิม แอดเดรสของรีจิสเตอร์อยู่ที่ตำแหน่ง 102FH ในการรับส่งข้อมูลนั้นสามารถที่จะกำหนดการรับส่งข้อมูลได้ว่าเป็นแบบ 10 บิทหรือ 11 บิท แต่ลักษณะพิเศษของพอร์ทอนุกรมสำหรับ 68HC11 นี้คือ สามารถที่จะทำการตรวจสอบสัญญาณรบกวนได้ โดยความกว้างของสัญญาณรบกวนจะต้องไม่เกิน 1/16 ของอัตราความเร็วในการส่ง

2.1.18 มัลติโปรเซสเซอร์

นอกจากการติดต่อข้อมูลแบบพอร์ทอินพุท เอาท์พุท และพอร์ทอนุกรมแล้ว 68HC11 ยังมีการติดต่อแบบมัลติโปรเซสเซอร์ที่สามารถนำซีพียูมาต่อหลาย ๆ ตัวได้โดยมีซีพียูตัวหนึ่งเป็นมาสเตอร์แล้วมีซีพียูตัวอื่น ๆ เป็นสเลฟจะใช้สัญญาณที่ขา SS, SCK, MISO, MOSI เป็นตัวติดต่อ ดังรูปที่ 2.11



รูปที่ 2.11 ไมโครคอนโทรลเลอร์ 68HC11 ที่ทำงานแบบมัลติโปรเซสเซอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.2 การอ้างแอดเดรสและชุดคำสั่งของ 68HC11

2.2.1 การอ้างแอดเดรสของ 68HC11

68HC11 มีกระบวนการอ้างแอดเดรส (Addressing) ทั้งสิ้น 6 โหมดด้วยกัน คือ

1. การอ้างแอดเดรสแบบทันทีทันใด (Immediat addressing)
2. การอ้างแอดเดรสแบบโดยตรง (Direct addressing)
3. การอ้างแอดเดรสแบบขยาย (Extended addressing)
4. การอ้างแอดเดรสแบบอินเด็กซ์ (Index addressing)
5. การอ้างแอดเดรสแบบอินเฮียเรนต์ (Inherent addressing)
6. การอ้างแอดเดรสแบบสัมพัทธ์ (Relative addressing)

ในบางคำสั่งเมื่อถูกใช้ในการอ้างแอดเดรสบางแบบ จำเป็นต้องเพิ่มข้อมูลเข้าไปอีก 1 ไบต์ก่อนออปโค้ดเพื่อปรับให้ 68HC11 สามารถทำงานได้ในลักษณะมัลติเพจออปโค้ดแมป (Multi page op-code map)ได้เหมาะสมยิ่งขึ้น ข้อมูลไบต์นั้นจะเรียกว่า (Prebyte)

2.2.1.1 การอ้างแอดเดรสแบบทันทีทันใด (Immediate Addressing)

ในการอ้างแอดเดรสแบบนี้เป็นการติดต่อเพื่อจัดการข้อมูลซึ่งเป็นค่าใด ๆ โดยตรงในทันทีทันใด จำนวนไบต์ของคำสั่งในการอ้างแอดเดรสแบบนี้จะมีขนาดตั้งแต่ 2 - 4 ไบต์ ขึ้นอยู่กับขนาดของรีจิสเตอร์ที่เกี่ยวข้องด้วย เช่น ถ้าเป็นแอกคิวมูเลเตอร์ A ก็จะมีจำนวนไบต์ของคำสั่ง 2ไบต์(1 ไบต์แรกเป็นออปโค้ด อีก 1 ไบต์หลังเป็นขนาดของโอเปอร์แรนด์ ซึ่งเท่ากับขนาดของแอกคิวมูเลเตอร์ A)

รูปแบบคำสั่งที่มีการอ้างแอดเดรสแบบนี้ หลังจากคำสั่งต้องตามด้วยเครื่องหมาย # เสมอ เพื่อเป็นการบ่งบอกให้ซีพียูทราบว่า คำสั่งที่จะทำต่อไปนี้ใช้ในการอ้างแอดเดรสแบบทันทีทันใด

2.2.1.2 การอ้างแอดเดรสแบบโดยตรง (Direct Addressing)

เป็นการติดต่อเพื่อเป็นการประมวลผลข้อมูลขนาด 8 บิท ที่อยู่ในหน่วยความจำภายในชิพซึ่งมีอยู่ 256 ไบต์ โดยมีแอดเดรสตั้งแต่ $0000 - $00FF ดังนั้นค่าโอเปอร์แรนด์ที่ตามหลังออปโค้ดคำสั่งก็ คือ ค่าแอดเดรสของแรมนั่นเอง

ในการอ้างแอดเดรสแบบนี้ บางทีเรียกว่า การอ้างแอดเดรสเพจศูนย์ (Zero page addressing)

2.2.1.3 การอ้างแอดเดรสแบบขยาย (Extended addressing)

ในการอ้างแอดเดรสแบบนี้ ข้อมูลในไบต์ที่ 2 และ 3 ที่ตามหลังออปโค้ด จะเก็บค่า

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์ ห้ามนำไปใช้เพื่อการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

แอดเดรสจริงของหน่วยความจำที่ต้องการนำข้อมูลในหน่วยความจำออกมาประมวลผล หรือจัดเก็บข้อมูลลงไปใหม่ ใช้การอ้างแอดเดรสแบบนี้คำสั่งจะมีขนาด 3-4 ไบต์ โดยออปโค้ดไบต์ที่ 1 หรือ 2 (กรณีถ้ามีขนาด 4 ไบต์) ส่วน 2 ไบต์หลังนี้จะเป็นค่าแอดเดรสที่อ้างถึงในหน่วยความจำ

2.2.1.4 การอ้างแอดเดรสแบบอินเด็กซ์ (Indexed Addressing)

การอ้างแอดเดรสแบบนี้จะมีการนำรีจิสเตอร์ชี้ข้อมูล (Index register) ทั้ง 2 ตัว คือ IX และ IY มาใช้ในการคำนวณค่าแอดเดรสที่ต้องการเรียกใช้ข้อมูล ดังนั้นค่าแอดเดรสที่ต้องการใช้งานจะเปลี่ยนแปลงหรือไม่ ขึ้นอยู่กับองค์ประกอบหลัก 2 ประการ คือ

1. ค่าแอดเดรสที่ถูกกำหนดอยู่ในปัจจุบัน
2. ค่าออฟเซ็ต 8 บิทที่บรรจุอยู่ในคำสั่งหรือค่าโอเปอร์แรนด์

ในการอ้างแอดเดรสแบบนี้ สามารถที่จะใช้หน่วยความจำที่ตำแหน่งใดก็ได้ในจำนวน 64 กิโลไบต์ เป็นจุดอ้างอิง ส่วนในด้านขนาดของคำสั่งในการอ้างแอดเดรสแบบนี้ ถ้าใช้รีจิสเตอร์ IX จะมีขนาด 2 ไบต์ โดยไบต์แรกเป็นออปโค้ด มีไบต์ที่สองเป็นค่า ออฟเซ็ต ถ้าหากใช้รีจิสเตอร์ IY จะมีขนาด 3 ไบต์ ไบต์แรกคือพรีไบต์ ตามด้วยออปโค้ด และค่าออฟเซ็ตขนาด 8 บิท

2.2.1.5 การอ้างแอดเดรสแบบอินเฮียเรนต์ (Inherent Addressing)

การอ้างแอดเดรสแบบนี้จะไม่ยุ่งเกี่ยวกับข้อมูลในหน่วยความจำแต่อย่างใด แต่เข้าไปจัดการในรีจิสเตอร์แทน ดังนั้นขนาดของคำสั่งในการอ้างแอดเดรสแบบนี้จะมีเพียง 1-2 ไบต์ โดยเป็นออปโค้ดทั้งสิ้นไม่มีโอเปอร์แรนด์

2.2.1.6 การอ้างแอดเดรสแบบสัมพัทธ์ (Relative Addressing)

การอ้างแอดเดรสแบบนี้จะใช้ชุดคำสั่งกระโดด (Jump and branch instructions) ถ้าเงื่อนไขในการกระโดดเป็นจริงค่าออฟเซ็ตขนาด 8 บิทที่อยู่ตามหลังออปโค้ดก็จะถูกบวกเข้าไปในรีจิสเตอร์โปรแกรมเคาน์เตอร์(PC)เพื่อกำหนดแอดเดรสต่อไปที่จะข้ามไปทำงานของซีพียู ปกติแล้วขนาดของคำสั่งที่ใช้ในการอ้างแอดเดรสแบบนี้จะเท่ากับ 2 ไบต์ โดยไบต์แรกเป็นออปโค้ด ส่วนไบต์ที่สองเป็นค่าออฟเซ็ตเพื่อบอกจำนวนที่จะเข้าไปเพิ่มใน PC

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.2.2 พรีไบต์ (Prebyte)

เป็นข้อมูลขนาด 8 บิท (1 ไบต์) ที่ใส่เข้าไปก่อนหน้าออปโค้ด เพื่อประโยชน์ในการบอกให้ซีพียูทราบถึงลักษณะของการจัดเพจออปโค้ด โดยถ้าหาก 68HC11 ทำงานในเพจ 1 ข้อมูลพรีไบต์ก็ไม่ต้องมี แต่ถ้าหากทำงานในเพจ 2 จะต้องเพิ่มค่าพรีไบต์เท่ากับ $18 เข้าไปก่อนออปโค้ดเสมอ แล้วตามด้วยออปโค้ด ในกรณีเพจ 3 จะใช้ค่า $1A และใช้ค่า $CD สำหรับเพจ 4 ค่าพรีไบต์จะเข้าไปเกี่ยวข้องเมื่อมีคำสั่งใด ๆให้รีจิสเตอร์ IY ทำงาน

2.2.3 ชุดคำสั่งของ 68HC11

ซีพียูภายในไมโครคอนโทรลเลอร์ 68HC11 จะมีลักษณะการทำงานที่คล้ายกับซีพียู MC6801 แต่ได้มีการเพิ่มเติมความสามารถของ 68HC11 ด้วยการเพิ่มรีจิสเตอร์และคำสั่งให้มากขึ้น เช่น เพิ่มรีจิสเตอร์อินเด็กซ์ขนาด 16 บิทอีก 1 ตัวคือ IY, เพิ่มคำสั่งเกี่ยวกับการหารเลข 16 บิทอีก 2 แบบ, การเพิ่มคำสั่งควบคุมและเพิ่มคำสั่งจัดการข้อมูลระดับบิท (Bit manipulation instruction) ในที่นี้จะแบ่งชุดคำสั่งของ 68HC11 ออกเป็น 7 กลุ่ม ดังต่อไปนี้

1. กลุ่มคำสั่งจัดการเกี่ยวกับข้อมูล (Data handling instructions)
2. กลุ่มคำสั่งทางคณิตศาสตร์ (Arithmatics instructions)
3. กลุ่มคำสั่งทางลอจิก (Logic instructions)
4. กลุ่มคำสั่งการกระโดด (Jump and Branch instructions)
5. กลุ่มคำสั่งเปรียบเทียบและตรวจสอบ (Data test instructions)
6. กลุ่มคำสั่งจัดการเกี่ยวกับรีจิสเตอร์รหัสเงื่อนไข (CCR instructions)
7. กลุ่มคำสั่งควบคุม (Control instructions)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.2.3.1 กลุ่มคำสั่งจัดการเกี่ยวกับข้อมูล

(1) กลุ่มคำสั่งเคลื่อนย้ายข้อมูล

	พึงคำสั่ง	การทำงาน	จำนวนไบต์ จำนวนไซเกิล และสัญญาณสถานะ						แฟลกรีจิสเตอร์ที่เปลี่ยนไป							
			แบบอ้างเลียเวนต์	แบบนัยทั่งไต	แบบตรง	แบบอ้อม	แบบอินเดกซ์									
							X	Y	S	X	H	I	N	Z	V	C
กลุ่มบิต	BSET	M ← M OR mm			3 5		3 7	4 8	-	-	-	-	↕	↕	0	-
	BCLR	M ← M AND (mm)			3 5		3 7	4 8	-	-	-	-	↕	↕	0	-
กลุ่มไบต์	LOAA (LOAB)	A ← M		2 2	2 3	3 4	2 4	3 5	-	-	-	-	↕	↕	0	-
	STAA (STAB)	M ← A			2 3	3 4	2 4	3 5	-	-	-	-	↕	↕	0	-
	TAB	B ← A	1/2						-	-	-	-	↕	↕	0	-
	TBA	A ← B	1/2						-	-	-	-	↕	↕	0	-
	TAP	CC ← A	1/2						↕	↕	↕	↕	↕	↕	↕	↕
	TPA	A ← CC	1/2						-	-	-	-	-	-	-	-
	CLRA (CLRB)	A ← 0	1/2						-	-	-	-	0	1	0	0
	CLR	M ← 0				3 5	2 5	3 7	-	-	-	-	0	1	0	0
	PSHA (PSHB)	Stack ← A	1/3						-	-	-	-	-	-	-	-
	PULA (PULB)	A ← Stack	1/4						-	-	-	-	-	-	-	-
กลุ่มเวิร์ด	LDD	A ← M B ← M+1		2 3	2 4	3 5	2 5	3 6	-	-	-	-	↕	↕	0	-
	STD	M ← A M+1 ← B			2 4	3 5	2 5	3 6	-	-	-	-	↕	↕	0	-
	XGDX	D ↔ X	1/3						-	-	-	-	-	-	-	-
	XGDY	D ↔ Y	2/4						-	-	-	-	-	-	-	-
	LDX	X ← M:M+1		3 3	2 4	3 5	2 5	3 5	-	-	-	-	↕	↕	0	-
	STX	M:M+1 ← X			2 4	3 5	2 5	3 5	-	-	-	-	↕	↕	0	-
	LDY	Y ← M:M+1		4 4	3 5	4 6	3 6	3 6	-	-	-	-	↕	↕	↕	-
	STY	M:M+1 ← Y			3 5	4 6	3 6	3 6	-	-	-	-	↕	↕	↕	-
	LDS	S ← M:M+1		3 3	2 4	3 5	2 5	3 5	-	-	-	-	↕	↕	0	-
	STS	M:M+1 ← S			2 4	3 5	2 5	3 5	-	-	-	-	↕	↕	↕	-
	PSHX	Stack ← X	1/4						-	-	-	-	-	-	-	-
	PULX	X ← Stack	1/5						-	-	-	-	-	-	-	-
	PSHY	Stack ← Y	2/5						-	-	-	-	-	-	-	-
	PULY	Y ← Stack	2/5						-	-	-	-	-	-	-	-
	TSX	X ← S+1	1/3						-	-	-	-	-	-	-	-
	TXS	S ← X-1	1/3						-	-	-	-	-	-	-	-
	TSY	Y ← S+1	2/4						-	-	-	-	-	-	-	-
	TYS	Y ← Y-1	2/4						-	-	-	-	-	-	-	-

ตารางที่ 2.7 กลุ่มคำสั่งเคลื่อนย้ายข้อมูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

(2) กลุ่มคำสั่งเปลี่ยนแปลงและแก้ไขข้อมูล

	วิธีคำสั่ง	สูตรคำนวณ	จำนวนไบต์/จำนวนไซเคิลของสัญญาณนาฬิกา E						แฟลกบิตเรจิสเตอร์ที่เปลี่ยนไป						
			แบบอ้างเนื้อความ	แบบสัมพัทธ์	แบบตรง	แบบขยาย	แบบอินเด็กซ์								
							X	Y	S	X	H	I	N	Z	V	C
เพิ่มค่า	INCA (INCB)	A ← A+1	1/2						·	·	·	·	↕	↕	↕	↕
	INC	M ← M+1				3/6	2/6	3/7	·	·	·	·	↕	↕	↕	↕
	INX	X ← X+1	1/3						·	·	·	·	↕	↕	·	·
	INY	Y ← Y+1	2/4						·	·	·	·	↕	↕	·	·
	INS	S ← S+1	1/3						·	·	·	·	·	·	·	·
ลดค่า	DECA (DECB)	A ← A-1	1/2						·	·	·	·	↕	↕	↕	·
	DEC	M ← M-1				3/6	2/6	3/7	·	·	·	·	↕	↕	↕	·
	DEX	X ← X-1	1/3						·	·	·	·	↕	↕	·	·
	DEY	Y ← Y-1	2/4						·	·	·	·	↕	↕	·	·
	DES	S ← S-1	1/3						·	·	·	·	·	·	·	·
แปลงค่า	COMA (COMB)	A ← FFH-A	1/2						·	·	·	·	↕	↕	0	1
	COM	M ← FFH-M				3/6	2/6	3/7	·	·	·	·	↕	↕	0	1

ตารางที่ 2.8 กลุ่มคำสั่งเปลี่ยนแปลงและแก้ไขข้อมูล

(3) กลุ่มคำสั่งเลื่อนและหมุนข้อมูล

	วิธีคำสั่ง	สูตรคำนวณ	จำนวนไบต์/จำนวนไซเคิลของสัญญาณนาฬิกา E						แฟลกบิตเรจิสเตอร์ที่เปลี่ยนไป						
			แบบอ้างเนื้อความ	แบบสัมพัทธ์	แบบตรง	แบบขยาย	แบบอินเด็กซ์								
							X	Y	S	X	H	I	N	Z	V	C
หมุนซ้ายบุ	ROLA (ROLB)	C A หรือ B หรือ M	1/2						·	·	·	·	↕	↕	↕	↕
	ROL					3/6	2/6	3/7	·	·	·	·	↕	↕	↕	↕
	RORA (RORB)	C A หรือ B หรือ M	1/2						·	·	·	·	↕	↕	↕	↕
	ROR					3/6	2/6	3/7	·	·	·	·	↕	↕	↕	↕
เลื่อนข้อมูลเลขคณิตซ้าย	ASLA (ASLB)	C A หรือ B หรือ M 0	1/2						·	·	·	·	↕	↕	↕	↕
	ASL					3/6	2/6	3/7	·	·	·	·	↕	↕	↕	↕
	ASLD	C D 0	1/3						·	·	·	·	↕	↕	↕	↕
เลื่อนข้อมูลเลขคณิตขวา	ASRA (ASRB)	C A หรือ B หรือ M	1/2						·	·	·	·	↕	↕	↕	↕
	ASR					3/6	2/6	3/7	·	·	·	·	↕	↕	↕	↕
เลื่อนข้อมูลลอจิก	LSLA (LSLB)	C A หรือ B หรือ M 0	1/2						·	·	·	·	↕	↕	↕	↕
	LSL					3/6	2/6	3/7	·	·	·	·	↕	↕	↕	↕
	LSLD	C D 0	1/3						·	·	·	·	↕	↕	↕	↕
	LSRA (LSRB)	C A หรือ B หรือ M	1/2						·	·	·	·	0	↕	↕	↕
	LSR					3/6	2/6	3/7	·	·	·	·	0	↕	↕	↕
	LSRD	C D 0	1/3						·	·	·	·	0	↕	↕	↕

ตารางที่ 2.9 กลุ่มคำสั่งการเลื่อนและหมุนข้อมูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.2.3.2 กลุ่มคำสั่งทางคณิตศาสตร์

	รหัสคำสั่ง	การทำงาน	จำนวนไบต์/จำนวนไซเกิลของลัญญาณนาฬิกา E						แฟลกวิจิสเตอร์รหัสเงื่อนไข							
			แบบอิน-เฮียเรนต์	แบบทันที ทันใด	แบบตรง	แบบจงใจ	แบบอินเด็กซ์									
							X	Y	S	X	H	I	N	Z	V	C
การบวก	ADDA (ADD8)	A ← A+M		2/2	2/3	3/4	2/4	3/5	↕	↕	↕	-	↕	↕	↕	↕
	ABA	A ← A+B	1/2						↕	↕	↕	-	↕	↕	↕	↕
	ADCA (ADC8)	A ← A+M+C		2/2	2/3	3/4	2/4	3/5	↕	↕	↕	-	↕	↕	↕	↕
	DAA	Adjust binary sum in A to BCD	1/2						↕	↕	-	-	↕	↕	↕	↕
	ADDD	D ← D+(M:M+1)		3/4	2/5	3/6	2/6	3/7	↕	↕	-	-	↕	↕	↕	↕
	ABX	X ← X+00:B	1/3						-	-	-	-	-	-	-	-
	ABY	Y ← Y+00:B	2/4						-	-	-	-	-	-	-	-
การลบ	SUBA (SUB8)	A ← A-M		2/2	2/3	3/4	2/4	3/5	↕	↕	↕	-	↕	↕	↕	↕
	SBA	A ← A-B	1/2						↕	↕	↕	-	↕	↕	↕	↕
	SBCA (SBC8)	A ← A-M-C		2/2	2/3	3/4	2/4	3/5	↕	↕	↕	-	↕	↕	↕	↕
	NEGA (NEG8)	A ← 0-A	1/2						↕	↕	↕	-	↕	↕	↕	↕
	NEG	M ← 0-M				3/6	2/6	3/7	↕	↕	↕	-	↕	↕	↕	↕
	SUBD	D ← D-(M:M+1)		3/4	2/5	3/6	2/6	3/7	↕	↕	↕	-	↕	↕	↕	↕
การคูณ และการหาร	MUL	D ← A*B	1/10						-	-	-	-	-	-	↕	↕
	IDIV	X ← D/X; D ← r	1/41						-	-	-	-	↕	0	↕	↕
	FDIV	X ← D/X; D ← r	1/41						-	-	-	-	↕	↕	↕	↕

ตารางที่ 2.10 กลุ่มคำสั่งทางคณิตศาสตร์

2.2.3.3 กลุ่มคำสั่งทางลอจิก

รหัสคำสั่ง	การทำงาน	จำนวนไบต์/จำนวนไซเกิลของลัญญาณนาฬิกา E						แฟลกวิจิสเตอร์รหัสเงื่อนไข							
		แบบอิน-เฮียเรนต์	แบบทันที ทันใด	แบบตรง	แบบจงใจ	แบบอินเด็กซ์									
						X	Y	S	X	H	I	N	Z	V	C
ANDA (AND8)	A ← A AND M		2/2	2/3	3/4	2/4	3/5	↕	↕	↕	-	↕	↕	0	-
ORAA (ORA8)	A ← A OR M		2/2	2/3	3/4	2/4	3/5	↕	↕	↕	-	↕	↕	0	-
EORA (EOR8)	A ← A EOR A		2/2	2/3	3/4	2/4	3/5	↕	↕	↕	-	↕	↕	0	-

ตารางที่ 2.11 กลุ่มคำสั่งทางลอจิก

2.2.3.4 กลุ่มคำสั่งการกระโดด

(1) กลุ่มคำสั่งกระโดดแบบไม่มีเงื่อนไข

เอกสารนี้มีเจตนาเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รหัสคำสั่ง	การทำงาน	จำนวนไบต์/จำนวนไซเกิลของสัญญาณนาฬิกา E						แฟลกรีจิสเตอร์รหัสเงื่อนไข							
		แบบอินเฮอเรนต์	แบบทันทีกันได	แบบตรง	แบบขยาย	แบบอินเด็กซ์ X	แบบอินเด็กซ์ Y	S	X	H	I	N	Z	V	C
JMP	PC ← M,M+1				3/3	2/3	3/4	-	-	-	-	-	-	-	-
JSR	Stack ← PC;			2/5	3/5	2/6	3/7	-	-	-	-	-	-	-	-
	PC ← M;M+1														
BRA	PC ← PC+M		2/3					-	-	-	-	-	-	-	-
BRN	PC ← PC+2		2/3					-	-	-	-	-	-	-	-
BSR	Stack ← PC;		2/6					-	-	-	-	-	-	-	-
	PC ← PC+M														
RTS	PC ← Stack	1/5						-	-	-	-	-	-	-	-
RTI	CC,B,A,X,Y,PC ← Stack	1/12						↕	↕	↕	↕	↕	↕	↕	↕

ตารางที่ 2.12 กลุ่มคำสั่งการกระโดดแบบไม่มีเงื่อนไข

(2) กลุ่มคำสั่งกระโดดแบบมีเงื่อนไข

	รหัสคำสั่ง	การทำงาน	จำนวนไบต์/จำนวนไซเกิลของสัญญาณนาฬิกา E						แฟลกรีจิสเตอร์รหัสเงื่อนไข							
			แบบอินเฮอเรนต์	แบบทันทีกันได	แบบตรง	แบบขยาย	แบบอินเด็กซ์ X	แบบอินเด็กซ์ Y	S	X	H	I	N	Z	V	C
จากอนพิกไบไนค์	BITSET	Branch if (M) AND mm = 00H			4/6		4/7	5/8	-	-	-	-	-	-	-	-
	BITCLR	Branch if M AND mm = 00H			4/6		4/7	5/8	-	-	-	-	-	-	-	-
แยกตามแฟลก	BCS			2/3					-	-	-	-	-	-	-	-
	BCC			2/3					-	-	-	-	-	-	-	-
	BEQ			2/3					-	-	-	-	-	-	-	-
	BNE			2/3					-	-	-	-	-	-	-	-
	BMI			2/3					-	-	-	-	-	-	-	-
	BPL			2/3					-	-	-	-	-	-	-	-
	BVS			2/3					-	-	-	-	-	-	-	-
	BVC			2/3					-	-	-	-	-	-	-	-
เปรียบเทียบกับข้อมูลตัวเลข	BLT			2/3					-	-	-	-	-	-	-	-
	BLE			2/3					-	-	-	-	-	-	-	-
	BEQ			2/3					-	-	-	-	-	-	-	-
	BGE			2/3					-	-	-	-	-	-	-	-
	BGT			2/3					-	-	-	-	-	-	-	-
	BNE			2/3					-	-	-	-	-	-	-	-
เปรียบเทียบข้อมูล 2 ชิ้นงาน	BLO			2/3					-	-	-	-	-	-	-	-
	BLS			2/3					-	-	-	-	-	-	-	-
	BEQ			2/3					-	-	-	-	-	-	-	-
	BHS			2/3					-	-	-	-	-	-	-	-
	BHI			2/3					-	-	-	-	-	-	-	-
	BNE			2/3					-	-	-	-	-	-	-	-

ตารางที่ 2.13 กลุ่มคำสั่งการกระโดดแบบมีเงื่อนไข

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.2.3.5 กลุ่มคำสั่งเปรียบเทียบและตรวจสอบข้อมูล

	รหัสคำสั่ง	การทำงาน	จำนวนไบต์/จำนวนไซเกิลของสัญญาณนาฬิกา E						แฟลกบิตสเตตัสเงื่อนไข							
			แบบอิน-เฮ้อเรนต์	แบบทันที กันได	แบบตรง	แบบจัดเก็บ	แบบอินเด็กซ์ X	แบบอินเด็กซ์ Y	S	X	H	I	N	Z	V	C
จัดบุลขนาด 8 บิต	CMPA (CMP8)	A - M		2/2	2/3	3/4	2/4	3/5	-	-	-	-	↕	↕	↕	↕
	CBA	A - B	1/2						-	-	-	-	↕	↕	↕	↕
	TSTA (TSTB)	A - 0	1/2						-	-	-	-	↕	↕	0	0
	TST	M - 0				3/6	2/6	3/7	-	-	-	-	↕	↕	0	0
	BITA (BITB)	A AND M		2/2	2/3	3/4	2/4	3/5	-	-	-	-	↕	↕	0	-
จัดบุลขนาด 16 บิต	CPD	D - (M:M+1)		4/5	3/6	4/7	3/7	3/7	-	-	-	-	↕	↕	↕	↕
	CPX	X - (M:M+1)		3/4	2/5	3/6	2/6	3/7	-	-	-	-	↕	↕	↕	↕
	CPY	Y - (M:M+1)		4/5	3/6	4/7	3/7	3/7	-	-	-	-	↕	↕	↕	↕

ตารางที่ 2.14 กลุ่มคำสั่งเปรียบเทียบและตรวจสอบข้อมูล

2.2.3.6 กลุ่มคำสั่งจัดการกับรีจิสเตอร์รหัสเงื่อนไข

รหัสคำสั่ง	การทำงาน	จำนวนไบต์/จำนวนไซเกิลของสัญญาณนาฬิกา E						แฟลกบิตสเตตัสเงื่อนไข							
		แบบอิน-เฮ้อเรนต์	แบบทันที กันได	แบบตรง	แบบจัดเก็บ	แบบอินเด็กซ์ X	แบบอินเด็กซ์ Y	S	X	H	I	N	Z	V	C
SEC	C ← 1	1/2						-	-	-	-	-	-	-	1
CLC	C ← 0	1/2						-	-	-	-	-	-	-	0
SEV	V ← 1	1/2						-	-	-	-	-	-	1	-
CLV	V ← 0	1/2						-	-	-	-	-	-	0	-
SEI	I ← 1 (disable interrupts)	1/2						-	-	-	1	-	-	-	-
CLI	I ← 0 (enable interrupts)	1/2						-	-	-	0	-	-	-	-

ตารางที่ 2.15 กลุ่มคำสั่งจัดการกับรีจิสเตอร์รหัสเงื่อนไข

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.2.3.7 กลุ่มคำสั่งควบคุม

รหัสคำสั่ง	การทำงาน	จำนวนไบต์/จำนวนไซเกิลของอัญญาณนาฬิกา E.						แฟลกซ์ซิสเตอร์/บิตเงื่อนไข							
		แบบอิมเมดิเอต	แบบทันที ทันใด	แบบตรง	แบบจดจำ	แบบอินเด็กซ์ X	แบบอินเด็กซ์ Y	S	X	H	I	N	Z	V	C
SWI	Stack ← PC,Y,X,A,B,CC; I ← 1; PC ← M:M+1	1/14						-	-	-	1	-	-	-	-
WAI	Stack ← PC,Y,X,A,B,CC;Halt	2/12						-	-	-	-	-	-	-	-
STOP	- -	1/2						-	-	-	-	-	-	-	-
NOP	PC ← PC+1	1/2						-	-	-	-	-	-	-	-

ตารางที่ 2.16 กลุ่มคำสั่งควบคุม

2.2.4 ความแตกต่างของการ JUMP และ BRANCH

คำสั่ง JUMP อันได้แก่ JMR, JSR จะสามารถกระโดดไปที่ใดในหน่วยความจำก็ได้ โดยกำหนดค่าแอดเดรสลงไปชัดเจน ทำให้ต้องใช้หน่วยความจำสิ้นเปลือง แต่ถ้าเป็นคำสั่ง BRA หรือ BSR จะสามารถกระโดดไปได้ในขอบเขต +127 และ -128 ตำแหน่งจากจุดที่กระทำคำสั่ง ทำให้ใช้คำสั่งในการเขียนโปรแกรมน้อยกว่า และซีพียูทำงานเร็วกว่า แต่มีข้อด้อยกว่าตรงที่มีขอบเขตของการกระโดดจำกัด

คำสั่งเกี่ยวกับการ BRANCH ถ้าจะเทียบกับไมโครโปรเซสเซอร์ Z80 ก็คือคำสั่ง Jump Relative หรือ JR นั่นเอง ซึ่งค่าสัมพัทธ์ (Relative) จะต้องมีการคำนวณด้วยดังสูตร

```
ค่าสัมพัทธ์ (relative : rr)  = PC ใหม่ - PC เก่า -  2
```

โดย rr จะอยู่ที่ค่า 8 บิทท้ายของผลที่ได้ เช่น PC ใหม่มีค่าเท่ากับ $E009, PC เก่ามีค่าเท่ากับ $E012 ดังนั้นค่า

```
rr = $E009 - $E012 - 2 = $FFF5
```

นั่นคือเราจะได้ค่า rr ที่แท้จริงก็คือ $F5

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.3 การเขียนโปรแกรมเพื่ออ้างแอดเดรส

68HC11 มีการอ้างแอดเดรส (Addressing) อยู่ด้วยกัน 6 แบบคือ

1. แบบทันทีทันใด (Immediate addressing)
2. แบบโดยตรง (Direct addressing)
3. แบบขยาย (Extended addressing)
4. แบบอินเด็กซ์ (Indext addressing)
5. แบบอินเฮียเรนต์ (Inherent addressing)
6. แบบสัมพัทธ์ (Relative addressing)

ดังนั้นการเขียนโปรแกรมเพื่ออ้างแอดเดรสในแต่ละแบบก็จะแตกต่างกันไป

2.3.1 โปรแกรมสำหรับการอ้างแอดเดรสแบบทันทีทันใด

การเขียนโปรแกรมจะมีรูปแบบการเขียนจะเป็นดังนี้

คำสั่ง #$ค่าที่ต้องการ

$ คือ ตัวอักษรที่ใช้บ่งบอกว่า ค่าที่ตามหลังนี้จะเป็นเลขฐานสิบหก

ตัวอย่างของโปรแกรมมีดังนี้

```
LDAA #$5C
```

หมายถึง การนำค่า $5C ไปเก็บไว้ในแอกคิวมูเลเตอร์ A

```
LDX #$3B27
```

หมายถึง เป็นการนำค่า $3B27 ไปเก็บไว้ในรีจิสเตอร์อินเด็กซ์ X (IX)โดย $3B จะถูกเก็บไว้ในไบต์สูงของ IX ส่วน $27 จะเก็บไว้ในไบต์ต่ำของ IX

2.3.2 โปรแกรมสำหรับการอ้างแอดเดรสแบบโดยตรง

รูปแบบการเขียนโปรแกรมมีดังนี้

คำสั่ง $ค่าแอดเดรสของแรมที่อยู่ภายในชิพ (มีค่าได้ตั้งแต่ $00-$FF)

ตัวอย่างของโปรแกรมมีดังนี้

```
LDAA $1B
```

เป็นการนำค่าที่อยู่ในแรมแอดเดรสที่ $1B มาเก็บไว้ในแอกคิวมูเลเตอร์ A ไม่ใช่เป็นการนำค่า $1B มาเก็บไว้ในแอกคิวมูเลเตอร์ A

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.3.3 โปรแกรมสำหรับการอ้างแอดเดรสแบบขยาย

รูปแบบการเขียนโปรแกรมเป็นดังนี้

คำสั่ง $ค่าแอดเดรสหน่วยความจำที่อยู่ภายนอกชิพ

ตัวอย่างของโปรแกรมมีดังนี้

LDAB $2000

เป็นการนำข้อมูลในหน่วยความจำแอดเดรส $2000 มาเก็บในแอกคิวมูเลเตอร์ B

2.3.4 โปรแกรมสำหรับการอ้างแอดเดรสแบบอินเด็กซ์

รูปแบบการเขียนโปรแกรมเป็นดังนี้

คำสั่ง $ค่าออฟเซ็ต, X (หรือ Y)

ตัวอย่างของโปรแกรมมีดังนี้

STAA $05, X

สมมติว่า IX มีแอดเดรสก่อนหน้าที่จะเอ็กซ์คิวต์คำสั่งนี้อยู่ที่ $2000 เมื่อเอ็กซ์คิวต์คำสั่งในโปรแกรมนี้ เป็นการนำค่าที่อยู่ในแอกคิวมูเลเตอร์ A ไปเก็บลงในหน่วยความจำที่มีแอดเดรสเท่ากับ IX + 5 นั่นคือ $2000 + $5 = $2005

2.3.5 โปรแกรมสำหรับการอ้างแอดเดรสแบบอินเฮียเรนต์

เนื่องจากการอ้างแอดเดรสแบบนี้ จะไม่ยุ่งเกี่ยวกับหน่วยความจำแต่อย่างใด ดังนั้นทุก ๆ คำสั่งเป็นการกระทำกับรีจิสเตอร์เท่านั้น จึงเป็นการอ้างแอดเดรสแบบอินเฮียเรนต์ทั้งสิ้น ดังตัวอย่างต่อไปนี้

CLRA : เป็นการเคลียร์ค่าในแอกคิวมูเลเตอร์ A

XGDY : สลับค่าในแอกคิวมูเลเตอร์ D กับรีจิสเตอร์ IY

DEX : ลดค่าของ IX

2.3.6 โปรแกรมสำหรับการอ้างแอดเดรสแบบสัมพัทธ์

รูปแบบการเขียนโปรแกรมเป็นดังนี้

คำสั่ง ค่าออฟเซ็ตสัมพัทธ์ที่ต้องการ

เนื่องจากการอ้างแอดเดรสแบบนี้ มักจะใช้ในโปรแกรมที่มีการกระโดด ค่าออฟเซ็ต

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จึงเป็นค่าที่ใช้บอกให้ทราบว่า หลังจากที่เอ็กซ์คิวต์คำสั่งนี้แล้ว จะต้องกระโดดไปทำงานที่แอดเดรสใด ดังเช่น

```
$1000          BEQ JMP
                 .
                 .
                 .
$1007      JMP  LDAA #$05
```

ที่บรรทัด $1000 นั่นคือโปรแกรมที่เขียนขึ้นโดยอาศัยการอ้างแอดเดรสแบบสัมพัทธ์ ค่าออฟเซ็ตสัมพัทธ์ที่ต้องการทราบคำนวณจาก ค่าแอดเดรสเก่า - ค่าแอดเดรสใหม่ - T โดย T จะมีค่าเท่ากับ 2 สำหรับคำสั่งปกติ จะเป็น 4 เมื่อใช้คำสั่ง BRSET และ BRCLR สำหรับ IX และเท่ากับ 5 เมื่อใช้คำสั่ง BRSET และ BRCLR สำหรับ IY จากตัวอย่างจะได้ค่าออฟเซ็ตสัมพัทธ์เท่ากับ $1007-$1000-$2 = $05

2.4 รูปแบบการเขียนโปรแกรมทั่วไป

ทุกครั้งที่เริ่มต้นเขียนโปรแกรมภาษาแอสเซมบลีของ 68HC11 ต้องเริ่มต้นด้วย ORG แล้วตามด้วยแอดเดรสเริ่มต้นของโปรแกรมทุกครั้ง และเมื่อเขียนโปรแกรมจบแล้วจะต้องปิดท้ายด้วยคำสั่ง END เสมอ ดังตัวอย่าง

```
ORG  $1000
LDAA #$5C
STAA $1007
END
```

ในการเขียนโปรแกรมด้วยภาษาแอสเซมบลีทุกครั้ง จะต้องนำโปรแกรมไปคอมไพล์ให้เป็นออฟโค้ดเสียก่อน เพื่อให้ซีพียูภายใน 68HC11 เข้าใจคำสั่ง และนี่คือตัวอย่างของโปรแกรมที่ผ่านการคอมไพล์แล้ว

ออปโค้ดที่ผ่านการคอมไพล์แล้ว	โปรแกรมภาษาแอสเซมบลี
	ORG $4000

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
4000    CE 8B 26          LDX   #$8B26
4003    BF                XGDX
4004    C0 12             SUBB  #12
```

โปรแกรมที่ใช้คอมไพล์เรียกว่า ครอสแอสเซมเบลอร์ (Cross assembler) ซึ่งนอกจากจะใช้แปลง ภาษาแอสเซมบลี ยังสามารถแปลงโปรแกรมที่เป็น ภาษาซี ได้ด้วย ทำให้การพัฒนา 68HC11 ทำได้ง่ายและทรงประสิทธิภาพมากขึ้นเพราะการเขียนโปรแกรมด้วยภาษาซีจะทำได้ง่ายกว่า และจะประหยัดเวลาได้มากกว่าการเขียนโปรแกรมด้วยภาษาแอสเซมบลี ดังตัวอย่าง

```
time()
     /* function time () increments sec, min, hour based onthe 24
hour clock format */
     {
     sec+=1;
     if(sec>=60)
          {sec=0;
          min+=1
          if(min>=60)
               {min=0;
                    hour+=1;
               if(hour>=24)
               {hour=0}
               }
          }
     }
```

นั่นคือโปรแกรมนาฬิกา จะเห็นว่ามีเพียงไม่กี่บรรทัดเท่านั้น แต่ถ้าเป็นการเขียนด้วยภาษาแอสเซมบลี จะยาวกว่านี้มาก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.5 โปรแกรมคณิตศาสตร์

2.5.1 โปรแกรมบวกลบเลข

ตัวอย่างของโปรแกรมทางคณิตศาสตร์อย่างง่าย ที่ยกมาเป็นตัวอย่างนี้เป็นโปรแกรมลบเลขฐาน โดยตัวตั้งมีค่าเท่ากับ $05 เก็บไว้ในหน่วยความจำแอดเดรส $3000 ตัวบวกมีค่าเท่ากับ $03 อยู่ในแอดเดรส $3001 ตัวลบมีค่า $02 อยู่ที่แอดเดรส $3002 ผลลัพธ์จากการคำนวณให้มาเก็บไว้ที่ $3003

```
ORG  $2000
LDAA $3000    ; A = $05
ADCA $3001    ; A = $05 + $03 = $08
SBCA $3002    ; A = $08 - $02 = $06
STAA $3003    ; A  to $3003
END
```

2.5.2 โปรแกรมคูณเลข

ต้องการคูณเลขฐานสิบหกระหว่าง $20 กับ $35

```
ORG $2000
LDD #$2035   ; $20 to A, $35 to B
MUL          ; A*B = 06A0  to D
ADCA #$00    ; adjust to 8 bit
END
```

2.5.3 โปรแกรมหารเลข

ต้องการหาร 6 ด้วย 4 คำตอบก็จะได้ 1 เศษ 2 สามารถเขียนโปรแกรมได้ดังนี้

```
ORG $2000
LDD #$0006
LDX #$0004
IDIV
STX $00
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
LDX #$0004
FDIV
END
```

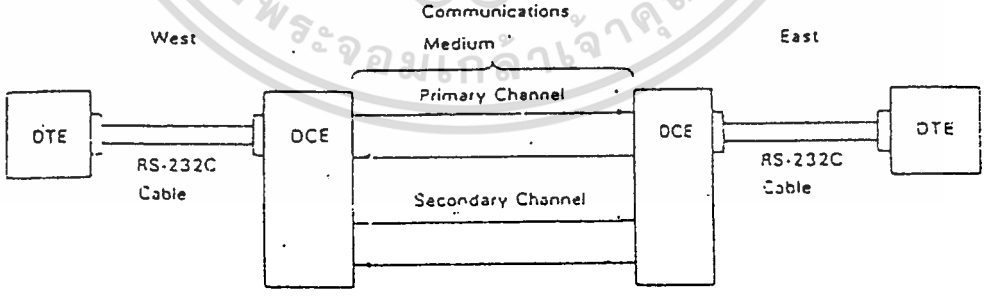
หลังจากกระทำคำสั่ง IDIV จะได้ผลการหารออกมาเป็น 1 เก็บไว้ที่รีจิสเตอร์ IX ส่วนเศษ 2 จะถูกเก็บไว้ในรีจิสเตอร์ จากนั้นจะทำการหารเศษ 2 ด้วย 4 จึงต้องเก็บค่าของผลลัพธ์ในช่วงแรกไว้ใน IX จากนั้นก็จะมาทำงานด้วยคำสั่ง FDIV

หลังจากที่เอ็กซ์คิวต์คำสั่ง FDIV จะได้ผลหารเท่ากับ 0.5 ถ้าแปลงเป็นเลขฐานสิบหก จะได้ค่า $8000 เก็บในรีจิสเตอร์ IX เหตุผลที่เป็นค่า $8000 เนื่องจากเป็นค่าครึ่งหนึ่งจาก $0000 ถึง $FFFF พอดี ดังนั้นผลหารจึงมีค่าเท่ากับ $1.8 หรือ 1.5 ในเลขฐานสิบ

2.6 อินเตอร์เฟส RS-232 C

2.6.1 แบบจำลองของวงจรการสื่อสารแบบ RS-232 C

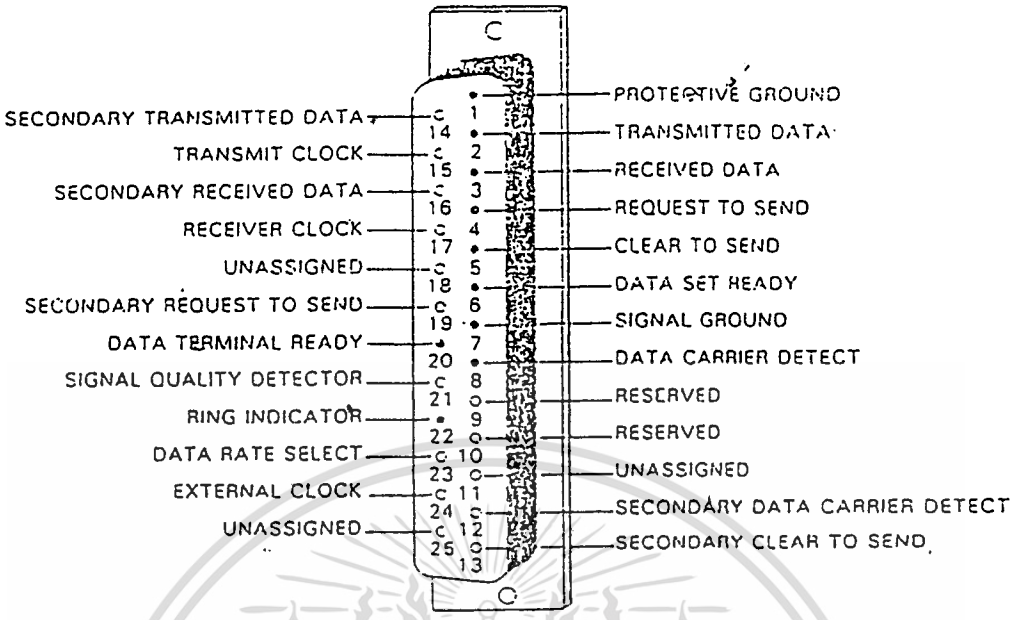
ข้อกำหนดตาม RS-232 C นั้น บอกถึงการเดินสายในเคเบิ้ลที่ต่อระหว่าง DTE (Data Terminal Equiment) กับ DCE (Data Communication Equipment) ตามรูปที่ 2.12 เป็นเคเบิ้ลที่ต่อกับปลั๊ก 25 ขา ที่เสียบเข้ากับคอนเนคเตอร์ "serial port" ที่ตั้ง PC หรืออุปกรณ์ปลายทางต่าง ๆ ปลั๊กนี้มีลักษณะดังรูปที่ 2.13



รูปที่ 2.12 แบบจำลองของวงจรการสื่อสารที่ใช้ RS-232 C

ในประการแรกข้อกำหนดจะกำหนดระดับแรงดันและคุณสมบัติทางไฟฟ้าอย่างอื่นของสายในเคเบิ้ลพร้อมทั้งอธิบายหน้าที่ของมันซึ่งจะได้กล่าวถึงในหัวข้อต่อไป และเพื่อที่จะให้เข้าใจ DTE เราจำเป็นต้องเข้าใจแบบจำลองหรือรูปแสดงแนวความคิดของตัวกลางการ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.13 ขั้วต่อตัวเมียของ RS-232 C

สื่อสารที่ใช้ ในข้อกำหนด RS-232 C ด้วย รูปแบบจำลองนี้แสดงในรูปที่ 2.12 ต่อไปเมื่อกล่าวถึงอุปกรณ์ที่อยู่ทางด้านซ้ายมือและด้านขวามือของตัวกลางการสื่อสาร เราจะอ้างว่าเป็นอุปกรณ์ที่อยู่ทางด้านซ้ายมือ และด้านขวามือของตัวกลางการสื่อสารเราจะอ้างว่าเป็นอุปกรณ์ด้านตะวันตก และด้านตะวันออก ตามลำดับ

ตัวกลางการสื่อสารประกอบด้วย ช่องสัญญาณหลัก (primary channel) และช่องสัญญาณรอง(secondary channel) ช่องสัญญาณนั้นเป็นช่องทาง (pipeline) ที่สัญญาณไหลผ่านเท่านั้นดังนั้นช่องสัญญาณหลักและช่องสัญญาณรองนั้นอาจมีอยู่ในสายคู่เดียวกัน หรือคนละคู่ก็ได้ เกี่ยวเนื่องกับช่องสัญญาณ ก็คือแนวความคิดเรื่อง คลื่นพาห์ (carrier) คลื่นพาห์คือ สัญญาณอนาล็อก (ปกติเป็นคลื่นรูปไซน์) ที่ความถี่ที่เหมาะสมที่จะเคลื่อนผ่านตัวกลางการสื่อสารได้ ตัวพาห์เป็นตัวบรรทุกข้อมูลที่เรียกว่ามอดดูเลทข้อมูล เพื่อนำข้อมูลผ่านตัวกลางการสื่อสารไปจุดหมายมีวิธีการมอดดูเลทหลายวิธี

ในแต่ละช่องสัญญาณมีสัญญาณคลื่นพาห์ 2 สัญญาณ แต่ละสัญญาณใช้สำหรับการส่งแต่ละทิศทาง ถ้าคลื่นพาห์ทั้งสองสามารถใช้พร้อมกันได้ ช่องสัญญาณก็จะเป็นชนิดฟูลดูเพล็กซ์ แต่ถ้าต้องใช้สลับเวลากัน ช่องสัญญาณนั้นก็เป็นชนิด ฮาล์ฟดูเพล็กซ์

ช่องสัญญาณมีค่าความจุ ซึ่งมีค่าเท่ากับจำนวนบิทต่อวินาทีที่มากที่สุดที่ช่องสัญญาณยอมให้ผ่านตัวมันไปได้ ช่องสัญญาณรองไม่จำเป็นต้องมีใช้ แต่ถ้ามีอยู่แล้วค่าความจุ(ความเร็ว)ของมันจะน้อยกว่าค่าความจุของช่องสัญญาณหลัก เพราะถ้าไม่เป็นเช่นนั้นก็จะไม่มีแนวความ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

แตกต่างระหว่างช่องสัญญาณหลักกับช่องสัญญาณรอง ข้อกำหนด RS-232 C กำหนดว่ามันไม่สามารถจะประยุกต์ใช้ได้กับตัวกลางการสื่อสารที่มีค่าสูงกว่า 200,000 บิทต่อวินาที

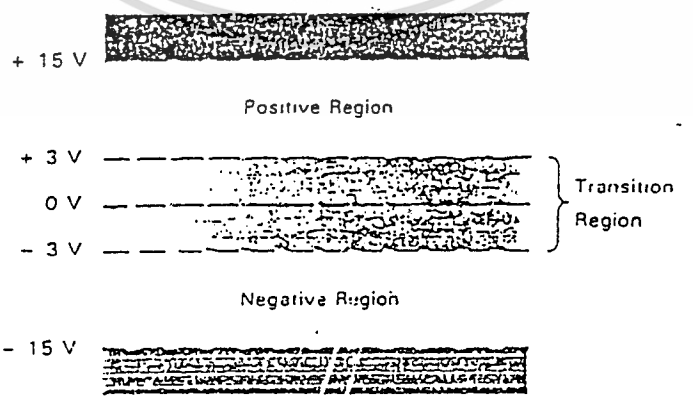
RS-232 C ได้ปล่อยการเลือกหลายอย่างไว้ให้ขึ้นอยู่กับผู้ออกแบบ เป็นต้นว่า

- ช่องสัญญาณจะเป็น ฮาล์ฟ หรือ ฟูลดูเพล็กซ์
- ตัวกลางการสื่อสารจะเป็นแบบสวิตซ์หรือแบบเช่าเฉพาะ
- ช่องสัญญาณจะเป็นแบบซิงโครนัสหรืออะซิงโครนัส
- ช่องสัญญาณรองจะมีหรือไม่

สิ่งเหล่านี้ก็ล้วนแล้วแต่ผู้ออกแบบว่าต้องการอย่างไร ผลก็คือจำนวนของสายที่อยู่ในเคเบิ้ล RS-232 C ที่ใช้ควบคุม DCE จะถูกใช้ไม่เท่ากัน ขึ้นอยู่กับการออกแบบ บางทีอาจจะไม่มีแบบหนึ่งแบบใดเลยก็ได้ที่ใช้สายครบหมดทุกสายตาม RS-232 C

2.6.2 ข้อกำหนดทางไฟฟ้า

มีรายละเอียดเกี่ยวกับคุณสมบัติของสัญญาณไฟฟ้าบนสายแต่ละสายในเคเบิล RS-232 มากมายแต่เพราะเราเกี่ยวข้องกับสัญญาณไบนารี ข้อจำกัดเกี่ยวกับแรงดันจึงถูกกำหนดลงในสองบริเวณ ดังแสดงในรูปที่ 2.14 บริเวณด่านบวก (positive region)อยู่ระหว่าง +3 โวลท์ ถึง +15 โวลท์ และบริเวณลบ (negative region) อยู่ระหว่าง -3 โวลท์ ถึง -15 โวลท์ บริเวณระหว่าง -3 โวลท์ ถึง 3 โวลท์ถือเป็นบริเวณเปลี่ยนถ่ายสถานะ (transittion region) มีข้อกำหนดให้สัญญาณจะมีสถานะอยู่ในบริเวณนี้ได้ไม่เกิน 1 มิลิวินาที ในบริเวณเปลี่ยนถ่ายสถานะนี้จะไม่มีการกำหนดสถานภาพให้กับสัญญาณแต่อย่างใด



รูปที่ 2.14 ระดับแรงดันของสัญญาณ RS-232 C ที่กำหนดให้ใช้งาน

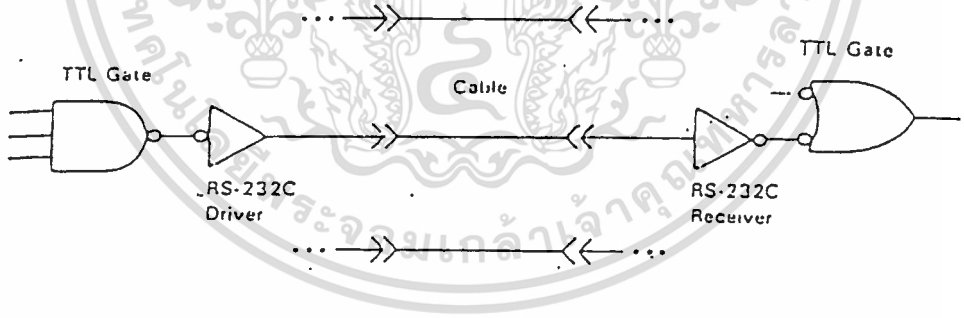
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บริเวณเหล่านี้ใช้เกี่ยวพันกับสถานะไบนารีของสัญญาณในลักษณะที่เป็นเอกเทศ การแปลความหมายของระดับแรงดันขึ้นอยู่กับหน้าที่ของสาย กำหนดได้ดังตารางที่ 2.17

WIRE FUNCTION	VOLTAGE LEVEL	
	Positive	Negative
Delta Modem Control & Timing	SPACE(0) on (asserted)	MARK(1) off (negated)

ตารางที่ 2.17 หน้าที่ของสายและความหมายของโวลท์เตจที่กำหนดให้

เนื่องจากระดับแรงดันดังกล่าวไม่พอเหมาะที่จะใช้กับ TTL ดังนั้นการอินเตอร์เฟสไปสู่เคเบิ้ล RS-232 C จึงต้องการวงจรภาคขับและภาครับเป็นพิเศษดังแสดงในรูที่ 2.15 ภาคขับและภาครับนี้สามารถหาได้ในรูป IC ตามท้องตลาดทั่วไป



รูปที่ 2.15 วงจรอินเตอร์เฟส RS-232 C

RS-232 C นั้นจำกัดค่าความจุไฟฟ้าของสายสัญญาณโดยวัดเทียบกับขากราวด์ อย่างมากสุดคือ 2,500 pF สำหรับสายเคเบิ้ลที่มีฉนวนและช่องว่างระหว่างสายอย่างปกติทั่วไป ถ้าสายยาวประมาณ 50 ฟุต จะมีค่าความจุไฟฟ้าประมาณนั้น ดังนั้นถ้าไม่ใช้สายชนิดพิเศษแล้ว ระยะห่างมากสุดระหว่าง DTE และ DCE คือ 50 ฟุต หรือประมาณ 15 เมตร

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Pin	Signal Name	Direction DTE DCE	Abbreviatio
1	PROTECTIVE (FRAME) GROUND		
2	TRANSMIT DATA	---->	XMT
3	RECEIVE DATA	<----	RCV
4	REQUEST TO SEND	---->	RTS
5	CLEAR TO SEND	<----	CTS
6	DATA SER READY	<----	DSR
7	SIGNAL GROUND (COMMON RETURN)		GRD
8	CARRIER DETECT	<----	CAR_DET
12	SECONDARY CARRIER DETECT	<----	SEC_CAR_DET
13	SECONDARY CLEAR TO SEND	<----	SEC_CTS
14	SECONDARY TRANSMIT DATA	---->	SEC_XMT
15	TRANSMIT CLOCK (DCE SOURCE)	<----	XMT_CLK
16	SECONDARY RECEIVE DATA	<----	SEC_RCV
17	RECEIVE CLOCK	<----	RCV_CLK
18	_		
19	SECONDARY REQUEST TO SEND	---->	SEC_RTS
20	DATA TERMINAL READY	---->	DTR
21	SIGNAL QUALITY DETECTOR	<----	SQD
22	RING INDICATOR	<----	RI
23	DATA RATE SELECTOR	---->	DR_SEL
24	TRANSMIT CLOCK (DTE SOURCE)	---->	XMT_CLK

ตารางที่ 2.18 การกำหนดขาขั้วต่อ RS-232 C

(ringing tone) เข้ามา และโมเด็มจะทำการตอบรับโดยการจัดวงจรเสมือนมีการยกหู โทรศัพท์รับ เมื่อได้คำสั่งจาก DTE ซึ่งตามปกติ DTE จะสั่งให้โมเด็มตอบรับการสื่อสารนั้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โดยใช้สัญญาณ DTE ส่งผ่านขาที่ 20

ปกติสำหรับเครื่องคอมพิวเตอร์เล็ก ๆ ตามท้องตลาด มักจะเกี่ยวข้องอยู่กับขาขั้วต่อ 10 ขาที่กล่าวมาแล้วข้างต้นเป็นสำคัญ ขาอื่น ๆ นั้นจะมีความสำคัญรองลงไป

ขาที่ 15, 17, 21 และ 24 จะให้สัญญาณเมื่อทำงานแบบซิงโครนัสเพราะทางด้านส่งจะต้องส่งข้อมูลบางอย่างไป (0 หรือ 1) ที่แต่ละช่องเวลาบิท (bit time)โมเด็มจะควบคุมจังหวะสัญญาณนาฬิกาจาก DTE และในทำนองเดียวกันโมเด็มที่ทำหน้าที่บก็จะต้องส่งบิทข้อมูลและจังหวะสัญญาณนาฬิกาที่ร่วมมาด้วยกันออกมาด้วย ขาที่ 15 และ 17 ใช้สนองความต้องการเหล่านี้และในกรณีที่สัญญาณควบคุมผ่านขาที่ 21 นั้นจะใช้เพื่อแสดงว่าคลื่นพาห์ที่รับเข้ามานั้นมีคุณสมบัติเป็นไปตามเงื่อนไขที่ได้กำหนดไว้ก่อนแล้ว

ขาที่ 23 ใช้เพื่อส่งสัญญาณเลือกอัตราการส่งสัญญาณข้อมูล ในกรณีเป็นชนิดที่สามารถเปลี่ยนอัตราส่งข้อมูล ขานี้จะแจ้งให้ทั้ง DTE ทางด้านส่งและด้านรับให้บอก DTE ทางด้านรับอีกต่อหนึ่ง

ขาที่ 12, 13, 14, 16 และ 19 เป็นขาสัญญาณที่ใช้กับช่องสัญญาณรอง ในโมเด็มบางเครื่องจะมีช่องสัญญาณใช้สองช่องคือช่องสัญญาณหลักและช่องสัญญาณรอง โดยขาสัญญาณทั้ง 5 ขาของช่องสัญญาณรองนั้น จะมีหน้าที่เหมือนกับหน้าที่ทางช่องสัญญาณหลักต่างกันแต่ว่าอัตราการส่งสัญญาณหลักต่างกัน แต่ว่าอัตราการส่งสัญญาณทางช่องสัญญาณรองนั้น ปกติมักจะช้ากว่าอัตราการส่งของช่องสัญญาณหลักและช่องสัญญาณรองนั้นจะมีทิศทางการส่งสัญญาณสวนกันกับทิศทางของช่องสัญญาณหลัก

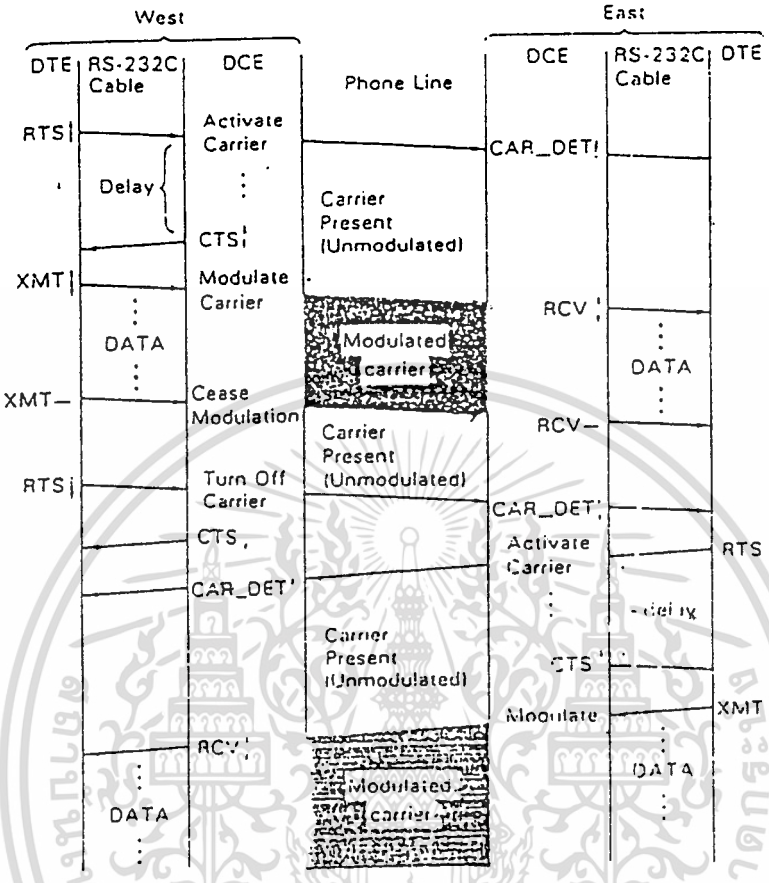
ลักษณะของข้อมูลที่ส่งผ่านอินเตอร์เฟส RS-232 C นั้นเป็นการส่งแบบลำดับอาจจะเป็นแบบอะซิงโครนัสหรือแบบซิงโครนัสก็ได้ขึ้นอยู่กับ DTE

2.6.4 การควบคุมคลื่นพาห์และการส่งข้อมูล

กระบวนการควบคุมคลื่นพาห์และการส่งข้อมูลสามารถอธิบายให้เข้าใจได้ง่ายโดยใช้รูปที่ 2.16 ในรูปนี้ใช้เครื่องหมายลูกศรที่พุ่งขึ้นท้ายตัวอักษรย่อแสดงถึงการเปิดส่งสัญญาณนั้นและจะใช้เครื่องหมายลูกศรพุ่งลงแสดงการปิดเลิกสัญญาณ เครื่องหมายลูกศรสองหัวชี้ขึ้นและลงแสดงถึงการเริ่มรับส่งข้อมูลการส่งข่าวสาร และการตอบรับแสดงโดยลูกศรที่พุ่งผ่านข้ามช่องการสื่อสาร โดยเวลาที่ผ่านไปนั้นแสดงโดยการเคลื่อนลงแนวดิ่งของรูปไดอะแกรมนั้น เครื่องหมายขีด(-)แสดงสถานะสงบ (quiescent)

เราเริ่มต้นด้วย DTE ด้านตะวันตกกระตุ้น DCE ด้วยสัญญาณ RST ทำให้ DCE ส่งคลื่นพาห์ไปกระตุ้น CAR-DET ด้านตะวันออก ให้แจ้งให้ DTE ด้านตะวันออกรู้ตัว หลังจากหน่วงเวลาประมาณ 150 มิลลิวินาที DCE ด้านตะวันตกจะส่ง CTS ไปแจ้ง DTE ว่าพร้อม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.16 การควบคุมคลื่นพาห์ในระบบฮาล์ฟดูเพล็กซ์

แล้วจึงได้มีการส่งข้อมูลเกิดขึ้น ซึ่งข้อมูลจะถูกมอดดูเลทกับคลื่นพาห์ส่งไปยังสถานีตะวันออก เมื่อสถานีตะวันตกส่งข้อมูลหมดก็จะปิดลด RTS ลง คลื่นพาห์ก็จะถูกปิดพร้อมกับ DCE ก็จะปิดตัด CAR-DET ลงด้วย ในตอนนี้ถ้า DTE ของสถานีตะวันออกต้องการจะส่งข้อมูลไปถึงสถานีด้านตะวันตกบ้าง จะต้องปฏิบัติกระบวนการเหมือนกับกระบวนการที่สถานีด้านตะวันตกได้กระทำมาแล้วทุกประการ จะเห็นได้ว่าการสื่อสารแบบที่ได้อธิบายอยู่นี้ เป็นการสื่อสารที่เรียกว่า ฮาล์ฟดูเพล็กซ์

ในระบบฮาร์ฟดูเพล็กซ์ต้องมีการกลับทิศทางการส่งสัญญาณและในการกลับทิศทางของการสื่อสารนี้ คลื่นพาห์ในระบบจะต้องถูกจัดส่งใหม่ เพราะคลื่นพาห์ต้องออกจากเครื่องส่งเสมอ ปรากฏการณ์สลับทิศทางของคลื่นพาห์นี้มีศัพท์เฉพาะเรียกว่า "สายวกกลับ" (line-turn around) ซึ่งจะเกิดขึ้นในระบบฮาล์ฟดูเพล็กซ์เป็นปกติ ดังที่ได้อธิบายมาแล้วว่าการที่ DCE จะตอบรับสัญญาณ RTS จาก DTE ด้วย CTS นั้นจะต้องหน่วงเวลาไว้ก่อนประมาณ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

150 มิลลิวินาที ดังนั้นการวกกลับสายแต่ละครั้งต้องใช้เวลาไม่น้อยกว่า 150 มิลลิวินาที และในระบบสื่อสารข้อมูลระยะไกล (long-distance data call) จะเกิดการสะท้อน (echo) ของสัญญาณอยู่เป็นปกติ ในการที่จะวกกลับสายได้จะต้องรอให้คลื่นพาห์ที่เกิดจากการสะท้อนลดหายไปก่อนด้วย ซึ่งจะต้องรอเวลาประมาณ 150 มิลลิวินาที ดังนั้นช่วงเวลาการวกกลับสายทำให้ประสิทธิภาพในการใช้สายลดลงเป็นอย่างมาก เวลา 150 มิลลิวินาที อาจดูเล็กน้อยแต่ลองพิจารณาถึงระบบที่ใช้อัตราส่งข้อมูล 9,600 บิท/วินาที ที่ใช้รหัสอักษรชิงโครนัสที่มีความยาว 8 บิท เราจะพบว่าจำนวนอักษรที่ส่งต่อวินาที คือ

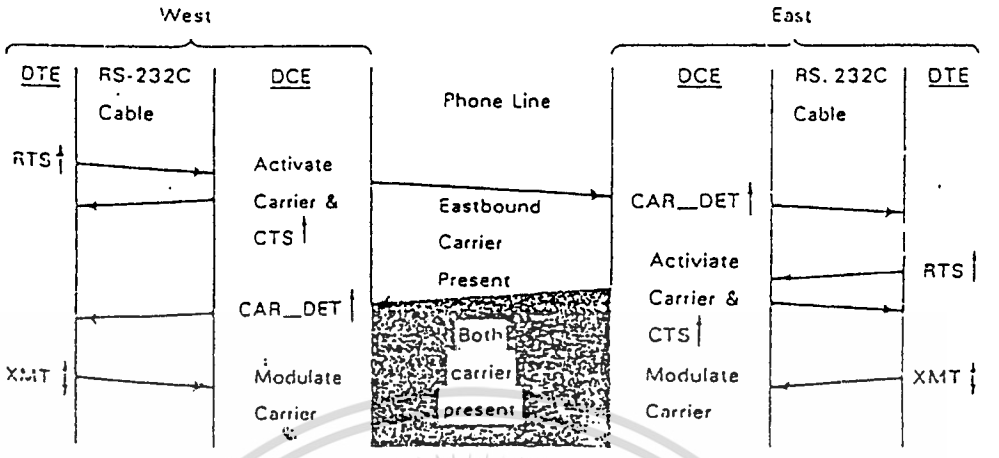
9,600/8 = 1,200 อักษร/วินาที

ดังนั้นในเวลา 150 มิลลิวินาที จะเป็นเวลาที่ส่งตัวอักษรได้เท่ากับ

1,200 x 0.15 = 180 ตัวอักษร

จะเห็นว่าในช่วงเวลาการวกกลับสายนี้ทำให้การส่งอักษรช้าลงไปถึง 180 ตัวอักษร ซึ่งปกติแล้วข้อมูล 180 ตัวอักษรนี้จะเป็นขนาดของบล็อกตัวอักษรที่ใช้กันในการสื่อสารข้อมูลธรรมดา ดังนั้นจึงนับว่าการวกกลับสายนั้นจะลดประสิทธิภาพในการใช้สายลงมากมาย ด้วยเหตุดังกล่าวนี้ในการส่งข้อมูลที่มีความเร็วสูงจึงนิยมทำการส่งในระบบฟูลดูเพล็กซ์ ในระบบฟูลดูเพล็กซ์ไม่มีการวกกลับสาย เพราะว่าคลื่นพาห์ที่ใช้กันสถานีละความถี่ จึงสามารถรักษาคลื่นพาห์ให้มีอยู่บนสายการสื่อสารได้ตลอดเวลาการสื่อสาร เมื่อไม่มีการเปลี่ยนส่งคลื่นพาห์จึงไม่มีความจำเป็นในการสลับสาย ตัวอย่างการส่งข้อมูลระบบฟูลดูเพล็กซ์แบบหนึ่งมีลักษณะดังในรูปที่ 2.17 ตามรูปนี้สถานีด้านตะวันตกได้เริ่มทำการสื่อสารโดยการส่ง RST ไปก่อน DEC ด้านตะวันตกเมื่อรับสัญญาณ RST ก็จะส่งคลื่นพาห์หลักไปสู่ DCE ด้านตะวันออก เมื่อตรวจพบคลื่นพาห์บนสาย DCE ก็จะส่ง CAR-DET ไปยัง DTE ด้านเดียวกัน DTE ทางด้านตะวันออกเมื่อได้รับ CAR-DET ก็จะจ่ง RST กลับมายัง DCE ให้ส่งคลื่นพาห์รองไปสู่ DCE ด้านตะวันตกพร้อมกับส่ง CTS มาที่ DTE ด้านตะวันออกด้วย เมื่อ DCE ด้านตะวันตกตรวจจับคลื่นพาห์รองได้ ก็จะส่ง CAR-DET ไปบอก DTE ด้านตะวันตกว่ามีการจัดระบบคลื่นพาห์บนสายการสื่อสารเรียบร้อยแล้ว จากช่วงเวลานี้การสื่อสารข้อมูลก็จะเริ่มต้นได้พร้อมกันทั้งสองสถานี

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



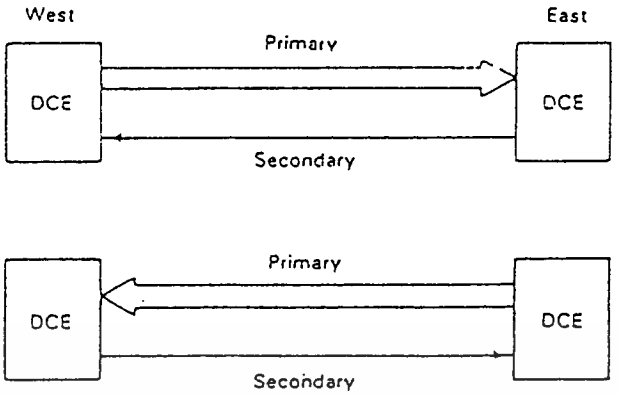
รูปที่ 2.17 การควบคุมคลื่นพาห์ของระบบฟูลดูเพล็กซ์

เพราะคลื่นพาห์นั้นจะปรากฏอยู่บนสายตลอดเวลาที่ทำการสื่อสารจึงไม่มีปรากฏการณ์สายวกกลับที่เป็นเหตุให้ประสิทธิภาพในการใช้สายเสียไป จึงสามารถปรับปรุงเวลาการส่งผ่านข้อมูลเพิ่มขึ้นเป็นอย่างมาก

2.6.5 ช่องสัญญาณรอง

ดังที่ได้กล่าวมาแล้วว่าหน้าที่ของช่องสัญญาณรองจะคล้ายกับหน้าที่ของช่องสัญญาณหลัก ช่องสัญญาณรองนี้ปกติจะมีบอดเรท (baud rate) ต่ำกว่าบอดของช่องสัญญาณหลัก และใช้ในทิศทางสวนกันกับทิศทางของช่องสัญญาณหลัก ดังนั้นก็จึงมีชื่ออีกอย่างหนึ่งว่าช่องย้อนกลับ อาจทำการสับเปลี่ยนการใช้ช่องสัญญาณได้สองแบบดังในรูปที่ 2.18 โดยที่ช่องสัญญาณหลัก (แสดงโดยลูกศรใหญ่) เป็นฮาล์ฟดูเพล็กซ์และถูกควบคุมด้วยวิธีการเหมือนที่ได้อธิบายมาแล้ว เครื่องส่งที่ใช้คลื่นพาห์รองจะถูกขัดสลัก (inter lock) กับคลื่นพาห์รองของสถานีนี้จะไม่มีโอกาสทำปฏิบัติการขึ้นมาได้เลยแม้ว่าสัญญาณร้องขอเพื่อการส่ง (request to sent) ของช่องสัญญาณรองจะถูกยืนยันขึ้นมาก็ตาม ดังนั้นจึงเรียกการส่งสัญญาณในระบบตามรูปที่ 2.18 นี้ว่าระบบดูเพล็กซ์สามในสี่ (threequarter duplex) กล่าวคือ ระบบนี้มีการติดต่อกันได้ในสองทิศทางพร้อมกัน แต่ในทิศทางหนึ่งจะมีความจุของการสื่อสารในช่องสัญญาณรองเพียง 5 บอดเท่านั้น เพราะวัตถุประสงค์ของการใช้ช่องสัญญาณรองนี้ก็เพียงเพื่อใช้ในการตอบรับการติดต่อเกี่ยวกับเรื่องการควบคุมการไหลของสัญญาณ (flow control) และการควบคุมความผิดพลาด (error control) หรือการส่งสัญญาณควบคุมเท่านั้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.18 การสลับสถานะแบบที่มีช่องย้อนกลับ

2.6.6 การตอบรับอย่างอัตโนมัติ

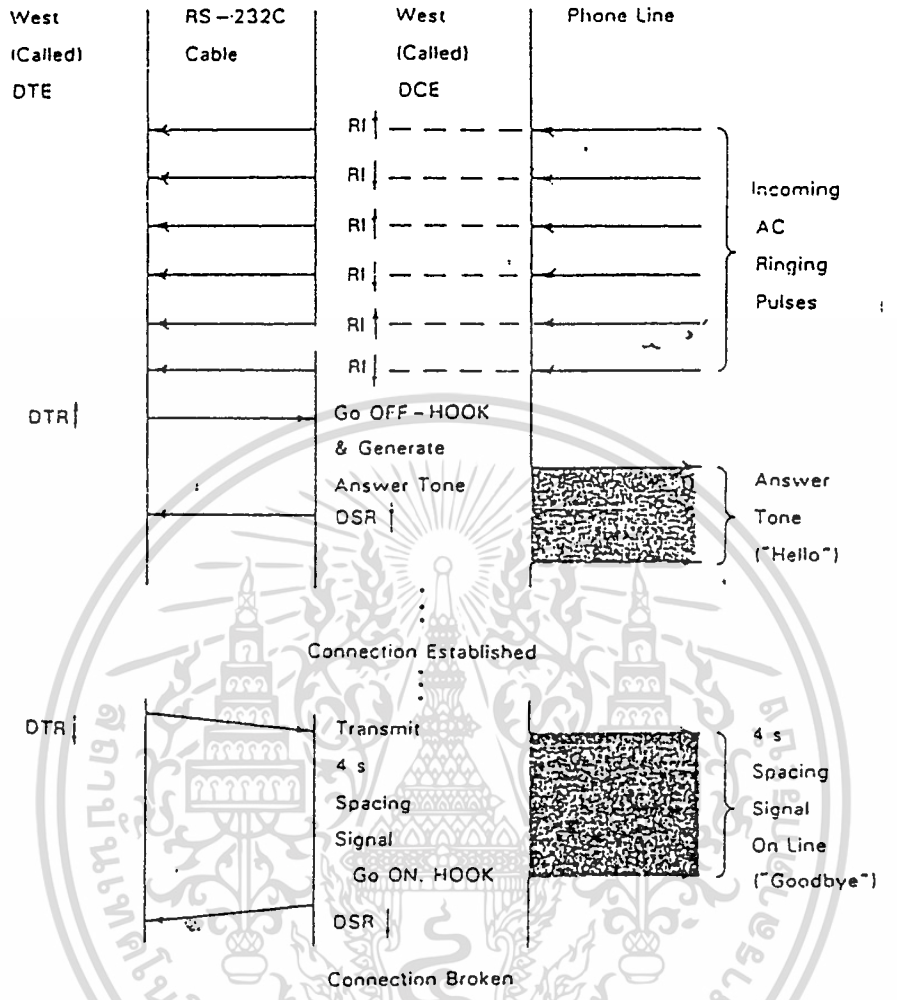
ในการตอบรับการติดต่ออย่างอัตโนมัติ เราอาจจัดระบบโดยการอาศัยสัญญาณบนขา DSR, DTR และ RI มาใช้ประโยชน์ โดยมีแผนภูมิเวลาการทำงานสังรูปที่ 2.19 ในรูปนี้ เราสมมุติว่ามีการเรียกจากสถานีตะวันออกมายังสถานีตะวันตก สัญญาณกระดิ่งโทรศัพท์ก็จะทำให้ DCE ด้านตะวันตกตรวจจับสัญญาณกระดิ่งได้แล้วส่งสัญญาณ RI ไปหา DTE ทางด้านตะวันตกเพื่อบอกว่ามีการติดต่อเข้ามาเมื่อสัญญาณ RI ได้รับการยืนยันแล้ว 3 ครั้งถ้า DTE ด้านตะวันตกพร้อมจะรับข้อมูลก็จะตอบรับด้วยการยืนยันสัญญาณ DTE มายัง DCE ทางด้านตะวันตก DCE จะตัดต่อวงจรเพื่อเลียนแบบการยกหูโทรศัพท์ พร้อมกับการส่งสัญญาณตอบรับซึ่งปกติจะเป็นสัญญาณรูปไซน์ความถี่ประมาณ 2,000 เฮิร์ต ติดต่อกันประมาณ 2 วินาที ไป DCE ด้านตะวันออก หลังจากนั้นก็จะยืนยันด้วยสัญญาณ DSR กลับไปยังที่ DTE ด้านตะวันตกเพื่อบอกให้ DTE รู้ว่าข่ายการสื่อสารถูกต่อให้เรียบร้อยแล้ว จากนี้จะเป็นกระบวนติดต่อข้อมูลกันระหว่างสถานีด้านตะวันออกและด้านตะวันตก

ในกระบวนการที่จะยกเลิกการใช้ข่ายสาย DTE จะตัดสัญญาณ DTR ออก DCE ก็จะส่งสัญญาณเป็นสเปซไป 4 วินาที เพื่อ"บอกลา" สถานีคู่สื่อสารว่าการติดต่อได้เสร็จสิ้นแล้ว และจะตัดสัญญาณ DSR ออกเป็นการแจ้งให้ DTE จองตัวเองรู้ว่าได้ยกเลิกสายการติดต่อเรียบร้อยแล้ว

2.6.7 การให้จังหวะเวลาแก่วงจรซิงโครนัส

เราได้กล่าวถึงรูปแบบข้อมูลซิงโครนัสมาบ้างแล้ว ข้อดีของระบบซิงโครนัสก็คือได้ลดช่วงเวลาว่างระหว่างการส่งตัวอักษรลง ตัวอย่างส่วนหนึ่งของสัญญาณข้อมูลซึ่งตรงกับอักษร

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.19 DTE การควบคุมการตอบรับและหยุดการติดต่อการเรียกทางโทรศัพท์

SOS ในระบบรหัสแอสกีมีดังแสดงในรูปที่ 2.20 จะเห็นได้ว่าเพราะไม่มีขีดแบ่งระหว่างรหัสตัวอักษรอย่างชัดเจน ดังนั้นถ้าหากข้อมูลติดต่อกันมายาว ๆ เช่น 200 ตัวอักษร ก็จะมีปัญหาเกิดขึ้น ทั้งนี้เพราะไม่มีการใช้บิทเริ่มต้นเหมือนอย่างในระบบอะซิงโครนัส

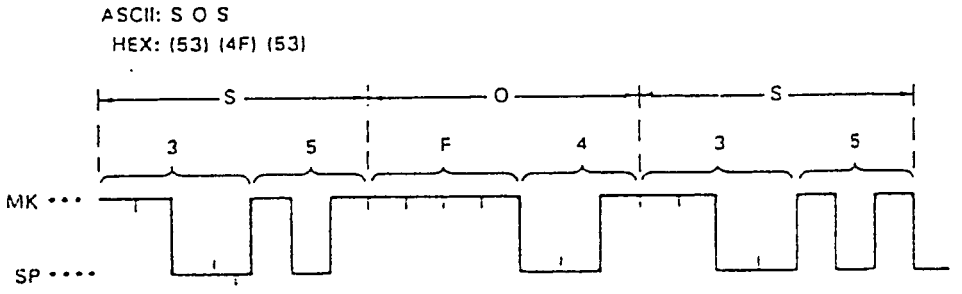
ในระบบอะซิงโครนัส บิทเริ่มต้นจะมีหน้าที่ที่สำคัญ 2 ประการคือ

1. แยกตัวอักษรออกเป็นหน่วยที่บ่งชัดออกมา

2. ทำให้เกิดความสัมพันธ์ระหว่างสัญญาณนาฬิกาที่ใช้ในชิฟรีจิสเตอร์ของภาครับกับที่จุดกึ่งกลางบิทของข้อมูลที่มีเข้ามา

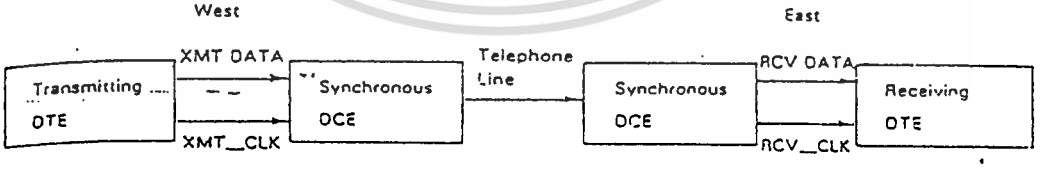
เพื่อจะให้ระบบซิงโครนัสทำงานได้จะต้องมีการแก้ปัญหา 2 ประการที่เกิดจากการตัดบิทเริ่มต้นนั้นออกให้ได้ การแก้ปัญหาอันดับแรกเป็นเรื่องของการจัดระบบโปรโตคอลให้

เอกสารนี้ ... ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.20 รูปแบบข้อมูลเนื้อความแอสกี้แบบซิงโครนัส

เหมาะสม ปัญหาที่สองจะเป็นเรื่องของซิงโครนัสและ RS-232 C อินเตอร์เฟสซิงโครนัส DTE จะมีอุปกรณ์ที่แปลงสัญญาณแบบลำดับเป็นสัญญาณแบบขนาน คล้ายกับ UART ในระบบซิงโครนัส เพราะฉะนั้นจะต้องมีการอาศัยชิฟรีจิสเตอร์และสัญญาณนาฬิกาสำหรับเลื่อนข้อมูลในการส่งและรับข้อมูลตามตารางที่ 2.18 นั้นจะมีสายสัญญาณนาฬิกาภาคส่งและภาครับอยู่ สัญญาณนาฬิกาภาคส่งจาก DTE จะใช้เลื่อนข้อมูลจากซิงโครนัส DCE ไปสู่สายการสื่อสารตามจังหวะนาฬิกานั้น และสัญญาณนาฬิกาภาครับจาก DCE จะต้องมีค่าเท่ากับสัญญาณนาฬิกาภาคส่งพอดี เพื่อเลื่อนข้อมูลเข้าสู่ DTE ภาครับได้อย่างถูกต้องที่จุดกึ่งกลางบิทพอดี ปัญหาก็เกิดขึ้นว่าจะมีการส่งสัญญาณนาฬิกาไปในสายการสื่อสารระหว่าง DCE ได้อย่างไร เพราะสายการสื่อสารมีอยู่คู่เดียว ดังในรูปที่ 2.21 ปัญหานี้ได้แก้โดยอาศัยเทคนิคการมอดดูเลชั่นที่บรรจุทั้งสัญญาณข้อมูลและสัญญาณนาฬิกาบนคลื่นพาห์คลื่นเดียวกันจากเครื่องส่งไปเครื่องรับ



รูปที่ 2.21 บล็อกไดอะแกรมของการส่งระบบซิงโครนัส

เกี่ยวกับอินเตอร์เฟสดังที่กล่าวมาแล้ว จะเห็นได้ว่ามีรูปแบบให้จัดการได้อย่างหลากหลายตามรูปที่ 2.21 สัญญาณนาฬิกาภาคส่งบนอินเตอร์เฟส RS-232 C ระหว่าง DTE และ DCE ด้านตะวันตก ต้องเป็นสัญญาณที่ใช้กับข้อมูลภาคส่งใน DTE ด้านตะวันตก ตามรูปที่ 2.21 เราใช้สัญญาณนาฬิกาภายใน DTE ทำหน้าที่ดังกล่าว ดังนั้นจะส่งสัญญาณนาฬิกานี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ผ่านสายที่ต่อขาที่ 24 คือ transmit clock(DTE ด้านส่ง)ของ RS-232 C ในบางกรณีอาจใช้สัญญาณนาฬิกาจาก DCE ควบคุมการขับข้อมูลของ DTE ก็ได้ ดังนั้นในกรณีนี้เราอาจจะใช้ขา transmit clock(DTE ด้านส่ง) ซึ่งเป็นขา 15 ของ RS-232 C แทนขา 14 ตามปกติแล้วจะมีการต่อสวิตช์ดังในรูปที่ 2.22 เพื่อให้เลือกแหล่งกำเนิดสัญญาณนาฬิกาได้

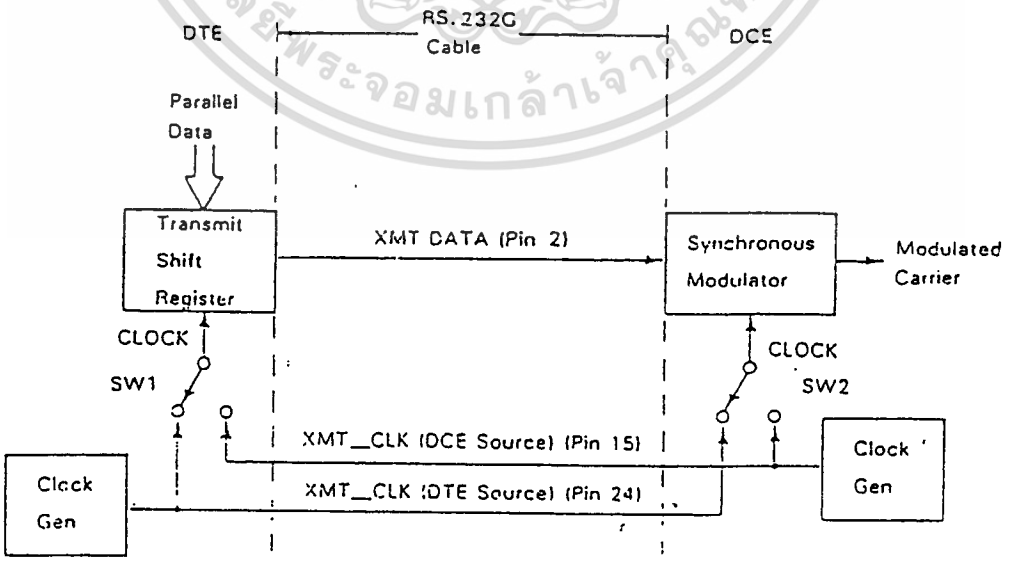
การใช้สัญญาณนาฬิกาจาก DCE มีข้อได้เปรียบอยู่บ้างเล็กน้อยคือเราสามารถทำการเปลี่ยนสวิตช์ไปยังบอดเรทที่แตกต่างกันออกไปได้ โดยที่ DTE ไม่ต้องถูกจัดระบบใหม่

2.6.8 ข้อจำกัดของ RS-232 C

ตัวประกอบหลัก 3 ประการที่จำกัดการใช้งานของอินเตอร์เฟส RS-232 C นั่นก็คือ ระยะทาง อัตราส่งข้อมูล และ สัญญาณรบกวน

2.6.8.1 ระยะทาง

ในเคเบิล RS-232 C ค่าความจุไฟฟ้า ระหว่างสายสัญญาณต่าง ๆ กับสายกราวด์เป็นพารามิเตอร์หลักที่จำกัดระยะทางการใช้เคเบิลนั้น ตามข้อกำหนดมีว่าค่าความจุไฟฟ้าที่มองไปจากวงจรขับ (drive) จะต้องไม่เกิน 2,500 pF การเพิ่มระยะทางของเคเบิ้ลก็จะเพิ่มค่าความจุไฟฟ้าที่มาเป็นภาระของวงจรขับ ตามปกติเคเบิ้ลที่ใช้อยู่จะมีความจุไฟฟ้า 2,500 pF ที่ความยาวประมาณ 50 ฟุต (15 เมตร)



รูปที่ 2.22 การเลือกขา trans clock ของ ซิงโครนัสโมเด็ม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.6.8.2 อัตราส่งข้อมูล

ข้อจำกัดที่ชัดเจนอีกอย่างหนึ่งก็คือ อัตราสัญญาณจะต้องไม่เกิน 20,000 บิท/วินาที ปัญหานี้ก็เกี่ยวข้องกับค่าความจุไฟฟ้าของสายเคเบิ้ลอีกเช่นกัน RS-232 C นั้นกำหนดค่าความต้านทานอินพุทของวงจรภาครับไว้ค่อนข้างสูง (3,000-7,000 โอห์ม) เราจะเห็นว่าค่าความต้านทานนี้จำกัดอัตราการประจุ และการคายประจุของ เคเบิลคาปาซิเตอร์ ทำให้เกิดการวน (round) ที่ขอบของสัญญาณและเกิดความผิดเพี้ยนที่อัตราข้อมูลสูง การมีขอบหน้า (leading edge) จะเป็นส่วนสำคัญที่ทำให้ค่าเวลาบิทผิดไป

ค่าจำกัดนี้แตกต่างไปจากทางปฏิบัติเล็กน้อย เพราะที่เร็วที่สุดนั้นจะทำงานในอัตรา 19,200 บิท/วินาที และส่วนใหญ่จะมีอัตราส่งสัญญาณต่ำกว่านี้ มีบางส่วนที่สูงกว่าอัตราดังกล่าวนั้น แต่วงจร RS-232 C จะไม่สามารถใช้ได้ในกรณีดังกล่าวนั้น

2.6.8.3 สัญญาณรบกวน

สัญญาณรบกวนอาจทำให้สัญญาณข้อมูลผิดพลาดได้ มีสาเหตุจากแหล่งกำเนิด 2 แบบด้วยกัน คือ

1. แหล่งกำเนิดสัญญาณเทียมได้แก่ จากมอเตอร์ไฟฟ้า คลื่นสัญญาณจากสถานีส่งวิทยุ หลอดฟลูออเรสเซนต์ เป็นต้น

2. แหล่งกำเนิดภายใน เช่น สายกราวด์สัญญาณร่วม(common return signal) เกิดไม่เป็นตัวนำที่สมบูรณ์คือ ค่าความต้านทานไม่เป็นศูนย์ การกระจายของกระแสจากสายสัญญาณต่าง ๆ ที่ใช้สายกราวด์ร่วมกันจะทำให้เกิดศักย์ไฟฟ้าในลักษณะต่าง ๆ ตกคร่อมสายกราวด์สัญญาณร่วมนี้และส่งผลไปที่ภาครับสัญญาณ ในลักษณะที่เป็นตัวก่อสัญญาณรบกวนเพิ่มขึ้น จากสาเหตุดังกล่าวทั้ง 2 ประการนี้ เมื่อสายเคเบิ้ลยาวมากขึ้นจะก่อให้เกิดสัญญาณรบกวนมากขึ้น สายที่ยาวจะทำตัวคล้ายสายอากาศคอยตรวจจับสัญญาณรบกวนที่เข้ามาและจะเกิดค่าความต้านทานสูงขึ้นทำให้เกิดสัญญาณรบกวนเพิ่มขึ้น ข้อกำหนดใหม่นี้จะได้รับการปรับปรุงทำให้ลดจุดอ่อนของ RS-232 C ลงไม่มากก็น้อย เช่น การใช้วงจรอินเตอร์เฟสสมดุลย์เข้ามาแทนวงจรอินเตอร์ไม่เฟสสมดุลย์ (unbalanced interface circuit)

วงจรสมดุลย์นั้นใช้สายคู่ตีเกลียวสำหรับสัญญาณแต่ละสัญญาณและแนวของการออกแบบวงจรภาคส่งและภาครับต่างไปจากวงจรแบบไม่สมดุลย์ วงจรแบบไม่สมดุลย์นั้นใช้สายเพียงเส้นเดียวสำหรับแต่ละสัญญาณและใช้สายกราวด์ร่วมกัน

2.7 นาฬิกาบอกเวลาที่แท้จริง (REAL TIME CLOCK)

RTC คือ REAL TIME CLOCK หรือนาฬิกาบอกเวลาที่แท้จริง (เวลาที่เราใช้กันประจำ) ให้กับไมโครโปรเซสเซอร์ โดยการเชื่อมต่อเข้ากับซีพียู ก็สามารถทำให้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ซีพียูรู้เวลาได้ทุกขณะโดยที่ซีพียูสามารถที่จะอ่านเวลาจากตัว RTC ได้ทุกครั้งที่ซีพียูต้องการ จึงทำให้ซีพียูสามารถไปทำงานอื่นๆ ได้ RTC นี้จะกินไฟน้อยมาก เมื่อเราต่อแบตเตอรี่สำรองให้กับมันจะทำให้ใช้ RTC ทำงานอยู่ได้นานเป็นเดือน แต่ก่อนที่จะให้ RTC ทำงาน เราต้องให้ ซีพียู เขียนข้อมูลเกี่ยวกับเวลาจริงให้กับมันเสียก่อน เมื่อเรียบร้อยแล้วเราเลิกจ่ายไฟเลี้ยง RTC ก็ยังทำงานเป็นนาฬิกาเหมือนทั่วๆ ไป (แต่ต้องต่อแบตเตอรี่สำรองด้วย) คือสามารถบอกเวลาได้ตั้งแต่วินาที,นาที,ชั่วโมง วันในรอบสัปดาห์(อาทิตย์-เสาร์) เดือน และปี ให้แก่ไมโครโปรเซสเซอร์ได้อย่างเที่ยงตรง โดยที่ ซีพียู เพียงแต่ต้องติดกับ RTC เหมือนกับติดต่อกับหน่วยความจำหรือพอร์ทเท่านั้น RTC มีอยู่ด้วยกันหลายเบอร์ เช่น MM58167,MM58174,MM58274,MSM6242B และในที่นี้เราจะกล่าวถึงเบอร์ MSM6242B ซึ่งสามารถที่จะปรับวันที่ให้ถูกต้องกับเดือนได้ ไม่ว่าจะเป็นเดือนที่ลงท้ายด้วย "คม" หรือลงท้ายด้วย "ยน" หรือแม้กระทั่งเดือนกุมภาพันธ์ ซึ่งปกติจะมี 28 วัน แต่ในปีอธิกสุรทินเดือนกุมภาพันธ์จะมี 29 วัน MSM 6242B ก็สามารถปรับวันที่ได้อย่างถูกต้อง

ไอซีเบอร์ MSM 6242B เป็นไอซี REAL TIME CLOCK /CALENDAR ชนิดซีมอสใช้ต่อกับบัสของไมโครโปรเซสเซอร์ได้โดยตรง มีแอดเดรสบัสและดาต้าบัส ขนาด 4 บิท มีรีจีสเตอร์ควบคุม ขนาด 4 บิท 3 ตัว คือ CD, CE และ CF ต่อร่วมกับ คริสตอล 32.768 KHz และตัวเก็บประจุเพื่อสร้างสัญญาณนาฬิกาให้กับวงจร และใช้แบตเตอรี่ขนาด 3.6 โวลท์ ในกรณีไฟฟ้าดับเวลาก็ยังคงเดินเป็นปกติ และใช้ไอซีเบอร์ 74HTC138 เป็นตัวถอดรหัสแอดเดรสเพื่อเลือกการทำงานของไอซี MSM 6242B

2.7.1 การจัดขาและหน้าที่ของขาต่าง ๆ

MSM 6242B ได้ถูกออกแบบมา ให้อินเตอร์เฟสเข้ากับ ซีพียู ในตระกูล 8085, MCS 48 และ Z80 ได้ด้วย หน้าที่ของขาต่างๆ มีดังนี้

D0-D3 (DATA BUS) : เป็นบัสข้อมูล อินพุท/เอาท์พุท ใช้ในการอ่าน และเขียนข้อมูลของรีจีสเตอร์ภายใน

A0-A3 (ADDRESS BUS) : ใช้สำหรับติดต่อกับรีจีสเตอร์ภายในของ RTC เพื่อที่จะเขียน หรืออ่านข้อมูลในตำแหน่งนั้น

ALE (ADDRESS LATCH ENABLE) : เมื่อ CS0=0 และALE เปลี่ยนจากลอจิก 1 ไปเป็นลอจิก 0 แอดเดรสจะถูกแลทช์เอาไว้ภายในตัวของ RTC

RD (READ ENABLE) : ACTIVE 0 สำหรับอ่านข้อมูลจาก RTC

WR (WRITE ENABLE) : ACTIVE 0 สำหรับเขียนข้อมูลเข้าไปยัง RTC

CS0, CS1 (CHIP SELECT 0,1) : ทำหน้าที่ ENABLE/DISABLE การทำงาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ของ ALE, RD และ WR โดยที่ CS0 และ ALE จะทำงานร่วมกัน ส่วน CS1 กับ ALE จะทำงานแยกกัน

STD.P (STANDARD PULSE OUTPUT):เป็นขาเอาท์พุทชนิด N-CH OPEN DRAIN ใช้ต่อกับขา INTERRUPT ของซีพียู

XT,XT (XTAL OSCILLATOR INPUT/OUTPUT):ต่อเข้ากับคลิสตอล 32.768KHz ถ้าต้องการป้อนความถี่จากภายนอก ทำได้โดยการป้อนความถี่ 32.768 KHz เข้าที่ขา XT

VDD : เป็นขา POWER SUPPLY +2~+6 โวลท์

GND : ขากราวด์

2.8 วงจรโซลิดสเตทรีเลย์ (SOLID STATE RELAY)

อุปกรณ์ที่เหมาะสมในการตัดต่อระบบไฟ AC 220 V คืออุปกรณ์ทาง SOLID-STATE RELAY หรือ เราเรียกสั้นๆว่า SSR ซึ่งเราสามารถนำมาใช้แทนระบบการเคลื่อนไหวทางกลในการตัดต่อวงจร

การทำงานของโซลิดสเตทรีเลย์ (SSR) เราแบ่งได้เป็น 2 ส่วนใหญ่ๆ คือ

1. ส่วนอินพุท (INPUT)

2. ส่วนเอาท์พุท (OUTPUT)

1. ภาคอินพุท (INPUT) จะเป็นตัวรับสัญญาณจากพอร์ท 8255 ซึ่งจะส่งมาให้ ไอซี 74LS245 ซึ่งเป็นบัฟเฟอร์ และมาขับให้ทรานซิสเตอร์เบอร์ BC557 ทำงานให้ไอซีออปโต้ไตรแอคเบอร์ MOC3082 เปลี่ยนสัญญาณจากไฟฟ้าให้เป็นสัญญาณทางแสงส่งไปภาคเอาท์พุทอีกทีหนึ่ง ซึ่งจะทำให้เป็นการตัดขาดระหว่างอินพุท และเอาท์พุทจริงๆ

2. ภาคเอาท์พุท (OUTPUT) จะรับสัญญาณจากไอซีเบอร์ MOC3082 ซึ่งจะเป็นการทำงานในลักษณะ ZERO SWITCHING โดยจะต่อวงจรขับไตรแอค (TRIAC) ก็ต่อเมื่อไฟ AC LINE มีค่าใกล้เคียงศูนย์โวลท์

2.8.1 ZERO SWITCHING

อีกส่วนหนึ่งที่มีความสำคัญสำหรับ การทำงานของ โซลิดสเตทรีเลย์ (SSR) ที่ใช้กับเอาท์พุทที่เป็นเอซี โดยปกติในการทำงานของสวิทช์ธรรมดาต่างๆ อาจจะมีการทำงานที่ไม่สมควรคือ เปิดวงจรในขณะที่สัญญาณของไฟเอซี อาจจะอยู่ในค่าสูงสุด ซึ่งทำให้เกิดกระแสและสัญญาณรบกวนที่สูงมาก โดยเฉพาะถ้าต้องต่อร่วมกับวงจรทางคอมพิวเตอร์แล้ว ก็อาจจะทำให้เครื่องเกิดอาการเสียหรือทำงานผิดปกติก็ได้ฉะนั้น ZERO SWITCHING ก็คือเป็นการทำงานให้เปิดวงจรในขณะที่สัญญาณเอซีอยู่ในระยะ 0 โวลท์ แต่ในการใช้งานจริงจะมี

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ค่ามากกว่า 0 โวลท์เล็กน้อย เพื่อให้อุปกรณ์เซมิคอมตัดเตอร์ที่ต่ออยู่ สามารถทำงานเปิดวงจรได้ เช่น ตัวอย่างเราต้องการให้วงจรทำงานเปิดวงจรที่ 15 โวลท์จาก 220 โวลท์ ก็จะเป็นมุมทางสัญญาณเอซี 2.77 องศา ตามสูตร

$$\text{มุมทางสัญญาณเอซี} = \text{SIN}^{-1} \frac{\text{Zsw.max}}{\text{Line Vrms} \times \sqrt{2}}$$

$$= \text{SIN}^{-1} \frac{15}{220*1.414}$$

$$= 2.77 \text{ องศา}$$

หรือในทางกลับกันเราอาจจะคำนวณระยะเวลาหลังจาก 0 โวลท์ ที่เกิดขึ้นแล้วนานเท่าไรจึงจะเปิดวงจร

$$\text{เวลาในการเปิดวงจร (T)} = \frac{1/2 \times \text{ส่วนกลับของความถี่ไฟสลับ}}{1/2 \times \text{มุมของสัญญาณต่อหนึ่งรูปคลื่น}} \times \text{มุมทางสัญญาณเอซี}$$

$$= \frac{1/2\ (1/50)}{1/2 \times 360} \times 2.77$$

$$= 0.153 \text{ ms}$$

2.8.2 ข้อดีของโซลิดสเตทรีเลย์

1. การทำงานจะต่อวงจรที่โวลท์เตจ (VOLTAGE) ขณะเป็นศูนย์หรือใกล้จุดศูนย์
2. มีอายุการทำงานที่ยาวนานมาก
3. ไม่มีเสียงรบกวนเกิดขึ้นในเวลาทำงาน
4. สามารถต่อร่วมกับระบบไมโครโปรเซสเซอร์ได้โดยง่าย
5. สามารถตัดต่อวงจรได้อย่างรวดเร็ว
6. ไม่มีส่วนเคลื่อนไหวทางกลในการทำงาน
7. ไม่เกิดอาการ BOUNCE ที่หน้าสัมพันธ์ซึ่งเป็นตัวก่อให้เกิดสัญญาณรบกวน
8. ไม่เป็นตัวก่อให้เกิดสัญญาณรบกวนต่อระบบไฟฟ้าที่ตัวมันต่ออยู่

2.8.3 ข้อเสียของโซลิดสเตทรีเลย์

1. ไม่สามารถใช้กับวงจรที่มี VOLTAGE ที่สูงมากๆได้
2. ไม่สามารถใช้โซลิดสเตทรีเลย์ (SSR) สำหรับ AC LINE หรือ DC LINE ได้ในตัวเดียวกัน

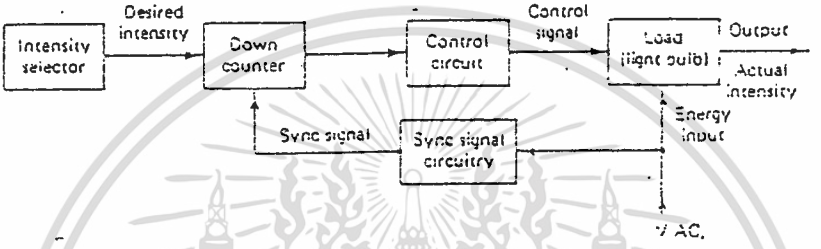
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3. เวลาใช้งานจะเกิดความร้อนขึ้นที่ตัวโซลิดสเตทรีเลย์ (SSR) นั้นๆ จำเป็นต้องติดตั้งแผ่นระบายความร้อน (HEAT SINK) ช่วยในการทำงาน

4. ตัวโซลิดสเตทรีเลย์ (SSR) จะเสียโดยง่ายถ้าเกิดการลัดวงจรขึ้นจึงควรระวัง

2.9 วงจรควบคุมความเข้มของแสงไฟด้วยสัญญาณดิจิตอล

2.9.1 บล๊อกไดอะแกรมของวงจรควบคุมความเข้มของแสงไฟด้วยสัญญาณดิจิตอล



รูปที่ 2.23 บล๊อกไดอะแกรมของวงจรควบคุมความเข้มของแสงไฟด้วยสัญญาณดิจิตอล

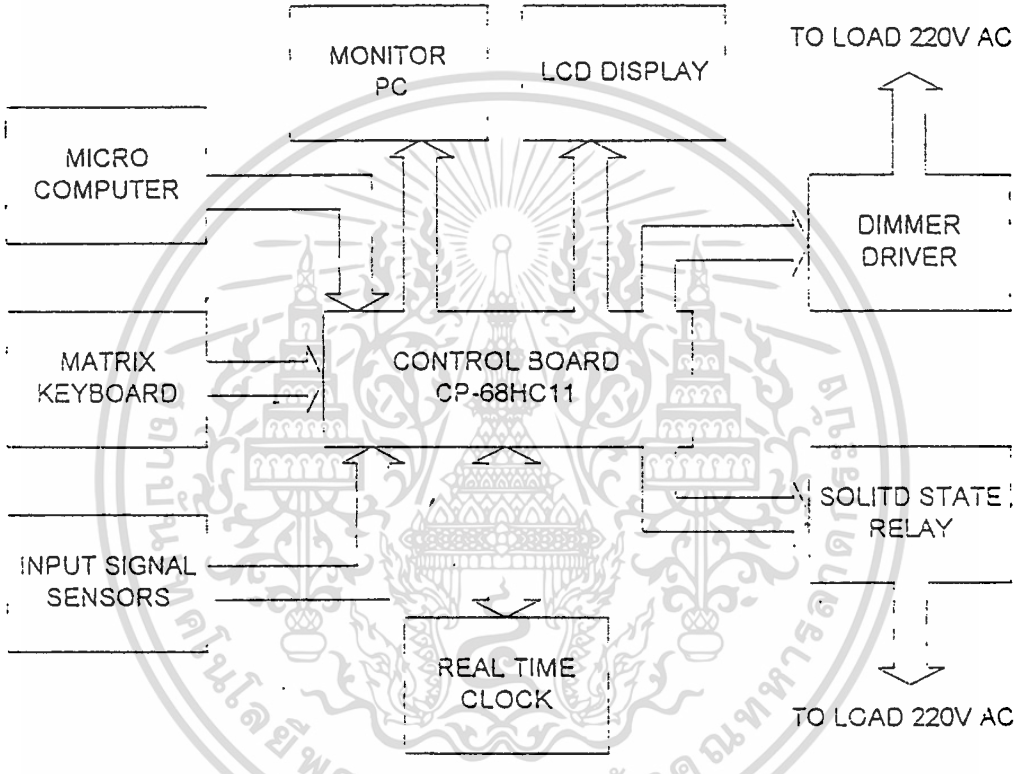
2.9.2 การทำงานของวงจรควบคุมความเข้มของแสงไฟด้วยสัญญาณดิจิตอล

จากบล๊อกไดอะแกรม เราจะนำสัญญาณดาต้า D0-D7 จากไมโครคอนโทรลเลอร์ โดยการกำหนดเป็นรหัสบีซีดี (BCD) ส่งเข้าวงจรนับลง (DOWN COUNTER) ซึ่งใช้ไอซีเบอร์ 40102B เป็นตัวนับค่ารหัสบีซีดี (BCD) ทำการควบคุมระดับแสงสว่าง โดยถ้ารหัสบีซีดี (BCD) มีค่าต่ำ จะให้ระดับความสว่างสูง ในทำนองกลับกัน ถ้ารหัสบีซีดี (BCD) มีค่าสูง จะให้ความสว่างต่ำ โดยจะส่งเป็นสัญญาณพัลซ์เข้าไปที่วงจรควบคุม ซึ่งเป็นวงจรโซลิดสเตทรีเลย์ควบคุมด้วยไอซีออปโต้ไตรแอค ไปขับไตรแอคเพื่อควบคุมหลอดไฟอีกทีหนึ่ง และจะมีวงจรซิงค์สัญญาณทางเอาท์พุทเข้ามาควบคุมการทำงานของไอซี 40102B ให้ควบคุมระดับความสว่างของหลอดไฟฟ้าให้คงที่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 3
การออกแบบและการสร้าง

3.1 บล็อกไดอะแกรมของระบบตรวจจับและควบคุมด้วยเครื่องไมโครคอมพิวเตอร์



รูปที่ 3.1 บล็อกไดอะแกรมของระบบตรวจจับและควบคุมด้วยเครื่องไมโครคอมพิวเตอร์

จากรูปที่ 3.1 จะเห็นว่าผู้ใช้สามารถใช้งานระบบ ได้ 2 ลักษณะ คือ

3.1.1 ใช้งานบนเครื่องไมโครคอมพิวเตอร์ (PC)

ใช้เครื่องไมโครคอมพิวเตอร์เป็นตัวควบคุมการทำงานส่วนต่าง ๆ ของระบบ โดยการส่งสัญญาณผ่านพอร์ทอนุกรม RS-232C สื่อสารกับบอร์ดไมโครคอนโทรลเลอร์ 68HC11 ซึ่งใช้โปรแกรมประเภท VISUAL ในการเขียนโปรแกรมสำหรับวินด์โดวส์ ทำหน้าที่ในการควบคุมระบบ ซึ่งแสดงผลการใช้งานบนไมโครซอฟท์วินด์โดวส์ของเครื่องคอมพิวเตอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.1.2 ใช้งานบนชุดสาธิตระบบตรวจจับและควบคุม (STAND ALONE)

ใช้เมตริกซ์คีย์บอร์ด ขนาด 4x4 บนตัวเครื่องที่สร้างขึ้น ทำหน้าที่ในการติดต่อกับระบบ และควบคุมการทำงานส่วนต่างๆ ของระบบ ซึ่งบนคีย์บอร์ดของเครื่องที่สร้างขึ้นจะประกอบด้วยคีย์ต่างๆ เพื่ออำนวยความสะดวกในการใช้งาน และแสดงผลการใช้งานบนจอแอลซีดี ขนาด 20 ตัวอักษร x 2 แถว

3.2 ส่วนประกอบโดยทั่วไป

ระบบตรวจจับและควบคุมด้วยเครื่องไมโครคอมพิวเตอร์ มีส่วนประกอบต่างๆ ดังนี้

3.2.1 ส่วนชุดควบคุม (CP-68HC11 BOARD)

เป็นส่วนที่ควบคุมการทำงานในส่วนต่างๆ ของระบบตามโปรแกรมมอนิเตอร์ที่เขียนขึ้นโดยใช้บอร์ด CP-68HC11 ซึ่งเป็นไมโครคอนโทรลเลอร์

3.2.2 ส่วนวงจรนาฬิกาบอกเวลาที่แท้จริง (REAL TIME CLOCK)

ทำหน้าที่ในการบอก วัน, เดือน, ปี และเวลา ให้กับซีพียู ซึ่งเราสามารถเขียนและอ่านข้อมูลเกี่ยวกับเวลาจากวงจร RTC ได้

3.2.3 ส่วนเมตริกซ์คีย์บอร์ด (MATRIX KEYBOARD)

เป็นส่วนที่รับค่า หรือเป็นอินพุทที่ใช้สั่งงานของระบบควบคุม โดยค่าที่รับนี้จะส่งผ่านพอร์ท C ของ 8255 เข้าไปที่ซีพียู

3.2.4 ส่วนการแสดงผลการใช้งานบนจอแอลซีดี (LCD DISPLAY)

ทำหน้าที่เป็นมอนิเตอร์ของระบบควบคุม ซึ่งใช้ จอแอลซีดีโมดูล ขนาด 20 ตัวอักษร x 2 แถว ติดต่อกับบอร์ด CP-68HC11

3.2.5 ส่วนวงจรถอดรหัสพอร์ทภายนอก (DECODER CIRCUIT)

เป็นส่วนสร้างสัญญาณสำหรับการเลือกการติดต่อกับอุปกรณ์ภายนอกที่นอกเหนือไปจากการสร้างสัญญาณเลือกอุปกรณ์บนชุดควบคุม ซึ่งส่วนนี้จะใช้สำหรับถอดรหัสเพื่อเลือกช่องการทำงานของวงจรหรี่ไฟ ซึ่งมี 4 ช่อง โดยจะติดต่อผ่าน Z80 BUS บนบอร์ด CP-68HC11

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.2.6 ส่วนวงจรควบคุมความเข้มของแสงไฟด้วยไมโครคอนโทรลเลอร์

ทำหน้าที่เป็นวงจรหรี่ไฟ ซึ่งใช้สำหรับติดต่อระหว่างอุปกรณ์ไฟฟ้าจากภายนอกกับส่วนวงจรถอดรหัสพอร์ทภายนอก โดยออกแบบไว้ทั้งหมด 4 ช่อง และสามารถพักข้อมูลที่กำหนดระดับแสงสว่างไว้แต่ละช่องด้วย ไอซีบัฟเฟอร์ 74LS373

3.2.7 ส่วนวงจรโซลิดสเตทรีเลย์ (SOLID STATE RELAY)

ทำหน้าที่เป็นสวิทช์อิเล็กทรอนิกส์ ใช้ในการตัดต่อวงจรไฟฟ้ากระแสสลับ 220 โวลท์ เพื่อควบคุมการเปิดและปิดอุปกรณ์ไฟฟ้า 220 โวลท์ ซึ่งออกแบบไว้ทั้งหมด 12 ช่อง ใช้สำหรับควบคุมแบบ เปิด/ปิด ธรรมดา หรือตั้งเวลา เปิด/ปิด 8 ช่อง และใช้สำหรับแสดงผลการตรวจจับสัญญาณจากภายนอก 4 ช่อง

3.2.8 ส่วนวงจรดีซีโวลท์มิเตอร์ (DC VOLTMETER CIRCUIT)

ทำหน้าที่ในการวัดระดับแรงดันไฟฟ้ากระแสตรง ซึ่งออกแบบสำหรับวัดแรงดัน 0-5 โวลท์ ไว้ 3 ช่อง และสำหรับวัดแรงดัน 0-500 โวลท์ ซึ่งจะเปลี่ยนย่านการวัดได้ 3 ย่าน คือ 0-5 โวลท์, 0-50 โวลท์ และ0-500 โวลท์ แบบอัตโนมัติ ไว้ 1 ช่อง และจะแสดงผลเป็นตัวเลขบนจอแอลซีดี โดยสัญญาณที่ได้จะถูกแปลงให้เป็นสัญญาณดิจิตอลด้วยวงจรเอทูดี บนบอร์ด CP-68HC11

3.2.9 ส่วนของวงจรตรวจจับ (SENSORS CIRCUIT)

ทำหน้าที่ในการตรวจจับแรงดันสัญญาณอนาลอกจากภายนอกเข้ามาที่ วงจรเอทูดีของบอร์ด CP-68HC11 ซึ่งออกแบบไว้ 4 แชลแนล และจะแสดงผลการตรวจจับสัญญาณที่โซลิดสเตทรีเลย์ตามค่าของแรงดัน 0-5 โวลท์ที่ผู้ใช้สามารถโปรแกรมไว้ได้แต่ละแชลแนล

3.2.10 ส่วนของแหล่งจ่ายไฟ (POWER SUPPLY)

ทำหน้าที่แปลงแรงดันไฟเอซี 220 โวลท์ เป็นแรงดันไฟกระแสตรง 12 โวลท์ แล้วผ่านวงจรเรคกูเลเตอร์จ่ายไฟกระแสตรง 5 โวลท์ส่งไปเลี้ยงวงจรในส่วนต่าง ๆ

3.3 การออกแบบและการทำงานในส่วนต่างๆ

3.3.1 ส่วนชุดควบคุม

คุณสมบัติทางเทคนิคของบอร์ดไมโครคอนโทรลเลอร์ CP-68HC11 ได้แก่

ไมโครคอนโทรลเลอร์ : MC68HC11A1FN

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หน่วยความจำ : แรม 6264 (8 กิโลไบต์) หรือ 62256 (32 กิโลไบต์) ใช้งานที่แอดเดรส $2000-$9FFF อีพรอม 2764 ขนาด 8 กิโลไบต์ หรือ 27256 (32 กิโลไบต์) แต่จะใช้งานได้ สูงสุด 24 กิโลไบต์ มีช่วงแอดเดรสคือ $A000-$FFFF มีแรมภายใน 256 ไบต์และอีอีพรอมขนาด 512 ไบต์

พอร์ทอินพุทเอาท์พุท : ขนาด 24 บิท โดยใช้ชิพ 8255

พอร์ทเครื่องพิมพ์ : มีจำนวน 1 พอร์ท

พอร์ทเชื่อมต่อคีย์บอร์ด : แบบ Matrix ขนาด 3x4 หรือ 4x4 จำนวน 1 พอร์ท

พอร์ทแอลซีดี : มี 1 พอร์ท ใช้ได้ทั้งแบบตัวอักษรและแบบกราฟฟิก

พอร์ทตั้งเวลา : มีพอร์ทอินพุท 3 ช่อง, พอร์ทเอาท์พุท 4 ช่อง และยังมีพอร์ทอินพุทเอาท์พุทอีก 4 ช่อง

พอร์ท A/D : มีพอร์ทแปลงสัญญาณอนาล็อกเป็นดิจิตอล 8 บิท 8 ช่อง

พอร์ทอนุกรม : มีพอร์ทอนุกรม 2 ช่อง แบบ SCI และ SPI

สัญญาณนาฬิกา : ใช้สัญญาณนาฬิกา 8 MHz

ขนาดของชุดควบคุม : 9.3 x 14 เซนติเมตร

3.3.1.1 การทำงานของชุดควบคุม

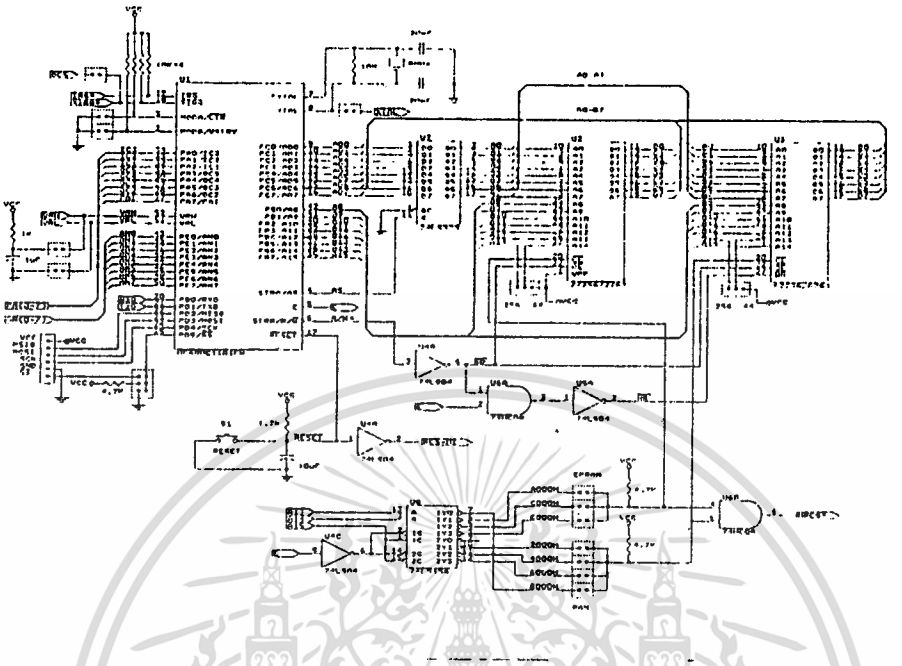
หัวใจการในการทำงานของบอร์ดขึ้นอยู่กับ U1 โดยไมโครคอนโทรลเลอร์ 68HC11 สามารถเลือกโหมดการทำงานให้แก่ U1ได้โดยการเลือกจั๊มเปอร์ที่ขา 2 และ 3 ของ U1 ดังรูปที่ 3.2

จากรูปที่ 3.2 ไมโครคอนโทรลเลอร์ 68HC11 นี้สามารถเชื่อมต่อกับหน่วยความจำทั้งอีพรอมและแรมได้โดยการเลือกโหมดการทำงานเป็นแบบมัลติเพล็กซ์ขยาย ในการเลือกโหมดนี้ขา PC0-PC7 ทำหน้าที่เป็นบัสแอดเดรสไบต์ต่ำสลับกับการเป็นบัสข้อมูล โดยมี U7 เบอร์ 74LS373 เป็นตัวแยกสัญญาณหรือเรียกว่าเป็นตัวดีมัลติเพล็กซ์ก็ได้ ซึ่งจะต้องทำงานร่วมกับสัญญาณ AS ที่ขา 4 ของ U1 ด้วย ส่วนขา PB0-PB7 จะเป็นขาของบัสแอดเดรสในไบต์สูงคือ A8-A15

U4B, U6A และ U5A จะทำหน้าที่เป็นวงจรสร้างสัญญาณ WR เพื่อเขียนข้อมูลลงในหน่วยความจำแรม U3 ซึ่งสามารถเลือกได้ว่าเป็นแรมเบอร์ 6264 (8 กิโลไบต์) หรือว่าเบอร์ 62256 (32 กิโลไบต์)ก็ได้ เอาท์พุทจาก U4B ยังใช้เป็นสัญญาณ RD เพื่อทำการส่งผ่านข้อมูลจากอีพรอม U2 และแรม U3

ในวงจรถอดรหัสแอดเดรสของหน่วยความจำจะเป็นหน้าที่ของ U8 เบอร์ 74LS156 สามารถเลือกแอดเดรสได้ตั้งแต่ $2000-$9FFF นอกจากนั้นยังมีการนำสัญญาณจากที่ U8

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.2 วงจรเชื่อมต่อ 68HC11 กับหน่วยความจำภายนอก

มาจัดเป็นสัญญาณ MREQ สำหรับบัสของระบบเพื่อประโยชน์ในการเชื่อมต่อกับอุปกรณ์อื่น ๆ ของ Z-80

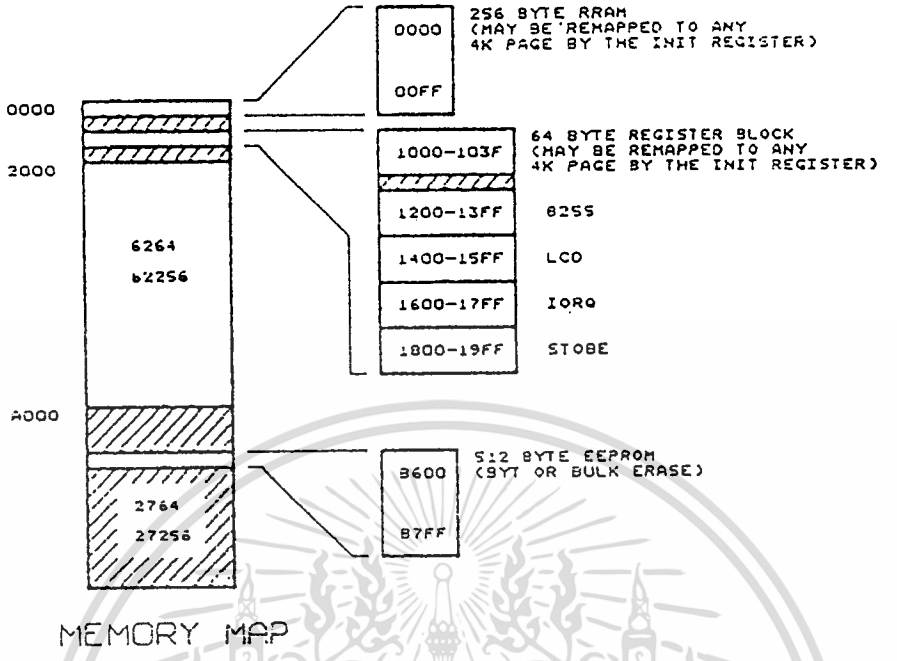
วงจรกำเนิดสัญญาณนาฬิกาของ U1 จะใช้ผลึกคริสตอลขนาด 8 MHz เป็นตัวกำเนิดความถี่ และสามารถนำสัญญาณนาฬิกานี้ออกไปใช้งานกับฮาร์ดแวร์ภายนอกได้

3.3.1.2 การจัดหน่วยความจำของบอร์ด

ที่บอร์ด CP-68HC11 จะมีการจัดสรรหน่วยความจำ ดังแสดงในรูปที่ 3.3 ส่วนแรกมีขนาด 256 กิโลไบต์ ที่แอดเดรส $0000-$00FF จัดสรรไว้สำหรับแรมภายในชิพ โดยที่แอดเดรส $1000-$103F เป็นพื้นที่ของรีจิสเตอร์ควบคุมขนาด 64 กิโลไบต์ ส่วนที่ขา 3 แอดเดรสตั้งแต่ $12000-$13FF เป็นแอดเดรสของพอร์ท 8255 ใช้สำหรับแอดเดรสของพอร์ทแอลซีดีจัดสรรไว้ที่ 1400-$15FF

ที่แอดเดรส $1600-$17FF เป็นของสัญญาณ IORQ และแอดเดรส $1800-$19FF เป็นของสัญญาณสโตรบ ในแอดเดรสระหว่าง $B600-$B7FF เป็นหน่วยความจำอีอีพรอมภายในชิพ 68HC11 สุดท้ายและแอดเดรส $A000-$FFFF เป็นพื้นที่สำหรับหน่วยความจำอีพรอมภายนอก ซึ่งก็คือ U2 นั่นเอง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



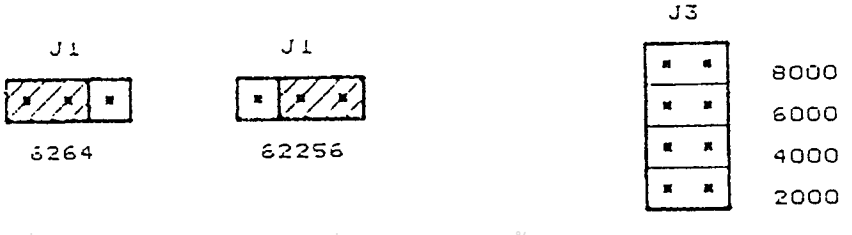
รูปที่ 3.3 การจัดหน่วยความจำของบอร์ด

3.3.1.3 ระบบบัส

เป็นคอนเนคเตอร์ขนาด 40 ขาใช้ในการขยายระบบ โดยมีการจัดขาสัญญาณต่าง ๆ ให้ใกล้เคียงกับบัส 40 ขาของ Z-80

3.3.1.4 การเลือกหน่วยความจำแรม

สามารถเลือกแรมได้ 2 เบอร์คือ 6264 ขนาด 8 กิโลไบต์ และ 62256 ขนาด 32 กิโลไบต์ โดยเลือกจั๊มเปอร์ J1 สำหรับการกำหนดตำแหน่งของหน่วยความจำสามารถเลือกได้ที่จั๊มเปอร์ J3 ซึ่งการใส่จั๊มเปอร์ 1 ตัว สามารถอ้างตำแหน่งในหน่วยความจำได้ 8 กิโลไบต์ ถ้าใส่ครบทั้ง 4 ตัวก็สามารถอ้างตำแหน่งได้ 32 กิโลไบต์ โดยที่ตำแหน่งของหน่วยความจำจะอยู่ในช่วง $2000 - $9FFF



รูปที่ 3.4 การเลือกจั๊มเปอร์ของแรม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.3.1.5 การเลือกหน่วยความจำอีพรอม

เราสามารถที่จะเลือกใช้อีพรอมได้ 2 เบอร์ คือเบอร์ 2764 (8 กิโลไบต์)และเบอร์ 27256 (32 กิโลไบต์)โดยการเลือกที่จั๊มเปอร์ J2 สำหรับการกำหนดตำแหน่งของหน่วยความจำสามารถเลือกได้ที่จั๊มเปอร์ J4

ซึ่งการใส่จั๊มเปอร์ 1 ตัว สามารถอ้างตำแหน่งในหน่วยความจำได้ 8 กิโลไบต์ ถ้าใส่ครบทั้ง 3 ตัว จะสามารถอ้างตำแหน่งในหน่วยความจำได้ 24 กิโลไบต์ โดยที่ตำแหน่งหน่วยความจำจะอยู่ในช่วง $A000-$FFFF โดยการเลือกจั๊มเปอร์แสดง ดังรูปที่ 3.5

รูปที่ 3.5 การเลือกจั๊มเปอร์ของอีพรอม

3.3.1.6 แอดเดรสของพอร์ทอินพุทเอาท์พุท 8255

เป็นคอนเนคเตอร์ 34 ขา ที่ต่อออกมาจากพอร์ทของ 8255 สำหรับแอดเดรสติดต่อกับพอร์ทของ 8255 มีรายละเอียดดังรูปที่ 3.6

PORT A	$1200
PORT B	$1201
PORT C	$1202
CONTROL PORT	$1203

รูปที่ 3.6 แอดเดรสของพอร์ทอินพุทเอาท์พุท 8255

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

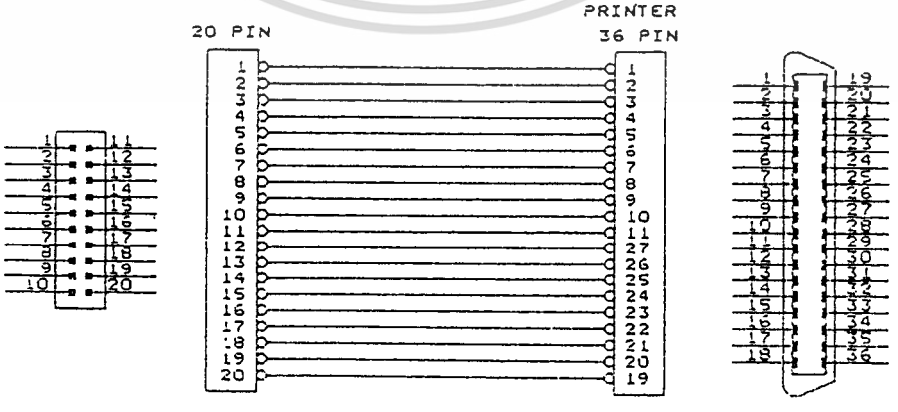
3.3.1.7 การเชื่อมต่อกับคีย์บอร์ด

เป็นคอนเนคเตอร์ขนาด 10 ขา ที่ต่อมาจากพอร์ท C ของ 8255 สามารถต่อเข้ากับคีย์บอร์ดขนาด 4x4 หรือ 3x4 ได้โดยการเลือกที่จั๊มเปอร์ J5 ในกรณีที่เลือกคีย์บอร์ดเป็นแบบ 3x4 ขา PC0 ของ 8255 จะถูกนำมาใช้เป็นสัญญาณ BUSY เพื่ออ่านสถานะของเครื่องพิมพ์ ส่วนในกรณีที่เลือกคีย์บอร์ดเป็นแบบ 4x4 จะไม่สามารถเชื่อมต่อกับเครื่องพิมพ์ได้ เนื่องจากขา PC0 ของ 8255 นี้จะถูกต่อเข้ากับคอนเนคเตอร์ 10 ขา ของคีย์บอร์ดแทนการเลือกจั๊มเปอร์ ดังแสดงในรูปที่ 3.7

รูปที่ 3.7 การเลือกลักษณะของคีย์บอร์ดโดยใช้ J5

3.3.1.8 แอดเดรสของพอร์ทเครื่องพิมพ์

เป็นคอนเนคเตอร์ขนาด 20 ขา ใช้ต่อกับเครื่องพิมพ์ธรรมดาทั่วไป สำหรับสัญญาณ STROBE นั้นจะใช้พอร์ทเอาท์พุทของ 74HC138 ซึ่งได้จากการถอดรหัสแอดเดรสของพอร์ทโดยการส่งข้อมูลอะไรก็ได้ส่งออกไปที่พอร์ทแอดเดรสที่ $1800 ก็จะเป็นการสโตรบข้อมูล สำหรับแอดเดรสที่ใช้ติดต่อกับเครื่องพิมพ์มีรายละเอียดดังนี้



รูปที่ 3.8 ลักษณะการเชื่อมต่อ คอนเนคเตอร์ 20 ขา เข้ากับ คอนเนคเตอร์เซนทรอนิกส์ขนาด 36 ขา ของเครื่องพิมพ์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.3.1.9 การเชื่อมต่อกับจอแสดงผลแบบแอลซีดี

เป็นคอนเนคเตอร์ขนาด 20 ขา สามารถใช้กับโมดูลแอลซีดีได้ 2 แบบ คือ แบบตัวอักษรและแบบกราฟฟิกโดยเลือกที่จั๊มเปอร์ J6 ดังรูปที่ 3.9 และสามารถปรับความสว่างของจอแอลซีดีได้โดยการปรับที่ตัวต้านทานปรับค่าได้ 10 กิโลโอห์ม แอดเดรสที่ใช้ในการติดต่อกับแอลซีดีมีรายละเอียดสำหรับแอลซีดีแบบกราฟฟิกดังนี้

WRITE DATA INSTRUCTION	$1400
READ BUSY FLAG ANND ADDRESS	$1400
WRITE DATA	$1401
READ DATA	$1401

Page 1 อยู่ที่แอดเดรส $1400-$1401

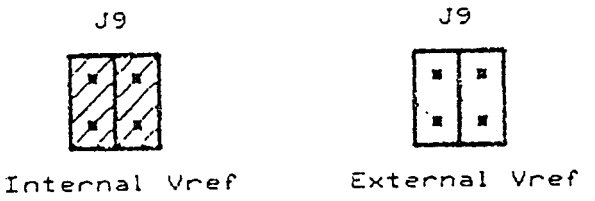
Page 2 อยู่ที่แอดเดรส $1402-$1403

รูปที่ 3.9 การเลือก J6 เพื่อเลือกชนิดของโมดูลแอลซีดี

3.3.1.10 พอร์ทเชื่อมต่อวงจร A/D

มีคอนเนคเตอร์ขนาด 14 ขา ซึ่งต่อออกมาจากขา PE0-PE7 ของ 68HC11 เป็นอินพุทของวงจรแปลงสัญญาณอนาล็อกให้เป็นดิจิตอล (A/D) ขนาด 8 บิท 8 ช่องโดยจะใช้แรงดันอ้างอิงภายในบอร์ด 5 โวลท์ สามารถใช้แรงดันอ้างอิงจากภายนอกได้ ให้แรงดันนี้อยู่ในช่วง 2.5-5 โวลท์โดยการถอดจั๊มเปอร์ J9

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



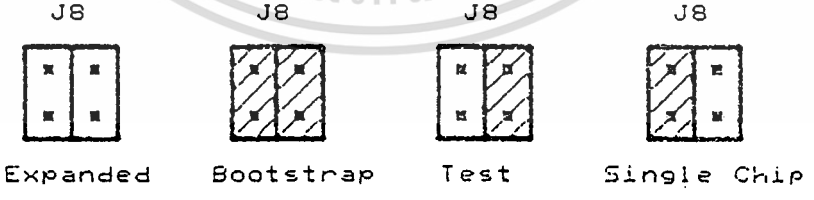
รูปที่ 3.10 การต่อแรงดันอ้างอิงโดย A/D โดย J9

3.3.1.11 พอร์ทตัวตั้งเวลาและตัวนับ

เป็นคอนเนคเตอร์ขนาด 10 ขา ต่อออกมาจาก PA0-PA7 ของ 68HC11 สามารถนำจะไปประยุกต์ใช้งานอื่น ๆ ได้หลายอย่างเช่น การนับความถี่, นับจำนวนพัลซ์คาบเวลา กำเนิดสัญญาณนาฬิการูปสี่เหลี่ยมหรือจะใช้เป็นพอร์ทอินพุทเอาท์พุทก็ได้

3.3.1.12 การเลือกโหมดการทำงาน

มีคอนเนคเตอร์ 4 ขา เพื่อที่จะใช้เลือกโหมดการทำงานของไมโครคอนโทรลเลอร์ ซึ่งสามารถกำหนดโหมดการทำงานได้ 4 โหมด คือ โหมดชิงเกิลชิพ (Single chip), โหมดมัลติเพล็กซ์ขยาย (Expanded multiplex), โหมดบูตสแตร์ปพิเศษ (Special-bootstrap) และโหมดทดสอบพิเศษ (Special test) แต่การทำงานโดยทั่วไปจะอยู่ที่โหมดมัลติเพล็กซ์ขยายซึ่งเป็นการติดต่อกับหน่วยความจำภายนอก การเลือกโหมดจะกระทำที่จั๊มเปอร์ J8 ดังรูปที่ 3.11

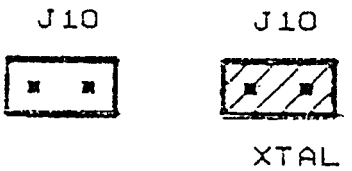


รูปที่ 3.11 การเลือกโหมดการทำงานของ 68HC11

3.3.1.13 การนำสัญญาณนาฬิกาออกไปใช้งานภายนอก

ในกรณีที่ต้องการสัญญาณความถี่ 8 MHz ออกไปยังคอนเนคเตอร์ 40 ขา เพื่อเชื่อมต่อระบบฮาร์ดแวร์ภายนอกจะต้องลัดจั๊มเปอร์ J10 ด้วย การนำสัญญาณนาฬิกาออกไปใช้งานก็เพื่อให้ ระบบฮาร์ดแวร์ภายนอกกับบอร์ดมีจังหวะการทำงานที่สัมพันธ์สอดคล้องกัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.12 การเลือกจั๊มเปอร์ J10

3.3.1.14 การต่อจั๊มเปอร์ JOC5

เป็นคอนเนคเตอร์ 2 ขา ซึ่งต่ออยู่ระหว่างขา OC5 กับขา XIRQ ของ 68HC11 ซึ่งจะถูกใช้ ในกรณีที่ต้องการทำประมวลผลแบบ Single Step เป็นการทำ Single Step โดยจะใช้ฮาร์ดแวร์ซึ่งจะทำให้เกิดอินเตอร์รัพต์ขึ้นที่ขา XIRQ ดังนั้นการนำขา OC5 ไปใช้งานอื่นๆ ควรนำจั๊มเปอร์นี้ออกได้

รูปที่ 3.13 การเลือกจั๊มเปอร์ J11

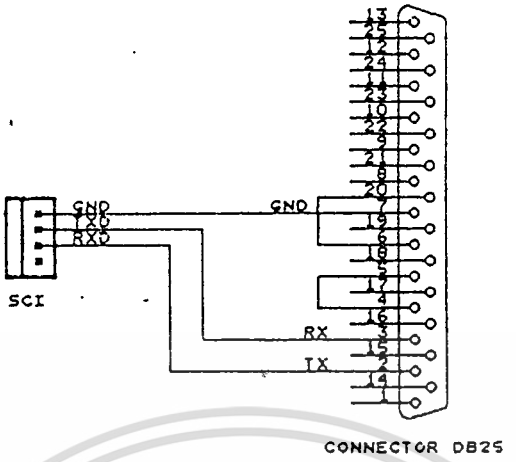
3.3.1.15 การส่งผ่านข้อมูลอนุกรมและพอร์ทสำหรับสื่อสารข้อมูลอนุกรม (SCI)

การเชื่อมต่อกับพอร์ทอนุกรมของเครื่องคอมพิวเตอร์ผ่านทาง RS-232 จะใช้ U9 เบอร์ MAX232 เป็นผู้ทำหน้าที่นี้ ถ้าต้องการเชื่อมต่อผ่านพอร์ท RS-232 ให้เปลี่ยน U9 จาก MAX232 เป็น 75176 แทน และต้องต่อตัวต้านทาน R7 ด้วย โดยค่าของ R7 จะขึ้นอยู่กับความยาวของสายที่ต่อใช้งานกับ พอร์ท RS-422 โดยปกติพอร์ท RS-422 สามารถจะใช้สายได้ยาวสูงสุด 4,000 ฟุต หรือประมาณ 1.2 กิโลเมตร และอัตราบอดในการส่งข้อมูลปกติแล้วจะใช้อัตราบอดเท่ากับ 9,600 บิทต่อวินาทีที่ระยะทาง 30.5 เมตร ค่า R7 จะอยู่ที่ 100 โอห์ม ถ้าระยะทางมากขึ้นค่า R7 ก็จะเพิ่มขึ้น ในอัตราบอดที่เท่าเดิม

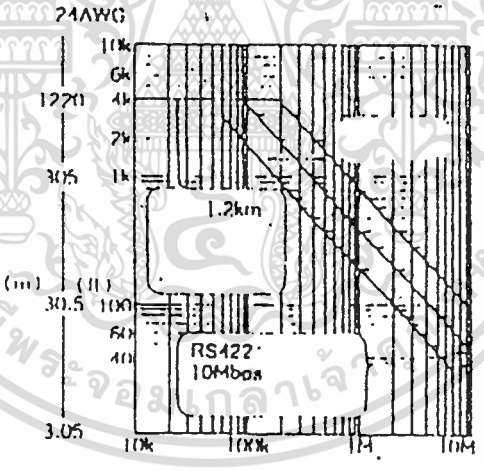
คอนเนคเตอร์ขนาด 4 ขาที่ต่อเข้ากับ MAX232 เป็นการเชื่อมต่อกับพอร์ท RS-232 เพื่อใช้งานสื่อสารข้อมูลอนุกรม เราสามารถเชื่อมต่อกับพอร์ท RS-232 ได้ โดยการถอด MAX232 ออกแล้วใส่ 75176 แทนแล้วกำหนดค่า Rz ใหม่ ดังรูปที่ 3.15

3.3.1.16 พอร์ทสำหรับเชื่อมต่ออุปกรณ์อนุกรม (SPI)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.14 การเชื่อมต่อ 68HC11 เพื่อส่งข้อมูลอนุกรม (SCI) โดยการต่อพอร์ทอนุกรมกับคอนเนคเตอร์ DB25



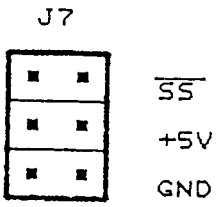
รูปที่ 3.15 กราฟสำหรับการเลือกค่าความต้านทาน Rz

เป็นพอร์ทใช้งานด้านเชื่อมต่อกับอุปกรณ์อนุกรม โดยจั๊มเปอร์ J7 เป็นตัวเลือกว่าจะใช้ต่อกับขา SS ของ 68HC11ไปต่อเข้ากับแรงดัน +5 โวลท์, กราวด์ (GND) หรือต่อไปคอนเนคเตอร์

3.3.1.17 การขยายพอร์ทเพิ่มเติมบนชุดควบคุม

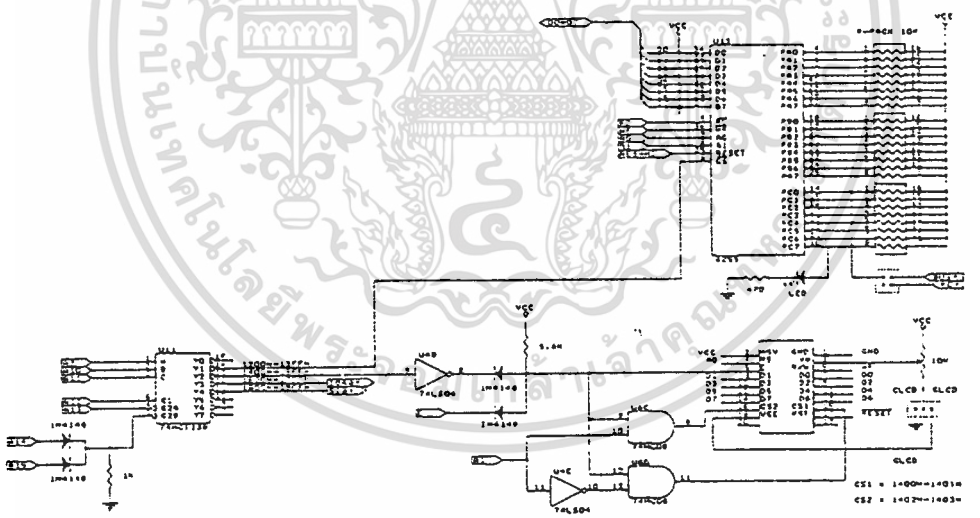
ในส่วนของการขยายพอร์ทเพิ่มเติม U11 ในเบอร์ 74HCT138 จะทำหน้าที่ถอดรหัส

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.16 จั๊มเปอร์ J7

แอดเดรสพอร์ทให้แก่ 68HC11 โดยพอร์ทเพิ่มเติมนั้นจะใช้ U12 เบอร์ 8255 นอกจากนั้น U11 ยังถอดรหัสแอดเดรสสำหรับพอร์ทที่เชื่อมต่อกับแอลซีดีโมดูลด้วย โดยจะทำงานร่วมกับ U4D, U4E, U6C และ U6D ความต้านทานปรับค่าได้ 10 กิโลโอห์ม จะปรับความสว่างของแอลซีดีโมดูล



รูปที่ 3.17 การขยายพอร์ทเพิ่มเติมบนชุดควบคุม

3.3.1.16 การสร้าง

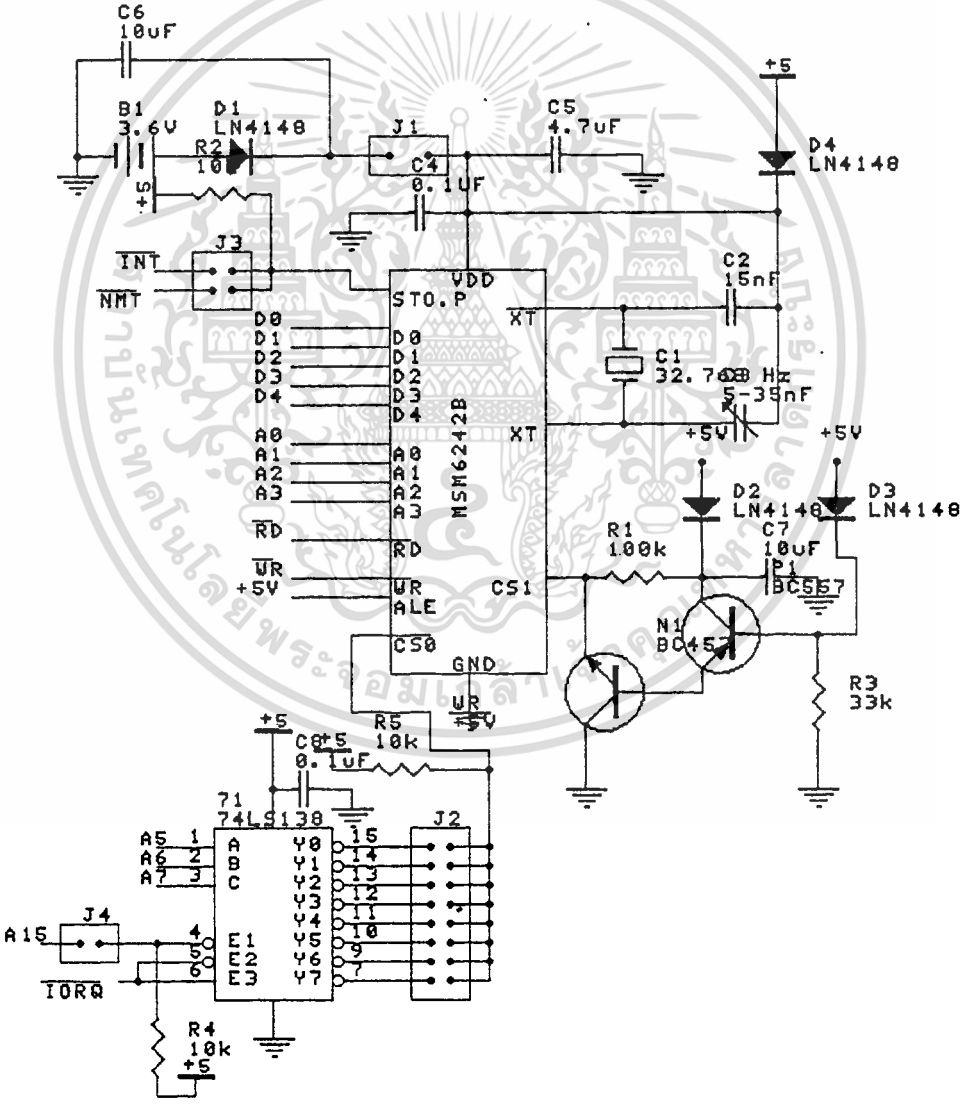
บอร์ด CP-68HC11 ใช้แผ่นวงจรพิมพ์ 2 หน้า ชนิดเพลตทรูโฮลด์ ความยุ่งยากอยู่ที่การติดตั้ง U1 68HC11 เพราะเป็นตัวถังแบบ PLCC ต้องติดตั้งซ็อกเก็ตก่อน ตำแหน่งและทิศทาง ดูดังภาคผนวก

การลงอุปกรณ์ควรใส่อุปกรณ์ที่มีขนาดเล็กก่อนเช่น ตัวต้านทาน ไดโอด ตัวเก็บประจุ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

และใส่ให้ถูกขั้วด้วย สำหรับการใส่ชิ๊อกเก็ต 52 ขา PLCC พื้นของชิ๊อกเก็ตและที่ไอซีจะมีรอยบากอยู่ ให้รอยบากนี้อยู่ทางด้านซ้ายบนโดยจะใส่ไอซีให้รอยบากอยู่ทางด้านเดียวกัน ที่ตัวไอซีจะมีจุดวงกลมเล็ก ๆ อยู่ ในรูปการจัดวางอุปกรณ์ในภาคผนวกจะแสดงตำแหน่งขา 1 ของไอซี สำหรับการใส่คอนเนคเตอร์ของพอร์ทอนุกรมทั้ง 2 ช่องนั้น ให้หันด้านพลาสติกเข้าด้านใน ด้านที่เป็นขั้วคอนเนคเตอร์จะหันออกด้านนอกแผ่นวงจรพิมพ์

3.3.2 ส่วนวงจรนาฬิกาบอกเวลาที่แท้จริง (REAL TIME CLOCK)



รูปที่ 3.18 แสดงวงจรนาฬิกาบอกเวลาที่แท้จริง REAL TIME CLOCK

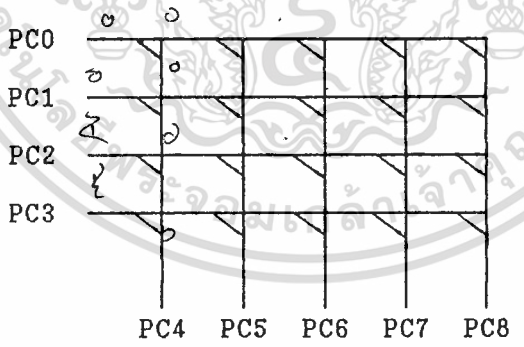
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากรูปที่ 3.18 เราใช้ไอซีเบอร์ MSM 6242B ซึ่งเป็นไอซี REAL TIME CLOCK /CALENDAR ชนิดซีมอสใช้ต่อกับบัสของไมโครโปรเซสเซอร์ได้โดยตรง มีแอดเดรสบัสและดาต้าบัส ขนาด 4 บิท มีรีจีสเตอร์ควบคุม ขนาด 4 บิท 3 ตัว คือ CD, CE และ CF ต่อร่วมกับคริสตอล 32.768 KHz และตัวเก็บประจุเพื่อสร้างสัญญาณนาฬิกาให้กับวงจรและใช้แบตเตอรี่ขนาด 3.6 โวลท์ ในกรณีไฟฟ้าดับเวลาก็ยังคงเดินเป็นปกติ และใช้ไอซีเบอร์ 74HTC138 เป็นตัวถอดรหัสแอดเดรสเพื่อเลือกการทำงานของไอซี MSM 6242B ดังรูป

3.3.2.1 การสร้างและการปรับแต่ง

บอร์ด RTC ใช้แผ่นวงจรพิมพ์ 2 หน้าชนิดเพลดทรูโฮลด์ ใส่อุปกรณ์ที่มีขนาดเล็กก่อน เช่น ตัวต้านทาน ไดโอด ตัวเก็บประจุ ซอกเก็ตไอซี และที่ใส่ BATTARY และให้ระวังเรื่องขั้วด้วย จากนั้นก็บัดกรีอุปกรณ์ต่างๆ และใส่ไอซีลงซอกเก็ต เมื่อเสร็จแล้วก็นำบอร์ด RTC เสียบลงที่คอนเนคเตอร์ 40 ขา ซึ่งเป็น Z80 บัสของบอร์ด CP-68HC11 แล้วจ่ายไฟเข้าบอร์ด จากนั้นใช้เครื่องวัดความถี่วัดที่ขา 17 (XT) แล้วปรับทริมเมอร์ ให้อ่านค่าความถี่ได้ 32768 Hz

3.3.3 ส่วนของเมตริกซ์คีย์บอร์ด (MATRIX KEYBOARD)



รูปที่ 3.19 การต่อวงจรส่วนคีย์บอร์ด

จากรูปที่ 3.19 เป็นการออกแบบวงจรเมตริกซ์คีย์บอร์ดขนาด 4x4 ซึ่งใช้ต่อเข้ากับคอนเนคเตอร์ขนาด 10 ขา ซึ่งต่อมาจากพอร์ท C ของ ไอซี 8255

การทำงานของวงจรจะใช้วิธีการสแกน เพื่อหาว่าสวิทช์ใดที่ถูกกด โดยอาจเริ่มต้นที่คอลัมน์ 0 โดยส่งค่าลอจิก "0" มาทางพอร์ทเอาท์พุทให้ U6 ซึ่งเป็นตัวสแกนคอลัมน์ เมื่อได้รับลอจิก "0" ทางอินพุทจึงทำให้ U6 ให้เอาท์พุท Y0 เป็นลอจิก "0"ส่วนคอลัมน์อื่น ๆ

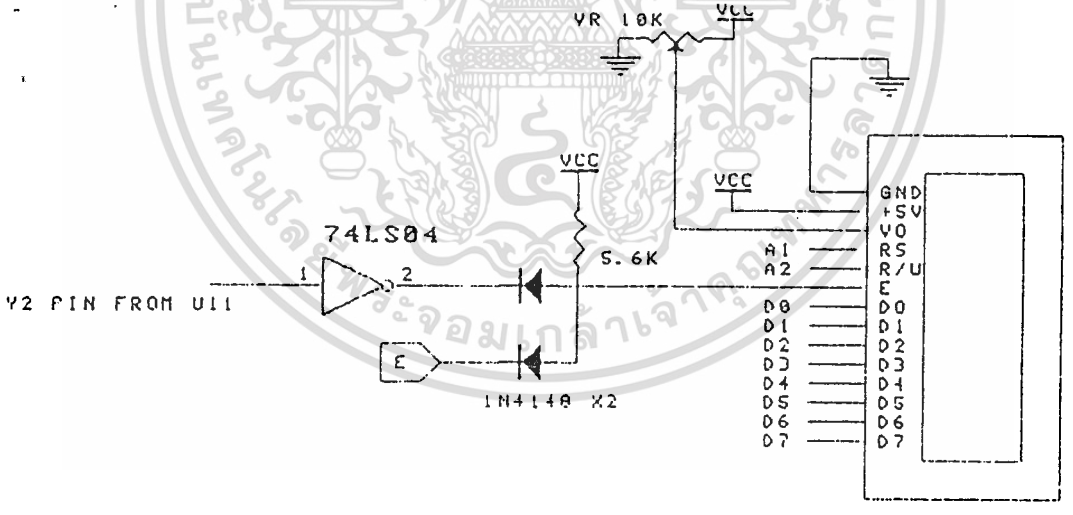
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เป็นลอจิก "1" แล้วทำการตรวจสอบทีละแถว โดยจะเริ่มจากแถว 0 ตรวจสอบว่ามีระดับลอจิกเป็น "0" หรือไม่ ถ้ามีระดับลอจิกเป็น "0" แสดงว่าสวิทช์ตำแหน่งนี้ถูกกด ถ้าไม่มีก็เลื่อนไปแถวถัดไปแล้วทำการเลื่อนและตรวจสอบทั้งด้านคอลัมน์และแถวต่อไปเรื่อย ๆ จนกว่าจะหมดหรือจนกระทั่งพบสวิทช์ที่ถูกกด และเมื่อพบว่าสวิทช์ใดถูกกดแล้ว จะต้องทราบว่าสวิทช์นี้คือสวิทช์ตำแหน่งใด เพื่อจะได้ตีความหมายหรือสร้างรหัสขึ้นตามสวิทช์ตำแหน่งนั้น ๆ เพื่อนำรหัสนั้นไปใช้งานต่อไป

3.3.3.1 การสร้างคีย์บอร์ดขนาด 4x4

ออกแบบลายทองแดงตามรูปวงจร รูปที่ 3.19 โดยใช้แผ่นปริ้นท์หน้าเดียว จากนั้นใช้สวิทช์ชนิด กดติด ปล่อยดับ และคอนเนคเตอร์ 10 ขา ใส่ลงบนแผ่นปริ้นท์ แล้วบัดกรีขาให้เรียบร้อย

3.3.4 ส่วนการแสดงผลการใช้งานบนจอแอลซีดี (LCD DISPLAY)



รูปที่ 3.20 วงจรเชื่อมต่อกับแอลซีดีขนาด 20 ตัวอักษร x 2 แถว

จากรูปที่ 3.20 เป็นการออกแบบการแสดงผลแอลซีดี แอลซีดีที่ใช้จะเป็นโมดูลแอลซีดีขนาด 20 ตัวอักษร 2 แถว ซึ่งภายในโมดูลประกอบด้วย

1. DOT MATRIX LCD เป็นตัวแสดงผลให้เรามองเห็นในลักษณะการปิดและเปิดตัวเองกับแสงก็คือ ส่วนที่เป็นกระจกบรรจุผลึก

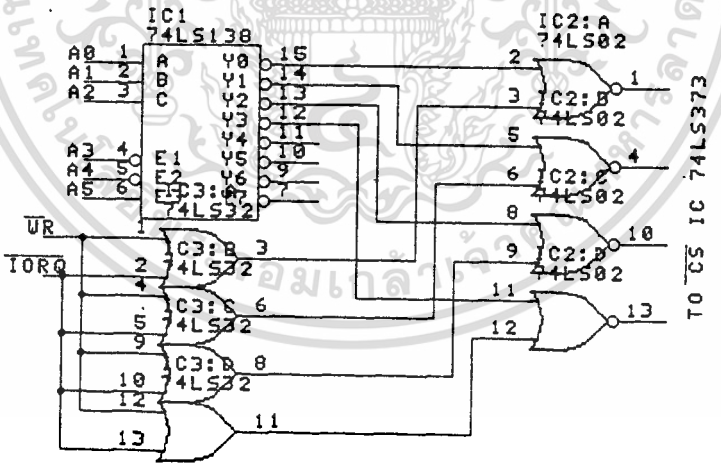
เอกสารนี้ใช้เพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2. DRIVER เป็นตัวรับสัญญาณจากตัวควบคุมมาขับผลิกแอลซีดีอีกตัวหนึ่ง โดยมีเบอร์ที่นิยมใช้ในแอลซีดีโมดูล เช่น HD44100H และ MSM5259

3. CONTROLLER เป็นตัวที่รับข้อมูลจากอุปกรณ์ภายนอกมาและจัดการควบคุมแอลซีดีโมดูลให้ทำงานแสดงผลต่าง ๆ เช่น การลบจอภาพ การเกิดตัวอักษร เป็นต้น โดยมีเบอร์ไอซีที่นิยมใช้กัน คือ HD44780 ซึ่งจะใช้แบบ CHARACTER แอลซีดี MODULE เป็นส่วนใหญ่

เพราะฉะนั้น ในการออกแบบเราเพียงแต่นำข้อมูลมาเชื่อมต่อกับ 68HC11 เพื่อส่งสัญญาณควบคุมและข้อมูลในการแสดงผลและการอ่านข้อมูลต่าง ๆ กับโมดูลเท่านั้น ในการเชื่อมต่อขา D0-D7 ซึ่งเป็นขาบัสข้อมูลของแอลซีดีสามารถที่จะต่อเข้ากับ PC0-PC7 ของ 68HC11 โดยตรง ขา RS และ R/W จะต่อเข้ากับแอดเดรส A1 และ A2 โดยขา E จะใช้เอาท์พุท Y2 ของ 74HC138 เป็นตัวเลือก โดยมีแอดเดรสของพอร์ทที่ใช้งานอยู่ในช่วง 1400H-15FFH การปรับระดับความสว่างของจอแอลซีดีอาศัยการปรับไบอัสให้ขา V0 โดยต่อตัวต้านทานปรับค่าได้ 10 กิโลโอห์ม ปรับระดับการไบอัส

3.3.5 ส่วนวงจรถอดรหัสพอร์ทภายนอก (DECODER CIRCUIT)



รูปที่ 3.21 วงจรถอดรหัสพอร์ทภายนอก

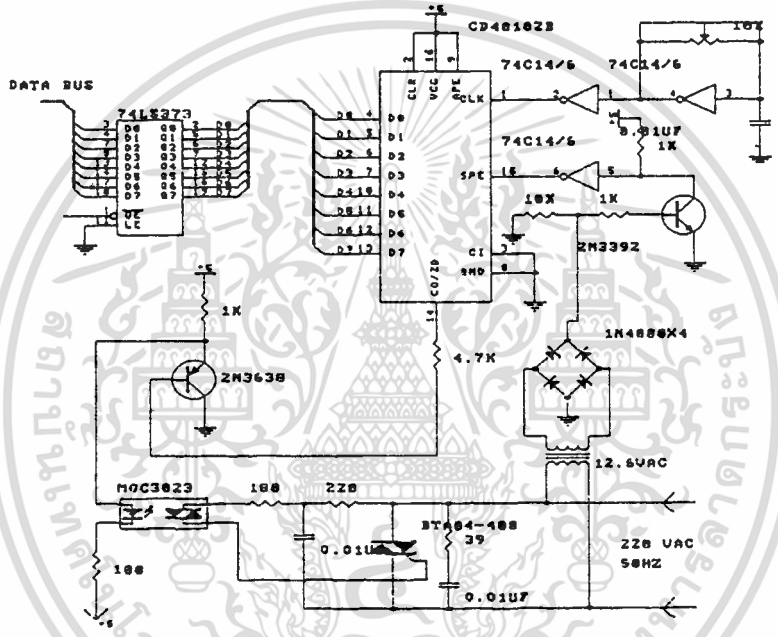
จากรูปวงจรรูปที่ 3.21 เราจะใช้ไอซี 74LS138 ซึ่งเป็นไอซีชนิด 3 to 8 Line Demultiplexer สำหรับถอดรหัสพอร์ทจากแอดเดรส A0-A5 โดยจะต่อตามลักษณะในรูปวงจร และทำงานร่วมกับสัญญาณ WR และ IORQ ซึ่งต่อเข้ากับเกทต่างๆ ดังรูปที่ 3.21 เพื่อนำสัญญาณที่ได้แต่ละช่องไปเลือกพอร์ทไอซี 74LS373 ในวงจรหรี่ไฟซึ่งต่อ D0-D7 ไว้

เอกสารนี้ [illegible] ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.3.5.1 การสร้างวงจรถอดรหัสพอร์ทภายนอก

ออกแบบลายทองแดงตามรูปวงจร รูปที่ 3.21 โดยใช้แผ่นปริ้นท์หน้าเดียว ด้วยโปรแกรม PROTEL จากนั้นใส่อุปกรณ์พวกซอกเก็ตไอซีและคอนเนคเตอร์ต่างๆ ลงบนแผ่นปริ้นท์ แล้วบัดกรีขาให้เรียบร้อย แล้วจึงใส่ไอซีเบอร์ต่างๆ ลงในซอกเก็ต

3.3.6 ส่วนวงจรควบคุมความเข้มของแสงไฟด้วยไมโครคอนโทรลเลอร์



รูปที่ 3.22 วงจรควบคุมความเข้มของแสงไฟด้วยไมโครคอนโทรลเลอร์

จากรูปที่ 3.22 ทำการออกแบบวงจรโดยใช้ไอซีเบอร์ 74LS373 เป็นตัวพักข้อมูล (Buffer) D0-D7 จากไมโครคอนโทรลเลอร์ ซึ่งต่อมาจาก บอร์ดวงจรถอดรหัสพอร์ทภายนอก โดยออกแบบไว้ทั้งหมด 4 แชลแนล ทำการเลือกแชลแนลด้วยสัญญาณ Chip Select ที่ได้จากบอร์ดวงจรการถอดรหัสพอร์ท แล้วส่งดาต้า D0-D7 เข้าไปควบคุมระดับความเข้มของแสง ที่ไอซีเบอร์ CD 40102B ซึ่งมีวงจรผลิตความถี่ต่ออยู่ภายนอกเพื่อใช้กระตุ้นการทำงานของไอซี แล้วส่งสัญญาณออกจากขา 14 นำไปเข้าวงจรควบคุมหลอดไฟโดยมีออปโต้ไตรแอคเบอร์ MOC 3023 เป็นตัวควบคุมระหว่างภาคอินพุทและเอาท์พุทของวงจร และจะนำสัญญาณไฟเอซีจากเอาท์พุทไปผ่าน วงจรซิงค์สัญญาณ ซึ่งประกอบไปด้วยหม้อแปลง 12 โวลท์ ผ่านวงจรบริดจ์เร็คติฟาย และเกทชมิทท์ทริกเกอร์ เข้าไปที่ขา 15

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

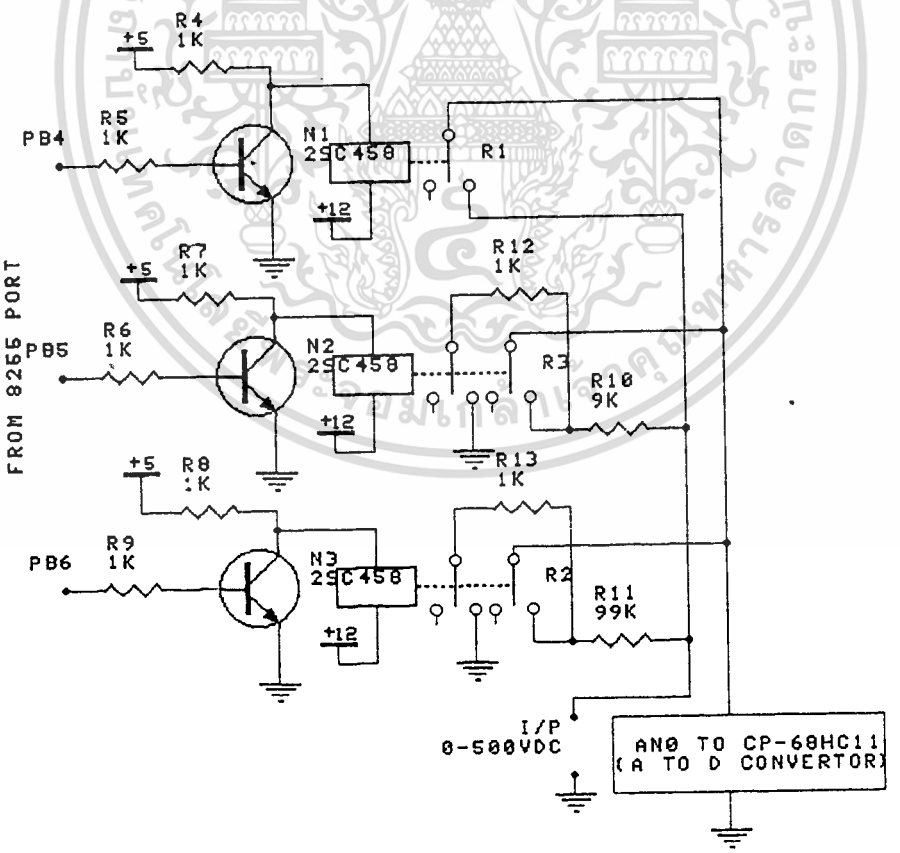
ทำงาน ไม่ให้ไตรแอคและออปโต้ไตรแอคเกิดการชำรุดเสียหาย

3.3.7.1 การสร้างวงจรโซลิดสเตทรีเลย์

ออกแบบลายทองแดงตามรูปวงจร รูปที่ 3.23 โดยใช้แผ่นปริ้นท์หน้าเดียว ด้วยโปรแกรม PROTEL ให้มีลักษณะเป็นบอร์ดควบคุม โดยใช้เฮดเดอร์ขนาด 34 ขาเป็นจุดต่อเข้ากับคอนเนคเตอร์ของพอร์ท 8255 บนบอร์ด CP-68HC11 ซึ่งออกแบบไว้ทั้งหมด 12 แชลแนล อยู่บนบอร์ดเดียวกันทั้งหมด จากนั้นใส่อุปกรณ์ตัวเล็กๆก่อนตามด้วยพวกซอกเก็ตไอซีและเฮดเดอร์ลงบนแผ่นปริ้นท์ แล้วบัดกรีขาให้เรียบร้อยรวมทั้งติดแผ่นระบายความร้อนให้กับไตรแอคเบอร์ BT 136 ด้วย แล้วจึงใส่ไอซีออปโต้ไตรแอคในซอกเก็ต

3.3.8 ส่วนวงจรดีซีโวลท์มิเตอร์ (DC VOLTMETER CITCUIT)

3.3.8.1 วงจรวัดแรงดันไฟตรง 0-500 โวลท์ เปลี่ยนย่านการวัดอัตโนมัติ



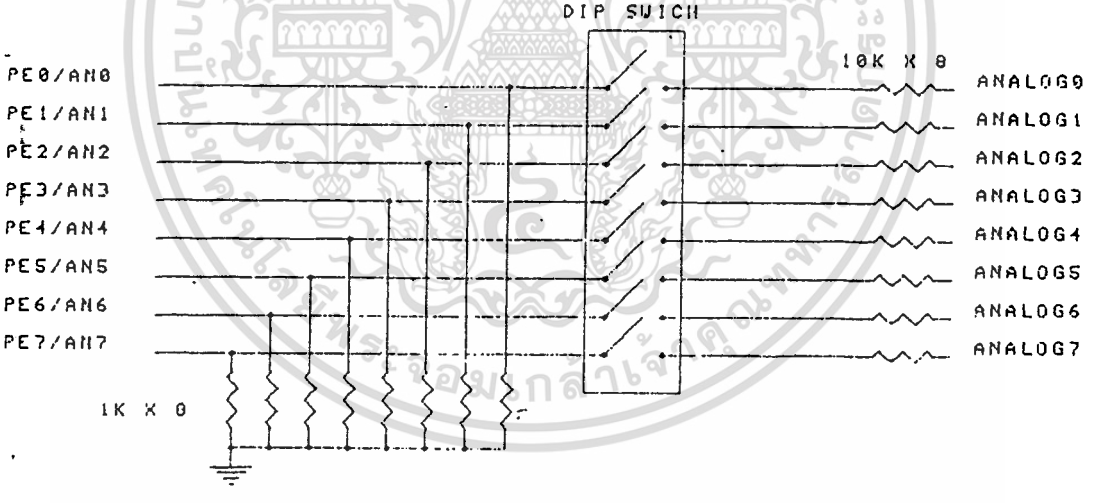
รูปที่ 3.24 วงจรวัดแรงดันไฟตรง 0-500 โวลท์ เปลี่ยนย่านการวัดอัตโนมัติ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากรูปที่ 3.24 ออกแบบวงจรโดยใช้พอร์ทจาก 8255 คือ PB4, PB5 และPB6 มาทำการควบคุมรีเลย์ของแต่ละย่านการวัดว่าจะให้ทำการวัดค่าที่เหมาะสมกับย่านนั้น คือ ย่าน 0-5 โวลท์, 0-50 โวลท์ และย่าน 0-500 โวลท์ โดยใช้ตัวต้านทานมาต่อแบบดีไวเดอร์ เพื่อจำกัดค่าแรงดันให้อยู่ในระดับ 0-5 โวลท์ก่อนจะเข้าวงจรเอทูดีดังรูป

3.3.8.2 วงจรวัดแรงดันไฟตรง 0-5 โวลท์

การออกแบบภาคเชื่อมต่อในการนำวงจรแปลงสัญญาณอนาล็อกเป็นสัญญาณดิจิตอลของบอร์ด CP-68HC11 ที่ไปใช้วัดระดับแรงดันสัญญาณอนาล็อกที่เป็นไฟตรง 0-5 โวลท์ จะใช้ตัวต้านทาน 10 กิโลโอห์ม ต่อลงกราวด์เพื่อแก้สภาวะอิมพีแดนซ์สูงของอินพุท ในขณะไม่มีสัญญาณอนาล็อกเข้ามา การต่อตัวต้านทานค่า 10 กิโลโอห์ม จะจำกัดกระแสเพื่อป้องกันความเสียหายของพอร์ทแปลงสัญญาณของบอร์ด CP-68HC11 ในส่วนนี้ได้ทำการออกแบบไว้ 3 แชนแนล คือใช้งานที่ขา AN1, AN2, AN3 ของวงจรเอทูดีของบอร์ด CP-68HC11



รูปที่ 3.25 วงจรวัดแรงดันไฟตรง 0-5 โวลท์

3.3.9 ส่วนของวงจรตรวจจับ (SENSORS CIRCUIT)

การออกแบบภาคเชื่อมต่อในการนำวงจรแปลงสัญญาณอนาล็อกเป็นสัญญาณดิจิตอลของบอร์ด CP-68HC11 ที่ไปใช้ในการตรวจจับแรงดันสัญญาณอนาล็อกที่เป็นไฟตรง 0-5 โวลท์ จะใช้ตัวต้านทาน 10 กิโลโอห์ม ต่อลงกราวด์เพื่อแก้สภาวะอิมพีแดนซ์สูงของอินพุท ในขณะไม่มีสัญญาณอนาล็อกเข้ามา การต่อตัวต้านทานค่า 10 กิโลโอห์ม จะจำกัดกระแสเพื่อป้อง

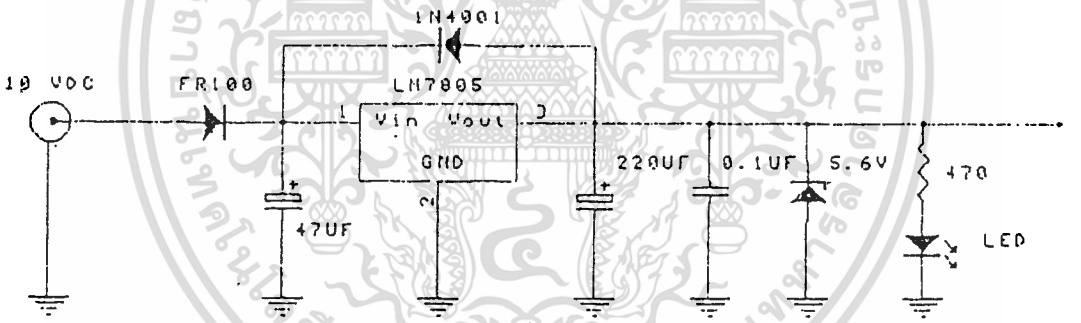
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กันความเสียหายของพอร์ทแปลงสัญญาณของบอร์ด CP-68HC11 ในส่วนนี้ได้ทำการออกแบบไว้ 4 แชนแนล คือใช้งานที่ขา AN4, AN5, AN6 และ AN7 ของวงจรเอทูดีของบอร์ด CP-68HC11 ซึ่งวงจรนี้จะไปทำการควบคุมวงจรโซลิดสเตทรีเลย์ แชลแนลที่ 9-12 ตามลำดับ โดยที่โซลิดสเตทรีเลย์จะต่อวงจรเมื่อมีสัญญาณตรวจจับที่เป็นไฟตรงตามที่ผู้ใช้งานได้ทำการตั้งค่าระดับแรงดันไว้ในแต่ละแชลแนล ตามที่ได้โปรแกรมไว้ขณะใช้งาน

3.3.10 ส่วนของแหล่งจ่ายไฟ (POWER SUPPLY)

ส่วนของแหล่งจ่ายไฟจะใช้ 3 ชุดเพื่อจ่ายให้กับวงจรส่วนต่างๆ คือ

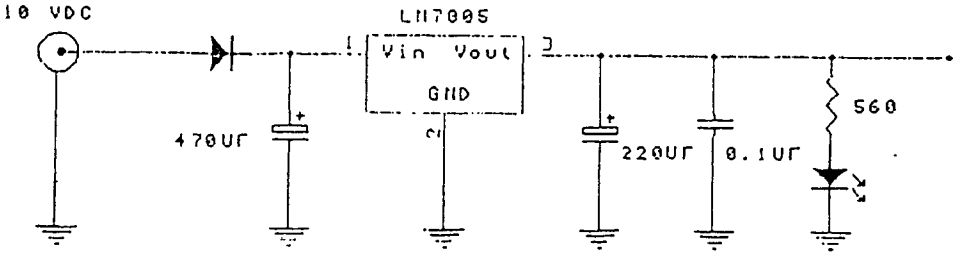
1. วงจรรักษาระดับแรงดันไฟตรงในบอร์ดควบคุม CP-68HC11
2. วงจรรักษาระดับแรงดันไฟตรงในบอร์ดโซลิดสเตทรีเลย์ (SSRAC)
3. วงจรรักษาระดับแรงดันไฟตรงในบอร์ดวงจรถอดรหัสพอร์ทจากภายนอก
4. วงจรแปลงแรงดันไฟสลับเป็นไฟตรงสำหรับวงจรวัดแรงดันไฟตรง 0-500 โวลท์



รูปที่ 3.26 วงจรรักษาระดับแรงดันไฟตรงในบอร์ดควบคุม

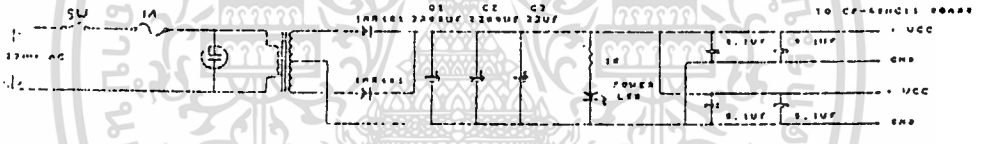
การออกแบบวงจรจะใช้ LM7805 เป็นตัวเรกกูเลตไฟตรง 5 โวลท์ จะใช้ไดโอด FR100 เป็นอุปกรณ์ป้องกันการต่อแหล่งจ่ายไฟกลับขั้ว ตัวไดโอด 1N4001 ใช้บายพาสการคายประจุของตัวเก็บประจุไม่ให้คายประจุผ่านไอซีเรกกูเลต สำหรับซีเนอร์ไดโอดมีขนาด 5.6 โวลท์ ใช้ป้องกันการผิดพลาดของตัวเรกกูเลต จะจำกัดระดับแรงดันไฟเอาท์พุทไม่ให้เกิน 5.6 โวลท์ ตัวต้านทาน 470 โอห์ม นำมาใช้ลดแรงดันและจำกัดกระแสที่ไบอัสให้แก่แอลอีดีในสภาวะที่เปิดเครื่องอยู่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.27 วงจรรักษาระดับแรงดันไฟตรงในบอร์ดโซลิดสเตทรีเลย์ (SSRAC) และวงจรรักษาระดับแรงดันไฟตรงในบอร์ดวงจรถอดรหัสพอร์ทจากภายนอก

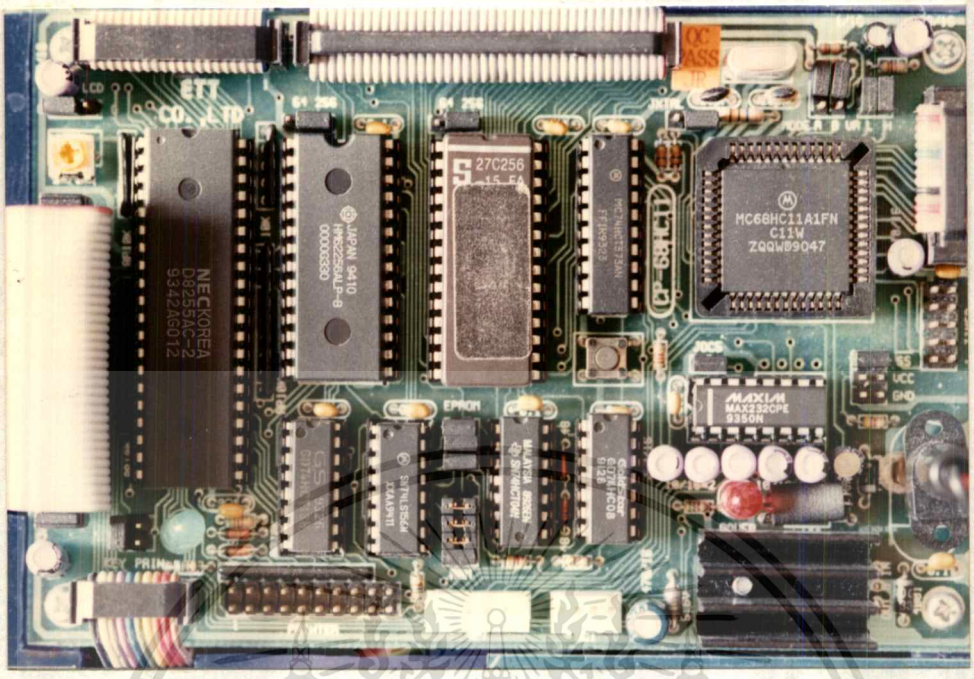
การออกแบบวงจรจะใช้ LM7805 เป็นตัวเรกกูเลตไฟตรง 5 โวลท์ไปเลี้ยงวงจร โดยมีตัวเก็บประจุค่า 470uF ใช้เป็นตัวฟิลเตอร์แรงดันไฟตรงให้เรียบยิ่งขึ้น โดยมีวงจรบริดจ์ไดโอด ทำหน้าที่ในการเรียงกระแสจากเอซีเป็นดีซีก่อนดังรูปที่ 3.28



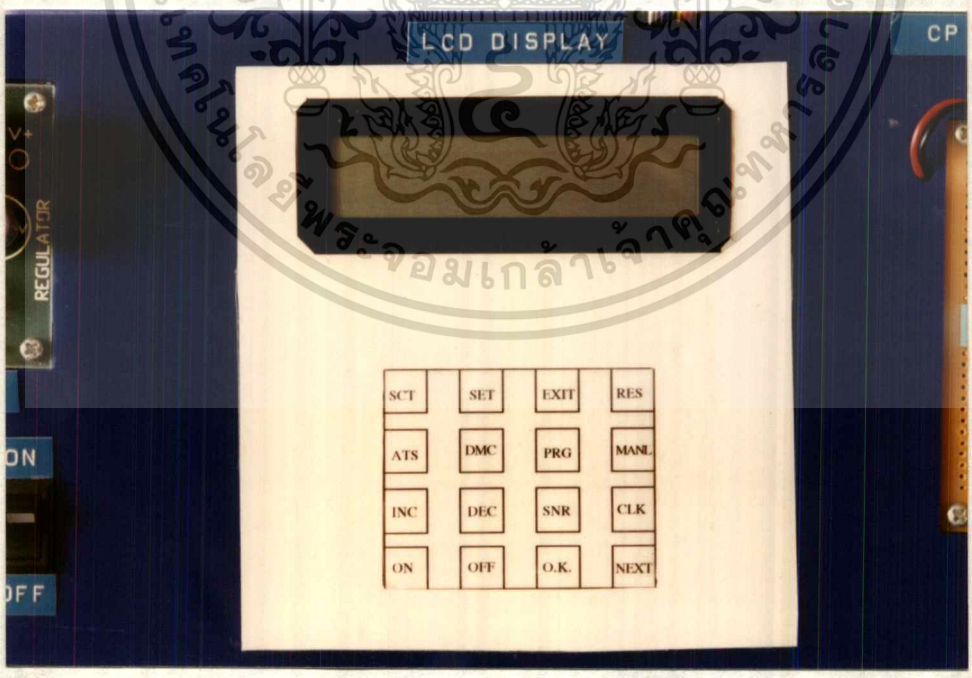
รูปที่ 3.28 วงจรแปลงไฟสลับเป็นไฟตรง

การออกแบบวงจรนั้นจะใช้ไดโอด 1N4001 4 ตัว เป็นวงจรเรียงกระแสแบบบริดจ์ ไฟตรงที่ได้จากการเรียงกระแส จะถูกฟิลเตอร์ และกรองสัญญาณรบกวน โดยตัวเก็บประจุแรงดันไฟตรงเอาท์พุทที่ฟิลเตอร์แล้วจะมีค่าประมาณ 12 โวลท์ ใช้สำหรับควบคุมรีเลย์ในวงจรวัดแรงดันไฟตรง 0-500 โวลท์ เปลี่ยนย่านการวัดอัตโนมัติ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



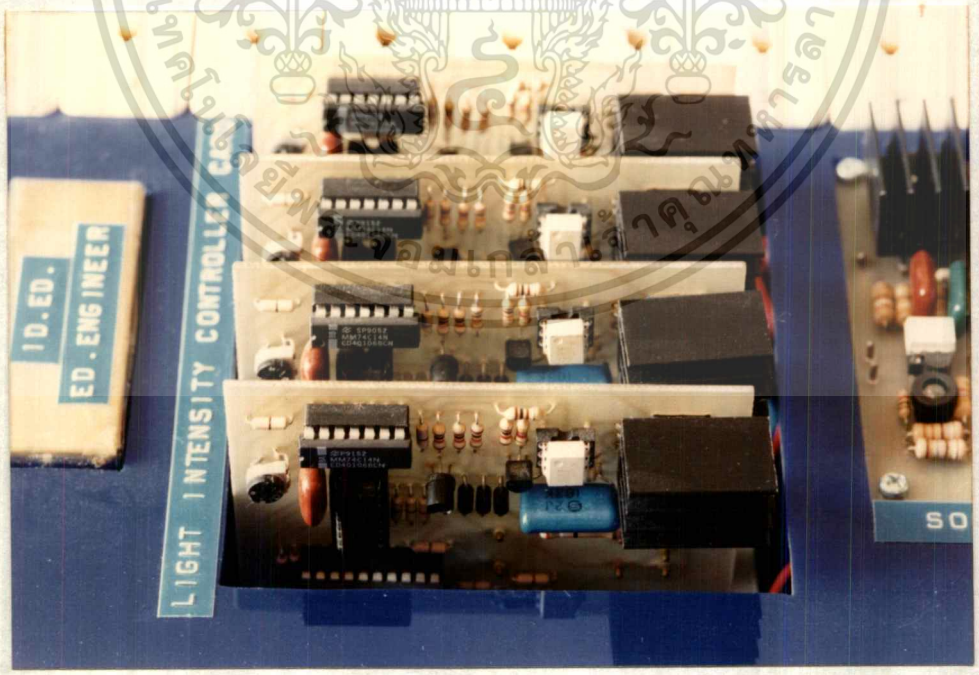
รูปที่ 3.29 ชุดควบคุม (CP-68HC11)



รูปที่ 3.30 เมตริกซ์คีย์บอร์ดขนาด 4x4

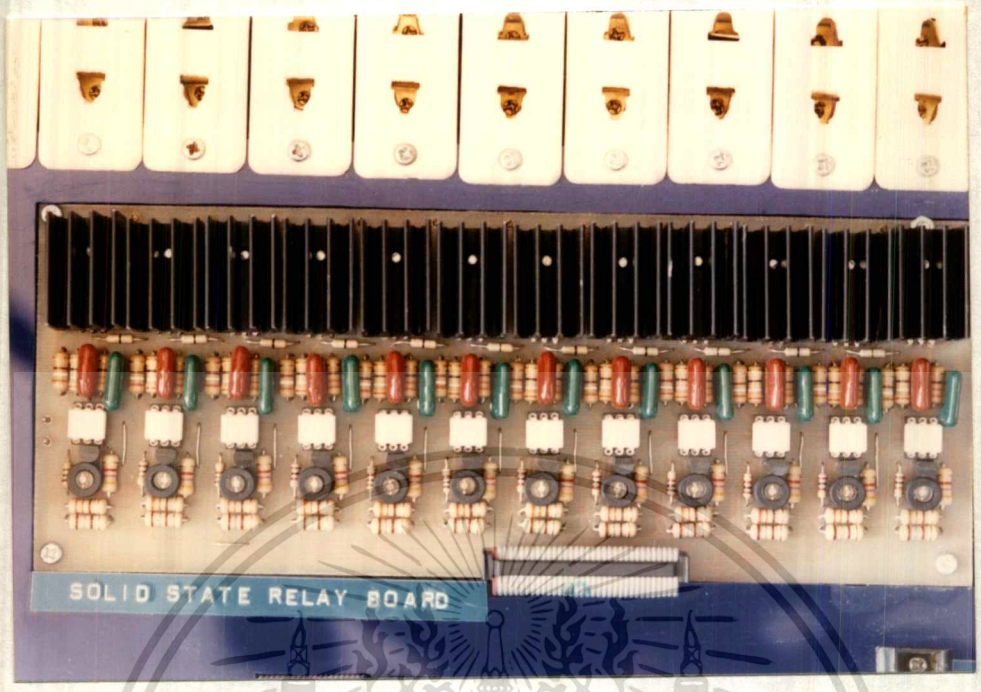
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 3.31 การแสดงผลบนจอแอลซีดีขนาด 20 ตัวอักษร 2 แถว

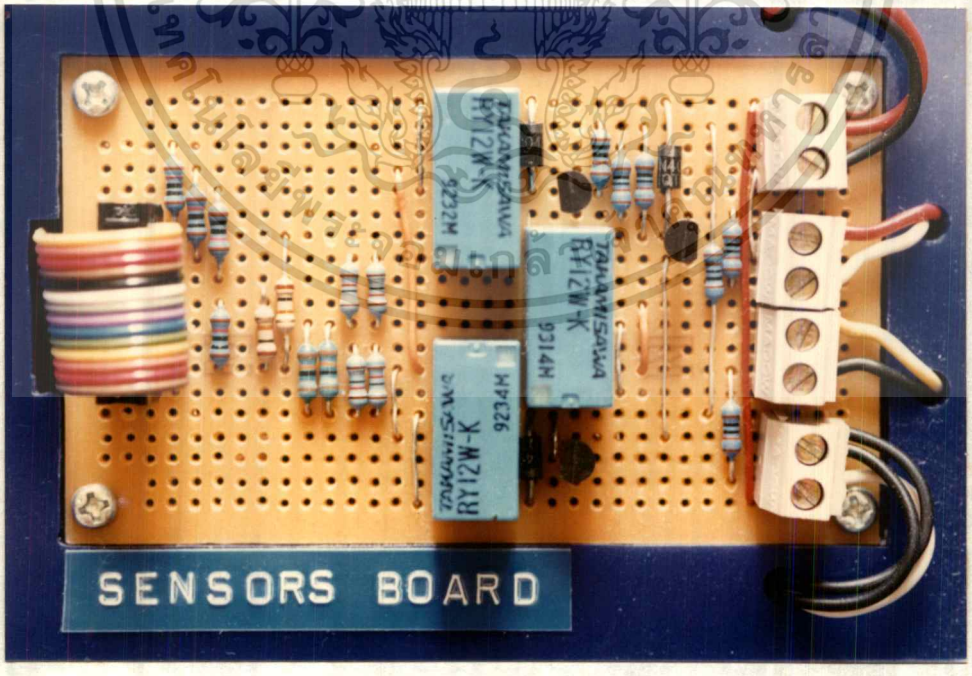


รูปที่ 3.32 การ์ดควบคุมความเข้มแสงสว่างของหลอดไฟ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

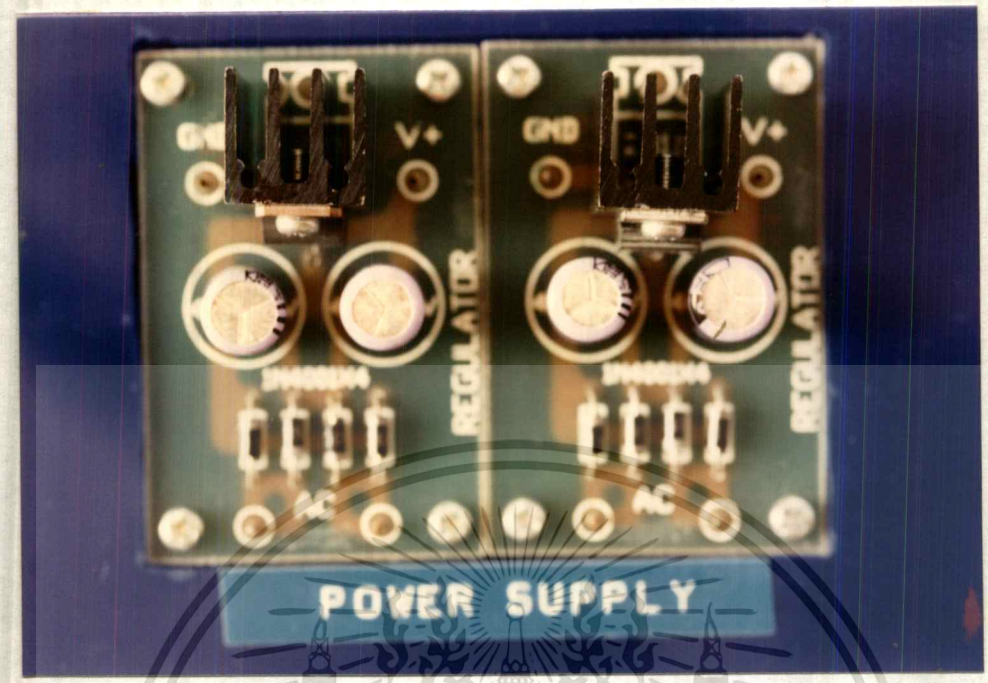


รูปที่ 3.33 บอร์ดโซลิดสเตทรีเลย์ 220 โวลท์เอซี

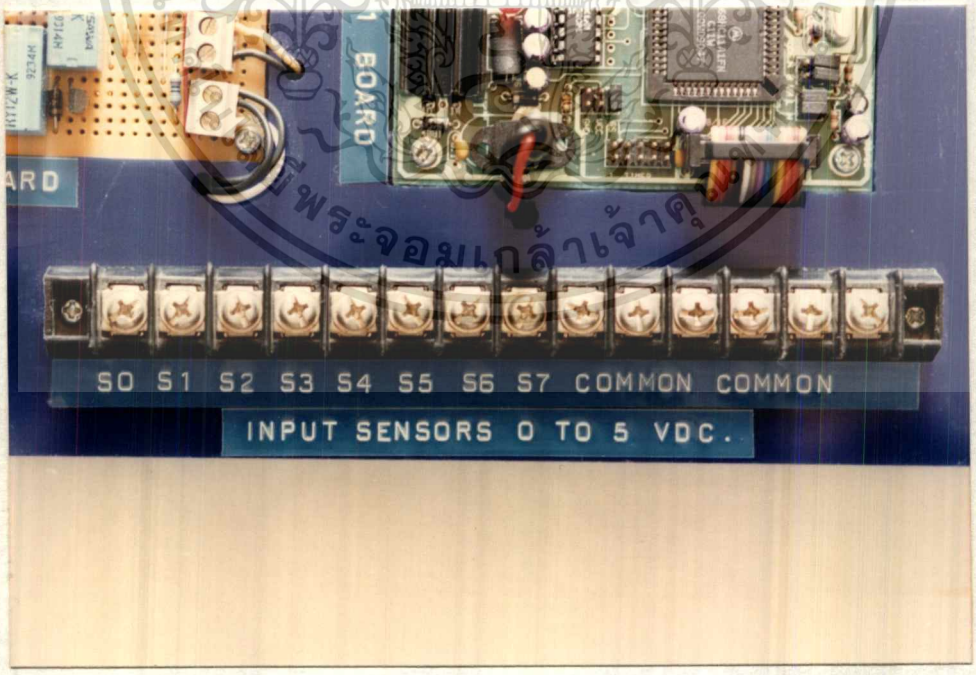


รูปที่ 3.34 บอร์ดตรวจจับสัญญาณแรงดันจากภายนอก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

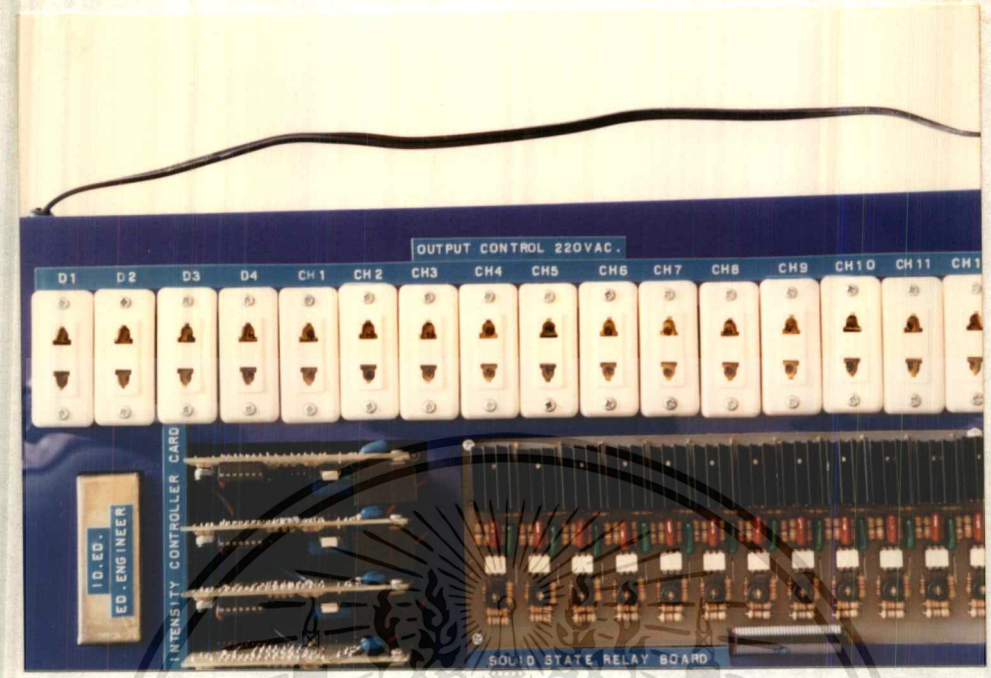


รูปที่ 3.35 แหล่งจ่ายไฟ

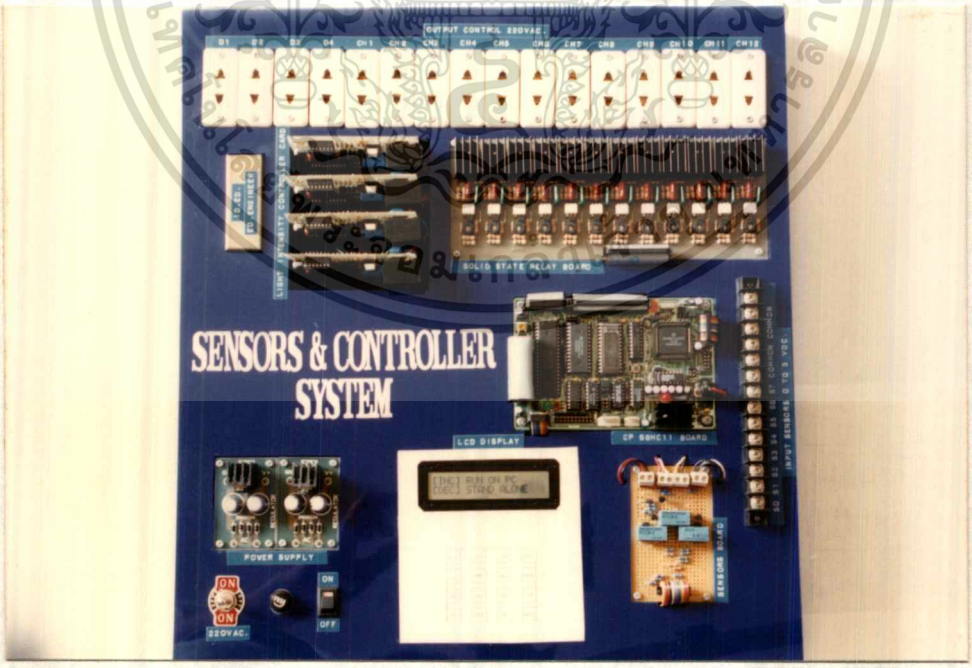


รูปที่ 3.36 จุดต่อรับสัญญาณตรวจจับแรงดันจากภายนอก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

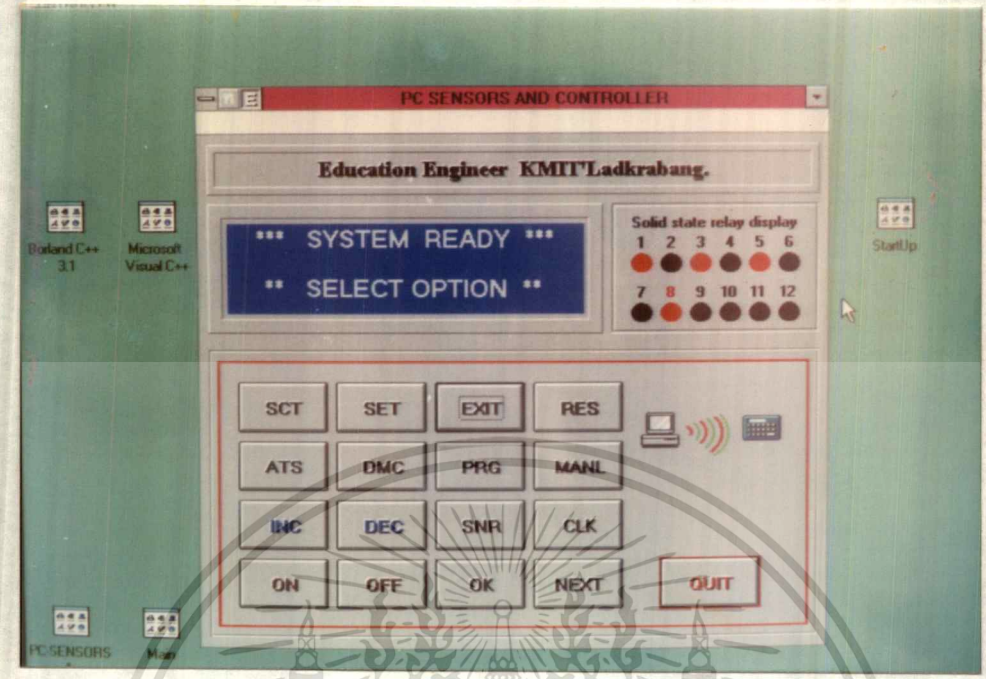


รูปที่ 3.37 จุดต่ออุปกรณ์ไฟฟ้า 220 โวลท์เอซี

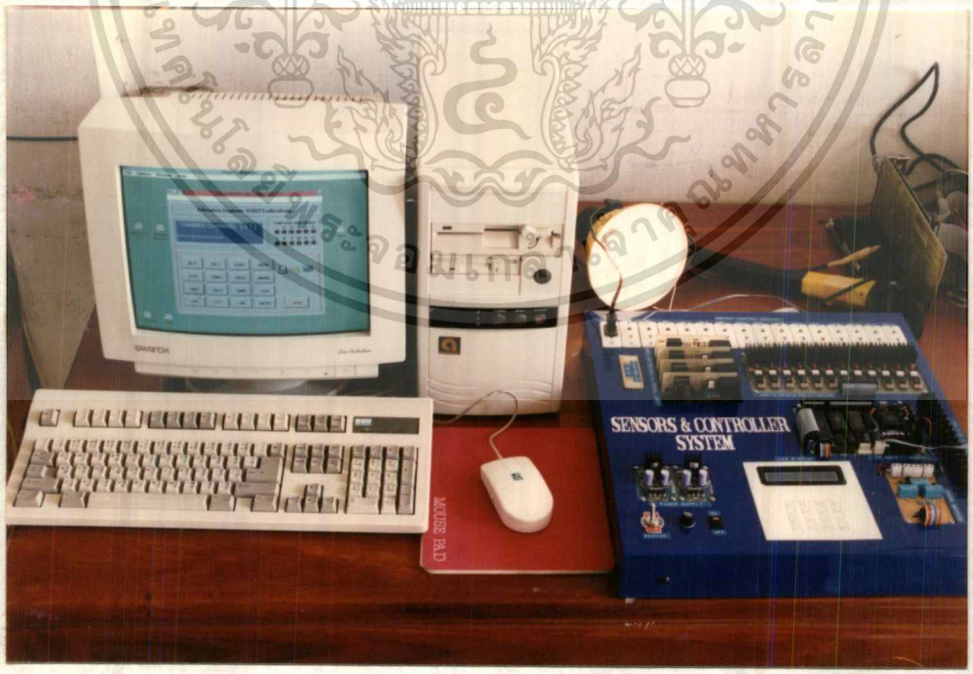


รูปที่ 3.38 ลักษณะสมบูรณ์ของระบบตรวจจับและควบคุม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.39 การควบคุมระบบโดยใช้โปรแกรมบนไมโครซอฟท์วินด์โดวส์



รูปที่ 3.40 แสดงการอินเตอร์เฟสกับไมโครคอมพิวเตอร์ผ่านพอร์ท RS-232C

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 4

การทดลองและผลการทดลอง

ในบทนี้จะกล่าวถึงการนำฮาร์ดแวร์ในส่วนต่างๆ ที่ได้จากการออกแบบและการสร้างในบทที่ 3 มาทำการทดลอง ซึ่งแบ่งเป็นส่วนต่างๆ ดังต่อไปนี้

4.1 การทดลองการทำงานของโซลิดสเตทรีเลย์

4.1.1 ลำดับขั้นการทดลอง

1. ต่ออุปกรณ์ไฟฟ้าที่จุดต่อต่างๆ บนบอร์ดโซลิดสเตทรีเลย์ ซึ่งมีทั้งหมด 12 ช่อง
2. ป้อนแรงดันจากแหล่งจ่ายไฟ 5 โวลท์ดีซี และ 220 โวลท์เอซี เข้าที่จุดต่อบนบอร์ด
3. ป้อนสัญญาณลอจิก "1" เข้าที่อินพุททีละช่อง ตั้งแต่ ช่องที่ 1-12

4.1.2 ผลการทดลอง

จากการทดลองปรากฏว่าเมื่อป้อนสัญญาณลอจิก "1" เข้าที่อินพุทช่องใดใน 12 ช่อง แอลอีดี ที่ใช้แสดงผลบนบอร์ดในช่องนั้นจะสว่าง และอุปกรณ์ไฟฟ้าที่ต่ออยู่จะทำงาน

4.2 การทดลองการทำงานของวงจรถอดรหัสพอร์ทภายนอก

4.2.1 ลำดับขั้นการทดลอง

1. ป้อนแรงดันจากแหล่งจ่ายไฟ 5 โวลท์ดีซี เข้าที่จุดต่อบนบอร์ด
2. ป้อนสัญญาณลอจิก "0" และลอจิก "1" เข้าที่เฮ็ดเดอร์ขนาด 40 ขาตามจุดต่อสัญญาณของ Z80 BUS บนบอร์ดวงจรถอดรหัสตามตารางที่ 4.1

4.2.2 ผลการทดลอง

จากการทดลองป้อนสัญญาณลอจิกตามตารางที่ 4.1 ปรากฏว่า จะได้สัญญาณลอจิก "1" ที่ขา 11 ของไอซี 74LS373 ของแต่ละแชลแนล ซึ่งทำให้สัญญาณข้อมูล D0-D7 ถูกส่งออกไปที่เอาท์พุทดาต้าของวงจร

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ADD CH	A5	A4	A3	A2	A1	A0	IORQ	WR
1	1	0	0	0	0	0	0	0
2	1	0	0	0	0	1	0	0
3	1	0	0	0	1	0	0	0
4	1	0	0	0	1	1	0	0

ตารางที่ 4.1 แสดงการทดลองการถอดรหัสพอร์ทภายนอก

4.3 การทดลองวงจรควบคุมความเข้มของแสงไฟ (LIGHT INTENSITY CONTROLLER)

4.3.1 ลำดับขั้นการทดลอง

1. นำการ์ดวงจรควบคุมความเข้มของแสงสว่าง ซึ่งมีทั้งหมด 4 ชุด เสียบลงบนบอร์ดวงจรถอดรหัสพอร์ทภายนอก ที่แชลแนล 1 ทีละชุด โดยการทดลองดังนี้

2. ป้อนแรงดัน 220 โวลท์เอซี เข้าที่จุดต่อ 220 โวลท์บนการ์ด แล้วต่อหลอดไฟแสดงผลที่จุดต่อเอาท์พุทของการ์ด

3. ป้อนสัญญาณข้อมูล D0-D7 , สัญญาณแอดเดรส A0-A5 และ สัญญาณลอจิก "0" เข้าที่ขา WR กับ IORQ เข้าที่เฮดเดอร์ของบอร์ดวงจรถอดรหัส

4. กำหนดสัญญาณแอดเดรส ให้ถอดรหัสไปที่แชลแนลที่1

5. เปลี่ยนแปลงระดับข้อมูล D0-D7 ด้วยรหัส BCD 8421 ตั้งแต่ค่าต่ำ (00) ไปจนถึงค่าสูงสุด (FF)

4.3.2 ผลการทดลอง

จากการทดลองจะเห็นว่า หลอดไฟจะสว่างเต็มที่เมื่อ ข้อมูลมีค่าเป็น "00" และจะค่อยๆ หรี่ลงมา ตามลำดับการเปลี่ยนแปลงของข้อมูลที่เพิ่มขึ้น จนเมื่อข้อมูลถึงค่า "FF" หลอดไฟจะดับ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.4 การทดลองการทำงานของเมตริกซ์คีย์บอร์ด

4.4.1 ลำดับขั้นการทดลอง

1. ใช้โอห์มมิเตอร์ วัดที่จุดร่วมของคีย์บอร์ดที่เฮดเดอร์ ขนาด 10 ขา ในลักษณะเมตริกซ์

2. ทดลองกดปุ่มที่คีย์บอร์ดทีละปุ่ม ในแถวที่ใช้โอห์มมิเตอร์วัด ทั้งหมด 4 แถว

4.4.2 ผลการทดลอง

เมื่อกดคีย์ที่ปุ่มใดในแถวที่ต่อร่วมกับคอลัมน์ในลักษณะเมตริกซ์แล้ว จะมีความต้านทานเป็น 0 โอห์ม หรือช็อต

4.5 การทดลองการทำงานชุดควบคุม (CP-68HC11 BOARD)

4.5.1 ลำดับขั้นการทดลอง

1. นำบอร์ดโซลิดสเตทรีเลย์, บอร์ดวงจรถอดรหัสภายนอก, เชื่อมต่อเข้ากับบอร์ด CP-68HC11 ด้วยสายคอนเนคเตอร์ขนาดต่างๆ

2. ป้อนแรงดันจากแหล่งจ่ายไฟตรง เข้าที่ขั้วบนบอร์ด CP-68HC11

3. ต่อสาย RS-232C จากเครื่องคอมพิวเตอร์เข้ากับบอร์ด CP-68HC11

4. เรียกใช้โปรแกรม PROCOM PLUS บนพีซี ทดลองส่งค่าแอดเดรสและข้อมูลออกไปที่ตำแหน่งของพอร์ท 8255 บนบอร์ดควบคุม ด้วยคำสั่ง o 1200 01 และ o 1200 00

5. ทดลองส่งค่าแอดเดรสและข้อมูลไปถอดรหัสพอร์ทภายนอก เพื่อควบคุมการหรี่ไฟ

4.5.2 ผลการทดลอง

จากการทดลองถึงข้อที่ 4 จะเห็นว่า เมื่อเข้าโปรแกรม PROCOM PLUS แล้ว จะเกิด.LOGO ของโปรแกรม และมีเคอร์เซอร์ปรากฏเป็นการเตรียมพร้อมสำหรับการสื่อสารสัญญาณผ่าน พอร์ท RS-232C จากนั้นเมื่อทดลองส่งค่าออกไปที่พอร์ท 8255 ทีละแชลแนล แล้ว โซลิดสเตทรีเลย์จะแสดงผลการทำงานตามการส่งค่าของแต่ละแชลแนล และเมื่อทดลองถึงข้อ 5 จะเห็นว่า หลอดไฟจะเปลี่ยนระดับความสว่างตามค่าของข้อมูลที่ส่งออกไป

4.6 การทดลองโปรแกรมมอนิเตอร์ของบอร์ด CP-68HC11

4.6.1 ลำดับขั้นการทดลอง

1. นำจอแอลซีดี เมตริกซ์คีย์บอร์ดและบอร์ดฮาร์ดแวร์อื่นๆอินเตอร์เฟสเข้ากับบอร์ด CP-68HC11 และจ่ายไฟตรงให้กับระบบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2. ใช้สายคอนเนคเตอร์ เชื่อมต่อบอร์ดควบคุมเข้ากับเครื่องคอมพิวเตอร์ โดยผ่านพอร์ท RS-232C

3. ใช้โปรแกรม PROGRAM PLUS ในการโหลดโปรแกรมสำหรับระบบที่เขียนขึ้นจาก ET-DEBUGGER 68HC11 ลงบนบอร์ด CP-68HC11

4. ทดลอง RUN โปรแกรมโดยใช้คำสั่ง G 2200 เริ่มต้นซึ่งเป็นตำแหน่งแอดเดรส

4.6.2 ผลการทดลอง

จากการทดลอง RUN โปรแกรมให้กับระบบ จะปรากฏข้อความสำหรับรอรับคำสั่งในการเลือกใช้งานระหว่างเครื่องคอมพิวเตอร์หรือใช้งานบนชุดสาธิตระบบ(STAND ALONE) บนจอแสดงผล แอลซีดี ซึ่งวิธีการใช้งานต่างๆ ของระบบตามโปรแกรมมอนิเตอร์ที่เขียนไว้จะกล่าวไว้ในภาคผนวก

4.7 การทดลองวงจรนาฬิกาบอกเวลาจริง (REAL TIME CLOCK)

4.7.1 การทดลอง

1. ต่อจอแอลซีดีเข้ากับบอร์ด CP-68HC11 และเสียบบอร์ด RTC ลงบน Z80 BUS ของบอร์ด CP-68HC11 แล้วจ่ายไฟตรงให้กับระบบ

2. ใช้สายคอนเนคเตอร์ เชื่อมต่อบอร์ดควบคุมเข้ากับเครื่องคอมพิวเตอร์ โดยผ่านพอร์ท RS-232C

3. เขียนโปรแกรมการตั้งเวลาบน DEBUGGER แล้วสั่ง RUN โปรแกรมที่แอดเดรสเริ่มต้น

4. ทดลองตั้งเวลาจริงให้กับ RTC ในหลักวินาที, นาที, ชั่วโมง และวัน/เดือน/ปี

4.7.2 ผลการทดลอง

เมื่อทำการทดลองในข้อ 3 แล้ว ตัวเลขจะแสดงเวลาเดิมที่อยู่ใน RTC และจะเดินไปเรื่อยๆ จากนั้นเมื่อทดลองตั้งเวลาที่ถูกต้องให้กับ RTC ใหม่แล้วกดคีย์สั่งให้ทำงานต่อไป โปรแกรมจะเซ็ทให้หลักวินาทีเป็น "00" หลังจากนั้นเวลาก็จะเดินตามเวลาใหม่ที่เราได้ตั้งไว้

4.8 การทดลองวงจรตรวจจับแรงดันไฟตรง 0-500 โวลท์ เปลี่ยนย่านการวัดอัตโนมัติ

4.8.1 ลำดับขั้นการทดลอง

1. จ่ายไฟตรง 2 ชุด คือ 12 โวลท์ และ 5 โวลท์ ให้กับ บอร์ดตรวจจับแรงดัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ไฟตรง เข้าที่จุดต่อบนบอร์ด

2. ป้อนสัญญาณลอจิก "1" เข้าที่จุดต่อ PB4 บนบอร์ด สังเกตการทำงานของรีเลย์ตัวที่ 1 (สำหรับย่านการวัด 0-5 โวลท์) แล้วนำสัญญาณลอจิก "1" ออก

3. ป้อนสัญญาณลอจิก "1" เข้าที่จุดต่อ PB5 บนบอร์ด สังเกตการทำงานของรีเลย์ตัวที่ 2 (สำหรับย่านการวัด 0-50 โวลท์) แล้วนำสัญญาณลอจิก "1" ออก

4. ป้อนสัญญาณลอจิก "1" เข้าที่จุดต่อ PB6 บนบอร์ด สังเกตการทำงานของรีเลย์ตัวที่ 3 (สำหรับย่านการวัด 0-500 โวลท์) แล้วนำสัญญาณลอจิก "1" ออก

4.8.2 ผลการทดลอง

จากการทดลองข้อที่ 2 รีเลย์ตัวที่ 1 จะทำการต่อหน้าสัมผัสไปที่ตำแหน่ง NO และเมื่อนำสัญญาณลอจิก "1" ออก รีเลย์ตัวที่ 1 จะทำการต่อหน้าสัมผัสไปที่ตำแหน่ง NC และจากการทดลองในข้อที่ 3 และ 4 ก็จะได้ผลการทดลองในลักษณะเดียวกัน

4.9 สรุปผลการทดลอง

จากการทดลอง การนำฮาร์ดแวร์ที่ได้จากการออกแบบ และการสร้างในบทที่ 3 มาทำการทดลอง การทำงานของวงจรในส่วนต่างๆ ตามที่ได้กล่าวไว้ทั้งหมด ก็จะได้ผลการทดลองตามความคาดหมายและสามารถนำวงจรส่วนต่างๆ อินเตอร์เฟสเข้ากันเป็นชุดสาธิตระบบตรวจจับและควบคุมด้วยเครื่องไมโครคอมพิวเตอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 5
บทวิจารณ์ สรุป และแนวทางในการพัฒนา

5.1 บทสรุป

จากขอบเขตของการออกแบบระบบตรวจจับและควบคุมด้วยไมโครคอมพิวเตอร์ ที่ได้เสนอขออนุมัติ หัวข้อปริญญานิพนธ์ การใช้งานระบบจะแบ่งออกเป็น 2 อย่างคือ

(1) ควบคุมการทำงานของระบบบนตัวเครื่องโดยตรง (STAND ALONE)

(2) ควบคุมการทำงานของระบบด้วยไมโครคอมพิวเตอร์ (PC)

5.1.1 ควบคุมการทำงานของระบบบนตัวเครื่องโดยตรง (STAND ALONE)

การทำงานในส่วนนี้จะประกอบด้วย

- ระบบตรวจจับและแสดงผลแรงดันไฟตรง 0-5 โวลท์ 3 แชลแนล
- ระบบตรวจจับและแสดงผลแรงดันไฟตรง 0-500 โวลท์ เลือกย่านวัดอัตโนมัติ
- ระบบควบคุมระดับสัญญาณตรวจจับเพื่อควบคุมการเปิด/ปิด อุปกรณ์ไฟฟ้า 4 ช่อง
- ระบบควบคุมความเข้มของแสงไฟ ซึ่งมีการปรับความสว่างได้ 20 ระดับ
- ระบบการ เปิด/ปิด ไฟสลับ 220 โวลท์ ซึ่งใช้โซลิดสเตทรีเลย์ ทำหน้าที่ในการเปิด/ปิดวงจร มีการทำงาน 2 ฟังก์ชั่นคือ ฟังก์ชั่น MANUAL จะเป็นการเปิด/ปิด โดยการควบคุมโดยตรงจากปุ่ม ON หรือ OFF และ ฟังก์ชั่น PROGRAM เป็นการตั้งเวลาเปิด/ปิดอุปกรณ์ไฟฟ้าไว้ล่วงหน้า ซึ่งมีทั้งหมด 8 แชลแนล
- ระบบฐานเวลาบนบอร์ด โดยมีการแสดงผล และตั้งเวลาปัจจุบัน เพื่อเป็นฐานเวลาจริง สำหรับการควบคุมการทำงานบนบอร์ด

การทำงานในส่วนนี้ จัดได้ว่ามีการทำงานที่สมบูรณ์ตามวัตถุประสงค์ที่ได้เสนอไว้ ในการขออนุมัติหัวข้อปริญญานิพนธ์ ซึ่งอาจมีบางส่วนที่นอกเหนือจากการเสนอ ที่มีการทำงานผิดพลาดบ้าง เนื่องจากปัญหาการอินเตอร์เฟสของฮาร์ดแวร์

5.1.2 ควบคุมการทำงานของระบบด้วยไมโครคอมพิวเตอร์ (PC)

การใช้งานควบคุมระบบในส่วนนี้ จะใช้โปรแกรมที่สร้างขึ้น สำหรับใช้งานควบคุมระบบบนไมโครซอฟท์วินโดวส์ ซึ่งการใช้งานในโปรแกรมจะปรากฏเป็นภาพจำลองของปุ่มต่างๆ ซึ่งมีลักษณะเป็นเมตริกซ์คีย์บอร์ดขนาด 4x4 โดยการใช้เมาส์คลิกที่ปุ่มจำลองหรือใช้คีย์ลูกศรของแป้นพิมพ์จากพีซีในการควบคุมการใช้งาน และแสดงผลบนจอแอลซีดีที่จำลองขึ้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5.2 ปัญหาและแนวทางการแก้ไข

ปัญหาในการทำงานต่างๆซึ่งทำให้งานล่าช้าออกไป ได้เสนอแนวทางการแก้ไขดังนี้

ปัญหา	แนวทางแก้ไข
เนื้อหาที่ใช้สำหรับ ใช้งานจริงของซีพียู MC 68HC11 ซึ่งส่วนใหญ่ยังเป็นตำราภาษาอังกฤษ จึงทำให้เกิดปัญหาล่าช้าในการเขียนโปรแกรมควบคุม	ศึกษาแนวทางการทำงาน จากบทความที่เป็นภาษาไทย ประกอบร่วมกับการแปลภาษาอังกฤษและปรึกษาอาจารย์ที่ปรึกษา
เขียนโปรแกรมถูกต้องแต่ไม่สามารถที่จะควบคุมวงจรหรือฮาร์ดแวร์ได้	ให้ทดลองการทำงานของฮาร์ดแวร์ โดยใช้การจำลองสัญญาณควบคุมจากภายนอกให้สามารถทำงานได้สมบูรณ์ก่อน แล้วจึงนำไปอินเตอร์เฟสกับซีพียู
การเขียนโปรแกรม ซึ่งอาศัยการ DOWN LOAD โปรแกรม จะ EXECUTE ในพื้นที่ของ RAM และเมื่อโปรแกรมมีขนาดใหญ่ การทำงานและการแสดงผลจะผิดพลาด	เมื่อโปรแกรมมีขนาดใหญ่จะไปทับกับส่วนที่เป็น บัฟเฟอร์ ของระบบ จึงทำให้การทำงานผิดพลาด ให้เปลี่ยนขนาดความจุของ RAM ให้มากขึ้น
ระบบตรวจจับและแสดงผลแรงดันไฟตรง 0-500 โวลท์ โดยการเปลี่ยนย่านวัดอัตโนมัติ เมื่อนำวงจรมาควบคุม โดยอินเตอร์เฟสกับ พอร์ท 8255 วงจรไม่สามารถทำงานได้ตามโปรแกรมที่เขียน	ศึกษาการออกแบบวงจร ที่ใช้กับพอร์ท 8255 ใหม่ และพยายามแก้ไขให้สมบูรณ์
การเขียนโปรแกรมบนพีซี ล่าช้ากว่าการโปรแกรมมอนิเตอร์ ทำให้ไม่สามารถที่ทดสอบการเขียนโปรแกรมรับ-ส่งค่าจาก RS-232C ของโปรแกรมมอนิเตอร์ได้	ให้ผู้เขียนโปรแกรมทั้ง 2 ส่วน ศึกษาวิธีการสื่อสารสัญญาณจากพอร์ท RS-232C ร่วมกันจากทฤษฎี

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5.3 แนวทางในการพัฒนา

ในการจัดทำปริญญานิพนธ์ระบบตรวจจับและควบคุมด้วยไมโครคอมพิวเตอร์ ยังมีการทำงานในหลายส่วนที่ควรทำการพัฒนาระบบให้มีประสิทธิภาพมากยิ่งขึ้น ดังต่อไปนี้

- พัฒนาโปรแกรมควบคุมการใช้งานระบบด้วยเครื่องไมโครคอมพิวเตอร์ ด้วยโปรแกรมบนไมโครซอฟท์วินโดวส์ ให้มี OPTION ในการเลือกใช้งานให้มากกว่านี้ เช่น มีการบันทึกค่าที่ใช้ในการควบคุมระบบ ในแชลแนลต่างๆ เอาไว้ในส่วนของ การเก็บข้อมูลสำรอง เพื่อที่จะได้นำค่าเดิมที่เซ็ทไว้มาใช้ใหม่ได้ในทันที และมี OPTION สำหรับสาธิตการใช้ระบบ ในแต่ละขั้นตอน โดยมีลักษณะเป็นภาพให้ดูเข้าใจง่าย

- พัฒนาโปรแกรมมอนิเตอร์ให้มีการทำงานหลายๆ อย่างได้ในเวลาเดียวกันตามที่ต้องการ เช่นในขณะที่ทำการควบคุมอุปกรณ์ไฟฟ้าอยู่ ให้ซีพียูตรวจจับสัญญาณภายนอกเข้ามาที่ระบบและแสดงผลได้ในทันที โดยอาศัยหลักการแบ่งเวลาการทำงานของซีพียู หรือที่เรียกว่า มัลติทรากซ์

- ทำการสร้างวงจรตรวจจับในลักษณะต่างๆ โดยผ่านทรานสดิวเซอร์ ก่อนที่จะเข้าในระบบตรวจจับ เพื่อใช้ในการประยุกต์ใช้งานจริง หรือใช้เป็นการสาธิตจริงก็ได้

- พัฒนาระบบตรวจจับให้มีลักษณะเป็นการควบคุมป้อนกลับ เช่น รับสัญญาณตรวจจับจากภายนอกเข้ามาที่ระบบแล้วให้ระบบส่งสัญญาณกลับไปควบคุมที่จุดนั้นให้มีสภาวะเช่นเดิม

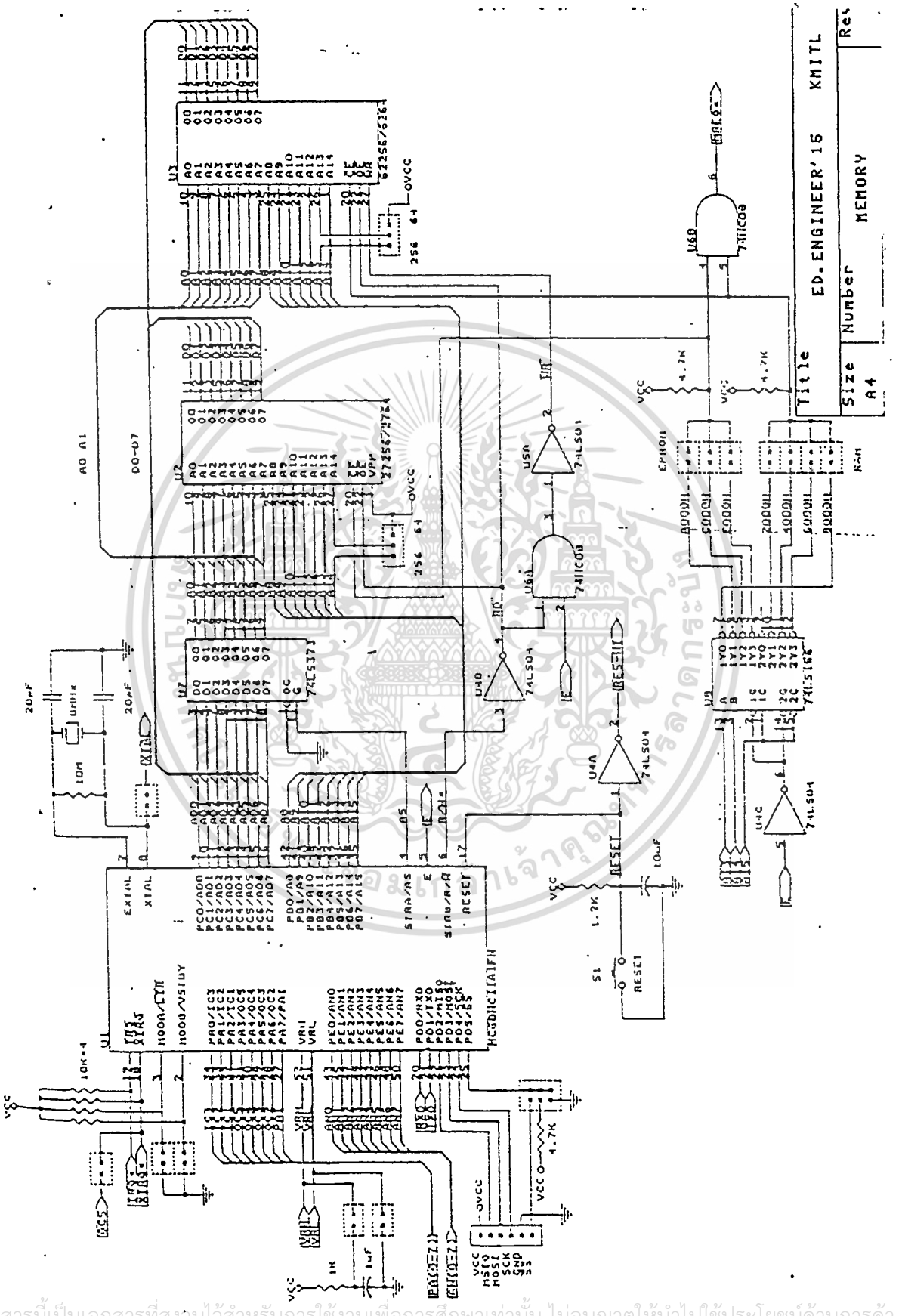
- สามารถนำระบบชุดสาธิตนี้ไปสร้างเป็นเครื่องต้นแบบที่ใช้สำหรับงานควบคุมจริงๆ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

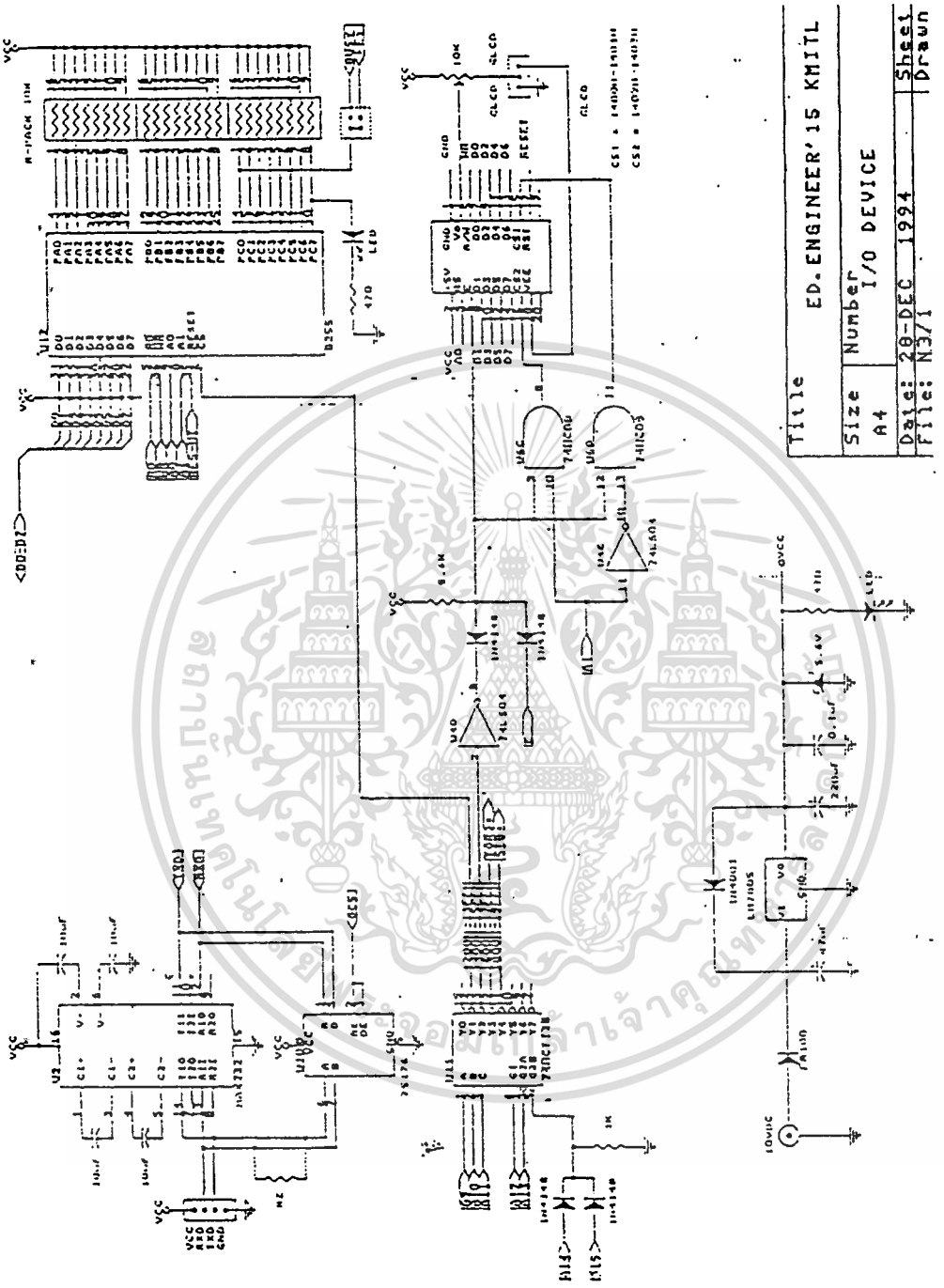
ภาคผนวก ก

วงจรส่วนต่างๆ ของระบบตรวจจับและควบคุม

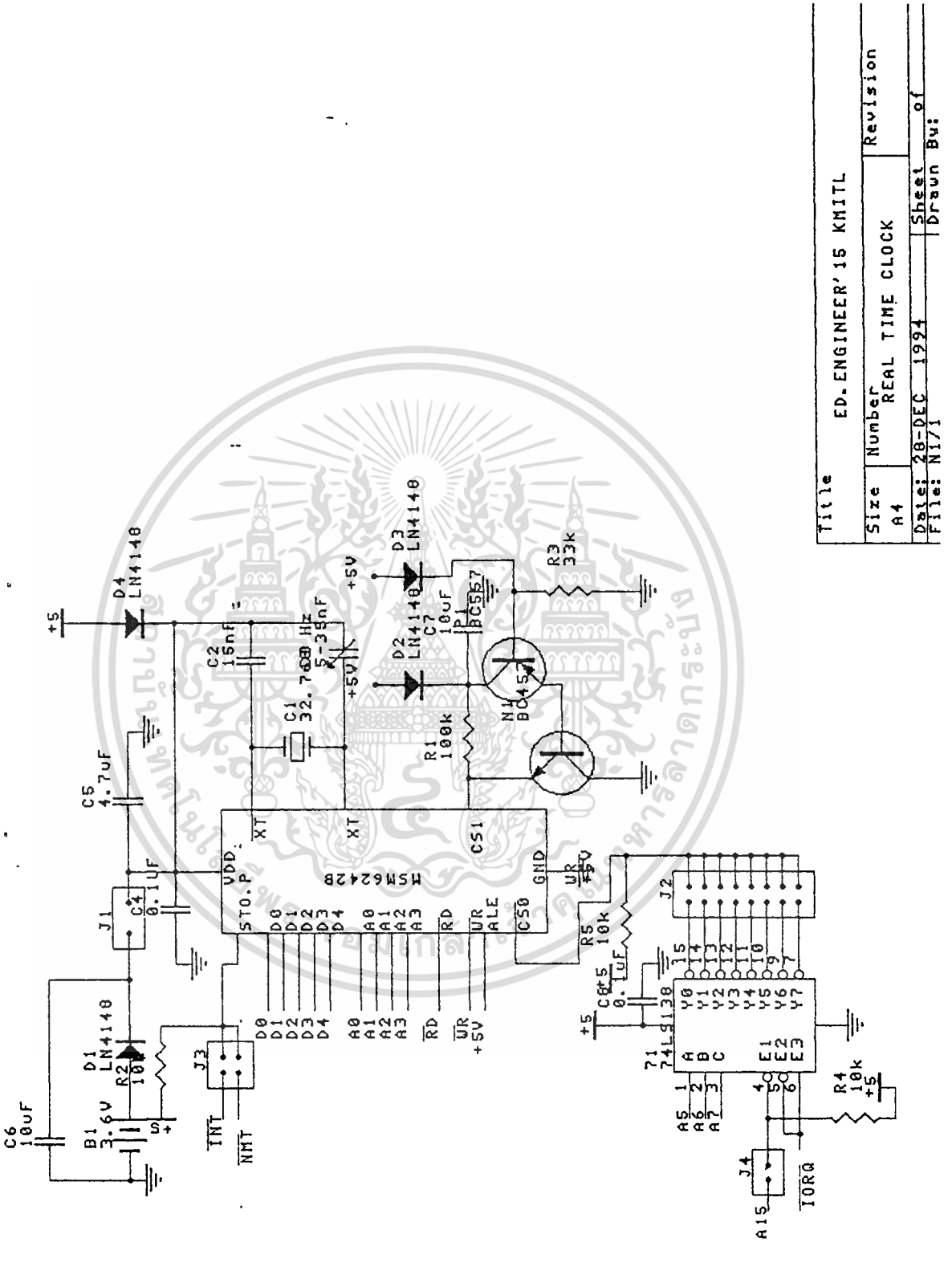
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



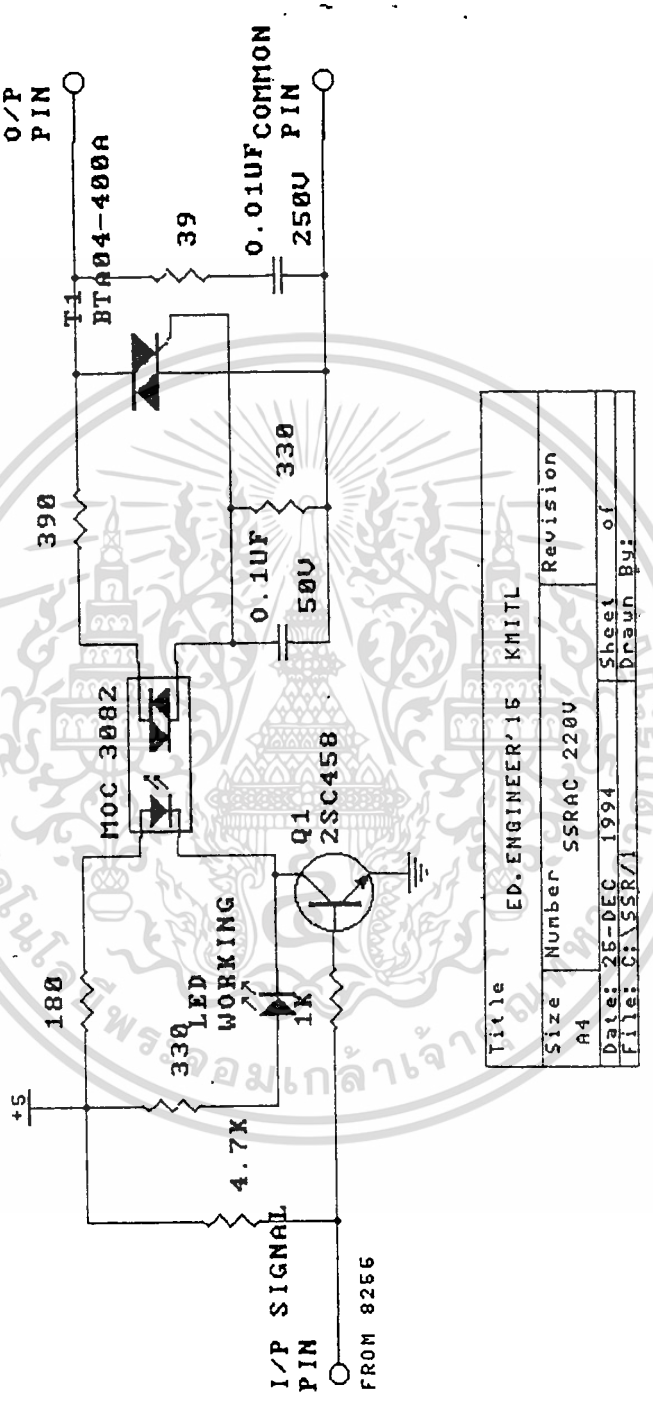
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



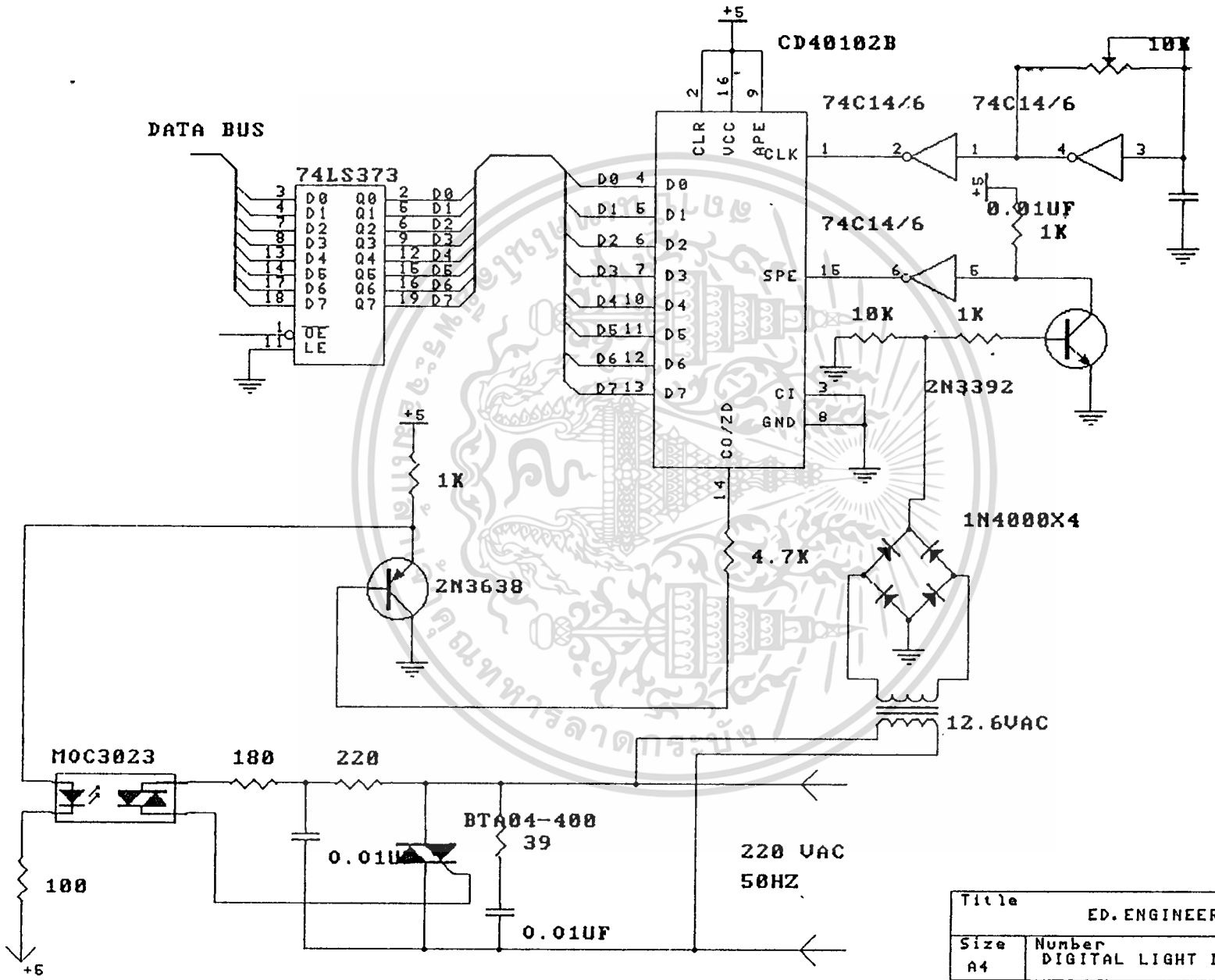
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



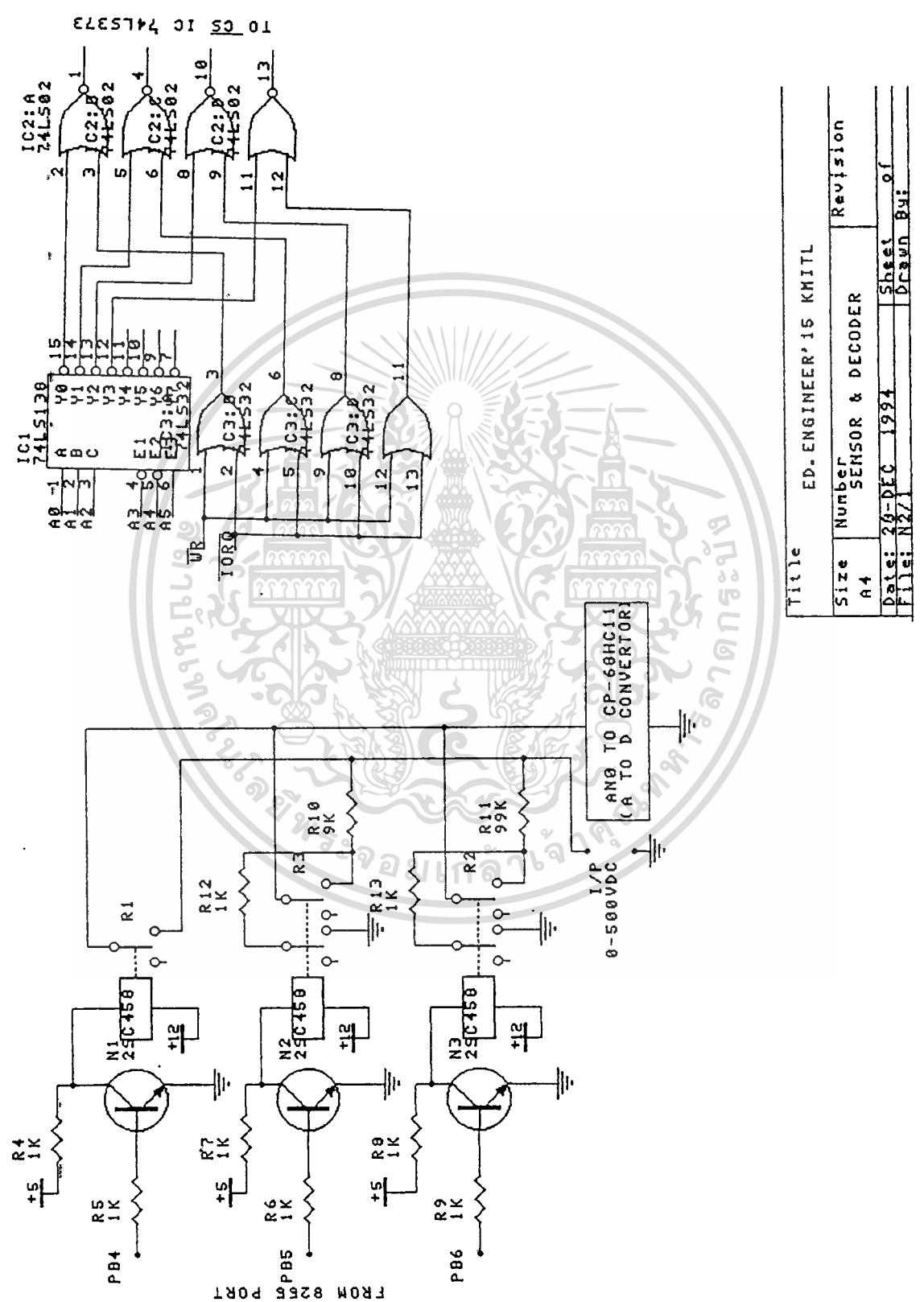
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



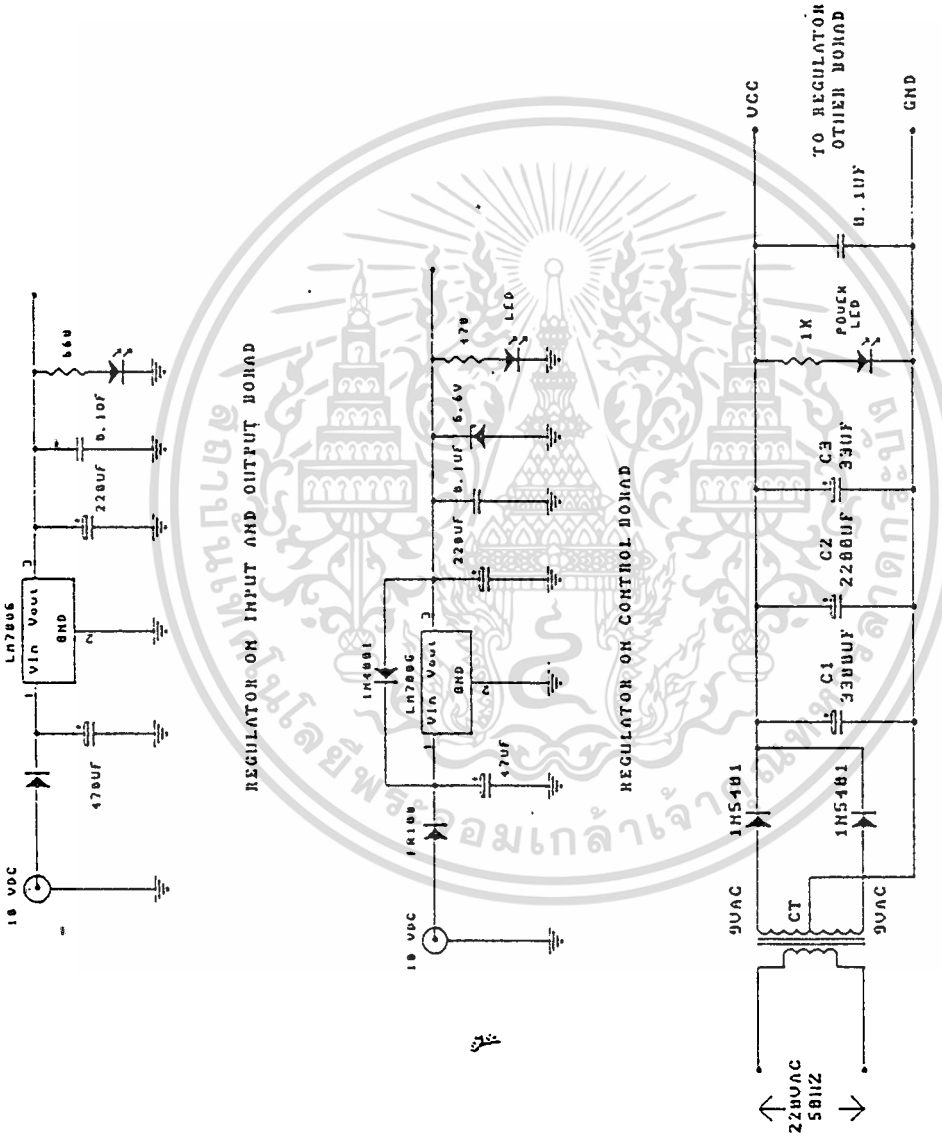
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



Title	ED.ENGINEER'15 KMITL	
Size A4	Number DIGITAL LIGHT INTENSITY	Revision
Date: 25-DEC 1994	Sheet of	



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

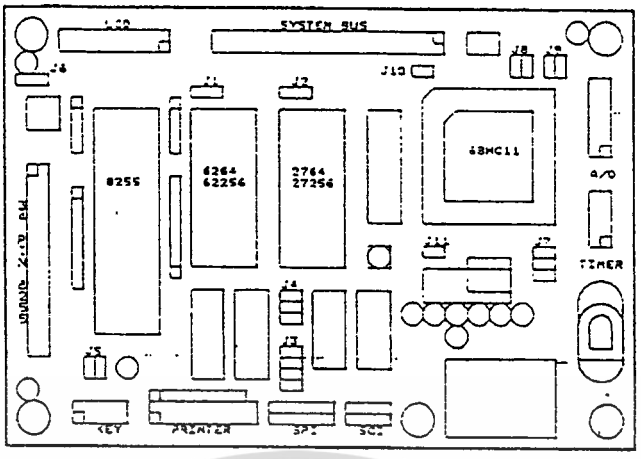


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

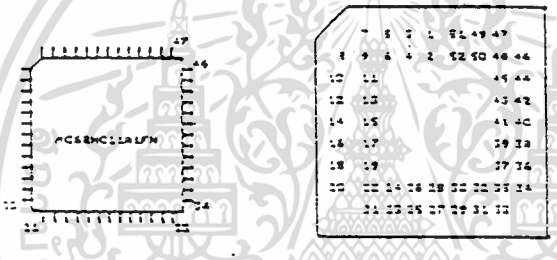
ภาคผนวก ข

ลักษณะการวางอุปกรณ์และการจัดวางขาคอนเนคเตอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



ลักษณะการวางอุปกรณ์บนบอร์ด CP-68HC11

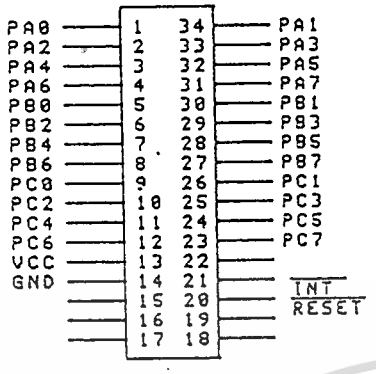


SOCKET 52 PIN (TOP VIEW)

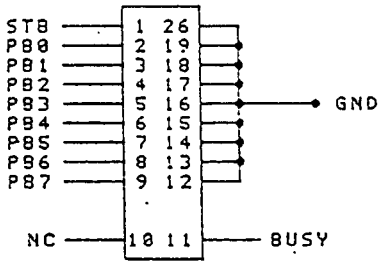
ลักษณะการจัดวางขาไอซี MC68HC11

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

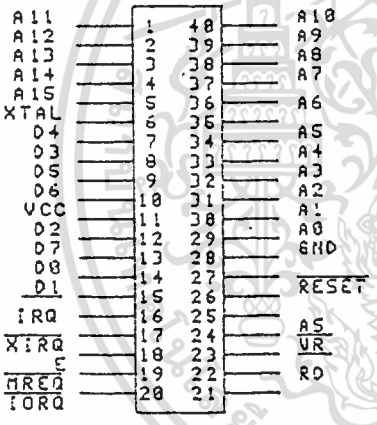
ลักษณะการจัดวางขาคอนเนคเตอร์บนบอร์ด CP-68HC11

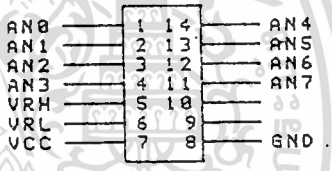


8255 CONNECTOR
34 PIN



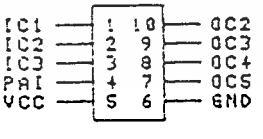
PRINTER PORT

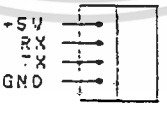




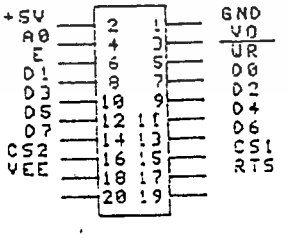
A-D CONNECTOR

40PIN Z80 BUS



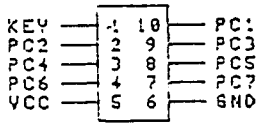


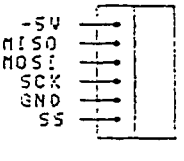
SCI



TIMER 68HC11

XEYBORAD 4X4 OR 3X4
8255





LCD

SPI

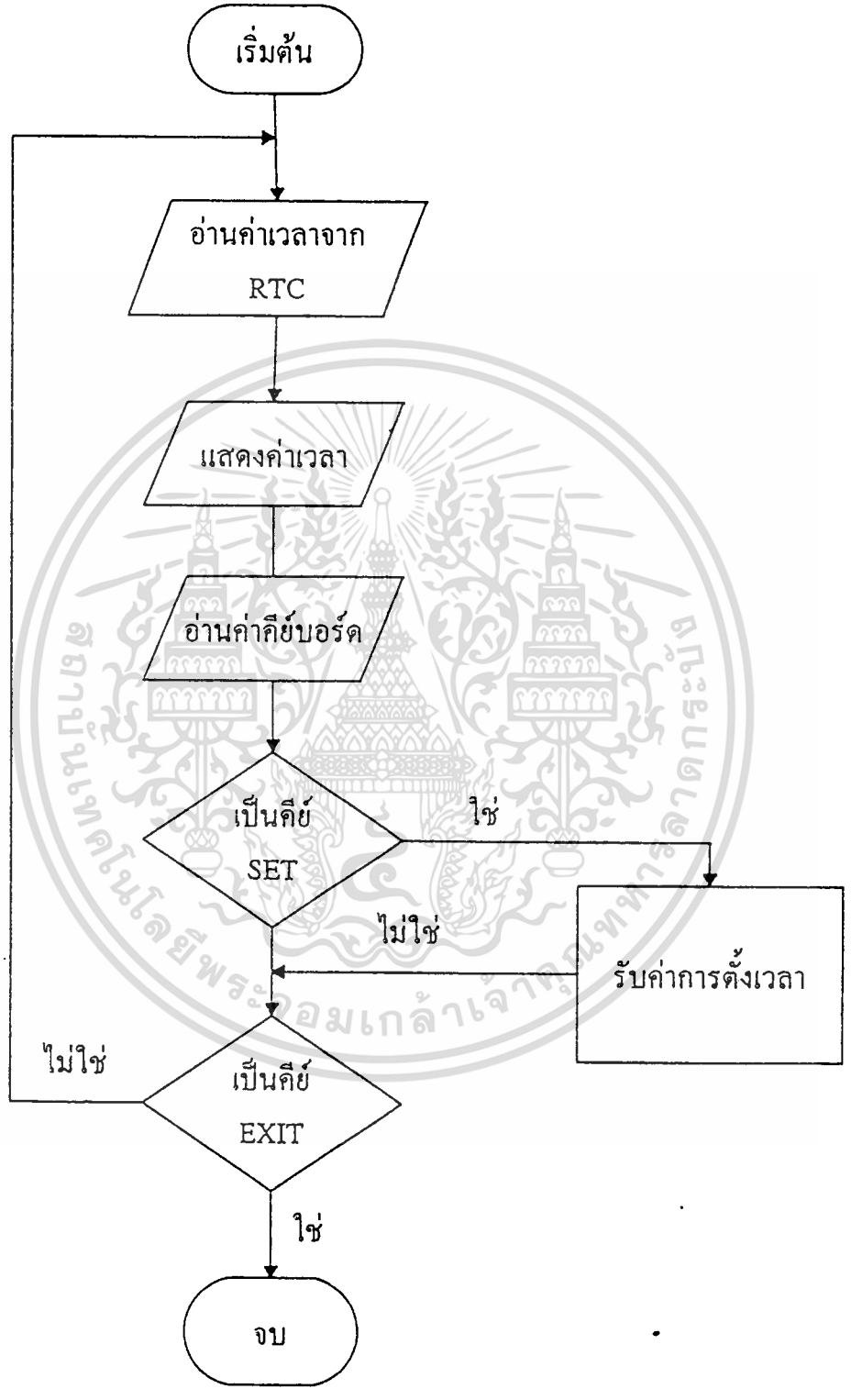
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ค

โฟลว์ชาร์ตระบบตรวจจับและควบคุมด้วยเครื่องไมโครคอมพิวเตอร์

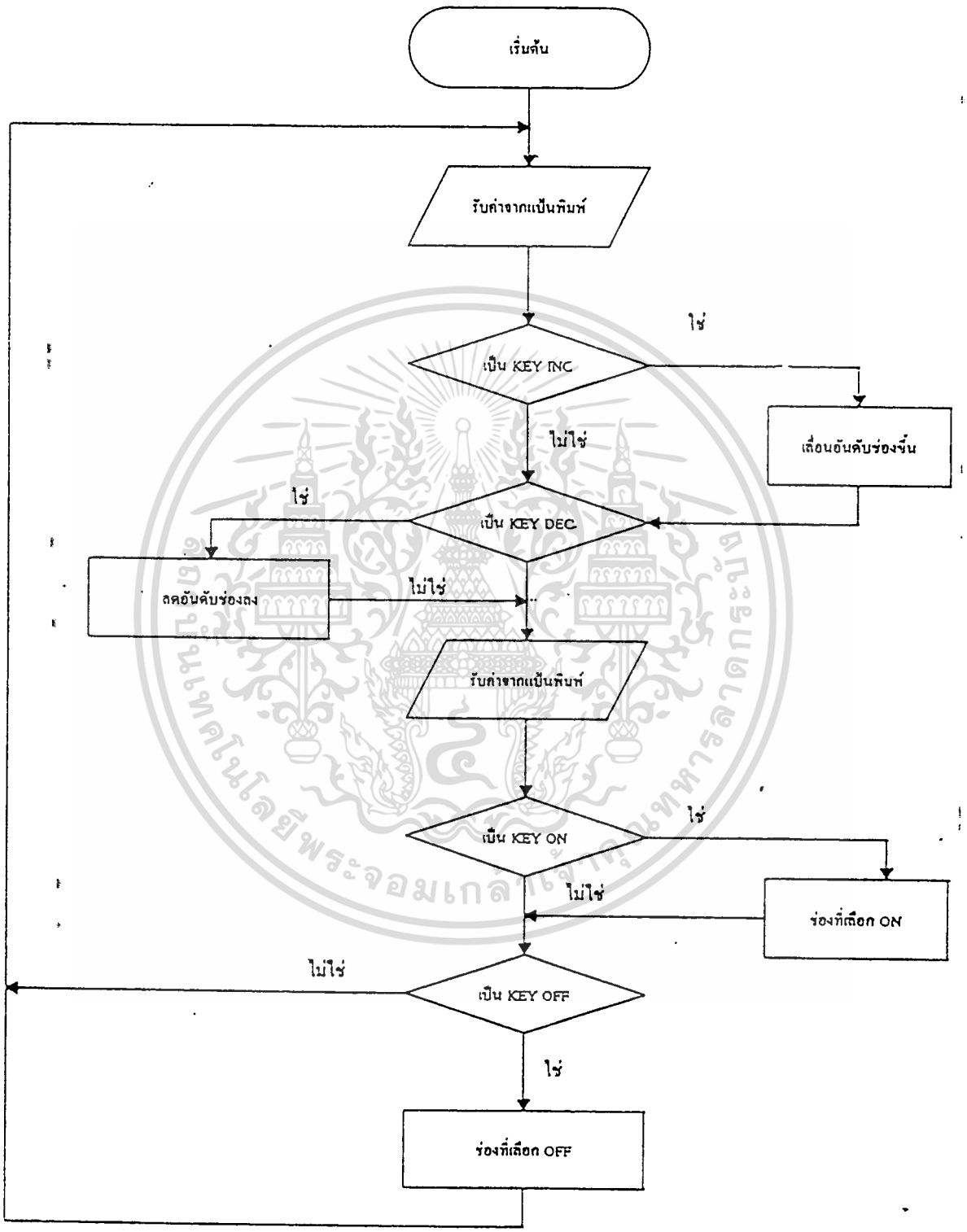
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

แสดงการทำงานของฟังก์ชั่นตั้งค่าแสดงเวลา



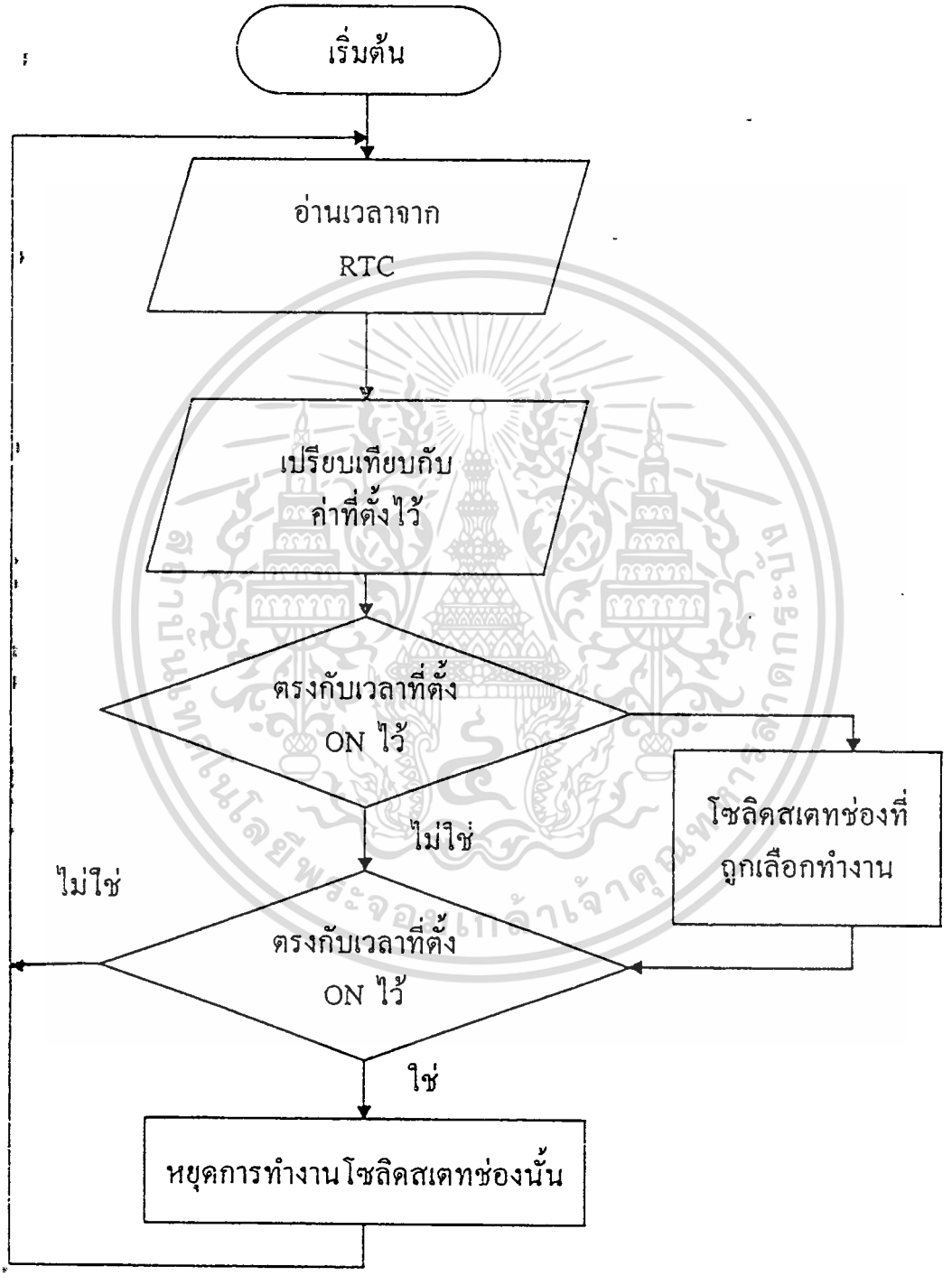
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

แสดงการทำงาน การเปิดปิดอุปกรณ์ไฟฟ้า

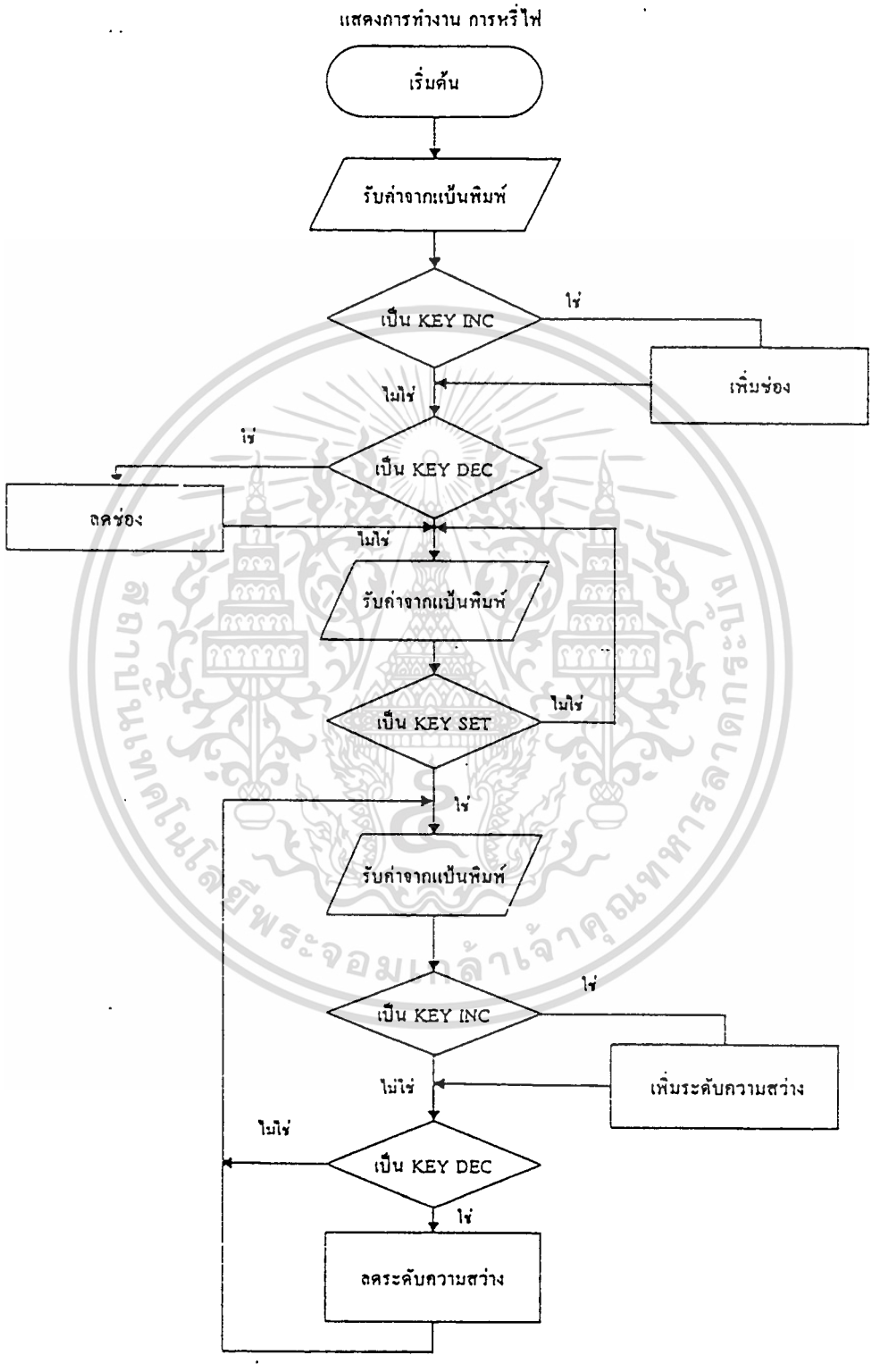


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

แสดงการทำงานการควบคุมการเปิดปิดอุปกรณ์ไฟฟ้า

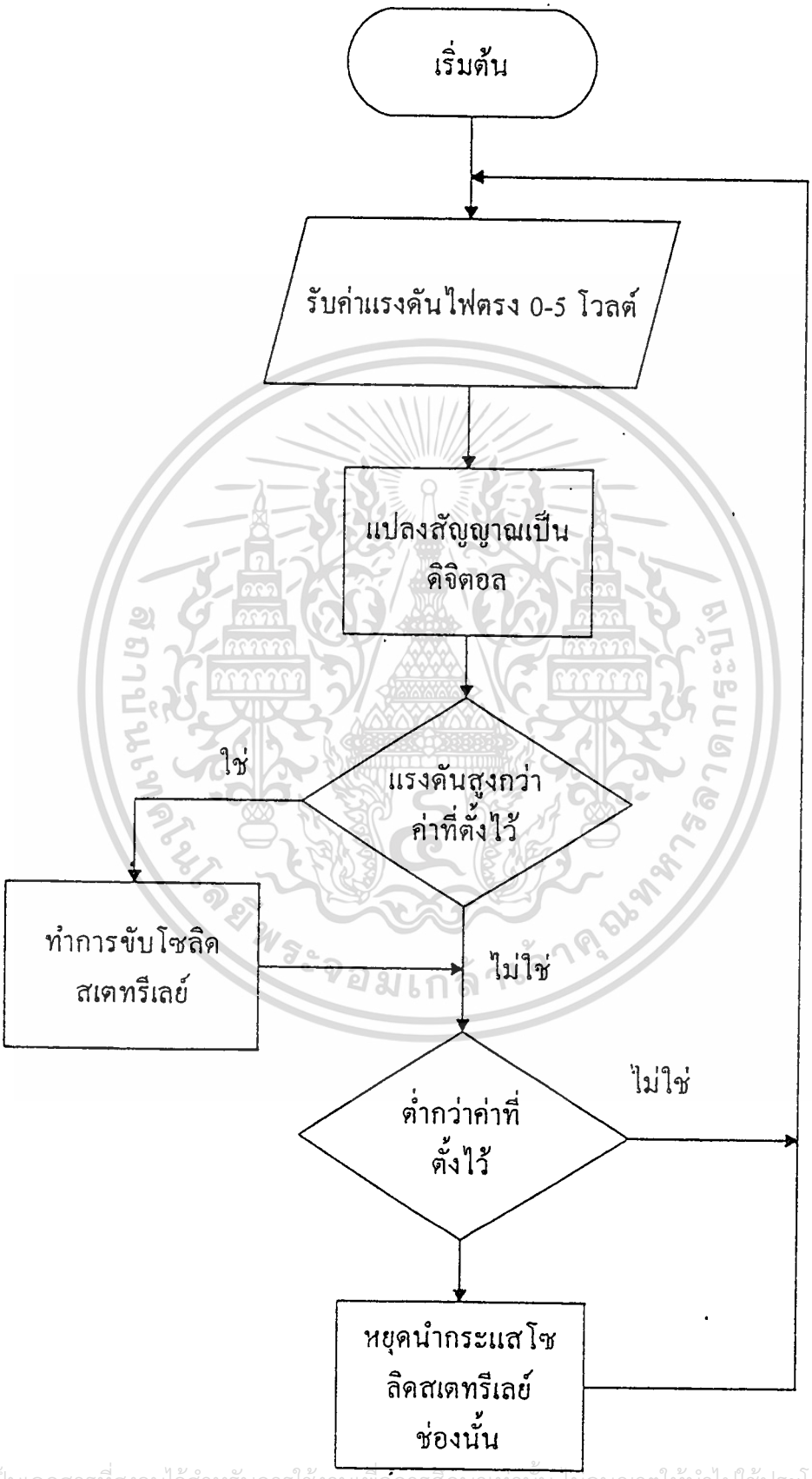


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



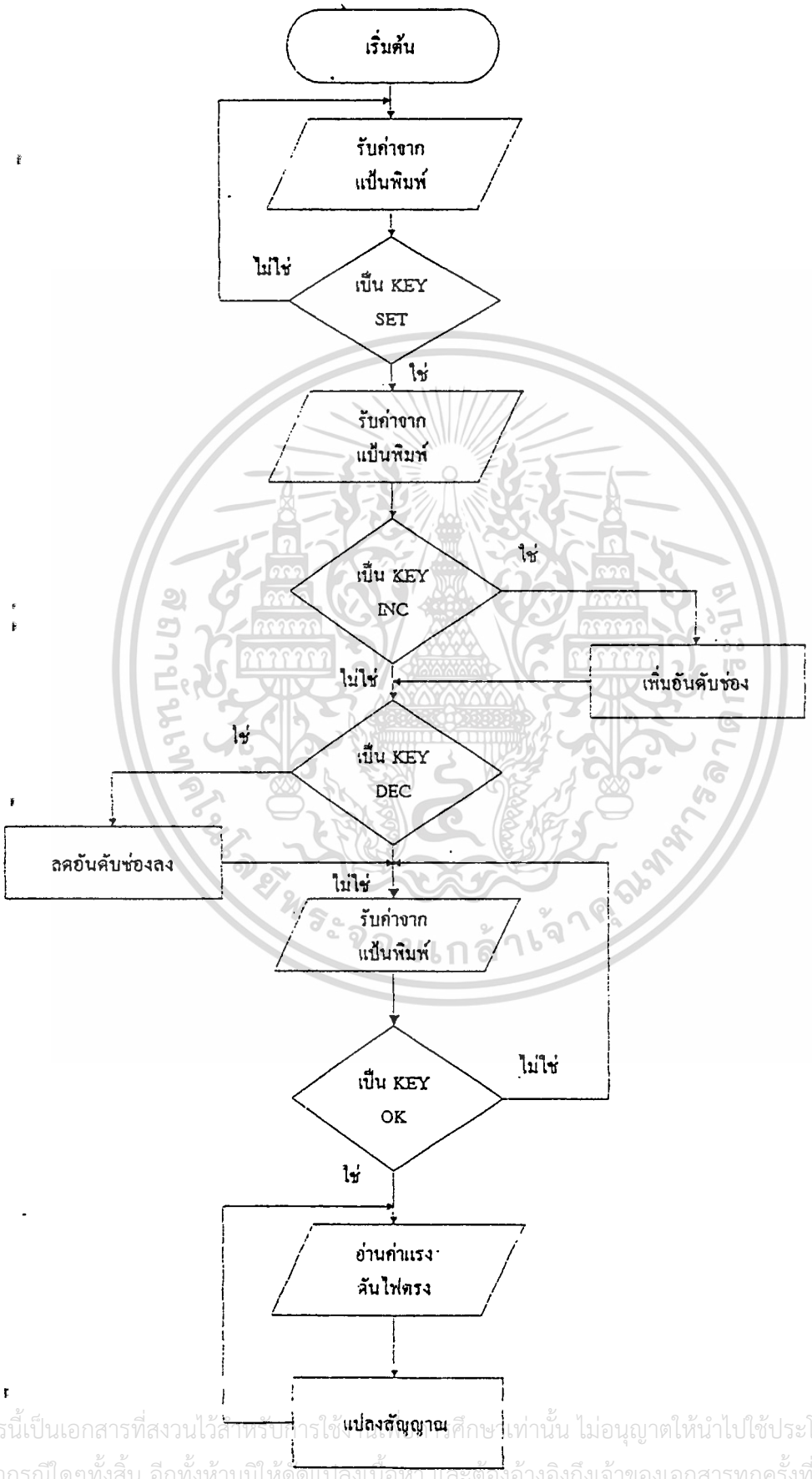
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

แสดงการทำงาน การควบคุมโซลิดสเตทรีเลย์



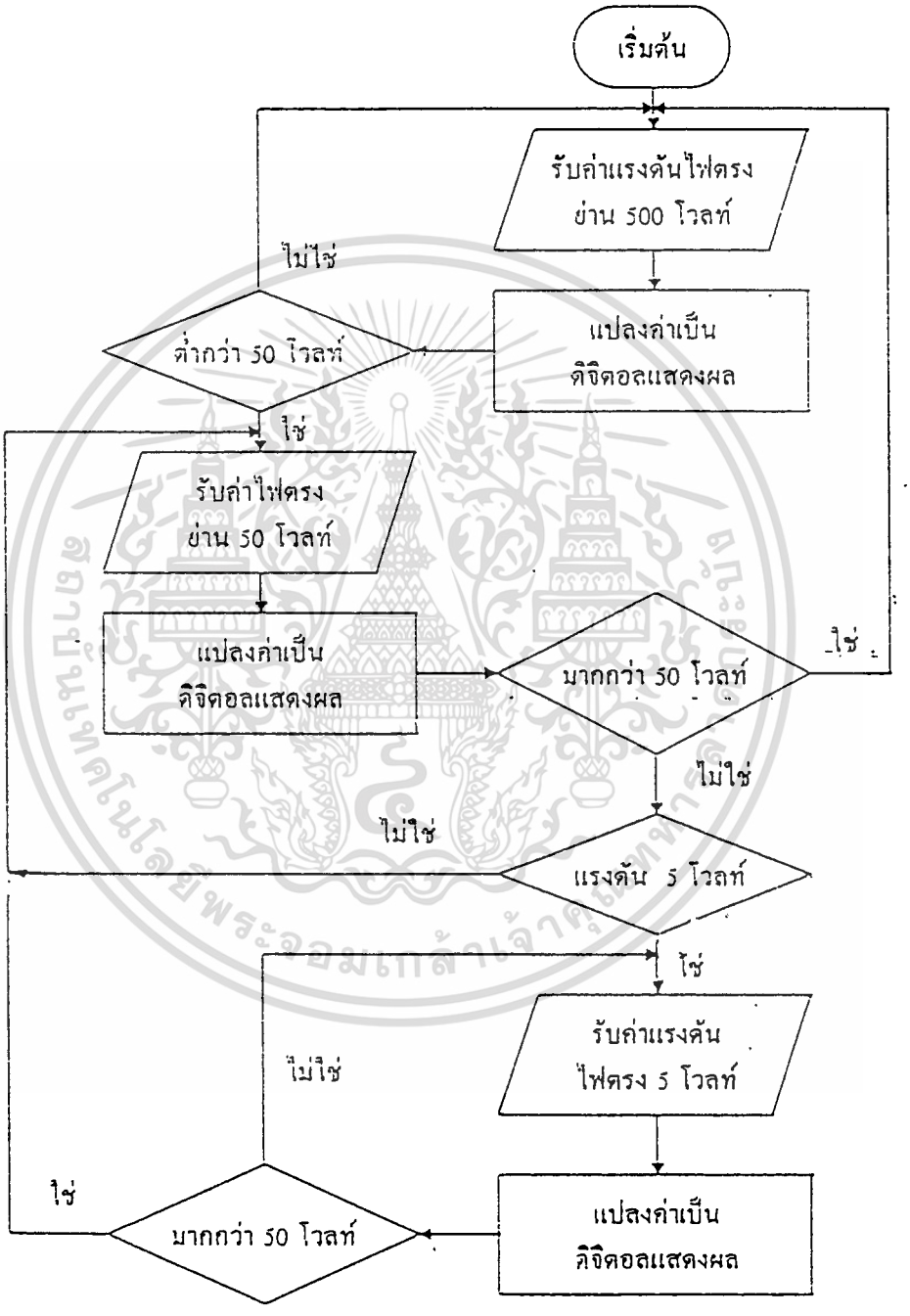
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

แสดงการทำงานของโวลท์มิเตอร์เลือกช่องได้



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

แสดงการทำงานของวงจร
วัดไฟตรงเลือกย่านอัตโนมัติ 0 - 500 โวลท์



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ง

โปรแกรมมอนิเตอร์บนบอร์ด CP-68HC11

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

```
          Freeware assember ASxx.EXE Ver 1.03.
        ****************************************
        *            MONITOR PROGRAM           *
        *      PC - SENSORS AND CONTROLLER     *
        *                68HC11                *
        ****************************************

          ******** VARIABLE SET EQUATES *********

REGBS       EQU   $1000          ; Start Of Registers
TMSK2       EQU   REGBS+$24      ; Timer Mark 2
OPTION      EQU   REGBS+$39      ; Option Reg.
ADCTL       EQU   $30            ; Offset A/D Control Register
EOT         EQU   $04            ; End Of Text
CODE        EQU   $A3            ; Code Power On
SET1        EQU   $1400          ; Port Set LCD
WR1LCD      EQU   $1401          ; Port Write LCD
CALL        EQU   $FF00          ; System Call
DIMMER_OUT  EQU   $1820          ; Dimmer Port
OUT_CTRL    EQU   $1200          ; Out Port On/Off Device
SSR_CTRL    EQU   $1201          ; SSR Control Port
KEY_PORT    EQU   $1202          ; Input Key From Keyboard
PORT_CTRL   EQU   $1203          ; Control Port 8255
BAUD        EQU   $102B
SCCR1       EQU   $102C
SCCR2       EQU   $102D
SCSR        EQU   $102E
SCDR        EQU   $102F
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
JTOC4           RMB  3                  ; Timer Output Compare 4
JTOC3           RMB  3                  ; Timer Output Compare 3
JTOC2           RMB  3                  ; Timer Output Compare 2
JTOC1           RMB  3                  ; Timer Output Compare 1
JTIC3           RMB  3                  ; Timer Input Capture 3
JTIC2           RMB  3                  ; Timer Input Capture 2
JTIC1           RMB  3                  ; Timer Input Capture 1
JRTI            RMB  3                  ; Real Time Interrupt
JIRQ            RMB  3                  ; IRQ
JXIRQ           RMB  3                  ; XIRQ
JSWI            RMB  3                  ; SWI
JILLOP          RMB  3                  ; Illegal Opcode
JCOP            RMB  3                  ; COP Fail
JCLM            RMB  3                  ; COP Clock Monitor Fail

                ORG  $2200

*****************************************
*        SYSTEM  MONITOR  PROGRAM       *
*           ON EXTERNAL RAM             *
*****************************************

SCAN_BUF        RMB  1
                RMB  30                 ; User Stack
USTACK          RMB  20                 ; Of Monitor
STACK           RMB  1                  ; Program

*======== CURRENT TIME  BUFFER ========*

DC_SEC          RMB  1
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
DC_MIN          RMB  1
DC_HOUR         RMB  1

DC_WEEK         RMB  1
DC_DAY          RMB  1
DC_MONTH        RMB  1
DC_YEAR         RMB  1

*========= BUFFER ON HEX TIME ========*

ON_DAY1         RMB  1                  ; CH1 On Hex Time
ON_MONTH1       RMB  1                  ; Of Day,Month
ON_HOUR1        RMB  1                  ; Hour,Minute
ON_MIN1         RMB  1                  ; And Second
ON_SEC1         RMB  1                  ;
ON_DAY2         RMB  1                  ; CH2 On Hex Time
ON_MONTH2       RMB  1                  ; Of Day,Month
ON_HOUR2        RMB  1                  ; Hour,Minute
ON_MIN2         RMB  1                  ; And Second
ON_SEC2         RMB  1                  ;
ON_DAY3         RMB  1                  ; CH3 On Hex Time
ON_MONTH3       RMB  1                  ; Of Day,Month
ON_HOUR3        RMB  1                  ; Hour,Minute
ON_MIN3         RMB  1                  ; And Second
ON_SEC3         RMB  1                  ;
ON_DAY4         RMB  1                  ; CH4 On Hex Time
ON_MONTH4       RMB  1                  ; Of Day,Month
ON_HOUR4        RMB  1                  ; Hour,Minute
ON_MIN4         RMB  1                  ; And Second
ON_SEC4         RMB  1                  ;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
ON_DAY5         RMB  1                  ; CH5 On Hex Time
ON_MONTH5       RMB  1                  ; Of Day,Month
ON_HOUR5        RMB  1                  ; Hour,Minute
ON_MIN5         RMB  1                  ; And Second
ON_SEC5         RMB  1                  ;
ON_DAY6         RMB  1                  ; CH6 On Hex Time
ON_MONTH6       RMB  1                  ; Of Day,Month
ON_HOUR6        RMB  1                  ; Hour,Minute
ON_MIN6         RMB  1                  ; And Second
ON_SEC6         RMB  1                  ;
ON_DAY7         RMB  1                  ; CH7 On Hex Time
ON_MONTH7       RMB  1                  ; Of Day,Month
ON_HOUR7        RMB  1                  ; Hour,Minute
ON_MIN7         RMB  1                  ; And Second
ON_SEC7         RMB  1                  ;
ON_DAY8         RMB  1                  ; CH8 On Hex Time
ON_MONTH8       RMB  1                  ; Of Day,Month
ON_HOUR8        RMB  1                  ; Hour,Minute
ON_MIN8         RMB  1                  ; And Second
ON_SEC8         RMB  1                  ;

*======== BUFFER OFF HEX TIME ========*

OFF_DAY1        RMB  1                  ; CH1 Off Hex Time
OFF_MONTH1      RMB  1                  ; Of Day,Month
OFF_HOUR1       RMB  1                  ; Hour,Minute
OFF_MIN1        RMB  1                  ; And Second
OFF_SEC1        RMB  1                  ;
OFF_DAY2        RMB  1                  ; CH2 Off Hex Time
OFF_MONTH2      RMB  1                  ; Of Day,Month
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
OFF_HOUR2       RMB  1                  ; Hour,Minute
OFF_MIN2        RMB  1                  ; And Second
OFF_SEC2        RMB  1                  ;
OFF_DAY3        RMB  1                  ; CH3 Off Hex Time
OFF_MONTH3      RMB  1                  ; Of Day,Month
OFF_HOUR3       RMB  1                  ; Hour,Minute
OFF_MIN3        RMB  1                  ; And Second
OFF_SEC3        RMB  1                  ;
OFF_DAY4        RMB  1                  ; CH4 Off Hex Time
OFF_MONTH4      RMB  1                  ; Of Day,Month
OFF_HOUR4       RMB  1                  ; Hour,Minute
OFF_MIN4        RMB  1                  ; And Second
OFF_SEC4        RMB  1                  ;
OFF_DAY5        RMB  1                  ; CH5 Off Hex Time
OFF_MONTH5      RMB  1                  ; Of Day,Month
OFF_HOUR5       RMB  1                  ; Hour,Minute
OFF_MIN5        RMB  1                  ; And Second
OFF_SEC5        RMB  1                  ;
OFF_DAY6        RMB  1                  ; CH6 Off Hex Time
OFF_MONTH6      RMB  1                  ; Of Day,Month
OFF_HOUR6       RMB  1                  ; Hour,Minute
OFF_MIN6        RMB  1                  ; And Second
OFF_SEC6        RMB  1                  ;
OFF_DAY7        RMB  1                  ; CH7 Off Hex Time
OFF_MONTH7      RMB  1                  ; Of Day,Month
OFF_HOUR7       RMB  1                  ; Hour,Minute
OFF_MIN7        RMB  1                  ; And Second
OFF_SEC7        RMB  1                  ;
OFF_DAY8        RMB  1                  ; CH8 Off Hex Time
OFF_MONTH8      RMB  1                  ; Of Day,Month
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
OFF_HOUR8       RMB  1                  ; Hour,Minute
OFF_MIN8        RMB  1                  ; And Second
OFF_SEC8        RMB  1                  ;

*============ PSEUDO REGISTER ============*

REGD            RMB  1
REGE            RMB  1
REGC            RMB  1
REGH            RMB  1
REGL            RMB  1
REGP            RMB  1
REGI            RMB  1

*======== OTHER INTERNAL SYSTEM ==========*

SYS_FLAG        RMB  1                  ; System Internal Flag
                                        ; Enable And Dissable
                                        ; b0 Dissable Check Command

CHAN_MAN        RMB  1                  ; Store Condition On/Off
CTRL_BUF        RMB  1
POWER_ON        RMB  1                  ; Power-On Code
MAX_BUF         RMB  1
MIN_BUF         RMB  1
BUFF_ADC        RMB  4
ADC1            RMB  1
ADC2            RMB  1
ADC3            RMB  1
ADC4            RMB  1
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
*====== ASCII DEMMER LIGHT LEVEL ======*

A_DIMMER1     RMB   20
A_DIMMER2     RMB   20
A_DIMMER3     RMB   20
A_DIMMER4     RMB   20

*====== CONDITION ON/OFF CONTROL ======*

CHANNEL0      RMB    1
CHANNEL1      RMB    1
CHANNEL2      RMB    1
CHANNEL3      RMB    1
CHANNEL4      RMB    1
CHANNEL5      RMB    1
CHANNEL6      RMB    1
CHANNEL7      RMB    1
CHANNEL8      RMB    1
ADJ_BUF       RMB  100

              ORG  $C000

************ POWER - UP **************

POWER         CLRB                    ; Power - On Delay
POWER1        DECB                    ; Count Down Delay
              NOP
              NOP
              CMPB  #$00
              BNE  POWER1
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
******* INITIAL INTERNAL SYSTEM *******

INIT_SYSTEM   LDS  #STACK
              LDAA  #$88
              STAA  PORT_CTRL
              LDAA  #$93
              STAA  OPTION
              CLRA
              STAA  TMSK2
SCAN_SCI      LDAA  #$30              ; Set Baud Rate 9600
              STAA  BAUD              ; 64 E cycles
              CLR   SCCR1             ; Set 8 Bit Data
              LDAA  #$0C
              STAA  SCCR2
              LDX   #SYS_FLAG
              BCLR  0,X $01
              LDAB  #$7E
              STAB  JIRQ
              LDX   #INT_IRQ
              STX   JIRQ+1
              CLI
              LDX   #DIMMER_OUT
              LDAA  #$FF
INIT_DIM      STAA  0,X
              INX
              CPX   #DIMMER_OUT+4
              BNE   INIT_DIM
CHECK_ON      LDAA  POWER_ON
              CMPA  #CODE
              BNE   CHECK_ON1
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                JMP   EXIT_EXE
CHECK_ON1       LDX   #$2200
                CLRA
CLEAR_CON       STAA  0,X
                INX
                CPX   #$3FFF
                BNE   CLEAR_CON
                LDAA  #CODE
                STAA  POWER_ON
                LDX   #A_DIMMER1
                LDAB  #' '
PUT_ADIM        STAB  0,X
                INX
                CPX   #A_DIMMER4+20
                BNE   PUT_ADIM
                LDX   #CHANNEL0
                LDAB  #$00
CH_VALUE        STAB  0,X
                INX
                INCB
                CMPB  #$09
                BNE   CH_VALUE
                LDX   #ADJ_BUF
                LDAB  #$64
                CLRA
PUT_VALUE       STAA  0,X
                INX
                INCA
                DAA
                DECB
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

```
                BNE   PUT_VALUE

********* SHOWN STRING ON LCD *********

                JSR   INIT_LCD
                LDAA  #$01
                LDX   #PRJ1_TAB
                JSR   WR_LINE
                LDAA  #$02
                LDX   #PRJ2_TAB
                JSR   WR_LINE
                JSR   DELAYL
                JSR   DELAYL
                JSR   DELAYL
SEL_CHECK       LDX   #ON_PC_TAB
                LDAA  #$01
                JSR   WR_LINE
                LDX   #ON_BOARD_TAB
                LDAA  #$02
                JSR   WR_LINE

SELECT_KEY      LDAA  #$0B
                STAA  KEY_PORT
                LDAA  KEY_PORT
                ANDA  #$F0
                CMPA  #$70
                BNE   BOARD
                CLRB
                STAB  SCAN_BUF
                BRA   EXIT_EXE
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
BOARD          LDAA  #$07
               STAA  KEY_PORT
               LDAA  KEY_PORT
               ANDA  #$F0
               CMPA  #$70
               BNE   SELECT_KEY
               LDAB  #$AA
               STAB  SCAN_BUF
EXIT_EXE       JSR   INIT_LCD
               LDAA  #$01
               LDX   #READY_TAB
               JSR   WR_LINE
               LDAA  #$02
               LDX   #OPTION_TAB
               JSR   WR_LINE

********** SCAN KEYBOARD 4*4 **********
******** SCAN SERIAL PORT PC **********

SCAN_KEY       LDAB  SCAN_BUF
               CMPB  #$00
               BNE   SCAN_BOARD
               BRA   RECEIVE
SCAN_BOARD     LDAB  #$04
               LDAA  #$F7
               STAA  REGE
SCAN_KEY1      STAA  KEY_PORT
               LDAA  KEY_PORT
               ANDA  #$F0
               CMPA  #$F0
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                BNE  CODE_KEY
                ROR  REGE
                LDAA  REGE
                DECB
                BNE  SCAN_KEY1
                BRA  SCAN_KEY
CODE_KEY        JSR  DELAYL
                PSHA
                LDAA  REGE
                ANDA  #$0F
                STAA  REGE
                PULA
                ANDA  #$F0
                ORAA  REGE
                LDX  #KEY_TAB
                LDAB  #$00
                STAB  REGD
                LDAB  #$10
FIND_CODE       CMPA  0,X
                BEQ  CHECK_CODE
                INC  REGD
                INX
                DECB
                BNE  FIND_CODE
                JMP  SCAN_KEY
CHECK_CODE      LDAA  REGD
                LDX  #SYS_FLAG
                LDAB  0,X
                BITB  #$01
                BEQ  EXE_KEY
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
END_CHECK       NOP
                RTS

RECEIVE         LDAA  SCSR
                BITA  #$20
                BEQ   RECEIVE
                LDAA  SCDR
                LDX   #SCI_TAB
                LDAB  #$00
                STAB  REGD
                LDAB  #$10
FIND_KEY        CMPA  0,X
                BEQ   END_FIND
                INC   REGD
                INX
                DECB
                BNE   FIND_KEY
                BRA   RECEIVE
END_FIND        LDAA  REGD
                LDX   #SYS_FLAG
                LDAB  0,X
                BITB  #$01
                BEQ   EXE_KEY
                NOP
                RTS

********* EXECUTE  COMMAND KEY ********

EXE_KEY         CMPA  #$06
                BNE   EXE_KEY1
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                JSR   ADC_KEY
                BRA   END_EXE
EXE_KEY1        CMPA  #$07
                BNE   EXE_KEY2
                JSR   SETCLK_KEY
                BRA   END_EXE
EXE_KEY2        CMPA  #$08
                BNE   EXE_KEY3
                JSR   AUTOS_KEY
                BRA   END_EXE
EXE_KEY3        CMPA  #$09
                BNE   EXE_KEY4
                JSR   DIMMER_KEY
                BRA   END_EXE
EXE_KEY4        CMPA  #$0A
                BNE   EXE_KEY5
                JSR   PRGRM_KEY
                BRA   END_EXE
EXE_KEY5        CMPA  #$0B
                BNE   EXE_KEY6
                JSR   MANUAL_KEY
                BRA   END_EXE
EXE_KEY6        CMPA  #$0C
                BNE   EXE_KEY7
                JSR   SEN_CTRL
                BRA   END_EXE
EXE_KEY7        CMPA  #$0F
                BNE   END_EXE
                CLRA
                STAA  POWER_ON
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                JMP  POWER
END_EXE         JMP  EXIT_EXE

*****************************************
*   COMMAND  KEY  EXECUTE  PROGRAM      *
*****************************************

****** ANALOG TO DIGITAL DISPLAY ******

ADC_KEY         LDX  #SYS_FLAG
                BSET  0,X $01
                LDAA  #$01
                LDX   #SENSOR_TAB
                JSR  WR_LINE
                LDX  #CH_SENSOR
                LDY  #LINE2
                JSR  M_ASCII
                LDX  #CHANNEL1
                STX  BUF2
LOOP_ADC        LDX  BUF2
                LDY  #LINE2
                LDAB  0,X
                ANDB  #$0F
SHOWN_CH        STAB  REGC
                LDX  #NUM_TAB
                ABX
                LDAA  0,X
                LDX  #LINE2
                STAA  $06,X
                LDAA  #$02
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                LDX   #LINE2
                JSR   WR_LINE
                JSR   SCAN_KEY
INC_CH          CMPA  #$04
                BNE   DEC_CH
                LDX   BUF2
                CPX   #CHANNEL3
                BEQ   END_INC_CH
                INX
END_INC_CH      STX   BUF2
                BRA   END_CH_ADC
DEC_CH          CMPA  #$05
                BNE   OK_CH
                LDX   BUF2
                CPX   #CHANNEL1
                BEQ   END_DEC_CH
                DEX
END_DEC_CH      STX   BUF2
                BRA   END_CH_ADC
OK_CH           CMPA  #$02
                BNE   END_CH_ADC
                LDAB  REGC
                ORAB  #$20
                LDX   #REGBS
                STAB  ADCTL,X
FLAG            BCLR  ADCTL,X $80
                LDX   #BUFF_ADC
                LDY   #$1031
READ            LDAA  0,Y
                STAA  0,X
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                INX
                INY
                CPY  #$1035
                BNE  READ

*=====================================*
* $FF  =  255  =  5V  ;  1  =  0.0196V *
*       ACT * 196/10  = ADCTUAL mV     *

                LDX  #BUFF_ADC
                LDAA  0,X
                LDAB  #196
                MUL
                STD  $2147
                LDAB  #$22
                JSR  CALL
                LDX  #LINE2
                LDD  $214A
                STD  $0B,X
                LDD  $214C
                STD  $0D,X
                LDAA  $214E
                STAA  $0E,X
                LDAA  #$02
                JSR  WR_LINE
                LDAA  #$0E
                STAA  KEY_PORT
                LDAA  KEY_PORT
                ANDA  #$F0
                CMPA  #$D0
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                BEQ   EXIT_ADC
                LDAA  SCDR
                CMPA  #$45
                BEQ   EXIT_ADC
                JSR   DELAYL
                JSR   DELAYL
                BRA   FLAG
END_CH_ADC      JMP   LOOP_ADC
EXIT_ADC        LDX   #SYS_FLAG
                BCLR  0,X $01
                RTS

**** SET AND DISPLAY  CURRENT TIME ****

SETCLK_KEY      LDX   #SYS_FLAG
                BSET  0,X $01
                LDAA  #$01
                LDX   #SETCLK_TAB
                JSR   WR_LINE
                LDAA  #$02
                LDX   #SETUP_TAB
                JSR   WR_LINE
                JSR   DELAYL
                JSR   DELAYL
                JSR   DELAYL
                LDX   #DATE_TAB
                LDY   #LINE1
                JSR   M_ASCII
                LDX   #TIME_TAB
                LDY   #LINE2
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                JSR  M_ASCII
                LDAA  #$01
                STAA  CREG_F
                LDAA  #$04
                STAA  CREG_F
                STAA  CREG_E
                CLRA
                STAA  CREG_D
LOOP_CTIME      LDX  #DC_SEC
                JSR  READ_TIME
                LDX  #DC_WEEK
                JSR  READ_DATE
                JSR  DISY_CTIM
                JSR  DISY_CDATE
                LDAA  #$01
                LDX  #LINE1
                JSR  WR_LINE
                LDAA  #$02
                LDX  #LINE2
                JSR  WR_LINE
                LDAA  #$07
                STAA  KEY_PORT
                LDAA  KEY_PORT
                ANDA  #$F0
                CMPA  #$D0
                BNE  EXIT_CHECK
                JSR  SET_TIME
EXIT_CHECK      LDAA  #$0E
                STAA  KEY_PORT
                LDAA  KEY_PORT
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                ANDA  #$F0
                CMPA  #$D0
                BEQ  EXIT_TIME
                LDAA  SCDR
                CMPA  #$44
                BNE  SCI_CHECK
SCI_CHECK       LDAA  SCDR
                CMPA  #$45
                BEQ  EXIT_TIME
                JSR  DELAYL
                BRA  LOOP_CTIME
EXIT_TIME       LDX  #SYS_FLAG
                BCLR  0,X $01

******** AUTO RANGE SENSORS ADC *******

AUTOS_KEY       LDX  #SYS_FLAG
                BSET  0,X $01
                LDAA  #$01
                LDX  #AUTOS_TAB
                JSR  WR_LINE
                LDAA  #$02
                LDX  #LIMIT_TAB
                JSR  WR_LINE
GET_KEY         JSR  SCAN_KEY
OK_AUTO         CMPA  #$02
                BNE  EXIT_AKEY
                BRA  RUN_AUTO
EXIT_AKEY       CMPA  #$0E
                BNE  GET_KEY
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                JMP   EXIT_AUTO
RUN_AUTO        LDX   #LVOLT_TAB
                LDY   #LINE2
                JSR   M_ASCII
                LDAB   #$40
                STAB   SSR_CTRL
LOOP1_AUTO      JSR   AUTO_RANGE
                LDX   #LINE2
                LDD   $214A
                STD   $05,X
                LDAA   $214C
                STAA   $07,X
                LDAA   #'.'
                STAA   $08,X
                LDAA   $214D
                STAA   $09,X
                LDX   #LINE2
                LDAA   #$02
                JSR   WR_LINE
                LDAA   #$0E
                STAA   KEY_PORT
                LDAA   KEY_PORT
                ANDA   #$F0
                CMPA   #$D0
                BNE   CONTINUE1
                JMP   EXIT_AUTO
                LDAA   SCDR
                CMPA   #$45
                BNE   CONTINUE1
                JMP   EXIT_AUTO
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
CONTINUE1       JSR  DELAYL
                LDAB  #$19
                CMPB  BUFF_ADC
                BCC  LOOP2_AUTO
                BRA  LOOP1_AUTO
LOOP2_AUTO      LDAB  #$20
                STAB  SSR_CTRL
LOOP2           JSR  AUTO_RANGE
                LDX  #LINE2
                LDD  $214A
                STD  $05,X
                LDAA  #'.'
                STAA  $07,X
                LDAA  $214C
                STAA  $08,X
                LDAA  $214D
                STAA  $09,X
                LDX  #LINE2
                LDAA  #$02
                JSR  WR_LINE
                LDAA  #$0E
                STAA  KEY_PORT
                LDAA  KEY_PORT
                ANDA  #$F0
                CMPA  #$D0
                BNE  CONTINUE2
                JMP  EXIT_AUTO
                LDAA  SCDR
                CMPA  #$45
                BNE  CONTINUE2
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                JMP   EXIT_AUTO
CONTINUE2       JSR   DELAYL
                LDAB  #$19
                CMPB  BUFF_ADC
                BCC   LOOP3_AUTO
                LDAB  #$FA
                CMPB  BUFF_ADC
                BCC   LIP2
                JMP   RUN_AUTO
LIP2            BRA   LOOP2
LOOP3_AUTO      LDAB  #$10
                STAB  SSR_CTRL
LOOP3           JSR   AUTO_RANGE
                LDX   #LINE2
                LDAA  $214A
                STAA  $05,X
                LDAA  #'.'
                STAA  $06,X
                LDD   $214B
                STD   $07,X
                LDAA  $214D
                STAA  $09,X
                LDX   #LINE2
                LDAA  #$02
                JSR   WR_LINE
                LDAA  #$0E
                STAA  KEY_PORT
                LDAA  KEY_PORT
                ANDA  #$F0
                CMPA  #$D0
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                BEQ   EXIT_AUTO
                LDAA  SCDR
                CMPA  #$45
                BNE   CONTINUE3
                JMP   EXIT_AUTO
CONTINUE3       JSR   DELAYL
                LDAB  #$FA
                CMPB  BUFF_ADC
                BCC   LIP3
                JMP   LOOP2_AUTO
LIP3            JMP   LOOP3
EXIT_AUTO       LDX   #SYS_FLAG
                BCLR  0,X $01
                RTS

********** DIMMER LIGHT  4 CH *********

DIMMER_KEY      LDX   #SYS_FLAG
                BSET  0,X $01
                LDX   #DIMMER_TAB
                LDY   #LINE1
                JSR   M_ASCII
                LDX   #A_DIMMER1
                STX   BUF6
                LDAA  #$02
                JSR   WR_LINE
                LDX   #H_DIMMER1
                STX   BUF7
                LDX   #CHANNEL1
                STX   BUF5
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
DIM_LOOP        LDX   BUF5
                LDAB   0,X
                ANDB   #$0F
                LDX   #NUM_TAB
                ABX
                LDAA   0,X
                LDX   #LINE1
                STAA   $11,X
                LDAA   #$01
                JSR   WR_LINE
                LDX   BUF6
                LDAA   #$02
                JSR   WR_LINE
                JSR   SCAN_KEY
INC_CH_DIM      CMPA   #$04
                BNE   DEC_CH_DIM
                LDX   BUF5
                CPX   #CHANNEL4
                BEQ   END_INC_DIM
                INX
                LDY   BUF7
                INY
                STY   BUF7
                JSR   INC_BUF6
END_INC_DIM     STX   BUF5
                BRA   END_CH_DIM
DEC_CH_DIM      CMPA   #$05
                BNE   SET_LEV_DIM
                LDX   BUF5
                CPX   #CHANNEL1
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                BEQ   END_DEC_DIM
                DEX
                LDY   BUF7
                DEY
                STY   BUF7
                JSR   DEC_BUF6
END_DEC_DIM     STX   BUF5
                BRA   END_CH_DIM
SET_LEV_DIM     CMPA  #$0D
                BNE   EXIT_DIM
                JSR   SET_LEVEL
                BRA   END_CH_DIM
EXIT_DIM        CMPA  #$0E
                BNE   END_CH_DIM
                BRA   EXIT_DIMMER
END_CH_DIM      JMP   DIM_LOOP
EXIT_DIMMER     LDX   #SYS_FLAG
                BCLR  0,X $01
                RTS

******* PROGRAM ON/OFF  CONTROL *******

PRGRM_KEY       LDX   #SYS_FLAG
                BSET  0,X $01
                LDAA  #$01
                LDX   #PRGRM_TAB
                JSR   WR_LINE
                LDAA  #$02
                LDX   #CH_PRGRM
                LDY   #LINE2
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                JSR  M_ASCII
                LDX  #CHANNEL1
                STX  BUF2
LOOP_SHOWN      JSR  ON_OFF
                LDAA  #$02
                LDX  #LINE2
                JSR  WR_LINE
                JSR  DELAYL
                JSR  SCAN_KEY
INC_SHOWN       CMPA  #$04
                BNE  DEC_SHOWN
                LDX  BUF2
                CPX  #CHANNEL8
                BEQ  END_INCS
                INX
END_INCS        STX  BUF2
                BRA  END_SHOWN
DEC_SHOWN       CMPA  #$05
                BNE  SET_SHOWN
                LDX  BUF2
                CPX  #CHANNEL1
                BEQ  END_DECS
                DEX
END_DECS        STX  BUF2
                BRA  END_SHOWN
SET_SHOWN       CMPA  #$0D
                BNE  END_SHOWN
                BRA  SET_CH_P
END_SHOWN       BRA  LOOP_SHOWN
SET_CH_P        LDX  #CH_PRGRM
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                LDY  #LINE1
                JSR  M_ASCII
                LDX  #TP_SET_TAB
                LDY  #LINE2
                JSR  M_ASCII
                LDX  #CHANNEL1
                STX  BUF2
                LDY  #ON_DAY1
                STY  BUF10
                LDY  #OFF_DAY1
                STY  BUF11
LOOP_PRGM       LDX  BUF10
                STX  BUF6
                JSR  TIME_PRGM
                JSR  ON_A
                LDAA  #$01
                LDX   #LINE1
                JSR  WR_LINE
                LDAA  #$02
                LDX  #LINE2
                JSR  WR_LINE
                JSR  DELAYL
                JSR  SCAN_KEY
INC_MAN_A       CMPA  #$04
                BNE  DEC_MAN_A
                LDX  BUF2
                CPX  #CHANNEL8
                BEQ  END_INC_A
                INX
                LDY  BUF10
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                LDAB  #$05
                ABY
                STY   BUF10
                LDY   BUF11
                LDAB  #$05
                ABY
                STY   BUF11
END_INC_A       STX   BUF2
                BRA   END_MAN_A
DEC_MAN_A       CMPA  #$05
                BNE   SET_PRGM
                LDX   BUF2
                CPX   #CHANNEL1
                BEQ   END_DEC_A
                DEX
                LDY   BUF10
                DEY
                DEY
                DEY
                DEY
                DEY
                STY   BUF10
                LDY   BUF11
                DEY
                DEY
                DEY
                DEY
                DEY
                STY   BUF11
END_DEC_A       STX   BUF2
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                BRA   END_MAN_A
SET_PRGM        CMPA  #$0D
                BNE   EXIT_MAN_A
                JSR   SET_ON_OFF
                BRA   END_MAN_A
EXIT_MAN_A      CMPA  #$0E
                BNE   END_MAN_A
                BRA   EXIT_COMM_A
END_MAN_A       JMP   LOOP_PRGM
EXIT_COMM_A     LDX   #SYS_FLAG
                BCLR  0,X $01
                RTS

******** MANUAL ON/OFF CONTROL ********

MANUAL_KEY      LDX   #SYS_FLAG
                BSET  0,X $01
                LDAA  #$01
                LDX   #MANUAL_TAB
                JSR   WR_LINE
                LDX   #CH_MANUAL
                LDY   #LINE2
                JSR   M_ASCII
                LDX   #CHANNEL1
                STX   BUF2
LOOP_MAN        JSR   ON_OFF
S_CONDIT        LDAA  #$02
                LDX   #LINE2
                JSR   WR_LINE
                JSR   SCAN_KEY
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
INC_MANUL       CMPA  #$04
                BNE   DEC_MANUL
                LDX   BUF2
                CPX   #CHANNEL8
                BEQ   END_INC
                INX
END_INC         STX   BUF2
                BRA   END_MANUL
DEC_MANUL       CMPA  #$05
                BNE   ON_MANUL
                LDX   BUF2
                CPX   #CHANNEL1
                BEQ   END_DEC
                DEX
END_DEC         STX   BUF2
                BRA   END_MANUL
ON_MANUL        CMPA  #$00
                BNE   OFF_MANUL
                LDX   BUF2
                BSET  0,X $80
                LDAB  0,X
                ANDB  #$0F
                LDX   #ON_KEY
                DECB
                ABX
                LDAB  0,X
                ORAB  CHAN_MAN
                STAB  CHAN_MAN
                STAB  OUT_CTRL
                BRA   END_MANUL
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
OFF_MANUL       CMPA  #$01
                BNE   EXIT_MANUL
                LDX   BUF2
                BCLR  0,X $80
                LDAB  0,X
                ANDB  #$0F
                LDX   #OFF_KEY
                DECB
                ABX
                LDAB  0,X
                ANDB  CHAN_MAN
                STAB  CHAN_MAN
                STAB  OUT_CTRL
                BRA   END_MANUL
EXIT_MANUL      CMPA  #$0E
                BNE   END_MANUL
                BRA   EXIT_COMM
END_MANUL       JMP   LOOP_MAN
EXIT_COMM       LDX   #SYS_FLAG
                BCLR  0,X $01
                RTS

** SENSOR CONTROL  SOLID STATE RELAY **

SEN_CTRL        LDX   #SYS_FLAG
                BSET  0,X $01
                LDAA  #$01
                LDX    #ADC_CTL_TAB
                JSR   WR_LINE
                LDAA  #$02
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                LDX   #CH_SSR_TAB
                JSR   WR_LINE
WAIT_KEY        JSR   SCAN_KEY
SET_ADC         CMPA  #$0D
                BNE   WAIT_KEY
                LDX   #CH_AUTOC
                LDY   #LINE1
                JSR   M_ASCII
                LDX   #LCH_AUTOC
                LDY   #LINE2
                JSR   M_ASCII
                LDAB  #$01
                STAB  REGI
                LDX   #ADC1
                STX   BUF3
LOOP_CH_ADC     LDAB  REGI
                LDX   #LINE1
                LDY   #NUM_TAB
                ABY
                LDAB  0,Y
                STAB  $0F,X
                JSR   CONVERT
                LDX   #LINE1
                LDAA  #$01
                JSR   WR_LINE
                LDAA  #$0F
                JSR   CURSOR_L
                JSR   SCAN_KEY
INC_CH_ADC      CMPA  #$04
                BNE   DEC_CH_ADC
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                LDAB  REGI
                CMPB  #$04
                BEQ   END_INC_ADC
                LDX BUF3
                INX
                STX   BUF3
                INCB
END_INC_ADC     STAB  REGI
                BRA   END_CHEK_CH
DEC_CH_ADC      CMPA  #$05
                BNE   SET_LEV_ADC
                LDAB  REGI
                CMPB  #$01
                BEQ   END_DEC_ADC
                LDX   BUF3
                DEX
                STX   BUF3
                DECB
END_DEC_ADC     STAB  REGI
                BRA   END_CHEK_CH
SET_LEV_ADC     CMPA  #$0D
                BNE    OK_ADCC
                JSR   SET_LADC
                BRA   END_CHEK_CH
OK_ADCC         CMPA  #$02
                BNE   EXIT_ADCC
                JSR   RUN_ADCC
                BRA   EXIT_ACTRL
EXIT_ADCC       CMPA  #$0E
                BEQ   EXIT_ACTRL
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
END_CHEK_CH    JMP  LOOP_CH_ADC
EXIT_ACTRL     LDX  #SYS_FLAG
               BCLR  0,X #01
               RTS

****************************************
*    SUBROUTINE   MONITOR  PROGRAM     *
****************************************

****** CONVERT ANALOG TO DIGITAL ******

AUTO_RANGE     LDX  #REGBS
               LDAA #$20
               STAA ADCTL,X
RANGE_FLAG     LDX  #REGBS
               BCLR ADCTL,X $80
               LDX  #BUFF_ADC
               LDY  #$1031
READ_AUTO      LDAA 0,Y
               STAA 0,X
               INY
               INX
               CPY  #$1035
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
*****************************************
*   DISPLAY       $FF = 255 = 5V        *
*                   1 = 0.0196V         *
*          ACT*196/10 = ACTUAL mv       *
*****************************************

            LDX  #BUFF_ADC
            LDAA  0,X
            LDAB  #196
            MUL
            STD  $2147
            LDAB  #$22
            JSR  CALL
            RTS

******** RUN ADC CONTROL  SSR. ********

RUN_ADCC     LDX  #CH_CTL_TAB
             LDAA  #$01
             JSR  WR_LINE
             LDX  #CONDIT_TAB
             LDY  #LINE2
             JSR  M_ASCII
LOOP_RUN     LDX  #REGBS
             LDAA  #$14
             STAA  ADCTL,X
START        BCLR  ADCTL,X $80
             LDX  #BUFF_ADC
             LDY  #$1031
READ_DATA    LDAA  0,Y
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                STAA  0,X
                INX
                INY
                CPY  #$1035
                BNE  READ_DATA
                LDAA  #$00
                STAA  REGI
CHECK1          LDAA  ADC1
                CMPA  BUFF_ADC
                BCS  ON1
                JSR  OFF_SSR
                LDX  #LINE2
                STAA  $02,X
                STAA  $03,X
                BRA  CHECK2
ON1             JSR  ON_SSR
                LDX  #LINE2
                STAA  $02,X
                STAB  $03,X
CHECK2          INC  REGI
                LDAA  ADC2
                CMPA  BUFF_ADC+1
                BCS  ON2
                JSR  OFF_SSR
                LDX  #LINE2
                STAA  $07,X
                STAA  $08,X
                BRA  CHECK3
ON2             JSR  ON_SSR
                LDX  #LINE2
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                STAA  $07,X
                STAB  $08,X
CHECK3          INC   REGI
                LDAA  ADC3
                CMPA  BUFF_ADC+2
                BCS   ON3
                JSR   OFF_SSR
                LDX   #LINE2
                STAA  $0C,X
                STAA  $0D,X
                BRA   CHECK4
ON3             JSR   ON_SSR
                LDX   #LINE2
                STAA  $0C,X
                STAB  $0D,X
CHECK4          INC   REGI
                LDAA  ADC4
                CMPA  BUFF_ADC+3
                BCS   ON4
                JSR   OFF_SSR
                LDX   #LINE2
                STAA  $11,X
                STAA  $12,X
                BRA   END_SSR
ON4             JSR   ON_SSR
                LDX   #LINE2
                STAA  $11,X
                STAB  $12,X
END_SSR         LDX   #LINE2
                LDAA  #$02
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                JSR  WR_LINE
                LDAA  #$0E
                STAA  KEY_PORT
                LDAA  KEY_PORT
                ANDA  #$F0
                CMPA  #$D0
                BEQ  EXIT_SSR
                LDAA  SCDR
                CMPA  #$45
                BEQ  EXIT_SSR
                JSR  DELAYL
                JMP  LOOP_RUN
EXIT_SSR        RTS
ON_SSR          LDAB  REGI
                LDX  #ON_KEY
                ABX
                LDAB  0,X
                ORAB  CTRL_BUF
                STAB  CTRL_BUF
                STAB  SSR_CTRL
                LDAA  #'N'
                LDAB  #' '
                RTS
OFF_SSR         LDAB  REGI
                LDX  #OFF_KEY
                ABX
                LDAB  0,X
                ANDB  CTRL_BUF
                STAB  CTRL_BUF
                STAB  SSR_CTRL
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                LDAA  #'F'
                RTS

******** SET LEVEL ADC COMPARE *********

SET_LADC        JSR   CONVERT
                JSR   SCAN_KEY
INC_LIMIT       CMPA  #$04
                BNE   DEC_LIMIT
                LDX   BUF3
                INC   0,X
                BRA   END_LIMIT
DEC_LIMIT       CMPA  #$05
                BNE   EXIT_LIMIT
                LDX   BUF3
                DEC   0,X
                BRA   END_LIMIT
EXIT_LIMIT      CMPA  #$0E
                BNE   END_LIMIT
                RTS
END_LIMIT       BRA   SET_LADC

******* RESULT  CONVERT TO ASCII ******

CONVERT         LDY   BUF3
                LDAA  0,Y
                LDAB  #196
                MUL
                STD   $2147
                LDAB  #$22
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                JSR  CALL
                LDX  #LINE2
                LDD  $214A
                STD  $0A,X
                LDD  $214C
                STD  $0C,X
                LDAA  #$02
                JSR  WR_LINE
                LDAA  #$4D
                JSR  CURSOR_L
                RTS

****** SET LEVEL DIMMER CONTROL *******

SET_LEVEL       LDX  BUF6
                LDAA  #$02
                JSR  WR_LINE
                LDX  BUF7
                LDAA  0,X
                ADDA  #$40
                JSR  CURSOR_L
                JSR  SCAN_KEY
INC_LEVEL       CMPA  #$04
                BNE  DEC_LEVEL
                LDX  BUF7
                LDAB  0,X
                CMPB  #$13
                BEQ  END_ILEVEL
                INCB
END_ILEVEL      STAB  0,X
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                LDAA  #'>'
                LDX   BUF6
                ABX
                STAA  0,X
                BRA   OK_LEVEL
DEC_LEVEL       CMPA  #$05
                BNE   EXIT_LEVEL
                LDX   BUF7
                LDAB  0,X
                CMPB  #$00
                BEQ   END_DLEVEL
                DECB
END_DLEVEL      STAB  0,X
                LDAA  #' '
                LDX   BUF6
                INCB
                ABX
                STAA  0,X
                BRA   OK_LEVEL
OK_LEVEL        LDX   BUF7
                LDAB  0,X
                LDY   #DIMMER_TABL
                ABY
                LDAA  0,Y
                STAA  REGC
                LDX   BUF5
                LDAB  0,X
                ANDB  #$0F
                LDX   #DIMMER_OUT
                DECB
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                ABX
                LDAA  REGC
                STAA  0,X
END_OK          JMP   SET_LEVEL
EXIT_LEVEL      CMPA  #$0E
                BEQ   END_LEVEL
                BRA   END_OK
END_LEVEL       RTS

**** INC/DEC ASCII BUFFER6 ADDRESS ****

INC_BUF6        LDAB  #20
                LDY   BUF6
LOOP_INC6       INY
                DECB
                BNE   LOOP_INC6
                STY   BUF6
                RTS
DEC_BUF6        LDAB  #20
                LDY   BUF6
LOOP_DEC6       DEY
                DECB
                BNE   LOOP_DEC6
                STY   BUF6
                RTS

****** INTERRUPT SERVICE ROUNTINE *****

INT_IRQ         SEI
                LDX #DC_SEC
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                JSR  READ_TIME
                LDX  #DC_WEEK
                JSR  READ_DATE
                LDX  #CHANNEL8
                STX  BUF9
                LDX  #ON_SEC8
                STX  BUF8
                LDAB  #$08
                STAB  REGL
                LDAA  #$FF
                STAA  ON_OFF_BUF
                JSR  LOOP_COMP
                LDX  #CHANNEL8
                STX  BUF9
                LDX  #OFF_SEC8
                STX  BUF8
                LDAB  #$08
                STAB  REGL
                CLRA
                STAA  ON_OFF_BUF
                JSR  LOOP_COMP
                CLI
                RTI
LOOP_COMP       LDX  BUF8
                LDAA  DC_SEC
                CMPA  0,X
                BEQ  COMP_MIN
                BRA  NEXT
COMP_MIN        DEX
                LDAA  DC_MIN
                CMPA  0,X
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                BEQ   COMP_HOUR
                BRA   NEXT
COMP_HOUR       DEX
                LDAA  DC_HOUR
                CMPA  0,X
                BEQ   COMP_DAY
                BRA   NEXT
COMP_DAY        DEX
                DEX
                LDAA  DC_DAY
                CMPA  0,X
                BEQ   COMP_MONTH
                BRA   NEXT
COMP_MONTH      INX
                LDAA  DC_MONTH
                CMPA  0,X
                BEQ   RUN_ON_OFF
                BRA   NEXT
NEXT            LDY   BUF8
                DEY
                DEY
                DEY
                DEY
                DEY
                STY   BUF8
                LDX   BUF9
                DEX
                STX   BUF9
                DEC   REGL
                BEQ   END_INT
                JMP   LOOP_COMP
END_INT         RTS
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

```
RUN_ON_OFF    LDX   BUF9
              LDAB  0,X
              LDAA  #$00
              CMPA  ON_OFF_BUF
              BEQ   PRGM_OFF
PRGM_ON       BSET  0,X $80
              ANDB  #$0F
              LDX   #ON_KEY
              DECB
              ABX
              LDAB  0,X
              ORAB  CHAN_MAN
              STAB  CHAN_MAN
              STAB  OUT_CTRL
              BRA   NEXT
PRGM_OFF      BCLR  0,X $80
              ANDB  #$0F
              LDX   #OFF_KEY
              DECB
              ABX
              LDAB  0,X
              ANDB  CHAN_MAN
              STAB  CHAN_MAN
              STAB  OUT_CTRL
              BRA   NEXT

********** TIME ON/OFF ALARM **********

SET_ON_OFF     LDX  BUF10
               STX  BUF6
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                JSR  SET_ATIME
                LDX  BUF11
                STX  BUF6
                JSR  TIME_PRGM
                JSR  OFF_A
                LDX  #LINE1
                LDAA  #$01
                JSR  WR_LINE
                LDX  #LINE2
                LDAA  #$02
                JSR  WR_LINE
                LDX  BUF11
                STX  BUF6
                JSR  SET_ATIME
                RTS
SET_ATIME       LDAB  #$3B          ; Sec
                STAB  MAX_BUF
                CLRB
                STAB  MIN_BUF
                JSR  MAX_MIN
                LDAB  #$04
                LDY  BUF6
                ABY
                STY  BUF7
                LDX   BUF7
                LDAA  #$53
                STAA  REGH
                JSR  SET_PTIME
                LDAB  #$03
                LDY  BUF6
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
ABY
STY  BUF7
LDX  BUF7                ; Min
LDAA  #$50
STAA  REGH
JSR  SET_PTIME
LDAB  #$18               ; Hour
STAB  MAX_BUF
LDAB  #$01
STAB  MIN_BUF
JSR  MAX_MIN
LDAB  #$02
LDY  BUF6
ABY
STY  BUF7
LDX  BUF7
INC  0,X
LDAA  #$4D
STAA  REGH
JSR  SET_PTIME
LDAB  #$0C               ; Month
STAB  MAX_BUF
JSR  MAX_MIN
LDAB  #$01
LDY  BUF6
ABY
STY  BUF7
LDX  BUF7
INC  0,X
LDAA  #$4A
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                STAA  REGH
                JSR  SET_PTIME
                LDAB  #$1F              ; Day
                STAB  MAX_BUF
                LDAB  #$01
                STAB  MIN_BUF
                JSR  MAX_MIN
                LDX  BUF6
                INC  0,X
                LDAA  #$47
                STAA  REGH
                JSR  SET_PTIME
                RTS

********** ADJUST PROGRAM TIME **********

SET_PTIME       LDAB  0,X
                LDAA  0,X
                LDY  #ADJ_BUF
LOOP_PFND       CMPA  0,Y
                BEQ  ADJ_PTIM1
                INY
                BRA  LOOP_PFND
ADJ_PTIM1       PSHY
LOOP_PADJ       STAB  0,X
                PSHX
                JSR  TIME_PRGM
                LDAA  #$02
                LDX  #LINE2
                JSR  WR_LINE
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                LDAA  REGH
                JSR   CURSOR_L
                JSR   SCAN_KEY
INC_PTIM        CMPA  #$04
                BNE   DEC_PTIM
                PULX
                PULY
                CPY   BUF4
                BNE   INC_PTIM1
                LDY   BUF5
                BRA   END_PADJ
INC_PTIM1       INY
                LDAB  0,Y
                PSHY
                BRA   LOOP_PADJ
DEC_PTIM        CMPA  #$05
                BNE   EXIT_PADJ
                PULX
                PULY
                CPY   BUF5
                BNE   DEC_PTIM1
                LDY   BUF4
                BRA   END_PADJ
DEC_PTIM1       DEY
                LDAB  0,Y
                PSHY
                BRA   LOOP_PADJ
EXIT_PADJ       CMPA  #$03
                BNE   END_ADJP_P
                PULX
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                PULY
                RTS
END_ADJP_P      PULX
                PULY
END_PADJ        LDAB  0,Y
                PSHY
                BRA  LOOP_PADJ

********* ALARM TIME  PROGRAM *********

TIME_PRGM       LDY  BUF6
                LDAB  0,Y
                LDX  #LINE2              ; Display Day
                LDY  #NUM_TAB
                STAB  REGC
                ANDB  #$F0
                JSR  R_SHIFT
                ABY
                LDAB  $0,Y
                STAB  $06,X
                LDAB  REGC
                ANDB  #$0F
                LDY   #NUM_TAB
                ABY
                LDAB  0,Y
                STAB  $07,X
                LDY  BUF6
                LDAB  $01,Y              ; Display Month
                STAB  REGC
                LDY  #NUM_TAB
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
ANDB  #$F0
JSR  R_SHIFT
ABY
LDAB  0,Y
STAB  $09,X
LDAB  REGC
ANDB  #$0F
LDY   #NUM_TAB
ABY
LDAB  0,Y
STAB  $0A,X
LDY  BUF6
LDAB  $02,Y          ; Display Hour
STAB  REGC
LDY  #NUM_TAB
ANDB  #$F0
JSR  R_SHIFT
ABY
LDAB  0,Y
STAB  $0C,X
LDAB  REGC
ANDB  #$0F
LDY   #NUM_TAB
ABY
LDAB  0,Y
STAB  $0D,X
LDY  BUF6
LDAB  $03,Y              ; Display Min
STAB  REGC
LDY  #NUM_TAB
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
ANDB  #$F0
JSR  R_SHIFT
ABY
LDAB  0,Y
STAB  $0F,X
LDAB  REGC
ANDB  #$0F
LDY   #NUM_TAB
ABY
LDAB  0,Y
STAB  $10,X
LDY  BUF6
LDAB  $04,Y          ; Display Sec
STAB  REGC
LDY  #NUM_TAB
ANDB  #$F0
JSR  R_SHIFT
ABY
LDAB  0,Y
STAB  $12,X
LDAB  REGC
ANDB  #$0F
LDY  #NUM_TAB
ABY
LDAB  0,Y
STAB  $13,X
RTS
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
**** ON/OFF SET  PROGRAM CONDITION ****

ON_A        LDX  BUF2
            LDY  #LINE1
            LDAB  0,X
            STAB  REGC
            LDAA  #'N'
            STAA  $10,Y
            LDAA  #' '
            STAA  $11,Y
            LDAB  REGC
            ANDB  #$0F
            LDX  #NUM_TAB
            ABX
            LDAA  0,X
            LDX  #LINE1
            STAA  $0B,X
            RTS
OFF_A       LDX  BUF2
            LDY  #LINE1
            LDAB  0,X
            STAB  REGC
            LDAA  #'F'
            STAA  $10,Y
            STAA  $11,Y
            LDAB  REGC
            ANDB  #$0F
            LDX  #NUM_TAB
            ABX
            LDAA  0,X
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                LDX  #LINE1
                STAA  $0B,X
                RTS

******** ON/OFF CHECK CONDITION *******

ON_OFF          LDX  BUF2
                LDY  #LINE2
                LDAB  0,X
                STAB  REGC
                BITB  #$80
                BEQ  GET_OFF
                LDAA  #'N'
                STAA  $10,Y
                LDAA  #' '
                STAA  $11,Y
                BRA  GET_NUMBER
GET_OFF         LDAA  #'F'
                STAA  $10,Y
                STAA  $11,Y
GET_NUMBER      LDAB  REGC
                ANDB  #$0F
                LDX  #NUM_TAB
                ABX
                LDAA  0,X
                LDX  #LINE2
                STAA  $0B,X
                RTS
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
********** SET CURRENT TIME ***********

SET_TIME        LDX   #CREG_D
                BSET  0,X $01
                LDX   #CREG_F
                BSET  0,X $02
LOOP_SET        LDAB  #$3B                ; Sec
                STAB  MAX_BUF
                CLRB
                STAB  MIN_BUF
                JSR   MAX_MIN
                LDX   #DC_SEC
                LDAA  #$4E
                STAA  REGH
                JSR   ADJ_TIME
                LDX   #DC_MIN             ; Min
                LDAA  #$4B
                STAA  REGH
                JSR   ADJ_TIME
                LDAB  #$18                ; Hour
                STAB  MAX_BUF
                LDAB  #$01
                STAB  MIN_BUF
                JSR   MAX_MIN
                LDX   #DC_HOUR
                LDAA  #$48
                STAA  REGH
                JSR   ADJ_TIME
                LDAB #$06                 ; Week
                STAB  MAX_BUF
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

```
                STAA  REGH
                JSR  ADJ_TIME
                JSR  SCAN_KEY
CANCEL_TIME     CMPA  #$02
                BEQ  EXIT_SET
                JMP  LOOP_SET
EXIT_SET        JSR  TIM_TO_RTC
                LDX  #CREG_F
                BCLR  0,X $02
                LDX  #CREG_D
                BCLR  0,X $01
                RTS

********* SED SET TIME TO RTC *********

TIM_TO_RTC      LDAB  DC_SEC
                STAB  REGD
                ANDB  #$0F
                STAB  SEC1
                LDAB  REGD
                JSR  R_SHIFT
                ANDB  #$0F
                STAB SEC10
                LDAB  DC_MIN
                STAB  REGD
                ANDB  #$0F
                STAB  MIN1
                LDAB  REGD
                JSR  R_SHIFT
                ANDB  #$0F
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
STAB  MIN10
LDAB  DC_HOUR
STAB  REGD
ANDB  #$0F
STAB  HOUR1
LDAB  REGD
JSR  R_SHIFT
ANDB  #$0F
STAB  HOUR10
LDAB  DC_WEEK
ANDB  #$0F
STAB  WEEK
LDAB  DC_DAY
STAB  REGD
ANDB  #$0F
STAB  DAY1
LDAB  REGD
JSR  R_SHIFT
ANDB  #$0F
STAB  DAY10
LDAB  DC_MONTH
STAB  REGD
ANDB  #$0F
STAB  MONTH1
LDAB  REGD
JSR  R_SHIFT
ANDB  #$0F
STAB  MONTH10
LDAB  DC_YEAR
STAB  REGD
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                ANDB   #$0F
                STAB   YEAR1
                LDAB   REGD
                JSR   R_SHIFT
                ANDB   #$0F
                STAB   YEAR10
                RTS

*** READ TIME FROM  REAL TIME CLOCK ***

READ_TIME       PSHA
                LDAA   SEC1
                ANDA   #$0F
                STAA   REGP
                LDAA   SEC10
                JSR   L_SHIFT
                ANDA   #$F0
                ORAA   REGP
                STAA   0,X
                INX
                LDAA   MIN1
                ANDA   #$0F
                STAA   REGP
                LDAA   MIN10
                JSR   L_SHIFT
                ANDA   #$F0
                ORAA   REGP
                STAA   0,X
                INX
                LDAA   HOUR1
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                ANDA  #$0F
                STAA  REGP
                LDAA  HOUR10
                JSR  L_SHIFT
                ANDA  #$F0
                ORAA  REGP
                STAA  0,X
                PULA
                RTS

*** READ DATE FROM  REAL TIME CLOCK ***

READ_DATE       PSHA
                LDAA  WEEK
                ANDA  #$0F
                STAA  0,X
                INX
                LDAA  DAY1
                ANDA  #$0F
                STAA  REGP
                LDAA  DAY10
                JSR  L_SHIFT
                ANDA  #$F0
                ORAA  REGP
                STAA  0,X
                INX
                LDAA  MONTH1
                ANDA  #$0F
                STAA  REGP
                LDAA  MONTH10
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                JSR  L_SHIFT
                ANDA  #$F0
                ORAA  REGP
                STAA  0,X
                INX
                LDAA  YEAR1
                ANDA  #$0F
                STAA  REGP
                LDAA  YEAR10
                JSR   L_SHIFT
                ANDA  #$F0
                ORAA  REGP
                STAA  0,X
                PULA
                RTS

***** DATE FROM DECIMAL TO ASCII ******

DISY_CDATE      LDX  #LINE1              ; Display Week
                LDY  #WEEK_TAB
                LDAB  DC_WEEK
                ASLB
                ASLB
                ABY
                LDD  0,Y
                STD  $07,X
                INY
                INY
                LDD  0,Y
                STD  $09,X
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        LDAB  DC_DAY              ; Display Day
        STAB  REGC
        LDY   #NUM_TAB
        ANDB  #$F0
        JSR   R_SHIFT
        ABY
        LDAB  0,Y
        STAB  $0B,X
        LDAB  REGC
        ANDB  #$0F
        LDY   #NUM_TAB
        ABY
        LDAB  0,Y
        STAB  $0C,X
        LDAB  DC_MONTH            ; Display Month
        STAB  REGC
        LDY   #NUM_TAB
        ANDB  #$F0
        JSR   R_SHIFT
        ABY
        LDAB  0,Y
        STAB  $0E,X
        LDAB  REGC
        ANDB  #$0F
        LDY   #NUM_TAB
        ABY
        LDAB  0,Y
        STAB  $0F,X
        LDAB  DC_YEAR             ; Display Year
        STAB  REGC
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                LDY  #NUM_TAB
                ANDB  #$F0
                JSR  R_SHIFT
                ABY
                LDAB  0,Y
                STAB  $11,X
                LDAB  REGC
                ANDB  #$0F
                LDY  #NUM_TAB
                ABY
                LDAB  0,Y
                STAB  $12,X
                RTS

***** TIME FROM DECIMAL TO ASCII ******

DISY_CTIM       LDX  #LINE2                ; Display Sec.
                LDY  #NUM_TAB
                LDAB  DC_SEC
                STAB  REGC
                LDY  #NUM_TAB
                ANDB  #$F0
                JSR  R_SHIFT
                ABY
                LDAB  0,Y
                STAB  $0D,X
                LDAB  REGC
                ANDB  #$0F
                LDY   #NUM_TAB
                ABY
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
LDAB  0,Y
STAB  $0E,X
LDAB  DC_MIN              ; Display Min
STAB  REGC
LDY   #NUM_TAB
ANDB  #$F0
JSR   R_SHIFT
ABY
LDAB  0,Y
STAB  $0A,X
LDAB  REGC
ANDB  #$0F
LDY   #NUM_TAB
ABY
LDAB  0,Y
STAB  $0B,X
LDAB  DC_HOUR             ; Display Hour
STAB  REGC
LDY   #NUM_TAB
ANDB  #$F0
JSR   R_SHIFT
ABY
LDAB  0,Y
STAB  $07,X
LDAB  REGC
ANDB  #$0F
LDY   #NUM_TAB
ABY
LDAB  0,Y
STAB  $08,X
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                RTS

********* MAX AND MIN CONTROL *********

MAX_MIN         LDX   #ADJ_BUF
                LDAB   MAX_BUF
                ABX
                STX   BUF4
                LDX   #ADJ_BUF
                LDAB   MIN_BUF
                ABX
                STX   BUF5
                RTS

********* ADJUST CURRENT TIME *********

ADJ_TIME        LDAB   0,X
                LDAA   0,X
                LDY   #ADJ_BUF
LOOP_FIND       CMPA   0,Y
                BEQ   ADJ_TIME1
                INY
                BRA   LOOP_FIND
ADJ_TIME1       PSHY
LOOP_ADJ        STAB   0,X
                PSHX
                JSR   DISY_CTIM
                JSR   DISY_CDATE
                LDAA   #$01
                LDX   #LINE1
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                JSR  WR_LINE
                LDAA  #$02
                LDX  #LINE2
                JSR  WR_LINE
                LDAA  REGH
                JSR  CURSOR_L
                JSR  SCAN_KEY
INC_TIME        CMPA  #$04
                BNE  DEC_TIME
                PULX
                PULY
                CPY  BUF4
                BNE  INC_TIME1
                LDY  BUF5
                BRA  END_ADJ
INC_TIME1       INY
                LDAB  0,Y
                PSHY
                BRA  LOOP_ADJ
DEC_TIME        CMPA  #$05
                BNE  EXIT_ADJ
                PULX
                PULY
                CPY  BUF5
                BNE  DEC_TIME1
                LDY  BUF4
                BRA  END_ADJ
DEC_TIME1       DEY
                LDAB  0,Y
                PSHY
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                BRA   LOOP_ADJ
EXIT_ADJ        CMPA  #$03
                BNE   END_ADJP
                PULX
                PULY
                RTS
END_ADJP        PULX
                PULY
END_ADJ         LDAB  0,Y
                PSHY
                BRA   LOOP_ADJ

****** SHIFT 4 BIT DATA  TO LEFT ******

L_SHIFT         ASLA
                ASLA
                ASLA
                ASLA
                RTS

****** SHIFT 4 BIT DATA  TO RIGHT ******

R_SHIFT         CLC
                RORB
                RORB
                RORB
                RORB
                RTS
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                STAA  SET1              ;  D=1,C=1,B=1
                JSR  DELAY_L
                LDAA  #$38              ; Set Function
                STAA  SET1              ; DL=1,N=1,F=0
                JSR  DELAY_L
                LDAA  #$06              ; Set Mode
                STAA  SET1              ; I/D=1,S=0
                JSR  DELAY_L
                LDAA  #$01              ; Clear Display
                STAA  SET1
                JSR  DELAY_L
                PULA
                RTS

********** CLEAR DISPLAY LCD *********

CLEAR_L         JSR  READY_L
                PSHA
                LDAA  #$01
                STAA  SET1
                PULA
                RTS

********** CURSOR LOCATION ************
*           A = ADDRESS               *

CURSOR_L        JSR  READY_L
                PSHA
                ADDA  #$80
                STAA  SET1
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                PULA
                RTS

********** WRITE BYTE OF LCD **********
*          A = DATA (ASCII)            *

OUT_L           JSR  READY_L
                STAA  WR1LCD
                RTS

********** WRITE  LINE OF LCD *********
*          IX = TABLE OF DATA          *

OUT_LINE        PSHB
                PSHA
                LDAB  #$0A
OUT1_LINE       LDAA  0,X
                JSR  OUT_L
                INX
                DECB
                BNE  OUT1_LINE
                PULA
                PULB
                RTS

************ READ BUSY FLAG ************

READY_L         PSHA
READY1_L        LDAA  SET1
                ANDA  #$80
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                BNE   READY1_L
                PULA
                RTS

************ DELAY   SUB. ************

DELAY_L         PSHA
                LDAA  #$00
DELAY1_L        NOP
                DECA
                BNE   DELAY1_L
                PULA
                RTS

DELAYL          PSHX
                LDX   #$0000
DELAYL1         NOP
                DEX
                BNE   DELAYL1
                PULX
                RTS

********* MOVE ASCII TABLE ***********
*       IX = START ASCII TABLE        *
*       IY = NEW ADDRESS              *

M_ASCII         PSHA
M_ASCII1        LDAA  0,X
                CMPA  #EOT
                BEQ   LM_ASCII
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

```
                STAA  0,Y
                INX
                INY
                BRA  M_ASCII1
LM_ASCII        PULA
                RTS

****************************************
*   LOOK UP TABLE  MONITOR PROGRAM   *
****************************************

KEY_TAB         FCB  $BB,$B7,$BE,$BD     ; Key Code Get
                FCB  $7B,$77,$7E,$7D     ; From Keyboard
                FCB  $EB,$E7,$EE,$ED
                FCB  $DB,$D7,$DE,$DD

NUM_TAB         FCB  $30,$31,$32,$33,$34 ; Ascii of The
                FCB  $35,$36,$37,$38,$39 ; Number

ON_KEY          FCB  $01,$02,$04,$08     ; OR Key System
                FCB  $10,$20,$40,$80     ; On Control

OFF_KEY         FCB  $FE,$FD,$FB,$F7     ; AND Key System
                FCB  $EF,$DF,$BF,$7F     ; Off Control

DIMMER_TABL     FCB  $FF,$F3,$E7,$DB,$CF ; Data Dimmer Level
                FCB  $C3,$B7,$AB,$9F,$93 ; Table
                FCB  $87,$7B,$6F,$63,$57
                FCB  $4B,$3F,$33,$27,$10
                FCB  $00
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
SCI_TAB         FCB  $30,$31,$32,$33,$34 ; Data Ascii Key
                FCB  $35,$36,$37,$38,$39 ; From PC
                FCB  $41,$42,$43,$44,$45
                FCB  $46

WEEK_TAB        FCC  'SUN '
                FCC  'MON '
                FCC  'THE '
                FCC  'WED '
                FCC  'THU '
                FCC  'FRI '
                FCC  'SAT '

****************************************
*      TABLE ASCII DISPLAY ON LCD      *
****************************************

ON_PC_TAB       FCC  '[INC] RUN ON PC     '

ON_BOARD_TAB    FCC  '[DEC] STAND ALONE   '

PRJ1_TAB        FCC  '******** PC ********'

PRJ2_TAB        FCC  'SENSORS & CONTROLLER'

SENSOR_TAB      FCC  '<< DC. VOLTSMETER >>'

CH_SENSOR       FCC  'CH : 0    [ 0000 mV.]'

                FCB  EOT
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
SETCLK_TAB    FCC  '<<  CURRENT TIME  >>'

SETUP_TAB     FCC  '  SETUP & DISPLAY   '

DATE_TAB      FCC  'DATE=>       /  /   '

              FCB  EOT
TIME_TAB      FCC  'TIME=>   :  :       '

              FCB  EOT
AUTOS_TAB     FCC  '<<  AUTO   RANGE  >>'

LIMIT_TAB     FCC  ' LIMIT  0-500 Volts '

LVOLT_TAB     FCC  '          Volts     '

              FCB  EOT
DIMMER_TAB    FCC  'DIMMER CHANNEL[ 0  ]'

              FCB  EOT
PRGRM_TAB     FCC  '<< PROGRAM OPTION >>'

CH_PRGRM      FCC  'CHANNEL : 0  [ 0   ]'

              FCB  EOT
TP_SET_TAB    FCC  'TIME>   /  /  :  :  '

              FCB  EOT
MANUAL_TAB    FCC  '<< MANUAL  OPTION >>'
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
CH_MANUAL     FCC  'CHANNEL : 0  [ 0   ]'

              FCB  EOT
READY_TAB     FCC  '*** SYSTEM READY ***'

OPTION_TAB    FCC  '** SELECT  OPTION **'

ADC_CTL_TAB   FCC  '<< SENSOR CONTROL >>'

CH_SSR_TAB    FCC  'CHANNEL1 - CHANNEL4 '

CH_CTL_TAB    FCC  '[CH1][CH2][CH3][CH4]'

              FCB  EOT
CONDIT_TAB    FCC  ' O    O    O    O   '

              FCB  EOT
CH_AUTOC      FCC  '    CHANNEL : 0     '

              FCB  EOT
LCH_AUTOC     FCC  '  LEVEL : 0000 mV.  '

              FCB  EOT

              ORG  $DFFF
              FCB  $A3
              END
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก จ

โปรแกรมการทำงานบนไมโครซอฟท์วินด์โดวส์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
VERSION 2.00
Begin Form main
   BackColor       =   &H00C0C0C0&
   BorderStyle     =   3  'Fixed Double
   Caption         =   "PC SENSORS AND CONTROLLER"
   ClientHeight    =   6270
   ClientLeft      =   2895
   ClientTop       =   735
   ClientWidth     =   7680
   FontBold        =   -1  'True
   FontItalic      =   0   'False
   FontName        =   "MS Sans Serif"
   FontSize        =   7.5
   FontStrikethru  =   0   'False
   FontUnderline   =   0   'False
   Height          =   6960
   Icon            =   PCSC.FRX:0000
   Left            =   2835
   LinkTopic       =   "Form1"
   MaxButton       =   0   'False
   ScaleHeight     =   6270
   ScaleWidth      =   7680
   Top             =   105
   Width           =   7800
   Begin SSPanel Panel3D3
      BackColor       =   &H00C0C0C0&
      BevelInner      =   1  'Inset
      BorderWidth     =   5
      Caption         =   "Education Engineer  KMIT'Ladkrabang."
      Font3D          =   2  'Raised w/heavy shading
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    FontBold        =   -1  'True
    FontItalic      =   0   'False
    FontName        =   "Book Antiqua"
    FontSize        =   13.5
    FontStrikethru  =   0   'False
    FontUnderline   =   0   'False
    ForeColor       =   &H00000000&
    Height          =   615
    Left            =   120
    TabIndex        =   20
    Top             =   120
    Width           =   7455
End
Begin SSPanel Panel3D2
    BackColor       =   &H00C0C0C0&
    BevelInner      =   2   'Raised
    BevelOuter      =   1   'Inset
    BorderWidth     =   1
    Font3D          =   0   'None
    Height          =   1575
    Left            =   5040
    TabIndex        =   18
    Top             =   840
    Width           =   2535
    Begin Label noled
        BackStyle       =   0   'Transparent
        Caption         =   "12"
        Height          =   255
        Index           =   11
        Left            =   2040
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
   TabIndex        =   22
   Top             =   960
   Width           =   255
End
Begin Label noled
   BackStyle       =   0  'Transparent
   Caption         =   "11"
   Height          =   255
   Index           =   10
   Left            =   1680
   TabIndex        =   23
   Top             =   960
   Width           =   255
End
Begin Label noled
   BackStyle       =   0  'Transparent
   Caption         =   "10"
   Height          =   255
   Index           =   9
   Left            =   1320
   TabIndex        =   35
   Top             =   960
   Width           =   255
End
Begin Label noled
   BackStyle       =   0  'Transparent
   Caption         =   " 9"
   Height          =   255
   Index           =   8
   Left            =   960
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    TabIndex        =   34
    Top             =   960
    Width           =   255
End
Begin Label noled
    BackStyle       =   0  'Transparent
    Caption         =   " 8"
    Height          =   255
    Index           =   7
    Left            =   600
    TabIndex        =   33
    Top             =   960
    Width           =   255
End
Begin Label noled
    BackStyle       =   0  'Transparent
    Caption         =   " 7"
    Height          =   255
    Index           =   6
    Left            =   240
    TabIndex        =   32
    Top             =   960
    Width           =   255
End
Begin Label noled
    BackStyle       =   0  'Transparent
    Caption         =   " 6"
    Height          =   255
    Index           =   5
    Left            =   2040
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    TabIndex        =   31
    Top             =   360
    Width           =   255
End
Begin Label noled
    BackStyle       =   0  'Transparent
    Caption         =   " 5"
    Height          =   255
    Index           =   4
    Left            =   1680
    TabIndex        =   30
    Top             =   360
    Width           =   255
End
Begin Label noled
    BackStyle       =   0  'Transparent
    Caption         =   " 4"
    Height          =   255
    Index           =   3
    Left            =   1320
    TabIndex        =   29
    Top             =   360
    Width           =   255
End
Begin Label noled
    BackStyle       =   0  'Transparent
    Caption         =   " 3"
    Height          =   255
    Index           =   2
    Left            =   960
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    TabIndex        =   28
    Top             =   360
    Width           =   255
End
Begin Label noled
    BackStyle       =   0   'Transparent
    Caption         =   " 2"
    Height          =   255
    Index           =   1
    Left            =   600
    TabIndex        =   27
    Top             =   360
    Width           =   255
End
Begin Label noled
    BackStyle       =   0   'Transparent
    Caption         =   " 1"
    Height          =   255
    Index           =   0
    Left            =   240
    TabIndex        =   26
    Top             =   360
    Width           =   255
End
Begin Label Label3
    BackStyle       =   0   'Transparent
    Caption         =   "Solid state relay display"
    FontBold        =   -1  'True
    FontItalic      =   0   'False
    FontName        =   "MS Sans Serif"
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    FontSize        =   7.5
    FontStrikethru  =   0   'False
    FontUnderline   =   0   'False
    Height          =   375
    Left            =   240
    TabIndex        =   21
    Top             =   120
    Width           =   2535
End
Begin Shape led
    BackColor       =   &H00000000&
    BackStyle       =   1   'Opaque
    BorderColor     =   &H00000000&
    BorderStyle     =   0   'Transparent
    BorderWidth     =   3
    FillStyle       =   0   'Solid
    Height          =   255
    Index           =   11
    Left            =   2040
    Shape           =   3   'Circle
    Top             =   1200
    Width           =   255
End
Begin Shape led
    BackColor       =   &H00000000&
    BackStyle       =   1   'Opaque
    BorderStyle     =   0   'Transparent
    BorderWidth     =   3
    FillStyle       =   0   'Solid
    Height          =   255
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    Index          =   10
    Left           =   1680
    Shape          =   3  'Circle
    Top            =   1200
    Width          =   255
End
Begin Shape led
    BackColor      =   &H00000000&
    BackStyle      =   1  'Opaque
    BorderColor    =   &H00000000&
    BorderStyle    =   0  'Transparent
    BorderWidth    =   3
    FillStyle      =   0  'Solid
    Height         =   255
    Index          =   9
    Left           =   1320
    Shape          =   3  'Circle
    Top            =   1200
    Width          =   255
End
Begin Shape led
    BackColor      =   &H00000000&
    BackStyle      =   1  'Opaque
    BorderStyle    =   0  'Transparent
    BorderWidth    =   3
    FillStyle      =   0  'Solid
    Height         =   255
    Index          =   8
    Left           =   960
    Shape          =   3  'Circle
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    Top             =   1200
    Width           =   255
End
Begin Shape led
    BackColor       =   &H00000000&
    BackStyle       =   1  'Opaque
    BorderColor     =   &H00000000&
    BorderStyle     =   0  'Transparent
    BorderWidth     =   3
    FillStyle       =   0  'Solid
    Height          =   255
    Index           =   7
    Left            =   600
    Shape           =   3  'Circle
    Top             =   1200
    Width           =   255
End
Begin Shape led
    BackColor       =   &H00000000&
    BackStyle       =   1  'Opaque
    BorderColor     =   &H00000000&
    BorderStyle     =   0  'Transparent
    BorderWidth     =   3
    FillStyle       =   0  'Solid
    Height          =   255
    Index           =   6
    Left            =   240
    Shape           =   3  'Circle
    Top             =   1200
    Width           =   255
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
   End
   Begin Shape led
      BackColor       =   &H00000000&
      BackStyle       =   1  'Opaque
      BorderColor     =   &H00000000&
      BorderStyle     =   0  'Transparent
      BorderWidth     =   3
      FillStyle       =   0  'Solid
      Height          =   255
      Index           =   5
      Left            =   2040
      Shape           =   3  'Circle
      Top             =   600
      Width           =   255
   End
   Begin Shape led
      BackColor       =   &H00000000&
      BackStyle       =   1  'Opaque
      BorderColor     =   &H00000000&
      BorderStyle     =   0  'Transparent
      BorderWidth     =   3
      FillStyle       =   0  'Solid
      Height          =   255
      Index           =   4
      Left            =   1680
      Shape           =   3  'Circle
      Top             =   600
      Width           =   255
   End
   Begin Shape led
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    BackColor       =   &H00000000&
    BackStyle       =   1  'Opaque
    BorderColor     =   &H00000000&
    BorderStyle     =   0  'Transparent
    BorderWidth     =   3
    FillStyle       =   0  'Solid
    Height          =   255
    Index           =   3
    Left            =   1320
    Shape           =   3  'Circle
    Top             =   600
    Width           =   255
End
Begin Shape led
    BackColor       =   &H00000000&
    BackStyle       =   1  'Opaque
    BorderColor     =   &H00000000&
    BorderStyle     =   0  'Transparent
    BorderWidth     =   3
    FillStyle       =   0  'Solid
    Height          =   255
    Index           =   2
    Left            =   960
    Shape           =   3  'Circle
    Top             =   600
    Width           =   255
End
Begin Shape led
    BackColor       =   &H00000000&
    BackStyle       =   1  'Opaque
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

```
      BorderStyle     =   0  'Transparent
      BorderWidth     =   3
      FillStyle       =   0  'Solid
      Height          =   255
      Index           =   1
      Left            =   600
      Shape           =   3  'Circle
      Top             =   600
      Width           =   255
   End
   Begin Shape led
      BackColor       =   &H00000000&
      BackStyle       =   1  'Opaque
      BorderColor     =   &H00000000&
      BorderStyle     =   0  'Transparent
      BorderWidth     =   3
      FillStyle       =   0  'Solid
      Height          =   255
      Index           =   0
      Left            =   240
      Shape           =   3  'Circle
      Top             =   600
      Width           =   255
   End
End
Begin SSPanel Panel3D1
   BackColor       =   &H00C0C0C0&
   BevelInner      =   1  'Inset
   BorderWidth     =   10
   Caption         =   "Panel3D1"
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
   Font3D          =   0  'None
   Height          =   1575
   Left            =   120
   TabIndex        =   19
   Top             =   840
   Width           =   4815
   Begin Label line2
      BackColor       =   &H00FF0000&
      FontBold        =   -1  'True
      FontItalic      =   0   'False
      FontName        =   "MS Sans Serif"
      FontSize        =   13.5
      FontStrikethru  =   0   'False
      FontUnderline   =   0   'False
      ForeColor       =   &H00E0E0E0&
      Height          =   495
      Left            =   240
      TabIndex        =   25
      Top             =   840
      Width           =   4335
   End
   Begin Label line1
      Alignment       =   2  'Center
      BackColor       =   &H00FF0000&
      FontBold        =   -1  'True
      FontItalic      =   0   'False
      FontName        =   "MS Sans Serif"
      FontSize        =   13.5
      FontStrikethru  =   0   'False
      FontUnderline   =   0   'False
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
      ForeColor       =   &H00E0E0E0&
      Height          =   615
      Left            =   240
      TabIndex        =   24
      Top             =   240
      Width           =   4335
   End
End
Begin SSFrame Frame3D1
   Font3D          =   0  'None
   Height          =   3615
   Left            =   120
   TabIndex        =   16
   Top             =   2520
   Width           =   7455
   Begin Timer Timer2
      Enabled         =   0   'False
      Interval        =   250
      Left            =   6600
      Top             =   1560
   End
   Begin SSCommand clk
      Caption         =   "CLK"
      Font3D          =   2  'Raised w/heavy shading
      FontBold        =   -1  'True
      FontItalic      =   0   'False
      FontName        =   "MS Sans Serif"
      FontSize        =   9.75
      FontStrikethru  =   0   'False
      FontUnderline   =   0   'False
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    Height          =   615
    Left            =   3960
    TabIndex        =   10
    Top             =   1920
    Width           =   1095
End
Begin Timer Timer1
    Enabled         =   0   'False
    Interval        =   1000
    Left            =   6120
    Top             =   1560
End
Begin SSCommand res
    Caption         =   "RES"
    Font3D          =   2   'Raised w/heavy shading
    FontBold        =   -1  'True
    FontItalic      =   0   'False
    FontName        =   "MS Sans Serif"
    FontSize        =   9.75
    FontStrikethru  =   0   'False
    FontUnderline   =   0   'False
    ForeColor       =   &H00000004&
    Height          =   615
    Left            =   3960
    TabIndex        =   3
    Top             =   480
    Width           =   1095
End
Begin SSCommand exit
    Caption         =   "EXIT"
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
      Font3D          =   2  'Raised w/heavy shading
      FontBold        =   -1  'True
      FontItalic      =   0   'False
      FontName        =   "MS Sans Serif"
      FontSize        =   9.75
      FontStrikethru  =   0   'False
      FontUnderline   =   0   'False
      Height          =   615
      Left            =   2760
      TabIndex        =   2
      Top             =   480
      Width           =   1095
   End
   Begin SSCommand set
      Caption         =   "SET"
      Font3D          =   2  'Raised w/heavy shading
      FontBold        =   -1  'True
      FontItalic      =   0   'False
      FontName        =   "MS Sans Serif"
      FontSize        =   9.75
      FontStrikethru  =   0   'False
      FontUnderline   =   0   'False
      Height          =   615
      Left            =   1560
      TabIndex        =   1
      Top             =   480
      Width           =   1095
   End
   Begin SSCommand sct
      Caption         =   "SCT"
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    Font3D          =   2  'Raised w/heavy shading
    FontBold        =   -1  'True
    FontItalic      =   0   'False
    FontName        =   "MS Sans Serif"
    FontSize        =   9.75
    FontStrikethru  =   0   'False
    FontUnderline   =   0   'False
    Height          =   615
    Left            =   360
    TabIndex        =   0
    Top             =   480
    Width           =   1095
End
Begin SSCommand ats
    Caption         =   "ATS"
    Font3D          =   2  'Raised w/heavy shading
    FontBold        =   -1  'True
    FontItalic      =   0   'False
    FontName        =   "MS Sans Serif"
    FontSize        =   9.75
    FontStrikethru  =   0   'False
    FontUnderline   =   0   'False
    Height          =   615
    Left            =   360
    TabIndex        =   4
    Top             =   1200
    Width           =   1095
End
Begin SSCommand dmc
    Caption         =   "DMC"
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    Font3D          =   2  'Raised w/heavy shading
    FontBold        =   -1  'True
    FontItalic      =   0   'False
    FontName        =   "MS Sans Serif"
    FontSize        =   9.75
    FontStrikethru  =   0   'False
    FontUnderline   =   0   'False
    Height          =   615
    Left            =   1560
    TabIndex        =   5
    Top             =   1200
    Width           =   1095
End
Begin SSCommand prg
    Caption         =   "PRG"
    Font3D          =   2  'Raised w/heavy shading
    FontBold        =   -1  'True
    FontItalic      =   0   'False
    FontName        =   "MS Sans Serif"
    FontSize        =   9.75
    FontStrikethru  =   0   'False
    FontUnderline   =   0   'False
    Height          =   615
    Left            =   2760
    TabIndex        =   6
    Top             =   1200
    Width           =   1095
End
Begin SSCommand manl
    Caption         =   "MANL"
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    Font3D          =   2  'Raised w/heavy shading
    FontBold        =   -1  'True
    FontItalic      =   0   'False
    FontName        =   "MS Sans Serif"
    FontSize        =   9.75
    FontStrikethru  =   0   'False
    FontUnderline   =   0   'False
    Height          =   615
    Left            =   3960
    TabIndex        =   7
    Top             =   1200
    Width           =   1095
End
Begin SSCommand inc
    Caption         =   "INC"
    Font3D          =   2  'Raised w/heavy shading
    FontBold        =   -1  'True
    FontItalic      =   0   'False
    FontName        =   "MS Sans Serif"
    FontSize        =   9.75
    FontStrikethru  =   0   'False
    FontUnderline   =   0   'False
    ForeColor       =   &H00FF0000&
    Height          =   615
    Left            =   360
    TabIndex        =   8
    Top             =   1920
    Width           =   1095
End
Begin SSCommand dec
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
      Caption         =   "DEC"
      Font3D          =   2  'Raised w/heavy shading
      FontBold        =   -1  'True
      FontItalic      =   0   'False
      FontName        =   "MS Sans Serif"
      FontSize        =   9.75
      FontStrikethru  =   0   'False
      FontUnderline   =   0   'False
      ForeColor       =   &H00FF0000&
      Height          =   615
      Left            =   1560
      TabIndex        =   17
      Top             =   1920
      Width           =   1095
   End
   Begin SSCommand snr
      Caption         =   "SNR"
      Font3D          =   2  'Raised w/heavy shading
      FontBold        =   -1  'True
      FontItalic      =   0   'False
      FontName        =   "MS Sans Serif"
      FontSize        =   9.75
      FontStrikethru  =   0   'False
      FontUnderline   =   0   'False
      Height          =   615
      Left            =   2760
      TabIndex        =   9
      Top             =   1920
      Width           =   1095
   End
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
Begin SSCommand swon
    Caption         =   "ON"
    Enabled         =   0   'False
    Font3D          =   2   'Raised w/heavy shading
    FontBold        =   -1  'True
    FontItalic      =   0   'False
    FontName        =   "MS Sans Serif"
    FontSize        =   9.75
    FontStrikethru  =   0   'False
    FontUnderline   =   0   'False
    ForeColor       =   &H00404080&
    Height          =   615
    Left            =   360
    TabIndex        =   11
    Top             =   2640
    Width           =   1095
End
Begin SSCommand off
    Caption         =   "OFF"
    Enabled         =   0   'False
    Font3D          =   2   'Raised w/heavy shading
    FontBold        =   -1  'True
    FontItalic      =   0   'False
    FontName        =   "MS Sans Serif"
    FontSize        =   9.75
    FontStrikethru  =   0   'False
    FontUnderline   =   0   'False
    ForeColor       =   &H00404080&
    Height          =   615
    Left            =   1560
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    TabIndex        =   12
    Top             =   2640
    Width           =   1095
End
Begin SSCommand ok
    Caption         =   "OK"
    Font3D          =   2  'Raised w/heavy shading
    FontBold        =   -1  'True
    FontItalic      =   0   'False
    FontName        =   "MS Sans Serif"
    FontSize        =   9.75
    FontStrikethru  =   0   'False
    FontUnderline   =   0   'False
    Height          =   615
    Left            =   2760
    TabIndex        =   13
    Top             =   2640
    Width           =   1095
End
Begin SSCommand nex
    Caption         =   "NEXT"
    Font3D          =   2  'Raised w/heavy shading
    FontBold        =   -1  'True
    FontItalic      =   0   'False
    FontName        =   "MS Sans Serif"
    FontSize        =   9.75
    FontStrikethru  =   0   'False
    FontUnderline   =   0   'False
    Height          =   615
    Left            =   3960
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
      TabIndex        =   14
      Top             =   2640
      Width           =   1095
   End
   Begin SSCommand quit
      Caption         =   "QUIT"
      Font3D          =   2  'Raised w/heavy shading
      FontBold        =   -1  'True
      FontItalic      =   0   'False
      FontName        =   "MS Sans Serif"
      FontSize        =   9.75
      FontStrikethru  =   0   'False
      FontUnderline   =   0   'False
      ForeColor       =   &H000000FF&
      Height          =   615
      Left            =   5520
      TabIndex        =   15
      Top             =   2640
      Width           =   1215
   End
   Begin MSComm Comm1
      CommPort        =   2
      DTREnable       =   0   'False
      Interval        =   1000
      Left            =   5640
      Top             =   1560
   End
   Begin Image Image5
      Height          =   585
      Left            =   5760
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    Picture         =   PCSC.FRX:0302
    Top             =   840
    Width           =   660
End
Begin Image Image4
    Height          =   435
    Left            =   6480
    Picture         =   PCSC.FRX:0724
    Top             =   840
    Width           =   570
End
Begin Image Image3
    Height          =   585
    Left            =   5760
    Picture         =   PCSC.FRX:09E2
    Top             =   840
    Width           =   660
End
Begin Image Image2
    Height          =   585
    Left            =   5760
    Picture         =   PCSC.FRX:0E04
    Top             =   840
    Width           =   660
End
Begin Shape Shape3
    BorderColor     =   &H000000FF&
    BorderWidth     =   3
    Height          =   615
    Left            =   5520
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
            Top             =   2640
            Width           =   1215
         End
         Begin Shape Shape2
            Height          =   615
            Left            =   5520
            Top             =   2640
            Width           =   1215
         End
         Begin Shape Shape1
            BackColor       =   &H00FFFFFF&
            BorderColor     =   &H000000FF&
            BorderWidth     =   2
            Height          =   3255
            Left            =   120
            Top             =   240
            Width           =   7215
         End
         Begin Image Image1
            Height          =   480
            Left            =   5280
            Picture         =   PCSC.FRX:1226
            Top             =   840
            Width           =   480
         End
      End
      Begin Menu mnuTime
      End
End
Dim Comcon As Integer
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
Dim IDX As Integer
Dim IDXX As Integer
Dim CHN$
Dim OFFLABEL$
Dim CHNIDX As Integer
Dim CHN2$
Dim DIMM$
Dim I As Integer
Dim CONIDX As Integer
Dim CONIDX2 As Integer

Sub ats_Click ()
     Beep
    main.Comm1.Output = Chr$(56)
End Sub
Sub  ats_MouseDown  (Button  As  Integer,  Shift As Integer, X As
                    Single, Y As Single)
     image5.Picture = image3.Picture
End Sub
Sub  ats_MouseUp  (Button  As  Integer,  Shift  As  Integer, X As
                   Single, Y As Single)
    image5.Picture = image2.Picture
End Sub
Sub clk_Click ()
     Beep
    main.Comm1.Output = Chr$(55)
End Sub
Sub  clk_MouseDown  (Button  As  Integer,  Shift As Integer, X As
                    Single, Y As Single)
     image5.Picture = image3.Picture
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
End Sub

Sub clk_MouseUp (Button As Integer, Shift As Integer, X As
                 Single, Y As Single)
    image5.Picture = image2.Picture
End Sub

Sub dec_Click ()
    Beep
   main.Comm1.Output = Chr$(53)
   Select Case CONIDX
     Case 1
  If IDX > 0 Then
    IDX = IDX - 1
    noled(IDX).ForeColor = &HFF&
    noled(IDX + 1).ForeColor = &H0&
  Else
       IDX = 7
       noled(IDX).ForeColor = &HFF&
       noled(0).ForeColor = &H0&
  End If
     CHNIDX = IDX + 1
     CHN$ = "0" & Str(CHNIDX)
     DISPLAY
   Case 2
    Select Case CONIDX2
     Case 3
    I = I - 1
       If I < 1 Then
I = 1
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        End If
      Case 2
      If IDXX > 0 Then
        IDXX = IDXX - 1
      Else
        IDXX = 3
      End If
        CHNIDX2 = IDXX + 1
        CHN2$ = "0" & Str(CHNIDX2)
        LINE1 = "DIMMER CHANNEL  [ " & CHN2$ & " ]"
    End Select
  End Select
End Sub

Sub dec_MouseDown (Button As Integer, Shift As Integer, X As
                   Single, Y As Single)
    image5.Picture = image3.Picture
End Sub
Sub dec_MouseUp (Button As Integer, Shift As Integer, X As
                   Single, Y As Single)
image5.Picture = image2.Picture
End Sub

Sub DISPLAY ()
    LINE1 = "<<  MANUAL  OPTION  >>"
     LN$ = CHN$ & " [ " & OFFLABEL$ & " ]"
    LINE2 = "  CHANNEL : " & LN$
End Sub

Sub dmc_Click ()
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    Beep
    main.Comm1.Output = Chr$(57)
  If CONIDX2 = 1 Then
    CHN2$ = "01"
    LINE1 = "DIMMER CHANNEL  [ " & CHN2$ & " ]"
    LINE2 = ""
    CONIDX = 2
    CONIDX2 = 2
    swon.Enabled = False
    OFF.Enabled = False

  Else
  End If
End Sub

Sub  dmc_MouseDown  (Button  As  Integer,  Shift As Integer, X As
                    Single, Y As Single)
     image5.Picture = image3.Picture
End Sub

Sub  dmc_MouseUp  (Button  As  Integer,  Shift  As  Integer, X As
                   Single, Y As Single)
     image5.Picture = image2.Picture
End Sub

Sub exit_Click ()
    Beep
   main.Comm1.Output = Chr$(69)
      If CONIDX2 = 3 Then
 LINE2 = ""
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
 TIMER2.Enabled = False
      End If
    CONIDX2 = CONIDX2 - 1
      If CONIDX2 < 1 Then
        CONIDX2 = 1
      End If
      ' If CONIDX2 = 0 Then
      '    LINE2 = ""
      ' End If

    If CONIDX2 = 1 Then
     LINE1 = "***   SYSTEM  READY  ***"
      LINE2 = "      **   SELECT OPTION  ** "
    TIMER2.Enabled = False
    Else
    End If
End Sub

Sub  exit_MouseDown  (Button  As  Integer, Shift As Integer, x As
                      Single, Y As Single)
     image5.Picture = image3.Picture
End Sub

Sub  exit_MouseUp  (Button  As  Integer,  Shift  As Integer, X As
                    Single, Y As Single)
    image5.Picture = image2.Picture
End Sub

Sub Form_Load ()
     ' Use COM1.
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    Comm1.CommPort = 2
    ' 9600 baud, no parity, 8 data, and 1 stop bit.
    Comm1.Settings = "9600,N,8,1"
    ' Tell the control to read entire buffer when Input is used.
    Comm1.InputLen = 0
    ' Open the port.
    Comm1.PortOpen = True
    CHN$ = "0" & Str(1)
    OFFLABEL$ = "OFF"
    LINE1 = "***   SYSTEM  READY  ***"
    LINE2 = "      **   SELECT OPTION  ** "
    CHN2$ = "0" & Str(1)
    I = 1
    DIMM$ = String$(I, ">")
    CONIDX2 = 2
    CONIDX2 = 1
End Sub

Sub inc_Click ()
   Beep
  main.Comm1.Output = Chr$(52)
  Select Case CONIDX
   Case 1
  If IDX < 7 Then
    IDX = IDX + 1
    noled(IDX).ForeColor = &HFF&
    noled(IDX - 1).ForeColor = &H0&
  Else
       IDX = 0
       noled(IDX).ForeColor = &HFF&
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        noled(7).ForeColor = &H0&
    End If
      CHNIDX = IDX + 1
      CHN$ = "0" & Str(CHNIDX)
     DISPLAY
    Case 2
      Select Case CONIDX2
       Case 3
       I = I + 1
         If I > 23 Then
I = 23
         End If
       Case 2
        If IDXX < 3 Then
          IDXX = IDXX + 1
        Else
          IDXX = 0
        End If
          CHNIDX2 = IDXX + 1
          CHN2$ = "0" & Str(CHNIDX2)
          LINE1 = "DIMMER CHANNEL  [ " & CHN2$ & " ]"
      End Select
  End Select
End Sub

Sub  inc_MouseDown  (Button  As  Integer,  Shift As Integer, X As
                     Single, Y As Single)
      image5.Picture = image3.Picture

End Sub
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
Sub inc_MouseUp (Button As Integer, Shift As Integer, X As
                 Single, Y As Single)
   image5.Picture = image2.Picture
End Sub

Sub manl_Click ()
   Beep
   main.Comm1.Output = Chr$(66)
  If CONIDX2 = 1 Then
    IDX = 0
    noled(0).ForeColor = &HFF&
    noled(1).ForeColor = &H0&
    noled(2).ForeColor = &H0&
    noled(3).ForeColor = &H0&
    noled(4).ForeColor = &H0&
    noled(5).ForeColor = &H0&
    noled(6).ForeColor = &H0&
    noled(7).ForeColor = &H0&
   inc.Enabled = True
   dec.Enabled = True
   ok.Enabled = True
   nex.Enabled = True
   CONIDX2 = 2
   CONIDX = 1
   DISPLAY
  Else
  End If
   swon.Enabled = True
   OFF.Enabled = True
End Sub
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
Sub man1_MouseDown (Button As Integer, Shift As Integer, X As
                    Single, Y As Single)
    image5.Picture = image3.Picture
End Sub

Sub man1_MouseUp (Button As Integer, Shift As Integer, X As
                  Single, Y As Single)
    image5.Picture = image2.Picture
End Sub

Sub mnucom1_Click ()
    Comcon = 1
End Sub

Sub mnucom2_Click ()
    Comcon = 2
End Sub

Sub nex_Click ()
    Beep
    main.Comm1.Output = Chr$(51)
End Sub

Sub nex_MouseDown (Button As Integer, Shift As Integer, X As
                   Single, Y As Single)
    image5.Picture = image3.Picture
End Sub

Sub nex_MouseUp (Button As Integer, Shift As Integer, X As
                 Single, Y As Single)
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    image5.Picture = image2.Picture
End Sub

Sub off_Click ()
        Beep
       main.Comm1.Output = Chr$(49)
       led(IDX).FillColor = &H0&
       OFFLABEL$ = "OFF"
       DISPLAY
End Sub

Sub  off_MouseDown  (Button  As  Integer,  Shift As Integer, X As
                    Single, Y As Single)
     image5.Picture = image3.Picture
End Sub

Sub  off_MouseUp  (Button  As  Integer,Shift  As  Integer,  X  As
                    Single, Y As Single)
image5.Picture = image2.Picture
End Sub

Sub ok_Click ()
      Beep
     main.Comm1.Output = Chr$(50)
End Sub

Sub  ok_MouseDown  (Button  As  Integer,  Shift  As Integer, X As
                     Single, Y As Single)
      image5.Picture = image3.Picture
End Sub
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
Sub ok_MouseUp (Button As Integer, Shift As Integer, X As  Single,
                Y As Single)
     image5.Picture = image2.Picture
End Sub

Sub prg_Click ()
      Beep
     main.Comm1.Output = Chr$(65)
End Sub
Sub  prg_MouseDown  (Button  As  Integer,  Shift As Integer, X As
                     Single, Y As Single)
     image5.Picture = image3.Picture
End Sub
Sub  prg_MouseUp  (Button  As  Integer,  Shift  As  Integer, X As
                   Single, Y As Single)
     image5.Picture = image2.Picture
End Sub

Sub quit_Click ()
   ' Close the serial port.
    Comm1.PortOpen = False
    Unload main
End Sub

Sub res_Click ()
 Beep
   main.Comm1.Output = Chr$(70)
End Sub
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
Sub res_MouseDown (Button As Integer, Shift As Integer, X As
                   Single, Y As Single)
    image5.Picture = image3.Picture
End Sub

Sub res_MouseUp (Button As Integer, Shift As Integer, X As
                 Single, Y As Single)
    image5.Picture = image2.Picture
End Sub

Sub sct_Click ()
    Beep
    main.Comm1.Output = Chr$(67)
End Sub
Sub sct_MouseDown (Button As Integer, Shift As Integer, X As
                   Single, Y As Single)
    image5.Picture = image3.Picture
End Sub

Sub sct_MouseUp (Button As Integer, Shift As Integer, X As
                 Single, Y As Single)
    image5.Picture = image2.Picture
End Sub

Sub set_Click ()
    Beep
    main.Comm1.Output = Chr$(68)
  If CONIDX = 2 Then
  If CONIDX2 = 2 Then
     I = 1
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
      TIMER2.Enabled = True
      CONIDX2 = 3
     End If
   Else
   End If
End Sub

Sub set_MouseDown (Button As Integer, Shift As Integer, X As
                   Single, Y As Single)
     image5.Picture = image3.Picture
End Sub
Sub set_MouseUp (Button As Integer, Shift As Integer, X As
                 Single, Y As Single)
    image5.Picture = image2.Picture
End Sub

Sub snr_Click ()
     Beep
    main.Comm1.Output = Chr$(54)
End Sub

Sub snr_MouseDown (Button As Integer, Shift As Integer, X As
                   Single, Y As Single)
     image5.Picture = image3.Picture
End Sub

Sub snr_MouseUp (Button As Integer, Shift As Integer, X As
                 Single, Y As Single)
     image5.Picture = image2.Picture
End Sub
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
Sub swon_Click ()
    Beep
   main.Comm1.Output = Chr$(48)
   led(IDX).FillColor = &HFF&
   OFFLABEL$ = "ON"
   DISPLAY
End Sub
Sub swon_MouseDown (Button As Integer, Shift As Integer, X As
                    Single, Y As Single)
image5.Picture = image3.Picture
End Sub
Sub swon_MouseUp (Button As Integer, Shift As Integer, X As
                  Single, Y As Single)
     image5.Picture = image2.Picture
End Sub
Sub Timer1_Timer ()
   main.mnuTime.Caption = Time$
End Sub

Sub Timer2_Timer ()
     DIMM$ = String$(I, ">")
     Static PickBmp As Integer
     If PickBmp Then
LINE2 = DIMM$ & "_"
     Else
LINE2 = DIMM$
     End If
     PickBmp = Not PickBmp
End Sub
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ฉ

การใช้งานระบบตรวจจับและควบคุมด้วยเครื่องไมโครคอมพิวเตอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ระบบตรวจจับและควบคุมด้วยเครื่องไมโครคอมพิวเตอร์ แบ่งการใช้งานการควบคุมระบบเป็น 2 ลักษณะ คือควบคุมการใช้งานที่เครื่องคอมพิวเตอร์ และใช้งานที่ตัวเครื่องชุดสาธิตระบบตรวจจับและควบคุมที่สร้างขึ้น ดังนั้นจึงแบ่งการทดลองเป็น 2 ส่วน คือ

- การใช้งานระบบตรวจจับและควบคุมบนชุดสาธิตที่สร้างขึ้น
- การใช้งานระบบตรวจจับและควบคุมบนเครื่องคอมพิวเตอร์

ขั้นตอนการใช้งาน

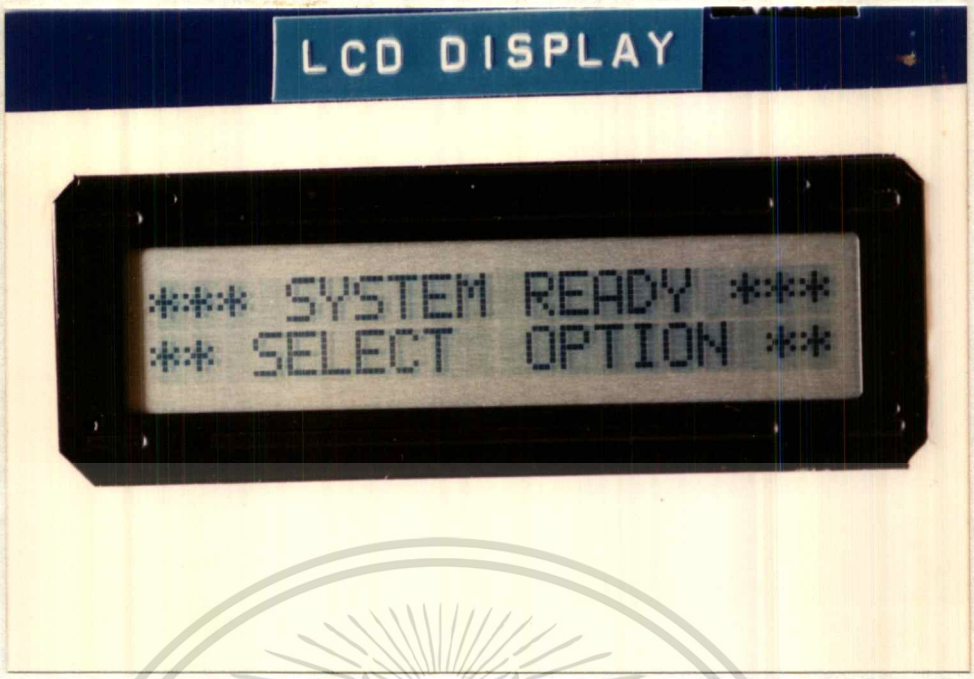
- เปิดสวิทช์จ่ายไฟดีซีให้กับชุดสาธิตระบบตรวจจับและควบคุม จะปรากฏข้อความดังรูปที่ 1 บนจอแอลซีดีสักครู่แล้วหายไป ซึ่งเป็นชื่อของระบบที่สร้างขึ้น จากนั้นจะปรากฏข้อความบนจอ แอลซีดี ดังนี้

```
[INC] RUN ON PC
[DEC] STAND ALONE
```

เป็นการแสดงสภาวะเตรียมพร้อมในการรอรับคำสั่งในการเลือกใช้งาน ซึ่งสามารถเลือกใช้งานบนเครื่องคอมพิวเตอร์ โดยการกดปุ่ม INC ที่เมตริกซ์คีย์บอร์ดบนชุดสาธิตระบบ หรือเลือกใช้งานบนชุดสาธิตโดยการกดปุ่ม DEC จากนั้นจะปรากฏข้อความ ดังรูปที่ 2 เป็นการแสดงสภาวะเตรียมพร้อมในการรอรับคำสั่งในการเลือกใช้งานต่างๆในระบบ

รูปที่ 1 แสดงสภาวะเริ่มต้นการทำงานของระบบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2 แสดงสภาวะเตรียมพร้อมของการใช้งานระบบ

การใช้งานระบบตรวจจับและควบคุมบนชุดสาธิตที่สร้างขึ้น

เป็นการใช้งานในลักษณะควบคุมบนชุดสาธิตระบบโดยอิสระ (STAND ALONE) โดยใช้เมตริกซ์คีย์บอร์ดขนาด 4x4 ซึ่งแสดงผลการใช้งานบนจอ แอลซีดี ขนาด 20 ตัวอักษร 2 แถว

การตั้งนาฬิกาเวลาที่แท้จริงให้กับระบบ (REAL TIME CLOCK)

ลำดับขั้นการใช้งาน

- กดปุ่ม | CLK | จะปรากฏ | << CURRENT TIME >> SETUP & DISPLAY | บนจอแอลซีดีสักครู่ แล้วจึงหายไป และปรากฏ

```
DATE=> MON 05/12/94
TIME=> 01:09:59
```

ซึ่งเป็นเวลาเดิมที่ตั้งเอาไว้และนาฬิกาจะเดินอยู่

- กดปุ่ม | SET | จะปรากฏเคอร์เซอร์กระพริบที่หลักของวินาที ให้ใส่ค่าวินาที

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ที่หลักวินาทีโดยการกดที่ปุ่ม [INC] หรือปุ่ม [DEC] จากนั้นให้กดปุ่ม [NEXT] เคอร์เซอร์จะเลื่อนไปที่หลักของนาที ให้ใส่ค่านาทีด้วยปุ่ม [INC] หรือกดที่ปุ่ม [DEC] เช่นกัน จากนั้นให้กดปุ่ม [NEXT] ไปตั้งหลักชั่วโมง, วัน, วันที่, เดือน และ ปี ตามลำดับ ในลักษณะเดิม

- เมื่อตั้งค่าถึงหลักของปีแล้วให้กดปุ่ม [NEXT] หนึ่งครั้งแล้วกดปุ่ม [O.K.]

- เมื่อควบคุมการตั้งเวลาเรียบร้อยแล้วให้กดปุ่ม [EXIT] ออกจากการทำงาน ที่หน้าจอแอลซีดีจะปรากฏข้อความดังรูปที่ 2 เพื่อรอการใช้งานในส่วนอื่นๆ ต่อไป

การใช้งานควบคุมอุปกรณ์ไฟฟ้าในลักษณะเปิด/ปิด (ON/OFF) โดยปกติ

ลำดับขั้นการใช้งาน

- ต่ออุปกรณ์ไฟฟ้าตามที่ต้องการจะควบคุม ที่ปลั๊กบนชุดสาธิต ตามช่องต่าง ๆ ซึ่งออกแบบไว้ทั้งหมด 8 ช่อง (CH1-CH8)

- กดปุ่ม [MANL] (MANUAL) จะปรากฏ

```
<< MANUAL  OPTION >>
CHANNEL  : 01  [OFF]
```

บนจอแสดงผลด้วยแอลซีดี

- จะเห็นว่าระบบจะรอการควบคุมอยู่ที่แชลแนลที่ 1 จากนั้นทดลองกดปุ่ม [ON] และ [OFF] บนคีย์บอร์ด สังเกตอุปกรณ์ไฟฟ้าที่ต่ออยู่ที่จุด CH1

- ทดลองควบคุมอุปกรณ์ไฟฟ้าในแชลแนลอื่นๆ ซึ่งมีทั้งหมด 8 แชลแนล โดยการใช้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ปุ่ม [INC] และ [DEC] ในการเลือกช่องการทำงาน

- เมื่อควบคุมการเปิด/ปิด (ON/OFF) อุปกรณ์ไฟฟ้าแชลแนลต่างๆ เรียบร้อยแล้ว ให้กดปุ่ม [EXIT] ออกจากการทำงาน ที่หน้าจอแอลซีดีจะปรากฏข้อความ ดังรูปที่ 2 เพื่อรอการใช้งานในส่วนอื่นๆ ต่อไป

การใช้งานควบคุมอุปกรณ์ไฟฟ้าในลักษณะตั้งเวลาเปิด/ปิด (PROGRAM)

ลำดับขั้นการใช้งาน

- ต่ออุปกรณ์ไฟฟ้าตามที่ต้องการจะควบคุม ที่ปลั๊กบนชุดสาธิต ตามช่องต่าง ๆ ซึ่งออกแบบไว้ทั้งหมด 8 ช่อง (CH1-CH8)

- กดปุ่ม [PRG] (PROGRAM) จะปรากฏ

```
<< PROGRAM OPTION >>
CHANNEL : 01   [OFF]
```

บนจอแสดงผลด้วยแอลซีดี

- กดปุ่ม [SET] จะปรากฏ

```
CHANNEL : 01  [ ON ]
TIME> 00/00/00:00:00
```

บนจอแสดงผลด้วยแอลซีดี

- ใช้ปุ่ม [INC] หรือ [DEC] ในการเลือกช่องการทำงาน ซึ่งมีทั้งหมด 8 ช่องด้วยกัน

- กดปุ่ม [SET] จะเกิดเคอร์เซอร์กระพริบรอรับการตั้งเวลา จากหลักวินาที, นาที, ชั่วโมง, เดือน และวันที่ตามลำดับจากขวาไปซ้าย ด้วยปุ่ม [INC] หรือ [DEC]

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

และใช้ปุ่ม | NEXT | ในการเลื่อนตำแหน่งเคอร์เซอร์ไปที่หลักต่างๆ เมื่อตั้งถึงหลักวันที่แล้ว ระบบจะเปลี่ยนหน้าจอเป็น

```
CHANNEL : 01 [ OFF ]
TIME> 00/00/00:00:00
```

โดยอัตโนมัติ เพื่อให้ตั้งเวลาปิด (OFF) ด้วยวิธีการตั้งเช่นเดิม เมื่อตั้งเสร็จแล้วเคอร์เซอร์จะหายไป

- ทดลองควบคุมอุปกรณ์ไฟฟ้าในแชลแนลอื่นๆ ซึ่งมีทั้งหมด 8 แชลแนล โดยการใช้ปุ่ม | INC | และ | DEC | ในการเลือกช่องการทำงาน

- เมื่อควบคุมการตั้งเวลา เปิด/ปิด อุปกรณ์ไฟฟ้าแชลแนลต่างๆ เรียบร้อยแล้ว ให้กดปุ่ม | EXIT | ออกจากการทำงาน ที่หน้าจอแอลซีดีจะปรากฏข้อความ ดังรูปที่ 2 เพื่อรอการใช้งานในส่วนอื่นๆ ต่อไป

การใช้งานควบคุมความเข้มของหลอดไฟ (LIGHT INTENSITY CONTROL)

ลำดับขั้นการใช้งาน

- ต่ออุปกรณ์ไฟฟ้าตามที่ต้องการจะควบคุม ที่ปลั๊กบนชุดสาธิต ตามช่องต่าง ๆ ซึ่งออกแบบไว้ทั้งหมด 4 ช่อง (D1-D4)

- กดปุ่ม | DMC | (DIMMER CONTROL)ปรากฏ

```
DIMMER CHANNEL[ 01 ]
```

บนจอแสดงผลด้วยแอลซีดี

- ใช้ปุ่ม | INC | หรือ | DEC | ในการเลือกช่องการทำงาน ซึ่งมีทั้งหมด 4 ช่องด้วยกัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- กดปุ่ม | SET | จะเกิดเคอร์เซอร์กระพริบรอรับการควบคุม ที่ต้นบรรทัดที่ 2

- จากนั้นให้กดปุ่ม | INC | หรือ | DEC | เพื่อเพิ่ม หรือลดระดับความสว่างของหลอดไฟแสดงผลที่ต่อไว้ ในระดับความสว่างที่ต้องการ

- เมื่อกำหนดความสว่างของหลอดไฟแล้ว ต้องการควบคุมช่องการทำงานอื่น ให้กดปุ่ม | EXIT | หนึ่งครั้ง

- เมื่อควบคุมความสว่างของหลอดไฟแต่ละแชลแนลเรียบร้อยแล้ว ต้องการออกจากการทำงานให้กดปุ่ม | EXIT | อีกหนึ่งครั้ง ที่หน้าจอ แอลซีดี จะปรากฏข้อความดังรูปที่ 2 เพื่อรอการใช้งานในส่วนอื่นๆ ต่อไป

การใช้งานควบคุมการตรวจจับสัญญาณจากภายนอก (SENSORS CONTROL)

ลำดับขั้นการใช้งาน

- ต่ออุปกรณ์ไฟฟ้าตามที่ต้องการใช้แสดงผลการตรวจจับ ที่ปลั๊กบนชุดสาธิตตามช่องต่างๆ ซึ่งออกแบบไว้ทั้งหมด 4 ช่อง (CH9-CH12)

- กดปุ่ม | SCT | (SENSORS CONTROL) ปรากฏ

```
<< SENSOR CONTROL >>
CHANNEL1 -  CHANNEL4
```

บนจอแสดงผลด้วยแอลซีดี

- กดปุ่ม | SET | จะปรากฏ

```
CHANNEL : 01
LEVEL : 0000  mV.
```

บนจอแสดงผลด้วยแอลซีดี

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- ใช้ปุ่ม INC หรือ DEC ในการเลือกช่องการทำงาน ซึ่งมีทั้งหมด 4 ช่องด้วยกัน

- กดปุ่ม SET จะเกิดเคอร์เซอร์กระพริบ เพื่อรอรับการตั้งค่าระดับแรงดัน จากนั้นให้ตั้งค่าระดับแรงดันในการตรวจจับสัญญาณ ด้วยปุ่ม INC หรือ DEC และกดปุ่ม EXIT หนึ่งครั้ง เคอร์เซอร์กลับไปที่ตำแหน่งการเลือกแชลแนล

- เมื่อกำหนดระดับแรงดันในการตรวจจับแต่ละแชลแนลเรียบร้อยแล้ว ให้กดปุ่ม O.K. ที่หน้าจอ แอลซีดี จะปรากฏ

```
[CH1][CH2][CH3][CH4]
 OFF  OFF  OFF  OFF
```

- ทดลองป้อนระดับแรงดันเข้าที่จุดต่อ (TERMINAL) ตามช่องต่าง ๆ (S4-S7) ให้เกินค่าของระดับแรงดันที่ตั้งไว้ในแต่ละแชลแนล (ระวัง! อย่าป้อนแรงดันเกิน 5 โวลท์ เป็นอันขาด) แล้วดูผลการทำงานบน แอลซีดี และที่อุปกรณ์เอาท์พุท (CH9-CH12)

- กดปุ่ม EXIT ออกจากการทำงาน ที่หน้าจอแอลซีดี จะปรากฏข้อความดังรูปที่ 2 เพื่อรอการใช้งานในส่วนอื่นๆ ต่อไป

การใช้งานดีซีโวลท์มิเตอร์ 0-5 โวลท์

ลำดับขั้นการใช้งาน

- กดปุ่ม SNR (SENSORS) จะปรากฏ

```
<< DC. VOLTSMETER >>
CH : 01 [0000 mV.]
```

บนจอแสดงผลด้วยแอลซีดี

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- กดปุ่ม [INC] หรือ [DEC] ในการเลือกช่องการทำงาน ซึ่งมีทั้งหมด 3 ช่องด้วยกัน

- จากนั้นกดปุ่ม [O.K.] แล้วทดลองป้อนแรงดันที่ต้องการวัด (ไม่เกิน 5 V) เข้าที่จุดต่อ (TERMINAL) S1 หรือ S2 หรือ S3 แล้วแต่การเลือกแชลแนล ในข้อที่ 3 (ระวัง! อย่าป้อนแรงดันเกิน 5 โวลท์เป็นอันขาด) แล้วดูผลการทำงานบน แอลซีดี)

- กดปุ่ม [EXIT] ออกจากการทำงาน ที่หน้าจอแอลซีดี จะปรากฏข้อความดังรูปที่ 2 เพื่อรอการใช้งานในส่วนอื่นๆ ต่อไป

การใช้งานระบบตรวจจับและควบคุมบนเครื่องคอมพิวเตอร์

การใช้งานระบบตรวจจับและควบคุมบนเครื่องคอมพิวเตอร์ ทำได้โดยการเรียกใช้โปรแกรมวินด์โดวส์ และทำการรันโปรแกรมที่สร้างขึ้นบนไมโครซอฟท์วินโดวส์ จากนั้นใช้เมาส์หรือคีย์ลูกศรทำการควบคุมการใช้งานจากภาพคีย์บอร์ดที่จำลองขึ้นในโปรแกรม ซึ่งมีวิธีการใช้งานระบบเช่นเดียวกันกับการใช้งานบนชุดสาธิต

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บรรณานุกรม

บริษัท ซีเอ็ดยูเคชั่นจำกัด."การเขียนโปรแกรมบนไมโครซอฟท์วินโดวส์ ด้วย - Visual Basic ภาคปฏิบัติ": พิมพ์ที่ เอช.เอ็นกรุ๊ปจำกัด, 2537.

บริษัท ซีเอ็ดยูเคชั่นจำกัด."เซมิคอนดัคเตอร์อิเลคทรอนิคส์ ฉบับที่ 127 พฤศจิกายน 2536, ฉบับที่ 128 มิถุนายน 2536, ฉบับที่ 133 พฤศจิกายน 2536, ฉบับที่ 135 มกราคม 2537, ฉบับที่ 136 กุมภาพันธ์ 2537, ฉบับที่ 137 มีนาคม 2537, ฉบับที่ 138 เมษายน 2537, ฉบับที่ 140 มิถุนายน 2537": กรุงเทพมหานคร: OS PRINTING HOUSE.

บริษัท อีทีทีจำกัด. "CP-68HC11 COMPLETE EMBEDDED.": กรุงเทพมหานคร: โรงพิมพ์ทิพยวิสุทธิ์, 2536.

บริษัท อีทีทีจำกัด. "DOT MATRIX LCD MODULE.": กรุงเทพมหานคร: โรงพิมพ์ทิพยวิสุทธิ์, 2534.

บริษัท อีทีทีจำกัด. "ET-DEBUGGER 68HC11.": กรุงเทพมหานคร: โรงพิมพ์ทิพยวิสุทธิ์, 2536.

บริษัท อีทีทีจำกัด. "RTC V4.0.": กรุงเทพมหานคร: โรงพิมพ์ทิพยวิสุทธิ์, 2536.

Joseph D. Greefield. "The 68HC11 Microcontroller": Saunders College Publishing, 1991.

Motorola Inc. "M68HC11EVBU Universal Evalution Board - User's Manual": Motorola Inc., 1992.

Ramakant Gayakwad, Lednard Sokoloff. "Analog And Digital - Control Systems": Prentice-Hall Inc., 1988.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้