ปริญญานิพนธ์

แผงแสดงผล
LIGHT DISPLAY BOARD

นายวีระ สุขพันธุ์

วัน เดือน ปี ๔ ต.ค. ๒๕๔๑
เลขทะเบียน 038351
เลขเรียกหนังสือ T 99311 ว๓๔๑ พ

ปริญญานิพนธ์ฉบับนี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรอุตสาหกรรมศาสตร์บัณฑิต
สาขาวิชาเทคโนโลยีอิเล็กทรอนิกส์
ภาควิชาเทคนิคอุตสาหกรรม
คณะวิศวกรรมศาสตร์
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหาร ลาดกระบัง
ปีการศึกษา 2539

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้
038351

ปริญญานิพนธ์

เรื่อง แผงแสดงผล

LIGHT DISPLAY BOARD

ผู้จัดทำ

นายวีระ สุขพันธุ์

อาจารย์ที่ปรึกษา

ลงนาม.....

(อ. อรลาภ แสงอรุณ)

ลงนาม.....

(อ. ชวลิต เบญจางคประเสริฐ)

หัวหน้าภาควิชา

ลงนาม.....

(อ. อุทัย ศรีธีระวิโรจน์)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ปริญญานิพนธ์

เรื่อง แผงแสดงผล

LIGHT DISPLAY BOARD

จุดประสงค์

1. เพื่อศึกษาการทำงานของไมโครโปรเซสเซอร์เบอร์ Z-80
2. เพื่อศึกษาการนำไมโครโปรเซสเซอร์ไปใช้ในงานที่เกี่ยวข้องกับเวลา
3. เพื่อศึกษาการทำงานของระบบคอมพิวเตอร์แบบแผ่นวงจรเดียว
4. เพื่อศึกษาการนำเอาบอร์ดควบคุมZ-80 ไปใช้ในการควบคุมหลอดไฟ 220V
5. เพื่อศึกษาโปรแกรมภาษาแอสแซมบลี้ที่ใช้กับไมโครโปรเซสเซอร์เบอร์Z-80
6. เพื่อศึกษาการเขียนโปรแกรมภาษาแอสแซมบลี้ สั่งให้ไมโครโปรเซสเซอร์ Z-80 ไปควบคุมการเปิดปิดหลอดไฟ 220 V
7. ศึกษาการทำงานของวงจร SOLID-STATE RELAY

ประโยชน์ที่คาดว่าจะได้รับ

1. ได้ศึกษาพื้นฐานการทำงานของระบบคอมพิวเตอร์
2. ได้ศึกษาการเขียนโปรแกรมสั่งงานไมโครโปรเซสเซอร์ สามารถนำไปควบคุมการทำงานของอุปกรณ์ชนิดอื่นๆได้
3. ได้ศึกษาการทำงานของแผงแสดงผลที่ใช้หลอดไฟ
4. นำแผงแสดงผลที่ประกอบขึ้นไปใช้ในงานต่างๆ เช่น การแสดงผลของเวลา หรือใช้เป็น SCORE BOARD เป็นต้น
5. ใช้แผงแสดงผลที่ได้ นำไปศึกษา ค้นคว้า และพัฒนาให้มีประสิทธิภาพในการแสดงผลให้ดียิ่งขึ้นต่อไป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

แผงแสดงผล

นายวีระ สุขพันธุ์

อาจารย์ที่ปรึกษา

อ. อรลาภ แสงอรุณ

อ. ชวลิต เบญจางคประเสริฐ

บทคัดย่อ

ปริญญานิพนธ์ฉบับนี้ได้นำเสนอ แผงแสดงผล โดยใช้หลอดไฟเป็นตัวแสดงผล และควบคุมการแสดงผลด้วยบอร์ดควบคุมที่ใช้ไมโครโปรเซสเซอร์เบอร์ Z-80 เป็นหน่วยประมวลผลกลาง โดยเน้นให้แสดงผลเป็นตัว SCORE BOARD เป็นหลัก แต่สามารถนำไปประยุกต์ใช้เป็นตัวแสดงผลอื่นๆได้ เช่น เป็นตัวนาฬิกาบอกเวลา เป็นต้น

อุปกรณ์ที่ใช้ในการต่อระบบไฟ AC 220 V ที่จ่ายให้หลอดแสดงผลคือ SSR (SOLID-STATE RELAY) ซึ่งรับสัญญาณมาจาก PORT 8255 บนบอร์ดควบคุม

หลอดไฟที่ใช้แสดงผลเป็นหลอดไฟ 220 โวลท์ 40 วัตต์ จำนวน 127 หลอด นำมาเรียงกันเพื่อให้แสดงผลเป็นตัวเลขหรือตัวอักษรภาษาอังกฤษได้บางตัว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

LIGHT DISPLAY BOARD

Mr. WEERA SUKAPAN

ADVISOR

Miss ORALAP SEANAROON

Mr. CHAOWARIT BENJANGKAPRASURD

ABSTRACT

This thesis presents the light display board,which used the display lamp 220 V,40 W. It has 127 lamp on the board. The light display board can be display time and other display for example "SEIKO". At the control board has a microprocessor , the Zilog Z-80 is a central processing unit. The device that used for ON or OFF AC supply 220 V is Solid State Relay

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กิตติกรรมประกาศ

บริญญานิพนธ์ฉบับนี้สำเร็จลุล่วงลงได้เนื่องจากการได้รับความช่วยเหลือจากท่านอาจารย์ที่ปรึกษาและอาจารย์ประจำภาควิชาเทคนิคอุตสาหกรรมทุกท่าน ขอขอบคุณเพื่อนพนักงานบริษัทวิทยุการบินแห่งประเทศไทย ลังกัดกองวิศวกรรมติดตามอากาศยานทุกท่าน ที่ได้ให้ความช่วยเหลือในการประกอบอุปกรณ์และให้ข้อแสนอแนะในการแก้ไขปัญหาต่างๆที่เกิดขึ้นในระหว่างการดำเนินงาน รวมถึงบริษัทวิทยุการบินแห่งประเทศไทยที่ให้การสนับสนุนทางด้านเงินทุนในการจัดซื้ออุปกรณ์ทั้งหมด และได้เอื้อเฟื้อสถานที่รวมทั้งอุปกรณ์เครื่องไม้เครื่องมือต่างๆที่ใช้ประกอบแผงแสดงผลนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ

เรื่อง			หน้า
บทคัดย่อภาษาไทย			I
บทคัดย่อภาษาอังกฤษ			II
กิตติกรรมประกาศ			III
สารบัญ			IV
สารบัญรูปภาพ			VI
สารบัญตาราง			VIII
บทที่ 1	บทนำ		1
บทที่ 2	ทฤษฎีและหลักการทำงาน		2
	2.1	หลักการในการทำให้เกิดตัวเลขหรือตัวอักษรบนแผงแสดงผล	
	2.2	KEYBOARD MAYRIX 3*4	
	2.3	REAL TIME CLOCK	
	2.4	SOLID STATE RELAY	
บทที่ 3	ส่วนประกอบของแผงแสดงผล		22
	3.1	CONTROL BOARD Z-80	
	3.2	SSR-AC	
	3.3	การเชื่อมต่ออุปกรณ์	
	3.4	การใช้งานแผงแสดงผล	
บทที่ 4	การทดลองและผลการทดลอง		37
บทที่ 5	บทสรุป		44
	5.1	สรุปผลการทดลอง	
	5.2	แนวทางในการพัฒนา	
	5.3	ปัญหาที่เกิดขึ้นในระหว่างการทดลอง	

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เรื่อง		หน้า
บรรณานุกรม		46
ภาคผนวก ก.	Z80 CPU Specification	47
ภาคผนวก ข.	วิธีการพัฒนาโปรแกรมควบคุม	48
ภาคผนวก ค.	KEYBOARD 3*4	49
ภาคผนวก ง.	ส่วนประกอบของ CONTROL BOARD Z-80	50
ภาคผนวก จ.	ส่วนประกอบของ SSR-AC	51
ภาคผนวก ฉ.	โปรแกรมการใช้งาน	52

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูปภาพ

รูปภาพ		หน้า
2.1	หลักการแสดงผล	
	รูปที่ 1 รูปด้านหน้าแผงแสดงผล	3
	รูปที่ 2 บล็อกไดอะแกรมแผงแสดงผล	3
2.2	KEYBOARD MATRIX 3*4	
	รูปที่ 1 วงจร KEYBOARD แบบต่างๆ	6
	รูปที่ 2 บล็อกไดอะแกรมของวงจรเข้ารหัส KEYBOARD	6
	รูปที่ 2 รูปคลื่นการตัด-ต่อหน้าสัมผัส	7
	รูปที่ 3 รูปคลื่นการตัดต่อ SWITCH	7
	รูปที่ 4 วิธีการแก้ DEBOUNCE ของ SWITCH	8
	รูปที่ 5 การต่อ KEYBOARD เข้ากับ IC 8255	9
	รูปที่ 6 FLOWCHART ของการแก้ BOUNCING	10
2.3	RTC MSM 6242B	
	รูปที่ 1 การจัดขาของ IC MSM 6242B	12
2.4	SOLID STATE RELAY	
	รูปที่ 1 วงจร SSR	16
	รูปที่ 2 อุปกรณ์ทาง OPTICAL และการต่อใช้งาน	17
	รูปที่ 3 TRANSFORMER ในวงจร SSR	28
	รูปที่ 4 วงจร SSR	20
3.1	CONTROL BOARD Z-80	
	รูปที่ 1 JP1 กำหนดหน่วยความจำ U3	23
	รูปที่ 2 JP2 เลือกเบอร์หน่วยความจำ U4	23
3.2	SSR-AC	
	รูปที่ 1 การต่อ SSR-AC กับ CONTROL BOARD Z-80	28
	รูปที่ 2 การต่อ SSR-AC แบบขยาย	29

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปภาพ หน้า

3.3 การเชื่อมต่ออุปกรณ์

รูปที่ 1 ต่อหลอดไฟทำงานที่ PORT A 32

รูปที่ 2 การต่อ SWITCH เข้ากับ IC 8255 ตัวที่ 1 34

รูปที่ 3 แสดงการต่อ IC 8255 เข้ากับส่วนแสดงผล 35

4 การทดลองและการแสดงผล

รูปที่ 1 การต่อแผงแสดงเพื่อทดลอง 37

รูปที่ 2 ผลที่ได้จากการทดลอง 43

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญตาราง

ตาราง หน้า

2.3 RTC MSM 6242B

ตารางที่ 1 REGISTER 13

ตารางที่ 2 REGISTER W 14

3.1 CONTROL BOARD Z-80

ตารางที่ 1 ตำแหน่ง ADDRESS ของ RTC 24

ตารางที่ 2 JP3 26

ตารางที่ 3 ADDRESS ที่ใช้ติดต่อกับ EEPROM 26

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 1
บทนำ

ในปัจจุบัน ความก้าวหน้าในด้านเทคโนโลยี่การแสดงผลได้เจริญก้าวหน้าไปตามลำดับ ป้ายโฆษณาต่างๆได้เปลี่ยนแปลงรูปแบบการแสดงผล ที่สามารถแสดงผลเป็นรูปภาพหรือตัวอักษรที่สามารถเคลื่อนไหวได้ ทำให้สามารถดึงดูดความสนใจแก่ผู้ที่ได้มากขึ้น บางคนที่เคยได้พบเห็นแผ่นป้ายโฆษณาต่างๆเหล่านี้อาจจะนึกสงสัยว่าเขาทำได้อย่างไร

ปริญญานิพนธ์ฉบับนี้ได้จัดทำแผงแสดงผลที่สามารถแสดงผลเป็นตัวเลขหรือตัวอักษรได้บางตัว เพื่อให้เรียนรู้ของการแสดงผลที่เราได้พบเห็นโดยทั่วไป และสามารถนำความรู้ที่ได้รับนำไปปรับปรุงและประยุกต์ใช้งานให้เกิดประโยนช์มากยิ่งขึ้น

ในปริญญานิพนธ์ฉบับนี้ มีเนื้อหาต่างๆ ดังต่อไปนี้

บทที่ 1 ได้เกริ่นนำถึงความเป็นมาและอธิบายเนื้อหาคร่าวๆ ในแต่ละบทของปริญญานิพนธ์

บทที่ 2 ทฤษฎีและหลักการ เป็นทฤษฎีและหลักการที่เกี่ยวข้องกับแผงแสดงผล ได้แก่ หลักในการทำให้เกิดตัวเลขหรือตัวอักษร,คีย์บอร์ด, REAL TIME CLOCK,SOLID STATE RELAY

บทที่ 3 ส่วนประกอบของแผงแสดงผล เป็นการอธิบายเกี่ยวกับอุปกรณ์ต่างๆที่นำมาประกอบเป็นแผงแสดงผล ได้แก่ CONTROL BOARD Z-80,SSR-AC และการเชื่อมต่ออุปกรณ์เหล่านี้ รวมถึงขั้นตอนการใช้งานเครื่อง

บทที่ 4 การทดลองและผลการทดลอง เป็นการทดสอบเครื่องเพื่อที่จะนำไปใช้งานและได้แสดงโปรแกรมที่ทำให้เกิดการแสดงผลแต่ละอย่าง อธิบายขั้นตอนการทดลองรวมถึงผลการทดลองที่เกิดขึ้น

บทที่ 5 บทสรุปและแนวทางในการพัฒนา ได้สรุปผลการทดลองและเสนอปัญหาที่เกิดขึ้น ตลอดทั้งแนวทางในการพัฒนาของโครงงานนี้

ในภาคผนวก ได้ให้รายละเอียดของไมโครโปรเซสเซอร์ Z-80,วิธีการพัฒนาโปรแกรม,โปรแกรมการใช้งานและรายละเอียดของรายการอุปกรณ์ที่ใช้ในโครงงานนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 2
ทฤษฎีและหลักการทำงาน

2.1 หลักการแสดงผล

การทำให้เกิดตัวอักษรหรือตัวเลขบนแผงแสดงผล สามารถทำได้ 2 ลักษณะใหญ่ๆ คือ

1. LATCH
2. SCAN

1. LATCH วิธีนี้ ในการเขียนโปรแกรมแสดงผลจะง่ายและแผงแสดงผลจะมีความสว่างเท่ากันทุกดวง แต่จะมีข้อเสียคือต้นทุนในการผลิตมีราคาสูงมาก เพราะ 1 ดวงของหลอดไฟ จะเท่ากับชุด DRIVE 1 ชุด

2. SCAN วิธีนี้ ในการเขียนโปรแกรมการแสดงผลจะมีความยุ่งยากมาก แต่มีข้อดีในการประหยัดทางด้าน HARDWARE ได้เป็นจำนวนมาก วิธีการ SCAN ยังแบ่งออกได้ 2 ลักษณะ คือ

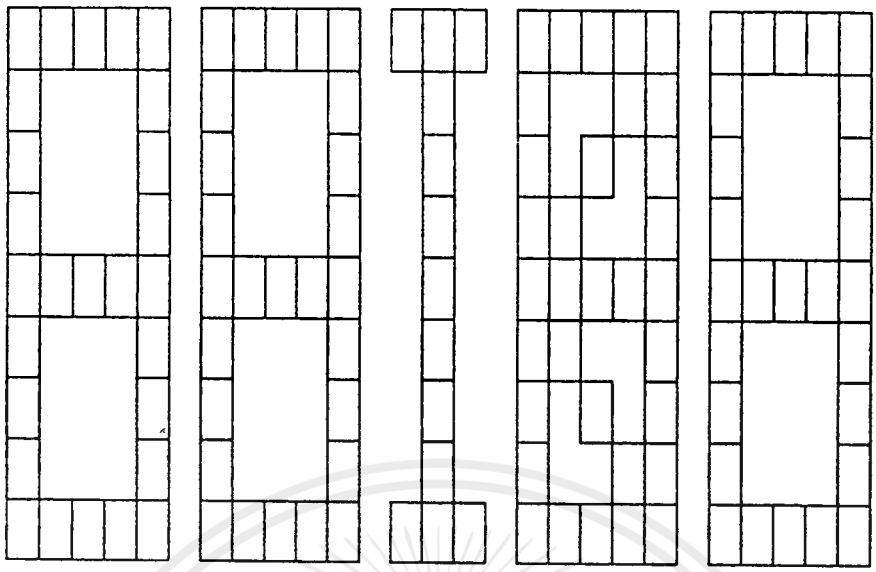
   1. SCAN ทาง COLUMN
   2. SCAN ทาง ROW

ในโครงงานนี้จะใช้วิธีการแสดงผลแบบ LATCH เนื่องจากต้องใช้กับไฟ AC 220 โวลท์

การจัดวางหลอดไฟ

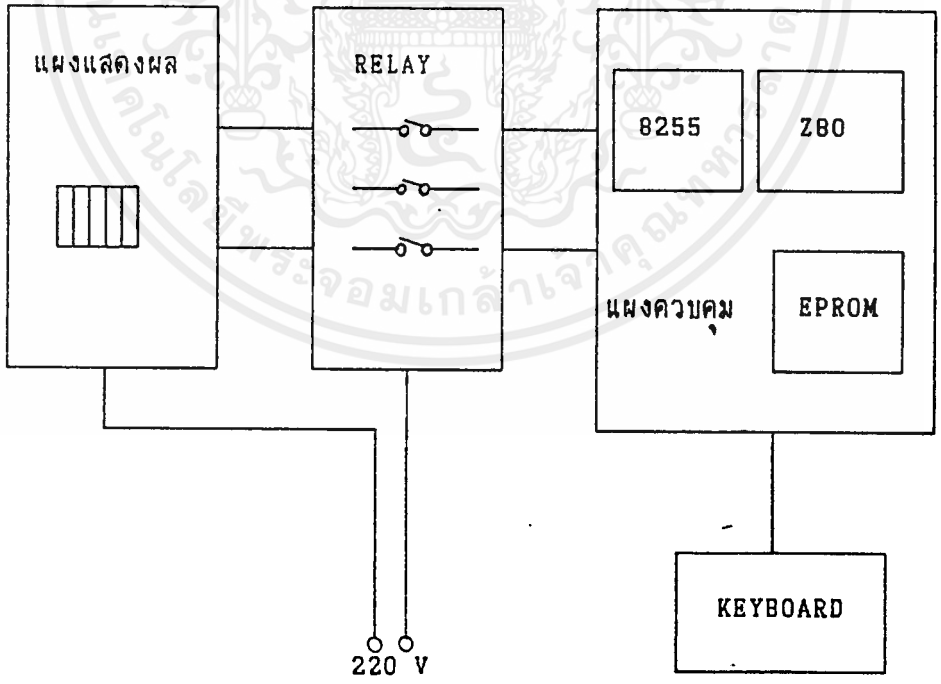
นำหลอดไฟ AC 220 V,40 W มาวางเพื่อให้แสดงผลตัวอักษรตางๆ ดังรูป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 1 รูปด้านหน้าแผงแสดงผล

วงจรการเชื่อมต่อ



รูปที่ 2 บล็อกไดอะแกรมของแผงแสดงผล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การทำงาน

การสั่งให้แสดงผลต่างๆ สั่งงานโดยผ่านทาง KEYBOARD

ภายในแผงควบคุมประกอบด้วยอุปกรณ์หลักๆ 3 ส่วน คือ

1. Z80 CPU เป็นไมโครโปรเซสเซอร์ที่ใช้ในการประมวลผลข้อมูลต่างๆ
2. IC 8255 เป็น I/O PORT ทำหน้าที่ส่งข้อมูลที่ได้จาก CPU ไปเปิดปิด RELAY
3. EPROM # 27256 เป็นที่เก็บโปรแกรมให้กับ CPU

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.2 KEYBOARD MATRIX 3*4

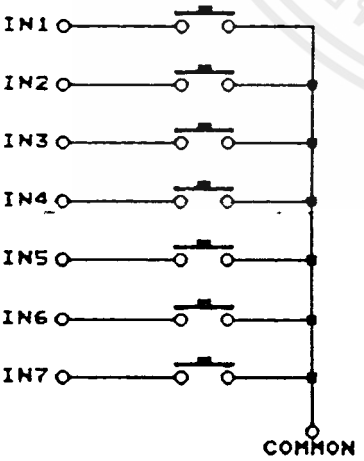
KEYBOARD นับเป็นสิ่งสำคัญอย่างหนึ่ง เพราะเป็นสิ่งที่ทำให้ผู้ใช้สามารถติดต่อกับเครื่องควบคุมในการทำงานต่างๆได้ โดย KEYBOARD จะประกอบส่วนสำคัญ 3 ส่วน คือ

1. ส่วนของ SWITCH KEYBOARD
2. ส่วนของวงจรเข้ารหัส KEYBOARD โดยส่วนนี้จะทำหน้าที่เข้ารหัสเมื่อรับรู้ตำแหน่งของ KEY และสามารถให้ค่า KEY ของแต่ละตัว เมื่อมีการกด KEY ที่เรียกว่า รหัส KEY (KEY CODE) ได้
3. ส่วนของวงจรถอดรหัส KEYBOARD จะเป็นส่วนที่ทำหน้าที่เปลี่ยนรหัส KEY CODE ให้เป็นรหัสที่นำไปใช้งานได้เช่น รหัส ASCII,รหัส BCD,รหัส HEX เป็นต้น

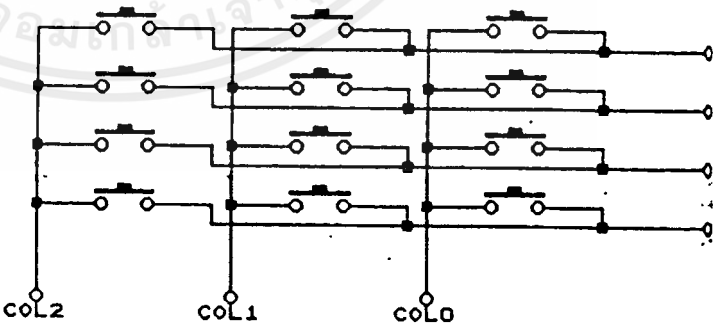
วงจร KEYBOARD

แบ่งได้เป็น 2 ลักษณะใหญ่ๆ คือ

1. แบบขาร่วม
2. แบบ MATRIX



แบบขาร่วม

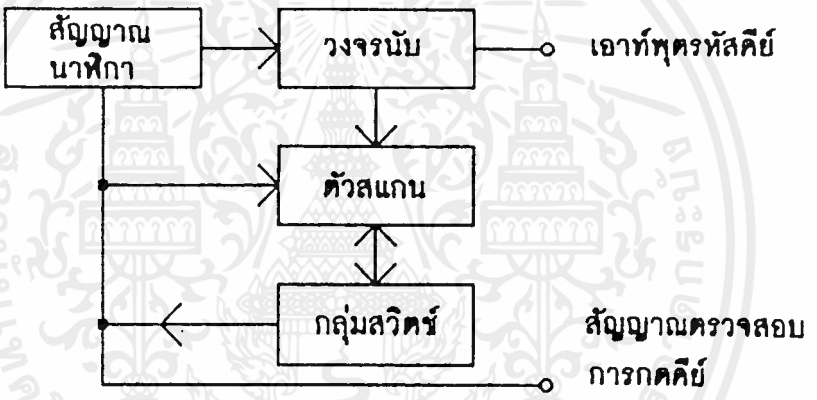


แบบ MATAIX

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 1 วงจร KEYBOARD แบบต่างๆ

จะเห็นว่า แบบขาร่วมจะเป็นแบบที่เข้าใจง่ายและรับรู้การกด KEY จะเป็นไปในเฉพาะตัวของแต่ละ KEY ซึ่งทำให้ง่ายต่อการออบแบบและสร้าง แต่ว่าจะสะดวกและประหยัดในกรณีที่ใช้ SWITCH ไม่มากนัก ส่วนแบบ MATRIX จะใช้สายในการต่อเข้ากับวงจรเข้ารหัสได้น้อยกว่า โดยให้จำนวนจุดของ KEY ได้จำนวนมาก ซึ่งจำนวน KEYจะขึ้นอยู่กับ สายทางด้าน ROW * สายทางด้าน COLUMN แต่การออกแบบวงจรเข้ารหัสจะยุ่งยากขึ้น โดยสามารถแสดงบล็อกไดอะแกรมของวงจรเข้ารหัส KEYBOARD ไว้ดังรูป



รูปที่ 2 บล็อกไดอะแกรมของวงจรเข้ารหัส KEYBOARD

จากรูป สัญญาณนาฬิกา จะใช้ SCAN ว่า KEY ใดถูกกด โดยเมื่อเราไม่มีการกด KEY ตัว SCAN จะส่งสัญญาณ SCAN ไปที่ SWITCH และเมื่อมีการกด KEY สัญญาณนาฬิกาจะหยุดทำงาน ทำให้ตัว SCAN หยุดการ SCAN วงจรนับจะหยุดนับและคงค่าสุดท้ายไว้(รหัส KEY ที่ถูกกด) จนกว่าจะมีการปล่อย KEY ถึงจะรับค่า KEYBOARD ตัวใหม่ได้ ซึ่งลักษณะการ SCAN แบบ MATIRX ของ KEY นี้ จะเริ่ม SCAN จาก ROW แรกไปยัง ROW สุดท้าย และ COLUMN แรกไปยัง COLUMN สุดท้ายพร้อมกัน ดังนั้นเมื่อ KEY ใดถูกกด ก็จะทำให้ทราบ KEY ที่ถูกกดว่าอยู่ ROW และ COLUMNใดได้จากส่วนของ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

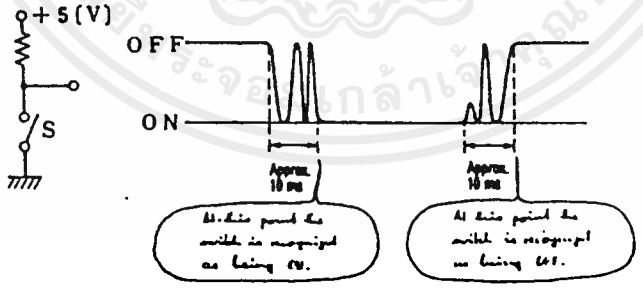
วงจรนับ

SWITCH BOUNCE

เนื่องจากเมื่อมีการกด KEY SWITCH ก็จะทำให้หน้าสัมผัสต่อ แต่การต่อของหน้าสัมผัสนี้จะไม่เป็นไปตามอุดมคติ ดังรูป

รูปที่ 2 รูปคลื่นการตัด-ต่อหน้าสัมผัส

ด้วยเหตุว่า หน้าสัมผัสจริงๆ ไม่ได้ราบเรียบหรือสะอาดอยู่เสมอและแรงกดนิ้วบนแป้นกด SWITCH ก็ยังแตกต่างกันไป ทำให้เกิดการสะบัดหรือแกว่งของแป้นกดชั่วขณะหนึ่งหลังจากกด SWITCH ลงไป ซึ่งเรียกว่าเป็นการกระเด้งของหน้าสัมผัส(BOUNCING) ซึ่งจะทำให้การต่อวงจรจริงของ SWITCH จะเป็น ดังนี้

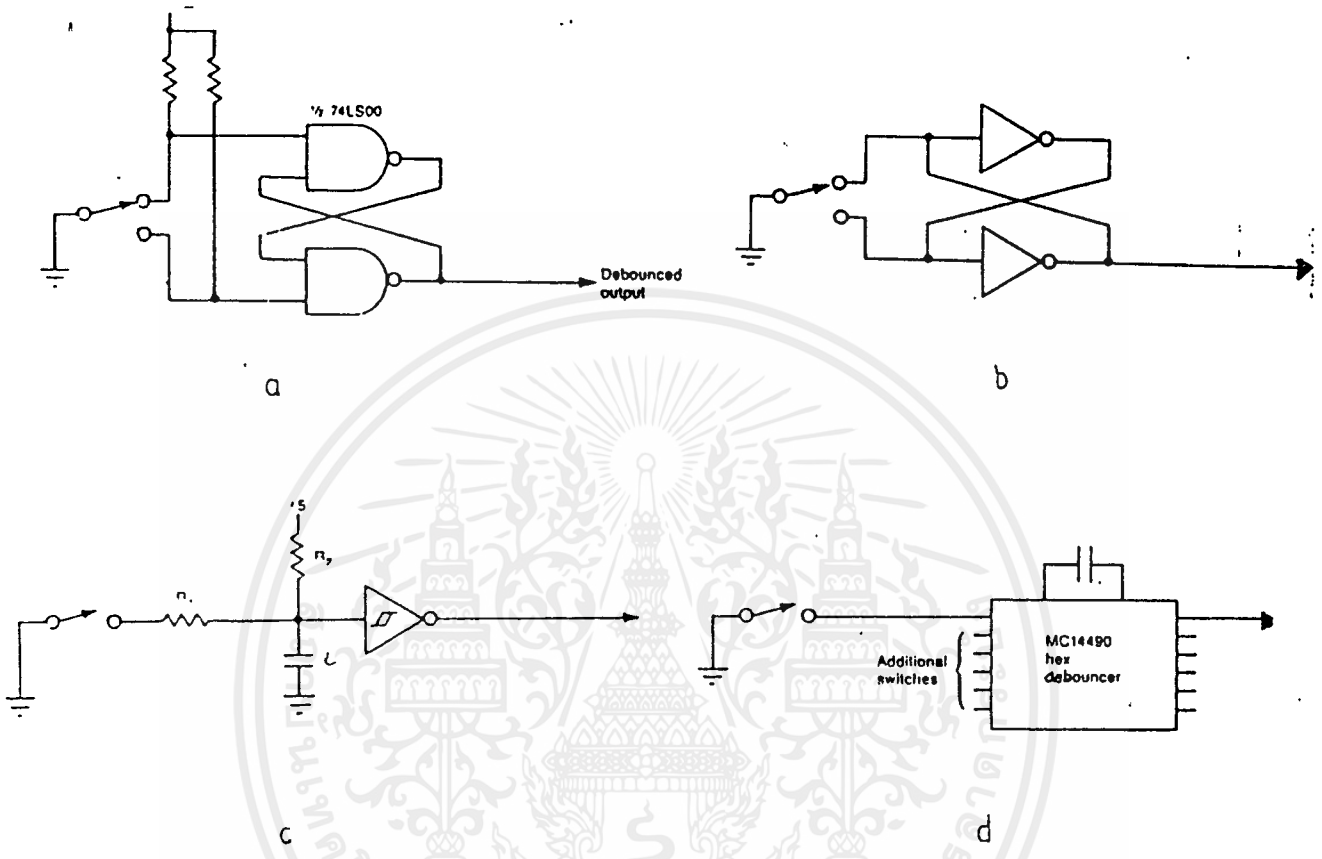


รูปที่ 3 รูปคลื่นการตัด-ต่อ SWITCH

จึงทำให้ตัวตรวจจับ KEY ที่กด ตรวจพบว่า KEY ดังกล่าวมีการกดหลายครั้ง ทั้งที่ความจริง ผู้ใช้ต้องการกดเพียงครั้งเดียว ซึ่งช่วงการเกิดของสัญญาณที่ไม่ต้องการนี้จะ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

มีช่วงเวลาประมาณ 10 MS เราสามารถแก้การ DEBOUNCE ของ SWITCHได้หลายวิธี ดังรูป



รูปที่ 4 วิธีการแก้การ DEBOUNCE ของ SWITCH

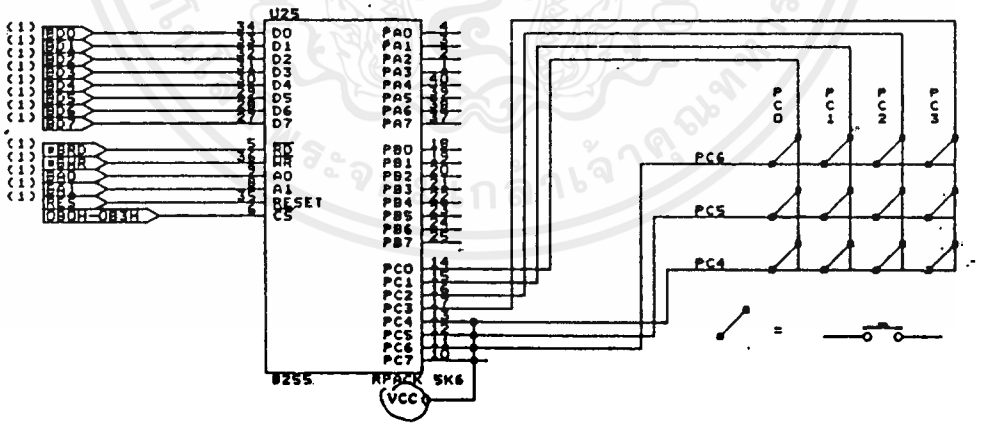
ใน แบบ RS LATCH ขณะ SWITCH ไม่ถูกเปลี่ยนทาง OUTPUT NAND GATE ตัวบนจะมี LOGIC "1" เนื่องจากเมื่อมีขา INPUT ใด INPUT หนึ่งเป็น "0"จะทำให้ INPUT ของ NAND GATE ตัวล่างมี INPUT เป็น "1" ทั้งคู่ ดังนั้น OUTPOT DEBOUNCE ก็จะมี LOGIC เป็น "0" แต่เมื่อมีการเปลี่ยนทิศทาง SWITCH ทำให้หน้าสัมผัส SWITCH ด้าน NAND GATE ตัวบนลอย ส่วนหน้าสัมผัส SWITCH NAND GATE ตัวล่างจะมีการต่อและจะมีการ BOUNCE ที่หน้าสัมผัสนี้เกิดขึ้น แต่ช่วงขณะนี้ NAND GATE ตัวล่าง ได้รับ INPUT เป็น "0" ครั้งแรก OUTPUT DEBOUNCE ก็จะเป็น "1" และเป็น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ผลให้ NAND GATE ตัวบนมี INPUT เป็น "1" ทั้งคู่ ทำให้ OUTPUTตัวบนเป็น "0"และเป็นผลให้ NAND GATE ตัวล่างมี INPUT ด้างเป็น "0" อยู่ด้านหนึ่ง อันเนื่องจาก OUTPUT NAND GATE ตัวบน ดังนั้นแม้ SWITCH ทางด้าน NAND GATE ตัวล่างจะ BOUNCE อย่างไรก็ตาม OUTPUT ที่ได้ก็จะไม่เปลี่ยนแปลงเพราะ INPUT เป็น "0"ค้างอยู่ด้านหนึ่ง แต่วิธีนี้จำเป็นต้องใช้ SWITCH เป็นแบบ 2 ทาง จึงจะแก้ BOUNCE ได้

ดังนั้นถ้า SWITCH เราเป็นแบบทางเดียวหรือแบบกดติด-ปล่อยดับแล้วในรูป Cจะช่วยได้ โดยการแก้การ BOUNCE ขึ้นอยู่กับค่า R และ C เช่น เมื่อเรากด SWITCHจะทำให้ C ถูก DISCHARGE ผ่าน R1 จนถึงระดับหนึ่งถึงจะทำให้ OUTPUT ของSCHMITT-TRIGGER เป็น LOGIC "1" ดีอจะไม่เป็น "1" ทันทีทันใด ซึ่งช่วงที่เป็น "1" นี้ก็คือช่วงที่เลยเวลาของการ BOUNCE ไปแล้ว และในขณะ SWITCH OPEN ก็จะไม่เป็น"0" ทันที เพราะจะมีการ CHARGE ของ C ผ่าน R2 จนถึงช่างเวลาหนึ่งถึงจะได้ LOGIC "0" เพราะคุณสมบัติของ SCHMITT-TRIGGER ด้วย

การใช้โปรแกรมในการทำ KEYBOARD แบบ MATRIX

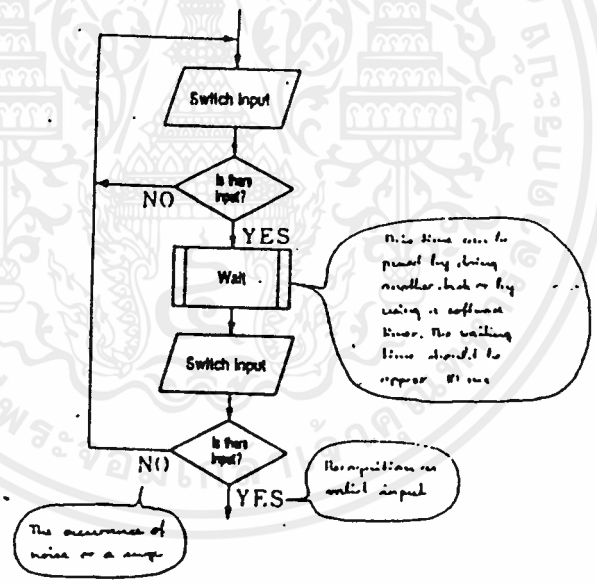


รูปที่ 5 การต่อ KEYBOARD เข้ากับ IC 8255

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากรูป จะเป็นวงจรของ KEYBOARD ขนาด 3*4 คือมี ROW ที่ใช้เป็น INPUT (อ่านข้อมูลของ KEY) 3 เส้น โดยจำเป็นต้องมี R PULL UP ไว้เพื่อยังไม่ได้กด KEY จะได้มีสถานะคงที่อยู่ค่าๆ ค่าหนึ่ง เพราะเราจะ SCAN LOGIC "0" ทางด้าน COLUMN หลักการก็คือ ส่งข้อมูล LOGIC "0" ให้กับ COLUMN แรกและอ่านข้อมูลทางด้าน ROW ดูว่ามี ROW เส้นใดเป็น "0" บ้าง ถ้ามี แสดงว่ามีการกด KEY ก็ไปทำการตรวจเช็คว่าเป็น ROW ใดและ COLUMN ใด เพื่อเข้ารหัส KEY แล้วแปลงเป็นค่าที่ต้องการไปใช้งาน แต่ถ้าอ่านแล้วไม่มี ROW เส้นใดเป็น "0" ก็ ACTIVE COLUMN ถัดไปให้เป็น "0" แล้วก็อ่าน ROW ใหม่ ทำอย่างนี้จนกว่าจะหมด COLUMN

การแก้ BOUNCING โดย SOFTWARE



รูปที่ 6 FLOWCHART ของการแก้ BONCING

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

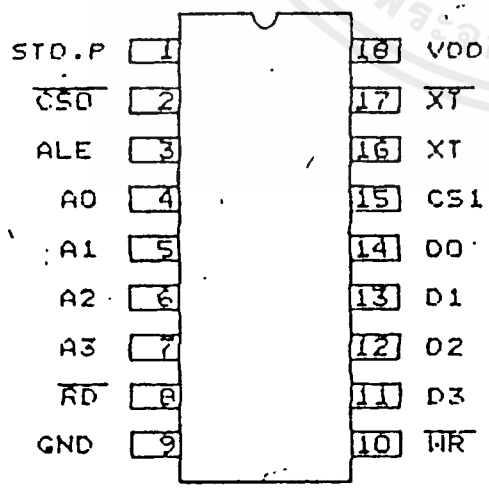
2.3 RTC MSM 6242B

การนำไมโครโปรเซสเซอร์ไปใช้งานที่เกี่ยวข้องกับเวลา โดยใช้ RTC เป็นตัวบอก วัน,เดือน,ปี,วันในรอบสัปดาห์,ชั่วโมง,นาที,วินาที การติดต่อกับ RTC เปรียบเสมือนกับการติดต่อกับพอร์ตอินพุท/เอาท์พุท คือ สามารถเขียนข้อมูลเกี่ยวกับเวลาไปที่เบอร์พอร์ทของ RTC ตัวนับเวลาภายในก็จะเดินตามเวลาที่ตั้งไว้และสามารถอ่านข้อมูลจากRTC ได้เช่นกัน

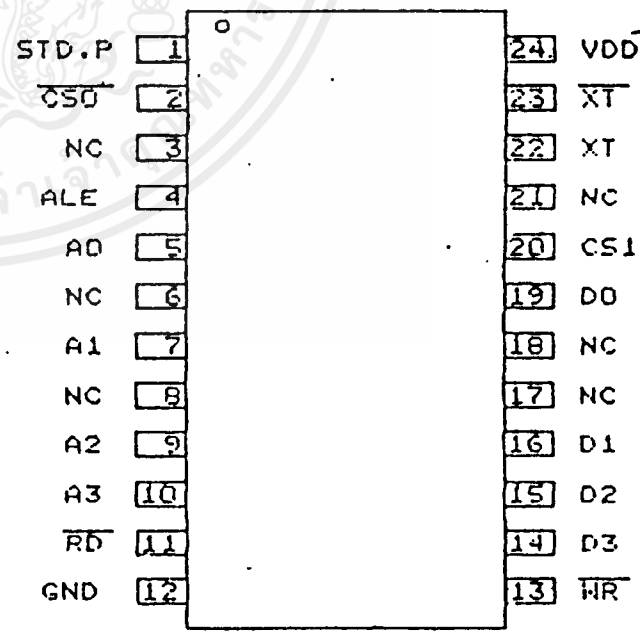
MSM 6242B สามารถปรับวันที่ให้ถูกต้องกับเดือนได้ ทั้งเดือนที่ลงท้ายด้วย"คม" และ "ยน" หรือเดือนกุมภาพันธ์ที่มี 28 และ 29 วัน

MSM 6264B เป็นไอซี REAL TIME CLOCK/CALENDAR ชนิด CMOS ใช้ต่อกับบัสของ MICROPROCESSOR/MICROCOMPUTER ได้โดยตรง มี ADDRESS BUS และ DATA ขนาด 4 บิท มี CONTROL REGISTER ขนาด 4 บิท 3 ตัว คือ CD,CE,CF

MSM 6264B ทำงานที่ 5 V +10% ที่ -30 ถึง 25 องศา C มี PACKAGE3 แบบ คือ 18 PIN PLASTIC DIP , 24 PIN PLASTIC FLAT PACKAGE และแบบ12 PIN PLCC PACKAGE การจัดขาต่างๆ แสดงในรูปที่ 1



18 pin Plastic DIP Package



24 pin Plastic Flat Package

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

A0-A3:	Address input
D0-D3:	Data input/output
CS0,CS1	CHIP SELECTS 0,1
RD:	READ enable
WR:	WRITE enable
ALE:	Address latch enable
STD.P:	Standard pulse output
XT,XT:	XTAL oscillator input/output
VDD:	+5V supply
GND:	ground

รูปที่ 1 การจัดขาของ IC MSM 6264B

การจัดขาและหน้าที่ต่างๆ

MSM 6242B ได้ถูกออกแบบมาให้อินเตอร์เฟสเข้ากับ CPU Z80ได้ สำหรับหน้าที่ของขาต่างๆ มีดังนี้

D0-D3 (DATA BUS)เป็นบัสข้อมูลอินพุท/เอาท์พุท สามารถต่อเข้ากับบัสของ MICROCONTROLLER ได้โดยตรง โดย D0=LSB , D3=MSB

A0-A3 (ADDRESS BUS) เป็นบัสแอดแดรส สำหรับติดต่อกับรีจิสเตอร์ภายในของ RTC ตำแหน่งของรีจิสเตอร์ดูได้ที่ตารางที่ 1

ALE (ADDRESS LATCH ENABLE) เมื่อ CS0=0 และ ALE เปลี่ยนจาก "1" ไปเป็นลอจิก "0"แอดแดรสจะถูกแลตช์เอาไว้ภายในตัวของRTC

WR (WRITE ENABLE) ใช้เขียนข้อมูลเข้าไปที่รีจิสเตอร์ของ RTC

RD (READ ENABLE) ใช้อ่านข้อมูลจากรีจิสเตอร์ของ RTC

CS0,CS1 (CHIP SELECTS 0,1) เป็น CHIP SELECT ทำหน้าที่ ENABLE/

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

DISABLE การทำงานของ ALE,RD และ WR

STD.P (STANDARD PULSE OUTPUT) เป็นขาเอาท์พุท ชนิด N-CH OPEN DRAIN ใช้ต่อเข้ากับขา INTERRUPT ของ CPU

XT,XT XTAL OSCILATOR INPUT/OUTPUT) ต่อเข้ากับตัวคริสตอล 32.768 KHZ

VDD เป็นขา POWER SUPPLY +2 - +6 V

GND เป็น GROUND

ตารางที่ 1 REGISTER

Address Input	Address Input				Register Name	Data				Count value	Discription
	A_3	A_2	A_1	A_0		D_3	D_2	D_1	D_0		
0	0	0	0	0	S_1	S_8	S_4	S_2	S_1	0 ~ 9	1-second digit register
1	0	0	0	1	S_{10}	X	S_{40}	S_{20}	S_{10}	0 ~ 5	10-seccod digit register
2	0	0	1	0	MI_1	mi_8	mi_4	mi_2	mi_1	0 ~ 9	1-minute digit register
3	0	0	1	1	MI_{10}	X	mi_{40}	mi_{20}	mi_{10}	0 ~ 5	10-minute digit register
4	0	1	0	0	H_1	h_8	h_4	h_2	h_1	0 ~ 9	1-hour digit register
5	0	1	0	1	H_{10}	X	PM/AM	h_{20}	h_{10}	0 ~ 2 or 0~1	PM/AM, 10-hour digit register
6	0	1	1	0	D_1	d_8	d_4	d_2	d_1	0 ~ 9	1-day digit register
7	0	1	1	1	D_{10}	X	X	d_{20}	d_{10}	0 ~ 3	10-dey digit register
8	1	0	0	0	MO_1	MO_8	MO_4	MO_2	MO_1	0 ~ 9	1-month digit register
9	1	0	0	1	MO_{10}	X	X	X	MO_{10}	0 ~ 1	10-month digit register
A	1	0	1	0	Y_1	y_8	y_4	y_2	y_1	0 ~ 9	1-year digit register
B	1	0	1	1	Y_{10}	y_{80}	y_{40}	y_{20}	y_{10}	0 ~ 9	10-year digit register
C	1	1	0	0	W	X	W_4	W_2	W_1	0 ~ 6	Week register

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

D	1	1	0	1	C_D	30sec. ADJ	IRQ	BUSY	HOLD	-	Control register D
E	1	1	1	0	C_E	t_1	t_0	ITRPT /STND	MASK	-	Control register E
F	1	1	1	1	C_F	TEST	24/12	STOP	REST	-	Control register F

REST = RESET

ITRPT/STND = INTERRUPT/STANDARD

Note 1) - Bit x does not exit (unrecognized during a write and held at "0" during a read).

Note 2) - Be sure to mask the AM/PM bit when processing 10's of hour's data.

Note 3) - BUSY bit is read only. The IRQ FLAG bit can only be set to a "0". Setting the IRQ FLAG to a "1" is d

REGISTER ต่างๆ

S1,S10,MI1,MI10,H1,H10,D1,D10,MO1,MO10,Y1,Y10,W

กลุ่มอักษรเหล่านี้เป็นชื่อย่อของรีจิสเตอร์ตามลำดับ คือ SECOND1,SECOND10, MINUTE1,MINUTE10,HOUR1,HOUR10,DAY1,DAY10,MONTH1,MONTH10,YEAR1,YEAR10 และ WEEK ในการกำหนดค่าให้รีจิสเตอร์เหล่านี้จะต้องให้เป็นระหัส BCD เช่น S1(S8,S4,S2,S1)=1001 ซึ่งจะหมายถึง 9 วินาที

PM/AM,h20,h10 ในโหมด 24 ชม. บิท PM/AM จะไม่ใช้และบิทนี้จะอ่านได้เป็น "0" ตลอด ในขณะที่อยู่ในโหมด 12 ชม. บิท h20 จะถูกเซตในการอ่าน

รีจิสเตอร์ W สามารถมีข้อมูลได้ตั้งแต่ 0-6 ดังตารางที่ 2

ตารางที่ 2 REGISTER W

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

W4	W2	W1	DAY OF WEEK
0	0	0	SUNDAY
0	0	1	MONDAY
0	1	0	TUESDAY
0	1	1	WEDNESDAY
1	0	0	THURSDAY
1	0	1	FRIDAY
1	1	0	SATURDAY

CD REGISTER (CONTROL D REGISTER)

เซทการอ่านและเขียนรีจิสเตอร์ S1-W

CE REGISTER (CONTROL E REGISTER)

ควบคุมเอาท์พุทของ STD.P

CF REGISTER (CONTROL F REGISTER)

เซทค่า CLOCK

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

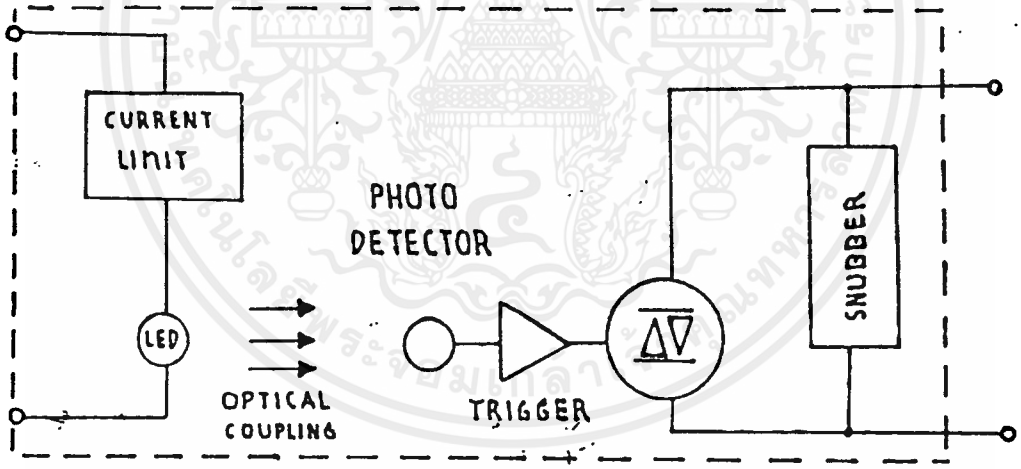
2.4 SOLID STATE RELAY

SOLID STATE RELAY หรือเรียกสั้นๆ ได้ว่า SSR ได้เข้ามาใช้งานในการเป็นอุปกรณ์ติดต่อทางไฟฟ้า โดยเฉพาะแล้วในระบบไฟฟ้าที่ใช้การทำงานร่วมกับ MICROCOMPUTER แล้ว จะเข้ามาใช้แทน RELAY ธรรมดาๆ โดยใช้หลักการทำงานของอุปกรณ์ที่เป็นเซมิคอนดัคเตอร์แทนระบบการเคลื่อนไหวทางกลในการตัดต่อวงจร

การทำงานของ SSR

แบ่งการทำงานของ SSR ได้เป็น 2 ภาคใหญ่ๆ คือ

1. ภาค INPUT
2. ภาค OUTPUT



AC SOLID-STATE RELAY (SSR)

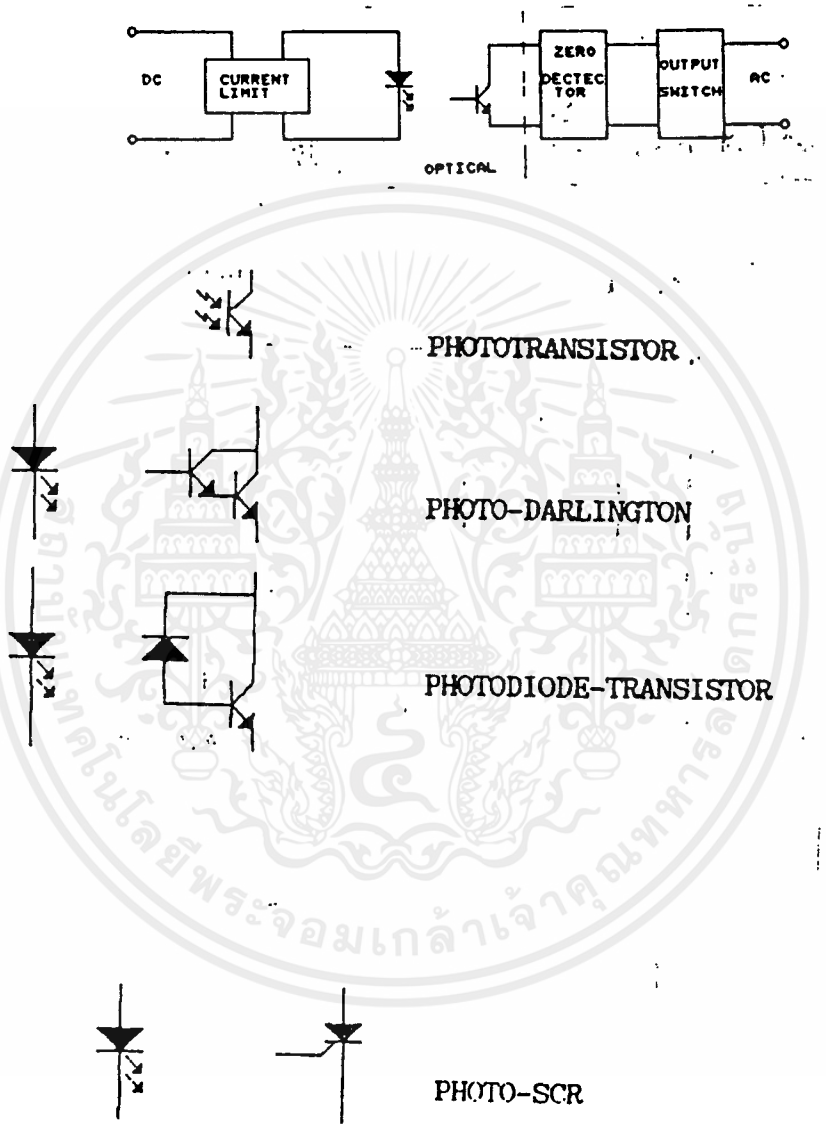
รูปที่ 1 วงจร SSR-AC

1. ภาค INPUT จะเป็นภาครับสัญญาณจากการควบคุมมาทำงาน โดยเป็นการ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การส่งผ่านสัญญาณไปควบคุมอุปกรณ์ติอต่อ ซึ่งจะแยกกันในทางไฟฟ้า โดยที่มีใช้กันอยู่พอแบ่งออกได้เป็น 2 ลักษณะ คือ

1.1 ใช้อุปกรณ์ทาง OPTICAL(PHOTO)



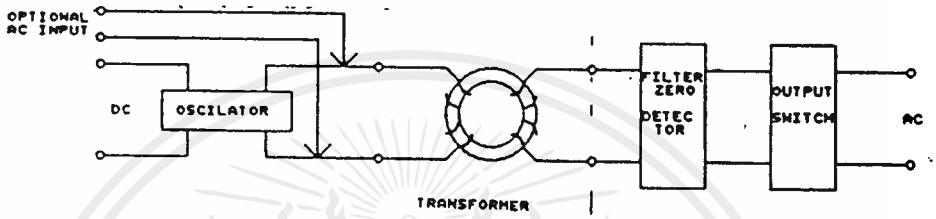
รูปที่ 2 อุปกรณ์ทาง OPTICAL และการต่อใช้งาน

เช่น อาจจะใช้ PHOTO TRANSISTOR ,PHOTO-DARLINGTON,PHOTODIODE,

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

PHOTO-SCR หรือ PHOTO RESISTOR เป็นต้น ตัวอุปกรณ์จะเปลี่ยนสัญญาณไฟฟ้าที่ส่งมาให้อยู่ในรูปของแสงไปควบคุมส่วนOUTPUTอีกทีหนึ่ง ซึ่งจะทำให้เป็นการตัดขาดระหว่าง INPUT และ OUTPUT ทางไฟฟ้าอย่างแท้จริง

1.2 ใช้อุปกรณ์ TRANSFORMER



รูปที่ 3 TRANSFORMER ในวงจร SSR-AC

โดยจะเป็น FERRITE-CORED TOROIDAL TRANSFERMER โดยจะทำงานร่วมกันระหว่าง INPUT และ OSCILLATOR โดยมีการ OSCILLATOR ความถี่สูง 50 KHZ-10 MHZ ในการส่งสัญญาณจาก INPUT ไปยัง OUTPUT

2. ภาค OUTPUT เป็นส่วนรับสัญญาณจาค INPUT ซึ่งอาจจะอยู่ในรูปของแสงมาทำงาน โดยต่อวงจรหรือตัดวงจร โดยอุปกรณ์นี้อาจจะเป็น TRANSISTOR,SCR หรือ TRIAC ก็ได้

ZERO SWITCHING

อีกส่วนหรือที่มีความสำคัญสำหรับการทำงานของ SSR ที่ใช้กับ OUTPUT ที่เป็นAC โดยปกติการทำงานของ SW ธรรมดาต่างๆอาจจะมีการทำงานที่ไม่สมควรเกิดขึ้น คือเปิดวงจรในขณะที่สัญญาณของไฟ AC อาจจะอยู่ในค่าสูงสุด ซึ่งทำให้เกิดกระแสและสัญญาณรบกวนที่สูงมาก โดยเฉพาะถ้าต่อรวมอยู่กับวงจรทาง COMPUTER แล้ว ก็อาจทำให้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เครื่องเกิดอาการเสียหรือทำงานผิดพลาดได้ ฉนั้น ZERO SWITCHING ก็คือเป็นการทำงานให้เปิดวงจรในขณะที่สัญญาณ AC อยู่ในระยะ 0 V แต่ในการใช้งานจริงจะมีค่ามากกว่า0 V เล็กน้อย เพื่อให้อุปกรณ์เซมิคอนดัคเตอร์ที่ต่ออยู่สามารถทำงานเปิดวงจรได้ เช่น ตัวอย่างเราต้องการให้วงจรทำงานเปิดวงจนที่ 15 V จาก 220 V ก็จะเป็นมุมทางสัญญาณ AC 2.77 องศา ตามสูตร

$$\phi = \text{SIN} \frac{\text{Vsw.max}}{\text{Line Vrms}\sqrt{2}}$$

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

$$= 2.77$$

หรือในทางกลับกัน เราอาจคำนวณระยะเวลาหลังจาก 0 V ที่เกิดขึ้นแล้วนานเท่าไรจึงจะเปิดวงจร

$$T = \frac{1/2 \text{ cyc.ms}}{1/2 \text{ cyc.deg}} * \phi$$

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

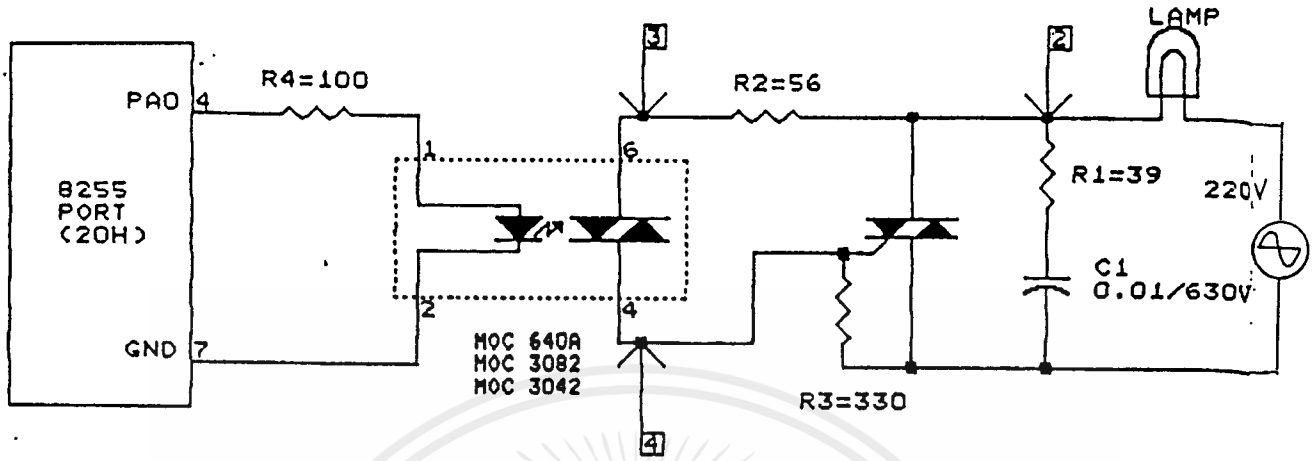
$$= 0.153 \text{ ms}$$

ตัวอย่างวงจร SSR OUTPUT AC 220 V

เราสามารถสั่งทางโปรแกรมให้ SSR ทำงานได้โดยส่ง OUTPUT จาก PORT 8255โดยให้ LOGIC "0" SSR จะทำงาน ให้ SSR ไม่ทำงานโดยส่ง LOGIC "1" ออกไปที่ PORT

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

038351



รูปที่ 4 วงจร SSR-AC

โปรแกรม

```
        ORG   2000H
DELAY   EQU   0FH
SYSCAL  EQU   10H
START    LD   A,80H
        OUT   (23H)
START1   LD   A,0
        OUT   (20H),A
         LD   A,DELAY
        RST   SYSCAL
         LD   A,0FFH
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
OUT     (20H),A
 LD     A,DELAY
RST     SYSCAL
 JR     START1
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 3
ส่วนประกอบของแผ่งแสดงผล

3.1 CONTROL BOARD Z-80

CONTROL BOARD Z-80 เป็นบอร์ดไมโครคอมพิวเตอร์ที่ใช้ในงานควบคุมมีคุณสมบัติดังนี้

-CPU Z84C00-6 (6 MHZ)

-REAL TIME CLOCK 6242B

-POWER MONITER & SYSTER RESET

-EEPROM 64*16 BIT

-96 BIT I/O (8255*4)

-EPROM 32K

-RAM 8 KB

-XTAL 4 MHZ

-KEYBOARD CONNECTOR (4*4)

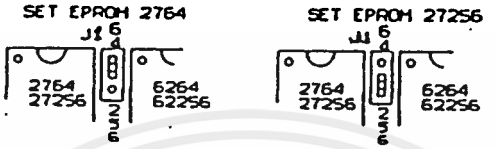
รายละเอียดและการใช้งาน

CPU ใช้ไอซี Z84C00-6 ของบริษัท ZILOG ซึ่งเป็น CMOS ใช้กำลังทำงานต่ำ ทำงานที่ความเร็วสูงสุดของ XTAL ได้ 6 MHZ สำหรับ CONTROL BOARD Z-80จะใช้ XTAL 4 MHZ เนื่องจากอุปการณ์จำพวก EPROM ACCESS TIME TIME จะช้ากว่า CPU มาก ซึ่งอาจทำงานไม่ทัน

หน่วยความจำ

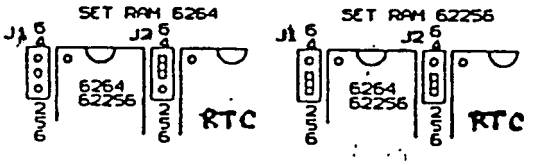
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1. ROM บนบอร์ดควบคุม ใช้ IC เบอร์ 27256 มีความจุ 32 K ใช้ใส่ตัวโปรแกรมที่พัฒนาขึ้น สามารถเลือกใส่หน่วยความจำขนาด 8 K โดยการกำหนดจาก JUMPER J1 ดังรูปที่ 1



รูปที่ 1 JP1 กำหนดหน่วยความจำ U3

2. RAM บนบอร์อควบคุม ใช้ IC เบอร์ 62256 มีความจุ 32 K มีตำแหน่งแอดแดรสที่ 8000H-FFFFH หน่วยความจำ RAM ยังมีวงจร BACK UP ข้อมูล โดยการใส่BATTERY ซึ่งจะเก็บข้อมูลไว้ได้นาน โดยใส่ JUMPER "JBAT" และสามารถเลือกหน่วยความจำเป็นขนาด 8 K โดยการเซท JUMPER JP2 ดังรูปที่ 2



รูปที่ 2 JP2 เลือกเบอร์หน่วยความจำ U4

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

พอร์ต

CONTROL BOARD Z-80 มี IC 8255 จำนวน 4 ตัว โดยสัญญาณ I/O จะถูกต่อไว้ที่ CONNECTOR 34 PIN จำนวน 4 ชุด นอกจากนี้ยังมีขั้ว 10 PIN สำหรับ KEYBOARD ซึ่งต่อมาจาก PORT C ของ 8255 U8

POWER SUPPLY

CONTROL BOARD Z-80 ต้องการแหล่งจ่ายไฟ 5 VDC จ่ายเข้าทางTERMINAL 2 ขา ในกรณีจ่ายไฟเกิน 5 VDC จะมี ZENER ขนาด 5.6 V ไว้ป้องกัน กรณีจ่ายไฟกลับขั้ว จะมี DIODE 1N4001 ต่อในทาง FORWARD BIAS(ขณะต่อกลับขั้ว) จะทำให้กระแสส่วนใหญ่ไหลผ่าน DIODE และ ZENER DIODE

การติดต่อกับ RTC

-RTC จะใช้ CHIP เบอร์ MSM6242B
-ตำแหน่งแอดแดรสของ RTC จะเป็น ดังนี้

ตารางที่ 1 ตำแหน่ง ADDRESS ของ RTC

ตำแหน่ง	JP3	
หลักหน่วยของวินาที	40H	C0H
หลักสิบของวินาที	41H	C1H
หลักหน่วยของนาที	42H	C2H
หลักสิบของนาที	43H	C3H
หลักหน่วยของชั่วโมง	44H	C4H
หลักสิบของชั่วโมง/AM,PM	45H	C5H
หลักหน่วยของวัน	46H	C6H
หลักสิบของวัน	47H	C7H

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หลักหน่วยของเดือน	48H	C8H
หลักสิบของเดือน	49H	C9H
หลักหน่วยของปี	4AH	CAH
หลักสิบของปี	4BH	CBH
วันในสัปดาห์	4CH	CCH
รีจิสเตอร์ D	4DH	CDH
รีจิสเตอร์ E	4EH	CEH
รีจิสเตอร์ F	4FH	CFH

เนื่องจาก RTC เป็น CHIP ขนาด 4 BIT จึงต้องมีการ ACCESS ข้อมูล 2 ครั้ง (หลักหน่วย/สิบ)

ตัวอย่าง ต้องการอ่านเวลาในหลักวินาที โปรมแกรมจะเป็น ดังนี้

```
IN A,(40H)
AND A,0FH
LD B,A
IN A,(41H)
SLA A          ;เลื่อนข้อมูลไปทางซ้าย 4 บิท
SLA A
SLA A
SLA A
OR A,B         ;ได้ค่าวินาทีเป็น BCD
```

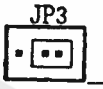
JUMPER JP6 สำหรับผู้ที่ต้องการให้ RTC INTERRUPT CPU ตามเวลาที่โปรแกรมไว้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

คีย์บอร์ด

ใช้ 8255(U8) PORT C เป็นคีย์บอร์ด โดยอยู่ที่ ตำแหน่งแอดแอรส ดังนี้

ตารางที่ 2 ตำแหน่ง JP3

คีย์	JP3	JP3 
คีย์อินพุท (ROW)	02	82H
คีย์เอ้าท์พุท (COLUMN)	02	82H

เนื่องจาก PORT C สามารถโปรแกรมให้เป็น INPUT หรือ OUTPUT ทั้ง PC-HI(PC4 -PC7) หรือ PC-LO(PC0-PC3) ขึ้นอยู่กับผู้ใช้เป็นผู้กำหนด

EEPROM

การติดต่อกับ EEPROM เบอร์ 93C46 ซึ่งเป็น SERIAL I/O โดยมีสัญญาณสำคัญอยู่ 4 ขา คือ CS,DI,DO,SK

-CS,DI,SK เป็นขา INPUT ซึ่งติดต่อกับ PORT C ของ U11 ฉนั้นพอร์ต PC0-PC2 ต้องเซทให้เป็น OUTPUT

-DO เป็นขา OUTPUT ต่อกับ PC4 ซึ่งต้องเซทให้เป็น INPUT

แอดแดรสที่ใช้ติดต่อ EEPROM

ตารางที่ 3 ADDRESS ที่ใช้ติดต่อ EEPROM

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

JP3

หน้าที่		
CS =PC1 (CHIP SELECT,O/P)	32H	B2H
SK =PC2 (SERIAL CLOCK O/P)	32H	B2H
DI =PC2 (DATA IN; O/P)	32H	B2H
DO =PC4 (DATA OUT I/O)	32H	B2H

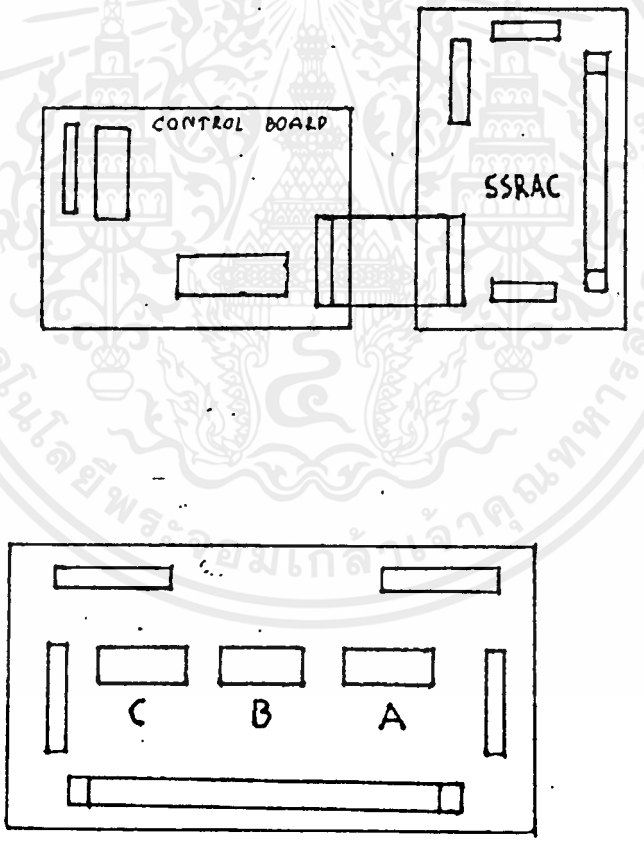
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.2 SSR-AC

SSR-AC

SSR-ACเป็นชุด BOARD ขยายระบบไมโครของ CONTROL BOARD Z-80 ขยายการทำงานเข้าสู่ระบบไฟ AC 220 V. โดยสามารถควบคุมได้ BOARD ละ 8 ช่องสัญญาณ และตัว BOARD สามรถนำมาต่อพ่วงกัน ขยายได้เป็น 24 ช่องสัญญาณได้

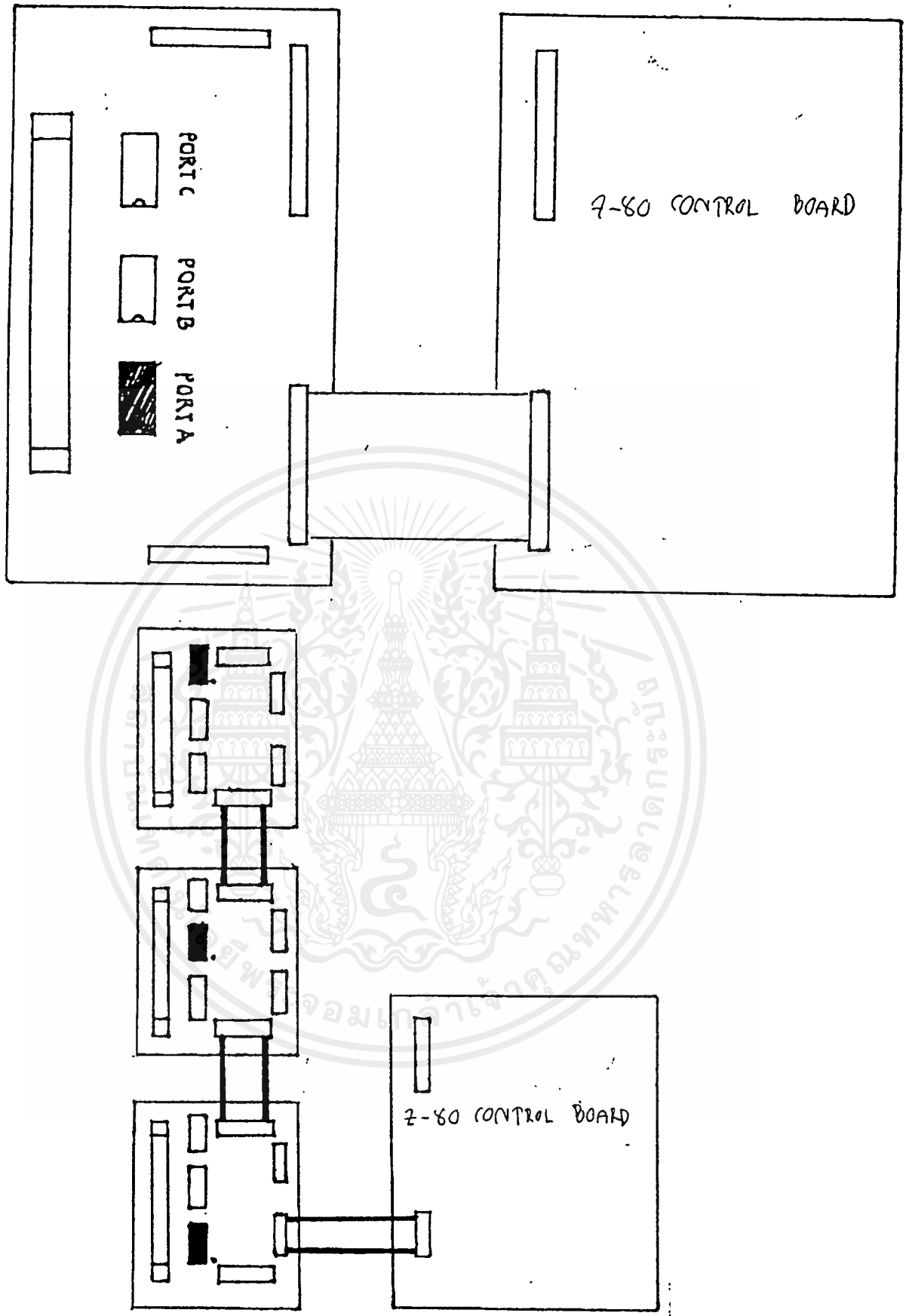
SSR-AC สามารถต่อรับสัญญาณจาก PORT 8255 บน CONTROL BOARD Z-80 และสามารถต่อพ่วงกัน 3 BOARD ได้ ดังรูป



รูปที่ 1 ก. การต่อ SSR-AC กับ Z-80 CONTROL BOARD

ข. ด้านหน่ง 74LS245 PORT A,B,และ C

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2 ก. ต่อแบบ 1 BOARD

ข. ต่อแบบขยายเต็ม 3 BOARD

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การทำงานของ SSR-AC

อุปกรณ์ที่เหมาะสมในการตัดต่อระบบไฟ AC 220 V. ก็คือ อุปกรณ์ทาง SOLID-STATE RELAY หรือ SSR ซึ่งสามารถนำมาใช้แทนระบบการเคลื่อนไหวทางกลในการตัดต่อวงจร

ข้อดีของ SSR

1. การทำงานจะต่อวงจรที่ VOLTAGE ขณะเป็นศูนย์หรือใกล้จุดศูนย์
2. มีอายุการทำงานที่ยาวนานมาก
3. ไม่มีเสียงรบกวนเกิดขึ้นในเวลาทำงาน
4. สามารถต่อรวมกับระบบไมโครโปรเซสเซอร์ได้โดยง่าย
5. สามารถตัดต่อวงจรได้อย่างรวดเร็ว
6. ไม่มีส่วนเคลื่อนไหวทางกลในการทำงาน
7. ไม่เกิดอาการ BOUNCE ที่หน้าสัมผัส ซึ่งเป็นตัวก่อให้เกิดสัญญาณรบกวน
8. ไม่เป็นตัวก่อให้เกิดสัญญาณรบกวนต่อระบบไฟฟ้าที่ตัวมันต่ออยู่

ข้อเสียของ SSR

1. ไม่สามารถใช้กับวงจรที่มี VOLTAGE ที่สูงมากๆได้
2. ไม่สามารถใช้ SSR สำหรับ AC LINE หรือ DC LINE ได้ในตัวเดียวกัน
3. เวลาใช้งานจะเกิดความร้อนขึ้นที่ตัว SSR นั้นๆ จำเป็นต้องติดตั้ง HEAT SING ช่วยในการทำงาน
4. ตัว SSR จะเสียโดยง่ายดาย ถ้าเกิดการลัดวงจรขึ้น จึงควรระมัดระวัง

การทำงานของ SSR

แบ่งได้เป็น 2 ส่วนใหญ่ๆ คือ

1. ส่วน INPUT
2. ส่วน OUTPUT

1. ภาค INPUT จะเป็นตัวรับสัญญาณจาก PORT 8255 ซึ่งจะส่งมาให้ IC 74LS245 ซึ่งเป็น BUFFER และมาขับให้ TR BC557 ทำงานให้ IC OPTICAL(PHOTO)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

MOC3082 เปลี่ยนสัญญาณจากไฟฟ้าให้เป็นสัญญาณแสงส่งไปภาค OUTPUT อีกทีหนึ่ง ซึ่งจะทำให้เป็นการตัดขาดระหว่าง INPUT และ OUTPUT จริงๆ

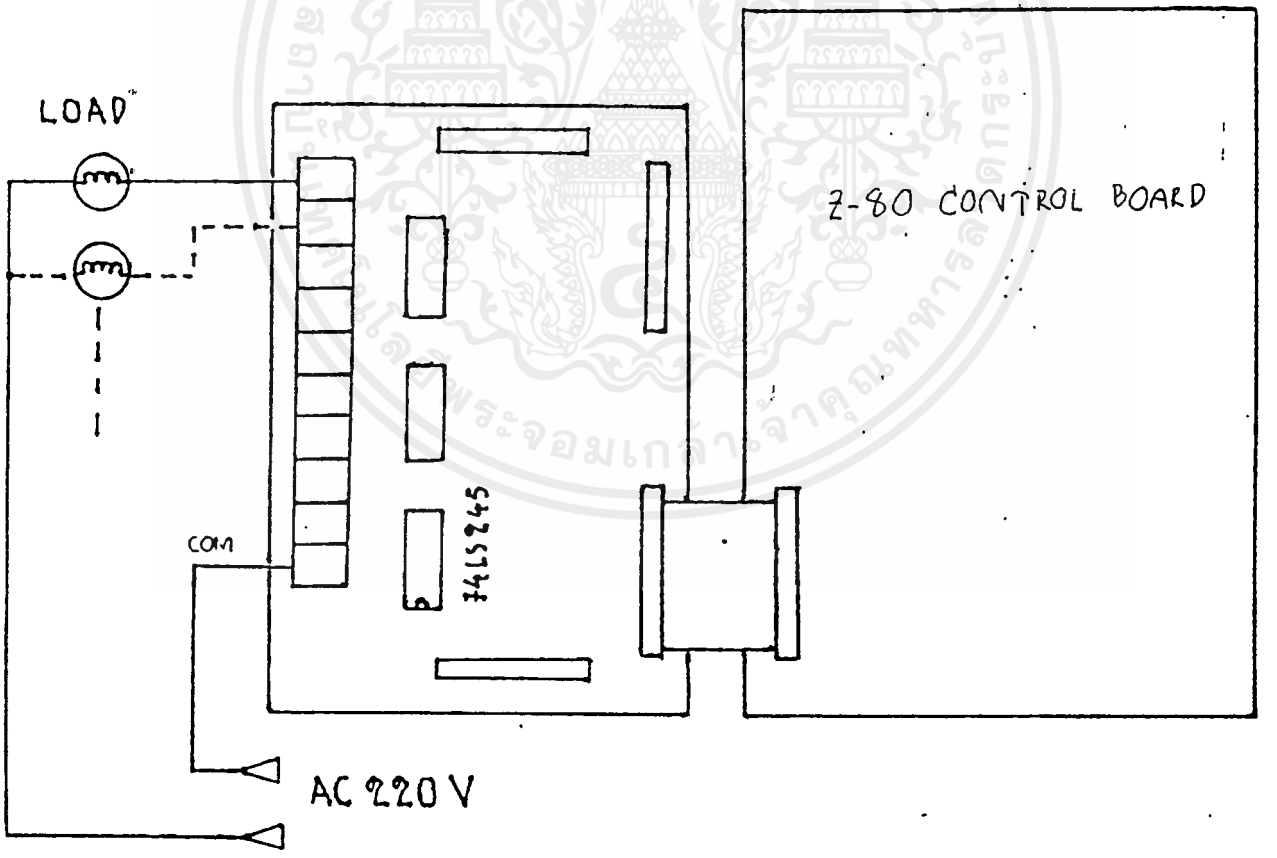
2. ภาค OUTPUT จะรับสญญาณจาก IC MOC3082 ซึ่งจะเป็นการทำงานในลักษณะ ZERO SWITCH โดยจะต่อวงจรขับ TRIAC ก็ต่อเมื่อ ไฟ AC LINE มีค่าใกล้เคียง 0 โวลท์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.3 การเชื่อมต่ออุปกรณ์

นำ BOARD SSR-AC ต่อรวมกับ Z-80 CONTROL BOARD ทางด้านส่วนขยาย 8255 ดังรูป โดยต่อ LOAD เป็นหลอดไฟ จะใช้ขนาด AC 220 V. 100W หรือน้อยกว่าก็ได้

จะเห็นว่า BOARD SSR-AC จะมี SOCKET ให้ใส่ IC 74LS245 ได้ 3 ตัว จากรูปใส่ IC74LS245 เพียงตัวเท่านั้นเพื่อเลือกให้ BOARD SSR-AC นี้ทำงานร่วมกับ 8255 เพียง 1 PORT เช่น ถ้าเราต้องการให้ SSR-AC นี้ติดต่อกับ8255 PORT A ก็ให้ใส่ IC 74LS245 ที่ SOCKET IC ที่เขียนว่า PORT A เป็นต้น โดยวิธีนี้เราสามารถใช้ 8255 1 ตัว ต่อร่วมกับ SSR-AC ได้ 3 BOARDโดยสลับตำแหน่ง IC74LS245 ระหว่าง BOARD ไม่ให้ตรงกันได้ ดังรูป



รูปที่ 1 ต่อหลอดไฟเลือกทำงานที่ PORT A

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การสั่งให้ BOARD SSR-AC ทำงาน ON/OFF ได้โดยการส่ง OUTPUT ออกจาก PORT 8255 โดย

LOGIT 1 = SSR-AC "OFF"

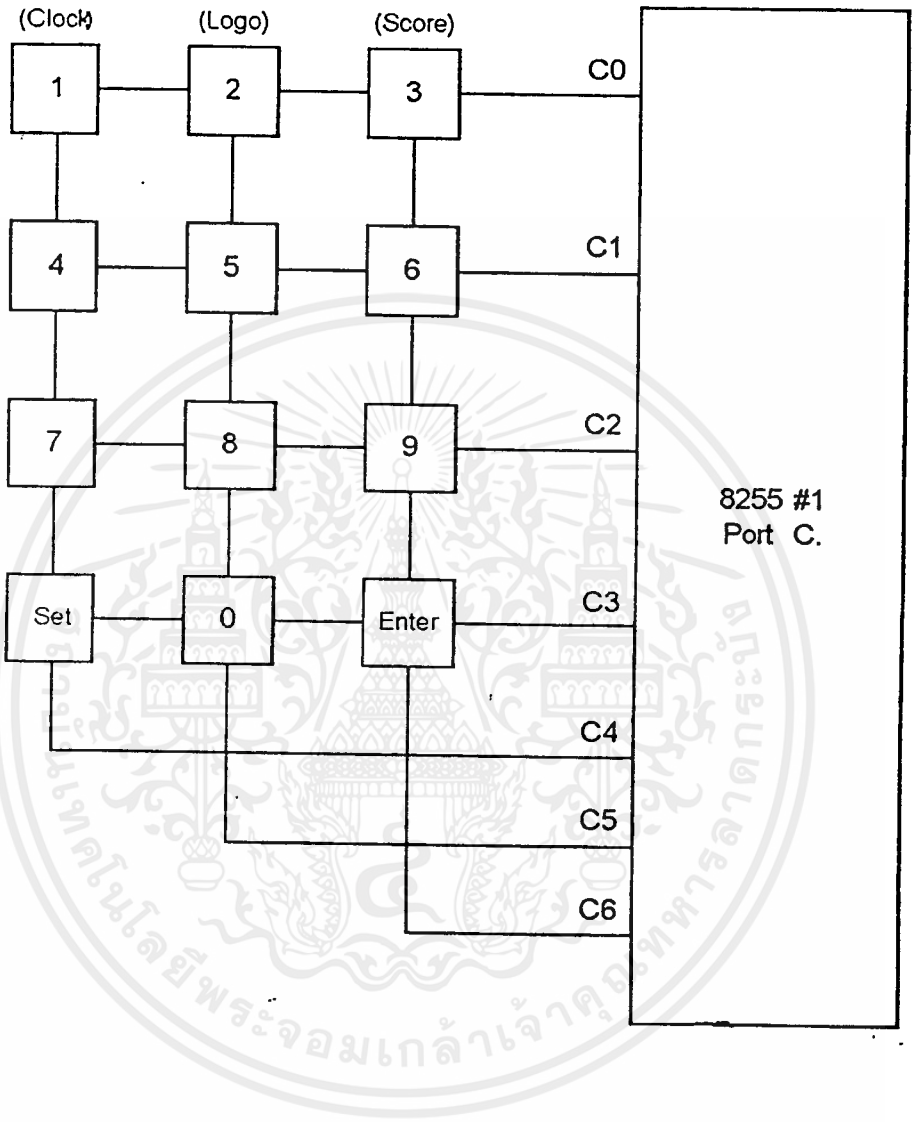
LOGIT 2 = SSR-AC "ON"

ตัวอย่างโปรแกรม

```
LD A,80H        ; SET 8255 เป็น OUTPUT PORT
OUT (23H),A     ; ส่ง CONTROL PORT
LD A,0FFH       ; ส่ง CLEAR PORT OFF OUTPUT
OUT (20H),A     ; PORT A
LD A,0FEH       ; เลือกให้ OUTPUT "1" ON
OUT (20H),A     ; PORT A
RST 18          ; หยุดโปรแกรม
```

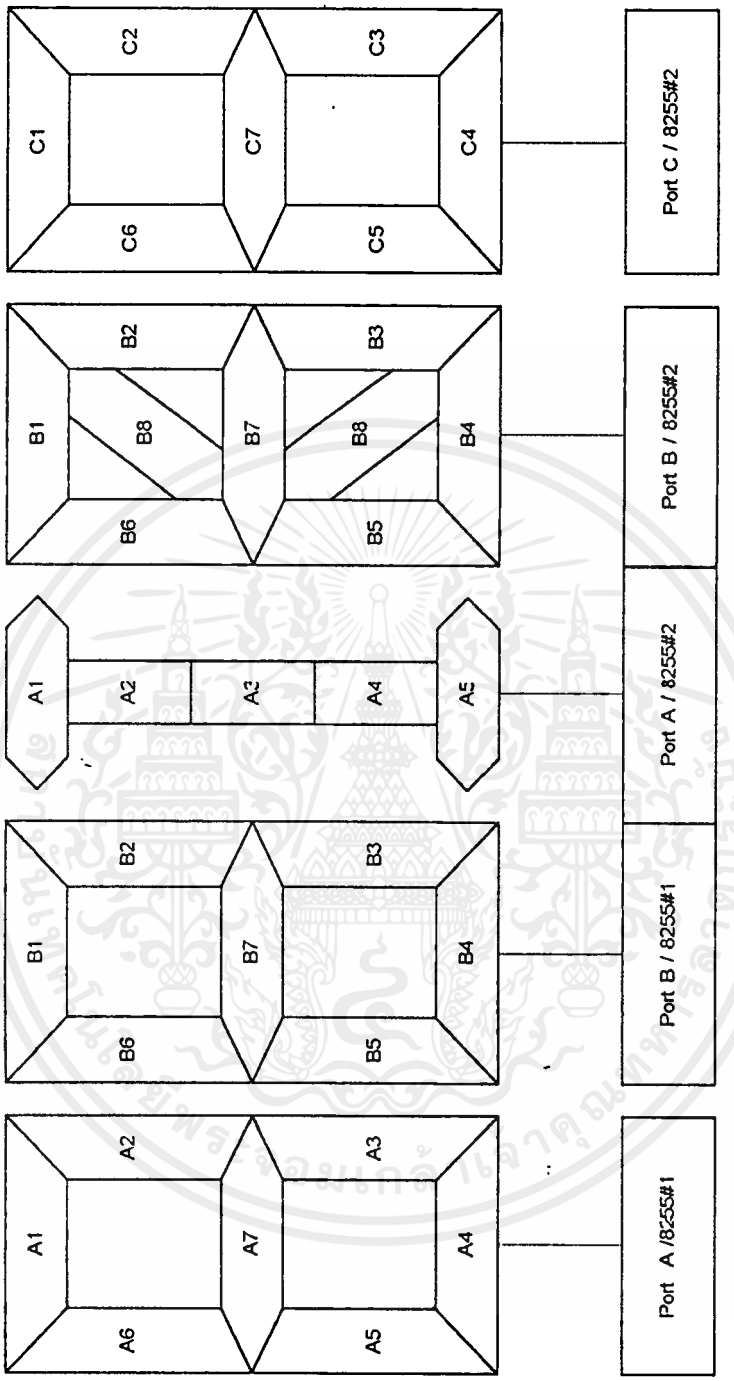
จากโปรแกรม หลอดไฟที่ต่ออยู่ จะติดสว่างขึ้น ถ้าต้องการ OFF ก็ให้ส่งข้อมูลเป็น "1" ออกไป หรือ OFF ได้โดยการ KEY RESET ก็ได้ เพราะจะเป็นการ RESET PORT 8255 ด้วย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2 การต่อ SWITCH เข้ากับ IC 8255 ตัวที่ 1

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3 แสดงการต่อ IC 8255 เข้ากับส่วนแสดงผล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.4 การใช้งานแผงแสดงผล

การเปิดเครื่อง

1. ON SWITCH POWER 220 V.
2. ON SWITCH CONTROL BOARD
3. เมื่อเริ่มเปิดเครื่อง แผงแสดงผลจะแสดง "SEIKO"
4. ต้องการแสดงเป็น SCORE BOARD กด "3"
5. แสดงผล "SEIKO" กด "2"
6. ต้องการแสดงผลเป็นพยางค์ กด "1"
7. แสดงผลตัวเลข กด 0-9
8. ตั้งเวลา กด "*" แล้วตามด้วยตัวเลข แล้วกด "=" 2 หลักหลังใช้วิธีเดียวกัน

การปิดเครื่อง

1. OFF SWITCH POWER 220 V.
2. OFF SWITCH CONTROL

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 4
การทดลองและผลการทดลอง

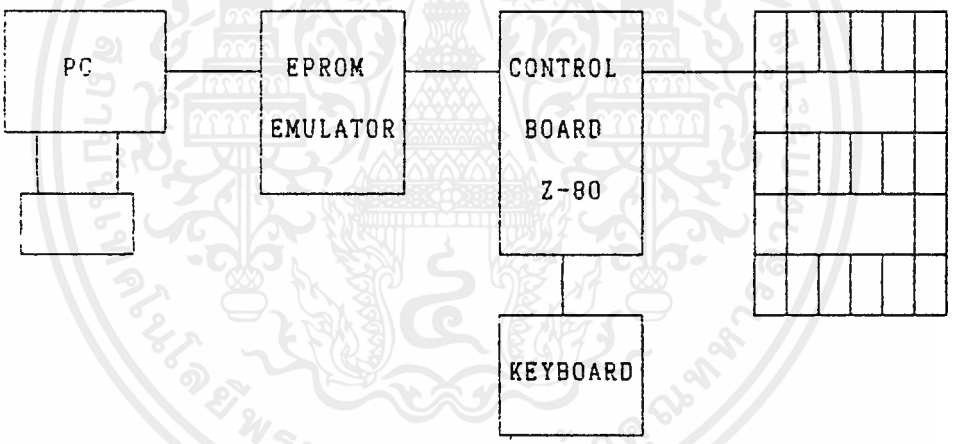
ในบทนี้จะกล่าวถึงการทดลองการทำงานของแผงแสดงผล ว่าสามารถที่จะทำงานได้ตามโปรแกรมที่เราเขียนขึ้นหรือไม่ มีข้อผิดพลาดจากการทำงานอย่างไร

การทดลอง

การทดสอบให้แสดงผลเป็นตัวเลขและตัว LOGO "SEIKO"

ขั้นตอนการทดลอง

1. ต่อวงจรดังรูป



รูปที่ 1 การต่อแผงแสดงผลเพื่อทดลอง

2. เขียนโปรแกรมผ่านทางเครื่องคอมพิเตอร์ PC
3. นำโปรแกรมที่เขียนไว้โหลดลงไปใน EPROM EMULATOR
4. ทดลองกดคีย์บอร์ดที่ตัว CONTROL BOARD Z-80 เพื่อทดสอบผลการทำงาน

โปรแกรมที่ใช้ในการทดลอง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
0000                    CPU     "Z80.TBL"       :USE Z80 TABLE
0000                    HOF     "INT8"          :HEX Output formate

0080 =          PORT1   EQU     80H
0090 =          PORT2   EQU     90H
00A0 =          PORT3   EQU     0A0H
00B0 =          PORT4   EQU     0B0H
00C0 =          RTC     EQU     0C0H

0000                    ORG     0000H

0000            POWER ON:
0000 AF                 XOR     A
0001 3D         LOOP:   DEC     A
0002 20FD               JR      NZ,LOOP
0004 31009F             LD      SP,9F00H

0007            SET8255:
                ;****************************************
                ;Set 8255 #1  PORT A= OUT , PORT B = OUT
                ;PORT C(UP) = OUT , PORT C(LOW) = IN
                ;****************************************
0007 3E88               LD      A,88H
0009 D383               OUT     (PORT1+3),A
                ;****************************************
                ;Set 8255 #2,3,4 PORT A,B,C = OUT
                ;****************************************
000B 3E80               LD      A,80H
000D D393               OUT     (PORT2+3),A
000F D3A3               OUT     (PORT3+3),A
0011 D3B3               OUT     (PORT4+3),A

                ;****************************************
                ;Clera PORT Control SSR. (OFF LIGHT)
                ;****************************************
0013 3EFF               LD      A,0FFH
0015 D390               OUT     (PORT2),A
0017 D391               OUT     (PORT2+1),A
0019 D392               OUT     (PORT2+2),A
001B D3A0               OUT     (PORT3),A
001D D3A1               OUT     (PORT3+1),A
001F D3A2               OUT     (PORT3+2),A
0021 D3B0               OUT     (PORT4),A
0023 D3B1               OUT     (PORT4+1),A
0025 D3B2               OUT     (PORT4+2),A
                ;****************************************
                ;******CLEAR BUFFER DISPLAY**************
                ;****************************************
0027 0605               LD      B,05H
0029 210080             LD      HL,8000H
002C 3EFF               LD      A,0FFH
002E            LOOP2:
002E 77                 LD      (HL),A
002F 23                 INC     HL
0030 10FC               DJNZ    LOOP2
                ;****************************************
                ;************SCAN SWITCH*****************
                ;****************************************
0032            SW_SCAN:
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งไม่อนุญาตให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของผลงานทุกครั้งที่มีการนำไปใช้

```
0032 0604              LD      B,04H           ;SET COUNTER ROW SW.
0034 0E01              LD      C,01H
0036            LOOP_SW1:
0036 79                LD      A,C
0037 D382              OUT     (PORT1+2),A     ;OUT ROW SW.
0039 DB82              IN      A,(PORT1+2)     ;INPUT SW. DATA
003B E670              AND     70H             ;Clrae INPUT ROW SW.
003D FE00              CP      00H             ;CKD. DATA INPUT
003F C44C00            CALL    NZ,INDATA       ;Have DATA CALL INDATA
0042 CB01              RLC     C               ;Shift ROW SW. Active
0044 10F0              DJNZ    LOOP_SW1

0046 CD7D01            CALL    DISPLAY
0049 C33200            JP      SW_SCAN


004C            INDATA:
004C 57                LD      D,A             ;Load DATA TO D.
004D DB82              IN      A,(PORT1+2)     ;CKD. INPUT DATA
004F E670              AND     70H
0051 BA                CP      D
0052 28F8              JR      Z,INDATA        ;KEY ACTIVE LOOP
0054 79                LD      A,C             ;Load A = ROW SW.
0055 FE01              CP      01H             ;CKD  ROW 0
0057 2004              JR      NZ,NO_ROW0
0059 CD7500            CALL    KEY_ROW0        ;CALL A = ROW 0
005C C9                RET
005D            NO_ROW0:
005D FE02              CP      02H             ;CKD ROW 1
005F 2004              JR      NZ,NO_ROW1
0061 CD8E00            CALL    KEY_ROW1        ;CALL A = ROW 1
0064 C9                RET
0065            NO_ROW1:
0065 FE04              CP      04H             ;CKD ROW 2
0067 2004              JR      NZ,NO_ROW2
0069 CDA700            CALL    KEY_ROW2        ;CALL A = ROW 2
006C C9                RET
006D            NO_ROW2:
006D FE08              CP      08H             ;CKD ROW 3
006F 2003              JR      NZ,NO_ROW3
0071 CDC000            CALL    KEY_ROW3        ;CALL A = ROW 3
0074            NO_ROW3:
0074 C9                RET

0075            KEY_ROW0:
0075 7A                LD      A,D             ;Load A = DATA IN
0076 FE40              CP      40H             ;CKD SW 1
0078 2004              JR      NZ,NO0_1        ;JUMP IF NOT SW1
007A CDD900            CALL    SW1             ;SET A = 1
007D C9                RET
007E FE20       NO0_1: CP      20H             ;CKD SW 2
0080 2004              JR      NZ,NO0_2        ;JUMP IF NOT SW2
0082 CDE400            CALL    SW2             ;SET A = 2
0085 C9                RET
0086 FE10       NO0_2: CP      10H             ;CKD SW 3
0088 2003              JR      NZ,NO0_3        ;JUMP IF NOT SW3
008A CDEF00            CALL    SW3             ;SET A = 3
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
008D C9          NO0_3:  RET

008E             KEY_ROW1:
008E 7A                  LD      A,D
008F FE40                CP      40H             ;CKD SW 4
0091 2004                JR      NZ,NO1_1        ;JUMP IF NOT SW4
0093 CDFA00              CALL    SW4             ;SET A = 4
0096 C9                  RET
0097 FE20        NO1_1:  CP      20H             ;CKD SW 5
0099 2004                JR      NZ,NO1_2        ;JUMP IF NOT SW5
009B CD0501              CALL    SW5             ;SET A = 5
009E C9                  RET
009F FE10        NO1_2:  CP      10H             ;CKD SW 6
00A1 2003                JR      NZ,NO1_3        ;JUMP IF NOT SW 6
00A3 CD1001              CALL    SW6             ;SET A =6
00A6 C9          NO1_3:  RET

00A7             KEY_ROW2:
00A7 7A                  LD      A,D
00A8 FE40                CP      40H             ;CKD SW 7
00AA 2004                JR      NZ,NO2_1        ;JUMP IF NOT SW 7
00AC CD1B01              CALL    SW7             ;SET A = 7
00AF C9                  RET
00B0 FE20        NO2_1:  CP      20H             ;CKD SW 8
00B2 2004                JR      NZ,NO2_2        ;JUMP IF NOT SW 8
00B4 CD2601              CALL    SW8             ;SET A = 8
00B7 C9                  RET
00B8 FE10        NO2_2:  CP      10H             ;CKD SW 9
00BA 2003                JR      NZ,NO2_3        ;JUMP IF NOT SW 9
00BC CD3101              CALL    SW9             ;SET A = 9
00BF C9          NO2_3:  RET

00C0             KEY_ROW3:
00C0 7A                  LD      A,D
00C1 FE40                CP      40H             ;CKD SW #
00C3 2004                JR      NZ,NO3_1        ;JUMP IF NOT SW#
00C5 CD4701              CALL    SWA             ;SET A =0A
00C8 C9                  RET
00C9 FE20        NO3_1:  CP      20H             ;CKD SW 0
00CB 2004                JR      NZ,NO3_2        ;JUMP IF NOT SW 0
00CD CD3C01              CALL    SW0             ;SET A = 0
00D0 C9                  RET
00D1 FE10        NO3_2:  CP      10H             ;CKD SW *
00D3 2003                JR      NZ,NO3_3        ;JUMP IF NOT SW *
00D5 CD4B01              CALL    SWB             ;SET A = 0B
00D8 C9          NO3_3:  RET


00D9             SW1:
00D9 DD21A801            LD      IX,NUMBER       ;SET IX = TABLE
00DD DD7E01              LD      A,(IX+1)        ;OPEN TABLE ADDRESS 1
00E0 CD5F01              CALL    SHIFT_DATA
00E3 C9                  RET
00E4             SW2:
00E4 DD21A801            LD      IX,NUMBER
00E8 DD7E02              LD      A,(IX+2)
00EB CD5F01              CALL    SHIFT_DATA
00EE C9                  RET
00EF             SW3:
00EF DD21A801            LD      IX,NUMBER
```

เอกสารนี้สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
00F3 DD7E03             LD      A,(IX+3)
00F6 CD5F01             CALL    SHIFT_DATA
00F9 C9                 RET
00FA            SW4:
00FA DD21A801           LD      IX,NUMBER
00FE DD7E04             LD      A,(IX+4)
0101 CD5F01             CALL    SHIFT_DATA
0104 C9                 RET
0105            SW5:
0105 DD21A801           LD      IX,NUMBER
0109 DD7E05             LD      A,(IX+5)
010C CD5F01             CALL    SHIFT_DATA
010F C9                 RET
0110            SW6:
0110 DD21A801           LD      IX,NUMBER
0114 DD7E06             LD      A,(IX+6)
0117 CD5F01             CALL    SHIFT_DATA
011A C9                 RET
011B            SW7:
011B DD21A801           LD      IX,NUMBER
011F DD7E07             LD      A,(IX+7)
0122 CD5F01             CALL    SHIFT_DATA
0125 C9                 RET
0126            SW8:
0126 DD21A801           LD      IX,NUMBER
012A DD7E08             LD      A,(IX+8)
012D CD5F01             CALL    SHIFT_DATA
0130 C9                 RET
0131            SW9:
0131 DD21A801           LD      IX,NUMBER
0135 DD7E09             LD      A,(IX+9)
0138 CD5F01             CALL    SHIFT_DATA
013B C9                 RET

013C            SW0:
013C DD21A801           LD      IX,NUMBER
0140 DD7E00             LD      A,(IX+0)
0143 CD5F01             CALL    SHIFT_DATA
0146 C9                 RET
0147            SWA:
0147 CD9701             CALL    CLEAR
014A C9                 RET

014B            SWB:
014B 210080             LD      HL,8000H        ;HL = BUFFER DISPLAY
014E DD21A301           LD      IX,SEIKO        ;IX = TABLE DATA off LOGO
0152 1E05               LD      E,05H
0154            LOOP7:
0154 DD7E00             LD      A,(IX+0)        ;COPY DATA LOGO TO BUFFER-
0157 77                 LD      (HL),A          ;DISPLAY
0158 DD23               INC     IX
015A 23                 INC     HL
015B 1D                 DEC     E
015C 20F6               JR      NZ,LOOP7
015E C9                 RET

015F            SHIFT_DATA:
015F DD210080           LD      IX,8000H
0163 DD5E01             LD      E,(IX+1)
```

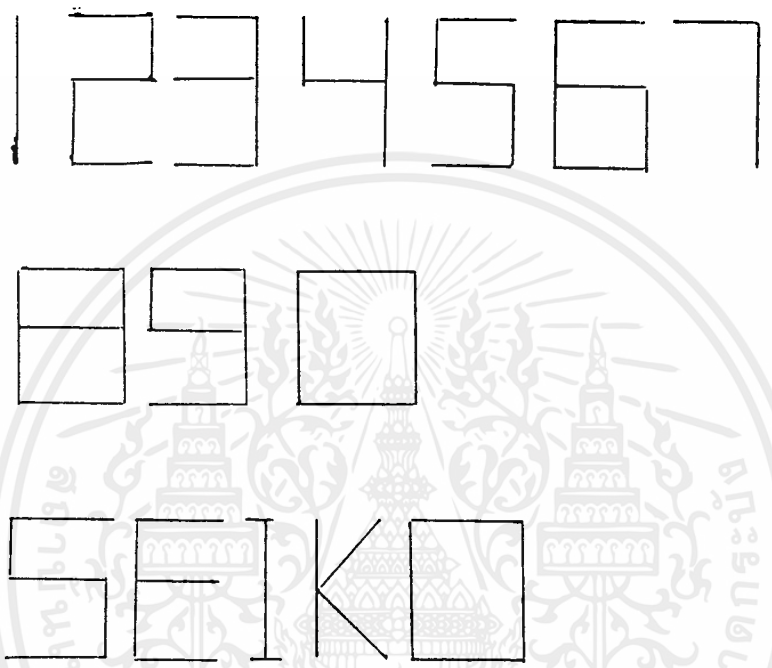
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
0166 DD7300            LD      (IX+0),E        ;SHIFT DATA 1 DIGIT
0169 DD5E02            LD      E,(IX+2)
016C DD7301            LD      (IX+1),E        ;SHIFT DATA 1 DIGIT
016F DD5E03            LD      E,(IX+3)
0172 DD7302            LD      (IX+2),E        ;SHIFT DATA 1 DIGIT
0175 DD7703            LD      (IX+3),A        ;NOW DATA TO LSB
0178 DD3604FF          LD      (IX+4),0FFH
017C C9                RET

017D             DISPLAY:
017D 210080            LD      HL,8000H        ;HL=START BUFFER ADDRESS
0180 0603              LD      B,03H
0182 0E90              LD      C,PORT2         ;C = ADDRESS PORT 8255#2
0184             DISPLAY1:
0184 7E                LD      A,(HL)          ; A =DATA IN BUFFER
0185 ED79              OUT     (C),A           ;OUT DATA TO PORT
0187 0C                INC     C               ;INC PORT
0188 23                INC     HL              ;INC BUFFER DATA
0189 10F9              DJNZ    DISPLAY1
018B 0602              LD      B,02H
018D 0EA0              LD      C,PORT3         ;C = ADDRESS PORT 8255#3
018F             DISPLAY2:
018F 7E                LD      A,(HL)
0190 ED79              OUT     (C),A
0192 0C                INC     C
0193 23                INC     HL
0194 10F9              DJNZ    DISPLAY2
0196 C9                RET
                 ;************************************
                 ;***PROGRAM CLEAR DATA IN DUFFER****
                 ;************************************
0197             CLEAR:
0197 0605              LD      B,05H           ;B = COUNTER BUFFER DISPLAY
0199 210080            LD      HL,8000H        ;HL = START ADDRESS BUFFER
019C 3EFF              LD      A,0FFH          ;A = DATA OFF LIGHT
019E             LOOP_CLEAR:
019E 77                LD      (HL),A
019F 23                INC     HL
01A0 10FC              DJNZ    LOOP_CLEAR
01A2 C9                RET

01A3 92864FC0CFSEIK0:  DFB     92H,86H,4FH,0C0H,3FH
01A8 C0F9A4B099NUMBER: DFB     0C0H,0F9H,0A4H,0B0H,99H
01AD 9282F88090        DFB     92H,82H,0F8H,80H,90H

01B2             END


0000                   END
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ผลการทดลอง

สามารถแสดงผลได้ตามต้องการ ดังนี้



รูปที่ 2 ผลที่ได้จากการทดลอง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 5
สรุปผลการทดลองและวิจารณ์

5.1 สรุปผลการทดลอง

จากการทดลองในบทที่ 4 สามารถสรุปได้ว่า แผงแสดงผลสามารถทำงานได้จริงตามโปรแกรมควบคุมที่เขียนขึ้น แต่เนื่องจากการจัดวางหลอดแผงแสดงผลเน้นให้แสดงผลเป็นตัวเลขมากกว่าตัวอักษร เพื่อไม่ให้งบประมาณในการทำโครงงานสูงมากนัก ฉนั้นจึงแสดงผลได้เฉพาะตัวเลขและตัวอักษรบางตัวเท่านั้น

5.2 แนวทางในการพัฒนา

แผงแสดงผลนี้สามารถดัดแปลงให้แสดงผลตัวอักษรภาษาไทยหรือภาษาอังกฤษก็ได้ โดยการเพิ่มหลอดไฟและวางหลอดไฟเป็นแบบ MATRIX แต่จะทำให้งบประมาณในการผลิตสูงขึ้น เนื่องจากว่าต้องใช้ SSR-AC 1 ชุดต่อหลอดไฟ 1 หลอด เช่น ถ้าจัดวางหลอดไฟเป็นแบบ MATRIX 8*8 จะต้องใช้ SSR-AC จำนวน 64 ชุด

แผงแสดงผลในโครงงานนี้ ผู้จัดทำโครงงานได้พยายามทำให้แสดงผลได้สมบูรณ์ได้เฉพาะงานบางอย่างเช่น การแสดงผลการแข่งขันกีฬาบางชนิด ที่ไม่จำเป็นในการแสดงข้อความเป็นตัวอักษรมากนัก

5.3 ปัญหาที่เกิดขึ้นในระหว่างการทดลอง

1. หลอดไฟที่ใช้เป็นหลอดไฟ AC 220 V จำเป็นจะต้องระมัดระวังในการต่อเพื่อไม่ให้เกิดการลัดวงจร

2. หลอดไฟ AC 220 V มีข้อดีเหนือกว่า LED คือสามารถให้ความสว่างมากกว่าและมองเห็นในระยะทางไกลๆได้ แต่มีข้อเสียคือ จะเกิดความร้อนขึ้นเมื่อ ON ทิ้งไว้นานๆ ฉนั้นต้องให้หลอดไฟติด-ดับตลอดเวลา เพื่อไม่ให้เกิดความร้อนมากนัก

3. การจัดวางหลอดแสดงผลแต่ละหลักพยายามลดชุด SSR-AC ให้มากที่สุดเพื่อ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ไม่ให้งบประมาณในการจัดทำโครงงานสูงมากนัก ฉนั้นการเขียนโปรแกรมให้แสดงผลเป็นตัวอักษรเลื่อนไปแต่ละหลักจึงทำไม่ได้ แต่เนื่องจากหลอดไฟจะต้องติด-ดับตลอดเวลาเพื่อไม่ให้เกิดความร้อนมากนัก ฉนั้นการแสดงผลจะทำให้ตัวเลขหรือตัวอักษรกระพริบอยู่ตลอดเวลา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บรรณานุกรม

1. RTC V4.0 USER'S MANUAL, ETT CO.,LTD.

2. ET-SSRAC SOLID-STATE RELAY,ETT CO.,LTD.

3. CONTROL PACK CP-Z80V3,ETT CO.,LTD.

4. ET-HARDWARE LAB,ETT.,LTD.

5. ET-Z80 HARDWARE EXPERIMENT MANUAL,ETT.,LTD.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ก.

Z80 CPU Specification

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Z80®-CPU
Z80A-CPU

Product Specification

MARCH 1978

The Zilog Z80 product line is a complete set of microcomputer components, development systems and support software. The Z80 microcomputer component set includes all of the circuits necessary to build high-performance microcomputer systems with virtually no other logic and a minimum number of low cost standard memory elements.

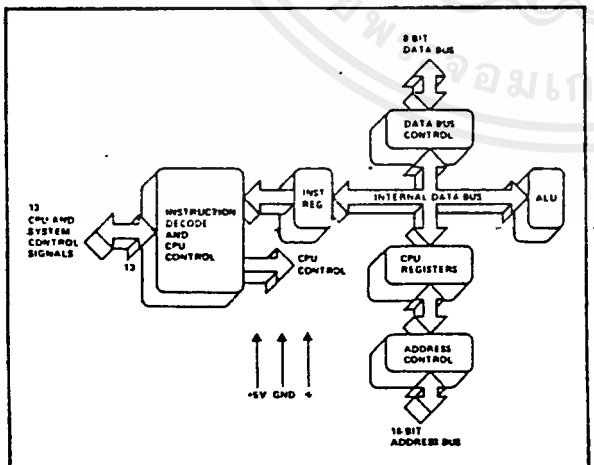
The Z80 and Z80A CPU's are third generation single chip microprocessors with unrivaled computational power. This increased computational power results in higher system through-put and more efficient memory utilization when compared to second generation microprocessors. In addition, the Z80 and Z80A CPU's are very easy to implement into a system because of their single voltage requirement plus all output signals are fully decoded and timed to control standard memory or peripheral circuits. The circuit is implemented using an N-channel, ion implanted, silicon gate MOS process.

Figure 1 is a block diagram of the CPU, Figure 2 details the internal register configuration which contains 208 bits of Read/Write memory that are accessible to the programmer. The registers include two sets of six general purpose registers that may be used individually as 8-bit registers or as 16-bit register pairs. There are also two sets of accumulator and flag registers. The programmer has access to either set of main or alternate registers through a group of exchange instructions. This alternate set allows foreground/background mode of operation or may be reserved for very fast Interrupt response. Each CPU also contains a 16-bit stack pointer which permits simple implementation of multiple level interrupts, unlimited subroutine nesting and simplification of many types of data handling.

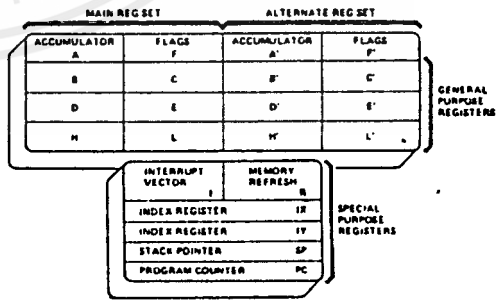
The two 16-bit index registers allow tabular data manipulation and easy implementation of relocatable code. The Refresh register provides for automatic, totally transparent refresh of external dynamic memories. The I register is used in a powerful interrupt response mode to form the upper 8 bits of a pointer to a interrupt service address table, while the interrupting device supplies the lower 8 bits of the pointer. An indirect call is then made to this service address.

FEATURES

- Single chip, N-channel Silicon Gate CPU.
- 158 instructions–includes all 78 of the 8080A instructions with total software compatibility. New instructions include 4-, 8- and 16-bit operations with more useful addressing modes such as indexed, bit and relative.
- 17 internal registers.
- Three modes of fast interrupt response plus a non-maskable interrupt.
- Directly interfaces standard speed static or dynamic memories with virtually no external logic.
- 1.0 μs instruction execution speed.
- Single 5 VDC supply and single-phase 5 volt Clock.
- Out-performs any other single chip microcomputer in 4-, 8-, or 16-bit applications.
- All pins TTL Compatible
- Built-in dynamic RAM refresh circuitry.



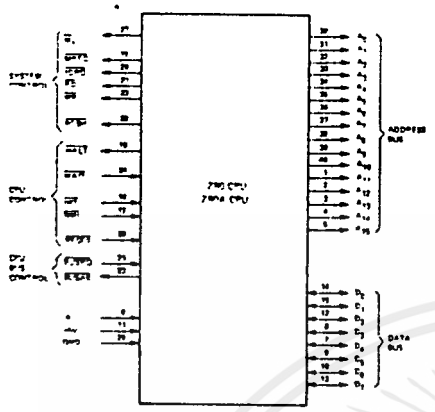
Z80, Z80A CPU BLOCK DIAGRAM



Z80, Z80A CPU REGISTERS

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Z80, Z80A-CPU Pin Description



Z80, Z80A CPU PIN CONFIGURATION

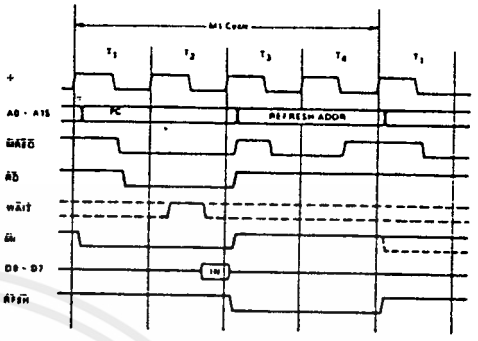
A_0-A_{15} (Address Bus)	Tri-state output, active high. A_0-A_{15} constitute a 16-bit address bus. The address bus provides the address for memory (up to 64K bytes) data exchanges and for I/O device data exchanges.
D_0-D_7 (Data Bus)	Tri-state input/output, active high. D_0-D_7 constitute an 8-bit bidirectional data bus. The data bus is used for data exchanges with memory and I/O devices.
$\overline{M_1}$ (Machine Cycle one)	Output, active low. $\overline{M_1}$ indicates that the current machine cycle is the OP code fetch cycle of an instruction execution.
$\overline{MREQ}$ (Memory Request)	Tri-state output, active low. The memory request signal indicates that the address bus holds a valid address for a memory read or memory write operation.
$\overline{IORQ}$ (Input/ Output Request)	Tri-state output, active low. The $\overline{IORQ}$ signal indicates that the lower half of the address bus holds a valid I/O address for a I/O read or write operation. An $\overline{IORQ}$ signal is also generated when an interrupt is being acknowledged to indicate that an interrupt response vector can be placed on the data bus.
$\overline{RD}$ (Memory Read)	Tri-state output, active low. $\overline{RD}$ indicates that the CPU wants to read data from memory or an I/O device. The addressed I/O device or memory should use this signal to gate data onto the CPU data bus.
$\overline{WR}$ (Memory Write)	Tri-state output, active low. $\overline{WR}$ indicates that the CPU data bus holds valid data to be stored in the addressed memory or I/O device.
$\overline{RFSH}$ (Refresh)	Output, active low. $\overline{RFSH}$ indicates that the lower 7 bits of the address bus contain a refresh address for dynamic memories and the current $\overline{MREQ}$ signal should be used to do a refresh read to all dynamic memories.
$\overline{HALT}$ (Halt state)	Output, active low. $\overline{HALT}$ indicates that the CPU has executed a HALT software instruction and is awaiting either a non-maskable or a maskable interrupt (with the mask enabled) before operation can resume. While halted, the CPU executes NOP's to maintain memory refresh activity.
$\overline{WAIT}$ (Wait)	Input, active low. $\overline{WAIT}$ indicates to the Z-80 CPU that the addressed memory or I/O devices are not ready for a data transfer. The CPU continues to enter wait states for as long as this signal is active.
$\overline{INT}$ (Interrupt Request)	Input, active low. The Interrupt Request signal is generated by I/O devices. A request will be honored at the end of the current instruction if the internal software controlled interrupt enable flip-flop (IFF) is enabled.
$\overline{NMI}$ (Non Maskable Interrupt)	Input, active low. The non-maskable interrupt request line has a higher priority than $\overline{INT}$ and is always recognized at the end of the current instruction, independent of the status of the interrupt enable flip-flop. $\overline{NMI}$ automatically forces the Z-80 CPU to restart to location 0066_H.
$\overline{RESET}$	Input, active low. $\overline{RESET}$ initializes the CPU as follows: reset interrupt enable flip-flop, clear PC and registers I and R and set interrupt to 8080A mode. During reset time, the address and data bus go to a high impedance state and all control output signals go to the inactive state.
$\overline{BUSRQ}$ (Bus Request)	Input, active low. The bus request signal has a higher priority than $\overline{NMI}$ and is always recognized at the end of the current machine cycle and is used to request the CPU address bus, data bus and tri-state output control signals to go to a high impedance state so that other devices can control these busses.
$\overline{BUSAK}$ (Bus Acknowledge)	Output, active low. Bus acknowledge is used to indicate to the requesting device that the CPU address bus, data bus and tri-state control bus signals have been set to their high impedance state and the external device can now control these signals.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Timing Waveforms

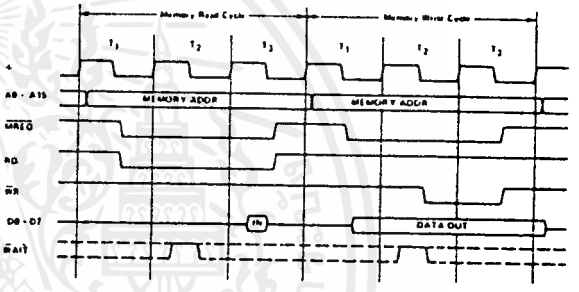
INSTRUCTION OP CODE FETCH

The program counter content (PC) is placed on the address bus immediately at the start of the cycle. One half clock time later $\overline{MREQ}$ goes active. The falling edge of $\overline{MREQ}$ can be used directly as a chip enable to dynamic memories. $\overline{RD}$ when active indicates that the memory data should be enabled onto the CPU data bus. The CPU samples data with the rising edge of the clock state T_3. Clock states T_3 and T_4 of a fetch cycle are used to refresh dynamic memories while the CPU is internally decoding and executing the instruction. The refresh control signal $\overline{RFSH}$ indicates that a refresh read of all dynamic memories should be accomplished.



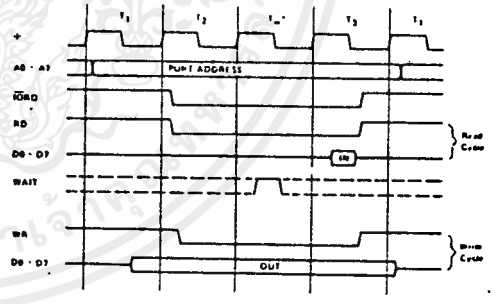
MEMORY READ OR WRITE CYCLES

Illustrated here is the timing of memory read or write cycles other than an OP code fetch (M_1 cycle). The $\overline{MREQ}$ and $\overline{RD}$ signals are used exactly as in the fetch cycle. In the case of a memory write cycle, the $\overline{MREQ}$ also becomes active when the address bus is stable so that it can be used directly as a chip enable for dynamic memories. The $\overline{WR}$ line is active when data on the data bus is stable so that it can be used directly as a R/W pulse to virtually any type of semiconductor memory.



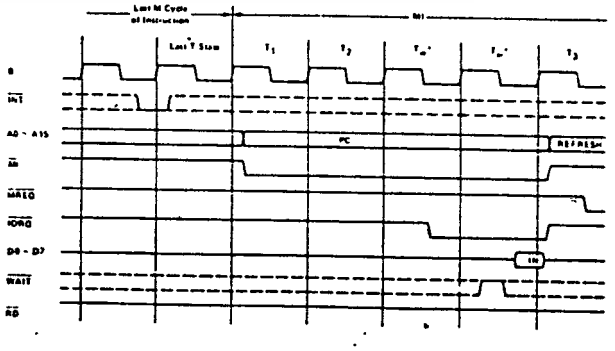
INPUT OR OUTPUT CYCLES

Illustrated here is the timing for an I/O read or I/O write operation. Notice that during I/O operations a single wait state is automatically inserted (Tw*). The reason for this is that during I/O operations this extra state allows sufficient time for an I/O port to decode its address and activate the $\overline{WAIT}$ line if a wait is required.



INTERRUPT REQUEST/ACKNOWLEDGE CYCLE

The interrupt signal is sampled by the CPU with the rising edge of the last clock at the end of any instruction. When an interrupt is accepted, a special M_1 cycle is generated. During this M_1 cycle, the $\overline{IORQ}$ signal becomes active (instead of $\overline{MREQ}$) to indicate that the interrupting device can place an 8-bit vector on the data bus. Two wait states (Tw*) are automatically added to this cycle so that a ripple priority interrupt scheme, such as the one used in the Z80 peripheral controllers, can be easily implemented.



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Z80, Z80A Instruction Set

The following is a summary of the Z80, Z80A instruction set showing the assembly language mnemonic and the symbolic operation performed by the instruction. A more detailed listing appears in the Z80-CPU technical manual, and assembly language programming manual. The instructions are divided into the following categories:

8-bit loads	Miscellaneous Group
16-bit loads	Rotates and Shifts
Exchanges	Bit Set, Reset and Test
Memory Block Moves	Input and Output
Memory Block Searches	Jumps
8-bit arithmetic and logic	Calls
16-bit arithmetic	Restarts
General purpose Accumulator & Flag Operations	Returns

In the table the following terminology is used.

- b ≡ a bit number in any 8-bit register or memory location
- cc ≡ flag condition code
  - NZ ≡ non zero
  - Z ≡ zero
  - NC ≡ non carry
  - C ≡ carry
  - PO ≡ Parity odd or no over flow
  - PE ≡ Parity even or over flow
  - P ≡ Positive
  - M ≡ Negative (minus)
- d ≡ any 8-bit destination register or memory location
- dd ≡ any 16-bit destination register or memory location
- e ≡ 8-bit signed 2's complement displacement used in relative jumps and indexed addressing
- L ≡ 8 special call locations in page zero. In decimal notation these are 0, 8, 16, 24, 32, 40, 48 and 56
- n ≡ any 8-bit binary number
- nn ≡ any 16-bit binary number
- r ≡ any 8-bit general purpose register (A, B, C, D, E, H, or L)
- s ≡ any 8-bit source register or memory location
- s_b ≡ a bit in a specific 8-bit register or memory location
- ss ≡ any 16-bit source register or memory location
- subscript "L" ≡ the low order 8 bits of a 16-bit register
- subscript "H" ≡ the high order 8 bits of a 16-bit register
- ( ) ≡ the contents within the ( ) are to be used as a pointer to a memory location or I/O port number

8-bit registers are A, B, C, D, E, H, L, I and R
16-bit register pairs are AF, BC, DE and HL
16-bit registers are SP, PC, IX and IY

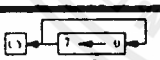
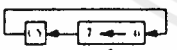
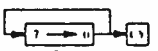
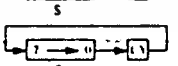
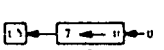
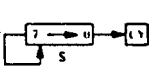
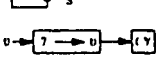
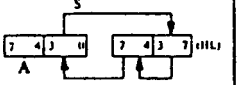
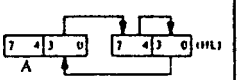
Addressing Modes implemented include combinations of the following:

Immediate	Indexed
Immediate extended	Register
Modified Page Zero	Implied
Relative	Register Indirect
Extended	Bit

	Mnemonic	Symbolic Operation	Comments
8-BIT LOADS	LD r, s	r ← s	s ≡ r, n, (HL), (IX+e), (IY+e)
	LD d, r	d ← r	d ≡ (HL), r (IX+e), (IY+e)
	LD d, n	d ← n	d ≡ (HL), (IX+e), (IY+e)
	LD A, s	A ← s	s ≡ (BC), (DE), (nn), I, R
	LD d, A	d ← A	d ≡ (BC), (DE), (nn), I, R
16-BIT LOADS	LD dd, nn	dd ← nn	dd ≡ BC, DE, HL, SP, IX, IY
	LD dd, (nn)	dd ← (nn)	dd ≡ BC, DE, HL, SP, IX, IY
	LD (nn), ss	(nn) ← ss	ss ≡ BC, DE, HL, SP, IX, IY
	LD SP, ss	SP ← ss	ss = HL, IX, IY
	PUSH ss	$(SP-1) \leftarrow ss_H$; $(SP-2) \leftarrow ss_L$	ss = BC, DE, HL, AF, IX, IY
	POP dd	$dd_L \leftarrow (SP)$; $dd_H \leftarrow (SP+1)$	dd = BC, DE, HL, AF, IX, IY
EXCHANGES	EX DE, HL	DE ↔ HL	
	EX AF, AF'	AF ↔ AF'	
	EXX	$\begin{pmatrix} BC \\ DE \\ HL \end{pmatrix} \leftrightarrow \begin{pmatrix} BC' \\ DE' \\ HL' \end{pmatrix}$	
	EX (SP), ss	$(SP) \leftarrow ss_L$, $(SP+1) \leftarrow ss_H$	ss ≡ HL, IX, IY

	Mnemonic	Symbolic Operation	Comments
MEMORY BLOCK MOVES	LDI	(DE) ← (HL), DE ← DE+1 HL ← HL+1, BC ← BC-1	
	LDIR	(DE) ← (HL), DE ← DE+1 HL ← HL+1, BC ← BC-1 Repeat until BC = 0	
	LDD	(DE) ← (HL), DE ← DE-1 HL ← HL-1, BC ← BC-1	
	LDDR	(DE) ← (HL), DE ← DE-1 HL ← HL-1, BC ← BC-1 Repeat until BC = 0	
MEMORY BLOCK SEARCHES	CPI	A-(HL), HL ← HL+1 BC ← BC-1	A-(HL) sets the flags only. A is not affected
	CPIR	A-(HL), HL ← HL+1 BC ← BC-1, Repeat until BC = 0 or A = (HL)	
	CPD	A-(HL), HL ← HL-1 BC ← BC-1	
	CPDR	A-(HL), HL ← HL-1 BC ← BC-1, Repeat until BC= 0 or A = (HL)	
8-BIT ALU	ADD s	A ← A + s	CY is the carry flag
	ADC s	A ← A + s + CY	
	SUB s	A ← A - s	s ≡ r, n, (HL) (IX+e), (IY+e)
	SBC s	A ← A - s - CY	
	AND s	A ← A ∧ s	
	OR s	A ← A ∨ s	
	XOR s	A ← A ⊕ s	

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

	Mnemonic	Symbolic Operation	Comments
8-BIT ALU	CP s	A - s	s = r, n (HL) (IX+e), (IY+e)
	INC d	d ← d + 1	d = r, (HL) (IX+e), (IY+e)
	DEC d	d ← d - 1	
16-BIT ARITHMETIC	ADD HL, ss	HL ← HL + ss	ss ≡ BC, DE HL, SP
	ADC HL, ss	HL ← HL + ss + CY	
	SBC HL, ss	HL ← HL - ss - CY	
	ADD IX, ss	IX ← IX + ss	ss ≡ BC, DE, IX, SP
	ADD IY, ss	IY ← IY + ss	ss ≡ BC, DE, IY, SP
	INC dd	dd ← dd + 1	dd ≡ BC, DE, HL, SP, IX, IY
	DEC dd	dd ← dd - 1	dd ≡ BC, DE, HL, SP, IX, IY
GP ACC. & FLAG	DAA	Converts A contents into packed BCD following add or subtract.	Operands must be in packed BCD format
	CPL	$A \leftarrow \overline{A}$	
	NEG	A ← 00 - A	
	CCF	$CY \leftarrow \overline{CY}$	
	SCF	CY ← 1	
MISCELLANEOUS	NOP	No operation	
	HALT	Halt CPU	
	DI	Disable Interrupts	
	EI	Enable Interrupts	
	IM 0	Set interrupt mode 0	8080A mode
	IM 1	Set interrupt mode 1	Call to 0038_H
	IM 2	Set interrupt mode 2	Indirect Call
ROTATES AND SHIFTS	RLC s		s ≡ r, (HL) (IX+e), (IY+e)
	RL s		
	RRC s		
	RR s		
	SLA s		
	SRA s		
	SRL s		
	RLD		
	RRD		

	Mnemonic	Symbolic Operation	Comments
BIT S. R. & T	BIT b, s	$Z \leftarrow \overline{s_b}$	Z is zero flag
	SET b, s	$s_b \leftarrow 1$	s ≡ r, (HL) (IX+e), (IY+e)
	RES b, s	$s_b \leftarrow 0$	
INPUT AND OUTPUT	IN A, (n)	A ← (n)	
	IN r, (C)	r ← (C)	Set flags
	INI	(HL) ← (C), HL ← HL + 1 B ← B - 1	
	INIR	(HL) ← (C), HL ← HL + 1 B ← B - 1 Repeat until B = 0	
	IND	(HL) ← (C), HL ← HL - 1 B ← B - 1	
	INDR	(HL) ← (C), HL ← HL - 1 B ← B - 1 Repeat until B = 0	
	OUT(n), A	(n) ← A	
	OUT(C), r	(C) ← r	
	OUTI	(C) ← (HL), HL ← HL + 1 B ← B - 1	
	OTIR	(C) ← (HL), HL ← HL + 1 B ← B - 1 Repeat until B = 0	
	OUTD	(C) ← (HL), HL ← HL - 1 B ← B - 1	
	OTDR	(C) ← (HL), HL ← HL - 1 B ← B - 1 Repeat until B = 0	
JUMPS	JP nn	PC ← nn	cc ≡ NZ PO, Z PE, NC P, C M
	JP cc, nn	If condition cc is true PC ← nn, else continue	
	JR e	PC ← PC + e	
	JR kk, e	If condition kk is true PC ← PC + e, else continue	kk ≡ NZ NC, Z C
	JP (ss)	PC ← ss	ss = HL, IX, IY
	DJNZ e	B ← B - 1, if B = 0 continue, else PC ← PC + e	
CALLS	CALL nn	$(SP-1) \leftarrow PC_H$ $(SP-2) \leftarrow PC_L$, PC ← nn	cc ≡ NZ PO, Z PE, NC P, C M
	CALL cc, nn	If condition cc is false continue, else same as CALL nn	
RESTARTS	RST L	$(SP-1) \leftarrow PC_H$ $(SP-2) \leftarrow PC_L$, $PC_H \leftarrow 0$ $PC_L \leftarrow L$	
RETURNS	RET	$PC_L \leftarrow (SP)$, $PC_H \leftarrow (SP+1)$	cc ≡ NZ PO, Z PE, NC P, C M
	RET cc	If condition cc is false continue, else same as RET	
	RETI	Return from interrupt, same as RET	
	RETN	Return from non-maskable interrupt	

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

A.C. Characteristics Z80-CPU

T_A = 0°C to 70°C, Vcc = +5V ± 5%, Unless Otherwise Noted.

Signal	Symbol	Parameter	Min	Max	Unit	Test Condition	
Φ	t_c	Clock Period	4	1121	μsec		[12] $t_c = t_{w(\Phi H)} + t_{w(\Phi L)} + t_r + t_f$
	$t_{w(\Phi H)}$	Clock Pulse Width, Clock High	180	[E]	nsec		
	$t_{w(\Phi L)}$	Clock Pulse Width, Clock Low	180	2000	nsec		
	t_r, t_f	Clock Rise and Fall Time		30	nsec		
A_{15-0}	$t_{D(AD)}$	Address Output Delay		145	nsec	C_L = 50pF	
	$t_{F(AD)}$	Delay to Float		110	nsec		
	t_{acm}	Address Stable Prior to MREQ (Memory Cycle)	[1]		nsec		[1] $t_{acm} = t_{w(\Phi H)} + t_f - 75$
	t_{aci}	Address Stable Prior to IORQ, RD or WR (I/O Cycle)	[2]		nsec		[2] $t_{aci} = t_c - 80$
	t_{ca}	Address Stable from RD, WR, IORQ or MREQ	[3]		nsec		[3] $t_{ca} = t_{w(\Phi L)} + t_f - 40$
	t_{caf}	Address Stable From RD or WR During Float	[4]		nsec		[4] $t_{caf} = t_{w(\Phi L)} + t_f - 60$
D_{7-0}	$t_{D(D)}$	Data Output Delay		230	nsec	C_L = 50pF	[5] $t_{dcm} = t_c - 210$
	$t_{F(D)}$	Delay to Float During Write Cycle		90	nsec		[6] $t_{dci} = t_{w(\Phi L)} + t_f - 210$
	$t_{S(D)}$	Data Setup Time to Rising Edge of Clock During M1 Cycle	50		nsec		[7] $t_{cdf} = t_{w(\Phi L)} + t_f - 80$
	$t_{S(D)}$	Data Setup Time to Falling Edge of Clock During M2 to M5	60		nsec		
	t_{dcm}	Data Stable Prior to WR (Memory Cycle)	[5]		nsec		
	t_{dci}	Data Stable Prior to WR (I/O Cycle)	[6]		nsec		
	t_{cdf}	Data Stable From WR	[7]				
	t_H	Any Hold Time for Setup Time	0		nsec		
MREQ	$t_{DL(MR)}$	MREQ Delay From Falling Edge of Clock, MREQ Low		100	nsec	C_L = 50pF	
	$t_{DH(MR)}$	MREQ Delay From Rising Edge of Clock, MREQ High		100	nsec		
	$t_{DF(MR)}$	MREQ Delay From Falling Edge of Clock, MREQ High		100	nsec		
	$t_{w(MRL)}$	Pulse Width, MREQ Low	[8]		nsec		[8] $t_{w(MRL)} = t_c - 40$
	$t_{w(MRH)}$	Pulse Width, MREQ High	[9]		nsec		[9] $t_{w(MRH)} = t_{w(\Phi H)} + t_f - 30$
IORQ	$t_{DL(IR)}$	IORQ Delay From Rising Edge of Clock, IORQ Low		90	nsec	C_L = 50pF	
	$t_{DF(IR)}$	IORQ Delay From Falling Edge of Clock, IORQ Low		110	nsec		
	$t_{DH(IR)}$	IORQ Delay From Rising Edge of Clock, IORQ High		100	nsec		
	$t_{DF(IR)}$	IORQ Delay From Falling Edge of Clock, IORQ High		110	nsec		
RD	$t_{DL(RD)}$	RD Delay From Rising Edge of Clock, RD Low		100	nsec	C_L = 50pF	
	$t_{DF(RD)}$	RD Delay From Falling Edge of Clock, RD Low		130	nsec		
	$t_{DH(RD)}$	RD Delay From Rising Edge of Clock, RD High		100	nsec		
	$t_{DF(RD)}$	RD Delay From Falling Edge of Clock, RD High		110	nsec		
WR	$t_{DL(WR)}$	WR Delay From Rising Edge of Clock, WR Low		80	nsec	C_L = 50pF	
	$t_{DF(WR)}$	WR Delay From Falling Edge of Clock, WR Low		90	nsec		
	$t_{DH(WR)}$	WR Delay From Falling Edge of Clock, WR High		100	nsec		
	$t_{w(WRL)}$	Pulse Width, WR Low	[10]		nsec		[10] $t_{w(WRL)} = t_c - 40$
M1	$t_{DL(M1)}$	M1 Delay From Rising Edge of Clock, M1 Low		130	nsec	C_L = 50pF	
	$t_{DH(M1)}$	M1 Delay From Rising Edge of Clock, M1 High		130	nsec		
RFSH	$t_{DL(RF)}$	RFSH Delay From Rising Edge of Clock, RFSH Low		180	nsec	C_L = 50pF	
	$t_{DH(RF)}$	RFSH Delay From Rising Edge of Clock, RFSH High		150	nsec		
WAIT	$t_{s(WT)}$	WAIT Setup Time to Falling Edge of Clock	70		nsec		
HALT	$t_{D(HT)}$	HALT Delay Time From Falling Edge of Clock		300	nsec	C_L = 50pF	
INT	$t_{s(IT)}$	INT Setup Time to Rising Edge of Clock	80		nsec		
NMI	$t_{w(NMI)}$	Pulse Width, NMI Low	80		nsec		
BUSRQ	$t_{s(BQ)}$	BUSRQ Setup Time to Rising Edge of Clock	80		nsec		
BUSAK	$t_{DL(BA)}$	BUSAK Delay From Rising Edge of Clock, BUSAK Low		120	nsec	C_L = 50pF	
	$t_{DH(BA)}$	BUSAK Delay From Falling Edge of Clock, BUSAK High		110	nsec		
RESET	$t_{s(RS)}$	RESET Setup Time to Rising Edge of Clock	90		nsec		
	$t_{F(C)}$	Delay to Float (MREQ, IORQ, RD and WR)		100	nsec		
	t_{mr}	M1 Stable Prior to IORQ (Interrupt Ack.)	[11]		nsec		[11] $t_{mr} = 2t_c + t_{w(\Phi H)} + t_f - 80$

NOTES

A Data should be enabled onto the CPU data bus when RD is active. During interrupt acknowledge data should be enabled when M1 and IORQ are both active.

B All control signals are internally synchronized, so they may be totally asynchronous with respect to the clock.

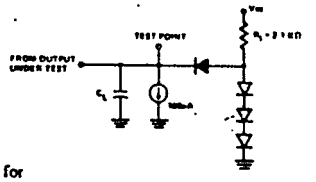
C The RESET signal must be active for a minimum of 3 clock cycles.

D Output Delay vs. Loaded Capacitance

TA = 70°C Vcc = +5V ±5%

Add 10nsec delay for each 50pf increase in load up to a maximum of 200pf for the data bus & 100pf for address & control lines

E Although static by design, testing guarantees $t_{w(\Phi H)}$ of 200 μsec maximum



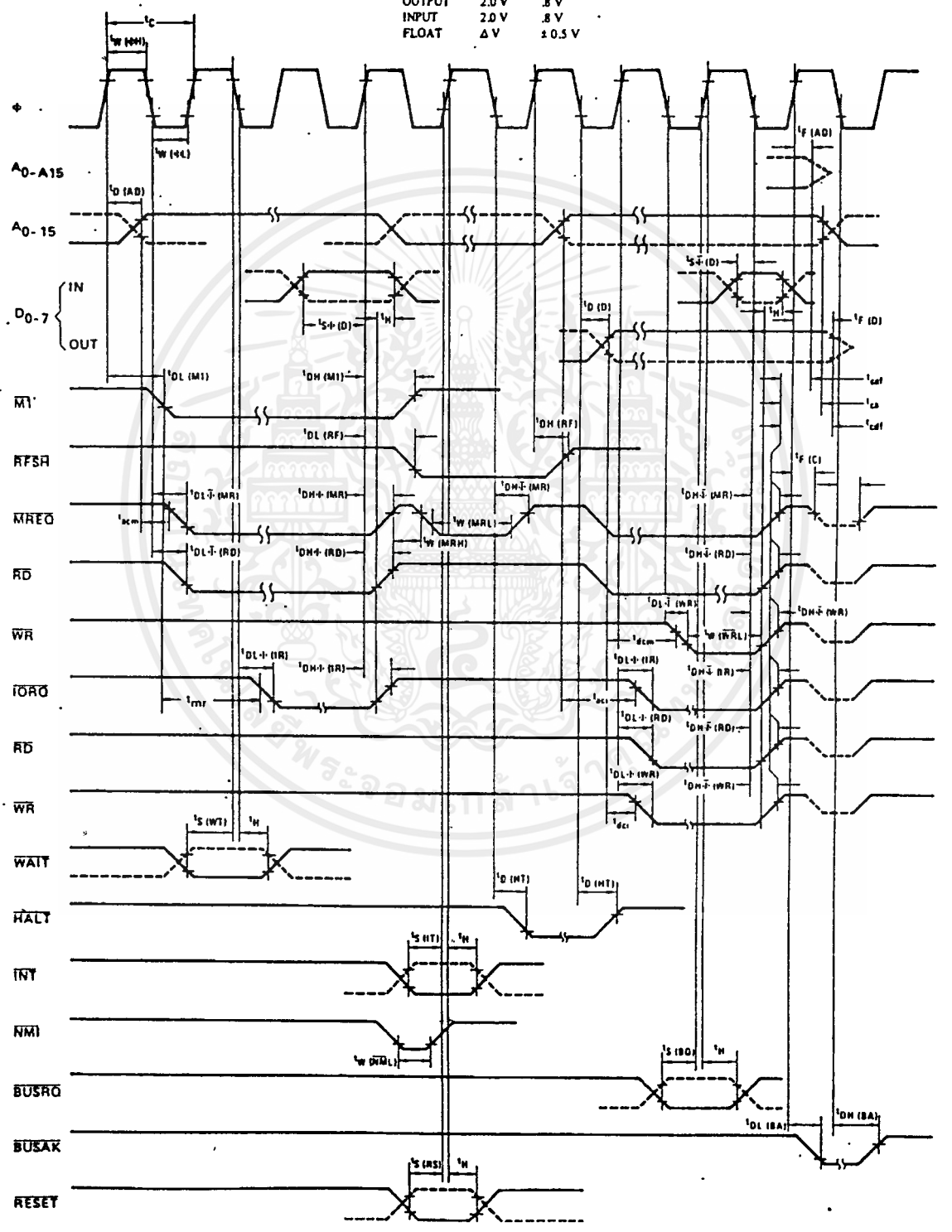
Load circuit for Output

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

A.C. Timing Diagram

Timing measurements are made at the following voltages, unless otherwise specified:

	"1"	"0"
CLOCK	V_{cc} -.6V	.45V
OUTPUT	2.0 V	.8 V
INPUT	2.0 V	.8 V
FLOAT	Δ V	± 0.5 V



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Absolute Maximum Ratings

Temperature Under Bias	Specified operating range.
Storage Temperature	-65°C to +150°C
Voltage On Any Pin with Respect to Ground	-0.3V to +7V
Power Dissipation	1.5W

*Comment

Stresses above those listed under "Absolute Maximum Rating" may cause permanent damage to the device. This is a stress rating only and functional operation of the device at these or any other condition above those indicated in the operational sections of this specification is not implied. Exposure to absolute maximum rating conditions for extended periods may affect device reliability.

Note. For Z80-CPU all AC and DC characteristics remain the same for the military grade parts except I_{cc}.

I_{cc} = 200 mA

Z80-CPU D.C. Characteristics

T_A = 0°C to 70°C, V_{cc} = 5V ± 5% unless otherwise specified

Symbol	Parameter	Min.	Typ.	Max.	Unit	Test Condition
V_{ILC}	Clock Input Low Voltage	-0.3		0.45	V	
V_{IHC}	Clock Input High Voltage	V_{cc} -.6		V_{cc}+.3	V	
V_{IL}	Input Low Voltage	-0.3		0.8	V	
V_{IH}	Input High Voltage	2.0		V_{cc}	V	
V_{OL}	Output Low Voltage			0.4	V	I_{OL}=1.8mA
V_{OH}	Output High Voltage	2.4			V	I_{OH} = -250μA
I_{cc}	Power Supply Current			150	mA	
I_{LI}	Input Leakage Current			10	μA	V_{IN}=0 to V_{cc}
I_{LOH}	Tri-State Output Leakage Current in Float			10	μA	V_{OUT}=2.4 to V_{cc}
I_{LOL}	Tri-State Output Leakage Current in Float			-10	μA	V_{OUT}=0.4V
I_{LD}	Data Bus Leakage Current in Input Mode			±10	μA	$0 \le V_{IN} \le V_{cc}$

Capacitance

T_A = 25°C, f = 1 MHz,
unmeasured pins returned to ground

Symbol	Parameter	Max.	Unit
C_Φ	Clock Capacitance	35	pF
C_{IN}	Input Capacitance	5	pF
C_{OUT}	Output Capacitance	10	pF

Z80-CPU Ordering Information

C – Ceramic
P – Plastic
S – Standard 5V ±5% 0° to 70°C
E – Extended 5V ±5% -40° to 85°C
M – Military 5V ±10% -55° to 125°C

Z80A-CPU D.C. Characteristics

T_A = 0°C to 70°C, V_{cc} = 5V ± 5% unless otherwise specified

Symbol	Parameter	Min	Typ.	Max.	Unit	Test Condition
V_{ILC}	Clock Input Low Voltage	-0.3		0.45	V	
V_{IHC}	Clock Input High Voltage	V_{cc} -.6		V_{cc}+.3	V	
V_{IL}	Input Low Voltage	-0.3		0.8	V	
V_{IH}	Input High Voltage	2.0		V_{cc}	V	
V_{OL}	Output Low Voltage			0.4	V	I_{OL}=1.8mA
V_{OH}	Output High Voltage	2.4			V	I_{OH} = -250μA
I_{cc}	Power Supply Current		90	200	mA	
I_{LI}	Input Leakage Current			10	μA	V_{IN}=0 to V_{cc}
I_{LOH}	Tri-State Output Leakage Current in Float			10	μA	V_{OUT}=2.4 to V_{cc}
I_{LOL}	Tri-State Output Leakage Current in Float			-10	μA	V_{OUT}=0.4V
I_{LD}	Data Bus Leakage Current in Input Mode			±10	μA	$0 \le V_{IN} \le V_{cc}$

Capacitance

T_A = 25°C, f = 1 MHz,
unmeasured pins returned to ground

Symbol	Parameter	Max.	Unit
C_Φ	Clock Capacitance	35	pF
C_{IN}	Input Capacitance	5	pF
C_{OUT}	Output Capacitance	10	pF

Z80A-CPU Ordering Information

C – Ceramic
P – Plastic
S – Standard 5V ±5% 0° to 70°C

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

A.C. Characteristics — Z80A-CPU

T_A = 0°C to 70°C, Vcc = +5V ± 5%, Unless Otherwise Noted.

Signal	Symbol	Parameter	Min	Max	Unit	Test Condition
Φ	t_c	Clock Period	.25	[12]	μsec	
	$t_w(\Phi H)$	Clock Pulse Width, Clock High	110	[E]	nsec	
	$t_w(\Phi L)$	Clock Pulse Width, Clock Low	110	2000	nsec	
	t_r, t_f	Clock Rise and Fall Time		30	nsec	
A_{0-15}	$t_{D(AD)}$	Address Output Delay		110	nsec	C_L = 50pF
	$t_{F(AD)}$	Delay to Float		90	nsec	
	t_{acm}	Address Stable Prior to $\overline{MREQ}$ (Memory Cycle)	[1]		nsec	
	t_{aci}	Address Stable Prior to $\overline{IORQ}$, $\overline{RD}$ or $\overline{WR}$ (I/O Cycle)	[2]		nsec	
	t_{ca}	Address Stable from $\overline{RD}$, $\overline{WR}$, $\overline{IORQ}$ or $\overline{MREQ}$	[3]		nsec	
	t_{caf}	Address Stable From $\overline{RD}$ or $\overline{WR}$ During Float	[4]		nsec	
D_{0-7}	$t_{D(D)}$	Data Output Delay		150	nsec	C_L = 50pF
	$t_{F(D)}$	Delay to Float During Write Cycle		90	nsec	
	$t_{S\Phi(D)}$	Data Setup Time to Rising Edge of Clock During M1 Cycle	35		nsec	
	$t_{S\overline{\Phi}(D)}$	Data Setup Time to Falling Edge of Clock During M2 to M5	50		nsec	
	t_{dcm}	Data Stable Prior to $\overline{WR}$ (Memory Cycle)	[5]		nsec	
	t_{dci}	Data Stable Prior to $\overline{WR}$ (I/O Cycle)	[6]		nsec	
	t_{cdf}	Data Stable From $\overline{WR}$	[7]			
	t_H	Any Hold Time for Setup Time		0	nsec	
$\overline{MREQ}$	$t_{DL\overline{\Phi}(MR)}$	$\overline{MREQ}$ Delay From Falling Edge of Clock, $\overline{MREQ}$ Low		85	nsec	C_L = 50pF
	$t_{DH\Phi(MR)}$	$\overline{MREQ}$ Delay From Rising Edge of Clock, $\overline{MREQ}$ High		85	nsec	
	$t_{DH\overline{\Phi}(MR)}$	$\overline{MREQ}$ Delay From Falling Edge of Clock, $\overline{MREQ}$ High		85	nsec	
	$t_{w(MRL)}$	Pulse Width, $\overline{MREQ}$ Low	[8]		nsec	
	$t_{w(MRH)}$	Pulse Width, $\overline{MREQ}$ High	[9]		nsec	
$\overline{IORQ}$	$t_{DL\Phi(IR)}$	$\overline{IORQ}$ Delay From Rising Edge of Clock, $\overline{IORQ}$ Low		75	nsec	C_L = 50pF
	$t_{DL\overline{\Phi}(IR)}$	$\overline{IORQ}$ Delay From Falling Edge of Clock, $\overline{IORQ}$ Low		85	nsec	
	$t_{DH\Phi(IR)}$	$\overline{IORQ}$ Delay From Rising Edge of Clock, $\overline{IORQ}$ High		85	nsec	
	$t_{DH\overline{\Phi}(IR)}$	$\overline{IORQ}$ Delay From Falling Edge of Clock, $\overline{IORQ}$ High		85	nsec	
$\overline{RD}$	$t_{DL\Phi(RD)}$	$\overline{RD}$ Delay From Rising Edge of Clock, $\overline{RD}$ Low		85	nsec	C_L = 50pF
	$t_{DL\overline{\Phi}(RD)}$	$\overline{RD}$ Delay From Falling Edge of Clock, $\overline{RD}$ Low		95	nsec	
	$t_{DH\Phi(RD)}$	$\overline{RD}$ Delay From Rising Edge of Clock, $\overline{RD}$ High		85	nsec	
	$t_{DH\overline{\Phi}(RD)}$	$\overline{RD}$ Delay From Falling Edge of Clock, $\overline{RD}$ High		85	nsec	
$\overline{WR}$	$t_{DL\Phi(WR)}$	$\overline{WR}$ Delay From Rising Edge of Clock, $\overline{WR}$ Low		65	nsec	C_L = 50pF
	$t_{DL\overline{\Phi}(WR)}$	$\overline{WR}$ Delay From Falling Edge of Clock, $\overline{WR}$ Low		80	nsec	
	$t_{DH\overline{\Phi}(WR)}$	$\overline{WR}$ Delay From Falling Edge of Clock, $\overline{WR}$ High		80	nsec	
	$t_{w(WRL)}$	Pulse Width, $\overline{WR}$ Low	[10]		nsec	
$\overline{M1}$	$t_{DL(M1)}$	$\overline{M1}$ Delay From Rising Edge of Clock, $\overline{M1}$ Low		100	nsec	C_L = 50pF
	$t_{DH(M1)}$	$\overline{M1}$ Delay From Rising Edge of Clock, $\overline{M1}$ High		100	nsec	
$\overline{RFSH}$	$t_{DL(RF)}$	$\overline{RFSH}$ Delay From Rising Edge of Clock, $\overline{RFSH}$ Low		130	nsec	C_L = 50pF
	$t_{DH(RF)}$	$\overline{RFSH}$ Delay From Rising Edge of Clock, $\overline{RFSH}$ High		120	nsec	
$\overline{WAIT}$	$t_{s(WT)}$	$\overline{WAIT}$ Setup Time to Falling Edge of Clock	70		nsec	
$\overline{HALT}$	$t_{D(HT)}$	$\overline{HALT}$ Delay Time From Falling Edge of Clock		300	nsec	C_L = 50pF
$\overline{INT}$	$t_{s(IT)}$	$\overline{INT}$ Setup Time to Rising Edge of Clock	80		nsec	
$\overline{NMI}$	$t_{w(NML)}$	Pulse Width, $\overline{NMI}$ Low	80		nsec	
$\overline{BUSRQ}$	$t_{s(BQ)}$	$\overline{BUSRQ}$ Setup Time to Rising Edge of Clock	50		nsec	
$\overline{BUSAK}$	$t_{DL(BA)}$	$\overline{BUSAK}$ Delay From Rising Edge of Clock, $\overline{BUSAK}$ Low		100	nsec	C_L = 50pF
	$t_{DH(BA)}$	$\overline{BUSAK}$ Delay From Falling Edge of Clock, $\overline{BUSAK}$ High		100	nsec	
$\overline{RESET}$	$t_{s(RS)}$	$\overline{RESET}$ Setup Time to Rising Edge of Clock	60		nsec	
	$t_{F(C)}$	Delay to Float ($\overline{MREQ}$, $\overline{IORQ}$, $\overline{RD}$ and $\overline{WR}$)		80	nsec	
	t_{mi}	M1 Stable Prior to IORQ (Interrupt Ack.)	1111		nsec	

[12] $t_c = t_{w(\Phi H)} + t_{w(\Phi L)} + t_r + t_f$

[1] $t_{acm} = t_{w(\Phi H)} + t_f - 65$

[2] $t_{aci} = t_c - 70$

[3] $t_{ca} = t_{w(\Phi L)} + t_r - 50$

[4] $t_{caf} = t_{w(\Phi L)} + t_r - 45$

[5] $t_{dcm} = t_c - 170$

[6] $t_{dci} = t_{w(\Phi L)} + t_f - 170$

[7] $t_{cdf} = t_{w(\Phi L)} + t_f - 70$

[8] $t_{w(\overline{MRL})} = t_c - 30$

[9] $t_{w(\overline{MRH})} = t_{w(\Phi H)} + t_f - 20$

[10] $t_{w(\overline{WRL})} = t_c - 30$

[11] $t_{mr} = 2t_c + t_{w(\Phi H)} + t_f - 65$

NOTES:

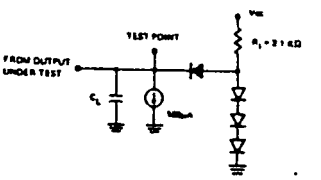
A. Data should be enabled onto the CPU data bus when $\overline{RD}$ is active. During interrupt acknowledge data should be enabled when $\overline{M1}$ and $\overline{IORQ}$ are both active.

B. All control signals are internally synchronized, so they may be totally asynchronous with respect to the clock.

C. The $\overline{RESET}$ signal must be active for a minimum of 3 clock cycles.

D. Output Delay vs. Loaded Capacitance
TA = 70°C Vcc = +5V ±5%
Add 10nsec delay for each 50pf increase in load up to maximum of 200pf for data bus and 100pf for address & control lines.

E. Although static by design, testing guarantees $t_{w(\Phi H)}$ of 200 μsec maximum



Load circuit for Output

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ข.

วิธีการพัฒนาโปรแกรมควบคุม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

วิธีการพัฒนาโปนแกรมควบคุม

วิธีการพัฒนาโปรแกรมควบคุม

วิธีการพัฒนาโปรแกรมควบคุมที่ใช้ในโครงงานนี้มี 2 วิธี คือ

1. พัฒนาโปรแกรมผ่าน DEBUGGER
2. พัฒนาโปรแกรมผ่าน EPROM EMULATOR

ขั้นตอนวิธีการทางคอมพิวเตอร์ (COMPUTER ALGORITHM)

ขั้นตอนวิธีการแก้ปัญหาด้วยคอมพิวเตอร์ แบ่งเป็นขั้นตอน ดังนี้

1. ASSIGN PROBLEM กำหนดปัญหา,ความต้องการและเงื่อนไขของงาน
2. FLOW CHART กำหนดผังระบบ(SYSTEM FLOW)และผังโปรแกรม(PROGRAM FLOW)

   ผังระบบ DATA FLOW ที่ต้องทำและความสัมพันธ์แต่ละส่วน

   ผังโปรแกรม งานของส่วนย่อยๆ,ตัวแปรและพารามิเตอร์ต่างๆ
3. PROGRAMMING ขั้นตอนของโปรแกรมเมอร์ที่จะเขียนโปรแกรมให้ทำงานตามความต้องการหรือเงื่อนไขของ PROGRAM FLOW
4. TESTING AND DEBUGGING ทดสอบและแก้ไขโปรมแกรมให้ทำงานได้ถูกต้องตามความต้องการของงาน
5. DOCUMENTATION คือ การจัดทำเอกสาร แบ่งเป็น 2 อย่าง คือ

   -USER MANUAL คือ คู่มือของผู้ใช้

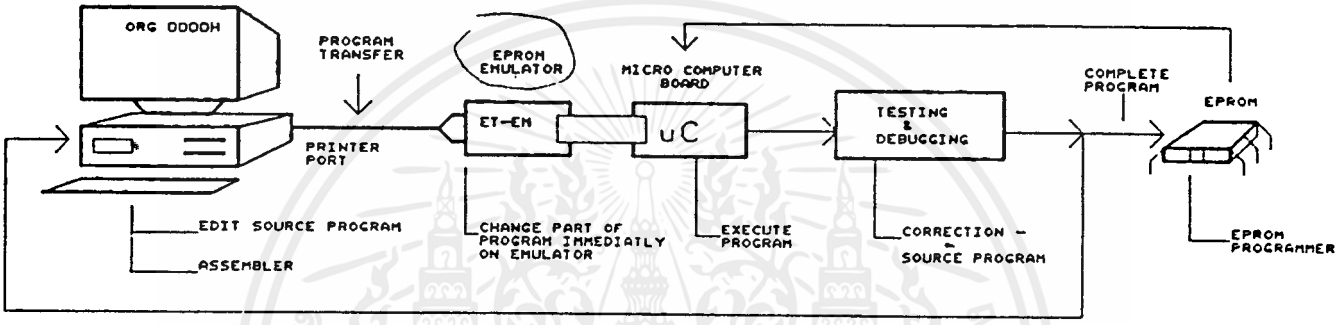
   -TECNICAL REFERENCE MANUAL คือ คู่มือของโปรแกรมเมอร์เป็นรายละเอียดการทำงานของโปรแกรม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

วิธีการพัฒนาโปรแกรมสำหรับบอร์ดไมโครคอมพิวเตอร์/ไมโครคอนโทรลเลอร์

ในที่นี้หมายถึง โปรแกรมที่ถูกพัฒนาขึ้นเพื่อใช้ควบคุม ซึ่งโปรแกรมจะถูก EXECUTE ด้วย CPU บอร์ดควบคุมใดๆก็ตาม จะมีขบวนการปฏิบัติ ดังนี้

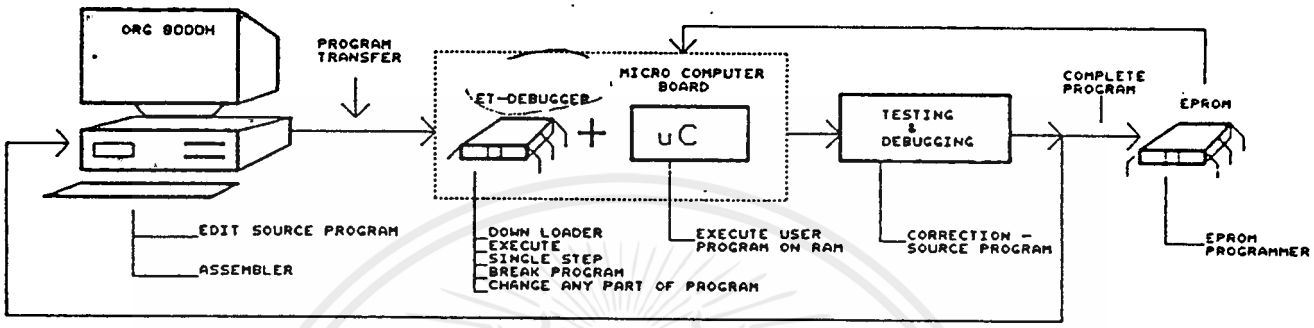
วิธีที่ 1 PROGRAM + EPROM EMULATOR + MICROCOMPUTER BOARD



ข้อดี	ข้อเสีย
-โปรแกรมสามารถแก้ไข้ทันทีโดยไม่ต้องนำอีพรอมไปทำการลบข้อมูล -สามารถแก้ไขข้อมูลบางส่วนในตัว CODE โดยไม่ต้องทำการ ASSEMBLE SOURCE PROGRAM -ทำการ RUN ที่ตำแหน่งแอดเดรสจริงเมื่อพัฒนาโปรแกรมเสร็จสมบูรณ์ สามารถนำ CODE ไปโปรมแกรมลงในอีพรอม โดยไม่ต้องแก้ไขใหม่	-การแก้ไขหรือค้นหาที่ผิดของโปรแกรมกระทำได้ยากเนื่องจากไม่สามารถติดตาม การทำงานของโปรแกรมได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

วิธีที่ 2 PROGRAM + DEBUGGER + MICROCOMPUTER BOARD



ข้อดี

-ทำการแก้ไขโปรแกรมและ RUN ได้ทันที โดยไม่ต้องเขียน/ลบอีพรอม

-สามารถหยุดการทำงานของโปรแกรมหรือให้ทำงานทีละ STEP เพื่อตรวจสอบการทำงานได้

-มีคำสั่งติดต่อโดยตรงกับ REGISTER ของ CPU หรือติดต่อกับอุปกรณ์ I/O โดยตรง ทำได้ง่ายและสะดวกในการพัฒนาโปรแกรม

-เข้าถึงหน่วยความจำได้ทุกตำแหน่งของระบบ

ข้อเสีย

-ต้องทำการปรับตำแหน่งของโปรแกรมหรือตำแหน่งอินเตอร์รัพท์เนื่องจากจะไม่ได้ RUN อยู่บนตำแหน่งแอดเดรสจริงของ CPU เมื่อเปิดเครื่อง (POWER ON หรือ RESET)

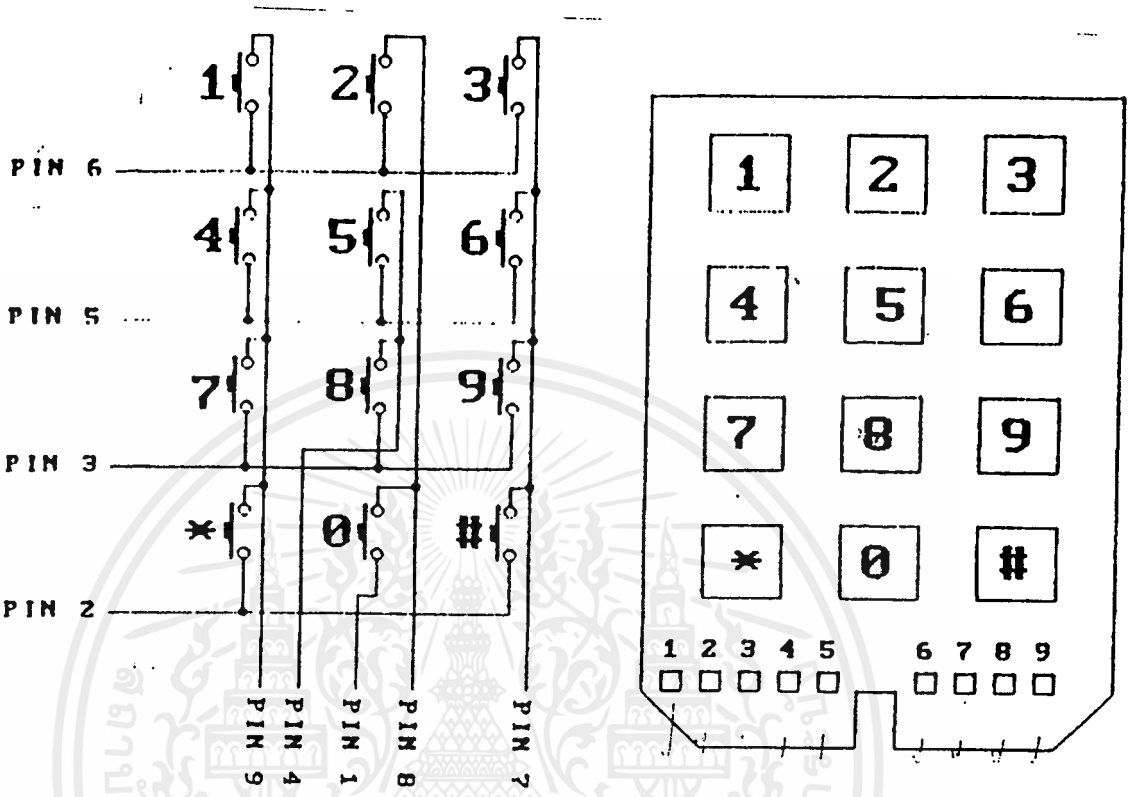
-พื้นที่บางส่วนของหน่วยความจำ หรือทรัพยากรบางส่วนของระบบถูกใช้โดย DEBUGGER

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ค.

KEYBOARD 3*4

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



KEY 4X3 DECODINGS

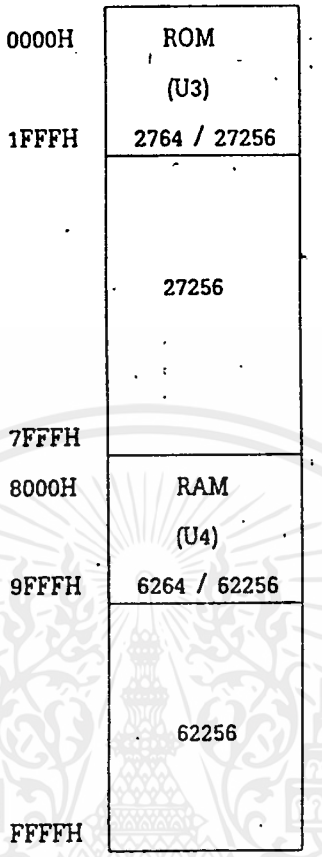
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ง.

ส่วนประกอบของ CONTROL BOARD Z-80

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

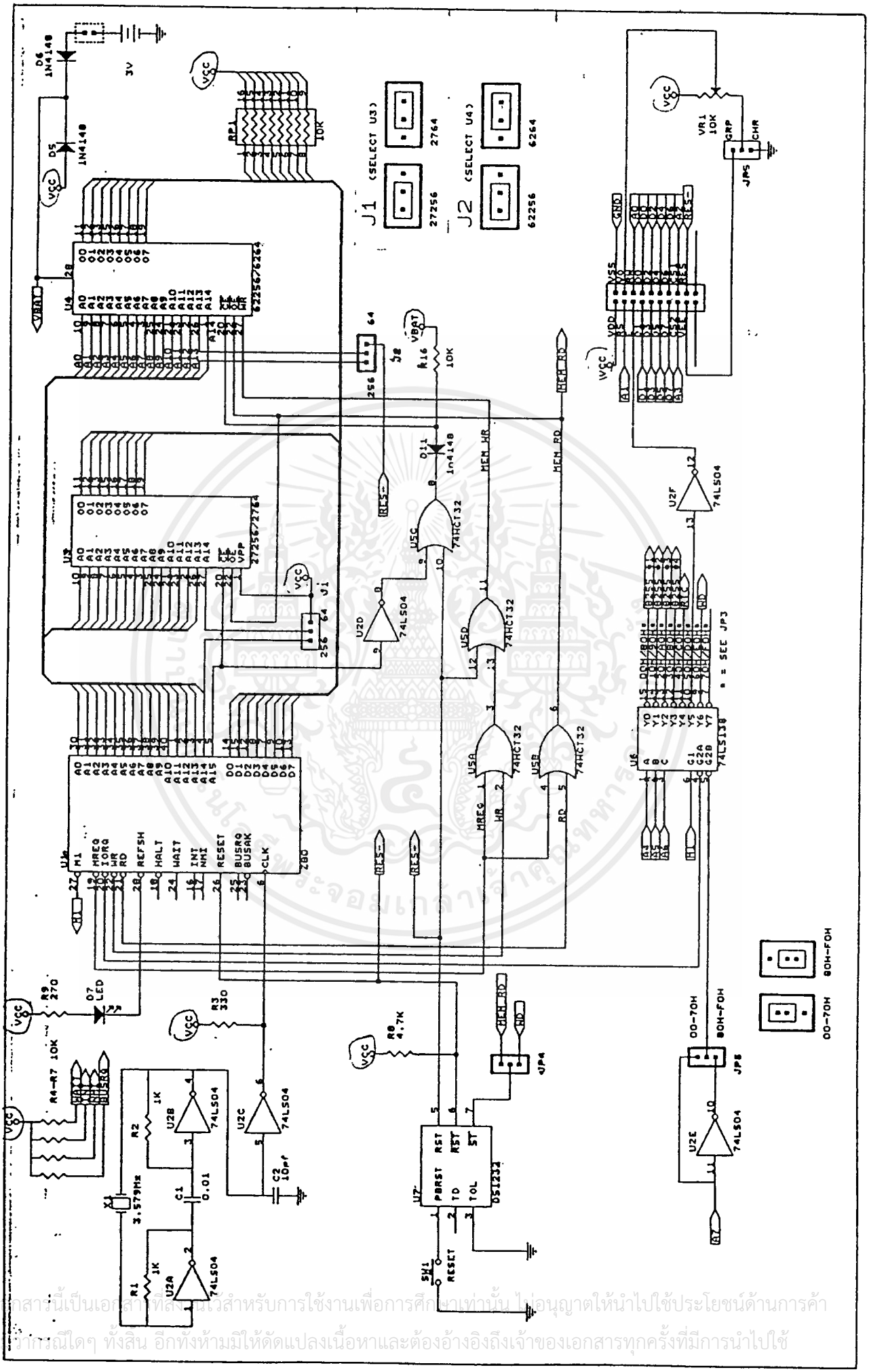
MEMORY MAP



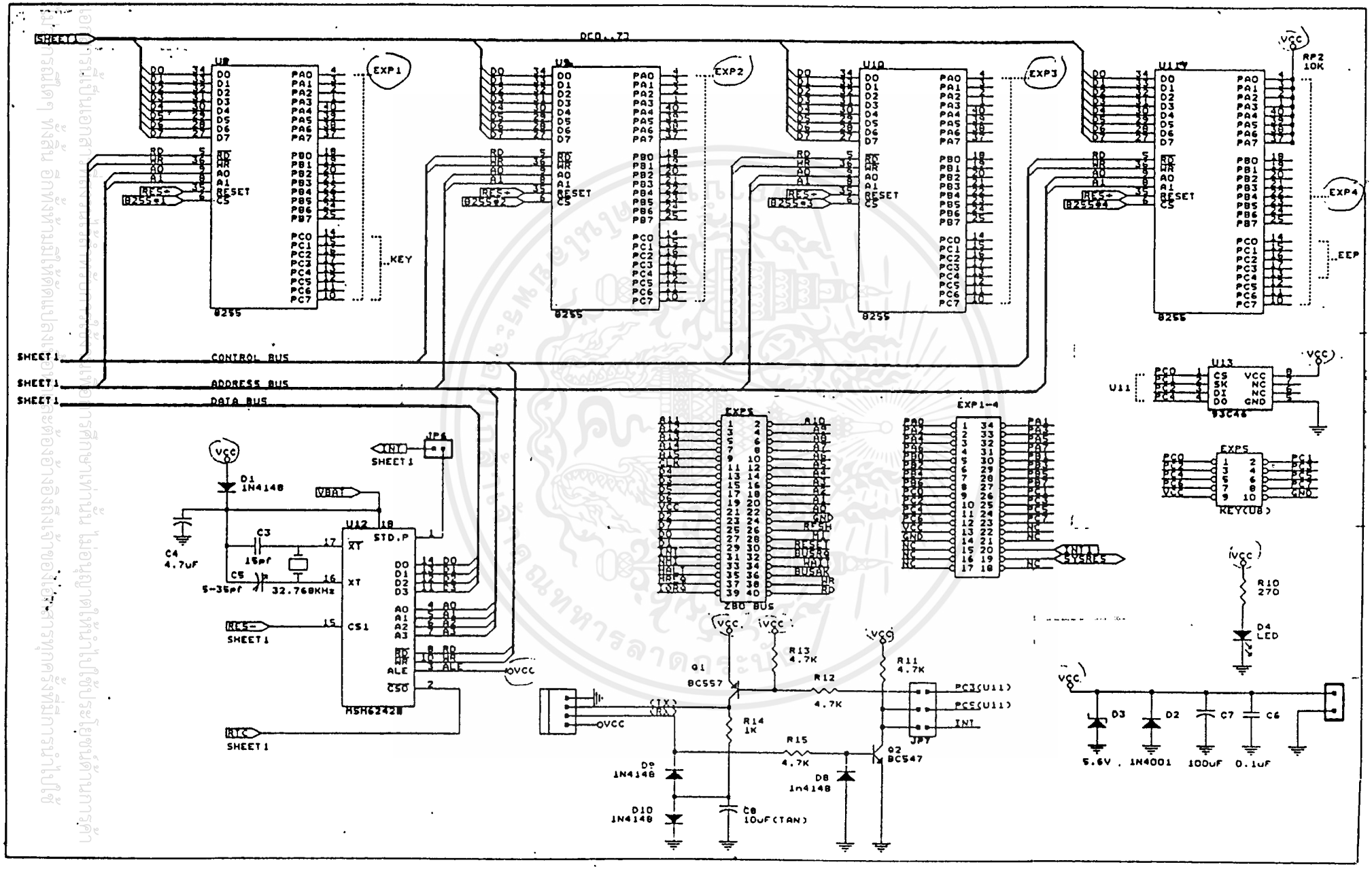
I/O MAP

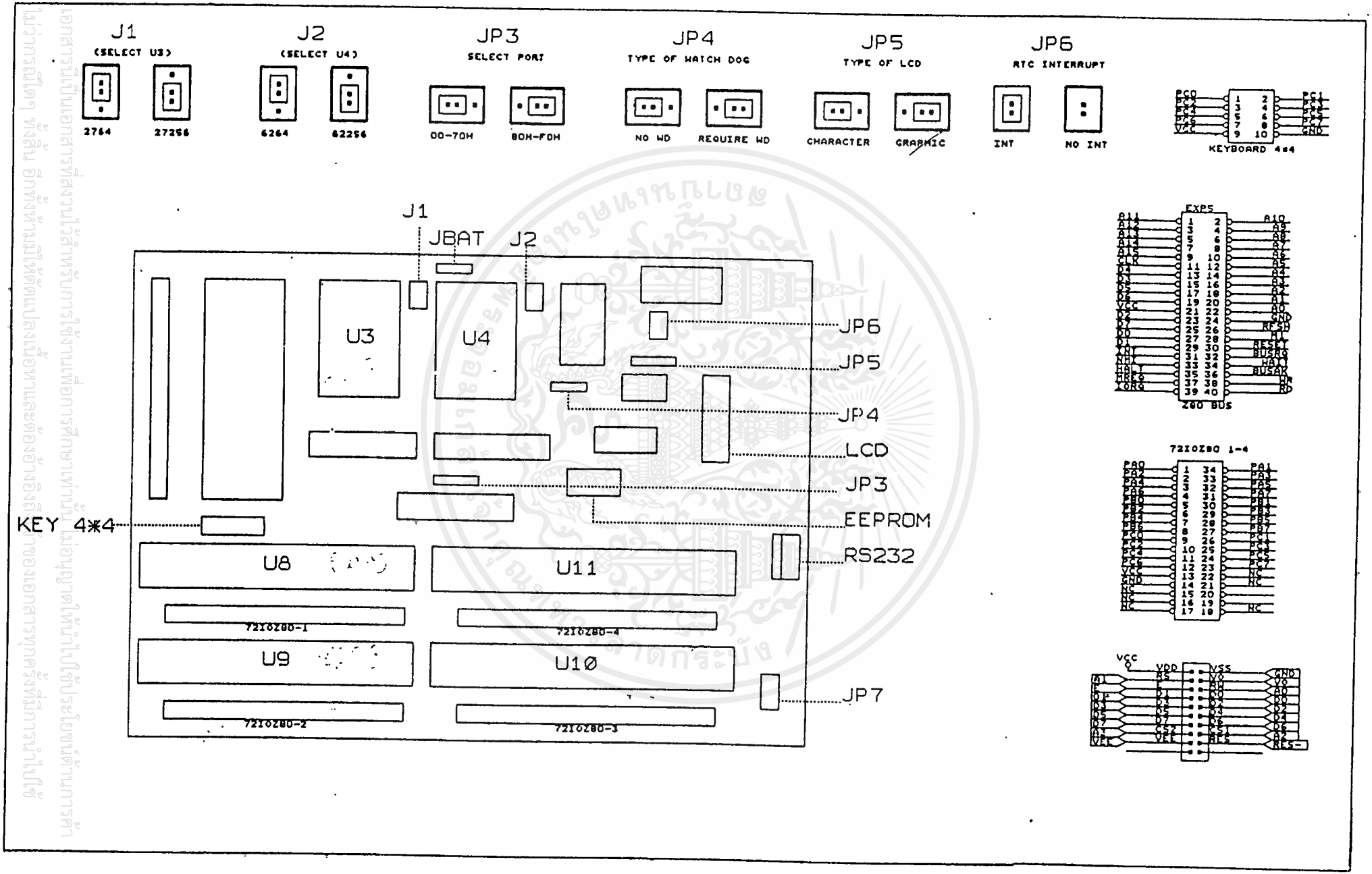
JP3		DISCRIPTION
00H	80H	8255 U8 General purpose KEY BOARD (PC0-PC7)
10H	90H	8255 U9 General purpose
20H	A0H	8255 U10 General purpose
30H	B0H	8255 U11 General purpose EEPROM,RS232
40H	C0H	RTC (C0H-CFH)
50H	D0H	LCD Character/Graphics
60H	E0H	Watch dog
70H	F0H	Reserve

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

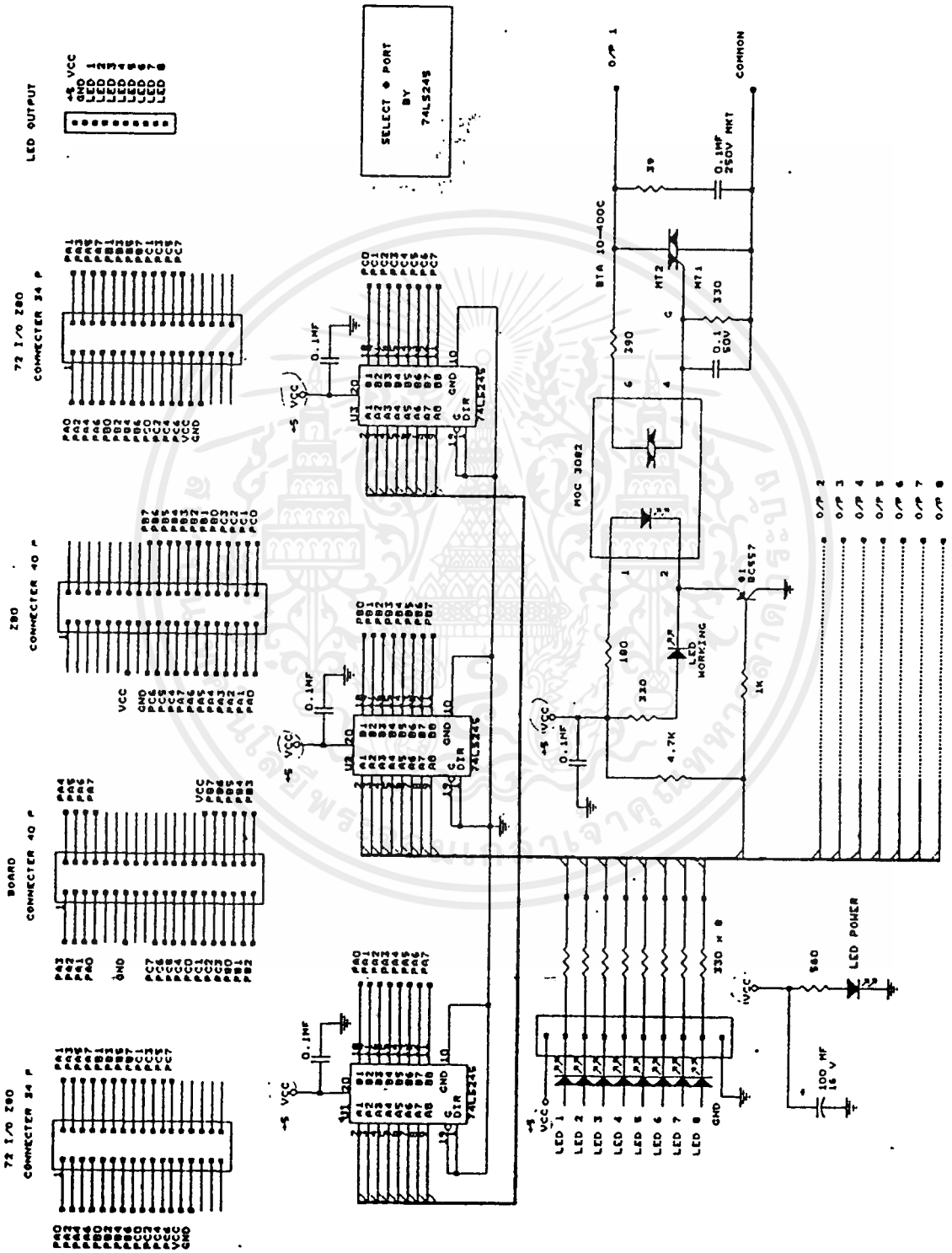

SHEET1
PC0..7
U8
U9
U10
U11
8255
EXP1
EXP2
EXP3
EXP4
KEY
EEP
RP2
10K
VCC
CONTROL BUS
ADDRESS BUS
DATA BUS
U13
93C46
EXP5
KEY(U8)
EXP1-4
Z80 BUS
U12
STD.P
HSH62428
D1
1N4148
VBAT
C3
15pf
C4
4.7uF
C5
5-35pf
32.768KHz
JP6
INT
OVCC
Q1
BC557
Q2
BC547
R13
4.7K
R12
4.7K
R11
4.7K
R14
1K
R15
4.7K
PC3(U11)
PC5(U11)
INT
JP7
D9
1N4148
D10
1N4148
D8
1n4148
C8
10uF(TAN)
R10
270
D4
LED
D3
5.6V
D2
1N4001
C7
100uF
C6
0.1uF


J1
(SELECT U3)
2764
27256
J2
(SELECT U4)
6264
62256
JP3
SELECT PORT
00-70H
80H-F0H
JP4
TYPE OF WATCH DOG
NO WD
REQUIRE WD
JP5
TYPE OF LCD
CHARACTER
GRAPHIC
JP6
RTC INTERRUPT
INT
NO INT
KEYBOARD 4*4
J1
JBAT
J2
U3
U4
JP6
JP5
JP4
LCD
JP3
EEPROM
KEY 4*4
RS232
U8
U11
U9
U10
JP7
7210Z80-1
7210Z80-4
7210Z80-2
7210Z80-3
EXP5
Z80 BUs
7210Z80 1-4
VCC
GND

ภาคผนวก จ.

ส่วนประกอบของ SSR-AC

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

SPECIFICATIONS

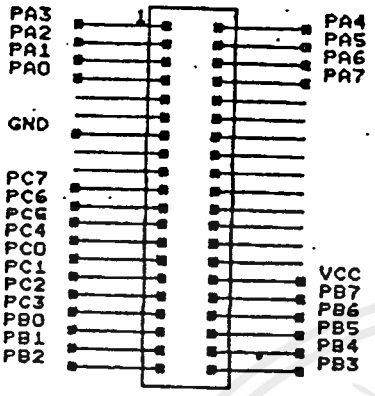
HARDWARE

NUMBER OF MODULES	:	10
OUTPUT MODULE CURRENT RATING	:	6A AT 220V AC
OPERATE TIME	:	20 ms MAX
ISOLATION	:	7500V AC MIN
POWER CONSUMPTION 5 VOITS (BOARD ONLY)	:	250 mA
LOGIC LEVEL TO "TURN ON"	:	"0" (LOW)
NUMBER OF LEDS	:	8 RED FOR INPUT
TRANSITER	:	8 BC557
TTL IC	:	1
OPTO IC	:	8
TRIACS	:	8
CONNECTOR	:	1 (40 PIN-HEADER STRIP FOR 8255 BOARD
	:	1 (40 PIN-HEADER STRIP FOR 8255
	:	2 (34 PIN-HEADER STRIP
DIMENSIONS	:	16cm L X 13cm W X 2.5 cm H

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

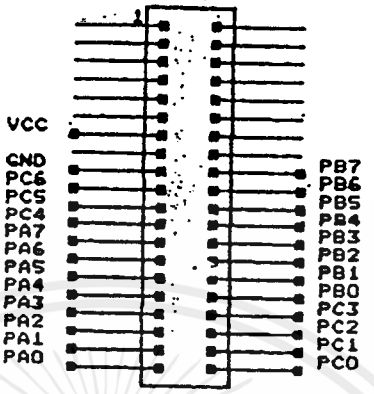
CONNECTOR PINOUT

40 PIN



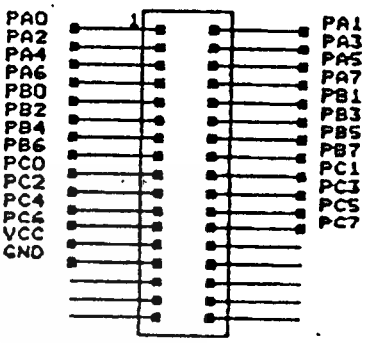
8255

40 PIN

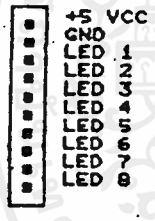


8255

34 PIN



10 PIN



* LED O/P

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รายการอุปกรณ์ SSRAC

IC	:	MOC 3082	= 8	ตัว	SOCKET	:	8 PIN	= 8	ตัว
		74LS245	= 1	ตัว			20 PIN	= 3	ตัว
R	:	330 1/4 W.	= 24	ตัว	C	:	0.1uF MUTI	= 11	ตัว
		180 1/4 W.	= 8	ตัว			0.1uF 50V.	= 8	ตัว
		390 1/4 W.	= 8	ตัว			0.1uF 250V AC	= 8	ตัว
		1K 1/4 W.	= 8	ตัว			100uF 16V.	= 1	ตัว
		4.7K 1/4 W.	= 8	ตัว					
		560 1/4 W.	= 1	ตัว					
		39 1 W.	= 8	ตัว					

เบ็ดเตล็ด

แผ่น PRINT SSRAC	=	1	แผ่น
HEADER STRIP 40 PIN	=	2	ตัว
HEADER STRIP 34 PIN	=	2	ตัว
TERMINAL 10 จุด	=	1	ตัว
LED	=	9	ตัว
TR BC557	=	8	ตัว
หมุดยึด PCB	=	4	ตัว
BTA 06600S	=	8	ตัว
HEAT SINK	=	8	ตัว
น็อต HEAT SINK	=	8	ตัว
น็อต TERMINAL	=	2	ตัว
พลาสติก TERMINAL	=	1	แผ่น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ฉ.

โปรแกรมการใช้งาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
0000                    CPU     "Z80.TBL"       ;USE Z80 TABLE
0000                    HOF     "INT8"          ;HEX Output formate

0080 =          PORT1   EQU     80H
0090 =          PORT2   EQU     90H
00A0 =          PORT3   EQU     0A0H
00B0 =          PORT4   EQU     0B0H
00C0 =          RTC     EQU     0C0H

0000                    ORG     0000H

0000            POWER_ON:
0000 AF                 XOR     A
0001 3D         LOOP:   DEC     A
0002 20FD               JR      NZ,LOOP
0004 31009F             LD      SP,9F00H

0007            SET8255:
                ;****************************************
                ;Set 8255 #1  PORT A= OUT , PORT B = OUT
                ;PORT C(UP) = OUT , PORT C(LOW) = IN
                ;****************************************
0007 3E88               LD      A,88H
0009 D383               OUT     (PORT1+3),A
                ;****************************************
                ;Set 8255 #2,3,4 PORT A,B,C = OUT
                ;****************************************
000B 3E80               LD      A,80H
000D D393               OUT     (PORT2+3),A
000F D3A3               OUT     (PORT3+3),A
0011 D3B3               OUT     (PORT4+3),A

                ;****************************************
                ;Clera PORT Control SSR. (OFF LIGHT)
                ;****************************************
0013 3EFF               LD      A,0FFH
0015 D390               OUT     (PORT2),A
0017 D391               OUT     (PORT2+1),A
0019 D392               OUT     (PORT2+2),A
001B D3A0               OUT     (PORT3),A
001D D3A1               OUT     (PORT3+1),A
001F D3A2               OUT     (PORT3+2),A
0021 D3B0               OUT     (PORT4),A
0023 D3B1               OUT     (PORT4+1),A
0025 D3B2               OUT     (PORT4+2),A
                ;****************************************
                ;******CLEAR BUFFER DISPLAY*************
                ;****************************************
0027 0605               LD      B,05H
0029 210080             LD      HL,8000H
002C 3EFF               LD      A,0FFH
002E            LOOP2:
002E 77                 LD      (HL),A
002F 23                 INC     HL
0030 10FC               DJNZ    LOOP2
                ;****************************************
                ;***Clear Address config display & sw***
                ;****************************************
0032 0604               LD      B,04H
0034 210081             LD      HL,8100H
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
0037 3E00               LD      A,00H
0039                CLEAR_LOOP:
0039 77                 LD      (HL),A
003A 23                 INC     HL
003B 10FC               DJNZ    CLEAR_LOOP
                    ;****************************************
                    ;************SCAN SWITCH*****************
                    ;****************************************
003D                SW_SCAN:
003D 0604               LD      B,04H           ;SET COUNTER ROW SW.
003F 0E01               LD      C,01H
0041                LOOP_SW1:
0041 79                 LD      A,C
0042 D382               OUT     (PORT1+2),A     ;OUT ROW SW.
0044 DB82               IN      A,(PORT1+2)     ;INPUT SW. DATA
0046 E670               AND     70H             ;Clrae INPUT ROW SW.
0048 FE00               CP      00H             ;CKD. DATA INPUT
004A C48500             CALL    NZ,INDATA       ;Have DATA CALL INDATA
004D CB01               RLC     C               ;Shift ROW SW. Active
004F 10F0               DJNZ    LOOP_SW1
0051 210081             LD      HL,8100H        ;HL =
0054 7E                 LD      A,(HL)
0055 FE00               CP      00H
0057 2008               JR      NZ,NO_LOGO
0059 CD1503             CALL    LOGO_DISPLAY
005C CD6003             CALL    DISPLAY
005F 1816               JR      END_DISPLAY
0061                NO_LOGO:
0061 FE01               CP      01H
0063 2008               JR      NZ,NO_CLOCK
0065 CD2903             CALL    CLOCK_DISPLAY
0068 CD6003             CALL    DISPLAY
006B 180A               JR      END_DISPLAY
006D                NO_CLOCK:
006D FE02               CP      02H
006F 20CC               JR      NZ,SW_SCAN
0071 CD4A03             CALL    SCORE_DISPLAY
0074 CD6003             CALL    DISPLAY
0077                END_DISPLAY:
0077 CD8603             CALL    DELAY
007A CD7A03             CALL    CLEAR
007D CD6003             CALL    DISPLAY
0080 CD8603             CALL    DELAY
0083 18B8               JR      SW_SCAN

0085                INDATA:
0085 57                 LD      D,A             ;Load DATA TO D.
0086 DB82               IN      A,(PORT1+2)     ;CKD. INPUT DATA
0088 E670               AND     70H
008A BA                 CP      D
008B 28F8               JR      Z,INDATA        ;KEY ACTIVE LOOP
008D 79                 LD      A,C             ;Load A = ROW SW.
008E FE01               CP      01H             ;CKD  ROW 0
0090 2004               JR      NZ,NO_ROW0
0092 CDAE00             CALL    KEY_ROW0        ;CALL A = ROW 0
0095 C9                 RET
0096                NO_ROW0:
0096 FE02               CP      02H             ;CKD ROW 1
0098 2004               JR      NZ,NO_ROW1
009A CDC700             CALL    KEY_ROW1        ;CALL A = ROW 1
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
009D C9                 RET
009E            NO_ROW1:
009E FE04               CP      04H             ;CKD ROW 2
00A0 2004               JR      NZ,NO_ROW2
00A2 CDE000             CALL    KEY_ROW2        ;CALL A = ROW 2
00A5 C9                 RET
00A6            NO_ROW2:
00A6 FE08               CP      08H             ;CKD ROW 3
00A8 2003               JR      NZ,NO_ROW3
00AA CDF900             CALL    KEY_ROW3        ;CALL A = ROW 3
00AD            NO_ROW3:
00AD C9                 RET

00AE            KEY_ROW0:
00AE 7A                 LD      A,D             ;Load A = DATA IN
00AF FE40               CP      40H             ;CKD SW 1
00B1 2004               JR      NZ,NO0_1        ;JUMP IF NOT SW1
00B3 CD1201             CALL    SW1             ;SET A = 1
00B6 C9                 RET
00B7 FE20       NO0_1:  CP      20H             ;CKD SW 2
00B9 2004               JR      NZ,NO0_2        ;JUMP IF NOT SW2
00BB CD3B01             CALL    SW2             ;SET A = 2
00BE C9                 RET
00BF FE10       NO0_2:  CP      10H             ;CKD SW 3
00C1 2003               JR      NZ,NO0_3        ;JUMP IF NOT SW3
00C3 CD6401             CALL    SW3             ;SET A = 3
00C6 C9         NO0_3:  RET

00C7            KEY_ROW1:
00C7 7A                 LD      A,D
00C8 FE40               CP      40H             ;CKD SW 4
00CA 2004               JR      NZ,NO1_1        ;JUMP IF NOT SW4
00CC CD8D01             CALL    SW4             ;SET A = 4
00CF C9                 RET
00D0 FE20       NO1_1:  CP      20H             ;CKD SW 5
00D2 2004               JR      NZ,NO1_2        ;JUMP IF NOT SW5
00D4 CDB001             CALL    SW5             ;SET A = 5
00D7 C9                 RET
00D8 FE10       NO1_2:  CP      10H             ;CKD SW 6
00DA 2003               JR      NZ,NO1_3        ;JUMP IF NOT SW 6
00DC CDD301             CALL    SW6             ;SET A =6
00DF C9         NO1_3:  RET

00E0            KEY_ROW2:
00E0 7A                 LD      A,D
00E1 FE40               CP      40H             ;CKD SW 7
00E3 2004               JR      NZ,NO2_1        ;JUMP IF NOT SW 7
00E5 CDF601             CALL    SW7             ;SET A = 7
00E8 C9                 RET
00E9 FE20       NO2_1:  CP      20H             ;CKD SW 8
00EB 2004               JR      NZ,NO2_2        ;JUMP IF NOT SW 8
00ED CD1902             CALL    SW8             ;SET A = 8
00F0 C9                 RET
00F1 FE10       NO2_2:  CP      10H             ;CKD SW 9
00F3 2003               JR      NZ,NO2_3        ;JUMP IF NOT SW 9
00F5 CD3C02             CALL    SW9             ;SET A = 9
00F8 C9         NO2_3:  RET

00F9            KEY_ROW3:
00F9 7A                 LD      A,D
```

ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
00FA FE40               CP      40H             ;CKD SW #
00FC 2004               JR      NZ,NO3_1        ;JUMP IF NOT SW#
00FE CD8202             CALL    SWA             ;SET A =0A
0101 C9                 RET
0102 FE20       NO3_1:  CP      20H             ;CKD SW 0
0104 2004               JR      NZ,NO3_2        ;JUMP IF NOT SW 0
0106 CD5F02             CALL    SW0             ;SET A = 0
0109 C9                 RET
010A FE10       NO3_2:  CP      10H             ;CKD SW *
010C 2003               JR      NZ,NO3_3        ;JUMP IF NOT SW *
010E CDB102             CALL    SWB             ;SET A = 0B
0111 C9         NO3_3:  RET

                ;************************************************
                ;*** SWITCH NO.1 and SW SELECT CLOCK DISPLAY ***
                ;************************************************
0112            SW1:
0112 210181             LD      HL,8101H        ;HL = BUFFER CONFIG SET S
0115 7E                 LD      A,(HL)          ;LOAD DATA CONFIG TO A.
0116 CB47               BIT     0,A             ;CKD. BIT SELECT
0118 2007               JR      NZ,SET_SW1      ;SW SET ACTIVE IF JUMP
011A 210081             LD      HL,8100H        ;HL = BUFFER TYPE DISPLAY
011D 3E01               LD      A,01H           ;SET DATA TYPE DISPLAY TO
011F 77                 LD      (HL),A          ;LOAD DATA TYPE DISPLAY T
0120 C9                 RET
0121            SET_SW1:
0121 CB4F               BIT     1,A             ;CDK. COUNTER ENTER
0123 200B               JR      NZ,SECON_SETSW1 ;SECON ENTER IF JUMP
0125 DD213E04           LD      IX,NUMBER       ;IX = TABLE NUMBER
0129 DD7E01             LD      A,(IX+1)        ;OPEN TABLE DATA 1
012C CDBF02             CALL    SET1            ;CALL   PROGRAM SET1
012F C9                 RET
0130            SECON_SETSW1:
0130 DD213E04           LD      IX,NUMBER       ;IX = TABLE NUMBER
0134 DD7E01             LD      A,(IX+1)        ;OPEN TABLE DATA 1
0137 CDEA02             CALL    SET2            ;CALL   PROGRAM SET2
013A C9                 RET

                ;************************************************
                ;*** SWITCH NO. 2 AND SW. SELECT LOGO DISPLAY ****
                ;************************************************
013B            SW2:
013B 210181             LD      HL,8101H
013E 7E                 LD      A,(HL)
013F CB47               BIT     0,A
0141 2007               JR      NZ,SET_SW2
0143 210081             LD      HL,8100H
0146 3E00               LD      A,00H
0148 77                 LD      (HL),A
0149 C9                 RET
014A            SET_SW2:
014A CB4F               BIT     1,A
014C 200B               JR      NZ,SECON_SETSW2
014E DD213E04           LD      IX,NUMBER
0152 DD7E02             LD      A,(IX+2)
0155 CDBF02             CALL    SET1
0158 C9                 RET
0159            SECON_SETSW2:
0159 DD213E04           LD      IX,NUMBER
015D DD7E02             LD      A,(IX+2)
```

ลิขสิทธิ์เอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
0160 CDEA02             CALL    SET2
0163 C9                 RET

                ;************************************************
                ;*** SWITCH NO. 3 and SW. SELECT SCORE DISPLAY ***
                ;************************************************
0164            SW3:
0164 210181             LD      HL,8101H
0167 7E                 LD      A,(HL)
0168 CB47               BIT     0,A
016A 2007               JR      NZ,SET_SW3
016C 210081             LD      HL,8100H
016F 3E02               LD      A,02H
0171 77                 LD      (HL),A
0172 C9                 RET
0173            SET_SW3:
0173 CB4F               BIT     1,A
0175 200B               JR      NZ,SECON_SETSW3
0177 DD213E04           LD      IX,NUMBER
017B DD7E03             LD      A,(IX+3)
017E CDBF02             CALL    SET1
0181 C9                 RET
0182            SECON_SETSW3:
0182 DD213E04           LD      IX,NUMBER
0186 DD7E03             LD      A,(IX+3)
0189 CDEA02             CALL    SET2
018C C9                 RET
                ;****************************************
                ;********* SWITCH NUMBER 4. *************
                ;****************************************
018D            SW4:
018D 210181             LD      HL,8101H
0190 7E                 LD      A,(HL)
0191 CB47               BIT     0,A
0193 2001               JR      NZ,SET_SW4
0195 C9                 RET
0196            SET_SW4:
0196 CB4F               BIT     1,A
0198 200B               JR      NZ,SECON_SETSW4
019A DD213E04           LD      IX,NUMBER
019E DD7E04             LD      A,(IX+4)
01A1 CDBF02             CALL    SET1
01A4 C9                 RET
01A5            SECON_SETSW4:
01A5 DD213E04           LD      IX,NUMBER
01A9 DD7E04             LD      A,(IX+4)
01AC CDEA02             CALL    SET2
01AF C9                 RET

                ;****************************************
                ;********* SWITCH NUMBER 5. *************
                ;****************************************
01B0            SW5:
01B0 210181             LD      HL,8101H
01B3 7E                 LD      A,(HL)
01B4 CB47               BIT     0,A
01B6 2001               JR      NZ,SET_SW5
01B8 C9                 RET
01B9            SET_SW5:
01B9 CB4F               BIT     1,A
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า การแสวงหากำไร ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
01BB 20E8              JR      NZ,SECON_SETSW4
01BD DD213E04          LD      IX,NUMBER
01C1 DD7E05            LD      A,(IX+5)
01C4 CDBF02            CALL    SET1
01C7 C9                RET
01C8                SECON_SETSW5:
01C8 DD213E04          LD      IX,NUMBER
01CC DD7E05            LD      A,(IX+5)
01CF CDEA02            CALL    SET2
01D2 C9                RET

                       ;*****************************************
                       ;********* SWITCH NUMBER 6. *************
                       ;*****************************************
01D3                   SW6:
01D3 210181            LD      HL,8101H
01D6 7E                LD      A,(HL)
01D7 CB47              BIT     0,A
01D9 2001              JR      NZ,SET_SW6
01DB C9                RET
01DC                   SET_SW6:
01DC CB4F              BIT     1,A
01DE 20C5              JR      NZ,SECON_SETSW4
01E0 DD213E04          LD      IX,NUMBER
01E4 DD7E06            LD      A,(IX+6)
01E7 CDBF02            CALL    SET1
01EA C9                RET
01EB                SECON_SETSW6:
01EB DD213E04          LD      IX,NUMBER
01EF DD7E06            LD      A,(IX+6)
01F2 CDEA02            CALL    SET2
01F5 C9                RET

                       ;*****************************************
                       ;********* SWITCH NUMBER 7. *************
                       ;*****************************************
01F6                   SW7:
01F6 210181            LD      HL,8101H
01F9 7E                LD      A,(HL)
01FA CB47              BIT     0,A
01FC 2001              JR      NZ,SET_SW7
01FE C9                RET
01FF                   SET_SW7:
01FF CB4F              BIT     1,A
0201 200B              JR      NZ,SECON_SETSW7
0203 DD213E04          LD      IX,NUMBER
0207 DD7E07            LD      A,(IX+7)
020A CDBF02            CALL    SET1
020D C9                RET
020E                SECON_SETSW7:
020E DD213E04          LD      IX,NUMBER
0212 DD7E07            LD      A,(IX+7)
0215 CDEA02            CALL    SET2
0218 C9                RET

                       ;*****************************************
                       ;********* SWITCH NUMBER 8. *************
                       ;*****************************************
0219                   SW8:
0219 210181            LD      HL,8101H
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
021C 7E                 LD      A,(HL)
021D CB47               BIT     0,A
021F 2001               JR      NZ,SET_SW8
0221 C9                 RET
0222            SET_SW8:
0222 CB4F               BIT     1,A
0224 200B               JR      NZ,SECON_SETSW8
0226 DD213E04           LD      IX,NUMBER
022A DD7E08             LD      A,(IX+8)
022D CDBF02             CALL    SET1
0230 C9                 RET
0231            SECON_SETSW8:
0231 DD213E04           LD      IX,NUMBER
0235 DD7E08             LD      A,(IX+8)
0238 CDEA02             CALL    SET2
023B C9                 RET

                ;****************************************
                ;********* SWITCH NUMBER 9. *************
                ;****************************************
023C            SW9:
023C 210181             LD      HL,8101H
023F 7E                 LD      A,(HL)
0240 CB47               BIT     0,A
0242 2001               JR      NZ,SET_SW9
0244 C9                 RET
0245            SET_SW9:
0245 CB4F               BIT     1,A
0247 200B               JR      NZ,SECON_SETSW9
0249 DD213E04           LD      IX,NUMBER
024D DD7E09             LD      A,(IX+9)
0250 CDBF02             CALL    SET1
0253 C9                 RET
0254            SECON_SETSW9:
0254 DD213E04           LD      IX,NUMBER
0258 DD7E09             LD      A,(IX+9)
025B CDEA02             CALL    SET2
025E C9                 RET

                ;****************************************
                ;********* SWITCH NUMBER 0. *************
                ;****************************************
025F            SW0:
025F 210181             LD      HL,8101H
0262 7E                 LD      A,(HL)
0263 CB47               BIT     0,A
0265 2001               JR      NZ,SET_SW0
0267 C9                 RET
0268            SET_SW0:
0268 CB4F               BIT     1,A
026A 200B               JR      NZ,SECON_SETSW0
026C DD213E04           LD      IX,NUMBER
0270 DD7E00             LD      A,(IX+0)
0273 CDBF02             CALL    SET1
0276 C9                 RET
0277            SECON_SETSW0:
0277 DD213E04           LD      IX,NUMBER
027B DD7E00             LD      A,(IX+0)
027E CDEA02             CALL    SET2
0281 C9                 RET
```

เอกสารนี้สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                ;***********************************************
                ;************** SWITCH ENTER *******************
                ;***********************************************
0282            SWA:
0282 210181             LD      HL,8101H
0285 7E                 LD      A,(HL)
0286 CB47               BIT     0,A             ;CKD. BIT SET DATA (= 1
0288 2026               JR      NZ,END_SWA      ;BIT 0 = 1 , Z NO SET
028A CB4F               BIT     1,A             ;CKD. BIT FIRST ENTER
028C 2003               JR      NZ,NO_FIRST     ;JUMP BIT 1 = 1 (SECON )
028E CBCF               SET     1,A             ;SET FIRST ENTER
0290 C9                 RET
0291            NO_FIRST:
0291 CB8F               RES     1,A             ;CLEAR BUFFER ENTER KEY
0293 3E00               LD      A,00H
0295 320181             LD      (8101H),A       ;CLEAR BUFFER SET KEY
0298 210081             LD      HL,8100H        ;HL = DATA TYPE DISPLAY
029B 7E                 LD      A,(HL)          ;LOAD A FORM DATA TYPE
029C FE01               CP      01H             ;CKD CLOCK DISPLAY
029E 200A               JR      NZ,AAA
02A0 CD9203             CALL    WRITE_CLOCK     ;CALL PROGRAM WRITE CLO
02A3 210480             LD      HL,8004H
02A6 3EBF               LD      A,0BFH
02A8 77                 LD      (HL),A
02A9 C9                 RET
02AA            AAA:
02AA 210480             LD      HL,8004H
02AD 3E7F               LD      A,7FH
02AF 77                 LD      (HL),A

02B0            END_SWA:
02B0 C9                 RET

                ;***************************************
                ;********** SWITCH SET *****************
                ;***************************************
02B1            SWB:
02B1 210181             LD      HL,8101H
02B4 7E                 LD      A,(HL)
02B5 F601               OR      01H
02B7 77                 LD      (HL),A
02B8 210480             LD      HL,8004H
02BB 3E3F               LD      A,3FH
02BD 77                 LD      (HL),A
02BE C9                 RET

                ;*****************************************;********** PR
                ;***************************************
02BF            SET1:
02BF 5F                 LD      E,A             ;LOAD DATA TO E.
02C0 210081             LD      HL,8100H        ;HL = CONFIG TYPE DISPL
02C3 FE01               CP      01H             ;CKD TYPE CLOCK DISPLAY
02C5 2805               JR      Z,SET_CLOCK1    ;JP. IT CLOCK DISPLAY
02C7 FE02               CP      02H             ;CKD. TYPE SCORE DISPLA)
02C9 2810               JR      Z,SET_SCORE1    ;JP IT SCORE DISPLAY
02CB C9                 RET
02CC            SET_CLOCK1:
02CC DD211080           LD      IX,8010H        ;IX = BUFFER CLOCK DATA
02D0 DD7E01             LD      A,(IX+1)        ;SHIFT LEFT DATA 1 BYTE
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้ง

```
02D3 DD7700             LD      (IX+0),A
02D6 7B                 LD      A,E
02D7 DD7701             LD      (IX+1),A        ;LOAD NEW DATA TO LSB.
02DA C9                 RET
02DB                SET_SCORE1:
02DB DD212080           LD      IX,8020H        ;IX = BUFFER SCORE DATA
02DF DD7E01             LD      A,(IX+1)        ;SHIFT LEFT DATA 1 BYTE
02E2 DD7700             LD      (IX+0),A
02E5 7B                 LD      A,E
02E6 DD7701             LD      (IX+1),A        ;LOAD NEW DATA TO LSB.
02E9 C9                 RET

                    ;*****************************************;********** PRO(
                    ;*************************************
02EA                SET2:
02EA 5F                 LD      E,A             ;LOAD DATA TO E.
02EB 210081             LD      HL,8100H        ;HL = CONFIG TYPE DISPLA)
02EE FE01               CP      01H             ;CKD TYPE CLOCK DISPLAY
02F0 2805               JR      Z,SET_CLOCK2    ;JP. IT CLOCK DISPLAY
02F2 FE02               CP      02H             ;CKD. TYPE SCORE DISPLAY
02F4 2810               JR      Z,SET_SCORE2    ;JP IT SCORE DISPLAY
02F6 C9                 RET
02F7                SET_CLOCK2:
02F7 DD211280           LD      IX,8012H        ;IX = BUFFER CLOCK DATA
02FB DD7E01             LD      A,(IX+1)        ;SHIFT LEFT DATA 1 BYTE
02FE DD7700             LD      (IX+0),A
0301 7B                 LD      A,E
0302 DD7701             LD      (IX+1),A        ;LOAD NEW DATA TO LSB.
0305 C9                 RET
0306                SET_SCORE2:
0306 DD212280           LD      IX,8022H        ;IX = BUFFER SCORE DATA
030A DD7E01             LD      A,(IX+1)        ;SHIFT LEFT DATA 1 BYTE
030D DD7700             LD      (IX+0),A
0310 7B                 LD      A,E
0311 DD7701             LD      (IX+1),A        ;LOAD NEW DATA TO LSB.
0314 C9                 RET

                    ;*************************************
                    ;PROGRAM LOAD LOGO DATA TO BUFFER DISPLAY
                    ;*************************************
0315                LOGO_DISPLAY:
0315 210080             LD      HL,8000H        ;HL = BUFFER DISPLAY
0318 DD213904           LD      IX,SEIKO        ;IX = TABLE DATA off LOG(
031C 1E05               LD      E,05H
031E                LOOP7:
031E DD7E00             LD      A,(IX+0)        ;COPY DATA LOGO TO BUFFE)
0321 77                 LD      (HL),A          ;DISPLAY
0322 DD23               INC     IX
0324 23                 INC     HL
0325 1D                 DEC     E
0326 20F6               JR      NZ,LOOP7
0328 C9                 RET

                    ;*****************************************
                    ;PROGRAM LOAD CLOCK DATA TO BUFFER DISPLAY
                    ;*****************************************
0329                CLOCK_DISPLAY:
0329 210181             LD      HL,8101H
032C 7E                 LD      A,(HL)
032D CB47               BIT     0,A
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
032F 2803              JR      Z,NO_SETCLOCK.
0331 CDB003            CALL    READ_CLOCK
0334                NO_SETCLOCK:
0334 210080            LD      HL,8000H        ;HL = ADDRESS BUFFER DIS
0337 DD211080          LD      IX,8010H        ;IX = ADDRESS BUFFER CLO
033B 0604              LD      B,04H           ; B = COUNTER CLOCK DATA
033D                CLOCK_LOOP:
033D DD7E00            LD      A,(IX+0)        ;LOAD DATA CLOCK TO A.
0340 77                LD      (HL),A          ;LOAD A. TO BUFFER DISPL
0341 DD23              INC     IX              ;INC. INDEX BUFFER CLOCK
0343 23                INC     HL              ;INC. INDEX BUFFER DISPL
0344 10F7              DJNZ    CLOCK_LOOP
0346 3E7F              LD      A,7FH
0348 77                LD      (HL),A
0349 C9                RET

                    ;*********************************************
                    ;PROGRAM LOAD SCORE DATA TO BUFFER DISPLAY
                    ;*********************************************
034A                SCORE_DISPLAY:
034A 210080            LD      HL,8000H        ;HL = ADDRESS BUFFER DIS
034D DD212080          LD      IX,8020H        ;IX = ADDRESS BUFFER SCO
0351 0604              LD      B,04H           ; B = COUNTER SCORE DATA
0353                CSORE_LOOP:
0353 DD7E00            LD      A,(IX+0)        ;LOAD DATA SCORE TO A.
0356 77                LD      (HL),A          ;LOAD A. TO BUFFER DISPL
0357 DD23              INC     IX              ;INC. INDEX BUFFER SCORE
0359 23                INC     HL              ;INC. INDEX BUFFER DISPL
035A 10F7              DJNZ    CSORE_LOOP
035C 3EBF              LD      A,0BFH
035E 77                LD      (HL),A
035F C9                RET

                    ;*****************************************
                    ;************DISPLAY PROGRAM**************
                    ;*****************************************
0360                DISPLAY:
0360 210080            LD      HL,8000H        ;HL=START BUFFER ADDRESS
0363 0603              LD      B,03H
0365 0E90              LD      C,PORT2         ;C = ADDRESS PORT 8255#2
0367                DISPLAY1:
0367 7E                LD      A,(HL)          ; A =DATA IN BUFFER
0368 ED79              OUT     (C),A           ;OUT DATA TO PORT
036A 0C                INC     C               ;INC PORT
036B 23                INC     HL              ;INC BUFFER DATA
036C 10F9              DJNZ    DISPLAY1
036E 0602              LD      B,02H
0370 0EA0              LD      C,PORT3         ;C = ADDRESS PORT 8255#3
0372                DISPLAY2:
0372 7E                LD      A,(HL)
0373 ED79              OUT     (C),A
0375 0C                INC     C
0376 23                INC     HL
0377 10F9              DJNZ    DISPLAY2
0379 C9                RET

                    ;*************************************
                    ;***PROGRAM CLEAR DATA IN DUFFER*****
                    ;*************************************
037A                CLEAR:
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งไม่อนุญาตให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
037A 0605              LD      B,05H           ;B = COUNTER BUFFER DISI
037C 210080            LD      HL,8000H        ;HL = START ADDRESS BUFI
037F 3EFF              LD      A,0FFH          ;A = DATA OFF LIGHT
0381                LOOP_CLEAR:
0381 77             Z  LD      (HL),A          ;LOAD A. TO BUFFER DISPI
0382 23                INC     HL              ;INC. INDEX BUFFER DISPI
0383 10FC              DJNZ    LOOP_CLEAR
0385 C9                RET

                    ;************************************
                    ;****** DELAY TIME PROGRAM ********
                    ;************************************
0386                DELAY:
0386 D9                EXX
0387 06C0              LD      B,0C0H
0389                DELAY_LOOP2:
0389 0E00              LD      C,00H
038B                DELAY_LOOP1:
038B 0D                DEC     C
038C 20FD              JR      NZ,DELAY_LOOP1
038E 10F9              DJNZ    DELAY_LOOP2
0390 D9                EXX
0391 C9                RET

                    ;*******************************************
                    ;******** PROGRAM WRITE DATA TO RTC.********
                    ;*******************************************
0392                WRITE_CLOCK:
0392 DD211080          LD      IX,8010H        ;IX = BUFFER CLOCK DATA
0396 DD7E00            LD      A,(IX+0)        ;LOAD A. = HSB HOR
0399 0EC5              LD      C,RTC+5         ;C = PORT RTC.
039B ED79              OUT     (C),A
039D DD7E01            LD      A,(IX+1)        ;A. = LSB HOR
03A0 0D                DEC     C
03A1 ED79              OUT     (C),A
03A3 DD7E02            LD      A,(IX+2)        ;A. = HSB Min
03A6 0D                DEC     C
03A7 ED79              OUT     (C),A
03A9 DD7E03            LD      A,(IX+3)        ;A. = LSB Min
03AC 0D                DEC     C
03AD ED79              OUT     (C),A
03AF C9                RET

                    ;*******************************************
                    ;******** PROGRAM READ DATA FROM RTC.*******
                    ;*******************************************
03B0                READ_CLOCK:

03B0 DD211080          LD      IX,8010H        ;IX = BUFFER CLOCK DATA
03B4 0EC5              LD      C,RTC+5         ;C = PORT RTC.
03B6 ED78              IN      A,(C)           ;IN DATA HSB HOR.
03B8 E60F              AND     0FH
03BA CDE203            CALL    TABLE
03BD DD7700            LD      (IX+0),A        ;LOAD DATA TO BUFFER
03C0 0D                DEC     C
03C1 ED78              IN      A,(C)           ;IN DATA LSB HOR.
03C3 E60F              AND     0FH
03C5 CDE203            CALL    TABLE
03C8 DD7701            LD      (IX+1),A        ;LOAD DATA TO BUFFER
03CB 0D                DEC     C
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
03CC ED78              IN      A,(C)           ;IN DATA LSB HOR.
03CE E60F              AND     0FH
03D0 CDE203            CALL    TABLE
03D3 DD7702            LD      (IX+2),A        ;LOAD DATA TO BUFFER
03D6 0D                DEC     C
03D7 ED78              IN      A,(C)           ;IN DATA LSB Min.
03D9 E60F              AND     0FH
03DB CDE203            CALL    TABLE
03DE DD7703            LD      (IX+3),A
03E1 C9                RET
                       ;*********************************
                       ;******* READ DATA FRO TABLE ******
                       ;*********************************
03E2                   TABLE:
03E2 DD213E04                  LD      IX,NUMBER
03E6 FE00                      CP      00H
03E8 2004                      JR      NZ,NO_0
03EA DD7E00                    LD      A,(IX+0)
03ED C9                        RET
03EE FE01              NO_0:   CP      01H
03F0 2004                      JR      NZ,NO_1
03F2 DD7E01                    LD      A,(IX+1)
03F5 C9                        RET
03F6 FE02              NO_1:   CP      02H
03F8 2004                      JR      NZ,NO_2
03FA DD7E02                    LD      A,(IX+2)
03FD C9                        RET
03FE FE03              NO_2:   CP      03H
0400 2004                      JR      NZ,NO_3
0402 DD7E03                    LD      A,(IX+3)
0405 C9                        RET
0406 FE04              NO_3:   CP      04H
0408 2004                      JR      NZ,NO_4
040A DD7E04                    LD      A,(IX+4)
040D C9                        RET
040E                   NO_4:
040E FE05                      CP      05H
0410 2004                      JR      NZ,NO_5
0412 DD7E05                    LD      A,(IX+5)
0415 C9                        RET
0416 FE06              NO_5:   CP      06H
0418 2004                      JR      NZ,NO_6
041A DD7E06                    LD      A,(IX+6)
041D C9                        RET
041E FE07              NO_6:   CP      07H
0420 2004                      JR      NZ,NO_7
0422 DD7E07                    LD      A,(IX+7)
0425 C9                        RET
0426 FE08              NO_7:   CP      08H
0428 2004                      JR      NZ,NO_8
042A DD7E08                    LD      A,(IX+8)
042D C9                        RET
042E FE09              NO_8:   CP      09H
0430 2004                      JR      NZ,NO_9
0432 DD7E09                    LD      A,(IX+9)
0435 C9                        RET
0436 3EFF              NO_9:   LD      A,0FFH
0438 C9                        RET

0439 92864FC03FSEIKO:  DFB     92H,86H,4FH,0C0H,3FH
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
043E C0F9A4B099NUMBER: DFB     0C0H,0F9H,0A4H,0B0H,99H
0443 9282F88090        DFB     92H,82H,0F8H,80H,90H

0448                   END     :


0000                   END
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

02AA	AAA	037A	CLEAR	0039	CLEAR_LOOP
0329	CLOCK_DISPLAY	033D	CLOCK_LOOP	0353	CSORE_LOOP
0386	DELAY	038B	DELAY_LOOP1	0389	DELAY_LOOP2
0360	DISPLAY	0367	DISPLAY1	0372	DISPLAY2
0448	END	0077	END_DISPLAY	02B0	END_SWA
0085	INDATA	00AE	KEY_ROW0	00C7	KEY_ROW1
00E0	KEY_ROW2	00F9	KEY_ROW3	0315	LOGO_DISPLAY
0001	LOOP	002E	LOOP2	031E	LOOP7
0381	LOOP_CLEAR	0041	LOOP_SW1	00B7	NO0_1
00BF	NO0_2	00C6	NO0_3	00D0	NO1_1
00D8	NO1_2	00DF	NO1_3	00E9	NO2_1
00F1	NO2_2	00F8	NO2_3	0102	NO3_1
010A	NO3_2	0111	NO3_3	03EE	NO_0
03F6	NO_1	03FE	NO_2	0406	NO_3
040E	NO_4	0416	NO_5	041E	NO_6
0426	NO_7	042E	NO_8	0436	NO_9
006D	NO_CLOCK	0291	NO_FIRST	0061	NO_LOGO
0096	NO_ROW0	009E	NO_ROW1	00A6	NO_ROW2
00AD	NO_ROW3	0334	NO_SETCLOCK	043E	NUMBER
0080	PORT1	0090	PORT2	00A0	PORT3
00B0	PORT4	0000	POWER_ON	03B0	READ_CLOCK
00C0	RTC	034A	SCORE_DISPLAY	0277	SECON_SETSW0
0130	SECON_SETSW1	0159	SECON_SETSW2	0182	SECON_SETSW3
01A5	SECON_SETSW4	01C8	SECON_SETSW5	01EB	SECON_SETSW6
020E	SECON_SETSW7	0231	SECON_SETSW8	0254	SECON_SETSW9
0439	SEIKO	02BF	SET1	02EA	SET2
0007	SET8255	02CC	SET_CLOCK1	02F7	SET_CLOCK2
02DB	SET_SCORE1	0306	SET_SCORE2	0268	SET_SW0
0121	SET_SW1	014A	SET_SW2	0173	SET_SW3
0196	SET_SW4	01B9	SET_SW5	01DC	SET_SW6
01FF	SET_SW7	0222	SET_SW8	0245	SET_SW9
025F	SW0	0112	SW1	013B	SW2
0164	SW3	018D	SW4	01B0	SW5
01D3	SW6	01F6	SW7	0219	SW8
023C	SW9	0282	SWA	02B1	SWB
003D	SW_SCAN	03E2	TABLE	0392	WRITE_CLOCK
0381	Z				

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้