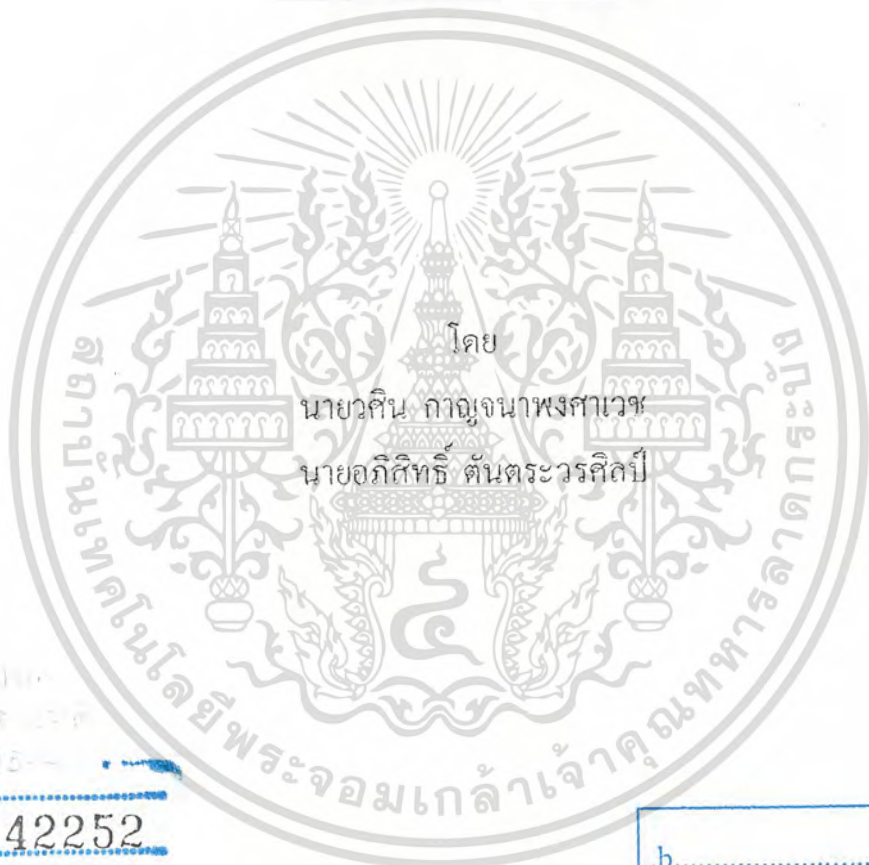


เครื่องโปรแกรมสวิทช์หลายช่อง
PROGRAMMABLE MULTI-SWICTH



โดย
นายวศิน กาญจนางศาเวช
นายอภิสิทธิ์ ตันตระกูลปี่

เลขที่.....
เลขทะเบียน 42252
วัน, เดือน, ปี 16 พ.ค. 2545

b.....
i.....

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาอุตสาหกรรมศาสตรบัณฑิต
สาขาเทคโนโลยีอิเล็กทรอนิกส์
ภาควิชาเทคนิคอุตสาหกรรม คณะวิศวกรรมศาสตร์
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
ปีการศึกษา 2544

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้


b 11204151

หัวข้อปริญญานิพนธ์	เครื่องโปรแกรมสวิตช์หลายช่อง
ชื่อนักศึกษา	นายวศิน กาญจนางสงาวงษ์ รหัสประจำตัว 42015568 นายอภิสิทธิ์ ต้นตระกูลปี รหัสประจำตัว 42015585
อาจารย์ที่ปรึกษา	อ.อรรถสิทธิ์ หล้าสกุล
ระดับการศึกษา	ปริญญาอุตสาหกรรมศาสตรบัณฑิต สาขาวิชาเทคโนโลยีอิเล็กทรอนิกส์
ภาควิชา	เทคนิคอุตสาหกรรม
ปีการศึกษา	2544

คณะกรรมการศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง อนุมัติ
ให้ปริญญานิพนธ์ฉบับนี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรอุตสาหกรรมศาสตรบัณฑิต

.....หัวหน้าภาควิชาเทคนิคอุตสาหกรรม
(ผศ.อุทัย ศรีธีระวิโรจน์)


คณะกรรมการสอบปริญญานิพนธ์


.....(อาจารย์ที่ปรึกษา)
(อ.อรรถสิทธิ์ หล้าสกุล)
.....(กรรมการ)
(.....)
.....(กรรมการ)
(.....)
.....(กรรมการ)
(.....)
.....(กรรมการ)
(.....)
.....(กรรมการ)
(.....)

ลิขสิทธิ์ของคณะกรรมการศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปเผยแพร่
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Thesis Title PROGRAMMABLE MULTI-SWITCH
Student Mr. Wasin Karnjanapongsawech
 Mr. Apicit Tantaworrasain
Thesis Advisor Mr. Attasit Lasakul
Level of study Bachelor degree of industrial technology electronics
Department Industrial Technology
Academic Year 2001

Accepted by the Faculty of Engineering, King Mongkut's Institute of Technology
 Ladkrabang in partial fulfillment of the requirements for the bachelor's degree



..... Chairman
 (Asst.Prof. U-thai Sritheeravirojana)

Project Report Committee

..... Advisor
 (Mr. Attasit Lasakul)

..... Member
 (.....)

..... Member
 (.....)

..... Member
 (.....)

..... Member
 (.....)

..... Member
 (.....)

..... Member
 (.....)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
 ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หัวข้อปริญญานิพนธ์

เครื่องโปรแกรมสวิทช์หลายช่อง

ชื่อนักศึกษา

นายวศิน กาญจนางพจนวช รหัสประจำตัว 42015568

นายอภิสิทธิ์ ตันตระวรศิลป์ รหัสประจำตัว 42015585

อาจารย์ที่ปรึกษา

อ.อรรถสิทธิ์ หล้าสกุล

ระดับการศึกษา

ปริญญาอุตสาหกรรมศาสตรบัณฑิต

สาขาวิชาเทคโนโลยีอิเล็กทรอนิกส์

ภาควิชา

เทคนิคอุตสาหกรรม

ปีการศึกษา

2544

บทคัดย่อ

ในโรงงานอุตสาหกรรมทั่วไป จะประกอบไปด้วยเครื่องมือเครื่องจักรจำนวนมากมายที่ใช้กระแสไฟฟ้าในระดับที่ต่างกันและมีหน้าที่การทำงานที่ต่างกันทั้งยังอาจทำงานกันคนละเวลา แม้แต่ระบบไฟฟ้าแสงสว่างธรรมดาที่มีอยู่ในโรงงานขนาดใหญ่ ก็จะมีลำดับเวลาของการเปิดเปิดในแต่ละตำแหน่งที่แตกต่างกัน ดังนั้นเครื่องโปรแกรมสวิทช์แบบหลายช่องที่ทำงานเป็นอิสระต่อกันจึงนับได้ว่าเป็นประโยชน์มากในงานดังกล่าว ในงานปริญญานิพนธ์ฉบับนี้จะได้นำเสนอถึงการออกแบบสร้างเครื่องโปรแกรมสวิทช์แบบหลายช่องที่มีความเป็นอิสระต่อกัน เพื่อใช้งานในลักษณะดังกล่าว ทั้งนี้ได้เพิ่มประสิทธิภาพมากขึ้นไปอีก คือ ระบบการตรวจสอบระดับกระแสไฟฟ้าอัตโนมัติ, ระบบการตั้งโปรแกรมการทำงานแบบเชื่อมต่อกับคอมพิวเตอร์ทั่วไป ทำให้ง่ายแก่กำหนดการควบคุมสวิทช์แต่ละช่อง และที่สำคัญราคาถูกกว่าเครื่องของต่างๆประเทศ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Thesis Title	PROGRAMMABLE MULTI-SWITCH
Student	Mr. Wasin Karnjanapongsawech Mr. Apicit Tantaworrasain
Thesis Advisor	Mr. Attasit Lasakul
Level of study	Bachelor degree of industrial technology electronics
Department	Industrial Technology
Academic Year	2001

ABSTRACT

There are many kinds of machinery in a factory. Each has a sequence of operating it self (Time-table), including the illumination system. It should able to be programmed to switch on/off base on minimum of power consumption. In this thesis, we proposed the Programmable multi-switch system that can be programmed for many kinds of application. Furthermore, the system has been attached special function to connect with microcomputer (PC). Therefore, it could be very easy to program a soft ware time-sequence (for any channels of system) and development of hard ware machine it self

กิตติกรรมประกาศ

โครงการนี้สามารถสำเร็จได้ ด้วยความกรุณาในการให้คำปรึกษา แนะนำ และความช่วยเหลือด้วยความรู้, เครื่องมือ และอื่นๆ จาก อ.อรรถสิทธิ์ หล้าสกุล อาจารย์ที่ปรึกษา และคณาจารย์ประจำภาควิชาเทคนิคอุตสาหกรรม คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง ขอกราบขอบพระคุณเป็นอย่างสูง

นายวสิน กาญจนางศาเวช
นายอภิสิทธิ์ ตันตระกูลศิลป์



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ

	หน้า
บทคัดย่อภาษาไทย	ก
บทคัดย่อภาษาอังกฤษ	ง
กิตติกรรมประกาศ	จ
สารบัญ	ฉ
สารบัญรูป	ช
สารบัญตาราง	ญ
บทที่ 1 บทนำ	
1.1 ความสำคัญของโครงการ	1
1.2 วัตถุประสงค์	2
1.3 ขอบเขตของงาน	2
1.4 ขั้นตอนการดำเนินงาน	2
บทที่ 2 ทฤษฎี	
2.1 บอร์ดควบคุม CP-JR180 V2.0	4
2.2 LCD (Liquid Crystal Display)	13
2.3 พอร์ตอนุกรม (Serial Port)	21
2.3.1 การสื่อสารแบบอนุกรม	21
2.3.2 การสื่อสารแบบอะซิงโครนัส	22
2.3.3 มาตรฐานของพอร์ตอนุกรมแบบ RS-232	23
2.3.4 UART	27
2.3.5 ลักษณะสัญญาณอินพุตและเอาต์พุตของพอร์ตอนุกรม	28
2.3.6 แอคเตอรส์ของพอร์ตอนุกรม	29
2.3.7 การเขียนโปรแกรมเพื่อใช้งานพอร์ตอนุกรม	29
2.4 Current Transformer	40
บทที่ 3 การสร้างและพัฒนาโปรแกรม	
3.1 โครงสร้างของเครื่องโปรแกรมสวิตซ์หลายช่อง	42
3.2 การออกแบบและหลักการทำงานของวงจรในภาคต่างๆ	44

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น เมื่ออนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.2.1	ภาคประมวลผลรวมของวงจรทั้งหมด	44
3.2.2	ภาควงจรควบคุมคีย์บอร์ด (และรหัสของคีย์)	47
3.2.3	ภาคควบคุมการทำงานโหลด (Drive Load)	51
3.2.4	ภาคการควบคุมกระแสของโหลด (Limited Current)	53
3.2.4.1	วงจรแปลงสัญญาณดิจิทัลเป็นสัญญาณอนาล็อก (Digital to Analog Circuit)	53
3.2.4.2	วงจรหน่วงเวลา (Monostable Multivibrator)	54
3.2.4.3	วงจรเปรียบเทียบแรงดัน (Comparator Circuit)	55
	และวงจรตรวจจับกระแสไฟฟ้า	
3.2.5	ภาคแสดงผล	59
3.2.6	ภาครับส่งข้อมูลผ่านพอร์ตอนุกรม RS-232	59
3.3	การเขียน โปรแกรมควบคุมการทำงานเครื่อง โปรแกรมสวิตซ์หลายช่อง	61
3.3.1	ไฟล์ชาร์ตการเขียน โปรแกรมควบคุมคีย์บอร์ด	61
3.3.2	ไฟล์ชาร์ตการเขียน โปรแกรมควบคุม LC	62
3.3.3	ไฟล์ชาร์ตการเขียน โปรแกรมควบคุมหลัก	63
3.3.4	ไฟล์ชาร์ตการเขียน โปรแกรมควบคุมคีย์ Pgm	64
3.3.5	ไฟล์ชาร์ตการเขียน โปรแกรมควบคุมคีย์ Mual	65
3.3.6	ไฟล์ชาร์ตการเขียน โปรแกรมควบคุมคีย์ Amp	65
3.3.7	ไฟล์ชาร์ตการเขียน โปรแกรมควบคุมคีย์ Time	66
บทที่ 4	การทดลองและผลการทดลอง	
	การทดลองที่ 4.1 ทดลองวัดค่าแรงดัน ไฟฟ้าที่ได้ จากวงจรตรวจจับกระแส	68
บทที่ 5	บทสรุป	70
	บรรณานุกรม	74
	ภาคผนวก ก.	75
	ภาคผนวก ข.	
	ภาคผนวก ค.	
	ภาคผนวก ง.	

สารบัญรูป

รูปที่	หน้า
2-1 แสดง โครงสร้างภายในของ CPU เบอร์ Z80-180	4
2-2 แสดงการเลือกขนาดและชนิดของหน่วยความจำ	5
2-3 แสดง MEMORY MAP ของบอร์ด CP-JR180 V2.0	6
2-4 แสดงลักษณะและตำแหน่งขาใช้งานของ RTC6242	7
2-5 การกำหนด Jumper ใช้งาน ของหน่วยความจำ U4	12
2-6 แสดง MEMORY MAP ของบอร์ด CP-JR180 V2.0	12
2-7 แสดง I/O MAP ของบอร์ด CP-JR180 V2.0	13
2-8 ตัวอย่างการต่อใช้งาน LCD ร่วมกับ ET-BOARD	13
2-9 ลักษณะของ LCD ขนาด 16 ตัวอักษร 1 แถว (16*1)	14
2-10 รูปแบบอย่างง่ายที่สุดของข้อมูลแบบอนุกรม	21
2-11 รูปแบบอย่างง่ายที่สุดของข้อมูลอนุกรมแบบอซิงโครนัส	22
2-12 การจัดขาของคอนเนคเตอร์พอร์ตอนุกรมตามมาตรฐาน RS-232 ทั้งแบบ DB-9 และ DB-25 -	25
2-13 การต่ออุปกรณ์ภายนอกกับพอร์ตอนุกรมของคอมพิวเตอร์ใน ลักษณะต่างๆ	26
2-14 ไออะแกรมแสดงโครงสร้างทางฮาร์ดแวร์ของพอร์ตอนุกรม	28
3-1 แสดงบล็อก ไดอะแกรมการทำงานของวงจรทั้งหมด	42
3-2 แสดงวงจรบอร์ดใช้งาน CP-Jr180 V2.0	46
3-3 แสดงวงจรควบคุมคีย์บอร์ด	47
3-4 โฟล์ทชาร์ต โปรแกรมย่อยของการตรวจสอบการกดคีย์ และการตรวจสอบรหัสคีย์บอร์ด	49
3-5 แสดงรหัสประจำคีย์บอร์ด	50
3-6 แสดงตำแหน่งการใช้งานของคีย์บอร์ด	50
3-7 แสดงวงจรควบคุมการทำงานของรีเลย์โดย ใช้กับ โหลด ไฟฟ้ากระแสสลับ 220 โวลท์	51
3-8 แสดงวงจรควบคุมการทำงานของรีเลย์โดยผู้	51

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ใช้สามารถป้อนแรงดันตามต้องการ ได้โดยการปรับสวิทช์

3-9	แสดงวงจรการแปลงสัญญาณดิจิทัลเป็นสัญญาณอนาล็อก	53
3-10	แสดงวงจรแบ่งแรงดัน	54
3-11	แสดงวงจรโมโนสเตเบิลมีลติไวเบเรเตอร์	55
3-12	แสดงวงจรตรวจจับกระแสของไหล และวงจรเปรียบเทียบแรงดัน	56
3-13	แสดงวงจรการต่อใช้งาน LCD Module ร่วมกับ Z80-180	59
3-14	แสดงวงจรต่อใช้งานของ MAX232	59
3-15	แสดงตำแหน่งขาใช้งานของพอร์ต RS232 คอนเน็คเตอร์แบบ DB-9	60
3-16	แสดงไฟล์ซาร์ตการเขียนโปรแกรมของการควบคุมคีย์บอร์ด	61
3-17	แสดงไฟล์ซาร์ตการเขียนโปรแกรมควบคุม LCD	62
3-18	แสดงไฟล์ซาร์ตการเขียนโปรแกรมควบคุมหลัก	63
3-19	แสดงไฟล์ซาร์ตการเขียนโปรแกรมการทำงานของคีย์ Pgm	64
3-20	แสดงไฟล์ซาร์ตการเขียนโปรแกรมการทำงานของคีย์ Mual	65
3-21	แสดงไฟล์ซาร์ตการเขียนโปรแกรมการทำงานของคีย์ Amp	65
3-22	แสดงไฟล์ซาร์ตการเขียนโปรแกรมการทำงานของคีย์ Time	66
4-1	แสดงวงจรที่ใช้ในการทดลองวัดค่าของแรงดันเอาท์พุท	68
5-1	แสดงวงจรเลขชี้ข้อมูล	72

สารบัญตาราง

ตารางที่	หน้า
2-1 ตัวอย่างการเลือกขาสัญญาณควบคุม (ORG)	6
2-2 แสดงตำแหน่งและหน้าที่การทำงานของ RTC6242	8
2-3 ตำแหน่งพอร์ตใช้งานของ S255	9
2-4 แสดงหน้าที่ของสัญญาณ 8255 System	10
2-5 คำสั่ง HD44780	15
2-6 แสดงข้อมูลในแอดเดรส 0000 : 0411H ที่ใช้แจ้งจำนวน พอร์ตอนุกรม	29
3-1 ตารางการทำงานของไอซีที่หมายเลข 74LS145	48
4-1 แสดงค่าของแรงดันที่วัดได้จากการทดลองที่ 4.1	69

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 1

บทนำ

1.1 ความสำคัญของโครงการ

ปัจจุบันการควบคุมเครื่องจักรในโรงงานอุตสาหกรรมมีทั้งควบคุมด้วยระบบคอมพิวเตอร์ และควบคุมด้วยคนงาน ซึ่งในการควบคุมเครื่องจักรหลายๆ เครื่องจำเป็นต้องมีคนงานคอยดูแลเครื่องจักรเหล่านั้นให้ทำงานเป็นปกติและเปิดและปิดเป็นไปตามระยะเวลาที่กำหนด ดังนั้น ความผิดพลาดอาจเกิดขึ้นได้จากมนุษย์(Human Error) ทำให้ต้องสูญเสียพลังงานและเวลาในการผลิต ดังนั้นจึงนำระบบไมโครโพรเซสเซอร์มาช่วยในการควบคุมการเปิดและปิดเครื่องจักรเหล่านั้นให้มีประสิทธิภาพยิ่งขึ้น

การนำไมโครโพรเซสเซอร์มาประยุกต์ใช้งานเป็นเครื่องโปรแกรมสวิทช์หลายช่อง มาควบคุมเครื่องจักรให้ทำงานเป็นปกติ โดยสามารถเช็คเครื่องจักรว่าอยู่ในสภาพที่ปกติหรือไม่(คือเครื่องไม่ลัดวงจร) ถ้าปกติให้ทำการเปิดหรือปิดตามโปรแกรมที่ได้ตั้งเอาไว้ ถ้าหากดำเนินการเปิดเครื่องจักรไปแล้วเครื่องจักรผิดปกติเครื่องควบคุมก็สามารถสั่งปิดเครื่องจักรนั้นๆโดยอัตโนมัติ เช่น ต้องการเปิดเครื่องจักร A ที่เวลา 9.00 น. หลังจากนั้นเปิดเครื่องจักร B และ C ที่เวลา 11.00 น. และปิดเครื่องจักร A ที่เวลา 16.00 แต่ระหว่างนี้เครื่องจักร C ผิดปกติ โปรแกรมจะปิดเครื่องจักร C ก่อนที่จะถึงเวลาที่โปรแกรมไว้ และต่อจากนั้นปิดเครื่องจักร B ที่ 5.00 เป็นต้น ในการใช้เครื่องควบคุมสามารถนำไปใช้ควบคุมเครื่องไฟฟ้าภายในบ้านได้เช่นเดียวกันเช่น เปิดและปิดเครื่องฉีดน้ำ เปิดและปิดแอร์ เป็นเวลา คือเปิดแอร์เวลา 6.00 น. ปิดเวลา 18.00 น. เป็นต้น แล้วถ้าหากเครื่องจักรหรือเครื่องใช้ไฟฟ้านั้นอยู่ใกล้กันหลายๆ ตัว และแต่ละตัวก็มีเวลาเปิด และปิดที่แตกต่างกัน ซึ่งจะทำให้เกิดการสับสน และถ้าหากเกิดการช็อตกันของเครื่องใช้ไฟฟ้าก็จะไม่สามารถปิดเครื่องใช้ไฟฟ้านั้นได้ทันเวลา ดังนั้นจึงควรที่จะนำเครื่องโปรแกรมสวิทช์หลายช่องมาควบคุมเครื่องจักรหรือเครื่องใช้ไฟฟ้าเพื่อให้มีความสะดวกมากขึ้น

กระผมจึงคิดที่จะทำโครงการนี้ขึ้นมาเพื่อแก้ปัญหาต่างๆที่ได้กล่าวไว้ข้างต้น และเนื่องจากโครงการนี้ได้จัดทำขึ้นมาก่อนแล้ว แต่กระผมเห็นว่าน่าจะนำมาพัฒนาปรับปรุงให้มีประสิทธิภาพมากขึ้นได้

1.2 วัตถุประสงค์

1. เพื่อพัฒนาเครื่องมือที่อยู่ในห้องตลาดให้มีประสิทธิภาพการทำงานมากขึ้น
2. เพื่อควบคุมการทำงานของเครื่องจักรในโรงงานอุตสาหกรรมได้หลายเครื่องตามเวลาที่กำหนด และช่วยประหยัดพลังงานในโรงงานอุตสาหกรรม
3. เพื่อศึกษาการนำไมโครโปรเซสเซอร์ไปประยุกต์ใช้งานด้านอุตสาหกรรม
4. เพื่อใช้ควบคุมเครื่องใช้ไฟฟ้าได้อย่างสะดวก และประหยัดพลังงาน
5. เพื่อป้องกันอุปกรณ์เครื่องใช้ไฟฟ้าเสียหายเนื่องจากเกิดไฟฟ้าช็อต

1.3 ขอบเขตของงาน

1. ควบคุมการทำงานของเครื่องใช้ไฟฟ้าได้ 12 ช่อง
2. แสดงผลทาง LCD (Liquid Crystal Display) 2 แถว 16 ตัวอักษร
3. ควบคุมการทำงานผ่านคีย์บอร์ด ขนาด 24 คีย์
4. ควบคุมกระแสไฟฟ้าของอุปกรณ์ไฟฟ้าได้ 4 ช่อง
5. ควบคุมการเปิดเปิดได้ 8 ช่อง
6. ตั้งเวลาเปิดและปิดในแต่ละช่อง เป็น วัน สัปดาห์ เดือน และปี โดยแต่ละช่องจะทำงานอิสระต่อกัน โดยใช้สัญญาณนาฬิกาจาก RTC(Real Time Clock) 6242
7. ถ้าไฟดับสัญญาณนาฬิกาก็ยังทำงานต่อได้
8. มีระบบสำรองข้อมูลในขณะที่ไฟดับ
9. สามารถสั่งเปิดและปิดเครื่องใช้ไฟฟ้าได้โดยตรงจากคีย์บอร์ด
10. สามารถโปรแกรมผ่านเครื่องคอมพิวเตอร์ได้ โดยติดต่อผ่านพอร์ตอนุกรม RS-232 และมีเซอร์ฟแวร์รองรับเพื่อให้สะดวกต่อการใช้งาน

1.4 ขั้นตอนการดำเนินงาน

1. ศึกษาการประยุกต์ใช้งานบอร์ดไมโครโปรเซสเซอร์ Z80-180
2. ศึกษาการเขียนโปรแกรมภาษา assembly ของ Z80-180
3. ศึกษาการเขียนโปรแกรม Visual Basic
4. ศึกษาการควบคุมการทำงานของคีย์บอร์ด
5. ศึกษาการควบคุมการทำงานของจอแสดงผล LCD (Liquid Crystal Display)
6. ศึกษาอุปกรณ์ควบคุมการเปิดและปิดของอุปกรณ์ไฟฟ้า
7. ศึกษาอุปกรณ์ที่ใช้ในควบคุมกระแสของอุปกรณ์ไฟฟ้า

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์ไว้เพื่อใช้ในการศึกษาเท่านั้น เมื่ออนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

8. ศึกษาอุปกรณ์ที่ใช้เป็นฐานเวลาให้แก่วงจร
9. ศึกษาการรับส่งข้อมูลผ่านพอร์ตอนุกรม RS 232
10. ออกแบบวงจรควบคุมคีย์บอร์ด
11. ออกแบบวงจรที่ใช้ในการควบคุมการเปิดและปิดอุปกรณ์ไฟฟ้า
12. ออกแบบวงจรที่ใช้ในการควบคุมกระแสของอุปกรณ์ไฟฟ้า
13. ทดสอบวงจรคีย์ และวงจรควบคุมการเปิดและปิดอุปกรณ์ไฟฟ้า
14. ทดสอบวงจรควบคุมกระแสของอุปกรณ์ไฟฟ้า
15. เขียนโปรแกรมควบคุมการทำงานทั้งหมดของวงจร
โดยใช้ภาษาแอสเซมบลี (Assembly) ของ Z80-180
16. เขียนโปรแกรมรับส่งข้อมูลผ่านพอร์ตอนุกรม RS 232
โดยใช้โปรแกรม Visual Basic
17. ทดสอบระบบการทำงานของเครื่องโปรแกรมสวิตซ์หลายช่อง

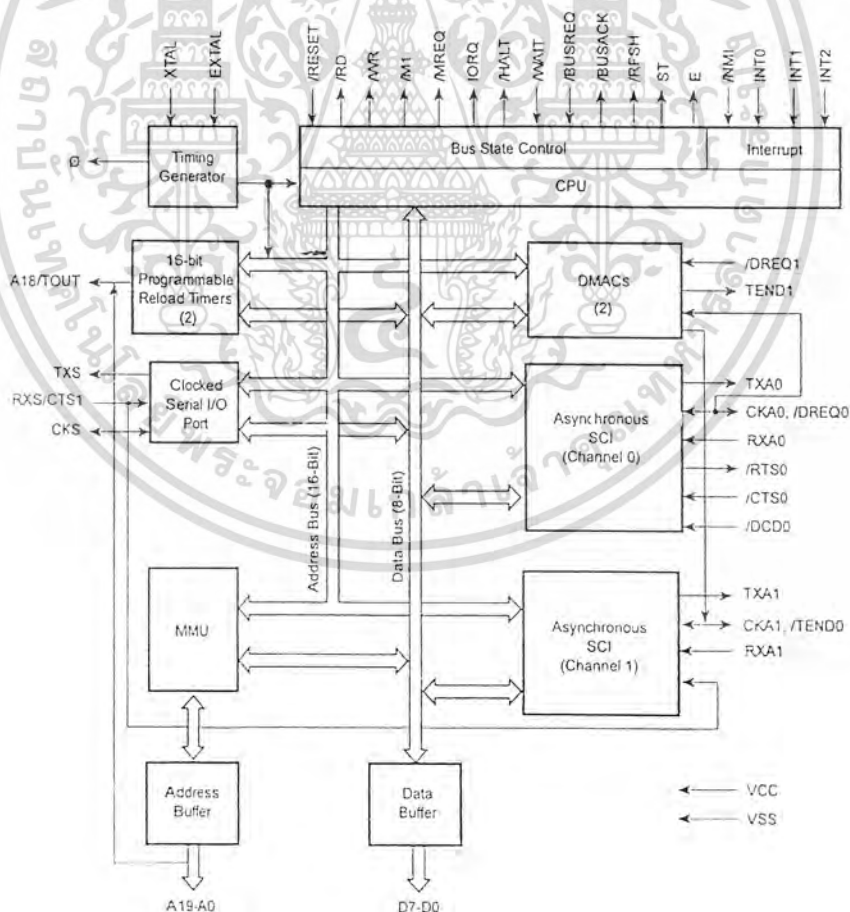
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 2

ทฤษฎี

การออกแบบเครื่อง โปรแกรมสวิตช์หลายช่องด้วยไมโครโปรเซสเซอร์ Z80-180 ต้องอาศัยความรู้เกี่ยวกับการใช้งานบอร์ดควบคุมการทำงานของไมโครโปรเซสเซอร์ Z80 CP-JR180 , การเขียนโปรแกรมด้วยภาษา assembly ของ Z80-180 , โปรแกรม Visual Basic , การใช้อุปกรณ์ฐานเวลา , การใช้งาน LCD module (Liquid Crystal Display) และการตรวจสอบการลัดวงจร ซึ่งจะกล่าวถึงทฤษฎีดังนี้

2.1 บอร์ดควบคุม CP-JR180 V2.0



รูปที่ 2-1 แสดง โครงสร้างภายในของ CPU เบอร์ Z80-180

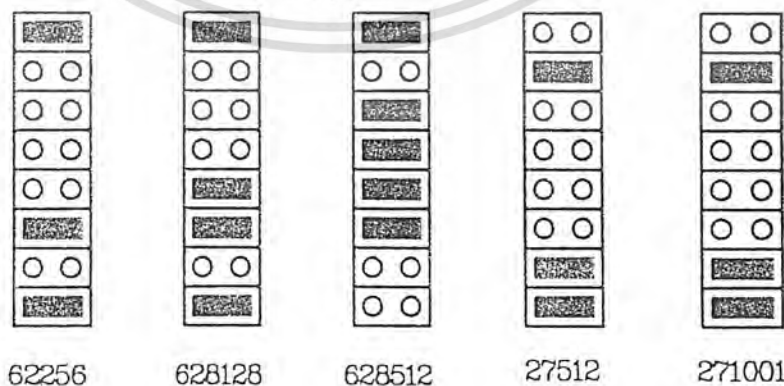
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่นอญญาติหนาไปไซ้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บอร์ด CP-JR180 V2.0 จะใช้ CPU ของ ZILOG เบอร์ Z80180 เป็นหน่วยประมวลผลหลักของระบบ ทั้งนี้เนื่องจาก CPU เบอร์ Z80180 จัดว่าเป็น CPU ที่มี ความสามารถสูงตัวหนึ่งที่ได้พัฒนาปรับปรุงมาจาก CPU ขอดนิมม เบอร์ Z80 ซึ่งใช้ชุดคำสั่งทั้งหมดยังคงเป็นของ Z80 อยู่เหมือนเดิม เพียงแต่ได้มีการเพิ่มเติมชุดคำสั่งและความสามารถเพิ่มเติมอื่นๆให้สามารถใช้งานได้กว้างขวางและสะดวกมากขึ้น โดยได้รวบรวมเอาส่วนประกอบที่สำคัญๆหลายอย่างเอาไว้ภายในตัว CPU อย่างครบถ้วน ซึ่งเมื่อพิจารณาแล้วนับว่ามีความเหมาะสมกับการใช้งานในด้านระบบการควบคุมต่างๆเป็นอย่างมาก ไม่ว่าจะเป็นระบบการควบคุมในโรงงานอุตสาหกรรม หรือระบบการสื่อสารข้อมูลเพราะ CPU เบอร์ Z80180 สามารถอ้างหน่วยความจำได้สูงถึง 1 MB มีระบบการทำ DMA ถึง 2 วงจร มีพอร์ตสื่อสารแบบอนุกรม UART 2 ชุด ซึ่งสามารถทำการสื่อสารแบบ Multi Process Communication ได้โดยตรงและยังมีระบบฐานเวลา Timer/Counter และการอินเทอร์รัพท์ต่างๆอีก

วงจรมีพอร์ตสื่อสารแบบอนุกรม UART 2 ชุด ซึ่งสามารถทำการสื่อสารแบบ Multi Process Communication ได้โดยตรงและยังมีระบบฐานเวลา Timer/Counter และอินเทอร์รัพท์ต่างๆอีก

หน่วยความจำ บอร์ด CP-JR180 V2.0 สามารถเชื่อมต่อกับหน่วยความจำทั้งแบบ ROM และ RAM ได้ Page ละ 64KB จำนวน 6 Page โดยวงจรในส่วนของ การ Decode ตำแหน่งของหน่วยความจำนั้น เราจะเลือกใช้การ Decode ออกเป็น 6 Page โดยมีขนาด Page ละ 64KB โดยเราสามารถจัดแบ่งหน่วยความจำออกเป็นดังนี้คือ

สำหรับการเลือกขนาดและชนิดของหน่วยความจำนั้น เราสามารถทำการเลือกได้โดย Jumper JP2 ซึ่งมีดังนี้



รูปที่ 2-2 แสดงการเลือกขนาดและชนิดของหน่วยความจำโดย Jumper JP2

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หน่วยความจำ U4 (EPROM หรือ RAM) 62256 628128/628512 27512/271001	5FFFFH ถึง 20000H
หน่วยความจำ U3 (RAM 32KB) เบอร์ 62256	1FFFFH ถึง 10000H
หน่วยความจำ U2 (EPROM 32/64KB) เบอร์ 27256/27512	0FFFFH ถึง 00000H

รูปที่ 2-3 แสดง MEMORY MAP ของบอร์ด CP-JR180 V2.0

EEPROM สำหรับหน่วยความจำในส่วนนี้จะไม่สัมพันธ์เกี่ยวข้องกับตำแหน่งของหน่วยความจำที่ไว้ในระบบแต่อย่างใด ทั้งนี้เพราะเราใช้ หน่วยความจำแบบ EEPROM ชนิด Serial-IO ซึ่งจะต้องใช้สัญญาณจากระบบ Clock Serial I/O Port ของ CPU ในการอ่าน / เขียนข้อมูลกับ EEPROM ซึ่งในส่วนของหน่วยความจำแบบ EEPROM นี้สามารถเลือกใส่ EEPROM ได้ทั้งหมด 3 ขนาด คือ 1KB/2KB/4KB แต่เนื่องจาก EEPROM ที่มีจำหน่ายกันอยู่ทั่วไปตามท้องตลาดนั้น ในปัจจุบันจะมีอยู่มากมายหลายยี่ห้อ ซึ่งในบางยี่ห้อก็ใช้วิธีการอ่านเขียนข้อมูลที่แตกต่างกันไปถึงแม้ว่าจะมีขนาดความจุ ของหน่วยความจำที่เท่ากันก็ตามที กล่าวคือ ในบางยี่ห้อใช้วิธีการอ่านเขียนข้อมูลครั้งละ 8 บิต บางยี่ห้อใช้วิธีการอ่านเขียนข้อมูลแบบ 16 บิต หรือบางยี่ห้อก็สามารถทำการอ่านเขียนข้อมูลได้ทั้งแบบ 8 บิต และ 16 บิต โดยการเลือกจากขาสัญญาณควบคุม(ORG) ก็ได้ ดังตัวอย่างในตาราง

EEPROM เบอร์	อ่าน/เขียนข้อมูล แบบ 8 บิต	อ่าน/เขียนข้อมูล แบบ 16 บิต	ขนาด ความจุ
93xx46	128ไบต์ x 8บิต	64ไบต์ x 16บิต	1 KB
93xx56	256ไบต์ x 8บิต	128ไบต์ x 16บิต	2 KB
93xx66	512ไบต์ x 8บิต	256ไบต์ x 16บิต	4 KB

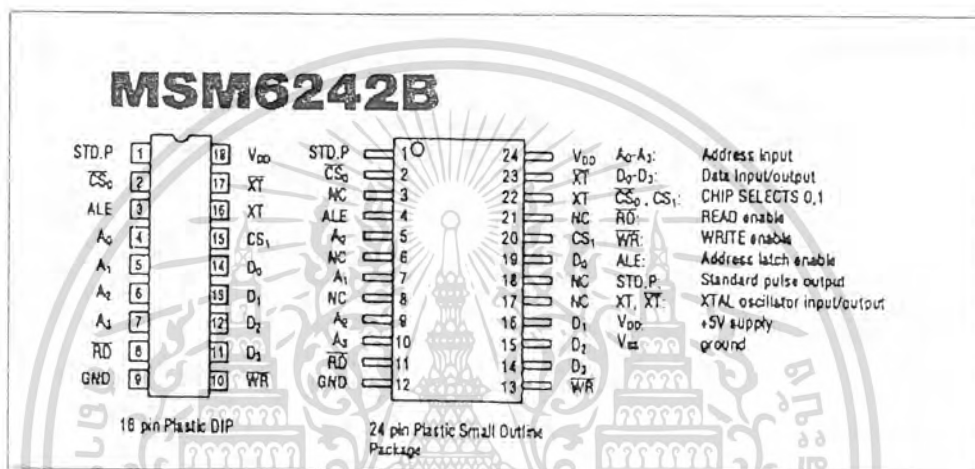
ตารางที่ 2-1 ตัวอย่างการเลือกขาสัญญาณควบคุม (ORG)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ซึ่งโดยทั่วไป EEPROM ที่สามารถเลือกวิธีการอ่าน / เขียน ข้อมูลได้ทั้งแบบ 8 บิต และ 16 บิต จะใช้ขาสัญญาณ ORG (ขา-6)เป็นตัวกำหนด โดยการเลือก Logic ให้กับขาสัญญาณนี้โดย

- ถ้าขา ORG ได้รับ Logic “0” จะเป็นการอ่าน / เขียนแบบ 8 บิต
- ถ้าขา ORG ได้รับ Logic “1” จะเป็นการอ่าน / เขียนแบบ 16 บิต

วงจรฐานเวลาฬิกา RTC6242



รูปที่ 2-4 แสดงภาพและตำแหน่งขาใช้งานของ RTC6242

สำหรับวงจรในส่วนของฐานเวลาฬิกานี้เราจะใช้ IC RTC6242 ซึ่งเป็น Real Time Clock/Calendar ในตัวเดียวกัน ซึ่งจะทำให้เราสามารถอ่านเขียนค่า เวลาวินาที นาที ชั่วโมง วัน เดือน ปี วันในรอบสัปดาห์ และยังมีส่วนของวันในปีอธิกสุรทิน อีกด้วย กล่าวคือสามารถตรวจสอบและปรับเวลาของปีที่มี 28 หรือ 29 วัน ได้เองโดยอัตโนมัติ ซึ่งการเชื่อมต่อของวงจรในส่วนนี้จะใช้วิธีการ Decode ตำแหน่งไว้ เราจึงสามารถทำการอ่านเขียนข้อมูลกับ RTC ผ่านทางพอร์ตได้ เลข ซึ่ง RTC นี้จะใช้ตำแหน่งพอร์ตในการอ่านเขียนข้อมูลกับส่วนต่างๆ ไม่ว่าจะเป็น เวลา วัน เดือน หรือ ปี ซึ่งมีด้วยกันทั้งหมด 16 ตำแหน่งด้วยกัน โดยมีตำแหน่งการทำงานอยู่ที่ A0H-AFH ตามลำดับ ดังแสดงในตาราง

ตำแหน่ง Address	ชื่อ รีจิสเตอร์	ค่าของบิตข้อมูล				ค่า นับ	หน้าที่และ การทำงาน
		D3	D2	D1	D0		
A0H	Sec1	S8	S4	S2	S1	0-9	ค่าวินาทีหลักหน่วย
A1H	Sec10	-	S40	S20	S10	0-5	ค่าวินาทีหลักสิบ
A2H	Min1	mi8	mi4	mi2	mi1	0-9	ค่านาทีหลักหน่วย
A3H	Min10	-	mi40	mi20	mi10	0-5	ค่านาทีหลักสิบ
A4H	Hour1	h8	h4	h2	h1	0-9	ค่าชั่วโมงหลักหน่วย
A5H	Hour10	-	PM/ AM	h20	h10	0-2 0-1	ค่าชั่วโมงหลักสิบ & สถานะ AM/PM
A6H	Date1	d8	d4	d2	d1	0-9	ค่าวันหลักหน่วย
A7H	Date10	-	-	d20	d10	0-3	ค่าวันหลักสิบ
A8H	Mon1	mo8	mo4	mo2	mo1	0-9	ค่าเดือนหลักหน่วย
A9H	Mon10	-	-	-	mo10	0-1	ค่าเดือนหลักสิบ
AAH	Year1	y8	y4	y2	y1	0-9	ค่าปีหลักหน่วย
ABH	Year10	y80	y40	y20	y10	0-9	ค่าปีหลักสิบ
ACH	Week	-	w4	w2	w1	0-6	ค่าสัปดาห์
ADH	Control Reg. D	30 sec ADJ	IRQ FLAG	BUSY	HOLD	-	รีจิสเตอร์ควบคุม D
AEH	Control Reg. E	n	10	ITRPT /STND	MASK	-	รีจิสเตอร์ควบคุม E
A FH	Control Reg. F	TEST	24/12	STOP	RESET	-	รีจิสเตอร์ควบคุม F

หมายเหตุ ITRPT/STND =Interrupt/Standard

ตารางที่ 2-2 แสดงตำแหน่งและหน้าที่การทำงานของ RTC6242

จอแสดงผล LCD (Liquid Crystal Display)

สำหรับส่วนของการแสดงผล แบบ LCD นี้เราสามารถเลือกใช้ออกแสดงผล LCD ได้ 2 ชนิด คือ ทั้งแบบรูปภาพ (Graphic) และแบบอักขระตัวอักษร(Character Dot-matrix) โดยมี Jmper7 เป็นตัวเลือกว่าจะต่อกับ LCD ชนิดใด และใช้ตัวต้านทานปรับค่าได้แบบเก็อกมีทำหน้าที่เป็นตัวปรับ

ระดับความสว่าง ของหน้าจอ LCD โดยในการต่อใช้งาน LCD แต่ละชนิดจะใช้สัญญาณในการเอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ในการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เชื่อมต่อไม่เท่ากันคือถ้าเป็นจอ LCD ชนิดตัวอักษร Character Dot-Matrix จะใช้สัญญาณ 14 เส้น แต่ถ้าเป็นจอ LCD ชนิดรูปภาพ Graphic ต้องใช้สัญญาณทั้งหมด 18 เส้น แต่ถ้าขั้วต่อของ LCD ที่ออกแบบไว้บนบอร์ด CP-JR180 V2.0 นี้จะเป็นขนาด 20 ขา เพื่อให้สามารถต่อกับ LCD ได้ทั้ง 2 แบบ ดังนั้นถ้าต้องการใช้กับ LCD ชนิดตัวอักษร Character Dot-Matrix ก็ให้ต่อสายสัญญาณเพียงแค่ 14 เส้นแรกเท่านั้น ส่วนสัญญาณที่เกินมาก็ไม่ต้องสนใจให้ปล่อยวางไว้

I/O PORT

อุปกรณ์อินพุตเอาต์พุตของบอร์ด CP-JR180 V2.0 นี้จะใช้ Chip Support เบอร์ 8255 จำนวน 2 ตัว ซึ่งขั้วต่อของพอร์ตจะเป็นขั้วต่อมาตรฐานอีทีที(72-IOZ80) สามารถเชื่อมต่อกับอุปกรณ์อินพุตเอาต์พุตของอีทีทีได้ทันทีโดย 8255 นี้จะมีตำแหน่งพอร์ตใช้งานดังต่อไปนี้คือ

8255 System (72IOZ80-1)		8255 User (72IOZ80-2)	
Port A	60H	Port A	80H
Port B	61H	Port B	81H
Port C	62H	Port C	82H
Port Control	63H	Port Control	83H

ตารางที่ 2-3 ตำแหน่งพอร์ตใช้งานของ 8255

ซึ่งพอร์ต 8255 นี้จะแบ่งออกเป็น 2 ส่วน คือ 8255 System เป็นพอร์ต 8255 ซึ่งสัญญาณบางส่วนจาก 8255 นี้จะถูกนำไปใช้ในระบบเช่น ควบคุมการอ่านเขียนของวงจรร EEPROM วงจร A/D วงจรแสดงผล SDP8 และคีย์บอร์ด กับ ลำโพง เป็นต้น โดยถ้าหากว่าผู้ใช้ไม่ได้ใช้งานอุปกรณ์ที่กล่าวมาแล้วข้างต้นก็สามารถนำสัญญาณของ 8255 System นี้ไปใช้งานอื่นตามความต้องการได้ ซึ่งถ้าหากมีความต้องการจะนำสัญญาณจาก 8255 System นี้ไปใช้งานภายนอกแล้วจะต้องไม่ใส่อุปกรณ์ที่กล่าวมาแล้วข้างต้นในบอร์ด เพราะจะทำให้เกิดการชนกันได้โดยพอร์ต 8255 System จะมีขั้วต่อของสัญญาณเป็นแบบ 34 PIN(72IOZ80-1)

ชื่อสัญญาณ	สัญญาณการเชื่อมต่อ	หน้าที่
PA0-PA7	ว่างสำหรับใช้งานตามต้องการอย่างอิสระ	-
PB0	สัญญาณ CS# ของ EEPROM	OUTPUT
PB1	สัญญาณ CLK ของ A/D (LTC1298)	OUTPUT
PB2	สัญญาณ DI ของ A/D (LTC1298)	OUTPUT
PB3	สัญญาณ CS# ของ A/D (LTC1298)	OUTPUT
PB4	สัญญาณ CLK ของชุดแสดงผล SDP8	OUTPUT
PB5	สัญญาณ DIN ของชุดแสดงผล ADP8	OUTPUT
PB6	สัญญาณ LOAD ของชุดแสดงผล SDP8	OUTPUT
PB7	สัญญาณขับลำโพงเล็กเพื่อกำเนิดเสียง	OUTPUT
PC0-PC7	ใช้สำหรับ Scan Keyboard ขนาด 4x4	IN & OUT

ตารางที่ 2-4 แสดงหน้าที่ของสัญญาณ 8255 System

ส่วนพอร์ต 8255 User นั้นสัญญาณทุกเส้นจาก 8255 ในส่วนนี้จะวางไว้สำหรับผู้ใช้สามารถต่อไปใช้งานได้อย่างอิสระ โดยมีขั้วต่อของสัญญาณเป็นแบบ 34 PIN(72IOZ80-2)

พอร์ตอนุกรม UART

สำหรับวงจร ในส่วนของการรับส่งข้อมูลแบบอนุกรม UART นั้น ในบอร์ด CP-JR180 V2.0 จะมีอยู่ 2 Channel โดยในวงจรของ Channel 0(RS232-1)จะทำการติดต่อสื่อสารแบบ RS232 ได้อย่างดีซึ่งใน Channel 0 นี้สามารถรับส่งข้อมูลแบบ Full Duplex ก็คือสามารถทำการรับส่งข้อมูลกันเป็นแบบ CPA-4PIN(RS232-1)

สำหรับในส่วนของวงจร Channel 1(RS232-2) จะสามารถเลือกใช้วงจร Line Driver ได้หลายลักษณะไม่ว่าจะเป็น RS232 หรือ RS422 หรือ RS485 ก็ได้ตามความเหมาะสมและตามความต้องการของผู้ใช้ โดยถ้าต้องการ วิธีการรับส่งข้อมูลแบบ RS232 ชนิด Full Duplex ต้องต่อสายสัญญาณส่งข้อมูลเข้ากับขั้วต่อแบบ CPA-4PIN(RS232-2)และต้องเลือก Jumper ตำแหน่ง JP8 (MAX/75176)ไว้ยังตำแหน่ง MAX เพื่อเลือกวงจร Line Driver เป็นเบอร์ MAX232 สำหรับส่งข้อมูลแบบ RS232 นั่นเอง

ซึ่งในการรับส่งข้อมูลแบบ RS232 นั้นจะมีข้อจำกัดในเรื่องของระยะทางอยู่มาก คือสามารถรับส่งกันได้สูงสุดประมาณ 50 ฟุต เท่านั้น ถ้าระยะทางไกลกว่านี้ก็จะเกิดความผิดพลาดสูง เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หรือรับส่งกันไม่ได้เลย ถ้าผู้ใช้งานมีความต้องการที่จะรับส่งข้อมูลให้ได้ระยะทางที่ไกลกว่า 50 ฟุต แล้วต้องเปลี่ยนวงจร Line Driver ให้เป็นแบบ RS422 ซึ่งจะทำได้ทำให้สามารถรับข้อมูลกันได้ในระยะทางที่ไกลขึ้นกว่าแบบ RS232 มากกล่าวคือ การรับส่งข้อมูลแบบ RS422 สามารถทำการรับส่งข้อมูลกันได้ในไกลสูงสุดถึง 4,000 ฟุตหรือประมาณ 1.2 กิโลเมตร เลขที่เคียว ซึ่งการเปลี่ยนวงจรการรับส่งแบบ RS232 มาเป็นแบบ RS422 นี้ยังไม่จำเป็นต้องเปลี่ยนแปลงโปรแกรมสำหรับควบคุมการรับส่งข้อมูลใหม่แต่อย่างใด ยังคงใช้โปรแกรมเดิมเกี่ยวกับการรับส่งแบบ RS232 อยู่เช่นเดิม เพียงแต่ว่าทั้งฝ่ายรับและฝ่ายส่งต้องเปลี่ยนวงจร Line Driver ของวงจรการรับส่งให้เป็นแบบ RS422(ใช้ Line Driver เบอร์ 75176) ให้เหมือนกันทั้งคู่เท่านั้นก็สามารถรับส่งข้อมูลกันได้แล้ว ซึ่งในบอร์ด CP-JR180 V2.0 นี้สามารถเปลี่ยนได้โดยการนำไอซี Line Driver เบอร์ 75176 มาใส่ลงในบอร์ดแล้วเลือก Jumper ตำแหน่ง JP8(MAX/75176)มาไว้ที่ตำแหน่ง 75176 และเลือก Jumper 9 ตำแหน่ง JP9 (422/485) มาไว้ในตำแหน่ง 422 เท่านั้น ส่วนสายสัญญาณในการรับส่งก็ให้ต่อเข้าที่ขั้วแบบ CPA-64PIN (422/485)

แต่ในกรณีที่ต้องการต่อวงจรหลายๆชุดเข้าด้วยกันเพื่อทำการรับส่ง ข้อมูลซึ่งกันและ (MultiProcessor Communication System) วงจรการรับส่งแบบ RS232 และแบบ RS422 ไม่สามารถใช้ได้ ในกรณีนี้เพราะจะเกิดการชนกันของข้อมูลในสายทำให้เกิดความผิดพลาดและรับส่งข้อมูลกันไม่รู้เรื่อง ดังนั้นจึงต้องหลีกเลี่ยงมาใช้วงจรการรับส่งแบบ RS485 แทน โดยในการรับส่งข้อมูลแบบ RS485 นี้จะสามารถรับส่งข้อมูลกันได้ในไกลสูงสุดถึง 4000 ฟุต หรือประมาณ 1.2 กิโลเมตร เช่นเดียวกับแบบ RS422 แต่ไม่สามารถทำการรับและส่งข้อมูลพร้อมกันในเวลาเดียวกันได้ ต้องใช้วิธีการผลัดกันรับและผลัดกันส่งทีละครั้ง หรือเรียกว่า “Half Duplex” ซึ่งในการรับส่งข้อมูลแบบนี้จะมีข้อดีคือสามารถเชื่อมต่อบนไมโครโปรเซสเซอร์หลายๆตัว เข้าด้วยกันได้ แต่มีข้อเสียคือต้องมีโปรแกรมสำหรับควบคุมการรับส่งโดยเฉพาะ เพื่อจัดลำดับและกำหนดทิศทางการรับส่งข้อมูลว่าเมื่อใดควรจะให้วงจรใดทำหน้าที่รับและ ให้วงจรใดทำหน้าที่ส่งข้อมูล ซึ่งต้องใช้วิธีที่ซับซ้อนมากขึ้น

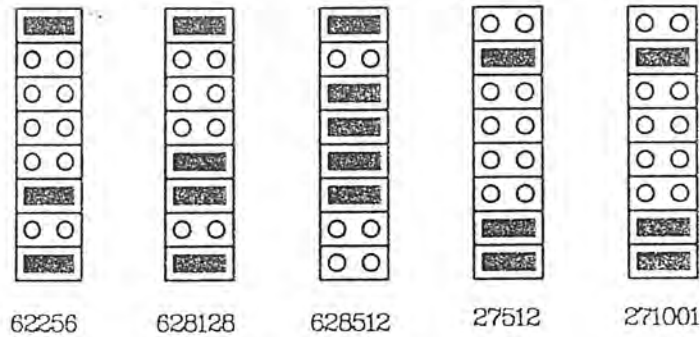
สรุปหน้าที่ของ JUMPER ต่างๆ

JP1 ใช้สำหรับเลือกขนาด EPROM MONITOR(U2)ว่าจะใช้หน่วยความจำขนาด 3 KB หรือ 64 KB(27256/27512)

JP2 ใช้กำหนดเลือกเบอร์หน่วยความจำ Expansion U4

JP3 ใช้เลือกแหล่งจ่ายไฟให้หน่วยความจำ U4ว่าจะใช้แหล่งจ่าย +VCC จากระบบ หรือ +VBAT จาก Battery(สำหรับ Backup)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2-5 การกำหนด Jumper ใช้งานของหน่วยความจำ U4

JP4 ใช้เลือกว่าจะใช้ WatchDog ควบคุมการรีเซ็ตหรือไม่(WD/CLK)

JP5 ใช้เลือกสัญญาณ CS# ของหน่วยความจำ EEPROMว่าจะใช้สัญญาณ RST0 จาก CPU หรือสัญญาณ PB0 ของ 8255 (System)

JP6 ใช้เลือกวิธีการอ่าน EEPROMว่าจะใช้แบบ 8 บิต หรือ 16 บิต

JP7 ใช้เลือกชนิดของจอแสดงผล LCDว่าจะใช้รุ่นรูปภาพ Graphic หรือรุ่นตัวอักษร Dotmatrix(GR/CH)

JP8 ใช้เลือกวงจร Line Driver ของพอร์ตสื่อสาร UART Channel-1ว่าจะใช้ MAX232 สำหรับการรับส่งแบบ RS232 หรือใช้ 75176 สำหรับการรับส่งแบบ RS422 และ RS485

JP9 ใช้เลือกรูปแบบการรับส่งของพอร์ตสื่อสาร UART Channel-1 ในกรณีที่ใช้ Line Driver เป็น 75176 แล้วว่าจะเป็นแบบ RS422 (Full Duplex) หรือ RS485 (Half Duplex)

หน่วยความจำ U4 (EPROM หรือ RAM)	5FFFFH
62256	ถึง
628128/628512	
27512/271001	20000H
หน่วยความจำ U3 (RAM 32KB)	1FFFFH
เบอร์ 62256	ถึง
	10000H
หน่วยความจำ U2 (EPROM 32/64KB)	0FFFFH
เบอร์ 27256/27512	ถึง
	00000H

รูปที่ 2-6 แสดง MEMORY MAP ของบอร์ด CP-JR180 V2.0

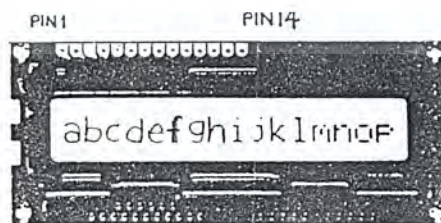
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

WatchDog	FFH
	EOH
จอแสดงผล LCD Write Instruction = C0H,C1H Write Data = C2H,C3H Read Busy&Addr= C4H,C5H	DFH
	COH
RTC6242	BFH
	A0H
8255 User PA = 80H PB = 81H PC = 82H PCC = 83H	9FH
	80H
8255 System PA = 60H PB = 61H PC = 62H PCC = 63H	7FH
	60H
	5FH
ว่างสำหรับผู้ใช้	00H

รูปที่ 2-7 แสดง I/O MAP ของบอร์ด CP-JR180 V2.0

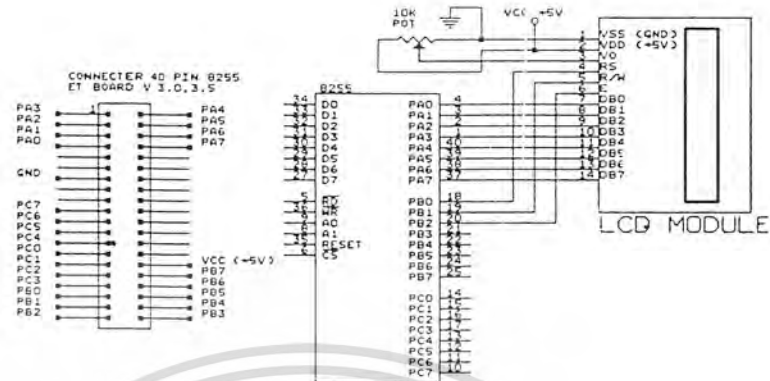
2.2 LCD (Liquid Crystal Display)

HD44780 เป็นไอซี LSI ตัวหนึ่งใช้ควบคุม LCD โดยแสดงผลในรูปแบบตัวอักษรหรือสัญลักษณ์ต่างๆตัวมันเองสามารถต่อใช้งานแบบ 4 Bit หรือ 8 Bit ก็ได้ โดยถ้าเราต่อแบบ 4 Bit จะต่อใช้งานที่ DB7-DB4 เท่านั้นโดยข้อมูลครั้งแรกที่ส่งนั้น HD44780 จะถือเป็นข้อมูล 4 Bit บนและข้อมูลที่ส่งต่อมานั้นเป็นข้อมูล 4 bit ล่าง



รูปที่ 2-8 ลักษณะของ LCD ขนาด 16 ตัวอักษร 1 แถว (16 * 1)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2-9 ตัวอย่างการต่อใช้งาน LCD ร่วมกับ ET-BOARD V.3.0, ET-BOARD V3.5

จากวงจรเป็นการต่อ 8255 ให้เข้าใช้กับ LCD โดยเราจะจำลองสัญญาณต่างๆ ขึ้นมาโดยการใช้ PORT A และ PORT B โดย PORT A นั้นเราให้เป็น DATA PORT และ PORT B นั้นเราให้เป็นสัญญาณควบคุม ไปใช้เมื่อเราเริ่มเปิดไฟป้อนให้ HD44780 นั่นก็จะทำการ RESET ตัวมันเอง โดยจะใช้เวลาประมาณ 10 ms หลังจากไฟ VDD ถึง 4.5 Volt แล้ว โดยจะ SET ตัวเองดังนี้ :

1. DISPLAY CLEAR จะทำการลบข้อมูลจอภาพ LCD
2. FUNCTION SET โดยจะ SET ค่าภายใน
 - DL = 1 : เป็นการ SET ให้การติดต่อแบบ 8 BIT
 - N = 0 SET เป็น 1 บรรทัดการแสดงผล
 - F = 0 : 5X7 DOT ต่อหนึ่งตัวอักษร
3. DISPLAY ON/OFF D = 0 : DISPLAY OFF
 - C = 0 : CURSOR OFF
 - B = 0 : BLINK OFF
4. ENTRY MODE SET I/O = 1 : +1 (เพิ่มค่า COUNTER ขึ้น 1)
 - S = 0 : NO SHIFT

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 2-5 คำสั่ง HD44780

Instructions

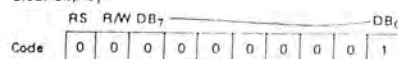
Instruction	Code										Description	Execution Time (max) (when fcp or fosc is 250 kHz)		
	RS	R/W	DB ₇	DB ₆	DB ₅	DB ₄	DB ₃	DB ₂	DB ₁	DB ₀				
Clear Display	0	0	0	0	0	0	0	0	0	0	1	Clears entire display and sets DD RAM address 0 in address counter.	1.64 ms	
Return Home	0	0	0	0	0	0	0	0	0	1	*	Sets DD RAM address 0 in address counter. Also returns display being shifted to original position. DD RAM contents remain unchanged.	1.64 ms	
Entry Mode Set	0	0	0	0	0	0	0	1	I/D	S		Sets cursor move direction and specifies shift of display. These operations are performed during data write and read.	40µs	
Display On/Off Control	0	0	0	0	0	0	1	D	C	B		Sets ON/OFF of entire display (D), cursor ON/OFF (C), and blink of cursor-position character (B).	40µs	
Cursor or Display Shift	0	0	0	0	0	1	SC	R/L	*	*		Moves cursor and shifts display without changing DD RAM contents.	40µs	
Function Set	0	0	0	0	1	DL	N	F	*	*		Sets interface data length (DL), number of display lines (L) and character font (F).	40µs	
Set CG RAM Address	0	0	0	1	ACG							Sets CG RAM address. CG RAM data is sent and received after this setting.	40µs	
Set DD RAM Address	0	0	ADD							Sets DD RAM address. DD RAM data is sent and received after this setting.	40µs			
Read Busy Flag & Address	0	1	BF		AC							Reads Busy flag (BF) indicating internal operation is being performed and reads address counter contents.	0µs	
Write Data to CG or DD RAM	1	0	Write Data										Writes data into DD RAM or CG RAM.	40µs
Read Data from CG or DD RAM	1	1	Read Data										Reads data from DD RAM or CG RAM.	40µs
	I/D=1: Increment I/D=0: Decrement S =1: Accompanies display shift SC=1: Display shift SC=0: Cursor move R/L=1: Shift to the right R/L=0: Shift to the left DL=1: 8 bits, DL=0: 4 bits N =1: 2 lines, N=0: 1 line F =1: 5x10 dots, F=0: 5x7 dots BF =1: Internally operating BF =0: Can accept instruction										DD RAM: Display data RAM CG RAM: Character generator RAM ACG: CG RAM address ADD: DD RAM Address Corresponds to cursor address AC: Address counter used for both DD and CG RAM address.	Execution time changes when frequency changes Example: When fcp or fosc is 270 kHz: $40\mu s \times \frac{250}{270} = 37\mu s$		

* No effect

รายละเอียดของคำสั่ง HD44780

1. CLEAR DISPLAY

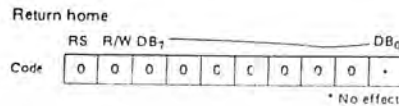
Clear display



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

คำสั่งนี้จะเป็นการเขียนช่องว่างหรือ SPACE (ASCII 20H) เข้าไปใน DD RAM ทั้งหมด และทำการ SET DD RAM ADDRESSSER เป็นศูนย์ ตัว CURSOR จะกลับไปอยู่ ตำแหน่ง บนสุดซ้ายมือของจอภาพ SET I/D = 1 , S ไม่มีการเปลี่ยน

2. RETURN HOME



คำสั่งนี้จะทำการ SET DD RAM ADDRESSSER เป็นศูนย์ ตัว CURSOR จะกลับไปอยู่ ตำแหน่งบนสุดซ้ายมือของจอภาพข้อมูลในจอภาพไม่เปลี่ยน

3. ENTRY MODE SET



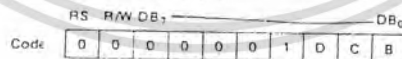
BIT I/D : โดยจะเป็นตัวกำหนดให้ DD RAM ADDRESS เพิ่มขึ้นหนึ่งโดย

1 = เพิ่ม

0 = ลดลงหนึ่ง

BIT S : เป็นตัวกำหนดแสดงผลโดยถ้า S = 1 จะเป็นการใส่ข้อมูลแล้วตัว CURSOR อยู่ที่ ที่ข้อมูลจะถูกค้นไปทางซ้าย ถ้า S = 0 ข้อมูลจะอยู่ที่ตัว CURSOR จะถูกค้น ไปทางขวามือ

4. DISPLAY ON/OFF CONTROL



BIT D : เป็น BIT ให้เปิดปิดหน้าจอภาพโดยถ้า

D = 1 จะ ON และ D = 0 จะ OFF

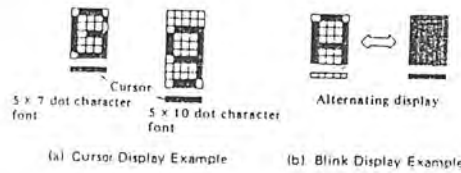
BIT C : จะให้แสดง CURSOR ให้ BIT C = 1 และถ้าไม่ต้องการแสดง CURSOR BIT C =

0 โดยตัว CURSOR จะอยู่ที่ LINE 8 ในแบบ 5X7 DOT และจะอยู่ LINE ที่ 11

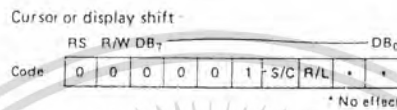
ในแบบ 5X10 DOT

BIT B : เป็น BIT SET การกระพริบของ CURSOR โดย B = 1 การกระพริบ B = 0 ไม่มี

การกระพริบ โดยมีระยะเวลาการกระพริบประมาณ 379.2 ms



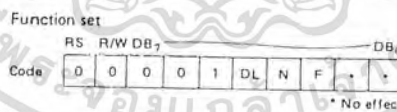
5. CURSOR OR DISPLAY SHIFT



เป็นคำสั่งกำหนดให้ตำแหน่ง CURSOR หรือข้อมูล ไปเกิดทางซ้ายหรือขวาโดยไม่ต้องใช้คำสั่งเขียนหรืออ่าน โดย

S/C	R/L	
0	0	ทำการย้าย CURSOR ไปจากตำแหน่งเดิมไปซ้ายมือ 1 ตำแหน่ง
0	1	ทำการย้าย CURSOR ไปจากตำแหน่งเดิมไปขวามือ 1 ตำแหน่ง
1	0	เป็นการค้นตัวอักษรที่เกิดไปทางซ้าย
1	1	เป็นการค้นตัวอักษรที่เกิดไปทางขวามือ

6. FUNCTION SET



BIT DL : เป็นการ SET การติดต่อว่าจะให้เป็นแบบ 8 BIT หรือ 4 BIT โดยถ้าต้องการติดต่อ 4 BIT DL = 0 และ 8 BIT DL = 1

N : เป็นการ SET บรรทัดการแสดงผล N = 0 แสดง 1 บรรทัด

N = 1 แสดง 2 บรรทัด ในกรณีมากกว่า 2 บรรทัด ก็ให้ SET N = 1

F : เป็นการ SET ขนาด DOT การแสดงผล 5X7 หรือ 5X10 โดย

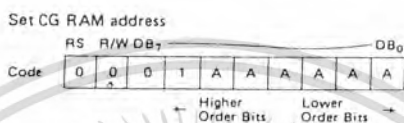
F = 0 เป็นแบบ 5X7 F = 1 เป็นแบบ 5X10

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

N F	No. of display lines	Character font	Duty factor	Remarks
0 0	1	5 x 7 dots	1/8	
0 1	1	5 x 10 dots	1/11	
1 *	2	5 x 7 dots	1/16	Cannot display 2 lines with 5 x 10 dot character font.

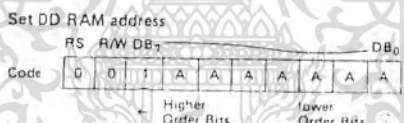
* No effect

7. SET CG RAM ADDRESS



ใน HD44780 นั้นจะมีหน่วยความจำอยู่ 2 ชุด คือ DISPLAY DATA RAM (DD RAM) จำนวน 80X8 BIT และ CHARACTER GENERATOR RAM CG RAM จำนวน 512 BIT และ 7200 BIT คำสั่งนี้จะเป็นการ SET ADDRESS ใน CG RAM โดยทำการ SET ADDRESS ก่อนเขียนหรืออ่านข้อมูลจาก CG RAM ด้วย

8. SET DD RAM ADDRESS



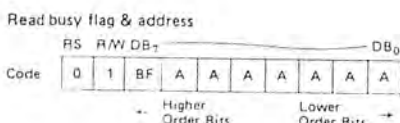
เป็นคำสั่ง SET ค่า ADDRESS ใน DD RAM ในการเขียนหรืออ่านค่าจาก DD RAM (DD RAM คือ ส่วนที่จะแสดงผลหน้าจอ LCD) โดยจำนวน ADDRESS ที่จะเกิดขึ้นบนจอ LCD จะอยู่กับ SET ค่า N ด้วย

N = 0 (1 บรรทัด) ADDRESS จะอยู่ 00H-4FH

N = 1 (2 บรรทัด) ADDRESS จะอยู่ 00H-27H สำหรับบรรทัดที่ 1 และ 40H-67H สำหรับ บรรทัดที่ 2

แบบการจัด ADDRESS ของ DD RAM หน้าจอ LCD แบบ 16 ตัวอักษร 1 บรรทัด , 16 ตัวอักษร 2 บรรทัด , 16 ตัวอักษร 4 บรรทัด , 20 ตัวอักษร 1 บรรทัด , 20 ตัวอักษร 2 บรรทัด และ 40 ตัวอักษร 2 บรรทัด

9. READ BUSY FLAG AND ADDRESS



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่นอนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

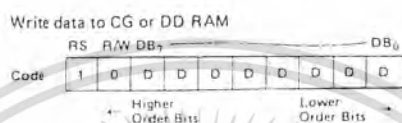
BUSY FLAG ซึ่งจะเป็นตัวบอกว่าตัว HD44780 นี้อยู่ในขบวนการทำงานภายในอยู่หรืออยู่ในสภาพพร้อมจะรับข้อมูล โดย

BF = 1 อยู่ในขบวนการทำงานภายในไม่พร้อมจะรับข้อมูลหรือคำสั่ง

BF = 0 พร้อมจะรับข้อมูลหรือคำสั่งได้

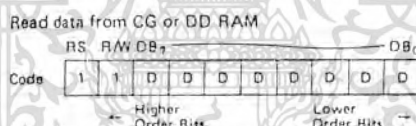
และนอกจากนี้ยังเป็นคำสั่งอ่านค่าข้อมูล ADDRESS ของ CG RAM หรือ DD RAM ด้วย

10. WRITE DATA TO CG หรือ DD RAM



เป็นคำสั่งเขียนข้อมูลเข้าไปใน CG หรือ DD RAM โดยเมื่อเขียนข้อมูลและ ADDRESS จะเพิ่มหรือลดโดยอัตโนมัติตามคำสั่งที่ SET ใน ENTRY MODE ข้อกำหนดที่จะรู้ว่าเป็นการเขียนข้อมูลของ CG RAM หรือ DD RAM ทำได้โดยการ SET ADDRESS ของ CG RAM หรือ DD RAM ขึ้นมาก่อนจะเขียนข้อมูล

11. READ DATA FROM CG OR DD RAM



เป็นคำสั่งอ่านค่าข้อมูลจาก CG RAM หรือ DD RAM โดยก่อนอ่านค่าจาก DD RAM หรือ CG RAM นี้ควรจะใช้คำสั่ง SET ADDRESS ก่อนเพื่อให้รู้ว่าข้อมูลที่อ่านได้นั้นเป็น DD หรือ CG RAM จากตารางการทำงานจะเห็นว่าการใช้งาน LCD MODULE นั้นง่ายเพียงแต่เราส่งคำสั่งเริ่มแรกและ SET ความต้องการในขนาดตัวอักษร , CURSOR หลังจากนั้นเราก็สามารถเขียนตัวอักษรเข้าไปใน DD RAM ตามตารางตัวอักษรที่ให้มานั้นก็จะเกิดอักษรในจอภาพ LCD เรายังสามารถกำหนดตำแหน่งตัวอักษรที่จะเกิดบนจอได้โดยการ SET DD RAM ADDRESS ตามตารางที่ให้มาในหัวข้อ SET DD RAM ADDRESS

ข่าต่างๆในการต่อใช้งาน HD44780

1. RS (REGISTOR SELECTION) จะเป็นขาเลือก REGISTOR ภายในซึ่งมีอยู่ 2 ตัวคือ INSTRUCTION REGISTOR (IR) และ DATA REGISTOR (DR)

โดยถ้าเป็น 1 จะเป็นการเลือก DATA และถ้าเป็น 0 จะเป็นการเลือก INSTRUCTION

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่นอนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- R/W (READ/WRITE) เป็นตัวเลือกว่าจะเขียนหรือจะอ่านข้อมูลจากตัว IC โดยอ่านข้อมูล = 1, เขียนข้อมูล = 0
- E (ENABLE SIGNAL) เป็นขากำหนดสภาพการรับเขียนอ่านข้อมูล

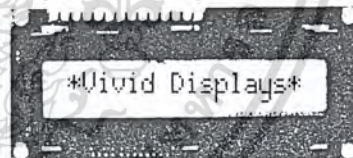
Register selection

RS	R/W	E	Operation
0	0		IR write as internal operation (Display clear, etc.)
0	1		Read busy flag (DB ₇) and address counter (DB ₀ - DB ₄)
1	0		DR write as internal operation (DR to DD or CG RAM)
1	1		DR read as internal operation (DD or CG RAM to DR)

- DB0 - DB7 เป็นขารับส่งข้อมูลจากตัว IC
- VDD ไฟเลี้ยงตัววงจร +5V
- VSS เป็นขา GND
- VO เป็นขารับ VOLTAGE ในการขับ LCD ให้สว่างหรือมืด

PIN CONNECTION

Pin No	Symbol	Level	Function
1	V _{SS}	—	DV
2	V _{DD}	—	+5V
3	V _{DD}	—	
4	RS	H/L	1: Instruction write mode 0: Data input
5	R/W	H/L	1: Data read (LCD module - MPU) 0: Data write (LCD module - MPU)
6	E	H, H-L	Enable signal
7	DB0	H/L	Data bus line Note (1), (2)
8	DB1	H/L	
9	DB2	H/L	
10	DB3	H/L	
11	DB4	H/L	
12	DB5	H/L	
13	DB6	H/L	
14	DB7	H/L	



เราสามารถต่อ VR ปรับค่าได้ 2 แบบ

- ต่อ GND LCD
- LCD LM2017

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

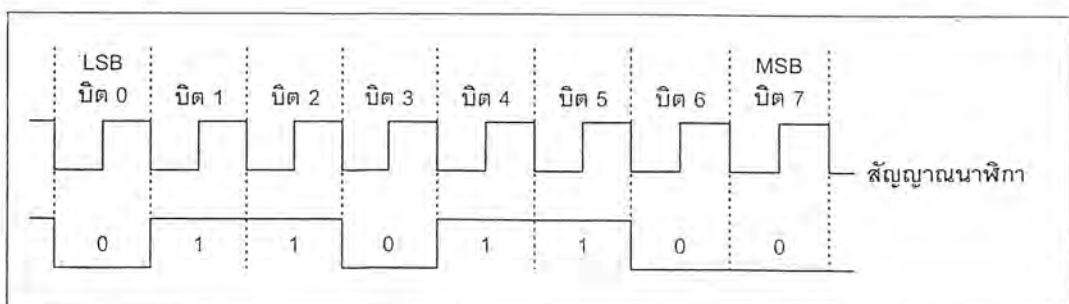
2.3 พอร์ตอนุกรม (Serial Port)

การเคลื่อนย้ายข้อมูลจากคอมพิวเตอร์ไปยังอุปกรณ์ต่อพ่วงอื่นๆ หรือคอมพิวเตอร์ด้วยกัน นั้นคือการรับส่งข้อมูลแบบขนานและการรับส่งข้อมูลแบบอนุกรม การรับส่งข้อมูลแบบขนาน จะเป็นการรับส่งข้อมูลคราวละ 4 หรือ 8 บิต ในเวลาเดียวกัน ซึ่งจะทำให้การรับส่งข้อมูลทำได้ที่ ความเร็วสูง ซึ่งก็หมายความว่าจำนวนของสายที่ใช้ในการส่งจะต้องมีมากเท่ากับจำนวนบิตของข้อมูลที่จะส่งด้วย นอกจากนี้ยังจะต้องรวมถึงสายที่ใช้สำหรับการควบคุมและการตรวจสอบการรับส่งข้อมูล ซึ่งอาจจะต้องใช้สายมากเป็น 2 เท่าของจำนวนบิตข้อมูลที่จะส่งได้ ซึ่งก็เป็นปัญหาในเรื่องราคาของสายที่ใช้ในการเชื่อมต่อแบบขนานมักจะมีราคาแพง

ในขณะที่การรับส่งข้อมูลแบบอนุกรมเป็นการรับส่งข้อมูลครั้งละ 1 บิต แต่ก็สามารถรับส่งข้อมูลได้คราวละหลายๆบิตได้ หากแต่จะต้องมีการตกลงกันระหว่างตัวส่งและตัวรับว่า จะรับส่งข้อมูลคราวละกี่บิต ตัวรับจะต้องรอข้อมูลมาให้ครบทุกบิตเสียก่อนจึงทำการประมวลผล ส่งผลให้ การสื่อสาร ข้อมูลอนุกรมอาจมีความเร็วต่ำกว่าแบบขนาน ในด้านจำนวนสายสัญญาณการรับส่งข้อมูลแบบอนุกรมจะใช้จำนวนสายที่น้อยกว่ามาก อย่างน้อยที่สุดใช้เพียง 2-3 เส้นเท่านั้น แต่อัตราเร็วในการรับส่งข้อมูลอาจต่ำกว่าแบบขนาน อย่างไรก็ตามการรับส่งข้อมูลแบบอนุกรมสามารถใช้สายสัญญาณที่มีความยาวมากกว่าแบบขนาน ทำให้ระยะทางในการสื่อสารข้อมูลแบบอนุกรมสามารถทำได้มากกว่า

2.3.1 การสื่อสารแบบอนุกรม

การสื่อสารแบบอนุกรมนั้นจะแบ่งออกได้เป็น 2 แบบคือการสื่อสารอนุกรมแบบซิงโครนัส และการสื่อสารอนุกรมแบบอะซิงโครนัส การสื่อสารแบบซิงโครนัสจะมีสัญญาณนาฬิกา ร่วมอยู่กับการรับและส่งสัญญาณด้วย ตัวอย่างการส่งข้อมูลแบบซิงโครนัสก็คือซีบีอาร์ตของคอมพิวเตอร์ ซึ่งสายเส้นหนึ่งจะเป็นสายของสัญญาณนาฬิกา ส่วนสายอีกเส้นจะเป็นสายของข้อมูล ดังนั้นการติดต่อกันแบบซิงโครนัสนี้จะต้องใช้สายในการเชื่อมต่ออย่างน้อยที่สุด 3 เส้นคือ สัญญาณนาฬิกา, ข้อมูล และกราวด์รูปที่ 2-10 แสดงให้เห็นถึงไทมิง ไดอะแกรมของการส่งข้อมูลแบบซิงโครนัส



รูปที่ 2-10 รูปแบบอย่างง่ายที่สุดของข้อมูลแบบอนุกรม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับใช้ภายในเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

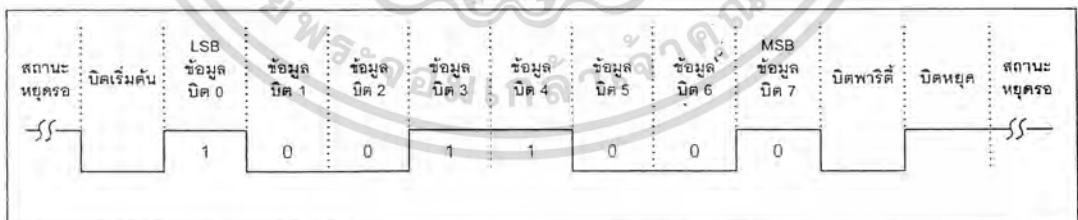
2.3.2 การสื่อสารข้อมูลแบบอะซิงโครนัส

การสื่อสารข้อมูลแบบอะซิงโครนัส คือการรับส่งข้อมูลไปในสายโดยไม่จำเป็นต้องมีสัญญาณนาฬิกาาร่วมด้วยเหมือนกับการรับส่งข้อมูลแบบซิงโครนัส แต่จะใช้การกำหนดค่าสัญญาณนาฬิกาทั้งภาครับและภาคส่งให้มีค่าเท่ากัน ซึ่งเรียกว่าสัญญาณนาฬิกาที่ใช้ในการกำหนดค่าให้ค่าให้ภาครับและภาคส่งนี้ว่า อัตราการถ่ายทอดข้อมูล หรือ บอดเรต (baudrate) มีหน่วยเป็นบิตต่อวินาที (bit per second : bps)

รูปแบบของข้อมูลที่ใช้ในการรับส่งข้อมูลแบบอะซิงโครนัสประกอบด้วย 4 ส่วนด้วยกันคือ

1. บิตเริ่มต้น (Start Bit) ซึ่งมีขนาด 1 บิต
2. บิตข้อมูลแบบอนุกรมจะมีขนาด 5,6,7 หรือ 8 บิต
3. บิตตรวจสอบพาริตี (Parity Bit) จะมีขนาด 1 บิตหรือไม่
4. บิตปิดท้าย (Stop Bit) จะมีขนาด 1,1.5 หรือ 2 บิต

รูปที่ 2-11 แสดงรูปแบบของข้อมูลอนุกรมแบบอะซิงโครนัส ซึ่งเมื่อไม่มีข้อมูลที่จะส่งหา DATA จะมีสถานะลอจิก “1” ซึ่งจะเรียกสถานะนี้ว่าสถานะหยุดรอ (waiting state) การเริ่มต้นส่งข้อมูลจะเริ่มจากการให้หา DATA มีลอจิก “0” ด้วยช่วงระยะเวลา 1 บิต ซึ่งจะเรียกบิตนี้ว่าบิตเริ่มต้น จากนั้นบิตข้อมูลจะถูกส่งออกไป โดยเริ่มจากบิตที่มีนัยสำคัญต่ำสุด (LSB) ก่อน ซึ่งข้อมูลในไบต์ที่จะส่งอาจจะมีจำนวนบิต 5, 6, 7 หรือ 8 บิตก็ได้ จากนั้นจะตามด้วยพาริตี ซึ่งใช้เพื่อตรวจสอบความผิดพลาดที่เกิดขึ้นจากการส่งข้อมูล บิตสุดท้ายที่จะส่งคือบิตปิดท้าย ซึ่งจะให้ขาดามีสถานะลอจิก 1 อีกครั้งด้วยระยะเวลาอย่างน้อย 1 บิต, 1.5 บิตหรือ 2 บิต เพื่อเป็นการแสดงว่าสิ้นสุดข้อมูลแล้ว



รูปที่ 2-11 รูปแบบอย่างง่ายที่สุดของข้อมูลอนุกรมแบบอะซิงโครนัส

อุปกรณ์พิเศษที่ได้รับการออกแบบมาสำหรับการรับและการส่งข้อมูลแบบอะซิงโครนัส เรียกว่า Universal Asynchronous Receiver /Transmitter หรือ UART อัตราความเร็วในการรับและส่งข้อมูลของการรับส่งข้อมูลแบบอะซิงโครนัส คือ ค่าบอดเรต ซึ่งก็คือค่าจำนวนบิตต่อวินาทีที่ใช้

ในการรับส่งข้อมูล บอดเรตมาตรฐานที่ใช้สำหรับพอร์ตอนุกรม RS-232 ได้แก่ 110, 150, 300, 600, เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่นอนุญาตให้นำไปใช้ประโยชน์ทางการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1200, 2400, 4800, 9600 และ 19200 บิตต่อวินาที และมีค่าเพิ่มมากขึ้นตามเทคโนโลยีของคอมพิวเตอร์

การตรวจสอบพาริตีสามารถกำหนดให้เป็นแบบคี่ (odd), แบบคู่ (even) หรือไม่มีการตรวจสอบพาริตีก็ได้ การตรวจสอบพาริตีเป็นการตรวจสอบจำนวนรวมของของบิตที่เป็นลอจิก “1” ภายในข้อมูลที่ส่งไป 1 ไบต์ว่ามีจำนวนรวมเป็นเลขคู่หรือเลขคี่โดยต้องรวมบิตพาริตีเข้าไปด้วย บิตพาริตีถูกสร้างขึ้นจากภาคส่งข้อมูลของ UART ซึ่งทางภาครับจะต้องทำการกำหนด คุณสมบัติการตรวจสอบพาริตีให้ตรงกันว่าจะตรวจสอบพาริตีคู่หรือพาริตีคี่ จากนั้นภาครับของ UART จะทำการตรวจสอบค่าพาริตีที่เกิดขึ้นว่าเป็นคู่หรือคี่ โดยการนับจำนวนลอจิก “1” ทั้งหมดรวมทั้งภาคบิตพาริตีด้วย ถ้ากำหนดพาริตีไว้เป็นคู่แต่อ่านค่าตัวเลขในการนับออกมาได้เป็นเลขคี่ทางภาครับจะแสดงข้อผิดพลาดออกมาให้ผู้ใช้งาน นับเป็นการตรวจสอบความผิดพลาดที่เกิดขึ้นในการถ่ายทอดข้อมูลที่ง่ายที่สุด แต่จะเชื่อถือได้เมื่อมีบิตข้อมูลที่ทำการส่งผิดพลาดเพียงบิตเดียวเท่านั้น ถ้าข้อมูลที่ทำการส่งมีบิตที่ผิดพลาดมากกว่า 1 บิต การตรวจสอบด้วยวิธีนี้จะไม่ได้ผล สำหรับการตั้งพาริตีบิตเป็น NONE นั้นทั้งภาครับและภาคส่ง จะไม่มีการตรวจสอบพาริตี

คอมพิวเตอร์ในรุ่น AT เกือบทั้งหมดจะใช้ UART เบอร์ 16450 และ 16550 ส่วนคอมพิวเตอร์ในรุ่น XT ใช้ UART เบอร์ 8250 ชิพเหล่านี้มีระดับแรงดันเป็นแบบทีทีแอลแค้เพื่อให้มีแรงดันเป็นไปตามมาตรฐาน RS 232 และเพื่อให้การรับส่งข้อมูลสามารถทำได้ที่ระยะทางไกลมากขึ้น ระดับแรงดัน ทีทีแอลจะถูกแปลงเป็นระดับแรงดันที่สูงขึ้น โดยลอจิก “0” มีระดับแรงดัน +3V ถึง -12V ในขณะที่ลอจิก “1” มีระดับแรงดัน -3V ถึง -12V

2.3.3 มาตรฐานของพอร์ตอนุกรมแบบ RS-232

มาตรฐานของการเชื่อมต่อแบบอนุกรม RS-232 เป็นมาตรฐานอุตสาหกรรมที่ออกแบบมาเพื่อใช้ในการส่งข้อมูลอนุกรมแบบอะซิงโครนัส 2 ทิศทาง โดยมาตรฐาน RS-232 ในอดีตนั้นถูกออกแบบมาเพื่อการส่งผ่านข้อมูลจากคอมพิวเตอร์ไปยังโมเด็มเพียงอย่างเดียว เพื่อที่จะนำข้อมูลจากโมเด็มนี้สื่อสารผ่านสายโทรศัพท์ไปยังคอมพิวเตอร์อีกชุดหนึ่งซึ่งอยู่ห่างไกลกัน โดยคณะกรรมการที่เรียกว่า สมาคมอุตสาหกรรมอิเล็กทรอนิกส์ (Electronic Industrial Association : EIA) ได้วางมาตรฐานที่มีชื่อเรียกกันว่า EIA RS-232 มาตรฐานนี้ในช่วงแรกจะใช้คอนเน็กเตอร์เป็นแบบ DB-25 โดยกำหนดความยาวสูงสุดของสายสัญญาณไว้ที่ 50 ฟุตมีระดับสัญญาณตั้งแต่ -3 ถึง -12 V แสดงว่ามีข้อมูล(Mark) และ +3 ถึง +12V แสดงว่าเป็นช่องว่าง (Space)

มาตรฐาน RS-232 ได้กำหนดรูปแบบของอุปกรณ์เชื่อมต่อข้อมูล (Data Terminal Equipment : DTE) กับวงจรปลายทาง (Data Circuit Terminating : DCE) ไว้ว่า อุปกรณ์ DTE จะเอกลำนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์การคำนวณว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ต้องเป็นอุปกรณ์ที่มีการประมวลผลในตัวเช่น ไมโครคอนโทรลเลอร์หรือ ไมโครคอมพิวเตอร์ ซึ่งมี
ความสามารถในการสร้างบิตข้อมูลแบบอนุกรมได้ ส่วนอุปกรณ์ DCE จะทำหน้าที่เป็นเพียงตัวรับ
ข้อมูลที่ส่งมาจาก DCE เท่านั้น โดยการรับส่งข้อมูลระหว่างอุปกรณ์ทั้งสองจะกระทำผ่านมาตรฐาน
RS-232

ข้อแตกต่างของอุปกรณ์ DTE และอุปกรณ์ DCE อย่างหนึ่งที่ได้เห็นได้ชัดคือ คอนเน็กเตอร์
ของ DTE จะเป็นตัวผู้ ส่วนคอนเน็กเตอร์ของ DCE จะเป็นตัวเมีย ซึ่งพอร์ตอนุกรมของคอมพิวเตอร์
ที่ใช้กันอยู่ทั่วไปจะเป็นแบบ DTE ส่วนคอนเน็กเตอร์ที่อยู่ที่ไม่เต็มจะเป็นแบบ DCE
คอนเน็กเตอร์สำหรับพอร์ต RS-232 และการเชื่อมต่อ

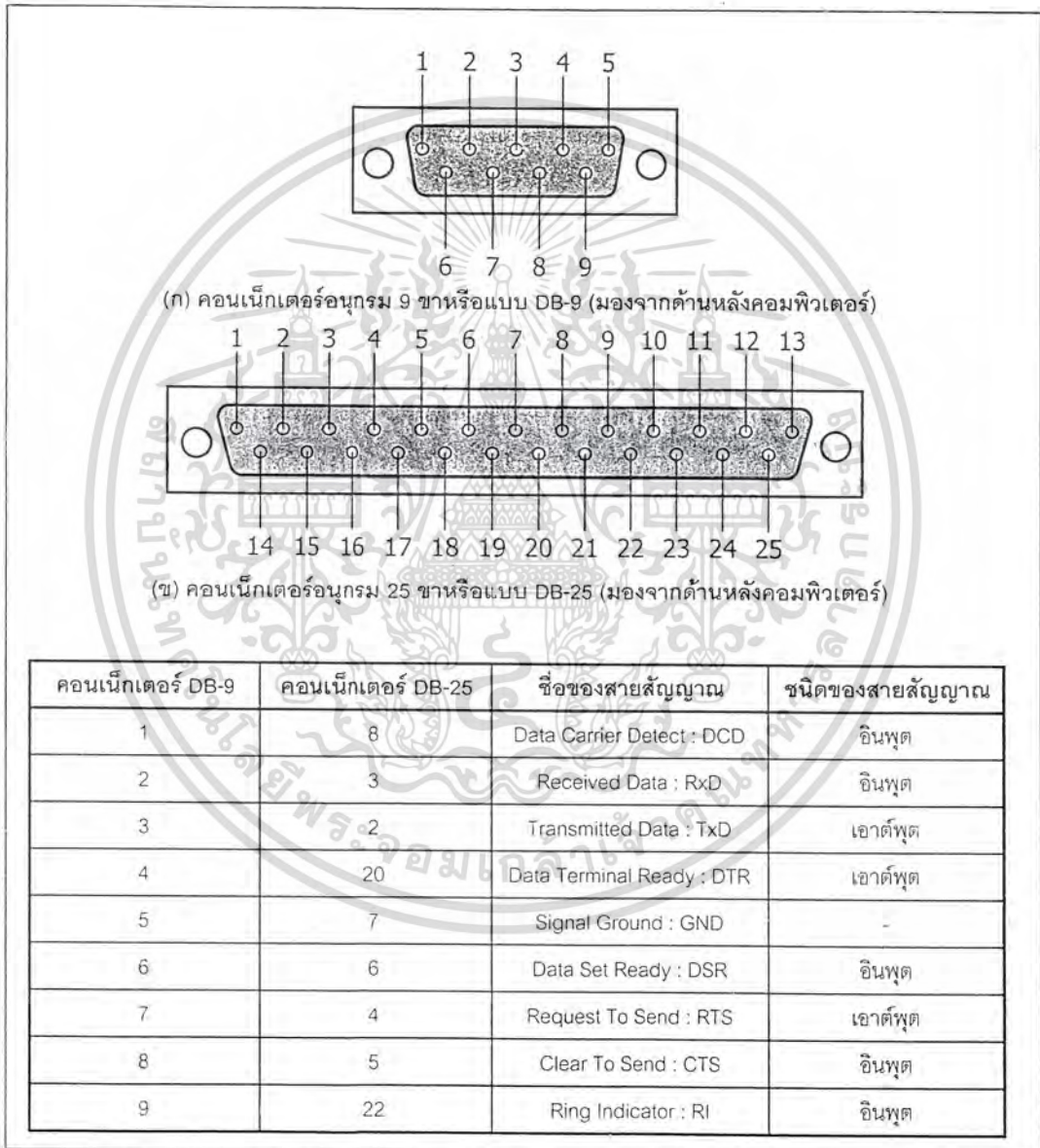
มาตรฐานการเชื่อมต่อแบบ RS-232 จะใช้คอนเน็กเตอร์แบบ DB-25 ตัวผู้หรือ DB-9 ตัวผู้
ซึ่งคอนเน็กเตอร์แบบ DB-25 จะมีขาต่อใช้งานเพียง 9 เส้นเช่นเดียวกับคอนเน็กเตอร์แบบ DB-9
เนื่องจากขาอื่นๆ ที่เคยใช้งานในอดีต ปัจจุบันมีการใช้งานไม่มากนัก จึงถูกยกเลิกไป โดยแสดงรูป
ร่างและตำแหน่งขาในรูปที่ 2-12

สำหรับการเชื่อมต่อคอมพิวเตอร์กับอุปกรณ์ภายนอกแสดงดังในรูปที่ 2-13 ลูกศรในรูป
แสดงทิศทางของข้อมูล ในรูป 2-13 (ก) เป็นการเชื่อมต่อแบบ Null modem หรือการเชื่อมต่อโดย
ตรงโดยไม่ต้องผ่านโมเด็ม โดยมีการตรวจสอบแฮนด์เช็กเต็มรูปแบบ ส่วนในรูปที่ 2-13 (ข) เป็นการ
เชื่อมต่อแบบ Null modem ในลักษณะที่ใช้สายสัญญาณเพียง 3 เส้นโดยเส้นหนึ่งสำหรับส่งข้อมูล
อีกเส้นสำหรับรับข้อมูล และเส้นสุดท้ายเป็นกราวด์ สำหรับรายละเอียดหน้าที่การทำงานของพอร์ต
อนุกรม RS-232 มีดังนี้

- **Data Carrier Detect** : DCD หรืออาจเรียกว่า Carrier Detect : CD ขานี้จะแอกติฟเมื่อมีการส่ง
สัญญาณพาห้จากอุปกรณ์สื่อสารข้อมูล เช่น โมเด็ม สำหรับการใช้งานปกติ ขานี้ จะไม่ได้ใช้
งานมากนัก
- **Receive Data** : RD หรือ RxD ขานี้ใช้เพื่อรับสัญญาณอนุกรมเข้ามายังคอมพิวเตอร์ โดยนำข้อ
มูลที่อ่านได้ไปเก็บไว้ในรีจิสเตอร์ บัฟเฟอร์
- **Transmitted Data** : TD หรือ TxD ขานี้ใช้เพื่อส่งข้อมูลออกจากคอมพิวเตอร์ โดยนำข้อมูลที่
เก็บอยู่ในบัฟเฟอร์สำหรับส่งข้อมูลสำหรับส่งข้อมูลออกไป
- **Data Terminal Ready** : DTR เป็นขาสัญญาณที่ส่งออกจากคอมพิวเตอร์ เพื่อให้อุปกรณ์ปลายทาง
ทางรับรู้ว่า ต้องการติดต่อด้วย โดยขา DTR นี้จะต้องเชื่อมต่อกับขา DSR ของอุปกรณ์ปลายทาง
และขา DTR ของอุปกรณ์ปลายทางจะต้องเชื่อมต่อกับขา DSR ของคอมพิวเตอร์ ถ้าใช้การเชื่อม
ต่อเป็นแบบ Null modem ซึ่งใช้สายในการเชื่อมต่อเพียง 3 เส้น จะต้องต่อขา DTR และ DSR

ของตัวมันเองเข้าด้วยกันและจะต้องต่อขา DCD ด้วยในกรณีที่โปรแกรมสื่อสารที่ใช้มีการตรวจ
จับสัญญาณพาห์

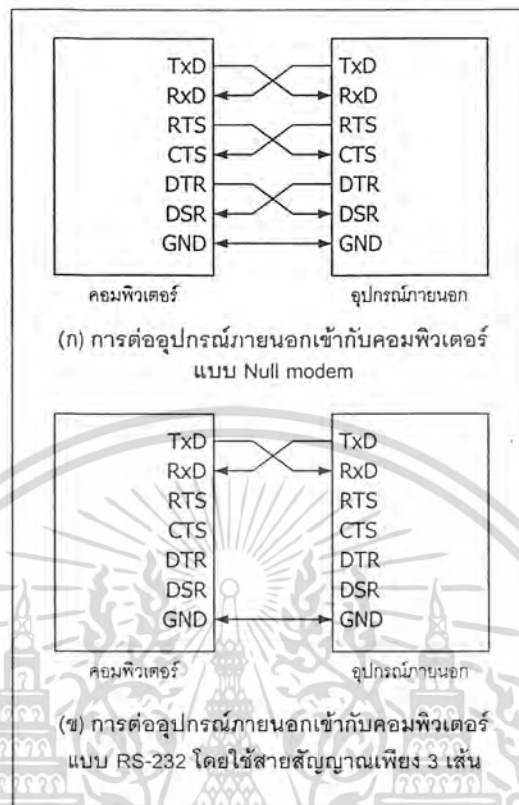
- Signal Ground : GND ขากราวด์ของระบบ



รูปที่ 2-12 การจัดขาของคอนเน็กเตอร์พอร์ตอนุกรม

ตามมาตรฐาน RS-232 ทั้งแบบ DB-9 และ DB-25

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2-13 การต่ออุปกรณ์ภายนอกกับพอร์ตอนุกรมของคอมพิวเตอร์ในลักษณะต่างๆ

- Data Set Ready : DSR ขานี้จะใช้คู่กับขา DTR เพื่อตรวจสอบการเชื่อมต่อกันระหว่างคอมพิวเตอร์กับอุปกรณ์ปลายทาง ซึ่งขา DSR นี้จะเป็นขาสำหรับข้อมูลจากภายนอกซึ่งถูกส่งมาจากขา DTR
- Request To Send : RTS เป็นขาสำหรับส่งสัญญาณร้องขอให้ทางอุปกรณ์ปลายทางส่งข้อมูลมายังคอมพิวเตอร์ โดยที่ขารับสัญญาณ RTS ก็คือขา CTS ในกรณีที่ใช้การเชื่อมต่อแบบ Null modem 3 สาย จะต้องเชื่อมต่อขา RTS และ CTS ของตัวมันเองเข้าด้วยกัน เพื่อให้การรับและส่งข้อมูลสามารถเกิดขึ้นได้ตลอดเวลา
- Clear To Send : CTS ขานี้จะคอยรับสัญญาณจากขา RTS เมื่อ รับสัญญาณได้ ข้อมูลที่ขา TxD จะถูกส่งออกไป ดังนั้นขานี้จึงถูกใช้เพื่อตรวจสอบอุปกรณ์ต่อพ่วงว่าพร้อมที่จะรับข้อมูลหรือไม่

- Ring Indicator : RI ใช้แสดงสถานะสัญญาณเรียกจากโทรศัพท์ ปกติในการสื่อสารโดยทั่วไป สายนี้จะไม่ถูกใช้งาน จะใช้งานก็ต่อเมื่อมีการเชื่อมต่อกับโมเด็มและโปรแกรมมีการตรวจสอบสัญญาณนี้เท่านั้น

2.3.4 UART

UART มาจากคำว่า Universal Asynchronous Receiver Transmitter ซึ่งหมายถึงอุปกรณ์ที่ทำหน้าที่รับและส่งข้อมูลแบบอะซิงโครนัสนั่นเอง

หน้าที่หลักของ UART คือทำหน้าที่แปลงข้อมูลที่อยู่ในรูปแบบขนานจากคอมพิวเตอร์ให้อยู่ในรูปแบบอนุกรมแบบอะซิงโครนัส แล้วส่งออกไป และทำหน้าที่แปลงสัญญาณอนุกรมแบบอะซิงโครนัสที่ป้อนเข้ามายัง UART ให้เป็นแบบขนานก่อนที่จะส่งเข้าสู่คอมพิวเตอร์ ซึ่งนอกจาก UART จะส่งข้อมูลไปยังคอมพิวเตอร์แล้ว ยังแจ้งข้อมูลอื่นๆ ให้กับคอมพิวเตอร์รับทราบด้วย เช่น อัตราเร็วในการรับส่งข้อมูล (บอดเรต), รูปแบบการส่งข้อมูล, ความผิดพลาดที่เกิดขึ้นระหว่างการถ่ายทอดข้อมูล (ผิดพลาดจากพาริตี, เฟรมข้อมูล, โอเวอร์รัน) เป็นต้น

ภายใน UART จะมีส่วนของวงจรสร้างบอดเรตแบบโปรแกรมได้ (Programmable baudrate generator) โดยการกำหนดค่าตัวหารให้กับสัญญาณนาฬิกาของ UART โดยตัวหารนี้มีขนาด 16 บิต ดังนั้นจึงสามารถกำหนดตัวหารอยู่ในช่วง 1- 65,535 UART สามารถรับส่งข้อมูลได้ทั้งแบบฮาล์ฟดูเพล็กซ์ (half duplex) และฟูลดูเพล็กซ์ (full duplex) โดยการส่งแบบฮาล์ฟดูเพล็กซ์เป็นการส่งทิศทางเดียว ส่วนการส่งแบบฟูลดูเพล็กซ์นั้นสามารถรับและส่งข้อมูลได้ในคราวเดียวกัน

ชนิดของ UART

ในเครื่องคอมพิวเตอร์ทั่วไปมี UART ที่ใช้งานกันอยู่ 2 เบอร์คือ 8250 ซึ่งเป็น UART มาตรฐานที่มีการใช้กันมายาวนาน UART เบอร์นี้จะมียัพเพอร์สำหรับรับและส่งข้อมูลตำแหน่งเดียวกัน ทำให้การรับและส่งข้อมูลถูกจำกัดด้วยความเร็วอยู่ที่ 57.6 กิโลบิตต่อวินาทีเท่านั้น แต่ UART เบอร์นี้ก็ถือว่าเป็นต้นแบบของ UART ที่ใช้ในคอมพิวเตอร์ โดยคอมพิวเตอร์ทุกรุ่นจะต้องสนับสนุนการทำงานตามรูปแบบของ UART เบอร์นี้

UART อีกเบอร์หนึ่งคือ 16450 มีความสามารถรับส่งข้อมูลได้ด้วยความเร็ว 115,200 บิตต่อวินาที และเพิ่มรีจิสเตอร์สำหรับพักข้อมูลสำหรับ UART นอกจากนั้นยังเพิ่มส่วนของชิพรีจิสเตอร์แบบ FIFO (First In First Out) ขนาด 16 ไบต์เข้าไปทำให้สามารถสนับสนุนความเร็วในการรับส่งข้อมูลที่ 256 กิโลบิตต่อวินาทีได้ โดยคอมพิวเตอร์ในปัจจุบันใช้ UART เบอร์นี้หรือใหม่กว่า เช่น เบอร์ TL16C750 ซึ่งมีรีจิสเตอร์แบบ FIFO ขนาด 64 ไบต์ ทำงานได้ที่ระดับแรงดัน +5V และ +3V *

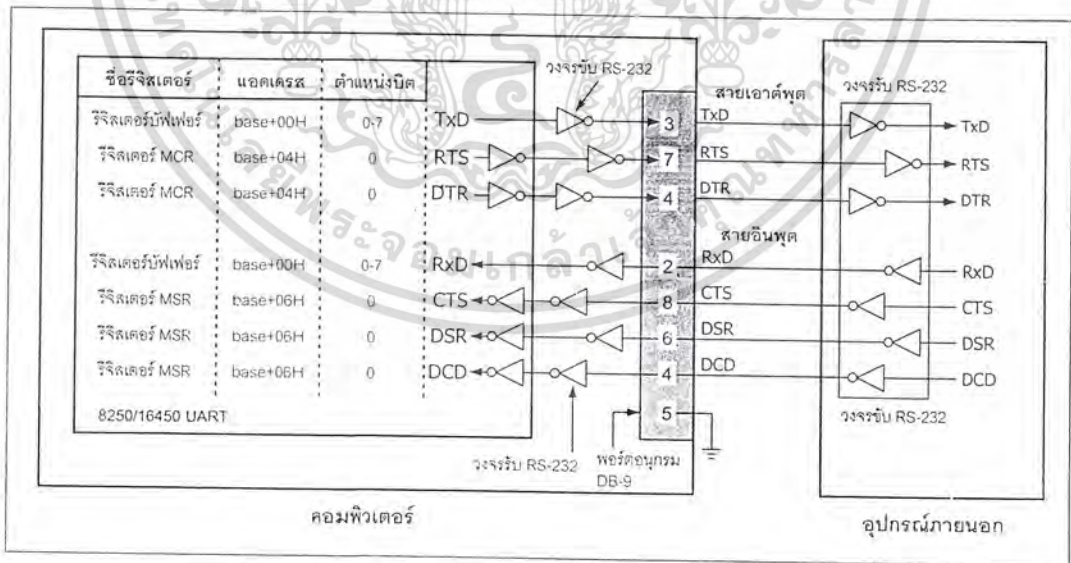
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการศึกษาเท่านั้น ไม่อนุญาตให้เผยแพร่หรือใช้ซ้ำโดยไม่ได้รับอนุญาต
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

มีโหมดประหยัดพลังงาน สามารถรับส่งข้อมูลได้ด้วยความเร็ว 1 เมกะบิตต่อวินาที เมื่อใช้สัญญาณนาฬิกา 16 MHz

อย่างไรก็ตาม ความเร็วในการส่งข้อมูลที่มากมายของ UART เบอร์ใหม่ๆ ก็ไม่ได้ช่วยให้การรับส่งข้อมูลของคอมพิวเตอร์เร็วขึ้น เนื่องจากว่าคอมพิวเตอร์ยังใช้ความถี่ของสัญญาณนาฬิกาในการแปลงข้อมูลเพียง 1.8432 MHz เท่านั้น

2.3.5 ลักษณะสัญญาณอินพุตและเอาต์พุตของพอร์ตอนุกรม

สัญญาณเอาต์พุตที่ใช้ควบคุม (RTS และ DTR) และสัญญาณแสดงสถานะอินพุต (CTS, DSR และ DCD) ของพอร์ตอนุกรม RS-232 จะถูกกลับสถานะภายในตัว UART ส่วนสัญญาณทั้งภาคส่งและรับจะไม่ถูกกลับสถานะ UART จะให้ระดับสัญญาณเอาต์พุตออกมาเป็นแบบทีทีแอลเท่านั้น ดังนั้นเมื่อสัญญาณถูกส่งออกมาจาก UART จึงต้องส่งเข้าสู่วงจรขับ เพื่อปรับระดับแรงดันให้ได้ระดับสัญญาณเป็นไปตามมาตรฐาน RS-232 ก่อนส่งออกไปจากคอมพิวเตอร์ สำหรับอุปกรณ์ต่อเชื่อมปลายทางก็จะต้องมีวงจรขับในลักษณะนี้เช่นเดียวกัน เพื่อให้ระดับสัญญาณในระดับเดียวกัน แต่วงจรขับที่ใช้ทั้งภายในคอมพิวเตอร์และอุปกรณ์ต่อเชื่อมปลายทางนั้นจะถูกกลับสถานะ ดังแสดงบล็อกไดอะแกรมในรูปที่ 2-14



รูปที่ 2-14 ไดอะแกรมแสดง โครงสร้างทางฮาร์ดแวร์ของพอร์ตอนุกรม

2.3.6 แอดเดรสของพอร์ตอนุกรม

แอดเดรสพื้นฐานของพอร์ตอนุกรมมีอยู่ 4 ตำแหน่งคือ

COM1 : 3F8H

COM2 : 2F8H

COM3 : 3E8H

COM4 : 2E8H

เมื่อเริ่มเปิดเครื่องภายในคอมพิวเตอร์ ไบออสภายในคอมพิวเตอร์จะทำการตรวจสอบแอดเดรสของพอร์ตอนุกรมทั้งหมด ถ้าไบออสพบแอดเดรสของพอร์ตอนุกรม ไบออสจะนำแอดเดรสที่ตรวจพบไปเก็บไว้ในหน่วยความจำขนาด 2 ไบต์ สำหรับพอร์ตอนุกรม COM1 จะเก็บไว้ที่แอดเดรส 0000:0401H ส่วนตำแหน่งอื่นๆมีรายละเอียดดังนี้

COM2 = 0000:0402H - 0000:0403H

COM3 = 0000:0404H - 0000:0405H

COM4 = 0000:0406H - 0000:0407H

นอกจากนี้ที่หน่วยความจำแอดเดรส 0000:0411H ยังใช้สำหรับแสดงจำนวนของพอร์ตอนุกรมที่มีใช้ภายในคอมพิวเตอร์อีกด้วย โดยมีรายละเอียดดังแสดงในตารางที่ 2-6

บิต 3	บิต 2	บิต 1	จำนวนพอร์ต
0	0	0	ไม่มีพอร์ตอนุกรม
0	0	1	มีพอร์ตอนุกรม 1 พอร์ต
0	1	0	มีพอร์ตอนุกรม 2 พอร์ต
0	1	1	มีพอร์ตอนุกรม 3 พอร์ต
1	0	0	มีพอร์ตอนุกรม 4 พอร์ต

ตารางที่ 2-6 แสดงข้อมูลในแอดเดรส 0000:0411H ที่ใช้แจ้งจำนวนพอร์ตอนุกรม

2.3.7 การเขียนโปรแกรมเพื่อใช้งานพอร์ตอนุกรม

มีหลากหลายวิธีในการเขียนโปรแกรมเพื่อการรับและส่งข้อมูลแบบอนุกรมผ่านพอร์ตอนุกรม RS-232 สำหรับการวิธีการที่จะกล่าวถึงจะเป็นใช้คอนโทรล MSCOMM32.OCX ของ Visual BASIC ในการติดต่อกับพอร์ตอนุกรมผ่านระบบปฏิบัติการวินโดวส์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

คอนโทรล MSComm

สำหรับการใช้งาน Visual BASIC ตั้งแต่เวอร์ชัน 2 เป็นต้นมา ใน Visual BASIC จะมีคอนโทรลสำหรับการสื่อสารอนุกรมผ่านทางพอร์ตอนุกรมของคอมพิวเตอร์มาให้ โดยใน Visual BASIC เวอร์ชัน 2 และ 3 จะใช้ชื่อว่า MSCOMM.VBX ส่วนเวอร์ชัน 4 ใช้ชื่อว่า MSCOMM16.OCX สำหรับการทำงานกับระบบปฏิบัติการ 16 บิต และ MSCOMM32.OCX สำหรับการทำงานกับระบบปฏิบัติการ 32 บิต สำหรับใน Visual BASIC เวอร์ชัน 5 จะมีเพียง MSCOMM32.OCX เท่านั้นเพราะถูกออกแบบมาให้ใช้งานกับระบบปฏิบัติการ 32 บิต

MSComm จัดเตรียมทางเลือกไว้ 2 ทาง เพื่อความสะดวกในการสื่อสารข้อมูล ทางแรกคือการสื่อสารข้อมูลที่กระตุ้นด้วยเหตุการณ์ (event-driven communication) เป็นรูปแบบการใช้งานที่มีประสิทธิภาพมากสำหรับการตอบสนองแบบทันทีทันใด เช่น เมื่อตัวอักษรถูกส่งมาที่พอร์ตอนุกรมหรือเกิดการเปลี่ยนแปลงที่ขา Data Carrier Detect (DCD) หรือขา Request To Send (RTS) เหตุการณ์ Oncomm ของ MSComm จะสามารถตรวจจับสัญญาณได้ทันที ซึ่งกล่าวถึงรายละเอียดในหัวข้อคุณสมบัติ CommEvent ต่อไป ส่วนทางเลือกที่สองเป็นการคอยตรวจสอบค่าเหตุการณ์และความผิดพลาดที่เกิดขึ้นด้วยการดูค่าที่เปลี่ยนแปลงภายในคุณสมบัติ CommEvent หลังจากให้โปรแกรมทำงานฟังก์ชันต่างๆ ไปเรียบร้อยแล้ว ซึ่งวิธีนี้ใช้งานได้ดีในกรณีที่โปรแกรมมีขนาดเล็ก

คอนโทรล MSComm ด้วสามารถควบคุมการทำงานของพอร์ตอนุกรมได้ 1 พอร์ตถ้าในโปรแกรมที่ใช้งานต้องการติดต่อกับพอร์ตอนุกรมมากกว่า 1 พอร์ตจะต้องใช้คอนโทรล MSComm มากกว่า 1 ตัวเพื่อควบคุมพอร์ตอนุกรมในแต่ละพอร์ต แอคเตสของพอร์ตอนุกรมและแอคเตสของการเกิดอินเตอร์รัปต์สามารถเปลี่ยนแปลงได้จากค่าที่ Control Panel

ถึงแม้ว่าคอนโทรล MSComm จะมีคุณสมบัติ (property) มากมายแต่สามารถทำความเข้าใจได้ไม่ยากดังนี้

● CommPort

ใช้ในการกำหนดและอ่านค่าพอร์ตอนุกรมที่ติดต่อกันอยู่ (COM1, COM2, COM3, COM4)

รูปแบบการใช้งานคำสั่ง `object.CommPort [= Value]`

โดย Value เป็นค่าของพอร์ตอนุกรม ชนิดของข้อมูลเป็น Integer ค่า Value สามารถกำหนดได้ในช่วง 1-16 (ค่าเริ่มต้นกำหนดไว้ที่ 1) เมื่อมีการกำหนดแล้วทำการเปิดพอร์ตโดยใช้คุณสมบัติ PortOpen แต่พอร์ตนั้นไม่มีอยู่ในระบบ MSComm จะแสดงข้อผิดพลาด error 68 ขึ้นมาซึ่งหมายถึงว่าอุปกรณ์ตัวนี้ไม่ได้อยู่ในระบบ ดังนั้นการเขียนโปรแกรมจึงจำเป็นต้องกำหนดตำแหน่งของพอร์ตอนุกรมก่อนที่ใช้คำสั่ง PortOpen

- **Setting**

ใช้ในการกำหนดและอ่านค่าอัตราบอด, พาริตี, จำนวนบิตของข้อมูล, จำนวนบิตปิดท้าย

รูปแบบการใช้งานคำสั่ง `object.Settings [= Value]`

ค่า Value มีชนิดของข้อมูลเป็นแบบ String มีรูปแบบเป็น “BBBB, P, D, S” โดย BBBB เป็นค่าอัตราบอด, P เป็นค่าพาริตี, D เป็นจำนวนของบิตข้อมูล และ S เป็นจำนวนของบิตปิดท้าย ปกติแล้วค่านี้ถูกกำหนดไว้เป็น “9600, N, 8, 1”

ค่าบอดเรตมาตรฐานที่ใช้กับ MSComm มีดังนี้

110 บิตต่อวินาที

300 บิตต่อวินาที

600 บิตต่อวินาที

1,200 บิตต่อวินาที

2,400 บิตต่อวินาที

9,600 บิตต่อวินาที (ค่าปกติ)

14,400 บิตต่อวินาที

19,200 บิตต่อวินาที

28,800 บิตต่อวินาที

38,400 บิตต่อวินาที (สงวน)

56,000 บิตต่อวินาที (สงวน)

128,000 บิตต่อวินาที (สงวน)

256,000 บิตต่อวินาที (สงวน)

สำหรับค่ามาตรฐานในการกำหนดพาริตีมีดังนี้

สัญลักษณ์ **รายละเอียด**

E พาริตีคู่

M ลอจิก “1” (Mark)

N ไม่มีใช้ (ค่าปกติ)

O พาริตีคี่ (Odd)

S ลอจิก “0” (Space)

ค่าที่ใช้ในการกำหนดจำนวนบิตมี 5 ค่าคือ 4, 5, 6, 7 และ 8 (เป็นค่าปกติ)

ค่าที่ระบุจำนวนบิตปิดท้ายมี 3 ค่าคือ 1 (เป็นค่าปกติ), 1.5 และ 2

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หมายเหตุ สาเหตุที่ค่าที่กำหนดจะต้องอยู่ในเครื่องหมายคำพูด “9600,N,8,1” เนื่องจาก ค่าที่กำหนดนี้อยู่ในรูปตัวแปร String

- **PortOpen**

ในการใช้งานและอ่านค่าสถานะของพอร์ตอนุกรม เพื่อเปิดและปิดพอร์ตอนุกรม

รูปแบบการใช้งานคำสั่ง `object.PortOpen [= Value]`

ค่า Value มีชนิดข้อมูลที่เป็นแบบบูลีนคือ True กับ False โดย True หมายถึงเปิดพอร์ตอนุกรม และ False หมายถึงปิดพอร์ตอนุกรม สำหรับการปิดพอร์ตนั้นจะมีการเคลียร์บัฟเฟอร์รับข้อมูลและบัฟเฟอร์ส่งข้อมูลด้วย คอนโทรล MSComm จะปิดพอร์ตอัตโนมัติเมื่อมีการออกจากโปรแกรม ก่อนที่จะใช้คุณสมบัติ PortOpen ต้องตรวจสอบให้แน่ใจก่อนว่าคุณสมบัติ CommPort นั้น ได้ทำการกำหนดตำแหน่งของพอร์ตอนุกรมไว้ถูกต้องหรือไม่ มิเช่นนั้น MSComm จะแสดงข้อผิดพลาด Error 68 แจ้งแก่ผู้ใช้งาน หรือถ้าพอร์ตถูกเปิดเอาไว้แล้ว โปรแกรมก็จะแจ้งข้อความผิดพลาดออกมาเช่นเดียวกัน

- **Input**

อ่านค่าและลบค่าขบวนข้อมูลจากบัฟเฟอร์ภาครับ

รูปแบบการใช้งานคำสั่ง `object.Input`

คุณสมบัติ InputLen เป็นตัวกำหนดจำนวนของตัวอักษรที่จะอ่านโดยคุณสมบัติ Input การกำหนดค่าให้ InputLen เท่ากับ 0 เป็นการกำหนดให้คุณสมบัติ Input ทำการอ่านค่าข้อมูลในบัฟเฟอร์รับข้อมูลทั้งหมด

คุณสมบัติ InputMode เป็นตัวกำหนดชนิดของข้อมูลที่คุณสมบัติ Input รับเข้ามา ถ้า InputMode ถูกกำหนดเป็น comInputModeText คุณสมบัติ Input จะส่งค่าข้อมูลกลับมาในรูปแบบของข้อความชนิดข้อมูลเป็นแบบ Variant ถ้า InputMode กำหนดเป็น comInputModeBinary คุณสมบัติ Input จะส่งข้อมูลกลับมาในรูปแบบของ ไบนารีและชนิดข้อมูลเป็นแบบ Variant

- **InBufferCount**

ส่งค่าจำนวนของตัวอักษรที่อยู่ในบัฟเฟอร์ภาครับ

รูปแบบการใช้งานคำสั่ง `object.InBufferCount [= Value]`

คำสั่ง `InBufferCount` จะแสดงค่าจำนวนของตัวอักษร ซึ่งรับมาจากภายนอกและยังเก็บอยู่ในบัฟเฟอร์ภาครับ เพื่อให้ผู้ใช้งานอ่านค่าออกไป สำหรับการเคลียร์ค่าบัฟเฟอร์ภาครับทำได้โดยกำหนดให้ `InBufferCount` มีค่าเป็น 0

- **InBufferSize**

กำหนดและคืนค่าขนาดของบัฟเฟอร์ภาครับในหน่วยเป็นไบต์

รูปแบบการใช้งานคำสั่ง `object.InBufferSize [= Value]`

คำสั่ง `InBufferSize` ใช้เพื่อกำหนดขนาดของบัฟเฟอร์ภาครับ โดยค่าเริ่มต้นถูกกำหนดไว้ที่ 1,024 ไบต์

หมายเหตุ การกำหนดค่าบัฟเฟอร์ภาครับขนาดใหญ่จะทำให้ หน่วยความจำที่เหลือสำหรับการใช้งานส่วนอื่นๆ จะเหลือน้อย อย่างไรก็ตามการกำหนดค่า บัฟเฟอร์ภาครับที่น้อยเกินไปจะทำให้เกิดการโอเวอร์โฟลวหรือข้อมูลล้นบัฟเฟอร์ เว้นแต่จะมีการใช้แฮนด์เช็ค ดังนั้นค่าปานกลางที่เหมาะสมก็คือค่า 1,024 ไบต์ ซึ่งเป็นค่าเริ่มต้นนั่นเอง แต่ถ้าโปรแกรมมีการเกิด โอเวอร์โฟลวแล้วจึงค่อยปรับเพิ่มค่าขนาดของบัฟเฟอร์ให้มีค่ามากขึ้น

- **InputLen**

กำหนดค่าและคืนค่าจำนวนของตัวอักษรที่อ่านจากบัฟเฟอร์ภาครับ

รูปแบบการใช้งานคำสั่ง `object.InputLen [= Value]`

ค่าเริ่มต้นของคุณสมบัติ `InputLen` มีค่าเท่ากับ “0” การกำหนดค่าเท่ากับ “0” จะทำให้คำสั่ง `Input` ของ `MSComm` อ่านค่าข้อมูลที่อยู่ภายในบัฟเฟอร์ภาครับทั้งหมด

ถ้าไม่มีข้อมูลอยู่ในบัฟเฟอร์ภาครับมากเท่ากับจำนวน `InputLen` คำสั่ง `Input` จะส่งค่าว่าง (“”) กลับออกมา ผู้ใช้งานสามารถตรวจสอบข้อมูลในบัฟเฟอร์ภาครับได้โดยใช้คุณสมบัติ `InBufferCount` โดยการกำหนดให้มีข้อมูลอยู่ในบัฟเฟอร์ภาครับก่อนแล้วจึงค่อยอ่านข้อมูลจากบัฟเฟอร์ภาครับ คุณสมบัตินี้มักใช้กับการอ่านค่าข้อมูลจากเครื่องมือหรือเครื่องจักรที่มีการกำหนดค่าขนาดความยาวของข้อมูลเอาไว้แล้ว

- **InputMode**

กำหนดค่าและคืนค่าชนิดของข้อมูลที่รับ โดยคำสั่ง `Input`

รูปแบบการใช้งานคำสั่ง `object.InputMode [= Value]`

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

คุณสมบัติ `InputMode` ใช้กำหนดว่าข้อมูลชนิดไหนที่รับเข้ามาผ่านคำสั่ง `Input` โดยข้อมูลจะเลือกได้ 2 ประเภทคือ

`comInputModeText` สำหรับข้อมูลที่อยู่ในรูปข้อความตัวอักษรตามมาตรฐาน ANSI โดยจะต้องกำหนดค่าเป็น “0” และค่าเริ่มต้นของการรับค่าข้อมูลก็จะเป็นค่านี้

`comInputModeBinary` สำหรับข้อมูลอื่นๆ ซึ่งจะเก็บในรูปแบบไบนารีรวมกันอยู่เป็น ไบต์ข้อมูล

● Output

ใช้ในการรับส่งขบวนของข้อมูล ไปยังบัฟเฟอร์ส่งข้อมูล

รูปแบบการใช้งานคำสั่ง `object.Output [= Value]`

ค่า `Value` เป็นค่าของตัวอักษรที่เขียนไปยังบัฟเฟอร์ส่งข้อมูล คุณสมบัติ `Output` สามารถใช้ในการส่งข้อมูลตัวอักษรหรือข้อมูลไบนารีก็ได้ โดยการส่งข้อมูลเป็นรูปแบบตัวอักษรจะต้องกำหนดชนิดข้อมูลเป็นแบบ `Variant` และมีข้อมูลภายในเป็นแบบ `String` สำหรับการส่งข้อมูลไบนารีจะต้องกำหนดชนิดข้อมูลเป็นแบบ `Variant` และมีข้อมูลภายในเป็นแบบ `Byte`

● OutBufferCount

คือค่าจำนวนของข้อมูลตัวอักษรที่เก็บอยู่ในบัฟเฟอร์ภาคส่ง และสามารถใช้คำสั่งนี้เพื่อเคลียร์บัฟเฟอร์ภาคส่งได้ด้วย

รูปแบบการใช้งานคำสั่ง `object.OutBufferCount [= Value]`

ผู้ใช้งานสามารถเคลียร์บัฟเฟอร์ภาคส่งได้โดยการกำหนดค่า `OutBufferCount` เท่ากับ “0”

● OutBufferSize

กำหนดคิณค่าขนาดของบัฟเฟอร์ภาคส่ง ชนิดตัวแปรเป็นแบบไบต์

รูปแบบการใช้งานคำสั่ง `object.OutBufferSize [= object]`

คุณสมบัติ `OutBufferSize` ใช้สำหรับกำหนดขนาดของบัฟเฟอร์ภาคส่ง โดยค่าปกติที่ใช้งานจะมีค่าเท่ากับ 512 ไบต์

หมายเหตุ การกำหนดค่าบัฟเฟอร์ภาคส่งที่มากเกินไปจะทำให้ มีหน่วยความจำเหลือให้ใช้งานแต่น้อย แต่อย่างไรก็ตามถ้ากำหนดค่าน้อยเกินไป จะทำให้เกิดข้อมูลสันบัฟเฟอร์ขึ้น ได้ยกเว้นจะมีการใช้ `แฮนด์เช็ก` วิธีการที่ถูกต้องในการกำหนดค่าคือ ทดลองใช้ค่าเริ่มต้นคือค่า 512 ไบต์ดูก่อน ถ้าโปรแกรมทำงานแล้วเกิดการสันของข้อมูลบ่อยเพิ่มค่าของ `OutBufferSize` ให้มากขึ้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- **ParityReplace**

กำหนดและคืนค่าตัวอักษรที่ไปวางแทนในตำแหน่งที่เกิดข้อผิดพลาดจากพาริตี

รูปแบบการใช้งานคำสั่ง `object.ParityReplace [= Value]`

บิตพาริตี เป็นบิตที่ทางภาคส่งข้อมูลทำการส่งมาพร้อมกับข้อมูล เพื่อทำการตรวจสอบข้อผิดพลาดของข้อมูล โดยเมื่อมีการใช้บิตพาริตี คอนโทรล MSComm จะทำการบวกทุกบิตที่มีค่าลอจิก “1” ในแต่ละ ไบต์ และทำการตรวจสอบผลลัพธ์ว่าบิตที่อ่านได้นั้นมีจำนวนลอจิก “1” เป็นเลขคู่หรือคี่ และตรงกับค่าที่กำหนดไว้แต่ต้นหรือไม่ ถ้าค่าที่นำมาบวกแล้วมีพาริตีไม่ตรงแสดงว่าการรับส่งข้อมูลผิดพลาด

การกำหนดค่า เริ่มต้นให้กับ ParityReplace นั้นกำหนดให้ใช้เครื่องหมาย (?) ไปวางไว้ที่ตำแหน่งที่เกิดพาริตีผิดพลาด ถ้ากำหนดค่า ParityReplace ให้เป็นค่าว่าง (“”) จะเป็นการยกเลิกการใช้งาน ParityReplace และ ไม่มีการป้อนข้อมูลแทนเมื่อตรวจพบข้อผิดพลาด

ParityReplace ใช้ชนิดข้อมูลเป็นแบบสตริง แต่จากการกำหนด จะกำหนดได้เพียง ไบต์เดียวเท่านั้น ซึ่งจะสามารถใช้ค่าใดๆก็ได้ที่เป็นโค้ด ANSI มีค่าอยู่ระหว่าง 0-255

- **DTREnable**

ใช้ในการกำหนดสถานะลอจิกของขา Data Terminal Ready (DTR) โดยสัญญาณของขา DTR จะส่งจากคอมพิวเตอร์ไปยังโมเด็มเพื่อแสดงว่าคอมพิวเตอร์พร้อมที่จะรับข้อมูลแล้ว ชนิดของข้อมูลเป็นแบบบูลีน

รูปแบบการใช้งานคำสั่ง `object.DTREnable [= Value]`

ค่า Value เป็นค่าสถานะ True หรือ False เพื่อกำหนดลอจิกของขา DTR ให้เป็น “0” หรือ “1” โดย

True หมายถึง ให้ขา DTR มีลอจิก “1”

False หมายถึง ให้ขา DTR มีลอจิก “0” (เป็นค่าปกติ)

หมายเหตุ เมื่อขา DTR ถูกกำหนดสถานะให้เป็น True ที่ขา DTR จะมีสถานะลอจิก “1” เมื่อทำการเปิดพอร์ตและจะมีสถานะเป็น “0” เมื่อมีการปิดพอร์ต เมื่อขา DTR ถูกกำหนดสถานะเป็น False ที่ขา DTR จะมีสถานะลอจิก “0” ตลอดเวลาไม่ว่าจะใช้คำสั่งเปิดพอร์ตหรือปิดพอร์ต

สำหรับการใช้งานกับโมเด็ม การทำให้ขา DTR เป็นลอจิก “0” จะเป็นการวางหูโทรศัพท์หรือยกเลิกการติดต่อ

- **RTSEnable**

ใช้เพื่อกำหนดสถานะลอจิกให้ขา Request To Send (RTS) โดยขา RTS จะเป็นสัญญาณที่ส่งจากคอมพิวเตอร์ไปยังโมเด็มเพื่อร้องขอส่งข้อมูล ชนิดของข้อมูลเป็นแบบ Boolean

รูปแบบการใช้งานคำสั่ง `object.RTSEnable[= value]`

ค่า Value เป็นค่าสถานะ True หรือ False เพื่อกำหนดลอจิก “0” หรือ “1” ให้ขา RTS โดย

True หมายถึง ให้ขา RTS มีลอจิก “1”

False หมายถึง ให้ขา RTS มีลอจิก “0” (เป็นค่าปกติ)

หมายเหตุ เมื่อขา RTSEnable ถูกกำหนดให้เป็น True ขา RTS จะมีสถานะลอจิก “1” เมื่อเปิดพอร์ตและมีสถานะลอจิก “0” เมื่อปิดพอร์ต

- **EOFEnable**

เป็นการกำหนดให้ MScComm รอสัญลักษณ์แสดงว่าส่วนท้ายสุดของไฟล์ (End of file : EOF) ระหว่างการรับอินพุตเข้ามา ถ้าพบสัญลักษณ์ EOF ภาคอินพุตจะหยุดรับข้อมูล และเหตุการณ์ OnComm จะถูกกระตุ้นให้ทำงาน คุณสมบัติ CommEvent จะมีค่าเท่ากับ 7 หรือ ComEvEOF

รูปแบบการใช้งานคำสั่ง `object.EOFEnable [= Value]`

โดย Value เป็นค่าสถานะ True หรือ False เพื่ออีนามิเบิลหรือดีสเอมิเบิลการทำงานของเหตุการณ์ OnComm เมื่อตรวจพบสัญลักษณ์ EOF โดย

True หมายถึง เหตุการณ์ OnComm จะถูกกระตุ้นให้ทำงานด้วย EOF

False หมายถึงเหตุการณ์ OnComm จะไม่ถูกกระตุ้นให้ทำงานด้วย EOF (เป็นค่าปกติ)

เมื่อ EOFEnable กำหนดให้เป็น False ส่วนควบคุมจะไม่มีกรตรวจสอบสัญลักษณ์ EOF

- **CTSHolding**

ผู้ใช้งานสามารถตรวจสอบการทำงานของขา Clear To Send (CTS) ได้ว่ามีสถานะลอจิก “0” หรือ “1” โดยค่าที่อ่านได้จะเป็นบูลีน True และ False ถ้าค่า CTSHolding เป็น True ขา CTS จะมีสถานะลอจิกเป็น “1” ถ้าค่า CTSHolding เป็น False ขา CTS จะมีสถานะลอจิกเป็น “0”

รูปแบบการใช้งานคำสั่ง `object.CTSHolding`

เมื่อขา CTS เป็นลอจิก “0” (CTSHolding = False) และเกิด ไทม์เอาต์ คอนโทรล MScComm จะกำหนดให้คุณสมบัติ CommEvent มีค่าเป็น comEventCTSTO (Clear To Send Timeout) และกระตุ้นให้เกิดเหตุการณ์ OnComm

- **CDHolding**

ผู้ใช้งานสามารถตรวจสอบการทำงานของ Data Carrier Detect (DCD) ได้ว่ามีสถานะลอจิก เป็น “1” หรือ “0” โดยค่าที่อ่านได้จะเป็นบูลีน True และ False ถ้าค่า CDHolding เป็น True ฆ DCD จะมีสถานะลอจิก “1” ถ้าค่า CDHolding เป็น False ฆ DCD จะมีสถานะลอจิก “0”

รูปแบบการใช้งานคำสั่ง `object.CDHolding`

เมื่อฆ DCD มีลอจิก “1” (CDHolding = True) และเกิดไทม์เอาต์ คอนโทรล MSCComm จะ กำหนด ให้คุณสมบัติ CommEvent มีค่าเป็น comEventCDTO (Carrier Detect Timeout Error)และ กระตุ้นให้เกิดเหตุการณ์ OnComm

- **DSRHolding**

ผู้ใช้งานสามารถตรวจสอบการทำงานของฆ DSR (DSR) ได้ว่ามีสถานะลอจิก “1” หรือ “0” โดยค่าที่อ่านได้จะเป็นบูลีน True และ False ถ้าค่า DSRHolding เป็น True ฆ DSR จะมีสถานะลอจิก “1” ถ้าค่า DSRHolding เป็น False ฆ DSR จะมีสถานะลอจิก “0”

รูปแบบการใช้งานคำสั่ง `object.DSRHolding`

เมื่อฆ DSR มีลอจิก “1” (DSRHolding = True) และเกิดไทม์เอาต์ คอนโทรล MSCComm จะ กำหนด ให้คุณสมบัติ CommEvent มีค่าเป็น comEventDSRTO (Data Set Ready Timeout Error) และกระตุ้นให้เกิดเหตุการณ์ OnComm

- **Handshaking**

กำหนดคุณสมบัติและคิ่ค่ารูปแบบแฮนด์เช้คทางฮาร์ดแวร์

รูปแบบการใช้งานคำสั่ง `object.HandShaking [= Value]`

ค่าตัวแปร Value ที่ใช้กำหนดมี 4 รูปแบบด้วยกันคือ

1. comNone ค่าที่กำหนดคือ 0 เป็นการกำหนดให้ไม่มีการแฮนด์เช้ค (เป็นค่าเริ่มต้น)
2. comXOnXOff ค่าที่กำหนดคือ 1 เป็นการกำหนดให้ใช้แฮนด์เช้คแบบ XON/XOFF
3. comRTS ค่าที่กำหนดคือ 2 เป็นการกำหนดให้ใช้ฆ RTS/CTS (Request To Send/Clear To Send)
4. comRTS ค่าที่กำหนดคือ 3 เป็นการกำหนดให้ใช้ทั้งแบบ Request To Send และ XON/XOFF

คุณสมบัติ Handshaking ใช้เพื่อกำหนดรูปแบบการสื่อสารภายใน ระหว่างข้อมูลถูกส่งไปยังบัพ

เพอร์ภากรับ เมื่อข้อมูลตัวอักษรถูกส่งมาถึงพอร์ตอนุกรม อุปกรณ์สื่อสารข้อมูลจะทำการย้ายข้อมูล เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่นอญูดเนาไปใช้ประโยชน์ด้านกราคา ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ไปยังบัพเฟอร์ภาครับ เพื่อที่จะโปรแกรมสามารถอ่านค่าไปใช้งานได้ ถ้าไม่มีบัพเฟอร์ภาครับ โปรแกรมที่ใช้งานจะต้องทำการอ่านค่าข้อมูลโดยตรงจากฮาร์ดแวร์ของพอร์ตอนุกรม ซึ่งผู้ใช้งาน จะเกิดปัญหาข้อมูลสูญหายได้ เนื่องจากว่าการเปลี่ยนแปลงของข้อมูลที่ส่งเข้ามามีการเปลี่ยนแปลง อย่างรวดเร็ว

คุณสมบัติ Handshaking ช่วยให้ผู้ใช้งานแน่ใจได้ว่าข้อมูลที่ได้รับนั้นไม่มีการสูญหายเมื่อบัพเฟอร์ภาครับที่รับข้อมูลนั้นเกิดข้อมูลล้นหรือโอเวอร์โฟลว์ (Overflow) โดยใช้วิธีการตรวจสอบ ความพร้อมของบัพเฟอร์ว่าพร้อมรับข้อมูลหรือไม่ก่อนที่จะส่งข้อมูลมาให้

- **Break**

ใช้ในการเซตและเคลียร์ค่าสัญญาณ Break ชนิดของข้อมูลเป็นแบบ Boolean

รูปแบบการใช้งาน object.Break [= Value]

โดย Value เป็นค่าบูลีน ถ้า Value = True หมายถึง การส่งสัญญาณ Break ออกไป ถ้า Value = False หมายถึงการเคลียร์สัญญาณ Break

เมื่อกำหนดให้สัญญาณ Break เป็น True จะเป็นการหยุดการส่งข้อมูลชั่วคราวจนกว่าจะมีการ ตั้งให้สัญญาณ Break เป็น False

- **OnComm**

เหตุการณ์ OnComm จะถูกสร้างขึ้นเมื่อค่าของคุณสมบัติ ComEvent มีการเปลี่ยนแปลงเพื่อ แสดงผลการเปลี่ยนแปลงเหล่านั้นแบบทันทีทันใดหรือแสดงข้อผิดพลาดที่เกิดขึ้น ตัวอย่าง โปรแกรมย่อย OnComm เพื่อนำเหตุการณ์ CommEvent มาแสดง

```
Private Sub MSComm_OnComm()
```

```
    Select Case MSComm1.CommEvent
```

```
        'Handle each event or error by placing
```

```
        'Code below each case statement
```

```
        'Errors
```

```
            Case comEventBreak           'A Break was received.
```

```
            Case comEventCDTO           'CD (RLSD) Timeout.
```

```
            Case comEventCTSTO          'CTS Timeout.
```

```
            Case comEventDSRTO          'DSR Timeout.
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

Case comEventFrme          'Framing Error
Case comEventOverrun      'Data Lost.
Case comEventRxOver       'Receive buffer overflow,
Case comEventRxParity     'Parity Error.
Case comEventTxFull
    'Event
    Case comEvCD           'Change in the CD line.
    Case comEvCTS          'Change in the CTS line.
    Case comEvDSR          'Change in the DSR line.
    Case comEvRing         'Change in the Ring Indicator.
    Case comEvReceive      'Receive RThreshold # of Chars.
    Case comEvSend         'Sthreshold number in the transmit buffer.
Case comEvEof              'An EOF character was found in the input stream
End Select
End Sub

```

การใช้ MSComm เพื่อการติดต่อฮาร์ดแวร์

จากรายละเอียดของ MSComm ที่กล่าวไปแล้วตอนต้นนั้น จะเห็นได้ว่าวิธีการที่จะอ่านค่าหรือเขียนค่าไปยังสถานะและควบคุมของพอร์ตอนุกรมสามารถทำได้ง่ายดายนมาก โดยใช้คำสั่งเหล่านี้

DTREnable	สำหรับการสั่งให้ขา DTR มีลอจิก “0” หรือ “1”
RTSEnable	สำหรับการสั่งให้ขา RTS มีลอจิก “0” หรือ “1”
CTSHolding	สำหรับอ่านค่าสถานะจากขา CTS ว่ามีลอจิก “0” หรือ “1”
CDHolding	สำหรับอ่านค่าสถานะจากขา DCD ว่ามีลอจิก “0” หรือ “1”
DSRHolding	สำหรับอ่านค่าสถานะจากขา DSR ว่ามีลอจิก “0” หรือ “1”
Break	สำหรับการสั่งให้ขา TxD มีลอจิก “0” หรือ “1”

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.4 Current Transformer

วิธีการพันหม้อแปลง

เนื่องจากวงจรลิมิตกระแส(Current limit) ของโหลด อาศัยหลักของ Current Transformer ในการวัดกระแส โดยเอาต์พุตที่ออกมาจะเป็นแรงดันที่จะมีลักษณะเป็นเชิงเส้น (Linear Voltage) ซึ่งหลักการพัน Current Transformer จะใช้แกนเทอร์รอยล์ ในการพันหม้อแปลงชนิดนี้โดยมีวิธีการดังนี้

1. หาพื้นที่หน้าตัดของเส้นลวดที่ใช้ในการพันขดไฟเข้า (Primary) และขดไฟออก (Secondary) เพื่อที่จะนำไปเปิดตารางหาเบอร์ของขดลวดที่ใช้ โดยมีหลักการดังนี้

พื้นที่หน้าตัดของขดลวด 400 เซอร์ควิลาร์มิล จะสามารถทนกระแสไฟฟ้าได้ 1 A

หรือ พื้นที่หน้าตัดของขดลวด 0.2 ตารางเมตร จะสามารถทนกระแสไฟฟ้าได้ 1 A

ตัวอย่าง ขดไฟเข้า 220 V 6 A ขดไฟออก 0.1 A

ขดไฟเข้า

พื้นที่หน้าตัดของขดลวดที่ใช้ $6 \times 400 = 2400$ เซอร์ควิลาร์มิล

หรือ ความโตที่ใช้ $6 \times 0.2 = 1.2$ ตร.ม.

เมื่อเปิดตารางเทียบเบอร์ได้ SWG เบอร์ 17

ขดไฟออก

พื้นที่หน้าตัดของขดลวดที่ใช้ $0.1 \times 400 = 40$ เซอร์ควิลาร์มิล

หรือ ความโตที่ใช้ $0.1 \times 0.2 = 0.02$ ตร.ม.

เมื่อเปิดตารางเทียบเบอร์ได้ SWG เบอร์ 17

2. การคำนวณจำนวนรอบในการพัน Current Transformer

ตัวอย่าง ขดไฟเข้า 250 V พันขดจำนวน 5 รอบ มีกระแสไหลที่ขดไฟออก 0.1 A

$$\frac{N_p}{N_s} = \frac{I_s}{I_p}$$

1. ต้องการสเกลสูงสุดที่ 5 A

$$\frac{N_p}{N_s} = \frac{I_s}{I_p}$$

$$\frac{5}{N_s} = \frac{0.1}{5}$$

$$0.1N_s = 25$$

$$N_s = 250$$

ดังนั้นต้องพันที่ขด Secondary จำนวน 250 รอบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2. ต้องการสเกลสูงสุดที่ 2.5 A

$$\frac{5}{N_s} = \frac{0.1}{2.5}$$

$$0.1N_s = 12.5$$

$$N_s = 125$$

ดังนั้นต้องพันที่ขด Secondary จำนวน 125 รอบ

3. ต้องการสเกลสูงสุดที่ 1 A

$$\frac{5}{N_s} = \frac{0.1}{1}$$

$$0.1N_s = 5$$

$$N_s = 50$$

ดังนั้นต้องพันที่ขด Secondary จำนวน 50 รอบ

4. ต้องการสเกลสูงสุดที่ 0.5 A

$$\frac{5}{N_s} = \frac{0.1}{0.5}$$

$$0.1N_s = 2.5$$

$$N_s = 25$$

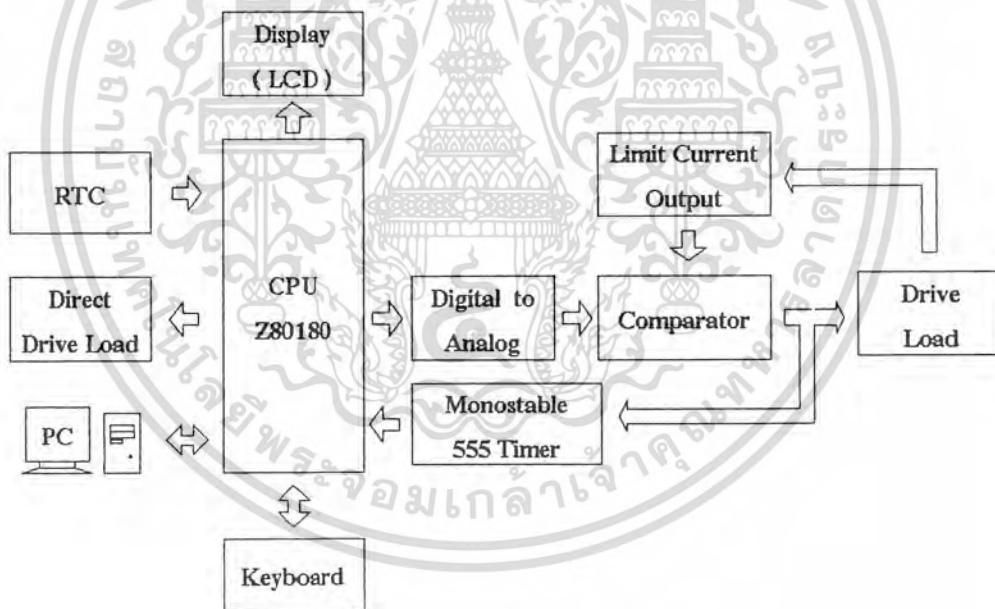
ดังนั้นต้องพันที่ขด Secondary จำนวน 25 รอบ

บทที่ 3

การสร้างและพัฒนาโปรแกรม

การสร้างและออกแบบเครื่องโปรแกรมสวิตซ์หลายช่องควบคุมด้วยไมโครโปรเซสเซอร์ Z80-180 ใช้บอร์ด CP-JR180 V2.0 ประกอบด้วย ส่วนที่สำคัญ 6 ส่วนคือ ภาคประมวลผลรวมของวงจรทั้งหมด ภาคควบคุมคีย์บอร์ด ภาคควบคุมการเปิดและปิดของอุปกรณ์ไฟฟ้า ภาคควบคุมตรวจจับกระแสของอุปกรณ์ไฟฟ้า ภาคแสดงผล และภาครับส่งข้อมูลผ่านพอร์ตอนุกรม ซึ่งจะแสดงโครงสร้างทั้งหมดของเครื่องโปรแกรมสวิตซ์หลายช่องได้ดังนี้

3.1 โครงสร้างของเครื่องโปรแกรมสวิตซ์หลายช่อง



รูปที่ 3-1 แสดงบล็อกไดอะแกรมการทำงานของวงจรทั้งหมด

จากบล็อกไดอะแกรมสามารถอธิบายแต่ละส่วนดังนี้

- ส่วน CPU ใช้ไมโครโปรเซสเซอร์เบอร์ Z80180 ทำหน้าที่ในการประมวลผลข้อมูล เป็นศูนย์กลางในการรับส่งข้อมูลเพื่อติดต่อกับวงจรอินเตอร์เฟสต่างๆ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- ส่วน Display ทำหน้าที่แสดงผลออกทางหน้าจอโดยใช้ LCD (Liquid Crystal Display) เพื่อให้ผู้ใช้สามารถตั้งโปรแกรมควบคุมได้
 - ส่วน RTC ทำหน้าที่เป็นสัญญาณนาฬิกาจะเป็นตัวนับ วัน เดือน ปี สัปดาห์ ชั่วโมง นาที และวินาที
 - ส่วน Keyboard ทำหน้าที่ป้อนข้อมูลต่างๆ ให้แก่ CPU เพื่อให้ CPU ประมวลผลคำสั่งนั้นๆ ต่อไป ซึ่งมีทั้งหมด 24 Key
 - ส่วน Direct Drive Load ทำหน้าที่ขับโหลดทางด้านเอาต์พุทโดยตรง โดยควบคุมจาก CPU
 - ส่วน Digital to Analog ทำหน้าที่แปลงสัญญาณ ดิจิตอล ให้เป็นสัญญาณ อนาล็อก เพื่อจะนำค่าแรงดันที่ได้ไปควบคุมกระแสเอาต์พุทต่อไป
 - ส่วน Comparator ทำหน้าที่เปรียบเทียบแรงดันที่ถูกกำหนดจากคีย์บอร์ด กับแรงดันที่ตรวจสอบจากกระแสที่ไหลผ่านโหลด ในส่วนของ Limit current output
 - ส่วน Drive Load ทำหน้าที่ขับโหลดทางด้านเอาต์พุท โดยจะถูกกำหนดการทำงานจากส่วนของ Comparator
 - ส่วน Limit current output ทำหน้าที่ตรวจสอบกระแสที่ไหลผ่านโหลดว่า ขณะนี้มีกระแสไหลเท่าไรจากนั้นจะทำการแปลงค่าเป็นแรงดันเพื่อไปเปรียบเทียบกับค่าที่ถูกตั้งไว้จากคีย์บอร์ด ในส่วน Comparator
 - ส่วน Analog to Digital for check current ทำหน้าที่แปลงสัญญาณอนาล็อกที่อยู่ในรูปของแรงดัน ให้เป็นสัญญาณดิจิตอล เพื่อใช้ในการอ่านค่ากระแสจากโหลดทางเอาต์พุท
 - ส่วน PC ทำหน้าที่โหลดข้อมูลจากโปรแกรมที่จัดทำขึ้นมาเพื่อความสะดวกในการตั้งเวลาในแต่ละช่อง
- โดยใช้โปรแกรม Visual Basic 5 ในการพัฒนาโปรแกรม ติดต่อพอร์ตอนุกรมแบบมาตรฐาน RS232 และ มาตรฐาน RS485

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.2 การออกแบบและหลักการทำงานของวงจรในภาคต่างๆ

3.2.1 ภาคประมวลผลรวมของวงจรทั้งหมด

ส่วนประกอบของภาคประมวลผลจะแบ่งออกเป็น 2 ส่วนคือ

1. ส่วนฐานเวลาซึ่งเราจะใช้ไอซี RTC (Real Time Clock) เบอร์ 6242 เพื่อเป็นฐานเวลาในการตรวจสอบ วินาที นาที ชั่วโมง วัน เดือน ปี และวันในรอบสัปดาห์ โดยจะทำการตรวจสอบโดยอ่านค่าจากตำแหน่งของหน่วยความจำที่ได้จัดสรรไว้ซึ่งมีตำแหน่งดังนี้

ตำแหน่ง A0H เป็นตำแหน่งประจำค่าของวินาทีหลักหน่วยซึ่งจะนับตั้งแต่ 0-9 โดยจะนับ ค่าเป็นฐาน 16 แบบ BCD Code คือ 00H,01H,02H.....09H แล้วก็จะวนมานับที่ 00H อีกครั้ง โดยเวลาที่ทำการเพิ่มขึ้นนี้จะมีเวลาเท่ากับ 1 วินาทีตามเวลามาตรฐานโลก โดยข้อมูลที่จะเป็นตัวบอกค่านั้นจะอยู่ที่บิตที่ 0(D0) ถึง บิตที่ 3(D3)

ตำแหน่ง A1H เป็นตำแหน่งประจำค่าของวินาทีหลักสิบจะนับตั้งแต่ 0-5 คือ 00H-05H

ตำแหน่ง A2H เป็นตำแหน่งประจำค่าของนาฬิกาหลักหน่วยจะนับตั้งแต่ 0-9 คือ 00H-09H

ตำแหน่ง A3H เป็นตำแหน่งประจำค่าของนาฬิกาหลักสิบจะนับตั้งแต่ 0-5 คือ 00H-05H

ตำแหน่ง A4H เป็นตำแหน่งประจำค่าของชั่วโมงหลักหน่วยจะนับตั้งแต่ 0-9 คือ 00H-09H

ตำแหน่ง A5H เป็นตำแหน่งประจำค่าของชั่วโมงหลักสิบ

ซึ่งค่าของชั่วโมงนี้จะแบ่งเป็น 2 ส่วนคือ

1. ส่วนนี้เป็นตัวบอกว่าเป็น Am หรือว่า Pm ซึ่งถ้าเป็น Am ค่า D2 จะเป็น 0 และถ้าเป็น Pm ค่า D2 จะเป็น 1 ซึ่งส่วนนี้จะนับอยู่ 2 สถานะคือ 0 และ 1
2. ส่วนนี้จะทำการนับค่าธรรมดาโดยจะนับตั้งแต่ 0-3 คือ 00H-03H

ตำแหน่ง A6H เป็นตำแหน่งประจำค่าของวันหลักหน่วยจะนับตั้งแต่ 0-9 คือ 00H-09H

ตำแหน่ง A7H เป็นตำแหน่งประจำค่าของวันหลักสิบจะนับตั้งแต่ 0-3 คือ 00H-03H

ตำแหน่ง A8H เป็นตำแหน่งประจำค่าของเดือนหลักหน่วยจะนับตั้งแต่ 0-9 คือ 00H-09H

ตำแหน่ง A9H เป็นตำแหน่งประจำค่าของเดือนหลักสิบจะนับตั้งแต่ 0-1 คือ 00H-01H

ตำแหน่ง AAH เป็นตำแหน่งประจำค่าของปีหลักหน่วยจะนับตั้งแต่ 0-9 คือ 00H-09H

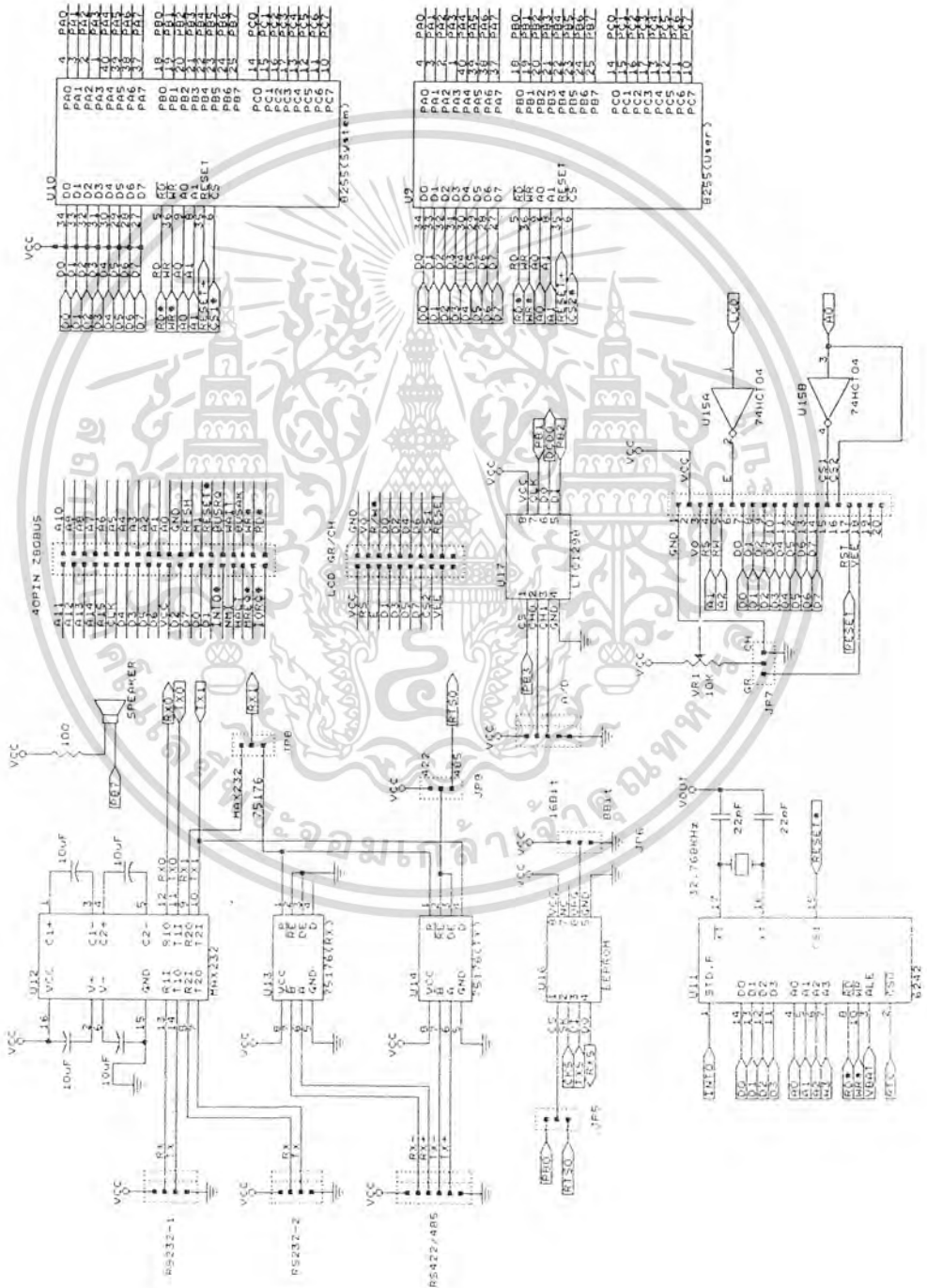
ตำแหน่ง ABH เป็นตำแหน่งประจำค่าของปีหลักสิบจะนับตั้งแต่ 0-9 คือ 00H-09H

ตำแหน่ง ACH เป็นตำแหน่งประจำค่าของวันในรอบสัปดาห์ จะนับตั้งแต่ 0-6 คือ 00H-06H ซึ่งคี่ของ 00H จะเป็นวันอาทิตย์ 01H จะเป็นวันจันทร์ ไปจนถึง 06H จะเป็นวันเสาร์

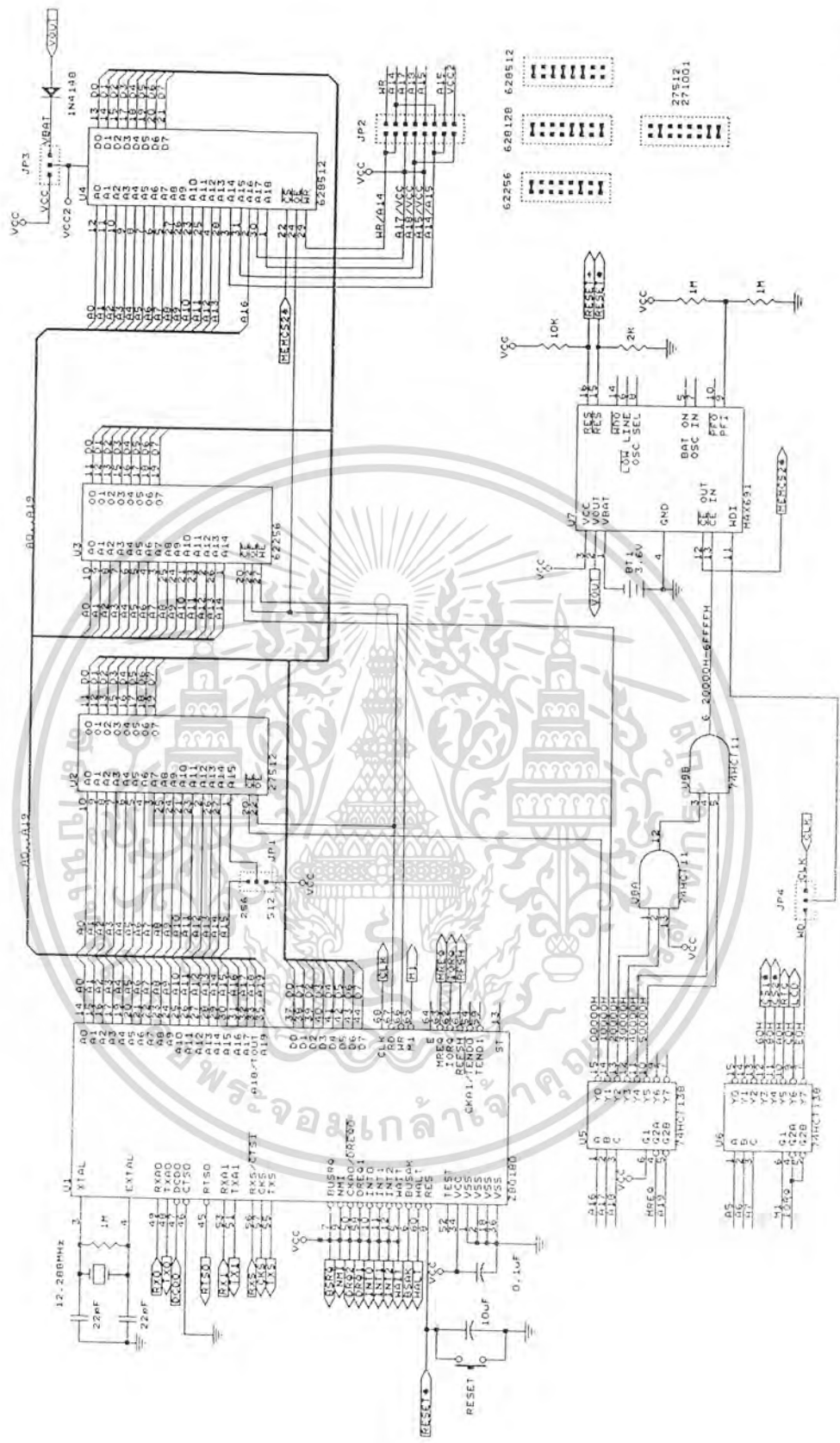
ตำแหน่ง AFH เป็นตำแหน่งของการกำหนดการนับชั่วโมงว่าจะนับ 24 ชั่วโมงต่อ 1 วันหรือว่าจะนับ 12 ชั่วโมงต่อ 1 วัน โดยถ้าบิตที่ 2 (D2) เป็น 0 จะนับ 12 ชั่วโมง แต่ถ้าเป็น 1 จะนับ 24 ชั่วโมง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2. ส่วนประมวลผลของโปรแกรมใช้ไมโครโปรเซสเซอร์เบอร์ Z80-180 ประมวลคำสั่งต่างๆ ที่ได้เขียนโปรแกรมควบคุมการทำงานของอุปกรณ์ต่างๆ ที่ได้ออกแบบไว้โดยบอร์ดควบคุม จะแสดงวงจรดังรูปที่ 3-2 ก และ ข



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



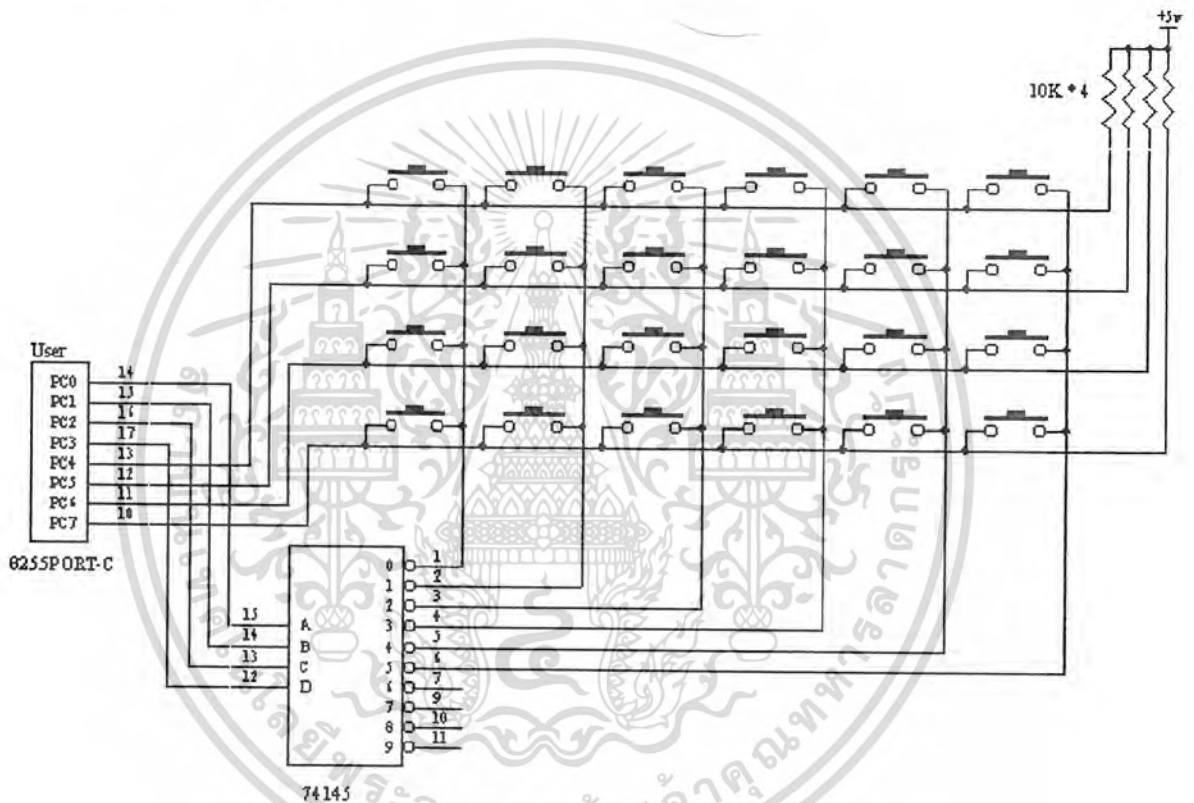
ข.

รูปที่ 3-2 แสดงวงจรของบอร์ดใช้งาน CP-JR 180 V2.0

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.2.2 ภาพวงจรควบคุมคีย์บอร์ด (และรหัสของคีย์)

ส่วนของวงจรคีย์บอร์ดนั้น การต่อวงจรจะต่อแบบเมทริก (Matrix) ขนาด 6×4 มีทั้งหมด 24 คีย์ และในการสร้างรหัสของคีย์นั้น จะใช้หลักการของการสแกนคีย์ โดยต่อคีย์บอร์ดทำงานร่วมกับ ไอซี ทีทีแอล เบอร์ 74LS145 ซึ่งเป็น ไอซีดีโค้ดเดอร์ (Decoder) เพื่อควบคุมการสร้างรหัสของคีย์ด้วยโปรแกรม ซึ่งในการต่อคีย์บอร์ดนั้นจะต่อใช้งานที่ PORT C ของ 8255 (User) ที่ตำแหน่ง 82H ดังรูป 3-3

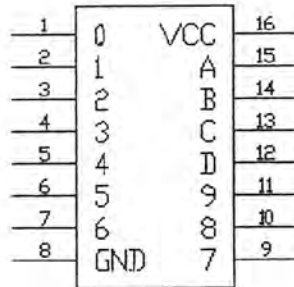


รูปที่ 3-3 แสดงวงจรควบคุมคีย์บอร์ด

หลักการการทำงานของวงจรถูก

จากวงจรจะเห็นว่าที่พอร์ต P3.0-3.3 จะมีตัวต้านทานค่า 10 กิโลโอห์มต่อในลักษณะพุช พู (Push Pull) เพื่อให้สถานะของพอร์ตมีเสถียรภาพขึ้น ซึ่งจะต่ออยู่ในแต่ละแถว ในขณะที่ยังไม่มี การป้อนแรงดันออกที่พอร์ต P3.4-3.7 จะมีค่าที่พอร์ต C เท่ากับ FOH โดยตารางการทำงานของไอซี ทีทีแอล 74LS145 แสดงดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



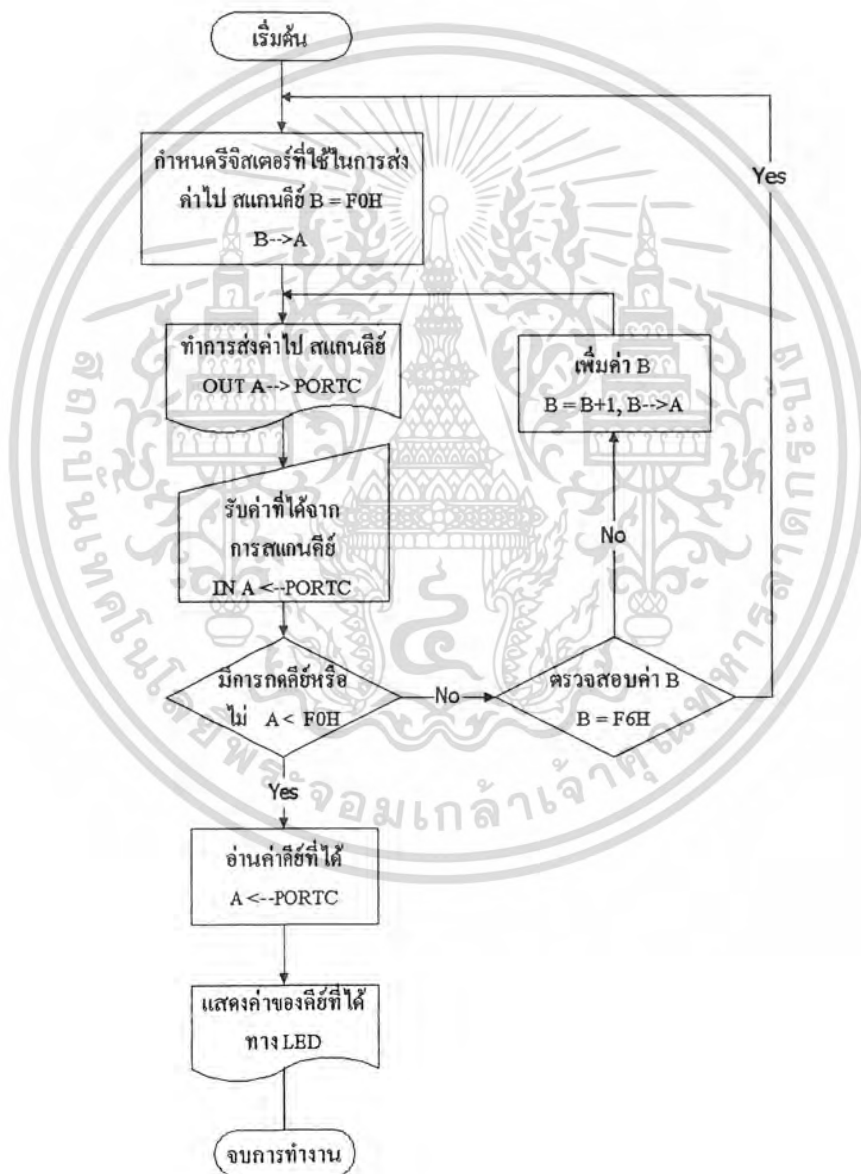
74LS145

NO	INPUTS				OUTPUTS									
	D	C	B	A	0	1	2	3	4	5	6	7	8	9
0	L	L	L	L	L	H	H	H	H	H	H	H	H	H
1	L	L	L	H	H	L	H	H	H	H	H	H	H	H
2	L	L	H	L	H	H	L	H	H	H	H	H	H	H
3	L	L	H	H	H	H	H	L	H	H	H	H	H	H
4	L	H	L	L	H	H	H	H	L	H	H	H	H	H
5	L	H	L	H	H	H	H	H	H	L	H	H	H	H
6	L	H	H	L	H	H	H	H	H	H	L	H	H	H
7	L	H	H	H	H	H	H	H	H	H	H	L	H	H
8	H	L	L	L	H	H	H	H	H	H	H	H	L	H
9	H	L	L	H	H	H	H	H	H	H	H	H	H	L

ตารางที่ 3-1 ตารางการทำงานของไอซี ทีทีแอล 74LS145

จากตารางการทำงานจะเห็นได้ว่าเป็นวงจร BCD-to-Decimal Decoder ซึ่งจากรูปวงจรที่ 3-3 บิตที่ 4-7 ของพอร์ต C ของ 8255 จะไปเข้าที่ขา 12-15 ของ 74LS145 เพื่อทำการป้อนรหัส BCD แล้วก็จะดีโค้ด ออกมาที่ตำแหน่งเอาต์พุตตามต้องการ โดยเอาต์พุตที่ออกมาจะทำงานที่ลอจิก 0 ที่เหลือจะเป็น 1 หมด ซึ่งลอจิก 0 นี้เองที่เรานำมาทำการตรวจสอบการกดของคีย์และสร้างรหัสคีย์ โดยพอร์ตที่ใช้ควบคุมคีย์บอร์ดนี้จะสามารถแยกได้เป็น 4 บิตบนและ 4 บิตล่าง โดยกำหนดให้ 4 บิตล่างทำหน้าที่เป็นอินพุต(input) และ 4 บิตบนทำหน้าที่เป็นเอาต์พุต(output) และที่ 4 บิตบนนี้ถ้ายังไม่มีการกดคีย์จะมีค่าเป็น FXH และถ้าหากมีการกดคีย์ค่าของ FXH นั้นจะเปลี่ยนไปตามแถวของเอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

คีย์ที่กด และในตัว X ที่เห็นนั้นจะถูกกำหนดด้วยการส่งเอาที่พูทออกจากพอร์ตเพื่อไปควบคุมการทำงานของ 74LS145 เพื่อทำการเลือกที่จะตรวจสอบที่แถวไหนของคีย์บอร์ด โดยค่า X ก็คือค่าของ 0-6 ในฐานสิบ หรือเท่ากับ 0000-0110 ในฐาน 2 นั้นเอง และเมื่อนำมารวมกันกับ 4 บิตบนจะได้รหัสของคีย์บอร์ดแต่ละคีย์ออกมา เช่น E3H (เกิดจากการกดคีย์ที่แถวที่ 2 และคอลัมน์ที่ 2 เป็นต้น ซึ่งในส่วนของการสร้างรหัสคีย์และควบคุมการทำงานของไอซี 74LS145 จะควบคุมด้วยโปรแกรมแอสเซมบลีซึ่งโปรแกรมที่ใช้สามารถเขียนเป็นโฟลว์ชาร์ต(Flow Chart)ดังนี้



รูปที่ 3-4 โฟลว์ชาร์ตโปรแกรมย่อยของการตรวจสอบและการสร้างรหัสคีย์บอร์ด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

และเมื่อเราทำการสร้างรหัสคีย์ จะได้รหัสคีย์ดังนี้

D5H	D3H	D2H	D1H	D0H	D4H
E5H	E3H	E2H	E1H	E0H	E4H
75H	73H	72H	71H	70H	74H
B5H	B3H	B2H	B1H	B0H	B4H

รูปที่ 3-5 แสดงรหัสประจำคีย์บอร์ด

เมื่อได้ทำการตรวจสอบรหัสของคีย์บอร์ดแล้ว จากนั้นก็ทำการกำหนดตำแหน่งการใช้งานของคีย์ต่างๆ ดังนี้

1	2	3	Pgm	Mual	Esc
4	5	6	Pc	Amp	Chk
7	8	9	Inc	Dec	Time
.	0	Clr	<--	-->	Ent

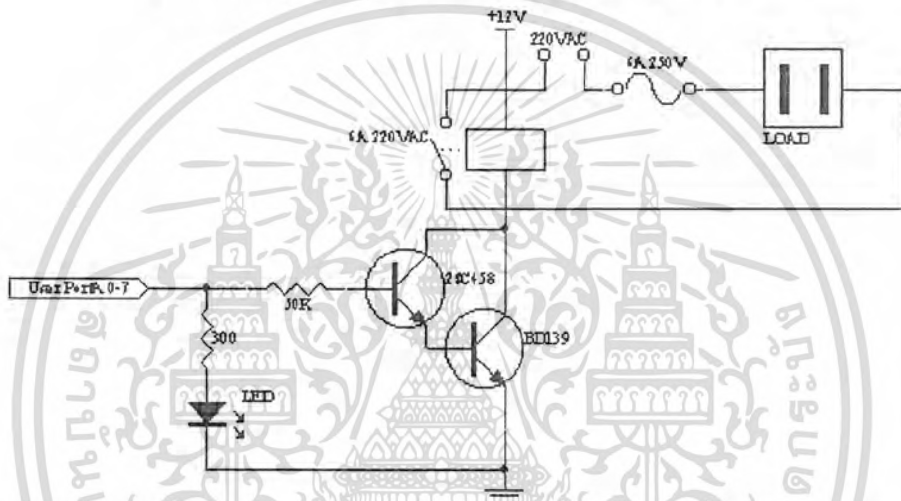
รูปที่ 3-6 แสดงตำแหน่งการใช้งานของคีย์บอร์ด

3.2.3 ภาคควบคุมการทำงานโหลด (Drive Load)

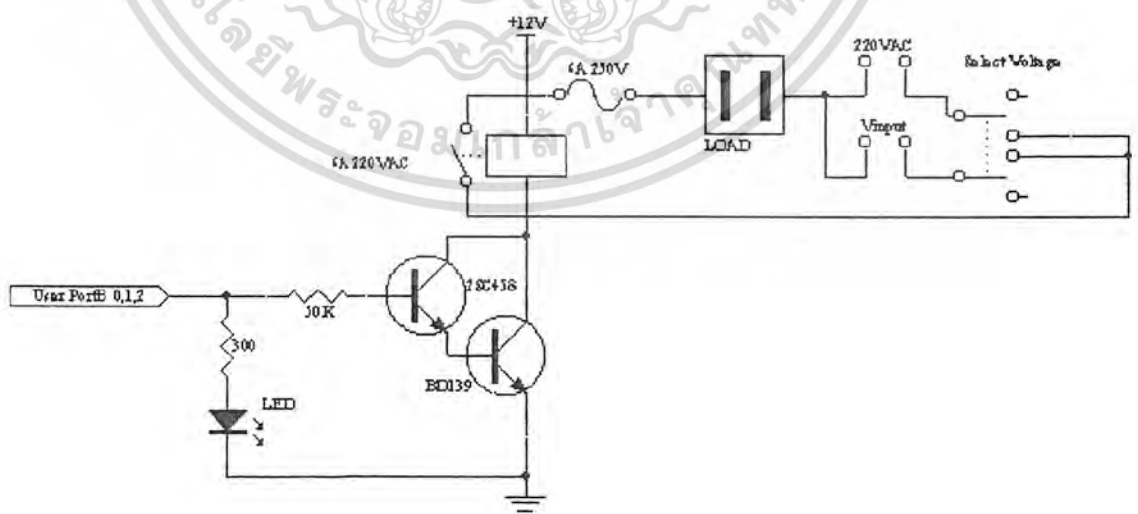
ในภาคนี้จะทำหน้าที่ควบคุมการทำงานของโหลด โดยจะทำการเปิดและปิดการทำงานของเอาต์พุต ซึ่งอุปกรณ์ที่ใช้คือ รีเลย์ (Relay) ซึ่งรีเลย์ที่ใช้จะมีขนาด 6แอมป์ 220โวลท์ ซึ่งวงจรที่ใช้ในการควบคุมการทำงานของรีเลย์จะมีการใช้งานอยู่ 2 แบบคือ

3.2.3.1 การต่อใช้งานโดยใช้กับไฟฟ้ากระแสสลับ 220โวลท์ แสดงดังรูปที่ 3-7

3.2.3.2 การต่อใช้งานโดยผู้ใช้สามารถป้อนแรงดันที่ต้องการให้กับอุปกรณ์ที่จะควบคุมได้ โดยการปรับสวิทช์ควบคุม แสดงการต่อวงจรดังรูปที่ 3-8



รูปที่ 3-7 แสดงวงจรควบคุมการทำงานของรีเลย์โดยใช้กับโหลดไฟฟ้ากระแสสลับ 220โวลท์



รูปที่ 3-8 แสดงวงจรควบคุมการทำงานของรีเลย์

โดยผู้ใช้สามารถป้อนแรงดันตามต้องการได้โดยการปรับสวิทช์

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์การเขียนเพื่อการศึกษาเท่านั้น เมื่อเข้าสู่ที่เห็นใบขอประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดลอกเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หลักการการทำงานของวงจร

จากรูปที่ 3-7 และ 3-8 จะมีหลักการการทำงานที่เหมือนกันจะต่างกันตรงที่การนำไปใช้งาน ซึ่งมีหลักการการทำงานคือ เมื่อมีแรงดันไฟมีค่าประมาณ (+5V) เข้าทางอินพุตที่ขาเบสของทรานซิสเตอร์ 2SC458 โดยผ่านตัวต้านทานค่า 50 กิโลโอห์ม ทำหน้าที่จำกัดกระแสเพื่อไม่ให้มีกระแสไฟฟ้าไหลเข้าขาเบสของทรานซิสเตอร์มากเกินไป ซึ่งจากการคำนวณ ค่ากระแสที่ไหลเข้าขาเบสมีค่าประมาณ 72 ไมโครแอมป์ โดยถ้าไม่มีตัวต้านทานมาช่วยในการจำกัดกระแสที่ไหลเข้าขาเบสแล้วจะทำให้ตัวทรานซิสเตอร์เสียหายได้ จากนั้นเมื่อทรานซิสเตอร์ 2SC458 ทำงาน กระแสทำให้มีกระแสไหลออกที่ขาคอนิเตอร์ ไปเข้าที่ขาเบสของทรานซิสเตอร์อีกตัวหนึ่ง (BD139) ซึ่งต่อแบบ ดาริงตันทรานซิสเตอร์จากนั้นเมื่อวงจรชุดนี้ทำงานจะทำให้มีกระแสไหลผ่านขดลวดของรีเลย์ เกิดการเหนี่ยวนำอีกขดหนึ่งทำให้หน้าสัมผัสของรีเลย์ต่อไฟ 220 โวลต์ครบวงจร จึงทำให้โหลดทำงานได้

และจะเห็นว่าที่อินพุตของวงจรจะมี ตัวต้านทานและ LED ต่ออยู่เพื่อแสดงผลการทำงานของวงจรรีเลย์ คือดูว่าเมื่อมีอินพุตเข้ามาทรานซิสเตอร์และรีเลย์จะทำงานได้ถูกต้องหรือไม่นั่นเอง

จากรูปที่ 3-7 จะกำหนดให้ใช้กับโหลดที่ต้องการแรงดันไฟฟ้าสลับ 220 โวลท์และกระแสสูงสุด 5 แอมป์ โดยจะมีช่อง (Channel) ใช้งานทั้งหมด 9 ช่อง คือช่องที่ 1 ถึงช่องที่ 9

จากรูปที่ 3-8 จะกำหนดให้ผู้ใช้สามารถที่จะใช้แรงดันเท่าไรก็ได้ตามต้องการ หรือจะใช้กับไฟสลับ 220 โวลท์กระแส 5 แอมป์ก็ได้ ทั้งนี้ก็เลือกโดยการปรับสวิทช์ โดยจะมีช่องที่สามารถใช้งานได้ 3 ช่องคือ ช่องที่ 10 ถึงช่องที่ 12

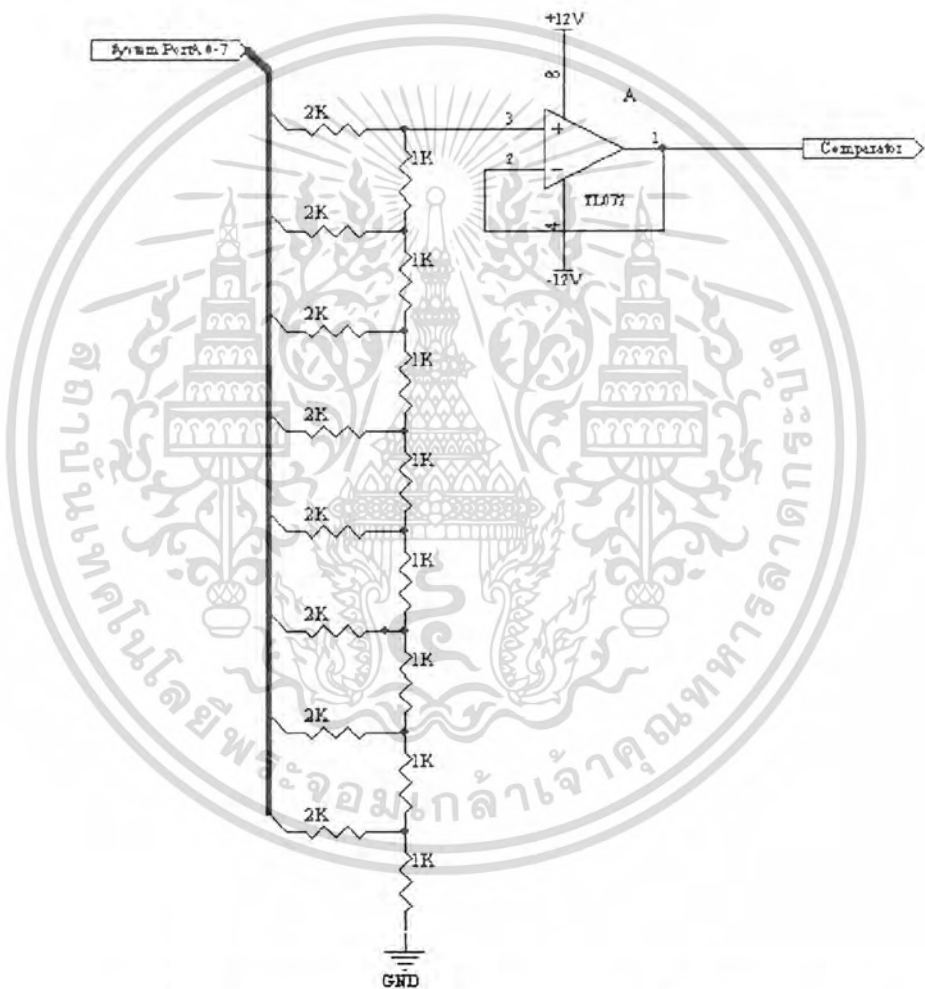
3.2.4 ภาคการควบคุมกระแสของโหลด (Limited current)

ในส่วนของวงจรควบคุมกระแสของโหลดนั้น จะประกอบด้วยวงจร 3 วงจรดังนี้

3.2.4.1 วงจรแปลงสัญญาณดิจิทัลเป็นสัญญาณอนาล็อก

(Digital to Analog Circuit)

วงจรนี้มีส่วนสำคัญในการแปลงสัญญาณดิจิทัลให้เป็นสัญญาณอนาล็อก เพื่อทำการควบคุมกระแสของโหลด ซึ่งค่าของกระแสที่จะควบคุมจะอยู่ในรูปของแรงดัน โดยค่าแรงดันที่ออกจาก



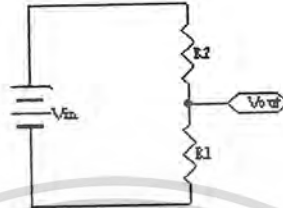
รูปที่ 3-9 แสดงวงจรการแปลงสัญญาณดิจิทัลเป็นสัญญาณอนาล็อก

ออปแอมป์ TL072 นั้น จะมีค่าแรงดันตามอินพุตที่ป้อนเข้ามา โดยแรงดันจะมีค่าเป็นลิเนียร์ คือจะเพิ่มขึ้นตามอินพุตที่ป้อนเข้ามา โดยค่าของอินพุตมากขึ้นเท่าไรก็จะทำให้ค่าของแรงดันมากขึ้นตามไปด้วย ซึ่งออปแอมป์จะทำหน้าที่เป็นบัฟเฟอร์ต่อแบบอนอินเวอร์คคิ่ง และทางเอาต์-

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

พหุจะมีตัวต้านทานปรับค่าได้เพื่อทำการปรับแต่ค่าของแรงดันให้ออกมาตามที่ต้องการ โดยการปรับแต่งจะให้หลักการของวงจรแบ่งแรงดัน ซึ่งมีสูตรการคำนวณดังนี้

$$V_{out} = V_{in} (R_1 / (R_1 + R_2))$$



รูปที่ 3-10 แสดงวงจรแบ่งแรงดัน

หลักการทํางานของวงจร

ลักษณะการต่อวงจรจะต่อแบบ R2R ดังรูปที่ 3-9 การคำนวณค่าของแรงดันคำนวณได้จากสมการดังนี้

$$V_o = V_{in} * ((D0/2) + (D1/4) + (D2/8) + (D3/16) + (D4/32) + (D5/64) + (D6/128) + (D7/256))$$

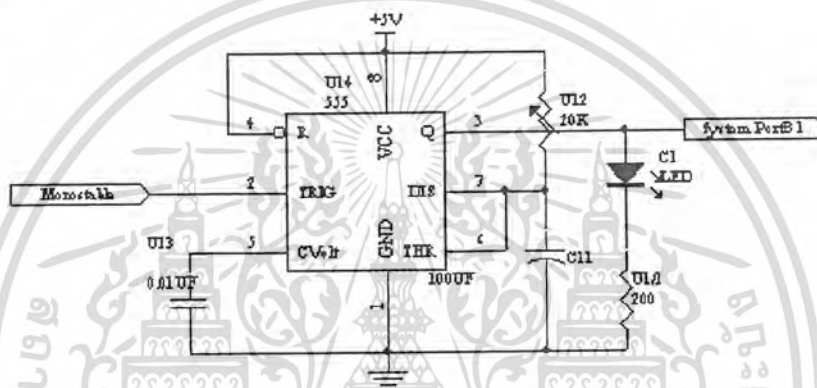
ซึ่งค่าแรงดันที่ได้มีจะนำไปเปรียบเทียบกับค่าของแรงดันที่ได้จากภาคตรวจจับกระแส โดยค่าแรงดันที่จะใช้ในการควบคุมกระแส นั้นจะไดจากการทดลอง วัดกระแสของโหลดโดยใช้แอมป์มิเตอร์วัดจากนั้นก็ดูว่าแรงดันที่ได้จากวงจรตรวจจับกระแส แล้วก็นับทีผลไว้จากนั้นก็นำค่าแรงดันเหล่านั้นมาเปรียบเทียบกับค่าแรงดันที่ได้ในวงจรแปลงสัญญาณดิจิตอลเป็นสัญญาณอนาล็อกว่าตรงกับการป้อนค่าของอินพุททำอะไร เช่น แรงดันที่ตรวจจับได้มีค่าเท่ากับ 2 โวลท์ที่กระแสโหลด 2 แอมป์ ตรงกับเลขฐาน 16 คือ 60H เป็นต้น จากนั้นเพื่อให้เกิดความแน่นอนก็อาจจะตรวจสอบโดยการป้อนค่า 60H ให้แก่วงจรอีกครั้งแล้วดูแรงดันทางเอาท์พุทว่ามีค่าออกมาตรงตามต้องการหรือไม่ถ้าไม่ถูกต้องก็จะต้องทำการตรวจสอบใหม่อีกครั้ง จนได้ค่าที่แน่นอนและใกล้เคียงค่าที่แท้จริงมากที่สุด

3.2.4.2 วงจรหน่วงเวลา (Monostable Multivibrator)

วงจรหน่วงเวลานี้ถูกนำมาใช้ในการตรวจสอบการตัดต่อของรีเลย์ โดยอินพุทที่ป้อนให้แก่วงจรจะมาจากเอาท์พุทของวงจรเปรียบเทียบแรงดัน โดยวงจรโมโนสเตเบิลมัลติไวเบรเตอร์นี้จะทำงานก็ต่อเมื่ออินพุทที่ป้อนเข้ามามีสถานะเป็น $-V$ ซึ่งจะทำให้เอาท์พุทของวงจรโมโนสเตเบิลมีแรงดันออกมาประมาณ $+5$ โวลท์ จากนั้นก็จะส่งแรงดันนี้ไปให้แก่พอร์ต B ของ 8255 (System) เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เพื่อให้ภาคประมวลผลรู้ว่าขณะนี้รีเลย์ทำการตัดแล้ว เมื่อรีเลย์ทำการตัดก็แสดงว่าเกิดกระแสเกินขึ้นที่โหลด ภาคประมวลผลจะสั่งรีเลย์หยุดทำงานถาวรทันทีจนกว่าจะมีการรีเซ็ตค่าใหม่อีกครั้งหนึ่ง ซึ่งการที่ต้องมีวงจรโมโนสเตเบิลมัลติไวเบรเตอร์ก็เพื่อป้องกันการตัดต่อของรีเลย์ เนื่องจากเกิดการเปรียบเทียบค่าแรงดันที่กำหนดกับค่าที่ตรวจจับได้เป็นแรงดันสลับ จึงทำมีค่าแรงดันขึ้นลง จึงเกิดการตัดต่ออย่างรวดเร็วขึ้นนั่นเอง ซึ่งจะทำให้รีเลย์เสียหายได้

การออกแบบวงจรโมโนสเตเบิลมัลติไวเบรเตอร์โดยใช้ไอซีไทม์เมอร์ (Timer) 555 แสดงดังรูปที่ 3-11



รูปที่ 3-11 แสดงวงจรโมโนสเตเบิลมัลติไวเบรเตอร์

ซึ่งในการคำนวณเวลาที่หน่วงนั้น คำนวณได้จากสมการ

$$T = 0.637RC$$

โดยจากวงจรกำหนดให้

$$C = 100\text{uf} , T = 1 \text{ วินาที}$$

ดังนั้น $R = 1 / (0.693 * 100\text{uf})$

$$R = 14.43\text{k}\Omega \text{ เลือกใช้ ตัวต้านทานปรับค่าได้ค่า } 20 \text{ K}\Omega$$

3.2.4.3 วงจรเปรียบเทียบแรงดัน (Comparator Circuit) และวงจรตรวจจับกระแสไฟฟ้า

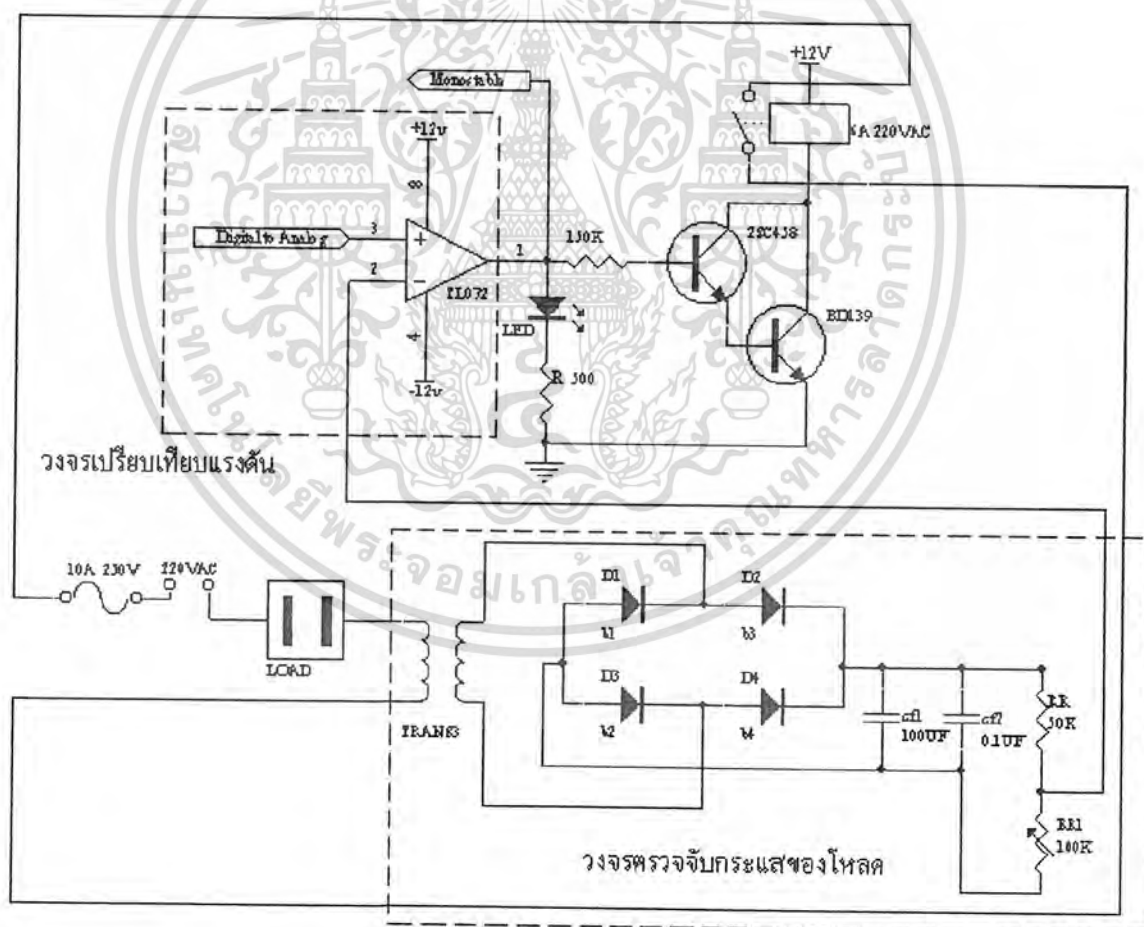
ในส่วนของวงจรเปรียบเทียบแรงดัน (Comparator circuit) ถูกนำมาใช้ในการเปรียบเทียบค่าของแรงดันระหว่างค่าแรงดันที่ผู้ใช้กำหนดกับค่าแรงดันที่ตรวจจับได้จากโหลดว่ามีค่าเป็นอย่างไร หากค่าแรงดันที่ตรวจจับได้จากโหลดมีค่ามากกว่าแรงดันที่ผู้ใช้กำหนด แสดงว่าที่โหลดนั้นมีค่าของกระแสมากกว่าที่กำหนดดังนั้นเราก็ทำการตัดวงจรทันที แต่ถ้าหากว่าค่าแรงดันที่

เอกสารนี้ตรวจจับได้จากโหลดมีค่าน้อยกว่าที่ผู้ใช้กำหนด แสดงว่าโหลดทำงานปกติตามที่ผู้ใช้ได้กำหนดไว้ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โดยจากหลักการที่ว่านี่เราจะนำมาออกแบบวงจร ซึ่งอุปกรณ์ที่เรานำมาใช้คือออปแอมป์ (Op Amp) ซึ่งเราจะใช้คุณสมบัติของออปแอมป์มาใช้ในการตรวจสอบแรงดันคั้งที่ได้กล่าวมาข้างต้น

และในส่วนของวงจรตรวจจ็กระแสนั้นเราจะใช้หม้อแปลงแกนทอรอยด์มาใช้ใน โดยใช้หลักการลดแรงดันลง โดยแรงดันที่ได้ทางขดทุติยภูมิ (Secondary) จะเป็นสัดส่วนกับกระแสที่ไหลผ่านทางด้านขดปฐมภูมิ (Primary) ซึ่งลักษณะการตรวจจ็บจะให้สายไฟฟ้าที่ต่ออยู่กับโหลดมาพันรอบแกนทอรอยด์ซึ่งก็คือขดปฐมภูมิ และเมื่อมีกระแสไหลภายในสายก็จะทำให้เกิดการเหนี่ยวนำรอบแกน ทำให้มีแรงดันเกิดขึ้นทางขดทุติยภูมิโดยแรงดันที่ได้นี้จะนำไปเปรียบเทียบกับแรงดันที่ผู้ใช้กำหนดเพื่อใช้ในการควบคุมกระแสที่โหลด

จากนั้นเราก็จะทำการออกแบบวงจร โดยใช้หลักการที่ได้กล่าวข้างต้นนำมาออกแบบวงจร ซึ่งแสดงวงจรดังรูปที่ 3-12



รูปที่ 3-12 แสดงวงจรตรวจจ็บกระแสของ โหลด และวงจรเปรียบเทียบแรงดัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หลักการการทำงานของวงจร

จากรูปที่ 3-12 ในส่วนของวงจรแบ่งแรงดัน ถ้าสมมุติว่าผู้ใช้ป้อนเลขฐาน 16 ขนาด 8 บิต ออกมาค่าหนึ่ง เมื่อผ่านวงจรแปลงสัญญาณดิจิตอลเป็นสัญญาณอนาล็อกแล้ว ได้แรงดันออกมา 3 โวลต์ ในขณะที่ส่วนของวงจรตรวจจับกระแสไม่มีแรงดันเนื่องรีเลย์ไม่ทำงาน จึงหวนนี้เองเมื่อออปแอมป์ทำการเปรียบเทียบแรงดันระหว่างขาอินเวอร์ตติ้ง (Inverting) และขาอนอินเวอร์ตติ้ง (Non-Inverting) ปรากฏว่าขาอนอินเวอร์ตมีแรงดันมากกว่าขาอินเวอร์ตจึงทำให้ที่เอาต์พุทของออปแอมป์มีแรงดันออกมาเท่ากับแรงดันช่วงบวกของแหล่งจ่าย ทำให้มีแรงดันไปกระตุ้นที่ขาเบสของทรานซิสเตอร์ในภาคขับรีเลย์

แล้วเมื่อมีแรงดันมาที่ขาเบสของทรานซิสเตอร์ 2SC458 ทำให้ทรานซิสเตอร์ BD139 นำกระแส ทำให้กระแสไหลผ่านขาคอลเลกเตอร์ ทำให้มีกระแสไหลผ่านรีเลย์ไปเหนี่ยวนำขดลวดของหม้อแปลง หม้อแปลงจึงทำการต่อวงจรโพลติงทำงาน และเมื่อโพลติงทำงานก็จะมีกระแสไหลผ่านหม้อแปลงที่ใช้ตรวจสอบกระแส เกิดการเหนี่ยวนำขึ้นทำให้มีแรงดันทางด้านทุติยภูมิซึ่งแรงดันนี้จะแปรผันตรงกับกระแสที่ไหลผ่านขดลวดทางปฐมภูมิ จากนั้นก็ผ่านวงจรบริดจ์เรกติฟายเออร์ (Bridge Rectifier) เพื่อทำเป็นแรงดันไฟตรง ซึ่งแรงดันที่จะใช้ในการเปรียบเทียบนั้นสามารถปรับได้ด้วยตัวต้านทานปรับค่าได้ VRL แล้วจะส่งค่าแรงดันนี้ไปเปรียบเทียบกับค่าของแรงดันที่กำหนดมา โดยถ้าค่าของแรงดันที่ตรวจสอบได้นั้น มีค่ามากกว่าแรงดันที่กำหนดจากผู้ใช้ จะทำให้ขาอินเวอร์ตติ้ง ของออปแอมป์มีค่ามากกว่าขาอนอินเวอร์ต ของออปแอมป์ ทำให้ทางเอาต์พุทของออปแอมป์มีแรงดันประมาณ 0 โวลต์ ไม่มีแรงดันไปเข้าขาเบสของ ทรานซิสเตอร์ 2SC458 ทำให้วงจรขับรีเลย์ไม่ทำงานจึงไม่ทำให้รีเลย์ทำงานโพลติงถูกปลดออกนั่นเอง และเมื่อโพลติงถูกตัดออกทำให้ไม่มีแรงดันที่วงจรตรวจจับและไปเปรียบเทียบกับแรงดันที่ผู้ใช้กำหนด ก็จะทำให้แรงดันที่ผู้ใช้กำหนดมากกว่าแรงดันที่ตรวจจับกระแสทำให้เอาต์พุทของออปแอมป์มีแรงดันไฟบวกไปทริกทรานซิสเตอร์รีเลย์ทำงานอีกครั้งหนึ่ง ซึ่งจะทำให้รีเลย์เกิดการตัดต่อตลอดเวลาอาจเสียหายได้ด้วยเหตุนี้เองจึง ได้ออกแบบวงจร โมโนสเตเบิลเพื่อมาทำการตรวจสอบการตัดต่อของออปแอมป์เพื่อที่จะส่งข้อมูลไปบอกกับหน่วยประมวลผลว่าขณะนี้โพลติงมีกระแสเกินกว่าที่กำหนดแล้วให้ทำการตัดวงจรออกจากโพลติงโดยการป้อนรหัส 00H ออกทางพอร์ตควบคุมทันทีซึ่งรหัส 00H นั้นเมื่อผ่านเข้าวงจรแปลงสัญญาณดิจิตอลเป็นอนาล็อกก็จะมีค่าเท่ากับ 0 โวลต์ ทำให้ไม่มีแรงดันไปเปรียบเทียบจึงไม่มีแรงดันไปกระตุ้นที่ขาเบสของทรานซิสเตอร์ รีเลย์จึงไม่ทำงานก็เปรียบเสมือนตัดวงจรที่โพลติงออกนั่นเอง

ซึ่งในการพันห่อแปลงมีวิธีการคำนวณดังนี้

1. หาพื้นที่หน้าตัดของเส้นลวดที่ใช้ในการพันขดไฟเข้า (Primary) และขดไฟออก (Secondary) เพื่อที่จะนำไปเปิดตารางเบอร์ของขดลวดที่ใช้ โดยมีหลักการดังนี้

พื้นที่หน้าตัดของขดลวด 400 เซอร์คิวลาร์มิล จะสามารถทนกระแสไฟฟ้าได้ 1 A หรือ พื้นที่หน้าตัดของขดลวด 0.2 ตารางเมตร จะสามารถทนกระแสไฟฟ้าได้ 1 A

ขดปฐมภูมิ 220 V 6 A ขดทุติยภูมิออก 0.1 A มีพื้นที่หน้าตัดของขดลวดที่ใช้ $6 \times 400 = 2400$ เซอร์คิวลาร์มิล (พื้นที่หน้าตัดของขดลวด 400 เซอร์คิวลาร์มิล จะทนกระแสได้ 1A) หรือ ความโตที่ใช้ $6 \times 0.2 = 1.2$ ตร.ม.

เมื่อเปิดตารางเทียบเบอร์ได้ SWG เบอร์ 17

ขดทุติยภูมิพื้นที่หน้าตัดของขดลวดที่ใช้ $0.1 \times 400 = 40$ เซอร์คิวลาร์มิลหรือ ความโตที่ใช้ $0.1 \times 0.2 = 0.02$ ตร.ม.

เมื่อเปิดตารางเทียบเบอร์ได้ SWG เบอร์ 17

2. การคำนวณจำนวนรอบในการพัน Current Transformer

$$\frac{N_p}{N_s} = \frac{I_s}{I_p}$$

ขดไฟเข้า 250 V พันขดจำนวน 5 รอบ มีกระแสไหลที่ขดไฟออก 0.1 A

ต้องการสเกลสูงสุดที่ 5 A

$$\frac{N_p}{N_s} = \frac{I_s}{I_p}$$

$$\frac{5}{N_s} = \frac{0.1}{5}$$

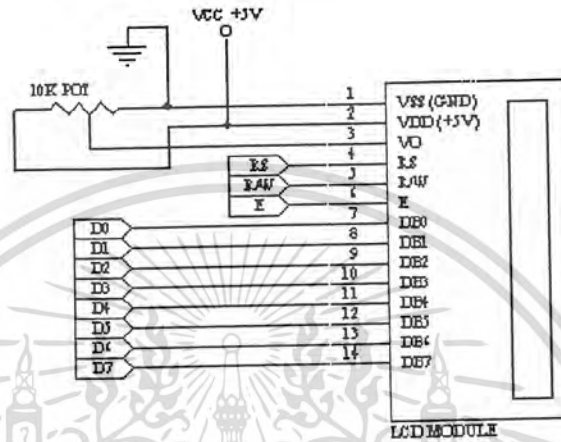
$$0.1 N_s = 25$$

$$N_s = 250$$

ดังนั้นต้องพันที่ขด Secondary จำนวน 250 รอบ

3.2.5 ภาคแสดงผล

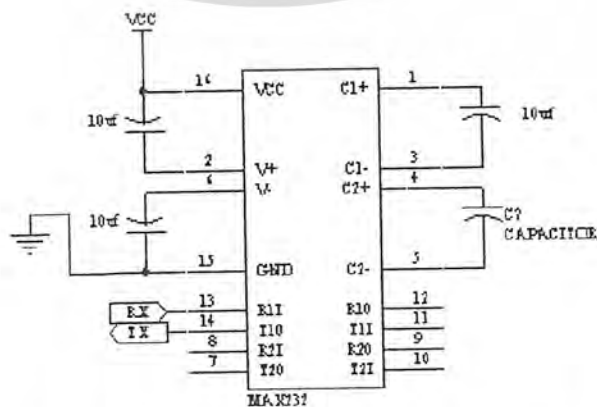
ส่วนของ ภาคแสดงผลนี้จะใช้ LCD Module (Liquid Crystal Display) มาแสดงผลให้แก่ผู้ใช้งานเนื่องจากการแสดงผลของ LCD มีความชัดเจน อ่านง่าย และมีความทนทาน ซึ่งในการต่อใช้งานของ LCD นี้แสดงดังรูปที่ 3-13



รูปที่ 3-13 แสดงวงจรการต่อใช้งาน LCD Module ร่วมกับ Z80-180

3.2.6 ภาครับส่งข้อมูลผ่านพอร์ตอนุกรม RS-232

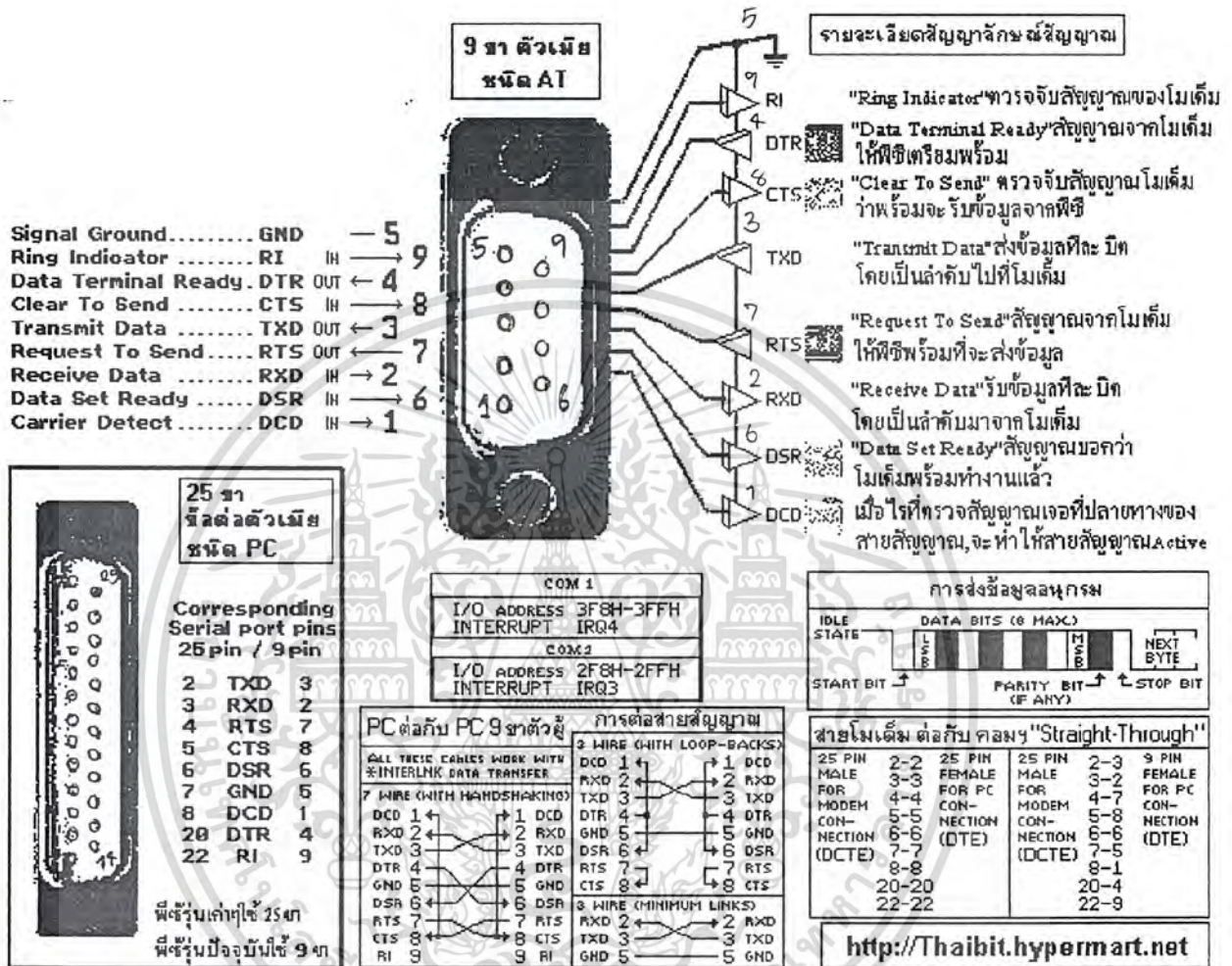
วงจรรับส่งข้อมูลผ่านพอร์ตอนุกรม RS-232 จะมีอุปกรณ์ที่ช่วยในการปรับค่าของแรงดันจากคอมพิวเตอร์ ซึ่งจะมีแรงดันตามมาตรฐาน RS-232 คือ +3V,-3V ถึง +12V,-12V เพื่อให้สามารถทำงานเข้ากันได้กับวงจรหลัก ซึ่งระดับแรงดันเป็นแบบทีทีแอล จึงต้องถือผ่านไอซี MAX232 หรือ ICL232 เพื่อแปลงระดับแรงดันให้อยู่ในระดับทีทีแอลเสียก่อน ซึ่งแสดงการต่อวงจรดังรูปที่ 3-14



รูปที่ 3-14 แสดงวงจรต่อใช้งานของ MAX232

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไมอนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โดยตำแหน่งขาของพอร์ต RS232 คอนเน็คเตอร์แบบ DB-9 แสดงรูปที่ 3-15



รูปที่ 3-15 แสดงตำแหน่งขาใช้งานของพอร์ต RS232 คอนเน็คเตอร์แบบ DB-9

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.3 การเขียนโปรแกรมควบคุมการทำงานของโปรแกรมสวิตช์หลายช่อง

ในการเขียนโปรแกรมนั้นจะแบ่งออกเป็นส่วนต่างๆ ซึ่งจะเขียนเป็นโฟลว์ชาร์ต ได้ดังนี้

3.3.1 โฟลว์ชาร์ตการเขียนโปรแกรมควบคุมคีย์บอร์ด



รูปที่ 3-16 แสดง โฟลว์ชาร์ตการเขียนโปรแกรมของการควบคุมคีย์บอร์ด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

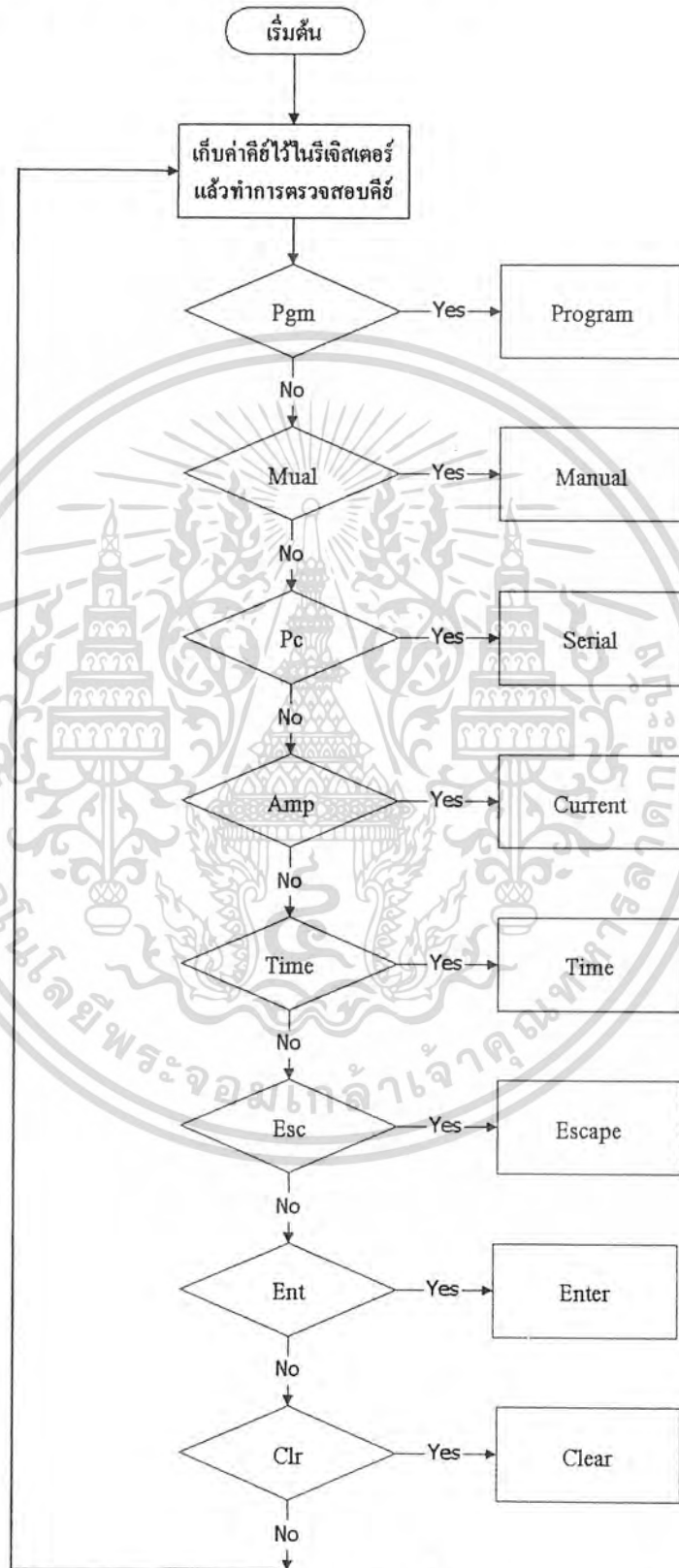
3.3.2 โฟลว์ชาร์ตการเขียนโปรแกรมควบคุม LCD



รูปที่ 3-17 แสดงโฟลว์ชาร์ตการเขียนโปรแกรมควบคุม LCD

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

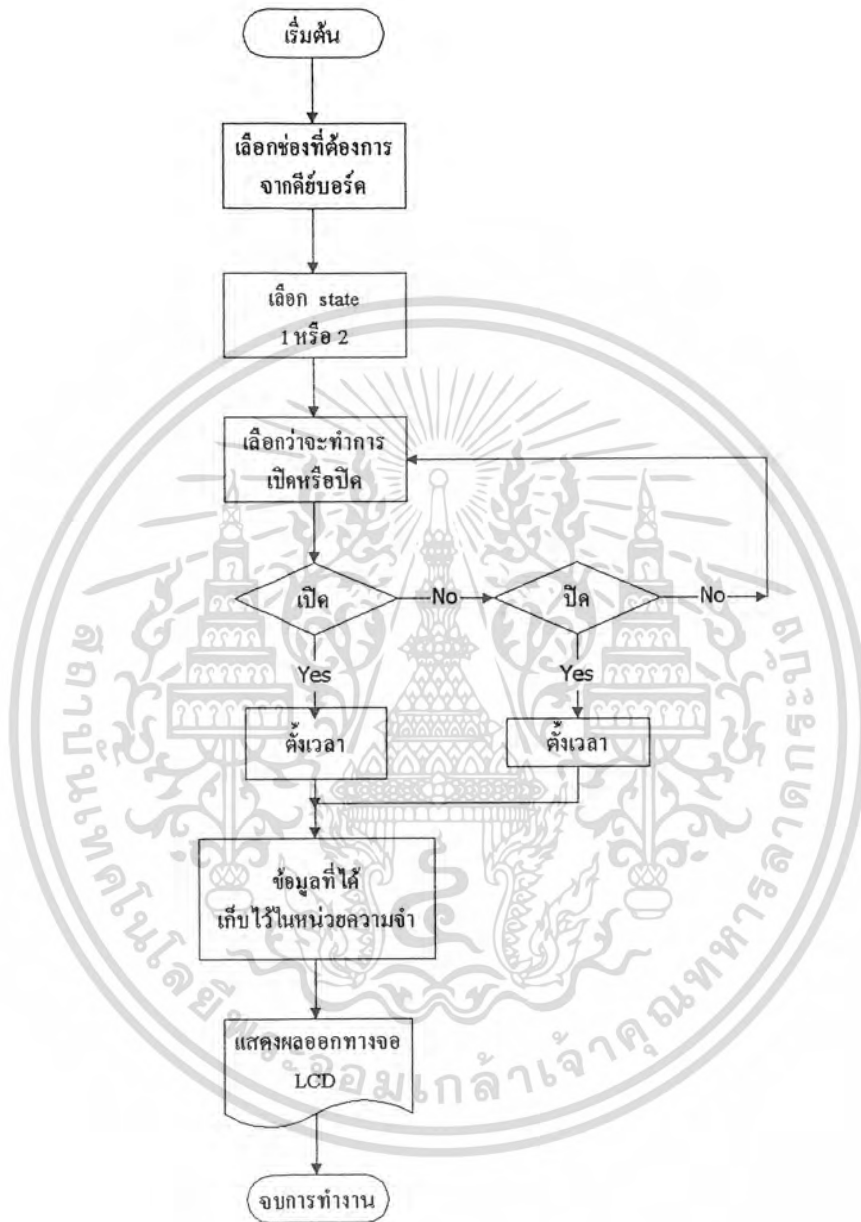
3.3.3 ไฟล์ชาร์ตการเขียนโปรแกรมควบคุมหลัก



รูปที่ 3-18 แสดงไฟล์ชาร์ตการเขียนโปรแกรมควบคุมหลัก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่นิยมนำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

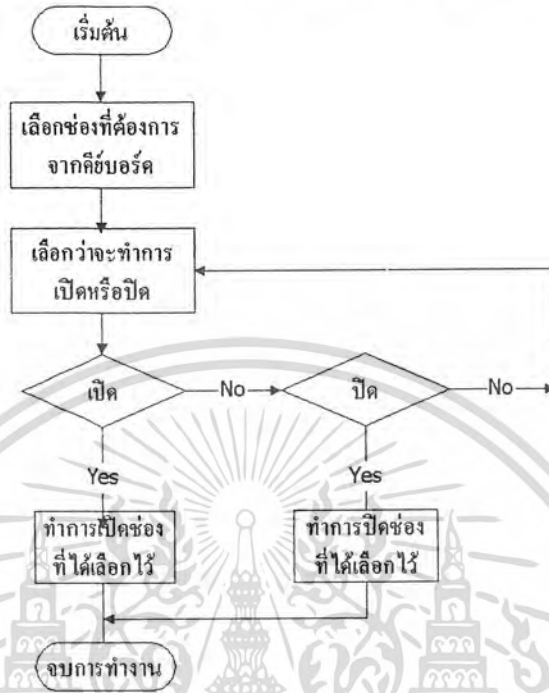
3.3.4 โฟลว์ชาร์ตการเขียนโปรแกรมการทำงานของคีย์ Pgm



รูปที่ 3-19 แสดงโฟลว์ชาร์ตการเขียนโปรแกรมการทำงานของคีย์ Pgm

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.3.5 ไฟล์ชาร์ตการเขียนโปรแกรมการทำงานของคีย์ Mual



รูปที่ 3-20 แสดงไฟล์ชาร์ตการเขียนโปรแกรมการทำงานของคีย์ Mual

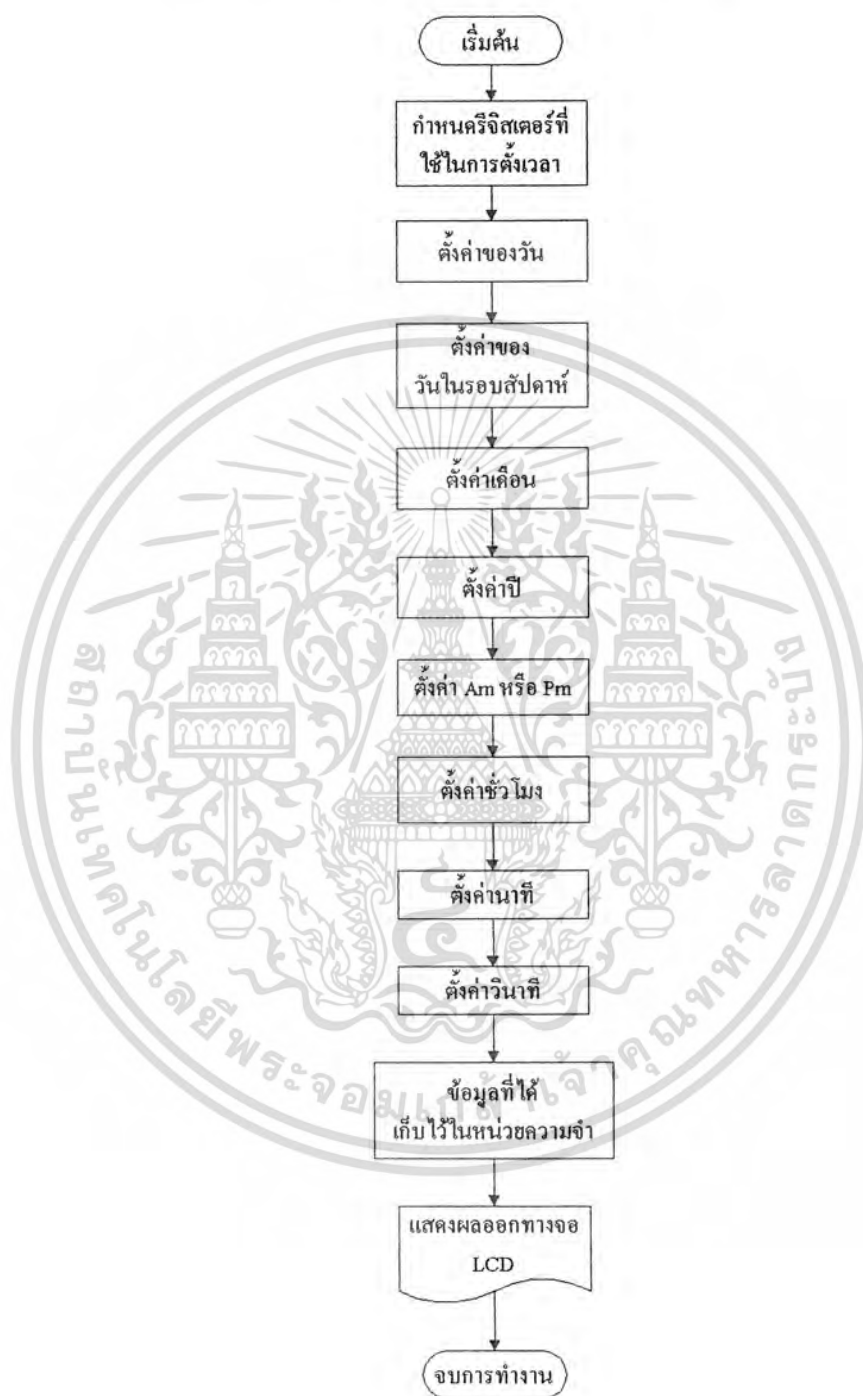
3.3.6 ไฟล์ชาร์ตการเขียนโปรแกรมการทำงานของคีย์ Amp



รูปที่ 3-21 แสดงไฟล์ชาร์ตการเขียนโปรแกรมการทำงานของคีย์ Amp

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์การเขียนโปรแกรมการทำงานของคีย์ Amp โยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.3.7 โฟลว์ชาร์ตการเขียนโปรแกรมการทำงานของกีย์ Time



รูปที่ 3-22 แสดงโฟลว์ชาร์ตการเขียนการทำงานของกีย์ Time

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ซึ่งไฟล์ชาร์ตในส่วนต่างๆที่ได้แสดงให้เห็นนั้น จะแสดงแนวคิดเบื้องต้นในการเขียนโปรแกรมเพื่อให้ผู้ที่จะนำโครงการนี้ไปพัฒนามีหลักการเบื้องต้นในการทำ และเป็นแนวทางให้มีความคิดแปลกใหม่มากยิ่งขึ้น

ซึ่งในส่วนขอของข้อโค้ดนั้นจะอยู่ในส่วนของภาคผนวก ข.



บทที่ 4

การทดลองและผลการทดลอง

บทนี้เป็นการทดลองวัดค่าของแรงดันไฟฟ้าที่ได้จากวงจรตรวจจับกระแส ซึ่งในการทดลองจะใช้อุปกรณ์ไฟฟ้าภายในบ้านมาทำการทดลอง เช่น พัดลม , เตารีด , หม้อหุงข้าว , และ กระจกน้ำร้อน ซึ่งค่าของแรงดันที่วัดได้ จะเทียบกับกระแสที่ไหลภายในสายไฟ และเทียบกับค่าเลขฐาน 16 ขนาด 8 บิต เพื่อป้อนให้แก่วงจรADC

การทดลองที่ 4.1

ทดลองวัดค่าของแรงดันไฟฟ้าที่ได้จากวงจรตรวจจับกระแส

จุดประสงค์

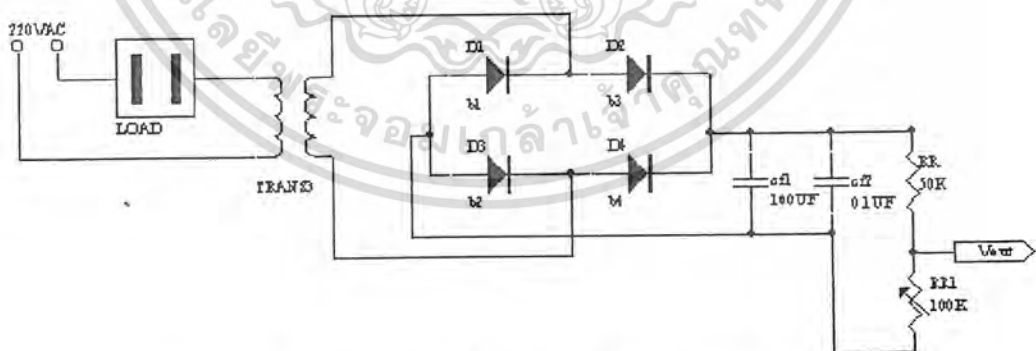
เพื่อหาค่าแรงดันที่ใช้ในการควบคุมกระแสไฟฟ้า

อุปกรณ์ที่ใช้ในการทดลอง

- | | | |
|------------------------|---|---------|
| 1. มัลติมิเตอร์ | 2 | เครื่อง |
| 2. ปลั๊กตัวเมีย | 2 | ตัว |
| 3. สายคียบ | 4 | เส้น |
| 4. ชุดวงจรตรวจจับกระแส | 1 | ชุด |

วิธีการทดลอง

- ประกอบวงจรดังรูปที่ 4-1



รูปที่ 4-1 แสดงวงจรที่ใช้ในการทดลองวัดค่าของแรงดันเอาต์พุต

- นำอุปกรณ์เครื่องใช้ไฟฟ้ามาต่อแล้วทำการวัดแรงดันที่วัด จากนั้นก็เพิ่มเครื่องใช้ไฟฟ้าขึ้นจนมีกระแสตามที่ต้องการ
- บันทึกผลที่ได้ลงในตารางที่ 4-1

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ผลการทดลอง

ค่ากระแสไฟฟ้า (Amp)	ค่าของแรงดันที่วัดได้ (Volt)	เลขฐาน 16
0.0	0	00H
0.5	0.28	15H
1.0	0.9	30H
1.5	1.4	4BH
2.0	1.76	5FH
2.5	2.3	77H
3.0	2.7	8FH
3.5	3	A0H
4.0	3.28	AFH
4.5	3.7	C4H
5.0	4	D5H

ตารางที่ 4-1 แสดงค่าของแรงดันที่วัดได้จากการทดลองที่ 4.1

สรุปผลการทดลอง

1. จากการทดลองสังเกตได้ว่า เมื่อกระแสที่ไหลภายในสายไฟฟ้ามีค่ามากขึ้น จะทำให้แรงดันเอาต์พุต (Vout) ที่ได้มีค่ามากขึ้นตามไปด้วย
2. ค่าแรงดันที่วัดได้จะเพิ่มขึ้นแบบไม่เป็นเชิงเส้น (Non-Linear) ทั้งนี้เนื่องจากชุดอุปกรณ์ตรวจจับกระแสเป็นอุปกรณ์ที่ไม่เป็นเชิงเส้น แต่ค่าที่วัดได้สามารถนำมาใช้งานได้ เนื่องจากว่าโครงการนี้ไม่ได้มีการวัดค่าละเอียด โดยย่านในการกำหนดกระแสจะอยู่ที่ 1 แอมป์ ซึ่งกว้างมาก จึงสามารถนำวงจรชุดนี้มาใช้งานได้

บทที่ 5

บทสรุป

เครื่องโปรแกรมสวิตช์หลายช่องนี้ สามารถนำมาใช้ควบคุมการเปิดและปิดของอุปกรณ์ไฟฟ้าได้หลายๆเครื่อง โดยไม่ต้องมีการควบคุมจากแรงงานคน ทั้งนี้เพราะเครื่องจะทำงานเองโดยอัตโนมัติตามที่ผู้ใช้ตั้งโปรแกรมไว้แล้ว และเวลาที่ได้โปรแกรมนั้นมีความแน่นอนและเที่ยงตรงสูง เนื่องจากผู้จัดทำโครงการได้ทดลองการจับเวลาในการเปิดและปิดของช่องต่างๆ ตามที่ได้โปรแกรมไว้ โดยเปรียบเทียบเวลากับเครื่องคอมพิวเตอร์ ปรากฏว่าเวลาที่วัดได้มีค่าที่ตรงกัน จึงมั่นใจได้ว่าเวลาที่ใช้มีความเที่ยงตรง จากนั้นได้ทำการทดสอบการควบคุมกระแสที่อุปกรณ์ไฟฟ้าที่นำมาต่อกับเครื่องโปรแกรมสวิตช์หลายช่อง (ช่องที่ 9) โดยการตั้งค่าควบคุมกระแสไว้ 2 แอมป์ ปรากฏว่า เมื่อทำการทดสอบการตัดของวงจรของเครื่องจะไปตัดที่กระแส 1.9 แอมป์ ซึ่งใกล้เคียงกับค่าที่ตั้งไว้

ส่วนการควบคุมเครื่องโปรแกรมสวิตช์หลายช่องผ่านคอมพิวเตอร์นั้น จะใช้โปรแกรม Visual Basic 6.0 ทำการรับส่งข้อมูลผ่านพอร์ตอนุกรม RS-232 โดยมีไอซี MAX232 ใช้ในการแปลงระดับแรงดัน ซึ่งจากการทดลองป้อนค่าต่างๆในโปรแกรมสำเร็จรูปที่ได้จัดทำขึ้นมา แล้วส่งข้อมูลไปยังเครื่องโปรแกรมสวิตช์หลายช่อง แล้วสังเกตการทำงานของเครื่องโปรแกรมสวิตช์ ปรากฏว่าเครื่องทำงานตามที่ได้โปรแกรมไว้

ซึ่งจากการทดลองวงจรในภาคต่างๆพบปัญหาดังนี้

1. ปัญหาในส่วนของวงจรควบคุมกระแสซึ่งค่ากระแสที่ควบคุมได้นั้นไม่ตรงกับค่าที่ตั้งไว้ซึ่งเกิดอาการผิดพลาด (Error) ขึ้น เนื่องจากเซนเซอร์ (Sensor) ที่ได้ทำการตรวจจับกระแสผู้จัดทำได้สร้างขึ้นมาเอง จึงทำให้อุปกรณ์ที่นำมาใช้ในวงจรไม่ค่อยมีเสถียรภาพ สาเหตุที่ผู้จัดทำสร้างขึ้นมาเองก็เนื่องจากว่าอุปกรณ์ตัวนี้ (ชุดควบคุมกระแส (Current Sensor)) ถ้าไปซื้อ ราคาจะค่อนข้างสูงประมาณตัวละ 500 บาท
2. จากวงจรจะเห็นว่ามิช่องที่ทำขึ้นมาเพื่อควบคุมกระแส 4 ช่องแต่ใช้เพียง 1 ช่องเนื่องจากมีปัญหาในเรื่องของงบประมาณในการทำ จึงได้ทำเพียง 1 ช่องเพื่อให้มีหลักการในการทำงานและสามารถเพิ่มช่องได้ในภายหลัง ซึ่งในการเพิ่มช่องควบคุมกระแส นั้นจะต้องเพิ่มทั้งในส่วนของวงจรและ ในส่วนของโปรแกรมควบคุมกันไปด้วย
3. การเขียนโปรแกรมสำหรับรับส่งข้อมูลยังมีขีดความสามารถที่น้อยอยู่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

แนวทางการพัฒนาและปรับปรุงต่อไป ผู้จัดทำโครงการมีข้อเสนอแนะดังนี้

1. ในส่วนของวงจรสามารถเพิ่มช่องที่จะใช้ในการควบคุมกระแสได้อีก 3 ช่องโดยส่วนที่ต้องเพิ่มเข้าไปในวงจรก็คือวงจรแลทช์ข้อมูล (Latch) เหตุที่ต้องใช้วงจรแลทช์ข้อมูลก็เพราะว่าเรามีพอร์ตไม่เพียงพอกับความต้องการของวงจร ซึ่งโดยปกติถ้าเราไม่ใช้วงจรแลทช์ข้อมูลวงจรจะต้องการใช้พอร์ตในการส่งข้อมูลทั้งหมด 4 พอร์ต (32บิต) คือ 1 ช่องต่อ 1 พอร์ต แต่ถ้าหากเรานำวงจรแลทช์ข้อมูลมาต่อเข้ากับวงจรก็จะทำให้ใช้พอร์ตในการส่งข้อมูลเพียง 2 พอร์ตเท่านั้นคือ พอร์ตที่ใช้ในการส่งข้อมูล (Data) และพอร์ตที่ใช้ในการควบคุมอุปกรณ์แลทช์ข้อมูลว่าจะให้อุปกรณ์แลทช์ข้อมูลตัวไหนรับข้อมูลแล้วคงสถานะทางเอาท์พุทไว้

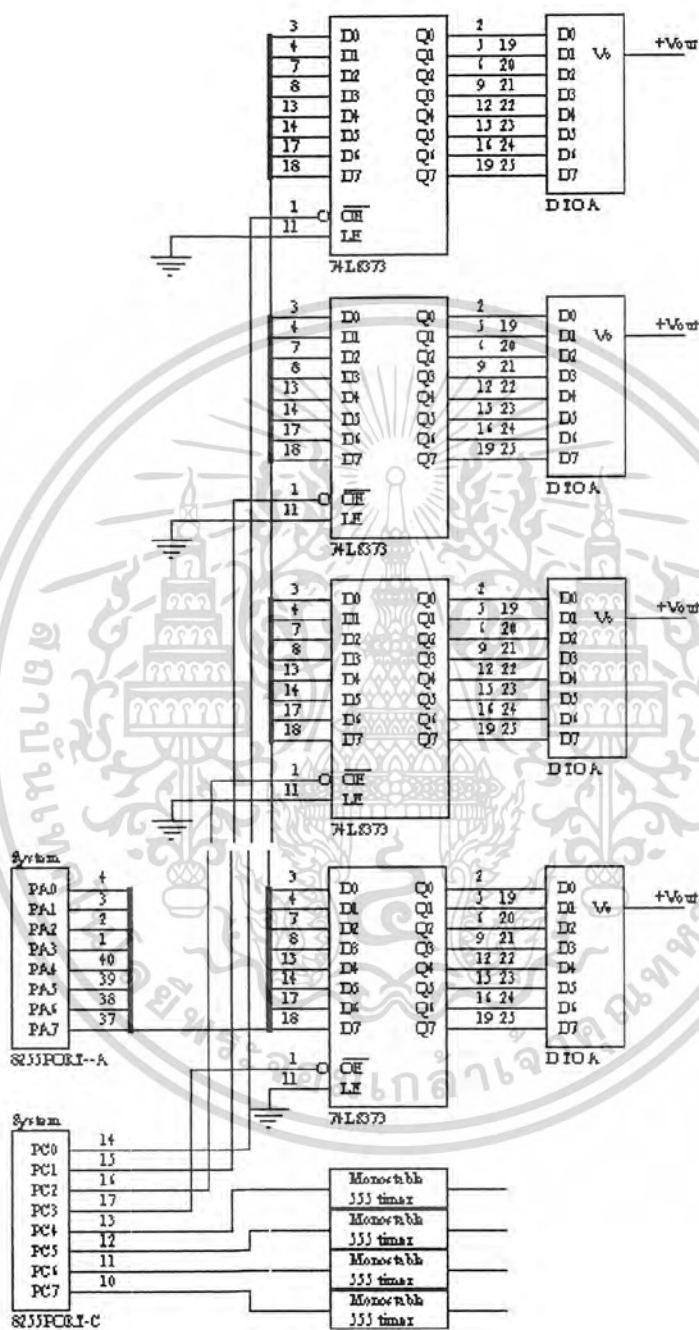
แล้วยังมีส่วนของวงจรที่สำคัญที่จะต้องเพิ่มเติมอีกคือ วงจรโมโนสเตเบิลมัลติไวเบรเตอร์ ซึ่งปกติจะมีอยู่ชุดเดียว แต่ถ้าหากต้องการเพิ่มช่องควบคุมกระแสของอุปกรณ์ไฟฟ้าก็ทำได้โดยการเพิ่มวงจรโมโนสเตเบิลมัลติไวเบรเตอร์ขึ้นตามจำนวนช่องที่จะทำการควบคุมกระแส

2. ในส่วนของโปรแกรมจะต้องเพิ่มเติมในส่วนของการควบคุมกระแส โดยเพิ่มช่องการใช้งานให้มากขึ้น ทั้งนี้ขึ้นอยู่กับการทำงานของวงจรที่ได้ออกแบบไว้ด้วย

3. การที่จะให้ภาคควบคุมกระแสมีการควบคุมที่แม่นยำ จะต้องใช้อุปกรณ์ตรวจจับกระแสที่มีประสิทธิภาพมากยิ่งขึ้น แต่ราคาของอุปกรณ์ก็จะยิ่งสูงขึ้นตามไปด้วย

และในส่วนของการพัฒนาวงจรมัน ผู้จัดทำได้ออกแบบในส่วนของวงจรแลทช์ข้อมูลสำหรับเพิ่มช่องในการควบคุมกระแสไฟฟ้าของอุปกรณ์ โดยจะแสดงวงจรดังรูปที่ 5-1 และจะอธิบายหลักการทำงานอย่างคร่าวๆ ดังนี้

วงจรแลทช์ข้อมูล (Latch Data)



รูปที่ 5-1 แสดงวงจรแลทช์ข้อมูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

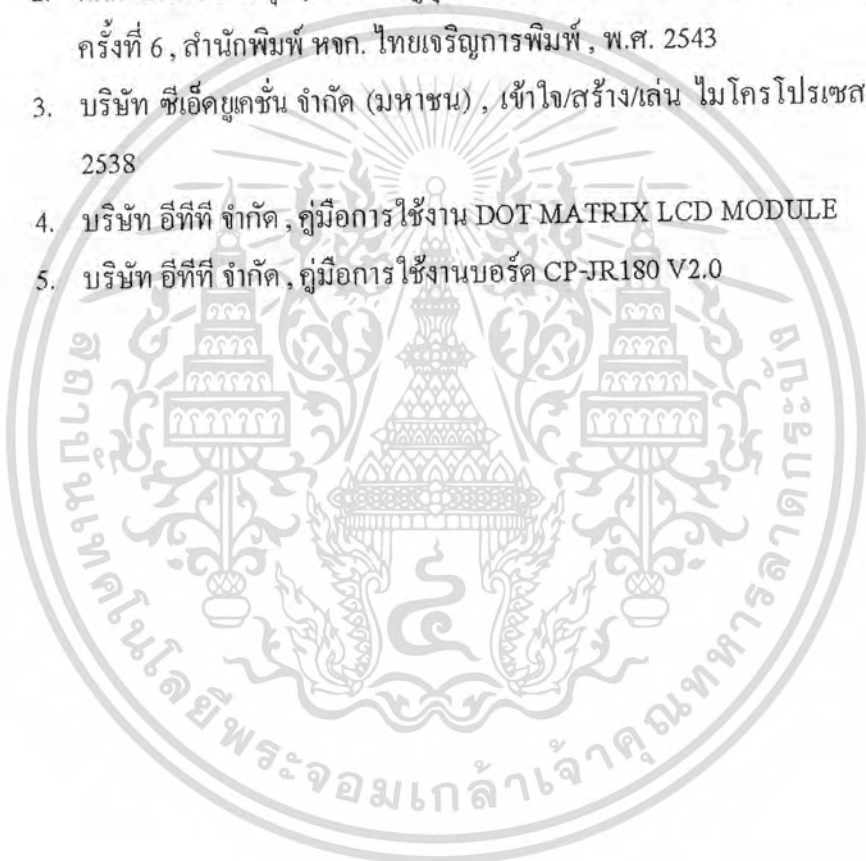
หลักการทํางานของวงจร

จากวงจรไอซี 74LS373 ใช้ทำหน้าที่ในการแลตซ์ข้อมูล หรือทำหน้าที่คงค่าข้อมูลเดิมไว้ถึงแม้ว่าจะไม่มีอินพุตเข้ามาก็ตาม โดยหลักการทํางานก็คือจะทำการป้อนข้อมูลเข้าที่ขา D0 - D7 จากนั้นจะทำการป้อนลอจิก “0” เข้าที่ขา Enable G (ขา11) โดยขา Output Control (ขา1) จะต่อลงกราวด์เพื่อให้อุปกรณ์ไตรสเตจทํางาน ซึ่งการคงค่าข้อมูลของ ไอซีตัวนี้ก็คือจะต้องป้อนลอจิก “0” ที่ขา 1 และขา 11 พร้อมกัน จึงจะทำให้ค่าของข้อมูลคงค่าเดิมถึงแม้จะเปลี่ยนข้อมูลทางด้านอินพุตใหม่ก็ตาม และถ้าหากป้อนลอจิก “1” ที่ขา G จะทำให้ค่าข้อมูลทางด้านเอาต์พุตเปลี่ยนแปลงตามข้อมูลทางด้านอินพุตทันที ซึ่งการต่อลักษณะนี้จะไม่คงค่าข้อมูลเดิมไว้ ถึงแม้ว่าข้อมูลทางอินพุตจะมีการเปลี่ยนแปลงก็ตาม ซึ่งคุณสมบัติตรงนี้เองเราจึงนำมาออกแบบวงจรเพื่อใช้ในการพัฒนาโครงการให้มีประสิทธิภาพสูงขึ้น



บรรณานุกรม

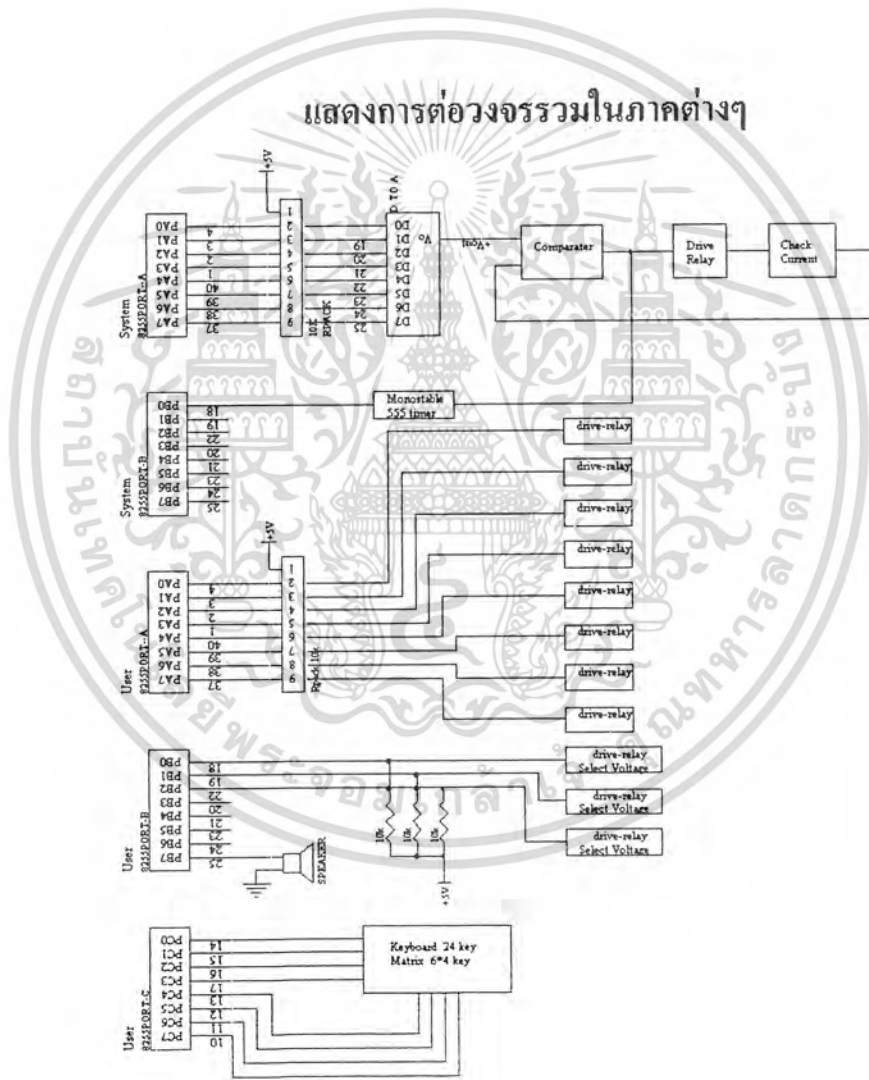
1. กฤษดา ไชเย็น, อรรถพล บุญยะ โภคา , ชัยวัฒน์ ลิ้มพรจิตรวิไล, เรียนรู้และปฏิบัติการเชื่อมต่อคอมพิวเตอร์กับอุปกรณ์ภายนอกผ่านพอร์ตอนุกรม , พ.ศ. 2540
2. กิตติ ภักดีวิวัฒนะกุล , จำลองกรอุตสาหะ , Visual Basic 6 ฉบับโปรแกรมเมอร์ , พิมพ์ครั้งที่ 6 , สำนักพิมพ์ หจก. ไทยเจริญการพิมพ์ , พ.ศ. 2543
3. บริษัท ซีเอ็ดดูเคชั่น จำกัด (มหาชน) , เข้าใจ/สร้าง/เล่น ไมโครโปรเซสเซอร์ 1 , พ.ศ. 2538
4. บริษัท อีทีที จำกัด , คู่มือการใช้งาน DOT MATRIX LCD MODULE
5. บริษัท อีทีที จำกัด , คู่มือการใช้งานบอร์ด CP-JR180 V2.0



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ก. รายละเอียดของวงจร

1. แสดงวงจรรวมทั้งหมดของเครื่องโปรแกรมสวิตช์หลายช่อง

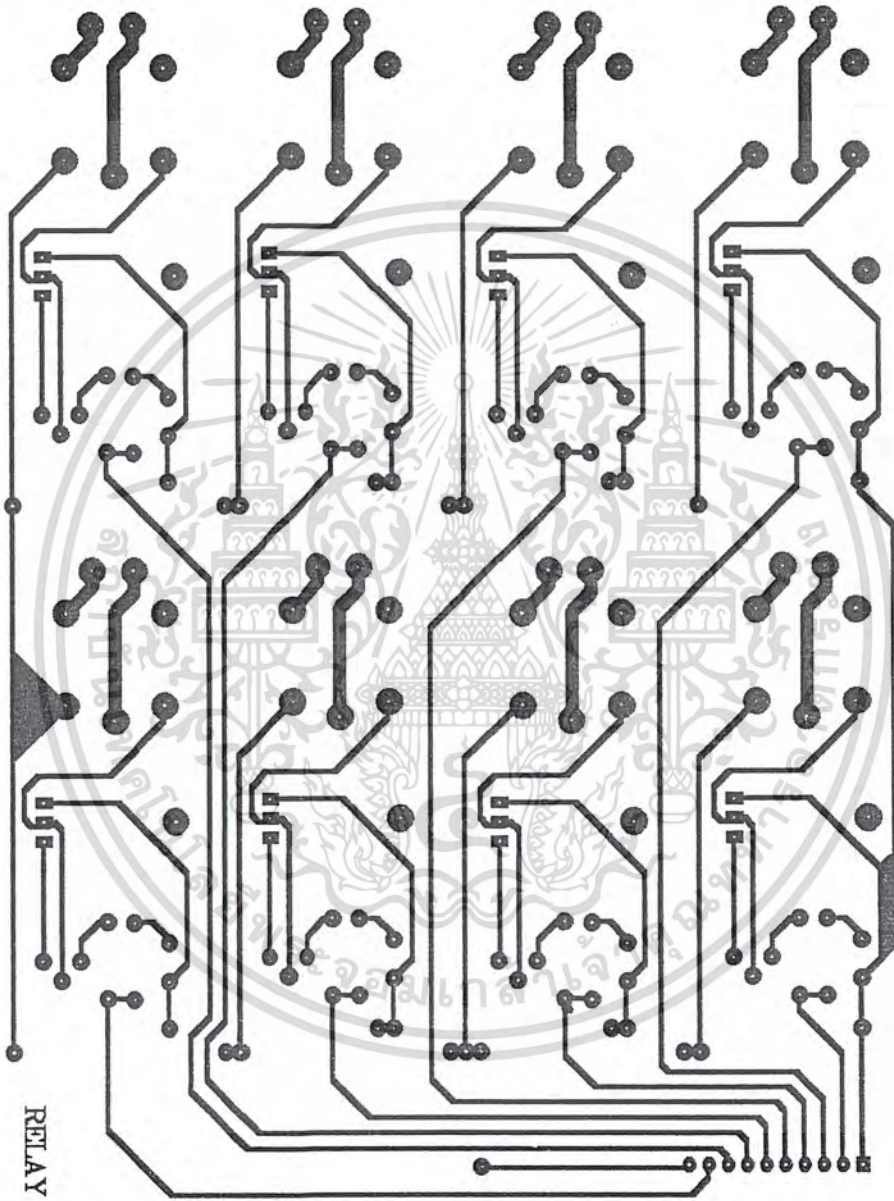


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

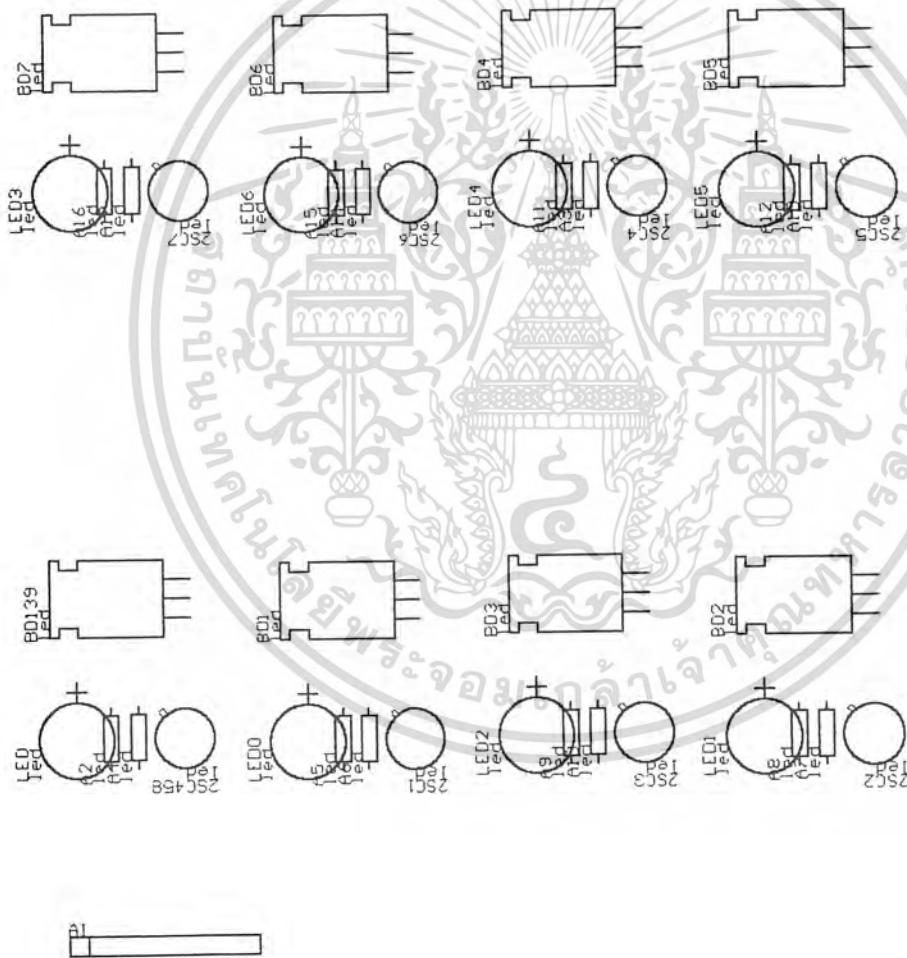


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2. ลายวงจรชุดควบคุมการเปิดและปิดของอุปกรณ์ไฟฟ้า

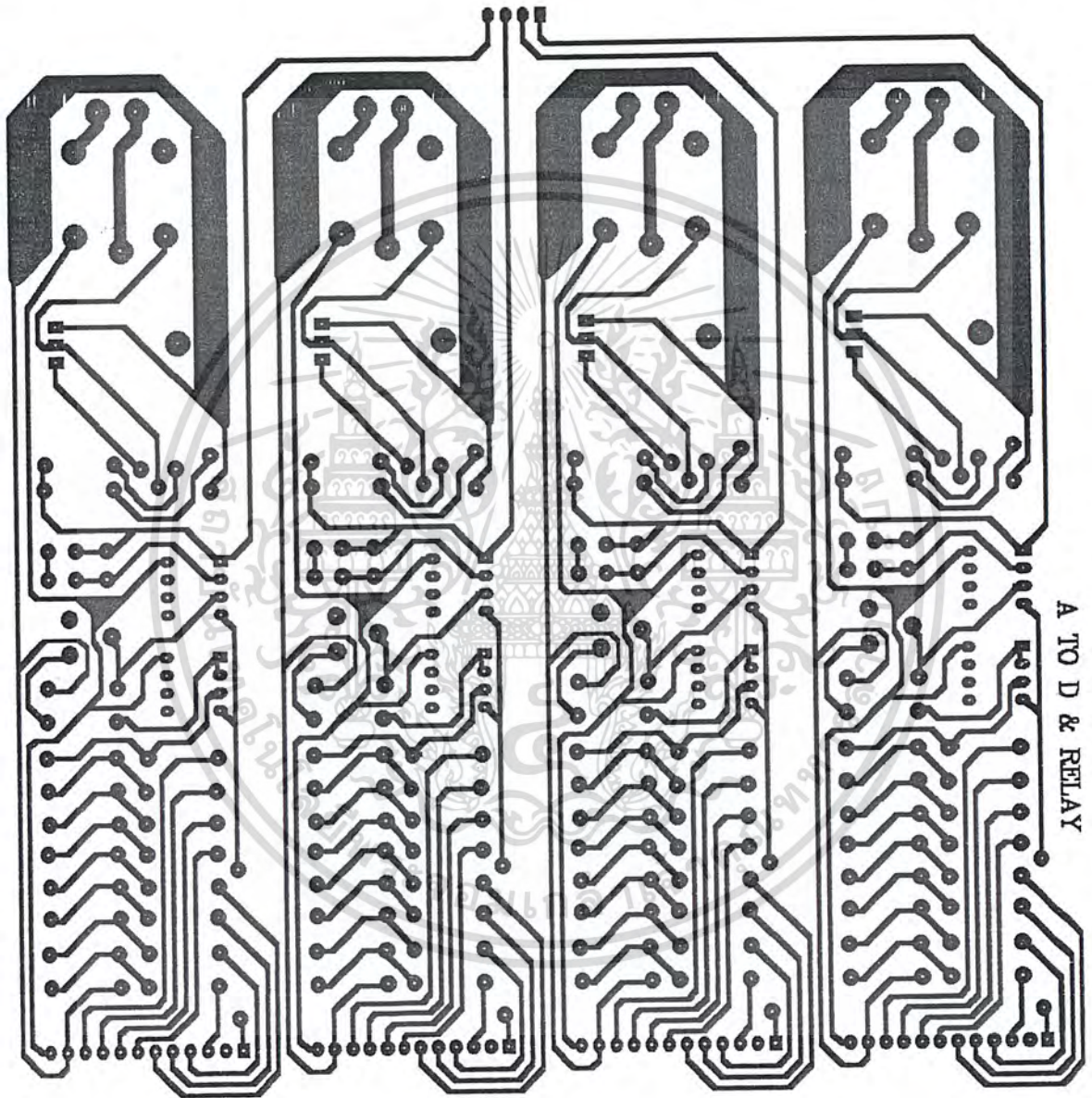


3. ตำแหน่งการวางตัวอุปกรณ์ของวงจรควบคุมการเปิดและปิดของอุปกรณ์ไฟฟ้า



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

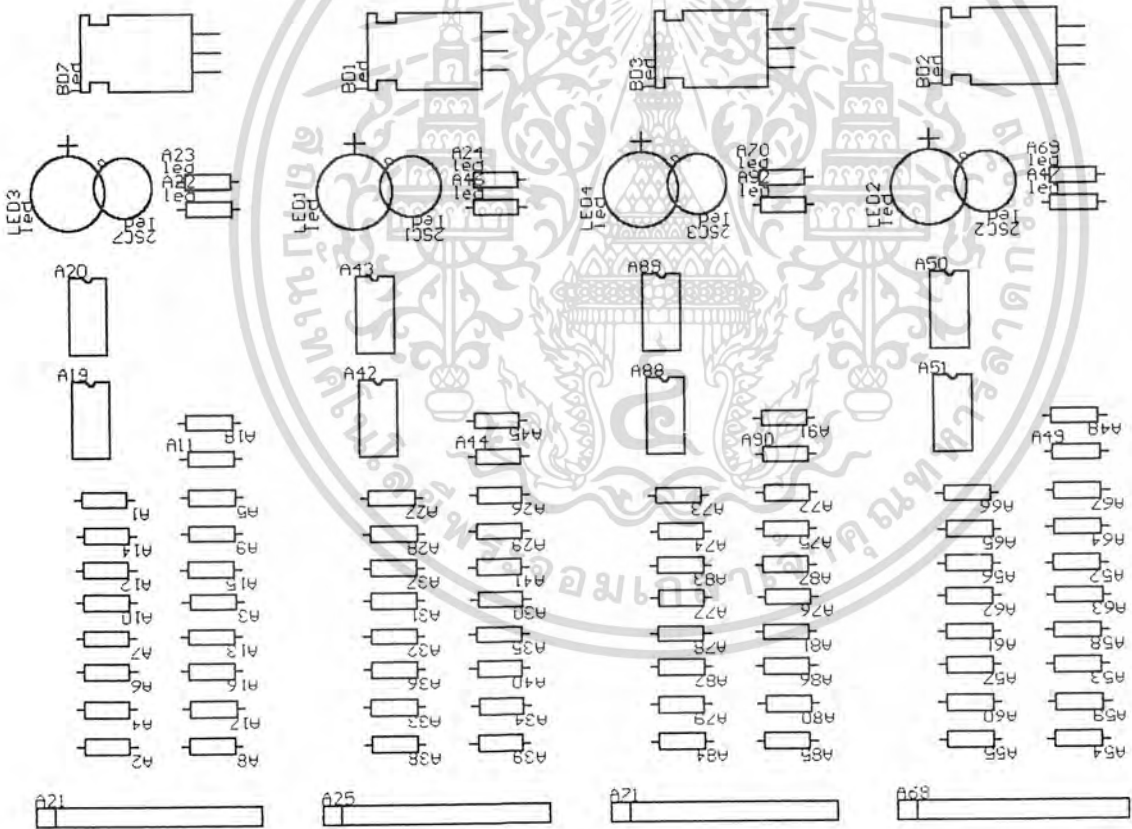
4. ลายวงจรของภาคควบคุมกระแส



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5. ตำแหน่งการวางอุปกรณ์ของภาคควบคุมกระแส

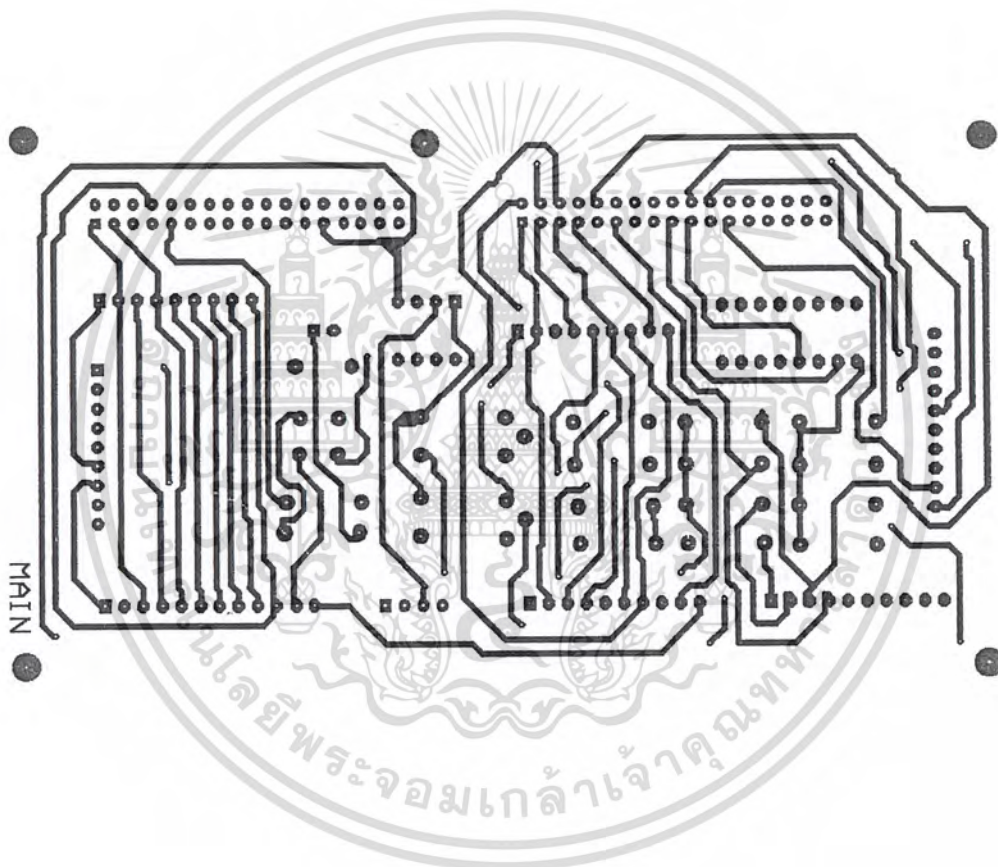
893



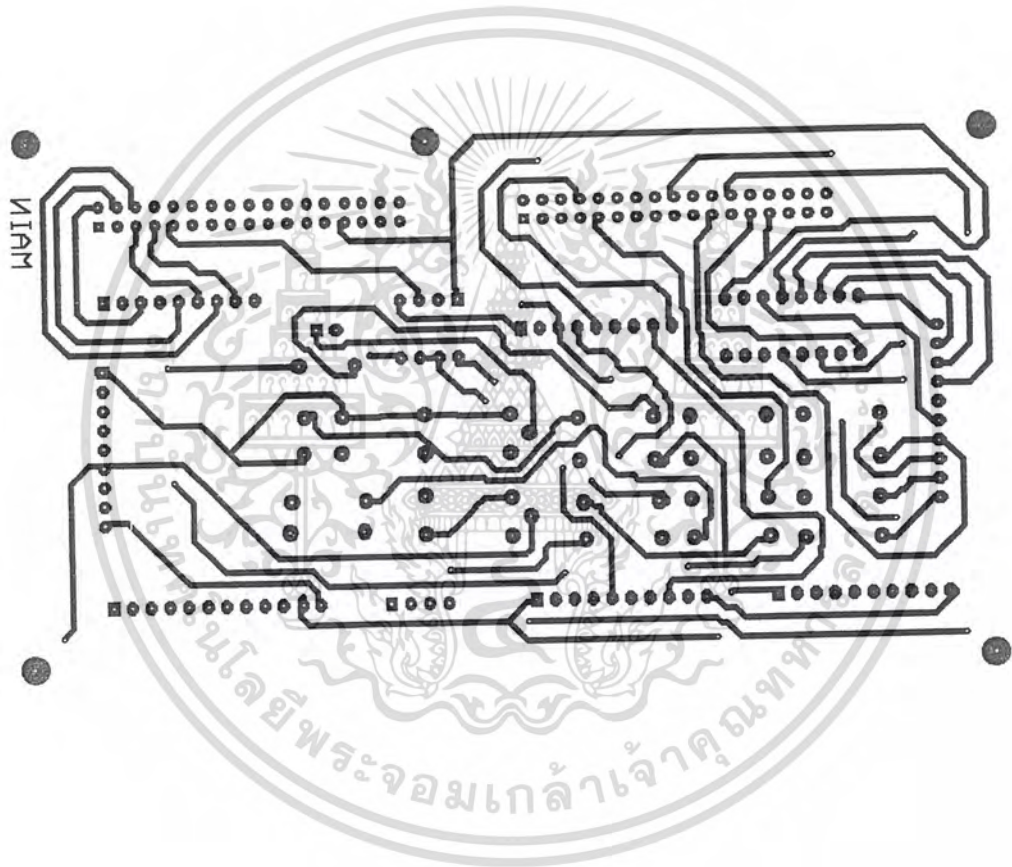
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

6. ลายวงจรของวงจรควบคุมหลัก

ด้านหน้า

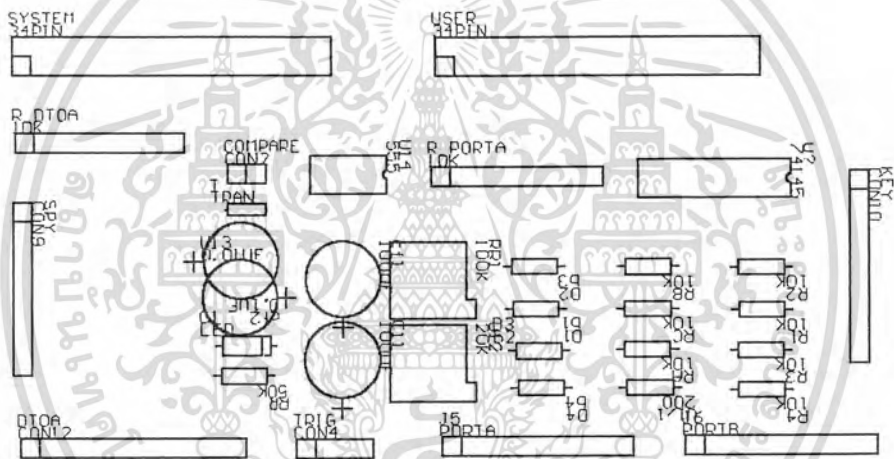


ด้านหลัง



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

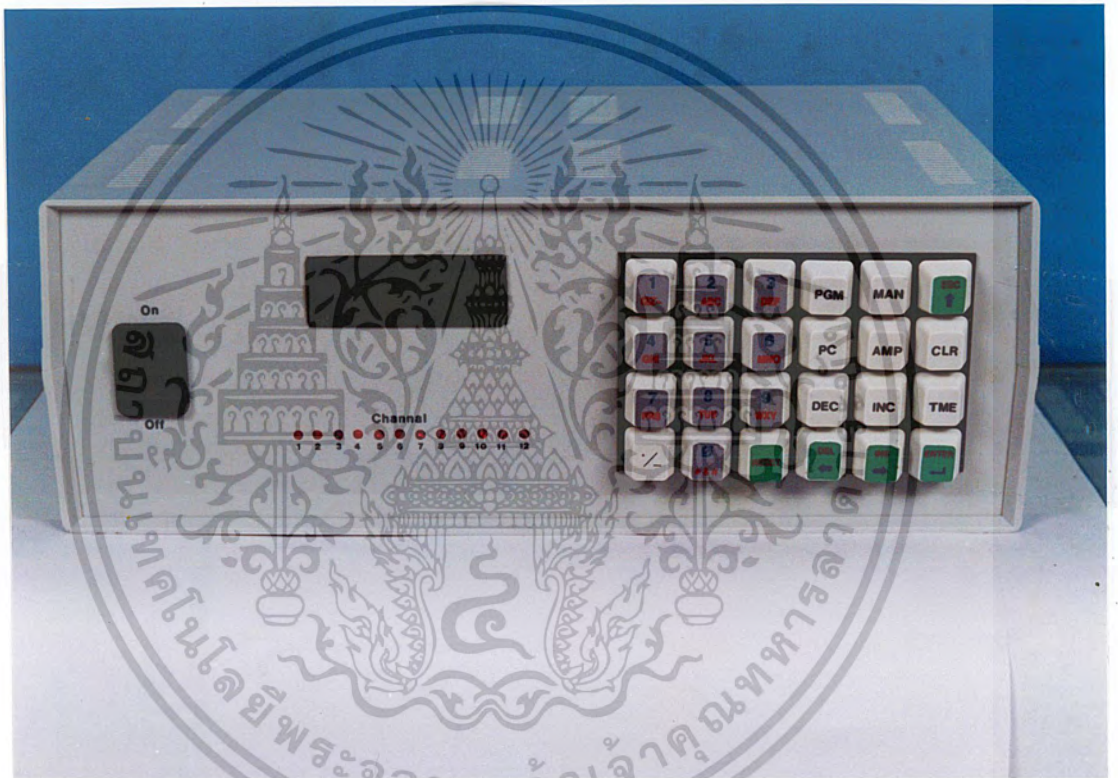
7. ตำแหน่งการวางอุปกรณ์ของวงจรควบคุมหลัก



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

8. ภาพถ่ายของเครื่องโปรแกรมสวิตช์หลายช่อง

ด้านหน้า

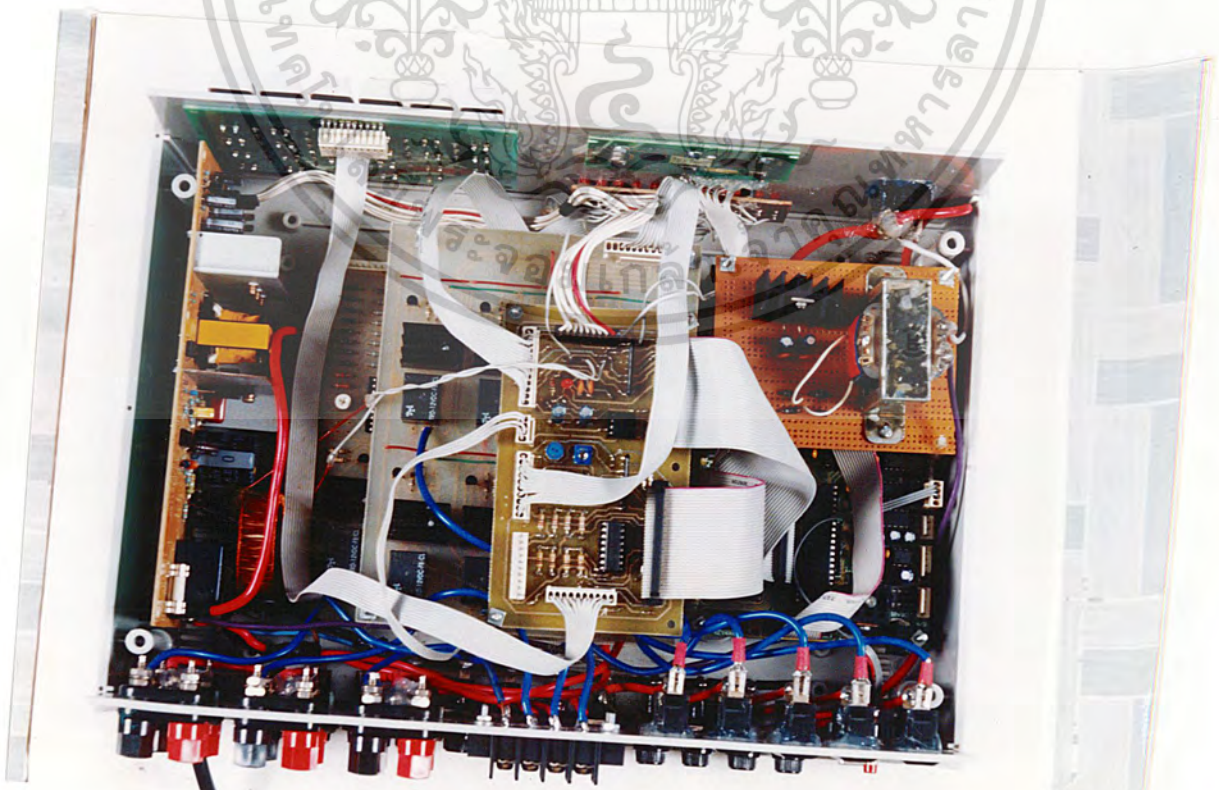


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ด้านหลัง



ด้านบน



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ข.

```

;*****
;PROGRAMMABLE MULTI-SWITCH * .EQU TCR,10H
;ASM SOFTWARE A180Z * .EQU TMDR1L,14H
;MONITOR PROGRAM COMMON AREA 0 * .EQU TMDR1H,15H
; (LOGICAL ADDRESS) 0000H-7FFFH * .EQU RLDR1L,16H
; (PHYSICAL ADDRESS) 0000H-7FFFH * .EQU RLDR1H,17H
; RAM BANK AREA * ;*** OTHER ***
; (LOGICAL ADDRESS) 8000H-EFFFH * ;*** DMA ****
; (PHYSICAL ADDRESS) 10000H-16FFFH * .EQU SAR0L,20H
; RAM COMMON AREA 1 * .EQU SAR0H,21H
; (LOGICAL ADDRESS) F000H-EFFFH * .EQU SAR0B,22H
; (PHYSICAL ADDRESS) 17000H-1FFFFH * .EQU DAR0L,23H
; .EQU DAR0H,24H
;*** ASCI ***** .EQU DAR0B,25H
; .EQU FR1,0DBH .EQU BCR0L,26H
; .EQU FR2,0B0H .EQU BCR0H,27H
; .EQU CNTLA0,0 .EQU MAR1L,28H
; .EQU CNTLA1,1 .EQU MAR1H,29H
; .EQU CNTLB0,2 .EQU MAR1B,2AE
; .EQU CNTLB1,3 .EQU IAR1L,2BH
; .EQU STAT0,4 .EQU IAR1H,2CH
; .EQU STAT1,5 .EQU BCR1L,2EH
; .EQU TDR0,6 .EQU BCR1H,2FH
; .EQU TDR1,7 .EQU DSTAT,30H
; .EQU RDR0,8 .EQU DMODE,31H
; .EQU RDR1,9 .EQU DCNTL,32H
;*** CSI/O *** ;*** INT ***
; .EQU CNTR,0AH .EQU IL,33H
; .EQU TRDR,0BH .EQU ITC,34H
;*** TIMER *** ;*** REFRESH ***
; .EQU TMDR0L,0CH .EQU RCR,36H
; .EQU TMDR0H,0DH ;*** MMU ***
; .EQU RLDR0L,0EH .EQU CBR,38H
; .EQU RLDR0H,0FH .EQU BBR,39H

```

```

.EQU CBAR,3AH
;*** I/O ***
.EQU OMCR,3EH
.EQU ICR,3FH
;*** I/O 8255
.EQU COM_PORT,0C0H
.EQU WR_PORT,0C2H
.EQU BUSY_PORT,0C4H
.EQU PORTA,80H
.EQU PORTB,81H
.EQU PORTC,82H
.EQU CPORT,83H
.EQU PORTAA,60H
.EQU PORTBB,61H
.EQU PORTCC,62H
.EQU CCPORT,63H
.EQU RTC,0A0H ;USED PORT
RTC6242
.EQU SEC1,RTC
.EQU SEC10,RTC+1
.EQU MIN1,RTC+2
.EQU MIN10,RTC+3
.EQU HOUR1,RTC+4
.EQU HOUR10,RTC+5
.EQU DAY1,RTC+6
.EQU DAY10,RTC+7
.EQU MON1,RTC+8
.EQU MON10,RTC+9
.EQU YEAR1,RTC+10
.EQU YEAR10,RTC+11
.EQU WEEK1,RTC+12
.EQU CREG_D,RTC+0DH
.EQU CREG_E,RTC+0EH
.EQU CREG_F,RTC+0FH
;
;
; END I/O MAP REG Z80180
.EQU ORIGIN,8000H
.ORG ORIGIN
;*****
: POWER UP DELAY *
;*****
INIT: LD HL,8000H ;DELAY
INIT1: DEC HL
LD A,H
OR L
JP NZ,INIT1
;*** GET MEMORY MAP ***
INIT2: LD A,0F8H
OUT0(CBAR),A
LD A,8H
OUT0(BBR),A
OUT0(CBR),A
LD SP,0FFFFH
;*** SET INTERRUPT VECTOR ***
LD A,0
LD IA ;VECTOR HIGH
LD A,40H ;VECTOR LOW
OUT0(IL),A
IM 1 ;FOR INTO
LD A,8 ;int ascii chanel 1
OUT0(STAT1),A
LD A,64H ;TX,RX,8BIT,1STOP
OUT0(CNTLA1),A
LD A,2H ;1200 AT X'TAL 12.288
MHz
OUT0(CNTLB1),A
;*** SET INITIAL INTERNAL I/O ***
LD A,40H

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

OUT0 (OMCR),A ;SAME Z80 LD (IX+4),00H
LD A,83H LD (IX+5),00H
OUT0 (RCR),A ;ENABLE REFRESH MAN_REM: LD (IX+21H),00H
IN0 A,(DCNTL) LD A,(IX+40H)
OR 70H CP 18H
OUT0 (DCNTL),A ;MEM 2 WIAT IO 4 JP NZ,TIME
;-----; JP MAIN
; CONTROL ANOTHER USER PORT : ;-----;
;-----; ; MAIN PROGRAM ;
CALL DELAY1 ;-----;
CALL DELAY1 MAIN: LD A,00001100B
LD A,88H ;-----> CONTROL CALL LCD_COM
PORT 8255 USER LD HL,SET_DAY2
OUT (CPORT),A CALL DIS_LCD
LD A,89H ;-----> CONTROL LD HL,SET_TME2
PORT 8255 SYSTEM CALL DIS_LCD1
OUT (CCPORT),A MAIN: LD B,00H
LD A,1 ;-----> CONTROL SKLP: LD A,B
DISPLAY_LCD OUT (PORTC),A
CALL LCD_COM IN A,(PORTC)
LD A,000111000B LD E,A
CALL LCD_COM CP 0F0H
LD A,00H JP C,NEXT_ST1
OUT (PORTAA),A INC B
LD A,88H LD A,B
LD (0D375H),A CP 06H
LD IX,0E000H JP NZ,SKLP
LD A,(IX+18H) PUSH BC
CP 44H CALL COMP
JP Z,MAN_REM1 CALL COUNT
LD (IX+19H),00H CALL CHANNAL1
MAN_REM1: LD A,(IX+23H) CALL GETCK
CP 50H CALL INTRTC1S
JP Z,MAN_REM POP BC

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

CALL OUTPUT	CALL DIS_LCD
JP MAIN1	LD HL,PLEASE
;	CALL DIS_LCD)
NEXT_ST1: CALL SOUND	SCANK: LD B,00H
CALL CHK_KEY	LOOPS: LD A,B
CALL DISCHKM	OUT (PORTC),A
CALL DIS_TM	IN A,(PORTC)
JP MAIN	LD E,A
;	CP 0F0H
CONST: LD A,00001100B	JP C,OPL
CALL LCD_COM	CALL SOUND
CALL SCANKEY	INC B
CALL SOUND	LD A,B
CALL CHK_KEY	CP 0AH
CALL DISCHKM	JP NZ,LOOPS
CALL DIS_TM	JP SCANK
RET	OPL: RET
;	
OUTPUT: LD IX,0E000H	MANUAL: LD IX,0E000H
LD A,(IX+4)	LD (IX+0),30H
OUT (PORTA),A	LD (IX+1),30H
LD A,(IX+5)	LD (IX+2),00H
OUT (PORTB),A	LD (IX+3),01H
;	LD (IX+0DH),00H
IN A,(PORTCC)	LD HL,MAN
AND 80H	CALL DIS_LCD
CP 80H	LD HL,PGM
JP NZ,OPL	CALL DIS_LCD1
LD (IX+19H),00H	LP_MAN: CALL SL_CHN
LD (IX+18H),00H	CALL DIS_MAN
LD A,(IX+19H)	JP LP_MAN
OUT (PORTAA),A	SL_CHN: LD A,00001101B
;	CALL LCD_COM
LD HL,OVER	CALL SHOW_MAN

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

CALL SCANKEY	CP 01H
CALL SOUND	JP NZ,TEST_E1
CALL CHK_KEY	LD A,(IX+2)
CALL DISCHK	CP 00H
RET	JP Z,MANUAL
:	LD A,(IX+1)
DIS_MAN: LD HL,CHN_KEY	CP 32H
LD A,L	JP C,NX_ENT
ADD A,B	CALL ERR
LD L,A	JP MANUAL
LD A,(HL)	:
LD B,A	TEST_E1: LD A,(IX+3)
CP 49H	CP 03H
JP Z,DEC_MAN	JP NZ,PGM_CHN
CP 44H	LD A,(IX+1)
JP Z,INC_MAN	CP 32H
CP 45H	JP C,C_CHK2
JP Z,ESC_MAN	CALL ERR
CP 3CH	JP CURRENT
JP Z,DEC_MAN	:
CP 3EH	C_CHK2: CP 31H
JP Z,INC_MAN	JP NZ,C_CHK2
CP 65H	LD A,(IX+0)
JP Z,ENT_MAN	CP 33H
CP 10H	JP C,ENT_CUR
JP Z,NON	CALL ERR
LD D,A	JP CURRENT
LD A,(IX+0)	:
LD (IX+0),D	C_CHK22: CP 33H
LD (IX+1),A	JP C,C_CHK23
INC (IX+2)	CALL ERR
NON: RET	JP CURRENT
:	C_CHK23: CP 30H
ENT_MAN: LD A,(IX+3)	JP NZ,MAIN

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

LD A,(IX+0)
CP 39H
JP Z,ENT_CUR
CP 30H
JP Z,ENT_CUR
CALL ERR
JP CURRENT
NX_ENT: CP 31H
JP NZ,STATUS
LD A,(IX+0)
CP 33H
JP C,STATUS
CALL ERR
JP MANUAL
;
ERR: LD A,45H
LD (IX+1),A
LD A,72H
LD (IX+0),A
CALL SHOW_MAN
CALL SD_ERR
RET
;
;
ESC_MAN: LD A,(IX+2)
CP 00H
JP Z,ESCAPE
LD A,(IX+0DH)
CP 00H
JP Z,MANUAL
JP ESCAPE
;
SHOW_MAN:LD IX,0E000H
LD A,0CEH
LD A,(IX+0)
CP 39H
JP Z,ENT_CUR
CP 30H
JP Z,ENT_CUR
CALL ERR
JP CURRENT
CALL LCD_COM
LD A,(IX+0)
CALL WR_ASCII
LD A,0CDH
CALL LCD_COM
LD A,(IX+1)
CALL WR_ASCII
RET
;
INC_MAN: INC (IX+2)
INC (IX+0)
LD A,(IX+0)
CP 3AH
JP C,LP_MAN
LD (IX+0),30H
INC (IX+1)
LD A,(IX+1)
CP 3AH
JP C,LP_MAN
LD (IX+1),30H
JP LP_MAN
;
DEC_MAN: LD A,(IX+0)
CP 30H
JP NZ,NX_DEC1
LD A,39H
LD (IX+0),A
LD A,(IX+1)
CP 30H
JP NZ,NX_DEC
LD A,39H
LD (IX+1),A
JP LP_MAN
NX_DEC: DEC (IX+1)

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

JP LP_MAN	JP Z,ENT_STA
NX_DEC1: DEC (IX+0)	RET
JP LP_MAN	:
STATUS: LD (IX+6),4FH	INC_STA: LD A,(IX+8)
LD (IX+7),66H	CP 66H
LD (IX+8),66H	JP NZ,STATUS
LD HL,STAS	LD (IX+6),20H
CALL DIS_LCD	LD (IX+7),4FH
DIS_OF: LD A,00001101B	LD (IX+8),6EH
CALL LCD_COM	JP DIS_OF
CALL SHOW_OF	ENT_STA: LD A,(IX+8)
CALL SCANKEY	CP 6EH
CALL SOUND	JP Z,SET_ON
CALL CHK_KEY	CP 66H
CALL DISCHK	JP Z,SET_OFF
CALL DIS_STA	JP MANUAL
JP DIS_OF	:
:	SET_ON: LD A,(IX+1)
DIS_STA: LD HL,CHN_KEY	CP 31H
LD A,L	JP Z,TEN_CHN
ADD A,B	LD A,(IX+0)
LD L,A	CP 31H
LD A,(HL)	JP Z,CHN1
CP 49H	CP 32H
JP Z,INC_STA	JP Z,CHN2
CP 44H	CP 33H
JP Z,INC_STA	JP Z,CHN3
CP 45H	CP 34H
JP Z,MANUAL	JP Z,CHN4
CP 3CH	CP 35H
JP Z,INC_STA	JP Z,CHN5
CP 3EH	CP 36H
JP Z,INC_STA	JP Z,CHN6
CP 65H	CP 37H

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

LD (IX+11H),00H	LD A,L
LD (IX+15H),00H	ADD A,B
LD (IX+16H),00H	LD L,A
LD A,00001101B	LD A,(HL)
CALL LCD_COM	LD B,A
EN_CURR: CALL SW_CURR	CP 49H
CALL SCANKEY	JP Z,DEC_CURR
CALL SOUND	CP 44H
CALL CHK_KEY	JP Z,INC_CURR
CALL DISCHK	CP 45H
CALL DIS_CURR	JP Z,ESC_CURR
JP EN_CURR	CP 3CH
;	JP Z,DEC_CURR
SW_CURR: LD IX,0E000H	CP 3EH
LD A,0CAH	JP Z,INC_CURR
CALL LCD_COM	CP 65H
LD A,(IX+10H)	JP Z,ENT_MANR
CALL WR_ASCII	CP 10H
LD A,0CBH	JP Z,NON3
CALL LCD_COM	INC (IX+11H) ;----> INC KEY
LD A,(IX+0FH)	VALUE FOR CHECK
CALL WR_ASCII	NON3: RET
LD A,0CCH	ENT_MANR: LD A,(IX+15H)
CALL LCD_COM	CP 00H
LD A,(IX+0EH)	JP Z,AAMP
CALL WR_ASCII	CP 05H
RET	JP Z,BAMP
;	CP 10H
ESC_CURR:LD A,(IX+11H)	JP Z,CAMP
CP 00H	CP 15H
JP NZ,ENT_CUR	JP Z,DAMP
JP MAIN	CP 20H
;	JP Z,EAMP
DIS_CURR:LD HL,CHN_KEY	CP 25H

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

JP Z,CHN7	CHN11: SET 1,(IX+5)
CP 38H	JP MN
JP Z,CHN8	CHN12: SET 2,(IX+5)
CP 39H	JP MN
JP Z,CHN9	;
JP MAIN	SET_OFF: LD A,(IX+1)
TEN_CHN:LD A,(IX+0)	CP 31H
CP 30H	JP Z,TEN_CHN)
JP Z,CHN10	LD A,(IX+0)
CP 31H	CP 31H
JP Z,CHN11	JP Z,CHN11)
CP 32H	CP 32H
JP Z,CHN12	JP Z,CHN2)
JP MAIN	CP 33H
;	JP Z,CHN3)
CHN1: SET 0,(IX+4)	CP 34H
JP MN	JP Z,CHN4)
CHN2: SET 1,(IX+4)	CP 35H
JP MN	JP Z,CHN5)
CHN3: SET 2,(IX+4)	CP 36H
JP MN	JP Z,CHN6)
CHN4: SET 3,(IX+4)	CP 37H
JP MN	JP Z,CHN7)
CHN5: SET 4,(IX+4)	CP 38H
JP MN	JP Z,CHN8)
CHN6: SET 5,(IX+4)	CP 39H
JP MN	JP Z,CHN9)
CHN7: SET 6,(IX+4)	JP MAIN
JP MN	TEN_CHN1:LD A,(IX+0)
CHN8: SET 7,(IX+4)	CP 30H
JP MN	JP Z,CHN10)
CHN9: JP CURRENT	CP 31H
CHN10: SET 0,(IX+5)	JP Z,CHN11)
JP MN	CP 32H

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

JP Z,CHN121
JP MAIN
;
CHN111: RES 0,(IX+4)
JP MN
CHN21: RES 1,(IX+4)
JP MN
CHN31: RES 2,(IX+4)
JP MN
CHN41: RES 3,(IX+4)
JP MN
CHN51: RES 4,(IX+4)
JP MN
CHN61: RES 5,(IX+4)
JP MN
CHN71: RES 6,(IX+4)
JP MN
CHN81: RES 7,(IX+4)
JP MN
CHN91: CALL DELAY1
LD A,00H
LD IX,0E000H
LD (IX+18H),A
LD (IX+19H),A
OUT (PORTAA),A
CALL DELAY1
JP MN
CHN101: RES 0,(IX+5)
JP MN
CHN111: RES 1,(IX+5)
JP MN
CHN121: RES 2,(IX+5)
MN: LD (IX+23H),50H
JP MAIN
;
SHOW_OF: LD A,8AH
CALL LCD_COM
LD A,(IX+6)
CALL WR_ASCII
LD A,8BH
CALL LCD_COM
LD A,(IX+7)
CALL WR_ASCII
LD A,8CH
CALL LCD_COM
LD A,(IX+8)
CALL WR_ASCII
RET
;
ESCAPE: LD HL,MENU
CALL DIS_LCD
LD HL,SELECT
CALL DIS_LCD1
JP MAIN
;
CURRENT: LD HL,LIM_CUR
CALL DIS_LCD
LD HL,AMP1
CALL DIS_LCD1
ENT_CUR: LD IX,0E000H
LD (IX+0),30H
LD (IX+1),30H
LD (IX+2),00H
LD (IX+3),03H
LD (IX+0DH),14H
LD (IX+0EH),30H
LD (IX+0FH),2EH
LD (IX+10H),30H

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

JP Z,FAMP
CP 30H
JP Z,GAMP
CP 35H
JP Z,HAMP
CP 40H
JP Z,LAMP
CP 45H
JP Z,JAMP
CP 50H
JP Z,KAMP
JP MAIN
AAMP: LD (IX+19H),00H
JP MA
BAMP: LD (IX+19H),15H
JP MA
CAMP: LD (IX+19H),30H
JP MA
DAMP: LD (IX+19H),4BH
JP MA
EAMP: LD (IX+19H),5FH
JP MA
FAMP: LD (IX+19H),77H
JP MA
GAMP: LD (IX+19H),8FH
JP MA
HAMP: LD (IX+19H),0A0H
JP MA
IAMP: LD (IX+19H),0AFH
JP MA
JAMP: LD (IX+19H),0C4H
JP MA
KAMP: LD (IX+19H),0D5H
JP MA
MA: LD IX,0E000H
LD (IX+18H),44H
LD A,(IX+19H)
OUT (PORTAA),A
CALL DELAY1
CALL DELAY1
CALL DELAY1
JP MAIN
:
OUT_CUR1: LD (IX+10H),A
:
MAIN FOR OUT PUSH1&2
LD (IX+0FH),2EH
LD (IX+0EH),30H
LPOUT_1: CALL OUT_TEST
JP LPOUT_1
:
OUT_TEST: OUT (PORTB),A
LD A,00001100B
CALL LCD_COM
CALL SW_CURR
CALL SCANKEY
CALL CHK_KEY
LD A,E
CP 0D4H
JP Z,MAIN
RET
:
ER_CUR: LD (IX+0EH),72H
LD (IX+0FH),45H
LD (IX+10H),20H
CALL SW_CURR
CALL SD_ERR
JP ENT_CUR

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

INC_CURR: LD A,(IX+15H)	TIME: LD IX,0E000H
CP 00H	LD (IX+2),00H
JP NZ,INCUR1	LD (IX+0DH),00H
INCUR2: ADD A,05H	LD HL,SET_DAY
LD (IX+15H),A	CALL DIS_LCD
JP CURR_COM	LD HL,SET_TIME1
INCUR1: CP 50H	CALL DIS_LCD1
JP NZ,INCUR2	STE_TIME: LD A,(IX+40H)
LD (IX+15H),00H	CP 18H
JP CURR_COM	JP Z,KAME
DEC_CURR: LD A,(IX+15H)	STE1: LD B,14H
CP 00H	LD IY,0D000H
JP NZ,DECUR1	LD A,2DH
LD (IX+15H),50H	CLR_IY: LD (IY+0),A
JP CURR_COM	INC IY
DECUR1: SUB 05H	DEC B
LD (IX+15H),A	JP NZ,CLR_IY
JP CURR_COM	LD IY,0D000H
CURR_COM: LD A,(IX+15H)	LD (IX+20H),00H
DAA	LD (IY+0DH),13H
LD (IX+15H),A	LD (IY+14H),07H
LD B,A	KAME: LD IY,0D000H
AND 0FH	-----
OR 30H	: BEGIN FOR SET TIME
LD (IX+0EH),A	-----
LD A,B	CALL COUNT
AND 0F0H	SET_WEEK: LD IY,0D000H
RR A	LD A,00001101B
RR A	CALL LCD_COM
RR A	LD A,(IY+10H)
RR A	CALL DS_WEEK
OR 30H	CALL WEEK
LD (IX+10H),A	CALL T_SET
JP EN_CURR	CALL DIS_WEEK

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

JP SET_WEEK
;
SET_ATE: CALL ATE
CALL T_SET
CALL DIS_ATE
JP SET_ATE
;
SET_MON: LD A,(IY+0DH)
LD L,A
CALL COMM
CALL MONTH
CALL T_SET
CALL DIS_MON
JP SET_MON
;
SET_YEAR: CALL YEAR
CALL T_SET
CALL DIS_YEAR
JP SET_YEAR
;
SET_XML: CALL XM
CALL T_SET
CALL DIS_XM
JP SET_XML
;
SET_HOUR: CALL HOUR
CALL T_SET
CALL DIS_HOUR
JP SET_HOUR
;
SET_MIN: CALL MIN
CALL T_SET
CALL DIS_MIN
JP SET_MIN
;
SET_SEC: CALL SEC
CALL T_SET
CALL DIS_SEC
JP SET_SEC
;
DIS_TIME: CALL DIS_COMT
CALL SETCK
JP MAIN
TIME_LP: LD A,00001100B
CALL LCD_COM
CALL COMP
CALL COUNT
CALL OUTPUT
CALL CHANNALI
CALL GETCK
CALL INTRTC1S
JP TIME_LP
;
T_SET: CALL SCANKEY
CALL SOUND
CALL CHK_KEY
CALL DISCHK
RET
;
DIS_WEEK: LD HL,CHM_KEY
LD A,L
ADD A,B
LD L,A
LD A,(HL)
LD B,A
CP 49H

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

JP Z,DEC_WEEK
CP 44H
JP Z,INC_WEEK
CP 45H
JP Z,ESC_WEEK
CP 3CH
JP Z,DEC_WEEK
CP 3EH
JP Z,INC_WEEK
CP 65H
JP Z,ENT_WEEK
RET
ESC_WEEK: LD IX,0E000H
LD A,(IX+40H)
CP 18H
JP Z,ESC
JP TIME
;
ENT_WEEK: LD A,(IX+0DH)
CP 00H
JP NZ,E1_WEEK
LD A,(IY+11H)
CP 2DH
JP NZ,E1_WEEK
LD (IY+14H),00H
JP SET_WEEK
E1_WEEK: JP SET_ATE
;
DEC_WEEK: LD A,(IY+14H)
CP 00H
JP NZ,D1_WEEK
LD (IY+14H),07H
JP SET_WEEK
D1_WEEK: DEC (IY+14H)

JP SET_WEEK
;
INC_WEEK: LD A,(IY+14H)
CP 07H
JP NZ,I1_WEEK
LD (IY+14H),00H
JP SET_WEEK
I1_WEEK: INC (IY+14H)
JP SET_WEEK
;
ESC: LD A,(IX+2)
CP 00H
JP Z,ESCAPE
JP ENT_STE
DIS_ATE: LD HL,CHM_KEY
LD A,L
ADD A,B
LD L,A
LD A,(HL)
LD B,A
CP 49H
JP Z,DEC_ATE
CP 44H
JP Z,INC_ATE
CP 45H
JP Z,SET_WEEK
CP 3CH
JP Z,DEC_ATE
CP 3EH
JP Z,INC_ATE
CP 65H
JP Z,ENT_ATE
CP 10H

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

JP Z, NON_ATE
LD D, A
LD A, (IY+6)
LD (IY+6), D
LD (IY+7), A
NON_ATE: RET
;
INC_ATE: LD A, (IY+7)
CP 2DH
JP Z, T6_ATE
LD A, (IY+6)
CP 2DH
JP Z, T6_ATE
LD A, (IY+7)
CP 33H
JP NC, T3_ATE
T4_ATE: LD A, (IY+6)
CP 39H
JP NZ, T1_ATE
LD (IY+6), 30H
JP T2_ATE
T1_ATE: INC (IY+6)
JP SET_ATE
T2_ATE: INC (IY+7)
JP SET_ATE
T3_ATE: LD A, (IY+7)
CP 33H
JP NZ, T6_ATE
LD A, (IY+6)
CP 31H
JP C, T1_ATE
T6_ATE: LD (IY+6), 31H
LD (IY+7), 30H
JP SET_ATE
;
DEC_ATE: LD A, (IY+7)
CP 2DH
JP Z, T6_ATE
LD A, (IY+6)
CP 2DH
JP Z, T6_ATE
LD A, (IY+7)
CP 30H
JP NZ, D3_ATE
LD A, (IY+6)
CP 30H
JP NZ, D2_ATE
LD (IY+6), 31H
LD (IY+7), 33H
JP SET_ATE
D3_ATE: CP 33H
JP NZ, D4_ATE
LD A, (IY+6)
CP 33H
JP NC, T6_ATE
CP 30H
JP NZ, D2_ATE
D1_ATE: DEC (IY+7)
LD (IY+6), 39H
JP SET_ATE
D2_ATE: DEC (IY+6)
JP SET_ATE
D4_ATE: CP 33H
JP NC, T6_ATE
LD A, (IY+6)
CP 30H
JP Z, D1_ATE
JP D2_ATE

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

;
ENT_ATE: LD A,(IX+0DH)
CP 00H
JP NZ,E2_ATE
LD A,(IY+7) ; WHEN ENTER
ATE
CP 2DH
JP Z,T6_ATE
LD A,(IY+6)
CP 2DH
JP Z,T6_ATE
;
E2_ATE: LD A,(IY+7)
CP 2DH
JP NZ,E3_ATE
LD A,(IY+6)
CP 2DH
JP NZ,T6_ATE
E4_ATE: LD A,(IY+7)
CP 30H
JP NZ,E1_ATE
LD A,(IY+6)
CP 30H
JP NZ,E1_ATE
JP T6_ATE
E1_ATE: LD A,(IY+7)
CP 33H
JP C,SET_MON
JP NZ,T6_ATE
LD A,(IY+6)
CP 32H
JP NC,T6_ATE
JP SET_MON
E3_ATE: LD A,(IY+6)
CP 2DH
JP Z,T6_ATE
LD A,(IY+6)
CP 2DH
JP Z,T6_ATE
;
INC_MON: LD A,(IY+0DH)
CP 09H
JP Z,D1_MON
CP 10H
JP Z,D2_MON
CP 11H
JP Z,D3_MON
CP 12H
JP Z,D4_MON
CP 13H
JP Z,I3_MON
INC (IY+0DH)
JP SET_MON
;
DEC_MON: LD A,(IY+0DH)
CP 01H
JP Z,D4_MON
CP 13H
JP Z,D3_MON
CP 12H
JP Z,D2_MON
CP 11H
JP Z,D1_MON
CP 10H
JP Z,D5_MON
DEC (IY+0DH)
JP SET_MON
;
D5_MON: LD (IY+0DH),09H
JP SET_MON

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่นอญูญาติให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

JP SET_WEEK
;
SET_ATE: CALL ATE
CALL T_SET
CALL DIS_ATE
JP SET_ATE
;
SET_MON: LD A,(IY+0DH)
LD L,A
CALL COMM
CALL MONTH
CALL T_SET
CALL DIS_MON
JP SET_MON
;
SET_YEAR: CALL YEAR
CALL T_SET
CALL DIS_YEAR
JP SET_YEAR
;
SET_XML: CALL XM
CALL T_SET
CALL DIS_XML
JP SET_XML
;
SET_HOUR: CALL HOUR
CALL T_SET
CALL DIS_HOUR
JP SET_HOUR
;
SET_MIN: CALL MIN
CALL T_SET
CALL DIS_MIN
JP SET_MIN
;
SET_SEC: CALL SEC
CALL T_SET
CALL DIS_SEC
JP SET_SEC
;
DIS_TIME: CALL DIS_COMT
CALL SETCK
JP MAIN
TIME_LP: LD A,00001100B
CALL LCD_COM
CALL COMP
CALL COUNT
CALL OUTPUT
CALL CHANNAL1
CALL GETCK
CALL INTRCIS
JP TIME_LP
;
T_SET: CALL SCANKEY
CALL SOUND
CALL CHK_KEY
CALL DISCHK
RET
;
DIS_WEEK: LD HL,CHM_KEY
LD A,L
ADD A,B
LD L,A
LD A,(HL)
LD B,A
CP 49H

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

JP Z,DEC_WEEK
CP 44H
JP Z,INC_WEEK
CP 45H
JP Z,ESC_WEEK
CP 3CH
JP Z,DEC_WEEK
CP 3EH
JP Z,INC_WEEK
CP 65H
JP Z,ENT_WEEK
RET
ESC_WEEK: LD IX,0E000H
LD A,(IX+40H)
CP 18H
JP Z,ESC
JP TIME
;
ENT_WEEK: LD A,(IX+0DH)
CP 00H
JP NZ,E1_WEEK
LD A,(IY+11H)
CP 2DH
JP NZ,E1_WEEK
LD (IY+14H),00H
JP SET_WEEK
E1_WEEK: JP SET_ATE
;
DEC_WEEK: LD A,(IY+14H)
CP 00H
JP NZ,D1_WEEK
LD (IY+14H),07H
JP SET_WEEK
D1_WEEK: DEC (IY+14H)
JP SET_WEEK
;
INC_WEEK: LD A,(IY+14H)
CP 07H
JP NZ,I1_WEEK
LD (IY+14H),00H
JP SET_WEEK
I1_WEEK: INC (IY+14H)
JP SET_WEEK
;
ESC: LD A,(IX+2)
CP 00H
JP Z,ESCAPE
JP ENT_STE
;
DIS_ATE: LD HL,CHM_KEY
LD A,L
ADD A,B
LD L,A
LD A,(HL)
LD B,A
CP 49H
JP Z,DEC_ATE
CP 44H
JP Z,INC_ATE
CP 45H
JP Z,SET_WEEK
CP 3CH
JP Z,DEC_ATE
CP 3EH
JP Z,INC_ATE
CP 65H
JP Z,ENT_ATE
CP 10H

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

JP Z, NON_ATE	;
LD D, A	DEC_ATE: LD A, (IY+7)
LD A, (IY+6)	CP 2DH
LD (IY+6), D	JP Z, T6_ATE
LD (IY+7), A	LD A, (IY+6)
NON_ATE: RET	CP 2DH
;	JP Z, T6_ATE
INC_ATE: LD A, (IY+7)	LD A, (IY+7)
CP 2DH	CP 30H
JP Z, T6_ATE	JP NZ, D3_ATE
LD A, (IY+6)	LD A, (IY+6)
CP 2DH	CP 30H
JP Z, T6_ATE	JP NZ, D2_ATE
LD A, (IY+7)	LD (IY+6), 31H
CP 33H	LD (IY+7), 33H
JP NC, T3_ATE	JP SET_ATE
T4_ATE: LD A, (IY+6)	D3_ATE: CP 33H
CP 39H	JP NZ, D4_ATE
JP NZ, T1_ATE	LD A, (IY+6)
LD (IY+6), 30H	CP 33H
JP T2_ATE	JP NC, T6_ATE
T1_ATE: INC (IY+6)	CP 30H
JP SET_ATE	JP NZ, D2_ATE
T2_ATE: INC (IY+7)	D1_ATE: DEC (IY+7)
JP SET_ATE	LD (IY+6), 39H
T3_ATE: LD A, (IY+7)	JP SET_ATE
CP 33H	D2_ATE: DEC (IY+6)
JP NZ, T6_ATE	JP SET_ATE
LD A, (IY+6)	D4_ATE: CP 33H
CP 31H	JP NC, T6_ATE
JP C, T1_ATE	LD A, (IY+6)
T6_ATE: LD (IY+6), 31H	CP 30H
LD (IY+7), 30H	JP Z, D1_ATE
JP SET_ATE	JP D2_ATE

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

;
ENT_ATE: LD A,(IX+0DH)
CP 00H
JP NZ,E2_ATE
LD A,(IY+7) ; WHEN ENTER
ATE
CP 2DH
JP Z,T6_ATE
LD A,(IY+6)
CP 2DH
JP Z,T6_ATE
;
E2_ATE: LD A,(IY+7)
CP 2DH
JP NZ,E3_ATE
LD A,(IY+6)
CP 2DH
JP NZ,T6_ATE
E4_ATE: LD A,(IY+7)
CP 30H
JP NZ,E1_ATE
LD A,(IY+6)
CP 30H
JP NZ,E1_ATE
JP T6_ATE
E1_ATE: LD A,(IY+7)
CP 33H
JP C,SET_MON
JP NZ,T6_ATE
LD A,(IY+6)
CP 32H
JP NC,T6_ATE
JP SET_MON
E3_ATE: LD A,(IY+6)
;
CP 2DH
JP Z,T6_ATE
JP E4_ATE
;-----
INC_MON: LD A,(IY+0DH)
CP 09H
JP Z,D1_MON
CP 10H
JP Z,D2_MON
CP 11H
JP Z,D3_MON
CP 12H
JP Z,D4_MON
CP 13H
JP Z,I3_MON
INC (IY+0DH)
JP SET_MON
;
DEC_MON: LD A,(IY+0DH)
CP 01H
JP Z,D4_MON
CP 13H
JP Z,D3_MON
CP 12H
JP Z,D2_MON
CP 11H
JP Z,D1_MON
CP 10H
JP Z,D5_MON
DEC (IY+0DH)
JP SET_MON
;
D5_MON: LD (IY+0DH),09H
JP SET_MON

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

D4_MON: LD (IY+0DH),13H	CP 13H
JP SET_MON	JP NZ,SET_YEAR
D1_MON: LD (IY+0DH),10H	LD (IY+0DH),01H
JP SET_MON	JP SET_MON
D2_MON: LD (IY+0DH),11H	;
JP SET_MON	DIS_YEAR: LD HL,CHM_KEY
D3_MON: LD (IY+0DH),12H	LD A,L
JP SET_MON	ADD A,B
I3_MON: LD (IY+0DH),01H	LD L,A
JP SET_MON	LD A,(HL)
;	LD B,A
DIS_MON: LD HL,CHM_KEY	CP 49H
LD A,L	JP Z,DEC_YEAR
ADD A,B	CP 44H
LD L,A	JP Z,INC_YEAR
LD A,(HL)	CP 45H
LD B,A	JP Z,SET_MON
CP 49H	CP 3CH
JP Z,DEC_MON	JP Z,DEC_YEAR
CP 44H	CP 3EH
JP Z,INC_MON	JP Z,INC_YEAR
CP 45H	CP 65H
JP Z,SET_MON	JP Z,ENT_YEAR
CP 3CH	CP 10H
JP Z,DEC_MON	JP Z,NON_YEAR
CP 3EH	LD D,A
JP Z,INC_MON	LD A,(IY+0BH)
CP 65H	LD (IY+0BH),D
JP Z,ENT_MON	LD (IY+0CH),A
RET	NON_YEAR: RET
ENT_MON: LD A,(IX+0DH)	;
CP 00H	YEAR_XML: LD A,(IY+0CH)
JP NZ,SET_YEAR	CP 2DH
LD A,(IY+0DH)	JP NZ,EA_YEAR

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่นอญญาติให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

LD A,(IY+0BH)	JP SET_YEAR
CP 2DH	I1_YEAR: INC (IY+0BH)
JP Z,SET_XML	JP SET_YEAR
JP EE_YEAR	I2_YEAR: LD A,(IY+0BH)
EA_YEAR: LD A,(IY+0BH)	CP 39H
CP 2DH	JP NZ,I1_YEAR
JP Z,EE_YEAR	LD (IY+0BH),30H
JP SET_XML	INC (IY+0CH)
;	JP SET_YEAR
ENT_YEAR: LD A,(IX+0DH)	;
CP 00H	DEC_YEAR: LD A,(IY+0CH)
JP NZ,YEAR_XML	CP 2DH
LD A,(IY+0BH)	JP Z,EE_YEAR
CP 2DH	LD A,(IY+0BH)
JP Z,EE_YEAR	CP 2DH
LD A,(IY+0CH)	JP Z,EE_YEAR
CP 2DH	LD A,(IY+0CH)
JP Z,EE_YEAR	CP 30H
JP SET_XML	JP NZ,D2_YEAR
;	LD A,(IY+0BH)
INC_YEAR: LD A,(IY+0CH)	CP 30H
CP 2DH	JP NZ,D1_YEAR
JP Z,EE_YEAR	LD (IY+0CH),39H
LD A,(IY+0BH)	LD (IY+0BH),39H
CP 2DH	JP SET_YEAR
JP Z,EE_YEAR	D1_YEAR: DEC (IY+0BH)
LD A,(IY+0CH)	JP SET_YEAR
CP 39H	D2_YEAR: LD A,(IY+0BH)
JP NZ,I2_YEAR	CP 30H
LD A,(IY+0BH)	JP NZ,D1_YEAR
CP 39H	LD (IY+0BH),39H
JP NZ,I1_YEAR	DEC (IY+0CH)
EE_YEAR: LD (IY+0CH),30H	JP SET_YEAR
LD (IY+0BH),30H	

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

;
DIS_XM: LD HL,CHM_KEY
LD A,L
ADD A,B
LD L,A
LD A,(HL)
LD B,A
CP 49H
JP Z,DEC_XM
CP 44H
JP Z,DEC_XM
CP 45H
JP Z,SET_YEAR
CP 3CH
JP Z,DEC_XM
CP 3EH
JP Z,DEC_XM
CP 65H
JP Z,ENT_XM
RET
DEC_XM: LD A,(IY+0EH)
CP 2DH
JP Z,XM4
CP 50H
JP Z,XM3
CP 41H
JP Z,XM1
XM4: LD A,41H
XM2: LD (IY+0EH),A
LD (IY+0FH),6DH
JP SET_XML
XM1: LD A,50H
JP XM2
XM3: LD (IY+0EH),2DH
LD (IY+0FH),2DH
JP SET_XML
ENT_XM: LD A,(IX+0DH)
CP 00H
JP NZ,SET_HOUR
LD A,(IY+0EH)
CP 2DH
JP Z,XM4
JP SET_HOUR
;
DIS_HOUR: LD HL,CHM_KEY
LD A,L
ADD A,B
LD L,A
LD A,(HL)
LD B,A
CP 49H
JP Z,DEC_HOUR
CP 44H
JP Z,INC_HOUR
CP 45H
JP Z,SET_XML
CP 3CH
JP Z,DEC_HOUR
CP 3EH
JP Z,INC_HOUR
CP 65H
JP Z,ENT_HOUR
CP 10H
JP Z,NON_HOUR
LD D,A
LD A,(IY+4)
LD (IY+4),D
LD (IY+5),A

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

NON_HOUR: RET	LD A,(IY+5)
;	CP 32H
INC_HOUR: LD A,(IY+5)	JP C,D1_HOUR
CP 2DH	D2_HOUR: LD (IY+5),31H
JP Z,I4_HOUR	LD (IY+4),32H
LD A,(IY+4)	JP SET_HOUR
CP 2DH	D1_HOUR: CP 30H
JP Z,I4_HOUR	JP NZ,D3_HOUR
LD A,(IY+5)	LD A,(IY+4)
CP 32H	CP 31H
JP C,I1_HOUR	JP Z,D2_HOUR
I4_HOUR: LD (IY+5),30H	D4_HOUR: DEC (IY+4)
LD (IY+4),31H	JP SET_HOUR
JP SET_HOUR	D3_HOUR: LD A,(IY+4)
I1_HOUR: CP 31H	CP 33H
JP NZ,I3_HOUR	JP NC,D2_HOUR
LD A,(IY+4)	CP 30H
CP 32H	JP NZ,D4_HOUR
JP Z,I4_HOUR	DEC (IY+5)
I3_HOUR: LD A,(IY+4)	LD (IY+4),39H
CP 39H	JP SET_HOUR
JP NZ,I2_HOUR	;
LD (IY+4),30H	ENT_HOUR: LD A,(IX+0DH)
INC (IY+5)	CP 00H
JP SET_HOUR	JP NZ,E1_HOUR
I2_HOUR: INC (IY+4)	LD A,(IY+5)
JP SET_HOUR	CP 2DH
;	JP Z,I4_HOUR
DEC_HOUR: LD A,(IY+5)	LD A,(IY+4)
CP 2DH	CP 2DH
JP Z,I4_HOUR	JP Z,I4_HOUR
LD A,(IY+4)	E1_HOUR: LD A,(IY+5)
CP 2DH	CP 2DH
JP Z,I4_HOUR	JP NZ,E4_HOUR

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

LD A,(IY+4)	JP Z,SET_HOUR
CP 2DH	CP 3CH
JP NZ,I4_HOUR	JP Z,DEC_MIN
E3_HOUR: LD A,(IY+5)	CP 3EH
CP 30H	JP Z,INC_MIN
JP NZ,E2_HOUR	CP 65H
LD A,(IY+4)	JP Z,ENT_MIN
CP 30H	CP 10H
JP Z,I4_HOUR	JP Z,NON_MIN
E2_HOUR: LD A,(IY+5)	LD D,A
CP 32H	LD A,(IY+2)
JP NC,I4_HOUR	LD (IY+2),D
CP 31H	LD (IY+3),A
JP NZ,SET_MIN	NON_MIN: RET
LD A,(IY+4)	:
CP 33H	INC_MIN: LD A,(IY+3)
JP NC,I4_HOUR	CP 2DH
JP SET_MIN	JP Z,I2_MIN
E4_HOUR: LD A,(IY+4)	LD A,(IY+2)
CP 2DH	CP 2DH
JP Z,I4_HOUR	JP Z,I2_MIN
JP E3_HOUR	LD A,(IY+3)
;	CP 35H
DIS_MIN: LD HL,CHM_KEY	JP NZ,I1_MIN
LD A,L	LD A,(IY+2)
ADD A,B	CP 39H
LD L,A	JP NZ,I4_MIN
LD A,(HL)	I2_MIN: LD (IY+3),30H
LD B,A	LD (IY+2),30H
CP 49H	JP SET_MIN
JP Z,DEC_MIN	I1_MIN: CP 36H
CP 44H	JP NC,I2_MIN
JP Z,INC_MIN	I4_MIN: LD A,(IY+2)
CP 45H	CP 39H

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

JP NZ,I3_MIN	CP 00H
LD (IY+2),30H	JP NZ,E1_MIN
INC (IY+3)	LD A,(IY+3)
JP SET_MIN	CP 2DH
I3_MIN: INC (IY+2)	JP Z,I2_MIN
JP SET_MIN	LD A,(IY+2)
;	CP 2DH
DEC_MIN: LD A,(IY+3)	JP Z,I2_MIN
CP 2DH	E1_MIN: LD A,(IY+3)
JP Z,I2_MIN	CP 2DH
LD A,(IY+2)	JP NZ,E3_MIN
CP 2DH	LD A,(IY+2)
JP Z,I2_MIN	CP 2DH
LD A,(IY+3)	JP NZ,I2_MIN
CP 30H	E2_MIN: LD A,(IY+3)
JP NZ,D1_MIN	CP 36H
LD A,(IY+2)	JP NC,I2_MIN
CP 30H	JP SET_SEC
JP NZ,D2_MIN	E3_MIN: LD A,(IY+2)
LD (IY+3),35H	CP 2DH
LD (IY+2),39H	JP Z,I2_MIN
JP SET_MIN	JP E2_MIN
D1_MIN: CP 36H	;
JP NC,I2_MIN	DIS_SEC: LD HL,CHM_KEY
D2_MIN: LD A,(IY+2)	LD A,L
CP 30H	ADD A,B
JP NZ,D3_MIN	LD L,A
LD (IY+2),39H	LD A,(HL)
DEC (IY+3)	LD B,A
JP SET_MIN	CP 49H
D3_MIN: DEC (IY+2)	JP Z,DEC_SEC
JP SET_MIN	CP 44H
;	JP Z,INC_SEC
ENT_MIN: LD A,(IX+0DH)	CP 45H

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

JP Z,SET_MIN	JP NZ,I3_SEC
CP 3CH	LD (IY+0),30H
JP Z,DEC_SEC	INC (IY+1)
CP 3EH	JP SET_SEC
JP Z,INC_SEC	I3_SEC: INC (IY+0)
CP 65H	JP SET_SEC
JP Z,ENT_SEC	:
CP 10H	DEC_SEC: LD A,(IY+1)
JP Z,NON_SEC	CP 2DH
LD D,A	JP Z,I2_SEC
LD A,(IY+0)	LD A,(IY+0)
LD (IY+0),D	CP 2DH
LD (IY+1),A	JP Z,I2_SEC
NON_SEC: RET	LD A,(IY+1)
:	CP 30H
INC_SEC: LD A,(IY+1)	JP NZ,D1_SEC
CP 2DH	LD A,(IY+0)
JP Z,I2_SEC	CP 30H
LD A,(IY+0)	JP NZ,D2_SEC
CP 2DH	LD (IY+1),35H
JP Z,I2_SEC	LD (IY+0),39H
LD A,(IY+1)	JP SET_SEC
CP 35H	D1_SEC: CP 36H
JP NZ,I1_SEC	JP NC,I2_SEC
LD A,(IY+0)	D2_SEC: LD A,(IY+0)
CP 39H	CP 30H
JP NZ,I4_SEC	JP NZ,D3_SEC
I2_SEC: LD (IY+1),30H	LD (IY+0),39H
LD (IY+0),30H	DEC (IY+1)
JP SET_SEC	JP SET_SEC
I1_SEC: CP 36H	D3_SEC: DEC (IY+0)
JP NC,I2_SEC	JP SET_SEC
I4_SEC: LD A,(IY+0)	:
CP 39H	ENT_SEC: LD IX,0E000H

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

LD (IX+40H),18H
LD A,(IX+0DH)
CP 00H
JP NZ,E1_SEC
LD A,(IY+1)
CP 2DH
JP Z,I2_SEC
LD A,(IY+0)
CP 2DH
JP Z,I2_SEC
E1_SEC: LD A,(IY+1)
CP 2DH
JP NZ,E3_SEC
LD A,(IY+0)
CP 2DH
JP NZ,I2_SEC
E2_SEC: LD A,(IY+1)
CP 36H
JP NC,I2_SEC
=====
TIMER
LD IX,0E000H
LD A,(IX+0DH)
CP 00H
JP Z,DIS_TIME
JP CONVERT
E3_SEC: LD A,(IY+0)
CP 2DH
JP Z,I2_SEC
JP E2_SEC
;
=====
; CHECK CHANNAL FOR SELECT TIME
ON/OFF OUTPUT

CHECK_CHN: LD IX,0E000H
LD A,(IX+1)
CP 31H
JP Z,MORE_CHN
LD A,(IX+0)
CP 31H
JP Z,MOVE_T1
CP 32H
JP Z,MOVE_T2
CP 33H
JP Z,MOVE_T3
CP 34H
JP Z,MOVE_T4
CP 35H
JP Z,MOVE_T5
CP 36H
JP Z,MOVE_T6
CP 37H
JP Z,MOVE_T7
CP 38H
JP Z,MOVE_T8
CP 39H
JP Z,MOVE_T9
JP MAIN
;
MORE_CHN: LD A,(IX+0)
CP 30H
JP Z,MOVE_T10
CP 31H
JP Z,MOVE_T11
CP 32H
JP Z,MOVE_T12
JP MAIN

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

; MOVE TIME VALUE FROM LCD BRING TO REGISTER IY FOR CHECK TIME
;-----
:STATE 1-2
:
MOVE_T1: LD IX,0E000H
        LD A,(IX+9)
        CP 32H
        JP Z,MT12_ON
        LD A,(IX+14H)
        CP 20H
        JP NZ,MT1_OFF
        LD IX,0D100H
;CHANNAL1---ON
        CALL MOVE_RAM
        JP ENT_STE
        MT1_OFF: LD IX,0D10DH
;CHANNAL1---OFF
        CALL MOVE_RAM
        JP ENT_STE
;
        MT12_ON: LD A,(IX+14H)
        CP 20H
        JP NZ,MT12_OFF
        LD IX,0D11AH
;CHANNAL1---ON
        CALL MOVE_RAM
        JP ENT_STE
        MT12_OFF: LD IX,0D127H
;CHANNAL1---OFF
        CALL MOVE_RAM
        JP ENT_STE
;-----
MOVE_T2: LD IX,0E000H
        LD A,(IX+9)
        CP 32H
        JP Z,MT22_ON
        LD A,(IX+14H)
        CP 20H
        JP NZ,MT2_OFF
        LD IX,0D134H
;CHANNAL2---ON
        CALL MOVE_RAM
        JP ENT_STE
        MT2_OFF: LD IX,0D141H
;CHANNAL2---OFF
        CALL MOVE_RAM
        JP ENT_STE
;
        MT22_ON: LD A,(IX+14H)
        CP 20H
        JP NZ,MT22_OFF
        LD IX,0D14EH
;CHANNAL2---ON
        CALL MOVE_RAM
        JP ENT_STE
        MT22_OFF: LD IX,0D15BH
;CHANNAL2---OFF
        CALL MOVE_RAM
        JP ENT_STE
;-----
MOVE_T3: LD IX,0E000H
        LD A,(IX+9)
        CP 32H
        JP Z,MT32_ON
        LD A,(IX+14H)
        CP 20H
        JP NZ,MT3_OFF

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

LD IX,0D168H
;CHANNAL3---ON
CALL MOVE_RAM
JP ENT_STE
MT3_OFF: LD IX,0D175H
;CHANNAL3---OFF
CALL MOVE_RAM
JP ENT_STE
;
MT32_ON: LD A,(IX+14H)
CP 20H
JP NZ,MT32_OFF
LD IX,0D182H
;CHANNAL3---ON
CALL MOVE_RAM
JP ENT_STE
MT32_OFF: LD IX,0D18FH
;CHANNAL3---OFF
CALL MOVE_RAM
JP ENT_STE
;-----
MOVE_T4: LD IX,0E000H
LD A,(IX+9)
CP 32H
JP Z,MT42_ON
LD A,(IX+14H)
CP 20H
JP NZ,MT4_OFF
LD IX,0D19CH
;CHANNAL4---ON
CALL MOVE_RAM
JP ENT_STE
MT4_OFF: LD IX,0D1A9H
;CHANNAL4---OFF
CALL MOVE_RAM
JP ENT_STE
;
MT42_ON: LD A,(IX+14H)
CP 20H
JP NZ,MT42_OFF
;CHANNAL4---OFF
CALL MOVE_RAM
JP ENT_STE
;-----
MOVE_T5: LD IX,0E000H
LD A,(IX+9)
CP 32H
JP Z,MT52_ON
LD A,(IX+14H)
CP 20H
JP NZ,MT5_OFF
LD IX,0D1D0H
;CHANNAL5---ON
CALL MOVE_RAM
JP ENT_STE
MT5_OFF: LD IX,0D1DDH
;CHANNAL5---OFF
CALL MOVE_RAM
JP ENT_STE
;
MT52_ON: LD A,(IX+14H)
CP 20H
JP NZ,MT52_OFF

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

LD IX,0D1EAH
;CHANNAL5---ON
CALL MOVE_RAM
JP ENT_STE
MT52_OFF: LD IX,0D1F7H
;CHANNAL5---OFF
CALL MOVE_RAM
JP ENT_STE
;-----
MOVE_T6: LD IX,0E000H
LD A,(IX+9)
CP 32H
JP Z,MT62_ON
LD A,(IX+14H)
CP 20H
JP NZ,MT6_OFF
LD IX,0D204H
;CHANNAL6---ON
CALL MOVE_RAM
JP ENT_STE
MT6_OFF: LD IX,0D211H
;CHANNAL6---OFF
CALL MOVE_RAM
JP ENT_STE
;
MT62_ON: LD A,(IX+14H)
CP 20H
JP NZ,MT62_OFF
LD IX,0D21EH
;CHANNAL6---ON
CALL MOVE_RAM
JP ENT_STE
MT62_OFF: LD IX,0D22BH
;CHANNAL6---OFF
CALL MOVE_RAM
JP ENT_STE
;-----
CALL MOVE_RAM
JP ENT_STE
;-----
MOVE_T7: LD IX,0E000H
LD A,(IX+9)
CP 32H
JP Z,MT72_ON
LD A,(IX+14H)
CP 20H
JP NZ,MT7_OFF
LD IX,0D238H
;CHANNAL7---ON
CALL MOVE_RAM
JP ENT_STE
MT7_OFF: LD IX,0D245H
;CHANNAL7---OFF
CALL MOVE_RAM
JP ENT_STE
;
MT72_ON: LD A,(IX+14H)
CP 20H
JP NZ,MT72_OFF
LD IX,0D252H
;CHANNAL7---ON
CALL MOVE_RAM
JP ENT_STE
;-----
MT72_OFF: LD IX,0D25FH
;CHANNAL7---OFF
CALL MOVE_RAM
JP ENT_STE
;-----
MOVE_T8: LD IX,0E000H
LD A,(IX+9)
CP 32H

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

JP Z,MT12_ON
LD A,(IX+14H)
CP 20H
JP NZ,MT8_OFF
LD IX,0D26CH
;CHANNAL8---ON
CALL MOVE_RAM
JP ENT_STE
MT8_OFF: LD IX,0D279H
;CHANNAL8---OFF
CALL MOVE_RAM
JP ENT_STE
;
MT82_ON: LD A,(IX+14H)
CP 20H
JP NZ,MT82_OFF
LD IX,0D286H
;CHANNAL8---ON
CALL MOVE_RAM
JP ENT_STE
MT82_OFF: LD IX,0D293H
;CHANNAL8---OFF
CALL MOVE_RAM
JP ENT_STE
;-----
MOVE_T9: LD IX,0E000H
LD A,(IX+9)
CP 32H
JP Z,MT92_ON
LD A,(IX+14H)
CP 20H
JP NZ,MT9_OFF
LD IX,0D2A0H
;CHANNAL9---ON
CALL MOVE_RAM
JP ENT_STE
MT9_OFF: LD IX,0D2ADH
;CHANNAL9---OFF
CALL MOVE_RAM
JP ENT_STE
;
MT92_ON: LD A,(IX+14H)
CP 20H
JP NZ,MT92_OFF
LD IX,0D2BAH
;CHANNAL9---ON
CALL MOVE_RAM
JP ENT_STE
MT92_OFF: LD IX,0D2C7H
;CHANNAL9---OFF
CALL MOVE_RAM
JP ENT_STE
;-----
MOVE_T10: LD IX,0E000H
LD A,(IX+9)
CP 32H
JP Z,MT102_ON
LD A,(IX+14H)
CP 20H
JP NZ,MT10_OFF
LD IX,0D2D4H
;CHANNAL10---ON
CALL MOVE_RAM
JP ENT_STE
MT10_OFF: LD IX,0D2E1H
;CHANNAL10---OFF
CALL MOVE_RAM
JP ENT_STE

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

; CALL MOVE_RAM
MT102_ON: LD A,(IX+14H) JP ENT_STE
CP 20H MT112_OFF: LD IX,0D32FH
JP NZ,MT102_OFF ;CHANNAL11---OFF
LD IX,0D2EEH CALL MOVE_RAM
;CHANNAL10---ON JP ENT_STE
CALL MOVE_RAM ;-----
JP ENT_STE MOVE_T12: LD IX,0E000H
MT102_OFF: LD IX,0D2FBH LD A,(IX+9)
;CHANNAL10---OFF CP 32H
CALL MOVE_RAM JP Z,MT122_ON
JP ENT_STE LD A,(IX+14H)
;----- CP 20H
MOVE_T11: LD IX,0E000H JP NZ,MT_OFF
LD A,(IX+9) LD IX,0D33CH
CP 32H ;CHANNAL12---ON
JP Z,MT112_ON CALL MOVE_RAM
LD A,(IX+14H) JP ENT_STE
CP 20H MT_OFF: LD IX,0D349H
JP NZ,MT11_OFF ;CHANNAL12---OFF
LD IX,0D308H CALL MOVE_RAM
;CHANNAL11---ON JP ENT_STE
CALL MOVE_RAM ;
JP ENT_STE MT122_ON: LD A,(IX+14H)
MT11_OFF: LD IX,0D315H CP 20H
;CHANNAL11---OFF JP NZ,MT122_OFF
CALL MOVE_RAM LD IX,0D356H
JP ENT_STE ;CHANNAL12---ON
; CALL MOVE_RAM
; JP ENT_STE
MT112_ON: LD A,(IX+14H)
CP 20H MT122_OFF: LD IX,0D363H
JP NZ,MT112_OFF ;CHANNAL12---OFF
LD IX,0D322H CALL MOVE_RAM
;CHANNAL11---ON LD A,88H

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

LD (0D370H),A	OR 0F0H
JP ENT_STE	LD (IX+4),A

MOVE_RAM: LD B,0DH	LD A,(IY+0EH)
LD IY,0D020H	CP 41H
MOVE_LP: LD A,(IY+0)	JP Z,CONAM
LD (IX+0),A	LD A,(IY+5)
INC IY	OR 0F4H
INC IX	CONAML: LD (IX+5),A
DEC B	LD A,(IY+6) ;ATE1
JP NZ,MOVE_LP	OR 0F0H
RET	LD (IX+6),A
CONAM: LD IY,0D000H	LD A,(IY+7) ;ATE10
LD A,(IY+5)	OR 0F0H
AND 0F3H	LD (IX+7),A
OR 0F0H	LD A,(IY+0DH)
JP CONAML	CP 13H
;	JP NZ,M2_LP
CONVERT: LD IX,0E020H	LD A,0DDH
LD IY,0D000H	M2_LP: OR 0F0H
LD IX,0E020H	LD (IX+8),A ;MON1
LD A,(IY+0) ;SEC1	LD A,(IY+0DH)
OR 0F0H	CP 13H
LD (IX+0),A	JP NZ,M1_LP
LD A,(IY+1) ;SEC10	LD A,0DDH
OR 0F0H	M1_LP: AND 0F0H
LD (IX+1),A	RL A
LD A,(IY+2) ;MIN1	RL A
OR 0F0H	RL A
LD (IX+2),A	RL A
LD A,(IY+3) ;MIN10	OR 0F0H
OR 0F0H	LD (IX+9),A ;MON10
LD (IX+3),A	LD A,(IY+0BH) ;YEAR1
LD A,(IY+4) ;HOUR1	OR 0F0H

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

LD (IX+0AH),A
LD A,(IY+0CH) ;YEAR10
OR 0F0H
LD (IX+0BH),A
LD A,(IY+14H) ;WEEK
CP 07H
JP NZ,W1_LP
LD A,0DDH
W1_LP: OR 0F0H
LD (IX+0CH),A
LD IX,0E020H
LD IY,0D020H
LD B,0DH
LPCON: LD A,(IX+0)
LD (IY+0),A
INC IX
INC IY
DEC B
JP NZ,LPCON
JP CHECK_CHN
PROGRAM: LD IX,0E000H ; on
(IX+0DH) -->01H
LD (IX+0DH),01H
LD (IX+0),30H
LD (IX+1),30H
LD (IX+2),01H
LD (IX+3),04H
LD (IX+9),31H
LD HL,PROG
CALL DIS_LCD
LD HL,PGM
CALL DIS_LCD1
JP LP_MAN
PGM_CHN: LD A,(IX+3)
CP 04H
JP NZ,MAIN
LD A,(IX+1)
CP 32H
JP C,PG_NX1
CALL ERR
JP PROGRAM
;
PG_NX1: CP 31H
JP NZ,PG_NX2
LD A,(IX+0)
CP 33H
JP C,STE_LOOP1
CALL ERR
JP PROGRAM
PG_NX2: LD A,(IX+0)
CP 30H
JP Z,PROGRAM
STE_LOOP1: LD HL,TIME1
CALL DIS_LCD
CALL DIS_PGM1
LD HL,STATE
CALL DIS_LCD1
;
E_LOOP: LD A,00001101B
CALL LCD_COM
CALL STATE_PGM
CALL T_SET
CALL DIS_STE
JP E_LOOP
;
STATE_PGM: LD IX,0E000H
LD A,0C8H
CALL LCD_COM

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

LD A,(IX+9)
CALL WR_ASCII
RET
;
DIS_STE: LD HL,CHM_KEY
LD A,L
ADD A,B
LD L,A
LD A,(HL)
LD B,A
CP 49H
JP Z,DEC_STE
CP 44H
JP Z,DEC_STE
CP 45H
JP Z,PROGRAM
CP 3CH
JP Z,DEC_STE
CP 3EH
JP Z,DEC_STE
CP 65H
JP Z,ENT_STE
RET
;-----GO TO SET TIME
DEC_STE: LD A,(IX+9)
CP 31H
JP NZ,D1_STE
LD (IX+9),32H
JP STE_LOOP
D1_STE: LD (IX+9),31H
JP STE_LOOP
;-----SELECT ON OR
OFF CHANNAL
ENT_STE: LD IX,0E000H
LD HL,TIME1
CALL DIS_LCD
LD HL,STATE
CALL DIS_LCD1
CALL STATE_PGM
CALL DIS_PGM1
LD (IX-12H),4FH
LD (IX+13H),6EH
LD (IX+14H),20H
LD (IX+0DH),01H
ET_ST: CALL ET_ONOFF
CALL T_SET
CALL DISET
JP ET_ST
DISET: LD HL,CHM_KEY
LD A,L
ADD A,B
LD L,A
LD A,(HL)
LD B,A
CP 49H
JP Z,DEC_STA
CP 44H
JP Z,DEC_STA
CP 45H
JP Z,STE_LOOP
CP 3CH
JP Z,DEC_STA
CP 3EH
JP Z,DEC_STA
CP 65H
JP Z,CHK_ONOFF
CHK_ONOFF: LD IX,0E000H

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

LD (IX+0DH),01H	LD A,0C2H
LD A,(IX+14H)	CALL LCD_COM
CP 20H	LD A,(IX+14H)
JP NZ,CHECK_OFF	CALL WR_ASCII
LD HL,CH_TIME	RET
CALL DIS_LCD	;
LD HL,ON_TIME	DEC_STA: LD A,(IX+14H)
CALL DIS_LCD1	CP 20H
CALL SHW_CHN	JP NZ,DEC_STA1
JP STE1	LD (IX+13H),66H
CHECK_OFF: LD HL,CH_TIME	LD (IX+14H),66H
CALL DIS_LCD	JP ET_ST
LD HL,OFF_TIME	DEC_STA1: JP ENT_STE
CALL DIS_LCD1	;
CALL SHW_CHN	ET_ONOFF: LD IX,0E000H
JP STE1	LD A,0CCH
SHW_CHN: LD IX,0E000H	CALL LCD_COM
LD A,80H	LD A,(IX+12H)
CALL LCD_COM	CALL WR_ASCII
LD A,(IX+1)	LD A,0CDH
CALL WR_ASCII	CALL LCD_COM
LD A,81H	LD A,(IX+13H)
CALL LCD_COM	CALL WR_ASCII
LD A,(IX+0)	LD A,0CEH
CALL WR_ASCII	CALL LCD_COM
LD A,0C0H	LD A,(IX+14H)
CALL LCD_COM	CALL WR_ASCII
LD A,(IX+12H)	RET
CALL WR_ASCII	;
LD A,0C1H	DIS_PGM1: LD A,8AH
CALL LCD_COM	CALL LCD_COM
LD A,(IX+13H)	LD A,(IX+1)
CALL WR_ASCII	CALL WR_ASCII
LD A,8BH	LD A,8BH

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

CALL LCD_COM	CALL RLP1
LD A,(IX+0)	LD B,A
CALL WR_ASCII	LD A,(IY+2)
RET	AND 0FH
;	OR B
; CONVERSE TIME VALUE FORM LCD	LD C,A
;	;
DIS_COMT: LD A,(IY+7) ;ATE	LD A,(IY+1) ;SEC
AND 0FH	AND 0FH
CALL RLP1	CALL RLP1
LD B,A	LD B,A
LD A,(IY+6)	LD A,(IY+0)
AND 0FH	AND 0FH
OR B	OR B
LD H,A	LD D,A
;	;
LD A,(IY+0DH) ;MONTH	LD A,(IY+5) ;HOUR
LD L,A	AND 0FH
;	CALL RLP1
LD A,(IY+14H) ;WEEK	LD B,A
OR 0F0H	LD A,(IY+4)
OUT (WEEK1),A	AND 0FH
;	OR B
LD A,(IY+0CH) ;YEAR	LD B,A
AND 0FH	;
CALL RLP1	LD A,(IY+0EH)
LD B,A	CP 50H
LD A,(IY+0BH)	JP NZ,AM
AND 0FH	LD A,B
OR B	OR 40H
LD E,A	LD B,A
;	JP END_COM
LD A,(IY+3) ;MIN	AM: LD A,B
AND 0FH	AND 0BFH

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

LD B,A                                RET
END_COM: RET                            ;
;-----;                               XM: LD A,0C6H ;AM/PM
; SUBROUTINE FOR TIME ;                 CALL LCD_COM
;-----;                               LD A,(IY+0FH)
SEC: LD IY,0D000H                       CALL WR_ASCII
LD A,0CFH ;SEC                          LD A,0C5H
CALL LCD_COM                             CALL LCD_COM
LD A,(IY+0)                              LD A,(IY+0EH)
CALL WR_ASCII                             CALL WR_ASCII
LD A,0CEH                                  RET
CALL LCD_COM                               ;
LD A,(IY+1)                               WEEK: LD A,85H ;WEEK
CALL WR_ASCII                             CALL LCD_COM
RET                                         LD A,(IY+11H)
;                                           CALL WR_ASCII
MIN: LD A,0CCH ;MIN                       LD A,84H
CALL LCD_COM                             CALL LCD_COM
LD A,(IY+2)                              LD A,(IY+12H)
CALL WR_ASCII                             CALL WR_ASCII
LD A,0CBH                                  LD A,83H
CALL LCD_COM                             CALL LCD_COM
LD A,(IY+3)                              LD A,(IY+13H)
CALL WR_ASCII                             CALL WR_ASCII
RET                                         RET
;                                           ;
HOUR: LD A,0C9H                          ATE: LD A,88H ;DAY
CALL LCD_COM ;HOUR                       CALL LCD_COM
LD A,(IY+4)                              LD A,(IY+6)
CALL WR_ASCII                             CALL WR_ASCII
LD A,0C8H                                  LD A,87H
CALL LCD_COM                             CALL LCD_COM
LD A,(IY+5)                              LD A,(IY+7)
CALL WR_ASCII                             CALL WR_ASCII

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

RET                                JP Z,C110N
;                                LD A,C
MONTH: LD A,8CH ; MONTH        CP 0DH
CALL LCD_COM                      JP NZ,C110N
LD A,(IY+8)                       LD IX,0E000H
CALL WR_ASCII                     SET 0,(IX+4)
LD A,8BH                          C110N: LD IX,0D10DH ;CH1_1OFF
CALL LCD_COM                      CALL COM_RAM
LD A,(IY+9)                       LD A,E
CALL WR_ASCII                     CP 0DH
LD A,8AH                          JP Z,C110FF
CALL LCD_COM                      LD A,C
LD A,(IY+0AH)                    CP 0DH
CALL WR_ASCII                     JP NZ,C110FF
RET                                LD IX,0E000H
;                                RES 0,(IX+4)
YEAR: LD A,8FH ; YEAR           ;
CALL LCD_COM                      C110FF: LD IX,0D11AH ;CH1_2ON
LD A,(IY+0BH)                    CALL COM_RAM
CALL WR_ASCII                     LD A,E
LD A,8EH                          CP 0DH
CALL LCD_COM                      JP Z,C120N
LD A,(IY+0CH)                    LD A,C
CALL WR_ASCII                     CP 0DH
RET                                JP NZ,C120N
;                                LD IX,0E000H
;---- SUPROUTINE FOR CHECK ANY   SET 0,(IX+4)
CHANNAL-----;                 C120N: LD IX,0D127H ;CH1_2OFF
;                                CALL COM_RAM
CHANNAL1: CALL LOAD_COM          LD A,E
LD IX,0D100H ;CH1_1ON           CP 0DH
CALL COM_RAM                    JP Z,C120FF
LD A,E                          LD A,C
CP 0DH                          CP 0DH

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

JP NZ,C12OFF
LD IX,0E000H
RES 0,(IX+4)
;=====
C12OFF: LD IX,0D134H ;CH2_1ON
CALL COM_RAM
LD A,E
CP 0DH
JP Z,C21ON
LD A,C
CP 0DH
JP NZ,C21ON
LD IX,0E000H
SET 1,(IX+4)
C21ON: LD IX,0D141H ;CH2_1OFF
CALL COM_RAM
LD A,E
CP 0DH
JP Z,C21OFF
LD A,C
CP 0DH
JP NZ,C21OFF
LD IX,0E000H
RES 1,(IX+4)
;
C21OFF: LD IX,0D14EH ;CH2_2ON
CALL COM_RAM
LD A,E
CP 0DH
JP Z,C22ON
LD A,C
CP 0DH
JP NZ,C22ON
LD IX,0E000H
;
SET 1,(IX+4)
C22ON: LD IX,0D15BH ;CH2_2OFF
CALL COM_RAM
LD A,E
CP 0DH
JP Z,C22OFF
LD A,C
CP 0DH
JP NZ,C22OFF
LD IX,0E000H
RES 1,(IX+4)
;=====
C22OFF: LD IX,0D168H ;CH3_1ON
CALL COM_RAM
LD A,E
CP 0DH
JP Z,C31ON
LD A,C
CP 0DH
JP NZ,C31ON
LD IX,0E000H
SET 2,(IX+4)
C31ON: LD IX,0D175H ;CH3_1OFF
CALL COM_RAM
LD A,E
CP 0DH
JP Z,C31OFF
LD A,C
CP 0DH
JP NZ,C31OFF
LD IX,0E000H
RES 2,(IX+4)
;
LD IX,0E000H

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

JP NZ,C12OFF
LD IX,0E000H
RES 0,(IX+4)
=====
C12OFF: LD IX,0D134H ;CH2_1ON
CALL COM_RAM
LD A,E
CP 0DH
JP Z,C21ON
LD A,C
CP 0DH
JP NZ,C21ON
LD IX,0E000H
SET 1,(IX+4)
C21ON: LD IX,0D141H ;CH2_1OFF
CALL COM_RAM
LD A,E
CP 0DH
JP Z,C21OFF
LD A,C
CP 0DH
JP NZ,C21OFF
LD IX,0E000H
RES 1,(IX+4)
;
C21OFF: LD IX,0D14EH ;CH2_2ON
CALL COM_RAM
LD A,E
CP 0DH
JP Z,C22ON
LD A,C
CP 0DH
JP NZ,C22ON
LD IX,0E000H
;
SET 1,(IX+4)
C22ON: LD IX,0D15BH ;CH2_2OFF
CALL COM_RAM
LD A,E
CP 0DH
JP Z,C22OFF
LD A,C
CP 0DH
JP NZ,C22OFF
LD IX,0E000H
RES 1,(IX+4)
=====
C22OFF: LD IX,0D168H ;CH3_1ON
CALL COM_RAM
LD A,E
CP 0DH
JP Z,C31ON
LD A,C
CP 0DH
JP NZ,C31ON
LD IX,0E000H
SET 2,(IX+4)
C31ON: LD IX,0D175H ;CH3_1OFF
CALL COM_RAM
LD A,E
CP 0DH
JP Z,C31OFF
LD A,C
CP 0DH
JP NZ,C31OFF
LD IX,0E000H
RES 2,(IX+4)
;
C21OFF: LD IX,0D14EH ;CH2_2ON
CALL COM_RAM
LD A,E
CP 0DH
JP Z,C22ON
LD A,C
CP 0DH
JP NZ,C22ON
LD IX,0E000H
;

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

C31OFF: LD IX,0D182H ;CH3_2ON LD A,E
CALL COM_RAM CP 0DH
LD A,E JP Z,C41OFF
CP 0DH LD A,C
JP Z,C32ON CP 0DH
LD A,C JP NZ,C41OFF
CP 0DH LD IX,0E000H
JP NZ,C32ON RES 3,(IX+4)
LD IX,0E000H ;
SET 2,(IX+4) C41OFF: LD IX,0D1B6H ;CH4_2ON
CALL COM_RAM
LD A,E
CP 0DH
JP Z,C42ON
LD A,C
CP 0DH
JP NZ,C42ON
LD IX,0E000H
SET 3,(IX+4)
RES 2,(IX+4) C42ON: LD IX,0D1C3H ;CH4_2OFF
CALL COM_RAM
LD A,E
CP 0DH
JP Z,C42OFF
LD A,C
CP 0DH
JP NZ,C42OFF
LD IX,0E000H
RES 3,(IX+4)
;
C32OFF: LD IX,0D19CH ;CH4_1ON
CALL COM_RAM
LD A,E
CP 0DH
JP Z,C41ON
LD A,C
CP 0DH
JP NZ,C41ON
LD IX,0E000H
SET 3,(IX+4)
C41ON: LD IX,0D1A9H ;CH4_1OFF
CALL COM_RAM
LD A,E
CP 0DH
JP Z,C41OFF
LD A,C
CP 0DH
JP NZ,C41OFF
LD IX,0E000H
RES 3,(IX+4)
;
C32ON: LD IX,0D18FH ;CH3_2OFF
CALL COM_RAM
LD A,E
CP 0DH
JP Z,C32ON
LD A,C
CP 0DH
JP NZ,C32OFF
LD IX,0E000H
RES 2,(IX+4)
C41OFF: LD IX,0D1B6H ;CH4_2ON
CALL COM_RAM
LD A,E
CP 0DH
JP Z,C42ON
LD A,C
CP 0DH
JP NZ,C42ON
LD IX,0E000H
SET 3,(IX+4)
RES 2,(IX+4)
C42ON: LD IX,0D1C3H ;CH4_2OFF
CALL COM_RAM
LD A,E
CP 0DH
JP Z,C42OFF
LD A,C
CP 0DH
JP NZ,C42OFF
LD IX,0E000H
RES 3,(IX+4)
;
C32OFF: LD IX,0D19CH ;CH4_1ON
CALL COM_RAM
LD A,E
CP 0DH
JP Z,C41ON
LD A,C
CP 0DH
JP NZ,C41ON
LD IX,0E000H
SET 3,(IX+4)
C41ON: LD IX,0D1A9H ;CH4_1OFF
CALL COM_RAM
LD A,E
CP 0DH
JP Z,C41OFF
LD A,C
CP 0DH
JP NZ,C41OFF
LD IX,0E000H
RES 3,(IX+4)
;
C42OFF: LD IX,0D1D0H ;CH5_1ON
CALL COM_RAM
LD A,E

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

CP 0DH		CP 0DH	
JP Z,C51ON		JP NZ,C52OFF	
LD A,C		LD IX,0E000H	
CP 0DH		RES 4,(IX+4)	
JP NZ,C51ON			
LD IX,0E000H			
SET 4,(IX+4)			
C51ON: LD IX,0D1DDH ;CH5_1OFF		C52OFF: LD IX,0D204H ;CH6_1ON	
CALL COM_RAM		CALL COM_RAM	
LD A,E		LD A,E	
CP 0DH		CP 0DH	
JP Z,C51OFF		JP Z,C61ON	
LD A,C		LD A,C	
CP 0DH		CP 0DH	
JP NZ,C51OFF		JP NZ,C61ON	
LD IX,0E000H		LD IX,0E000H	
RES 4,(IX+4)		SET 5,(IX+4)	
;		C61ON: LD IX,0D211H ;CH6_1OFF	
C51OFF: LD IX,0D1BAH ;CH5_2ON		CALL COM_RAM	
CALL COM_RAM		LD A,E	
LD A,E		CP 0DH	
CP 0DH		JP Z,C61OFF	
JP Z,C52ON		LD A,C	
LD A,C		CP 0DH	
CP 0DH		JP NZ,C61OFF	
JP NZ,C52ON		LD IX,0E000H	
LD IX,0E000H		RES 5,(IX+4)	
SET 4,(IX+4)		;	
C52ON: LD IX,0D1F7H ;CH5_2OFF		C61OFF: LD IX,0D21EH ;CH6_2ON	
CALL COM_RAM		CALL COM_RAM	
LD A,E		LD A,E	
CP 0DH		CP 0DH	
JP Z,C52OFF		JP Z,C62ON	
LD A,C		LD A,C	
		CP 0DH	

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

JP NZ,C62ON                                RES 6,(IX+4)
LD IX,0E000H                                ;
SET 5,(IX+4)                                C71OFF: LD IX,0D252H    ;CH7_2ON
C62ON: LD IX,0D22BH    ;CH6_2OFF            CALL COM_RAM
CALL COM_RAM                                LD A,E
LD A,E                                        CP 0DH
CP 0DH                                        JP Z,C72ON
JP Z,C62OFF                                  LD A,C
LD A,C                                        CP 0DH
CP 0DH                                        JP NZ,C72ON
JP NZ,C62OFF                                  LD IX,0E000H
LD IX,0E000H                                  SET 6,(IX+4)
RES 5,(IX+4)                                C72ON: LD IX,0D25FH    ;CH7_2OFF
;
C62OFF: LD IX,0D238H    ;CH7_1ON            CALL COM_RAM
CALL COM_RAM                                LD A,E
LD A,E                                        CP 0DH
CP 0DH                                        JP Z,C72OFF
JP Z,C71ON                                  LD A,C
LD A,C                                        CP 0DH
CP 0DH                                        JP NZ,C72OFF
JP NZ,C71ON                                  LD IX,0E000H
LD IX,0E000H                                  RES 6,(IX+4)
SET 6,(IX+4)                                ;
C71ON: LD IX,0D245H    ;CH7_1OFF            CALL COM_RAM
CALL COM_RAM                                LD A,E
LD A,E                                        CP 0DH
CP 0DH                                        JP Z,C81ON
JP Z,C71OFF                                  LD A,C
LD A,C                                        CP 0DH
CP 0DH                                        JP NZ,C81ON
JP NZ,C71OFF                                  LD IX,0E000H
LD IX,0E000H                                  SET 7,(IX+4)

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

C81ON: LD IX,0D279H ;CH8_1OFF CALL COM_RAM
      LD A,E
      CP 0DH
      JP Z,C81OFF
      LD A,C
      CP 0DH
      JP NZ,C81OFF
      LD IX,0E000H
      RES 7,(IX+4)
      ;
C81OFF: LD IX,0D286H ;CH8_2ON CALL COM_RAM
      LD A,E
      CP 0DH
      JP Z,C82ON
      LD A,C
      CP 0DH
      JP NZ,C82ON
      LD IX,0E000H
      SET 7,(IX+4)
C82ON: LD IX,0D293H ;CH8_2OFF CALL COM_RAM
      LD A,E
      CP 0DH
      JP Z,C82OFF
      LD A,C
      CP 0DH
      JP NZ,C82OFF
      LD IX,0E000H
      RES 7,(IX+4)
      ;
C91ON: LD IX,0D2ADH ;CH9_1OFF CALL COM_RAM
      LD A,E
      CP 0DH
      JP Z,C91OFF
      LD A,C
      CP 0DH
      JP NZ,C91OFF
      LD A,00H
      OUT (PORTAA),A
      ;
C91OFF: LD IX,0D2BAH ;CH9_2ON CALL COM_RAM
      LD A,E
      CP 0DH
      JP Z,C92ON
      LD A,C
      CP 0DH
      JP NZ,C92ON
      LD IX,0E000H
      LD A,(IX+19H)
C82OFF: LD IX,0D2A0H ;CH9_1ON CALL COM_RAM
      LD A,E
      CP 0DH
      JP Z,C91ON
      LD A,C
      CP 0DH
      JP NZ,C91ON
      LD IX,0E000H
      LD A,(IX+19H)
      OUT (PORTAA),A
      CALL DELAY1
      CALL DELAY1

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

OUT(PORTAA),A                                LD IX,0E000H
CALL DELAY1                                  RES 0,(IX+5)
CALL DELAY1                                  ;
C92ON: LD IX,0D2C7H ;CH9_2OFF                C101OFF: LD IX,0D2EEH ;CH10_2ON
CALL COM_RAM                                 CALL COM_RAM
LD A,E                                       LD A,E
CP 0DH                                       CP 0DH
JP Z,C92OFF                                  JP Z,C102ON
LD A,C                                       LD A,C
CP 0DH                                       CP 0DH
JP NZ,C92OFF                                 JP NZ,C102ON
LD A,00H                                     LD IX,0E000H
OUT(PORTAA),A                               SET 0,(IX+5)
JP C92OFF                                    C102ON: LD IX,0D2FBH ;CH10_2OFF
;                                             CALL COM_RAM
;                                             LD A,E
;                                             CP 0DH
C92OFF: LD IX,0D2D4H ;CH10_1ON              JP Z,C102OFF
CALL COM_RAM                                 LD A,C
LD A,E                                       CP 0DH
CP 0DH                                       JP NZ,C102OFF
JP Z,C101ON                                  LD IX,0E000H
LD A,C                                       RES 0,(IX+5)
CP 0DH
JP NZ,C101ON
LD IX,0E000H
SET 0,(IX+5)
;
C101ON: LD IX,0D2E1H ;CH10_1OFF              C102OFF: LD IX,0D308H ;CH11_1ON
CALL COM_RAM                                 CALL COM_RAM
LD A,E                                       LD A,E
CP 0DH                                       CP 0DH
JP Z,C101OFF                                  JP Z,C111ON
LD A,C                                       LD A,C
CP 0DH                                       CP 0DH
JP NZ,C101OFF                                 JP NZ,C111ON
LD IX,0E000H

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

SET 1,(IX+5)
C111ON: LD IX,0D315H ;CH11_1OFF
CALL COM_RAM
LD A,E
CP 0DH
JP Z,C111OFF
LD A,C
CP 0DH
JP NZ,C111OFF
LD IX,0E000H
RES 1,(IX+5)
;
C111OFF: LD IX,0D322H ;CH11_2ON
CALL COM_RAM
LD A,E
CP 0DH
JP Z,C112ON
LD A,C
CP 0DH
JP NZ,C112ON
LD IX,0E000H
SET 1,(IX+5)
C112ON: LD IX,0D32FH ;CH11_2OFF
CALL COM_RAM
LD A,E
CP 0DH
JP Z,C112OFF
LD A,C
CP 0DH
JP NZ,C112OFF
LD IX,0E000H
RES 1,(IX+5)
;
SET 2,(IX+5)
C112OFF: LD IX,0D33CH ;CH12_1ON
CALL COM_RAM
LD A,E
CP 0DH
JP Z,C121ON
LD A,C
CP 0DH
JP NZ,C121ON
LD IX,0E000H
SET 2,(IX+5)
C121ON: LD IX,0D349H ;CH12_1OFF
CALL COM_RAM
LD A,E
CP 0DH
JP Z,C121OFF
LD A,C
CP 0DH
JP NZ,C121OFF
LD IX,0E000H
RES 2,(IX+5)
;
C121OFF: LD IX,0D356H ;CH12_2ON
CALL COM_RAM
LD A,E
CP 0DH
JP Z,C122ON
LD A,C
CP 0DH
JP NZ,C122ON
LD IX,0E000H
SET 2,(IX+5)
C122ON: LD IX,0D363H ;CH12_2OFF
CALL COM_RAM
LD A,E

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

CP 0DH	IN A,(SEC1)
JP Z,C122OFF	LD (IY+0),A
LD A,C	IN A,(SEC10)
CP 0DH	LD (IY+1),A
JP NZ,C122OFF	IN A,(MIN1)
LD IX,0E000H	LD (IY+2),A
RBS 2,(IX+5)	IN A,(MIN10)
C122OFF: LD A,88H	LD (IY+3),A
LD (0D370H),A	IN A,(HOUR1)
RET	LD (IY+4),A
	IN A,(HOUR10)
	LD (IY+5),A
	IN A,(DAY1)
COM_RAM: LD B,0DH	LD (IY+6),A
LD C,00H	IN A,(DAY10)
LD E,00H	LD (IY+7),A
LD IY,0E200H	IN A,(MON1)
COMP_LP: LD A,(IY+0)	LD (IY+8),A
CP (IX+0)	IN A,(MON10)
JP NZ,COM_NXT	LD (IY+9),A
RE_COM: INC IY	IN A,(YEAR1)
INC IX	LD (IY+0AH),A
INC C	IN A,(YEAR10)
DEC B	LD (IY+0BH),A
JP NZ,COMP_LP	IN A,(WEEK1)
JP R_COM	LD (IY+0CH),A
COM_NXT: LD A,(IX+00H)	RET
CP 0FDH	;
JP NZ,R_COM	;-----;
INC E	; SUBROUTINE FOR CHECK KEY ;
JP RE_COM	;-----;
R_COM: RET	;
;	COMP: LD IY,0D000H
LOAD_COM: LD IY,0E200H	IN A,(SEC1)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

AND 3FH	AND 31H
LD (IY+00H),A	NEXT_DAY: LD (IY+5),A
;	;
IN A,(SEC10)	IN A,(DAY1)
AND 3FH	AND 3FH
LD (IY+1),A	LD (IY+6),A
;	IN A,(DAY10)
IN A,(MIN1)	AND 3FH
AND 3FH	LD (IY+7),A
LD (IY+2),A	;
;	CALL DS_WEEK
IN A,(MIN10)	;
AND 3FH	IN A,(YEAR1)
LD (IY+3),A	AND 3FH
;	LD (IY+0BH),A
IN A,(HOUR1)	IN A,(YEAR10)
AND 3FH	AND 3FH
LD (IY+4),A	LD (IY+0CH),A
;	;
IN A,(HOUR10)	COMM: LD A,L ;MOUTH
AND 3FH	CP 01H
CP 35H	JP Z,JAN
JP Z,DIS_PM	CP 02H
CP 34H	JP Z,FEB
JP Z,DIS_PM1	CP 03H
LD (IY+0EH),41H	JP Z,MAR
LD (IY+0FH),6DH	CP 04H
JP NEXT_DAY	JP Z,APR
DIS_PM1: LD (IY+0EH),50H	CP 05H
LD (IY+0FH),6DH	JP Z,MAY
AND 30H	CP 06H
JP NEXT_DAY	JP Z,JUN
DIS_PM: LD (IY+0EH),50H	CP 07H
LD (IY+0FH),6DH	JP Z,JLY

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

CP 08H	;-----WEEK
JP Z,AUG	NONE_WEEK: LD (IY+13H),2DH
CP 09H	LD (IY+12H),2DH
JP Z,SEP	LD (IY+11H),2DH
CP 10H	JP NX_COMP
JP Z,OCT	;
CP 11H	SUN: LD (IY+13H),55H
JP Z,NOV	LD (IY+12H),55H
CP 12H	LD (IY+11H),4EH
JP Z,DECS	JP NX_COMP
CP 13H	;
JP Z,NONE_MOUTH	MON: LD (IY+13H),4DH
RET_MOUTH: RET	LD (IY+12H),4FH
;	LD (IY+11H),4EH
DS_WEEK: LD IY,0D000H	JP NX_COMP
LD A,(IY+14H)	;
AND 0FH	TUE: LD (IY+13H),54H
CP 00H	LD (IY+12H),55H
JP Z,SUN	LD (IY+11H),45H
CP 01H	JP NX_COMP
JP Z,MON	;
CP 02H	WED: LD (IY+13H),57H
JP Z,TUE	LD (IY+12H),45H
CP 03H	LD (IY+11H),44H
JP Z,WED	JP NX_COMP
CP 04H	;
JP Z,THU	THU: LD (IY+13H),54H
CP 05H	LD (IY+12H),48H
JP Z,FRI	LD (IY+11H),55H
CP 06H	JP NX_COMP
JP Z,SAT	;
CP 07H	FRI: LD (IY+13H),46H
JP Z,NONE_WEEK	LD (IY+12H),52H
NX_COMP: RET	LD (IY+11H),49H

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

JP NX_COMP	LD (IY+9),41H
;	LD (IY+8),59H
SAT: LD (IY+13H),53H	JP RET_MOUTH
LD (IY+12H),41H	;
LD (IY+11H),54H	JUN: LD (IY+0AH),4AH
JP NX_COMP	LD (IY+9),55H
;	LD (IY+8),4EH
-----MOUTH	JP RET_MOUTH
NONE_MOUTH: LD (IY+0AH),2DH	;
LD (IY+9),2DH	JLY: LD (IY+0AH),4AH
LD (IY+8),2DH	LD (IY+9),4CH
JP RET_MOUTH	LD (IY+8),59H
;	JP RET_MOUTH
JAN: LD (IY+0AH),4AH	;
LD (IY+9),41H	AUG: LD (IY+0AH),41H
LD (IY+8),4EH	LD (IY+9),55H
JP RET_MOUTH	LD (IY+8),47H
;	JP RET_MOUTH
FEB: LD (IY+0AH),46H	;
LD (IY+9),45H	SEP: LD (IY+0AH),53H
LD (IY+8),42H	LD (IY+9),45H
JP RET_MOUTH	LD (IY+8),50H
;	JP RET_MOUTH
MAR: LD (IY+0AH),4DH	;
LD (IY+9),41H	OCT: LD (IY+0AH),4FH
LD (IY+8),52H	LD (IY+9),43H
JP RET_MOUTH	LD (IY+8),54H
;	JP RET_MOUTH
APR: LD (IY+0AH),41H	;
LD (IY+9),50H	NOV: LD (IY+0AH),4EH
LD (IY+8),52H	LD (IY+9),4FH
JP RET_MOUTH	LD (IY+8),56H
;	JP RET_MOUTH
MAY: LD (IY+0AH),4DH	;

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

DECS:  LD (IY+0AH),44H          ;
        LD (IY+9),45H          LD A,8CH      ; MOUNT
        LD (IY+8),43H          CALL LCD_COM
        JP RET_MOUTH          LD A,(IY+8)
        ;                      CALL WR_ASCII
RPL1:  RL A                      LD A,8BH
        RL A                      CALL LCD_COM
        RL A                      LD A,(IY+9)
        RL A                      CALL WR_ASCII
        RET                      LD A,8AH
        ;                      CALL LCD_COM
COUNT: LD IY,0D000H          LD A,(IY+0AH)
        LD A,85H      ; WEEK    CALL WR_ASCII
        CALL LCD_COM          ;
        LD A,(IY+11H)          LD A,8FH      ; YEAR
        CALL WR_ASCII        CALL LCD_COM
        LD A,84H              LD A,(IY+0BH)
        CALL LCD_COM        CALL WR_ASCII
        LD A,(IY+12H)        LD A,8EH
        CALL WR_ASCII        CALL LCD_COM
        LD A,83H              LD A,(IY+0CH)
        CALL LCD_COM        CALL WR_ASCII
        LD A,(IY+13H)        LD A,0CFH     ; SEC
        CALL WR_ASCII        CALL LCD_COM
        ;                      LD A,(IY+00H)
        LD A,88H      ; DAY    CALL WR_ASCII
        CALL LCD_COM        LD A,0CEH
        LD A,(IY+6)          CALL LCD_COM
        CALL WR_ASCII        LD A,(IY+1)
        LD A,87H              CALL WR_ASCII
        CALL LCD_COM        ;
        LD A,(IY+7)          LD A,0CCH     ; MIN
        CALL WR_ASCII        CALL LCD_COM

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

LD A,(IY+2)
CALL WR_ASCII
LD A,0CBH
CALL LCD_COM
LD A,(IY+3)
CALL WR_ASCII

LD A,0C9H
CALL LCD_COM ; HOUR
LD A,(IY+4)
CALL WR_ASCII
LD A,0C8H
CALL LCD_COM
LD A,(IY+5)
CALL WR_ASCII
:
LD A,0C6H ;AM/PM
CALL LCD_COM
LD A,(IY+0FH)
CALL WR_ASCII
LD A,0C5H
CALL LCD_COM
LD A,(IY+0EH)
CALL WR_ASCII
RET
;
;-----
C_CURRENT: LD A,1
CALL LCD_COM
LD IX,0E000H
LD (IX+15H),59H
LD (IX+16H),65H
LD (IX+17H),73H
LD (IX+03H),02H

LD HL,CLR_OUT
CALL DIS_LCD
JP LP_YESNO
;
;-----Clear all
program
CLEAR_C: LD IX,0E000H
LD (IX+3),05H
LD HL,RESET
CALL DIS_LCD
I2_RES: LD (IX+15H),59H
LD (IX+16H),65H
LD (IX+17H),73H
RES_LOOP: LD A,00001101B
CALL LCD_COM
LP_YESNO: LD HL,YESNO
CALL DIS_LCD1
CALL SHOW_CLR
CALL T_SET
CALL DIS_RES
JP RES_LOOP
;
DIS_RES: LD HL,CHM_KEY
LD A,L
ADD A,B
LD L,A
LD A,(HL)
LD B,A
CP 49H
JP Z,INC_RES
CP 44H
JP Z,INC_RES
CP 45H
JP Z,MAIN

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

CP 3CH                                JP CLR_LOPP
JP Z,INC_RES                            ;
CP 3EH                                SHOW_CLR: LD A,0CCH
JP Z,INC_RES                            CALL LCD_COM
CP 65H                                LD A,(IX+15H)
JP Z,ENT_RES                            CALL WR_ASCII
RET                                    LD A,0CDH
;                                       CALL LCD_COM
INC_RES: LD A,(IX+15H)                 LD A,(IX+16H)
CP 59H                                CALL WR_ASCII
JP NZ,I2_RES                            LD A,0CEH
LD (IX+15H),4EH                        CALL LCD_COM
LD (IX+16H),6FH                        LD A,(IX+17H)
LD (IX+17H),20H                        CALL WR_ASCII
JP RES_LOOP                            RET
;                                       CLR_PUT: LD (IX+4),00H
ENT_RES: LD A,(IX+15H)                 LD (IX+5),00H
CP 59H                                LD A,00H
JP NZ,MAIN                              OUT (PORTAA),A
LD IX,0E000H                            CALL OUTPUT
LD A,(IX+03H)                          JP MAIN
CP 02H                                ;
JP Z,CLR_PUT                            ;*****
CP 04H                                SERIAL: LD HL,LOAD
JP Z,SERJMP                            CALL DIS_LCD
CP 05H                                LD IX,0E000H
JP NZ,MAIN                              LD (IX+15H),59H
LD IX,0D100H                            LD (IX+16H),65H
CLR_LOPP: LD A,(IX+00H)                LD (IX+17H),73H
CP 88H                                LD (IX+03H),04H
JP Z,MAIN                              JP RES_LOOP
LD A,00H                                ;
LD (IX+00H),A                          SERJMP: LD A,5H ;1200 AT XTAL
INC IX                                  12.288 MHz

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

OUT0 (CNTLB1),A
LD A,00001100B
CALL LCD_COM
LD HL,CONNECT
CALL DIS_LCD1
CALL RXBYTE
CP 99H
JP NZ,.SERJMP
;
LD A,00001100B
NNSER: CALL LCD_COM
LD HL,LOAD
CALL DIS_LCD
LD HL,READY
CALL DIS_LCD1
CALL RXBYTE
CP 90H
JP Z,SE_RET
CP 60H
JP Z,LP_LOAD
JP NNSER
;
LP_LOAD: LD IX,0D100H
LP_LOAD1: CALL RXBYTE
CP 80H
JP Z,SERET
LD (IX+00H),A
INC IX
JP LP_LOAD1
SERET: LD IX,0D375H
SETRE: CALL RXBYTE
LD (IX+00H),A
CP 82H
JP Z,SE_RET1
;
INC IX
JP SETRE
;
SE_RET1: LD IX,0D375H
LD A,(IX+00H)
CP 0EEH
JP Z,NNLTR
LD A,(IX+1)
OUT (PORTA),A
LD IX,0E000H
LD (IX+19H),A
LD (IX+18H),44H
SE_RET: JP MAIN
NNLTR: LD IX,0E000H
LD (IX+19H),00H
LD (IX+18H),00H
JP MAIN
;-----
; SUBROUTINE FOR SCANKEY
;-----
SCANKEY: LD B,00H
LOOP: LD A,B
OUT (PORTC),A
IN A,(PORTC)
LD E,A
CP 0F0H
JP C,NEXT
INC B
LD A,B
CP 0AH
JP NZ,LOOP
JP SCANKEY
NEXT: RET
;

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

CHK_KEY: IN A,(PORTC) ;--> CHECK
FOR PUSH KEY
    NOR E
    JP Z,CHK_KEY
    RET
    ;
;-----;
: SUBROUTINE FOR COMPEAR KEY :
;-----;
;
DISCHK: LD B,00H
        LD HL,TABLE_KEY
DIS_LPT: LD A,(HL)
        CP E
        JP Z,DIS_L
        INC HL
        INC B
        LD A,(HL)
        CP OFFH
        JP NZ,DIS_LPT
DIS_L:  RET
;
DISPLAY_T: PUSH HL
        LD HL,COM_KEY
        LD A,L
        ADD A,B
        LD L,A
        LD A,(HL)
        LD B,A
        CALL CLEAR
        CALL C_KEY1
DIS_RT: RET
;
DISCHKM: LD B,00H

```

```

LD HL,TAB_KEY
DIS_LPTM: LD A,(HL)
        CP E
        JP Z,DIS_LM
        INC HL
        INC B
        LD A,(HL)
        CP OFFH
        JP NZ,DIS_LPTM
DIS_LM:  RET
;
DIS_TM:  LD HL,CC_KEY
        LD A,L
        ADD A,B
        LD L,A
        LD A,(HL)
        LD B,A
        CALL C_KEY1
DIS_RTM: RET
;
;-----;
: SUBROUTINE FOR CHECK KEY :
;-----;
;
C_KEY1: LD A,B
        CP 47H
        JP Z,PROGRAM
        CP 4DH
        JP Z,MANUAL
        CP 45H
        JP Z,ESCAPE
        CP 50H
        JP Z,SERIAL

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

CP 41H                                CALL S_DLY1
JP Z,CURRENT                          LD A,00H
CP 43H                                OUT (PORTBB),A
JP Z,C_CURRENT                        CALL S_DLY1
CP 54H                                DEC B
JP Z,TIME                             JP NZ,S_ER
CP 63H                                LD B,00H
JP Z,CLEAR_C                          RET
POP HL                                ;
RET                                    S_DLY1: PUSH BC
;                                     LD B,04H
;-----;                             S2: LD C,0EEH
; SUBROUTINE FOR SOUND ;             S3: DEC C
;-----;                             JP NZ,S3
;                                     DEC B
SOUND: PUSH BC                       JP NZ,S2
LD B,5FH ;-----> WHEN             POP BC
PUSH KEY                               RET
S_LOOP: LD A,80H ;----->         ;
SOUND BEEP                            ;-----;
OUT (PORTBB),A                        ; SUBROUTINE FOR LCD ;
CALL S_DLY                             ;-----;
LD A,00H                               ;
OUT (PORTBB),A                        CLEAR: LD A,1
CALL S_DLY                             CALL LCD_COM
DEC B                                  RET
JP NZ,S_LOOP                           ;
LD B,00H                               LCD_COM: CALL BUSY
POP BC                                 OUT (COM_PORT),A
RET                                    RET
;                                     ;
SD_ERR: LD B,60H                       WR_ASCII: CALL BUSY
S_ER: LD A,80H                         OUT (WR_PORT),A
OUT (PORTBB),A                         RET

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

;
; LD A,D
BUSY:  PUSH AF
; CALL LCD_COM
BUSY1: IN A,(BUSY_PORT)
; LD A,E
; CALL WR_ASCII
; INC D
; INC HL
; DEC B
; JP NZ,DIS_LP1
;
;-----;
; RET
; SUBROUTINE FOR DISPLAY ON LCD ;
; ADDRESS 80H ;
;-----;
DIS_LCD: LD B,10H
; *****
; GET CHARECTER *
; *****
; input -
; output AC
; reg C
;
LD D,80H
DIS_LP: LD A,(HL)
; RXBYTE: DI
; RXBYTE1: IN0 C,(STAT1) ;Rxready
; BIT 7,C
; JP Z,RXBYTE1
; IN0 C,(RDR1) ;RECEIVE
; LD A,C
; RXBYTE2: EI
; RET
;
;
;-----;
; *****
; SUBROUTINE FOR DISPLAY ON LCD ;
; TX ASCII ONE BYTE *
; *****
; input C
; output -
; reg AC
;
DIS_LCD1: LD B,10H
; TXBYTE: IN0 A,(STAT1)
; LD D,0C0H
DIS_LP1: LD A,(HL)
; LD E,A

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

BIT 1,A
JP Z,TXBYTE
OUT0 (TDR1),C
RET
;-----;
; SUBROUTINE FOR DELAY ;
;-----;
S_DLY:  PUSH BC ;----->
DELAY FOR SOUND KEY
LD B,28H
S1:  DEC B
JP NZ,S1
POP BC
RET
;-----;
DELAY:  PUSH HL ;----->
DELAY FOR SCAN KEY
PUSH DE
LD HL,8800H
LD DE,0001H
DLY:  SBC HL,DE
JP NZ,DLY
POP DE
POP HL
RET
;-----;
DELAY1:  PUSH HL ;----->
DELAY FOR SCAN KEY
PUSH DE
LD HL,0FFFFH
LD DE,0001H
DLY1:  SBC HL,DE
JP NZ,DLY1
POP DE

```

```

POP HL
RET
;-----;
DELAY2:  PUSH HL ;----->
DELAY FOR SCAN KEY
PUSH DE
LD HL,7000H
LD DE,0001H
DLY2:  SBC HL,DE
JP NZ,DLY2
POP DE
POP HL
RET
;-----;
; RTC SUBROUTINE *
;-----;
; OUTPUT SEC D
MIN C
HOR B
DAY H
MON L
YEAR E
WEEK (IY+10H)
SETCK: LD IY,0D000H
LD A,D
OUT (SEC1),A
CALL SHIPL
OUT (SEC10),A
LD A,C
OUT (MIN1),A
CALL SHIPL
OUT (MIN10),A
LD A,B
OUT (HOUR1),A

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

CALL SHIPL                                LD D,A   ;SEC
OUT (HOUR10),A                             IN A,(MIN1)
LD A,H                                     AND 0FH
OUT (DAY1),A                               LD B,A
CALL SHIPL                                IN A,(MIN10)
OUT (DAY10),A                              CALL SHIFTRTC
LD A,L                                     AND 0F0H
OUT (MON1),A                              ADD A,B
CALL SHIPL                                LD C,A   ;MIN
OUT (MON10),A                             IN A,(DAY1)
LD A,E                                     AND 0FH
OUT (YEAR1),A                             LD B,A
CALL SHIPL                                IN A,(DAY10)
OUT (YEAR10),A                            CALL SHIFTRTC
LD A,(IY+14H)                             AND 0F0H
OUT (WEEK1),A                             ADD A,B
RET                                         LD H,A   ;DAY
;***** GET CLOCK *****                IN A,(MON1)
; INPUT SEC D                             AND 0FH
; MIN C                                   LD B,A
; HOR B                                  IN A,(MON10)
; DAY L                                  CALL SHIFTRTC
; MON H                                  AND 0F0H
; YEAR E                                 ADD A,B
; WEEK (IY+14H)                          LD L,A   ;MON
;                                         IN A,(YEAR1)
;                                         AND 0FH
GETCK: LD IY,0D000H                       LD B,A
IN A,(SEC1)                               IN A,(YEAR10)
AND 0FH                                   CALL SHIFTRTC
LD B,A                                     AND 0F0H
IN A,(SEC10)                              ADD A,B
CALL SHIFTRTC                             LD E,A   ;YEAR
AND 0F0H
ADD A,B

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

IN A,(HOUR1)                                RET
AND 0FH                                     SHIPL: RRCA
LD B,A                                       RRCA
IN A,(HOUR10)                               RRCA
CALL SHIFTRTC                               RRCA
AND 0F0H                                     RET
ADD A,B                                     ;
LD B,A   ;HOUR                             ;***** SUB OUT WDT *****
;                                           ;EQU CSWDT,0E0H
IN A,(WEEK1)                                WDT: PUSH AF
LD (IY+14H),A ;WEEK                         IN A,(CSWDT)
RET                                           POP AF
;                                           RET
;-----;
;***** RTC INT *****                    ;-----;
;                                           ; DEFIND TABLE ;
;-----;
INTRTC: LD A,04H   ;set 24h                 ;-----;
OUT (CREG_F),A                               .ORG 0B100H
XOR A                                         TAB_KEY: .DB 0B4H,0B5H,0B1H
OUT (CREG_D),A   ;set HOLD bit=0           .DB 074H,075H,071H
RET                                           .DB 0D4H,0D5H,0D1H
;                                           .DB 0E3H,0E4H,0E5H,0E1H,0FFH
;***** RTC INT 1 SEC *****              CC_KEY: .DB 47H,4DH,45H
;                                           .DB 50H,41H,43H
INTRTC1S:LD A,04H   ;set 24h                .DB 49H,44H,54H
OUT (CREG_F),A                               .DB 63H,3CH,3EH,65H
OUT (CREG_E),A   ;set INT 1sec              ;
XOR A                                         TABLE_KEY:
OUT (CREG_D),A   ;set HOLD bit=0           .DB 0B0H,0B2H,0B3H,0B4H,0B5H,0B1H
RET                                           .DB 070H,072H,073H,074H,075H,071H
;                                           .DB 0D0H,0D2H,0D3H,0D4H,0D5H,0D1H
SHIFTRTC: RLCA                               .DB 0E0H,0E2H,0E3H,0E4H,0E5H,0E1H
RLCA                                           .DB 0FFH
RLCA                                           ;
RLCA                                           COM_KEY: .DB 31H,32H,33H,47H,4DH,45H

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่นอญญาติให้นำไปใช้ประโยชน์ดานการคาไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

.DB 34H,35H,36H,50H,41H,43H
.DB 37H,38H,39H,49H,44H,54H
.DB 0A5H,30H,63H,3CH,3EH,65H
;
CNI_KEY: .DB 31H,32H,33H,10H,10H,45H
.DB 34H,35H,36H,10H,10H,10H
.DB 37H,38H,39H,49H,44H,10H
.DB CA5H,30H,10H,3CH,3EH,65H
;
CHN_KEY: .DB 31H,32H,33H,10H,10H,45H
.DB 34H,35H,36H,10H,10H,10H
.DB 37H,38H,39H,49H,44H,10H
.DB 10H,30H,10H,3CH,3EH,65H
;
CHM_KEY: .DB 31H,32H,33H,10H,10H,45H
.DB 34H,35H,36H,10H,10H,10H
.DB 37H,38H,39H,49H,44H,10H
.DB 2DH,30H,10H,3CH,3EH,65H
;
CHD_KEY: .DB 10H,10H,10H,10H,10H,45H
.DB 10H,10H,10H,10H,10H,10H
.DB 10H,10H,10H,49H,44H,10H
.DB 2DH,10H,10H,3CH,3EH,65H
;
;-----;
; DISPLAY ON LCD FROM ANOTHER ;
; FUNCTION ;
;-----;
;
LIM_CUR: .DB " Limit Current "
PROG: .DB " Program System "
SELECT: .DB " Select Function"
SET_DAY: .DB ">> ---/--/--/--"

SET_TIME1: .DB "Time<-->--:--:--"
SET_DAY2: .DB ">> / / / "
SET_TME2: .DB "Time< > : : "
C_CUR: .DB " Current: A "
STAS: .DB " Status : "
MENU: .DB " Multi Channal "
MAN: .DB " Manual System "
PC: .DB " Select Menu "
PC_2: .DB " RS-232 RS-485 "
CLR_OUT: .DB " Clear all CHN. "
CLR_OUT1: .DB " Yes or No: "
AMP: .DB " Select Channal "
AMP1: .DB " CH <9>: "
CLR: .DB " Clear Program "
YES_NO: .DB " Yes No "
PGM: .DB " CH <1-12>: "
SEL_MENU: .DB " Select Menu "
TIME1: .DB " Channal < > "
STATE: .DB " State: ->On "
CH_TIME: .DB "00>---/--/--/--"
ON_TIME: .DB "On> <-->--:--:--"
OFF_TIME: .DB "Off <-->--:--:--"
RESET: .DB " Reset Program "
YESNO: .DB " Yes or No: "
OVER: .DB " Over Current "
PLEASE: .DB " Please any key "
LOAD: .DB " Program from PC"
READY: .DB " < Ready> "
AAM: .DB " Loading "
CONNECT: .DB " Connecting "
;-----;
.END

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ค.

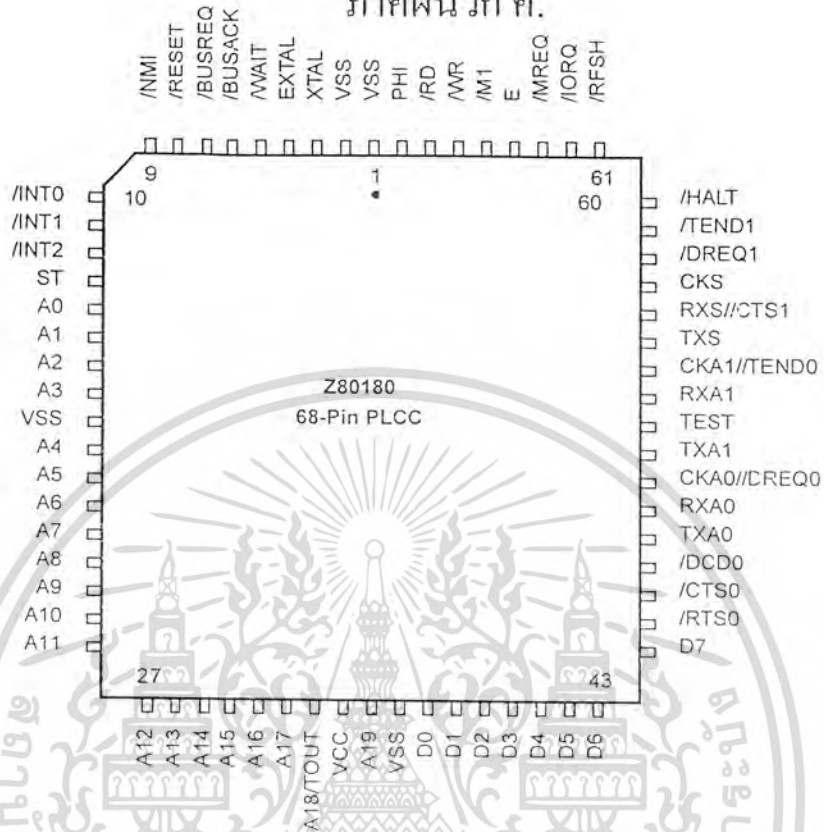


Figure 3. Z80180 68-Pin PLCC Pin Configuration

HM628512P/LP Series
HM628512FP/LFP Series

A18	1	32	Vcc
A16	2	31	A15
A14	3	30	A17
A12	4	29	WE
A7	5	28	A13
A6	6	27	A8
A5	7	26	A9
A4	8	25	A11
A3	9	24	OE
A2	10	23	A10
A1	11	22	CS
A0	12	21	I/O7
I/O0	13	20	I/O6
I/O1	14	19	I/O5
I/O2	15	18	I/O4
Vss	16	17	I/O3

(Top View)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

LM555/LM555C Timer

General Description

The LM555 is a highly stable device for generating accurate time delays or oscillation. Additional terminals are provided for triggering or resetting if desired. In the time delay mode of operation, the time is precisely controlled by one external resistor and capacitor. For astable operation as an oscillator, the free running frequency and duty cycle are accurately controlled with two external resistors and one capacitor. The circuit may be triggered and reset on falling waveforms, and the output circuit can source or sink up to 200 mA or drive TTL circuits.

Features

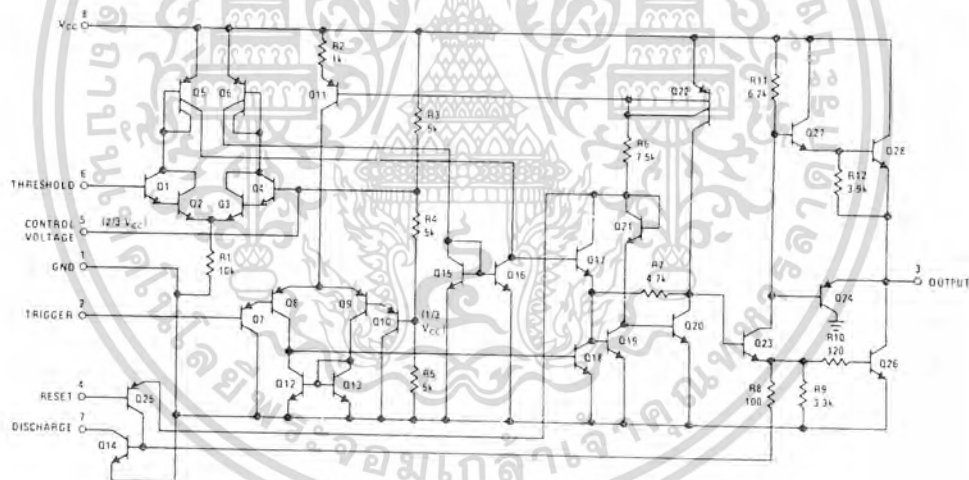
- Direct replacement for SE555/NE555
- Timing from microseconds through hours
- Operates in both astable and monostable modes

- Adjustable duty cycle
- Output can source or sink 200 mA
- Output and supply TTL compatible
- Temperature stability better than 0.005% per °C
- Normally on and normally off output

Applications

- Precision timing
- Pulse generation
- Sequential timing
- Time delay generation
- Pulse width modulation
- Pulse position modulation
- Linear ramp generator

Schematic Diagram



TL/H/7551-1

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Absolute Maximum Ratings

If Military/Aerospace specified devices are required, please contact the National Semiconductor Sales Office/Distributors for availability and specifications.

Supply Voltage	+18V
Power Dissipation (Note 1)	
LM555H, LM555CH	760 mW
LM555, LM555CN	1180 mW
Operating Temperature Ranges	
LM555C	0°C to +70°C
LM555	-55°C to +125°C

Storage Temperature Range -65°C to +150°C

Soldering Information

Dual-In-Line Package	
Soldering (10 Seconds)	260°C
Small Outline Package	
Vapor Phase (60 Seconds)	215°C
Infrared (15 Seconds)	220°C

See AN-450 "Surface Mounting Methods and Their Effect on Product Reliability" for other methods of soldering surface mount devices.

Electrical Characteristics ($T_A = 25^\circ\text{C}$, $V_{CC} = +5\text{V}$ to $+15\text{V}$, unless otherwise specified)

Parameter	Conditions	Limits						Units
		LM555			LM555C			
		Min	Typ	Max	Min	Typ	Max	
Supply Voltage		4.5		18	4.5		16	V
Supply Current	$V_{CC} = 5\text{V}, R_L = \infty$ $V_{CC} = 15\text{V}, R_L = \infty$ (Low State) (Note 2)		3 10	5 12		3 10	6 15	mA mA
Timing Error, Monostable								
Initial Accuracy			0.5			1		%
Drift with Temperature	$R_A = 1\text{k to }100\text{ k}\Omega$, $C = 0.1\ \mu\text{F}$, (Note 3)		30			50		ppm/°C
Accuracy over Temperature			1.5			1.5		%
Drift with Supply			0.05			0.1		%/V
Timing Error, Astable								
Initial Accuracy			1.5			2.25		%
Drift with Temperature	$R_A, R_B = 1\text{k to }100\text{ k}\Omega$, $C = 0.1\ \mu\text{F}$, (Note 3)		90			150		ppm/°C
Accuracy over Temperature			2.5			3.0		%
Drift with Supply			0.15			0.30		%/V
Threshold Voltage				0.667			0.667	$\times V_{CC}$
Trigger Voltage	$V_{CC} = 15\text{V}$ $V_{CC} = 5\text{V}$	4.8 1.45	5 1.67	5.2 1.9		5 1.67		V V
Trigger Current			0.01	0.5		0.5	0.9	μA
Reset Voltage		0.4	0.5	1	0.4	0.5	1	V
Reset Current			0.1	0.4		0.1	0.4	mA
Threshold Current	(Note 4)		0.1	0.25		0.1	0.25	μA
Control Voltage Level	$V_{CC} = 15\text{V}$ $V_{CC} = 5\text{V}$	9.6 2.9	10 3.33	10.4 3.6	9 2.6	10 3.33	11 4	V V
Pin 7 Leakage Output High			1	100		1	100	nA
Pin 7 Sat (Note 5)								
Output Low	$V_{CC} = 15\text{V}, I_7 = 15\text{ mA}$		150			180		mV
Output Low	$V_{CC} = 4.5\text{V}, I_7 = 4.5\text{ mA}$		70	100		80	200	mV

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Electrical Characteristics $T_A = 25^\circ\text{C}$, $V_{CC} = +5\text{V}$ to $+15\text{V}$, (unless otherwise specified) (Continued)

Parameter	Conditions	Limits						Units
		LM555			LM555C			
		Min	Typ	Max	Min	Typ	Max	
Output Voltage Drop (Low)	$V_{CC} = 15\text{V}$		0.1	0.15		0.1	0.25	V
	$I_{\text{SINK}} = 10\text{ mA}$		0.4	0.5		0.4	0.75	V
	$I_{\text{SINK}} = 50\text{ mA}$		2	2.2		2	2.5	V
	$I_{\text{SINK}} = 100\text{ mA}$		2.5			2.5		V
	$V_{CC} = 5\text{V}$		0.1	0.25				V
	$I_{\text{SINK}} = 8\text{ mA}$ $I_{\text{SINK}} = 5\text{ mA}$					0.25	0.35	V
Output Voltage Drop (High)	$I_{\text{SOURCE}} = 200\text{ mA}$, $V_{CC} = 15\text{V}$		12.5			12.5		V
	$I_{\text{SOURCE}} = 100\text{ mA}$, $V_{CC} = 15\text{V}$	13	13.3		12.75	13.3		V
	$V_{CC} = 5\text{V}$	3	3.3		2.75	3.3		V
Rise Time of Output			100			100		ns
Fall Time of Output			100			100		ns

Note 1: For operating at elevated temperatures the device must be derated above 25°C based on a -150°C maximum junction temperature and a thermal resistance of $164^\circ\text{C}/\text{w}$ (TO-5), $106^\circ\text{C}/\text{w}$ (DIP) and $170^\circ\text{C}/\text{w}$ (SO-8) junction to ambient.

Note 2: Supply current when output high typically 1 mA less at $V_{CC} = 5\text{V}$.

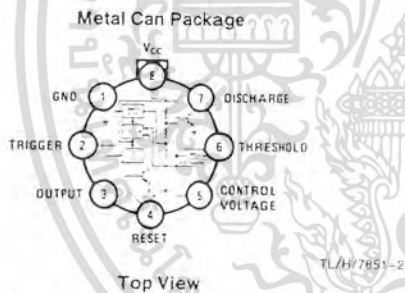
Note 3: Tested at $V_{CC} = 5\text{V}$ and $V_{CC} = 15\text{V}$.

Note 4: This will determine the maximum value of $R_A + R_B$ for 15V operation. The maximum total ($R_A + R_B$) is 20 M Ω .

Note 5: No protection against excessive pin 7 current is necessary providing the package dissipation rating will not be exceeded.

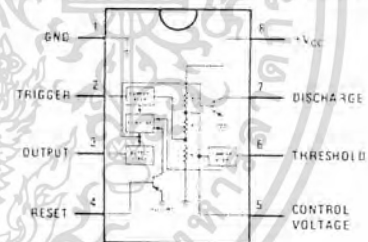
Note 6: Refer to RET555X drawing of military LM555H and LM555J versions for specifications.

Connection Diagrams



Order Number LM555H or LM555CH
See NS Package Number H08C

Dual-In-Line and Small Outline Packages

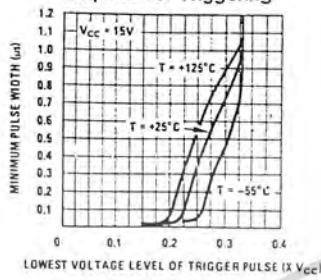


Order Number LM555J, LM555CJ,
LM555CM or LM555CN
See NS Package Number J08A, M08A or N08E

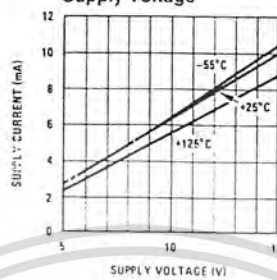
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Typical Performance Characteristics

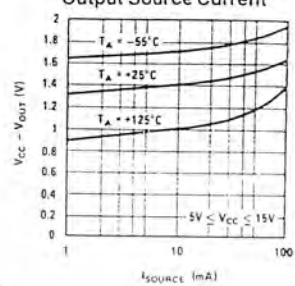
Minimum Pulse Width Required for Triggering



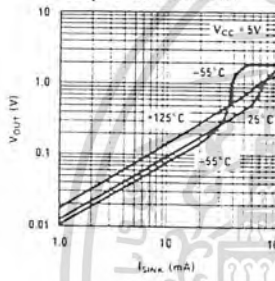
Supply Current vs Supply Voltage



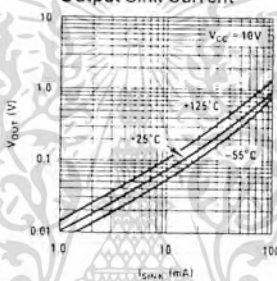
High Output Voltage vs Output Source Current



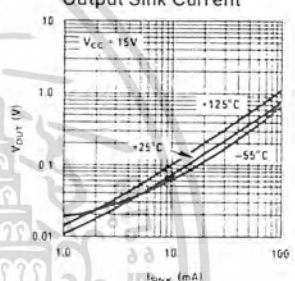
Low Output Voltage vs Output Sink Current



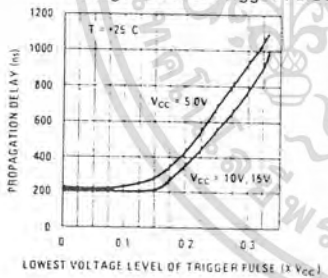
Low Output Voltage vs Output Sink Current



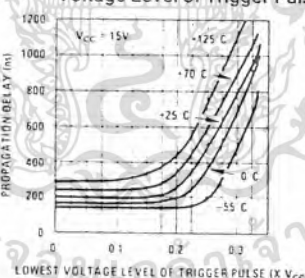
Low Output Voltage vs Output Sink Current



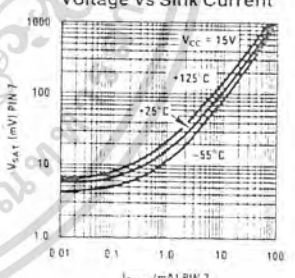
Output Propagation Delay vs Voltage Level of Trigger Pulse



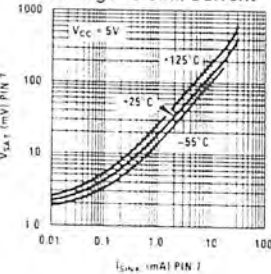
Output Propagation Delay vs Voltage Level of Trigger Pulse



Discharge Transistor (Pin 7) Voltage vs Sink Current



Discharge Transistor (Pin 7) Voltage vs Sink Current



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Applications Information

MONOSTABLE OPERATION

In this mode of operation, the timer functions as a one-shot (Figure 1). The external capacitor is initially held discharged by a transistor inside the timer. Upon application of a negative trigger pulse of less than $1/3 V_{CC}$ to pin 2, the flip-flop is set which both releases the short circuit across the capacitor and drives the output high.

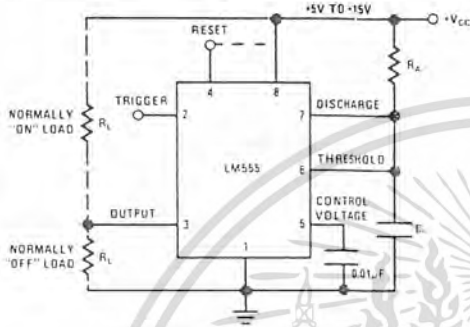
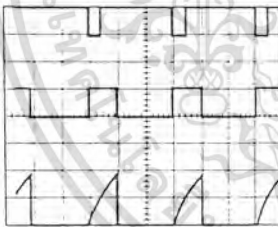


FIGURE 1. Monostable

The voltage across the capacitor then increases exponentially for a period of $t = 1.1 R_A C$, at the end of which time the voltage equals $2/3 V_{CC}$. The comparator then resets the flip-flop which in turn discharges the capacitor and drives the output to its low state. Figure 2 shows the waveforms generated in this mode of operation. Since the charge and the threshold level of the comparator are both directly proportional to supply voltage, the timing interval is independent of supply.



$V_{CC} = 5V$
TIME = 0.1 ms/DIV.
 $R_A = 9.1 k\Omega$
 $C = 0.01 \mu F$

Top Trace: Input 5V/Div.
Middle Trace: Output 5V/Div.
Bottom Trace: Capacitor Voltage 2V/Div

FIGURE 2. Monostable Waveforms

During the timing cycle when the output is high, the further application of a trigger pulse will not effect the circuit so long as the trigger input is returned high at least $10 \mu s$ before the end of the timing interval. However the circuit can be reset during this time by the application of a negative pulse to the reset terminal (pin 4). The output will then remain in the low state until a trigger pulse is again applied.

When the reset function is not in use, it is recommended that it be connected to V_{CC} to avoid any possibility of false triggering.

Figure 3 is a nomograph for easy determination of R, C values for various time delays.

NOTE: In monostable operation, the trigger should be driven high before the end of timing cycle.

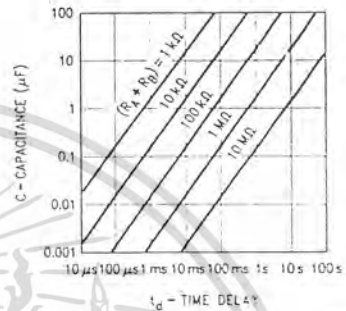


FIGURE 3. Time Delay

ASTABLE OPERATION

If the circuit is connected as shown in Figure 4 (pins 2 and 6 connected) it will trigger itself and free run as a multivibrator. The external capacitor charges through $R_A + R_B$ and discharges through R_B . Thus the duty cycle may be precisely set by the ratio of these two resistors.

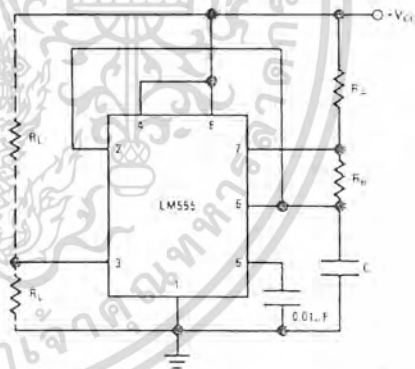
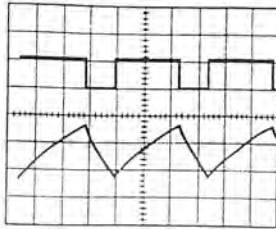


FIGURE 4. Astable

In this mode of operation, the capacitor charges and discharges between $1/3 V_{CC}$ and $2/3 V_{CC}$. As in the triggered mode, the charge and discharge times, and therefore the frequency are independent of the supply voltage.

Applications Information (Continued)

Figure 5 shows the waveforms generated in this mode of operation.



$V_{CC} = 5V$
 TIME = 20 μs /DIV.
 $R_A = 3.9 k\Omega$
 $R_B = 3 k\Omega$
 $C = 0.01 \mu F$

Top Trace: Output 5V/Div.
 Bottom Trace: Capacitor Voltage 1V/Div.

FIGURE 5. Astable Waveforms

The charge time (output high) is given by:

$$t_1 = 0.693 (R_A + R_B) C$$

And the discharge time (output low) by:

$$t_2 = 0.693 R_B C$$

Thus the total period is:

$$T = t_1 + t_2 = 0.693 (R_A + 2R_B) C$$

The frequency of oscillation is:

$$f = \frac{1}{T} = \frac{1.44}{(R_A + 2R_B) C}$$

Figure 6 may be used for quick determination of these RC values.

The duty cycle is:

$$D = \frac{R_B}{R_A + 2R_B}$$

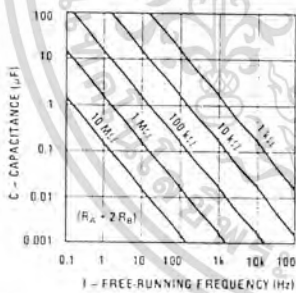
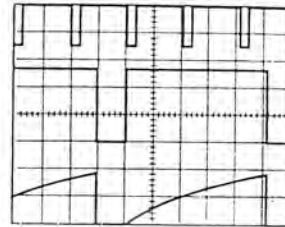


FIGURE 6. Free Running Frequency

FREQUENCY DIVIDER

The monostable circuit of Figure 1 can be used as a frequency divider by adjusting the length of the timing cycle. Figure 7 shows the waveforms generated in a divide by three circuit.



$V_{CC} = 5V$
 TIME = 20 μs /DIV.
 $R_A = 5.1 k\Omega$
 $C = 0.01 \mu F$

Top Trace: Input 4V/Div.
 Middle Trace: Output 2V/Div.
 Bottom Trace: Capacitor 2V/Div.

FIGURE 7. Frequency Divider

PULSE WIDTH MODULATOR

When the timer is connected in the monostable mode and triggered with a continuous pulse train, the output pulse width can be modulated by a signal applied to pin 5. Figure 8 shows the circuit, and in Figure 9 are some waveform examples.

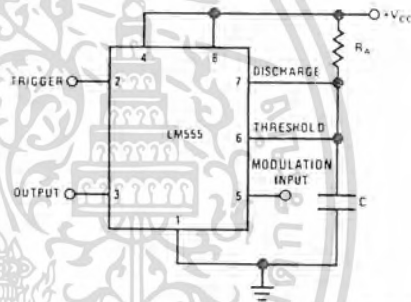
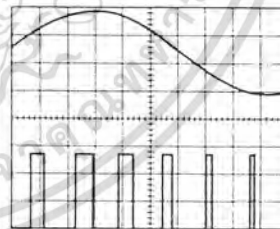


FIGURE 8. Pulse Width Modulator



$V_{CC} = 5V$
 TIME = 0.2 ms/DIV.
 $R_A = 2.1 k\Omega$
 $C = 0.01 \mu F$

Top Trace: Modulation 1V/Div.
 Bottom Trace: Output Voltage 2V/Div.

FIGURE 9. Pulse Width Modulator

PULSE POSITION MODULATOR

This application uses the timer connected for astable operation, as in Figure 10, with a modulating signal again applied to the control voltage terminal. The pulse position varies with the modulating signal, since the threshold voltage and hence the time delay is varied. Figure 11 shows the waveforms generated for a triangle wave modulation signal.

Applications Information (Continued)

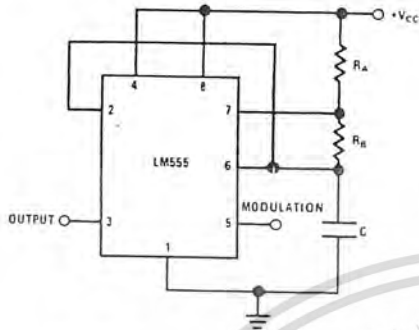


FIGURE 10. Pulse Position Modulator

TL/H/7851-14

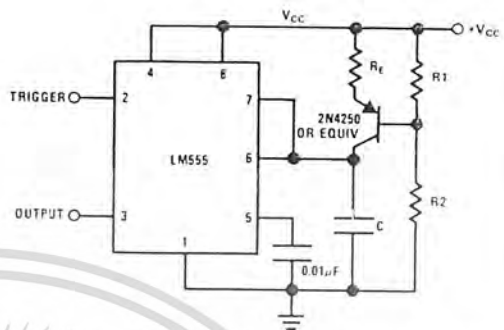
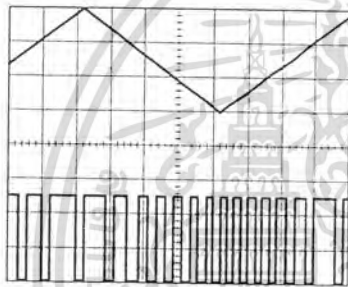


FIGURE 12

TL/H/7851-16



$V_{CC} = 5V$
 TIME = 0.1 ms/DIV
 $R_A = 3.9 k\Omega$
 $R_B = 3 k\Omega$
 $C = 0.01 \mu F$

FIGURE 11. Pulse Position Modulator

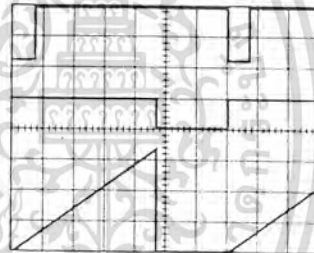
LINEAR RAMP

When the pullup resistor, R_A , in the monostable circuit is replaced by a constant current source, a linear ramp is generated. Figure 12 shows a circuit configuration that will perform this function.

Figure 13 shows waveforms generated by the linear ramp. The time interval is given by:

$$T = \frac{2/3 V_{CC} R_E (R_1 + R_2) C}{R_1 V_{CC} - V_{BE} (R_1 + R_2)}$$

$V_{BE} \approx 0.6V$



$V_{CC} = 5V$
 TIME = 20 μs /DIV
 $R_1 = 47 k\Omega$
 $R_2 = 100 k\Omega$
 $R_E = 2.7 k\Omega$
 $C = 0.01 \mu F$

FIGURE 13. Linear Ramp

TL/H/7851-17

50% DUTY CYCLE OSCILLATOR

For a 50% duty cycle, the resistors R_A and R_B may be connected as in Figure 14. The time period for the out-

Applications Information (Continued)

put high is the same as previous, $t_1 = 0.693 R_A C$. For the output low it is $t_2 =$

$$\left[(R_A R_B) / (R_A + R_B) \right] C \ln \left[\frac{R_B - 2R_A}{2R_B - R_A} \right]$$

Thus the frequency of oscillation is $f = \frac{1}{t_1 + t_2}$

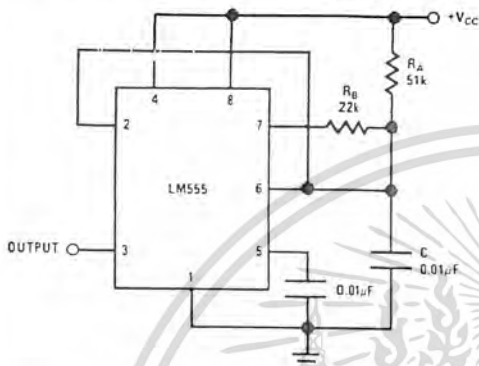


FIGURE 14. 50% Duty Cycle Oscillator

Note that this circuit will not oscillate if R_B is greater than $1/2 R_A$ because the junction of R_A and R_B cannot bring pin 2 down to $1/3 V_{CC}$ and trigger the lower comparator.

ADDITIONAL INFORMATION

Adequate power supply bypassing is necessary to protect associated circuitry. Minimum recommended is $0.1 \mu F$ in parallel with $1 \mu F$ electrolytic.

Lower comparator storage time can be as long as $10 \mu s$ when pin 2 is driven fully to ground for triggering. This limits the monostable pulse width to $10 \mu s$ minimum

Delay time reset to output is $0.47 \mu s$ typical. Minimum reset pulse width must be $0.3 \mu s$, typical.

Pin 7 current switches within $30 ns$ of the output (pin 3) voltage.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ICL232

+5V Powered Dual RS-232 Transmitter/Receiver

Features

- Meets All RS-232C Specifications
- Requires Only Single +5V Power Supply
- Onboard Voltage Doubler/Inverter
- Low Power Consumption
- 2 Drivers
 - ±9V Output Swing for +5V Input
 - 300Ω Power-off Source Impedance
 - Output Current Limiting
 - TTL/CMOS Compatible
 - 30V/μs Maximum Slew Rate
- 2 Receivers
 - ±30V Input Voltage Range
 - 3kΩ to 7kΩ Input Impedance
 - 0.5V Hysteresis to Improve Noise Rejection
- All Critical Parameters are Guaranteed Over the Entire Commercial, Industrial and Military Temperature Ranges

Applications

- Any System Requiring RS-232 Communications Port
 - Computer - Portable and Mainframe
 - Peripheral - Printers and Terminals
 - Portable Instrumentation
 - Modems
 - Dataloggers

Description

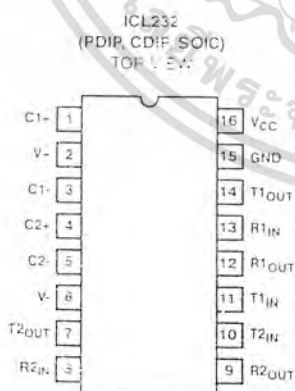
The ICL232 is a dual RS-232 transmitter/receiver interface circuit that meets all EIA RS-232C specifications. It requires a single +5V power supply, and features two onboard charge pump voltage converters which generate +10V and -10V supplies from the 5V supply.

The drivers feature true TTL/CMOS input compatibility, slew-rate-limited output, and 300Ω power-off source impedance. The receivers can handle up to ±30V, and have a 3kΩ to 7kΩ input impedance. The receivers also have hysteresis to improve noise rejection.

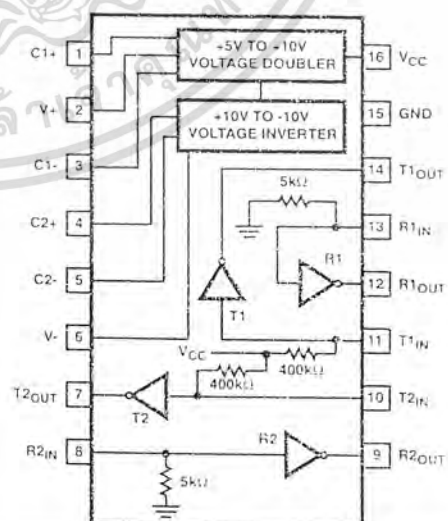
Ordering Information

PART NUMBER	TEMPERATURE RANGE	PACKAGE
ICL232CPE	0°C to +70°C	16 Lead Plastic DIP
ICL232CJE	0°C to +70°C	16 Lead Ceramic DIP
ICL232CBE	0°C to +70°C	16 Lead SOIC (W)
ICL232IFE	-40°C to +85°C	16 Lead Plastic DIP
ICL232IJE	-40°C to +85°C	16 Lead Ceramic DIP
ICL232IBE	-40°C to +85°C	16 Lead SOIC (W)
ICL232MJE	-55°C to +125°C	16 Lead Ceramic DIP

Pinouts



Functional Diagram



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Specifications ICL232

Absolute Maximum Ratings

V_{CC} to Ground	$(GND - 0.3V) < V_{CC} < 6V$
$V+$ to Ground	$(V_{CC} - 0.3V) < V+ < 12V$
$V-$ to Ground	$-12V < V- < (GND + 0.3V)$
Input Voltages	
$T1_{IN}, T2_{IN}$	$(V- - 0.3V) < V_{IN} < (V+ + 0.3V)$
$R1_{IN}, R2_{IN}$	$\pm 30V$
Output Voltages	
$T1_{OUT}, T2_{OUT}$	$(V- - 0.3V) < V_{TXOUT} < (V+ + 0.3V)$
$R1_{OUT}, R2_{OUT}$	$(GND - 0.3V) < V_{RXOUT} < (V_{CC} + 0.3V)$
Short Circuit Duration	
$T1_{OUT}, T2_{OUT}$	Continuous
$R1_{OUT}, R2_{OUT}$	Continuous
Storage Temperature Range	$-65^{\circ}C$ to $+150^{\circ}C$
Lead Temperature (Soldering 10s)	$+300^{\circ}C$

Thermal Information

Thermal Resistance	θ_{JA}	θ_{JC}
Ceramic DIP Package	80°C/W	24°C/W
Plastic DIP Package	100°C/W	-
SOIC Package	100°C/W	-
Maximum Power Dissipation	250mW	
Operating Temperature Range		
ICL232C	0°C to +70°C	
ICL232I	-40°C to +85°C	
ICL232M	-55°C to +125°C	

CAUTION: Stresses above those listed in "Absolute Maximum Ratings" may cause permanent damage to the device. This is a stress only rating and operation of the device at these or any other conditions above those indicated in the operational sections of this specification is not implied.

Electrical Specifications

Test Conditions: $V_{CC} = +5V \pm 10\%$, $T_A =$ Operating Temperature Range. Test Circuit as in Figure 8 Unless Otherwise Specified

PARAMETER	TEST CONDITIONS	LIMITS			UNITS
		MIN	TYP	MAX	
Transmitter Output Voltage Swing, T_{OUT}	$T1_{OUT}$ and $T2_{OUT}$ loaded with 3k Ω to Ground	-5	± 9	± 10	V
Power Supply Current, I_{CC}	Outputs Unloaded, $T_A = +25^{\circ}C$	-	5	10	mA
T_{IN} Input Logic Low, V_{IL}		-	-	0.8	V
T_{IN} Input Logic High, V_{IH}		2.0	-	-	V
Logic Pullup Current, I_p	$T1_{IN}, T2_{IN} = 0V$	-	15	200	μA
RS-232 Input Voltage Range, V_{IR}		-30	-	+30	V
Receiver Input Impedance, R_{IN}	$V_{IN} = \pm 3V$	3.0	5.0	7.0	k Ω
Receiver Input Low Threshold, V_{IL} (H-L)	$V_{CC} = 5.0V, T_A = +25^{\circ}C$	0.8	1.2	-	V
Receiver Input High Threshold, V_{IH} (L-H)	$V_{CC} = 5.0V, T_A = +25^{\circ}C$	-	1.7	2.4	V
Receiver Input Hysteresis, V_{HYST}		0.2	0.5	1.0	V
TTL/CMOS Receiver Output Voltage Low, V_{OL}	$I_{OUT} = 3.2mA$	-	0.1	0.4	V
TTL/CMOS Receiver Output Voltage High, V_{OH}	$I_{OUT} = -1.0mA$	3.5	4.6	-	V
Propagation Delay, t_{PD}	RS-232 to TTL	-	0.5	-	μs
Instantaneous Slew Rate, SR	$C_L = 10pF, R_L = 3k\Omega, T_A = +25^{\circ}C$ (Notes 1, 2)	-	-	30	V/ μs
Transition Region Slew Rate, SR_T	$R_L = 3k\Omega, C_L = 2500pF$ Measured from -3V to -3V or -3V to +3V	-	3	-	V/ μs
Output Resistance, R_{OUT}	$V_{CC} = V+ = V- = 0V, V_{OUT} = \pm 2V$	300	-	-	Ω
RS-232 Output Short Circuit Current, I_{SC}	$T1_{OUT}$ or $T2_{OUT}$ shorted to GND	-	-10	-	mA

NOTES:

1. Guaranteed by design.
2. See Figure 4 for definition.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ICL232

Typical Performance Curves

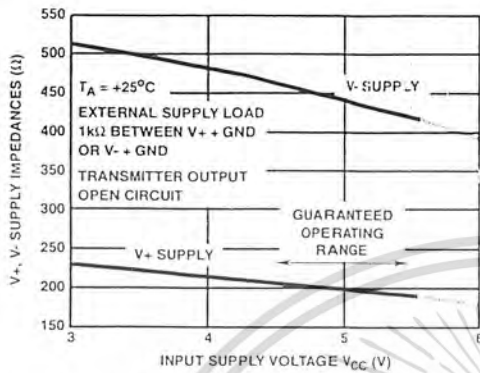


FIGURE 1. V+, V- OUTPUT IMPEDANCES vs V_{CC}

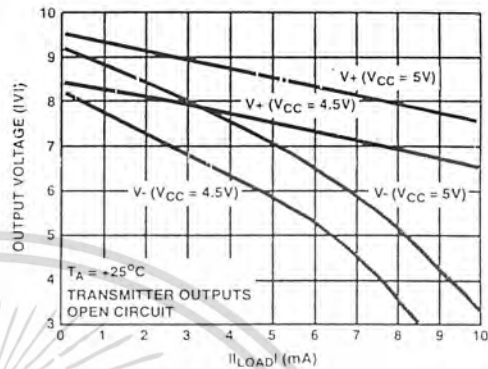


FIGURE 2. V+, V- OUTPUT VOLTAGES vs LOAD CURRENT

Pin Descriptions

PLASTIC DIP, CERAMIC DIP	SOIC	PIN NAME	DESCRIPTION
1	1	C1-	External capacitor "-" for internal voltage doubler.
2	2	V+	Internally generated +10V (typical) supply.
3	3	C1-	External capacitor "-" for internal voltage doubler.
4	4	C2+	External capacitor "+" internal voltage inverter.
5	5	C2-	External capacitor "-" internal voltage inverter.
6	6	V-	Internally generated -10V (typical) supply.
7	7	T2 _{OUT}	RS-232 Transmitter 2 output ±10V (typical).
8	8	R2 _{IN}	RS-232 Receiver 2 input, with internal 5K pull-down resistor to GND.
9	9	R2 _{OUT}	Receiver 2 TTL/CMOS output.
10	10	T2 _{IN}	Transmitter 2 TTL/CMOS input, with internal 400K pull-up resistor to V _{CC} .
11	11	T1 _{IN}	Transmitter 1 TTL/CMOS input, with internal 400K pull-up resistor to V _{CC} .
12	12	R1 _{OUT}	Receiver 1 TTL/CMOS output.
13	13	R1 _{IN}	RS-232 Receiver 1 input, with internal 5K pull-down resistor to GND.
14	14	T1 _{OUT}	RS-232 Transmitter 1 output ±10V (typical).
15	15	GND	Supply Ground.
16	16	VCC	Positive Power Supply +5V ±10%

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Detailed Description

The ICL232 is a dual RS-232 transmitter/receiver powered by a single +5V power supply which meets all EIA RS232C specifications and features low power consumption. The functional diagram illustrates the major elements of the ICL232. The circuit is divided into three sections: a voltage doubler/inverter, dual transmitters, and dual receivers.

Voltage Converter

An equivalent circuit of the dual charge pump is illustrated in Figure 3.

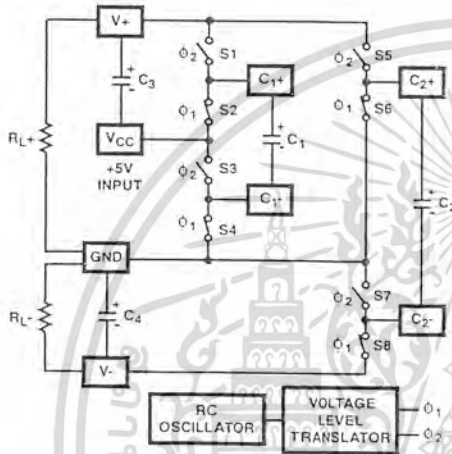
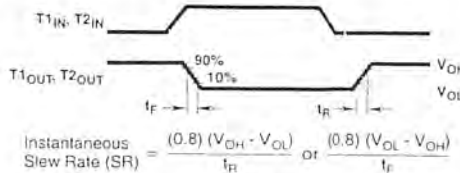


FIGURE 3. DUAL CHARGE PUMP

The voltage quadrupler contains two charge pumps which use two phases of an internally generated clock to generate +10V and -10V. The nominal clock frequency is 16kHz. During phase one of the clock, capacitor C1 is charged to V_{CC}. During phase two, the voltage on C1 is added to V_{CC}, producing a signal across C2 equal to twice V_{CC}. At the same time, C3 is also charged to 2V_{CC}, and then during phase one, it is inverted with respect to ground to produce a signal across C4 equal to -2V_{CC}. The voltage converter accepts input voltages up to 5.5V. The output impedance of the doubler (V₊) is approximately 200Ω, and the output impedance of the inverter (V₋) is approximately 450Ω. Typical graphs are presented which show the voltage converters output vs input voltage and output voltages vs load characteristics. The test circuit (Figure 8) uses 1μF capacitors for C1-C4, however, the value is not critical. Increasing the values of C1 and C2 will lower the output impedance of the voltage doubler and inverter, and increasing the values of the reservoir capacitors, C3 and C4, lowers the ripple on the V₊ and V₋ supplies.



$$\text{Instantaneous Slew Rate (SR)} = \frac{(0.8)(V_{OH} - V_{OL})}{t_r} \text{ or } \frac{(0.8)(V_{OL} - V_{OH})}{t_f}$$

FIGURE 4. SLEW RATE DEFINITION

Transmitters

The transmitters are TTL/CMOS compatible inverters which translate the inputs to RS-232 outputs. The input logic threshold is about 26% of V_{CC}, or 1.3V for V_{CC} = 5V. A logic 1 at the input results in a voltage of between -5V and V₋ at the output, and a logic 0 results in a voltage between +5V and (V₊ - 0.6V). Each transmitter input has an internal 400kΩ pullup resistor so any unused input can be left unconnected and its output remains in its low state. The output voltage swing meets the RS-232C specification of ±5V minimum with the worst case conditions of: both transmitters driving 3kΩ minimum load impedance, V_{CC} = 4.5V, and maximum allowable operating temperature. The transmitters have an internally limited output slew rate which is less than 30V/μs. The outputs are short circuit protected and can be shorted to ground indefinitely. The powered down output impedance is a minimum of 300Ω with ±2V applied to the outputs and V_{CC} = 0V.

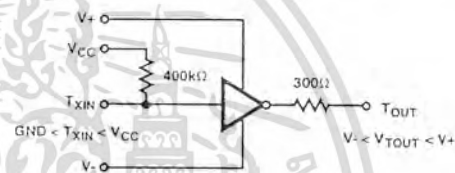


FIGURE 5. TRANSMITTER

Receivers

The receiver inputs accept up to ±30V while presenting the required 3kΩ to 7kΩ input impedance even if the power is off (V_{CC} = 0V). The receivers have a typical input threshold of 1.3V which is within the ±3V limits, known as the transition region, of the RS-232 specification. The receiver output is 0V to V_{CC}. The output will be low whenever the input is greater than 2.4V and high whenever the input is floating or driven between +0.8V and -30V. The receivers feature 0.5V hysteresis to improve noise rejection.

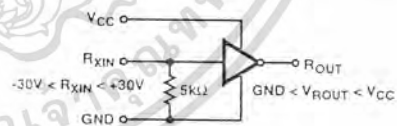
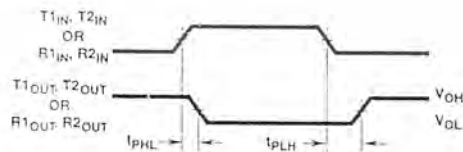


FIGURE 6. RECEIVER



$$\text{Average Propagation Delay} = \frac{t_{PHL} + t_{PLH}}{2}$$

FIGURE 7. PROPAGATION DELAY DEFINITION

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Test Circuits

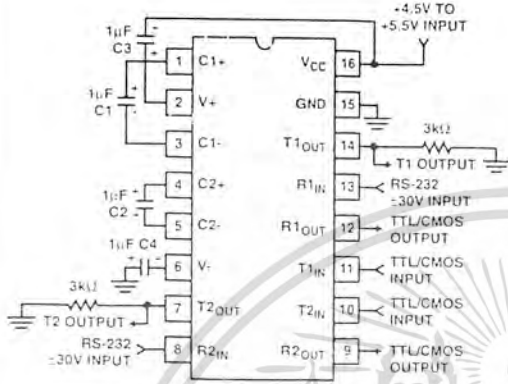


FIGURE 8. GENERAL TEST CIRCUIT

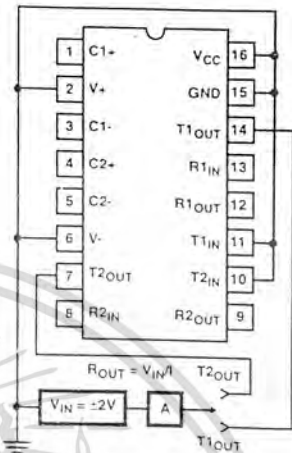


FIGURE 9. POWER-OFF SOURCE RESISTANCE CONFIGURATION

Applications

The ICL232 may be used for all RS-232 data terminal and communication links. It is particularly useful in applications where $\pm 12V$ power supplies are not available for conventional RS-232 interface circuits. The applications presented represent typical interface configurations.

capacitors (C3 and C4). The benefit of sharing common reservoir capacitors is the elimination of two capacitors and the reduction of the charge pump source impedance which effectively increases the output swing of the transmitters.

A simple duplex RS-232 port with CTS/RTS handshaking is illustrated in Figure 10. Fixed output signals such as DTR (data terminal ready) and DSRS (data signaling rate select) is generated by driving them through a 5kΩ resistor connected to V_{-} .

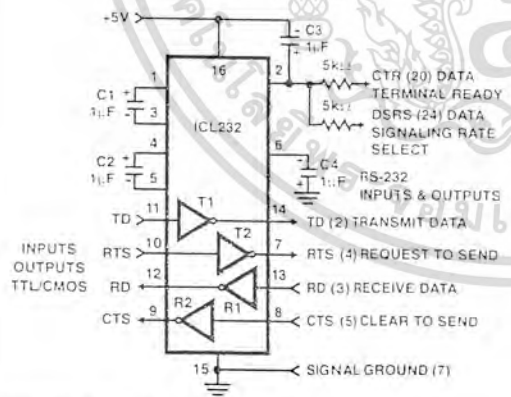


FIGURE 10. SIMPLE DUPLEX RS-232 PORT WITH CTS/RTS HANDSHAKING

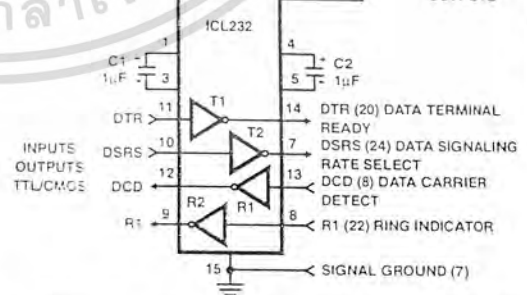
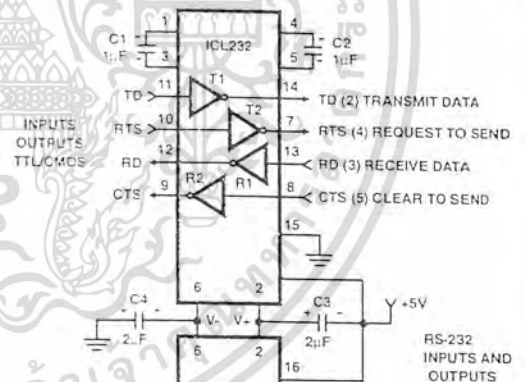


FIGURE 11. COMBINING TWO ICL232s FOR 4 PAIRS OF RS-232 INPUTS AND OUTPUTS

In applications requiring four RS-232 inputs and outputs (Figure 11), note that each circuit requires two charge pump capacitors (C1 and C2) but can share common reservoir

54145 / 74145

BCD-to-Decimal Decoder/Driver with Open-Collector Output (15V Output)

	Schottky TTL				High-Speed TTL				Low-Power Schottky TTL				Standard TTL				Low-Power TTL					
	Device Type		Package		Device Type		Package		Device Type		Package		Device Type		Package		Device Type		Package			
	C	P	M	CF	C	P	M	CF	C	P	M	CF	C	P	M	CF	C	P	M	CF		
T. I.									SN54LS145	J	L	N	W	SN54145	J	L	N	W				
FAIRCHILD									SN74LS145	J	L	N	W	SN74145	J	L	N	W				
MOTOROLA													FM54145	F	M	3145	D					
N. S. C.													FC74145	F	C	93145	D					
PHILIPS													MC54145	L	M							
SIGNETICS									N74LS145				N74145									
SIEMENS									N74LS145	A			S54145	F								
FUJITSU													N74145									
HITACHI									74LS145	M			MB423	J	M							
mitsubishi													HD74145	H	D	2555	T	P				
NEC									M74LS145	P			M53345									
TOSHIBA																						

Electrical Characteristics SN54LS145-SN74LS145

absolute maximum ratings over operating free-air temperature range

Supply voltage, V _{CC}	7V	Operating frequency	SN54LS	55°C to 125°C
Input voltage	7V	temperature range	SN74LS	0°C to 70°C
		Storage temperature range		-65°C to 150°C

recommended operating conditions

	SN54145			SN74145			UNIT
	MIN	NOM	MAX	MIN	NOM	MAX	
Supply voltage, V _{CC}	4.5	5	5.5	4.75	5	5.25	V
Off-state output voltage			15			15	V
Operating free-air temperature, T _a		-55	125		0	70	°C

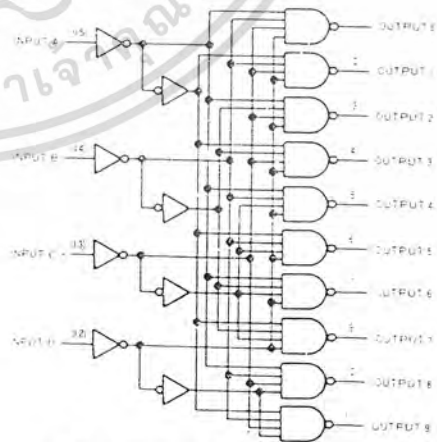
electrical characteristics over recommended operating free-air temperature range

PARAMETER	TEST CONDITIONS*	MIN	TYP†	MAX	UNIT
V _{IH}	High-level input voltage		2		V
V _{IL}	Low-level input voltage		0.8		V
V _I	Input clamp voltage	V _{CC} = MIN, I _I = -10mA		-1.5	V
V _{O(oh)}	On-state output voltage	V _{CC} = MIN, V _{IH} = 2V, I _{O(oh)} = 50µA	2.3		V
V _{O(otr)}	Off-state output voltage	V _{CC} = MIN, V _{IH} = 2V, V _{IL} = 0.8V, I _{O(otr)} = 250µA	0.25	0.5	V
I _I	Input current at maximum input voltage	V _{CC} = MAX, V _I = 5V	0.1		µA
I _{IH}	High-level input current	V _{CC} = MAX, V _I = 2.4V	20		µA
I _{IL}	Low-level input current	V _{CC} = MAX, V _I = 0.4V	0.4		mA
I _{CC}	Supply current	V _{CC} = MAX, SN54LS145 See Note	2	13	mA
t _{PLH}	Propagation delay time, low-to-high-level output	V _{CC} = 5V, T _a = 25°C		50	ns
t _{PHL}	Propagation delay time, high-to-low-level output	C _L = 45pF, R _L = 665Ω		56	ns

FUNCTION TABLE

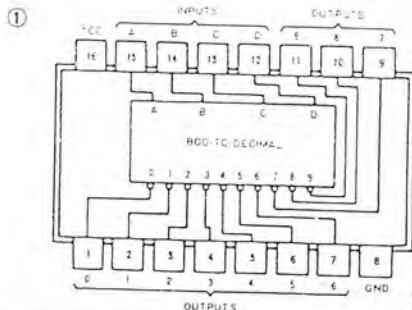
NO	INPUTS				OUTPUTS									
	D	C	B	A	0	1	2	3	4	5	6	7	8	9
0	L	L	L	L	H	H	H	H	H	H	H	H	H	H
1	L	L	L	H	L	H	H	H	H	H	H	H	H	H
2	L	L	H	L	L	H	H	H	H	H	H	H	H	H
3	L	L	H	H	L	L	H	H	H	H	H	H	H	H
4	L	H	L	L	L	L	H	H	H	H	H	H	H	H
5	L	H	L	H	L	L	L	H	H	H	H	H	H	H
6	L	H	H	L	L	L	L	L	H	H	H	H	H	H
7	L	H	H	H	L	L	L	L	L	H	H	H	H	H
8	H	L	L	L	L	L	L	L	L	L	H	H	H	L
9	H	L	L	H	L	L	L	L	L	L	L	H	H	L
10	H	L	H	L	L	L	L	L	L	L	L	L	H	L
11	H	L	H	H	L	L	L	L	L	L	L	L	L	H
12	H	H	L	L	L	L	L	L	L	L	L	L	L	H
13	H	H	L	H	L	L	L	L	L	L	L	L	L	H
14	H	H	H	L	L	L	L	L	L	L	L	L	L	H
15	H	H	H	H	L	L	L	L	L	L	L	L	L	H

Functional Block Diagram



145 BCD-TO-DECIMAL DECODER DRIVER

Pin Assignment (Top View)



* For conditions shown as MIN or MAX, use the appropriate value specified under recommended operating conditions for the applicable type.
 † All typical values are at V_{CC} = 5V, T_a = 25°C.
 NOTE: I_{CC} is measured with all inputs grounded and outputs open.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
 ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

DMC162

● Display Format(16character ×2line) ● Display Fonts(5×8dots) ● Driving method(1/4D)

ABSOLUTE MAXIMUM RATINGS

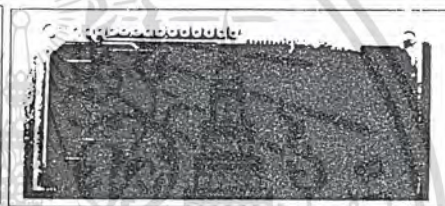
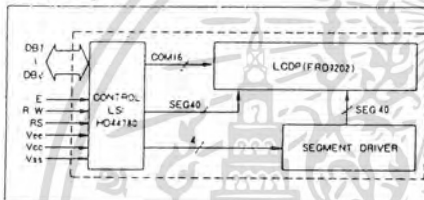
Item	Sym- bol	Test Condition	Standard Value			Unit
			min.	typ.	max.	
Power Supply Voltage for Logic	Vcc - Vss		0		7	V
Power Supply Voltage for LCD Drive	Vcc - Vee		0		13.5	V
Input "High" Voltage	Vi		Vss		Vcc	V
Operating Temperature	Ta		0		+50	°C
Storage Temperature	Tstg		-20		+70	°C

ELECTRICAL CHARACTERISTICS

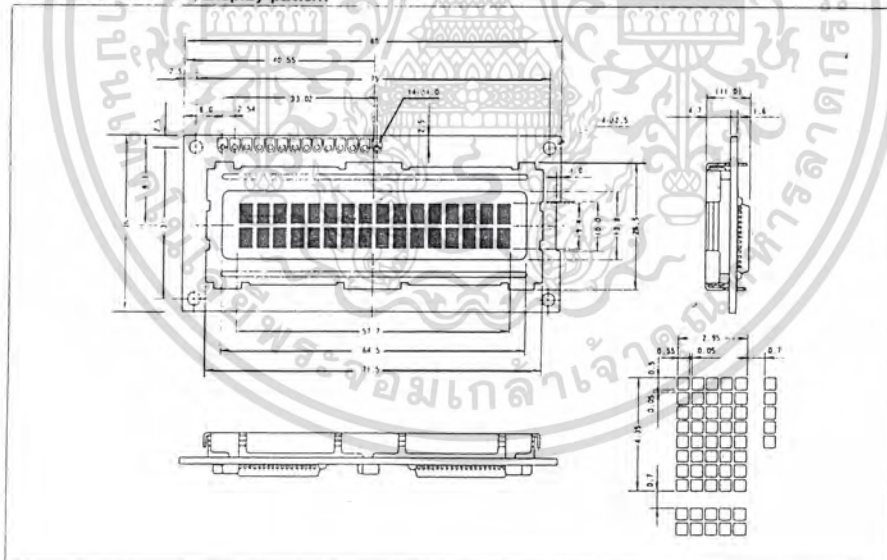
Item	Sym- bol	Test Condition	Standard Value			Unit
			min.	typ.	max.	
Input Voltage	ViH		2.2		Vcc	V
Input "Low" Voltage	ViL		-0.3		0.6	V
Output "High" Voltage	VoH	-IoH=0.205mA	2.4			V
Output "Low" Voltage	VoL	IoL=1.2mA			0.4	V
Power Supply Current	Icc	Vcc=5.0V	0.5		2.0	mA

* Vcc=5.0V ± 5%, Ta=25°C

Block diagram



External dimensions / Display pattern



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ง.

คู่มือการใช้งานเครื่องโปรแกรมสวิตซ์หลายช่อง

1. เมื่อเราเปิดเครื่องครั้งแรกจะทำการตั้งเวลาให้ทำการตั้งค่าของเวลาดังนี้

>	>	-	-	/	-	-	/	-	-	/	-	-
T	i	m	e	<	-	-	>	-	-	:	-	-

1.1. ทำการตั้งค่าของวันในรอบสัปดาห์ ซึ่งจะแสดงเป็น

>	>	S	U	N	/	-	-	/	-	-	/	-	-		
T	i	m	e	<	-	-	>	-	-	:	-	-	:	-	-

SUN หมายถึงอาทิตย์

MON หมายถึงวันจันทร์

TUE หมายถึงวันอังคาร

WED หมายถึงวันพุธ

THU หมายถึงวันพฤหัสบดี

FRI หมายถึงวันศุกร์

SAT หมายถึงวันเสาร์

เมื่อทำการตั้งค่าเสร็จแล้วให้กด ENT หรือถ้าต้องการยกเลิกการตั้งเวลาที่ทำการกดคีย์ ESC

1.2. ทำการตั้งค่าของวัน ซึ่งก็คือ 1 – 30,31 ซึ่งสามารถป้อนได้จากคีย์เลข 0-9 และการกดเพิ่ม INC และลดวัน DEC โดยถ้าตั้งค่าเสร็จแล้วให้กดคีย์ ENT หรือถ้าหากต้องการกลับไปแก้วันในรอบสัปดาห์ใหม่อีกครั้งให้กดคีย์ ESC

>	>	S	U	N	/	0	1	/	-	-	/	-	-		
T	i	m	e	<	-	-	>	-	-	:	-	-	:	-	-

1.3. ทำการตั้งค่าของเดือน โดยการกดคีย์ INC และ DEC ซึ่งแสดงค่าของเดือนดังนี้

>	>	S	U	N	/	0	1	/	J	A	N	/	-	-	
T	i	m	e	<	-	-	>	-	-	:	-	-	:	-	-

เอกสารนี้เป็นเอกสารที่สงวนไว้
ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- JAN หมายถึงเดือนมกราคม
- FEB หมายถึงเดือนกุมภาพันธ์
- MUC หมายถึงเดือนมีนาคม
- APR หมายถึงเดือนเมษายน
- MAY หมายถึงเดือนพฤษภาคม
- JUN หมายถึงเดือนมิถุนายน
- JLY หมายถึงเดือนกรกฎาคม
- AUG หมายถึงเดือนสิงหาคม
- SEP หมายถึงเดือนกันยายน
- OCT หมายถึงเดือนตุลาคม
- NOV หมายถึงเดือนพฤศจิกายน
- DEC หมายถึงเดือนธันวาคม

ถ้าผู้ใช้ตั้งค่าเสร็จแล้วให้กดคีย์ ENT หรือถ้าต้องการกลับไปแก้วันที่ให้กดคีย์ ESC

1.4. ทำการตั้งค่าของปีที่ต้องการ โดยสามารถเลือกจากการกดคีย์ 0-9 หรือกด INC และ DEC ก็ได้ โดยป้อนเฉพาะ 2 หลักสุดท้ายเท่านั้น เช่น ค.ศ. 2001 ก็ป้อน 01 เป็นต้น เมื่อทำการตั้งค่าเสร็จแล้วก็ให้กดคีย์ ENT หรือหากต้องการกลับไปแก้เดือนก็กด ESC

>	>	S	U	N	/	0	1	/	J	A	N	/	0	1
T	i	m	e	<	-	-	>	-	-	:	-	-	:	-

1.5. ทำการตั้งค่าของเวลาที่จะกำหนดให้เป็นเวลาช่วงบ่าย หรือว่าเวลาช่วงเช้า โดยถ้าเลือกช่วงเช้า (Am) คือ เวลา 24.01น. ถึง 12.00น. และช่วงบ่าย (Pm) คือช่วงเวลา 12.01น. ถึง 24.00 น. เมื่อตั้งค่าเสร็จแล้วก็ให้กดคีย์ ENT หรือถ้าต้องการตั้งค่าของปีก็ให้กดคีย์ ESC

>	>	S	U	N	/	0	1	/	J	A	N	/	0	1
T	i	m	e	<	A	m	>	-	-	:	-	-	:	-

1.6. ทำการตั้งค่าของชั่วโมงโดยป้อนค่า 1-12 โดยการป้อนด้วยคีย์ 0-9 หรือคีย์ INC และ DEC เมื่อทำการตั้งค่าเสร็จแล้วให้กดคีย์ ENT ถ้าต้องการแก้ไขค่าของ เวลา ก็ให้กด ESC

>	>	S	U	N	/	0	1	/	J	A	N	/	0	1
T	i	m	e	<	A	m	>	1	2	:	-	-	:	-

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับใช้ภายในเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1.7. ทำการตั้งค่าของนาฬิกา โดยการใช้คีย์ 0-9 หรือคีย์ INC และ DEC เมื่อทำการตั้งเสร็จแล้ว ให้กดคีย์ ENT ถ้าต้องการแก้ไขค่าของชั่วโมง ก็ให้กด ESC

>	>	S	U	N	/	0	1	/	J	A	N	/	0	1
T	i	m	e	<	A	m	>	1	2	:	5	9	:	-

1.8. ทำการตั้งค่าของวินาที โดยการ ใช้คีย์ 0-9 หรือคีย์ INC และ DEC เมื่อทำการตั้งค่าเสร็จแล้ว ให้กดคีย์ ENT ถ้าต้องการแก้ไขค่าของนาฬิกาให้กดคีย์ ESC

>	>	S	U	N	/	0	1	/	J	A	N	/	0	1
T	i	m	e	<	A	m	>	1	2	:	5	9	:	5

2. คีย์ MAN (Manual)

เป็นคีย์ที่ใช้สำหรับควบคุมการทำงานของแต่ละช่อง โดยจะทำการเปิดและปิด โดยตรง วิธีการใช้งานคือ เมื่อกดปุ่ม MAN ที่จอ LCD จะปรากฏคำว่า

M	a	n	a	l	S	y	s	t	e	m
C	H	<	1	-	12	>	:		0	1

ให้ผู้ใช้ทำการเลือกว่าจะทำการควบคุมช่องใด ซึ่งสามารถเลือกได้ทั้งหมด 12 ช่อง คือ 1-12 โดยการกดเลือกช่องได้ตามหมายเลขคีย์บอร์ด หรือสามารถเพิ่มช่อง หรือลดช่องได้โดยกดคีย์ INC หรือ DEC โดยถ้าเลือกกด INC ก็เพิ่มขึ้นหนึ่งช่อง และถ้าเลือกกด DEC ก็ลดช่องลงมาหนึ่งช่อง ถ้าเลือกช่องตามต้องการแล้วให้กดคีย์ ENT ซึ่งคือคีย์ตกลง จากนั้นที่หน้าจอก็จะแสดงดังนี้

S	t	a	t	u	s	:	O	f	f	
C	H	<	1	-	12	>	:	0	1	

ผู้ใช้สามารถเลือกว่าจะให้ ON หรือ OFF ได้โดยการกดปุ่ม INC และ DEC ก็จะแสดงผล ON และ OFF สลับกัน ซึ่งเมื่อทำการเลือกแล้วก็ให้กด ENT ก็เสร็จสิ้นการใช้คีย์ MAN ถ้าหากว่าผู้ใช้เลือกที่จะ ON ที่ช่อง 9 ผู้ใช้จะตั้งค่าของกระแสที่ต้องการจะควบคุมที่อุปกรณ์ก่อน โดยจะแสดงหน้าจอดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

L	i	m	i	t	C	u	r	r	e	n	t		
C	H	<	9	>	:	0	.	0					

ผู้ใช้สามารถเลือกกระแสที่ต้องการได้ตั้งแต่ 0 – 5 แอมป์ โดยสามารถเลือกได้ที่ละ 0.5 แอมป์ คือ 0.5,1,1.5,2 5 เป็นต้น หากผู้ใช้เลือกเสร็จแล้วก็ให้กดคีย์ ENT ก็จะเสร็จสิ้นการใช้คีย์ MAN

3. คีย์ PGM (Program)

เป็นคีย์ที่ใช้สำหรับโปรแกรมเวลาเปิดและปิดของช่องต่างๆ โดยเมื่อกดคีย์ PGM จะแสดงดังนี้

P	r	o	g	r	a	m	S	y	s	t	e	m		
C	H	<	1	-	1	2	>	:	0	1				

ผู้ใช้สามารถเลือกช่องในการโปรแกรมได้โดยการเลือกจากคีย์ 0-9 หรือทำการเพิ่มทีละช่องโดยปุ่ม INC หรือทำการลดทีละช่องโดยปุ่ม DEC ตามแต่ผู้ใช้นัดจากนั้นเมื่อเลือกช่องที่จะทำการโปรแกรมได้กดคีย์ ENT จากนั้นหน้าจอจะแสดงดังนี้

C	h	a	n	n	a	l	<	0	1	>		
S	t	a	t	e	:	1	-	->	O	n		

ให้ผู้ใช้เลือกว่าจะทำการตั้งโปรแกรมที่ state ที่เท่าไร ซึ่งคำว่า state ก็คือเราสามารถที่จะทำการตั้งเวลาเปิดและปิดได้ 2 ช่วงในหนึ่งช่อง เช่น ตั้งเวลา 9.00 น. ถึง 11.00น. และเวลา 15.00น. ถึง 19.00น. เป็นต้น ซึ่ง state 1 ก็คือเวลา 9.00 น.ถึง 11.00น. และ state 2 ก็คือเวลา 15.00น. ถึง 19.00น.

โดยสามารถเลือก state ได้ด้วยคีย์ INC และ DEC โดยจะสลับกันไป เมื่อเลือก state แล้วก็กด ENT จากนั้นก็จะแสดงหน้าจอดังนี้

C	h	a	n	n	a	l	<	0	1	>		
S	t	a	t	e	:	1	-	->	O	f	f	

ให้ทำการเลือกว่าจะ โปรแกรมเวลาในการเปิด ON หรือเวลาในการปิด OFF ซึ่งเลือกด้วย คีย์ INC และ DEC ซึ่งจากนั้นกด ENT

โดยในการเลือก ถ้าผู้ใช้เลือกตั้ง โปรแกรมเปิด ON หรือเลือกตั้งโปรแกรมปิด OFF ลักษณะ การโปรแกรมค่าต่างๆ ก็จะมีลักษณะที่เหมือนกัน คือ

3.1. จะทำการป้อนวันในรอบสัปดาห์ ซึ่งจะแสดงเป็น

0 1 >	S	U	N	/	-	-	/	-	-	-	/	-	-
O f f	<	-	-	>	-	-	:	-	-	:	-	-	

- SUN หมายถึงอาทิตย์
- MON หมายถึงวันจันทร์
- TUE หมายถึงวันอังคาร
- WED หมายถึงวันพุธ
- THU หมายถึงวันพฤหัสบดี
- FRI หมายถึงวันศุกร์
- SAT หมายถึงวันเสาร์
- -- หมายถึงสนใจค่าของวันในรอบสัปดาห์

เมื่อทำการเลือกเสร็จแล้วให้กด ENT หรือถ้าหากจะกลับไปเลือกว่าจะตั้งโปรแกรมเปิด หรือตั้งโปรแกรมปิดก็ทำการกดคีย์ ESC

3.2. เลือกค่าของวัน ซึ่งก็คือ 1 – 30,31 ซึ่งสามารถป้อนได้จากคีย์เลข 0-9 และ -- หรือกด เพิ่ม INC และลดวัน DEC โดยถ้าเลือกเสร็จแล้วให้กดคีย์ ENT หรือถ้าหากต้องการกลับไปแก้วันใน รอบสัปดาห์ใหม่อีกครั้งให้กดคีย์ ESC (หากเราไม่ต้องการสนใจวันก็ให้กดคีย์ ./ --)

0 1 >	S	U	N	/	0	1	/	-	-	-	/	-	-
O f f	<	-	-	>	-	-	:	-	-	:	-	-	

3.3. ทำการเลือกเดือน โดยการกดคีย์ INC และ DEC ซึ่งแสดงค่าของเดือนดังนี้

0 1 >	S	U	N	/	0	1	/	J	A	N	/	-	-
O f f	<	-	-	>	-	-	:	-	-	:	-	-	

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

JAN หมายถึงเดือนมกราคม
 FEB หมายถึงเดือนกุมภาพันธ์
 MUC หมายถึงเดือนมีนาคม
 APR หมายถึงเดือนเมษายน
 MAY หมายถึงเดือนพฤษภาคม
 JUN หมายถึงเดือนมิถุนายน
 JLY หมายถึงเดือนกรกฎาคม
 AUG หมายถึงเดือนสิงหาคม
 SEP หมายถึงเดือนกันยายน
 OCT หมายถึงเดือนตุลาคม
 NOV หมายถึงเดือนพฤศจิกายน
 DEC หมายถึงเดือนธันวาคม

--- หมายถึงไม่สนใจค่าของเดือน

ถ้าผู้ใช้เลือกเสร็จแล้วให้กดคีย์ ENT หรือถ้าต้องการกลับไปแก้วันให้กดคีย์ ESC

3.4. เลือกปีที่ต้องการโดยสามารถเลือกจากการกดคีย์ 0-9 และ -- หรือกด INC และ DEC ก็ได้ โดยป้อนเฉพาะ 2 หลักสุดท้ายเท่านั้น เช่น ค.ศ. 2001 ก็ป้อน 01 เป็นต้น เมื่อทำการเลือกเสร็จแล้วก็ให้กดคีย์ ENT หรือหากต้องการกลับไปแก้เดือนก็กด ESC (หากเราไม่ต้องการสนใจค่าของปี ก็ให้กดคีย์ ./--)

0	1	>	S	U	N	/	0	1	/	J	A	N	/	0	1
O	f	f	<	-	-	>	-	-	:	-	-	:	-	-	

3.5. เลือกค่าของเวลาว่าจะกำหนดให้เป็นเวลาช่วงบ่าย หรือว่าเวลาช่วงเช้า โดยฉันทเลือกช่วงเช้า (Am)

คือ เวลา 24.01น. ถึง 12.00น. และช่วงบ่าย (Pm) คือช่วงเวลา 12.01น. ถึง 24.00น. เมื่อตั้งค่าเสร็จแล้วก็ให้กดคีย์ ENT หรือถ้าต้องการตั้งค่าของปีก็ให้กดคีย์ ESC (หากเราไม่ต้องการสนใจค่าช่วงเวลา ก็ให้เลือก ---)

0	1	>	S	U	N	/	0	1	/	J	A	N	/	0	1
O	f	f	<	A	M	>	-	-	:	-	-	:	-	-	

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่นิยมนำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.6. เลือกค่าของชั่วโมง โดยป้อนค่า 1-12 โดยการป้อนด้วยคีย์ 0-9 หรือคีย์ INC และ DEC เมื่อทำการเลือกได้แล้วให้กดคีย์ ENT ถ้าต้องการแก้ไขค่าของ เวลา ก็ให้กด ESC (หากเราไม่ต้องการสนใจค่าของชั่วโมง ก็ให้กดคีย์ ./--)

0	1	>	S	U	N	/	0	1	/	J	A	N	/	0	1
O	f	f	<	A	M	>	1	2	:	-	-	:	-	-	

3.7. เลือกค่าของนาฬิกา โดยการใช้นคีย์ 0-9 หรือคีย์ INC และ DEC เมื่อทำการเลือกได้แล้วให้กดคีย์ ENT ถ้าต้องการแก้ไขค่าของชั่วโมง ก็ให้กด ESC (หากเราไม่ต้องการสนใจค่าของนาฬิกา ก็ให้กดคีย์ ./--)

0	1	>	S	U	N	/	0	1	/	J	A	N	/	0	1
O	f	f	<	A	M	>	1	2	:	5	9	:	-	-	

3.8. เลือกค่าของวินาที โดยการใช้นคีย์ 0-9 หรือคีย์ INC และ DEC เมื่อทำการเลือกได้แล้วให้กดคีย์ ENT ถ้าต้องการแก้ไขค่าของนาฬิกาให้กดคีย์ ESC (หากเราไม่ต้องการสนใจค่าของวินาที ก็ให้กดคีย์ ./--)

0	1	>	S	U	N	/	0	1	/	J	A	N	/	0	1
O	f	f	<	A	M	>	1	2	:	5	9	:	5	9	

หมายเหตุ

การที่เราไม่สนใจค่าโดยการกด - นั้น หมายถึงว่าในการตรวจสอบเวลาเครื่องจะไม่สนใจค่านั้น โดยถ้าสมมุติว่าเราต้องการให้เครื่องเปิดทุกวันจันทร์ เราก็เลือกเฉพาะ วันในรอบสัปดาห์ให้เป็นค่า MON จากนั้นค่าของปีของวันหรือชั่วโมง นาฬิกา วินาที ก็ให้ใส่ค่า - ไว้ เครื่องก็จะเปิดทุกๆ วันจันทร์นั่นเอง หรือถ้าต้องการให้เปิดทุกๆ 1 นาทีก็ให้เลือกนาฬิกาเปิดที่ 59 วินาที และปิดอาจจะเปิดที่ 15 วินาทีที่เปิด คือตั้งที่ 15 ส่วนค่าอื่นๆ ก็ไม่ต้องตั้งค่าคือใส่ -- เป็นต้น

เมื่อเลือกครบแล้วก็ให้ก็จะกลับไปให้เลือกการโปรแกรมค่าว่าจะโปรแกรมเวลา ON หรือ OFF ถ้าไม่ต้องการก็ให้กด ESC ไปเรื่อยๆ จนกว่าจะเข้าสู่หน้าจอหลักคือหน้าที่แสดงวันเวลาของเครื่องนั่นเอง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4. คีย์ AMP

คีย์นี้ใช้สำหรับกำหนดค่าของกระแสที่ต้องการจะควบคุมซึ่งจะตั้งไว้ที่ช่อง 9 เพียงช่องเดียวเท่านั้น ซึ่งเมื่อเรากดคีย์ AMP ที่หน้าจอก็จะแสดงดังนี้

L	i	m	i	t	C	u	r	r	e	n	t		
C	H									0	.	0	

ให้เราทำการเลือกกระแสที่ต้องการจะควบคุมโดยใช้คีย์ INC และ DEC ซึ่งค่าที่เลือกนั้นจะมีค่าตั้งแต่ 0-5 แอมป์ โดยแต่ละช่วงจะห่างกัน 0.5 แอมป์ และเมื่อเลือกเสร็จแล้วก็ให้กดคีย์ ENT หรือถ้าไม่ต้องการเลือกจะกลับไปสู่หน้าจอหลักก็ให้กดคีย์ ESC ไปจนกว่าจะเข้าสู่หน้าจอหลัก

5. คีย์ CLR

คีย์นี้มีไว้สำหรับเคลียร์หรือปิดทุกๆ ช่องที่ทำงานอยู่ทางเอาต์พุตของเครื่อง โดยเมื่อกดคีย์ CLR จะแสดงหน้าจอดังนี้

C	l	e	a	r	a	l	l	C	H	N	.			
Y	e	s			o	r			N	o	:	Y	e	s

โดยถ้าต้องการที่จะปิดการทำงานทุกช่องก็ให้เลือก Yes หรือถ้าไม่ต้องการปิดการทำงานทุกช่องให้เลือก No โดยสามารถเลือกกดที่คีย์ INC และ DEC เมื่อทำการเลือกเสร็จแล้วก็ให้กดคีย์ ENT หรือถ้าต้องการเข้าสู่หน้าจอหลักก็ให้กดคีย์ ESC ซึ่งการที่เราปิดที่เอาต์พุตของทุกช่องนั้นไม่ได้หมายความว่าค่าที่ตั้งไว้ในโปรแกรมจะหายไปด้วย เพียงแต่ว่าเราจะทำการปิดทุกช่องในขณะนั้นเท่านั้นเอง

6. คีย์ TME (Time)

เป็นการตั้งเวลาของเครื่องโดยวิธีการตั้งเวลานั้นให้อ่านในข้อที่ 1

7. คีย์ RESET

คีย์นี้จะเป็นการเคลียร์ค่าของโปรแกรมที่โปรแกรมไว้ในทุกๆ ช่อง ซึ่งเมื่อทำการกดคีย์นี้จะแสดงที่หน้าจอดังนี้

R	e	s	e	t	P	r	o	g	r	a	m			
Y	e	s			o	r			N	o	:	Y	e	s

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หากเราต้องการที่จะเคลียร์โปรแกรมทั้งหมดทุกช่องให้เลือก Yes และถ้าไม่ต้องการให้
เลือก No หรือกลับสู่หน้าจอหลักกดคีย์ ESC โดยสามารถเลือกกดที่คีย์ INC และ DEC เมื่อทำการ
เลือกเสร็จแล้วก็ให้กดคีย์ ENT

8. คีย์ ENT (Enter)

เป็นคีย์สุดท้ายที่ผู้ใช้กดเพื่อเลือกการทำงานของเมนูต่างๆ ที่ต้องการ

9. คีย์ ESC (Escape)

เป็นคีย์ที่ไว้สำหรับแสดงผลการทำงานย้อนหลังที่ได้ทำการกด ENT ไปแล้ว โดยอ้างอิง
จากหน้าจอหลัก ก็คือถ้าหน้าจอนั้นเป็นหน้าจอหลักอยู่แล้วเมื่อกดคีย์ ESC จะไม่แสดงผลการทำงาน
ย้อนหลัง และยังคงแสดงผลหน้าจอหลักเหมือนเดิม

10. คีย์ 0-9

เป็นคีย์ที่ใช้กำหนดค่าที่ต้องการในเมนูต่างๆ

11. คีย์ . / --

คีย์นี้จะแบ่งเป็น 2 ตัวคือ สำหรับ . นั้นจะใช้ในการกำหนดค่ากระแสที่ต้องการ ส่วน - นั้น
จะใช้ในการตั้งโปรแกรมควบคุมในแต่ละช่อง

12. คีย์ PC

คีย์นี้มีไว้เพื่อใช้รับข้อมูลที่ส่งมาจากคอมพิวเตอร์ โดยผ่านพอร์ตอนุกรม RS 232 ซึ่งข้อ
มูลที่ส่งมาจะอยู่ในรูปรหัส ASCII ซึ่งที่โปรแกรมบนคอมพิวเตอร์นั้นจะมีประโยชน์กับการ
โปรแกรมเวลาในแต่ละช่องมาก ซึ่งสามารถที่จะบันทึกโปรแกรมที่กำหนดไว้ได้ และก็สามารถที่
จะโหลดมาใช้ได้ ในการรับข้อมูลจากคอมพิวเตอร์นั้นมีวิธีการดังนี้

12.1 ทำการกดคีย์ PC จะแสดงหน้าจอดังนี้

P	r	o	g	r	a	m	f	r	o	m	P	C
Y	e	s	o	r	N	o	:	Y	e	s		

หากผู้ใช้ไม่ต้องการรับข้อมูลจากคอมพิวเตอร์ให้เลือก No แล้วกดคีย์ ENT ก็จะกลับสู่หน้า
จอหลัก แต่ถ้าต้องการรับข้อมูลจากคอมพิวเตอร์ก็เลือก Yes เมื่อผู้ใช้เลือกเสร็จแล้วให้กด ENT จาก
นั้นจอจะแสดงดังนี้

P	r	o	g	r	a	m	f	r	o	m	P	C
			C	o	n	n	e	c	t	i	n	g

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ขณะนี้เครื่องพร้อมที่จะรับข้อมูลจากคอมพิวเตอร์แล้ว ให้ทำการเปิดโปรแกรมบนเครื่องคอมพิวเตอร์ที่ใช้ในการส่งข้อมูล จากนั้นก็กดปุ่ม Connect บนวินโดว์ จากนั้นให้สังเกตที่หน้าจอ LCD ของเครื่องจะแสดงดังนี้

P	r	o	g	r	a	m	f	r	o	m	P	C
							R	e	a	d	y	

จากนั้นให้กดปุ่ม Sent บนเครื่องคอมพิวเตอร์ ข้อมูลก็จะถูกส่งมายังเครื่องโปรแกรมสวิตซ์หลายช่อง แล้วเมื่อส่งข้อมูลเสร็จแล้วที่หน้าจอของเครื่องโปรแกรมสวิตซ์หลายช่องก็จะกลับสู่หน้าจอหลัก เป็นการสิ้นสุดการรับข้อมูลจากเครื่องคอมพิวเตอร์



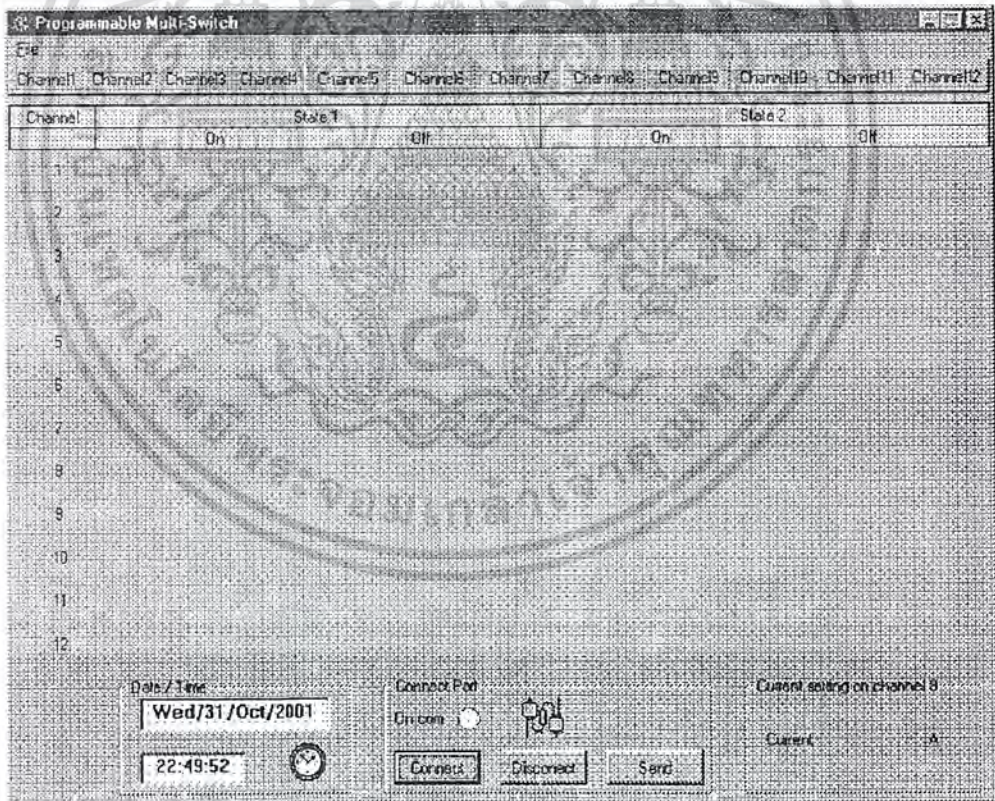
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การใช้งานโปรแกรมบนเครื่อง PC

การใช้งานโดยควบคุมผ่านทางคอมพิวเตอร์โดยใช้โปรแกรม Visual basic 6.0 สามารถที่จะควบคุมการเปิดปิดช่อง(Channel) ได้ 2 state ทั้งหมด 12 ช่องได้โดยง่ายและสะดวกต่อผู้ใช้งาน โดยที่ผู้ใช้งานสามารถที่จะตั้งค่าวันที่และเวลา และในส่วนของช่องที่ 9 สามารถที่จะตั้งค่ากระแสได้อีกด้วย อีกทั้งผู้ใช้งานยังสามารถที่จะตั้งค่าแล้วบันทึกเป็นโปรแกรม โดยในครั้งต่อไปผู้ใช้งานสามารถเปิดโปรแกรมที่ตั้งไว้แล้วส่งออกพอร์ตได้เลย เพื่อที่ผู้ใช้ไม่ต้องมาทำการตั้งค่าในแต่ละช่องทุกครั้งไป

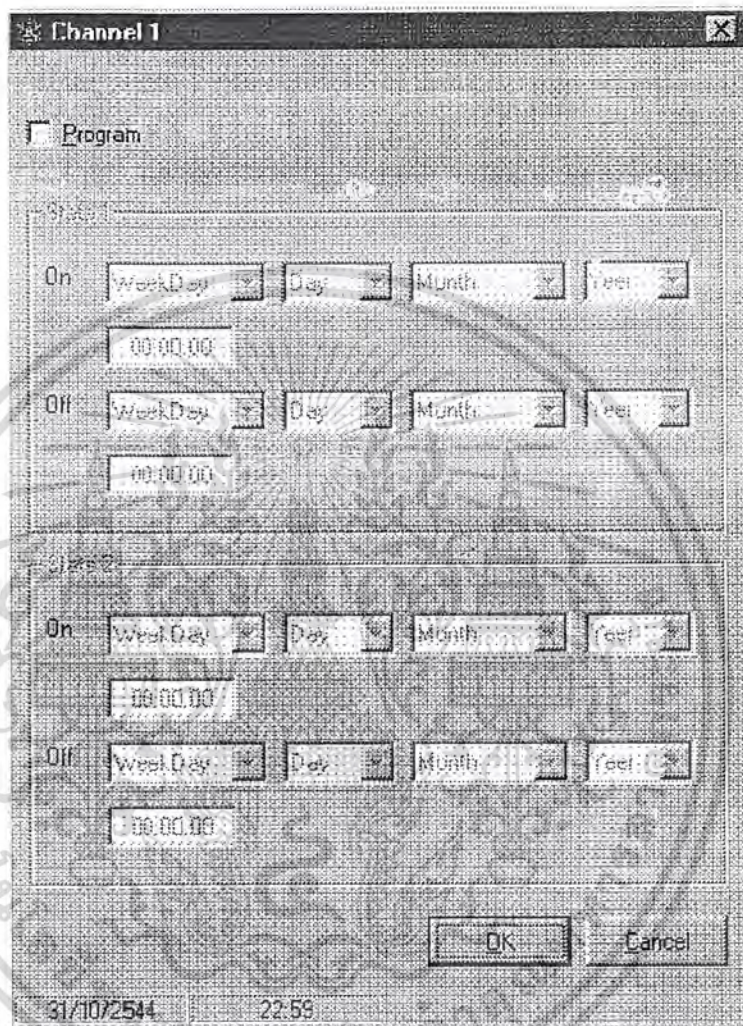
วิธีการใช้งานโปรแกรม

1.) เมื่อผู้ใช้งานเปิดโปรแกรมขึ้นมาจะเห็นส่วนของสถานะ(Status)ของการตั้งค่าในแต่ละช่องดังรูปที่ 1 ผู้ใช้สามารถที่จะเข้าไปตั้งค่าวันที่และเวลาทั้ง 2 state ในแต่ละช่องได้



รูปที่ 1 แสดงสถานะ โปรแกรมในแต่ละช่อง

2.) เมื่อคลิกที่ช่อง 1 สามารถที่จะเข้าไปตั้งค่าวันที่และเวลา โดยที่จะต้องคลิกที่ Check box เอกสารนี้ เกื่อนที่จะตั้งค่างรูปที่ 2 ับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

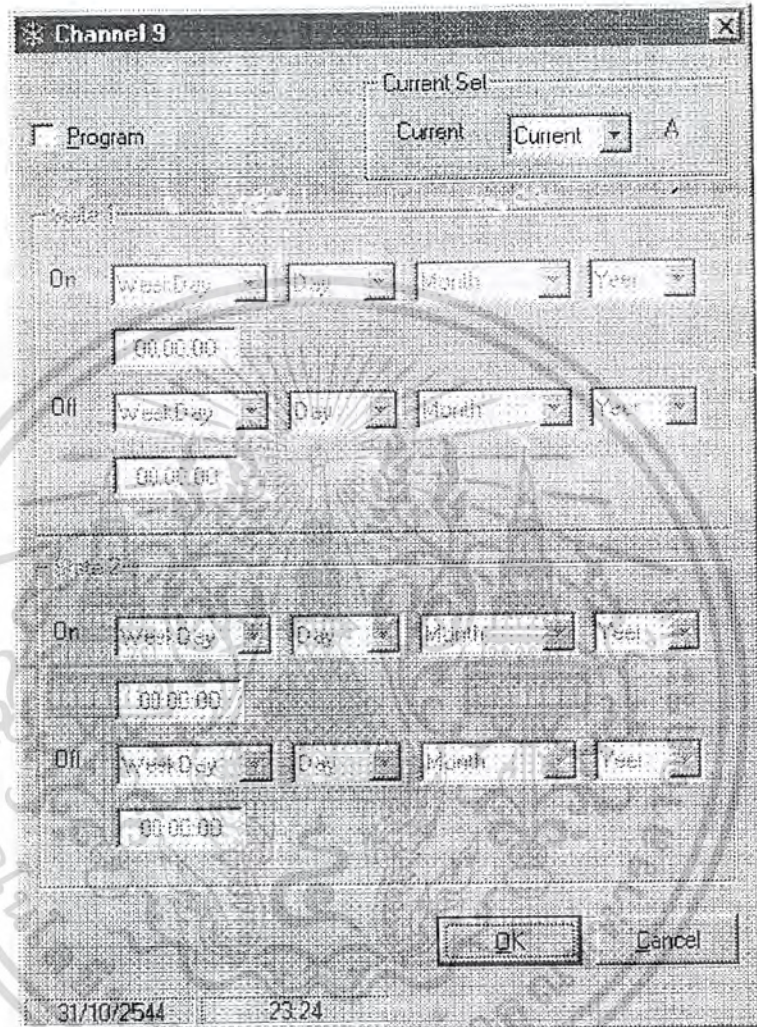


รูปที่ 2 แสดงการตั้งค่าวันที่เวลาทั้ง 2 state

วิธีการตั้งค่าในส่วนของวันที่เมื่อเราต้องการตั้งค่าก็สามารถเลือกได้ใน Combobox และในส่วนของเวลาสามารถตั้งค่าได้เป็นวินาที นาที และชั่วโมง ถ้าหากไม่ต้องการตั้งค่าใดก็ให้ขีดสองขีด เช่น --:--:30, --:20:-- และ 09:--:-- เป็นต้น เมื่อเลือกแล้วให้กดปุ่ม OK

3. การตั้งค่าจำกัดกระแสจะสามารถกระทำได้ในช่องที่ 9 เท่านั้นดังรูปที่ 3 คลิกไปที่ ช่องที่ 9 โดยที่ สามารถเลือก ได้ตั้งแต่ 0.0 A จนถึง 5.0 A เมื่อเลือกแล้วให้คลิกปุ่ม OK

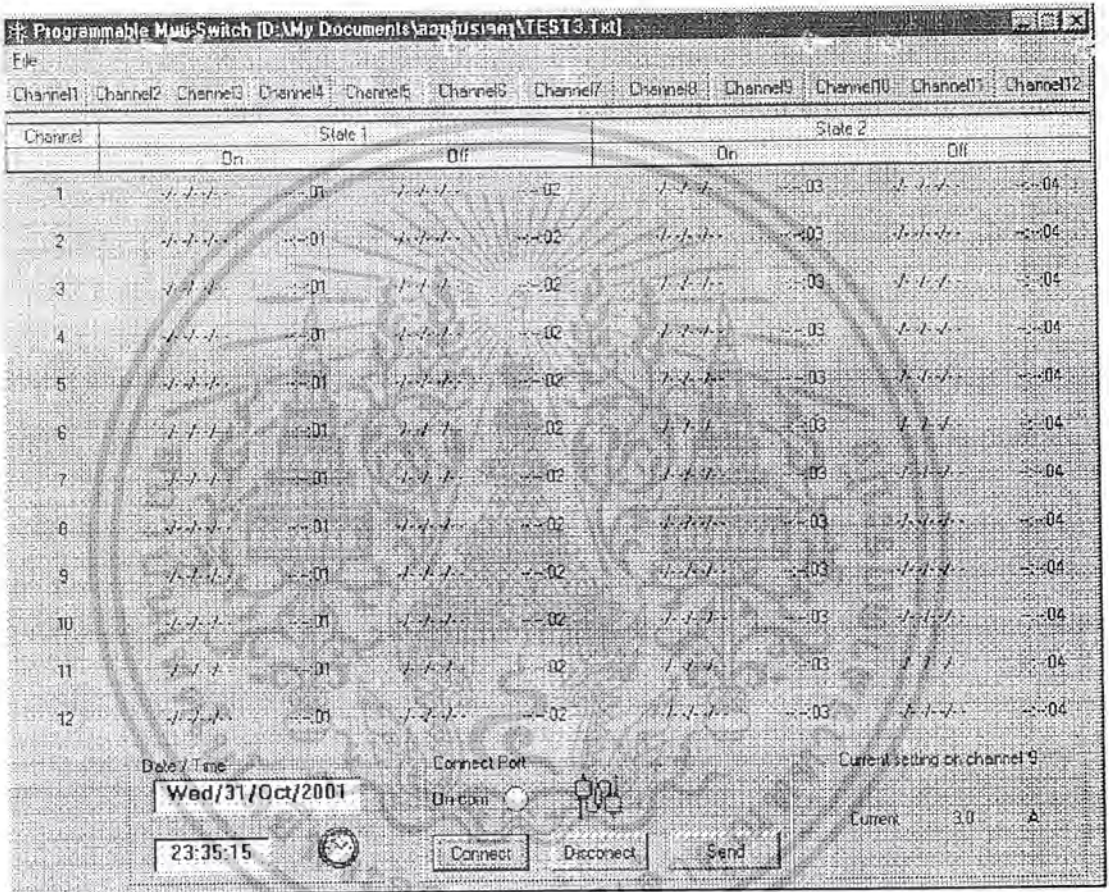
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3 แสดงการตั้งค่าจำกัดกระแสในช่องที่ 9

4. เมื่อเราตั้งค่าได้ในแต่ละช่องจะสามารถทราบค่าสถานะในแต่ละช่องแสดงได้ดังรูปที่ 4 เมื่อตั้งค่าตามต้องการได้แล้วมาที่ส่วนหน้าต่างสถานะ สามารถที่จะคลิกที่ปุ่ม Connect เพื่อทำการติดต่อกับพอร์ต จากนั้นทำการส่งออกพอร์ตโดยคลิกปุ่ม Send ส่งค่าวันที่และเวลาในแต่ละช่องออกพอร์ต ถ้าหากคลิกปุ่ม Disconnect ทำการปิดพอร์ต

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4 แสดงสถานะการตั้งค่าในแต่ละช่องและการตั้งค่ากระแสในช่องที่ 9

5. ผู้ใช้งานสามารถที่จะบันทึกโปรแกรมตั้งค่าวันที่ เวลา และกระแสได้ โดยการคลิกเลือกที่เมนู File แล้วเลือก Save แล้วตั้งชื่อเช่น Prog1 , Prog2,Prog3 เป็นต้น ซึ่งจะบันทึกเป็นไฟล์นามสกุล Text(Txt) แล้วกด Save ในกรณีต้องการเปิด โปรแกรมที่เราได้ตั้งค่าไว้แล้วก็ให้ไปคลิกเลือกที่เมนู File แล้วเลือก Open แล้วเลือกไฟล์ที่ต้องการ อีกทั้งยังสามารถที่ เลือก New และ Save As ได้ อีกด้วย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้