เครื่องควบคุมระยะไกลเอนกประสงค์แบบโปรแกรมได้

PROGRAMMABLE UNIVERSAL REMOTE CONTROL

โดย

นาย บัญชา เศรษฐนันท์ 33100185

ปริญญานิพนธ์นี้ เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตร์

สาขา วิศวกรรมอิเล็กทรอนิกส์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

033207

ปริญญานิพนธ์ปีการศึกษา 2536

ภาควิชาอิเลคทรอนิกส์

คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เรื่อง เครื่องควบคุมระยะไกลเอนกประสงค์แบบโปรแกรมได้

ผู้จัดทำ : นาย บัญชา เศรษฐนันท์ รหัสประจำตัว 33100185

..... อาจารย์ที่ปรึกษา

(อ.เทอดศักดิ์ ลิ่วหาทอง)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เครื่องควบคุมระยะไกลเอนกประสงค์แบบโปรแกรมได้

Programmable Universal Remote Control

โดย

นาย บัญชา เศรษฐนันท์ 33100185

อ.เทอดศักดิ์ ลิ่วหาทอง อาจารย์ที่ปรึกษา

ปีการศึกษา 2536

บทคัดย่อ

เครื่องควบคุมระยะไกลเอนกประสงค์แบบโปรแกรมได้เครื่องนี้ เป็นเครื่องควบคุมระยะไกลแบบที่ใช้สัญญาณอินฟราเรดในการส่งสัญญาณเพื่อควบคุมอุปกรณ์ต่างๆโดยที่สามารถ บันทึกสัญญาณควบคุมของเครื่องควบคุมระยะไกลแบบที่ใช้สัญญาณอินฟราเรดในการควบคุมอื่นๆได้ 118ช่องสัญญาณสามารถโปรแกรมเวลาในการส่งสัญญาณไปควบคุมอุปกรณ์ต่างๆได้มีจอ LCD ใช้สำหรับแสดงสถานะ และ แสดงผลในการทำงานเพื่อความสะดวกในการใช้งาน จะเห็นว่า เครื่องควบคุมระยะไกลเอนกประสงค์แบบโปรแกรมได้ให้ความสะดวกแก่ผู้ใช้ มากกว่าการที่จะใช้เครื่องควบคุมระยะไกลหลายๆเครื่องในการควบคุมอุปกรณ์ไฟฟ้าๆอุปกรณ์

ABSTRACT

Programmable Universal Remote Control is the remote control which uses infrared signal to control any other equipment. The ability of The Programmable Universal Remote Control is that can receive other infrared signals for control other equipment.It can be recorded up to 118 channel for control other equipment and it can program the time to control an equipment. The LCD display will help the user to use it more esier.Finally The Programmable Universal Remote Control will adventages to reduce any remote control to the one.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

คำนำ

ในปัจจุบันนี้เราจะเห็นได้ว่าสิ่งอำนวยความสะดวกสบายต่างๆ จะเข้ามามีบทบาทในการใช้ชีวิตประจำวันมากขึ้น ในอุปกรณ์เครื่องใช้ไฟฟ้าที่ใช้ในชีวิตประจำวันก็เช่นกัน ทางบริษัท ผู้ผลิตพยายามที่จะเพิ่มความสะดวกสบายในการใช้งานเครื่องใช้ไฟฟ้าต่างๆ ให้สามารถใช้งานได้ง่ายขึ้น อุปกรณ์ควบคุมระยะไกลก็เป็นสิ่งหนึ่งที่ทางบริษัทผู้ผลิตต่างๆ ได้ทำขึ้น เพื่อความสะดวกในการใช้งาน

ในการใช้งานอุปกรณ์เครื่องใช้ไฟฟ้านั้นเครื่องควบคุมระยะไกล 1 เครื่อง สามารถที่จะใช้ควบคุมเครื่องใช้ไฟฟ้าได้เพียง 1 ชนิดเท่านั้น ถ้าเรามีเครื่องใช้ไฟฟ้า 2 ชนิด ๆ ละ 1 เครื่อง ในการใช้งานเราจะมีเครื่องควบคุมระยะไกลถึง 2 เครื่อง ซึ่งอาจจะไม่สะดวกในการใช้งาน เพื่อความสะดวกในการใช้งานเครื่องควบคุมระยะไกลเอนกประสงค์แบบโปรแกรมได้ จะช่วยแก้ปัญหานี้ได้โดยการบันทึกสัญญาณควบคุมของเครื่องใช้ไฟฟ้าทุกตัวไว้ที่เครื่องนี้เวลาจะใช้งานก็ใช้เครื่องควบคุมเอนกประสงค์แบบโปรแกรมได้เครื่องนี้ เครื่องเดียวในการควบคุมเครื่องใช้ไฟฟ้าทั้ง 6 เครื่อง นอกจากนั้นเครื่องควบคุมแบบโปรแกรมได้เครื่องนี้ ยังสามารถตั้งเวลาในการส่งสัญญาณควบคุมไปควบคุมอุปกรณ์ได้

สุดท้ายนี้ หวังว่าโครงงานนี้คงเป็นประโยชน์ต่อบุคคลทั่วไปบ้างไม่มากก็น้อยและหากมีข้อผิดพลาดประการใด ผู้จัดทำก็ต้องขออภัยไว้ ณ ที่นี้ด้วย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ

เรื่อง		หน้า
บทที่ 1	บทนำ	1
บทที่ 2	ทฤษฎีและหลักการทำงานพื้นฐานของโครงงาน	4
บทที่ 3	การออกแบบและการสร้าง	59
บทที่ 4	การทดลองและผลการทดลอง	70
บทที่ 5	สรุปและวิจารณ์	71

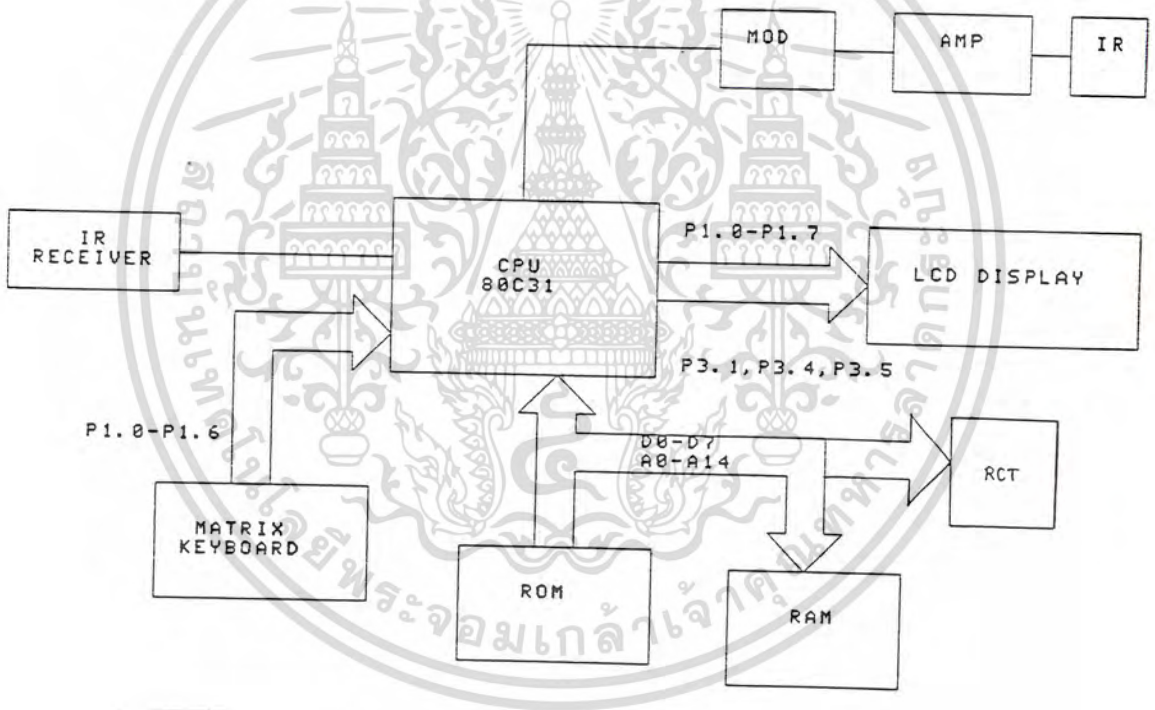
ภาคผนวก
- หนังสืออ้างอิง
- วงจรที่ใช้ในโครงงาน
- ลักษณะโครงสร้างของโปรแกรม
- การใช้งานเครื่องควบคุมระยะไกลเอนกประสงค์แบบโปรแกรมได้
- โปรแกรม
- กิตติกรรมประกาศ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 1

บทนำ

เครื่องควบคุมระยะไกลเอนกประสงค์แบบโปรแกรมได้นี้เป็นเครื่องควบคุมระยะไกลที่ใช้ไมโครคอนโทรลเลอร์เบอร์ 80C31 เป็นส่วนควบคุมซึ่งเครื่องควบคุมระยะไกลเครื่องนี้สามารถทำงานได้โดยไม่ต้องเชื่อมต่อกับอุปกรณ์ภายนอกอีก โครงสร้างของเครื่องดังบล็อกไดอะแกรมในรูปที่ 1.1



รูปที่ 1.1 แสดงบล็อกไดอะแกรมของเครื่องควบคุมระยะไกลเอนกประสงค์แบบโปรแกรมได้

จาก รูปที่ 1.1 จะเห็นได้ว่าโครงสร้างของเครื่องควบคุมระยะไกลเอนกประสงค์แบบโปรแกรมได้จะประกอบไปด้วยส่วนต่างๆดังต่อไปนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1.1 หน่วยประมวลผล (CPU)

เป็นส่วนที่ ทำหน้าที่ในการควบคุม และ ประมวลผลการทำงานของระบบทั้งหมด ที่อยู่ในโครงงานนี้ เราได้ใช้ไมโครคอนโทรลเลอร์ตระกูล MCS-51 โดยเลือกใช้เบอร์ 80C31 ซึ่งเป็นหน่วยประมวลผลแบบ 8 บิท

1.2 หน่วยความจำ

ในโครงงานนี้ใช้หน่วยความจำ 2 ชนิด คือ

1.หน่วยความจำแบบรอม ใช้เป็นโปรแกรมมอนิเตอร์

2.หน่วยความจำแบบแรม ใช้ในการเก็บข้อมูลของสัญญาณควบคุมระยะไกล

1.3 หน่วยสร้างสัญญาณพาหะ

ใช้วงจรสร้างสัญญาณความถี่พาหะ เพื่อใช้ในการมอดดูเลทกับ ข้อมูลของสัญญาณควบคุมที่ต้องการจะส่งออกไปควบคุมอุปกรณ์ภายนอกซึ่งในโครงงานนี้ จะใช้ วงจรสร้างความถี่ 50.1KHz

1.4 ส่วนรับ-ส่งสัญญาณควบคุม

เป็นส่วนที่ทำหน้าที่ในการรับ-ส่งสัญญาณควบคุมไปยังอุปกรณ์ ที่เราต้องการจะควบคุม โดยจะส่งไปในรูปของสัญญาณอินฟราเรด (ความยาวคลื่นประมาณ 850-1000 นาโนเมตร)

1.5 หน่วยแสดงผล

เป็นส่วนที่ทำหน้าที่แสดงถึงสถานะการทำงานของระบบว่า กำลังทำงานอยู่ในฟังก์ชันใดโดยในโครงงานนี้เราเลือกใช้จอภาพแบบ LCD ในการแสดงผล

1.6 นาฬิกาแสดงเวลาจริง (REAL TIME CLOCK)

ทำหน้าที่ในการแสดงเวลาและกำหนดเวลาในการส่งสัญญาณควบคุมไปควบคุม อุปกรณ์ต่างๆ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1.7 แหล่งจ่ายไฟ

ส่วนภาคจ่ายไฟที่ป้อนให้กับระบบต่างๆ ภายในเครื่องควบคุมระยะไกลแบบโปรแกรมได้ ทำงานได้นั้นเราใช้ไฟจาก แบตเตอรี่ 9 โวลต์ 2 ก้อนผ่านไอซีเรคกูเรเตอร์ 7805

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 2

ทฤษฎีและหลักการทำงานพื้นฐานของโครงงาน

ทฤษฎีและหลักการทำงานพื้นฐานของโครงงานแบ่งเป็นหัวข้อย่อยดังนี้

2.1 ความรู้เกี่ยวกับสัญญาณอินฟราเรด

2.2 โครงสร้างและหลักการทำงานของไมโครคอนโทรลเลอร์ 8031

2.3 โครงสร้างและหลักการทำงานของหน่วยแสดงผล

2.4 โครงสร้างของส่วนสัญญาณพาหะ

2.1 ความรู้พื้นฐานเกี่ยวกับสัญญาณ อินฟราเรด (Infrared)

สัญญาณอินฟราเรดนั้น เป็นส่วนหนึ่งของ สเปกตรัมแสง (Light Spectrum)ซึ่งจะมีความยาวคลื่นอยู่ในย่านของแสงที่ตามนุษย์ที่จะมองไม่สามารถมองเห็นได้ (มีค่าความยาวคลื่นประมาณ 850-1000 นาโนเมตร) โดยความยาวคลื่นของแสงที่ตามนุษย์จะมองเห็นได้จะอยู่ในช่วงประมาณ 400-700 นาโนเมตร สำหรับสัญญาณที่ใช้ในเครื่องควบคุมระยะไกล มีอยู่ 3 แบบคือ

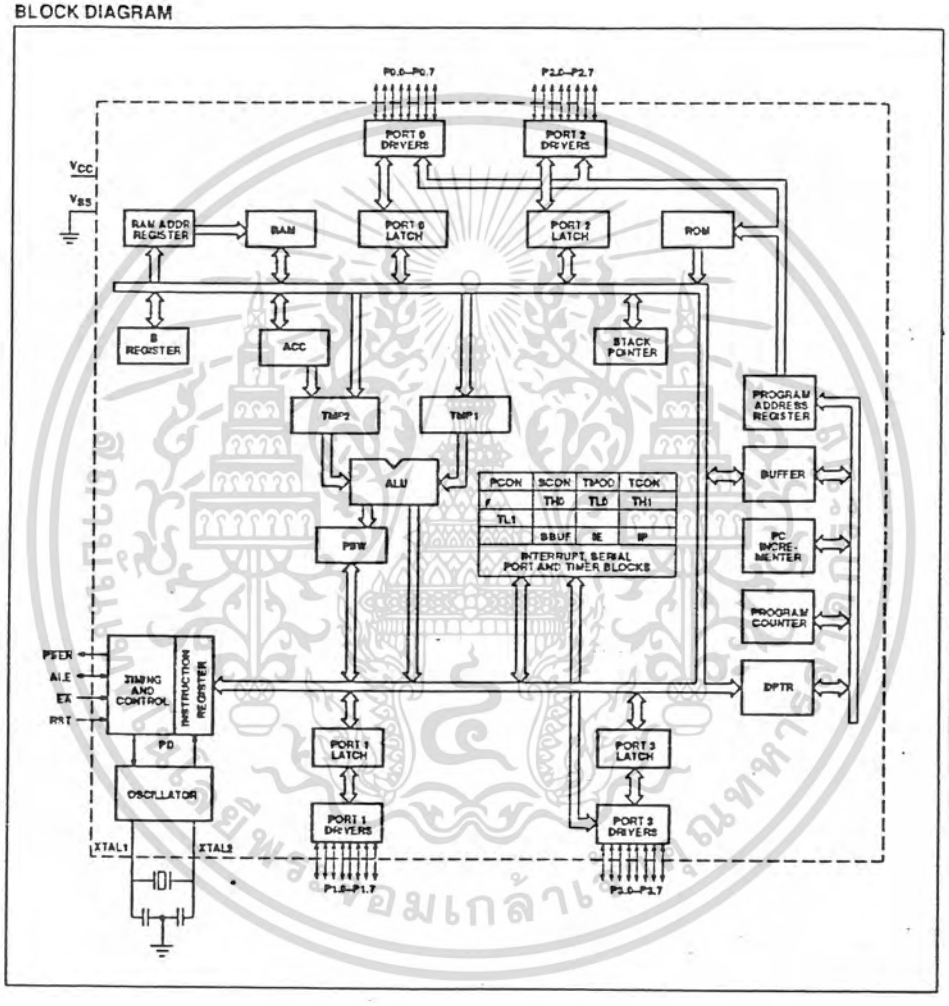
1. เครื่องควบคุมระยะไกลแบบที่ใช้สัญญาณอุลตร้าโซนิก
2. เครื่องควบคุมระยะไกลแบบที่ใช้สัญญาณอินฟราเรด
3. เครื่องควบคุมระยะไกลแบบที่ใช้สัญญาณคลื่นวิทยุ

สำหรับเครื่องควบคุมระยะไกลแบบที่ใช้สัญญาณอุลตร้าโซนิกค่อนข้างจะล้าสมัยเครื่องควบคุมระยะไกลแบบนี้มีใช้ในเครื่องรับโทรทัศน์รุ่นเก่าๆ ที่นิยมใช้ในปัจจุบันเป็นแบบที่ใช้สัญญาณอินฟราเรดและคลื่นวิทยุ ในที่นี้จะขอกล่าวเฉพาะเครื่องควบคุมระยะไกลแบบที่ใช้สัญญาณอินฟราเรดอุปกรณ์ที่จะติดต่อสื่อสารด้วยนั้นต้อง เป็นอุปกรณ์ที่สามารถจะส่งหรือตรวจจับสัญญาณแบบอินฟราเรดได้ ตัวอย่างเช่น TIL38,TIL39,TIL100 MRD821 เป็นต้นลักษณะของสัญญาณที่มีการส่งแบบอินฟราเรดนี้จะมีลักษณะเป็นพัลส์ที่มีความกว้างแตกต่างกันส่งออกมาแบบอนุกรม คือ ข้อมูลจะเรียงต่อๆ กันมาตามลำดับทำให้ได้ชุดสัญญาณที่ซ้ำๆกัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.2 โครงสร้างและหลักการทำงานของไมโครคอนโทรลเลอร์ 8031

8031 เป็นไมโครคอนโทรลเลอร์แบบชิพเดี่ยวซึ่งอยู่ในตระกูล MCS-51ลักษณะโครงสร้าง 8031 แสดงอยู่ในรูปที่ 2.1ซึ่งสถาปัตยกรรมของ 8031 สร้างขึ้นด้วย HMOS



รูปที่ 2.1 สถาปัตยกรรมของ 8031

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.2.1 ลักษณะของ 8031

-เป็น CPU แบบ 8 บิต

-มีวงจรออสซิลเลเตอร์และ CLOCK อยู่ในตัว

-มีขาสัญญาณเข้าและออก(I/O)32 ขา

-แยกหน่วยความจำของข้อมูลได้ 64K และหน่วยความจำโปรแกรมอีก 64K

-มี TIMER และ COUNTER แบบ 16 บิต ถึง 2 ตัว

-สัญญาณอินเทอรัพท์ 6 แหล่ง 5 VECTOR ซึ่งแบ่งระดับความสำคัญออกเป็น 2 ระดับ

-การทำงานแบบ FULL DUPLEX ในขณะส่งข้อมูลแบบอนุกรม

-มีการประมวลผลแบบบูลีน(AND,OR,XOR)ฯลฯ

2.2.2 การจัดหน่วยความจำของ 8031

8031 แบ่งหน่วยความจำสำหรับโปรแกรมและข้อมูลอย่างละ 64K และ ยังมีหน่วยความจำประเภทแรมอีก 128 ไบต์ ซึ่งอยู่ในตัวของ 8031 และบน 8031 ยังประกอบด้วยรีจิสเตอร์ที่ทำหน้าที่พิเศษ "SFR" (SPECIAL FUNCTION REGISTER) ซึ่งแสดงในตารางที่ 2.1

ตารางที่ 2.1

SYMBOL	NAME	ADDRESS
*ACC	ACCUMULATOR	OEOH
*B	B REGISTER	OFOH
*PSW	PROGRAM STSTUS WORD	ODOH
SP	STACK POINTER	81H
DPH	DATA POINTER(CONSIS OF DPH,DPL)	83H
DPL		82H
*PO	PORT O	80H

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ต่อจากตารางที่ 2.1

SYMBOL	NAME	ADDRESS
*P1	PORT 1	90H
*P2	PORT 2	0A0H
*P3	PORT 3	0B0H
*IP	INTERRUPT PIORITY CONTROL	0B8H
*IE	INTERRUPT ENABLE CONTROL	0A8H
TMOD	TIMER/COUNTER MODE CONTROL	89H
+*TCON2	TIMER/COUNTER 2 CONTROL	0C8H
TCON	TIMER/COUNTER CONTROL	88H
TH0	TIMER 0 (HIGH BYTE)	8CH
TL0	TIMER 0 (LOW BYTE)	8AH
TH1	TIMER 1 (HIGH BYTE)	8DH
TL1	TIMER 1 (LOW BYTE)	8BH
+TH2	TIMER 2 (HIGH BYTE)	0CDH
+TL2	TIMER 2 (LOW BYTE)	0CCH
+RCAP2H	TIMER/COUNTER2 CAPTURE (HI)	0CBH
+RCAP2L	TIMER/COUNTER2 CAPTURE (LO)	0CAH
*SCON	SERIAL CONTROL	98H
SBUF	SERIAL DATA BUFFER	99H
PCON	POWER CONTROL	87H

หมายเหตุ *SFR ที่มีเครื่องหมายดอกจันอยู่ข้างหน้า หมายความว่าสามารถเข้าถึงได้โดยการอ่านแอดเดรสแบบไบท์และแบบบิท คือสามารถเข้าถึงบิทใดบิทหนึ่งใน SFR ได้ +SFR ที่มีเครื่องหมายบวกแสดงว่ามีอยู่ใน 8032/8052 เท่านั้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รายละเอียดของ REGISTER ที่ทำหน้าที่พิเศษ

ACCUMULATOR

ทำหน้าที่ในการเก็บข้อมูล พักข้อมูล ในการกระทำทาง LOGIC และ ARITHMATIC

B REGITER

รีจิสเตอร์ B ใช้ในคำสั่งคูณและหาร ส่วนในคำสั่งอื่นสามารถใช้เหมือนกับรีจิสเตอร์ทั่วไป

PROGRAM STATUS WORD

ประกอบด้ายรายละเอียดดังในรูปที่ 2.2

SYMBOL	POSITION	NAME AND SINIFICAND
CY	PSW.7	CARRY FLAG
AC	PSW.6	AUXILLARY CARRY FLAG (FOR BCD OPERATION)
FO	PSW.5	FLAG O (FOR GERNERAL PURPOSE)
RS1	PSW.4	REGISTER BANK SELECT CONTROL BITSET/CLEAR BY SOFTWARE TO DETERMINE
RSO	PSW.3	WORKING REGISTER BANK (SEE NOTE)

รูปที่ 2.2 PSW:PROGRAM STATUS WORD

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

SYMBOL	POSITION	NAMME AND SINIFICAND
OV	PSW.2	OVER FLOW FLAG
-	PSW.1	(RESERVED)
P	PSW.0	PARITY FLAG SET/CLEAR BY HARDWARE EACH INSTRUCTION TO INDICATE AN ODD/EVEN NUMBER OF "ONE" BIT IN ACCUMULATER.

รูปที่ 2.2 PSW:PROGRAM STATUS WORD (ต่อ)

NOTE THE CONSISTS OF (RS1,RS0) ENABLE THE WORKING REGISTER BANKS AS FOLLOWS

(0.0)-BANK 0 (00H-07H)

(0.1)-BANG 1 (08H-0FH)

(1.0)-BANG 2 (10H-17H)

(1.1)-BANG 3 (18H-1FH)

STACK POINTER

SP เป็นรีจิสเตอร์ขนาด 8 BIT เมื่อใช้คำสั่งPUSH หรือ CALL SPจะเพิ่มขึ้นก่อนที่จะเก็บข้อมูลหรือ ADDRESS ซึ่ง STACK จะอยู่ที่ไหนก็ได้ในหน่วยความจำRAM ขน 8031 ภายหลังการรีเซ็ตSTACK จะมาอยู่ที่ 07H นั่นคือ STACK จะเริ่มที่ 08H

DATA POINTER

DPTR ประกอบด้วย DPH และ DPL ซึ่งจะรวมกันเป็นรีจิสเตอร์ขนาด 16 บิท การเข้าถึง DPTR สามารถทำได้ทั้งแบบ 16 บิท และแบบ 8 บิท หน้าที่ของ DPTR ถ้า

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการ
033207

ใช้แบบ 16 บิท จะทำหน้าที่เป็นตัวชี้ข้อมูลที่อยู่ในหน่วยความจำภายนอก

PORT 0-3

พอร์ท 0,พอร์ท 1,พอร์ท 2 และ พอร์ท 3 เป็น SFR ตัวหนึ่งที่สามารถแลตท์ข้อมูลได้สามารถใช้เป็นอินพุทพอร์ทและเอ๊าพุทพอร์ทได้ทั้ง 3 พอร์ท

SERIAL DATA BUFFER

แบ่งเป็นริจิสเตอร์บัฟเฟอร์ในทางส่งและบัฟเฟอร์ในทางรับเมื่อมีการส่งข้อมูลภายในให้ SBUF ข้อมูลจะถูกส่งมาที่ บัฟเฟอร์ในทางส่งที่ซึ่งจะใช้ในการส่งข้อมูลอนุกรม การส่งข้อมูล ให้ SBUF จะเป็นการเริ่มต้นการส่งด้วย และถ้าอ่านข้อมูลจาก SBUF ข้อมูลจะถูกอ่านจาก บัฟเฟอร์ในทางรับ

TIMER REGISTER

ริจิสเตอร์คู่ (TH0,TL0) และ (TH1,TL1)เป็นตัวจับเวลาและตัวนับขนาด 16 บิท จะกล่าวรายละเอียดภายหลัง

CONTROL REGISTERS

ริจิสเตอร์หน้าที่พิเศษ IP,IE,TMOD,TCON,T2CON,SCON และPCON ประกอบด้วยบิทควบ คุมและบิทสถานะสำหรับระบบการ อินเทอร์รัพท์

2.2.3 โครงสร้างและการทำงานของพอร์ท

8031 มี I/O พอร์ทอยู่ 4 พอร์ท โดยแต่ละพอร์ทจะเป็นพอร์ทแบบ 2 ทิศทางมีการแลตช์ข้อมูลได้ รวมทั้งวงจรขับทางด้านเอ๊าท์พุท และบัฟเฟอร์ทางด้านอินพุทพอร์ท 0 และพอร์ท 2 ใช้สำหรับติดต่อกับหน่วยความจำภายนอก ในการใช้ 8031 ติดต่อกับหน่วยความจำภายนอก พอร์ท 0 จะให้เอ๊าท์พุทเป็น LOW BYTE ของแอดเดรสของหน่วยความจำภายนอก การ MULTIPLEX กับข้อมูลที่จะเขียนหรืออ่าน
สรุป คือพอร์ท 0 จะเป็นทั้ง ADDRESS และDATA ส่วนพอร์ท 2 จะให้ ADDRESS HIGH BYTE ของหน่วยความจำภายนอกขาของพอร์ท 3 ทั้งหมดกับอีก 2 บิทของ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

พอร์ท 1 (8052)จะทำงานหลายหน้าที่ ดังรายละเอียดในรูปที่ 2.4

*P1.0	T2(TIMER/COUNTER 2 EXTERNAL INPUT)
*P1.1	T2EX(TIMER/COUNTER 2 CAPTURE/RELOAD TRIGGER
P3.0	RXD(SERIAL INPUT PORT)
P3.1	TXD(SERIAL OUTPUT PORT)
P3.2	INT0(EXTERNAL INTERUPT)
P3.3	INT1(EXTERNAL INTERUPT)
P3.4	T0(TIMER/COUNTER 0 EXTERNAL INPUT)
P3.5	T1(TIMER/COUNTER 1 EXTERNAL INPUT)
P3.6	WR(EXTERNAL DATA MEMORY WRITE STROBE)
P3.7	RD(EXTERNAL DATA MEMORY READ DATA)

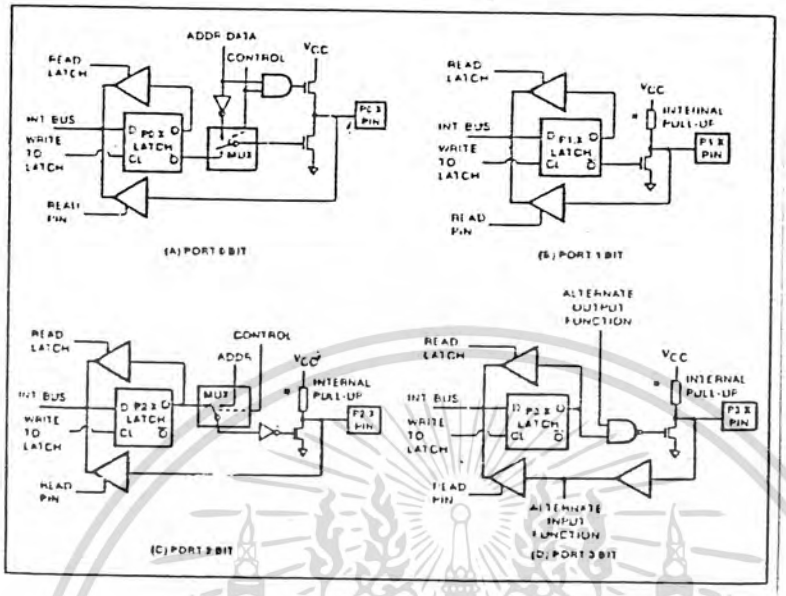
รูปที่ 2.8 แสดงหน้าที่ของพอร์ท 1

หมายเหตุ *P1.0 และ P1.1สงวนไว้สำหรับ 8052

โครงสร้างของพอร์ท

ในรูป 2.4 แสดงโครงสร้างของ I/O ของแต่ละพอร์ทโดยแสดงเพียงพอร์ทละบิท ส่วนของพอร์ทแลตช์ใช้ D-FLIP PLOP ซึ่งรับข้อมูลพัลส์จากภายนอกเพื่อตอบสนองต่อสัญญาณ "READ LATCH"จาก CPU เมื่อ CPU ต้องการอ่านพอร์ทส่วนสัญญาณที่ภายนอกของพอร์ทจะถูกต่อเข้ากับขาข้อมูลภายในและพอร์ทภายนอกนี้จะถูกอ่านโดยตอบสนองต่อ สัญญาณ "READ PIN" (ดูรูป 3 ประกอบ) ส่วนคำสั่งอื่นๆจะอ่านใช้สัญญาณ"READ PIN" ดูรายละเอียดคำสั่งที่อ่นข้อมูลจาก READ LATCH ในรูป 2.5

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.4 โครงสร้างแต่ละบิทของพอร์ท

```
ANL          (LOGIC AND,e.g.,ANL P1,A)
ORL          (LOGIC OR,e.g.,ORL P2,A)
XRL          (LOGIC EX-OR e.g.,XRL P3,A)
JBC          (JUMP IF BIT=1 AND CLEAR BIT
              e.g.,JBC P1.1,LABEL)
CPL          (COMPLEMENT BIT,e.g.,CPL P3.0)
INC          (INCREMENT, e.g.,INC P2)
DEC          (INCREMENT,e.g.,DEC P2)
DJNZ         (DECREMENT AND JUMP IF NOT ZERO
              e.g.,DJNZ P3,LABEL)
MOV PX.Y,C   (MOVE CARRY TO BIT Y OF PORT X)
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
CLR PX.Y     (CLEAR BIT Y OF PORT X)
SETB PX.Y    (SET BIT Y OF PORT X)
```

รูปที่ 2.5 คำสั่งที่อ่านจาก read pin (ต่อ)

เหตุที่บางคำสั่งต้องอ่านพอร์ตจาก "READ LATCH" เพราะในบางกรณี CPU อาจเข้าใจสัญญาณที่ขาของพอร์ตผิดพลาด ตัวอย่างเช่น PORT1 ถูกต่ออยู่กับขา BASE ของ TRANSISTOR และให้สัญญาณทางออกเป็น HIGH แก่ขา BASE ของทรานซิสเตอร์ ในขณะที่ทรานซิสเตอร์นำการแสจะทำให้แรงดันที่ขา BASEเหลือโดยประมาณ 0.6 โวลท์ ซึ่งขา BASE ถูกต่ออยู่กับพอร์ตเมื่อ CPU ทำการอ่านพอร์ตจะทำให้เข้าใจสัญญาณที่ขาของพอร์ตผิดไปทั้ง ๆ ที่ขณะนั้นขาของพอร์ตเป็น HIGH อยู่

จากรูปที่ 2.4 จะเห็นว่าพอร์ต 0 นอกจากจะเป็น I/O พอร์ตแล้วยังเป็นได้ทั้งแอดเดรสบัสและบัสข้อมูล ส่วนพอร์ต 2 เป็นทั้ง I/O พอร์ต และ แอดเดรสซึ่งควบคุมโดยสัญญาณภายในชิพในขณะทำการติดต่อกับหน่วยความจำภายนอกSFR ของพอร์ต 2 จะไม่เปลี่ยนค่าหลังจากให้แอดเดรสแล้วแต่ SFR ของพอร์ต 0 จะเปลี่ยนเป็น HIGH ทั้ง 8 บิท เพื่อที่จะรับข้อมูลที่จะส่งมาทางบัสข้อมูลของหน่วยความจำ (เนื่องจาก SFR สามารถแลตช์ข้อมูลได้ถ้าไม่ทำให้เป็น HIGH อาจเกิดในการผิดพลาดในการอ่านข้อมูลจากหน่วยความจำภายนอก)

พอร์ต 1, พอร์ต 2 และพอร์ต 3 มีการพูลอัพภายใน ส่วนในพอร์ต 0 จะเป็นแบบ OPENDRAIN OUTPUT แต่ในกรณีที่ใช้พอร์ต 1,2,3 เป็นอินพุทจะต้องทำให้แต่ละบิทเป็น HIGH ก่อนเพื่อทำให้ FET ที่อยู่ทางด้านทางออกหยุดนำกระแสและรักษาระดับสัญญาณ HIGH จากความต้านทานพูลอัพภายใน (ดูรูป 2.4 ประกอบ)ข้อแตกต่างของพอร์ต 0 ก็คือ ไม่มีการ พูลอัพภายใน (ดูรูป (2.4A)ประกอบ) FET ตัวที่ทำหน้าที่พูลอัพ (ตัวบน) จะใช้เพียงให้เอ๊าท์พุทเป็น 1 ในขณะที่ติดต่อกับหน่วยความจำภายนอกในกรณีอื่น ๆ FET ตัวนี้จะถูกทำให้คัทออฟ ฉะนั้นพอร์ต 0 ที่ใช้เป็นเอ๊าท์พุทพอร์ตจะเป็น OPENDRAIN

การเขียน 1 (FFH) ไปที่พอร์ต 0 จะทำให้ FET ทั้ง 2 ตัวหยุดทำงานผลก็

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

คือทำให้ขาพอร์ต 0 ลอยและสามารถใช้เป็นขาอินพุทแบบความต้านหน่วงสูง (HI-Z INPUT) การริเซท 8031 จะทำให้ทุกพอร์ตมีระดับสัญญาณเป็น HIGH

2.2.4 การติดต่อกับหน่วยความจำภายนอก

การติดต่อกับหน่วยความจำภายนอกกระทำได้ 2 แบบ คือ ติดต่อกับโปรแกรมภายนอกกับติดต่อกับหน่วยความจำภายนอกที่เก็บข้อมูลการติดต่อกับโปรแกรมภายนอกจะใช้ สัญญาณ PSEN (PROGRAM STORE ENABLE) เป็นสัญญาณอ่านโปรแกรมภายนอก ส่วนการติดต่อกับหน่วยความจำภายนอกที่เก็บข้อมูลจะใช้ RD (P3.7)และ WR (P3.6) เหมือนกับ CPU ทั่วไป

การติดต่อกับข้อมูลภายนอกจะใช้ได้ทั้ง 16 บิท (MOVX @DPTR) หรือ เป็นแบบ 8 บิท (MOVX @RI) เมื่อไรก็ตามที่ต้องใช้แอดเดรสขนาด 16 บิท แอดเดรสไบท์สูง (A8-A15) จะ ออกทางพอร์ต 2

เมื่อติดต่อกับหน่วยความจำของข้อมูลภายนอกแบบ 8 บิท (MOVX A, @RI)จะไม่มีผลกระทบกับ SFR ของพอร์ต 2 คือ SFR ของพอร์ต 2 คือ SFR ของพอร์ต 2 จะยังคง แลตช์ข้อมูลเดิมเอาไว้ ประโยชน์ คือ สามารถใช้พอร์ต 2 กำหนดหน้า (PAGE) ของหน่วยความจำ

การที่จะติดต่อกับหน่วยความจำภายนอกได้ขึ้นอยู่กับ 2 ประการคือ

1. เมื่อต่อขา EA ของ 8031 ลงกราวด์
2. เมื่อไรก็ตามที่ program counter (PC)มีค่ามากกว่า 0FFFH

สัญญาณ PSEN

เมื่อมีการเฟชคำสั่งจากโปรแกรมภายนอก สัญญาณ PSEN จะแอคทีฟ 2 ครั้งทุก ๆ 1 แมชีนไซเคิล (ยกเว้นคำสั่ง MOVX)

สัญญาณ ALE

จะเป็นขาสโตรบในการแลตช์แอดเดรสไบท์ต่ำ เมื่อทำการติดต่อกับหน่วยความจำภายนอก ALE จะแอคทีฟ 2 ครั้งทุก ๆแมชีนไซเคิล ยกเว้นขณะทำคำสั่งที่ติดต่อกับ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ข้อมูลในหน่วยความจำภายนอก (MOVX) ฉะนั้นในกรณีที่ระบบไม่ได้ใช้หน่วยความจำภายนอก (MOVX) การแอคทีฟของ ALE จะคงที่ในอัตรา 1/6 ของความถี่ออสซิลเลเตอร์ เพื่อใช้เป็น clock ให้กับวงจรภายนอกได้ การใช้งานบางกรณีต้องการที่จะใช้ทั้งโปรแกรมและข้อมูลอยู่ในเพจเดียวกัน (64K) เราสามารถทำได้โดยการรวมสัญญาณ PSEN และ RD โดยใช้ AND GATE

TIMER/COUNTER

8031 มี TIMER/COUNTER อยู่ 2 ตัว คือ T0 และ T1 สัญญาณ INPUT ที่จะป้อนให้ COUNTER นั้นทำงานที่ขอบขาลง (1 TO 0) คือ ต้องเป็นพัลส์ HIGH 1แมชินไซเคิลและเป็น LOW 1 แมชินไซเคิล ฉะนั้นความถี่สูงสุดที่ COUNTER จะนับได้นั้นประมาณ 1/24 ของความถี่ออสซิลเลเตอร์การทำงานของ TIMER/COUNTER แบ่งเป็น 3 โหมด ดังกล่าวต่อไป

โหมด0

การทำงานในโหมดนี้รีจิสเตอร์ถูกกำหนดให้เป็นแบบ 13 บิท โดยการนับจากค่าที่ทุกบิทเป็น HIGH ไปจนทุก ๆ บิทเป็น 0 เกิด OVERFLOW และจะให้สัญญาณอินเตอร์รัพท์โดยเซ็ทแฟลก TF0 หรือ TF1 การที่จะให้ TIMER/COUNTER ตัวใดอยู่ในโหมดใดนั้นกำหนดได้จากรีจิสเตอร์ TMOD (รูปที่ 2.6)

รูปที่ 2.6 TMOD: TIMER MODE CONTROL REGISTOR

(MSB) (LSB)

GATE C/T M1 M0	GATE C/T M1 M0
-----TIMER1-----	-----TIMER2-----

GATE	GATING CONTROL WHEN SET	M1	M0	OPERATION MODE
	TIMER/COUNTER "X" IS ENABLE	0	0	MCS-48 TIMER"TLX"
	ONAY WHILE "INTx" PIN IS SET			SERVES AS FIVE BIT

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

	WHEN CLEEARED TIMER "X"IS ENABLED WHENEVER"TEx"CONTROL BIT IS SET	0	1 PRESSCALER 16 BIT TIMER/ COUNTER "THx" AND"TLx" ARE CADCADED IS NO PRESCALE
C/T	TIMER OR COUNTER SELCTOR CLEAR FOR TIME OPERATION (INPUT FROM INTERNAL SYSTEM CLOCK)SET FOR COUNTER OPERATION (INPUT FROM "Tx"INPUT PIN)	1	0 8 BIT AUTO-RELOAD TIMER/COUNTER "THx"HOLD A VALUE WHICH IS TO BE RELOADED INTO TLx EACH TIME IT OVERFLOWS
		1	1 (TIMER 0)TLO IS AN EIGHT BITTIMER COUNTER CONTROLED BY STANDARD TIMER 0 CONTROL BITS THO IS THE EIGH BIT TIMER 1 CONTROL BIT

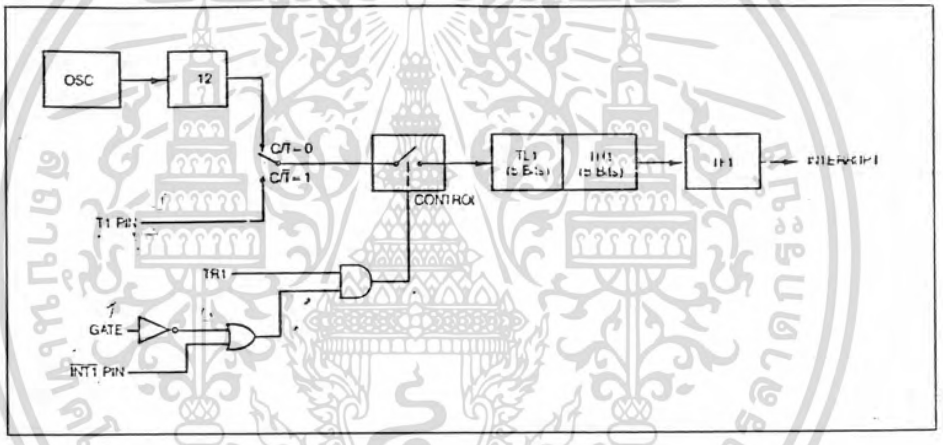
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1 1 (TIMER 1)TIMER

/COUNTER 1

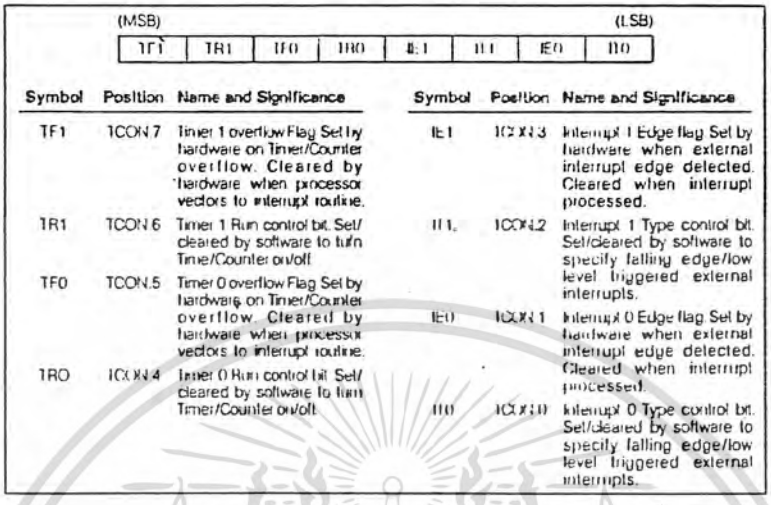
STOPED

COUNTER จะทำงานได้ก็ต่อเมื่อ TR1= 1และ GATE= 0 หรือ INT1 =1 TIMER /COUNTER จะถูกควบคุมด้วยสัญญาณ INT1 จากภายนอก ประโยชน์ในการทำงานแบบนี้คือใช้วัดความกว้างของพัลส์จากอินพุทภายนอก TR1เป็นบิทควบคุมอยู่ใน TCON ดังในรูปที่ 2.8



รูปที่ 2.7 TIMER/COUNTER 1 MODE 0 : 13 BIT COUNTER

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



(MSB)							(LSB)
TF1	TR1	TF0	TR0	IE1	IT1	IE0	IT0

Symbol	Position	Name and Significance
TF1	TCON.7	Timer 1 overflow Flag Set by hardware on Timer/Counter overflow. Cleared by hardware when processor vectors to interrupt routine.
TR1	TCON.6	Timer 1 Run control bit. Set/cleared by software to turn Timer/Counter on/off.
TF0	TCON.5	Timer 0 overflow Flag Set by hardware on Timer/Counter overflow. Cleared by hardware when processor vectors to interrupt routine.
TR0	TCON.4	Timer 0 Run control bit Set/cleared by software to turn Timer/Counter on/off.
IE1	TCON.3	Interrupt 1 Edge flag Set by hardware when external interrupt edge detected. Cleared when interrupt processed.
IT1	TCON.2	Interrupt 1 Type control bit. Set/cleared by software to specify falling edge/low level triggered external interrupts.
IE0	TCON.1	Interrupt 0 Edge flag Set by hardware when external interrupt edge detected. Cleared when interrupt processed.
IT0	TCON.0	Interrupt 0 Type control bit. Set/cleared by software to specify falling edge/low level triggered external interrupts.

รูปที่ 2.8 TCON:TIMER/COUNTER CONTROL REGISTER

ในโหมด 0 ที่จะแบ่ง TH1 เป็น 8 บิท กับ TL1 อีก 5 บิท โดยที่เลือกอีก 3 บิท นั้น ไม่ได้ใช้ และการใช้งานจะเหมือนกันทั้ง TIMER 1 และ TIMER 0

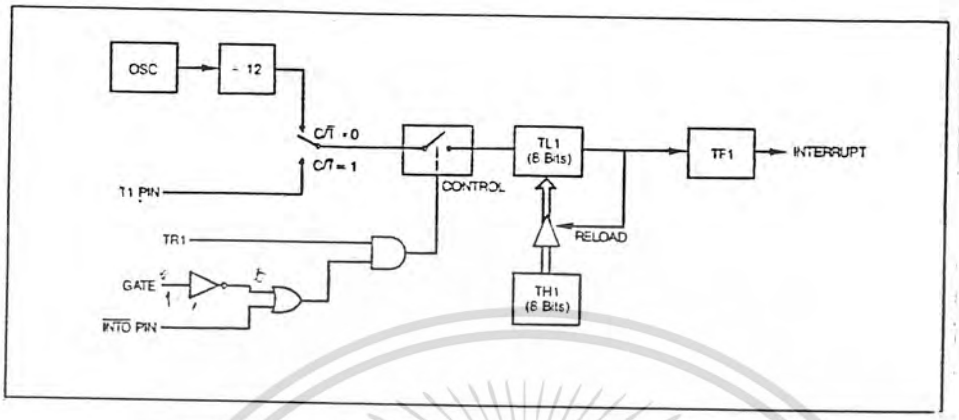
โหมด 1

การใช้งานเหมือนกับโหมด 0 ยกเว้นริจิสเตอร์ที่ใช้จะเป็นแบบ 16 บิท

โหมด 2

ในโหมด 2 ริจิสเตอร์จะเป็นแบบ 8 บิท โดยที่ TL1 จะสามารถโหมดข้อมูลจาก TH1ได้ใหม่ (AUTO-RELOAD) เมื่อเกิดโอเวอร์โฟลวจาก TL1 (ดูรูปที่ 2.9) โดยที่ค่าใน TH1 จะถูกเปลี่ยนการทำงานอื่น ๆจะไม่เหมือนกับโหมด 0

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

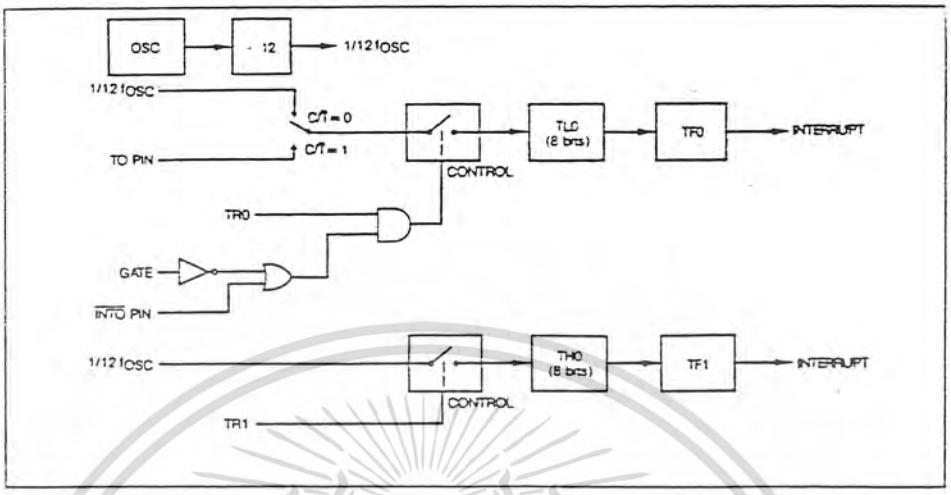


รูปที่ 2.9 TIMER/COUNTER 1 MODE 2 : 8 BIT AUTO-RELOAD

โหมด 3

ในโหมด 3 นี้จะแยก TL0 และ TH0 ของ TIMER 0 ใช้โดยอิสระ TL0 จะใช้บิทความควบคุมคือ GATE, TR0, INT0 ส่วน TH0 ถูกใช้เป็น TIMER (นับแมชินไซเคิล) และรับช่วงการใช้ TR1 และ TF1 ของ TIMER 1 ฉะนั้นในโหมด 3 TH0 จะควบคุมการอินเตอร์รัพท์ของ TIMER 1 (TF1) เมื่อใช้ TIMER 0 ในโหมด 3 แล้ว TIMER 1 สามารถจะสลับใช้ระหว่างโหมด 3 และโหมดอื่นได้หรือ ใช้เป็น BAUD RATE GENERATER

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.10 TIMER/COUNTER 0 MODE 3:TWO 8 BIT COUNTERS

พอร์ตอนุกรม

พอร์ตอนุกรมนี้เป็น FULL DUPLEX ที่สามารถรับข้อมูลใน BYTE แรกยังไม่ถูกอ่านออกไปจาก BUFFER แต่อย่างไรก็ตามข้อมูล BYTE แรกจะต้องถูกอ่านไปก่อนที่การรับข้อมูลใน BYTE ที่สองจะเสร็จสมบูรณ์มิฉะนั้นข้อมูล BYTE แรกสูญเสียไป (ถูกทับด้วยข้อมูลที่ตามมา) ข้อมูลที่จะใช้ในการส่งและรับถูกพักไว้ ณ ที่เดียวกันคือ SBUF การเขียนข้อมูลไปที่ SBUF จะเป็นการโหลดข้อมูลให้กับ TRANSMIT REGISTOR และในการอ่าน SBUF จะเป็นการอ่านข้อมูล RECEIVER REGISTOR พอร์ตอนุกรมแบ่งการทำงานออกเป็น 4 โหมด

โหมด 0 : ข้อมูลจะเข้ามาทาง RXD ส่วนข้อมูลทางออกจะออกทาง TXD ความเร็วในการส่ง (BAUD RATE) จะถูกกำหนดตายตัว เป็น 1/12 ของความถี่ออสซิลเลเตอร์ของระบบในโหมด 0 จะเป็นการส่งข้อมูลขนาด 8 บิท (โดย LSB ออกไปก่อน)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โหมด 1 : ส่งและรับขนาดข้อมูล 10 บิทซึ่งประกอบด้วย STARE BIT(0) , ข้อมูล 8 บิท (LSB ออกก่อน),STOP BIT ในขณะรับข้อมูล STOP BITจะถูกส่งให้ RB8ในริจิสเตอร์หน้าที่พิเศษ SCON ความเร็วในการส่งไม่กำหนดตายตัว(ดูการหา BAUD RATE)

โหมด 2 : ส่งและรับข้อมูลขนาด 11 บิท ประกอบด้วย START BIT (0), ข้อมูล 8 บิท (LSB ก่อน),ข้อมูลบิทที่ 9 ของข้อมูลสามารถ SET เป็น 0 หรือ 1 ก็ได้ประโยชน์อาจใช้เป็นตัวส่งพาริตี้บิทโดยนำค่าของแฟลก P ใน PSW มาไว้ใน TB8 ของ SCON ความเร็วในการส่งจะถูกโปรแกรมเป็น 1/32 หรือ 1/64 ของออสซิลเลเตอร์

โหมด 3 : การทำงานเหมือนกับโหมด 2 เพียงแต่ความเร็วในการส่งไม่กำหนดตัวตาย

การทำงานทั้ง 4 โหมด ทางด้านส่งจะเริ่มการส่งขึ้นก็ต่อเมื่อ SBUF ถูกใช้เป็นปลายทางของคำสั่งต่าง ๆ เช่น mov SBUF,A ในทางด้านการรับจะเริ่มก็เมื่อ RI=0 และ REN = 1 ในโหมด 0 ส่วนโหมดอื่น ๆ การรับข้อมูลจะเริ่มต้นเมื่อมี START BIT เข้ามาและ REN =1

MULTIPROCESSOR COMMUNICATIONS

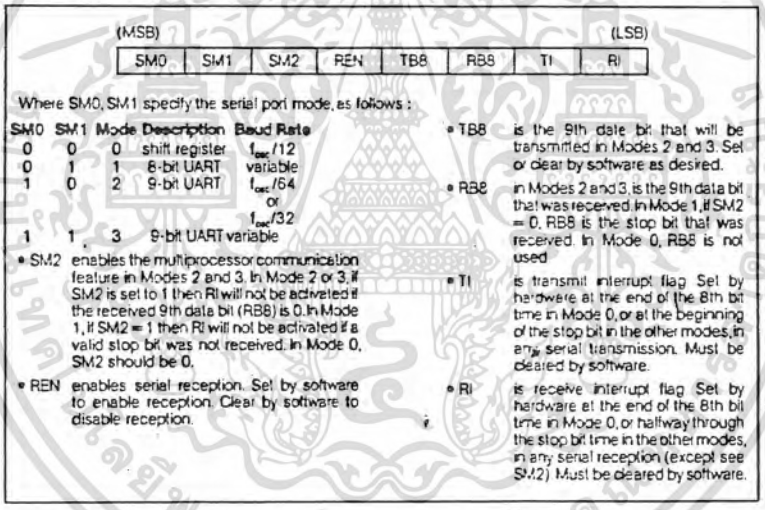
ในโหมด 2 และ 3 มีเงื่อนไขพิเศษสำหรับการติดต่อระหว่างหน่วยประมวลหลายตัว ในโหมดนี้ข้อมูลในบิทที่ 9 จะถูกโหลดเข้าไปใน RB8 ต่อจากนั้นจึงจะรับ STOP BIT อินเตอร์รัพท์แฟลกของพอร์ตอนุกรมนอดทีฟก็ต่อเมื่อ RB8 =1 การจะใช้ลักษณะพิเศษนี้ต้องเซ็ท SM2 ใน SCON เมื่อตัวประมวลผลตัวแม่ต้องการจะส่งข้อมูลเป็นบล็อกให้กับตัวลูกซึ่งมีหลายตัวฉะนั้น BYTE แรกที่จะต้องส่งคือ ตัวกำหนดว่าจะให้ตัวใดเป็นตัวรับ (การกำหนด address) โดยที่ BYTE ที่เป็น ADDRESS จะต้องแตกต่างจาก BYTE ข้อมูลซึ่งเราทำได้โดยให้บิทที่ 9 ของข้อมูลเป็น 1 ใน BYTE ที่กำหนดให้เป็น ADDRESS ของตัวลูก (SLAVE)ส่วนในตัวลูกถูกต้อง เซ็ท SM2 = เมื่อทำดังนี้หน่วยประมวลผลที่เป็นตัวลูกทุกตัวจะไม่ถูกอินเตอร์รัพท์ที่เกิดจากการส่งข้อมูลที่ส่งนั้นบิทที่ 9 ไม่ถูกเซ็ท ฉะนั้นหน่วยประมวลผลตัวลูกทุกตัวจะรับการส่งที่กำหนดให้เป็น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ADDRESS แต่จะมีเพียงตัวเดียวที่ตรวจสอบแล้วว่าถูกกำหนดให้เป็นปลายทางในการส่งข้อมูลจึงเตรียมการรับบล๊อกของข้อมูลและเคลียร์ ด้วย ส่วนตัวอื่น ๆ คงปล่อยให้ SM2 เซ็ทอยู่อย่างเดิมและทำงานตามปกติต่อไป SM2นี้จะไม่มีผลในการใช้งานในโหมด 0 ส่วนโหมด 1 จะใช้สำหรับตรวจ สอบความถูกต้องของ STOP BIT ในโหมด 1 เมื่อเซ็ท SM2 = 1 แล้ว RI(RCECIEVE INTERRUPT)จะไม่ถูก SET ถ้า ไม่ได้รับ STOP BIT ที่ถูกต้อง

SERIAL PORT CONTROL REGISTOR

ในรูปที่ 2.11 เป็นรายละเอียดของ SFR ที่ทำหน้าที่ควบคุมพอร์ตอนุกรม(SCON) ภายใน SCON ไม่เพียงแต่มีการเลือกโหมดเท่านั้นแต่ยังรวมถึงบิทที่ 9 ของข้อมูล (TB8, RB8)และ SERIAL PORT INTERUPT (T1 และ R1)



(MSB) (LSB)

SM0	SM1	SM2	REN	TB8	RB8	TI	RI

Where SM0, SM1 specify the serial port mode, as follows :

SM0	SM1	Mode	Description	Baud Rate
0	0	0	shift register	$f_{osc}/12$
0	1	1	8-bit UART	variable
1	0	2	9-bit UART	$f_{osc}/64$ or $f_{osc}/32$
1	1	3	9-bit UART	variable

- SM2 enables the multiprocessor communication feature in Modes 2 and 3. In Mode 2 or 3, if SM2 is set to 1 then RI will not be activated if the received 9th data bit (RB8) is 0. In Mode 1, if SM2 = 1 then RI will not be activated if a valid stop bit was not received. In Mode 0, SM2 should be 0.
- REN enables serial reception. Set by software to enable reception. Clear by software to disable reception.
- TB8 is the 9th date bit that will be transmitted in Modes 2 and 3. Set or clear by software as desired.
- RB8 in Modes 2 and 3, is the 9th data bit that was received. In Mode 1, if SM2 = 0, RB8 is the stop bit that was received. In Mode 0, RB8 is not used.
- TI is transmit interrupt flag. Set by hardware at the end of the 8th bit time in Mode 0, or at the beginning of the stop bit in the other modes, in any serial transmission. Must be cleared by software.
- RI is receive interrupt flag. Set by hardware at the end of the 8th bit time in Mode 0, or halfway through the stop bit time in the other modes, in any serial reception (except see SM2). Must be cleared by software.

รูปที่ 2.11 SCON : SERIAL PORT CONTROL REGISTOR

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

อัตราความเร็วในการส่ง (BAUD RATES)

ในโหมด 0 ความเร็วในการกำหนดไว้แน่นอนคือ

$$\text{MODE 0 BAUD RATE} = \frac{\text{OSCILLATOR FREQUENCY}}{12}$$

ในโหมด 2 ความเร็วในการส่งขึ้นอยู่กับ SMOD ซึ่งอยู่ใน PCON ถ้า SMOD = 0 ความเร็วจะเท่ากับ 1/64 ของความถี่ออสซิลเลเตอร์ ถ้า SMOD= 1 ความเร็วจะเป็น 1/32 ของความถี่ออสซิลเลเตอร์ ความเร็วในการส่งในโหมด 2 คำนวนได้จากสูตรดังนี้

$$\text{MODE 2 BAUD RATE} = ((2^{\text{SMOD}})/64) * (\text{OSCILLATOR FREQUENCY})$$

โหมด 1 และโหมด 3 ความเร็วในการส่งถูกกำหนดโดยอัตราของ OVERFLOW ของ TIMER 1

การใช้ TIMER 1 ในการกำเนิด BAUD RATE

เมื่อใช้ TIMER 1 เป็นตัวกำเนิด BAUD RATE ความเร็วจะขึ้นอยู่กับ OVERFOLW RATE และค่าที่อยู่ใน SMOD ความเร็วคำนวนได้จากสูตร

$$\text{MODE 1,3 BATE} = ((2^{\text{SMOD}})/32) * (\text{TIMER 1 OVERFOLW RATE})$$

ในการใช้ TIMER 1เป็นตัวกำเนิดความเร็วในการส่งข้อมูลนี้จะต้องไม่ยอมให้มีการอินเตอร์รัพท์ของ TIMER 1 วิธีการใช้ TIMER 1 เป็นตัวกำเนิดความเร็วนี้โดยทั่วไปเราจะเซ็ทให้ TIMER 1/COUNTER เป็น TIMER และในโหมด 2 ซึ่งTIMER ในโหมด 2 นี้ทำ AUTO-RELOAD ได้ (เซ็ทไบท์สูงของ TMOD =0010B) ในกรณีนี้ความเร็วจะคำนวณได้จากสูตร

$$\text{MODE 1,3 BAUD RATE} = \frac{(2^{\text{SMOD}}) * \text{OSCILLATOR FREQUENCY}}{32 \times 12 \times [256-(\text{th1})]}$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โดยที่ค่า TH1 จะเป็นค่าในช่อง RELODE ของตารางที่ 2.2

BUAD RATE	fosc	SMOD	TIME 1 C/T MODE	RELOAD	VALUE
MODE 0 MAX : 0MHZ	12MHz	X	X	X	X
MODE 1 MAX : 375K	12MHz	1	X	X	X
MODE 1,3 :62.5K	12MHz	1	0	2	FFH
19.2K	11.059MHz	1	0	2	FDH
9.6K	11.059MHz	0	0	2	FDH
4.8K	11.059MHz	0	0	2	FAH
2.4K	11.059MHz	0	0	2	F4H
1.2K	11.059MHz	0	0	2	E8H
137.5	11.059MHz	0	0	2	1DH
110	6MHz	0	0	2	72H
110	12MHz	0	0	1	FEEBH

รายละเอียดเพิ่มเติมเกี่ยวกับ MODE 0

ข้อมูลอนุกรมจะผ่านเข้าออกทางขา RXD ส่วนขา TXD นั้นจะกำเนิดสัญญาณนาฬิกาข้อมูลที่ใช้ในการส่งและรับมีขนาด 8 บิท และอัตราการส่งกำหนดตายตัวที่ 1/12 ของความถี่ออสซิลเลเตอร์ รูปที่ 2.12 แสดงแผนภาพเวลา (TIMING DIAGRAM) และการทำงานอย่างง่ายของพอร์ตอนุกรมในโหมด 0 การส่งจะเริ่มเมื่อมีการเขียนข้อมูลมาที่รีจิสเตอร์ SBUF สัญญาณ "WRITE TO SBUF" ที่ S6P2 จะให้ค่า 1 แก่บิทที่ 9 ของรีจิสเตอร์เลื่อนข้อมูล ทางส่ง (TRANSMIT SHIFT REGISTER) และบอกส่วนควบคุมทางการส่ง ให้เริ่มต้นการส่ง ระยะเวลาตั้งแต่เกิดสัญญาณ "WRITE TOSSBUF" จนกระทั่งสัญญาณ SEND เริ่มทำงานจะใช้เวลาทั้งสิ้น 1แมชีนไซเคิลสัญญาณ SEND ทำให้ข้อมูล ของรีจิสเตอร์เลื่อนข้อมูล(SHIFT REGISTER)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ถูกส่งมายังขา "ALTERNATE OUTPUT FUNCTION" ของ P3.0 (ดูรูปโครงสร้างพอร์ต 3)และยังทำให้ SHIFT CLOCK มายังขา "ALTERNATE OUTCTION" ของ P3.1 (TXD) ซึ่งจะเป็น LOW ในระหว่างเวลาของ S3, S4 และ S5 ของทุก ๆ แมชีนไซเคิล และเป็น HI ระหว่าง S6,S1 และ S2 ที่เวลา S6P2 ของทุก ๆ แมชีนไซเคิลที่สัญญาณ SEND แอคทีฟข้อมูลใน SHIFT REGISTER จะถูกเลื่อนไปทางขวา 1 บิทในขณะที่ข้อมูลถูกเลื่อนไปทางขวา 1 บิท ข้อมูล 0 จะถูกเลื่อนเข้ามาทางซ้าย 1 บิท และเมื่อข้อมูลบิทสำคัญสูงสุด (MSB) ของ SHIFT REGISTER มาถึงยังตำแหน่ง OUTPUT ของ SHIFT REGISTER แล้วสัญญาณ 1 ที่ถูกโหลดมายังบิทที่ 9 ของ SHIFT REGISTER ในตอนแรก จะเข้ามาทางซ้ายของบิท MSB และข้อมูลทางซ้ายทั้งหมดของบิทที่เข้ามาต่อท้ายบิท MSB จะเป็น 0 ทั้งหมด สภาวะเช่นนี้จะทำให้หน่วยควบคุมการส่ง(TX CONTROL BLOCK)จะทำการเลื่อนข้อมูลอีก 1 ครั้งเป็นครั้งสุดท้ายก่อนที่จะยกเลิกสัญญาณ SEND และเซ็ทบิท TI การกระทำทั้ง 2 อย่างนี้จะเกิดขึ้นที่ S1P1 ของของแมชีนไซเคิลที่ 10 หลังเกิดสัญญาณ "WRITE TO SUBF" ทางด้านรับจะเริ่มต้นเมื่อ RAN = 1 และ RI = 0 ที่ S6P2 ของแมชีนไซเคิลต่อไปหน่วยควบคุมการรับ (RX CONTROL UNIT) เขียนข้อมูล 1111110 ไปที่รีจิสเตอร์เลื่อนข้อมูลทางด้านรับ (RECEIVE SHIFT REGISTER)และสัญญาณนาฬิกาเฟสถัดไปจะทำขา RECEIVE ทำงาน (ดูรูป 2.12) ข้อมูลที่อยู่ในตัวเลื่อนข้อมูลด้านรับจะถูกเลื่อนมาทางด้านซ้าย 1 ตำแหน่ง ข้อมูลที่เข้ามาทางขวาคือข้อมูลบิทที่ถูกสุ่มตัวอย่างทางขา P3.0 ที่เวลา S5P2 ของแมชีนไซเคิลเดียวกัน (S5P2)ในขณะที่ข้อมูลเข้ามาทางขวาบิท 1จะถูกดันเข้ามาทางซ้ายและเมื่อบิทที่เป็น 0 ซึ่ง ถูกโหลดให้ในตอนเริ่มต้น (1111110) มาถึงทางด้านซ้ายมือสุดของตัวเลื่อนข้อมูล เป็นเหตุให้ RX CONTROL BLOCK จะทำการเลื่อนข้อมูลอีกครั้งซึ่งเป็นครั้งสุดท้าย แล้วทำการโหลดค่าข้อมูลที่รับเข้ามาให้กับ SBUF สัญญาณ RECEIVE จะถูกเคลียร์ RI จะถูกเซทที่ S1P1 ของแมชีนไซเคิลที่ 10 หลังจากบิท RI ถูกเคลียร์ในการเริ่มต้น

<u>รายละเอียดเพิ่มเติมเกี่ยวกับ MODE 1 (ดูรูปที่ 2.15)</u>

โหมด 1 จะเป็นการส่งรับข้อมูลขนาด 10 บิท โดยมี 1 START BIT (0),8 บิท

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ข้อ มูลและ 1 STOP บิท (1) การส่งจะเริ่มต้นเมื่อมีการใช้คำสั่งที่ต้องใช้ SBUF เป็นปลาทางเช่น MOV SBUF,A ในขณะที่เขียนข้อมูลไปที่ SBUF บิทที่ 9 ของ TX SHIFT REGISTOR จะถูกโหลดด้วย 1ในทางส่งจะมีการทำงานเหมือนกับโหมด 0 ดังที่ได้อธิบายมาข้างต้น ส่วนทางด้านรับจะเริ่มตั้งแต่ตรวจพบขอบขาลง (1 TO 0) ของ RXD เมื่อตรวจพบขอบขาลงของสัญญาณทางขา RXD ข้อมูล 1FFH จะถูกโหลดให้ INPUT SHIFT REGISTOR ข้อมูลจะถูกสุ่มตัวอย่าง 3 ครั้ง และข้อมูลที่ได้คือข้อมูลจากตัวอย่าง 2 ใน 3 ที่สุ่มมาได้เหตุที่ทำอย่างนี้เพื่อป้องกันสัญญาณรบกวนที่อาจเกิดขึ้นได้ในระหว่างบิทแรก (START BIT) ถ้าตัวอย่างที่สุ่มมาได้ไม่ใช่ 0 วงจรส่วนรับข้อมูลจะถูกรีเซ็ทและจะกลับไปรอรับการเกิด SHIFT BIT อีกครั้ง ในกรณีที่ START BIT เกิดขึ้นถูกต้องข้อมูลจะถูก SHIFT เข้าไปใน INPUT SHIFT REGISTER และจะรับข้อมูลส่วนที่เหลือต่อไป ในขณะที่บิทของข้อมูลถูกเลื่อนเข้ามาทางขวา "1" จะถูกเลื่อนออกทางซ้าย เมื่อ START BIT มาถึงตำแหน่งซ้ายสุดของ SHIFT REGISTOR จะทำให้ RX CONTROL BLOCK ทำการเลื่อนอีกพร้อมทั้งโหลด SBUF และ RB8 และเซ็ทแฟลก RI สัญญาณที่โหลด SBUF,RB และเซ็ท RI จะเกิดขึ้นได้ก็ต่อเมื่อในช่วงเวลาการเลื่อนข้อมูลครั้งสุดท้ายจากสภาวะต่อไปนี้

1. RI =0

2. SM2 = 0 หรือ THE ERCEIVED STOP BIT = 1

ถ้าไม่พบสภาวะทั้งสองอย่างนี้การรับข้อมูลนั้นถือว่าล้มเหลว ถ้าสภาวะทั้งสองเกิดขึ้นถูกต้อง STOP BIT จะเข้าไปที่ RB8 ข้อมูล 8 บิทจะเข้าไปที่ SBUF และ RIจะถูกกระตุ้นเมื่อถึงขั้นตอนนี้ไม่ว่าสภาวะทั้งสองจะเกิดขึ้นหรือไม่ระบบจะกลับเข้าสู่การคอย START BIT ต่อไป

<u>รายละเอียดเกี่ยวกับโหมด 2 และ 3</u>

เป็นการรับส่งข้อมูล 11 บิท ผ่านทาง RXD และ TXD โดยมีรูปแบบการส่งดังนี้ START BIT (0),8 DATA BIT (LSB ออกก่อน) บิทที่ 9 ซึ่งโปรแกรมได้ และ STOP BIT (1) ในการส่งข้อมูลบิทที่ 9 (TD8 ใน SCON) สามารถที่จะกำหนดให้เป็น 0 หรือ 1ได้ส่วนทางด้านรับข้อมูลบิทที่ 9 จะเข้าไปที่ RB8 (ในริจิสเตอร์ SCON)ในโหมด 2 ความเร็วในการส่งสามารถโปรแกรมให้เป็น 1/32 หรือ 1/64

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ของความถี่ออสซิลเลเตอร์ได้ด้วยบิท SMODในรีจิสเตอร์ PCON ส่วนโหมด 3 ความเร็วในการส่งถูกกำหนดโดย TIMER0 หรือ TIMER1 ขึ้นอยู่กับ TCLK และ RCLK รูปที่ 2.13 และ 2.14 แสดงแผนภาพเวลา และการทำงานอย่างง่ายของพอร์ตอนุกรมโหมด 2 และ 3 ตามลำดับการทำงานของภาครับจะเหมือนกับโหมด 1 ทุกประการยกเว้นภาคส่งที่มีส่วนแตกต่างกับโหมด 1 คือมีส่วนของบิทที่ 9 ของ TRANSMIT SHIFT REGISTER

เมื่อสัญญาณการส่งเริ่มขึ้น สัญญาณนาฬิกาการที่ใช้ในการเลื่อนข้อมูล จะให้ข้อมูลบิท 1 กับบิทที่ 9 ของ SHIFT REGISTER (เพื่อเป็น STOP BIT) หลังจากนั้นข้อมูล 0 จะถูกเลื่อนเข้ามาจากทางซ้าย (บิทข้อมูลจะเลื่อนออกทางขวา) เมื่อบิท TB8 มาถึงตำแหน่ง OUTPUT ของ SHIFT REGISTER บิทหยุด (STOP BIT) จะเข้ามาต่อท้ายบิท TB8 และ ขณะนี้ข้อมูลทางซ้ายจะเป็น 0 ทั้งหมด ในสภาวะนี้ TX CONTROL UNIT จะทำการเลื่อนข้อมูลอีก 1 ครั้งเป็นครั้งสุดท้าย

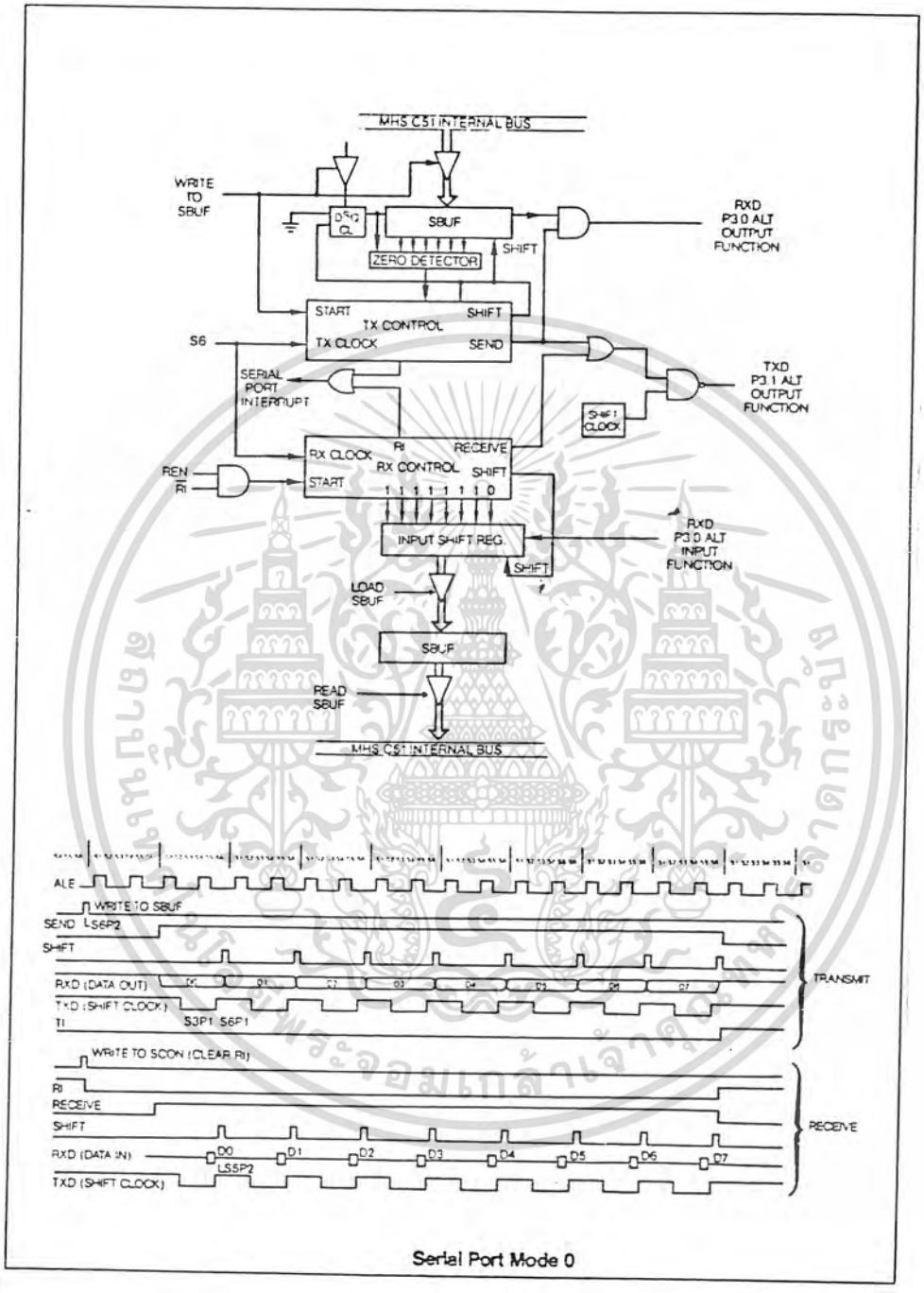
ทางด้านรับเมื่อ START BIT ถูกเลื่อนจนถึงOUTPUT ของ SHIFT REGISTER (ในโหมด 2 และ 3)จะทำให้ ข้อมูลถูกเลื่อนอีก 1 ครั้งซึ่งเป็นครั้งสุดท้าย ต่อจากนั้นจะโหลดข้อมูลให้ SBUF และ RB8 ต่อจากนั้นจำทำการเซทบิท RI ขบวนการเหล่านี้จะเกิดก็เมื่อ

1. RI =0 และ
2. SM2 = หรือ ข้อมูลบิทที่ 9 =1

<u>สรุป</u> ข้อมูลจะเข้ามาที่ SBUF และแฟลก RI จะเซ็ทก็ต่อเมื่อ RI สภาวะเดิมต้องเป็น 0 อยู่และอีกสภาวะคือ SM2 ต้องเป็น 0 หรือไม่เช่นนั้นข้อมูลบิทที่ 9 ต้องเป็น 1

<u>ข้อสังเกต</u> คือ เมื่อ RIให้สัญญาณอินเตอร์รัมท์แล้วในส่วนของโปรแกรมต้องเคลียร์ RI ให้ด้วย ส่วนโหมด 2 และ 3 ต้องริเซ็ทข้อมูลในบิทที่ 9 ต้องเป็น 1 ยกเว้นการใช้งานแบบ MULTI PROCESSOR COMMUNICATION ดังที่ กล่าวมาข้างต้น

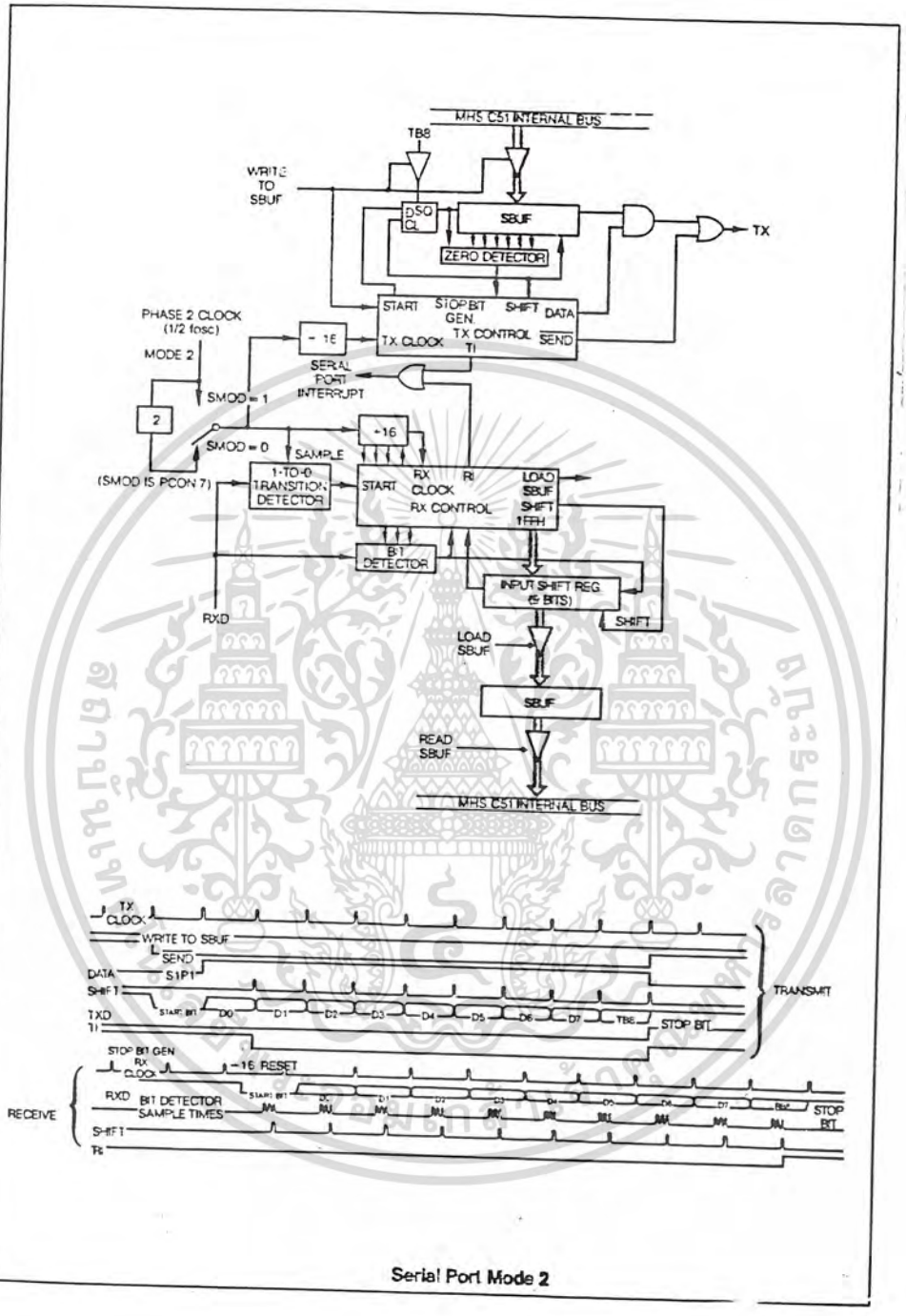
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



Serial Port Mode 0

รูปที่ 2.12 SERIAL PORT MODE 0

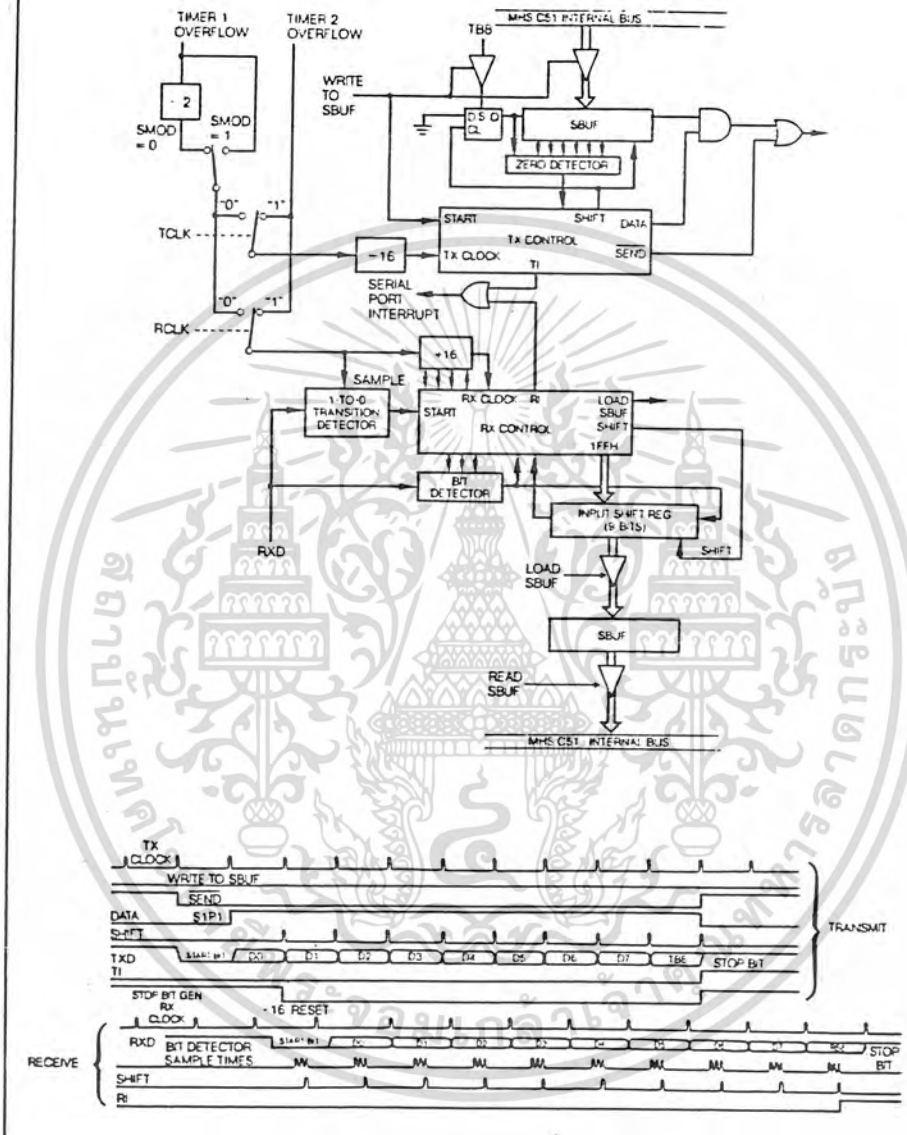
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



Serial Port Mode 2

รูปที่ 2.13 SERIAL PORT MODE 2

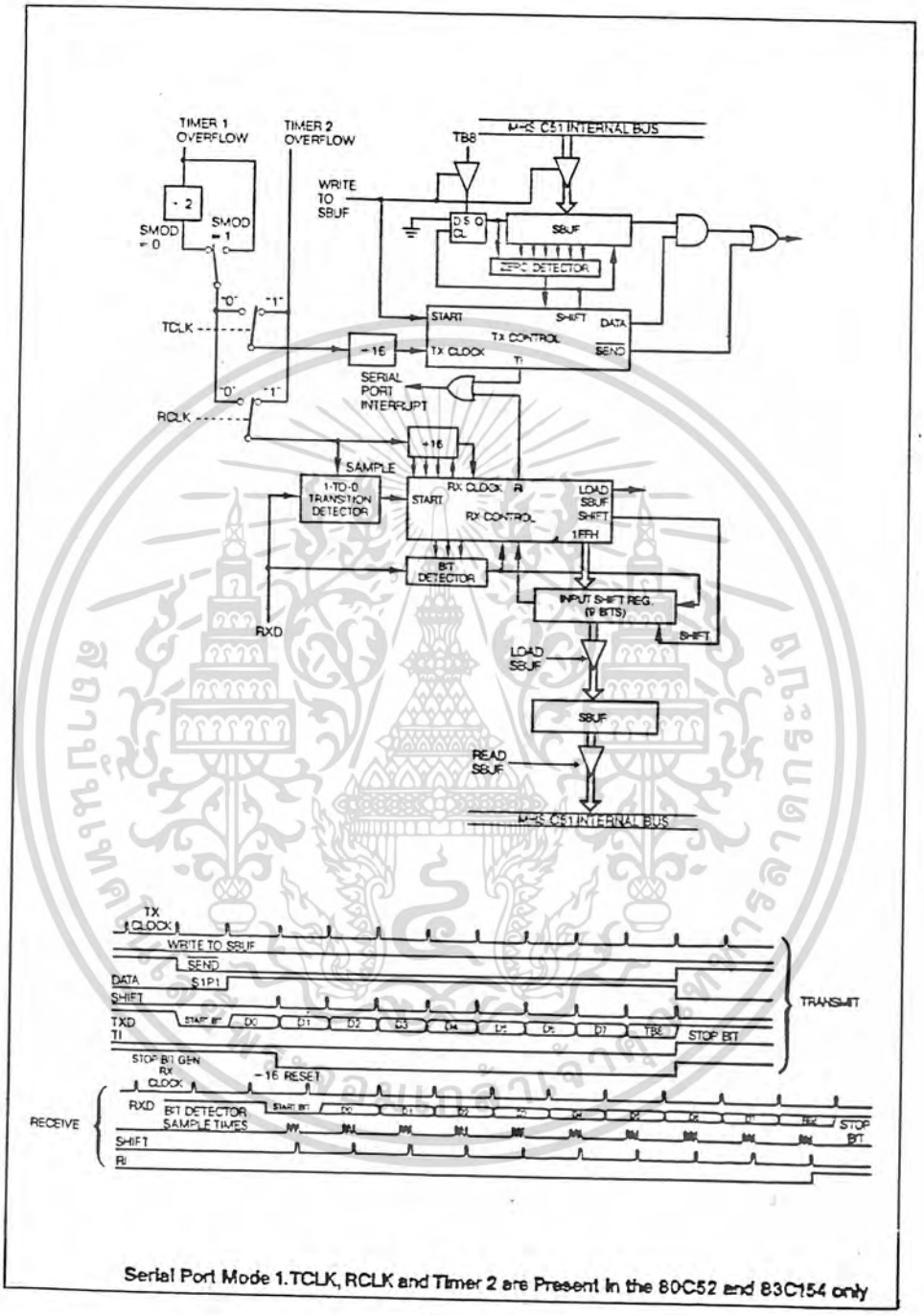
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



Serial Port Mode 3. TCLK, RCLK, and Timer 2 are Present in the 80C52/80C32 and 83C154/80C154

รูปที่ 2.14 SERIAL PORT MODE 3

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



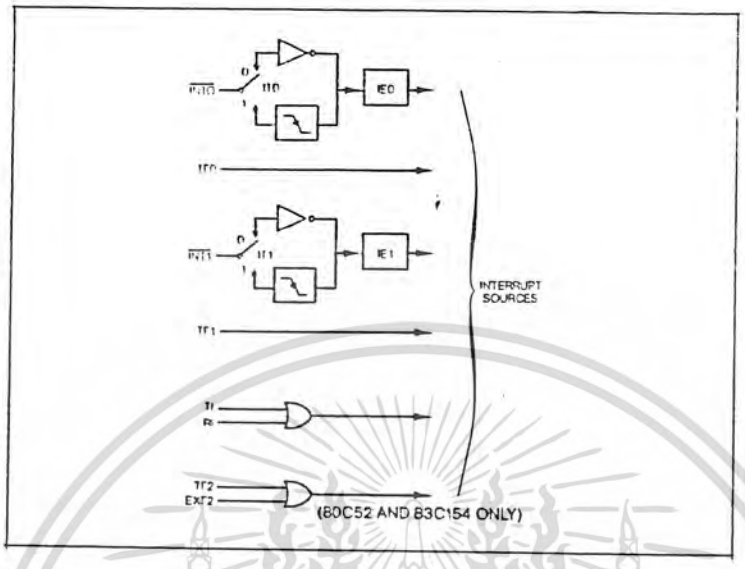
Serial Port Mode 1.TCLK, RCLK and Timer 2 are Present in the 80C52 and 83C154 only

รูปที่ 2.15 SERIAL PORT MODE 1

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.2.5 อินเทอร์เทอรัพท์

80C31 จะมีแหล่งกำเนิดสัญญาณอินเทอร์รัพท์ 5 อย่าง ดังในรูปที่ 2.16



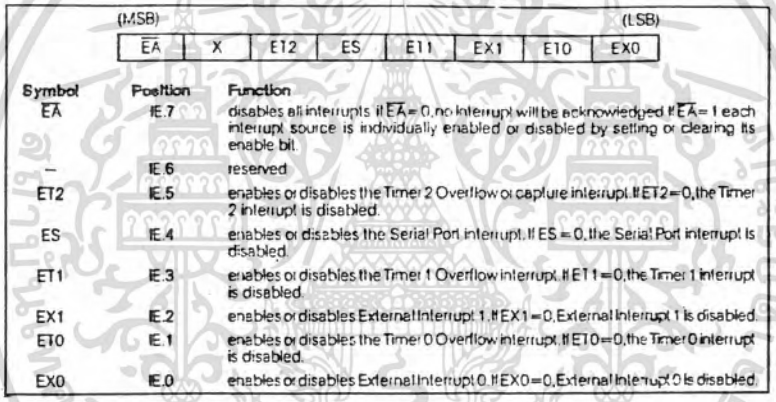
รูปที่ 2.16 INTERRUPT SOURCE

ขาอินเตอร์รัพท์ภายนอกมีเพียง 2 ขา คือ INT0 และ INT1 ซึ่งสามารถโปรแกรมให้เป็นแบบกระตุ้นด้วยระดับสัญญาณ (LEVEL-ACTIVATED)หรือกระตุ้นด้วยขอบของสัญญาณ (TRANSITION-ACTIVATED) ขึ้นอยู่กับบิท IT0 และIT1 ในรีจิสเตอร์ TCON แฟลกที่กำเนิดสัญญาอินเตอร์รัพท์ที่แท้จริงคือIE0 และIE1ในTCON เมื่อมีสัญญาณอินเตอร์รัพท์จากภายนอกแฟลกที่กำเนิด สัญญาณอินเตอร์รัพท์จะถูก เซ็ท เคลียร์โดยฮาร์ดแวร์ (ภายใน 8031)ขณะที่ CPU กระโดดไปทำในอินเตอร์รัพท์รูทีนโดยที่ต้องโปรแกรมให้รับอินเตอร์รัพท์แบบขอบของสัญญาณ ถ้าชนิดของอินเตอร์-รัพท์ถูกเซ็ทให้เป็นการอินเตอร์รัพท์โดยระดับของสัญญาณแฟลกอินเตอร์รัพท์ต้อง CLEAR ด้วยเซอร์ฟแวร์ ภายในซับรูทีนของอินเตอร์รัพท์

อินเตอร์รัพท์ของ TIMER 0 และ TIMER1 เกิดขึ้นโดยแฟลก TF0 และ TF1 เมื่อเกิดอินเตอร์รัพท์ขึ้นแฟลกจะถูกเครียร์โดยไปทำงานที่ SERVICE ROUTINE อินเตอร์รัพท์ของพอร์ตอนุกรมเกิดขึ้นจากทางรับหรือทางส่งข้อมูลโดยที่โปรแกรมต้องตรวจเช็คว่าเป็นอินเตอร์รัพท์จากด้านรับ(RI)หรือด้านส่ง(TI)และจะต้องทำเคลียร์แฟลก อินเตอร์รัพท์ด้วยซอร์ฟแวร์ บิทที่กำเนิดสัญญาณอินเตอร์รัพท์สามารถเซ็ทหรือ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เคลียร์ได้โดยซอร์ฟแวร์โดยจะให้ผลเหมือนกับสัญญาณที่กระทำโดยฮาร์ดแวร์นั่นหมายความว่าสัญญาณอินเตอร์รัพท์สามารถจะเกิดขึ้นหรือ อินเตอร์รัพท์ที่ค้างอยู่สามารถยกเลิกได้โดยซอร์ฟแวร์ แหล่งกำเนิดสัญญาณอินเตอร์รัพท์แต่ละตัวสามารถจะ ENABLE หรือ DISABLEโดยการเซ็ทหรือเคริยร์บิทที่ในรีจิสเตอร์ IE ข้อสังเกตุในรีจิสเตอร์ IE บิทที่7 คือ EA จะเป็นตัวควบคุมการ ENABLE หรือ DISABLE ของสัญญาณอินเตอร์รัพท์ทุกสัญญาณ ฉะนั้นเมื่อต้องการใช้อินเตอร์รัพท์ต้องไม่ลืมที่จะเซ็ทบิท EA ด้วย หลักจากนั้นจึงทำการ ENABLE สัญญาณอินเตอร์รัพท์ที่ต้องการ



(MSB) (LSB)

$\overline{EA}$	X	ET2	ES	ET1	EX1	ET0	EX0

Symbol	Position	Function
$\overline{EA}$	IE.7	disables all interrupts. If $\overline{EA}$ = 0, no interrupt will be acknowledged. If $\overline{EA}$ = 1 each interrupt source is individually enabled or disabled by setting or clearing its enable bit.
–	IE.6	reserved.
ET2	IE.5	enables or disables the Timer 2 Overflow or capture interrupt. If ET2=0, the Timer 2 interrupt is disabled.
ES	IE.4	enables or disables the Serial Port interrupt. If ES = 0, the Serial Port interrupt is disabled.
ET1	IE.3	enables or disables the Timer 1 Overflow interrupt. If ET1=0, the Timer 1 interrupt is disabled.
EX1	IE.2	enables or disables External Interrupt 1. If EX1=0, External Interrupt 1 is disabled.
ET0	IE.1	enables or disables the Timer 0 Overflow interrupt. If ET0=0, the Timer 0 interrupt is disabled.
EX0	IE.0	enables or disables External Interrupt 0. If EX0=0, External Interrupt 0 is disabled.

รูปที่ 2.17 IE : INTERUPT ENABLE REGISTER

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ลำดับความสำคัญของการอินเตอร์รัพท์

แหล่งสัญญาณอินเตอร์รัพท์แต่ละสัญญาณ สามารถโปรแกรมได้ว่าให้เป็นลำดับความสำคัญสูงหรือลำดับความสำคัญต่ำ โดยที่ลำดับความสำคัญต่ำจะถูกอินเตอร์รัพท์ ด้วยสัญญาณอินเตอร์รัพท์ที่มีความสำคัญสูงกว่า และลำดับความสำคัญสูงจะไม่ถูกอินเตอร์-รัพท์โดยสัญญาณอินเตอร์รัพท์ที่มีความสำคัญต่ำกว่า

ถ้ามีการอินเตอร์รัพท์ด้วยลำดับความสำคัญเท่ากันมากกว่า 1 สัญญาณ CPU จะทำการตรวจ(POLLING) และ ตัดสินใจว่าจะให้บริการกับสัญญาณอินเตอร์รัพท์ตัวใดฉะนั้นในแต่ระดับความสำคัญ ยังมีการจัดลำดับความสำคัญไว้อีกดังลายละเอียดข้างล่าง

SOURCE	PIORITY WITHIN LEUEL
1. IE0	(HIGHEST)
2. TF0	
3. IE1	
4. TF1	
5. RI+TI	
6. TF2+EXF2	(LOWEST)

หมายเหตุ ลำดับความสำคัญที่ใช้เฉพาะเมื่อมีสัญญาณอินเตอร์รัพท์ในลำดับความสำคัญเท่ากันมากกว่า 1 สัญญาณ

การทำงานของสัญญาณอินเตอร์รัพท์

แฟลกของสัญญาณอินเตอร์รัพท์จะถูกสุ่มตัวอย่างในเสตท 5 เฟสที่ 2 ของทุกๆแมทชีนไซเคิลและจะทำการตรวจ (POLLING) การอินเตอร์รัพท์จาก 5 แหล่งสัญญาณในแมชีนไซเคิลต่อมาถ้า ผลของการสุ่มตัวอย่างพบว่าแฟลกอินเตอร์รัพท์ถูกเซ็ทในเสตทที่ 5 เฟสที่ 2 ของแมชีนไซเคิลที่ผ่านมาแล้ว จะมีการเรียกไปยังส่วนของโปรแกรมบริการอินเตอร์รัพท์ หากว่าไม่ถูกขัด ขวาง ด้วยสภาวะหนึ่งดังต่อไปนี้

1. กำลังทำคำสั่งอยู่ในโปรแกรมบริการอินเตอร์รัพท์ที่มีความสำคัญเท่ากันหรือสูง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กว่า

2. ไม่ใช่ไซเคิลสุดท้ายของคำสั่งที่กำลังปฏิบัติ

3. คำสั่งที่ปฏิบัตินั้นคือ RETI หรือ คำสั่งที่ติดต่อกับรีจิสเตอร์ IE หรือ IP

ในสภาวะตามข้อ 2. เพื่อเป็นการประกันว่าคำสั่งที่ปฏิบัติถึงไซเคิลสุดท้ายแล้วจะไม่ถูกอินเตอร์รัพท์จนกว่าจะปฏิบัติคำสั่งนั้นจนจบเสียก่อน

ตามข้อ 3. นั้นในกรณีที่ CUP กำลังทำคำสั่ง RETI หรือกำลังติดต่อกับ IE หรือ IP ตัวใดตัวหนึ่งอยู่แล้วเกิดอินเตอร์รัพท์ขึ้น CPUจะยอมให้มีการอินเตอร์รัพท์แต่ต้องปฏิยัติอย่างน้อยอีก 1 คำสั่งหลังจากทำคำสั่ง IE,IP หรือ RETI ตัวอย่าเช่น ถูกอินเตอร์รัพท์ในขณะที่กำลังทำคำสั่ง RETI หน่วยประมวลผลจะสั่งแอดเดรสให้ PC หลังจากคำสั่ง RETI และปฏิบัติอีก 1 คำสั่งในโปรแกรมหลักต่อจากนั้นจึงจะตอบสนองการอินเตอร์รัพท์ CPU รับรู้การอินเตอร์รัพท์ โดยกระโดดไปทำโปรแกรมบริการอินเตอร์รัพท์ ในบางกรณีที่ไม่เคลียร์แฟลกส่วนของโปรแกรมของผู้ใช้จะต้องมีคำสั่งเคลียร์แฟลกเอง เช่น การอินเตอร์รัพท์ที่เกิดจากพอร์ตอนุกรม ส่วนสัญญาณอินเตอร์รัพท์จากภายนอก แฟลกจะถูกเคลียร์ให้ถ้าเป็นการโปรแกรมให้รับการอินเตอร์รัพท์แบบการเปลี่ยนแปลงของสัญญาณ การตอบสนองสัญญาณอินเตอร์รัพท์ CPU จะกระโดดไปที่ตำแหน่งของโปรแกรมบริการอินเตอร์รัพท์ตาม ชนิดของอินเตอร์รัพท์ดังนี้

SOURCE	VECTOR ADDRESS
IEO	0003H
TFO	000BH
IE1	0013H
TF1	001BH
R1+T1	0023H

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การอินเตอร์รัพท์จากภายนอก

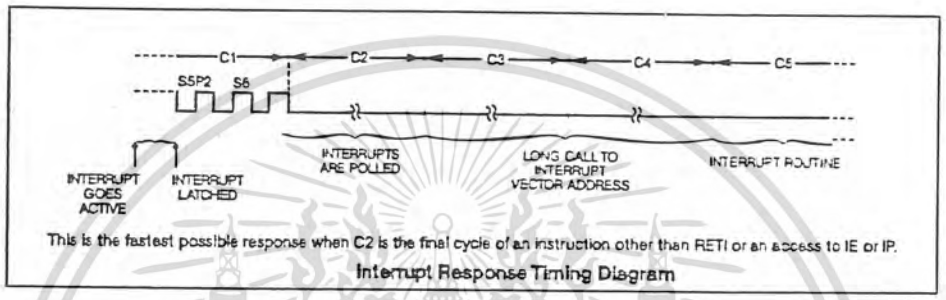
การอินเตอร์รัพท์ภายนอกกระทำได้ 2 อินพุทคือ INTTO และ INT1 โดยสามารถโปรแกรมให้เป็นแบบการเปลี่ยนแปลงขอบสัญญาณหรือเป็นแบบระดับสัญญาณก็ได้โดยเซ็ทหรือเคลียร์บิท IT1 หรือ IT0 ในริจิสเตอร์ TCON ถ้า ITx=0 จะเป็นการอินเตอร์รัพท์แบบระดับสัญญาณ (LOW) ถ้า ITx=1 จะเป็นการรับอินเตอร์รัพท์แบบการเปลี่ยนแปลงขอบของสัญญาณและในกรณีนี้ขา INTx ใน TCON ฉะนั้นสัญญาณที่เป็น HIGH 1 ไซเคิล และ LOW ในไซเคิลต่อไปส่วนของอินเตอร์รัพท์จะเซ็ท IEx ใน TOCN ฉะนั้นสัญญาณที่เป็น HIGH และ LOW ที่กล่าวมาจะต้องมีค่าอย่างน้อย12 คาบเวลาของความถี่ออสซิลเลเตอร์และ ITx จะถูกเคลียร์เมื่อโปรแกรมกระโดดไปทำงานในส่วนของเซอร์วิสรูทีนโดยอัตโนมัติ ถ้าสัญญาณอินเตอร์รัพท์จากภายนอกเป็นการอินเตอริรัพท์แบบระดับสัญญาณ (LOW) วงจรอินเตอร์รัพท์ภายนอกต้องรักษาระดับสัญญาณ 0 จนกว่าส่วนบริการอินเตอร์รัพท์จะทำงานและต้องถอนตัวจากการอินเตอร์รัพท์ก่อนที่ CUP จะเสร็จสิ้นโปรแกรมบริการอินเตอร์รัพท์

เวลาในการตอบสนองการอินเตอร์รัพท์

ถ้าการอินเตอร์รัพท์จากภายนอกเกิดขึ้นในภาวปกติ CPU จะใช้เวลาตั้งแต่เซ็ทอินเตอร์รัพท์แฟลก ตรวจสอบ (POLLING) ไปจนถึงกระโดดไปทำคำสั่งของซับรูทีนอย่างน้อยที่ สุด 3 แมชีนไซเคิล ในบางสภาวะจะใช้เวลามากกว่า 3 แมชีนไซเคิล ถ้าผลของอินเตอร์รัพท์ถูกขัดขวางด้วยสภาวะใดสภาวะหนึ่งใน 3 ข้อจากที่กล่าวมาแล้วข้างต้น เช่น ถ้าผลของการอินเตอร์รัพท์ในขณะที่ CPU ทำคำสั่งอยู่ซับรูทีนของอินเตอร์รัพท์ที่ลำดับความสำคัญเท่ากันหรือสูงกว่าเวลาจะมากหรือน้อยขึ้นอยู่กับโปรแกรมในซับรูทีนของอินเตอร์รัพท์ที่กำลังอยู่ถ้าคำสั่งที่กำลังดำเนินนั้นไม่ใช่ แมชีนไซเคิลสุดท้ายเวลาในการตอบสนองการอินเตอร์รัพท์จะมากขึ้นแต่จะไม่เกิน 3แมชีนไซเคิลเพราะว่าคำสั่งที่ยาวที่สุด (คูณและหาร)จะยาวเพียง 4 แมชีนไซเคิลและถ้าคำสั่งที่กำลังดำเนินอยู่เป็นคำสั่ง RETI หรือคำสั่งติดต่อกับรีจิสเตอร์ IE หรือ IP เวลาในการตอบสนองการอินเตอร์รัพท์จะเพิ่มขึ้นแต่ไม่มากกว่า 5 แมชีนไซเคิล (จะต้องทำอย่างมากที่สุดอีก 1 แมชีนไซเคิลสำหรับทำให้คำสั่ง (REIT) จบสมบูรณ์และกับอีก 4 แมชีนไซเคิลสำหรับคำสั่งที่ยาวที่ยาวสุด อีก 1 คำสั่ง)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

<u>สรุป</u> สัญญาณอินเตอร์รัพท์เดียว (โดยไม่ซ้อนกับอินเตอร์รัพท์อื่น) จะใช้เวลาในการตอบสนองมากกว่า 3 แมชีนไซเคิล และน้อยกว่า 8 แมชีนไซเคิล



รูปที่ 2.18 INTERUPT RESTPONSE TIMINGDIAGRAM

<u>การรีเซ็ท</u>

สัญญาณริเซ็ทเป็นสัญญาณอินพุททางขา 9 การรีเซ็ทจะสมบูรณ์ต้องรักษาระดับ HIGH อย่างน้อยที่สุด 2 แมชีนไซเคิล (24 คาบเวลาของออสซิลเลเตอร์) การรีเซ็ทภายใน ตัว CPU จะเริ่มในระหว่างไซเคิลที่ 2 นับตั้งแต่ขา RST เป็น HIGH ผลของการรีเซ็ทจะมีผลกับรีจิสเตอร์ ดังต่อไปนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

REGISTOR	CONTENT
PC	0000H
ACC	00H
B	00H
PSW	00H
SP	07H
DPTR	0000H
P0-P3	FFH
IP	(XX000000)
IE	(XX000000)
TMOD	00H
TCON	00H
TH0	00H
TL0	00H
TH1	00H
SCON	00H
SBUF	00H
PCON	00H

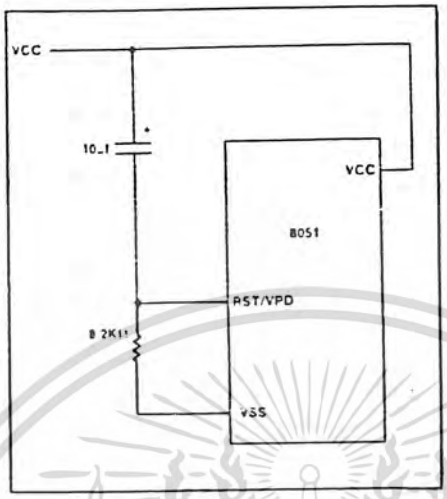
RAM ภายในจะไม่ถูกเคลียร์เมื่อ CUP ถูกริเซ็ท

<u>การริเซ็ทเมื่อเปิดเครื่อง</u>

เมื่อจ่ายไฟเข้าระบบควรมีการริเซ็ท CPU ก่อนเพื่อรอให้ทั้งระบบอยู่ในสภาวะพื้นที่จะทำงานซึ่งทำได้โดยต่อ C ขนาด 10UF จาก VCC มาที่ขา 9 (RST)และจากขา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

9 ต่อ R ขนาด 8.2K ลงกราวด์



รูปที่ 2.19 POWER ON RESET CIRCUIT

IDLE MODE

8031 มีคำสั่งเซ็ท PCON.0 ซึ่ง CPU ทำคำสั่งนี้จะเป็นคำสุดท้ายก่อนจะเข้าสู่ IDLE โหมด (ไม่ทำงาน) IDLE โหมดนี้สัญญาณนาฬิกาภายในจะหยุดส่งให้ CPU แต่ยัง จ่ายสัญญาณนาฬิกาให้กับอินเตอร์รัพท์,TIMER และพอร์ตอนุกรมสถานะต่าง ๆ ของ CPU ถูกเก็บไว้และรีจิสเตอร์ต่าง ๆ ยังคงรักษาข้อมูลเดิมไว้ในระหว่างอยู่ใน IDEL โหมด การที่จะทำให้ CPU กลับสู่โหมดปกติทำได้ 2 อย่างคือทำการอินเตอร์รัพท์จากภายนอกเมื่อ CPU ถูกอินเตอร์รัพท์จะให้บริการอินเตอร์รัพท์เมื่อจบบริการอินเตอร์รัพท์ CPU จะกลับมาหาคำสั่งต่อไปของโปรแกรมซึ่งต่อจากคำสั่งที่ทำ IDLE โหมด การประยุกต์ใช้งานเราอาจจะใช้ประโยชน์จาก GF1,GF0 ซึ่งเป็นแฟลกที่ใช้ในจุดประสงค์ทั่ว ๆไปโดยการเซ็ตบิทใดบิทหนึ่งหรือทั้งสองก่อนคำสั่งIDLE โหมด และเมื่อถูกอินเตอร์รัพท์ในโปรแกรมบริการอินเตอร์รัพท์ตรวจดูว่าเป็นการอินเตอร์รัพท์หลังจากทำ คำสัง IDLE โหมดหรือไม่เพื่อที่จะให้บริการที่ถูกต้อง อีกวิธีหนึ่งที่จะออกจาก IDLE โหมดได้คือ ทำการรีเซ็ท CPU ทางขารีเซ็ท

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รายละเอียดของขาต่าง ๆ

VCC	:	ต่อกับไฟเลี้ยงของระบบ (5VDC)
VSS	:	กราวด์
PORTO	:	พอร์ต 0 เป็นอินพุท/เอ้าท์พุทพอร์ตแบบ 8 บิท (OPEN DRAIN) สามารถรับกระแส SINK จาก LSTTL ได้ 8 ตัว ถ้าเราเขียน 1 ไปที่พอร์ต 0 จะทำให้พอร์ต 0 เป็น HI-Z อินพุทพอร์ต 0 สามารถ MULTIPLEXระหว่างบัสข้อมูลกับแอดเดรสไบท์
PORT1	:	พอร์ต 1 เป็น 8 บิทสองทางมีการพูลอัพภายใน, พอร์ต 1 นี้ขับ TTL (LS)ได้ 4 ตัว เมื่อเขียนค่า 1H ไปที่พอร์ต 1 ขาของพอร์ต 1 จะเป็น HIGH โดยมีการพูลอัพภายในและสามารถใช้เป็นอินพุทพอร์ตได้
PORT2	:	เป็นอินพุท/เอ้าท์พุทพอร์ตแบบ 2 ทิศทางมีการพูลอัพภายในสามารถขับ TTL (LS) ได้ 4 ตัว และพอร์ต 2 นี้จะให้แอดเดรสไบท์สูงใน ขณะติดต่อกับหน่วยความจำภายนอก
PORT3	:	พอร์ต 3 เป็นอินพุท/เอ้าท์พุทพอร์ตแบบ 2 ทิศทางมีการพูลอัพภายในพอร์ต 3 ขับ TTL (LS) ได้ 4 ตัวและยังสามารถใช้ทำงานในลักษณะพิเศษดังรายละเอียดข้างล่าง

PORT PIN	ALTERNATE FUNCTION
P3.0	RXD(SERIAL INPUT PORT)
P3.1	TXD(SERIAL OUTPUT PORT)
P3.2	INTO (EXTERNAL INTERUPTO)
P3.3	INT1 (EXTERNAL INTERUPT1)
P3.4	TO (TIME O EXTERNAL INPUT)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

P3.5		T1 (TIME 1 EXTERNAL INPUT)
P3.6		WR (EXTERNAL DATA MEMORY WRITE STROBE)
P3.7		RD (EXTERNAL DATA MEMORY READ STROBE)
RST	:	ขาริเซ็ทสัญญาณ HIGH ที่ขาริเซ็ทนี้นาน 2 แมชีนไซเคิลจะเป็นการริเซ็ท CPU
ALE/PROG	:	ADDRESS LATCH ENABLE พัลส์สำหรับแลตช์ แอดเดรสไบท์ต่ำในระหว่างการติดต่อกับหน่วยความจำภายนอก ALE นี้ยังจ่ายความถี่คงที่ 1/6 ของความถี่ออสซิลเลเตอร์(ถ้าไม่ติดต่อกับหน่วยความจำภายนอก) และ ALE อีกหน้าที่หนึ่ง คือเป็นอินพุทรับพัลส์ (LOW) ในระหว่างโปรแกรม EPROM (8751)
PSEN	:	เป็นขาสัญญาณ READ STROBE ในขณะอ่านโปรแกรมจากภายนอก PSEN จะแอดทีฟ 2 ครั้งต่อ 1 แมชีนไซเคิล
EA/UPP	:	ต่อ LOW จะทำคำสั่งในโปรแกรมภายนอกถ้าต่อ HIGH จะใช้ โปรแกรมภายใน และใช้เป็นอินพุท รับ 21V. ในการโปรแกรม 8751H
XTAL1	:	เป็นขาอินพุทของอินเวอร์เตอร์ของภาคขยายความถี่
XTAL2	:	เป็นขาเอ้าท์พุทของอินเวอร์เตอร์ของภาคขยายความถี่

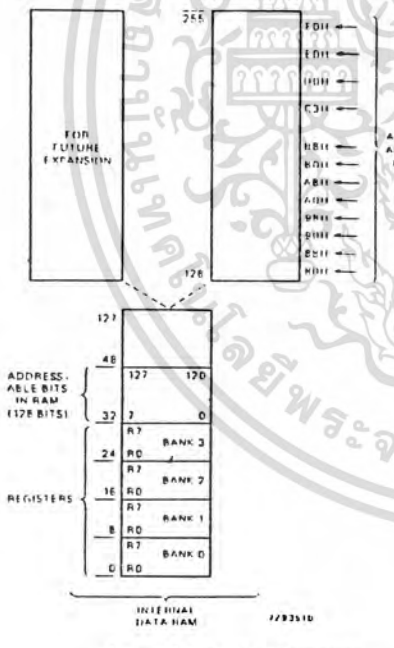
2.2.6 การจัดหน่วยความจำ

โปรแกรมแมมโมรี : ไมโครคอนโทรเลอร์ตระกูล MCS-51 จะแบ่งหน่วยความจำเป็นส่วนของโปรแกรมแมมโมรีและส่วนของข้อมูล (DATA) อย่างละ 64K ถ้า EA ต่อ HIGH จะเป็นการรันโปรแกรม ที่อยู่ภายในตัวโดยแอดเดรสจะไม่เกิน OFFFH และถ้า EA ต่อ LOW จะเป็นการรันโปรแกรมภายนอก ที่แอดเดรส 00 ถึง 23H จะเป็นส่วนของการอินเตอร์รัพท์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

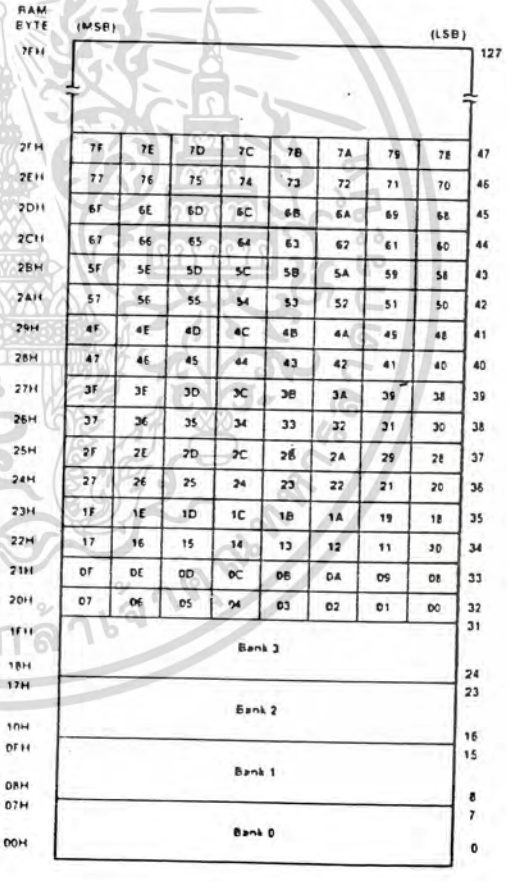
หน่วยความจำเก็บข้อมูล

แอดเดรสของข้อมูลจะประกอบด้วยหน่วยความจำภายใน และ ภายนอกโดยที่หน่วยความจำสำหรับข้อมูลภายนอกติดต่อได้โดยคำสั่ง MOVX และมี DPTRเป็นตัวชี้หน่วยความจำภายในถูกแบ่งเป็น 3 ส่วน คือ แอดเดรสด้านต่ำ 128 ไบท์ (RAM)และ SFR (SPECIAL FUNCTION REGISTER) ซึ่งแท้จริงก็คือ RAM อีก 128 ไบท์นั่นเอง แต่การเข้าถึงหน่วยความจำใน SFR นี้แตกต่างจาก 128 ไบท์ล่างหน่วยความจำทางด้านต่ำ (LOWER RAM 0-31H) ยังแบ่งเป็น 4 BANK โดยที่สามารถเข้าถึงได้ครั้งละ 1 BANK เท่านั้นแต่ละ BANK เลือกโดยเซ็ทบิท RO,R1ใน PSW หน่วยความจำอีก 16ไบท์ ตำแหน่ง 20H ถึง 47H ประกอบด้วยหน่วยความจำ ที่สามารถเข้าถึงแบบบิทได้อีก 128 บิท (รูป 2.21)



Internal data memory address space.

รูปที่ 2.20



RAM bit addresses.

รูปที่ 2.21

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ADDRESSING MODES

8031 แบ่งการเข้าถึงหน่วยความจำได้ 5 แบบ ดังรูปที่ 2.22

```
1.REGISTER ADDRESSING
        - R0-R7
        - ACC, B , CY (BIT), DPTR
2.DIRECT ADDRESSING
        - LOWER 128 BYTES OF INTERNAL RAM
        - SPEICIAL FUNCTION REGISTER
3.REGISTER INDIRECT ADDRESSING
        - INTERNAL RAM (@R1, @R0, SP)
        - EXTERNAL DATA MEMORY (@R1, @R0, @DPTR)
4.IMMEDIATE ADDRESSING
        - PROGRAM MEMORY
5.BASE-REGISTER PLUS INDEX-REGISTER INDIRECT ADDRESS
        - PROGRAM MEMORY (@DPTR+A, @PC+A)
```

รูปที่ 2.22.ADDRESSING MODE

REGISTER ADDRESSING

เป็นการติดต่อกับรีจิสเตอร์ทั้ง 8 ตัวในแต่ละ BANK โดยใช้ 3 บิทล่างของ OP-CODE เป็นตัวกำหนดรีจิสเตอร์ที่จะทำการติดต่อด้วย ACC, B, DPTR และ CY และการประมวลผลทางบูลีนถือว่าอยู่ในโหมด REGISTER ADDRESSING ตัวอย่าง เช่น MOV A, R0

DIRECT ADDRESS

เป็นเพียงวิธีเดียวที่จะเข้าถึง SFR (SPECIAL FUNCTION REGISTER) ได้ การติดต่อกับ RAM 128 ไบท์ล่างก็ใช้โหมดนี้ด้วย ตัวอย่างเช่น MOV A, SBUF

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

REGISTER-INDIRECT ADDRESSING

ในโหมดนี้ใช้ค่าที่อยู่ในริจิสเตอร์ R0 หรือ R1 (ใน BANK ที่เลือกไว้) เป็นตัวชี้ไปยังตำแหน่งต่าง ๆ ภายใน 256 ไบท์(RAM 128ไบท์ล่างหรือ 256 ไบท์ล่างของหน่วยความจำภายนอกข้อสังเกต คือ SFR ไม่สามารถติดต่อได้ด้วยวิธี REGISTER-INDIRECT นี้ส่วนการติดต่อกับหน่วยความจำข้อมูลภายนอก 64K นั้นใช้ตัวชี้ขนาด16 บิท (DPTR) คำสั่ง PUSH และ POP ก็ทำงานในโหมดนี้ด้วย (ใช้ SP เป็นตัวชี้) ตัวอย่าง MOV A,@R0

IMMEDIATE ADDRESSING

โหมดนี้อนุญาติให้ใช้ค่าคงที่เป็นส่วนหนึ่งของ OP-CODE

ตัวอย่าง MOV A,#01H

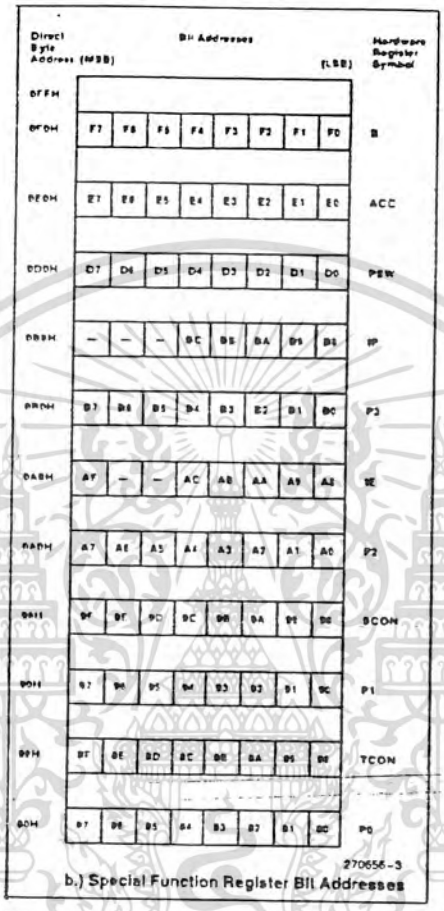
MOV DPTR,#1234H

BASE-REGISTER PLUS INDEX REGISTER-INDIRECT ADDRESSING

ในโหมดนี้จะใช้ค่าในแอคคิวมูเลเตอร์ A บวกกับเบสริจิสเตอร์ เช่น DPTR หรือ PC ประโยชน์ของโหมดนี้คือใช้ในการหาค่าที่ต้องการจากตาราง (TABLE)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

MOVC A, @A+PC



รูปที่ 2.23 SPECIAL FUNCTION REGISTER BIT ADDRES

ชุดคำสั่งของ 8031

8031 มีคำสั่งทั้งหมด 111 ประกอบด้วยคำสั่งไบท์เดียว 49 คำสั่ง 2 ไบท์ 45 คำสั่ง คำสั่ง 3 ไบท์ 17 คำสั่ง เราแบ่งคำสั่งตามหน้าที่การทำงานได้เป็น 4 แบบ

1.DATA TRANSFER

2.ARITHMETIC

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.LOGIC

4.CONTROL

1.1 DATA TRANSFER ยังแบ่งเป็น

- GENERAL PURPOSE
- ACCUMULATOR-SPECIFIC
- ADDRESS-OBJECT

ทั้งหมดนี้ไม่มีคำสั่งใดที่มีผลกระทบต่อ PSW ยกเว้นการ POP หรือ MOV โดยตรงไปที่ PSW

GENERAL PURPOSE TRANSFER

- MOV จะกระทำเป็นบิทหรือไบท์ก็ได้โดยข้อมูลจะย้ายจากต้นทาง (SOURCE) มายังปลายทาง (DESTINATION)
- PUSH คำสั่งนี้จะเพิ่มค่า SP ขึ้นไปอีก 1 ก่อนที่จะนำข้อมูลจากต้นทางไปเก็บไว้ในตำแหน่งที่ชี้โดย SP
- POP คำสั่งย้ายข้อมูลจากตำแหน่งที่ชี้โดย SP มายังปลายทาง และ SP จะลดลง 1

ACCUMULATOR SPECIFIC TRANSFER

- XCH (EXCHANGE) แลกเปลี่ยนข้อมูลของต้นทางกับแอคคิวมูเลเตอร์
- XCHD และเปลี่ยนค่า 4 บิทล่างของไบท์ข้อมูลกับข้อมูล 4 บิทล่างของแอคคิวมูเลเตอร์
- MOVX ทำการย้ายข้อมูลระหว่างแอคคิวมูเลเตอร์กับหน่วยความจำข้อมูลภายในออกโดยตำแหน่งของหน่วยความจำภายนอกกำหนดได้โดยค่าของ DPTR (16 บิท) หรือ รีจิเตอร์ R0, R1 (8 บิท)
- MOVC ย้ายข้อมูล 1 ไบท์จากหน่วยความจำของโปรแกรม (ในกรณีแยกหน่วยความจำของข้อมูลต่างหาก) มาไว้ที่ A โดยข้อมูลใน A ก่อนทำคำสั่งนี้จะรวมกับ PC หรือ DPTR เป็นตัวชี้ที่อยู่ของข้อมูลที่ต้องการ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตัวอย่าง REL-PC :

```
INC A
MOVC A, @A+PC
DB 66H
DB 77H
DB 88H
DB 89H
```

ถ้า SUBROUTINE นี้ถูกเรียกโดยที่ A มีค่า = 01H เมื่อออกจากรูทีนจะได้ค่า 77H อยู่ใน A

ARITHMETIC

8031 มีคำสั่งสนับสนุนการคำนวณเบื้องต้น เช่น บวก ลบ คูณ หาร แบบ 8 บิท โดยไม่คิดเครื่องหมาย

การบวก (ADDING)

- INC (INCREMENT) เพิ่มค่าในโอเปอร์แรนด์ขึ้นอีก 1
- ADD บวกค่าของโอเปอร์แรนด์ต้นทางกับแอคคิวมูเลเตอร์และเก็บค่าไว้ในแอคคิวมูเลเตอร์
- DA (DECIMAL-ADD-ADJUST FOR BCD ADDITION) เป็นการปรับแต่งผลรวมของการบวกเลข BCD และคืนผลลัพธ์นั้นให้รีจิสเตอร์ A แฟลก CY จะถูกเซ็ท ถ้าผลลัพธ์ที่ได้จากคำสั่ง DA มากกว่า 99 การทำงานคร่าว ๆของคำสั่งถ้าค่าที่จะทำการปรับแต่ง 4 บิทล่างของ A มากกว่า 9 หรือแฟลก AC = 1 จะทำการบวกด้วย 06H เช่นเดียวกัน ถ้า 4 บิทบนมากกว่า 9 ก็จะบวกด้วย 06H

การลบ (SUBTRACTION)

- SUBB (SUBTRACT WITH BORROW) ลบค่าในโอเปอร์แรนด์ตัวแรกด้วยโอเปอร์แรนด์ตัวที่ 2 และลบด้วย 1 ถ้า CY ถูก SET และคืนผลลัพธ์ให้แอคคิวมูเลเตอร์ A
- DEC (DECREMENT) ลดค่าในแอคคิวมูเลเตอร์ลง 1

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การคูณ (MULTIPLICATION)

- MUL เป็นการคูณกันระหว่างรีจิสเตอร์ A กับ B โดยที่ A จะรับค่าไบท์ต่ำและ B รับค่าไบท์สูงของผลคูณ OV จะถูกเคลียร์ถ้าข้อมูลครึ่งบนสุดของผลลัพธ์เป็น 0 ส่วน CY จะเคลียร์เสมอ

การหาร (DIVISION)

- DIV เป็นการหารระหว่างรีจิสเตอร์ A โดยริจิสเตอร์ B และ ให้ผลลัพธ์เลขจำนวนเต็มในรีจิสเตอร์ A ส่วนเศษของการหารจะอยู่ในรีจิสเตอร์ B การหารด้วยศูนย์จะไม่ให้ค่าผลลัพธ์ใน A และ B และ OV แฟลกจะถูกเซ็ทการทำคำสั่งหารนี้จะมีผลต่อแฟลกต่าง ๆ ดังนี้

- CY จะถูกเซ็ทถ้าผลของการหารทำให้เกิดตัวหดขึ้นจากบิทสูงสุด

- AC จะถูกเซ็ทถ้าผลของการหารทำให้เกิดตัวหดจาก 4 บิทล่างหรือเกิดการยืมบิทบนโดยบิทล่าง

- OV จะถูกเซ็ทถ้าจำนวนใด ๆ ถูกหารด้วย 0ในกรณีอื่น OV จะถูกเคลียร์ OV ถูกใช้ใน

การคำนวณแบบ TWO's COMPLEMENTเพราะว่า OV จะถูกเซ็ทเมื่อมีการใช้จำนวนที่มีเครื่องหมายและไม่แสดงผลใน 8 บิทได้

- P(PARITY) ถ้าผลลัพธ์ทำให้เกิดพาริตี้คี่ P จะไม่ถูกเซ็ท

LOGIC

8013 สามารถปฏิบัติการทาง LOGIC ได้ทั้งแบบบิทและไบท์

SINGLE-OPERAND OPERATION

- CLR เคลียร์รีจิสเตอร์ A ให้เป็น 0 หรือเคลียร์บิทใดบิทหนึ่งในรีจิสเตอร์หรือหน่วยความจำที่สามารถเข้าถึงแบบบิทให้เป็น 0 ได้

- SETB เซ็ทบิทต่าง ๆ ถ้าสามารถเข้าถึงแบบบิทได้ให้เป็น 1

- CPL คอมพลีเมนท์ค่าให้รีจิสเตอร์ A หรือคอมพลีเมนท์โดยตรงกับบิทใดบิทหนึ่งที่สามารถติดต่อแบบบิทได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- RL, RLC, RR, RRC, SWAP เป็นกลุ่มคำสั่งหมุน (ROTATE)ค่าในริจิสเตอร์ A SWAP ใช้เปลี่ยนค่าระหว่าง 4 บิทล่างกับ 4 บิทบนของริจิสเตอร์ A

TWO OPERAND OPERATIONS

- ANL คำสั่ง LOGIC"AND" แบบบิทหรือไบท์ด้วยรีจิสเตอร์ 2 ตัว โดยผลลัพธ์เก็บไว้ในริจิสเตอร์ตัวแรก
- ORL คำสั่ง LOGIC"OR" แบบบิทหรือไบท์โดยรีจิสเตอร์ 2 ตัว
- XOR คำสั่ง LOGIC"XOR"

CONTROL TRANSFER

8031 แบ่งการควบคุมการส่งผ่านข้อมูลออกเป็น 3 อย่างคือ CALL แบบไม่มีเงื่อนไข RETURN และ JUMP

UNCONDITION CALLS, RETURN AND JUMP

- ACALL และ LCALL เก็บค่า PC ของคำสั่งต่อไปลงในแสตคและเปลี่ยนการควบคุมระบบให้กับแอดเดรสปลายทางของคำสั่ง CALL
- LCALL เป็นคำสั่ง 3 ไบท์ ที่สามารถกระโดดไปยังตำแหน่งใด ๆ ภายใน 64K ได้
- ACALL เป็นคำสั่ง 2 ไบท์ ใช้เมื่อการ CALL ไปยังตำแหน่งที่ไม่ไกลเกิน 2K โดยการนำค่าของ A0-A10 ของตำแหน่งที่จะกระโดดไปมาเข้าโค๊ดในคำสั่งACALL

A10 A9 A8 1	0 0 0 1	A7 A6 A5 A4	A3 A2 A1 A0

<u>ข้อควรระวัง</u> ถ้าคำสั่ง AJMP อยู่ใน 2 ไบท์สุดท้ายของ PAGE (2K)เมื่อทำคำสั่งนี้ PAGE จะเปลี่ยนไปเพราะว่า PC จะต้องเพิ่มขึ้นอีก (AJMP = 2 ไบท์ ก่อนที่จะทำคำสั่งทำให้การกระโดดไม่ตรงตำแหน่งที่ต้องการเพราะในคำสั่ง AJMP จะเกี่ยวพันกัน A11-A15 ด้วยในกรณีนี้ A11 จะเปลี่ยนไป

- RET กลับสู่โปรแกรมหลักหลังจากการ CALL โดยการ POP ไบท์สูงให้ PC แล้ว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ลดค่า SP ลง 1 POP ไบท์ต่ำให้ PC และลดค่า SP ลงอีก 1

- AJMP LJMP และ SJMP คำสั่ง AJMP และ LJMP คล้ายกัน ACALL และ LCALL ส่วน SJMP (SHORT JUMP จะกระโดดได้เพียง 256 ไบท์โดยมีจุดศูนย์กลางอยู่ที่คำสั่งต่อจาก SJMP (-128 TO + 127)
- JMP @A+DPTR ใช้รีจิสเตอร์ A เป็นตัว OFFSET (0-255) โดยการบวกค่าในรีจิสเตอร์ A กับ DPTR

CONDITIONAL JUMP

เป็นการกระโดดแบบมีเงื่อนไขโดยระยะทางในการกระโดดอยู่ในช่วง 256 ไบท์โดยเดินหน้าและถอยหลังจากจุดศูนย์กลางอยู่ที่คำสั่งที่ต่อจากการกระโดดนั้น (-128 TO + 127)

- JZ กระโดดถ้าแอคคิวมูเลเตอร์เป็นศูนย์
- JNZ กระโดดถ้าแอคคิวมูเลเตอร์ไม่เท่ากับศูนย์
- JC กระโดดถ้าแฟลกตัวทดถูกเซ็ท
- JNC กระโดดถ้าแฟลกตัวทดไม่ถูกเซ็ท
- JB กระโดดถ้าบิทที่กำหนดถูกเซ็ท
- JNB กระโดดถ้าบิทที่กำหนดไม่ถูกเซ็ท
- JBC กระโดดถ้าบิทที่กำหนดถูกเซ็ทและทำการเคลียร์บิทที่กำหนดนั้นด้วย
- CJNE ทำการเปรียบเทียบโอเปอร์แรนด์ตัวแรกกับตัวที่ 2 และจะกระโดดถ้าโอเปอร์แรนด์ทั้งสองมีค่าไม่เท่ากัน แฟลก CY จะถูกเซ็ทถ้าโอเปอร์แรนด์ตัวแรกมีค่าน้อยกว่าตัวที่ 2
- DJNZ ลดค่าในโอเปอร์แรนด์และกระโดดถ้าผลของการลดค่านั้นไม่เท่ากับศูนย์

INTERUPT RETURNS

- RETI มีการทำงานเหมือนคำสั่ง RET แต่คำสั่ง RETI จะเพิ่มการ ENABLE อินเตอร์รัพท์ด้วย ในการดูคำสั่งของ 8031 จะมีสัญญาลักษณ์ที่ใช้ประกอบร่วมกับ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

คำสั่งดังต่อไปนี้

Rn - ริจิสเตอร์ R0-R7 ใน BANK ที่กำหนดไว้

DIRECT - ตำแหน่งของหน่วยความจำ (RAM) ภายใน (0-127) หรือ I/O พอร์ตริจิสเตอร์ควบคุม ฯลฯ

@Ri - ตำแหน่งของ RAM (0-225) ซึ่งถูกกำหนดโดยค่าของ R1 หรือ R0

#DATA - ค่าคงที่ขนาด 8 บิท

#DATA16 - ค่าคงที่ขนาด 16 บิท

addr 16 - ตำแหน่งปลายทางขนาด 16 บิท ใช้โดยคำสั่ง LCALL & LJMP

addr 11 - ตำแหน่งปลายทางขนาด 11 บิท ใช้โดยคำสั่ง ACALL & AJMP ซึ่งกระโดดได้ภายใน 2K ไบท์

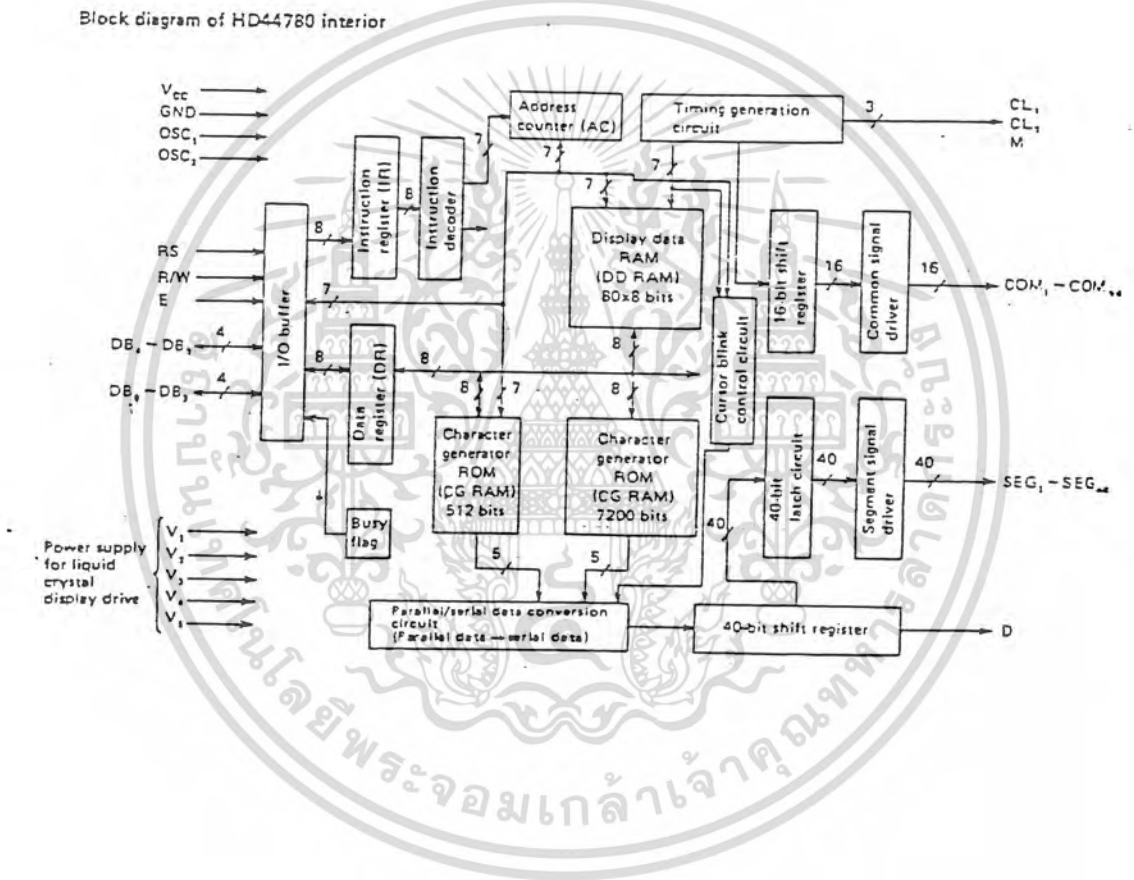
rel - ค่าความสัมพันธ์ (RELATIVE) โดยจะเป็นจำนวนที่มีเครื่องหมาย (-128 ถึง +127) ซึ่งจะสัมพันธ์กับตำแหน่งแรกของคำสั่งถัดไป

bit - ตำแหน่งของหน่วยความจำภายใน (RAM) ที่กำหนดได้โดยตรงหรือตำแหน่งในบิทต่าง ๆ ของ SFR

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.3 โครงสร้างหลักการทำงานของหน่วยแสดงผล

โครงสร้างภายในของวงจรควบคุมจอแสดงผล LCD



รูปที่ 2.24 โครงสร้างของ LCD

การใช้งานจอแสดงผล LCD นี้ จะมีลักษณะการใช้งานที่ง่ายและสะดวกโดยที่เราสามารถที่จะใช้รหัสต่าง ๆ โปรแกรมการใช้จอ LCD ได้โดยตรงรายละเอียดของ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

คำสั่งต่าง ๆ จะกล่าวดังต่อไปนี้

2.3.1 <u>คำสั่งเคลียร์หน้าจอ</u> (Clear Display)

	RS	R/W	DB7							DB0
รหัส	0	0	0	0	0	0	0	0	0	1

คำสั่งนี้จะเป็นการเขียนช่องว่าง (space) ซึ่งมีรหัสแอสกี้ (Ascll code) 2 H เข้าไปในตำแหน่งบนหน้าจอทั้งหมด และทำการเซทค่าของตำแหน่งอ้างอิงบนหน้าจอ (DD Ram Address) ให้มีค่าเป็นศูนย์ ตัวเคอร์เซอร์ (Cursor)จะกลับไปอยู่ตำแหน่งบนสุดซ้ายมือจอภาพ

2.3.2 <u>คำสั่งรีเทิร์นโฮม</u> (Return Home)

	RS	R/W	DB7							DB0
รหัส	0	0	0	0	0	0	0	0	1	*

* ไม่มีผล

คำสั่งนี้จะทำการเซทค่าของตำแหน่งอ้างอิงบนหน้าจอให้มีค่าเป็นศูนย์ ตัวเคอร์เซอร์จะกลับไปอยู่ตำแหน่งบนสุดซ้ายมือของจอภาพโดยที่ข้อมูลบนจอภาพจะไม่มีการเปลี่ยนแปลง

2.3.3 <u>คำสั่งเอนทรีโหมดเซท</u> (Entry Mode Set)

	RS	R/W	DB7							DB0
รหัส	0	0	0	0	0	0	0	1	I/D	S

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

คำสั่งนี้ การทำงานจะขึ้นอยู่กับค่าของบิทข้อมูล 2 บิท คือ I/D และ S บิท I/D จะเป็นตัวกำหนดว่า เมื่อเขียนหรืออ่านข้อมูลแล้วจะทำให้ตำแหน่งอ้างอิงบนจอภาพมีค่าเพิ่มขึ้นหรือลดลงหนึ่งตำแหน่งโดยถ้าบิทนี้ มีค่า = 1 เพิ่มมากขึ้นหนึ่งตำแหน่ง , = 0 ลดลงหนึ่งตำแหน่ง

บิท S : จะเป็นตัวกำหนดแสดงผล โดยถ้า S =1 จะเป็นการใส่ข้อมูลแล้วตัวของคอร์เซอร์อยู่กับที่ ตัวของข้อมูลจะถูกเลื่อนไปทางซ้าย และถ้า S =0 ตัวของข้อมูลจะอยู่กับที่แต่ตัวเคอร์เซอร์จะเลื่อนไปทางขวามือ

2.3.4 <u>คำสั่งเปิดปิดหน้าจอ</u> (Display On/Off Control)

	RS	R/W	DB7							DB0
รหัส	0	0	0	0	0	0	1	D	B	C

คำสั่งนี้การทำงานจะขึ้นอยู่กับค่าของบิทข้อมูล 3 บิท คือ D,C,B

บิท D : ถ้า D มีค่าเป็น 1 จะเป็นการเปิดหน้าจอ
ถ้า D มีค่าเป็น 0 จะเป็นการปิดหน้าจอ

บิท C : ถ้า C มีค่าเป็น 1 จะเป็นการแสดงเคอร์เซอร์
ถ้า C มีค่าเป็น 0 จะเป็นการไม่แสดงเคอร์เซอร์

บิท B : ถ้า B มีค่าเป็น 0 เป็นการกำหนดให้เคอร์เซอร์ไม่กระพริบ
ถ้า B มีค่าเป็น 1 เป็นการกำหนดให้เคอร์เซอร์กระพริบ

2.4.5 <u>คำสั่งเลื่อนเคอร์เซอร์หรืออักขระ</u>(Cursor Display Shift)

	RS	R/W	DB7							DB0
รหัส	0	0	0	0	0	1	S/C	R/L	*	*

* ไม่มีผล

เป็นคำสั่งที่กำหนดให้ตำแหน่งของเคอร์เซอร์หรือข้อมูลไปแสดงทางซ้ายหรือขวาโดย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ไม่ต้องใช้คำสั่งเขียนหรืออ่าน โดยที่มีชื่อรหัสการทำงานดังต่อไปนี้

บิท S/C R/L

0 0 ทำการย้ายเคอร์เซอร์ไปจากตำแหน่งเดิมทางซ้าย 1 ตำแหน่ง

0 1 ทำการย้ายเคอร์เซอร์ไปจากตำแหน่งเดิมทางขวา 1 ตำแหน่ง

1 0 เป็นการเลื่อนตัวอักษรที่จะไปแสดงทางซ้าย 1 ตำแหน่ง

1 1 เป็นการเลื่อนตัวอักษรที่จะแสดงไปทางขวา 1 ตำแหน่ง

2.3.6 <u>คำสั่งการเซตฟังก์ชั่น</u> (Function Set)

* ไม่มีผล

บิท DL : เป็นการเซตการติดต่อว่าจะเป็นแบบ 8 บิทหรือ 4 บิท โดยถ้าต้องการติดต่อแบบ 4 บิท ค่าในบิท DL จะเท่ากับ 0 แต่ถ้าต้องการที่จะติดต่อแบบ 8บิท ค่าในบิท ค่าในบิท DL จะเท่ากับ 1

N : เป็นการเซตบรรทัดในการแสดงผล โดยที่ถ้า N = 0 จะเป็นการแสดงผลแบบ 1 บรรทัด N = 1 จะเป็นการแสดงผล 2 บรรทัด ในกรณีที่จอแสดงผลLCD มีมากกว่า 2 บรรทัด ก็ให้เซตค่าในบิทนี้ = 1

F : เป็นการเซตขนาดของจุดแสดงผลว่าจะให้เป็นแบบ 5*7 หรือ 5*10 โดยที่ถ้า F = 0 จะเป็นแบบ 5*7 และถ้า F =1 จะเป็นแบบ 5*10

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.3.7 <u>คำสั่งที่ใช้กำหนดค่าแอดเดรสของตัวสร้างอักขระ</u>

(Set CG RAM Address)

RS R/W DB7 DB0

รหัส	0	0	0	1	A	A	A	A	A	A

เป็นการเซตค่าแอดเดรสของ CG RAM โดยที่จะต้องทำการเซตค่าแอดเดรสก่อนที่จะมีการเขียนหรืออ่านข้อมูลจาก CG RAM ด้วย

2.3.8 <u>คำสั่งที่ใช้กำหนดค่าแอดเดรสบนหน้าจอ</u>

LCD (Set DD RAM Address)

RS R/W DB7 DB0

รหัส	0	0	1	A	A	A	A	A	A	A

เป็นคำสั่งที่ใช้ในการเซตค่าแอดเดรสใน DD RAM ในการเขียนหรืออ่านค่าจาก DD RAM โดยที่จำนวนแอดเดรสที่เกิดขึ้นบนจอจะอยู่กับการเซตค่า N โดย

ถ้า N =1 (1บรรทัด) ค่าแอดเดรสจะอยู่ในช่วง 00H-4FH

ถ้า N =0 (2บรรทัด) ค่าแอดเดรสจะอยู่ในช่วง 00H-27H สำหรับบรรทัดที่ 1 และจะอยู่ในช่วง 40H-67H สำหรับบรรทัดที่ 2

2.3.9 <u>คำสั่งอ่านค่าสถานะและแอดเดรส</u> (Read Busy Flag & Address)

RS R/W DB7 DB0

รหัส	0	1	BF	A	A	A	A	A	A	A

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เป็นคำสั่งที่ใช้อ่านค่าสถานะการทำงานว่า ขณะนี้จอแสดงผล LCD พร้อมที่จะรับข้อมูลหรือไม่โดยที่ถ้า BF =1 แสดงว่าจอแสดงผลยังไม่พร้อมที่จะรับข้อมูลหรือคำสั่ง ถ้า BF =0 แสดงว่าจอแสดงผลพร้อมที่จะรับข้อมูลหรือคำสั่ง

2.3.10 <u>คำสั่งเขียนข้อมูลลงใน CG หรือ DD RAM</u>

(Write Data To CG or DD RAM)

	RS	R/W	DB7							DB0
รหัส	1	0	D	D	D	D	D	D	D	D

เป็นคำสั่งที่ใช้ในการเขียนข้อมูลเข้าไปใน CG RAM หรือ DD RAMโดยที่ก่อนที่จะทำการอ่านข้อมูลนั้น จะต้องมีการกำหนดค่าของแอดเดรสเริ่มต้นเสียก่อน

2.3.11 <u>คำสั่งอ่านข้อมูลจาก CG หรือ DD RAM</u>

(Read Data From CG or DD RAM)

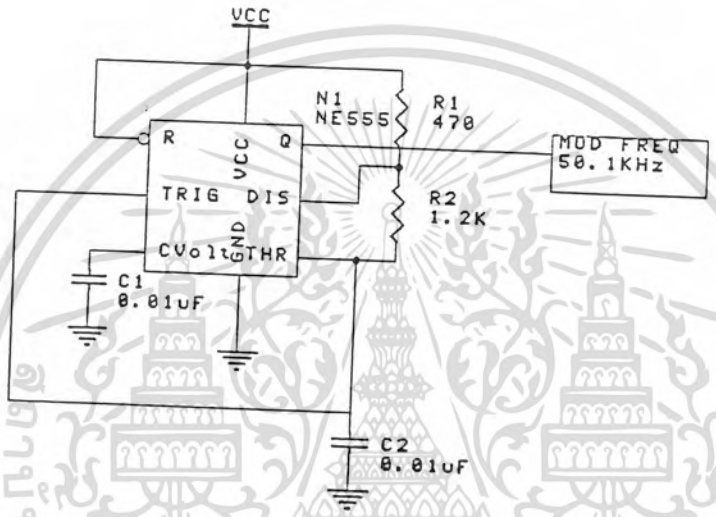
	RS	R/W	DB7							DB0
รหัส	1	1	D	D	D	D	D	D	D	D

เป็นคำสั่งที่ใช้ในการอ่านข้อมูลจาก CG RAM หรือ DD RAMโดยที่ก่อนที่จะทำการอ่านข้อมูลนั้น จะต้องมีการกำหนดค่าของแอดเดรสเริ่มต้นเสียก่อน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.4 โครงสร้างส่วนสร้างสัญญาณพาหะ

ใช้ไอซี NE555 เป็นตัวกำเนิดสัญญาณพาหะดังรูป



รูป 2.25 วงจรกำเนิดสัญญาณพาหะ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 3

การออกแบบและการสร้าง

ในการออกแบบและการสร้าง แบ่งออกเป็น 3 ส่วนใหญ่ๆดังต่อไปนี้

3.1 หน่วยประมวลผลกลางและหน่วยความจำ

3.2 ส่วนรับส่งข้อมูลจากเครื่องควบคุมระยะไกลภายนอก

3.3 ส่วนแสดงผล

3.4 ส่วนของนาฬิกาแสดงเวลาจริง(Real time clock)

3.5 ส่วนที่ใช้ในการ BACK UP หน่วยความจำ

3.6 คีย์บอร์ด

3.1 ส่วนประมวลผลกลางและหน่วยประมวลผล

ในส่วนนี้ได้เลือกใช้ไมโครคอนโทรลเลอร์เบอร์ 80C31 เป็นตัวประมวลผลกลาง โดยที่จะประกอบด้วยหน่วยความจำแบบรอมขนาด 8กิโลไบต์แบบแรมขนาด 32 กิโลไบต์ โดยที่จะมีการจัดแบ่งแอดเดรสของหน่วยความจำแต่ละแบบในการใช้งานดังต่อไปนี้

3.1.1 หน่วยความจำแบบรอม เราใช้ไอซีเบอร์ 27C64 เป็นหน่วยความจำแบบซีมอส ขนาด 8 กิโลไบต์ โดยที่หน่วยความจำส่วนนี้จะใช้ในการบันทึกมอนิเตอร์โปรแกรม ซึ่งจะอยู่ในช่วงแอดเดรส 0000-1FFFH

3.1.2 หน่วยความจำแบบแรม เราใช้ไอซีเบอร์ 62256 ซึ่งมีขนาด 32 กิโลไบต์ โดยหน่วยความจำส่วนนี้จะใช้ในการบันทึกสัญญาณข้อมูลที่รับมาจากเครื่องควบคุมระยะไกล

3.2 ส่วนรับส่งข้อมูลจากเครื่องควบคุมระยะไกลภายนอก

ในการออกแบบส่วนนี้ เราจะแบ่งการออกแบบเป็น 3 ส่วนดังต่อไปนี้

3.2.1 การออกแบบขั้นตอนการรับส่งสัญญาณ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.2.2 การออกแบบภาครับส่งสัญญาณอินฟราเรด

3.3.3 การออกแบบและคำนวณวงจรสร้างคลื่นพาหะ

3.2.1 การออกแบบขั้นตอนการรับส่งสัญญาณ

การใช้งานเครื่องควบคุมระยะไกลเอนกประสงค์แบบโปรแกรมได้เครื่องนี้นั้นขั้นแรกเราจะต้องบันทึกสัญญาณควบคุมแบบอินฟราเรดของเครื่องควบคุมระยะไกลที่เราใช้งานอยู่ทั่วๆไปเสียก่อนเพื่อที่จะเก็บรูปแบบของสัญญาณเพื่อควบคุมอุปกรณ์เครื่องใช้ตามที่ต้องการได้โดยเราจะแบ่งการทำงานออกเป็น 3 ส่วน

3.2.1.1 การรับสัญญาณ

3.2.1.2 การส่งสัญญาณ

3.2.1.1 การรับสัญญาณ

เมื่อเครื่องควบคุมระยะไกลเอนกประสงค์แบบโปรแกรมได้เครื่องนี้ถูกเลือกให้ทำการอ่านสัญญาณของเครื่องควบคุมระยะไกลเครื่องอื่นนั้น เครื่องควบคุมระยะไกลเอนกประสงค์แบบโปรแกรมได้นี้จะทำการเก็บรูปแบบของสัญญาณไว้ในหน่วยความจำตำแหน่งที่ได้กำหนดไว้ในหน่วยความจำแบบแรมในการบันทึกสัญญาณ 1 สัญญาณใช้หน่วยความจำ 256 ไบต์แอดเดรสที่เริ่มใช้ในการบันทึกสัญญาณจะเริ่มจากแอดเดรส RAM ที่ 0B6AH ข้อมูลที่บันทึก จะมีลักษณะเป็นความกว้างของสัญญาณ 0 และสลับกันไปสำหรับการนับช่วงความกว้างของสัญญาณจะอาศัยการทำงานของ COUNTER 0 และ COUNTER 1 ซึ่งจะอยู่ในตัว CPU 80C31โดยจะสลับกันทำงานในช่วงที่สัญญาณมีค่าเป็น 1 และในช่วงสัญญาณที่มีค่าเป็น 0 หลักการทำงานก็คือเมื่อเราส่งสัญญาณควบคุมจากเครื่องควบคุมระยะไกลเครื่องอื่น เข้ามาที่เครื่องควบคุมระยะไกลเอนกประสงค์แบบโปรแกรมได้สัญญาณก็จะเข้ามาที่ขา INT0 และ INT1 ของ 80C31 เมื่อสัญญาณที่เข้ามาที่ขา INT0เปลี่ยนจาก 1 เป็น 0 CPU จะกระโดดไปทำงานที่ตำแหน่ง 0003H ซึ่งเป็นตำแหน่งINTERRUPT ROUTINE ของ INT0 ในโปรแกรมส่วนนี้จะกำหนดให้ COUNTER 0 เริ่ม ทำการนับความกว้างของ pulse ที่เข้ามาที่ขา INT0 และให้ COUNTER 1 หยุดนับแล้วนำค่าที่ COUNTER 1 นับได้ไปเก็บไว้ที่หน่วยความจำ เมื่อสัญญาณที่เข้ามาที่ขาINT1เปลี่ยนจาก 1เป็น 0 CPU จะกระโดด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ไปทำงานที่ตำแหน่ง 0013H ซึ่งเป็นตำแหน่ง INTERRUPT ROUTINE ของ INT1 ในโปรแกรมส่วนนี้จะกำหนดให้ COUNTER 1 เริ่มทำการนับความกว้างของ พัลส์ที่เข้ามาที่ขา INT0 และให้ COUNTER 0 หยุดนับแล้วนำค่าที่ COUNTER 0 นับได้ไปเก็บไว้ที่หน่วยความจำ COUNTER 0 และ COUNTER 1 จะทำงานสลับกันไปเรื่อยๆจนกว่าจะได้จำนวน pulse ที่ต้องการ

3.2.1.2 การส่งสัญญาณ

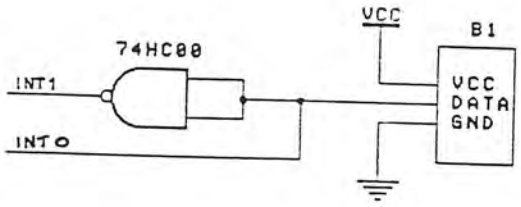
เมื่อเครื่องควบคุมระยะไกลเอนกประสงค์แบบโปรแกรมได้ถูกเลือกให้ทำหน้าที่ในการส่งสัญญาณหน่วยประมวลผลจะทำการนำเอาข้อมูลของสัญญาณแต่ละอุปกรณ์ควบคุมซึ่งถูกเก็บไว้ในหน่วยความจำภายนอกเข้ามาทำการเก็บในรีจิสเตอร์ของไทม์เมอร์0 หรือไทม์เมอร์ 1 แล้วทำการนับในขณะเดียวกันก็จะทำการส่งสัญญาณที่มีค่าเป็น1หรือ 0 ซึ่งจะสัมพันธ์กับที่นำเข้ามาเก็บไว้ในรีจิสเตอร์ของไทม์เมอร์แต่ละตัวโดยที่สัญญาณนี้จะถูกส่งออกมาที่ขา P3.0 ของ CPU เพื่อที่จะนำไปทำการมอดดูเลทกับสัญญาณคลื่นพาหะก่อนที่จะทำการแปลงเป็นสัญญาณอินฟราเรดเพื่อใช้ในการส่งเป็นสัญญาณควบคุมอุปกรณ์ต่างๆการทำงานของไทม์เมอร์ทั้ง 2 ตัว เราสามารถจะโปรแกรมให้ทำงานในโหมดการทำงานโหมด 1 และจะสลับกันทำงานเช่นเดียวกับตอนรับสัญญาณแต่การส่งสัญญาณจะอาศัยคุณสมบัติการทำงานของอินเทอร์รัปไทม์เมอร์ เมื่อเกิดการ OVER FLOW ในการเริ่มต้นและหยุดการนับของไทม์เมอร์พร้อมกับการเปลี่ยนสถานะของสัญญาณที่ขา P3.0 ไปด้วย

3.2.2 การออกแบบภาครับส่งสัญญาณอินฟราเรด

ภาครับ

ภาครับสัญญาณอินฟราเรดใช้ตัวรับของบริษัท SONYเบอร์ SBX 1610-52 แล้วส่งข้อมูลเข้าที่ขา INT0 และ INT1ของ CPU สัญญาณที่เข้ามาที่ขา INT1 เป็นอินเวอร์ทของสัญญาณที่เข้ามาขา INT0 ดังรูปที่ 3.1

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

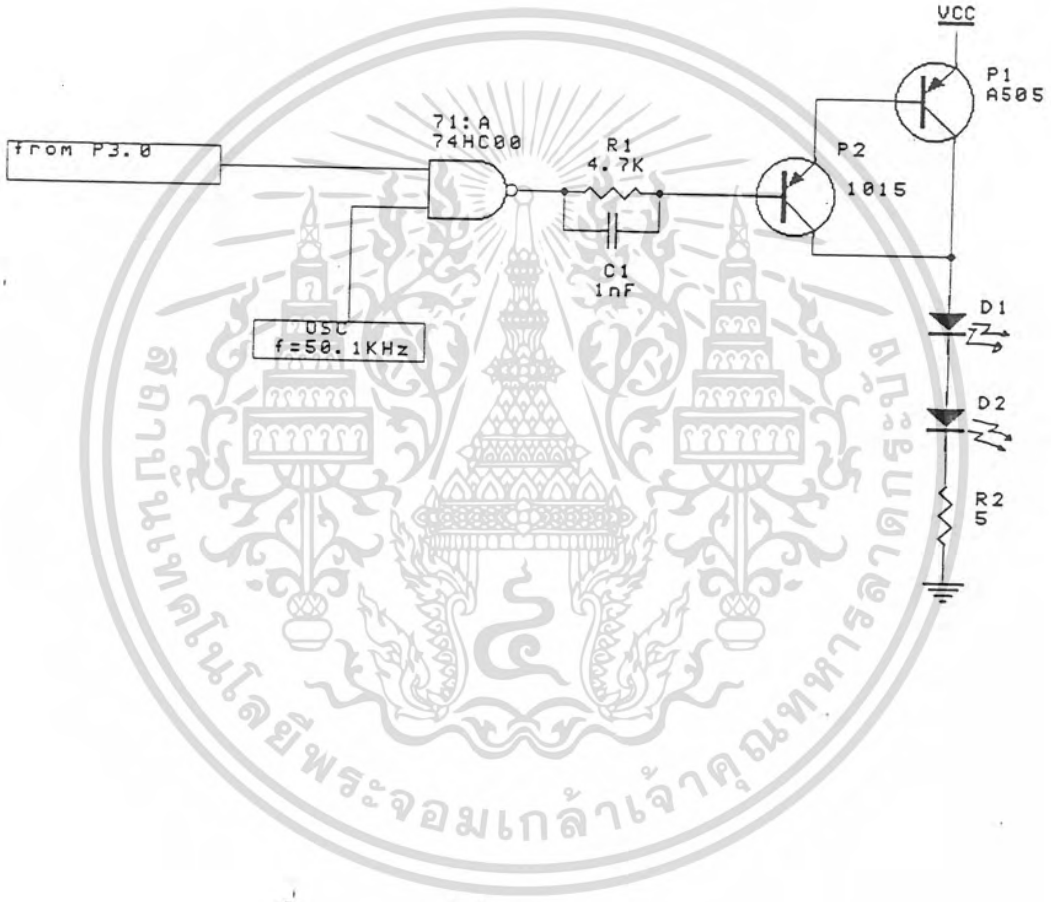


รูปที่ 3.1 ภาครับ

ภาคส่งสัญญาณ

ในส่วนของวงจรภาคส่งสัญญาณนี้จะเป็นส่วนของวงจรที่ใช้ในการส่งข้อมูลที่เก็บไว้ในหน่วยความจำของเครื่องควบระยะไกลแบบโปรแกรมได้นี้ ออกไปใช้ในการควบคุมอุปกรณ์ต่างๆโดยที่จะประกอบด้วยตัวส่งสัญญาณอินฟราเรด ที่ทำหน้าที่ในการแปลงข้อมูลที่เก็บไว้ในหน่วยความจำให้เป็นสัญญาณอินฟราเรด ก่อนที่จะทำการเปลี่ยนข้อมูลที่เก็บไว้ในหน่วยความจำให้เป็นสัญญาณอินฟราเรด จะต้องมีการนำเอาข้อมูลที่เก็บไว้ในหน่วยความจำมาทำการมอดดูเลทกับสัญญาณพาหะเสียก่อน โดยที่เราจะทำการมอดดูเลทสัญญาณควบคุมผ่านทาง NAND GATE หลังจากนั้นสัญญาณเอาท์พุทหลังจากการมอดดูเลทที่ได้จาก แนนเกท ก็จะถูกส่งไปยังวงจรขยายกระแสซึ่งจะประกอบทรานซิสเตอร์ PNP 2 ตัวต่อกันแบบ DALINGTON ต่อจากนั้นก็จะใช้สัญญาณที่ได้จากวงจรขยายนี้ไปขับตัวส่งอินฟราเรดอีกทีหนึ่งดังแสดงในรูป 3.2

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

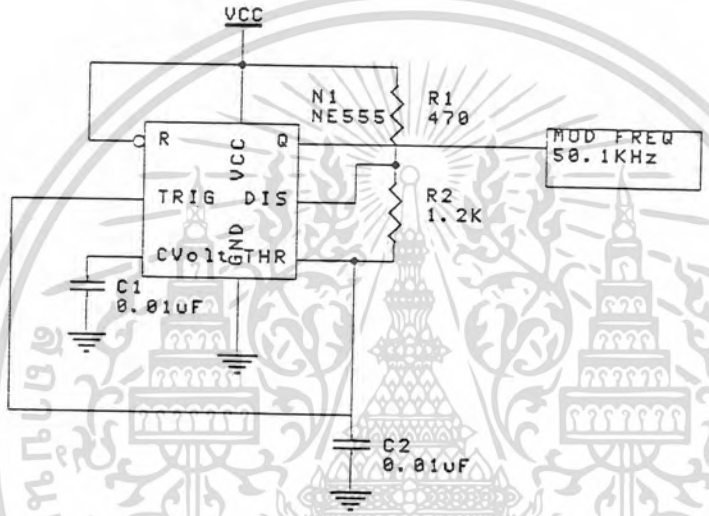


รูปที่ 3.2 ภาคส่งสัญญาณ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การออกแบบและคำนวณวงจรกำเนิดสัญญาณพาหะ

ในโครงงานนี้จะใช้ความถี่พาหะในการส่งข้อมูลเท่ากับ 50.1KHz โดยที่วงจรสร้างความถี่สร้างขึ้นจากวงจร Astable Multivibrator ของไอซีเบอร์ NE555 ดังในรูปที่ 3.3



รูปที่ 3.3 วงจรกำเนิดสัญญาณพาหะ

ความถี่เอาท์พุทของวงจร Astable Multivibrator หาได้จากสูตร

$$f=(1.43)/((R1+2R2)*C)$$

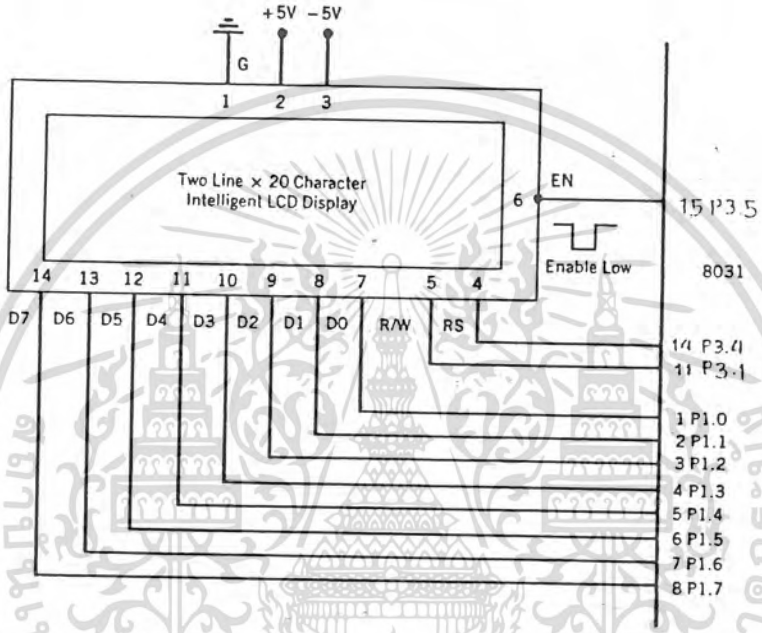
$$=(1.43)/((470+(2*1.2*10^{3}))*0.01X10^{-6})$$

$$=50.1KHz$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.3 ส่วนแสดงผล

สำหรับส่วนแสดงผลนี้เราเลือกใช้จอ LCD เป็นหน่วยแสดงผลโดยที่จะมีการรับส่งข้อมูล ระหว่างหน่วยประมวลผลกับจอ LCD ผ่านทางพอร์ท 1 ให้เป็นพอร์ทข้อมูล และพอร์ท 3 ใช้เป็นพอร์ทควบคุม (P3.1,P3.4,P3.5) LCD ดังในรูปที่ 3.4



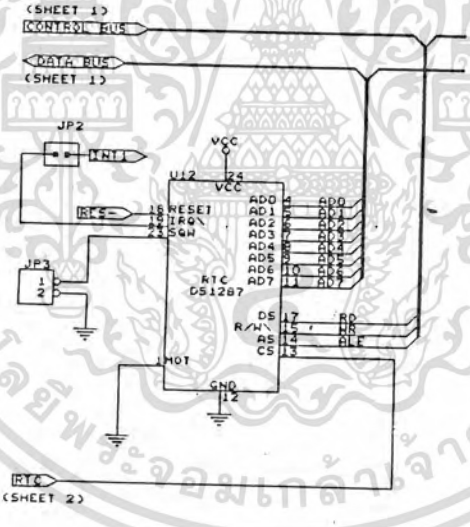
รูปที่ 3.4 ส่วนแสดงผล

3.4 ส่วน REAL TIME CLOCK

ใช้ในการแสดงเวลาจริงและใช้ในการตั้งเวลาส่งสัญญาณจากเครื่องควบคุมระยะไกลเอนกประสงค์แบบโปรแกรมได้ไปควบคุมอุปกรณ์ต่างๆในส่วน REAL TIME CLOCK เราใช้ไอซี เบอร์ DS1287 ในการตั้งเวลาหน่วยประมวลผลจะจะรับค่าเวลาที่ตั้งไว้เข้ามาทางคีย์บอร์ดเมื่อเข้าสู่ MODE การตั้งเวลา หน่วยประมวลผลจะเปรียบเทียบค่าที่อยู่หน่วยความ จำกับค่าเวลาใน rel time clockว่าตรงกันหรือไม่ถ้าไม่ตรงกันหน่วยประมวลผลจะนำค่าในหน่วยประผลต่อไปเรื่อย ๆ จนกว่าค่า

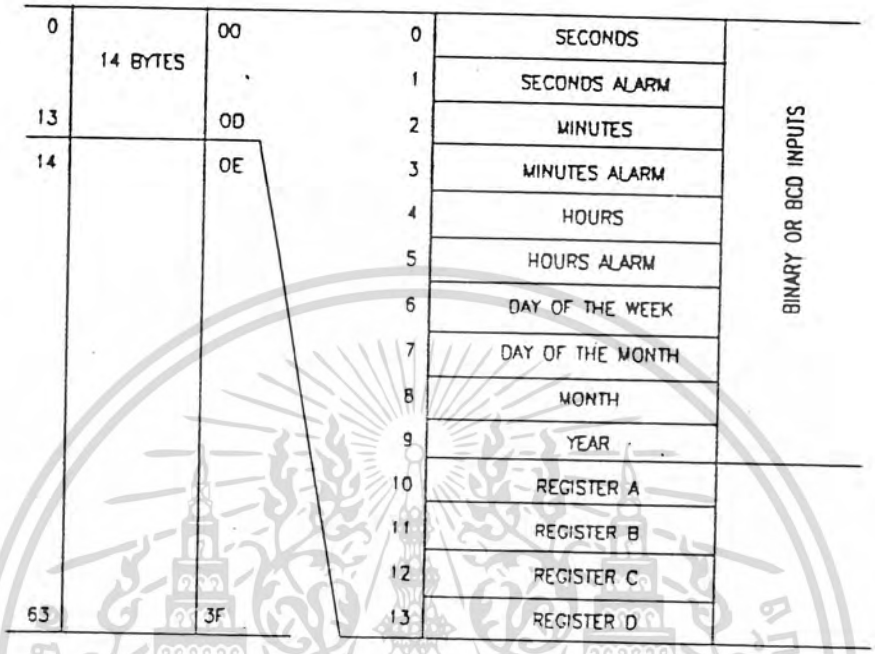
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เวลาจะตรงกันหน่วยประมวลผลก็จะนำข้อมูลของสัญญาณควบคุมส่งออกไปควบคุมอุปกรณ์ที่ได้ตั้งเวลาไว้ จากนั้นหน่วยความจำก็จะทำการเปรียบเทียบค่าในหน่วยความจำและ real time clock ต่อไปจนกว่าจะกดคีย์ 0 โปรแกรมจะออกจากการทำงานใน MODE นี้ ในโครงงานกำหนดให้ตำแหน่งของหน่วยความจำของreal time clock ตำแหน่งแรกอยู่ที่ address 8000H ของวงจรดังนั้นถ้าเราต้องการอ่านค่า วินาทีจาก real time clock จะสามารถอ่านได้ที่ address 8000H รูปข้างล่างจะแสดงการเชื่อมต่อวงจร real time clock



รูปที่ 3.5 วงจรเชื่อมต่อของ real time clock

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

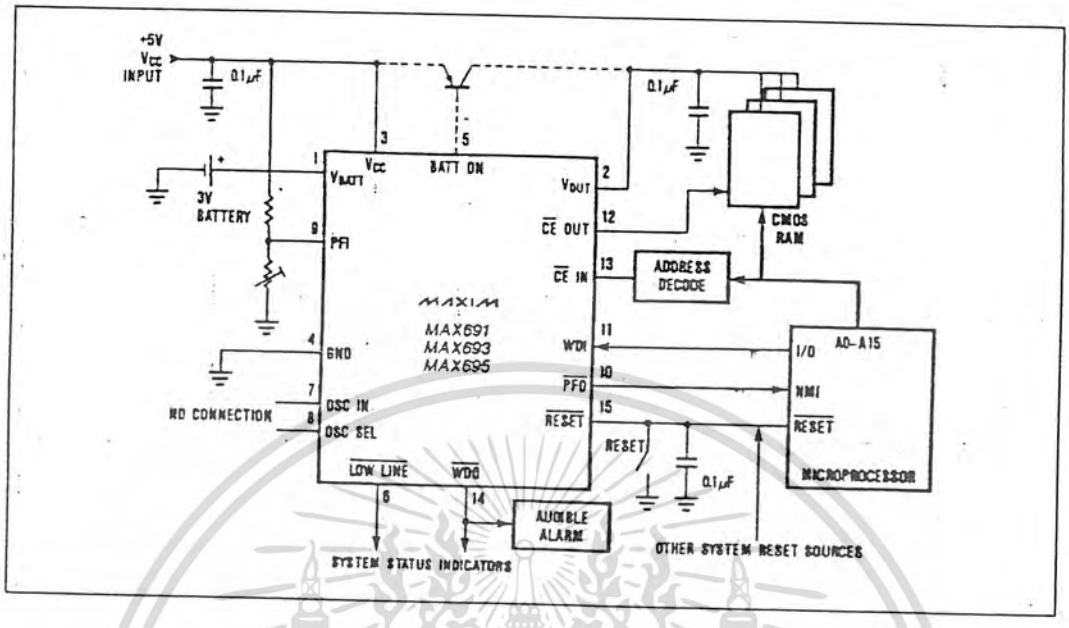


รูปที่ 3.6 ADDRESS MAP ของ real time clock

3.5 ส่วนที่ใช้ในการ BACK UP หน่วยความจำ

ในโครงงานนี้ใช้ ไอซีเบอร์ MAX 691 ดังแสดงในรูป 3.7

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



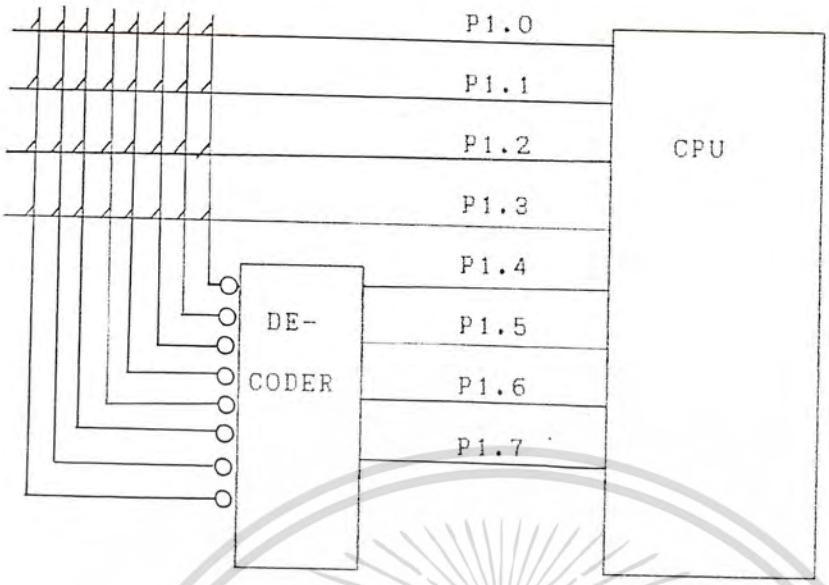
รูปที่ 3.7 วงจรที่ใช้ในการ BACK UP หน่วยความจำ

หน้าที่ของ MAX 691 จะป้องกันไม่ให้ข้อมูลสูญเสียไปขณะที่ถ่านหมดหรือปิดเครื่องในขณะที่ปิดเครื่องสัญญาณที่ /CE OUT จะเป็น HIGH ทำให้หน่วยความจำไม่ถูกเขียนข้อมูลเข้าไปในช่วงนี้ ในขณะที่ ปิดเครื่อง ไฟเลี้ยงจากแบตเตอรี่ 3 โวลท์ ถูก switch ให้ไปเลี้ยงหน่วยความจำ นอกจากนี้แล้ว MAX 691 ยังเป็นตัวกำเนิดสัญญาณ reset+ ให้กับ หน่วยประมวลผลและสัญญาณ reset- ให้กับ real time clock

3.6 คีย์บอร์ด

ในโครงงานนี้เราใช้คีย์บอร์ด 32คีย์เพื่อความสะดวกในการใช้งานยิ่งขึ้นโดยลักษณะของคีย์บอร์ดจะมีลักษณะเป็น matrix คีย์การเชื่อมต่อของคีบอร์ดและหน่วยประมวลผลแสดงไว้ดังรูปที่ 3.8

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.8 วงจรเชื่อมต่อคีย์บอร์ด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 4

การทดลองและผลการทดลอง

1. เมื่อใช้ความถี่พาหะ 16 KHz ในการมอดดูเลทกับข้อมูลที่ส่งออกไปจากเครื่องควบคุมเอนกประสงค์แบบโปรแกรมได้ไปควบคุมอุปกรณ์ต่างๆ เกิดผลดังนี้

ชื่ออุปกรณ์	ระยะทางที่รับได้(เมตร)
1. เครื่องเล่น CD SONY	7-8
2. TV สี SONY	2
3. เครื่องเล่นเทป,CD PANASONIC	1-2
4. TV สี JVC และ TANIN	3

2. เมื่อใช้ความถี่พาหะ 50.1 KHz ในการมอดดูเลทกับข้อมูลที่ส่งออกไปจากเครื่องควบคุมเอนกประสงค์แบบโปรแกรมได้ไปควบคุมอุปกรณ์ต่างๆ เกิดผลดังนี้

ชื่ออุปกรณ์	ระยะทางที่รับได้(เมตร)
1. เครื่องเล่น CD SONY	8-10
2. TV สี SONY	4-5
3. เครื่องเล่นเทป,CD PANASONIC	1-2
4. TV สี JVC และ TANIN	4-5

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 5

สรุปและวิจารณ์

โดยสรุปแล้วผลการทดลองอยู่ในระดับที่น่าพอใจคือสามารถควบคุมและตั้งเวลาในการส่งสัญญาณไปควบคุมเครื่องใช้ไฟฟ้าต่างๆในการทดลองได้ทุกชนิดโดยมีการแสดงสภาวะการทำงานของระบบด้วยจอแสดงผลชนิด LCD และมี คีย์บอร์ด 32 คีย์ซึ่งจะทำให้ผู้ใช้งานสามารถใช้งานเครื่องควบคุมระยะไกลเอนกประสงค์แบบโปรแกรมได้เครื่องนี้ง่ายขึ้น

ปัญหาที่พบ

1.ระยะทางในการส่งสัญญาณไปควบคุมเครื่องใช้ไฟฟ้าบางเครื่อง บางรุ่นระยะทางยังไม่ไกลพอ

2.วงจรกินกระแสไฟมากซึ่งวงจรทั้งหมดกินกระแสไฟประมาณ 120 mA ทำให้ใช้งานไม่ได้นาน

แนวทางแก้ไข

1.ในการรับสัญญาณที่เข้ามาเครื่องควบคุมระยะไกลเอนกประสงค์แบบโปรแกรมได้ควรเก็บค่าความค่าความถี่พาหะของสัญญาณที่รับเข้ามาด้วยเวลาส่งสัญญาณจากเครื่องควบคุมระยะไกลเอนกประสงค์แบบโปรแกรมได้ไปควบคุมเครื่องใช้ไฟฟ้าเครื่องอื่นก็ควรมอดดูเลทกับสัญญาณพาหะเดิมของสัญญาณที่รับเข้ามาซึ่งจะทำให้ระยะทางในการควบคุมไกลขึ้นซึ่งจะใช้หลักการทำงานของ timer มาแก้ปัญหานี้แทนการมอดดูเลทโดยใช้ไอซีเบอร์ NE555

2.ควรใช้ CPU เบอร์ 87C51 ซึ่งมี ROM ภายในตัว 4 กิโลไบต์กินไฟแค่ 11 mA จะช่วยแก้ปัญหาเรื่องแหล่งจ่ายไฟได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บรรณานุกรม

1. วารสารเซมิฯอิเล็กทรอนิกส์. หลักการเขียนโปรแกรมมอนิเตอร์:ซีเอ็ดยูเคชั่น, 2536.อาณัติ มุลิกฤกษ์ และ ชัยวัฒน์ ลิ้มพรจิตรวิไล

2. ET-8032 V2.0 MCS-51 SINGLE BOARD MICROCONTROLLER USER'S - MANUAL ETT CO.,LTD;2536

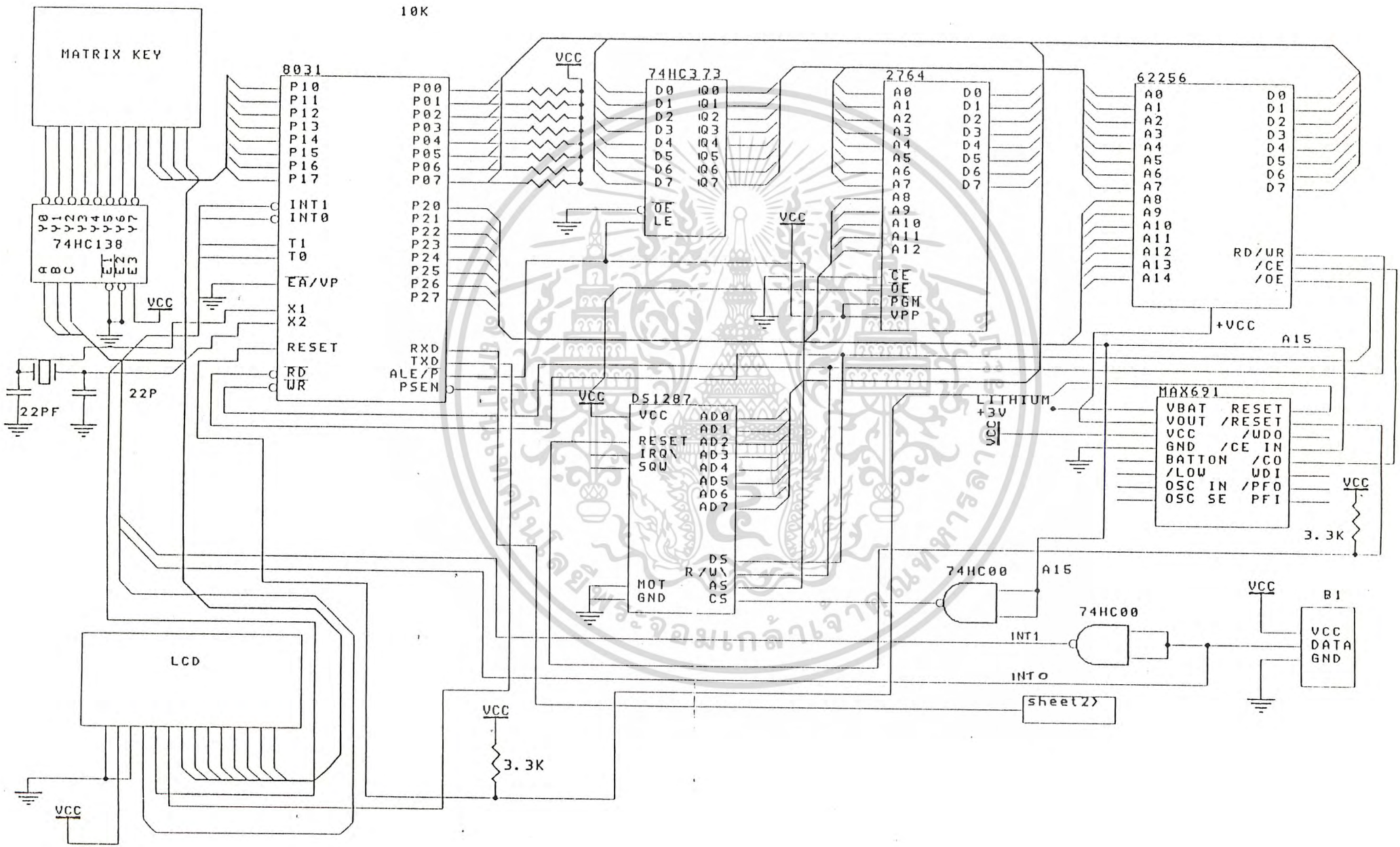
3. วิทยานิพนธ์ปีการศึกษา 2533,2534. เครื่องควบคุมระยะไกลเอนกประสงค์แบบโปรแกรมได้

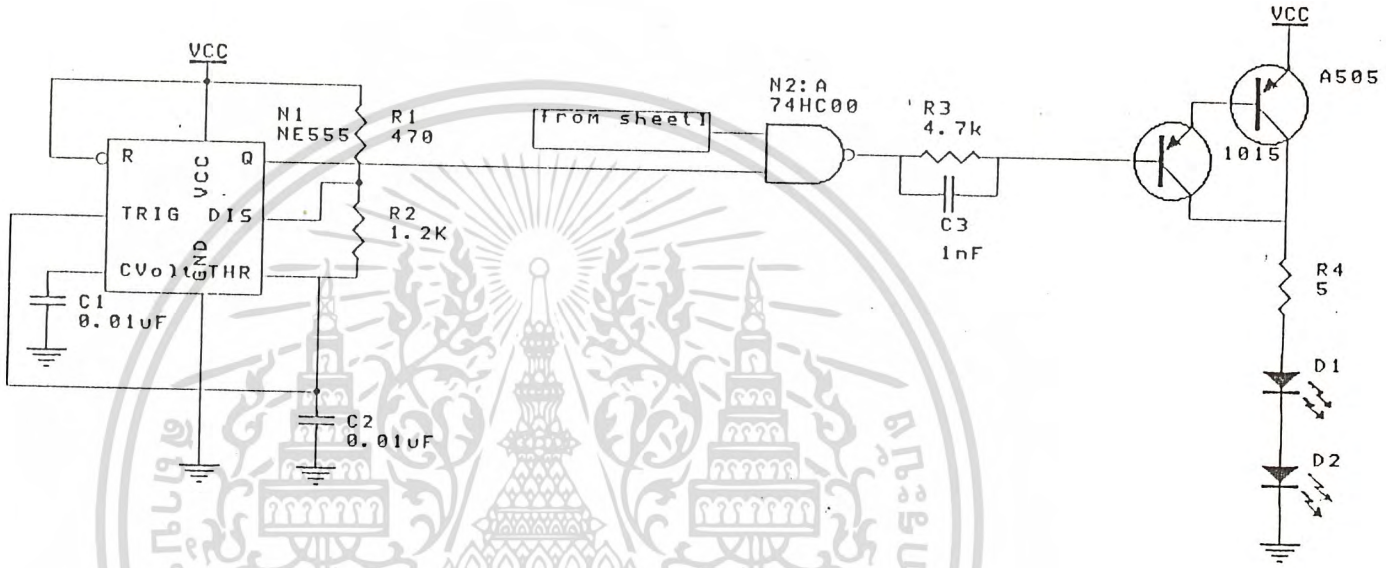
4. กฤษดา วิศวธิรานนท์ . เรียน/เล่น /ใช้ ไอซีดิจิตอล:ซีเอ็ดยูเคชั่น,2521

5. ETT MCS-51 MICROCONTROLLERS. ETT CO.,LTD

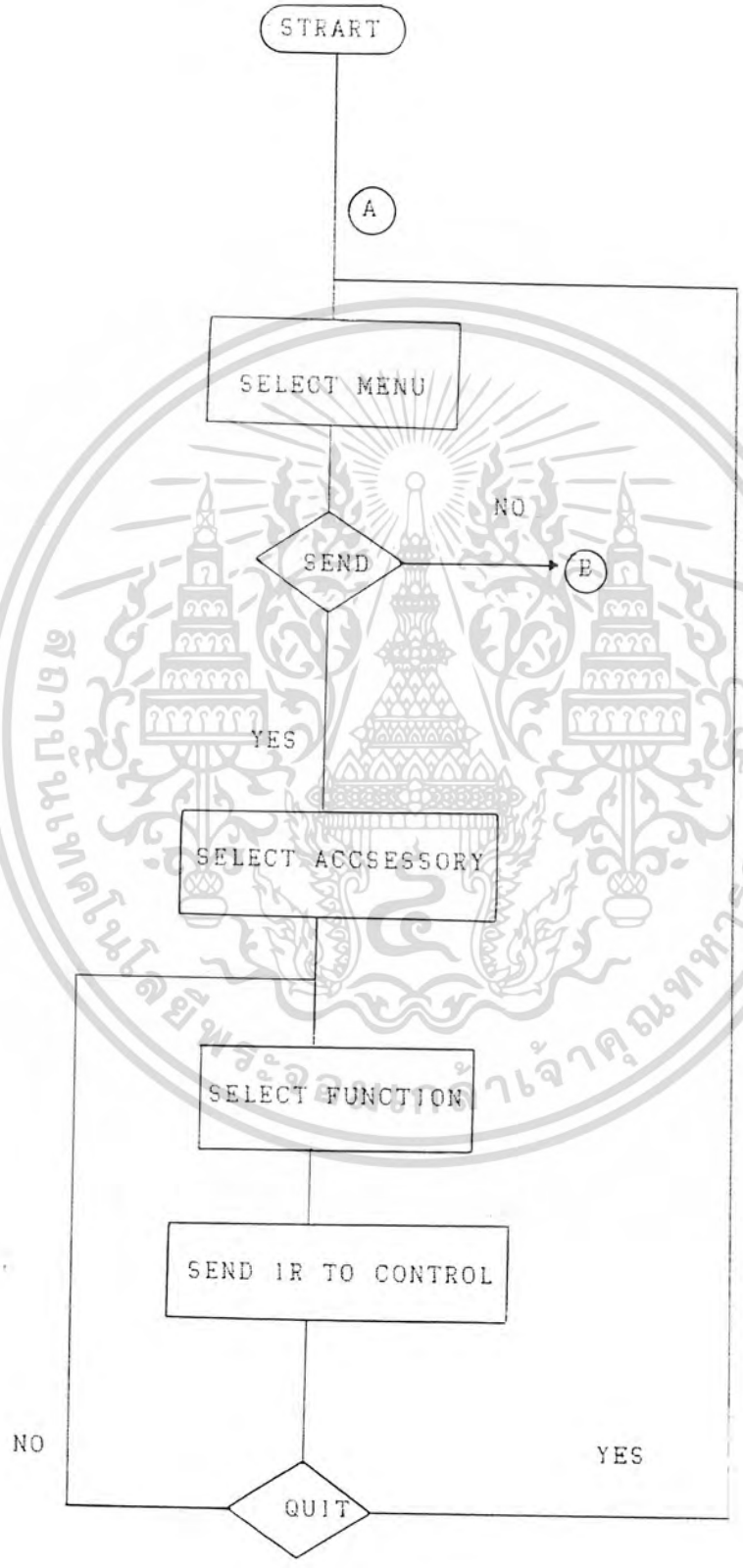
6. KENNETH J. AYALA. THE 8051 MICROCONTROLLER ARCHITECTURE, PROGRAMMING AND APPLICATION :1991

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

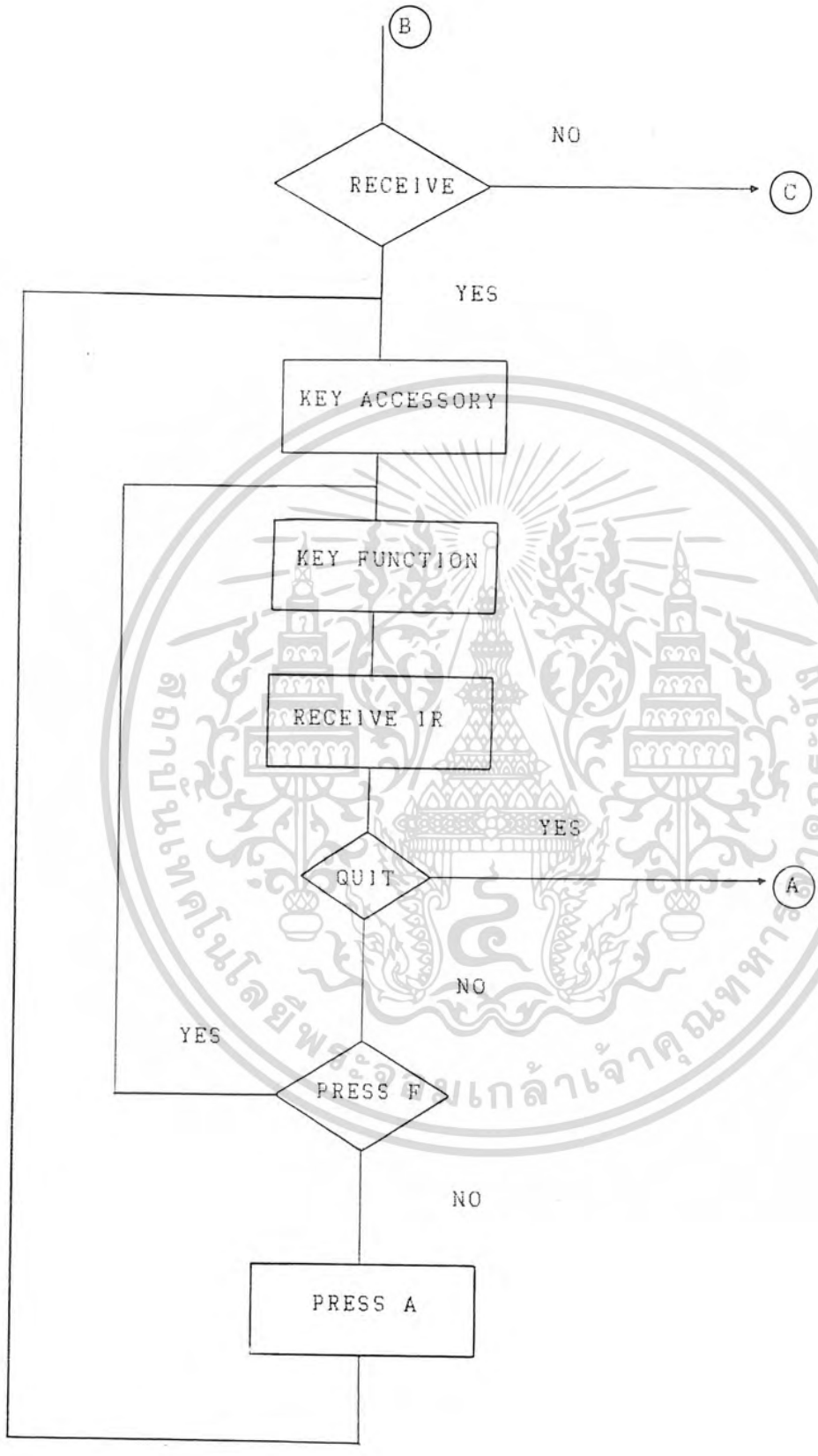

10K
MATRIX KEY
8031
P10
P11
P12
P13
P14
P15
P16
P17
INT1
INT0
T1
T0
EA/VP
X1
X2
RESET
RD
WR
P00
P01
P02
P03
P04
P05
P06
P07
P20
P21
P22
P23
P24
P25
P26
P27
RXD
TXD
ALE/P
PSEN
74HC373
D0
D1
D2
D3
D4
D5
D6
D7
OE
LE
IQ0
IQ1
IQ2
IQ3
IQ4
IQ5
IQ6
IQ7
2764
A0
A1
A2
A3
A4
A5
A6
A7
A8
A9
A10
A11
A12
CE
OE
PGM
VPP
62256
RD/WR
/CE
/OE
+VCC
A15
74HC138
VCC
22PF
22P
DS1287
VCC
RESET
IRQ\
SQW
MOT
GND
AD0
AD1
AD2
AD3
AD4
AD5
AD6
AD7
DS
R/W\
AS
CS
LITHIUM
+3V
MAX691
VBAT
VOUT
VCC
GND
BATTON
/LOW
OSC IN
OSC SE
RESET
/RESET
/WDO
/CE IN
/CO
WDI
/PFO
PFI
3.3K
74HC00
A15
INT1
INT0
B1
VCC
DATA
GND
LCD
3.3K
sheet2


VCC
N1
NE555
R
Q
TRIG
DIS
CVoltage
THR
VCC
GND
R1
470
R2
1.2K
C1
0.01uF
C2
0.01uF
from sheet1
N2:A
74HC00
R3
4.7k
C3
1nF
VCC
A505
1015
R4
5
D1
D2

FLOWCHART แสดงการทำงานของเครื่องควบคุมระยะไกลแบบโปรแกรมได้



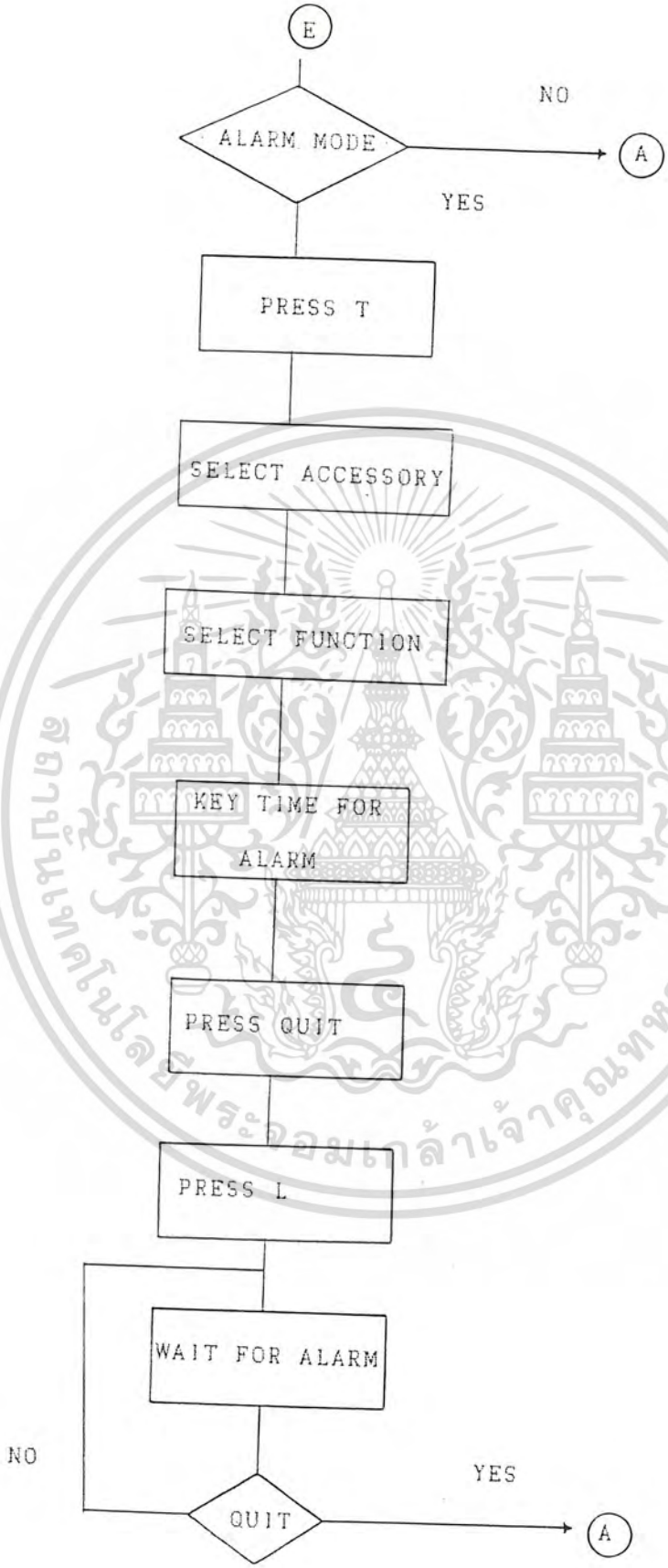
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การใช้งานเครื่องควบคุมระยะไกลเอนกประสงค์แบบโปรแกรมได้

1. เมื่อเริ่มเปิดเครื่อง เครื่องจะถามว่าต้องการ CLEAR หน่วยความจำหรือไม่ ถ้าไม่ ต้องการกด N ถ้าต้องการกด Y

2. เมื่อกด Y หรือ N เรียบร้อยแล้ว เครื่องจะพิมพ์คำว่า SELECT MENU จากนั้นเราก็เลือก MENU ถ้ากด

- S จะเป็นการส่งข้อมูล
- R จะเป็นการรับข้อมูล
- T จะเป็นการตั้งเวลาในการส่งสัญญาณไปควบคุมอุปกรณ์
- U จะเป็นการแก้ไขค่าเวลาใน Realtime clock
- C จะเป็นการแก้ไขข้อมูลของสัญญาณควบคุมที่รับเข้ามา

3. เมื่อกด S จะเป็นการส่งสัญญาณควบคุม เครื่องจะพิมพ์คำว่า SELECT ACCESSORY จากนั้นให้กด ENTER เครื่องจะพิมพ์ชื่ออุปกรณ์ที่เราได้บันทึกสัญญาณไว้ จากนั้นกด เพื่อเลือกอุปกรณ์ที่เราต้องการควบคุม เมื่อเลื่อนลงมาถึงชื่ออุปกรณ์ที่เราต้องการส่งสัญญาณไปควบคุมแล้ว ให้กด ENTER จากนั้นเครื่องจะพิมพ์คำว่า SELECT FUNCTION ให้กด ENTER 1 ครั้ง เครื่องก็จะพิมพ์คำว่า SELECT-FUNCTION เช่นเดิม จากนั้นเลือกกดปุ่ม A – Z ปุ่มใดปุ่มหนึ่ง จากนั้นเครื่องก็จะพิมพ์ชื่อฟังก์ชั่นประจำปุ่มนั้น ถ้าต้องการส่งสัญญาณควบคุมตามชื่อฟังก์ชั่นที่ แสดงที่ LCD ก็กดซ้ำปุ่มเดิมอีกครั้ง ถ้าไม่ต้องการที่จะส่งฟังก์ชั่นนั้นก็กดปุ่มอื่นหลังจากนั้น ถ้าต้องการส่งสัญญาณฟังก์ชั่นชื่อนี้ก็กดปุ่มเดิมอีกครั้งถ้าเครื่องพิมพ์คำว่า OUT OF KEY แสดงว่าปุ่มที่กดไม่ได้บันทึกสัญญาณควบคุมเอาไว้ ถ้าต้องการออกไปจากการส่งสัญญาณควบคุมให้กด quit

4. จากข้อ 2 เมื่อกด R ก็จะเป็นการรับข้อมูลเครื่องจะรายงานว่าตอนนี้มีเนื้อที่ในหน่วยความจำเพื่อที่จะเก็บสัญญาณได้อีกกี่สัญญาณจากนั้นกด ENTER ที่จอ LCDจะพิมพ์คำว่า WRITE ACCESSORY (คือเครื่องบอกให้เขียนชื่ออุปกรณ์ที่ต้องการรัยสัญญาณควบคุม) กดENTER ที่จอ LCD จะกระพริบรอรับการกดคีย์จากนั้นเราก็พิมพ์ชื่ออุปกรณ์ซึ่งชื่ออุปกรณ์ต้องไม่เกิน 16 ตัวอักษรเมื่อพิมพ์เสร็จกด ENTER ที่จอ LCDจะพิมพ์คำว่า WRITE FUNTION ให้กด ENTER จากนั้นที่จอ LCD จะกระพริบรอรับการกดคีย์ ให้พิมพ์ชื่อฟังก์ชันที่ต้องการรับ สัญญาณจากนั้นกด ENTER ที่จอ LCD จะ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

พิมพ์คำว่า ENTER FOR RECEIVE กด ENTER แล้วส่งสัญญาณควบคุมจากเครื่องควบคุมอุปกรณ์นั้นมาที่เครื่องควบคุมระยะไกลเอนกแบบโปรแกรมได้สัญญาณที่ส่งมาจะต้องเป็นฟังก์ชั่นเดียวกับที่พิมพ์ไว้จากนั้นที่จอ LCD จะพิมพ์คำว่า READY ถ้าต้องการรับฟังก์ชั่นต่อไปของเครื่องควบคุมเดิมให้กดปุ่ม F(funtion) ซึ่งขั้นตอนการใช้งานต่อไปก็จะเหมือนกับที่ได้กล่าวมาแล้วถ้าต้องการรับสัญญาณจากอุปกรณ์ตัวใหม่ให้กดคีย์ A (Accessory) ซึ่งขั้นตอนการใช้งานต่อไปก็จะเหมือนกับที่ได้กล่าวไปแล้วถ้าต้องการออกจากการรับข้อมูลให้กด quit

5.จากข้อ 2 เมื่อกด T จะเป็นการตั้งเวลาในการส่งสัญญาณออกไปควบคุมอุปกรณ์ที่ LCD ก็จะพิมพ์คำว่า SELECT ACCESSORY จากนั้นกด ENTER เครื่องก็จะพิมพ์ชื่ออุปกรณ์ กด ↑ หรือ ↓ ในการเลือกอุปกรณ์ถ้าเลือกได้แล้วกด ENTER ที่ LCD จะพิมพ์คำว่า SELECT FUNTION จากนั้นกด ↑ หรือ ↓ เพื่อเลือกฟังก์ชั่นเมื่อเลือกฟังก์ชั่นได้แล้วกด ENTER ที่LCD จะกระพริบ พิมพ์เวลาที่จะส่งฟังก์ชั่นที่เราเลือกไปควบคุมอุปกรณ์เมื่อพิมพ์เสร็จเครื่องจะพิมพ์ชื่อฟังก์ชั่นเดิมถ้าต้องการตั้งเวลาฟังก์ชั่นอื่นก็กด ↑ หรือ ↓ จากนั้นก็ทำเหมือนเดิมพอตั้งเวลาตามที่ต้องการแล้ว ก็กด quitที่ LCD จะพิมพ์คำว่า SELECT MENU จากนั้นกด L เครื่องก็จะแสดงเวลาจริง และพร้อมที่จะส่งสัญญาณไปควบคุมอุปกรณ์เมื่อถึงเวลาที่ตั้งไว้ ถ้าต้องการออกจากการตั้งเวลากด Q ก็จะกลับเข้าสู่เมนหลัก

6.จากข้อ 2.เมื่อกด U จะเป็นการแก้ไขค่าเวลาใน REAL RIME CLOCK ที่ LCD จะพิมพ์คำว่า UPDATE TIME กด ENTER ที่ LCD จะกระพริบพิมพ์ค่าเวลาใหม่แล้วกด ENTER ก็จะกลับเข้าสู่เมนหลักทันที

7.จากข้อ 2. เมื่อกด C เป็นการแก้ไขข้อมูลของสัญญาณควบคุมที่รับเข้ามาที่ LCD จะพิมพ์คำว่า SELECT ACCESSORY กด ENTER ที่ LCD จะพิมพ์ชื่ออุปกรณ์ กด หรือ เพื่อเลือกอุปกรณ์ที่ต้องการแก้ไขเมื่อเลือกได้แล้วกด ENTER เครื่องจะพิมพ์คำว่า SELECT FUNCTION เพื่อเลือกฟังก์ชั่นที่ที่จะทำการแก้ไข กด ENTER ที่ LCD จะพิมพ์คำว่า SELECT FUNCTION อยู่เช่นเดิม จากนั้นกด ENTER ที่ LCDจะกระพริบจากนั้นพิมพ์ชื่อใหม่เข้าไปแลวักด ENTER ที่จอ LCD จะพิมพ์คำว่า OK CH-CHANGED กด ENTER อีกครั้ง แล้วส่งสัญญาณจากเครื่องควบคุมในฟังก์ชั่นเดียวกันเข้ามาที่เครื่องควบคุมระยะไกลเอนกประสงค์แบบโปรแกรมได้จากนั้นเครื่องจะพิมพ์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

คำว่า READY ถ้าต้องการแก้ไขฟังก์ชั่นก็กดคีย์ quit แล้วกดคีย์ C อีกครั้งแล้วทำเหมือนที่ได้กล่าวมาแล้ว

หน้าที่ของคีย์พิเศษ

คีย์ SHIFT ใช้ในการเลือกตัวอักษรหรือตัวเลข

A<->1 ,B<->2,C<->3,D<->4,E<->5,F<->6,G<->7,H<->8,I<->9

,J<->10, K<->-,L<->+

คีย์ M-Z จะเปลี่ยนไปเมื่อกด SHIFT ต้องกดSHIFT อีกครั้งเพื่อให้เป็นอักษร M-Z เหมือนเดิม

คีย์ ← ใช้ในการลบตัวอักษรที่พิมพ์ผิด

คีย์ ↓ ใช้ในการเลื่อนดูอุปกรณ์และฟังก์ชั่น เป็นการเลื่อนลง

คีย์ ↑ ใช้ในการเลื่อนดูอุปกรณ์และฟังก์ชั่น เป็นการเลื่อนขึ้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                ORG     0000H
W1              EQU       00H
W2              EQU       01H
MG              EQU       5EH
F2              EQU       5FH
JIK             EQU       5AH
TOT             EQU       5BH
XDL             EQU       5CH
XDH             EQU       5DH
KDL             EQU       5EH
KDH             EQU       5FH
SREQ            EQU       60H
SRCQ            EQU       61H
SRR0            EQU       62H
SRR1            EQU       63H
STL             EQU       64H
STH             EQU       65H
COUNT           EQU       66H
SVT             EQU       67H
TDL             EQU       68H
TDH             EQU       69H
TCK             EQU       6AH
TCC             EQU       6BH
SRUL            EQU       6CH
SRUH            EQU       6DH
SRQU            EQU       6EH
SRKE            EQU       70H
SRPL            EQU       71H
SRPM            EQU       74H
SRTM            EQU       77H
SRDL            EQU       78H
SRDH            EQU       79H
CKR             EQU       7AH
COU             EQU       7BH
DA1             EQU       7CH
DA2             EQU       7DH

                LJMP START
                ORG 0003H
                SJMP INT00
                ORG 000BH
                LJMP  TF00
                ORG 0013H
                SJMP INT11
                ORG 001BH
                LJMP  TF11

INT00:          INC CKR
                INC CKR
                PUSH ACC
                SETB TCON.4
                CLR TCON.6
                MOV A,TL1
                MOVX @DPTR,A
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        INC DPTR
        MOV A,TH1
        MOVX @DPTR,A
        INC DPTR
        MOV TL1,#00H
        MOV TH1,#00H
        POP ACC
        RETI

TF00:   INC CKR
        INC CKR
        PUSH ACC
        CLR P3.0
        MOV TCON,#00H
        CLR C
        MOVX A,@DPTR
        SUBB A,#00H
        MOV R3,A
        INC DPTR
        MOVX A,@DPTR
        SUBB A,#00H
        MOV R4,A
        MOV A,#0DFH
        SUBB A,R3
        MOV TL1,A
        MOV A,#0FFH
        SUBB A,R4
        MOV TH1,A
        INC DPTR
        MOV TCON,#40H
        POP ACC
        RETI

INT11:  INC CKR
        INC CKR
        PUSH ACC
        SETB TCON.6
        CLR TCON.4
        MOV A,TL0
        MOVX @DPTR,A
        INC DPTR
        MOV A,TH0
        MOVX @DPTR,A
        INC DPTR
        MOV TL0,#00H
        MOV TH0,#00H
        POP ACC
        RETI

TF11:   INC CKR
        INC CKR
        PUSH ACC
        SETB P3.0
        MOV TCON,#00H
        CLR C
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        MOVX A,@DPTR
        SUBB A,#00H
        MOV R3,A
        INC DPTR
        MOVX A,@DPTR
        SUBB A,#00H
        MOV R4,A
        MOV A,#0DFH
        SUBB A,R3
        MOV TL0,A
        MOV A,#0FFH
        SUBB A,R4
        MOV TH0,A
        INC DPTR
        MOV TCON,#10H
        POP ACC
        RETI

WR16:   LCALL INIT2
        MOV P1,#80H
        LCALL GOTO
        MOV R0,#08H
ZEP:    MOV A,#00H
        MOVC A,@A+DPTR
        LCALL WRLCD
        INC DPTR
        DJNZ R0,ZEP
        MOV R0,#08H
        MOV P1,#0C0H
        LCALL GOTO
ZEP1:   MOV A,#00H
        MOVC A,@A+DPTR
        LCALL WRLCD
        INC DPTR
        DJNZ R0,ZEP1
        RET

BCDDIS: MOV A,R7
        ADD A,#30H
        MOV P1,#80H
        LCALL GOTO
        LCALL WRLCD
        MOV A,R6
        ANL A,#0F0H
        SWAP A
        ADD A,#30H
        MOV P1,#81H
        LCALL GOTO
        LCALL WRLCD
        MOV A,R6
        ANL A,#0FH
        ADD A,#30H
        MOV P1,#82H
        LCALL GOTO
        LCALL WRLCD
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        RET

BCD:    MOV R6,#00H
        MOV R7,#00H
        MOVX A,@DPTR
        JNB ACC.7,AT1
        MOV R6,#28H
        MOV R7,#01H

AT1:    JNB ACC.6,AT2
        PUSH ACC
        MOV A,#64H
        ADD A,R6
        DA A
        MOV R6,A
        MOV A,R7
        ADDC A,#00H
        MOV R7,A
        POP ACC
AT2:    JNB ACC.5,AT3
        PUSH ACC
        MOV A,#32H
        ADD A,R6
        DA A
        MOV R6,A
        MOV A,R7
        ADDC A,#00H
        MOV R7,A
        POP ACC
AT3:    JNB ACC.4,AT4
        PUSH ACC
        MOV A,#16H
        ADD A,R6
        DA A
        MOV R6,A
        MOV A,R7
        ADDC A,#00H
        MOV R7,A
        POP ACC
AT4:    JNB ACC.3,AT5
        PUSH ACC
        MOV A,#08H
        ADD A,R6
        DA A
        MOV R6,A
        MOV A,R7
        ADDC A,#00H
        MOV R7,A
        POP ACC
AT5:    JNB ACC.2,AT7
        PUSH ACC
        MOV A,#04H
        ADD A,R6
        DA A
        MOV R6,A
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        MOV A,R7
        ADDC A,#00H
        MOV R7,A
        POP ACC
AT8:    JNB ACC.0,AT9
        PUSH ACC
        MOV A,#01H
        ADD A,R6
        DA A
        MOV R6,A
        MOV A,R7
        ADDC A,#00H
        MOV R7,A
        POP ACC
AT9:    RET


START:  MOV DPTR,#1C90H
        LCALL WR16
        LCALL DELAY2
        MOV DPTR,#1CA0H
        LCALL WR16
        LCALL DELAY2
        MOV DPTR,#1D90H
        LCALL WR16
SR4:    LCALL SCANKEY
        MOV A,SRKE
        CJNE A,#0BH,SR1
        SJMP SR5
SR1:    CJNE A,#5EH,SR2
        LCALL SETUP
        SJMP SR5
SR2:    SJMP  SR4

SR5:    CLR P3.0
        SETB F2
START1: MOV KDL,#20H
        MOV KDH,#1BH
        MOV DPTR,#1CF0H
        LCALL WR16
AS1:    LCALL SCANKEY
        MOV A,SRKE
        CJNE A,#4DH,AOV1
        MOV DPTR,#0201H
        MOVX A,@DPTR
        INC A
        MOVX @DPTR,A
        MOV DPTR,#0209H
        MOV A,#01H
        MOVX @DPTR,A
        LJMP ASTR
AOV1:   CJNE A,#5DH,ASV
        LJMP DSENSEC
        SJMP AS1
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
ASV:    CJNE A,#1DH,OVM
        LCALL UT
        SJMP START1
OVM:    CJNE A,#2BH,OVM1
        LJMP ALM
OVM1:   CJNE A,#6DH,OVM2
        LJMP SELECTC
OVM2:   CJNE A,#3EH,IS1
        MOV DPL,KDL
        MOV DPH,KDH
        MOV A,KDL
        CJNE A,#0FH,IS7
        MOV KDL,#0F0H
        MOV KDH,#1BH
        SJMP IS2
IS7:    ADD A,#10H
        MOV KDL,A
        MOV A,KDH
        ADDC A,#00H
        MOV KDH,A
IS2:    MOV DPL,KDL
        MOV DPH,KDH
        LCALL WR16
        LJMP AS1
IS1:    CJNE A,#2EH,IS3
        MOV DPL,KDL
        MOV DPH,KDH
        MOV A,DPL
        CJNE A,#30H,IS4
        SJMP IS5
IS4:    CLR C
        SUBB A,#10H
        MOV KDL,A
        MOV A,KDH
        SUBB A,#00H
        MOV KDH,A
IS5:    MOV DPL,KDL
        MOV DPH,KDH
        LCALL WR16
        LJMP AS1
IS3:    CJNE A,#37H,IS6
        LJMP UCN
IS6:    AJMP AS1

ASTR:   MOV DPTR,#0200H
        LCALL BCD
        LCALL INIT2
        LCALL BCDDIS
        MOV DPTR,#1DA0H
        MOV P1,#83H
        LCALL GOTO
        MOV R0,#05H
TURN1:  MOV A,#00H
        MOVC A,@A+DPTR
        LCALL WRLCD
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        INC DPTR
        DJNZ R0,TURN1
        MOV R0,#08H
        MOV P1,#0C0H
        LCALL GOTO

TURN2:  MOV A,#00H
        MOVC A,@A+DPTR
        LCALL WRLCD
        INC DPTR
        DJNZ R0,TURN2
        LCALL ENT
        MOV DPTR,#1DE0H
        LCALL WR16
        LCALL ENT
EQUIP:  MOV DPTR,#0201H
        MOVX A,@DPTR
        ADD A,#30H
        LCALL INIT2
        MOV P1,#80H
        LCALL GOTO
        LCALL WRLCD
        MOV A,#2EH
        LCALL WRLCD
        MOV DPTR,#0202H
        MOVX A,@DPTR
        MOV R4,A
        ADD A,#13H
        MOVX @DPTR,A
        INC DPTR
        MOVX A,@DPTR
        MOV R5,A
        ADDC A,#00H
        MOVX @DPTR,A
        MOV DPTR,#020AH
        MOVX A,@DPTR
        MOV DPL,R4
        MOV DPH,R5
        MOVX @DPTR,A
        MOV DPTR,#0204H
        MOVX A,@DPTR
        MOV DPL,R4
        MOV DPH,R5
        INC DPTR
        MOVX @DPTR,A
        INC DPTR
        MOV R4,DPL
        MOV R5,DPH
        MOV DPTR,#0205H
        MOVX A,@DPTR
        MOV DPL,R4
        MOV DPH,R5
        MOVX @DPTR,A
        INC DPTR
        MOV R4,DPL
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        MOV R5,DPH
        LCALL ENT
        LCALL INIT1
        LCALL SCANDIS
        MOV DPTR,#1DD0H
        LCALL WR16
        LCALL ENT
FUNC:   LCALL INIT1
        MOV DPTR,#0204H
        MOVX A,@DPTR
        MOV R4,A
        ADD A,#13H
        MOVX @DPTR,A
        INC DPTR
        MOVX A,@DPTR
        MOV R5,A
        ADDC A,#00H
        MOVX @DPTR,A
        MOV DPTR,#0206H
        MOVX A,@DPTR
        MOV R6,A
        INC DPTR
        MOVX A,@DPTR
        MOV R7,A
        MOV DPL,R4
        MOV DPH,R5
        INC DPTR
        MOV A,R6
        MOVX @DPTR,A
        INC DPTR
        MOV A,R7
        MOVX @DPTR,A
        MOV DPTR,#0209H
        MOVX A,@DPTR
        MOV DPL,R4
        MOV DPH,R5
        MOVX @DPTR,A
        MOV DPTR,#0209H
        MOVX A,@DPTR
        INC A
        MOVX @DPTR,A
        MOV DPL,R4
        MOV DPH,R5
        INC DPTR
        INC DPTR
        INC DPTR
        MOV R4,DPL
        MOV R5,DPH
        LCALL SCANDIS
        MOV DPTR,#1DC0H
        LCALL WR16
        LCALL ENT
        MOV DPTR,#0206H
        MOVX A,@DPTR
        MOV R4,A
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        INC DPTR
        MOVX A,@DPTR
        MOV R5,A
        LCALL RECIVE
        MOV DPTR,#1DB0H
        LCALL WR16
        MOV DPTR,#0207H
        MOVX A,@DPTR
        INC A
        MOVX @DPTR,A
        MOV DPTR,#0200H
        MOVX A,@DPTR
        DEC A
        JNZ OV12
        LJMP START
OV12:   MOVX @DPTR,A
        MOV DPTR,#0202H
        MOVX A,@DPTR
        CLR C
        SUBB A,#13H
        MOV R4,A
        INC DPTR
        MOVX A,@DPTR
        SUBB A,#00H
        MOV DPH,A
        MOV DPL,R4
        MOVX A,@DPTR
        INC A
        MOVX @DPTR,A
RESA1:  LCALL SCANKEY
        MOV A,SRKE
        CJNE A,#5DH,OV13
        LJMP SENSEC
OV13:   CJNE A,#4BH,OV14
        LJMP FUNC
OV14:   CJNE A,#57H,ODP
        MOV DPTR,#0201H
        MOVX A,@DPTR
        INC A
        MOVX @DPTR,A
        MOV DPTR,#0209H
        MOV A,#01H
        MOVX @DPTR,A
        LJMP EQUIP
ODP:    CJNE A,#77H,ODP1
        LJMP START1
ODP1:   SJMP RESA1


SENSEC: MOV SRCQ,#00H
        MOV DPTR,#1D10H
        LCALL WR16
        LCALL ENT
        MOV DPTR,#0201H
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        MOV DPL,A
        MOV DPH,R5
        CJNE A,#0FDH,OV17
        MOV DPTR,#0210H
OV17:   SJMP BCR
OV16:   CJNE A,#4EH,RESA7
        MOV DPL,R4
        MOV DPH,R5
        MOVX A,@DPTR
        MOV R4,A
        INC DPTR
        MOVX  A,@DPTR
        MOV R5,A
        MOV DPTR,#1D00H
        LCALL WR16
        LCALL ENT
JE0:    MOV DPL,R4
        MOV DPH,R5
        MOV SRDL,R4
        MOV SRDH,R5
JE8:    MOV A,SRDL
        MOV R4,A
        MOV A,SRDH
        MOV R5,A
        MOV DPL,R4
        MOV DPH,R5
        MOV A,R4
        ADD A,#03H
        MOV R4,A
        MOV A,R5
        ADDC A,#00H
        MOV R5,A
        MOV DPL,R4
        MOV DPH,R5
        LCALL WRRAM
        LCALL SCANKEY
        MOV A,SRKE
        CJNE A,#4EH,OV18
        CLR C
        MOV A,R4
        SUBB A,#03H
        MOV DPL,A
        MOV A,R5
        SUBB A,#00H
        MOV DPH,A
        MOV SRDH,A
        MOV A,DPL
        MOV SRDL,A
        MOV A,R4
        INC DPTR
        MOVX A,@DPTR
        MOV R4,A
        INC DPTR
        MOVX A,@DPTR
        MOV R5,A
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        LCALL SEND
        LJMP JE8
OV18:   CJNE A,#2EH,OV19
        MOV A,SRDH
        MOV DPH,A
        MOV A,SRDL
        MOV DPL,A
        MOV A,SRTM
        CLR C
        SUBB A,#00H
        JNZ  PA1
        SJMP JE8
PA1:    MOV A,DPL
        CLR C
        SUBB A,#13H
        MOV R4,A
        MOV SRDL,A
        MOV A,DPH
        SUBB A,#00H
        MOV R5,A
        MOV SRDH,A
        DEC SRTM
        LJMP JE8
OV19:   CJNE A,#3EH,OV20
        INC SRTM
        MOV A,SRDH
        MOV DPH,A
        MOV A,SRDL
        MOV DPL,A
        MOV A,SRTM
        CLR C
        SUBB A,SRPM
        JNZ OV21
        MOV SRTM,#00H
        MOV A,SRDH
        MOV DPH,A
        MOV A,SRDL
        MOV DPL,A
        MOV A,SRPM
        DEC A
        MOV B,#13H
        MUL AB
        CLR C
        MOV DPL,A
        MOV A,SRDL
        SUBB A,DPL
        MOV SRDL,A
        MOV A,B
        MOV DPH,A
        MOV A,SRDH
        SUBB A,DPH
        MOV SRDH,A
        LJMP JE8
OV21:   MOV A,#01H
        MOV B,#13H
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        MOVX A,@DPTR
        MOV R6,A
        MOV A,#0FFH
        SUBB A,R6
        MOV TH0,A
        INC DPTR
        SETB TCON.4
        SETB P3.0
CH2:    MOV A,CKR
        CJNE A,#0FCH,CH2
        INC MG
        MOV A,MG
        CJNE A,#02H,SEND1
        CLR P3.0
        MOV TCON,#00H
        RET

INIT1:  CLR P3.1
        CLR P3.4
        CLR P3.5
        MOV P1,#38H
        SETB P3.5
        LCALL DELAY1
        CLR P3.5
        MOV P1,#0FH
        SETB P3.5
        LCALL DELAY1
        CLR P3.5
        MOV P1,#06H
        SETB P3.5
        LCALL DELAY1
        CLR P3.5
        MOV P1,#01H
        SETB P3.5
        LCALL DELAY1
        CLR P3.5
        RET
INIT2:  CLR P3.1
        CLR P3.4
        CLR P3.5
        MOV P1,#38H
        SETB P3.5
        LCALL DELAY1
        CLR P3.5
        MOV P1,#0CH
        SETB P3.5
        LCALL DELAY1
        CLR P3.5
        MOV P1,#06H
        SETB P3.5
        LCALL DELAY1
        CLR P3.5
        MOV P1,#01H
        SETB P3.5
        LCALL DELAY1
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        CLR P3.5
        RET

WRLCD:  SETB P3.4
        MOV P1,A
        SETB P3.5
        LCALL DELAY1
        CLR P3.5
        CLR P3.4
        LCALL CHF
        RET

  CHF:  SETB P3.1
        SETB P3.5
        LCALL DELAY1
        CLR P3.5
        JB P1.7,CHF
        CLR  P3.1
        RET

DELAY1: MOV R2,#02H
LOPP1:  MOV R3,#0FFH
LOPP:   DJNZ R3,LOPP
        DJNZ R2,LOPP1
        RET

DELAY2: MOV R2,#0FFH
L1:     MOV R3,#0FFH
L2:     DJNZ R3,L2
        DJNZ R2,L1
        MOV R2,#0FFH
L3:     MOV R3,#0FFH
L4:     DJNZ R3,L4
        DJNZ R2,L3
        MOV R2,#0FFH
L5:     MOV R3,#0FFH
L6:     DJNZ R3,L6
        DJNZ R2,L5
        MOV R2,#0FFH
L7:     MOV R3,#0FFH
L8:     DJNZ R3,L8
        DJNZ R2,L7
        MOV R2,#0FFH
Z7:     MOV R3,#0FFH
Z8:     DJNZ R3,Z8
        DJNZ R2,Z7
        MOV R2,#0FFH
Z5:     MOV R3,#0FFH
Z6:     DJNZ R3,Z6
        DJNZ R2,Z5
        MOV R2,#0FFH
Z4:     MOV R3,#0FFH
Z3:     DJNZ R3,Z3
        DJNZ R2,Z4
        MOV R2,#0FFH
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                MOVX A,@DPTR
                ANL A,#0F0H
                SWAP A
                ADD A,#30H
                MOV P1,#86H
                LCALL GOTO
                LCALL WRLCD
                MOV P1,#85H
                LCALL GOTO
                MOV A,#2EH
                LCALL WRLCD
                MOV DPTR,#8004H
                MOVX A,@DPTR
                ANL A,#0FH
                ADD A,#30H
                MOV P1,#84H
                LCALL GOTO
                LCALL WRLCD
                MOV DPTR,#8004H
                MOVX A,@DPTR
                ANL A,#0F0H
                SWAP A
                ADD A,#30H
                MOV P1,#83H
                LCALL GOTO
                LCALL WRLCD
                POP DPL
                POP DPH
                RET

KEYDOWN:        MOV A,#0FH
                ADD A,R0
                XRL A,P1
                RET

SCANKEY:        MOV R1,#08H
SCANKEY2:       MOV R0,#00H
SCANKEY1:       MOV A,#0FH
                ADD A,R0
                MOV P1,A
                LCALL KEYDOWN
                JZ NP
                MOV A,P1
                MOV SRKE,A
                LCALL DELAY3
                LCALL KEYDOWN
                JZ NP
                MOV A,P1
                CJNE A,SRKE,NP
                LCALL DELAY3
RECHK:          LCALL KEYDOWN
                JNZ  RECHK
                LCALL DELAY4
                LCALL KEYDOWN
                JNZ  RECHK
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        RET
NP:     MOV A,#10H
        ADD A,R0
        MOV R0,A
        DJNZ R1,SCANKEY1
        SJMP SCANKEY

SCANDIS: MOV SRPL,#80H
        LCALL INIT1
        MOV P1,#80H
        LCALL GOTO
RESCAN: LCALL SCANKEY
        MOV A,SRKE
        LCALL TRAN
        PUSH DPH
        PUSH DPL
        MOV DPL,R4
        MOV DPH,R5
        MOVX @DPTR,A
        INC DPTR
        MOV R4,DPL
        MOV R5,DPH
        POP DPL
        POP DPH
        LCALL WRLCD
        INC SRPL
        MOV A,SRPL
        JB ACC.6,CB
        XRL A,#88H
        JNZ  RESCAN
        MOV P1,#0C0H
        LCALL GOTO
        MOV SRPL,#0C0H
        SJMP RESCAN
CB:     XRL A,#0C8H
        JNZ  RESCAN
        MOV SRPL,#80H
        LCALL DELAY2
        MOV P1,#01H
        LCALL GOTO
        MOV P1,#80H
        LCALL GOTO
        SJMP RESCAN

TRAN:   CJNE A,#67H,K1
        JBC F2,TER
        SETB F2
TER:    POP ACC
        POP ACC
        LJMP RESCAN
K1:     CJNE A,#1EH,T0
        MOV A,SRPL
        CJNE A,#80H,K3
        MOV SRPL,#0C7H
        SJMP K4
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
K3:     CJNE A,#0C0H,K5
        MOV SRPL,#87H
        SJMP K4
K5:     DEC SRPL
K4:     MOV A,SRPL
        MOV P1,A
        LCALL GOTO
        POP ACC
        POP ACC
        MOV A,R4
        CLR C
        SUBB A,#01H
        MOV R4,A
        MOV A,R5
        SUBB A,#00H
        MOV R5,A
        LJMP RESCAN
T0:     CJNE A,#77H,T1
        POP ACC
        POP ACC
        POP ACC
        POP ACC
        LJMP START1
T1:     CJNE A,#2EH,T2
        MOV A,#20H
        RET
T2:     CJNE A,#3EH,T3
        MOV A,#2FH
        RET
T3:     CJNE A,#4EH,T4
        MOV P1,#01H
        LCALL GOTO
        MOV P1,#80H
        LCALL GOTO
        POP ACC
        POP ACC
        RET
T4:     JB  F2,K6
        MOV DPTR,#1C00H
        MOVC A,@A+DPTR
        RET
K6:     MOV DPTR,#1C01H
        MOVC A,@A+DPTR
        RET


WRRAM:  LCALL INIT2
        MOV R0,#08H
        MOV P1,#80H
        LCALL GOTO
GUO:    MOVX A,@DPTR
        LCALL WRLCD
        INC DPTR
        DJNZ R0,GUO
        MOV R0,#08H
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        MOV SREQ,A ;STORE NUMBER OF ACCESSORY
        LCALL BCD
        LCALL INIT2
        LCALL BCDDIS
        MOV DPTR,#0210H
BCRC:   MOV A,DPL
        ADD A,#01H
        MOV R4,A
        MOV A,DPH
        ADDC A,#00H
        MOV R5,A
        MOVX A,@DPTR
        MOV SRPM,A
        MOV SRTM,#00H
        INC DPTR
        INC DPTR
        INC DPTR
RESA7C: LCALL WRRAM
        LCALL SCANKEY
        MOV A,SRKE
        CJNE A,#3EH,OAA1C
        MOV A,R4
        ADD A,#12H
        MOV R4,A
        MOV DPL,A
        MOV A,R5
        ADDC A,#00H
        MOV R5,A
        MOV DPH,A
        MOV A,SREQ
        MOV B,#13H
        MUL AB
        ADD A,#10H
        MOV SRR0,A
        MOV A,B
        ADDC A,#02H
        MOV SRR1,A
        MOV A,R5
        CJNE A,SRR1,CEQ1C
        INC SRCQ
CEQ1C:  MOV A,R4
        CJNE A,SRR0,CEQ2C
        INC SRCQ
CEQ2C:  MOV A,SRCQ
        CJNE A,#02H,CEQ3C
        MOV DPTR,#0210H
        SJMP BCRC
CEQ3C:  MOV SRCQ,#00H
        SJMP BCRC
OAA1C:  CJNE A,#2EH,OV16C
        MOV A,R4
        CLR C
        SUBB A,#14H
        MOV R4,A
        MOV DPL,A
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        MOV DPH,R5
        CJNE A,#0FDH,OV17C
        MOV DPTR,#0210H
OV17C:  SJMP BCRC
OV16C:  CJNE A,#4EH,RESA7C
        MOV DPL,R4
        MOV DPH,R5
        MOVX A,@DPTR
        MOV R4,A
        INC DPTR
        MOVX  A,@DPTR
        MOV R5,A
        MOV DPTR,#1D00H
        LCALL WR16
        LCALL ENT
JE0C:   MOV DPL,R4
        MOV DPH,R5
        MOV SRDL,R4
        MOV SRDH,R5
JE8C:   MOV A,SRDL
        MOV R4,A
        MOV A,SRDH
        MOV R5,A
        MOV DPL,R4
        MOV DPH,R5
        MOV A,R4
        ADD A,#03H
        MOV R4,A
        MOV A,R5
        ADDC A,#00H
        MOV R5,A
        MOV DPL,R4
        MOV DPH,R5
        LCALL WRRAM
        LCALL SCANKEY
        MOV A,SRKE
        CJNE A,#4EH,OV18C
        CLR C
        MOV A,R4
        SUBB A,#03H
        MOV DPL,A
        MOV A,R5
        SUBB A,#00H
        MOV DPH,A
        MOV SRDH,A
        MOV A,DPL
        MOV SRDL,A
        MOV A,R4
        INC DPTR
        MOVX A,@DPTR
        MOV R4,A
        INC DPTR
        MOVX A,@DPTR
        MOV R5,A
        LCALL TS
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        LJMP JE8C
OV18C:  CJNE A,#2EH,OV19C
        MOV A,SRDH
        MOV DPH,A
        MOV A,SRDL
        MOV DPL,A
        MOV A,SRTM
        CLR C
        SUBB A,#00H
        JNZ  PA1C
        SJMP JE8C
PA1C:   MOV A,DPL
        CLR C
        SUBB A,#13H
        MOV R4,A
        MOV SRDL,A
        MOV A,DPH
        SUBB A,#00H
        MOV R5,A
        MOV SRDH,A
        DEC SRTM
        LJMP JE8C
OV19C:  CJNE A,#3EH,OV20C
        INC SRTM
        MOV A,SRDH
        MOV DPH,A
        MOV A,SRDL
        MOV DPL,A
        MOV A,SRTM
        CLR C
        SUBB A,SRPM
        JNZ OV21C
        MOV SRTM,#00H
        MOV A,SRDH
        MOV DPH,A
        MOV A,SRDL
        MOV DPL,A
        MOV A,SRPM
        DEC A
        MOV B,#13H
        MUL AB
        CLR C
        MOV DPL,A
        MOV A,SRDL
        SUBB A,DPL
        MOV SRDL,A
        MOV A,B
        MOV DPH,A
        MOV A,SRDH
        SUBB A,DPH
        MOV SRDH,A
        LJMP JE8C
OV21C:  MOV A,#01H
        MOV B,#13H
        MUL AB
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        ADD A,DPL
        MOV DPL,A
        MOV A,B
        ADDC A,DPH
        MOV DPH,A
        MOV R4,DPL
        MOV R5,DPH
        MOV SRDL,R4
        MOV SRDH,R5
        LJMP JE0C
OV20C:  CJNE A,#57H,OV22C
        LJMP SELECTC
OV22C:  CJNE A,#77H,OV23C
        LJMP START1
OV23C:  LJMP JE8C
  TS:   MOV A,STL
        MOV DPL,A
        MOV A,STH
        MOV DPH,A
        MOV A,R4
        MOVX @DPTR,A
        MOV A,R5
        INC DPTR
        MOVX @DPTR,A
        INC DPTR
        MOV R4,DPL
        MOV R5,DPH
        LCALL SCANDIS
        MOV STL,R4
        MOV STH,R5
        INC COUNT
        MOV A,COUNT
        CJNE A,#0AH,NT1
        MOV DPTR,#1CC0H
        LCALL WR16
        LCALL ENT
        MOV DPTR,#1CD0H
        LCALL WR16
        LCALL ENT
        MOV STL,#00H
        MOV STH,#01H
        RET
 NT1:   RET

 ALM:   MOV TCC,#00H
        MOV COU,#00H
        LCALL INIT2
        MOV DPTR,#0100H
ALM1:   INC COU
        MOV A,COU
        CJNE A,#0AH,ALM2
        MOV DPTR,#0100H
        MOV TDL,DPL
        MOV TDH,DPH
        MOV COU,#00H
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
ALM2:  MOVX A,@DPTR
       MOV R4,A
       LCALL TIME
       INC DPTR
       MOVX A,@DPTR
       MOV R5,A
       INC DPTR
       MOVX A,@DPTR
       ANL A,#0FH
       SWAP A
       MOV R0,A
       INC DPTR
       MOVX A,@DPTR
       ANL A,#0FH
       ORL A,R0
       MOV SVT,A
       INC DPTR
       MOV TDH,DPH
       MOV TDL,DPL
       MOV DPTR,#8004H
       MOVX A,@DPTR
       CJNE A,SVT,TIM6
       MOV DPL,TDL
       MOV DPH,TDH
       MOVX A,@DPTR
       ANL A,#0FH
       SWAP A
       MOV R0,A
       INC DPTR
       MOVX A,@DPTR
       ANL A,#0FH
       ORL A,R0
       MOV SVT,A
       INC DPTR
       MOV TDL,DPL
       MOV TDH,DPH
       MOV DPTR,#8002H
       MOVX A,@DPTR
TIM1:  CJNE A,SVT,TIM7
       MOV DPL,TDL
       MOV DPH,TDH
       MOVX A,@DPTR
       ANL A,#0FH
       SWAP A
       MOV R0,A
       INC DPTR
       MOVX A,@DPTR
       ANL A,#0FH
       ORL A,R0
       MOV SVT,A
       INC DPTR
       MOV TDL,DPL
       MOV TDH,DPH
       MOV DPTR,#8000H
       MOVX A,@DPTR
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
TIM2:  CJNE A,SVT,TIM3
       LCALL SEND
       INC TCC
       MOV A,TCC
       CJNE A,#20H,TIM4
       MOV DPTR,#1A00H
       LCALL WR16
       LCALL ENT
       LJMP START1
TIM3:  MOV DPL,TDL
       MOV DPH,TDH
TIM4:  MOV A,#0FH
       MOV R0,#30H
       ADD A,R0
       MOV P1,A
       LCALL KEYDOWN
       JNZ TIM5
       LJMP ALM1
TIM5:  MOV DPTR,#1CB0H
       LCALL WR16
       LJMP START1
TIM6:  MOV A,TDL
       ADD A,#04H
       MOV DPL,A
       MOV A,TDH
       ADDC A,#00H
       MOV DPH,A
       SJMP ALM3
TIM7:  MOV A,TDL
       ADD A,#02H
       MOV DPL,A
       MOV A,TDH
       ADDC A,#00H
       MOV DPH,A
ALM3:  MOV R0,#30H
       MOV A,#0FH
       ADD A,R0
       MOV P1,A
       LCALL KEYDOWN
       JNZ ALM4
       LJMP ALM1
ALM4:  LCALL KEYDOWN
       JNZ ALM4
       LJMP START1


UPC:   MOV SRCQ,#00H
       MOV TOT,#00H
       MOV DPTR,#1D10H
       LCALL WR16
       LCALL ENT
 AG:   MOV DPTR,#0210H
 C1:   MOV A,DPL
       MOV SRUL,A
       ADD A,#01H
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        MOV R4,A
        MOV A,DPH
        MOV SRUH,A
        ADDC A,#00H
        MOV R5,A
        MOVX A,@DPTR
        MOV SRPM,A
        MOV SRTM,#00H
        INC DPTR
        INC DPTR
        INC DPTR
C2:     LCALL WRRAM
        LCALL SCANKEY
        MOV A,SRKE
        CJNE A,#3EH,C3
        MOV A,R4
        ADD A,#12H
        MOV R4,A
        MOV DPL,A
        MOV A,R5
        ADDC A,#00H
        MOV R5,A
        MOV DPH,A
        MOV A,SREQ
        MOV B,#13H
        MUL AB
        ADD A,#10H
        MOV SRR0,A
        MOV A,B
        ADDC A,#02H
        MOV SRR1,A
        MOV A,R5
        CJNE A,SRR1,EQ1
        INC SRCQ
EQ1:    MOV A,R4
        CJNE A,SRR0,EQ2
        INC SRCQ
EQ2:    MOV A,SRCQ
        CJNE A,#02H,EQ3
        MOV DPTR,#0210H
        SJMP C1
EQ3:    MOV SRCQ,#00H
        SJMP C1
C3:     CJNE A,#2EH,C4
        MOV A,R4
        CLR C
        SUBB A,#14H
        MOV R4,A
        MOV DPL,A
        MOV DPH,R5
        CJNE A,#0FDH,C5
        MOV DPTR,#0210H
C5:     SJMP C1
C4:     CJNE A,#4EH,C2
        MOV DPL,R4
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        MOV DPH,R5
        MOVX A,@DPTR
        MOV R4,A
        INC DPTR
        MOVX  A,@DPTR
        MOV R5,A
        MOV A,R4
        PUSH A
        MOV A,R5
        PUSH A
        MOV DPTR,#1B00H
        LCALL WR16
        LCALL SDOWN
        MOV DPTR,#1B10H
        LCALL WR16
        LCALL SDOWN
        MOV DPTR,#1B20H
        LCALL WR16

UC11:   LCALL SCANKEY
        MOV A,SRKE
        CJNE A,#0BH,UC8
        SJMP UC10
UC8:    CJNE A,#5EH,UC9
        MOV DPH,SRUH
        MOV DPL,SRUL
        INC DPTR
        INC DPTR
        INC DPTR
        MOV R4,DPL
        MOV R5,DPH
        LCALL INIT1
        LCALL SCANDIS
        MOV DPTR,#1DD0H
        LCALL WR16
        LCALL ENT
        SJMP UC10

UC9:    SJMP UC11
UC10:   POP A
        MOV R5,A
        POP A
        MOV R4,A
UFC:    LCALL INIT1
        MOV DPH,R5
        MOV DPL,R4
        MOVX @DPTR,A
        INC DPTR
        MOVX A,@DPTR
        MOV R6,A
        INC DPTR
        MOVX A,@DPTR
        MOV R7,A
        INC DPTR
        MOV R4,DPL
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        LCALL INIT2
        LCALL BCDDIS
        MOV DPTR,#0210H
DBCR:   MOV A,DPL
        ADD A,#01H
        MOV R4,A
        MOV A,DPH
        ADDC A,#00H
        MOV R5,A
        MOVX A,@DPTR
        MOV SRPM,A
        MOV SRTM,#00H
        INC DPTR
        INC DPTR
        INC DPTR
DRESA7: LCALL WRRAM
        LCALL SCANKEY
        MOV A,SRKE
        CJNE A,#3EH,DOAA1
        MOV A,R4
        ADD A,#12H
        MOV R4,A
        MOV DPL,A
        MOV A,R5
        ADDC A,#00H
        MOV R5,A
        MOV DPH,A
        MOV A,SREQ
        MOV B,#13H
        MUL AB
        ADD A,#10H
        MOV SRR0,A
        MOV A,B
        ADDC A,#02H
        MOV SRR1,A
        MOV A,R5
        CJNE A,SRR1,DCEQ1
        INC SRCQ
DCEQ1:  MOV A,R4
        CJNE A,SRR0,DCEQ2
        INC SRCQ
DCEQ2:  MOV A,SRCQ
        CJNE A,#02H,DCEQ3
        MOV DPTR,#0210H
        SJMP DBCR
DCEQ3:  MOV SRCQ,#00H
        SJMP DBCR
DOAA1:  CJNE A,#2EH,DOV16
        MOV A,R4
        CLR C
        SUBB A,#14H
        MOV R4,A
        MOV DPL,A
        MOV DPH,R5
        CJNE A,#0FDH,DOV17
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        MOV DPTR,#0210H
DOV17:  SJMP DBCR
DOV16:  CJNE A,#4EH,DRESA7
        MOV DPL,R4
        MOV DPH,R5
        MOVX A,@DPTR
        MOV R4,A
        INC DPTR
        MOVX  A,@DPTR
        MOV R5,A
        MOV DPTR,#1D00H
        LCALL WR16
        LCALL ENT
DJE0:   MOV DPL,R4
        MOV DPH,R5
        MOV SRDL,R4
        MOV SRDH,R5
        CLR F2
 JF1:   LCALL SCANKEY
        MOV A,SRKE
        LCALL DTRS
 JF4:   MOV R6,A
        MOV A,SRDL
        MOV DPL,A
        MOV A,SRDH
        MOV DPH,A
        MOV A,R6
        MOV B,#13H
        MUL AB
        ADD A,DPL
        MOV DPL,A
        MOV A,B
        ADDC A,DPH
        MOV DPH,A
        MOV A,DPL
        ADD A,#03H
        MOV DPL,A
        MOV DA1,A
        MOV A,DPH
        ADDC A,#00H
        MOV DPH,A
        MOV DA2,A
        LCALL WRRAM
        MOV A,R6
        MOV B,A
        LCALL SCANKEY
        MOV A,SRKE
        LCALL TR2
        CJNE A,B,JF
        MOV DPL,DA1
        MOV DPH,DA2
        MOV A,DPL
        CLR C
        SUBB A,#02H
        MOV DPL,A
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        MOV A,DPH
        SUBB A,#00H
        MOV DPH,A
        MOVX A,@DPTR
        MOV R4,A
        INC DPTR
        MOVX A,@DPTR
        MOV R5,A
        LCALL SEND
        SJMP JF1
JF:     CJNE A,#77H,JF3
        LJMP START1
JF3:    LCALL UFG
        SJMP  JF4

TR2:    CJNE A,#67H,K12
        JBC F2,TER2
        SETB F2
TER2:   POP ACC
        POP ACC
        LJMP JF1
K12:    CJNE A,#77H,T44
        RET
T44:    JB  F2,K62
        MOV DPTR,#1900H
        MOVC A,@A+DPTR
        RET
K62:    MOV DPTR,#1980H
        MOVC A,@A+DPTR
        RET
DTRS:   CJNE A,#77H,DTR29
        LCALL START1
DTR29:  LCALL TR2
UFG:    PUSH A
        INC A
        MOV SRTM,A
        MOV A,SRPM
        CLR C
        SUBB A,SRTM
        JNC WS1
        MOV DPTR,#1A40H
        LCALL WR16
        SJMP WS2
WS1:    POP A
        RET
WS2:    POP A
        POP SRTM
        POP SRTM
        LJMP JF1

UCN:    MOV SRCQ,#00H
        MOV DPTR,#1D10H
        LCALL WR16
        LCALL ENT
        MOV DPTR,#0201H
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        MOVX A,@DPTR
        MOV SREQ,A ;STORE NUMBER OF ACCESSORY
        LCALL BCD
        LCALL INIT2
        LCALL BCDDIS
        MOV DPTR,#0210H
ZBCR:   MOV A,DPL
        ADD A,#01H
        MOV R4,A
        MOV A,DPH
        ADDC A,#00H
        MOV R5,A
        MOVX A,@DPTR
        MOV SRPM,A
        MOV SRTM,#00H
        INC DPTR
        INC DPTR
        INC DPTR
ZRESA7: LCALL WRRAM
        LCALL SCANKEY
        MOV A,SRKE
        CJNE A,#3EH,ZOAA1
        MOV A,R4
        ADD A,#12H
        MOV R4,A
        MOV DPL,A
        MOV A,R5
        ADDC A,#00H
        MOV R5,A
        MOV DPH,A
        MOV A,SREQ
        MOV B,#13H
        MUL AB
        ADD A,#10H
        MOV SRR0,A
        MOV A,B
        ADDC A,#02H
        MOV SRR1,A
        MOV A,R5
        CJNE A,SRR1,ZCEQ1
        INC SRCQ
ZCEQ1:  MOV A,R4
        CJNE A,SRR0,ZCEQ2
        INC SRCQ
ZCEQ2:  MOV A,SRCQ
        CJNE A,#02H,ZCEQ3
        MOV DPTR,#0210H
        SJMP ZBCR
ZCEQ3:  MOV SRCQ,#00H
        SJMP ZBCR
ZOAA1:  CJNE A,#2EH,ZOV16
        MOV A,R4
        CLR C
        SUBB A,#14H
        MOV R4,A
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                MOV DPL,A
                MOV DPH,R5
                CJNE A,#0FDH,ZOV17
                MOV DPTR,#0210H
ZOV17:          SJMP ZBCR
ZOV16:          CJNE A,#4EH,ZRESA7
                MOV DPL,R4
                MOV DPH,R5
                MOVX A,@DPTR
                MOV R4,A
                INC DPTR
                MOVX  A,@DPTR
                MOV R5,A
                MOV DPTR,#1D00H
                LCALL WR16
                LCALL ENT
ZJE0:           MOV DPL,R4
                MOV DPH,R5
                MOV SRDL,R4
                MOV SRDH,R5
                CLR F2
ZF1:            LCALL SCANKEY
                MOV A,SRKE
                LCALL ZDTRS
ZF4:            MOV JIK,A
                MOV A,SRDL
                MOV DPL,A
                MOV A,SRDH
                MOV DPH,A
                MOV A,JIK
                MOV B,#13H
                MUL AB
                ADD A,DPL
                MOV DPL,A
                MOV A,B
                ADDC A,DPH
                MOV DPH,A
                MOV A,DPL
                ADD A,#03H
                MOV DPL,A
                MOV DA1,A
                MOV R4,A
                MOV A,DPH
                ADDC A,#00H
                MOV DPH,A
                MOV DA2,A
                MOV R5,A
                LCALL WRRAM
                LCALL ENT
                LCALL INIT1
                LCALL SCANDIS
                MOV DPTR,#1A30H
                LCALL WR16
                MOV A,JIK
                MOV B,A
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        LCALL SCANKEY
        MOV A,SRKE
        LCALL ZTR2
        CJNE A,#7BH,ZF
        MOV DPL,DA1
        MOV DPH,DA2
        MOV A,DPL
        CLR C
        SUBB A,#02H
        MOV DPL,A
        MOV A,DPH
        SUBB A,#00H
        MOV DPH,A
        MOVX A,@DPTR
        MOV R4,A
        INC DPTR
        MOVX A,@DPTR
        MOV R5,A
        LCALL RECIVE
        MOV DPTR,#1DB0H
        LCALL WR16
        SJMP ZF1
ZF:     CJNE A,#77H,ZF3
        LJMP START1
ZF3:    LCALL ZUFG
        SJMP  ZF4

ZTR2:   CJNE A,#67H,ZK12
        JBC F2,ZTER2
        SETB F2
ZTER2:  POP ACC
        POP ACC
        LJMP ZF1
ZK12:   CJNE A,#77H,ZT44
        RET
ZT44:   JB  F2,ZK62
        MOV DPTR,#1900H
        MOVC A,@A+DPTR
        RET
ZK62:   MOV DPTR,#1980H
        MOVC A,@A+DPTR
        RET
ZDTRS:  CJNE A,#77H,ZDTR29
        LCALL START1
ZDTR29: LCALL TR2
ZUFG:   PUSH A
        INC A
        MOV SRTM,A
        MOV A,SRPM
        CLR C
        SUBB A,SRTM
        JNC ZWS1
        MOV DPTR,#1A40H
        LCALL WR16
        SJMP ZWS2
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
DB      '  RECEIVE <R>    '
DB      '   SEND <S>      '
DB      'SELECT FUNCTION '
DB      ' FOR ALARM <T>  '
DB      ' ALARM MODE<L>  '
DB      'UPDATE TIME <U> '
DB      'UPDATE CHANNEL  '
DB      '     <C>        '
DB      'ADD FUNCTION <F>'
DB      'ADD ACCESSORY<A>'
ORG     1C00H
DB      '          J0  NNVZZ'
DB      '          E5  MMUEE'
DB      '          D4  L+PEE'
DB      '          C3  K-QEE'
DB      '          B2  F6REE'
DB      '          A1  G7SYY'
DB      '          EE  H8TXX'
DB      '          EE  I9OWW'
ORG     1C90H
DB      'DO YOU WANT TO  '
DB      '  CLEAR MEMORY  '
DB      ' COMPLETE ALARM '
DB      ' MEMORY FULL FOR'
DB      ' ALARM PRESS ESC'
DB      '   UPDATE TIME  '
DB      '   SELECT MENU  '
DB      ' SELECT FUCTION '
DB      'SELECT ACCESSORY'
ORG     1D90H
DB      '  PRESS N OR Y  '
DB      ' FREE CHANNEL   '
DB      '     READY      '
DB      'ENTER FOR RECIVE'
DB      ' WRITE FUNCTION '
DB      'WRITE ACCESSORY '
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กิตติกรรมประกาศ

ขอขอบคุณ อาจารย์เทอดศักดิ์ ที่ให้คำปรึกษาและแนะนำในการทำโครงงานครั้งนี้ ขอบพี่ วีรพงษ์ ที่ให้ยืมเครื่องเล่น CD เพื่อใช้ในการทดสอบ ขอขอบคุณเพื่อนๆ และน้องๆ ที่ให้คำแนะนำต่างๆ และขอขอบคุณทุกคนมีส่วนทำให้โครงงานสำเร็จลงด้วยดี

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้