ปีการศึกษา 2533

เครื่องส่งงานทางโทรศัพท์

โดย

นาย ก่อกิจ กิตติวัฒนาวงศ์

นาย วิบูลย์พร นิลนาม

อาจารย์ที่ปรึกษา

ผศ. นิกร สุขตมตันติ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เครื่องสั่งงานทางโทรศัพท์

นาย ก่อกิจ กิตติวัฒนาวงศ์

MR. KORKIJ KITTIWATTARNARWONG

นาย วิบูลย์พร นิลนาม

MR. WIBULPORN NILNAM

ปริญญานิพนธ์สำหรับปริญญาอุตสาหกรรมศาสตร์บัณฑิต

สาขาวิชาเทคโนโลยีอิเลคทรอนิคส์

คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้า เจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2533

เลขหมู่ T 33114 ก5
เลขทะเบียน 027947
วัน, ปี 18 ก.ค. 39

วปค.
ก5ค
2533

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ปริญญานิพนธ์ปีการศึกษา 2533

ภาควิชา เทคนิคอุตสาหกรรม

คณะ วิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้า เจ้าคุณทหารลาดกระบัง

เรื่อง เครื่องสั่งงานทางโทรศัพท์

ผู้จัดทำ

1.นาย ก่อกิจ กิตติวัฒนาวงศ์ 323301

2.นาย วิบูลย์พร นิลนาม 323328

..... อาจารย์ที่ปรึกษา

(ผศ. นิกร สุขุตมตันติ)

..... กรรมการ

( )

..... กรรมการ

( )

..... กรรมการ

( )

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ

บทคัดย่อ	I
ABSTRACT	II
บทที่ 1 บทนำ	1-1
1.1) วัตถุประสงค์ของโครงงาน	1-1
1.2) การนำไปใช้งาน	1-1
บทที่ 2 ความรู้ทั่วไปเกี่ยวกับ IC ที่ใช้งาน	2-1
2.1) คุณสมบัติของ MT 8870	2-1
- การนำ MT 8870 ไปใช้งาน	2-1
- โครงสร้างของ MT 8870	2-1
- ฟังก์ชั่นการทำงานภายในของ MT 8870	2-3
- ภาคกรองสัญญาณความถี่	2-3
- ภาคถอดรหัส	2-3
- ภาคตรวจสอบสัญญาณ	2-3
- ภาคขยายสัญญาณความแตกต่าง	2-5
- ภาคกำเนิดความถี่	2-5
- อธิบายขั้นตอนการทำงานของ MT 8870	2-6
2.2) คุณสมบัติของ IC T6668	2-6
- เกี่ยวกับ MEMORY	2-8
บทที่ 3 สถาปัตยกรรมของ 8048	3-1
3.1) โครงสร้างของ 8048	3-3
3.1.1) ส่วนประมวลทางคณิตศาสตร์	3-3
3.1.1.1) หน่วยคำนวณทางคณิตศาสตร์	3-3
3.1.1.2) แอคคิวมูเลเตอร์	3-3
3.1.1.3) แฟลกตัวทด	3-3
3.1.1.4) ตัวแปรคำสั่ง	3-3
3.1.2) หน่วยความจำ	3-3
3.1.2.1) หน่วยความจำโปรแกรม	3-3
3.1.2.2) หน่วยความจำข้อมูล	3-4

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.1.3) ส่วนติดต่ออุปกรณ์ภายนอก 3-4
3.1.3.1) พอร์ท 3-4
3.1.3.2) บัส 3-5
3.1.4) การตรวจสอบและสัญญาณอินเตอร์รัพท์ 3-6
3.1.5) ตัวนับโปรแกรม 3-6
3.1.6) คำแสดงสถานะของโปรแกรม 3-6
3.1.7) ลอจิกของเงื่อนไขในการกระโดด 3-7
3.1.8) อินเตอร์รัพท์ 3-7
3.1.9) ตัวตั้งเวลาและตัวนับ 3-9
3.1.9.1) ตัวนับ 3-9
3.1.9.2) ตัวตั้งเวลา 3-9
3.1.10) สัญญาณนาฬิกา และวงจรฐานเวลา 3-10
3.1.10.1) ตัวกำเนิดความถี่ 3-11
3.1.10.2) ตัวนับสถานะ 3-11
3.1.10.3) ตัวนับรอบ 3-11
3.1.11) การรีเซ็ท 3-11
3.1.12) การทำคำสั่งทีละหนึ่งคำสั่ง 3-12
บทที่ 4 หลักการทำงานของโครงงาน 4-1
- ส่วนควบคุม 4-1
- ส่วน BUFFER หรือ PORT INPUT และ PORT OUTPUT 4-1
- ส่วน DETECT สัญญาณ TONE 4-2
- ภาคขับ THYRISTER 4-2
- ส่วนอัดเสียง 4-2
- ภาค POWER SUPPLY 4-3
บทที่ 5 สรุปผลการทดลองและวิจารณ์ 5-1

กิตติกรรมประกาศ

บรรณานุกรม

ภาคผนวก รูปแสดงวงจรที่ใช้ในโครงงาน PROGRAM MONITOR และ DATA SHEET

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เครื่องสั่งงานทางโทรศัพท์

นาย ก่อกิจ กิตติวัฒนาวงศ์
นาย วิบูลย์พร นิลนาม

ผศ. นิกร สุขุตมตันติ อาจารย์ที่ปรึกษา

ปีการศึกษา 2533

บทคัดย่อ

ปริญญานิพนธ์ฉบับนี้แสดงถึงการประยุกต์ใช้งานของ IC 8048 MICROPROCESSOR 8 BIT โดยนำมาควบคุมระบบการทำงานของเครื่องสั่งงานทางโทรศัพท์ ซึ่ง IC เบอร์นี้เหมาะสมต่อการใช้งานในด้านนี้มาก เพราะมีคำสั่งที่สั้นกว่า และ สามารถทำงานได้รวดเร็ว สำหรับโครงงานชิ้นนี้ สามารถที่จะนำไปดัดแปลงหรือเพิ่มเติมได้ในหลายๆส่วนเพื่อให้เหมาะสม กับสภาพการใช้งานได้เป็นอย่างดี โดยอาศัย พื้นฐานการทำงาน หลักๆ อันเดียวกัน ซึ่งนับว่าเหมาะสมอย่างยิ่ง ต่อนักศึกษา ที่จะสามารถพัฒนาขีดความสามารถของโครงงาน ให้ดียิ่งขึ้นต่อไป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

REMOTE CONTROL VIA TELEPHONE LINE

KORKIJ KITTIWATTARNARWONG

WIBULPORN NILNAM

ASSISTANT PROFESSOR NIKORN SUKUTAMATANTI

1990

ABSTARCT

THIS THESIS SHOWS IC 8048 μ-PROCESSOR 8 BIT FOR APPLICATION IN CONTROLED SYSTEM OF THE DEVICE THAT CONTROLLED HOME USED BY TELEPHONE. THIS IC HAVE PROPERTY IN THIS PROJECT. THIS PROJECT CAN BE APPLIED OR GET SOME OPTION FOR PROPERTY IN USED BY SAME PRINCIPLE. THUS THIS PROJECT IS ABSOLUTE FOR STUDENT BECAUSE OF THEY CAN IMPROVE THIS PROJECT FOR BETTER.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 1

บทนำ

ในปัจจุบันนี้ สิ่งที่เห็นว่ามีความสำคัญไม่ด้อยไปกว่าสิ่งอื่นก็คือ เวลา เวลาในแต่ละวินาทีนั้น ถ้าเราเสียไปโดยเปล่าประโยชน์ จะทำให้เกิดความสิ้นเปลือง ไปโดยใช่เหตุ ดังนั้นในปัจจุบันจึงได้มีการประดิษฐ์คิดค้น เทคโนโลยีใหม่ ๆ ขึ้นมาเพื่อตอบสนองความต้องการของมนุษย์และส่วนใหญ่คำนึงถึง ความสะดวกสบาย และ การประหยัดเวลา ในโครงงานนี้ เราก็ได้คำนึงเวลาที่จะเสียไป ถ้าในกรณีเราต้องการทำอะไรซักอย่าง แล้วเราต้องสั่งให้คนทำ ซึ่งการทำงานของคนนั้นนับว่าช้ามากเมื่อเทียบกับ CPU ดังนั้นโครงงานนี้ จึงเน้นไปถึงความสะดวกสบายและการประหยัดเวลา โดยเราจะตัดการสั่งงานไปที่บุคคลออก แล้วให้เครื่องทำงานแทนซึ่งจะรวดเร็วกว่าคนมาก

1.1) วัตถุประสงค์ของโครงงาน

1. เพื่อศึกษาถึงคุณสมบัติต่างๆของ CPU 8048
2. ให้สามารถนำ CPU 8048 ไปประยุกต์ใช้งานต่างๆ ได้
3. เพื่อให้มีทักษะในการเขียนคำสั่งของ CPU 8048

1.2) การนำไปใช้งาน

เครื่องที่ทำนี้เราสามารถนำไปติดตั้งขนานกับเครื่องโทรศัพท์ที่เรามีอยู่ที่บ้านได้เลย แต่จะได้เฉพาะเครื่องโทรศัพท์ชนิด TONE เท่านั้น และการติดตั้งควรจะติดตั้งตั้งแต่เริ่มแรก เพราะเราจะได้วางสายไฟของระบบให้เข้ากับเครื่องที่สร้างนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 2

ความรู้ทั่วไป เกี่ยวกับ IC ที่ใช้ในโครงงาน

2.1) คุณสมบัติของ MT8870

- เป็นตัวรับและถอดรหัสความถี่ (DTMF reciver)
- กินไฟน้อย ใช้ไฟเลี้ยงระดับเดียวกับ TTL
- สามารถตั้งอัตราขยายภายในตัวไอซีได้
- สามารถปรับการ์ดไทม์ (guard time) ได้
- เป็นไอซีคุณภาพสูง

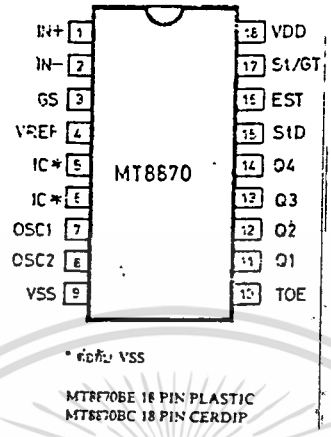
การนำ MT 8870 ไปใช้งาน

- นำไปใช้งานด้านรีโมตคอนโทรล
- เครื่องป้องกันโทรศัพท์ทางไกล
- ใช้ในงานเกี่ยวกันเครดิตการ์ด
- ใช้งานร่วมกับคอมพิวเตอร์
- ใช้ในเครื่องชุมสายขนาดย่อยหรือ PABX
- ใช้กับงานทางด้านโทรศัพท์ทั่วไป
- เครื่องกันขโมย
- การควบคุมอุปกรณ์ทางโทรศัพท์
- ใช้ทำเครื่องสอบถามทางโทรศัพท์

โครงสร้างของ MT8870

โครงสร้างภายในของ MT8870 ประกอบไปด้วยวงจรกรองความถี่ และ วงจรถอดรหัสฟังก์ชันทางดิจิตอล เป็นไอซีที่สร้างโดยใช้เทคโนโลยี ISO^2-CMOS ในส่วนของวงจรกรองความถี่ ใช้เทคนิคของ สวิตคาปาซิเดอร์ฟิลเตอร์ สำหรับกรองความถี่สูงและต่ำ ส่วนวงจรถอดรหัสใช้เทคนิคการนับทางดิจิตอล เพื่อตรวจจับและถอดรหัสทั้ง 16 ความถี่ออกเป็นเลขฐานสองขนาด 4 บิต และ เช็คช่วงเวลาที่สัญญาณเข้ามา ส่วนภาคอินพุตเป็นออปแอมป์ ซึ่งสามารถปรับอัตราขยายได้ โดยต่ออุปกรณ์ภายนอกเอาต์พุตเป็นวงจรแลตช์ 3 สถานะ รูปที่ 2.1แสดงขาของ MT8870 และรูปที่2.3 แสดงโครงสร้างภายในของ MT8870

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.1 แสดงลักษณะภายนอกของ MT8870

F_{LOW}	F_{HIGH}	NO	TOE	Q_4	Q_3	Q_2	Q_1
697	1209	1	H	0	0	0	1
697	1336	2	H	0	0	1	0
697	1477	3	H	0	0	1	1
770	1209	4	H	0	1	0	0
770	1336	5	H	0	1	0	1
770	1477	6	H	0	1	1	0
852	1209	7	H	0	1	1	1
852	1336	8	H	1	0	0	0
852	1477	9	H	1	0	0	1
941	1336	0	H	1	0	1	0
941	1209	*	H	1	0	1	1
941	1477	#	H	1	1	0	0
697	1633	A	H	1	1	0	1
770	1633	B	H	1	1	1	0
852	1633	C	H	1	1	1	1
941	1633	D	H	0	0	0	0
-	-	ANY	L	Z	Z	Z	Z

รูปที่ 2.2 แสดงรหัสซึ่งถอดได้ในแต่ละความถี่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ฟังก์ชั่นการทำงานภายใน MT8870

ภายใน MT8870 ประกอบด้วยส่วนสำคัญ 5 ส่วน คือ

1. ภาคกรองความถี่ (filter section)
2. ภาคถอดรหัส (decoder section)
3. ภาคขยายสัญญาณความแตกต่าง (differential input)
4. ภาคตรวจสอบสัญญาณ (steering circuit)
5. ภาคกำเนิดความถี่ (oscillator)

ภาคกรองสัญญาณความถี่

ในส่วนนี้จะแยกสัญญาณ DTMF ที่เข้ามาออกเป็น 2 กลุ่มความถี่ คือช่วงความถี่สูง และ ช่วงความถี่ต่ำ โดยใช้วงจรกรองแถบความถี่อันดับ 6 ชนิด สวิตช์คาปาซิเตอร์ (six-order switched capacitor band pass filter) ซึ่งความถี่ที่แยกได้มี 2 ช่วง คือ ช่วงความถี่สูงและช่วงความถี่ต่ำ

ภาคถอดรหัส

ความถี่ DTMF ที่ถูกกรองเรียบร้อยแล้ว จะผ่านเข้าวงจรถอดรหัสความถี่ออกเป็นตัวเลข โดยใช้เทคนิคการนับแบบดิจิตอล และมีการตรวจสอบความถี่ที่เข้ามาว่าเป็นความถี่มาตรฐาน DTMF หรือไม่ เพื่อป้องกันไม่ให้ความถี่อื่นเข้ามาผสมเมื่อตรวจสอบว่าความถี่นั้นถูกต้อง สัญญาณที่ขา EST (early steering) ก็จะแอคทีฟสำหรับค่าที่ถอดรหัสได้จากความถี่ต่าง ๆ นั้น แสดงในรูปที่ 2.2

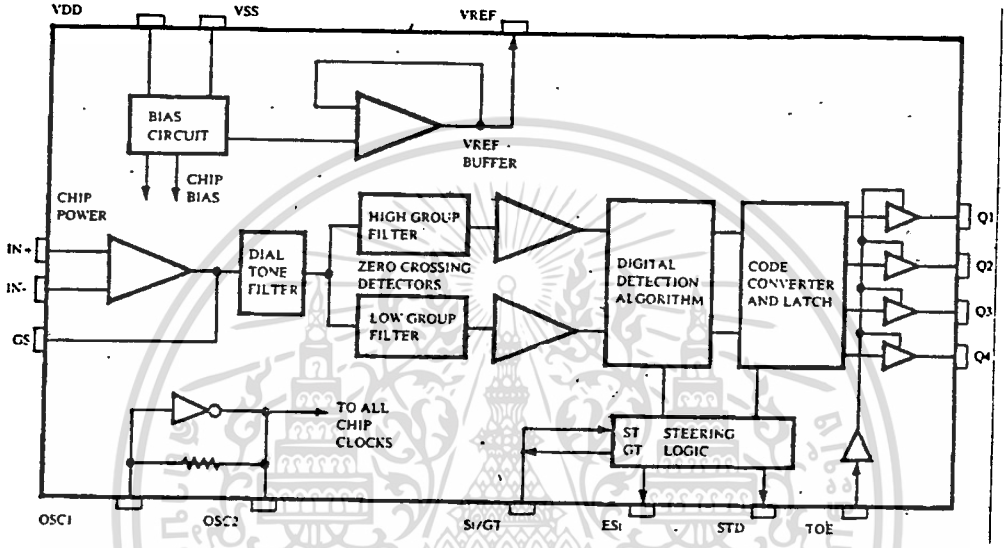
ภาคตรวจสอบสัญญาณ

ก่อนที่จะมีการถอดรหัสความถี่ออกไปที่เอาต์พุตจะมีการตรวจสอบช่วงความถี่ที่เข้ามาว่ามีระยะเวลาตามที่กำหนดหรือไม่ โดยสังเกตจากระยะเวลาการกดปุ่มโทรศัพท์ ซึ่งต้องกดปุ่มให้มีความถี่ออกมาเป็นช่วงเวลาพอสมควรมิฉะนั้นวงจรส่วนนี้จะไม่รับ โดยถือว่าสัญญาณนั้นไม่ถูกต้อง ส่วนช่วงเวลายาวเท่าใด สามารถตั้งได้โดยใช้ RC ต่อภายนอก สัญญาณที่ขา EST จะเป็น "high" นาน ใกล้เคียงกับระยะเวลาที่มีความถี่ DTMF เข้ามา จากรูปที่ 2.4เมื่อขา EST เป็น "high" ทำให้ V_c สูงขึ้นตัวเก็บประจุ C จะคายประจุ ทำให้แรงดัน V_c สูงขึ้นจนถึงค่าเทรชโอลด์ วงจรถอดรหัส จึงจะถอดรหัสออกเป็นตัวเลขขนาด 4 บิต รายละเอียดการทำงานขอ

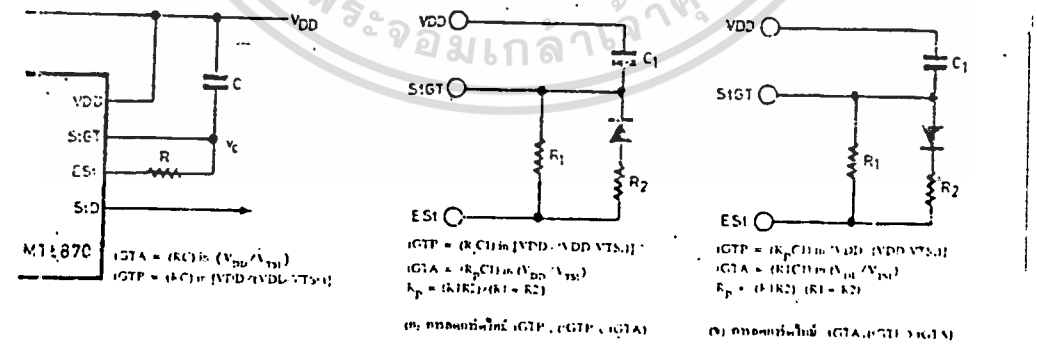
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ให้ดูจากแผนภูมิเวลาหรือไทมิ่งไดอะแกรม (timing diagram) ในรูปที่2.7จะเข้าใจได้ง่ายกว่า

สำหรับคำว่าการ์ดไทม์ (gard time) นั้น หมายถึง ช่วงคาบเวลาของความถี่ที่เข้ามา ซึ่งจะต้องนานเท่ากับหรือมากกว่าช่วงเวลาที่เราตั้งไว้ จึงจะได้รับการยอมรับว่าสัญญาณความถี่นั้นถูกต้องหรือพูดได้ว่าเวลาที่เราตั้งไว้โดย RC ก็คือการ์ดไทม์นั้นเอง เมื่อสัญญาณความถี่เข้ามานานเท่า หรือ มากกว่าเวลาที่ตั้งไว้ จึงจะสามารถแปลงเป็นตัวเลขออกไป การตั้งเวลาและคำนวณเวลาดูได้จากรูปที่2.4

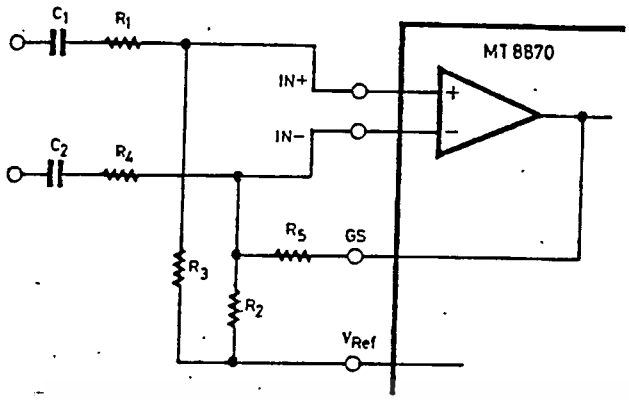


รูปที่ 2.3 แสดงโครงสร้างภายใน MT8870



รูปที่ 2.4 แสดงการต่อ และสูตรหาค่าเวลา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.5 แสดงการต่อภาคอินพุท และสูตรในการคำนวณ

ภาคขยายสัญญาณความแตกต่าง

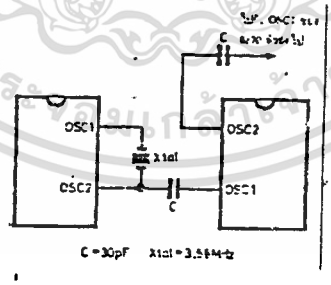
วงจรส่วนอินพุตของ MT8870 เป็นภาคขยายออปแอมป์ที่สามารถปรับอัตราขยายโดยต่อวงจรภายนอกเพิ่มเข้าไปรูปที่ 2.5 แสดงการต่อวงจรภายนอกเข้ากับอิตพุตซึ่งสามารถคำนวณอัตราขยายความแตกต่างของอินพุตและอิมพีแดนซ์ได้ดังนี้

อัตราขยาย (A_v diff) = R5/R1

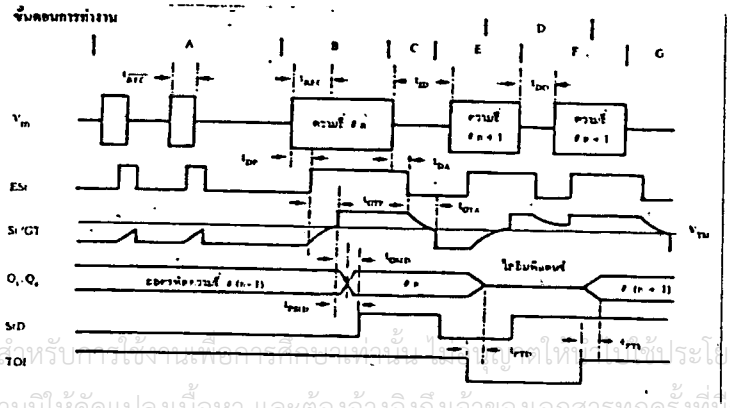
อินพุตอิมพีแดนซ์ (Zindiff) = $2\sqrt{R1^2 + (1/wc)^2}$

ภาคกำเนิดความถี่

ในภาคนี้ภายในไอซีจะมีวงจรเวลาอยู่ภายใน เพียงแต่ต่อแร่คริสตอลขนาด 3.58 MH_z ก็สามารถใช้งานได้ทันที การต่อวงจรกำเนิดความถี่แสดงในรูปที่ 2.6



รูปที่ 2.6 แสดงการต่อ X-TAL



รูปที่ 2.7 แสดงรูปสัญญาณตามขั้นตอนการทำงานต่าง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

อธิบายขั้นตอนการทำงาน

A - ตรวจพบความถี่เข้ามา แต่คาบเวลาไม่ถูกต้อง เอาต์พุตไม่เปลี่ยน

B - ความถี่ #n ถูกตรวจพบและมีคาบเวลาที่ถูกต้องความถี่ถูกถอดรหัสและแลตช์ไว้ที่เอาต์พุต

C - จบความถี่ #n ช่วงห่างถูกต้องเอาต์พุตยังคงแลตช์อยู่จนกว่าจะได้รับความถี่ที่ถูกต้องใหม่

D - เอาต์พุตเปลี่ยนเป็นไฮอิมพีแดนซ์

E - ความถี่ #n+1 ถูกตรวจพบ คาบเวลาถูกต้อง ความถี่ถูกถอดรหัสและแลตช์ไว้

F - ความถี่ #n+1 หายไป ช่วงห่างไม่ถูกต้อง เอาต์พุตยังคงแลตช์อยู่

G - จบความถี่ #n+1 ช่วงห่างถูกต้อง เอาต์พุตยังแลตช์อยู่จนถึงความถี่ใหม่ที่ถูกต้อง

อธิบายคำศัพท์

V_{in} - สัญญาณความถี่ DTMF ที่เข้ามา

EST - Early Steering output ใช้แสดงความถี่ที่ถูกต้อง

St/GT- Steering input / Guard Time output สำหรับต่อกับ RC ภายนอก

Q1-Q4- เอาต์พุต BCD ขนาด 4 บิต

STD - Delayed Steering output ใช้แสดงว่าความถี่ที่ได้รับหรือหายไป มีคาบเวลาตามที่กำหนด เพื่อแสดงความถูกต้องของสัญญาณ

TOE - Tone Output Enable (input) ใช้ควบคุม Q1-Q4 ให้เป็นไฮอิมพีแดนซ์

$t_{\overline{REC}}$ - คาบเวลานานสุดที่ตรวจพบความถี่ DTMF แล้วยังไม่ถูกต้อง

t_{REC} - คาบเวลาสั้นที่สุดที่ต้องการเพื่อแสดงว่าสัญญาณถูกต้อง

t_{ID} - เวลาสั้นที่สุดระหว่างสัญญาณ DTMF ที่ถูกต้อง 2 สัญญาณ

t_{DO} - เวลานานสุดที่ยอมให้สัญญาณหายไปได้ในคาบเวลาความถี่ที่ถูกต้อง

t_{DP} - เวลาที่ใช้ในการตรวจพบสัญญาณความถี่ DTMF ที่ถูกต้อง

t_{DA} - เวลาที่ใช้ในการตรวจการหายไปของสัญญาณความถี่ DTMF ที่ถูกต้อง

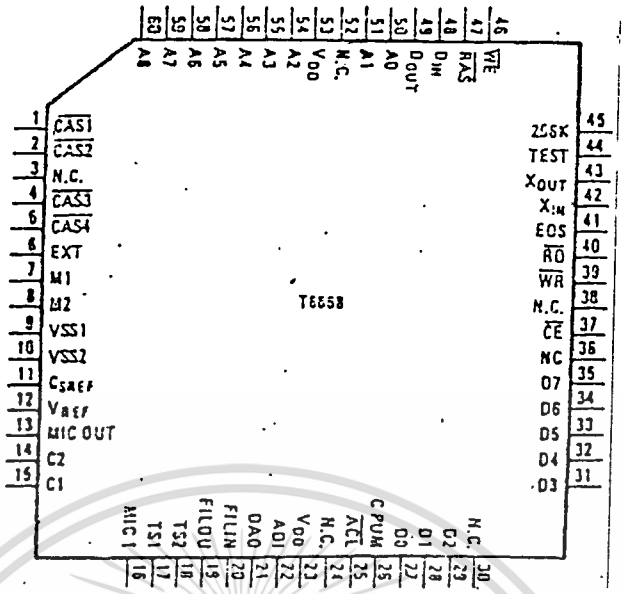
t_{GTP} - การ์ดไทม์ของการปรากฏความถี่ DTMF

t_{GTA} - การ์ดไทม์ของการหายไปของความถี่ DTMF

2.2) คุณสมบัติของ IC T6668

IC T6668 ถูกออกแบบขึ้นมาเพื่อใช้งานด้านวิเคราะห์เสียงดยเฉพาะ ซึ่งเป็นผลิตผลการประดิษฐ์ ของ บริษัทโตชิบาแห่งประเทศญี่ปุ่น ซึ่งเป็นไอซีชนิด CMOS LSI ลักษณะโครงสร้างภายนอกและตำแหน่งขาต่าง ๆ แสดงไว้ในรูปที่ 2.21

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.21 แสดงตำแหน่งขาต่างๆ ของT6668

การทำงาน IC จะทำการรับสัญญาณเสียงพูดเข้ามาจากนั้นทำการขยาย แล้วเปลี่ยนจากสัญญาณอนาล็อกไปเป็นข้อมูลดิจิตอล แล้วไปเก็บไว้ที่ไดนามิคแรม (DRAM) โดย CPU ภายในจะทำการเลื่อนแอดเดรส ที่จะนำเข้าไปเก็บเองโดยอัตโนมัติ เมื่อทำการแปลงข้อมูลจาก D/A จะใช้อัตรา 10 BIT D/A เพื่อเปลี่ยนกลับมา เป็นเสียงเช่นเดิม การอัดเข้าไป เราจะสามารถเลือก speed โดยเลือกที่ D6-D7 จากตารางที่ 2.21

KBPS	D7	D6
8	0	0
11	0	1
16	1	0
32	1	1

ตารางที่ 2.21 แสดงการเลือก SPEED ของการอัด

1. ถ้าเราเลื่อนสวิท D7-D6 ไปที่ 0,0 จะทำให้อัตราความเร็วของการแปลงข้อมูลเป็น 8K bit ต่อวินาที ทำให้อัดหรือเล่นได้นาน 128 วินาที

2. D7-D6 เป็น 0,1 จะทำให้อัตราการแปลงข้อมูลเป็น 11K bit ต่อวินาที ทำให้อัดหรือเล่นได้นาน 93 วินาที

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3. D7-D6 เป็น 1,0 ทำให้อัตราการแปลงข้อมูลเป็น 16K bit ทำให้อัดหรือเล่นได้นาน 64 วินาที

4. D7-D6 เป็น 1,1 ทำให้อัตราการแปลงข้อมูลเป็น 32K bit ทำให้อัดหรือเล่นได้นาน 32 วินาที

เกี่ยวกับ MEMORY

T6668 สามารถเลือกใช้ memory ได้ 2 ขนาดคือ 64K DRAM กับ 256K DRAM โดยการเลือกที่ขา 45 ของ IC (ที่เขียนไว้ว่า 256K) คือเมื่อเราจะต่อ DRAM 256K ให้กับ IC เราจะต้องเลือกต่อขา 45 กับไฟบวก และเมื่อเราจะต่อ DRAM 64K ให้กับ IC เราต้องต่อขา 45 กับกราวด์ T6668 ก็จะรู้ว่าเราใช้ MEMORY ขนาดเท่าใด

การเพิ่ม MEMORY ให้กับ IC T6668 สามารถกำหนดได้โดยการต่อขา M2 และ M1 ตามตาราง 2.22 นี้ คือ ถ้าเราต่อ M2,M1 ลงกราวด์ T6668 จะทำการเขียนหรืออ่านข้อมูลจาก 00000H ไปจนถึง 0FFFFH แล้วตัวมันเองก็จะเลิกการอ่าน หรือ การเขียนมารอการเริ่มต้นใหม่

ชนิดของ RAM	256K	M2	M1	ADDRESSที่หยุด
64K DRAM ตัวที่ 1	0	0	0	0FFFFH
64K DRAM ตัวที่ 2	0	0	1	1FFFFH
64K DRAM ตัวที่ 3	0	1	0	2FFFFH
64K DRAM ตัวที่ 4	0	1	1	3FFFFH
256K DRAM ตัวที่ 1	1	0	0	3FFFFH
256K DRAM ตัวที่ 2	1	0	1	7FFFFH
256K DRAM ตัวที่ 3	1	1	0	BFFFFH
256K DRAM ตัวที่ 4	1	1	1	FFFFFH

ตาราง 2.22 แสดงการกำหนด ขา M1,M2 เมื่อต่อ RAM ต่างชนิดกัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

D3	D2	D1	D0	ช่องที่
0	0	0	0	1
0	0	0	1	2
0	0	1	0	3
0	0	1	1	4
0	1	0	0	5
0	1	0	1	6
0	1	1	0	7
0	1	1	1	8
1	0	0	0	9
1	0	0	1	10
1	0	1	0	11
1	0	1	1	12
1	1	0	0	13
1	1	0	1	14
1	1	1	0	15
1	1	1	1	16

รูปแสดงการเลือก page โดยกำหนดที่ bit D0-D3

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 3

สถาปัตยกรรมของ 8048

วิวัฒนาการทางด้านสารกึ่งตัวนำได้พัฒนาจากยุค LSI เข้าสู่ VLSI (Very large ScaleMu Integrated circuit) ทำให้แต่เดิมมีเพียงไมโครโปรเซสเซอร์ที่สร้างสารกึ่งตัวนำก็กลับกลายมาเป็นไมโครคอมพิวเตอร์บนแผ่นชิปเดียว โดยรวมไมโครโปรเซสเซอร์ หน่วยความจำและอุปกรณ์อินเตอร์เฟสอินพุทเอาท์พุทเข้าด้วยกัน

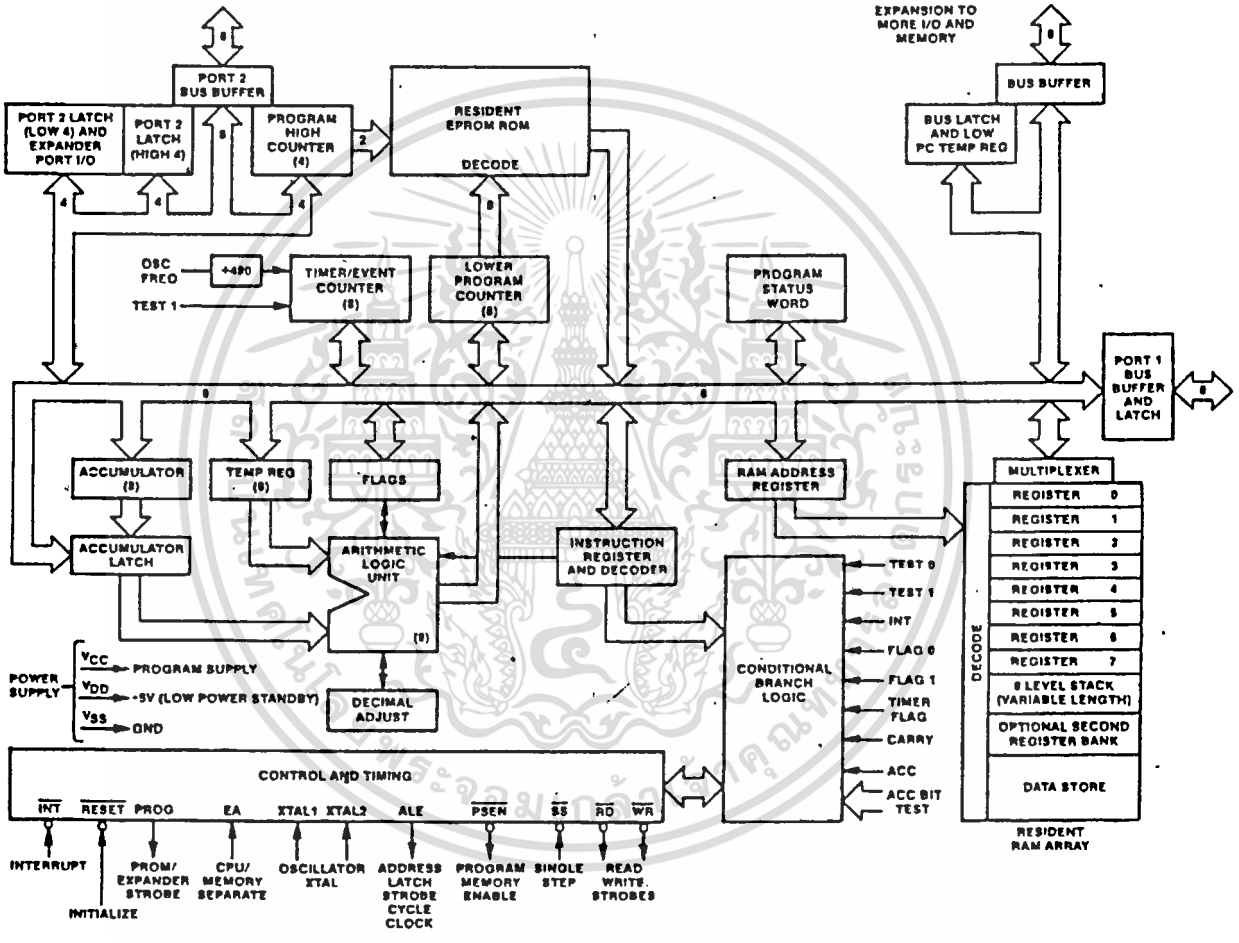
ในบรรดาไอซีไมโครคอมพิวเตอร์ทั้งหลาย ตระกูล 8048 เป็นไอซีที่มีแนวโน้มในการแพร่หลายและใช้งานกันมากที่สุด โดยมักจะพบเห็นในอุปกรณ์ของใช้ต่าง ๆ กันมากขึ้น เช่นในโทรศัพท์กดปุ่มที่มีหน่วยความจำ ในของเล่นเด็ก ในเครื่องดนตรี หรือออร์แกนอิเลคโทน ในเครื่องซักผ้าในเตาอบไมโครเวฟ ในจักรเย็บผ้า ฯลฯ

สืบเนื่องจากไอซีในตระกูล 8048 มีมากมายหลายเบอร์ แต่ละเบอร์มีโครงสร้างสถาปัตยกรรมในการรับรู้คำสั่งทางซอฟท์แวร์เหมือนกัน แต่จะแตกต่างกันทางฮาร์ดแวร์ ตารางที่ 1เป็นตารางแสดงรายละเอียดของไอซีตระกูล 8048

Device	Internal Memory		RAM Standby
8050AH	4K × 8 ROM	256 × 8 RAM	yes
8049AH	2K × 8 ROM	128 × 8 RAM	yes
8048AH	1K × 8 ROM	64 × 8 RAM	yes
8040AHL	none	256 × 8 RAM	yes
8039AHL	none	128 × 8 RAM	yes
8035AHL	none	64 × 8 RAM	yes

ตารางที่ 3.1 สรุปลักษณะของไอซีตระกูล 8048

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.1 แสดงรายละเอียดภายใน 8040

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.1 โครงสร้างของ 8048

โครงสร้างของ 8048 ก็เหมือนกับโครงสร้างของไมโครคอมพิวเตอร์ทั่วไป กล่าวคือประกอบด้วยบัสที่ทำหน้าที่เชื่อมต่อกับอุปกรณ์อื่น ๆ ดังรูปที่ 1.1 รายละเอียดของแต่ละส่วนเป็นดังนี้

3.1.1 ส่วนประมวลผลทางคณิตศาสตร์ (ARITHMEIIC SECTION)

ในส่วนประมวลผลทางคณิตศาสตร์ของ 8048 จะมีฟังก์ชั่นการทำงานพื้นฐาน ซึ่งสามารถแบ่งได้ดังนี้

หน่วยคำนวณคณิตศาสตร์และลอจิก (ARITHMETIC LOGIC UNIT = ALU)

แอคคิวมูเลเตอร์ (ACCUMULATOR) แฟลกตัวทด (CARRY FLAG)

ตัวแปลคำสั่ง (INSTRUCTION DECODER)

รายละเอียดของแตละส่วนเป็นดังนี้

3.1.1.1 หน่วยคำนวณคณิตศาสตร์ (ALU) ALU จะรับข้อมูลจากแหล่งข้อมูล 1 หรือ 2 แหล่งเพื่อไปทำการประมวลผลทางคณิตศาสตร์ภายใต้การควบคุมของตัวแปลคำสั่ง ALU มีหน้าที่ดังนี้

ADD WITH OR WITHOUT CARRY

AND, OR, EXCLUSIVE OR

INCREMENT/DECREMENT

BIT COMPLEMENT

SWAP NIBBLES

BCD DECIMAL ADJUST

หากการทำงานของ ALU ได้ผลลัพธ์มากกว่า 8 บิท แล้วแฟลกตัวทด (CARRY FLAG) ใน PROGRAM STATUS WORD จะถูกเซ็ท

3.1.1.2 แอคคิวมูเลเตอร์ เป็นรีจิสเตอร์ที่สำคัญที่สุดของ 8048 เพราะการทำงานทางคณิตศาสตร์ การติดต่ออุปกรณ์ภายนอก การติดต่อหน่วยความจำจะต้องผ่านแอคคิวมูเลเตอร์เสมอ

3.1.1.3 แฟลกตัวทด เป็นตัวบอกสภาวะการทำงานของ 8048 ว่าการทำงานเกินจำนวนบิทที่มีอยู่หรือไม่ ถ้าเกินแฟลกตัวทดนี้จะถูกเซ็ท

3.1.1.4 ตัวแปลคำสั่ง รหัสการทำงาน (OPERATION CODE = PO CODE) ของแต่ละคำสั่งจะเก็บอยู่ในตัวแปลคำสั่งนี้ จากนั้นจะถูกเปลี่ยนเป็นสัญญาณในการควบคุมตามหน้าที่ของคำสั่งต่อไป

3.1.2 หน่วยความจำ (MEMORY) หน่วยความจำของตระกูล 8048 นั้นแบ่งออกได้เป็น 2 ชนิด คือ

3.1.2.1 หน่วยความจำโปรแกรม (PROGRAM MEMORY) หน่วยความจำนี้จะใช้สำหรับเก็บชุดคำสั่งการทำงานของ 8048 ที่เขียนขึ้น หน่วยความจำสูงสุดของส่วนนี้คือ 4 KBYTES โดยส่วนหนึ่งจะอยู่ภายในชิปของไอซีและอีกส่วนหนึ่งเป็นการต่อเพิ่มภายนอก ในกรณีที่หน่วยความจำโปรแกรมภายในชิปจะมีขนาดความจุและชนิดต่างกันแล้วแต่ละเบอร์ในตระกูล 8048 เช่น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

8035 จะไม่มีหน่วยความจำส่วนนี้

8048 จะมีหน่วยความจำส่วนนี้เป็นรอม ขนาดความจุ 1 KBYTE

8049 จะมีหน่วยความจำส่วนนี้เป็นรอม ขนาดความจุ 2 KBYTE

8748 จะมีหน่วยความจำส่วนนี้เป็นอีพรอม ขนาดความจุ 1 KBYTE เป็นต้น

ตำแหน่งที่สำคัญในหน่วยความจำส่วนนี้คือ

ตำแหน่ง 0 (ADDRESS 000) : เมื่อถูก RESET 8048 จะทำการอ่านคำสั่ง แรกที่ตำแหน่งนี้

ตำแหน่งที่ 3 (ADDRESS 003): 8048 จะอ่านคำสั่งแรกที่ตำแหน่งนี้เมื่อมีการ INTERRUPT เข้ามา

ตำแหน่งที่ 7 (ADDRESS 007) : 8048 จะอ่านคำสั่งแรกที่ตำแหน่งนี้เมื่อมีการ TIMER/COUNTER INTERRUPT เข้ามา

ตำแหน่งทั้ง 3 ดังกล่าวโดยปกติแล้วจะเขียนด้วยคำสั่ง JUMP ไปยังโปรแกรมที่ได้เขียนไว้

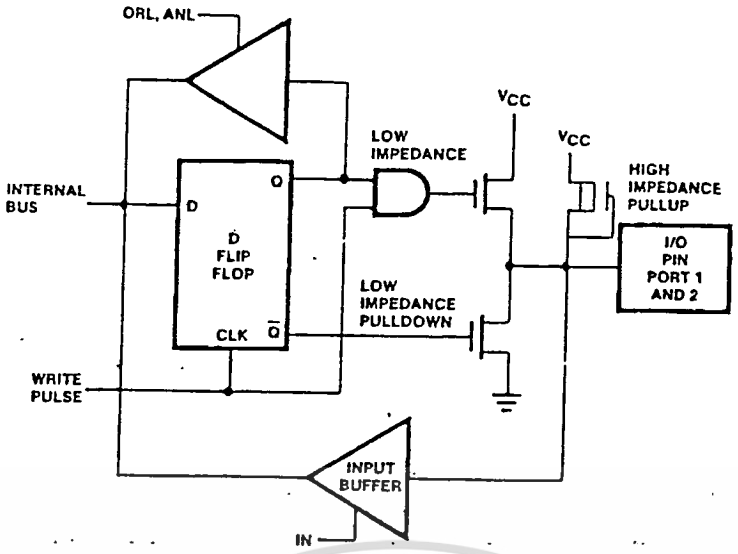
3.1.2.2 หน่วยความจำข้อมูล (DATA MEMORY) หน่วยความจำส่วนนี้จะเป็นชนิด RAM ซึ่งอยู่ภายในชิปของ 8048 โดยมีขนาด 64, 128 หรือ 256 BYTES ขึ้นอยู่กับเบอร์ของไอซีตระกูล 8048 ภายในส่วนของหน่วยความจำส่วนนี้จะทำหน้าที่เป็นรีจิสเตอร์ใช้งาน (WORKING REGISTER) 16 ตัว โดยแบ่งเป็น 2 ชุด ๆ ละ 8 ตัว ซึ่งชุดแรกอยู่ที่ตำแหน่ง 0-7 และชุดที่สองอยู่ที่ตำแหน่ง 24-31 ในขณะที่ 8048 ทำงานอยู่นั้นรีจิสเตอร์จะถูกใช้งานเพียงชุดเดียวโดยสามารถเลือกชุดของรีจิสเตอร์จากคำสั่ง REGISTER BANK AWITCH (SEL RB) นอกจากทำหน้าที่เป็นรีจิสเตอร์แล้วที่หน่วยความจำข้อมูลตำแหน่ง 8-23 จะใช้เป็นเนื้อที่ของ STACK และตั้งแต่ตำแหน่ง 32 ขึ้นไปผู้ใช้สามารถใช้เป็นส่วนเก็บข้อมูลชั่วคราวเช่นเดียวกับ RAM ทั่วไปด้วย

ในกรณีที่ผู้ใช้ต้องการใช้เนื้อที่ RAM มากกว่าที่มีอยู่ในตัว 8048 ก็สามารถเพิ่ม RAM ภายนอกได้อีก 256 BYTES โดยใช้คำสั่ง MOVX A, R หรือ MOVX R, A เมื่อ R เป็นรีจิสเตอร์ 0 หรือรีจิสเตอร์ 1 ทำหน้าที่เป็นตัวชี้ (POINTER)ในการติดต่อกับ RAM ที่เพิ่มขึ้นหน่วยความจำส่วนนี้โดยปกติจะใช้สำหรับเก็บข้อมูลชั่วคราว เมื่อมีการอ่านเข้าไป 8048 จะมองเห็นเช่นข้อมูลซึ่งผิดกับข้อมูลที่อยู่ในส่วนของหน่วยความจำโปรแกรมที่กล่าวมาแล้ว

3.1.3 ส่วนติดต่ออุปกรณ์ภายนอก (INPUT/OUTPUT) ในส่วนการติดต่ออุปกรณ์ภายนอกของตระกูล 8048 สามารถแบ่งออกได้เป็น 2 ลักษณะคือทำหน้าที่เป็น พอร์ท กับบัส โดยจะแยกกล่าวรายละเอียดดังต่อไปนี้

3.1.3.1 พอร์ท (PORT) ภายในตัวของไอซีตระกูล 8048 จะมีพอร์ทด้วย เป็นสองพอร์ทที่มีขนาดพอร์ท 8 บิท พอร์ททั้งสองเป็นพอร์ทชนิด OUAST-BIDIRECTIONAL ซึ่งแต่ละขาของพอร์ทสามารถเป็นได้ทั้งอินพุท เอาท์พุทได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.2 แสดงวงจรภายในของพอร์ท 1 และพอร์ท 2 แบบ QVASI BIDIRECTIONAL

จากรูปที่ 1.2 แสดงวงจรภายในของแต่ละขาของพอร์ททั้งสอง จะเห็นว่าความต้านทาน 50 K จะทำหน้าที่ในการยกระดับโวลเตจและกระแสให้เพียงพอกับระดับลอจิกของทีทีแอล. จากวงจรจะเห็นว่าในกรณีของการอินพุทนั้นจะสามารถรับการเปลี่ยนระดับลอจิกจากระดับสูงสู่ระดับต่ำเท่านั้น ดังนั้นในการใช้ขาใดขาหนึ่งของพอร์ทเป็นอินพุทสัญญาณนั้นต้องทำการเซ็ทให้ขานั้นอยู่ในระดับลอจิกหนึ่งเสมอ

ในกรณีที่ใช้หน่วยความจำโปรแกรมภายนอก 4 บิทแรกของพอร์ท 2 จะทำหน้าที่ในการให้ ADDRESS 4 บิทสูงของหน่วยความจำโปรแกรมเพื่อทำการอ่านคำสั่งการทำงานอีกด้วยในกรณีนี้ต้องการใช้ 4 บิทแรกของพอร์ท 2 ทำหน้าที่ติดต่ออุปกรณ์ภายนอกอีก จะเกิดปัญหาที่ว่า 8048 ใช้พอร์ท 2 เป็นตัวชี้ ADDRESS ของ 4 บิท สุดท้ายอยู่แล้ว สามารถแก้ปัญหานี้ได้โดยอาศัยการแลทช์เข้าช่วย เพราะสิ่งที่ออกมาที่ 4 บิทแรกของพอร์ท 2 จะเป็นลักษณะ MULTIPLEX ระหว่างข้อมูลที่จะติดต่อกับอุปกรณ์ภายนอกกับ ADDRESS ที่ส่งออกมา โดยรูปแบบของการติดต่อภายนอกจะออกมาขณะช่วงขึ้นของสัญญาณ ALE (สัญญาณนี้จะอธิบายภายหลัง) แต่รูปแบบ ADDRESS จะปรากฏเมื่อช่วงลงของสัญญาณ ALE ดังนั้นข้อมูลสำหรับติดต่ออุปกรณ์ภายนอกกับ ADDRESS จึงแยกออกมาได้

3.1.3.2 บัส (BUS) ในส่วนนี้จะทำหน้าที่สองอย่างขึ้นอยู่กับกรณีที่ใช้คือในกรณีที่ใช้หน่วยความจำโปรแกรมภายใน บัสจะทำหน้าที่เช่นเดียวกับพอร์ททั่วไปชนิด BIDIRECTIONAL PORT ซึ่งจะต่างกับพอร์ท 1 และพอร์ท 2 ที่กล่าวมาแล้ว แต่ในกรณีที่ใช้หน่วยความจำโปรแกรมภายนอก โดยขณะที่ทำการอ่านคำสั่งจะให้ ADDRESS และอ่านคำสั่งเข้าที่บัสนี้ ด้วยวิธี MULTIPLEX เช่นเดียวกับพอร์ท 2 ที่กล่าวมาแล้ว ในการ MULTIPLEX นั้น 8048 จะให้ ADDRESS ออกมาก่อนแล้วจึงทำการอ่านคำสั่งที่ได้จากหน่วยความจำโปรแกรมภายนอก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.1.4 การตรวจสอบและสัญญาณอินเตอร์รัพ (TEST AND INT INPUT)

ในตระกูล 8048 ได้จัดขาไว้ 3 ขา สำหรับเป็นอินพุท และไว้ใช้ทดสอบเงื่อนไขเพื่อทำงานร่วมกับคำสั่ง JUMP ซึ่งได้แก่ TO, T_1, INT หลังจากทำการทดสอบเงื่อนไขของสัญญาณทั้งสามแล้ว การทำงานของ 8048 สามารถข้ามไปทำตามคำสั่งในโปรแกรมย่อยที่เขียนไว้เพื่อทำเมื่อเงื่อนไขถูกต้อง โดยไม่จำเป็นต้องผ่านแอคคิวมูเลเตอร์ นอกจากนี้แล้วสัญญาณทั้งสามนี้ยังสามารถทำหน้าที่อย่างอื่นได้อีก ซึ่งจะอธิบายในส่วนถัดไป

3.1.5 ตัวนับโปรแกรม (PROGRAM COUNTER)

ตัวนับโปรแกรมเป็นตัวนับอิสระขนาด 12 บิท ในการทำงานปกติจะเพิ่มทีละหนึ่งเพื่อใช้สำหรับในการชี้ ADDRESS ของหน่วยความจำโปรแกรมเพื่ออ่านคำสั่งในการทำงาน สำหรับบิท 11 ของตัวนับโปรแกรมจะไม่เปลี่ยนไปตามการเพิ่มของตัวนับโปรแกรม สำหรับบิทนี้สามารถเปลี่ยนได้โดยคำสั่ง SELECT MEMORY BANK เป็นการเลือกชุดของหน่วยความจำโปรแกรม ส่วน 2 KBYTE แรกหรือส่วน 2 KBYTE หลัง ในกรณีที่ใช้หน่วยความจำโปรแกรมภายนอก BIT 0-7 ของตัวนับโปรแกรมนี้จะออกปรากฏที่บัสเมื่อสัญญาณ ALE เริ่มตกลงส่วน BIT 8-11 ของตัวนับโปรแกรมจะถูกส่งออกทาง 4 BIT ล่างของพอร์ท 2 เพื่อใช้สำหรับเป็นตัวชี้ ADDRESS ส่วนของหน่วยความจำโปรแกรม

3.1.6 คำแสดงสถานะของโปรแกรม (PROGRAM STATUS WORD)

คำแสดงสถานะของโปรแกรม มีขนาด 8 บิท สามารถอ่านและเขียนผ่านแอคคิวมูเลเตอร์ได้เพื่อดูหรือเซ็ทสถานะเริ่มต้นก่อนทำตามชุดคำสั่งที่ได้เขียนไว้คำแสดงสถานะของโปรแกรมเป็นรีจิส - เตอร์ที่เก็บตัวชี้สแตค และสถานะต่าง ๆ ของโปรแกรมที่ทำงานอยู่ การที่สามารถเขียนหรืออ่านได้ของคำแสดงสถานะของโปรแกรม ทำให้สามาถเก็บสถานะต่าง ๆหลังเกิดมีปัญหาเรื่องไฟเลี้ยงขึ้นได้

คำแสดงสถานะของโปรแกรมจะมีส่วนประกอบดังนี้

บิท 0-2 = เป็นตัวรีสแตค (STACK POINTER)

บิท 3 = เป็นบิทที่ไม่ได้ใช้งาน โดยปกติเมื่อมีการอ่านเข้ามาจะมีค่าเป็น 1 เสมอ

บิท 4 = เป็นบิทใช้สำหรับเลือกชุดรีจิสเตอร์ ดังนี้ 0=รีจิสเตอร์ชุดที่ 1 (ADDRESS 0-7) 1=รีจิสเตอร์ชุดที่ 2 (ADDRESS 24-31)

บิท 5 = เป็นแฟลกศูนย์ (F) สามารถเซ็ทหรือรีเซ็ทได้ตามสถานะของคำสั่ง

บิท 6 = AUXILIARY CARRY (AC) เป็นตัวทดจากบิท 3 ไปบิท 4 เมื่อใช้คำสั่ง ADD ใช้สำหรับเป็นเงื่อนไขในการปรับค่าเป็นเลขฐานสิบ (DECTMAL ADJUST INSTRUCTION)

บิท 7 = บิทตัวทด (CARRY FLAG) เป็นบิทที่แสดงสถานะเป็นตัวทดหรือกรณีที่ผลลัพธ์ที่ได้มากกว่า 8 บิท

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

คำแสดงสถานะของโปรแกรม บิท 4-7 จะถูกเก็บในสแตคเมื่อมีการเรียกโปรแกรมย่อมทำงาน และสามารถนำค่าแสดงสถานะของโปรแกรมกลับมาได้เมื่อใช้คำสั่ง RETR. แต่ถ้าใช้คำสั่ง RET จะเป็นการกลับจากโปรแกรมย่อยที่ไม่เก็บค่าแสดงสถานะของโปรแกรมเดิมไว้

3.1.7 ลอจิกของเงื่อนไขในการกระโดด (CONDITIONAL BRANCH LOGIC)

เงื่อนไขต่าง ๆ ที่เกิดขึ้นภายในและภายนอกของ 8048 สามารถที่จะทำการทดสอบเพื่อไปยังชุดของโปรแกรมที่ได้เขียนไว้เมื่อเงื่อนไขถูกต้องโดยใช้คำสั่งกระโดดแบบมีเงื่อนไข (CONDITIONAL JUMP INSTRUCTION) เงื่อนไขต่าง ๆ ที่ใช้สำหรับการกระโดดแบบมีเงื่อนไขดังตารางที่ 2

Device Testable	Jump Conditions (Jump On)	
	All zeros	not all zeros
Accumulator	—	1
Accumulator Bit	0	1
Carry Flag	—	1
User Flags (F0, F1)	—	1
Timer Overflow Flag	0	1
Test Inputs (T0, T1)	0	—
Interrupt Input ($\overline{INT}$)		

ตารางที่ 2.แสดงเงื่อนไขสำหรับใช้ในการกระโดด

3.1.8 อินเตอร์รัพท์ (INTERRUPT)

8048 จะรับการอินเตอร์รัพท์ต่อเมื่อได้ทำการอีนาเบิ้ลการอินเตอร์รัพท์แล้วด้วยคำสั่งอีนาเบิ้ล 8048 จะรับการอินเตอร์รัพท์ที่สถานะลอจิกเป็นศูนย์ ในกรณี ดิสเอเบิ้ลการอินเตอร์รัพท์ที่ขาอินเตอร์รัพท์นี้ยังใช้เป็นเงื่อนไขในการกระโดดได้ด้วย

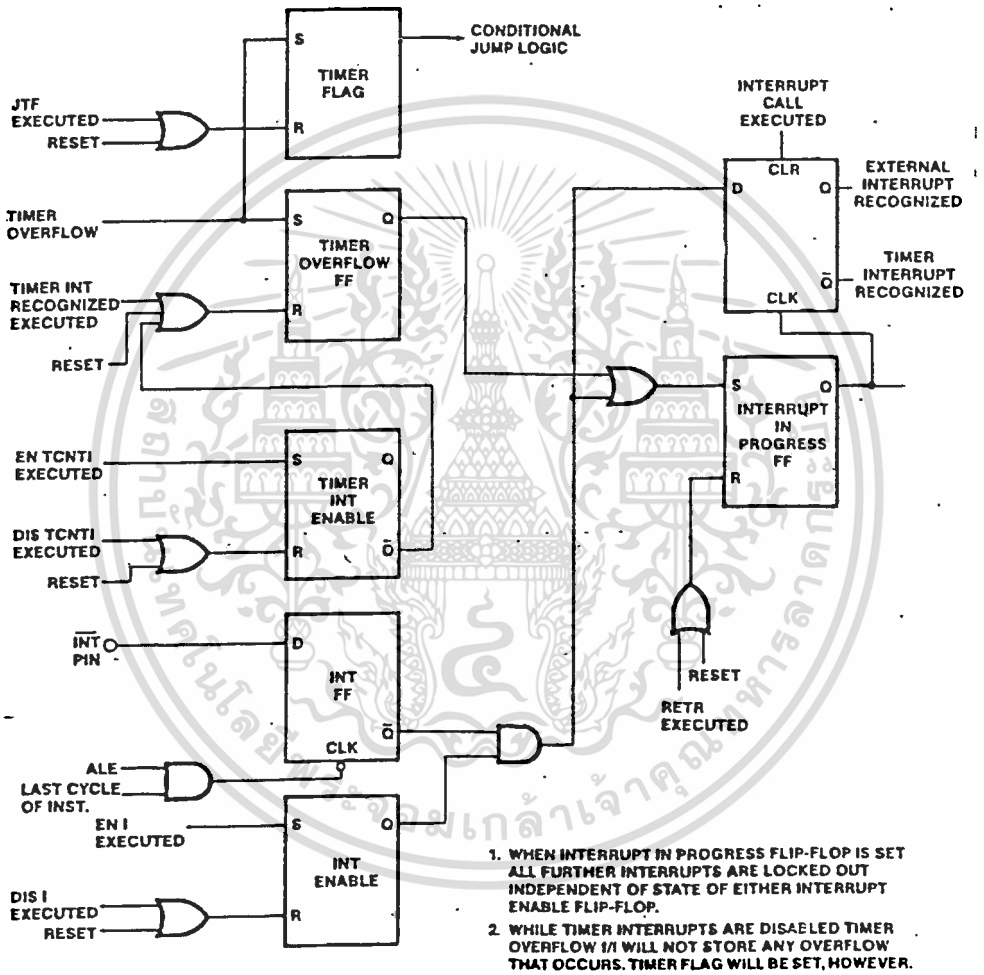
ระหว่างที่มีการอินเตอร์รัพท์ตัวนับโปรแกรมและบิท 4-7 ของคำแสดงสถานะของโปรแกรมจะถูกเก็บไว้ที่สแตค และการควบคุมการทำงานต่าง ๆเมื่อมีการอินเตอร์รัพท์นั้น จะถูกส่งผ่านไปทำงานที่ตำแหน่ง 003 ของหน่วยความจำโปรแกรมซึ่งโดยปกติจะเขียนด้วยคำสั่งกระโดดแบบไม่มีเงื่อนไข (UNCONDITIONAL JUMP) ไปยังโปรแกรมย่อยที่ใช้ตอบสนองการอินเตอร์รัพท์ การกลับหลังจากโปรแกรมย่อยของการอินเตอร์รัพท์จะกลับโดยคำสั่ง RETR. ซึ่งจะยังคงรักษาคำแสดงสถานะของโปรแกรมไว้ ขณะเดียวกันก็อีนาเบิ้ลการอินเตอร์รัพท์ใหม่ด้วย การอินเตอร์รัพท์ของ 8048 เป็นการอินเตอร์รัพท์ระดับเดียว ดังนั้นจะต้องหวังการอีนาเบิ้ลทุกครั้งหลังจากการอินเตอร์รัพท์ด้วยคำสั่ง RETR เสมอการอินเตอร์รัพท์ที่เกิดจากตัวตั้งเวลาและตัวนับ (TIMER/COUNTER) ก็เช่นเดียวกันการอินเตอร์รัพท์ภายนอก แต่จะต่างตรงที่ว่าการอินเตอร์รัพท์ที่เกิดจากตัวตั้งเวลาและตัวนับนั้นจะไปยังตำแหน่ง 007 ซึ่งเป็นตำแหน่งที่โปรแกรมย่อยการอินเตอร์รัพท์ของตัวตั้งเวลาและตัวนับอยู่การอินเตอร์รัพท์จากภายนอกจะมีลำดับสูงกว่าการอินเตอร์รัพท์จากตัวตั้งเวลาและตัวนับหมายความว่าถ้ามี

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การอินเตอร์รัพท์จากภายนอกพร้อมกับการอินเตอร์รัพท์จากตัวตั้งเวลาแล้ว 8048 จะตอบสนองการอินเตอร์รัพท์จากภายนอกก่อนเสมอ

เมื่อต้องการดีสเอเบิ้ลการอินเตอร์รัพท์ สามารถทำได้ 2 วิธีคือ การรีเซ็ท 8048 กับการใช้คำสั่งในการอินเตอร์รัพท์ภายนอกกับตัวตั้งเวลาและตัวนับ 8048 จะตรวจสอบการอินเตอร์รัพท์ทุก ๆ วงจรการทำงาน (MACHINE CYCLE) ของ ALE

ในรูปที่ 3 แสดงถึงวงจรลอจิกภายในของ 8048 ที่เกี่ยวกับการอินเตอร์รัพท์



รูปที่ 3.3 แสดงวงจรลอจิกของการอินเตอร์รัพ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.1.9 ตัวตั้งเวลาและตัวนับ (TIMER/COUNTER)

ภายในของ 8048 จะประกอบด้วยตัวตั้งเวลาและตัวนับ การทำงานของทั้งสองจะคล้ายกัน เพียงแต่ว่าตัวนับจะมีการอินพุทเท่านั้น

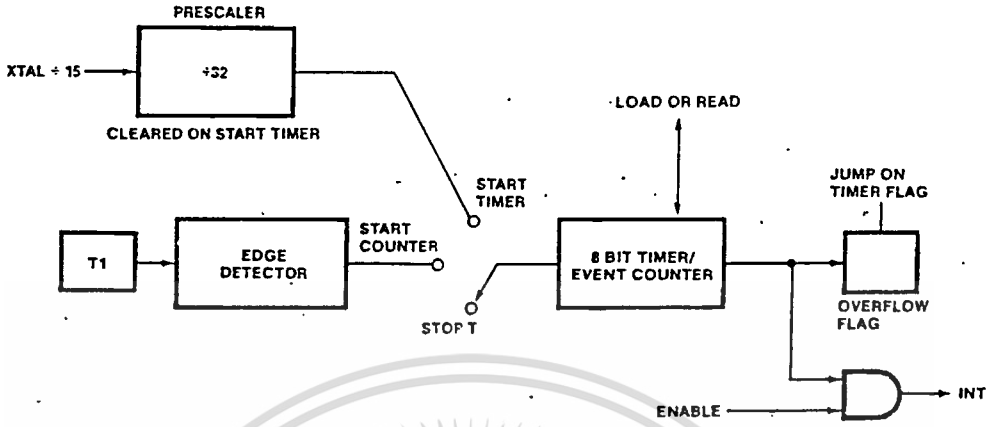
3.1.9.1 ตัวนับ เป็นตัวนับขึ้นแบบเลขฐานสองขนาด 8 บิท ที่สามารถกำหนดหรืออ่านค่าได้ด้วยคำสั่ง MOV. ที่ส่งผ่านทางแอคคิวมูเลเตอร์ การรีเซ็ทจะไม่มีผลต่อค่าที่อยู่ในตัวนับ ตัวนับจะหยุดก็ต่อเมื่อมีการรีเซ็ทหรือใช้คำสั่ง STOP TCNC และจะหยุดก็ต่อเมื่อมีการรีเซ็ทจนกระทั่งใช้คำ START T หรือ START CNT อีกครั้ง ทันทีที่ตัวนับเริ่มนับจะเพิ่มค่าตัวเองทีละ 1 จนกระทั่งถึงค่าสูงสุด FFH แล้วเริ่มนับที่ 00 ใหม่ การนับจะเป็นเช่นนี้เรื่อยไปจนกว่าจะมีคำสั่ง STOP TCNT หรือการรีเซ็ท

การเพิ่มค่าจากค่าสูงสุดไปยังค่าศูนย์ (OVERFLOW) จะมีผลไปเซ็ท OVERFLOW FLAG FLIP-FLOP และสร้างสัญญาณอินเตอร์รัพท์ขึ้น สถานะของ OVERFLOW FLAG สามารถตรวจสอบได้โดยคำสั่งกระโดดแบบมีเงื่อนไข (JIF) FLAG จะถูกรีเซ็ทก็ต่อเมื่อมีการทำงานของคำสั่ง JIF หรือการรีเซ็ท 8048 สัญญาณอินเตอร์รัพท์จะถูกแลทช์และ OR กับการอินเตอร์รัพท์ภายนอก การอินเตอร์รัพท์ของตัวนับสามารถอีนาเบิ้ลหรือดีสเอเบิ้ลด้วยคำสั่ง EN TCNTI และ DIS TCNTI หากมีการอีนาเบิ้ล เมื่อตัวนับเกิด OVERFLOW จะไปทำงานที่ตำแหน่ง 007 ทันทีตัวนับสามารถทำหน้าที่เป็นตัวทำหน้าที่นับเหตุการณ์ (EVENT COUNTER) ได้โดยที่ขา T1 จะทำหน้าที่เป็นอินพุทเมื่อมีการเปลี่ยนสถานะจาก "1" ไป "0" ตัวนับจะเพิ่มค่าขึ้นหนึ่งเสมอ ความเร็วของการนับสูงสุดคือ 3 รอบคำสั่ง (ทุก ๆ 7.5 เมื่อใช้คลิสตรอล 6 เมกกะเฮิทซ์) และอินพุทเข้า T1 จะต้องมีระดับลอจิกเป็น 1 อย่างน้อย 500 ns หลังจากการเปลี่ยนระดับในแต่ละครั้ง

3.1.9.2 ตัวตั้งเวลา เมื่อใช้คำสั่ง START T ตัวตั้งเวลาและตัวนับจะทำหน้าที่เป็นตัวตั้งเวลา ในการใช้คำสั่ง START T นั้นจะทำให้ฐานเวลาภายในหารด้วย 15 และ 32 ตามลำดับจากผลที่ได้ตั้งไว้ก่อนแล้ว เมื่อเกินค่าที่กำหนดจะทำการอินเตอร์รัพท์ของตัวตั้งเวลาและตัวนับ เพื่อไปยังโปรแกรมที่ต้องการ ตัวอย่าง เช่น

เมื่อใช้คริสตอลขนาด 11 เมกกะเฮิทซ์เป็นฐานเวลาให้ 8048 เมื่อผ่านตัวหาร 15 จะได้ความถี่ 733 กิโลเฮิทซ์ และความถี่นี้จะถูกหารด้วย 32 อีกจะได้ 22,917 เฮิทซ์ ซึ่งแสดงว่าตัวตั้งเวลาจะเพิ่มขึ้นทุก ๆ 44 ไมโครวินาที รูปที่ 1.4 เป็นการแสดงให้เห็นถึงส่วนประกอบการทำงานของตัวตั้งเวลาและตัวนับภายใน 8048

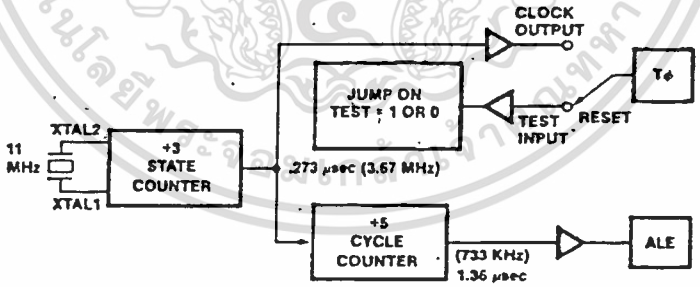
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 1.4 แสดงส่วนของตัวตั้งเวลาและตัวนับของ 8048

3.1.10 สัญญาณนาฬิกาและวงจรฐานเวลา (CLOCK AND TIMING CIRCUITS)

แหล่งกำเนิดฐานเวลาสำหรับ 8048 สามารถสร้างได้จากอุปกรณ์ภายนอก เช่น คริสตอลอินดัคเตอร์ หรือแหล่งกำเนิดสัญญาณนาฬิกาภายนอก การทำงานของส่วนสัญญาณนาฬิกาและวงจรฐานเวลาแสดงในรูปที่ 1.5 โดยแยกกล่าวเป็นส่วน ๆ ดังนี้



รูปที่ 3.5 แสดงโครงสร้างของส่วนสัญญาณนาฬิกาและวงจรฐานเวลา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

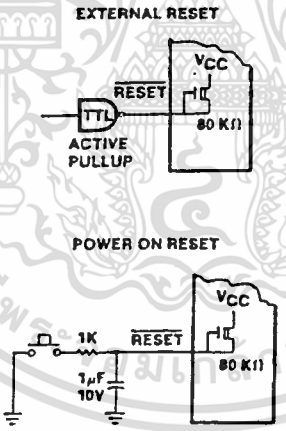
3.1.10.1 ตัวกำเนิดความถี่ (OSCILLATOR) ตัวกำเนิดความถี่เป็นวงจรแบบ SERIES RESONANT ที่มีอัตราขยายสูงโดยมีช่วงความถี่ระหว่าง 1 ถึง 6 เมกกะเฮิทซ์ ซึ่งขา X1 เป็นขาอินพุทเข้าสู่วงจรขยาย ขณะที่ขา X2 เป็นขาเอาท์พุทในการต่อคริสตอลหรืออินดัคเตอร์ระหว่างขา X1 และ X2 จะทำให้เกิดการป้อนกลับและการเลื่อนเฟสสำหรับการกำเนิดความถี่ในกรณีที่ไม่ต้องการใช้ความถี่ที่แน่นอนสามารถใช้อุปกรณ์พวกอินดัคเตอร์แทนคริสตอลได้ซึ่งจะให้ความถี่อยู่ในช่วง 3 ถึง 5 เมกกะเฮิทซ์ได้

3.1.10.2 ตัวนับสถานะ (STATE COUNTER) เอาท์พุทของตัวกำเนิดความถี่จะถูกหารด้วย 3 ในตัวนับสถานะ เพื่อสร้างสัญญาณนาฬิกาสำหรับการทำงานของ 8048 สัญญาณนาฬิกานี้สามารถส่งออกไปยังขา TO ได้โดยใช้คำสั่ง ENTO CLK สัญญาณนาฬิกาที่ขา TO จะหยุดเมื่อ 8048 ถูกรีเซ็ท

3.1.10.3 ตัวนับรอบ (CYCLE COUNTER) สัญญาณนาฬิกาจะถูกหารด้วย 5 ในตัวนับรอบเพื่อสร้างสัญญาณนาฬิกาที่ประกอบรอบการทำงาน 5 สถานะ สัญญาณนาฬิกานี้เรียกว่า ADDRESS LATCH ENABLE (ALE) เพื่อใช้เป็นสัญญาณติดต่อกับหน่วยความจำโปรแกรมภายนอกสัญญาณนี้จะออกมาอย่างต่อเนื่องตลอดเวลาขณะที่ 8048 ยังทำงานอยู่

3.1.11 การรีเซ็ท (RESET)

การรีเซ็ทจะทำให้ 8048 เริ่มการทำงานใหม่เมื่อขา RESET มีลอจิกศูนย์ซึ่งขานี้จะทำงานแบบ SCHMITT-TRIGGER โดยจะมีความต้านทานที่ใช้อยู่ภายใน 8048 และภายนอกจะต่อตัวเก็บประจุค่า1 ไมโครฟาร์ดอยู่เพื่อทำให้เกิดช่วงเวลาเพียงพอในการรีเซ็ท การรีเซ็ท 8048 สามารถทำได้ดังแสดงในรูปที่ 1.6



รูปที่ 3.6 แสดงการรีเซ็ท 8048

หากมีสัญญาณรีเซ็ทจากภายนอกที่ขา RESET จะต้องมีค่าลอจิกศูนย์อย่างน้อย 50 มิลลิวินาที หลังจากที่แหล่งจ่ายไฟอยู่ในสภาพที่พร้อมจะใช้งานแล้ว

เมื่อมีการรีเซ็ทให้ 8048 จะทำให้เกิด

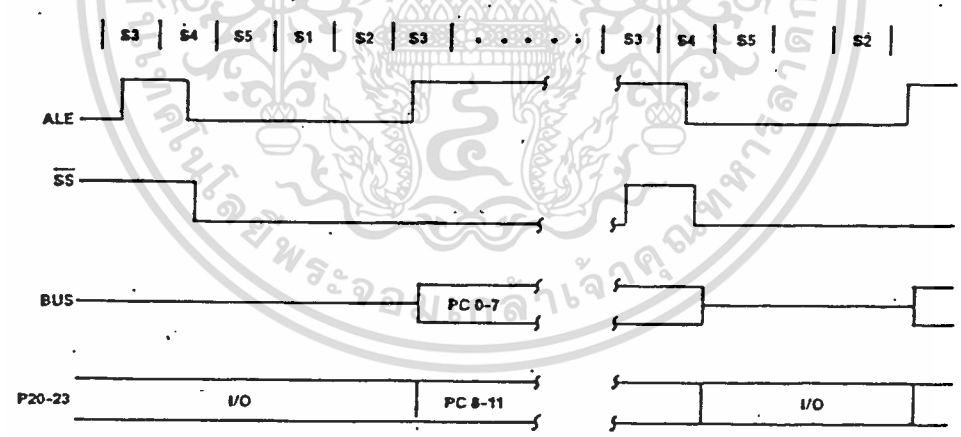
- ตัวนับโปรแกรมถูกเซ็ทให้เป็นศูนย์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- ตัวชี้สแตคถูกเซ็ทเป็นศูนย์
- รีจิสเตอร์ชุดศูนย์ถูกเลือก
- หน่วยความจำโปรแกรมชุดศูนย์ถูกเลือก
- บัสจะถูกเซ็ทเป็น HIGH IMPEDANCE ยกเว้นเมื่อขา EA มีสถานะลอจิกหนึ่ง
- พอร์ท 1 และพอร์ท 2 จะอยู่ในกรณีเป็นอินพุท
- อินเตอร์รัพท์ถูกดิสเอเบิ้ล
- ตัวตั้งเวลาจะหยุด
- แฟลกของตัวตั้งเวลาถูกทำให้เป็นศูนย์
- แฟลกศูนย์และแฟลกหนึ่งถูกทำให้เป็นศูนย์

3.1.12 <u>การทำคำสั่งทีละหนึ่งคำสั่ง</u> (SINGLE-STEP)

การทำคำสั่งทีละหนึ่งคำสั่ง เป็นวิธีที่มีประโยชน์แก่ผู้ใช้ในการตรวจสอบโปรแกรมที่ได้เขียนขึ้นมา โดยการทำงานเป็นแบบที่เรียกว่า HARDWARE หมายความว่าไอซีเบอร์ 8048 มีขาเฉพาะในการทำคำสั่งทีละหนึ่งคำสั่งคือขา ss โดยการทำงานจะทำงานร่วมกันสัญญาณ ALE เมื่อขา ss มีค่าลอจิกเป็นศูนย์ การทำงานของ 8048 จะหยุดโดยค่าตำแหน่งของคำสั่งต่อไปที่จะอ่านเข้ามาจะถูกส่งออกไปยังบัสและ 4 บิทแรกของพอร์ท 2 ดังนั้นผู้ใช้สามารถจะติดตามการทำงานของโปรแกรมที่เขียนขึ้นได้จากแผนผังเวลาของการทำงานที่แสดง ในรูปที่ 1.7 แสดงถึงการทำงานร่วมกันของสัญญาณ ALE และ ss สัญญาณของบัสจะหายไประหว่างการทำคำสั่งทีละหนึ่งคำสั่ง แต่อย่างไรก็ตามถ้าต้องการทราบสัญญาณขนบัส สามารถเห็นได้โดยการแลทช์เข้าช่วยเพื่อเก็บสัญญาณนี้ไว้ตรวจสอบได้



รูปที่ 3.7 แสดงแผนผังเวลาการทำงานทีละคำสั่ง

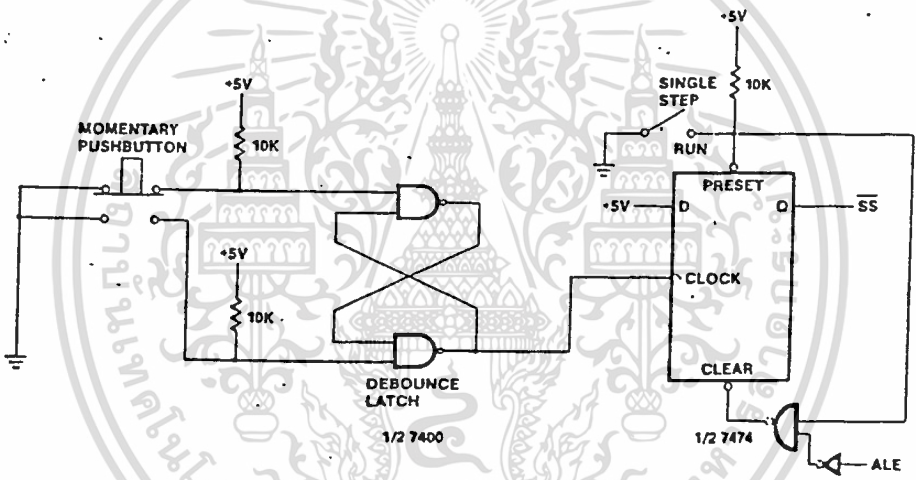
ขั้นตอนการทำงานของผังเวลาเมื่อ 8048 ทำคำสั่งทีละหนึ่งคำสั่ง ดังนี้

- 8048 จะถูกทำให้หยุดทำงานโดยสัญญาณ ss ทำงาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- 8048 จะหยุดการทำงานระหว่างที่จะทำการอ่านคำสั่งต่อไปหากเป็นคำสั่ง 2 ไบท์ จะทำงานจนเสร็จก่อนจึงหยุดทำตามสัญญาณ ss
- 8048 จะตอบรับการหยุดการทำงานโดยจะให้สัญญาณที่ขา ALE มีลอจิกหนึ่งตลอดเวลา ในสถานะอย่างนี้ตำแหน่งของคำสั่งต่อไปที่จะอ่านเข้ามาจะอยู่ที่บัสและ 4 บิทแรกของพอร์ท 2
- เมื่อขา ss มีลอจิกเป็นหนึ่งจะทำให้ 8048 เลิกจากการหยุดการทำงานเพื่ออ่านคำสั่งไป 8048 แสดงการออกจากสถานะการหยุดทำงานโดยทำให้สัญญาณที่ขา ALE อยู่ในลอจิกศูนย์
- เมื่อต้องการให้ 8048 หยุดการทำงานที่คำสั่งต่อไป ขา ss จะต้องมีค่าลอจิกศูนย์อีกครั้งทันทีที่ขา ALE มีสถานะทางลอจิกหนึ่งอยู่ 8048 จะยังคงอยู่ในช่วงการทำงานปกติ

เมื่อต่อกันให้ 8048 ทำงานทีละหนึ่งคำสั่งจำเป็นต้องเพิ่มวงจรการทำงานของส่วนนี้ดังแสดงในรูปที่ 3.8



รูปที่ 3.8 แสดงวงจรในการทำงานทีละหนึ่งคำสั่ง

<u>การทำงานของวงจรเป็นดังนี้</u>

เมื่ออยู่ระหว่างการทำงานปกติ ขา PRE 2 ของ DM74LS74 จะคงอยู่ที่สถานะลอจิกศูนย์ทำให้ขา Q2 มีสถานะลอจิกหนึ่ง เป็นผลให้ขา ss รักษาสถานะลอจิกหนึ่งอยู่ ดังนั้น 8048 ยังคงทำงานได้อย่างปกติ เมื่อสวิชเลือกการทำงานอยู่ที่ SINGLE STEP ทำให้ขา PRE 2 มีสถานะลอจิกหนึ่ง ขณะที่สัญญาณ ALE มีสถานะลอจิกศูนย์สัญญาณที่ขา ss จะถูกทำให้มีสถานะลอจิกศูนย์ด้วยเป็นเหตุให้ 8048 หยุดการทำงาน เมื่อต้องการให้คำสั่งต่อไป เพียงแต่กดปุ่ม MOMENTARY PUSH BUTION ถ้า ALE มีสถานะลอจิกเป็นหนึ่ง ฟรับ-ฟรอบ D1 จะส่งหนึ่งไปยัง D2

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 4

หลักการทำงานของโครงงาน

จากวงจรทั้งหมด เราสามารถแบ่งออกเป็นส่วนๆ ได้ดังนี้คือ

1. ส่วนควบคุม
2. ส่วน BUFFER หรือ PORT INPUT และ PORT OUTPUT
3. ส่วน DETECT สัญญาน TONE
4. ส่วน DRIVE THYRISTER
5. ส่วนอัดเสียง
6. ภาค SUPPLY

- ส่วนควบคุม

ส่วนควบคุมเราสามารถแสดงได้จากวงจรในภาคผนวก ก. ดังรูป A1 ซึ่งในที่นี้เราใช้ CPU เบอร์8048 เป็นตัวควบคุม การทำงานต่าง ๆ ซึ่งคุณสมบัติของ CPU 8048 เราได้ทราบมาแล้ว ในบทที่ 3 โดยการทำงานของมัน จะทำงานร่วมกับ IC74LS373 และ IC2716 (2732) ซึ่งIC74LS373 ทำหน้าที่เป็นตัว LATCH ค่า ADDRESS ออกมาจากสาย DATA BUS จากคุณสมบัติของ 8048ที่ให้ค่า ADDRESS กับค่า DATA ในสายเดียวกัน ซึ่งในที่นี้เราจะได้ ADDRESS A0-A7 ส่วนA8-A11 เราจะได้จาก 4 BIT ล่างของ PORT 2 (P20-P23)

โดยเราจะใช้สัญญาณ ALE ในการแลทช์ค่าของ ADDRESS โดย CPU 8048 จะเป็นตัวสร้างขึ้นมา เวลาเราจะนำค่าของ ADDRESS ไปใช้งานในที่นี้เราจะดึงจาก OUTPUT ของ IC74LS373 ไปใช้งานเลย ส่วนโปรแกรม MONITOR ในที่นี้เราใช้ EPROM เบอร์ 2716 โดยขา 21 เราจะต้องต่อกับไฟบวก 5V. และค่าของ ADDRESS ที่ได้จากการ LATCH ของ IC 74LS373 มันจะนำไปเป็น ADDRESS เพื่อเลือกค่า DATA ภายในตัวของ EPROM 2716 เพื่อเป็นคำสั่งของ CPU 8048 นำไปปฏิบัติการต่อไป จากเหตุผลที่ว่า สาย ADDRESS กับ DATA ของ CPU 8048 เป็นสายเดียวกัน ดังนั้น IC 74LS373 จึงมีความจำเป็นต่อการ DETECT ให้รู้ว่าตอนช่วงนี้ ค่าข้อมูล ที่ออกมาในช่วงนี้ เป็นค่าของ ADDRESS หรือ DATA และในการ FETCH คำสั่งของ CPU 8048 จาก หน่วยความจำภายนอก CPU 8048 ก็จะให้สัญญาน PSEN ACTIVE โดยจากวงจรเราจะนำสัญญานนี้ไปเข้าขา CE ของ IC 2716 เพื่อเป็นตัว SELECT IC 2716

- ส่วน BUFFER หรือ PORT INPUT และ PORT OUTPUT

ภาคนี้จะมีอยู่ในภาคผนวก ก. ทั้งในรูป A2 รูป A3 และรูป A4 โดย IC ที่สำคัญในภาคนี้คือ IC 74LS373 และ IC 74LS244 ซึ่งIC 74LS244 เป็น IC ซึ่งจะให้ OUTPUT เป็น 3 สถานะคือ LOW ,HIGH ,HIGH IMPEDANCE ซึ่งเราจะใช้ IC 74LS244 นี้เป็น PORT

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

INPUT เพราะคุณสมบัติของ PORT INPUT นั้นไม่จำเป็น จะต้อง LATCH ค่าของ OUTPUT ไว้ ดังนั้นเราจึงใช้IC 74LS244 สำหรับ IC 74LS373 นั้นเป็น D LATCH OUTPUT 3 STATE โดย IC ทั้งสองตัวนี้ จะทำงานได้ก็ต่อเมื่อได้รับสัญญาน ENABLE ซึ่งสัญญานนี้ได้มาจากการ นำสัญญาน RDหรือ WR นำมาร่วมกับสัญญานเลือก PORT ซึ่งได้มาจากการนำสาย ADDRESS(A0-A7) มาทำการ เลือก PORT โดยจะนำไป เข้า IC 74LS138 เช่น ในกรณี เราจะเลือก PORT OUTPUT เราจะต้องนำสายสัญญาน WR มาทำการ OR กับสัญญานเลือก PORT ซึ่งได้มาจาก IC 74LS138บางครั้งเราจะต้องนำมา NOR กันจะทำแบบใด ก็ขึ้นกับว่า สัญญานที่เราต้องการนั้น เป็น HIGH หรือ LOW

- ส่วน DETECT สัญญาน TONE

ภาคนี้ แสดงในภาคผนวก ก. ดังรูป A4 ซึ่ง IC ที่ใช้ในการ DETECT TONE จะใช้ IC เบอร์ MT8870 โดย IC MT8870 ตัวนี้เมื่อมีสัญญาน TONE ที่มีความแรงพอสมควรมาปรากฏที่ ขา INPUT ของมัน มันจะทำการ DETECT สัญญาน TONE ไปเป็นรหัส BCD ซึ่งได้ตั้งแต่ 0000-1111H และค่านี้ จะถูกเก็บไว้ที่ OUTPUT Q1-Q4 ซึ่งเราสามารถจะให้ปรากฏ หรือไม่ ก็ได้โดย ถ้าเราให้ที่ขา TOE เป็นสัญญาน HIGH จะทำให้ ที่ OUTPUT Q1-Q4 เป็นค่ารหัส BCD ที่ถอดได้ และถ้าเราให้ที่ขา TOE เป็น LOW จะทำให้ที่ OUTPUTของ IC MT8870เป็น HIGH IMPEDANCE โดยการทำงาน ของ IC MT8870 นี้ที่ OUTPUT ของมัน จะเป็นค่า BCD ของรหัสสัญญาน TONE ซึ่งเข้ามาครั้งล่าสุด

- ภาคขับ THYRISTER

ภาคนี้ เราจะใช้ OPTOISOLATOR เพื่อเป็นการแยกกันระหว่าง GROUND DC กับ GROUND AC เพื่อป้องกันความเสียหายเนื่องจากไฟ AC ซึ่งแสดงในภาคผนวก ก. ดังรูป A2 โดย เราใช้ OPTO เบอร์ MOC 3040 ซึ่งเป็น PHOTO TRIAC WITH ZERO CROSSING โดยที่ขา INPUT ของ MOC 3040 จะมี R อยู่ 1 ตัว เพื่อจำกัดกระแสให้พอเหมาะกับ INPUT ของ MOC 3040 โดยจะถูกขับโดย TR C458 ซึ่งทำหน้าที่เป็น SWITCH ON-OFF ตามสัญญานที่เข้ามาที่ INPUT ของมัน ส่วน OUTPUT ที่ได้จาก ICMOC 3040 จะนำไป DRIVE ตัว TRIAC ซึ่งทำหน้าที่เป็น SWITCH ๆ ไฟ 220 VAC ให้กับ LOAD ในแต่ละตัว

- ส่วนอัดเสียง

ภาคนี้แสดงวงจรได้ ในภาคผนวก ก. ดังรูป A3 ซึ่ง IC ที่เป็นหัวใจของวงจรนี้ คือ IC T6668 ซึ่งทำงานร่วมกับ PORT OUTPUT คือ IC 74LS373 อีก 2 ตัว ซึ่ง PORT OUTPUT สองตัวนี้ จะเป็นตัวควบคุมให้ T6668 ทำงานตามจังหวะที่เราสั่ง OUTPORT เข้าไปที่ตัว IC 74LS373 โดยช่วงแรกเราจะต้องเลือก PAGE ของการอัด และเลือก SPEED ของการอัด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การเลือก PAGE เราสามารถกำหนดได้โดยการ SET ที่ BIT D0-D3 ส่วนเลือก SPEED นั้น กำหนดที่ BIT D6-D7 สำหรับการเลือก SPEED ต่าง ๆ นั้นเราได้ทราบมาแล้วในบทที่ 2 เมื่อเราเลือก SPEED และ PAGE แล้ว คราวนี้เราจะเริ่มสั่งให้มันเริ่ม START การ RECORD หรือ PLAY ส่วนที่เรา RECORD ไว้ โดยกำหนดให้ที่ BIT D4 เป็น HIGH และ ถ้าเราต้องการให้ STOP ก็ทำได้โดยทำให้ BIT D5 เป็น HIGH ซึ่งการสั่งให้ T6668 ทำงานนี้เราสั่งโดยการใช้โปรแกรม ซึ่งตามปกติ T6668 จะทำงานร่วมกับ HARDWARE คือ เราใช้SWITCH เป็นตัวควบคุม ดังนั้นเมื่อเรานำ T6668 มาทำงานร่วมกับ CPU ซึ่งตามปกติทำงานเร็วมาก ดังนั้นการสั่ง T6668 ทำงานในแต่ละครั้ง จะต้องมีการ DELAY เพื่อมันจะได้ทำงานทันกับ CPU

- ภาค POWER SUPPLY

ภาคนี้ แสดงในภาคผนวก ก. ในรูป A5 ซึ่งเราใช้ IC 7805 ที่มีตัวถังเป็นเหล็ก (TO-8) ซึ่งสามารถจ่ายกระแสได้ 3 AMP และ ส่วนนี้ ยังประกอบด้วย BATTERY 12 VDC. เป็นชุดสำรองอีก 1 ชุด เพื่อป้องกันในกรณีไฟฟ้าดับ จะได้สามารถเก็บสภาวะต่างๆ ไว้ได้

ส่วนของ PROGRAM MONITOR

ในส่วนนี้เราจะแสดงไว้ในภาคผนวก ก. (B1-B23) ไว้ทั้งหมด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 5

สรุปผลการทดลองและวิจารณ์

จากเครื่องต้นแบบที่เราประกอบขึ้นมา จะเห็นว่าการใช้งานได้ผลยังไม่เป็นที่น่าพอใจนัก เพราะ เรายังขาดส่วนประกอบอีกบางส่วน ซึ่งถ้าเรามีส่วนนี้เข้ามา ก็จะทำให้โครงงานนี้สมบูรณ์มากยิ่งขึ้น ในเครื่องนี้ส่วนที่ใช้งานได้ก็คือ การสั่งเปิด ปิด อุปกรณ์ การตรวจเช็คสภาวะ และ ส่วนของการตอบรับโทรศัพท์ แต่ส่วนที่ผู้จัดทำบอกว่ายังขาดอยู่ก็คือ ส่วนของการอัดข้อความจากผู้ที่โทรศัพท์เข้ามา เนื่องจากทางผู้จัดทำยังไม่สามารถหา DATA SHEET ของ IC เบอร์ T6668 ได้ เพื่อนำมาประกอบกับการปรับแต่งวงจร ดังนั้นงานในส่วนนี้จึงยังขาดไป ซึ่งผู้จัดทำมีข้อเสนอแนะให้ไปหาดู DATA SHEETของ T6668 แล้วนำข้อมูลมาประยุกต์เข้ากับโครงงานนี้ ซึ่งผู้จัดทำคิดว่ามันจะทำให้โครงงานนี้สมบูรณ์มากยิ่งขึ้น และเราสามารถเพิ่มโปรแกรมการเรียกฟังเสียงที่อัดไว้ได้ โดยการกดรหัสเข้ามาบอก CPU ว่าเราต้องการฟัง และสามารถเลือกว่าจะฟังของช่องใด ซึ่งส่วนต่างๆนี้เราสามารถเลือกโดยผ่านโปรแกรม และปัญหาที่พบอีกอย่างหนึ่งก็คือ การรบกวนจากสัญญาณภายนอก ดังนั้นเราต้องคำนึงถึงในกรณีนี้ด้วย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กิตติกรรมประกาศ

ปริญญานิพนธ์เรื่อง เครื่องสั่งงานทางโทรศัพท์นี้ มีแนวความคิด มาจาก คุณ ฉัตรชัย ดอกนารี ผู้จัดทำก็ต้องขอขอบคุณยิ่ง และก็ต้องขอขอบพระคุณ อาจารย์นิกร สุขุตมตันติ ซึ่งท่านได้ให้ความช่วยเหลือทุกอย่าง จนสำเร็จ ลุล่วงไปด้วยดี และต้องขอขอบคุณเพื่อนๆ ที่ให้ความร่วมมือ และให้ความสนับสนุนด้วยดีตลอดเวลา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บรรณานุกรม

1.RODNAY ZAKS,AUSTIN LESEA,"MICROPROCESSOR INTERFACING TECHNIQUES", SYBEX

2.EDWARD M.NOLL,"MICROPROCESSOR CIRCUIT",HOWARD W.SAMS.

3.ยืน ภู่วรวรรณ,"เทคนิคการประยุกต์และการใช้งานไอซีทีทีแอล",ซีเอดยูเคชั่น,2523

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก

ประกอบด้วยวงจรทดลอง เนื้อโปรแกรม และ DATA SHEET ของอุปกรณ์ที่สำคัญ

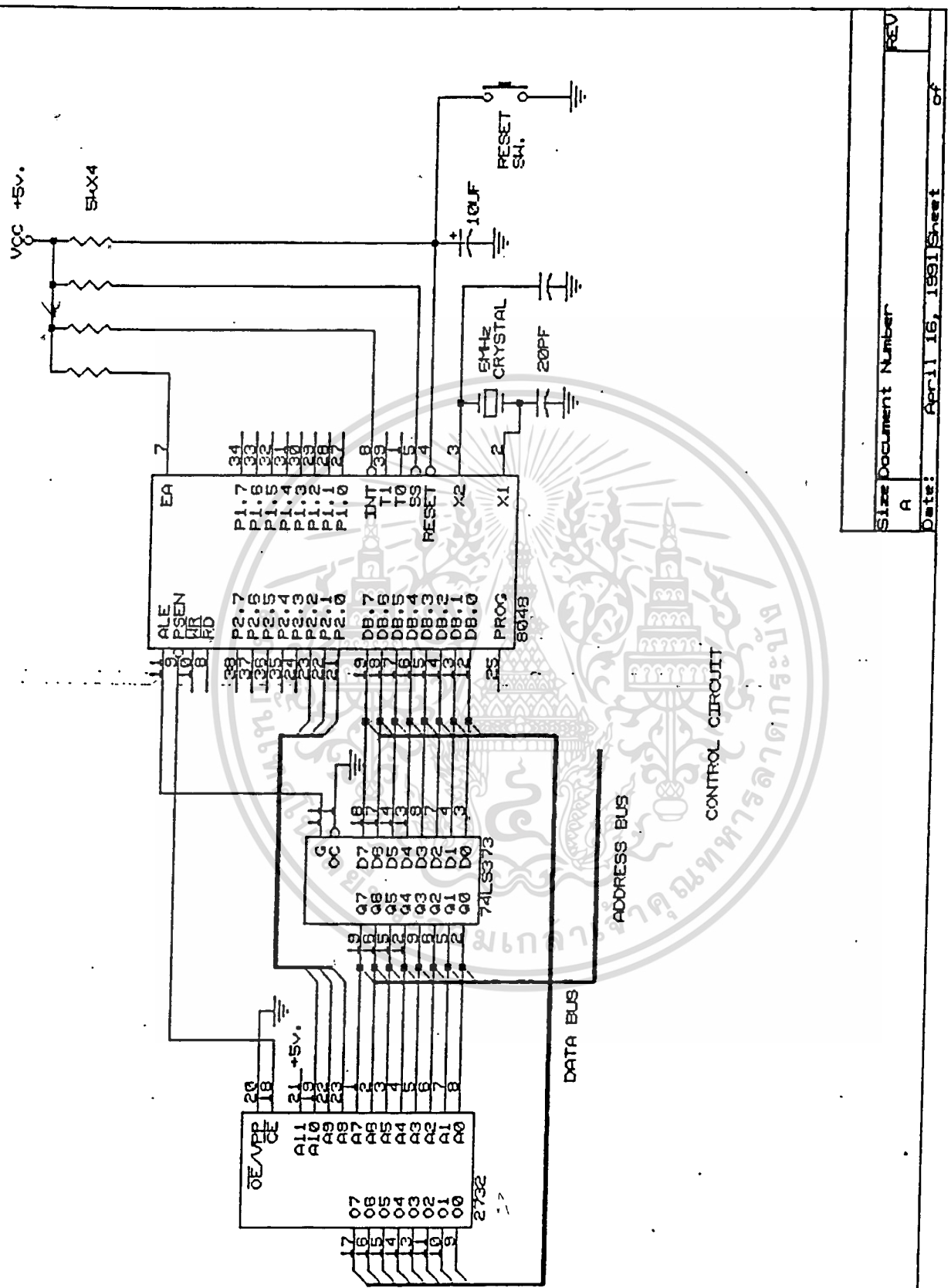
1. วงจรทดลองเครื่องสั่งงานทางโทรศัพท์ A1-A5

2. โปรแกรม

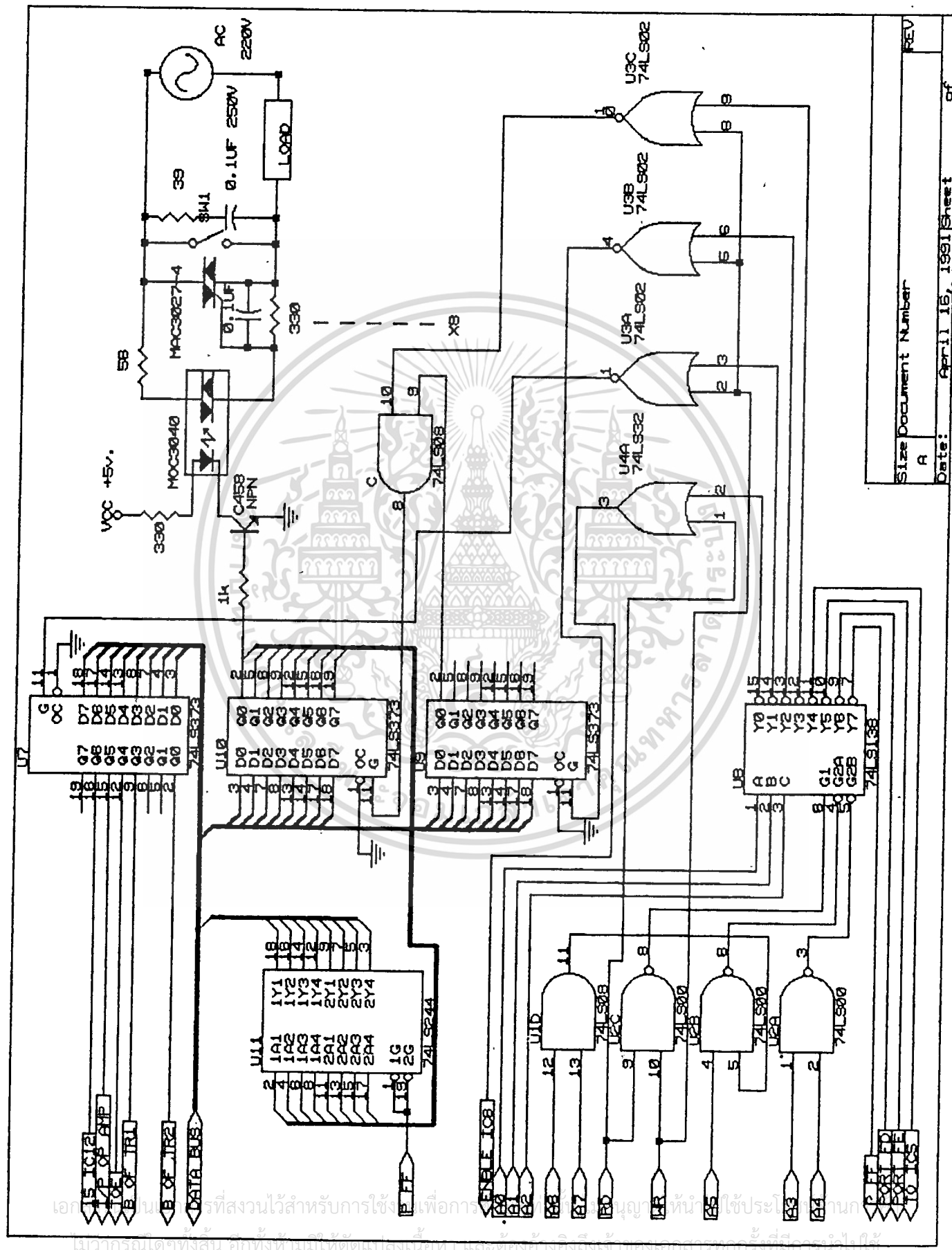
- MAIN PROGRAM B1-B3
- SUBPROGRAM RINGGING B3-B4
- SUBPROGRAM USERCODE B4-B6
- SUBPROGRAM DETECTTONE B6
- SUBPROGRAM OVER B6-B8
- SUBPROGRAM DELAYFORTONE B8-B9
- SUBPROGRAM MODE B9-B10
- SUBPROGRAM NOTUSE B10
- SUBPROGRAM RESET B10-B11
- SUBPROGRAM STATUS B11-B12
- SUBPROGRAM SETRESET B12-B13
- SUBPROGRAM OVER1 B13-B14
- SUBPROGRAM SONG B14-B15
- SUBPROGRAM SETSYS B15
- SUBPROGRAM DELAY B15-B16
- SUBPROGRAM DELAY2 B16
- SUBPROGRAM ANSWER B17
- SUBPROGRAM REC B17-B18
- SUBPROGRAM READ20 B18-B19
- SUBPROGRAM DELAY3 B20
- SUBPROGRAM DELAYFORRW B20
- SUBPROGRAM CHECK B20-B21
- SUBPROGRAM DELAYCHECK B22

3. DATA SHEET OF 8048CPU C1-C7

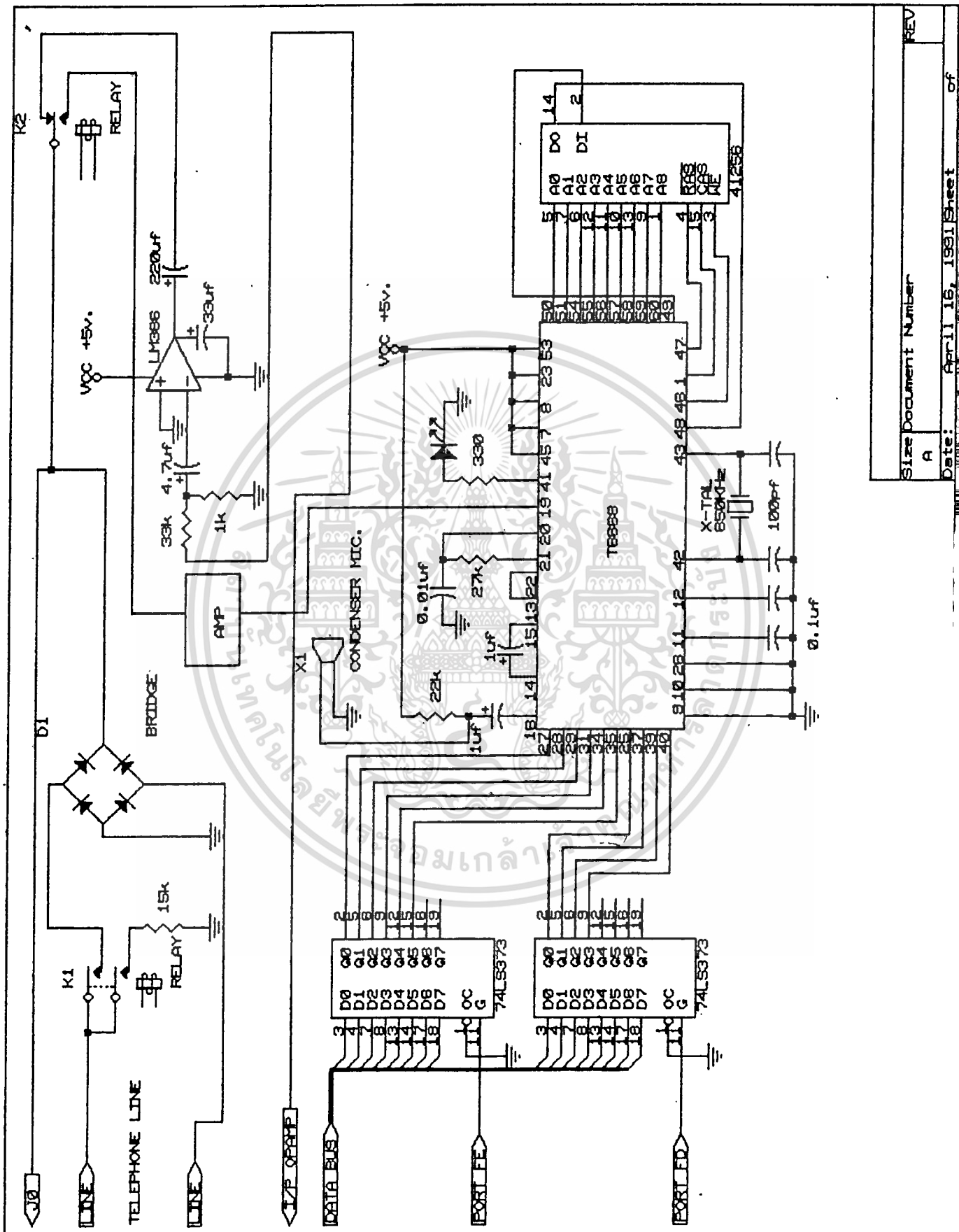
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



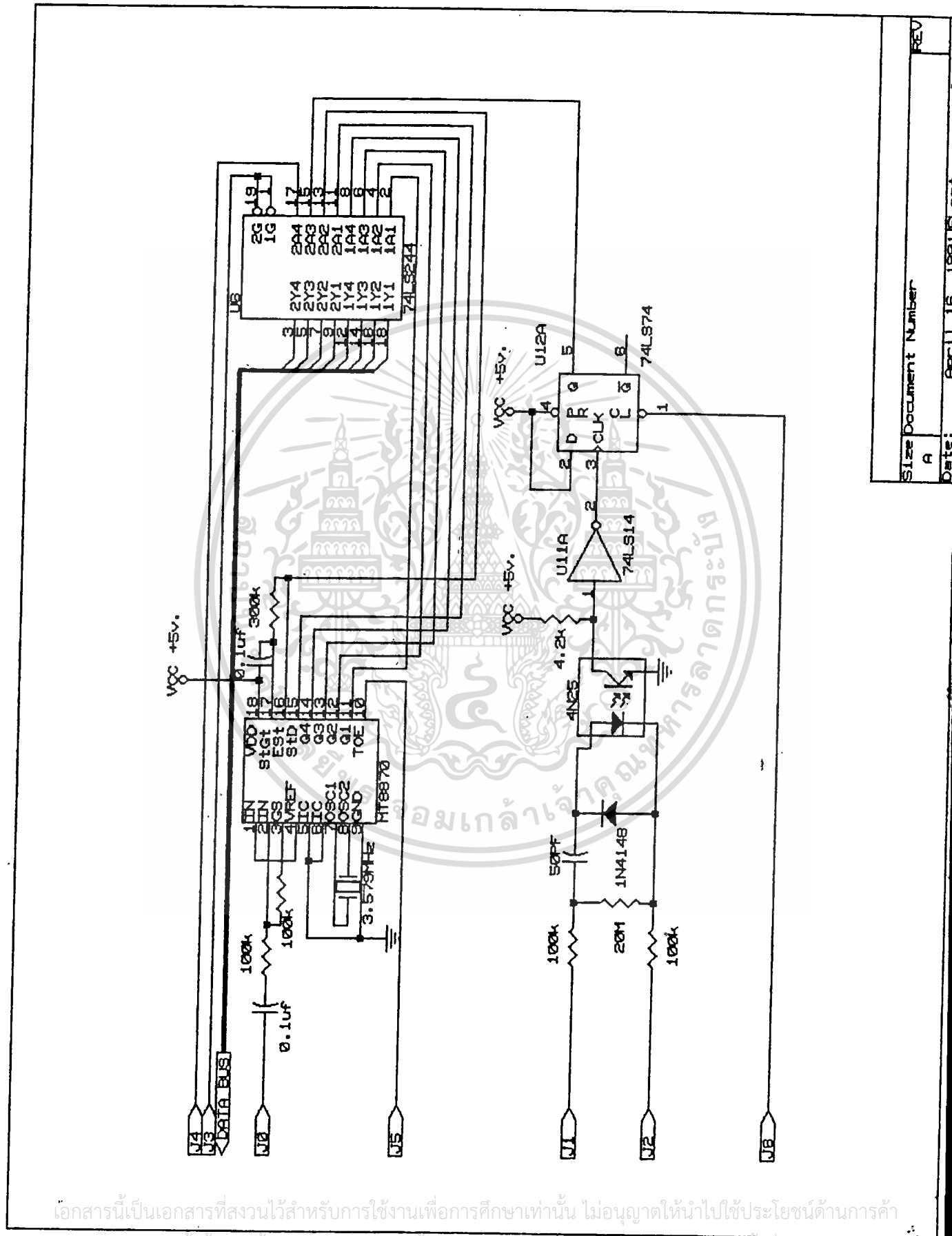
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



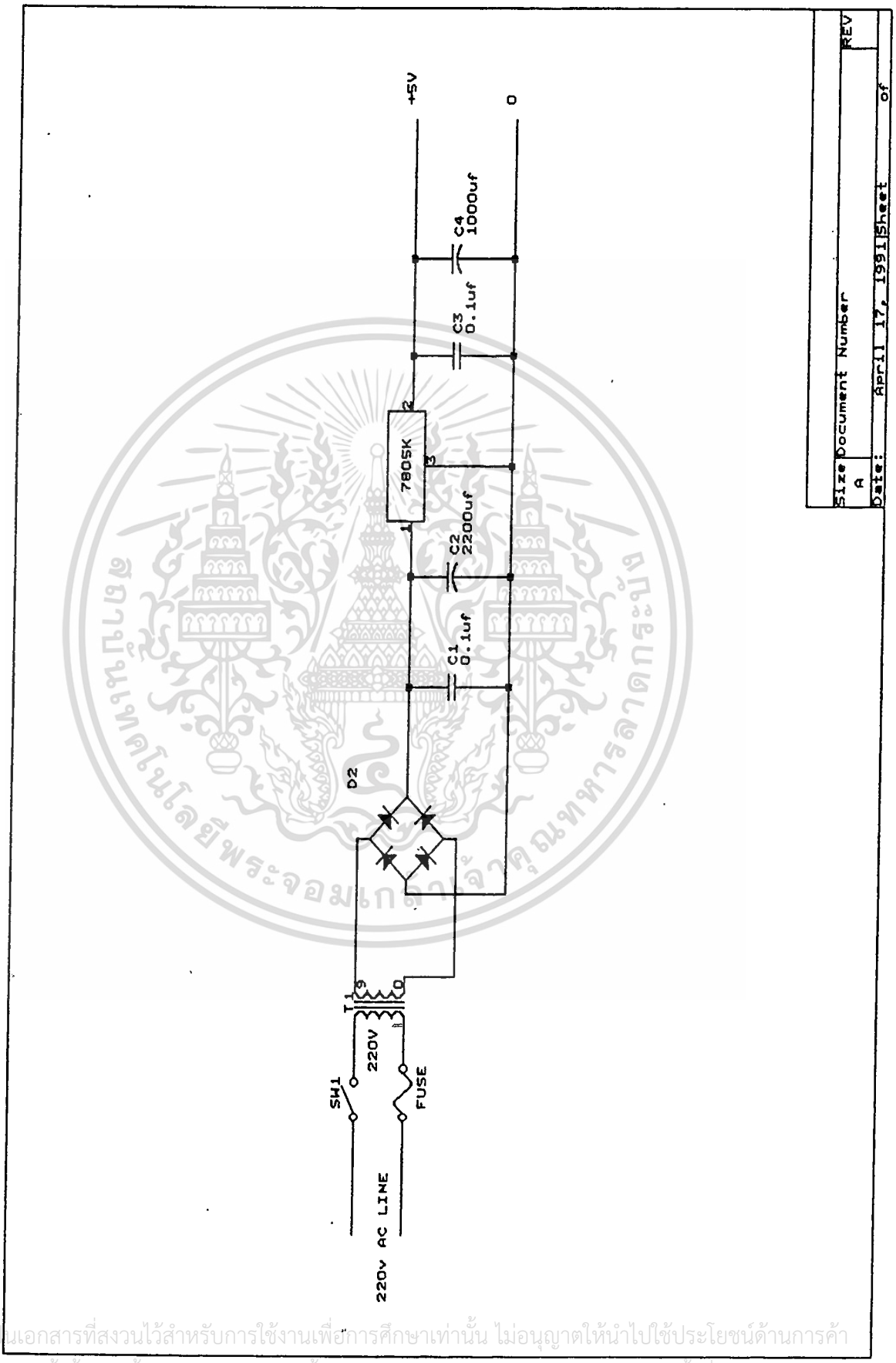
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
000000                          CPU "8048.TBL"
                                :MAIN PROGRAM
000000 B82F                     MOV R0,#2FH
000002 B001                     MOV @R0,#01H
000004 B82E                     MOV R0,#2EH
000006 B000                     MOV @R0,#00H
000008 27                       CLR A
000009 B8FD                     MOV R0,#0FDH
00000B 90                       MOVX @R0,A
00000C 945D                     CALL REC
00000E 5442                     CALL RESET
000010 9400             START:  CALL SETSYS
000012 147C                     CALL RINGING
000014 3400                     CALL USERCODE
000016 B822                     MOV R0,#22H
000018 F0                       MOV A,@R0
000019 53FF                     ANL A,#0FFH
00001B 962B                     JNZ NEXT1
00001D 2300             READ:   MOV A,#00H
00001F B8F9                     MOV R0,#0F9H
000021 90                       MOVX @R0,A
000022 B8F8                     MOV R0,#0F8H
000024 80                       MOVX A,@R0
000025 5340                     ANL A,#40H
000027 961D                     JNZ READ
000029 0410                     JMP START
00002B BA02             NEXT1:  MOV R2,#02H
00002D 9400             LOOP1:  CALL SETSYS
00002F 9426                     CALL DELAY2
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
000031 3470                        CALL DETECTONE
000033 3495                        CALL OVER
000035 B825                        MOV R0,#25H
000037 F0                          MOV A,@R0
000038 53FF                        ANL A,#0FFH
00003A 9640                        JNZ NEXT2
00003C EA2D                        DJNZ R2,LOOP1
00003E 0410                        JMP START
000040 5400              NEXT2:    CALL MODE
000042 00000000000000              DFB 00H,00H,00H,00H,00H,00H,00H
000049 B827                        MOV R0,#27H
00004B F0                          MOV A,@R0
00004C 53FF                        ANL A,#0FFH
00004E 9610                        JNZ START
000050 B828                        MOV R0,#28H
000052 F0                          MOV A,@R0
000053 53FF                        ANL A,#0FFH
000055 962B                        JNZ NEXT1
000057 B829                        MOV R0,#29H
000059 F0                          MOV A,@R0
00005A 53FF                        ANL A,#0FFH
00005C 962B                        JNZ NEXT1
00005E B82B                        MOV R0,#2BH
000060 F0                          MOV A,@R0
000061 53FF                        ANL A,#0FFH
000063 C674                        JZ NEXT3
000065 9426                        CALL DELAY2
000067 3470                        CALL DETECTONE
000069 54AF                        CALL OVER1
00006B B82C                        MOV R0,#2CH
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
00006D F0                       MOV A,@R0
00006E 53FF                     ANL A,#0FFH
000070 9676                     JNZ NEXT4
000072 0410                     JMP START
000074 548B         NEXT3:      CALL SETRESET
000076 9426         NEXT4:      CALL DELAY2
000078 5459                     CALL STATUS
00007A 042B                     JMP NEXT1

00007C                          ORG 007CH
                                ;SUBPROGRAM RINGING
00007C C5           RINGING:    SEL RB0
00007D B833         START10:    MOV R0,#33H
00007F B000                     MOV @R0,#00H
000081 27                       CLR A
000082 B8F9                     MOV R0,#0F9H
000084 90                       MOVX @R0,A
000085 2340                     MOV A,#40H
000087 90                       MOVX @R0,A
000088 BD01                     MOV R5,#01H
00008A BE07         LOOP70:     MOV R6,#07H
00008C BF11         LOOP71:     MOV R7,#11H
00008E B498         LOOP72:     CALL CHECK
000090 B833                     MOV R0,#33H
000092 F0                       MOV A,@R0
000093 D304                     XRL A,#04H
000095 C69F                     JZ STOP
000097 EF8E                     DJNZ R7,LOOP72
000099 EE8C                     DJNZ R6,LOOP71
00009B ED8A                     DJNZ R5,LOOP70
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
00009D 047D                     JMP START10
00009F D5           STOP:       SEL RB1
0000A0 83                       RET

000100                          ORG 0100H
                                ; SOBPROGRAM USERCODE
000100 C5           USERCODE:   SEL RB0
000101 2308                     MOV A,#08H
000103 B8F9                     MOV R0,#0F9H
000105 90                       MOVX @R0,A
000106 B901                     MOV R1,#01H
000108 BAFF         LOOP4:      MOV R2,#0FFH
00010A BBFF         LOOP5:      MOV R3,#0FFH
00010C B8F8         LOOP6:      MOV R0,#0F8H
00010E 80                       MOVX A,@R0
00010F 5310                     ANL A,#10H
000111 961D                     JNZ READ1
000113 EB0C                     DJNZ R3,LOOP6
000115 EA0A                     DJNZ R2,LOOP5
000117 E908                     DJNZ R1,LOOP4
000119 9440                     CALL ANSWER
00011B 2462                     JMP END1
00011D 2318         READ1:      MOV A,#18H
00011F B8F9                     MOV R0,#0F9H
000121 90                       MOVX @R0,A
000122 B8F8                     MOV R0,#0F8H
000124 80                       MOVX A,@R0
000125 530F                     ANL A,#0FH
000127 D30C                     XRL A,#0CH
000129 9662                     JNZ END1
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
00012B 9417                     CALL DELAY
00012D B821                     MOV R0,#21H
00012F B000                     MOV @R0,#00H
000131 54CF                     CALL SONG
000133 B901                     MOV R1,#01H
000135 BAFF         LOOP7:      MOV R2,#0FFH
000137 BBFF         LOOP8:      MOV R3,#0FFH
000139 B8F8         LOOP9:      MOV R0,#0F8H
00013B 80                       MOVX A,@R0
00013C 5310                     ANL A,#10H
00013E 9648                     JNZ READ2
000140 EB39                     DJNZ R3,LOOP9
000142 EA37                     DJNZ R2,LOOP8
000144 E935                     DJNZ R1,LOOP7
000146 2462                     JMP END1
000148 2318         READ2:      MOV A,#18H
00014A B8F9                     MOV R0,#0F9H
00014C 90                       MOVX @R0,A
00014D B8F8                     MOV R0,#0F8H
00014F 80                       MOVX A,@R0
000150 530F                     ANL A,#0FH
000152 D30B                     XRL A,#0BH
000154 9662                     JNZ END1
000156 B821                     MOV R0,#21H
000158 B010                     MOV @R0,#10H
00015A 54CF                     CALL SONG
00015C B822                     MOV R0,#22H
00015E B0FF                     MOV @R0,#0FFH
000160 2466                     JMP END10
000162 B822         END1:       MOV R0,#22H
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
000164 B000                         MOV @R0,#00H
000166 D5           END10:          SEL RB1
000167 83                           RET

000170                              ORG  0170H
                                    ;SUBPROGRAM DETECTONE
000170 C5           DETECTONE:      SEL RB0
000171 B821                         MOV R0,#21H
000173 B000                         MOV @R0,#00H
000175 54CF                         CALL SONG
000177 34CE                         CALL DELAYFORTONE
000179 B8F8                         MOV R0,#0F8H
00017B 80                           MOVX A,@R0
00017C 530F                         ANL A,#0FH
00017E B823                         MOV R0,#23H
000180 A0                           MOV @R0,A
000181 9426                         CALL DELAY2
000183 B821                         MOV R0,#21H
000185 B080                         MOV @R0,#80H
000187 54CF                         CALL SONG
000189 34CE                         CALL DELAYFORTONE
00018B B8F8                         MOV R0,#0F8H
00018D 80                           MOVX A,@R0
00018E 530F                         ANL A,#0FH
000190 B824                         MOV R0,#24H
000192 A0                           MOV @R0,A
000193 D5                           SEL RB1
000194 83                           RET
000195                              ORG 0195H
                                    ;SUBPROGRAM OVER
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
000195 C5              OVER:    SEL RB0
000196 B823                     MOV R0,#23H
000198 F0                       MOV A,@R0
000199 E7                       RL A
00019A E7                       RL A
00019B E7                       RL A
00019C E7                       RL A
00019D 0350                     ADD A,#50H
00019F F6C2                     JC END4
0001A1 F0                       MOV A,@R0
0001A2 D30A                     XRL A,#0AH
0001A4 C6B7                     JZ END3
0001A6 D309                     XRL A,#09H
0001A8 C6C2                     JZ END4
0001AA B824                     MOV R0,#24H
0001AC F0                       MOV A,@R0
0001AD E7                       RL A
0001AE E7                       RL A
0001AF E7                       RL A
0001B0 E7                       RL A
0001B1 0370                     ADD A,#70H
0001B3 F6C2                     JC END4
0001B5 24C8                     JMP NEXT9
0001B7 B824            END3:    MOV R0,#24H
0001B9 F0                       MOV A,@R0
0001BA E7                       RL A
0001BB E7                       RL A
0001BC E7                       RL A
0001BD E7                       RL A
0001BE 03B0                     ADD A,#0B0H
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
0001C0 E6C8                    JNC NEXT9
0001C2 B825          END4:     MOV R0,#25H
0001C4 B000                    MOV @R0,#00H
0001C6 24CC                    JMP END5
0001C8 B825          NEXT9:    MOV R0,#25H
0001CA B0FF                    MOV @R0,#0FFH
0001CC D5            END5:     SEL RB1
0001CD 83                      RET

0001CE                         ORG 01CEH
                               ;SUBPROGRAM DELAYFORTONE
0001CE C5            DELAYFORTONE: SEL RB0
0001CF B901                    MOV R1,#01H
0001D1 BAFF          CYCLE1:   MOV R2,#0FFH
0001D3 BBFF          CYCLE2:   MOV R3,#0FFH
0001D5 B8F8          CYCLE3:   MOV R0,#0F8H
0001D7 80                      MOVX A,@R0
0001D8 5310                    ANL A,#10H
0001DA 96E4                    JNZ CYCLE4
0001DC EBD5                    DJNZ R3,CYCLE3
0001DE EAD3                    DJNZ R2,CYCLE2
0001E0 E9D1                    DJNZ R1,CYCLE1
0001E2 24FE                    JMP END7
0001E4 B901          CYCLE4:   MOV R1,#01H
0001E6 BAFF          CYCLE5:   MOV R2,#0FFH
0001E8 BBFF          CYCLE6:   MOV R3,#0FFH
0001EA B8F8          CYCLE7:   MOV R0,#0F8H
0001EC 80                      MOVX A,@R0
0001ED 5310                    ANL A,#10H
0001EF C6F9                    JZ CYCLE8
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
0001F1 EBEA                     DJNZ R3,CYCLE7
0001F3 EAE8                     DJNZ R2,CYCLE6
0001F5 E9E6                     DJNZ R1,CYCLE5
0001F7 24FE                     JMP END7
0001F9 2318           CYCLE8:   MOV A,#18H
0001FB B8F9                     MOV R0,#0F9H
0001FD 90                       MOVX @R0,A
0001FE D5             END7:     SEL RB1
0001FF 83                       RET

000200                          ORG 0200H
                                ;SUBPROGRAM MODE
000200 C5             MODE:     SEL RB0
000201 B823                     MOV R0,#23H
000203 F0                       MOV A,@R0
000204 E7                       RL A
000205 E7                       RL A
000206 E7                       RL A
000207 E7                       RL A
000208 A9                       MOV R1,A
000209 B824                     MOV R0,#24H
00020B F0                       MOV A,@R0
00020C 69                       ADD A,R1
00020D B826                     MOV R0,#26H
00020F A0                       MOV @R0,A
000210 00000000000000           DFB 00H,00H,00H,00H,00H,00H,00H
000217 D3A1                     XRL A,#0A1H
000219 961D                     JNZ LOOP13
00021B 5436                     CALL NOTUSE
00021D F0             LOOP13:   MOV A,@R0
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
00021E D3A2                     XRL A,#0A2H
000220 9624                     JNZ LOOP14
000222 5478                     CALL SET
000224 F0           LOOP14:     MOV A,@R0
000225 D3A3                     XRL A,#0A3H
000227 962B                     JNZ LOOP15
000229 5442                     CALL RESET
00022B F0           LOOP15:     MOV A,@R0
00022C D3A4                     XRL A,#0A4H
00022E 9634                     JNZ NEXTA
000230 B82B                     MOV R0,#2BH
000232 B0FF                     MOV @R0,#0FFH
000234 D5           NEXTA:      SEL RB1
000235 83                       RET
000236                          ORG 0236H
                                ;SUBPROGRAM NOTUSE
000236 C5           NOTUSE:     SEL RB0
000237 2300                     MOV A,#00H
000239 B8F9                     MOV R0,#0F9H
00023B 90                       MOVX @R0,A
00023C B827                     MOV R0,#27H
00023E B0FF                     MOV @R0,#0FFH
000240 D5                       SEL RB1
000241 83                       RET

000242                          ORG 0242H
                                ;SUBPROGRAM RESET
000242 C5           RESET:      SEL RB0
000243 B8FA                     MOV R0,#0FAH
000245 23FF                     MOV A,#0FFH
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
000247 90                          MOVX @R0,A
000248 2300                        MOV A,#00H
00024A B8FB                        MOV R0,#0FBH
00024C 90                          MOVX @R0,A
00024D B8FA                        MOV R0,#0FAH
00024F 90                          MOVX @R0,A
000250 B82A                        MOV R0,#2AH
000252 A0                          MOV @R0,A
000253 B829                        MOV R0,#29H
000255 B0FF                        MOV @R0,#0FFH
000257 D5                          SEL RB1
000258 83                          RET
000259                             ORG 0259H
                                   ;SUBPROGRAM STATUS
000259 C5                STATUS:   SEL RB0
00025A B826                        MOV R0,#26H
00025C F0                          MOV A,@R0
00025D 530F                        ANL A,#0FH
00025F AA                          MOV R2,A
000260 B8FF                        MOV R0,#0FFH
000262 80                          MOVX A,@R0
000263 F7                LOOP16:   RLC A
000264 EA63                        DJNZ R2,LOOP16
000266 F670                        JC OPEN
000268 B821                        MOV R0,#21H          ;CLOSE
00026A B060                        MOV @R0,#60H
00026C 54CF                        CALL SONG
00026E 4476                        JMP END8
000270 B821                OPEN:   MOV R0,#21H
000272 B040                        MOV @R0,#40H
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
000274 54CF                     CALL SONG
000276 D5           END8:       SEL RB1
000277 83                       RET
000278                          ORG 0278H
                                ;SUBPROGRAM SET
000278 C5           SET:        SEL RB0
000279 B8FA                     MOV R0,#0FAH
00027B 23FF                     MOV A,#0FFH
00027D 90                       MOVX @R0,A
00027E B8FB                     MOV R0,#0FBH
000280 90                       MOVX @R0,A
000281 27                       CLR A
000282 B8FA                     MOV R0,#0FAH
000284 90                       MOVX @R0,A
000285 B828                     MOV R0,#28H
000287 B0FF                     MOV @R0,#0FFH
000289 D5                       SEL RB1
00028A 83                       RET
00028B                          ORG 028BH
                                ;SUBPROGRAM SETRESET
00028B C5           SETRESET:   SEL RB0
00028C B826                     MOV R0,#26H
00028E F0                       MOV A,@R0
00028F 530F                     ANL A,#0FH
000291 AA                       MOV R2,A
000292 AB                       MOV R3,A
000293 B8FF                     MOV R0,#0FFH
000295 80                       MOVX A,@R0
000296 97                       CLR C
000297 F7           LOOP17:     RLC A
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
000298 EA97                    DJNZ R2,LOOP17
00029A A7                      CPL C
00029B 67          LOOP18:     RRC A
00029C EB9B                    DJNZ R3,LOOP18
00029E A9                      MOV R1,A
00029F 23FF                    MOV A,#0FFH
0002A1 B8FA                    MOV R0,#0FAH
0002A3 90                      MOVX @R0,A
0002A4 F9                      MOV A,R1
0002A5 B8FB                    MOV R0,#0FBH
0002A7 90                      MOVX @R0,A
0002A8 2300                    MOV A,#00H
0002AA B8FA                    MOV R0,#0FAH
0002AC 90                      MOVX @R0,A
0002AD D5                      SEL RB1
0002AE 83                      RET
0002AF                         ORG 02AFH
                               ;SUBPROGRAM OVER1
0002AF C5          OVER1:      SEL RB0
0002B0 B823                    MOV R0,#23H
0002B2 F0                      MOV A,@R0
0002B3 E7                      RL A
0002B4 E7                      RL A
0002B5 E7                      RL A
0002B6 E7                      RL A
0002B7 A9                      MOV R1,A
0002B8 B824                    MOV R0,#24H
0002BA F0                      MOV A,@R0
0002BB 69                      ADD A,R1
0002BC B826                    MOV R0,#26H
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
0002BE A0                       MOV @R0,A
0002BF 03E7                     ADD A,#0E7H
0002C1 F6C9                     JC END9
0002C3 B82C                     MOV R0,#2CH
0002C5 B0FF                     MOV @R0,#0FFH
0002C7 44CD                     JMP NEXTC
0002C9 B82C             END9:   MOV R0,#2CH
0002CB B000                     MOV @R0,#00H
0002CD D5               NEXTC:  SEL RB1
0002CE 83                       RET
0002CF                          ORG 02CFH
                                ;SUBPROGRAM SONG
0002CF C5               SONG:   SEL RB0
0002D0 B821                     MOV R0,#21H
0002D2 F0                       MOV A,@R0
0002D3 AB                       MOV R3,A
0002D4 B92D             LOOP20: MOV R1,#2DH
0002D6 2328             LOOP21: MOV A,#28H
0002D8 B8F9                     MOV R0,#0F9H
0002DA 90                       MOVX @R0,A
0002DB 54EE                     CALL DELAY1
0002DD 2308                     MOV A,#08H
0002DF 90                       MOVX @R0,A
0002E0 54EE                     CALL DELAY1
0002E2 E9D6                     DJNZ R1,LOOP21
0002E4 FB                       MOV A,R3
0002E5 17                       INC A
0002E6 AB                       MOV R3,A
0002E7 E3                       MOVP3 A,@A
0002E8 D3FF                     XRL A,#0FFH
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
0002EA 96D4                        JNZ LOOP20
0002EC D5                          SEL RB1
0002ED 83                          RET
0002EE FB             DELAY1:      MOV A,R3
0002EF E3                          MOVP3 A,@A
0002F0 AA                          MOV R2,A
0002F1 BC12           LOOP22:      MOV R4,#12H
0002F3 ECF3           LOOP23:      DJNZ R4,LOOP23
0002F5 EAF1                        DJNZ R2,LOOP22
0002F7 83                          RET

000400                             ORG 0400H
                                   ;SUBPROGRAM SETSYS
000400 C5             SETSYS:      SEL RB0
000401 B827                        MOV R0,#27H
000403 B000                        MOV @R0,#00H
000405 B828                        MOV R0,#28H
000407 B000                        MOV @R0,#00H
000409 B829                        MOV R0,#29H
00040B B000                        MOV @R0,#00H
00040D B82B                        MOV R0,#2BH
00040F B000                        MOV @R0,#00H
000411 00000000                    DFB 00H,00H,00H,00H
000415 D5                          SEL RB1
000416 83                          RET
000417                             ORG 0417H
                                   ;SUBPROGRAM DELAY
000417 C5             DELAY:       SEL RB0
000418 B803                        MOV R0,#03H
00041A B9FF           LOOP24:      MOV R1,#0FFH
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
00041C BAFF          LOOP25:     MOV R2,#0FFH
00041E EA1E          LOOP26:     DJNZ R2,LOOP26
000420 E91C                      DJNZ R1,LOOP25
000422 E81A                      DJNZ R0,LOOP24
000424 D5                        SEL RB1
000425 83                        RET

                                 ;SUBPROGRAM DELAY2
000426 C5            DELAY2:     SEL RB0
000427 B801                      MOV R0,#01H
000429 B9FF          LOOP27:     MOV R1,#0FFH
00042B BAFF          LOOP28:     MOV R2,#0FFH
00042D EA2D          LOOP29:     DJNZ R2,LOOP29
00042F E92B                      DJNZ R1,LOOP28
000431 E829                      DJNZ R0,LOOP27
000433 D5                        SEL RB1
000434 83                        RET

000300                           ORG 0300H
000300 06050403020202  TABLE:    DFB 06H,05H,04H,03H,02H,02H,02H,02H,0FF
000309 00000000000000            DFB 00H,00H,00H,00H,00H,00H,00H
000310 0F0B0B0F0B0B              DFB 0FH,0BH,0BH,0FH,0BH,0BH
000316 0F0B0B0B0B0B              DFB 0FH,0BH,0BH,0BH,0BH,0BH
00031C 0B0D0A0A0909              DFB 0BH,0DH,0AH,0AH,09H,09H
000322 090A09090909              DFB 09H,0AH,09H,09H,09H,0DH
000328 09090B0B0B09              DFB 09H,09H,0BH,0BH,0BH,09H
00032E 09FF00000000              DFB 09H,0FFH,00H,00H,00H,00H
000334 000000000000             DFB 00H,00H,00H,00H,00H,00H
00033A 000000000000              DFB 00H,00H,00H,00H,00H,00H
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
00045D C5              REC:     SEL RB0
00045E 0000000000               DFB 00H,00H,00H,00H,00H
000463 2303                     MOV A,#03H
000465 B8FD                     MOV R0,#0FDH
000467 90                       MOVX @R0,A
000468 B82E                     MOV R0,#2EH
00046A F0                       MOV A,@R0
00046B 4390                     ORL A,#90H
00046D B8FC                     MOV R0,#0FCH
00046F 90                       MOVX @R0,A
000470 B47E                     CALL DELAY3
000472 53EF                     ANL A,#0EFH
000474 B8FC                     MOV R0,#0FCH
000476 90                       MOVX @R0,A
000477 B489                     CALL DELAYFORRW
000479 2320                     MOV A,#20H
00047B B8FC                     MOV R0,#0FCH
00047D 90                       MOVX @R0,A
00047E 000000000000             DFB 00H,00H,00H,00H,00H,00H
000484 0000000000               DFB 00H,00H,00H,00H,00H
000489 D5                       SEL RB1
00048A 83                       RET

000491                          ORG 0491H
                                ;SUBPROGRAM READ20
000491 C5              READ20:  SEL RB0
000492 27                       CLR A
000493 B8FD                     MOV R0,#0FDH
000495 90                       MOVX @R0,A
000496 B47E                     CALL DELAY3
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
000498 2309          MOV A,#09H
00049A B8F9          MOV R0,#0F9H
00049C 90            MOVX @R0,A
00049D 2301          MOV A,#01H
00049F B8FD          MOV R0,#0FDH
0004A1 90            MOVX @R0,A
0004A2 B47E          CALL DELAY3
0004A4 B82E          MOV R0,#2EH
0004A6 F0            MOV A,@R0
0004A7 4390          ORL A,#90H
0004A9 B8FC          MOV R0,#0FCH
0004AB 90            MOVX @R0,A
0004AC B47E          CALL DELAY3
0004AE 53EF          ANL A,#0EFH
0004B0 B8FC          MOV R0,#0FCH
0004B2 90            MOVX @R0,A
0004B3 B489          CALL DELAYFORRW
0004B5 2320          MOV A,#20H
0004B7 B8FC          MOV R0,#0FCH
0004B9 90            MOVX @R0,A
0004BA 2308          MOV A,#08H
0004BC B8F9          MOV R0,#0F9H
0004BE 90            MOVX @R0,A
0004BF B47E          CALL DELAY3
0004C1 27            CLR A
0004C2 B8FC          MOV R0,#0FCH
0004C4 90            MOVX @R0,A
0004C5 D5            SEL RB1
0004C6 83            RET
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
00057E                          ORG 057EH
                                ;SUBPROGRAM DELAY3
00057E C5           DELAY3:     SEL RB0
00057F B80D                     MOV R0,#0DH
000581 B9FF         LOOP44:     MOV R1,#0FFH
000583 E983         LOOP30:     DJNZ R1,LOOP30
000585 E881                     DJNZ R0,LOOP44
000587 D5                       SEL RB1
000588 83                       RET

                                ;SUBPROGRAM DELAYFORRW
000589 C5           DELAYFORRW: SEL RB0
00058A B813                     MOV R0,#13H
00058C B9FF           LOOP31:   MOV R1,#0FFH
00058E BAFF           LOOP32:   MOV R2,#0FFH
000590 EA90           LOOP33:   DJNZ R2,LOOP33
000592 E98E                     DJNZ R1,LOOP32
000594 E88C                     DJNZ R0,LOOP31
000596 D5                       SEL RB1
000597 83                       RET

                                ;SUBPROGRAM CHECK
000598 C5           CHECK:      SEL RB0
000599 B901         START11:    MOV R1,#01H
00059B BA01         LOOP73:     MOV R2,#01H
00059D BB01         LOOP74:     MOV R3,#01H
00059F BC01         LOOP75:     MOV R4,#01H
0005A1 BD01         LOOP76:     MOV R5,#01H
0005A3 BE01         LOOP77:     MOV R6,#01H
0005A5 BF01         LOOP78:     MOV R7,#01H
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
0005A7 B8F8          LOOP79:     MOV R0,#0F8H
0005A9 80                        MOVX A,@R0
0005AA 5340                      ANL A,#40H
0005AC C6C1                      JZ DEC
0005AE B4D8                      CALL DELAYCHECK
0005B0 B833                      MOV R0,#33H
0005B2 10                        INC @R0
0005B3 F0                        MOV A,@R0
0005B4 D304                      XRL A,#04H
0005B6 C6D1                      JZ OFFHUCK
0005B8 27                        CLR A
0005B9 B8F9                      MOV R0,#0F9H
0005BB 90                        MOVX @R0,A
0005BC 2340                      MOV A,#40H
0005BE 90                        MOVX @R0,A
0005BF A4A7                      JMP LOOP79
0005C1 EFA7          DEC:        DJNZ R7,LOOP79
0005C3 EEA5                      DJNZ R6,LOOP78
0005C5 EDA3                      DJNZ R5,LOOP77
0005C7 ECA1                      DJNZ R4,LOOP76
0005C9 EB9F                      DJNZ R3,LOOP75
0005CB EA9D                      DJNZ R2,LOOP74
0005CD E99B                      DJNZ R1,LOOP73
0005CF A4D6                      JMP GOMAIN
0005D1 B8F9          OFFHUCK:    MOV R0,#0F9H
0005D3 2308                      MOV A,#08H
0005D5 90                        MOVX @R0,A
0005D6 D5            GOMAIN:     SEL RB1
0005D7 83                        RET
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                                        ;SUBPROGRAM DELAYCHECK
0005D8 C5           DELAYCHECK:     SEL RB0
0005D9 B809                         MOV R0,#09H
0005DB B9FF         LOOP80:         MOV R1,#0FFH
0005DD BAFF         LOOP81:         MOV R2,#0FFH
0005DF EADF         LOOP82:         DJNZ R2,LOOP82
0005E1 E9DD                         DJNZ R1,LOOP81
0005E3 E8DB                         DJNZ R0,LOOP80
0005E5 D5                           SEL RB1
0005E6 83                           RET
0005E7                  END
```

00000440	ANSWER	00000598	CHECK	000001D1	CYCLE1
000001D3	CYCLE2	000001D5	CYCLE3	000001E4	CYCLE4
000001E6	CYCLE5	000001E8	CYCLE6	000001EA	CYCLE7
000001F9	CYCLE8	000005C1	DEC	00000417	DELAY
000002EE	DELAY1	00000426	DELAY2	0000057E	DELAY3
000005D8	DELAYCHECK	00000589	DELAYFORRW	000001CE	DELAYFORTON
00000170	DETECTONE	00000162	END1	00000166	END10
000001B7	END3	000001C2	END4	000001CC	END5
000001FE	END7	00000276	END8	000002C9	END9
000005D6	GOMAIN	0000002D	LOOP1	0000021D	LOOP13
00000224	LOOP14	0000022B	LOOP15	00000263	LOOP16
00000297	LOOP17	0000029B	LOOP18	000002D4	LOOP20
000002D6	LOOP21	000002F1	LOOP22	000002F3	LOOP23
0000041A	LOOP24	0000041C	LOOP25	0000041E	LOOP26
00000429	LOOP27	0000042B	LOOP28	0000042D	LOOP29
00000583	LOOP30	0000058C	LOOP31	0000058E	LOOP32
00000590	LOOP33	00000108	LOOP4	00000581	LOOP44
0000010A	LOOP5	0000010C	LOOP6	00000135	LOOP7
0000008A	LOOP70	0000008C	LOOP71	0000008E	LOOP72

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

0000059B	LOOP73	0000059D	LOOP74	0000059F	LOOP75
000005A1	LOOP76	000005A3	LOOP77	000005A5	LOOP78
000005A7	LOOP79	00000137	LOOP8	000005DB	LOOP80
000005DD	LOOP81	000005DF	LOOP82	00000139	LOOP9
00000200	MODE	0000002B	NEXT1	00000040	NEXT2
00000074	NEXT3	00000076	NEXT4	000001C8	NEXT9
00000234	NEXTA	000002CD	NEXTC	00000236	NOTUSE
000005D1	OFFHUCK	00000270	OPEN	00000195	OVER
000002AF	OVER1	0000001D	READ	0000011D	READ1
00000148	READ2	00000491	READ20	0000045D	REC
00000242	RESET	0000007C	RINGING	00000278	SET
0000028B	SETRESET	00000400	SETSYS	000002CF	SONG
00000010	START	0000007D	START10	00000599	START11
00000259	STATUS	0000009F	STOP	00000300	TABLE
00000100	USERCODE				

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

8048AH/8035AHL/8049AH
8039AHL/8050AH/8040AHL
HMOS SINGLE-COMPONENT 8-BIT MICROCOMPUTER

- **High Performance HMOS II**
- **Interval Timer/Event Counter**
- **Two Single Level Interrupts**
- **Single 5-Volt Supply**
- **Over 96 Instructions; 90% Single Byte**
- **Reduced Power Consumption**
- **Compatible with 8080/8085 Peripherals**
- **Easily Expandable Memory and I/O**
- **Up to 1.36 μSec Instruction Cycle**
  All Instructions 1 or 2 cycles

The Intel MCS®-48 family are totally self-sufficient, 8-bit parallel computers fabricated on single silicon chips using Intel's advanced N-channel silicon gate HMOS process.

The family contains 27 I/O lines, an 8-bit timer/counter, and on-board oscillator/clock circuits. For systems that require extra capability, the family can be expanded using MCS®-80/MCS®-85 peripherals.

To minimize development problems and provide maximum flexibility, a logically and functionally pin-compatible version of the ROM devices with UV-erasable user-programmable EPROM program memory is available with minor differences.

These microcomputers are designed to be efficient controllers as well as arithmetic processors. They have extensive bit handling capability as well as facilities for both binary and BCD arithmetic. Efficient use of program memory results from an instruction set consisting mostly of single byte instructions and no instructions over 2 bytes in length.

Device	Internal Memory		RAM Standby
8050AH	4K × 8 ROM	256 × 8 RAM	yes
8049AH	2K × 8 ROM	128 × 8 RAM	yes
8048AH	1K × 8 ROM	64 × 8 RAM	yes
8040AHL	none	256 × 8 RAM	yes
8039AHL	none	128 × 8 RAM	yes
8035AHL	none	64 × 8 RAM	yes

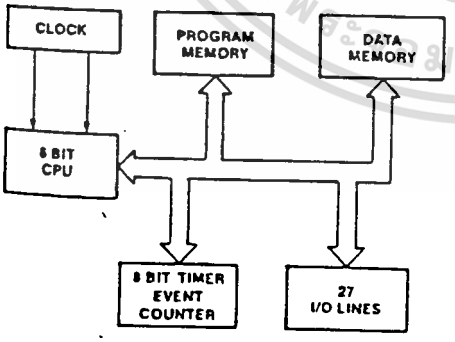


Figure 1.
Block Diagram

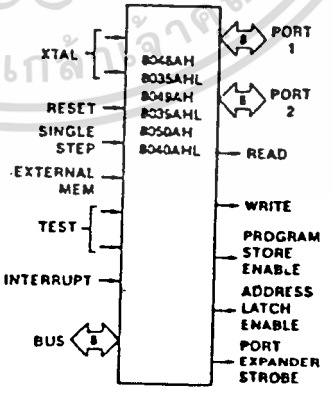


Figure 2.
Logic Symbol

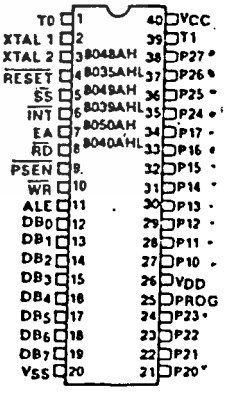


Figure 3.
Pin Configuration

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Table 1. Pin Description

Symbol	Pin No.	Function
V_{SS}	20	Circuit GND potential
V_{DD}	26	+5V during normal operation.
		Low power standby pin.
V_{CC}	40	Main power supply; +5V during operation.
PROG	25	Output strobe for 8243 I/O expander.
P10-P17 Port 1	27-34	8-bit quasi-bidirectional port.
P20-P23 P24-P27 Port 2	21-24 35-38	8-bit quasi-bidirectional port. P20-P23 contain the four high order program counter bits during an external program memory fetch and serve as a 4-bit I/O expander bus for 8243.
DB0-DB7 BUS	12-19	True bidirectional port which can be written or read synchronously using the $\overline{RD}$, $\overline{WR}$ strobes. The port can also be statically latched.
		Contains the 8 low order program counter bits during an external program memory fetch, and receives the addressed instruction under the control of $\overline{PSEN}$. Also contains the address and data during an external RAM data store instruction, under control of ALE, $\overline{RD}$, and $\overline{WR}$.
T0	1	Input pin testable using the conditional transfer instructions JT0 and JNT0. T0 can be designated as a clock output using ENT0 CLK instruction
T1	39	Input pin testable using the JT1, and JNT1 instructions. Can be designated the timer/counter input using the STRT CNT instruction.
$\overline{INT}$	6	Interrupt input. Initiates an interrupt if interrupt is enabled. Interrupt is disabled after a reset. Also testable with conditional jump instruction. (Active low) interrupt must remain low for at least 3 machine cycles for proper operation.

Symbol	Pin No.	Function
$\overline{RD}$	8	Output strobe activated during a BUS read. Can be used to enable data onto the bus from an external device. Used as a read strobe to external data memory. (Active low)
$\overline{RESET}$	4	Input which is used to initialize the processor. (Active low) (Non TTL V_{IH})
		Used during power down.
		Used during ROM verification.
$\overline{WR}$	10	Output strobe during a bus write. (Active low) Used as write strobe to external data memory.
ALE	11	Address latch enable. This signal occurs once during each cycle and is useful as a clock output. The negative edge of ALE strobes address into external data and program memory.
$\overline{PSEN}$	9	Program store enable. This output occurs only during a fetch to external program memory. (Active low)
$\overline{SS}$	5	Single step input can be used in conjunction with ALE to "single step" the processor through each instruction. (Active Low)
		Used in sync mode
EA	7	External access input which forces all program memory fetches to reference external memory. Useful for emulation and debug. (Active high)
		Used during ROM verification (12V)
XTAL1	2	One side of crystal input for internal oscillator. Also input for external source. (Non TTL V_{IH})
XTAL2	3	Other side of crystal input.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Table 2. Instruction Set

Accumulator

Mnemonic	Description	Bytes	Cycles
ADD A, R	Add register to A	1	1
ADD A, @R	Add data memory to A	1	1
ADD A, # data	Add immediate to A	2	2
ADDC A, R	Add register with carry	1	1
ADDC A, @R	Add data memory with carry	1	1
ADDC A, # data	Add immediate with carry	2	2
ANL A, R	And register to A	1	1
ANL A, @R	And data memory to A	1	1
ANL A, # data	And immediate to A	2	2
ORL A, R	Or register to A	1	1
ORL A @R	Or data memory to A	1	1
ORL A, # data	Or immediate to A	2	2
XRL A, R	Exclusive or register to A	1	1
XRL A, @R	Exclusive or data memory to A	1	1
XRL, A, # data	Exclusive or immediate to A	2	2
INC A	Increment A	1	1
DEC A	Decrement A	1	1
CLR A	Clear A	1	1
CPL A	Complement A	1	1
DA A	Decimal adjust A	1	1
SWAP A	Swap nibbles of A	1	1
RL A	Rotate A left	1	1
RLC A	Rotate A left through carry	1	1
RR A	Rotate A right	1	1
RRC A	Rotate A right through carry	1	1

Input/Output

Mnemonic	Description	Bytes	Cycles
IN A, P	Input port to A	1	2
OUTL P, A	Output A to port	1	2
ANL P, # data	And immediate to port	2	2
ORL P, # data	Or immediate to port	2	2
INS A, BUS	Input BUS to A	1	2
OUTL BUS, A	Output A to BUS	1	2
ANL BUS, # data	And immediate to BUS	2	2
ORL BUS, # data	Or immediate to BUS	2	2
MOVD A, P	Input expander port to A	1	2
MOVD P, A	Output A to expander port	1	2
ANLD P, A	And A to expander port	1	2
ORLD P, A	Or A to expander port	1	2

Registers

Mnemonic	Description	Bytes	Cycles
INC R	Increment register	1	1
INC @R	Increment data memory	1	1
DEC R	Decrement register	1	1

Branch

Mnemonic	Description	Bytes	Cycles
JMP addr	Jump unconditional	2	2
JMPP @A	Jump indirect	1	2
DJNZ R, addr	Decrement register and skip	2	2
JC addr	Jump on carry = 1	2	2
JNC addr	Jump on carry = 0	2	2
JZ addr	Jump on A zero	2	2
JNZ addr	Jump on A not zero	2	2
JT0 addr	Jump on T0 = 1	2	2
JNT0 addr	Jump on T0 = 0	2	2
JT1 addr	Jump on T1 = 1	2	2
JNT1 addr	Jump on T1 = 0	2	2
JF0 addr	Jump on F0 = 1	2	2
JF1 addr	Jump on F1 = 1	2	2
JTF addr	Jump on timer flag	2	2
JNI addr	Jump on INT = 0	2	2
JBb addr	Jump on accumulator bit	2	2

Subroutine

Mnemonic	Description	Bytes	Cycles
CALL addr	Jump to subroutine	2	2
RET	Return	1	2
RETR	Return and restore status	1	2

Flags

Mnemonic	Description	Bytes	Cycles
CLR C	Clear carry	1	1
CPL C	Complement carry	1	1
CLR F0	Clear flag 0	1	1
CPL F0	Complement flag 0	1	1
CLR F1	Clear flag 1	1	1
CPL F1	Complement flag 1	1	1

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Table 2. Instruction Set (Continued)

Data Moves

Mnemonic	Description	Bytes	Cycles
MOV A, R	Move register to A	1	1
MOV A, @R	Move data memory to A	1	1
MOV A, # data	Move immediate to A	2	2
MOV R, A	Move A to register	1	1
MOV @R, A	Move A to data memory	1	1
MOV R, # data	Move immediate to register	2	2
MOV @R, # data	Move immediate to data memory	2	2
MOV A, PSW	Move PSW to A	1	1
MOV PSW, A	Move A to PSW	1	1
XCH A, R	Exchange A and register	1	1
XCH A, @R	Exchange A and data memory	1	1
XCHD A, @R	Exchange nibble of A and register	1	1
MOVX A, @R	Move external data memory to A	1	2
MOVX @R, A	Move A to external data memory	1	2
MOVP A, @A	Move to A from current page	1	2
MOVP3 A, @	Move to A from page 3	1	2

Timer/Counter

Mnemonic	Description	Bytes	Cycles
MOV A, T	Read timer/counter	1	1
MOV T, A	Load timer/counter	1	1
STRT T	Start timer	1	1
STRT CNT	Start timer	1	1
STOP TCNT	Stop timer/counter	1	1
EN TCNTI	Enable timer/counter interrupt	1	1
DIS TCNTI	Disable timer/counter interrupt	1	1

Control

Mnemonic	Description	Bytes	Cycles
EN I	Enable external interrupt	1	1
DIS I	Disable external interrupt	1	1
SEL RB0	Select register bank 0	1	1
SEL RB1	Select register bank 1	1	1
SEL MB0	Select memory bank 0	1	1
SEL MB1	Select memory bank 1	1	1
ENT0 CLK	Enable clock output on T0	1	1

Mnemonic	Description	Bytes	Cycles
NOP	No operation	1	1
IDL	Select Idle Operation	1	1

ABSOLUTE MAXIMUM RATINGS*

Ambient Temperature Under Bias . . . 0°C to 70°C
Storage Temperature . . . . . -65°C to +150°C
Voltage On Any Pin With Respect to Ground . . . . . -0.5V to +7V
Power Dissipation . . . . . 1.5 Watt

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

D.C. CHARACTERISTICS: (T_A = 0°C to 70°C; V_{CC} = V_{DD} = 5V ± 10%; V_{SS} = 0V)

Symbol	Parameter	Limits Min	Typ	Max	Unit	Test Conditions	Device
V_{IL}	Input Low Voltage (All Except RESET, X1, X2)	-.5		.8	V		All
V_{IL1}	Input Low Voltage (RESET, X1, X2)	-.5		.6	V		All
V_{IH}	Input High Voltage (All Except XTAL1, XTAL2, RESET)	2.0		V_{CC}	V		All
V_{IH1}	Input High Voltage (X1, X2, RESET)	3.8		V_{CC}	V		All
V_{OL}	Output Low Voltage (BUS)			.45	V	I_{OL} = 2.0 mA	All
V_{OL1}	Output Low Voltage (RD, WR, PSEN, ALE)			.45	V	I_{OL} = 1.8 mA	All
V_{OL2}	Output Low Voltage (PROG)			.45	V	I_{OL} = 1.0 mA	All
V_{OL3}	Output Low Voltage (All Other Outputs)			.45	V	I_{OL} = 1.6 mA	All
V_{OH}	Output High Voltage (BUS)	2.4			V	I_{OH} = -400 μA	All
V_{OH1}	Output High Voltage (RD, WR, PSEN, ALE)	2.4			V	I_{OH} = -100 μA	All
V_{OH2}	Output High Voltage (All Other Outputs)	2.4			V	I_{OH} = -40 μA	All

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

D.C. CHARACTERISTICS: (T_A = 0°C to 70°C; V_{CC} = V_{DD} = 5V ± 10%; V_{SS} = 0V) (Continued)

Symbol	Parameter	Limits			Unit	Test Conditions	Device
		Min	Typ	Max			
I_{L1}	Leakage Current (T1, $\overline{INT}$)			±10	μA	$V_{SS} \leq V_{IN} \leq V_{CC}$	All
I_{LI1}	Input Leakage Current (P10-P17, P20-P27, EA, $\overline{SS}$)			−500	μA	$V_{SS}+.45 \leq V_{IN} \leq V_{CC}$	All
I_{LI2}	Input Leakage Current $\overline{RESET}$	20		300	μA	$V_{SS} \leq V_{IN} \leq 3.8V$	All
I_{L0}	Leakage Current (BUS, T0) (High Impedance State			±10	μA	$V_{SS} \leq V_{IN} \leq V_{CC}$	All
I_{DD}	V_{DD} Supply Current (RAM Standby)		3	5	mA		8048AH 8035AHL
			4	7	mA		8049AH 8039AHL
			5	10	mA		8050AH 8040AHL
$I_{DD}+I_{CC}$	Total Supply Current*		30	65	mA		8048AH 8035AHL
			35	70	mA		8049AH 8039AHL
			40	80	mA		8050AH 8040AHL
V_{DD}	RAM Standby Voltage	2.2		5.5	V	Standby Mode Reset $\leq V_{IL1}$	All

*$I_{CC}+I_{DD}$ is measured with all outputs disconnected; $\overline{SS}$, $\overline{RESET}$, and $\overline{INT}$ equal to V_{CC}; EA equal to V_{SS}.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

A.C. CHARACTERISTICS: (T_A = 0°C to 70°C; V_{CC} = V_{DD} = 5V ± 10%; V_{SS} = 0V)

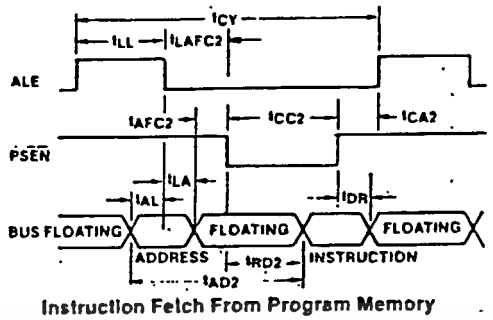
Symbol	Parameter	f (t) (Note 3)	11 MHz		Unit	Conditions (Note 1)
			Min	Max		
t	Clock Period	1/xtal freq	90.9	1000	ns	(Note 3)
t_{LL}	ALE Pulse Width	3.5t-170	150		ns	
t_{AL}	Addr Setup to ALE	2t-110	70		ns	(Note 2)
t_{LA}	Addr Hold from ALE	t-40	50		ns	
t_{CC1}	Control Pulse Width ($\overline{RD}$, $\overline{WR}$)	7.5t-200	480		ns	
t_{CC2}	Control Pulse Width ($\overline{PSEN}$)	6t-200	350		ns	
t_{DW}	Data Setup before $\overline{WR}$	6.5t-200	390		ns	
t_{WD}	Data Hold after $\overline{WR}$	t-50	40		ns	
t_{DR}	Data Hold ($\overline{RD}$, $\overline{PSEN}$)	1.5t-30	0	110	ns	
t_{RD1}	$\overline{RD}$ to Data in	6t-170		375	ns	
t_{RD2}	$\overline{PSEN}$ to Data in	4.5t-170		240	ns	
t_{AW}	Addr Setup to $\overline{WR}$	5t-150	300		ns	
t_{AD1}	Addr Setup to Data ($\overline{RD}$)	10.5t-220		730	ns	
t_{AD2}	Addr Setup to Data ($\overline{PSEN}$)	7.5t-200		460	ns	
t_{AFC1}	Addr Float to $\overline{RD}$, $\overline{WR}$	2t-40	140		ns	(Note 2)
t_{AFC2}	Addr Float to $\overline{PSEN}$	.5t-40	10		ns	(Note 2)
t_{LAFC1}	ALE to Control ($\overline{RD}$, $\overline{WR}$)	3t-75	200		ns	
t_{LAFC2}	ALE to Control ($\overline{PSEN}$)	1.5t-75	60		ns	
t_{CA1}	Control to ALE ($\overline{RD}$, $\overline{WR}$, $\overline{PROG}$)	t-65	25		ns	
t_{CA2}	Control to ALE ($\overline{PSEN}$)	4t-70	290		ns	
t_{CP}	Port Control Setup to $\overline{PROG}$	1.5t-80	50		ns	
t_{PC}	Port Control Hold to $\overline{PROG}$	4t-260	100		ns	
t_{PR}	$\overline{PROG}$ to P2 Input Valid	8.5t-120		650	ns	
t_{PF}	Input Data Hold from $\overline{PROG}$	1.5t	0	140	ns	
t_{DP}	Output Data Setup	6t-290	250		ns	
t_{PD}	Output Data Hold	1.5t-90	40		ns	
t_{PP}	$\overline{PROG}$ Pulse Width	10.5t-250	700		ns	
t_{PL}	Port 2 I/O Setup to ALE	4t-200	160		ns	
t_{LP}	Port 2 I/O Hold to ALE	.5t-30	15		ns	
t_{PV}	Port Output from ALE	4.5t+100		510	ns	
t_{0PRR}	T0 Rep Rate	3t	270		ns	
t_{CY}	Cycle Time	15t	1.36	15.0	μs	

Notes:

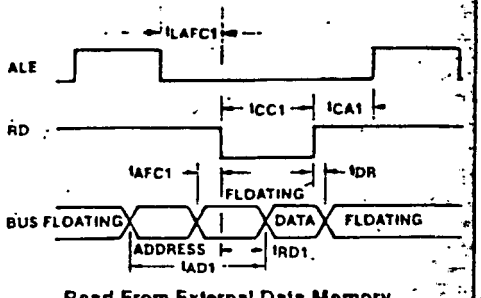
1. Control Outputs CL = 80pF
   BUS Outputs CL = 150pF
2. BUS High Impedance Load 20pF.
3. f(t) assumes 50% duty cycle on X1, X2. Max clock period is for a 1 MHz crystal input.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

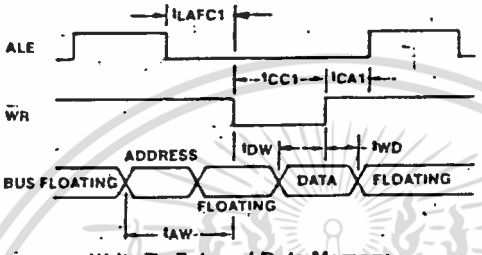
WAVEFORMS



Instruction Fetch From Program Memory



Read From External Data Memory

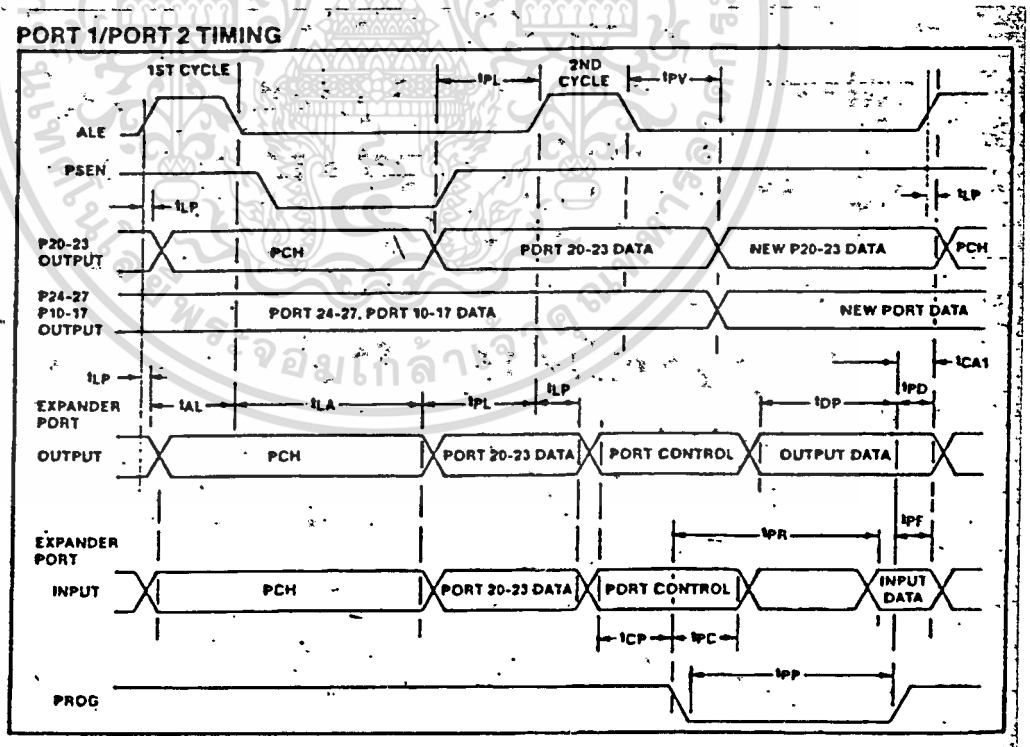


Write To External Data Memory

A.C. testing inputs are driven at 2.4V for a logic "1" and 0.45V for a logic "0." Output timing measurements are made at 2.0V for a logic "1" and 0.8V for a logic "0."

Input And Output For A.C. Tests

PORT 1/PORT 2 TIMING



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้