

เครื่องแสดงข้อมูลการใช้งานโทรศัพท์

Telephone usage reporting



โดย
นาย สุรัชย์ แก้วสุพรรณ
นาย อัครนัย โพธิ์ไทย

เลขหมู่.....
เลขทะเบียน.....42205
วัน, เดือน, ปี 5 พ.ค. 2545

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

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต

สาขาวิชาวิศวกรรมโทรคมนาคม

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2543

เครื่องแสดงข้อมูลการใช้งานโทรศัพท์

Telephone usage reporting

โดย

นาย สุรัชย์ แก้วสุพรรณ 41013035

นาย อัครนัย โพธิ์ไทย 41013038

อาจารย์ที่ปรึกษา

รศ.ดร. สุวิพล ลิทธิชีวกาศ

ผศ.เกรียงไกร วงศ์โรจนภรณ์

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต

สาขาวิชาวิศวกรรมโทรคมนาคม

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2543

ปริญญานิพนธ์ปีการศึกษา 2543

ภาควิชาวิศวกรรมโทรคมนาคม

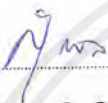
คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เรื่อง เครื่องแสดงข้อมูลการใช้งานโทรศัพท์

Telephone usage reporting

ผู้จัดทำ

1. นาย สุรชัย แก้วสุพรรณ 41013035
2. นาย อัสนัย โพธิ์ไทย 41013038


(รศ.ดร.สุวิมล ลิทธิชีวภาค) อาจารย์ที่ปรึกษา


(ผศ.เกรียงไกร วงศ์โรจนภรณ์) อาจารย์ที่ปรึกษา



เครื่องแสดงข้อมูลการใช้งานโทรศัพท์

Telephone usage reporting

โดย นาย สุรัชย์ แก้วสุพรรณ 41013035

นาย อศนัย โพธิ์ไทย 41013038

อาจารย์ที่ปรึกษา รศ.ดร.สุวิมล สิริธิชัยภัก

ผศ.เกรียงไกร วงศ์โรจนภรณ์

บทคัดย่อ

เครื่องแสดงข้อมูลการใช้งานโทรศัพท์นี้จะทำหน้าที่บันทึกการใช้โทรศัพท์ และสามารถตรวจสอบรายชื่อของผู้ใช้เลขหมายปลายทางที่เราทำการเรียกไป โดยรายชื่อของเบอร์ปลายทางจะถูกเก็บในรูปแบบของ Database และใน Database นี้จะต้องมีข้อมูลของชื่อ ที่อยู่ และเบอร์โทรศัพท์เก็บไว้เรียบร้อยแล้ว ข้อมูลของการใช้งานก็จะถูกเก็บเป็น Database เพื่อที่จะแสดงรายการ ซึ่งโครงการนี้เป็นการประยุกต์นำเอาการสื่อสารทางโทรศัพท์และการสืบค้นฐานข้อมูลมาประยุกต์ใช้งานพร้อมทั้งยังเป็นแนวทางในการประยุกต์ใช้งานโดยใช้ Visual Basic และการเขียนโปรแกรมโดยใช้ Visual Basic ด้านอื่นๆ ได้อีกในอนาคต

Abstract

The objective of " Telephone usage reporting " Project is to develop which can be used for store data of telephone usage and it has ability to show data user called its data is kept in database. Which much be has address field, name field and telephone number field. The data ready to reporting it keep in database too. This project is applied telephone communication and Visual Basic programming with database we wish that the idea of this can be applied to another project and actually work in the real work.

กิตติกรรมประกาศ

ปริญญานิพนธ์ฉบับนี้ สำเร็จลุล่วงไปด้วยดี โดยได้รับการสนับสนุนความช่วยเหลือ การแนะนำ การให้คำปรึกษาจากคณาจารย์ประจำภาควิชาวิศวกรรมโทรคมนาคมทุกท่านโดยเฉพาะอย่างยิ่งจาก อาจารย์ สุวิพล สิริชีวะภาค และอาจารย์ เกรียงไกร วงศ์โรจนภรณ์ ซึ่งเป็นอาจารย์ที่ปรึกษาของโครงการ ขอขอบพระคุณอาจารย์ทุกท่านที่ให้ความช่วยเหลือและคำปรึกษาทำให้ปริญญานิพนธ์ฉบับนี้เสร็จสมบูรณ์ และที่สำคัญที่สุด ขอกราบขอบพระคุณ คุณพ่อ และคุณแม่ ผู้บังเกิดเกล้าผู้เป็นแรงสนับสนุน อันยิ่งใหญ่ทั้งกำลังใจ กำลังทรัพย์และเป็นผู้ให้ตลอดมา



	หน้า
บทที่ 1 บทนำ	1
1.1 ความเป็นมาของโครงการ	1
1.2 วัตถุประสงค์ของโครงการ	1
1.3 เป้าหมายในการทำโครงการ	1
1.4 แนวคิดของโครงการ	2
บทที่ 2 ทฤษฎีหรือหลักการ	3
2.1 หน้าที่หลักของโทรศัพท์	3
2.2 ภาคถอดรหัสสัญญาณความถี่ DTMF	9
2.3 ส่วนควบคุมการทำงาน	15
2.4 การเชื่อมโยง MCS-51 กับหน่วยความจำ	17
2.5 มาตรฐานพอร์ตอนุกรมแบบ RS-232	24
2.6 Visual Basic เบื้องต้น	26
บทที่ 3 การคำนวณและการสร้าง	38
3.1 ส่วนของวงจร ดีเทคเตอร์	38
3.2 ส่วนควบคุมการทำงาน	44
3.3 ส่วนแสดงและประมวลผล	57
บทที่ 4 การทดลองและผลการทดลอง	68
4.1 วงจรตรวจสอบการยกหู	68
4.2 วงจรตรวจสอบสัญญาณเรียกกลับ	69
4.3 การทดลองและผลการทดลองในส่วนแสดงผลและประมวลผล	73
บทที่ 5 บทสรุปและวิจารณ์	76
5.1 ปัญหาและแนวทางการแก้ไข	76

รูปที่ 1.1 บล็อกไดอะแกรมโครงสร้างของโครงการ	2
รูปที่ 2.1 สัญญาณให้หมุน	4
รูปที่ 2.2 สัญญาณไม่ว่า	4
รูปที่ 2.3 สัญญาณเรียกกลับ	5
รูปที่ 2.4 สัญญาณกริ่งเรียก	5
รูปที่ 2.5 แสดงหน้าปัดมิเตอร์สวิตช์แบบกดปุ่มและความถี่ที่ใช้	6
รูปที่ 2.6 ลักษณะของสัญญาณเมื่อผู้เรียกเรียกเข้ามา	6
รูปที่ 2.7 ลักษณะทางไฟฟ้าที่ปรากฏที่คู่สายขณะทำการเรียก	7
รูปที่ 2.8 ผังการทำงานของโทรศัพท์	8
รูปที่ 2.9 โครงสร้างภายในของ MT 8870	9
รูปที่ 2.10 ความถี่ที่ได้จากภาคกรองความถี่	10
รูปที่ 2.11 วงจรตรวจสอบสัญญาณอย่างง่าย	11
รูปที่ 2.12 ผังเวลาของไอซี MT 8870	11
รูปที่ 2.13 การต่อวงจรภาคอินพุท	13
รูปที่ 2.14 การต่อวงจรผลิตความถี่	13
รูปที่ 2.15 วงจรใช้งานเบื้องของไอซี MT8870	14
รูปที่ 2.16 แสดงโครงสร้าง พอร์ต 0 (บิต)	15
รูปที่ 2.17 โครงสร้างของพอร์ต 1 (บิต)	16
รูปที่ 2.18 โครงสร้างของพอร์ต 2 (บิต)	16
รูปที่ 2.19 โครงสร้างของพอร์ต 3 (บิต)	17
รูปที่ 2.20 การจัดวางขาของ Static RAM เบอร์ TMS4016	18
รูปที่ 2.21 ผังเวลาของ TMS4016	19
รูปที่ 2.22 การจัดวางขาของ TMS2732A	20
รูปที่ 2.23 ผังเวลาในการอ่านและโปรแกรมของ TMS2732A	21
รูปที่ 2.24 การจัดวางขาและตารางความจริงของ 74LS138	22
รูปที่ 2.25 วงจรถอดรหัสสำหรับหน่วยความจำตาม ตารางที่ 2.4	23
รูปที่ 2.26 การต่อ MCS-51 เข้ากับ EPROM และ RAM	23
รูปที่ 2.27 การจัดขาของคอนเน็กเตอร์พอร์ตอนุกรมตามมาตรฐาน RS-232	25
รูปที่ 2.28 การต่ออุปกรณ์ภายนอกกับพอร์ตอนุกรมของคอมพิวเตอร์ในลักษณะต่างๆ	25
รูปที่ 2.29 ไดอะแกรมการทำงานภายในของพอร์ตอนุกรมของเครื่องคอมพิวเตอร์	28
รูปที่ 2.30 หน้าจอสำหรับเปิด Project ใหม่	29

รูปที่ 2.31 รูปแสดงจอภาพของ Visual basic	30
รูปที่ 2.32 แสดงการเลือก control ต่างๆที่ต้องการใช้งาน	31
รูปที่ 2.33 แถบเครื่องมือที่ประกอบด้วย Icon ต่างๆ	32
รูปที่ 2.34 แสดงตัวอย่างการเขียน From และการดึง Control มาใช้งาน	34
รูปที่ 2.35 แสดงผลโปรแกรมทดสอบการทำงานติดต่อกับพอร์ทอนุกรม	36
รูปที่ 3.1 บล็อกไดอะแกรมแสดงส่วนประกอบของโครงการ	38
รูปที่ 3.2 แสดงวงจรตรวจสอบการยกหูโทรศัพท์	39
รูปที่ 3.3 แสดงวงจรตรวจสอบสัญญาณเรียกกลับ	40
รูปที่ 3.4 วงจรถอดรหัสหมายเลขโทรศัพท์	41
รูปที่ 3.5 วงจรดีเทกเตอร์	42
รูปที่ 3.6 ลายวงจรของวงจรดีเทกเตอร์	43
รูปที่ 3.7 แสดงบล็อกไดอะแกรมของส่วนควบคุมการทำงาน	44
รูปที่ 3.8 แสดงวงจรส่วนควบคุมและหน่วยความจำ	45
รูปที่ 3.9 แผนผังแสดงการทำงานการรับข้อมูลจากวงจรดีเทกเตอร์	47
รูปที่ 3.10 แสดง Microsoft Comm Control	58
รูปที่ 3.11 แผนผังแสดงการทำงานการแสดงผลและประมวลผล	59
รูปที่ 4.1 แสดงรูปสัญญาณที่ได้จากวงจรตรวจสอบการยกหูโทรศัพท์	68
รูปที่ 4.2 แสดงรูปสัญญาณที่ได้จากเอาต์พุตของ IC LM 567	70
รูปที่ 4.3 แสดงรูปสัญญาณที่ได้จากเอาต์พุตของ IC 741	71
รูปที่ 4.4 แสดงรูปสัญญาณที่ได้จากเอาต์พุตของวงจรตรวจสอบสัญญาณเรียกกลับ	72
รูปที่ 4.5 แสดงรูปแบบเมื่อทำการ RUN โปรแกรม	73
รูปที่ 4.6 แสดงข้อมูลในฐานข้อมูลของ Microsoft Access	74
รูปที่ 4.7 แสดงข้อมูลต่างๆ ที่ได้จากผลการทดลอง	75

สารบัญตาราง

	หน้า
ตารางที่ 2.1 ความถี่มอดคูลเลตเมื่อกดหมายเลข	3
ตารางที่ 2.2 แสดงค่าถอดรหัสที่ได้จากควมถี่ต่างๆ	10
ตารางที่ 2.3 แสดงรายละเอียดการจัดวางขาของ EPROM ตระกูล 74xx	20
ตารางที่ 2.4 ตารางการถอดรหัสหน่วยความจำ	22
ตารางที่ 2.5 แสดงรายละเอียดการใช้งาน Toolbar ต่างๆของ Visual Basic	33



บทที่ 1

บทนำ

1.1 ความเป็นมาของโครงการ

ปัจจุบันเป็นยุคแห่งการติดต่อสื่อสารแลกเปลี่ยนข้อมูล การสื่อสารจึงมีความสำคัญอย่างมากในชีวิตประจำวัน “โทรศัพท์” เป็นการสื่อสารชนิดหนึ่งที่นิยมใช้กันอย่างแพร่หลายนั้น ก็เพราะเป็นการสื่อสารที่มีความสะดวกสบาย รวดเร็ว ปัจจุบันนี้ได้มีการนำเอาสัญญาณคู่สายโทรศัพท์มาประยุกต์ใช้ในด้านต่างๆมากมายตัวอย่างเช่น เป็นเครื่องควบคุมอุปกรณ์เครื่องมือเครื่องใช้ต่างๆ

ในการใช้โทรศัพท์ติดต่อกับลูกค้าคงจะคิดว่าสามารถรู้เบอร์หรือชื่อผู้ปลายทาง จะทำให้เรารู้ว่าเราโทรออกไปหาใครตอนเวลาเท่าไรและวันที่เท่าใด รายละเอียดของเบอร์ปลายทางเป็นสิ่งที่เราไม่สามารถรู้ได้เนื่องจากทางองค์การ โทรศัพท์ไม่ได้บอกเอาไว้ในใบเสร็จ(สำหรับเบอร์ในรหัสเดียวกัน)แต่ถ้าเรามีข้อมูลของลูกค้าเราก็นำมาใช้ได้ในการสืบค้น ปัจจุบันข้อมูลขององค์กร หรือ บริษัทส่วนมากจะถูกเก็บไว้ในรูปแบบของ Database เพื่อที่จะมีการนำมาใช้งานได้อย่างง่ายดาย แม้ในสถาบันการศึกษาการที่จะเก็บรายชื่อหรือประวัติของนักเรียนนักศึกษาจะมีการเก็บข้อมูลในรูปแบบของฐานข้อมูล

จากที่กล่าวผ่านมามีทำให้เราได้มีการประยุกต์นำเอาการสื่อสารโดยโทรศัพท์มาประยุกต์ใช้กับงานทางด้านข้อมูลที่เป็น Database เพื่อที่จะสามารถรู้รายละเอียดของการใช้โทรศัพท์ อาทิเช่น เบอร์ปลายทาง เวลาใช้งาน วัน/เดือน/ปี ที่ใช้งาน พร้อมทั้งคำนวณค่าโทรศัพท์

1.2 วัตถุประสงค์ของโครงการ

1. ศึกษาระบบการทำงานของโทรศัพท์
2. ออกแบบและสร้างวงจรเพื่อทดสอบสัญญาณโทรศัพท์ได้
3. ศึกษาการอินเตอร์เฟสกับคอมพิวเตอร์ผ่านพอร์ทอนุกรม
4. ออกแบบและสร้างวงจรเพื่ออินเตอร์เฟสกับคอมพิวเตอร์ผ่านพอร์ทอนุกรมได้
5. เขียนโปรแกรมเพื่อรับข้อมูลที่ได้มาจากวงจรได้
6. สร้างฐานข้อมูลที่ใช้ในการสืบค้นได้
7. เขียนโปรแกรมเพื่อสืบค้นข้อมูล(เบอร์โทรศัพท์)จากฐานข้อมูลได้
8. เขียนโปรแกรมเพื่อควบคุมการทำงานทั้งหมด

1.3 เป้าหมายในการทำโครงการ

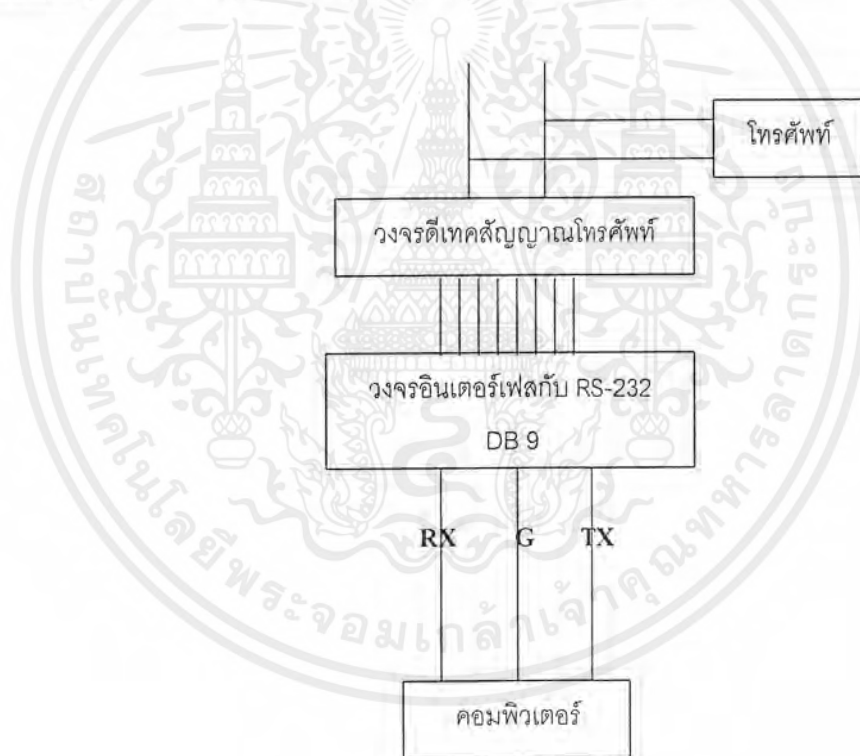
ในการทำโครงการชิ้นนี้ เป็นการออกแบบและสร้างโดยใช้งานได้จริง โดยทำการตรวจจับสัญญาณที่ได้มาจากการกดคีย์โทรศัพท์แต่ข้อมูลนั้นเป็นแบบขนาบจึงเปลี่ยนเป็นอนุกรมโดยวงจรอินเตอร์เฟสและใช้โปรแกรมเพื่อสืบค้นกับฐานข้อมูลและทำการคำนวณการใช้งานพร้อมทั้งแสดงผลออกมาทางจอ Monitor อีกทั้งยังสามารถแสดงผลออกทางเครื่องพิมพ์ได้ด้วย

แต่สิ่งที่คิดว่าจะได้รับนอกเหนือไปจากนั้นและเป็นสิ่งสำคัญในการทำโครงการนี้คือทำให้มีความรู้ด้านต่างๆที่ไม่เคยรู้มาก่อนเพิ่มขึ้นอีกมากมายทั้งในด้านการอินเตอร์เฟสกับอุปกรณ์ I/O ผ่านพอร์ทอนุกรมและที่สำคัญอย่างยิ่งคือการได้ศึกษาการพัฒนาโปรแกรมโดยใช้ Visual Basic ซึ่งกำลังเป็นที่

พอร์ทอนุกรมและที่สำคัญอย่างยิ่งคือ การได้ศึกษาการพัฒนาโปรแกรมโดยใช้ Visual Basic ซึ่งกำลังเป็นที่นิยมในปัจจุบันนี้เนื่องจากมีความสามารถที่สูงและศึกษาทำความเข้าใจได้ง่ายและมีหนังสือให้ค้นคว้ามากมายและสามารถที่จะนำไปประยุกต์ในการใช้งานด้านอื่นๆได้

1.4 แนวคิดของโครงการ

จากสถานะของกลุ่มสายโทรศัพท์ทำให้เราสามารถที่จะดีเทคเอาสถานะเหล่านั้นมาใช้งานได้ ทำอย่างไรที่จะทำให้คอมพิวเตอร์รับรู้ข้อมูลที่อ่านมาได้ เราได้ทำการสร้างวงจรอินเตอร์เฟส เพื่อให้การติดต่อกับอุปกรณ์ภายนอก ผ่าน RS-232 ได้ ส่วนของ Database เราได้ทำการสร้างขึ้นมาใช้งานโดยประกอบไปด้วย Record ของเบอร์โทรศัพท์ ชื่อและที่อยู่ สุดท้ายส่วนของโปรแกรม เราได้ทำการศึกษาความเป็นไปได้ว่าจะใช้ภาษาอะไรในการเขียนโปรแกรมเพื่อที่จะทำให้ตัวโปรแกรมมีเสถียรภาพที่ดี เช่น มีการแสดงผลที่สวยงาม ง่ายต่อการศึกษาและที่สำคัญควรจะเป็นโปรแกรมที่สามารถพัฒนาบน Windows ได้ เพราะว่าปัจจุบัน Windows เป็นที่นิยมใช้กันอย่างแพร่หลายและคุณสมบัติของ Windows ที่เป็น Multitasking จึงได้ทำการเลือกใช้ Visual Basic ในการเขียนโปรแกรม



รูปที่ 1.1 บล็อกไดอะแกรม โครงสร้างของโครงการ

บทที่ 2

ทฤษฎีหรือหลักการ

2.1 หน้าที่หลักของโทรศัพท์

1. เครื่องโทรศัพท์จะรับรู้ว่ามีผู้ต้องการใช้โทรศัพท์ เมื่อมีการยกหูโทรศัพท์ขึ้น
2. เครื่องโทรศัพท์จะส่งสัญญาณที่เรียกว่า “สัญญาณหมุน” บอกว่าพร้อมที่จะทำการกดหมายเลขที่จะติดต่อได้ ซึ่งก็คือ เสียงที่ได้ยินเวลาที่ยกหู เป็นสัญญาณเสียงที่มีความถี่ 350 Hz กับ 440 Hz มอดูเลตรวมกัน
3. เครื่องโทรศัพท์จะทำหน้าที่ ส่งรหัสหมายเลขที่ผู้เรียกจะติดต่อด้วย ไปยังชุมสายที่ควบคุม
4. เครื่องโทรศัพท์จะส่งสัญญาณบอกผู้เรียกว่าหมายเลขที่ต้องติดต่อด้วยว่าว่างหรือไม่ถ้าว่างก็จะส่งสัญญาณตอบกลับเป็นความถี่ 440 Hz โดยจะดัง 2 วินาที แล้วเงียบ 4 วินาที สลับกันไป แต่ถ้าหมายเลขที่เรียกไม่ว่าง ก็จะส่งสัญญาณความถี่ 480 Hz กับ 620 Hz มอดูเลตกันมา
5. สามารถเปลี่ยนพลังงานเสียงเป็นพลังงานไฟฟ้าและเปลี่ยนจากพลังงานไฟฟ้ามาเป็นพลังงานเสียง
6. โทรศัพท์จะปรับระดับแรงดันอย่างอัตโนมัติ ในกรณีที่เกิดการเปลี่ยนแปลงของแรงดันขึ้น
7. เครื่องโทรศัพท์จะส่งสัญญาณไปยังชุมสาย เพื่อแจ้งให้ทราบว่า สิ้นสุดการใช้งานแล้ว และให้ชุมสายทำการเลิกติดต่อกับอีกฝ่ายได้

ความถี่ (Hz)	รหัสหรือหมายเลข		
697	1	2	3
770	4	5	6
852	7	8	9
941	*	0	#
ความถี่ (Hz)	1209	1336	1477

ตารางที่ 2.1 ความถี่มอดูเลตเมื่อกดหมายเลข

2.1.1 การสนทนา

เมื่อปลายสายหรือผู้เรียก หรือผู้ถูกเรียก ยกหูโทรศัพท์ขึ้น การทำงานในส่วนวงจรควบคุมของชุมสายโทรศัพท์ก็จะหยุด เพื่อที่จะรอทำงานให้กับผู้อื่นที่จะเรียกเข้ามาต่อไป แต่หน้าที่ของชุมสายสำหรับตอนนี้ก็คือ การทำงานของมิเตอร์สำหรับเรียกเก็บค่าบริการภายหลัง

ในระหว่างที่ทำการสนทนาอยู่ เครื่องโทรศัพท์ก็จะทำงาน 2 รูปแบบไปพร้อม ๆ กัน คือ แปลงสัญญาณไฟฟ้าให้เป็นสัญญาณเสียง (Acoustic Energy) ซึ่งจะเรียกว่า รูปแบบการรับสัญญาณ

(Receiver Mode) และในทางกลับกัน รูปแบบที่ทำหน้าที่แปลงจากสัญญาณเสียงเป็นสัญญาณไฟฟ้า จะเรียกว่า รูปแบบการส่งสัญญาณ (Transmitter Mode) ในรูปแบบหลังนี้เอง ที่มีเรื่องของการป้อนกลับของสัญญาณเข้ามาเกี่ยวข้อง ก็คือ การที่ผู้พูดสามารถได้ยินเสียงตนเองจากหูฟังด้วย เรียกเสียงนี้ว่า Side Tone ซึ่งจำเป็นอย่างมากที่จะต้องป้อนกลับมา เพราะไม่เช่นนั้นก็ไม่สามารถรู้ได้เลยว่า ควรพูดให้มีเสียงดังในระดับใดจึงจะพอเหมาะที่คู่สนทนาจะได้ยินเสียงพูดของผู้เรียกได้อย่างชัดเจน

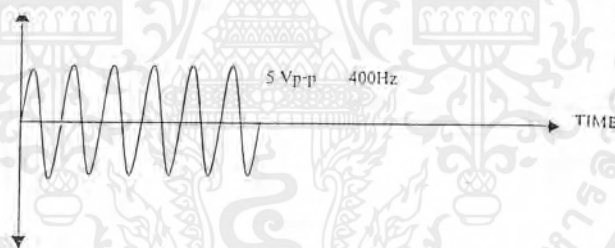
เมื่อสิ้นสุดการสนทนาทั้ง 2 ฝ่าย และวางหูโทรศัพท์ลง สัญญาณจากสวิตช์ก็จะบอกให้ชุมสายทำการเปิดวงจรที่ทำการติดต่ออยู่ออก อุปกรณ์ต่าง ๆ ก็จะว่าง และพร้อมสำหรับการติดต่อครั้งต่อไป

2.1.2 สัญญาณต่าง ๆ ในระบบชุมสายโทรศัพท์

สัญญาณต่าง ๆ ในระบบชุมสายโทรศัพท์ที่ส่งมายังสมาชิกผู้ใช้ (Subscriber) เป็นการบอกสถานะการติดต่อของอุปกรณ์ส่วนต่าง ๆ ในระบบโทรศัพท์ และแจ้งให้ผู้ใช้ทราบว่าควรทำอย่างไร เมื่อได้รับสัญญาณแต่ละชนิด รายละเอียดของสัญญาณต่าง ๆ มีดังนี้

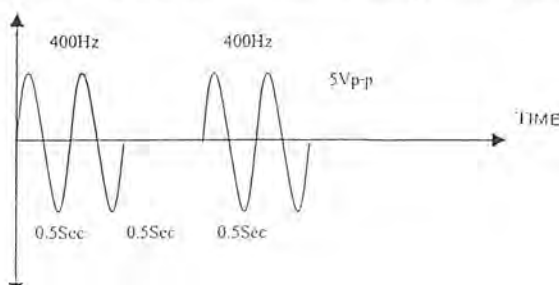
สัญญาณสมาชิก คือ สัญญาณที่เครื่องชุมสายโทรศัพท์จะแจ้งสภาวะต่าง ๆ ในการติดต่อให้ผู้เรียกทราบว่าควรทำอย่างไรเมื่อได้รับสัญญาณ สัญญาณสมาชิกประกอบด้วย

1. สัญญาณให้หมุน (DT : Dial Tone) สัญญาณให้หมุนนี้ เป็นสัญญาณแบบคลื่นไซน์ที่มีความถี่ 400 เฮิรตซ์ ส่งมาอย่างต่อเนื่อง และมีระดับขนาด 400 มิลลิโวลต์ พิกทูปิก ดังรูปที่ 2.1



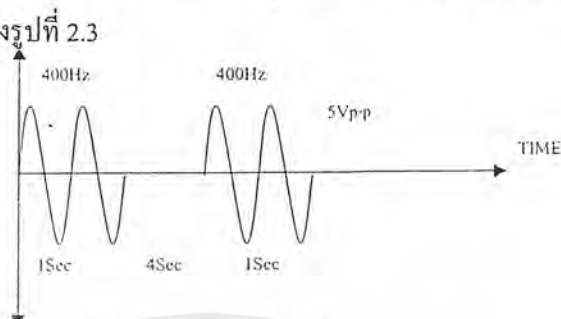
รูปที่ 2.1 สัญญาณให้หมุน

2. สัญญาณไม่ว่าง (BT : Busy Tone) เป็นสัญญาณเพื่อแสดงให้เห็นสมาชิกทราบว่า ผู้รับสายไม่ว่าง หรือการต่อระหว่างชุมสาย (Trunk) ไม่ว่าง ผู้เรียกจึงควรวางหูโทรศัพท์สักระยะหนึ่งแล้วจึงเริ่มทำการเรียกใหม่ สัญญาณไม่ว่างเป็นสัญญาณคลื่นไซน์ มีความถี่ 400 เฮิรตซ์ ส่งมาให้คู่สายเป็นช่วง ๆ โดยเป็นจังหวะดัง 0.5 วินาที สลับกันไป และมีขนาด 250 ถึง 300 มิลลิโวลต์พิกทูปิก ดังรูปที่ 2.2



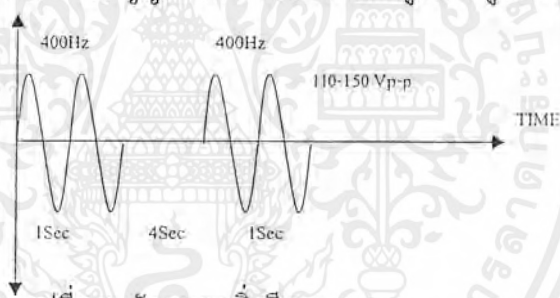
รูปที่ 2.2 สัญญาณไม่ว่าง

3. สัญญาณเรียกกลับ (RBT : Ring Back Tone) เป็นสัญญาณเพื่อแสดงว่า การต่อถูกขึ้นตอนตามความต้องการของผู้เรียกไปยังผู้รับ และส่งสัญญาณกริ่งให้กับผู้รับ เพื่อตอบรับการเรียก สัญญาณกริ่งเป็นสัญญาณคลื่นไซน์ที่มีความถี่ 20 เฮิร์ตซ์ มาเป็นช่วง ๆ โดยมีจังหวะดัง 1 วินาที และเงียบ 4 วินาที ระดับขนาด 5 โวลต์พีคทูพีค ดังรูปที่ 2.3



รูปที่ 2.3 สัญญาณเรียกกลับ

4. สัญญาณกริ่งเรียก (RGT : Ringing Tone) เป็นสัญญาณเพื่อแสดงว่าการต่อถูกขึ้นตอนเป็นไปตามความต้องการของผู้เรียกไปยังผู้รับ เครื่องชุมสายโทรศัพท์ สามารถดำเนินการติดต่อได้สำเร็จ และส่งสัญญาณกริ่งมาให้ผู้รับตอบรับการเรียก สัญญาณกริ่งเรียกเป็นสัญญาณคลื่นไซน์ มีความถี่ 20 เฮิร์ตซ์ ส่งมาเป็นช่วง โดยมีจังหวะดัง 1 วินาที มีระดับสัญญาณขนาด 100 โวลต์พีคทูพีค ดังรูปที่ 2.4



รูปที่ 2.4 สัญญาณกริ่งเรียก

2.1.3 ลักษณะของสัญญาณโทรศัพท์

สัญญาณโทรศัพท์ที่มีทั้งสัญญาณไฟตรง และไฟสลับ ซึ่งจะพิจารณาสัญญาณโทรศัพท์ได้เป็น 2 ลักษณะ ตามวิธีการใช้ คือ

1. เมื่อเป็นผู้เรียก

เมื่อโทรศัพท์ไม่ได้ใช้ หรือไม่ได้ยกหู สัญญาณระหว่างคู่สายจะเป็น 48 โวลต์ ดีซีเมื่อยกหูฟังขึ้น สัญญาณระหว่างโทรศัพท์จะตกลงเป็น 5 โวลต์ ดีซี ในขณะเดียวกันก็จะมีสัญญาณ 600 มิลลิโวลต์ ผสมมาด้วย เมื่อหมุนหรือกดหน้าปัทม์ เพื่อเรียกไปยังหมายเลขที่ต้องการติดต่อดังนั้น ในแบบหมุนหน้าปัทม์จะส่งพัลส์ที่หมุนโดยจะส่งพัลส์ในแบบ 100 พัลส์ต่อวินาที หรือ 20 พัลส์ต่อวินาที สำหรับในแบบกดปุ่มหน้าปัทม์นั้น จะส่งสัญญาณคู่ความถี่ ซึ่งเป็นสัญญาณของกลุ่มความถี่ต่ำและความถี่สูงรวมกัน ซึ่งจะเป็นความถี่มาตรฐานที่กำหนดไว้ดังรูป 2.5

		High Group Frequency (Hz)			
		1209	1336	1477	1633
Low Group Frequency (Hz)	697	1	2	3	A R1
	770	4	5	6	B R2
	852	7	8	9	C R3
	941	*	0	#	D R4
		C1	C2	C3	C4

R=ROW
C=COLUMN

รูปที่ 2.5 แสดงหน้าปัดมัลติมิเตอร์แบบกดปุ่มและความถี่ที่ใช้

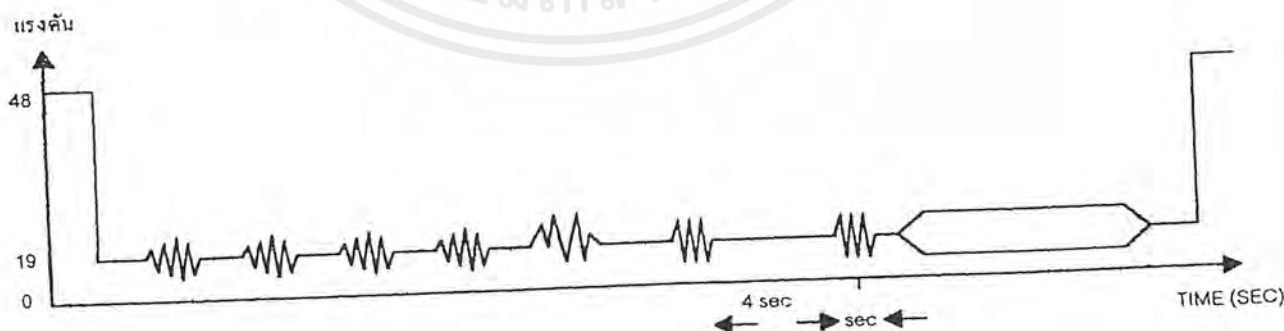
ขณะรอสัญญาณหลังการหมุน หรือกดปุ่มหน้าปัทม์ ถ้าได้รับสัญญาณเรียกกลับ แสดงว่ากำลังมีการเรียกไปยังหมายเลขที่กำลังติดต่อยู่ โดยเป็นสัญญาณเป็นจังหวะ 1 วินาที หยุด 4 วินาที สลับกัน ที่มี ความถี่ 440 เฮิรตซ์ และระดับสัญญาณ 200 มิลลิโวลต์ แต่ถ้าหากว่าได้รับสัญญาณสายไม่ว่าง ซึ่งจะเป็น สัญญาณดัง และหยุดสลับกันเป็นจังหวะทุก 0.5 เฮิรตซ์ และระดับสัญญาณ 400 มิลลิโวลต์

ขณะพูดสัญญาณระหว่างสายโทรศัพท์ยังคงเป็นสัญญาณ ดิจี 5 โวลต์ เช่นเดิม แต่จะมีสัญญาณ เสียงเป็นสัญญาณเอชี่ ระดับสัญญาณนาไม่เกิน 1 โวลต์ กร่อมอยู่บนสัญญาณดิจี

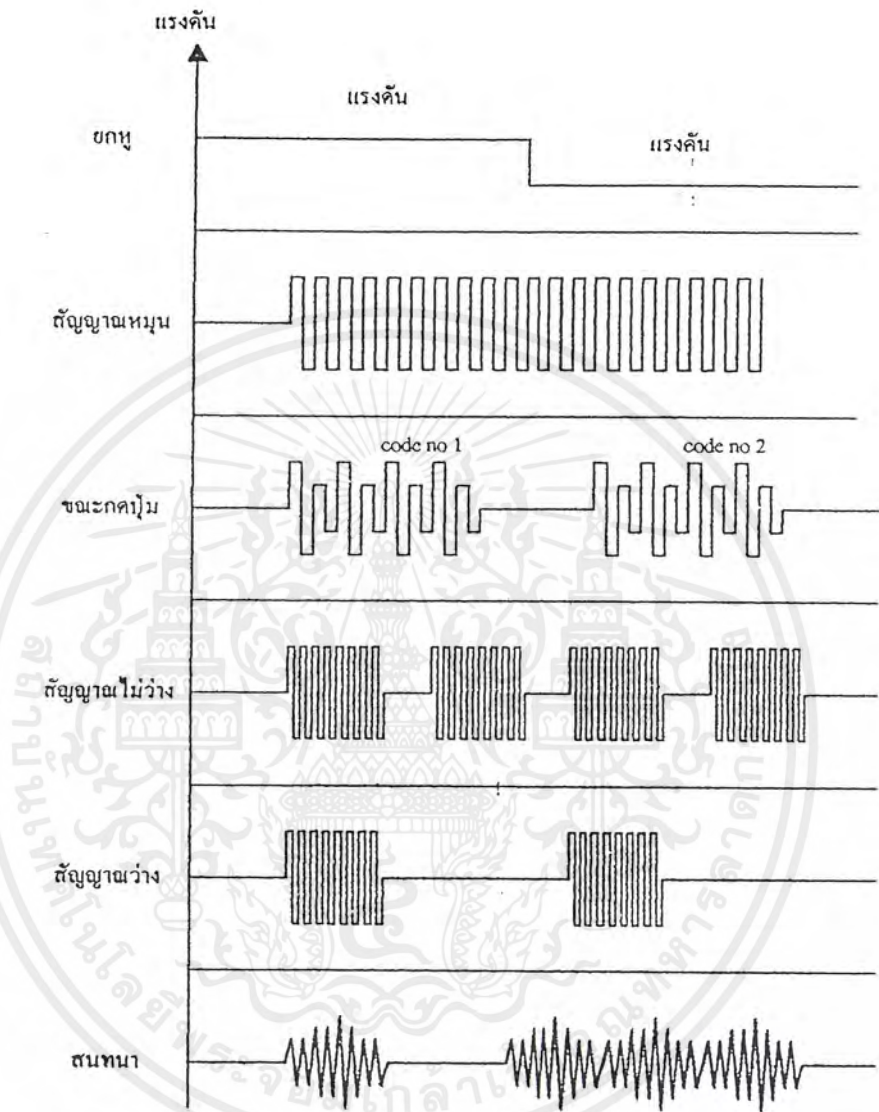
2. เมื่อเป็นผู้รับ

ขณะยังวางหูอยู่ สัญญาณระหว่างสายโทรศัพท์จะเป็น 48 โวลต์ ดิจี เมื่อมีสัญญาณกระดิ่ง จะมี สัญญาณเอชี่ ความถี่ 16 เฮิรตซ์ ระดับสัญญาณ 270 โวลต์ ดัง 1 วินาที และหยุด 4 วินาที สลับกัน

เมื่อยกหูโทรศัพท์ สัญญาณระหว่างสายโทรศัพท์จะตกลงไปเป็น 10 โวลต์ ดิจี และจะได้ยินเสียง จากผู้เรียก

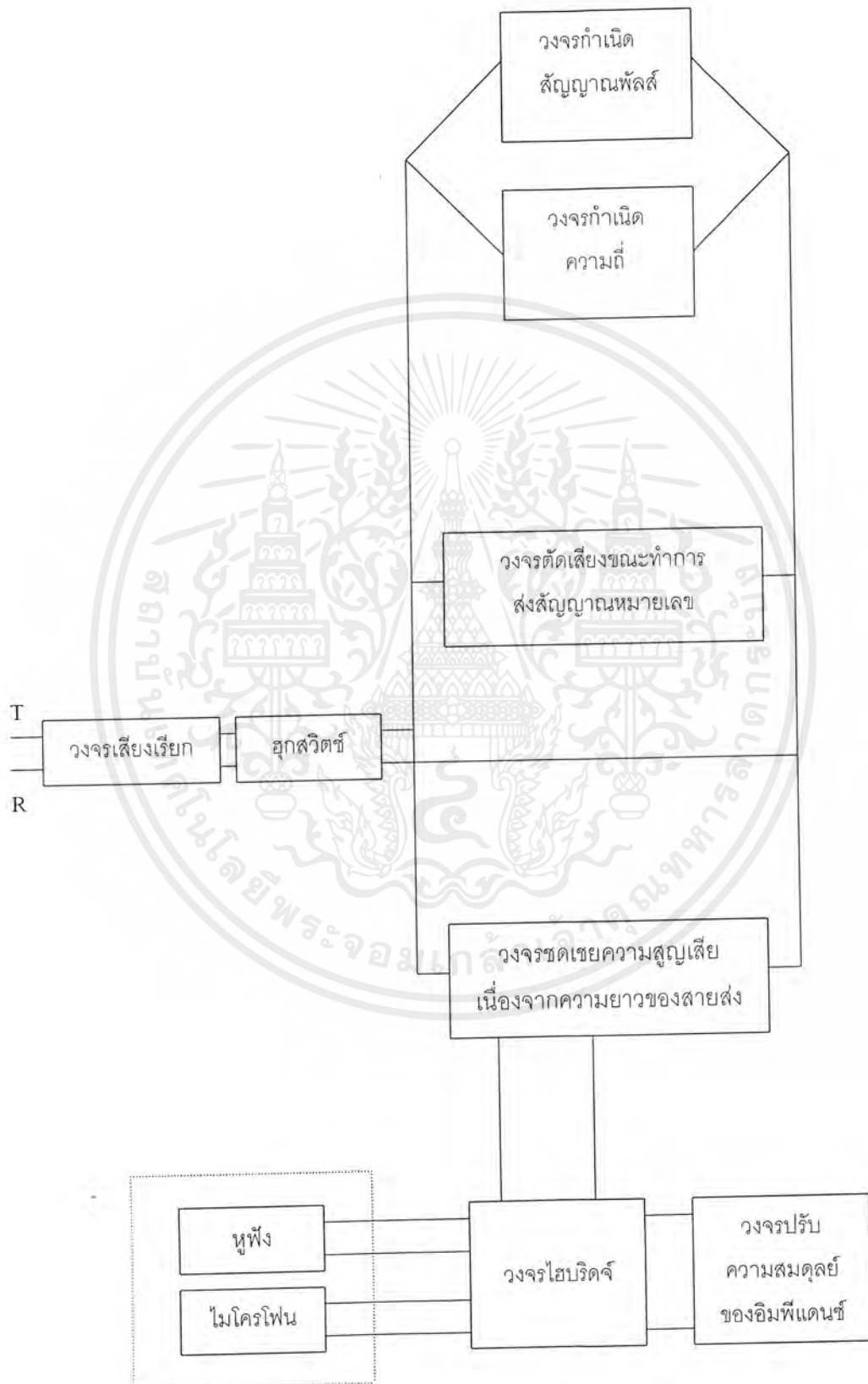


รูปที่ 2.6 ลักษณะของสัญญาณเมื่อผู้เรียกเรียกเข้ามา



รูปที่ 2.7 ลักษณะทางไฟฟ้าปรากฏที่ตู้สายในขณะทำการเรียก

2.1.4 ระบบทำงานของโทรศัพท์



รูปที่ 2.8 ผังการทำงานของโทรศัพท์

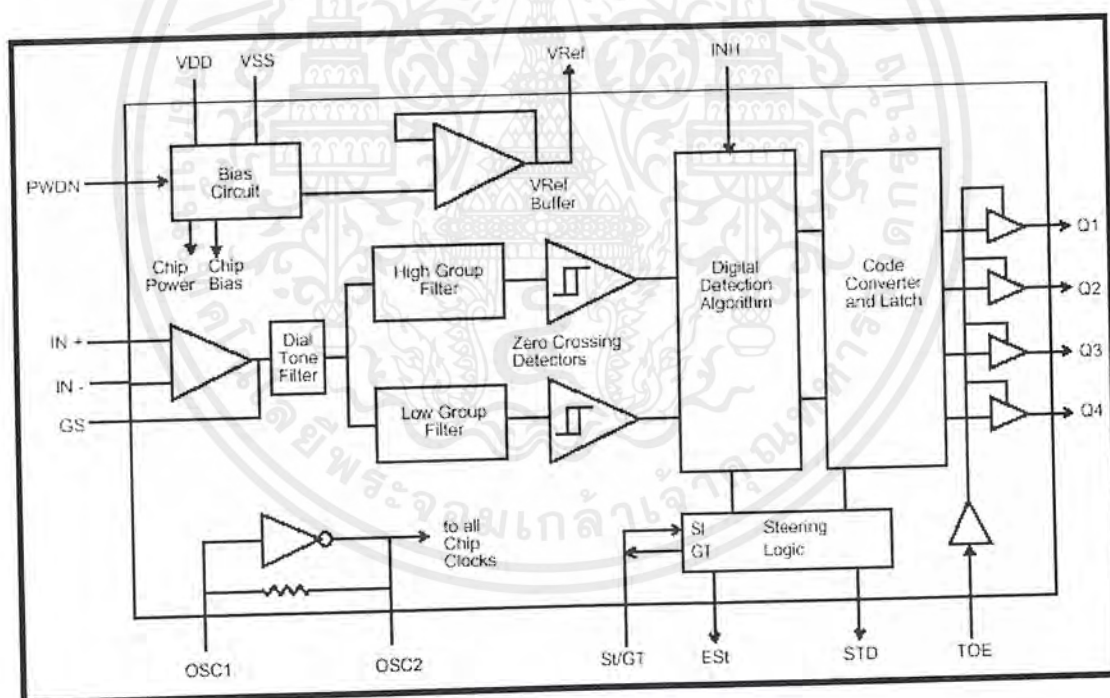
2.2 ภาคถอดรหัสสัญญาณความถี่ DTMF

2.2.1 คุณสมบัติพิเศษเฉพาะตัว MT 8870

1. เป็นตัวรับและถอดรหัสความถี่ (DTMF RECEIVER)
2. กินกระแสไฟน้อย ใช้ไฟเลี้ยงระดับเดียวกับไอซี TTL
3. สามารถตั้งอัตราการขยายภายในตัวไอซีได้
4. สามารถปรับช่วงคาบเวลาของความถี่ได้
5. เป็นไอซีคุณภาพสูง :

2.2.2 โครงสร้างไอซี MT 8870

โครงสร้างภายในของ MT 8870 ประกอบด้วยวงจรความถี่และวงจรถอดรหัสทางดิจิทัลเป็นไอซี ที่สร้างโดยใช้เทคโนโลยี IOS2-CMOS ในส่วนของวงจรกรองความถี่ใช้เทคนิคของ Switch Capacitor สำหรับการกรองความถี่สูงและกรองความถี่ต่ำ ส่วนวงจรถอดรหัสใช้เทคนิคการนับทางดิจิทัลเพื่อตรวจจับ และถอดรหัสทั้ง 16 ความถี่ออกเป็นเลขฐานสองขนาด 4 บิตและตรวจสอบช่วงเวลาที่สำคัญเข้ามา ส่วน Output เป็นวงจรค้าง 3 สถานะ



รูปที่ 2.9 โครงสร้างภายในของ MT 8870

2.2.3 หน้าที่การทำงานภายในไอซี MT 8870

ภายในไอซี MT 8870 ประกอบด้วยส่วนสำคัญ 5 ส่วน คือ

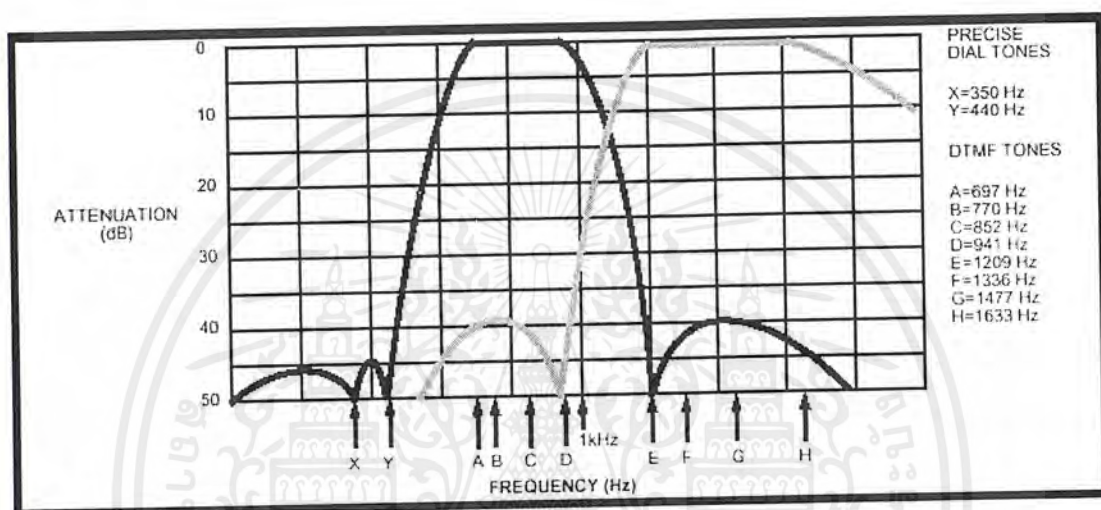
1. ภาคกรองความถี่ (Filter Section)
2. ภาคถอดรหัส (Decoder Section)
3. ภาคตรวจสอบสัญญาณ (Steering Circuit)

4. ภาคนัยสัญญาณความแตกต่าง (Differential input)

5. ภาคนัยความถี่ (Oscillator)

1. ภาครองความถี่

ในส่วนนี้จะแยกสัญญาณ DTMF ที่เข้ามาออกเป็นสองกลุ่มความถี่ คือ ช่วงความถี่สูงและช่วงความถี่ต่ำ โดยใช้วงจรความถี่อันดับ 6 ชนิด สวิตช์คาปาซิเตอร์ (Six-order switched capacitor band pass filter) ซึ่งความถี่ที่แยกได้มี 2 ช่วงความถี่ คือ ช่วงความถี่สูง (High Frequency) และช่วงความถี่ต่ำ (Low Frequency)



รูปที่ 2.10 ความถี่ที่ได้จากภาครองความถี่

2. ภาคนัยรหัส

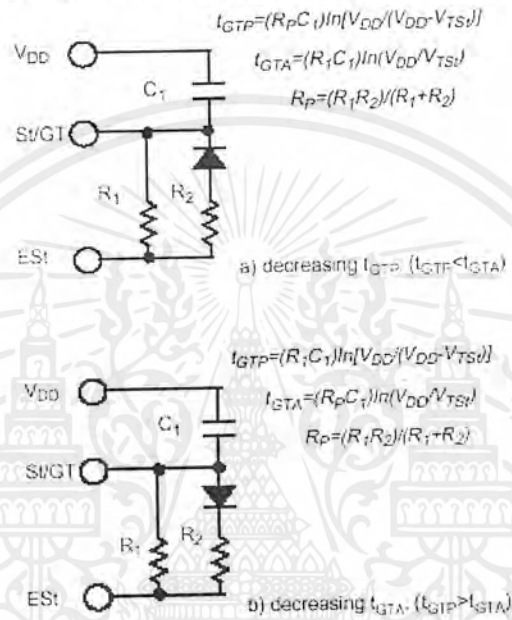
ความถี่ DTMF ที่ถูกกรองเรียบร้อยแล้วจะผ่านเข้าวงจรถอดรหัสความถี่ออกเป็นตัวเลขโดยใช้เทคนิคการนับแบบดิจิตอล และตรวจสอบความถี่ที่เข้ามาว่าเป็น ความถี่มาตรฐาน DTMF หรือไม่ เพื่อป้องกันความถี่อื่นเข้ามาปน เมื่อตรวจสอบว่าความถี่ถูกต้อง สัญญาณที่เข้าหา EAT (EARLY STREAMING) ก็จะทำงานสำหรับค่าที่ได้จากการถอดรหัสสัญญาณความถี่ต่าง ๆ นั้นแสดงในตารางที่ 2.2

Flow	Fhigh	NO	TOE	Q4	Q3	Q2	Q1
697	1209	1	H	0	0	0	1
697	1336	2	H	0	0	1	0
679	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

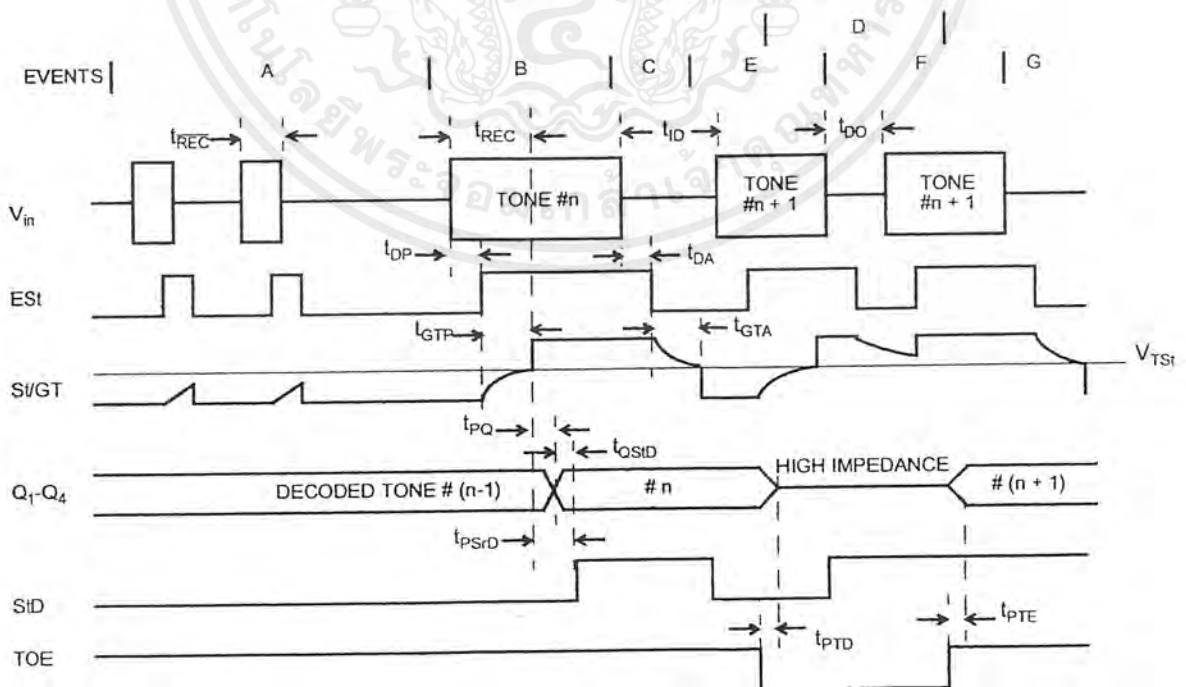
ตารางที่ 2.2 แสดงค่าถอดรหัสได้จากความถี่ต่างๆ

3. ภาพตรวจสอบสัญญาณ

ก่อนที่จะมีการถอดรหัสความถี่ไปที่ Output จะมีการตรวจสอบช่วงความถี่ที่เข้ามามีระยะเวลาตามที่กำหนดหรือไม่ โดยสังเกตจากระยะเวลาการกดปุ่มโทรศัพท์ ซึ่งต้องกดปุ่มให้มีความถี่ออกมาให้เป็นเวลาพอสมควร มิฉะนั้นวงจรส่วนนี้จะไม่รับ โดยถือว่าสัญญาณนั้นไม่ถูกต้อง ส่วนช่วงเวลายาวเท่าใดสามารถตั้งได้โดยใช้ RC ภายนอก สัญญาณที่เข้า EST จะเป็น 1 นานใกล้เคียงกับระยะเวลาที่มีความถี่ DTMF เข้ามา จากรูปที่ 2.11 เมื่อเข้า EST เป็น 1 ทำให้ VC สูงขึ้นจนถึงค่าเทรสโฮลด์ วงจรถอดรหัสจึงจะถอดรหัสออกเป็นตัวเลขขนาด 4 บิต รายละเอียดการทำงานพิจารณาจากแผนภูมิเวลาในรูปที่ 2.12



รูปที่ 2.11 วงจรตรวจสอบสัญญาณอย่างง่าย



รูปที่ 2.12 ผังเวลาของไอซี MT 8870

อธิบายขั้นตอนการทำงาน

- A – ตรวจสอบความถี่ที่เข้ามา แต่คาบเวลาไม่ถูกต้อง Output ไม่เปลี่ยนแปลง
- B – ความถี่ # n ถูกตรวจสอบพบ และมีคาบเวลาที่ถูกต้อง ความถี่ที่ถูกถอดรหัส และค้างไว้ที่

Output

- C – จบความถี่ # n ช่วงห่างถูกต้อง Output ยังคงค้างจนกว่าจะได้รับความถี่ที่ถูกต้องใหม่
- D – Output เปลี่ยนเป็นอิมพีแดนซ์สูง
- E – ความถี่ # n+1 ถูกตรวจพบช่วงเวลาที่ถูกต้อง ความถี่ที่ถูกถอดรหัสและค้างไว้
- F – ความถี่ # n+1 หายไป ช่วงห่างไม่ถูกต้อง Output ยังคงค้างอยู่
- G – ความถี่ # n+1 ช่วงห่างถูกต้อง Output ยังคงค้างอยู่จนถึงความถี่ใหม่

อธิบายคำศัพท์

- Vin – สัญญาณความถี่ DTMF ที่เข้ามา
- Est – Early Streering output ให้แสดงความถี่ที่ถูกต้อง
- St / Gt – Streering input / Guard Time output สำหรับต่อกับ RC ภายนอก
- Q1-Q4 – Output BCD ขนาด 4 บิต
- StD – Delayed Streering output ใช้แสดงว่าความถี่ที่ได้รับหรือที่หายไปมีค่าเวลาตามที่กำหนด

เพื่อแสดงความถูกต้องของสัญญาณ

- TOE – Tone Output Enable (input) ใช้ควบคุม Q1-Q2 ให้เป็นอิมพีแดนซ์สูง
- Trec – คาบเวลาที่นานที่สุดที่ตรวจพบความถี่ DTMF ที่ถูกต้อง 2 สัญญาณ
- Tid – เวลาสั้นที่สุดระหว่างสัญญาณ DTMF ที่ถูกต้อง 2 สัญญาณ
- Tdo – เวลานานที่สุดที่ยอมให้สัญญาณหายไป ในคาบเวลาความถี่ที่ถูกต้อง
- Tdp – เวลาที่ใช้ในการตรวจพบความถี่ DTMF ที่ถูกต้อง
- Tda – เวลาที่ใช้ในการตรวจหายิบของสัญญาณความถี่ DTMF ที่ถูกต้อง
- Tgtp – ช่วงคาบเวลาของความถี่ของการปรากฏความถี่ DTMF
- Tgta – ช่วงคาบเวลาของความถี่ของการหายไปของความถี่ DTMF

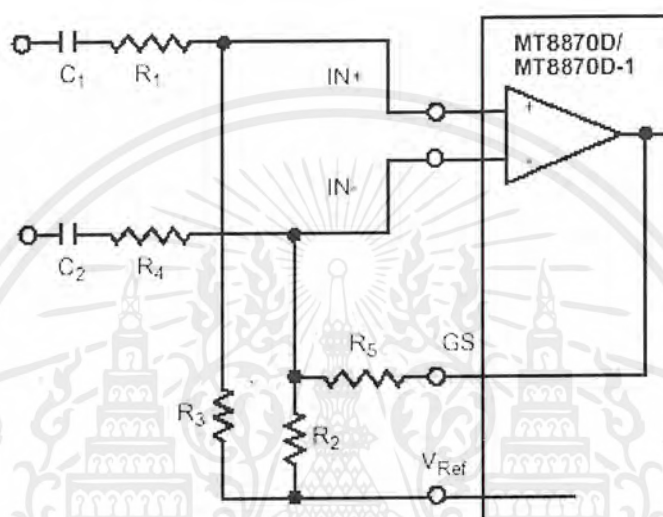
สำหรับคำว่า Grand Time หมายถึง ช่วงคาบเวลาความถี่ที่เข้ามาซึ่งจะต้องนานเท่ากับหรือมากกว่าเวลาที่ตั้งไว้ จึงจะได้การยอมรับว่าสัญญาณความถี่นั้นถูกต้อง หรือพูดว่า เวลาที่เราตั้งไว้โดย RC ก็คือ Grand Time นั่นเอง เมื่อสัญญาณความถี่เข้ามานานเท่ากับหรือมากกว่าเวลาที่ตั้งไว้จึงสามารถแปลงเป็นตัวเลขได้ ถ้าสัญญาณความถี่ที่เข้ามาสั้นกว่าก็จะไม่มีการถอดรหัสเป็นตัวเลขออกไป การตั้งเวลาและคำนวณดูได้จากรูป 2.11

4. ภาคขยายสัญญาณความแตกต่าง

วงจรส่วน Input ของ MT 8870 เป็นภาคขยายสัญญาณจากออปแอมป์ ที่สามารถปรับอัตราขยายโดยตรงต่อวงจรภายนอกเพิ่มเข้าไป และแสดงการต่อวงจรภายนอกเข้ากับ Input ซึ่งสามารถคำนวณอัตราขยายความแตกต่างของ Input และอิมพีแดนซ์ดังนี้

$$\text{อัตราขยาย (Av diff)} = \frac{R5}{R1}$$

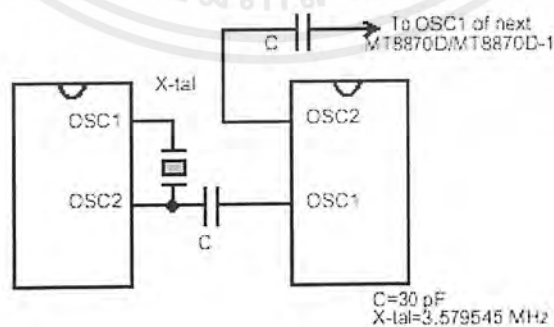
$$\text{อินพุทอิมพีแดนซ์ (Zin diff)} = \frac{2}{R1} + \left(\frac{1}{\omega C} \right)^2$$



รูปที่ 2.13 การต่อวงจรภาคอินพุท

5. กำหนดความถี่

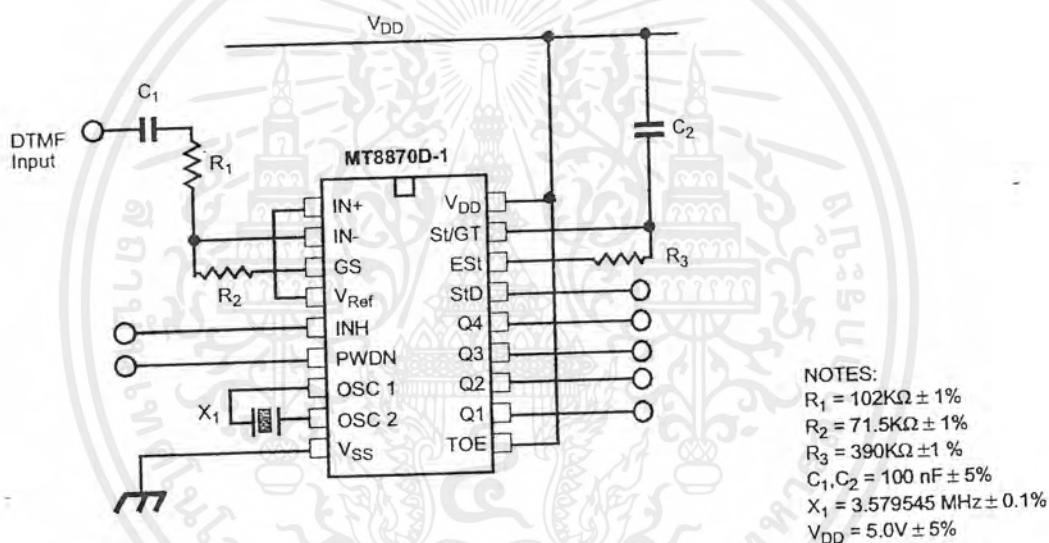
ในภาคนี้จะมีวงจรเวลาอยู่ในเพียงแต่ต่อ แร่ X-Tal ที่มีความถี่ 3.579545 MHz ก็จะสามารถใช้งานได้เลย



รูปที่ 2.14 การต่อวงจรผลิตความถี่

2.2.4 การนำ MT 8870 ไปใช้งาน

1. นำไปใช้งานด้านการควบคุมระยะไกล
2. เครื่องป้องกันโทรศัพท์ทางไกล
3. ใช้ในงานด้านเครดิตการ์ด
4. ใช้งานร่วมกับคอมพิวเตอร์
5. ใช้ในเครื่องชุมสายขนาดเล็กๆ หรือ PAPX
6. ใช้กับงานทางด้านโทรศัพท์ทั่วไป
7. เครื่องกันขโมย
8. การควบคุมอุปกรณ์ทางโทรศัพท์
9. ใช้เป็นเครื่องสอบถามทางโทรศัพท์



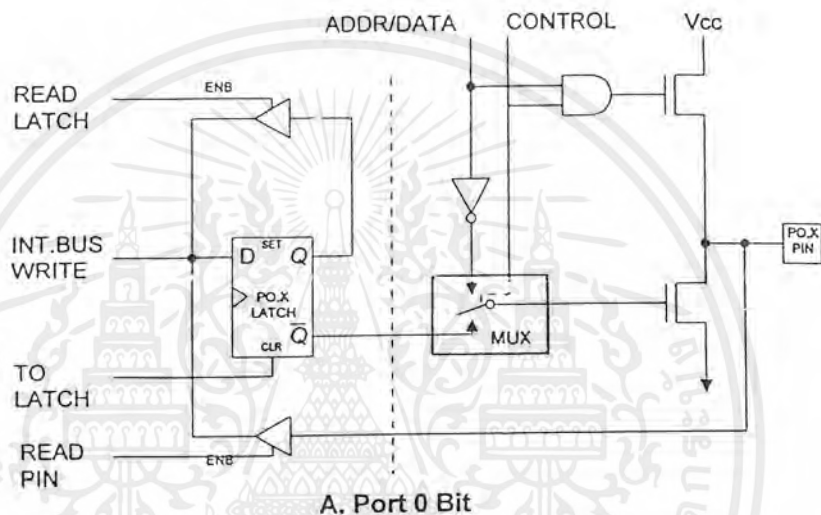
รูปที่ 2.15 วงจรใช้งานเบื้องต้นของไอซี MT 8870

2.3 ส่วนควบคุมการทำงาน

2.3.1 พอร์ตของ 8051

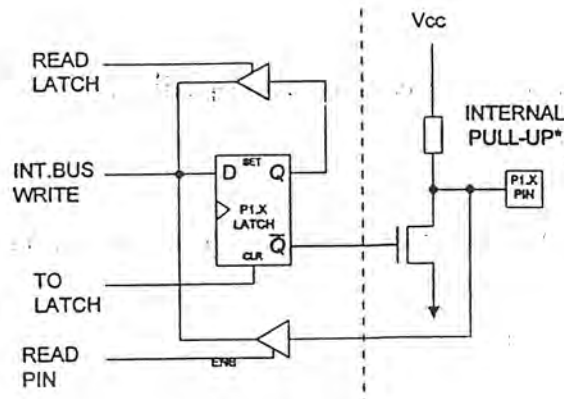
8051 เป็นไมโครคอนโทรลเลอร์ขนาด 40 ขา ซึ่งมีขาต่าง ๆ ดังนี้

- Vcc (ขา 40) ต่อกับ + 5V
- Vss (ขา 20) เป็นขา GND
- พอร์ต 0 (ขา 32-39) มีทั้งหมด 8 บิต คือ (P0.7-P0.0) มีโครงสร้างแบบ Open-Drain Bi-directional ดังแสดงในรูป 2.16



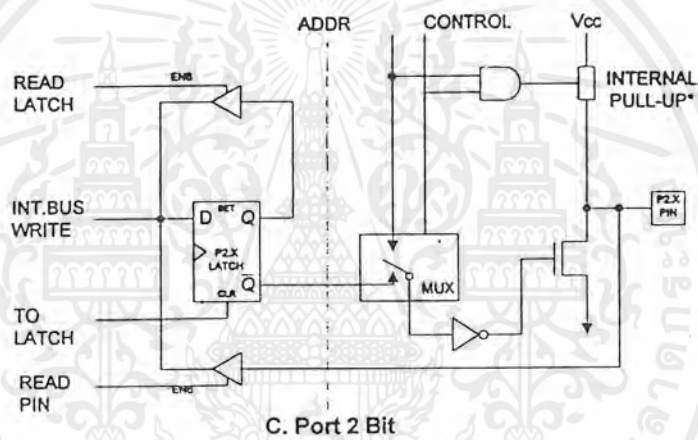
รูปที่ 2.16 แสดงโครงสร้าง พอร์ต 0 (บิต)

- พอร์ต 0 (ขา 32-39) มีทั้งหมด 8 บิต คือ (P0.7-P0.0) ใช้งานได้ 2 หน้าที่ คือแอดเดรสบััสและดาต้าบััสเมื่อต้องการติดต่อกับหน่วยความจำภายนอกหรือเป็นโอไอพอร์ท ถ้าต้องการให้ทำงานเป็นอินพุตพอร์ทต้องส่งลอจิก “1” ไปยังพอร์ตนี้นี้ จะมีผลให้ $\overline{Q}$ ของ D-FF เป็น “0” ทำให้ FET ตัวล่างมีสถานะ OFF สัญญาณที่ใช้อ่านอินพุตพอร์ทแลทช์โดยส่งสัญญาณ READLATCH ไปกระตุ้นที่ Tri-State Buffer ตัวบนและการอ่าน Port (pin) จะใช้สัญญาณ Read (pin)
- พอร์ต 1 (ขา 21-28) มีทั้งหมด 8 บิต คือ (P1.0-P1.7) มีโครงสร้างคล้ายพอร์ท 0 แต่จะใช้ความต้านทานภายในพูลอัพแทน Internal Pull up Register มีโครงสร้างดังรูป 2.17
- พอร์ต 2 (ขา 21-28) มีทั้งหมด 8 บิต คือ ขา (P2.7-P2.0) มีโครงสร้างคล้าย พอร์ท 0 โดยมี FET ตัวล่างตัวเดียวส่วนด้านบนใช้ความต้านทานพูลอัพแทน (Internal pull up) พอร์ตนี้นี้ทำงาน 2 หน้าที่ คือสามารถใช้เป็นแอดเดรสบััสขนาด 8 บิต (A15-A8) และเป็นโอไอพอร์ทใช้งานทั่วไป เมื่อจะใช้งานเป็นอินพุตพอร์ทต้องส่งลอจิก “1” มาที่พอร์ตนี้นก่อนเพื่อกันดึงให้ FET อยู่ในสภาวะ off ดังแสดงในรูป 2.18



B. Port 1 Bit

รูปที่ 2.17 โครงสร้างของพอร์ต 1 (บิต)



C. Port 2 Bit

รูปที่ 2.18 โครงสร้างของ พอร์ต 2 (บิต)

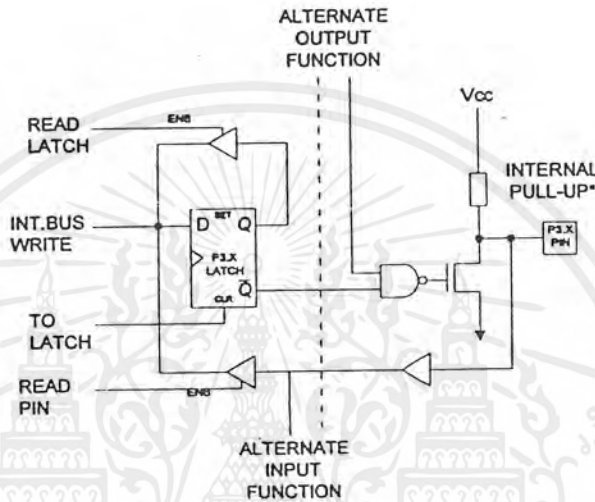
- พอร์ต 3 (ขา 10-17) มีทั้งหมด 8 บิต คือ ขา (P3.7-P3.0) มีโครงสร้างคล้ายพอร์ต 1 ทำงานได้ 2 หน้าที่คือเป็นไอโอพอร์ตถ้าจะโปรแกรมให้เป็นอินพุตพอร์ทต้องส่งลอจิก “1” มาที่พอร์ทนี้ ก่อนและอีกหน้าที่หนึ่งก็คือใช้ส่งสัญญาณควบคุมออกมาและรับสัญญาณเข้าไปสัญญาณต่าง ๆ มีดังนี้

P3.0/RXD (Serial Input Port)	เป็นขาที่ใช้รับข้อมูลแบบอนุกรม (UART)
P3.1/TXD (Serial Output Port)	เป็นขาที่ใช้ส่งข้อมูลแบบอนุกรม (UART)
P3.2/ $\overline{\text{INT0}}$ (External Interrupt 0)	ใช้รับสัญญาณการขัดจังหวะจากภายนอกเบอร์ 0
P3.3/ $\overline{\text{INT1}}$ (External Interrupt 1)	ใช้รับสัญญาณการขัดจังหวะจากภายนอกเบอร์ 1
P3.4/TO (Counter 0 External Input)	ขารับสัญญาณพัลส์อินพุตเข้าไปยังวงจร Counter 0 (เป็นอินพุตโหมดคาน์เตอร์)
P3.5/T1 (Counter 1 External Input)	ขารับสัญญาณพัลส์อินพุตเข้าไปยังวงจร Counter 1 (เป็นอินพุตโหมดคาน์เตอร์)

P3.6/ $\overline{\text{WR}}$ (External Data Memory Write Strobe) ขาสัญญาณควบคุมการเขียนข้อมูลลงหน่วยความจำภายนอก

P3.7/ $\overline{\text{RD}}$ (External Data Memory Read Strobe) ขาสัญญาณควบคุมการอ่านข้อมูลจากหน่วยความจำภายนอก

2.3.2 โครงสร้างของ (พอร์ท 3) ดังแสดงในรูป 2.19



รูปที่ 2.19 โครงสร้างของพอร์ท 3 (บิต)

- **ALE** (ขา 30) เป็นขาส่งสโตรบสำหรับใช้ในการแลตช์แอดเดรสไบต์ต่ำ (A7-A0) ที่ส่งออกมาจาก (พอร์ท 0) สัญญาณนี้จะแอตทีฟทุก 2 ครั้ง ใน 1 แมกซีนไซเคิล
- **$\overline{\text{PSEN}}$** (ขา 29) เป็นขาสโตรบที่ใช้สำหรับอ่านข้อมูลจาก Program Memory ภายนอกสัญญาณนี้จะส่งออกมา 2 ครั้ง ในแต่ละแมกซีนไซเคิลแต่ถ้าเป็นการอ่าน Internal program Memory จะไม่มีสัญญาณออกที่ขา
- **$\overline{\text{EA}}$** (ขา 30) ใช้เลือกหน่วยความจำโปรแกรมภายนอก
 บิต "0" จะอ่านโปรแกรมจากภายนอกชิพ
 บิต "1" จะอ่านโปรแกรมจากภายในชิพ
- **RST** (ขา 9) ขาสีท์ได้ก็ต่อเมื่อบิตลอจิก "1" เข้ามานี้นานอย่างน้อย 2 แมกซีนไซเคิล
- **XTAL1** (ขา 19) ใช้ต่อคริสตอลภายนอกโดยเป็นอินพุตเข้าสู่วงจรรอสซีสเลเตอร์ภายใน
- **XTAL2** (ขา 18) ใช้ต่อคริสตอลภายนอก โดยเป็นอินพุตของวงจรรอสซีสเลเตอร์ภายใน

2.4 การเชื่อมโยง MCS-51 กับหน่วยความจำ

2.4.1 การจัดหน่วยความจำสำหรับ MCS-51

แบ่งได้เป็น 2 ส่วนใหญ่ ๆ คือ

- Data Memory
- Program Memory

Data Memory

เป็นหน่วยความจำที่ใช้เก็บข้อมูล มีความจุได้ถึง 64 Kbytes เป็นหน่วยความจำประเภทอ่านและเขียนได้ (RAM) สัญญาณจาก MCS-51 ที่ใช้ในการอ่านข้อมูลจาก RAM คือ $\overline{RD}$ และสัญญาณที่ใช้ในการเขียนข้อมูลลง RAM คือ $\overline{WR}$

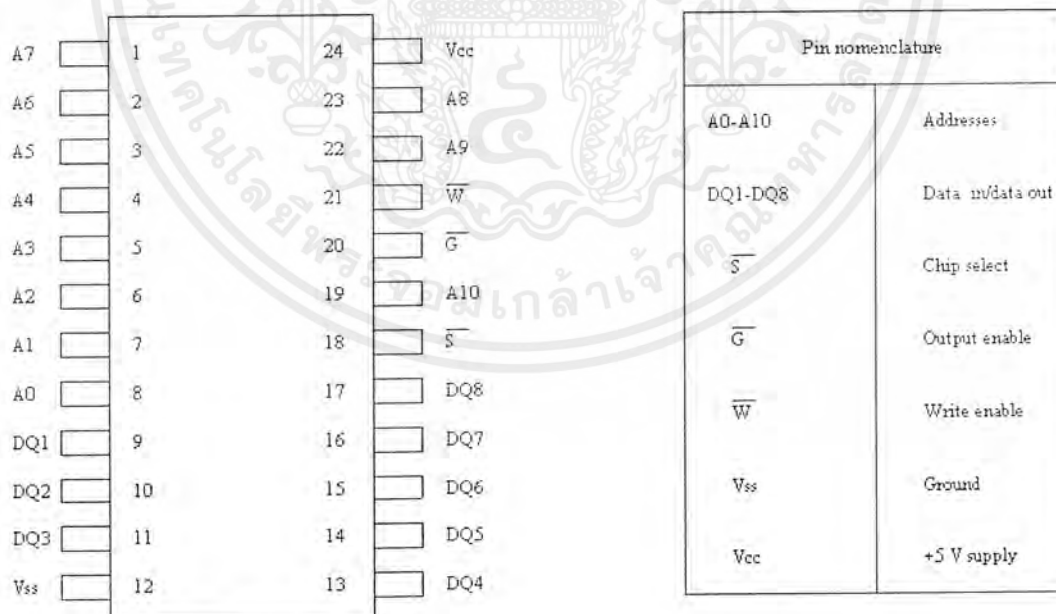
RAM (Random Access Memory)

เป็นหน่วยความจำที่สามารถเขียนและอ่านข้อมูลได้ ข้อมูลจะสูญหายทันทีที่ขาดไฟเลี้ยง แบ่งเป็น 2 ชนิด คือ Static RAM และ Dynamic RAM

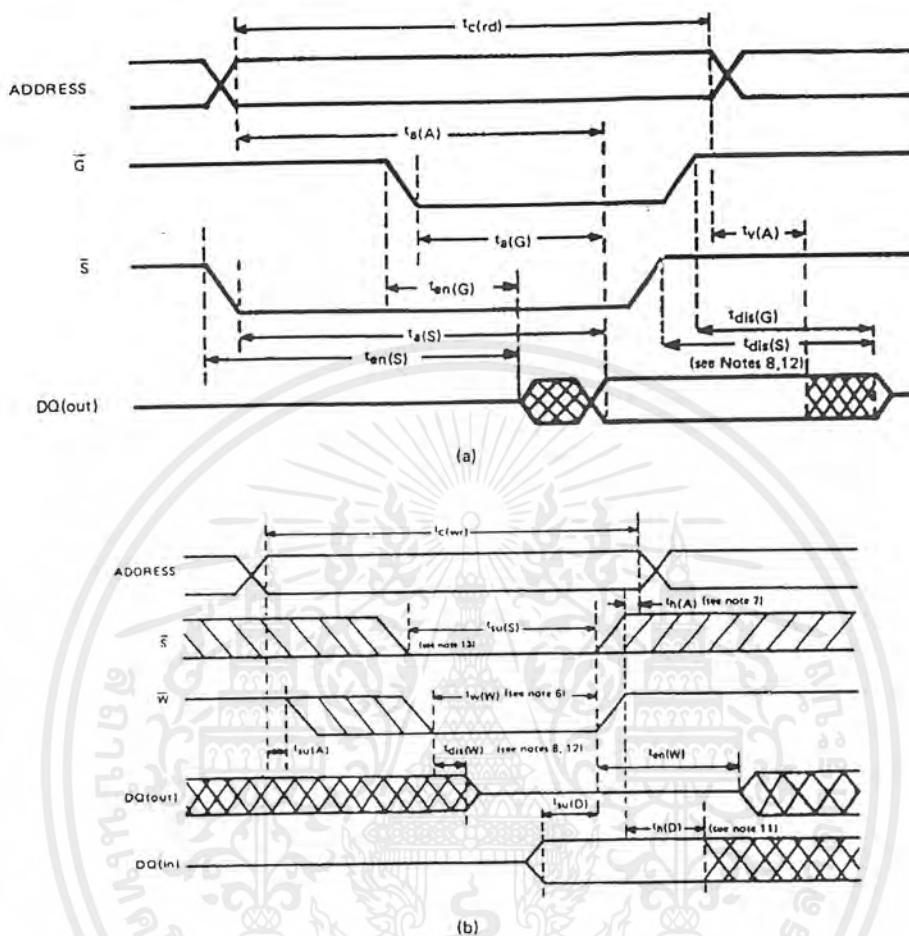
STATIC RAM

เป็นหน่วยความจำชนิดหนึ่งที่ย่ายในการนำมาใช้งานแต่ราคาสูงเมื่อเทียบกับ Dynamic RAM ซึ่งราคาถูกกว่าแต่ ยุ่งยากขึ้นมากก็ต้องทำการ refresh หน่วยความจำตลอดเวลา ส่วน Static RAM ที่มีใช้อย่างแพร่หลายมีอยู่ด้วยกันหลายเบอร์ตามขนาดความจุของหน่วยความจำเบอร์ที่นิยมกันมากสำหรับงานที่ไม่ใหญ่มากนัก คือ เบอร์ TMS4016 หรือ 6116 ซึ่งมีขนาดความจุ 2K X 8 ดังมีการจัดวางขาตามรูป 2.20 และผังเวลาของ TMS4016 ดังแสดงในรูป 2.21

TMS4016 24 – Pin plastic dual-in-line package (top view)



รูปที่ 2.20 การจัดวางขาของ Static RAM เบอร์ TMS4016



รูปที่ 2.21 (a) Read cycle timing for the TMS4016 and (b) write cycle timing.

(Courtesy of Texas Instruments, Inc.)

PROGRAM MEMORY

เป็นหน่วยความจำที่ใช้เก็บโปรแกรมเป็นหน่วยความจำที่อ่านข้อมูลได้อย่างเดียว(ROM)

สัญญาณจาก MCS-51 ที่ใช้ในการอ่าน ROM คือ $\overline{PSEN}$

ROM (Read Only Memory)

เป็นหน่วยความจำแบบถาวร เมื่อขาดไฟเลี้ยงข้อมูลไม่สูญหาย เป็นหน่วยความจำชนิดอ่านข้อมูลได้อย่างเดียว

EPROM (Erasable Programmable read-only Memory)

เป็นหน่วยความจำแบบถาวรชนิดหนึ่งซึ่งผู้ใช้นำมาโปรแกรมเองได้และสามารถใช้แสง UV ลบข้อมูลได้

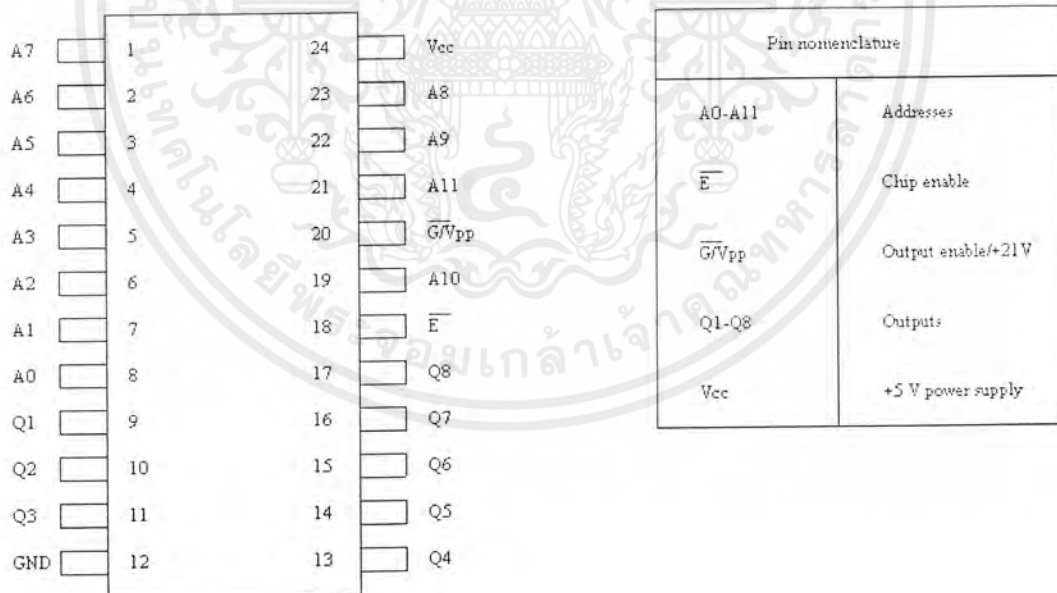
The 27xx Family of EPROM

ดังรายละเอียดดัง ตารางที่ 2.3 การจัดวางขา ดังรูป 2.22 และมีผังเวลาดังรูป 2.23

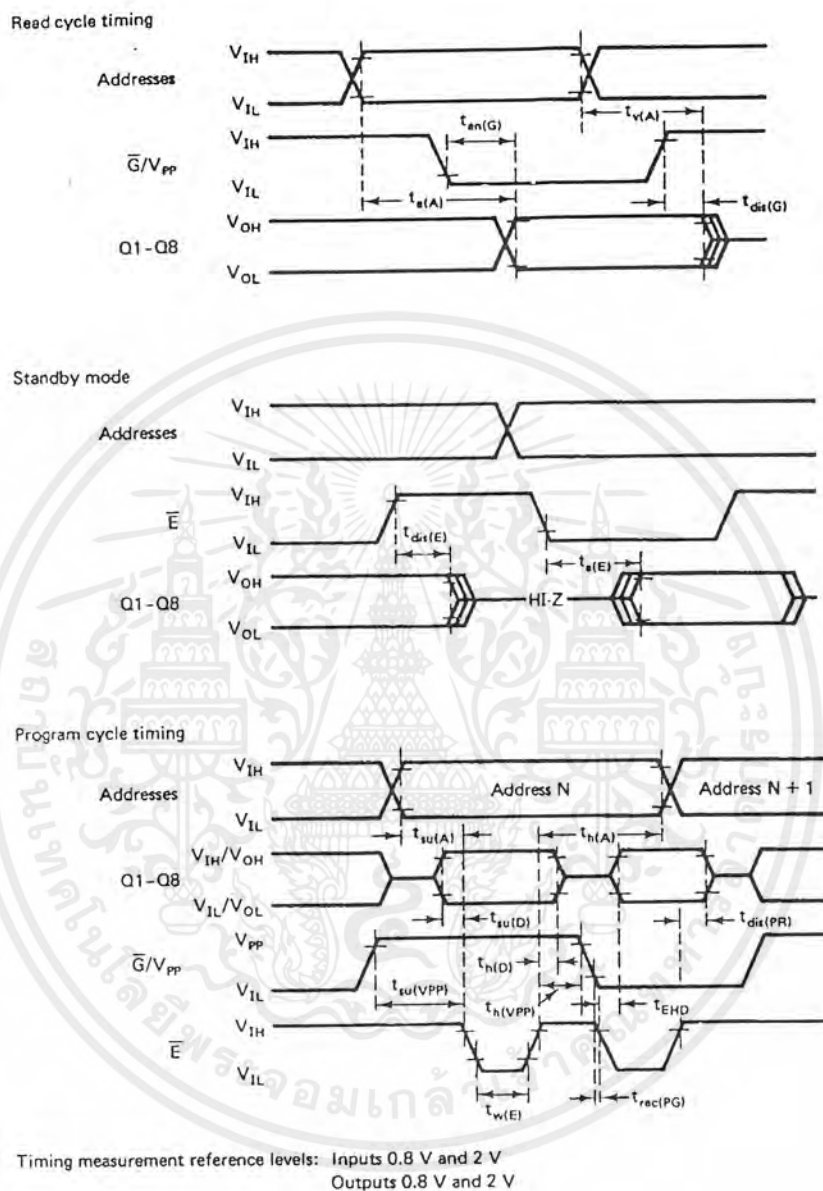
ตารางที่ 2.3 แสดงรายละเอียดการจัดวางขา

เบอร์	ความจุ
2704	512 x 8
2708	1K x 8
2716	2K x 8
2723	4K x 8
2764	8K x 8
27128	16K x 8
27256	32K x 8
27512	64K x 8

TMS2732A.....JL Package (top view)



รูปที่ 2.22 การจัดวางขาของ TMS 2732A



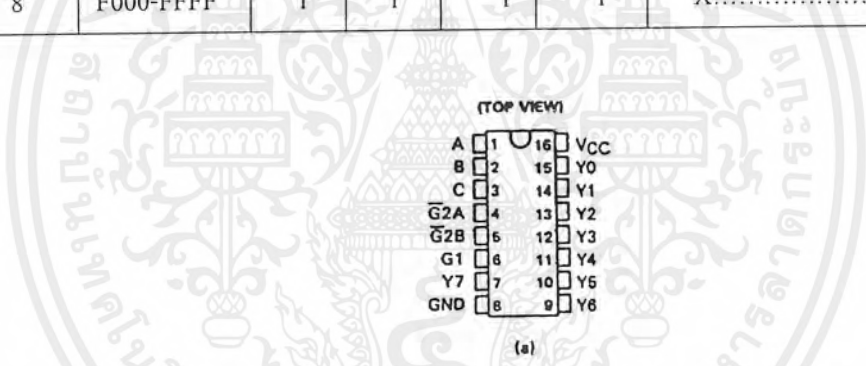
รูปที่ 2.23 ฟังก์ชันเวลาในการอ่านและโปรแกรมของ TMS 2732A

2.4.2 การถอดรหัสตำแหน่งของหน่วยความจำ

การถอดรหัสตำแหน่งของหน่วยความจำมีความจำเป็นอย่างยิ่งเพราะว่าหน่วยความจำที่นำมาใช้มีขนาดเล็กกว่า 64 Kbytes ดังนั้น จึงต้องนำมาเรียงกันแล้วใช้วงจรถอดรหัสตำแหน่งของหน่วยความจำแยกเป็นแต่ละช่วง ถ้าใช้หน่วยความจำขนาด 4 Kbytes จำนวน 8 ตัว จะได้เท่ากับ 32 Kbytes ถ้าให้ตัวแรกเริ่มที่แอดเดรส 8000H ตัวที่ 2 ก็จะเริ่มที่แอดเดรส 9000H ดังรายละเอียดดัง ตารางที่ 2.4

ตารางที่ 2.4 ตารางการถอดรหัสตำแหน่งหน่วยความจำ

EPROM ตัวที่	แอดเดรส (HEX)	A15	A14	A13	A12	A11.....A0
1	8000-8FFF	1	0	0	0	X.....X
2	9000-9FFF	1	0	0	1	X.....X
3	A000-AFFF	1	0	1	0	X.....X
4	B000-BFFF	1	0	1	1	X.....X
5	C000-BFFF	1	1	0	0	X.....X
6	D000-DFFF	1	1	0	1	X.....X
7	E000-EFFF	1	1	1	0	X.....X
8	F000-FFFF	1	1	1	1	X.....X



FUNCTION TABLE

ENABLE INPUTS		SELECT INPUTS			OUTPUTS							
G1	G2*	C	B	A	Y0	Y1	Y2	Y3	Y4	Y5	Y6	Y7
X	H	X	X	X	H	H	H	H	H	H	H	H
L	X	X	X	X	H	H	H	H	H	H	H	H
H	L	L	L	L	L	H	H	H	H	H	H	H
H	L	L	L	H	H	L	H	H	H	H	H	H
H	L	L	H	L	H	H	L	H	H	H	H	H
H	L	L	H	H	H	H	H	L	H	H	H	H
H	L	H	L	L	H	H	H	H	L	H	H	H
H	L	H	L	H	H	H	H	H	H	L	H	H
H	L	H	H	L	H	H	H	H	H	H	L	H
H	L	H	H	H	H	H	H	H	H	H	H	L

*G2 = G2A + G2B

(b)

รูปที่ 2.24 การจัดวางขาและตารางความจริงของ 74LS138

การถอดรหัสตำแหน่งของหน่วยความจำจะใช้ IC ถอดรหัส 3 to 8Line Decoder เบอร์ 74LS138
ดังมีการจัดวางขาตามรูป 2.24 (a) และตารางความจริง (b)

2.5 มาตรฐานพอร์ตอนุกรมแบบ RS-232

มาตรฐานการเชื่อมต่อแบบอนุกรม RS-232 เป็นมาตรฐานอุตสาหกรรมที่ออกแบบมาเพื่อใช้ในการส่งข้อมูลอนุกรมแบบอะซิงโครนัส 2 ทิศทาง โดยมาตรฐาน RS-232 ในอดีตนั้นถูกออกแบบมาเพื่อการส่งผ่านข้อมูลจากคอมพิวเตอร์ไปยังโมเด็มเพียงอย่างเดียว เพื่อที่จะนำข้อมูลจากโมเด็มนี้สื่อสารผ่านสายโทรศัพท์ไปยังคอมพิวเตอร์อีกชุดซึ่งอยู่ห่างไกลกัน โดยคณะกรรมการที่เรียกว่า สมาคมอุตสาหกรรมอิเล็กทรอนิกส์ (Electronic Industries Association : EIA) ได้วางมาตรฐานที่มีชื่อเรียกกันว่า EIA RS - 232 มาตรฐานนี้ในช่วงแรกจะใช้คอนเน็กเตอร์เป็นแบบ DB - 25 โดยกำหนดความยาวสูงสุดของสายสัญญาณไว้ที่ 50 ฟุต มีระดับสัญญาณตั้งแต่ -3 ถึง -12V แสดงว่ามีข้อมูล (Mark) และ +3 ถึง +12V แสดงว่าเป็นช่องว่าง (Space)

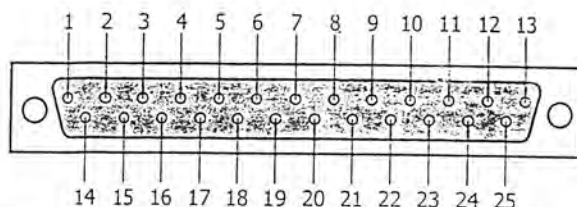
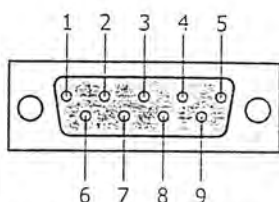
มาตรฐาน RS - 232 ได้กำหนดรูปแบบของอุปกรณ์เชื่อมต่อข้อมูล (Data Terminal Equipment : DTE) กับวงจรข้อมูลปลายทาง (Data Circuit Termination : DCE) ไว้ว่า อุปกรณ์ DTE จะต้องเป็นอุปกรณ์ที่มีการประมวลผลในตัวเช่น ไมโครคอนโทรลเลอร์หรือไมโครคอมพิวเตอร์ ซึ่งมีความสามารถในการสร้างบิตข้อมูลแบบอนุกรมได้ ส่วนอุปกรณ์ DCE จะทำหน้าที่เป็นเพียงตัวรับข้อมูลที่ส่งมาจาก DTE เท่านั้น โดยการรับส่งข้อมูลระหว่างอุปกรณ์ทั้งสองจะกระทำผ่านมาตรฐาน RS - 232

ข้อแตกต่างของอุปกรณ์ DTE และอุปกรณ์ DCE อย่างหนึ่งที่เราเห็นได้ชัดคือ คอนเน็กเตอร์ของ DTE จะเป็นตัวผู้ ส่วนคอนเน็กเตอร์ของ DCE จะเป็นตัวเมีย ซึ่งพอร์ตอนุกรมของคอมพิวเตอร์ที่ใช้กันอยู่ทั่วไปจะเป็นแบบ DTE ส่วนคอนเน็กเตอร์ที่อยู่ทีโมเด็มจะเป็นแบบ DCE

สำหรับการใช้งานบนคอมพิวเตอร์ พอร์ตอนุกรม RS - 232 มักถูกใช้เชื่อมต่อกับโมเด็มหรือเมาส์ โดยสามารถรับส่งข้อมูลได้ที่ความยาวของสายสัญญาณสูงสุดถึง 20 เมตร

2.5.1 คอนเน็กเตอร์สำหรับพอร์ต RS - 232 และการเชื่อมต่อ

มาตรฐานการเชื่อมต่อแบบ RS - 232 จะใช้คอนเน็กเตอร์แบบ DB - 25 ตัวผู้หรือ DB - 9 ตัวผู้ ซึ่งคอนเน็กเตอร์แบบ DB - 25 จะมีขาต่อใช้งานเพียง 9 เส้น เช่นเดียวกับคอนเน็กเตอร์แบบ DB - 9 เนื่องจากขาอื่น ๆ ที่เคยใช้งานในอดีต ปัจจุบันมีการใช้งานไม่มากนัก จึงถูกยกเลิกไป โดยแสดงรูปร่างและตำแหน่งขาในรูปที่ 2.27



(ก) คอนเน็กเตอร์อนุกรม 9 ขาหรือแบบ DB-9 (มองจากด้านหลังคอมพิวเตอร์) (ข) คอนเน็กเตอร์อนุกรม 25 ขาหรือแบบ DB-25 (มองจากด้านหลังคอมพิวเตอร์)

คอนเน็กเตอร์ DB-9	คอนเน็กเตอร์ DB-25	ชื่อของสายสัญญาณ	ชนิดของสายสัญญาณ
1	3	Data Carrier Detect : DCD	อินพุต
2	3	Received Data : RxD	อินพุต
3	2	Transmitted Data : TxD	เอาต์พุต
4	20	Data Terminal Ready : DTR	เอาต์พุต
5	7	Signal Ground : GND	-
6	5	Data Set Ready : DSR	อินพุต
7	4	Request To Send : RTS	เอาต์พุต
8	5	Clear To Send : CTS	อินพุต
9	22	Ring Indicator : RI	อินพุต

รูปที่ 2.27 การจัดขาของคอนเน็กเตอร์พอร์ตอนุกรมตามมาตรฐาน RS - 232 ทั้งแบบ DB - 9 และ DB - 25



รูปที่ 2.28 การต่ออุปกรณ์ภายนอกกับพอร์ตอนุกรมของคอมพิวเตอร์ในลักษณะต่าง ๆ

สำหรับการเชื่อมต่อคอมพิวเตอร์กับอุปกรณ์ภายนอกแสดงดังในรูปที่ 2.28 ลูกศรในรูปแสดงถึงทิศทางของข้อมูลในรูปที่ 2.28 (ก) เป็นการเชื่อมต่อแบบ Null modem หรือการเชื่อมต่อโดยตรงโดยไม่ต้องผ่านโมเด็ม โดยมีการตรวจสอบหรือแฮนด์เช็กเต็มรูปแบบ ส่วนในรูปที่ 2.28 (ข) เป็นการเชื่อมต่อแบบ Null modem ในลักษณะที่ใช้สายสัญญาณเพียง 3 เส้น โดยเส้นหนึ่งสำหรับส่งข้อมูล อีกเส้นหนึ่งสำหรับรับข้อมูล และเส้นสุดท้ายเป็นกราวด์ สำหรับรายละเอียดหน้าที่การทำงานในแต่ละขาของพอร์ตอนุกรม RS - 232 มีดังนี้

- **Data Carrier Detect : DCD** หรืออาจเรียกว่า **Carrier Detect : CD** ขานี้จะแอกติฟเมื่อมีการส่งสัญญาณพาจากอุปกรณ์สื่อสารข้อมูล เช่น โมเด็ม สำหรับการใช้งานปกติ ขานี้จะไม่ได้ถูกใช้งานมากนัก
- **Receive Data : RD หรือ RxD** ขานี้ใช้เพื่อรับสัญญาณอนุกรมเข้ามายังคอมพิวเตอร์โดยนำข้อมูลที่อ่านได้เก็บไว้ในรีจิสเตอร์ บัฟเฟอร์
- **Transmitted Data : TD หรือ TxD** ขานี้ใช้เพื่อส่งข้อมูลออกจากคอมพิวเตอร์ โดยนำเอาข้อมูลที่เก็บอยู่ในบัฟเฟอร์สำหรับส่งข้อมูลออกไป
- **Data terminal Ready : DTR** เป็นขาสัญญาณที่ส่งออกจากคอมพิวเตอร์เพื่อให้อุปกรณ์ปลายทางรับรู้ว่า ต้องการติดต่อด้วย โดยขา DTR นี้จะต้องเชื่อมต่อกับขา DSR ของอุปกรณ์ปลายทางและขา DTR ของอุปกรณ์ปลายทางจะต้องเชื่อมต่อกับขา DSR ของคอมพิวเตอร์ถ้าใช้การเชื่อมต่อเป็นแบบ Null Modem ซึ่งใช้สายในการเชื่อมต่อเพียง 3 เส้น จะต้องต่อขา DTR และ DSR ของตัวมันเองเข้าด้วยกัน และต้องต่อกับขา DCD ด้วยในกรณีที่โปรแกรมสื่อสารที่ใช้มีการตรวจจับสัญญาณพาห์
- **Signal Ground : GND** ขากราวด์ของระบบ
- **Data Set Ready : DSR** ขานี้จะใช้คู่กับขา DTR เพื่อตรวจสอบการเชื่อมต่อกันระหว่างคอมพิวเตอร์กับอุปกรณ์ปลายทาง ซึ่งขา DSR นี้จะเป็นขาสำหรับรับข้อมูลจากภายนอกซึ่งถูกส่งมาจากขา DTR
- **Request To Send : RTS** เป็นขาสำหรับส่งสัญญาณร้องขอให้ทางอุปกรณ์ปลายทางส่งข้อมูลกลับมายังคอมพิวเตอร์ โดยขาที่รับสัญญาณ RTS ก็คือขา CTS ในกรณีที่ใช้การเชื่อมต่อแบบ Null modem 3 สาย จะต้องเชื่อมต่อกับขา RTS และ CTS ของตัวมันเองเข้าด้วยกัน เพื่อจะให้การรับและส่งข้อมูลสามารถเกิดขึ้นได้ตลอดเวลา
- **Clear To Send : CTS** ขานี้จะคอยรับสัญญาณจาก RTS เมื่อรับสัญญาณได้ ข้อมูลที่ขา TxD จะถูกส่งออกไป ดังนั้นขานี้จึงถูกใช้เพื่อตรวจสอบอุปกรณ์ต่อพ่วงว่าพร้อมที่จะรับข้อมูลหรือไม่
- **Ring Indicator : RI** ใช้แสดงสถานะสัญญาณเรียกจากสายโทรศัพท์ ปกติในการสื่อสารโดยทั่วไปสายนี้จะถูกใช้งาน จะใช้งานเมื่อมีการเชื่อมต่อกับโมเด็มและโปรแกรมมีการตรวจสอบสัญญาณนี้เท่านั้น

2.5.2 UART

UART มาจากคำว่า **Universal Asynchronous Receiver Transmitter** ซึ่งหมายถึงอุปกรณ์ที่ทำหน้าที่รับและส่งข้อมูลแบบอะซิงโครนัสนั่นเอง สำหรับการสื่อสารอนุกรมบนคอมพิวเตอร์แล้ว UART ถือว่าเป็นหัวใจสำคัญของการสื่อสารอนุกรม

หน้าที่หลักของ UART คือทำหน้าที่แปลงข้อมูลที่อยู่ในรูปแบบขนานจากคอมพิวเตอร์ให้อยู่ในรูปแบบอนุกรมแบบอะซิงโครนัส แล้วส่งออกไป และทำหน้าที่แปลงสัญญาณอนุกรมแบบอะซิงโครนัสที่ป้อนเข้ามายัง UART ให้เป็นแบบขนานก่อนที่จะส่งเข้าสู่คอมพิวเตอร์ ซึ่งนอกจาก UART จะส่งข้อมูลไป

ยังคอมพิวเตอร์แล้ว ยังแจ้งข้อมูลอื่น ๆ ให้คอมพิวเตอร์รับทราบด้วย เช่น อัตราเร็วในการรับส่งข้อมูล (บอดเรต), รูปแบบการส่งข้อมูล, ความผิดพลาดที่เกิดขึ้นระหว่างการถ่ายทอดข้อมูล (ผิดพลาดจากพาริตี, เฟรมข้อมูล, โอเวอร์รัน) เป็นต้น

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

2.5.3 ชนิดของ UART

ในเครื่องคอมพิวเตอร์ทั่วไปมี UART ที่ใช้งานกันอยู่ 2 เบอร์ คือ 8250 ซึ่งเป็น UART มาตรฐานที่มีใช้กันมาช้านาน UART เบอร์นี้จะมีบัฟเฟอร์สำหรับรับและส่งข้อมูลตำแหน่งเดียวกัน ทำให้การรับและส่งข้อมูลถูกจำกัดความเร็วอยู่ที่ 57.6 กิโลบิตต่อวินาทีเท่านั้น แต่ UART เบอร์นี้ถือว่าเป็นต้นแบบของ UART ที่ใช้คอมพิวเตอร์ โดยคอมพิวเตอร์ทุก ๆ รุ่นจะต้องสนับสนุนการทำงานตามรูปแบบของ UART เบอร์นี้

UART อีกเบอร์หนึ่งคือ 16450 มีความสามารถรับส่งข้อมูลได้ด้วยความเร็ว 115,200 บิต ต่อวินาที และเพิ่มรีจิสเตอร์สำหรับพักข้อมูลสำหรับ UART นอกจากนั้นยังเพิ่มส่วนของชิฟต์รีจิสเตอร์ แบบ FIFO (First In First Out) ขนาด 16 ไบต์เข้าไป ทำให้สามารถสนับสนุนความเร็วในการรับส่งข้อมูลที่ 256 กิโลบิตต่อวินาทีได้ โดยคอมพิวเตอร์ในปัจจุบันใช้ UART เบอร์นี้หรือใหม่กว่านี้ เช่นเบอร์ TL16C750 ซึ่งมีรีจิสเตอร์แบบ FIFO ขนาด 64 ไบต์ ทำงานได้ที่ระดับแรงดัน +5V และ +3V มีโหมดประหยัดพลังงานสามารถรับส่งข้อมูลได้ด้วยความเร็ว 1 เมกะบิตต่อวินาทีเมื่อใช้สัญญาณนาฬิกา 16 MHz

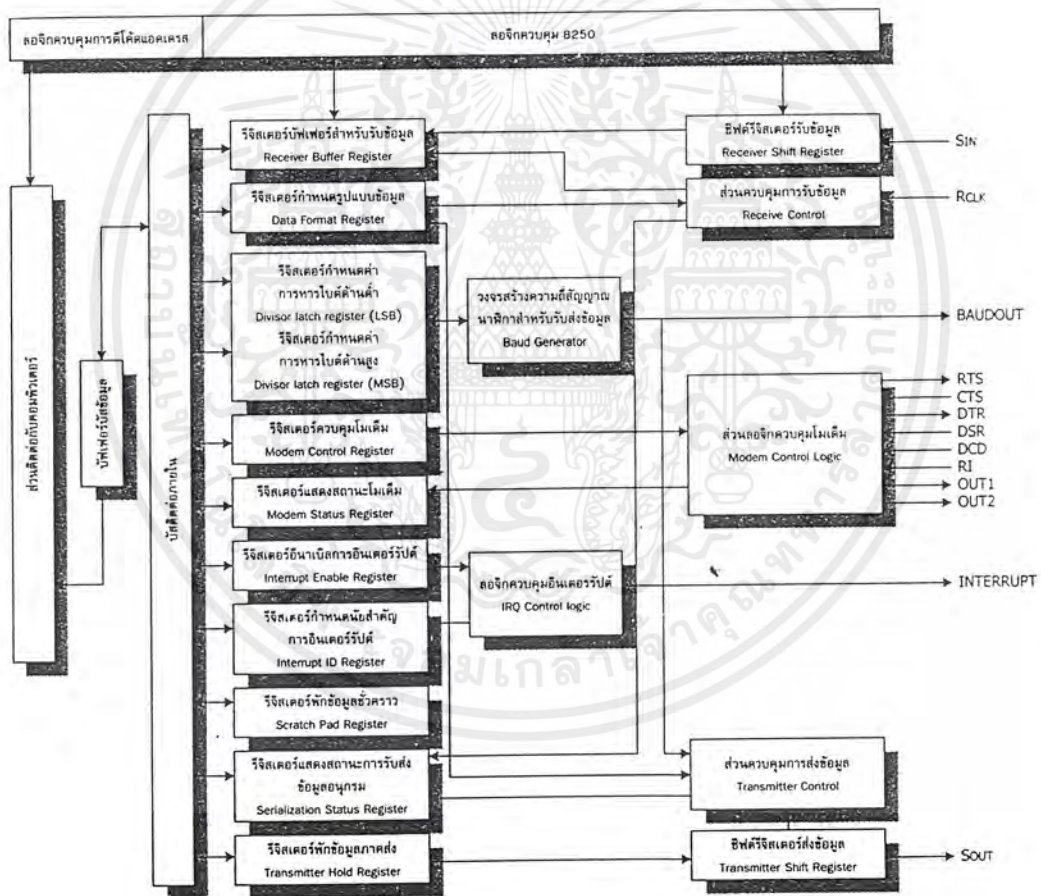
อย่างไรก็ตามความเร็วในการส่งข้อมูลที่มากมายของ UART เบอร์ใหม่ ๆ ก็ไม่ได้ช่วยให้การรับส่งข้อมูลของคอมพิวเตอร์เร็วขึ้น เนื่องจากว่าคอมพิวเตอร์ยังใช้ความถี่ของสัญญาณนาฬิกาในการแปลงข้อมูลเพียง 1.8432 MHz เท่านั้น

2.5.4 วงจรภายในและรีจิสเตอร์ของพอร์ตอนุกรม RS – 232

เครื่องคอมพิวเตอร์โดยทั่วไปสามารถต่อพอร์ตอนุกรม RS – 232 สูงสุดได้ 4 พอร์ต ซึ่งจะมียี่ห้อเรียกเป็น COM1, COM2, COM3 และ COM4 ซึ่งพอร์ตอนุกรมแต่ละตัวต่างก็ใช้งาน UART ภายในคอมพิวเตอร์ในการติดต่อกับอุปกรณ์ภายนอกเช่นเดียวกัน

ในรูปที่ 2.29 แสดงไดอะแกรมการทำงานภายในพอร์ตอนุกรม ซึ่งประกอบไปด้วยรีจิสเตอร์ขนาด 8 บิต 8 ตัวที่ใช้งานร่วมกับ UART แอดเดรสของรีจิสเตอร์ภายในพอร์ตอนุกรมสามารถคำนวณได้จากค่ารีจิสเตอร์พื้นฐานของพอร์ตอนุกรม ยกตัวอย่าง พอร์ตอนุกรม COM1 มีแอดเดรสอยู่ที่ 3F8H ตำแหน่งของรีจิสเตอร์ต่าง ๆ จะเป็นตำแหน่งที่บวกเข้าไปกับค่า 3F8H โดยรีจิสเตอร์ที่ใช้งานกับพอร์ตอนุกรมมีดังนี้

- 00H เป็นรีจิสเตอร์ไบเฟอร์สำหรับเก็บข้อมูลที่รับเข้ามาหรือเตรียมข้อมูลก่อนที่จะส่งออกไป
- 01H รีจิสเตอร์อีนابلการอินเตอร์รัปต์ ใช้ในการเซตโหมดการอินเตอร์รัปต์ของพอร์ตอนุกรม
- 02H รีจิสเตอร์แสดงโหมดการอินเตอร์รัปต์ ใช้เพื่อตรวจสอบโหมดของการอินเตอร์รัปต์เมื่อมีการอินเตอร์รัปต์เกิดขึ้น
- 03H รีจิสเตอร์กำหนดรูปแบบของข้อมูล
- 04H รีจิสเตอร์ควบคุมโมเด็ม ใช้ตรวจสอบบิตสำหรับติดต่อกับโมเด็ม เช่น RTS หรือ DTR
- 05H รีจิสเตอร์แสดงสถานะการรับและการส่งข้อมูลแบบอนุกรม
- 06H รีจิสเตอร์แสดงสถานะของโมเด็ม ซึ่งจะแสดงสถานะของขา DCD, RI, DSR และ CTS
- 07H รีจิสเตอร์สำหรับการเก็บข้อมูลชั่วคราว



รูปที่ 2.29 โค้ดแแกรมการทำงานภายในของพอร์ตอนุกรมของเครื่องคอมพิวเตอร์

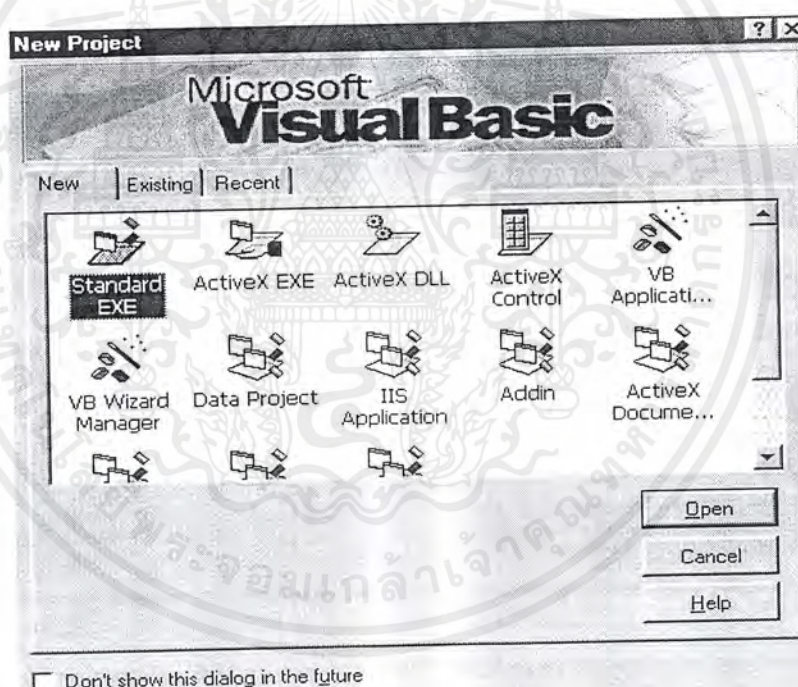
2.6 Visual Basic เบื้องต้น

เราจะพบว่าในปัจจุบันนี้ windows เป็นระบบปฏิบัติการที่นิยมมาก และการใช้ windows ก็แพร่หลายไปทั่วโลก ไม่ว่าใครก็สามารถใช้ windows อย่างง่ายดายนั้นเพราะ windows สามารถใช้งานและเรียนรู้ได้ง่าย และจุดเด่นที่สำคัญอีกอย่างนั้นก็คือมันสามารถที่จะทำงานหลายอย่างพร้อมๆ กันได้ (Multitasking) ดังนั้น Software ที่เป็นที่ยอมรับปัจจุบันนี้จึงเป็นโปรแกรมที่สามารถทำงานบน windows ได้

Visual Basic เป็นโปรแกรมภาษาของคอมพิวเตอร์ที่พัฒนาให้สามารถใช้งานได้ง่ายและสามารถพัฒนาโปรแกรมบน windows แต่ก่อนจะพบว่าการเขียนโปรแกรมบน windows การที่จะเขียนหน้าต่างของ windows ขึ้นมานั้นจะต้องใช้คำสั่งต่างๆมากมาย แต่ถ้าเป็น Visual Basic แล้ว สามารถสร้างได้ง่ายดาย ต่อไปจะได้กล่าวถึงการใช้งาน Visual Basic 6.0

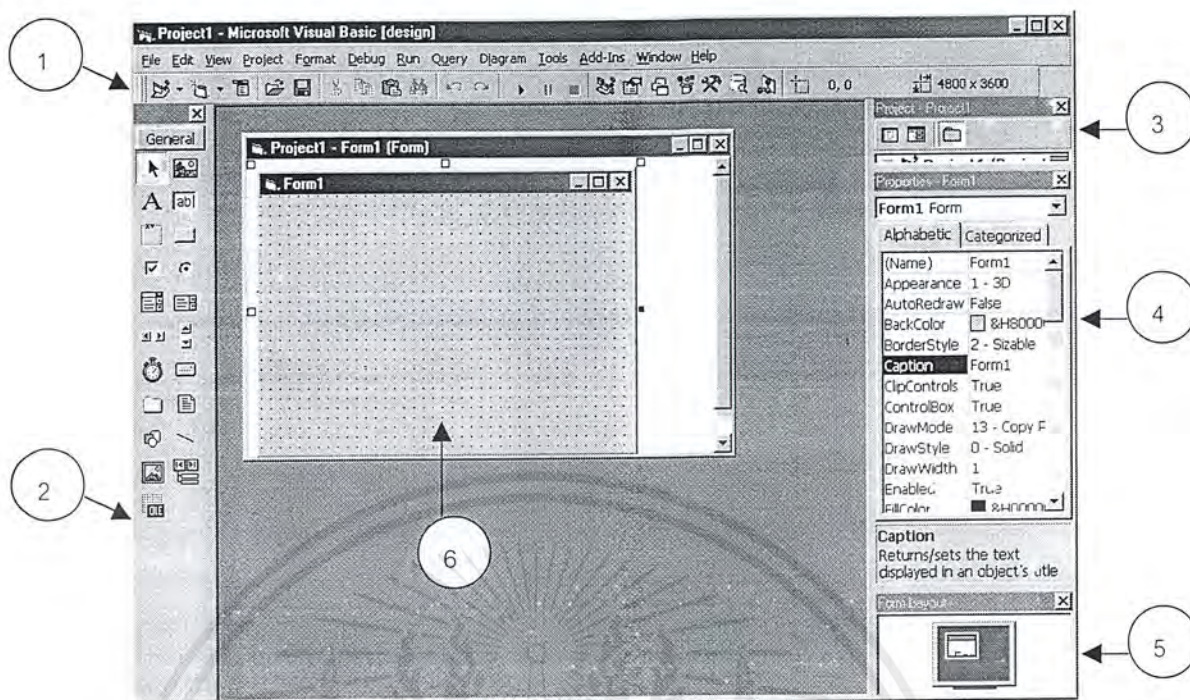
2.6.1 การใช้งาน Visual Basic เบื้องต้น

สิ่งแรกที่จะพบเมื่อเข้าสู่โปรแกรมก็คือหน้าจอสำหรับเปิด project ดังรูป 2.30



2.30 หน้าจอสำหรับเปิด project ใหม่

ในนี้จะเลือก “ Standard EXE “ เพื่อเข้าสู่การใช้ Visual Basic ที่ใช้ในการเขียนโปรแกรมต่อไป
 ดังรูปที่ 3.31

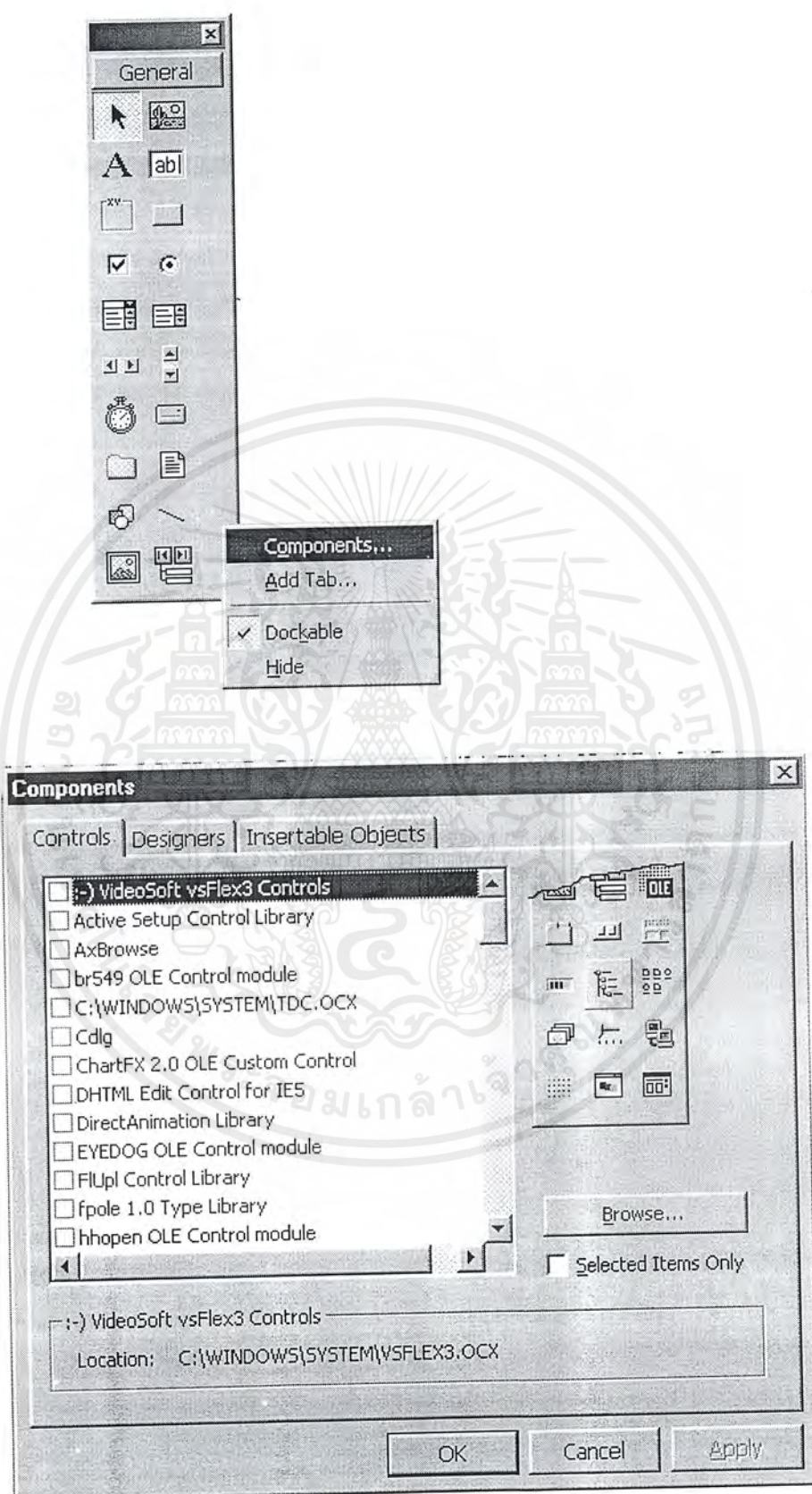


รูปที่ 2.31 รูปแสดงจอภาพของ Visual basic

2.6.2 ส่วนประกอบต่างๆของ Visual basic

1. Toolbar จะประกอบด้วย icon ที่ใช้ในการพัฒนา โปรแกรม เหมือนกับในโปรแกรมอื่นๆ
2. Toolbox จะประกอบด้วย icon ต่างๆ ซึ่งใน Visual Basic เราเรียกว่า control
3. Project Explorer Windows จะแสดงรายชื่อ Form ที่พร้อมจะถูกแก้ไข
4. Properties Windows ใช้สำหรับกำหนดคุณสมบัติ (Property) ให้กับ Form และ Object ต่างๆ ที่อยู่บน Form
5. Form Layout Windows จะแสดงตำแหน่งของ Form บนจอคอมพิวเตอร์เมื่อทำการ Run
6. Form เป็นส่วนที่ใช้ในการสร้างจอเมื่อทำการรัน แล้วจะให้หน้าจออย่างไรซึ่งเราสามารถที่จะตกแต่งให้สวยงามได้ และเป็นส่วนที่เราสามารถเลือกเอา Control ต่างๆ มาวางไว้และสามารถจัดตำแหน่งได้

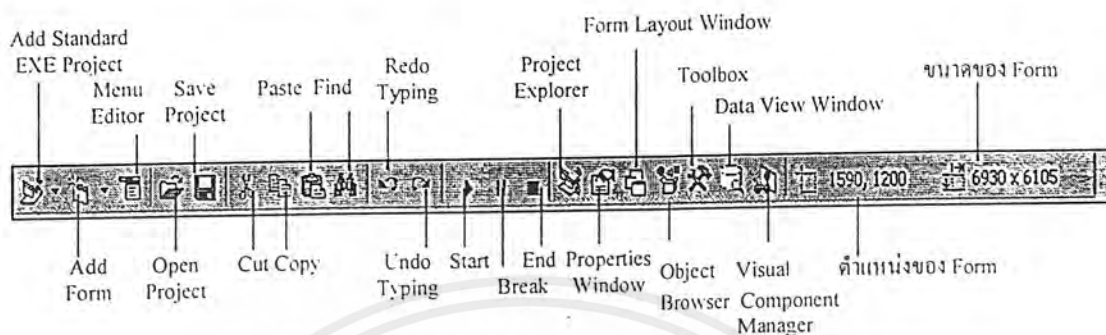
ส่วนของ Control ต่างๆ ที่สามารถดึงมาใช้ได้นอกจากนี้ยังมีอีกส่วน โดยการคลิกขวาที่บริเวณสีเทาแล้วเลือก Component แล้วทำการเลือก Control ที่ต้องการ ดังรูป 2.32



รูปที่ 2.32 แสดงการเลือก Control ต่างๆที่ต้องการใช้งาน

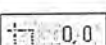
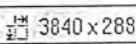
2.6.3 ส่วนของ Toolbar

เป็นแถบเครื่องมือที่ประกอบด้วย Icon ต่างๆ ดังรูปที่ 2.33 จะมีลักษณะที่คล้ายกับ Toolbar ของ MS-Word และมีหน้าที่ที่คล้ายๆ กัน



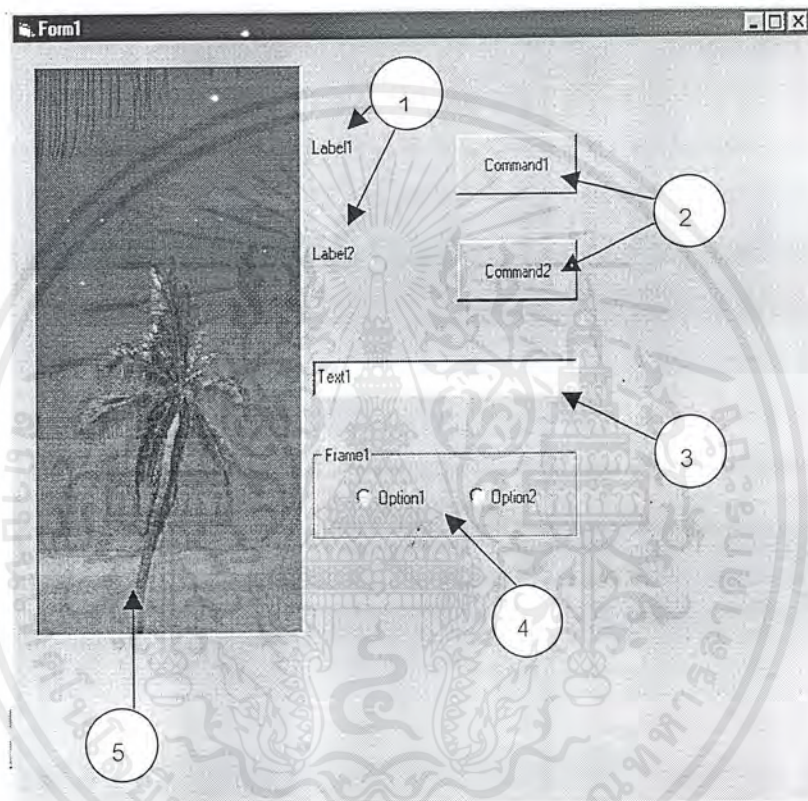
รูปที่ 2.33 แถบเครื่องมือที่ประกอบด้วย Icon ต่างๆ

รูป Icon	วิธีการ	รายละเอียดการใช้งาน
	Add Standard EXE-Project	ใช้สำหรับเปิด Project ใหม่ ในกรณีที่ไม่มีหลาย Project อยู่บนจอภาพ สามารถสลับไปมาระหว่าง Project ได้โดยเข้าไปเลือกในเมนู Windows
	Add Form	ใช้ในการเพิ่ม Form ให้กับ Project ซึ่งอาจเรียกจากเมนู Project และ Add Form ตามลำดับ
	Menu Editor	ใช้เรียก Menu Editor ซึ่งเป็น Tool สำหรับสร้างเมนูให้กับ Form ซึ่งอาจใช้การกดปุ่ม Ctrl+E หรือเลือกจากเมนู Tools และ Editor ตามลำดับ
	Open Project	ใช้สำหรับเปิด Project ซึ่งอาจใช้ Hot Keys Ctrl+O หรือเรียกจากเมนู File และ Open Project ตามลำดับ
	Save Project	ใช้สำหรับบันทึก Project และ Form ซึ่งอาจเลือกจากเมนู File และ Save Project หรือ Save Project As ตามลำดับ ถ้าต้องการระบุชื่อของ Project: ใหม่
	Cut	ใช้สำหรับตัด Object ต่าง ๆ บน Form ซึ่งอาจใช้ Hot Keys Ctrl+X หรือเลือกจากเมนู Edit และ Cut ตามลำดับ
	Copy	ใช้สำหรับ Copy Object บน Form ซึ่งอาจใช้ Hot Keys Ctrl+C หรือเลือกจากเมนู Edit และ Copy ตามลำดับ สำหรับ Object ที่ถูก Copy จะเรียกว่า Control Array ซึ่งจะกล่าวถึงในส่วนต่อไป
	Paste	ใช้สำหรับ Paste Object ที่ Cut หรือ Copy ไว้ ซึ่งอาจใช้ Hot Keys Ctrl+V หรือเลือกจากเมนู Edit และ Paste ตามลำดับ
	Find	ใช้สำหรับค้นหาคำใน Editor ซึ่งเป็น Tool ที่ใช้ในการเขียนโปรแกรม โดยอาจใช้ Hot Keys Ctrl+F หรือเลือกจากเมนู Edit และ Find ตามลำดับ

รูป Icon	ชื่อเรียก	รายละเอียดการใช้งาน
	Undo Typing	ใช้สำหรับยกเลิกค่าที่พิมพ์ใน Editor ซึ่งอาจใช้ Hot Keys Ctrl+Z หรือเลือกจากเมนู Edit และ Undo Typing ตามลำดับ ในกรณีที่ไม่ได้อยู่ในจอภาพ Editor จะปรากฏข้อความ Can't Undo แทนทั้งในส่วนของการชี้ Icon และในเมนู Edit
	Redo Typing	ใช้สำหรับทำซ้ำค่าที่พิมพ์ใน Editor ซึ่งอาจเลือกจากเมนู Edit และ Redo Typing ตามลำดับ และก็เช่นเดียวกับ Undo Typing ในกรณีที่ไม่ได้อยู่ในจอภาพ Editor ปรากฏข้อความ Can't Redo แทน
	Start	ใช้สำหรับ Run Project ที่จัดทำขึ้น ซึ่งอาจใช้ Hot Keys F5 หรือเลือกจากเมนู Run และ Start ตามลำดับ
	Break	ใช้สำหรับหยุดการทำงานของ Project ชั่วคราว ซึ่งอาจใช้ Hot Keys Ctrl+Break หรือเลือกจากเมนู Run และ Break ตามลำดับ
	End	ใช้สำหรับหยุดการทำงานของ Project ซึ่งอาจเลือกจากเมนู Run และ End ตามลำดับ
	Project Explorer	ใช้แสดงว่า Project นั้นประกอบไปด้วย Form และ Module ใดบ้าง ซึ่งอาจใช้ Hot Keys Ctrl-R หรือเลือกจากเมนู View และ Project Explorer ตามลำดับ
	Properties Window	ใช้สำหรับกำหนดคุณสมบัติ (Property) ของ Object และ Form ซึ่งอาจใช้ Hot Keys F4 หรือเลือกจากเมนู View และ Properties Window ตามลำดับ
	Form layout - Window	ใช้สำหรับเรียกจอภาพ Form Layout ซึ่งใช้แสดงตำแหน่งของ Form บนจอภาพ โดยอาจเลือกจากเมนู View และ Form Layout: Window ตามลำดับ
	Object Browser	ใช้สำหรับเรียกจอภาพ Object Browser ซึ่งใช้แสดงถึง Class และสมาชิกของแต่ละ Class อาจใช้ Hot Keys F2 หรือเลือกจากเมนู View และ Object Browser ตามลำดับ
	Toolbox	ใช้สำหรับเรียก Toolbox ขึ้นมาบนจอภาพ ซึ่งอาจเลือกจากเมนู View และ Toolbar ตามลำดับ
	ตำแหน่งของ Form	ใช้บอกตำแหน่งในแกน X และ Y ของ Form
	ขนาดของ Form	ใช้บอกขนาดของ Form ตามแนวแกน X และ Y

ตารางที่ 2.5 แสดงรายละเอียดการใช้งาน Toolbar ต่างๆ

Visual Basic เป็นโปรแกรมที่ง่ายต่อการเรียนรู้ และมีขั้นตอนที่ง่ายด้วยอาทิเช่นว่า เมื่อเราต้องการที่จะต้องการให้หน้าจอของเราเป็นอย่างไรก็แต่เพียงดึงเอา Control นั้นๆ มาวางไว้บน Form จึงทำให้เราไม่ต้องเขียน คำสั่งต่างๆ มากมายจึงทำให้ผู้ใช้สะดวกมาก บนหน้าจอของ Form ก็คือส่วนที่เราจะแสดงเมื่อมีการ Run เราสามารถที่จะตกแต่งให้ดูสวยงามได้ซึ่งจะยกตัวอย่างการเขียน Form และการดึง Control มาใช้งาน



รูปที่ 2.34 แสดงตัวอย่างการเขียน Form และการดึง Control มาใช้งาน

หมายเลข 1 เป็นการใช้ Control ชื่อ Label เมื่อเราดึงมาวางบน Form เราก็จะสามารถที่พิมพ์ข้อความใส่รูปแบบตัวอักษรหรือใส่สีให้ข้อความได้

หมายเลข 2 เป็นการใช้ Control ชื่อ CommandButton เป็นลักษณะของปุ่มให้เราคลิกซึ่งจะพบว่าเราจะคุ้นเคยกับ Control ตัวนี้มากในการใช้งาน Windows ซึ่งเราสามารถที่จะเขียน โปรแกรมให้กับตัวมัน โดยมองลักษณะที่ว่าเมื่อเกิดการคลิกแล้วให้ทำงานใดๆ ก็ได้

หมายเลข 3 เป็นการใช้ Control ชื่อ TextBox เป็นลักษณะ กล่องข้อความใช้แสดงข้อความ

หมายเลข 4 เป็นการใช้ Control Frame และ Control OptionButton ร่วมกันลักษณะเช่นนี้ก็จะพบบ่อยคือเราอาจจะเขียน โปรแกรมให้ผู้ใช้เลือกอย่างใดอย่างหนึ่งจึงเอามารวมเป็น Frame เดียวกัน

หมายเลข 5 เป็นการ ใช้ Control PictureBox ก็เป็นการใส่รูปภาพเพื่อเป็นการจัดให้โปรแกรมของเรา มี Presentation ที่สวยงาม

From ตัวอย่างข้างบนทำให้มองเห็นภาพการใช้งานของ Visual Basic ได้ชัดเจนขึ้นจะพบว่าเราสามารถที่จะจัดรูปแบบตามที่เรต้องการได้นั้น Visual Basic จะเป็น presentation ที่ใช้ได้ทีเดียวแถมยังสามารถที่จะเขียนคำนวณ และพัฒนาโปรแกรมที่ซับซ้อน อาทิเช่น พวกโปรแกรมที่ติดต่อกับอุปกรณ์ Input / Output , งานทางด้านฐานข้อมูล , Application ทางด้าน Internet และยังมีด้านต่างๆมากมาย ดังนั้นจึงพบว่า Visual Basic เป็นโปรแกรมภาษาที่น่าสนใจทีเดียว ที่สามารถรวมเอา Presentation ที่สวยงาม แดงง่ายและรวดเร็ว กับ ความสามารถที่สูงในการคำนวณและการพัฒนาโปรแกรม

ที่ยกตัวอย่างการใช้งานของ Control บาง Control เท่านั้นเองยังมี Control ที่ไม่ได้กล่าวถึงอีกมากมายและแต่ละ Control ก็มีเหตุการณ์ที่จะเกิดขึ้นกับตัวมันได้หลายเหตุการณ์เช่นกันที่กล่าวไว้เป็นเพียงการแนะนำเท่านั้นเอง ปัจจุบัน Visual Basic เป็นโปรแกรมที่น่าสนใจและมีหนังสือที่อธิบายไว้ค่อนข้างละเอียดสามารถที่จะทำตามเป็นขั้นตอนได้ แต่หนังสือมีในท้องตลาดนั้นจะกล่าวไว้เพียงส่วนที่เป็นการใช้งาน และพัฒนาด้านสำนักงาน ที่มี Presentation ที่สวยงาม และการคำนวณเบื้องต้น ท่านที่สนใจในงานวิศวกรรมก็ควรที่จะอ่านส่วนที่เป็นพื้นฐานให้เข้าใจและยังมีอีกแหล่งข้อมูลที่รวบรวมการใช้งานและยังมีโปรแกรมตัวอย่างไว้ค่อนข้างละเอียดก็คือ Library ของ Visual Basic ที่มีขายตามท้องตลาด

ตัวอย่างโปรแกรมติดต่อกับบอร์ดอินเตอร์เฟส เพื่อรับและส่งค่าข้อมูล เพื่อเปรียบเทียบให้เห็นถึงความแตกต่างระหว่างการโปรแกรมที่เขียนด้วยภาษา C กับการเขียนด้วย Visual Basic

```
#include<stdio.h>
#include<dos.h>
#include<conio.h>
#define PORT1 0x2F8
void main(void)
{
    int c;
    int ch;
    clrscr();
    outportb(PORT1+1,0x00);
    outportb(PORT1+3,0x80);
    outportb(PORT1+0,0x0C);
    outportb(PORT1+1,0x00);
    outportb(PORT1+3,0x03);
    outportb(PORT1+2,0xC7);
    outportb(PORT1+4,0x0B);
    outportb(PORT1+0,0xFF);
```

```

delay(5000);
outportb(PORT1+0,0x00);
do{
c=inportb(PORT1+5);
if (c&1)
{
ch=inportb(PORT1);clrscr();
printf("press ESC to quit. input data is %X==: ",ch);
outportb(PORT1,ch);
}
if(kbhit())
{
ch=getch();
outportb(PORT1,0x00);
}
}while(ch!=27);
}

```

Form1

โปรแกรมทดสอบการทำงานติดต่อกับ Port อนุกรม

รับข้อมูล ส่งข้อมูล

ป้อนเลขช่องออก Port เป็นฐาน 10 2

ข้อมูลที่ได้รับมาจาก อุปกรณ์ภายนอก 255

End

รูปที่ 2.35 แสดงผลโปรแกรมทดสอบการทำงานติดต่อกับ Port อนุกรม

ใช้ Visual Basic

```
Private Sub Command1_Click()
    Command2 = False
    MSComm1.CommPort = 2
    MSComm1.Settings = "9600,N,8,1"
    MSComm1.InputLen = 0
    MSComm1.PortOpen = True
    MSComm1.Output = Chr(Text1.Text)

    MSComm1.PortOpen = False
End Sub
```

- เมื่อมีการคลิกที่ปุ่ม ส่งข้อมูล
- ให้อุปกรณ์รับข้อมูลไม่ทำงานหรือไม่สามารถคลิกได้
- ติดต่อกับ port Com 2
- เซ็ตค่าการรับส่งข้อมูลของ port Com2
- เปิดพอร์ต
- เอาข้อมูลที่เก็บไว้ใน textbox ที่เราป้อนเข้ามาออกไปที่พอร์ต
- ปิดพอร์ต

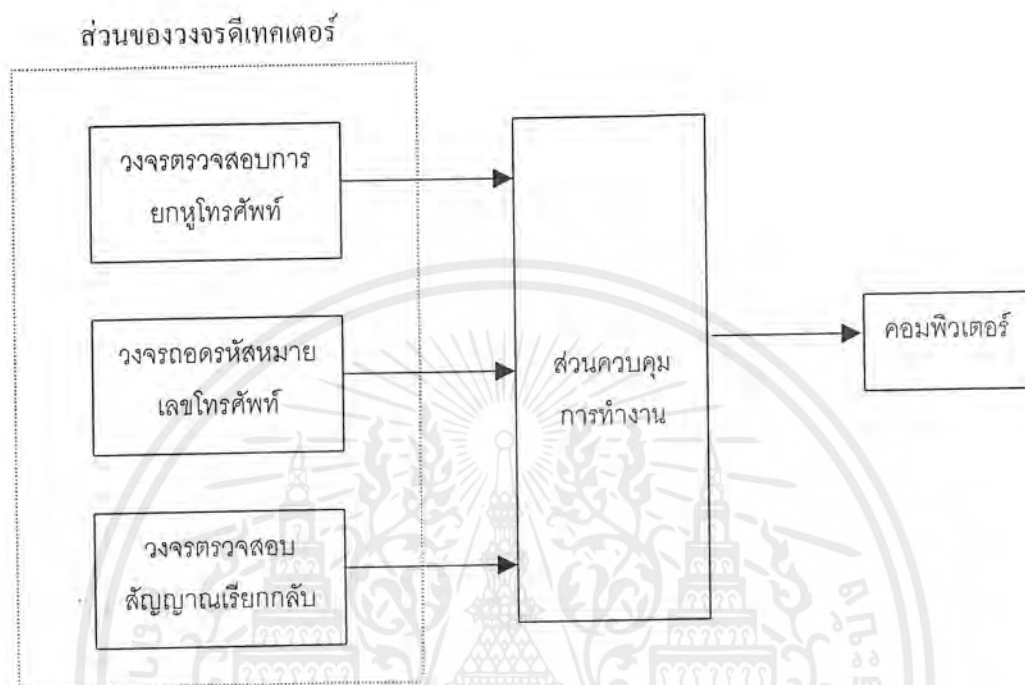
```
Private Sub Command2_Click()
    Dim InString As String
    Command1 = False
    MSComm1.CommPort = 2
    MSComm1.Settings = "9600,N,8,1"
    MSComm1.InputLen = 0
    MSComm1.PortOpen = True
    InString = MSComm1.Input
    Text2.Text = Asc(InString)
    MSComm1.PortOpen = False
End Sub
```

```
Private Sub Command3_Click()
End
End Sub
```

จะเห็นว่า Visual Basic จะมีการแสดงผลที่สวยงามและยังง่ายต่อการใช้งานทำให้ User ที่ใช้โปรแกรมสะดวกสบายและตัว Sourcecode ก็ศึกษาทำความเข้าใจง่าย จากที่ได้กล่าวไว้ตั้งแต่ตอนต้นแล้วว่า Visual Basic จะมองเป็นวัตถุ (Object) แต่ละวัตถุก็แตกต่างกันออกไปแล้วแต่เราจะเลือกใช้ Control อะไร แล้ววัตถุแต่ละวัตถุก็มีคุณสมบัติหลายอย่างซึ่งเราสามารถกำหนดให้มันได้

บทที่ 3

การคำนวณและการสร้าง



รูปที่ 3.1 บล็อกไดอะแกรมแสดงส่วนประกอบของโครงงาน

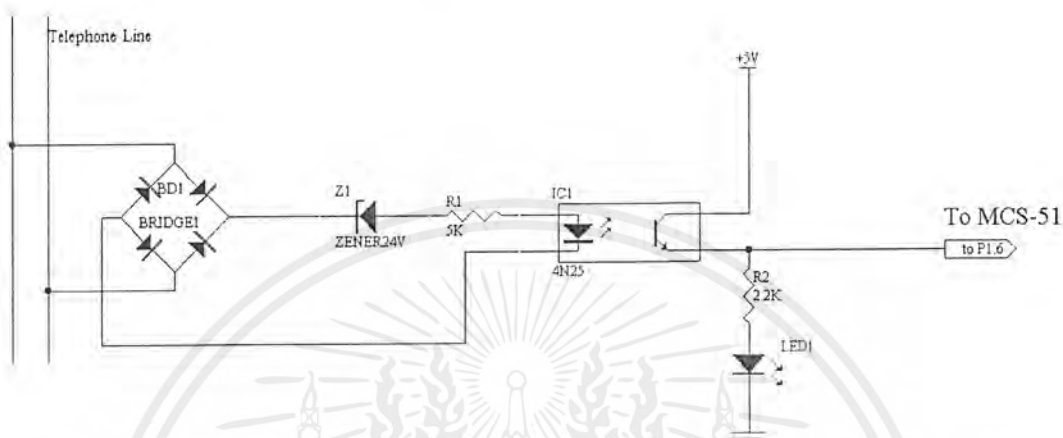
3.1 ส่วนของวงจรดีเทคเตอร์

วงจรส่วนนี้ประกอบด้วย 3 วงจรย่อย ได้แก่ วงจรตรวจสอบการยกหูโทรศัพท์ (Hook Switch Detector) วงจรถอดรหัสหมายเลขโทรศัพท์ (DTMF Decoder Circuit) และวงจรตรวจสอบสัญญาณเรียกกลับ (Ringback Tone Detector Circuit) วงจรเหล่านี้จะต่ออยู่กับคู่สายโทรศัพท์ของเครื่องที่ต้องการจะแสดงข้อมูลการใช้งานโทรศัพท์ เพื่อรับสัญญาณอินพุตจากคู่สายโทรศัพท์นั้น และเอาต์พุตของวงจรจะส่งไปยังส่วนควบคุมการทำงาน

3.1.1 วงจรตรวจสอบการยกหูโทรศัพท์ (Hook Switch Detector)

หลักการทำงานของวงจร คือ ใช้การเปรียบเทียบของแรงดันระหว่างการยกหูและการวางหู ซึ่งในขณะที่วางหูอยู่นั้นจะมีแรงดันตกคร่อมคู่สายโทรศัพท์ประมาณ 48 โวลต์ และในขณะที่ทำการวางหูจะทำให้แรงดันตกลงเหลือประมาณ 6 – 10 โวลต์ ซึ่งก็แล้วแต่ค่าความต้านทานของเครื่องโทรศัพท์แต่ละเครื่องพิจารณาในขณะที่ทำการวางหู แรงดันไฟ 48 โวลต์จะผ่านวงจรบริดจ์เรกติไฟเออร์ซึ่งจะให้ขั้วของแรงดันแน่นอน แรงดันดังกล่าวมีค่ามากกว่าแรงดัน zener ของซีเนอร์ไดโอดขนาด 24 โวลต์ เป็นผลทำให้มีกระแสไหลผ่านความต้านทาน 5 กิโลโอห์ม จึงทำให้มีกระแสไหลผ่านแอลอีดีในออปโตไอโซเลเตอร์ และไปกระตุ้นให้ทรานซิสเตอร์นำกระแสตามไปด้วย เกิดแรงดันตกคร่อมตัวต้านทาน 1 กิโลโอห์มที่

เอาต์พุตเป็น 5 โวลต์ ซึ่งแทนด้วยระดับลอจิก “1” พิจารณาในขณะที่ทำให้การยกหู แรงดันจะตกลงเหลือประมาณ 6 – 9 โวลต์ แรงดันดังกล่าวมีค่าน้อยกว่าแรงดันซีเนอร์ จึงไม่มีกระแสไหลผ่านซีเนอร์ไปได้ เป็นผลทำให้แอลอีดีและทรานซิสเตอร์ในออปโตไอโซเลเตอร์ไม่นำกระแส แรงดันที่เอาต์พุตจึงตกลงเป็น 0 โวลต์ ซึ่งแทนด้วยระดับลอจิก “0”



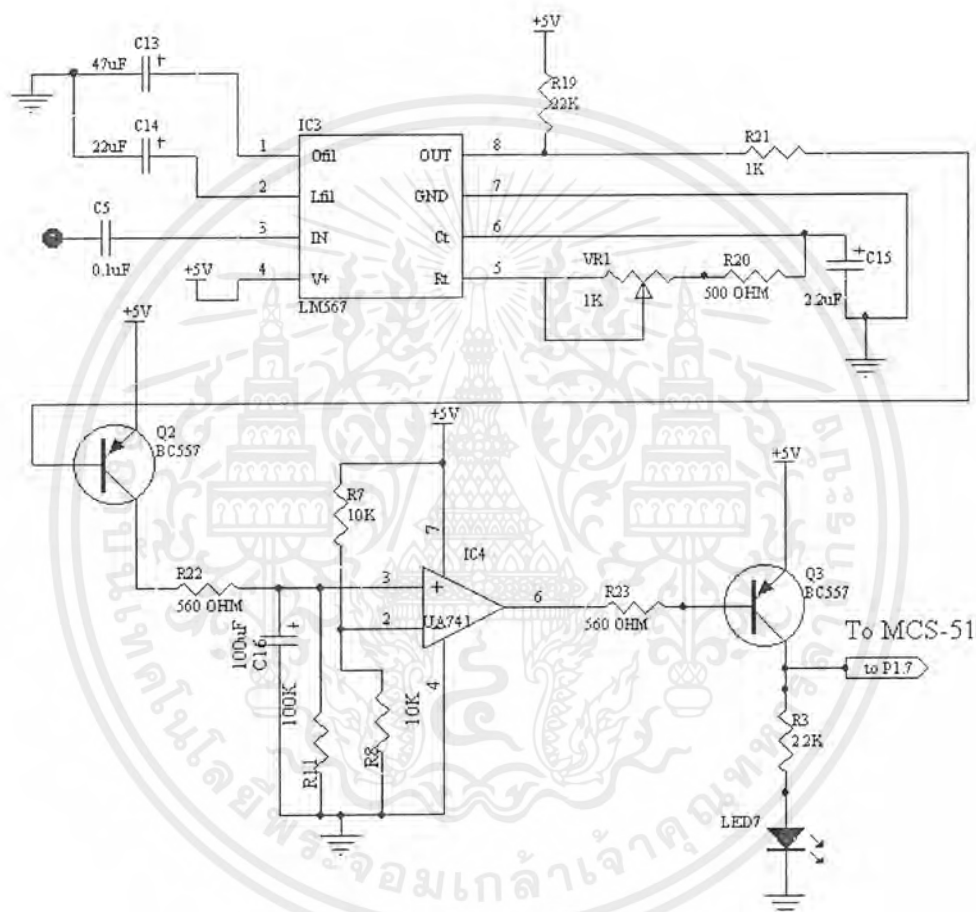
รูปที่ 3.2 แสดงวงจรตรวจสอบการยกหูโทรศัพท์

3.1.2 วงจรตรวจสอบสัญญาณเรียกกลับ (Ringback Tone Detector Circuit)

วงจรตรวจสอบสัญญาณเรียกกลับแสดงดังรูปที่ 3.3 เมื่อมีการยกหูโทรศัพท์แล้วรีเลย์จะทำการต่อคู่สายโทรศัพท์เข้ากับวงจรนี้ และเมื่อกดเลขหมายปลายทางแล้วก็จะได้รับสัญญาณตอบกลับมา ถ้าหากสายปลายทางว่างก็จะได้รับสัญญาณเรียกกลับ (Ring back tone) ความถี่ประมาณ 425 Hz ดัง 2 วินาทีดับ 4 วินาที แต่ถ้าสายปลายทางไม่ว่างก็จะได้รับสัญญาณสายไม่ว่าง (Busy tone) ความถี่ประมาณ 425 Hz ดัง 0.5 วินาทีดับ 0.5 วินาที สัญญาณจากคู่สายโทรศัพท์จะผ่านเข้ามา IC LM567 ซึ่งถูกต่อเป็นวงจรตรวจจับสัญญาณ (Tone Detector) เพื่อตรวจจับสัญญาณความถี่ประมาณ 425 Hz โดยการกำหนดค่าจากตัวต้านทานและตัวเก็บประจุซึ่งต่ออยู่ที่ขา 5 และ ขา 6 ของ IC LM567 จากความสัมพันธ์ $f_0 = 1.1/RC$ เมื่อมีสัญญาณอินพุตซึ่งมีความถี่ประมาณ 425 Hz มาเข้าวงจรนี้ เอาต์พุตที่ขา 8 จะเป็น “0” และเมื่อไม่มีอินพุตความถี่ดังกล่าว เอาต์พุตที่ขา 8 จะเป็น “1” เมื่ออินพุตเป็นสัญญาณสายไม่ว่าง เอาต์พุตที่ขา 8 จะเป็น “0” นาน 0.5 วินาที เป็น “1” นาน 0.5 วินาที สลับกันไป ถ้าอินพุตเป็นสัญญาณเรียกกลับ เอาต์พุตที่ขา 8 จะเป็น “0” นาน 2 วินาที เป็น “1” นาน 4 วินาทีสลับกันไป เมื่อสัญญาณนี้ผ่านทรานซิสเตอร์ BC557

ซึ่งทำหน้าที่เป็นอินเวอร์เตอร์ (Inverter) สัญญาณจะกลับสถานะ กล่าวคือ สัญญาณสายไม่ว่างก็จะเป็น “0” นาน 0.5 วินาที ตามเดิม แต่ถ้าเป็นสัญญาณเรียกกลับจะเป็น “0” นาน 4 วินาที “1” นาน 2 วินาที สัญญาณจะถูกลดแรงดันลงเมื่อผ่านตัวต้านทาน 560Ω และทำให้ตัวเก็บประจุ $100 \mu F$ เกิดการประจุและคายประจุ ช่วงเวลาการติดดับของสัญญาณไม่เพียงพอที่จะทำให้ตัวเก็บประจุสามารถคายประจุ

ได้ทัน แรงดันที่ขา 3 ของ IC LM741 จึงสูงกว่าแรงดันที่ขา 2 ทำให้เอาต์พุตที่ขา 6 เป็น “1” และทำให้เอาต์พุตของวงจรที่ขา 4 ของ IC 4N25 เป็น “1” แต่เมื่อปลายทางรับสายจะทำให้ตัวเก็บประจุไม่เกิดการประจุ แรงดันที่ขา 3 จึงต่ำกว่าแรงดันที่ขา 2 ทำให้เอาต์พุตที่ขา 6 และเอาต์พุตของวงจรเป็น “0” อย่างต่อเนื่อง แต่เนื่องจากเอาต์พุตของวงจรในทางปฏิบัติไม่เป็น “0” จริงตามทฤษฎีจึงต้องนำทรานซิสเตอร์ BC 557 มาทำการปรับระดับแรงดันให้ใกล้เคียงกับทฤษฎี แต่เอาต์พุตจะกลับสถานะเป็นตรงกันข้ามกับขา 6 ทำให้ส่วนควบคุมการทำงานทราบว่าปลายทางรับสายแล้ว ให้ทำการเริ่มจับเวลาการใช้งานได้

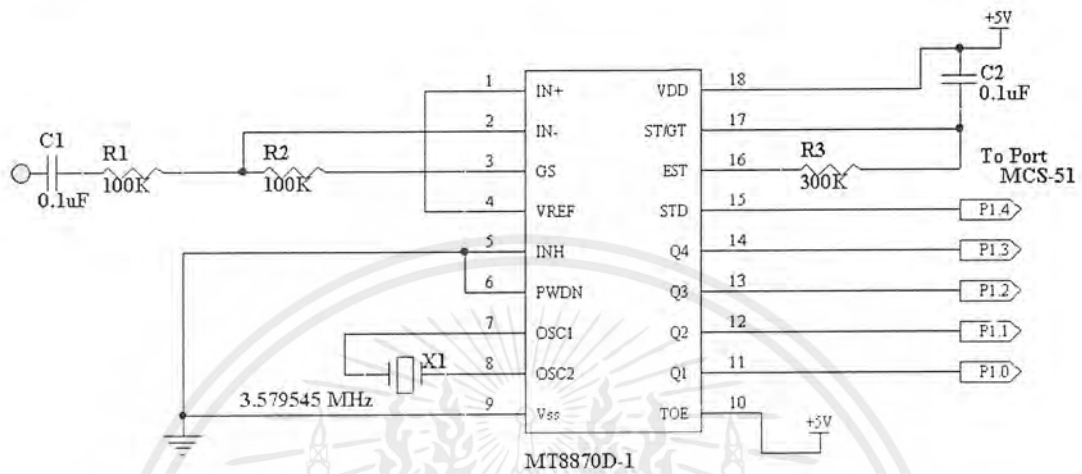


รูปที่ 3.3 แสดงวงจรตรวจสอบสัญญาณเรียกกลับ

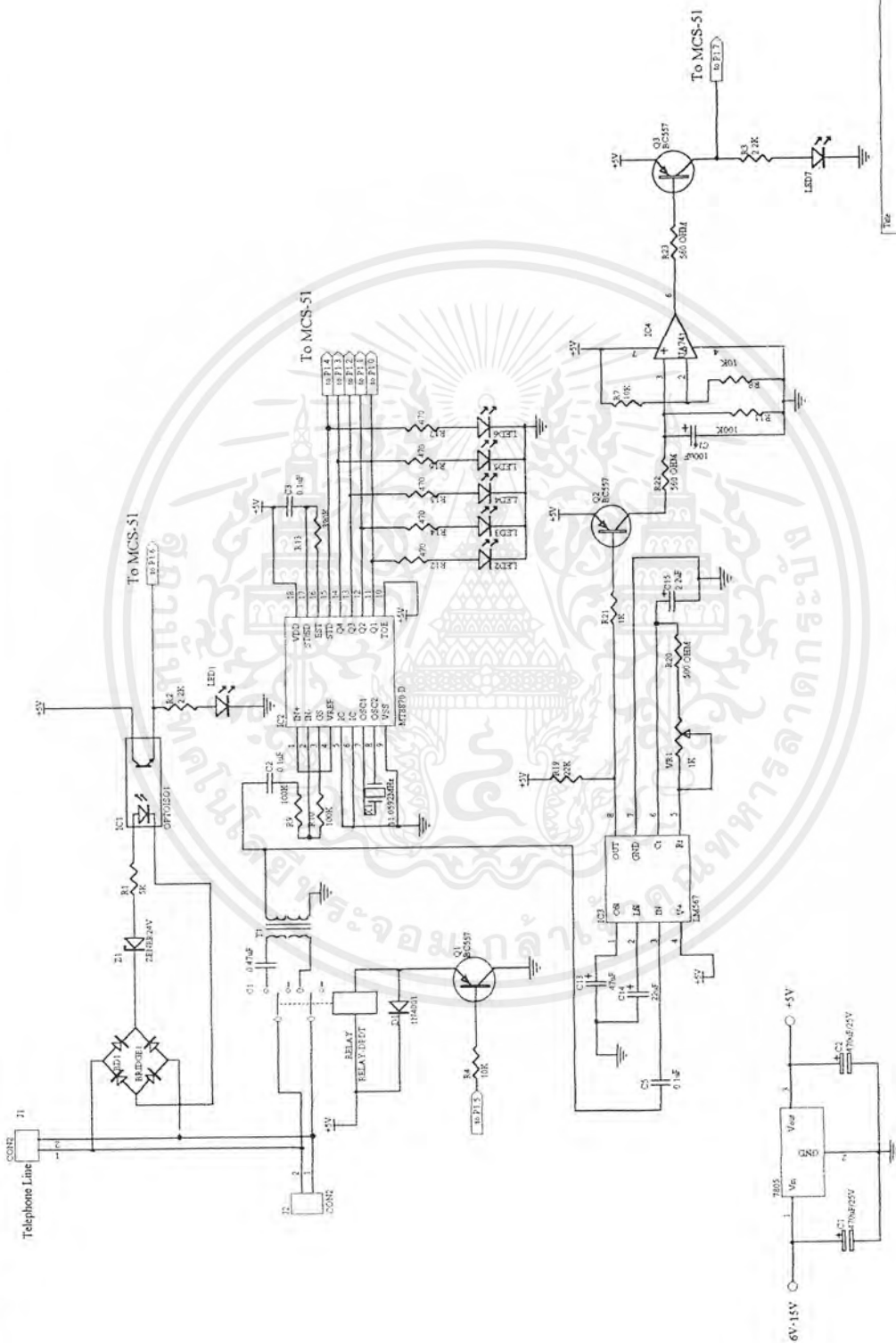
3.1.3 วงจรถอดรหัสหมายเลขโทรศัพท์ (DTMF Decoder Circuit)

จะใช้ไอซีสำเร็จรูปเบอร์ MT8870 ซึ่งจะทำหน้าที่ในการถอดรหัสความถี่ที่เอ็มเอฟออกมาเป็นรหัสไบนารี 4 บิต ภายใน MT8870 ซึ่งจะประกอบไปด้วยภาคกรองความถี่ ภาคถอดรหัส ภาคตรวจสอบสัญญาณ ภาคขยายสัญญาณความแตกต่าง และภาคกำเนิดความถี่ โดยจะใช้คริสตอล 3.579545 เมกกะเฮิร์ตซ์เป็นฐานเวลา จากวงจรจะใช้แมชซิง 600 โอห์มมาเป็นตัวคัปปลิงสัญญาณจากคู่สายโทรศัพท์มายังวงจรเฉพาะสัญญาณไฟกระส่ายเท่านั้น และยังทำหน้าที่ในการแยกการวัดของวงจรอีกด้วย เพราะบนคู่สายโทรศัพท์จะมีสัญญาณรบกวนต่างๆมากมาย ซึ่งอาจจะมีผลต่อการดีโค๊ดสัญญาณได้ สัญญาณจากการกดหมายเลข 1 ตัว จะถูกถอดรหัสออกเป็นเลขไบนารี

4 บิต ทางขา 11,12,13 และ 14 (Q1 – Q4) ส่วนสัญญาณที่ขา 15 จะเป็น “1” ขณะกดปุ่มโทรศัพท์ เมื่อไม่กดจะเป็น “0” สัญญาณนี้จะบอกให้ส่วนควบคุมทราบว่ามีการกดปุ่มโทรศัพท์ จากนั้นไอซีจะทำการแลตซ์ค่าสัญญาณนั้นเอาไว้จนกว่าจะได้รับสัญญาณใหม่เข้ามา แล้วจึงค่อยเปลี่ยนเป็นรหัสตัวใหม่

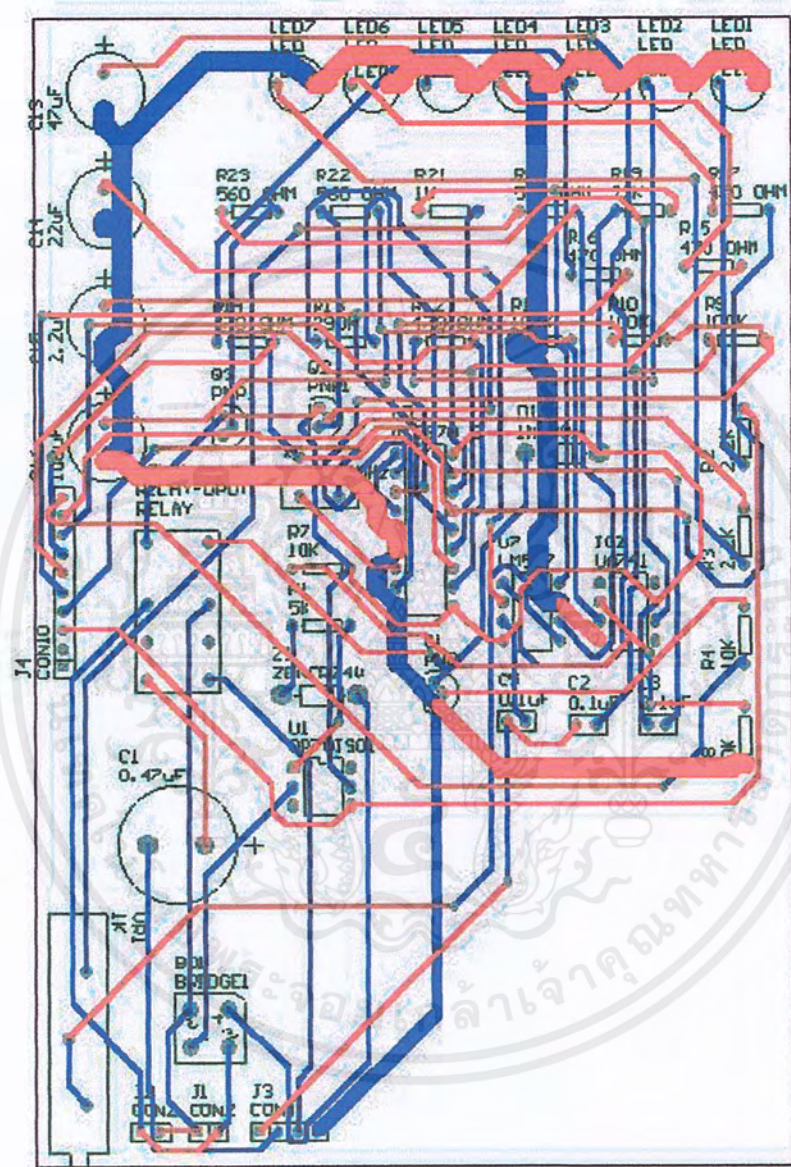


รูปที่ 3.4 วงจรถอดรหัสหมายเลขโทรศัพท์



รูปที่ 3.5 วงจรดีเทคเตอร์

Title		Date		Page	
Author	Number	16 Mar 2001	1	1	1
Revised	Number				
Drawn	Number				
File	Number				

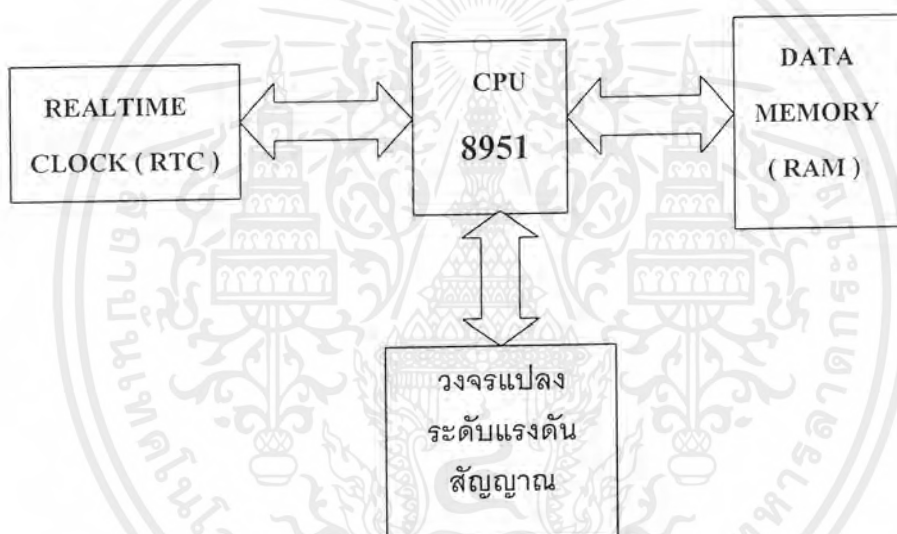


รูปที่ 3.6 ถายวงจรของวงจรดีเทคเตอร์

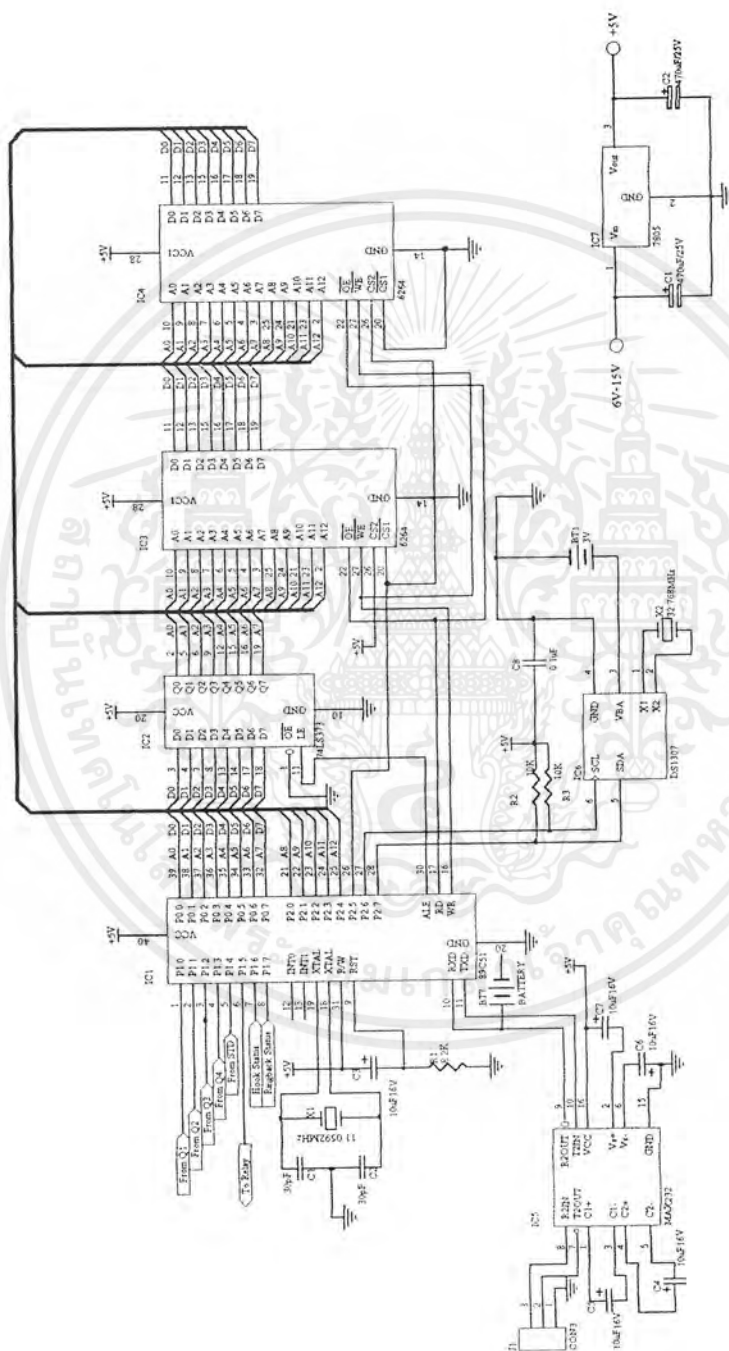
3.2 ส่วนควบคุมการทำงาน

ในโครงงานนี้จะใช้ไมโครคอนโทรลเลอร์เบอร์ 89C51 ทั้งนี้เพื่อที่สะดวกในการทดลองและใช้งานจริงได้ ซึ่งส่วนประกอบของส่วนควบคุมการทำงานนี้จะมีส่วนรายละเอียดของบล็อกไดวงจรและวงจรดังรูปที่ 3.7 และรูปที่ 3.8 ตามลำดับ วงจรดีเทคเตอร์จะทำการดีเทคสัญญาณในสถานะต่างๆบนคู่สายโทรศัพท์เข้ามาในส่วนควบคุมการทำงาน โดยไมโครคอนโทรลเลอร์ 89C51 จะเป็นตัวควบคุมการทำงานทั้งหมด ซึ่งข้อมูลต่างๆที่ส่งเข้ามาจะถูกเก็บเอาไว้ใน RAM ส่วนการถ่ายโอนข้อมูลระหว่าง RAM กับ CPU จะใช้ IC 74LS373 ในการแลตช์แอดเดรสเอาไว้ และส่วนควบคุมนี้จะมีวงจร RTC ทำหน้าที่ในการสร้างฐานเวลาให้กับ 89C51 ซึ่งจะส่งค่าเวลามาตรฐานเข้าไปเก็บไว้ใน RAM เช่นกัน

ส่วน IC MAX 232 จะเป็น IC ที่ใช้แปลงระดับแรงดันของสัญญาณให้ระดับแรงดันตามมาตรฐาน RS 232



รูปที่ 3.7 แสดงบล็อกไดอะแกรมของส่วนควบคุมการทำงาน



รูปที่ 3.8 แสดงวงจรส่วนควบคุมและหน่วยความจำ

[illegible]

3.2.1 ส่วนควบคุมการรับข้อมูลจากวงจรดีเทคเตอร์

สัญญาณจากคู่สายโทรศัพท์จะผ่านวงจรดีเทคเตอร์เพื่อแปลงให้เป็นสัญญาณดิจิทัล แล้วป้อนให้กับพอร์ทของ 8951 ซึ่งสัญญาณจากวงจรดีเทคเตอร์มีดังนี้

1. สัญญาณตรวจสอบการยกหูโทรศัพท์ เป็นสัญญาณข้อมูล 1 บิต ขนาดประมาณ 5 โวลต์ ซึ่งแทนด้วยระดับลอจิก “1” เมื่อมีการวางหู และมีขนาดประมาณ 0 โวลต์ ซึ่งแทนด้วยระดับลอจิก “0” เมื่อมีการยกหู สัญญาณดังกล่าวจะส่งเข้าพอร์ท 1 (ขา 7) ของ 8951

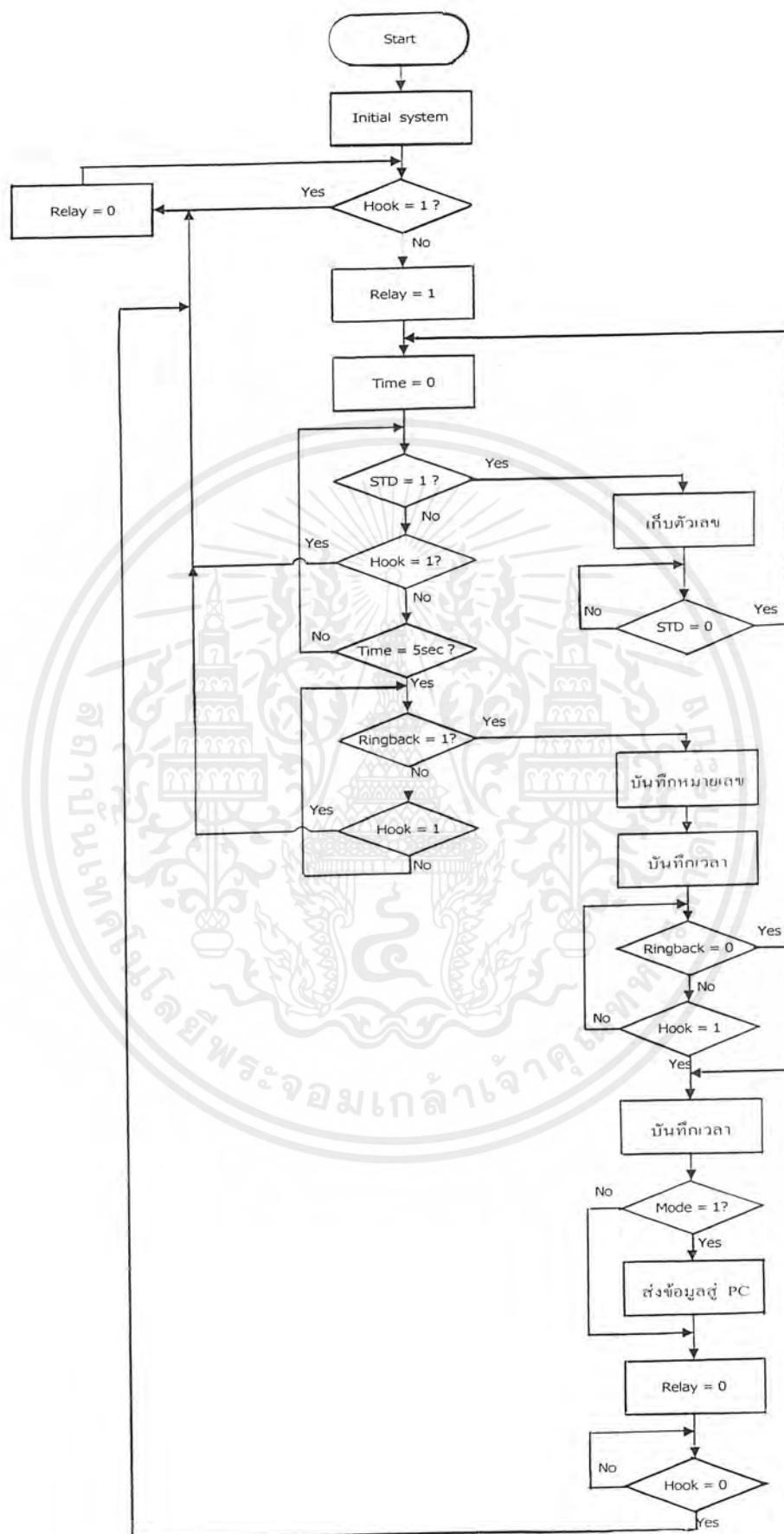
2. สัญญาณตรวจสอบการกดปุ่มเลขหมายโทรศัพท์ เป็นสัญญาณข้อมูล 1 บิตได้จากขา 15 ของ IC MT8870 ของวงจรถอดรหัสหมายเลขโทรศัพท์ ซึ่งแทนด้วยระดับลอจิก “1” ขณะกดปุ่มหมายเลขโทรศัพท์ และแทนด้วยระดับลอจิก “0” เมื่อไม่มีการกดปุ่มหมายเลขโทรศัพท์ สัญญาณดังกล่าวจะส่งเข้าพอร์ท 1 (ขา 5) ของ 8951

3. สัญญาณเลขหมายโทรศัพท์ เมื่อมีการกดเลขหมายโทรศัพท์ 1 ตัว IC MT8870 ของวงจรถอดรหัสเลขหมายโทรศัพท์จะทำการถอดรหัสออกเป็นเลขไบนารี 4 บิต สัญญาณจากขา 14 จะเป็นบิตที่มีนัยสำคัญสูงสุด แล้วลดลงมาจนถึงบิตที่มีนัยสำคัญต่ำสุดที่ขา 11 สัญญาณดังกล่าวจะส่งเข้าพอร์ท 1 (ขา 1-4) ของ 8951

4. สัญญาณการตรวจสอบการรับสายของปลายทาง สัญญาณนี้ได้จากวงจรตรวจสอบสัญญาณเรียกกลับ เป็นสัญญาณข้อมูล 1 บิต จะแทนด้วยระดับลอจิก “0” เมื่อปลายทางยังไม่รับสาย และจะแทนด้วยระดับลอจิก “1” เมื่อปลายทางรับสายแล้ว สัญญาณนี้จะถูกส่งเข้าพอร์ท 1 (ขา 8) ของ 8951

3.2.2 การทำงานของโปรแกรมควบคุมการรับข้อมูลจากวงจรดีเทคเตอร์

เมื่อเริ่มทำงานก็จะมีกร Initial ระบบต่างๆ ไม่ว่าจะเป็น RTC, การส่งสัญญาณผ่านพอร์ทอนุกรม, พื้นที่เก็บข้อมูลและตัวแปรต่างๆ จากนั้นโปรแกรมก็จะตรวจสอบว่ามีการยกหูโทรศัพท์หรือไม่ โดยจะตรวจสอบสัญญาณ Hook นั้นเอง ถ้าพบว่ามีกรยกหูโทรศัพท์ก็จะทำให้ Relay ทำงาน เพื่อต่อคู่สายโทรศัพท์เข้ากับวงจรที่เหลือ เมื่อมีการกดปุ่มหมายเลข สัญญาณ STD จะปรากฏขึ้นจึงสามารถอ่านตัวเลขเข้าสู่ระบบได้ โดยอ่านตัวเลขเข้ามาเรื่อยๆ ครั้งละ 1 ตัว โดยจะตรวจสอบสัญญาณ STD อยู่ตลอดเวลา แต่ถ้ามีการวางหูโทรศัพท์ในช่วงนี้ก็จะทำการยกเลิกการทำงาน แล้วกลับไปตรวจสอบสัญญาณ Hook ต่อไป แต่ถ้าไม่มีการกดตัวเลขแล้วเป็นเวลาประมาณ 5 วินาที จะมาทำการตรวจสอบว่าสัญญาณ Ringback เท่ากับ 1 หรือไม่ ถ้าใช่แสดงว่าผู้รับได้รับสายแล้ว เราก็จะทำการบันทึกเวลาเริ่มโทรออกเอาไว้ หลังจากนั้นก็จะทำการตรวจสอบสัญญาณ Ringback ไปเรื่อยๆ ถ้ามีค่าเป็น 0 เมื่อใดแสดงว่าอีกฝ่ายได้วางหูแล้ว ก็จะทำการบันทึกเวลาสิ้นสุดการใช้งานโทรศัพท์ แล้วก็จะกลับไปตรวจสอบสัญญาณ Hook ในขบวนการเริ่มต้นอีกครั้ง



รูปที่ 3.9 แผนผังแสดงการทำงานการรับข้อมูลจากวงจรดีเทคเตอร์

3.2.3 รายละเอียดของโปรแกรมควบคุมการรับข้อมูลจากวงจรดีเทคเตอร์

```

R_SET_T    EQU    'S'        ; SERIAL SIGNAL BYTE
R_READ_T   EQU    'R'
R_READ_D   EQU    'G'
R_DEL_D    EQU    'D'
R_SET_M    EQU    'M'
R_READ_M   EQU    'T'
S_END      EQU    'E'
S_WAIT     EQU    'W'
SETMODE    EQU    'Z'
RSTMODE    EQU    'C'
ACCEPT     EQU    'A'
SEPA1      EQU    ','
SEPA2      EQU    ';'
FIRSTPTR   EQU    0C010H    ; FIRST ADDRESS OF DATA (FIX VALUE)
DS1307R    EQU    0D1H      ; ADDRESS OF RTC WHEN READ
DS1307W    EQU    0D0H      ; ADDRESS OF RTC WHEN WRITE
L_FRAME    EQU    10H       ; LEANGE OF DATA FRAME (DATA+CHECKSUM)
FLAGS      EQU    20H       ; FLAGS
LASTPTRH   EQU    21H       ; LAST ADDRESS OF DATA (21H-22H)
LASTPTL    EQU    22H
SECS       EQU    24H       ; SECONDS STORAGE RAM
MINS       EQU    25H       ; MINUTES
HRS        EQU    26H       ; HOURS
DAY        EQU    27H       ; DAY
DATE       EQU    28H       ; DATE
MONTH      EQU    29H       ; MONTH
YEAR       EQU    2AH       ; YEAR
CONTROL    EQU    2BH       ; FOR STORAGE OF CONTROL REGISTER WHEN READ
BUFF       EQU    2DH       ; SERIAL SEND AND RECIVE BUFFER (2DH - 3DH)
LASTTIME   EQU    3EH       ; LAST TIME READ (3EH - 45H)
STACK      EQU    46H

MODE       BIT    FLAGS.0    ; 1 - ONLINE MODE, 0 - OFFLINE MODE
LASTREAD   BIT    FLAGS.1    ; RTC CONNECT
ACK        BIT    FLAGS.2    ; RTC ACK
BUS_FLT    BIT    FLAGS.3    ; RTC BUS ON
2W_BUSY    BIT    FLAGS.4
ENUM       BIT    FLAGS.5    ; NUMBER CHECK

STD        BIT    P1.4
RELAY      BIT    P1.5
HOOK       BIT    P1.6
RINGBACK   BIT    P1.7

SCL        BIT    P2.6
SDA        BIT    P2.7

ORG        0000H
AJMP       INITIAL

ORG        0023H
AJMP       RXISR        ; SERIAL INTERFACE INTERRUPT VECTOR

INITIAL:   MOV     SP,#STACK
           MOV     P1,#0FFH
           MOV     PCON,#00H        ; SET PCON
           MOV     SCON,#50H        ; SET SCON

```

```

MOV      IE,#90H      ; ENABLE SERIAL INTERRUPT
MOV      TMOD,#20H     ; SET TIMER MODE
MOV      TH1,#0FDH    ; SET TIMER FOR 9600 BORD RATE
SETB     TR1           ; ENABLE TIMER 1

ACALL    INIT_RTC      ; INITIAL RTC
ACALL    SET_RTC       ; SET RTC TO 0
ACALL    CLR_LAST      ; CLEAR LAST TIME
MOV      DPTR,#FIRSTPTR ; INITIAL LASTPTR
MOV      LASTPTRH,DPH
MOV      LASTPTRL,DPL

START:   CLR          RELAY      ; OFF RELAY
         JB           HOOK,START  ; HOOK CHECK
         SETB        RELAY
         MOV         DPH,LASTPTRH
         MOV         DPL,LASTPTRL
         CLR         ENUM

MDELAY:  MOV          R0,#1FH
MDT1:    MOV          R1,#0FFH
MDT2:    MOV          R2,#0FFH
STDC:    JB           STD,GETNUM
         JB           HOOK,START
         DJNZ        R2,STDC
         DJNZ        R1,MDT2
         DJNZ        R0,MDT1

WAITRING: JB          RINGBACK,SAVENO
          JB          HOOK,START
          AJMP        WAITRING

GETNUM:  MOV          A,P1
          ANL         A,#0FH
          JB          ENUM,GET1
          RL          A
          RL          A
          RL          A
          RL          A
          ORL         A,#0FH
          MOVX        @DPTR,A
          SETB        ENUM
          AJMP        GET2

GET1:    MOV          R0,A
          MOVX        A,@DPTR
          ANL         A,#0F0H
          ORL         A,R0
          MOVX        @DPTR,A
          INC         DPTR
          CLR         ENUM

GET2:    JNB          STD,MDELAY
          AJMP        GET2

SAVENO:  JNB          ENUM,SAVE1
          INC         DPTR

SAVE1:   MOV          A,#SEPA1
          MOVX        @DPTR,A
          INC         DPTR
          ACALL       READ_RTC
          MOV         R0,#YEAR
          MOV         R1,#44H
          MOV         R3,#07H

SAVE2:   MOV          A,@R1
          MOV         R4,A
          MOV         A,@R0
          CJNE        A,04H,SAVE3

```

```

DEC      R0
DEC      R1
DJNZ     R3,SAVE2
AJMP     SAVE4
SAVE3:   MOV      A,@R0
MOVX     @DPTR,A
INC      DPTR
DEC      R0
DJNZ     R3,SAVE3
SAVE4:   MOV      R0,#YEAR
MOV      R1,#44H
MOV      R3,#07H
SAVE41:  MOV      A,@R0
MOV      @R1,A
DEC      R0
DEC      R1
DJNZ     R3,SAVE41
SAVE5:   JNB      RINGBACK,SAVE6
JNB      HOOK,SAVE5
SAVE6:   MOV      A,#SEPA1
MOVX     @DPTR,A
INC      DPTR
ACALL    READ_RTC
MOV      R0,#YEAR
MOV      R1,#44H
MOV      R3,#07H
SAVE7:   MOV      A,@R1
MOV      R4,A
MOV      A,@R0
CJNE     A,04H,SAVE8
DEC      R0
DEC      R1
DJNZ     R3,SAVE7
AJMP     SAVE9
SAVE8:   MOV      A,@R0
MOVX     @DPTR,A
INC      DPTR
DEC      R0
DJNZ     R3,SAVE8
SAVE9:   MOV      R0,#YEAR
MOV      R1,#44H
MOV      R3,#07H
SAVE10:  MOV      A,@R0
MOV      @R1,A
DEC      R0
DEC      R1
DJNZ     R3,SAVE10
MOV      A,#SEPA2
MOVX     @DPTR,A
INC      DPTR
ACALL    BLANK
PUSH     DPL
PUSH     DPH
MOV      DPL, LASTPTRL
MOV      DPH, LASTPTRH
POP      LASTPTRH
POP      LASTPTRL
JNB      MODE,END_STA
PUSH     DPH
PUSH     DPL
PUSH     0E0H

```

```

END_STA:      ACALL    SEND_DATA2
              JNB      HOOK,END_STA
              AJMP     START

RXISR:        CLR      RI      ; SERIAL INTERRUPT SERVICE
              MOV      A,SBUF
              CJNE     A,#R_SET_T,MA1
              ACALL    SET_TIME
              AJMP     MA_END
MA1:          CJNE     A,#R_READ_T,MA2
              ACALL    SEND_TIME
              AJMP     MA_END
MA2:          CJNE     A,#R_READ_D,MA3
              ACALL    SEND_DATA
              AJMP     MA_END
MA3:          CJNE     A,#R_DEL_D,MA4
              ACALL    DEL_DATA
              AJMP     MA_END
MA4:          CJNE     A,#R_SET_M,MA5
              ACALL    SET_MODE
              AJMP     MA_END
MA5:          CJNE     A,#R_READ_M,MA_END
              ACALL    READ_MODE
MA_END:       RETI

SET_TIME:     ACALL    REC_FRAME      ; SET TIME OF RTC
              ACALL    RAM2TI
              ACALL    SET_RTC
              RET

SEND_TIME:    ACALL    READ_RTC      ; SEND TIME TO PC
              ACALL    TI2RAM
              ACALL    SENDFRAME
              RET

SEND_DATA:    PUSH     DPH      ; SEND DATA TO PC
              PUSH     DPL
              PUSH     0E0H
              MOV      DPTR,#FIRSTPTR
SEND_DATA2:   MOV      A,DPH
              CJNE     A,LASTPTRH,SE1
              MOV      A,DPL
              CJNE     A,LASTPTL,SE1
              AJMP     SE_END
SE1:          MOV      A,#S_WAIT
              MOV      SBUF,A
SEWAIT:       JNB      TI,SEWAIT
              CLR      TI
SE11:         JNB      RI,SE11
              CLR      RI
              MOV      A,SBUF
              CJNE     A,#ACCEPT,SE_END
              ACALL    MK_FRAME
              ACALL    SENDFRAME
SE2:          JNB      RI,SE3
SE3:          CLR      RI
              MOV      A,SBUF
              CJNE     A,#ACCEPT,SE1
              CLR      C
              MOV      A,DPH
              SUBB     A,LASTPTRH

```

```

JC          SE1
CLR         C
MOV        A,DPL
SUBB       A, LASTPTRL
JC         SE1
SE_END:    MOV        A, #S_END
MOV        SBUF, A
SEWAIT1:   JNB        TI, SEWAIT1
CLR        TI
POP        0E0H
POP        DPL
POP        DPH
RET

DEL_DATA:  PUSH        0E0H          ; DELETE DATA
PUSH        DPL
PUSH        DPH
MOV        DPTR, #FIRSTPTR
PUSH        DPL
PUSH        DPH
POP        LASTPTRH
POP        LASTPTRL
ACALL      CLR_LAST
MOV        A, #S_END
MOV        SBUF, A
DEL1:      JNB        TI, DEL1
CLR        TI
POP        DPH
POP        DPL
POP        0E0H
RET

SET_MODE:  PUSH        0E0H          ; SET MODE OF COMMUNICATION
SETM1:     JNB        RI, SETM1
CLR        RI
MOV        A, SBUF
CJNE      A, #SETMODE, SETM2
SETB       MODE
AJMP      SETM3
SETM2:     CLR        MODE
SETM3:     POP        0E0H
RET

READ_MODE: PUSH        0E0H          ; SEND MODE TO PC
JNB        MODE, READM1
MOV        A, #SETMODE
AJMP      READM2
READM1:    MOV        A, #RSTMODE
READM2:    MOV        SBUF, A
READM3:    JNB        TI, READM3
CLR        TI
POP        0E0H
RET

CLR_LAST:  PUSH        00H          ; CLEAR LAST TIME
PUSH        01H
MOV        R0, #LASTTIME
MOV        R1, #07H
CLR1:      MOV        @R0, #00H
INC        R0
DJNZ      R1, CLR1

```

```

        POP        01H
        POP        00H
        RET

RAM2TI:  PUSH        00H    ; CONVERT DATA TO TIME BUFFER
        PUSH        01H
        PUSH        02H
        PUSH        0E0H
        MOV         R2,#07H
        MOV         R1,#SECS
        MOV         R0,#BUFF
RA1:     MOV         A,@R0
        MOV         @R1,A
        INC         R1
        INC         R0
        DJNZ        R2,RA1
        POP         0E0H
        POP         02H
        POP         01H
        POP         00H
        RET

TI2RAM:  PUSH        00H    ; CONVERT TIME BUFFER TO DATA
        PUSH        01H
        PUSH        02H
        PUSH        0E0H
        MOV         R2,#07H
        MOV         R1,#SECS
        MOV         R0,#BUFF
TI1:     MOV         A,@R1
        MOV         @R0,A
        INC         R1
        INC         R0
        DJNZ        R2,TI1
        MOV         A,#00H
        MOV         @R0,A
        POP         0E0H
        POP         02H
        POP         01H
        POP         00H
        RET

MK_FRAME:  PUSH        00H    ; MAKE FRAME OF DATA
        PUSH        01H
        MOV         R1,#L_FRAME
        DEC         R1
        MOV         R0,#BUFF
MK1:     MOVX        A,@DPTR
        MOV         @R0,A
        INC         R0
        INC         DPTR
        DJNZ        R1,MK1
        POP         01H
        POP         00H
        RE

BLANK:    PUSH        DPH
        PUSH        DPL
        PUSH        00H
        PUSH        0E0H
        MOV         R0,#0FH

```

```

BLA:      MOV      A, #00H
          MOVX     @DPTR, A
          INC      DPTR
          DJNZ     R0, BLA
          POP      0E0H
          POP      00H
          POP      DPL
          POP      DPH
          RET

```

----- SERIAL CONNECT WITH RS232 -----

```

SENDFRAME:  PUSH  00H      ;SEND FRAME OF DATA FROM BUFFER TO PC
            PUSH  01H
            PUSH  0E0H
            ACALL STORE_CH
            MOV   R1, #L_FRAME
            MOV   R0, #BUFF
S1:         MOV   A, @R0
            MOV   SBUF, A
SWAIT:      JNB   TI, SWAIT
            CLR   TI
            INC   R0
            DJNZ  R1, S1
            POP   0E0H
            POP   01H
            POP   00H
            RET

REC_FRAME:  PUSH  00H
            PUSH  01H
            MOV   R0, #L_FRAME
            MOV   R1, #BUFF
REC1:       JNB   RI, REC1
            CLR   RI
            MOV   A, SBUF
            MOV   @R1, A
            INC   R1
            DJNZ  R0, REC1
            MOV   R0, A
            ACALL CH_GEN
            CLR   C
            SUBB  A, R0
            POP   01H
            POP   00H
            RET

CH_GEN:     PUSH  00H      ; CHECK SUM GENERATOR
            PUSH  01H
            PUSH  02H
            MOV   R0, #L_FRAME
            MOV   R1, #BUFF
            DEC   R0
            MOV   R2, #00H
CH1:        MOV   A, @R1
            CLR   C
            ADDC  A, R2
            MOV   R2, A
            INC   R1

```

```

        DJNZ      R0,CH1
        MOV       A,R2
        POP       02H
        POP       01H
        POP       00H
        RET

STORE_CH:  MOV     R1,#BUFF ; STORE CHECK SUM TO BUFFER
           MOV     RC,#L_FRAME
           DEC     R0
ST1:       INC     R1
           DJNZ    R0,ST1
           ACALL   CH_GEN
           MOV     @R1,A
           RET

;----- RTC WORK SPACE -----
;-----

INIT_RTC:  ACALL   RTCS_START ; GENERATE START CONDITION
           MOV     A,#DS1307W ; 1101 0000 ADDRESS + WRITE-BIT
           ACALL   RTCS_BYTE  ; SEND BYTE TO 1307
           MOV     A,#00H     ; ADDRESS BYTE TO REGISTER 00H
           ACALL   RTCS_BYTE  ; SECONDS REGISTER, ALWAYS LEAVE
           SETB    LASTREAD   ; REG 00H-BIT #7 = 0 (LOW)
           ACALL   RTCS_STOP  ; IF REG 00H-BIT #7 = 1 CLOCK
           ACALL   RTCS_START ; OSCILLATOR IS OFF.
           MOV     A,#DS1307R ; 1101 0001 ADDRESS + READ-BIT
           ACALL   RTCS_BYTE
           ACALL   RTCR_BYTE  ; READ A BYTE FROM THE 1307
           CLR     ACC.7      ; CLEAR REG 00H-BIT #7 TO
                               ; ENABLE OSCILLATOR.
OSC_SET:   PUSH    ACC        ; SAVE ON STACK
           ACALL   RTCS_STOP
           ACALL   RTCS_START
           MOV     A,#DS1307W ; SETUP TO WRITE
           ACALL   RTCS_BYTE
           MOV     A,#00H     ; REGISTER 00H ADDRESS
           ACALL   RTCS_BYTE
           POP     ACC        ; GET DATA TO START OSCILLATOR
           ACALL   RTCS_BYTE  ; SEND IT
           ACALL   RTCS_STOP
           RET

READ_RTC:  PUSH    00H
           PUSH    01H
           PUSH    02H
           MOV     R1,#24H    ; SECONDS STORAGE LOCATION
           MOV     R2,#00H
           CLR     LASTREAD
           ACALL   RTCS_START
           MOV     A,#DS1307W
           ACALL   RTCS_BYTE
           MOV     A,#00H
           ACALL   RTCS_BYTE
           ACALL   RTCS_STOP
           ACALL   RTCS_START
           MOV     A,#DS1307R
           ACALL   RTCS_BYTE
READ_LOOP: MOV     A,R2

```

```

NOT_LAST:    CJNE    A,#09H,NOT_LAST
              SETB    LASTREAD
              ACALL    RTCR_BYTE
              MOV      @R1,A
              MOV      A,R2
              CJNE    A,#00H,NOT_FIRST
              MOV      A,@R1
              CLR      ACC.7 ; ENSURE OSC BIT=0 (ENABLED)
              MOV      @R1,A
NOT_FIRST:   INC      R1
              INC      R2
              MOV      A,R2
              CJNE    A,#0AH,READ_LOOP
              ACALL    RTCS_STOP
              POP      02H
              POP      01H
              POP      00H
              RET

SET_RTC:     PUSH     01H
              ACALL    RTCS_START ; SEND 2-WIRE START
                                   ; CONDITION
              MOV      A,#DS1307W ; SEND 1307 WRITE COMMAND
              ACALL    RTCS_BYTE
              MOV      A,#00H ; SET DATA POINTER TO
                                   ; REGISTER 00H ON THE DS1307
              ACALL    RTCS_BYTE
              MOV      R1,#24H ; START WITH SECONDS
                                   ; REGISTER IN INTERNAL RAM
SEND_LOOP:   MOV      A,@R1 ; MOVE FIRST BYTE OF DATA TO ACC
              ACALL    RTCS_BYTE ; SEND DATA ON 2-WIRE BUT
              INC      R1 ; LOOP UNTIL CLOCK DATA SENT
              CJNE    R1,#2BH,SEND_LOOP
              ACALL    RTCS_STOP ; SEND 2-WIRE STOP
                                   ; CONDITION
              POP      01H
              RET

RTCS_START:  SETB     _2W_BUSY ; INDICATE THAT 2-WIRE
              CLR      ACK ; OPERATION IS IN PROGRESS
              CLR      BUS_FLT ; CLEAR STATUS FLAGS
              JNB      SCL,FAULT
              JNB      SDA,FAULT
              SETB     SDA ; BEGIN START CODITION
              SETB     SCL ; SET SCL HIGH
              JNB      SCL,$ ; LOOP UNTIL STRONG 1 ON SCL
              CLR      SDA
              ACALL    DELAY
              CLR      SCL
              RET
FAULT:       SETB     BUS_FLT
              RET

RTCR_BYTE:   PUSH     00H
              MOV      R0,#08H ;SET COUNTER FOR 8-BITS DATA
              MOV      A,#00H
              SETB     SDA ;SET SDA HIGH TO ENSURE LINE FREE
READ_BITS:   SETB     SCL ; SET SCL HIGH
              JNB      SCL,$ ; LOOP UNTIL STRONG 1 ON SCL
              MOV      C,SDA ; MOVE DATA BIT INTO CARRY
              RLC      A ; ROTATE CARRY-BIT INTO ACC.0

```

```

CLR          SCL          ; TRANSITION SCL HIGH-TO-LOW
DJNZ         R0,READ_BITS ; LOOP FOR 8-BITS
JB           LASTREAD,ACKN ; CHECK TO SEE
                        ; IF THIS IS THE LAST READ
ACKN:        CLR          SDA ; IF NOT LAST READ SEND ACK-BIT
SETB         SCL          ; SET SCL HIGH
JNB          SCL,$        ; LOOP UNTIL STRONG 1 ON SCL
CLR          SCL          ; OR NOT ACKNOWLEDGE BIT
POP          00H
RET

RTCS_STOP:   CLR          SDA
SETB         SCL          ; SET SCL HIGH
JNB          SCL,$        ; LOOP UNTIL STRONG 1 ON SCL
SETB         SDA
CLR          _2W_BUSY
RET

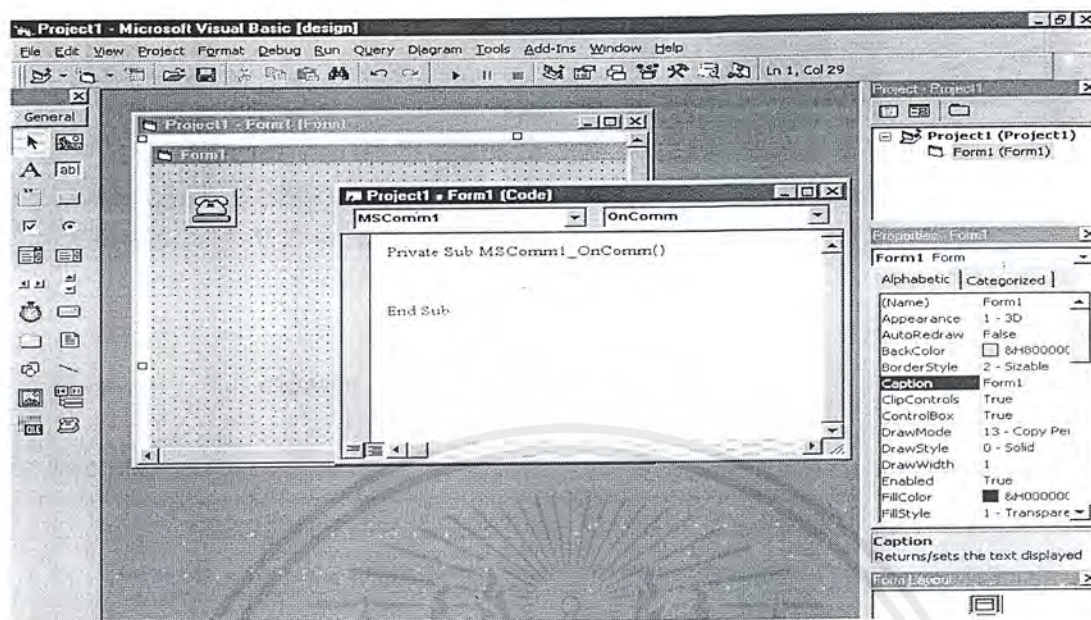
RTCS_BYTE:   PUSH        00H
MOV          R0,#08H      ; SET COUNTER FOR 8-BITS
SB_LOOP:    JNB          ACC.7,NOTONE ; CHECK TO SEE IF
                        ; BIT-7 OF
SETB         SDA          ; ACC IS A 1, AND SET SDA HIGH
JMP          ONE
NOTONE:      CLR          SDA ; CLR SDA LOW
ONE:         SETB         SCL ; SET SCL HIGH
JNB          SCL,$        ; LOOP UNTIL STRONG 1 ON SCL
RL           A            ; ROTATE ACC LEFT 1-BIT
CLR          SCL          ; TRANSITION SCL LOW-TO-HIGH
DJNZ         R0,SB_LOOP   ; LOOP FOR 8-BITS
SETB         SDA          ; SET SDA HIGH TO LOOK FOR
SETB         SCL          ; SET SCL HIGH
JNB          SCL,$        ; LOOP UNTIL STRONG 1 ON SCL
CLR          ACK
JNB          SDA,SB_EX     ; CHECK FOR ACK OR NOT ACK
SETB         ACK          ; SET ACKNOWLEDGE FLAG FOR
                        ; NOT ACK
SB_EX:       ACALL        DELAY ; DELAY FOR AN OPERATION
CLR          SCL          ; TRANSITION SCL HIGH-TO-LOW
ACALL        DELAY        ; DELAY FOR AN OPERATION
POP          00H
RET

DELAY:       NOP          ; DELAY FOR BUS TIMING
RET
END

```

3.3 ส่วนแสดงและประมวลผล

ในการติดต่อกับส่วนควบคุมนั้นเราจะใช้ Visual Basic ซึ่งเป็นภาษาที่ง่ายและได้รับความนิยมสูง อีกทั้งยังมี Component สำเร็จรูปต่างๆ ให้ใช้งานมากมาย ในการสื่อสารผ่านพอร์ทอนุกรมนี้เราใช้ Component ชื่อ Microsoft Comm Control ซึ่งเป็น Control ที่สร้างมาขึ้นเพื่อการสื่อสารกับพอร์ทอนุกรมโดยตรง ดังรูปที่ 3.10

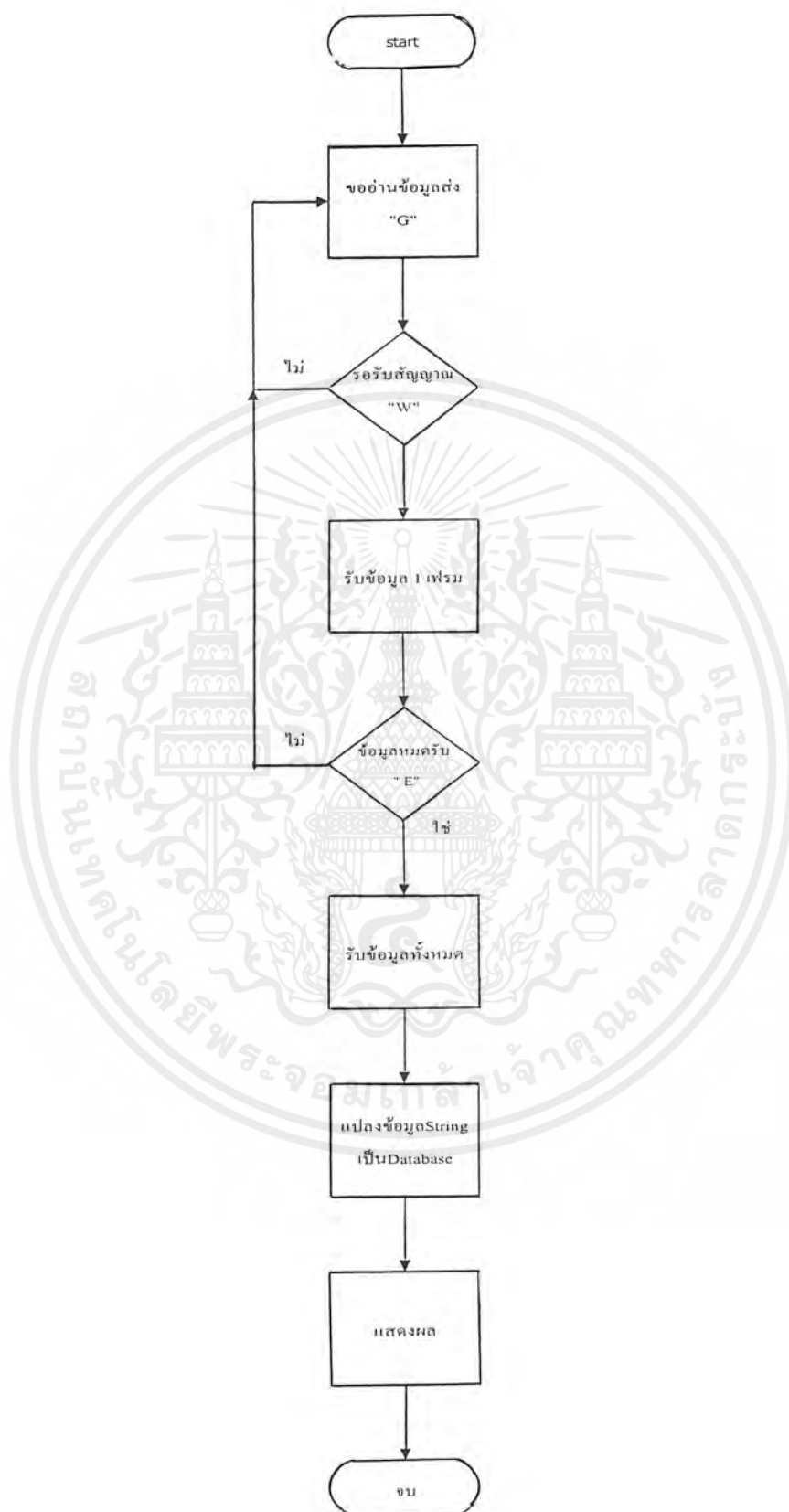


รูปที่ 3.10 แสดง Microsoft Comm Control

3.3.1 การทำงานของโปรแกรมแสดงและประมวลผล

เมื่อต้องการจะส่งข้อมูลจากส่วนควบคุม PC ก็จะส่ง Access “G” ออกไป เมื่อส่งไปแล้วก็จะรอสัญญาณตอบรับถ้าได้รับสัญญาณตอบรับเป็น “W” PC ก็จะเริ่มรอรับข้อมูลเข้ามาครั้งละ 1 เฟรม เมื่อได้รับข้อมูลแล้วก็จะส่ง Access “A” ออกไปเพื่อบอกว่าได้รับข้อมูลเรียบร้อยแล้ว จากนั้นก็จะรอรับข้อมูลต่อไป โดยถ้าได้รับ Access “W” ก็แสดงว่ามีข้อมูลอีก ก็จะรอรับข้อมูลอีก จนได้รับ Access “E” แสดงว่าข้อมูลหมดแล้ว จึงนำข้อมูลที่ได้นี้ไปแปลงเป็นข้อมูล String แล้วนำเข้าไปเก็บยัง Database ต่อไป และก็สามารถที่จะนำมาแสดงผลได้

Flowchart แสดงการรับข้อมูล จากไมโครคอนโทรลเลอร์



รูปที่ 3.11 แผนผังแสดงการทำงานการแสดงผลและประมวลผล

```
Private MainWs As Workspace
Private MainDb As Database
Private TelDyn As Recordset
```

```
Private Sub CmdClear_Click()
    MSComm1.InputLen = 0
    MSComm1.PortOpen = True
    Buf = ""
    MSComm1.Output = "D"
    Do
        DoEvents
        Buf = Buf + MSComm1.Input
    Loop Until Len(Buf) <> 0
    MSComm1.PortOpen = False
    If Buf = "E" Then
        k = MsgBox("ลบข้อมูลเรียบร้อยแล้ว", vbOKOnly, "Clear Data")
    Else
        k = MsgBox("การลบข้อมูลผิดพลาด", vbOKOnly, "Clear Data")
    End If
End Sub
```

```
Private Sub CmdDelete_Click()
    Dim RecStr As String
    On Error Resume Next

    If MsgBox("คุณยืนยันที่จะลบข้อมูลในฐานข้อมูลเก่าหรือไม่", vbCritical +
vbYesNo, "ลบข้อมูล") = vbYes Then
        Adodc2.Refresh
        Adodc2.Refresh
        RecCnt = Adodc2.Recordset.RecordCount
        If RecCnt > 0 Then
            Adodc2.Recordset.MoveFirst
            For i = 1 To RecCnt Step 1
                Adodc2.Recordset.Delete
                Adodc2.Recordset.MoveNext
            Next i
        End If
        Adodc2.Refresh
        Adodc2.Refresh
        Call ArrGrid2
    End If
End Sub
```

```
Private Sub CmdExit_Click()
    Unload Me
End Sub
```

```
Private Sub CmdNew_Click()
    FrNew.Visible = True
    FrOld.Visible = False
    CmdOld.Visible = True
    CmdNew.Visible = False
    Call ArrGrid
End Sub
```

```
Private Sub CmdOld_Click()
    FrNew.Visible = False
    FrOld.Visible = True
    CmdOld.Visible = False
```

```

CmdNew.Visible = True
Call ArrGrid2
End Sub

Private Sub CmdPrint_Click()
DataReport1.Show
Me.Hide
End Sub

Private Sub CmdRead_Click()
Dim P1 As Integer
Dim P2 As Integer
Dim TelNo As String
Dim Time1 As String
Dim Time2 As String
Dim RecStr As String
Dim D1, D2 As Date
'On Error Resume Next
Adodc1.Refresh
Adodc1.Refresh
RecCnt = Adodc1.Recordset.RecordCount
If RecCnt > 0 Then
    Adodc1.Recordset.MoveFirst
    For i = 1 To RecCnt Step 1
        Adodc1.Recordset.Delete
        Adodc1.Recordset.MoveNext
    Next i
End If
Adodc1.Refresh
Adodc1.Refresh
Call ArrGrid
'RecStr =
"7390150F,01031802112000,2135;1144,3520,210012;152F,2000,2501;"
RecStr = RecData
If Len(RecStr) = 0 Then
    k = MsgBox("ไม่พบข้อมูลที่บันทึกไว้", vbOKOnly, "Recive Data")
    Exit Sub
End If
P1 = InStr(1, RecStr, ",") + 1
TelNo = Mid(RecStr, 1, P1 - 2)
If Right(TelNo, 1) = "F" Then
    TelNo = Left(TelNo, Len(TelNo) - 1)
End If
P2 = InStr(P1, RecStr, ",") + 1
Time1 = Mid(RecStr, P1, P2 - P1 - 1)
D1 = CalTime(Now, Time1)
P1 = InStr(P2, RecStr, ";") + 1
Time2 = Mid(RecStr, P2, P1 - P2 - 1)
D2 = CalTime(D1, Time2)
Adodc1.Recordset.AddNew
Adodc1.Recordset.Fields(1) = TelNo
TelDyn.Requery
TelDyn.FindFirst ("TelNo='" + TelNo + "'")
If TelDyn.NoMatch Then
    Adodc1.Recordset.Fields(2) = "ไม่พบข้อมูล"
Else
    Adodc1.Recordset.Fields(2) = TelDyn(1)
    Adodc1.Recordset.Fields(3) = TelDyn(2)
End If
Adodc1.Recordset.Fields(4) = D1

```

```

D2 = DateAdd("s", 0, D2)
Adodc1.Recordset.Fields(5) = D2
Adodc1.Recordset.Fields(6) = CalMinute((DateDiff("s", D1, D2)))
Adodc1.Recordset.Update

Adodc2.Refresh
Adodc2.Recordset.AddNew
Adodc2.Recordset.Fields(1) = Adodc1.Recordset(1)
Adodc2.Recordset.Fields(2) = Adodc1.Recordset(2)
Adodc2.Recordset.Fields(3) = Adodc1.Recordset(3)
Adodc2.Recordset.Fields(4) = Adodc1.Recordset(4)
Adodc2.Recordset.Fields(5) = Adodc1.Recordset(5)
Adodc2.Recordset.Fields(6) = Adodc1.Recordset(6)
Adodc2.Recordset.Update

Do While (InStr(P1, RecStr, ";") > P1)
    P2 = InStr(P1, RecStr, ",") + 1
    TelNo = Mid(RecStr, P1, P2 - P1 - 1)
    If Right(TelNo, 1) = "F" Then
        TelNo = Left(TelNo, Len(TelNo) - 1)
    End If
    P1 = InStr(P2, RecStr, ",") + 1
    Time1 = Mid(RecStr, P2, P1 - P2 - 1)
    D1 = CalTime(D2, Time1)
    P2 = InStr(P1, RecStr, ";") + 1
    Time2 = Mid(RecStr, P1, P2 - P1 - 1)
    D2 = CalTime(D1, Time2)
    Adodc1.Recordset.AddNew
    Adodc1.Recordset.Fields(1) = TelNo
    TelDyn.Requery
    TelDyn.FindFirst ("TelNo='" + TelNo + "'")
    If TelDyn.NoMatch Then
        Adodc1.Recordset.Fields(2) = "Not Found"
    Else
        Adodc1.Recordset.Fields(2) = TelDyn(1)
        Adodc1.Recordset.Fields(3) = TelDyn(2)
    End If
    Adodc1.Recordset.Fields(4) = D1
    D2 = DateAdd("s", 0, D2)
    Adodc1.Recordset.Fields(5) = D2
    Adodc1.Recordset.Fields(6) = CalMinute((DateDiff("s", D1,
D2)))

    Adodc1.Recordset.Update

    Adodc2.Refresh
    Adodc2.Recordset.AddNew
    Adodc2.Recordset.Fields(1) = Adodc1.Recordset(1)
    Adodc2.Recordset.Fields(2) = Adodc1.Recordset(2)
    Adodc2.Recordset.Fields(3) = Adodc1.Recordset(3)
    Adodc2.Recordset.Fields(4) = Adodc1.Recordset(4)
    Adodc2.Recordset.Fields(5) = Adodc1.Recordset(5)
    Adodc2.Recordset.Fields(6) = Adodc1.Recordset(6)
    Adodc2.Recordset.Update
    t = P1
    P1 = P2
    P2 = t

Loop
End Sub

Private Sub CmdReadT_Click()
Dim Buf As String

```

```

Dim CD As Date
Dim Tp As String
LabTl.Caption = Format(Now, "dd mmmm yyyy hh:mm:ss")
MSComm1.InputLen = 0
MSComm1.PortOpen = True
Buf = ""
MSComm1.Output = "R"
Do
    DoEvents
    Buf = Buf + MSComm1.Input
Loop Until Len(Buf) > 14
MSComm1.PortOpen = False
Buf = Chr2Hex(Buf)
Tp = Mid(Buf, 9, 2) + "/" + Mid(Buf, 11, 2) + "/20" + Mid(Buf,
13, 2)
CD = Tp
LabTr.Caption = Format(CD, "dd mmmm yyyy") + " " + Mid(Buf, 5, 2)
+ ":" + Mid(Buf, 3, 2) + ":" + Mid(Buf, 1, 2)
End Sub

Private Sub CmdSetT_Click()
Dim Buf As String
Buf = Chr(Hex2Dec(CStr(Second(Now))))
Buf = Buf + Chr(Hex2Dec(CStr(Minute(Now))))
Buf = Buf + Chr(Hex2Dec(CStr(Hour(Now))))
Buf = Buf + Chr(CInt(Format(Now, "d")))
Buf = Buf + Chr(Hex2Dec(CStr(Day(Date))))
Buf = Buf + Chr(Hex2Dec(CStr(Month(Date))))
Buf = Buf + Chr(Hex2Dec(Format(Now, "yy")))
Buf = Buf + " "
MSComm1.InputLen = 0
MSComm1.PortOpen = True
MSComm1.Output = "S" + Buf
Do
    DoEvents
Loop Until MSComm1.OutBufferCount = 0
MSComm1.PortOpen = False
Call CmdReadT_Click
End Sub

Private Sub Form_Load()
Dim RecCnt As Integer
ChDir "c:\vb\"
Set MainWs = DBEngine.Workspaces(0)
Set MainDb = MainWs.OpenDatabase(App.Path + "\TelDb.mdb", False,
False)
Set TelDyn = MainDb.OpenRecordset("SELECT * FROM TelNumber ORDER
BY TelNo", dbOpenDynaset)

Adodc1.ConnectionString =
"Provider=Microsoft.Jet.OLEDB.3.51;Persist Security Info=False;Data
Source=" + App.Path + "\TelDb.mdb"
Adodc1.RecordSource = "SELECT * FROM Temp ORDER BY STime"
Adodc2.ConnectionString =
"Provider=Microsoft.Jet.OLEDB.3.51;Persist Security Info=False;Data
Source=" + App.Path + "\TelDb.mdb"
Adodc2.RecordSource = "SELECT * FROM Use ORDER BY STime"
Adodc1.Refresh
Adodc1.Refresh
RecCnt = Adodc1.Recordset.RecordCount
If RecCnt > 0 Then

```

```

        Adodcl.Recordset.MoveFirst
        For i = 1 To RecCnt Step 1
            Adodcl.Recordset.Delete
            Adodcl.Recordset.MoveNext
        Next i
    End If
    Adodcl.Refresh
    Adodcl.Refresh
    Call ArrGrid
    MSComm1.CommPort = 1
    MSComm1.Settings = "9600,N,8,1"
    Call CmdSetT_Click
End Sub

```

```

Private Function Hex2Dec(H As String) As Integer
Dim S As String
Dim Sum As Integer
Dim M As Integer
    For i = 1 To Len(H)
        S = Mid(H, i, 1)
        Select Case S
            Case "A"
                M = 10
            Case "B"
                M = 11
            Case "C"
                M = 12
            Case "D"
                M = 13
            Case "E"
                M = 14
            Case "F"
                M = 15
            Case Else
                M = CInt(S)
        End Select
        M = M * (16 ^ (Len(H) - i))
        Sum = Sum + M
    Next i
    Hex2Dec = Sum
End Function

```

```

Private Function Chr2Hex(Num As String) As String
Dim LStr As Integer
Dim StrOut As String
Dim Ch As String
Dim i As Integer
    LStr = Len(Num)
    For i = 1 To LStr Step 1
        Ch = Mid(Num, i, 1)
        Ch = Hex(Asc(Ch))
        If Len(Ch) = 1 Then
            Ch = "0" + Ch
        End If
        If Ch = "2C" Then
            Ch = ","
        End If
        If Ch = "3B" Then
            Ch = ";"
        End If
        If Mid(Ch, 2, 1) = "A" Then

```

```

        Ch = Mid(Ch, 1, 1) + "0"
    End If
    If Mid(Ch, 1, 1) = "A" Then
        Ch = "0" + Mid(Ch, 2, 1)
    End If
    If i Mod 16 = 0 Then
        Ch = ""
    End If
    StrOut = StrOut + Ch
Next i
Chr2Hex = StrOut
End Function

```

```

Private Function RecData() As String
    Dim RecStr As String
    Dim ConStr As String
    Dim RStr As String
    Dim AStr As String
    MSComm1.PortOpen = True
    MSComm1.InputLen = 0
    ConStr = ""
    MSComm1.Output = "G"
    Do
        Buf = ""
        Do
            DoEvents
            Buf = Buf + MSComm1.Input
        Loop Until Len(Buf) <> 0
        ConStr = Left(Buf, 1)
        If ConStr = "W" Then
            MSComm1.Output = "A"
            MSComm1.InputLen = 0
            Do
                DoEvents
                Loop Until MSComm1.OutBufferCount = 0
            Buf = ""
            Do
                DoEvents
                Buf = Buf + MSComm1.Input
            Loop Until Len(Buf) > 15
            RecStr = RecStr + Buf
            MSComm1.Output = "A"
            MSComm1.InputLen = 0
            Do
                DoEvents
                Loop Until MSComm1.OutBufferCount = 0
            End If
        Loop Until ConStr <> "W"
        MSComm1.PortOpen = False
        RStr = Chr2Hex(RecStr)
        RecData = RStr
        If ConStr <> "E" Then
            RecData = "Error"
        End If
    End Function

```

```

Private Sub ArrGrid()
    With DataGrid1
        .Columns(0).Visible = False
        .Columns(1).Caption = "เบอร์โทรศัพท์"
    End With

```

```

.Columns(1).Alignment = dbgLef
.Columns(1).Width = 1400
.Columns(2).Caption = "ชื่อ"
.Columns(2).Alignment = dbgLef
.Columns(2).Width = 1200
.Columns(3).Caption = "ที่อยู่"
.Columns(3).Alignment = dbgLef
.Columns(3).Width = 4100
.Columns(4).Caption = "เริ่มโทร"
.Columns(4).Alignment = dbgRight
.Columns(4).Width = 1900
.Columns(4).NumberFormat = "dd mmmmm yyyy hh:mm:ss "
.Columns(5).Caption = "วางหู"
.Columns(5).Alignment = dbgRight
.Columns(5).Width = 1900
.Columns(5).NumberFormat = "dd mmmmm yyyy hh:mm:ss "
.Columns(6).Caption = "เวลา"
.Columns(6).Alignment = dbgRight
.Columns(6).Width = 800
End With
End Sub

```

```

Private Sub ArrGrid2()
    With DataGrid2
        .Columns(0).Visible = False
        .Columns(1).Caption = "เบอร์โทรศัพท์"
        .Columns(1).Alignment = dbgLef
        .Columns(1).Width = 1400
        .Columns(2).Caption = "ชื่อ"
        .Columns(2).Alignment = dbgLef
        .Columns(2).Width = 1200
        .Columns(3).Caption = "ที่อยู่"
        .Columns(3).Alignment = dbgLef
        .Columns(3).Width = 4100
        .Columns(4).Caption = "เริ่มโทร"
        .Columns(4).Alignment = dbgRight
        .Columns(4).Width = 1900
        .Columns(4).NumberFormat = "dd mmmmm yyyy hh:mm:ss "
        .Columns(5).Caption = "วางหู"
        .Columns(5).Alignment = dbgRight
        .Columns(5).Width = 1900
        .Columns(5).NumberFormat = "dd mmmmm yyyy hh:mm:ss "
        .Columns(6).Caption = "เวลา"
        .Columns(6).Alignment = dbgLef
        .Columns(6).Width = 800
    End With
End Sub

```

```

Function CalTime(ByVal LTime As Date, NTime As String) As Date
Dim ArrTime(6) As String
Dim MTime(6) As String
Dim OrrTime(6) As String

```

```

Dim k As Integer
For i = 0 To (Len(NTime) / 2) - 1
    If i <> 3 Then
        ArrTime(k) = Mid(NTime, i * 2 + 1, 2)
        k = k + 1
    End If
Next i
OrrTime(0) = CStr(Second(LTime))
OrrTime(1) = CStr(Minute(LTime))
OrrTime(2) = CStr(Hour(LTime))
OrrTime(3) = CStr(Day(LTime))
OrrTime(4) = CStr(Month(LTime))
OrrTime(5) = Format(LTime, "yy")
k = k - 1
For i = 0 To 5
    If k >= i Then
        MTime(i) = ArrTime(k - i)
    Else
        MTime(i) = OrrTime(i)
    End If
Next i
CalTime = MTime(3) + "/" + MTime(4) + "/20" + MTime(5) + " " + MTime
(2) + ":" + MTime(1) + ":" + MTime(0)
End Function

Private Function CalMinute(Sec As Integer) As String
Dim Temp As String
Dim Result As String
Dim T2 As String
Dim Num As Integer
If Sec > 3600 Then
    Num = Sec \ 3600
    Result = Result + CStr(Num) + ":"
    Sec = Sec Mod 3600
End If
If Sec > 60 Then
    Num = Sec \ 60
    Result = Result + CStr(Num) + ":"
    Sec = Sec Mod 60
End If
T2 = CStr(Sec)
If Len(T2) = 1 Then
    T2 = "0" + T2
End If
Result = Result + T2
CalMinute = Result
End Function

```

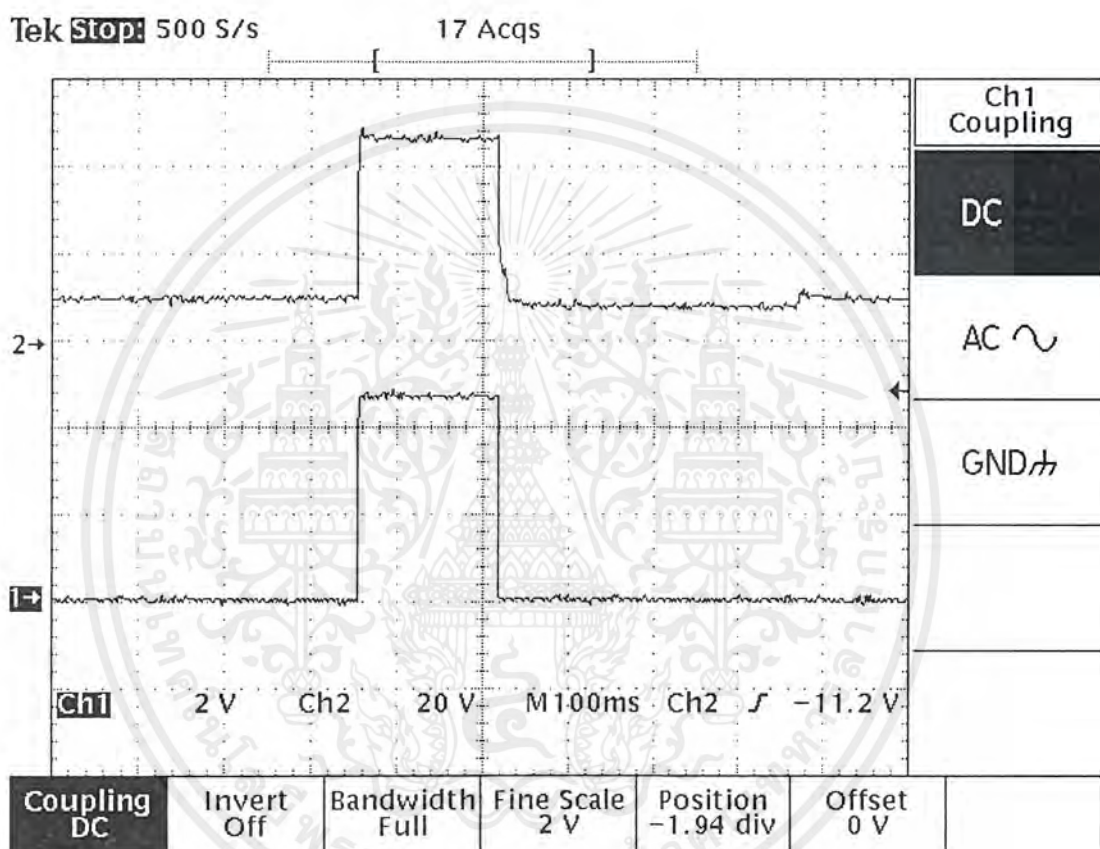
บทที่ 4

การทดลองและผลการทดลอง

4.1 การทดลองวงจรดีเทกเตอร์

4.1.1 วงจรตรวจสอบการยกหูโทรศัพท์

จากวงจรรูปที่ 3.2 จะได้ผลการทดลองดังรูป

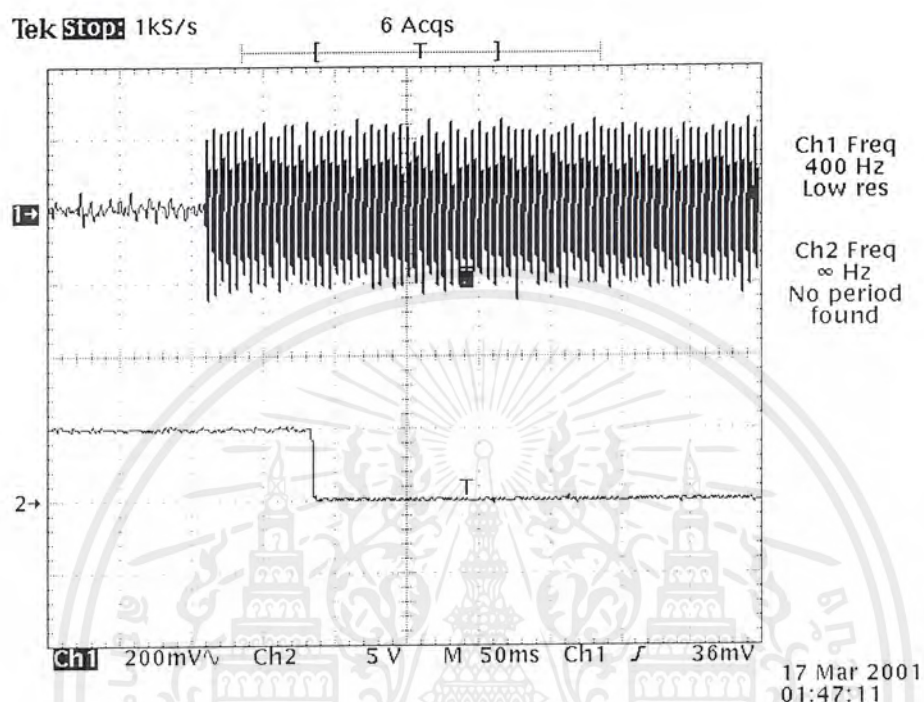


รูปที่ 4.1 แสดงรูปสัญญาณที่ได้จากวงจรตรวจสอบการยกหูโทรศัพท์

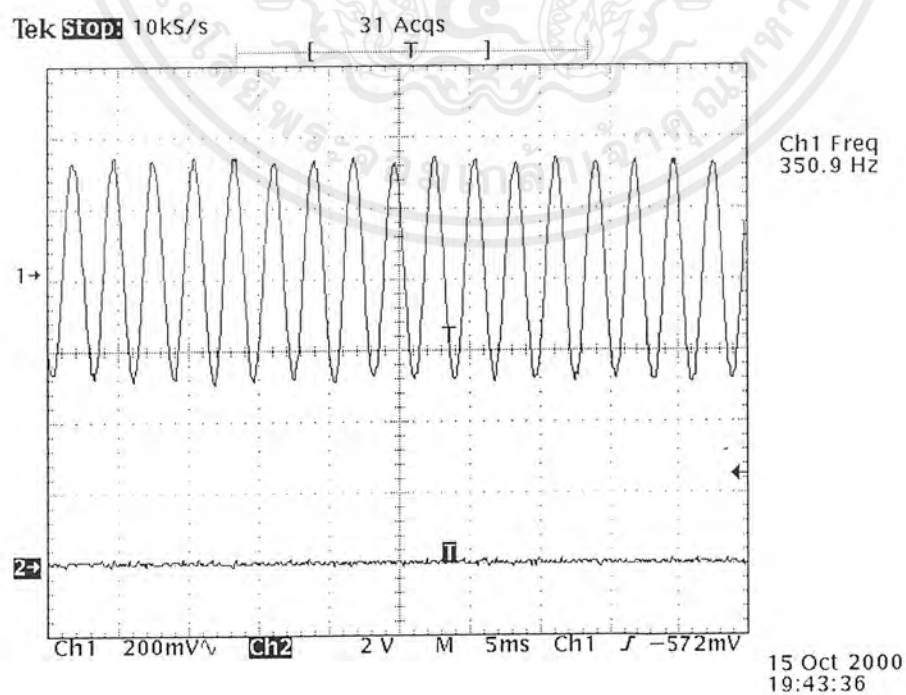
จากรูปที่ 4.1 รูปด้านล่างจะเป็นเอาต์พุตที่ได้จากวงจรตรวจสอบการยกหูโทรศัพท์ จะเห็นว่า ขณะที่คู่สายวางห้อยอยู่แรงดันไฟฟ้าจะถูกเปลี่ยนให้เป็นแรงดัน 5 โวลต์ แทนด้วยระดับของลอจิก “1” เพื่อให้สามารถใช้งานร่วมกับระบบไมโครคอนโทรลเลอร์ได้ ส่วนในขณะการยกหู แรงดันจะลดลงเหลือ 0 โวลต์ แทนด้วยระดับของลอจิก “0” รูปด้านบนเป็นสัญญาณที่วัดคร่อมคู่สายโทรศัพท์ ซึ่งในขณะทำการวางห้อยระดับแรงดันไฟฟ้าจะมีค่าประมาณ 48 โวลต์ ขณะที่ทำการยกหูแรงดันไฟฟ้าจะลดลงเหลือประมาณ 10 โวลต์

4.1.2 วงจรตรวจสอบสัญญาณเรียกกลับ

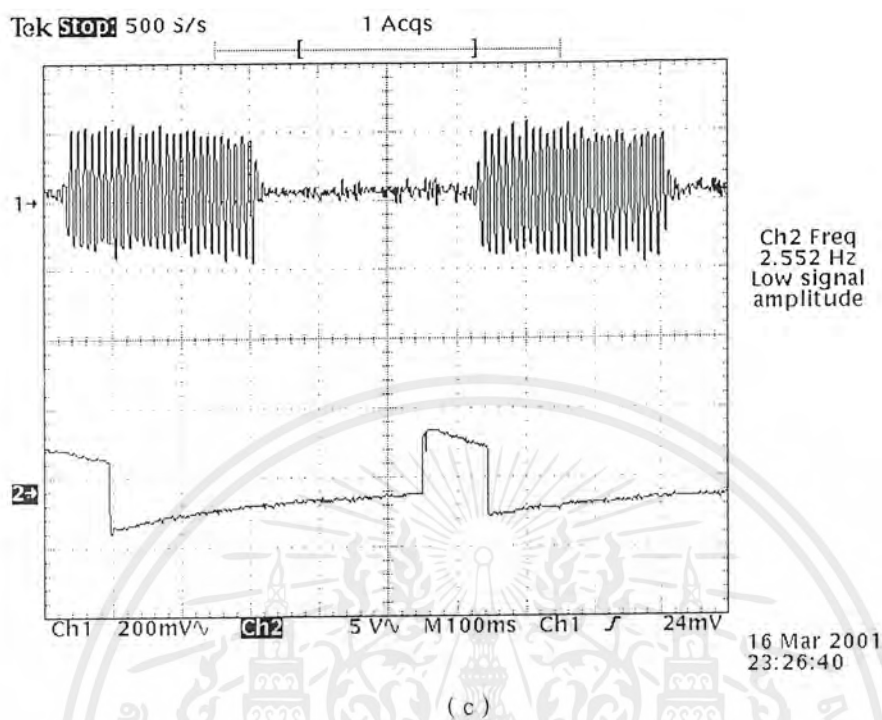
จากรูปที่ 3.3 จะได้ผลการทดลองดังรูป



(a)

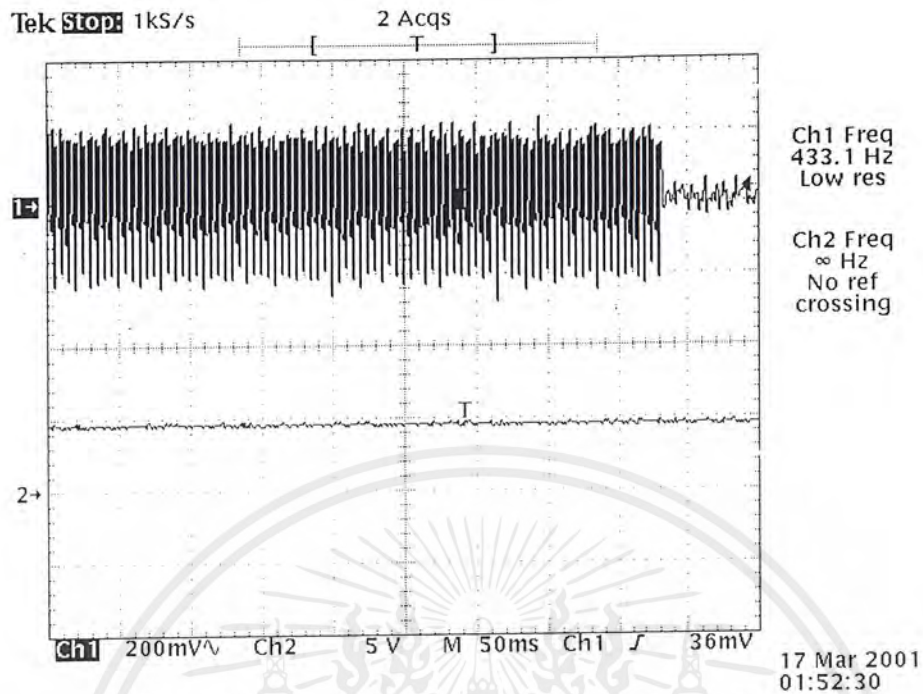


(b)

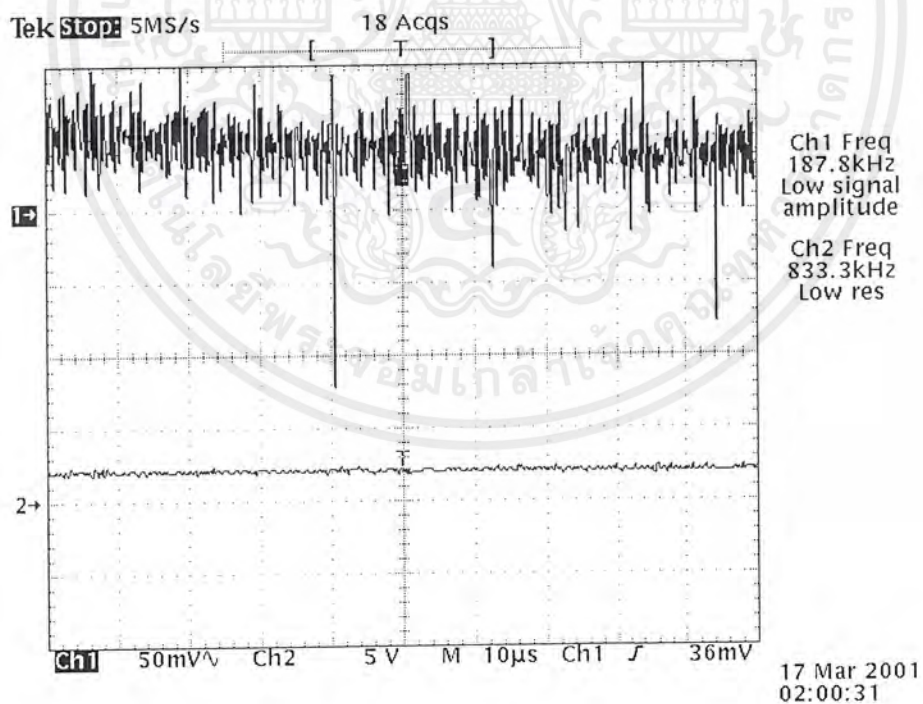


รูปที่ 4.2 แสดงรูปสัญญาณที่ได้จากเอาต์พุตของ IC LM567

จากรูปที่ 4.2 (a) เป็นสัญญาณที่ได้จากการตรวจสอบสัญญาณเรียกกลับ ซึ่งแทนด้วยระดับของลอจิก “0” เมื่อมีสัญญาณเรียกกลับความถี่ประมาณ 425 Hz เข้ามา และแทนด้วยระดับลอจิก “1” เมื่อมีความถี่อื่นเข้ามา ส่วนรูปที่ 4.2 (b) เป็นสัญญาณที่ได้จากการตรวจสอบสัญญาณให้หมุนหรือให้กดหมายเลขซึ่งจะเป็นระดับลอจิก “0” ตลอด ส่วนรูปที่ 4.2 (c) เป็นสัญญาณที่ได้จากการตรวจสอบสัญญาณไม่ว่าง



(a)

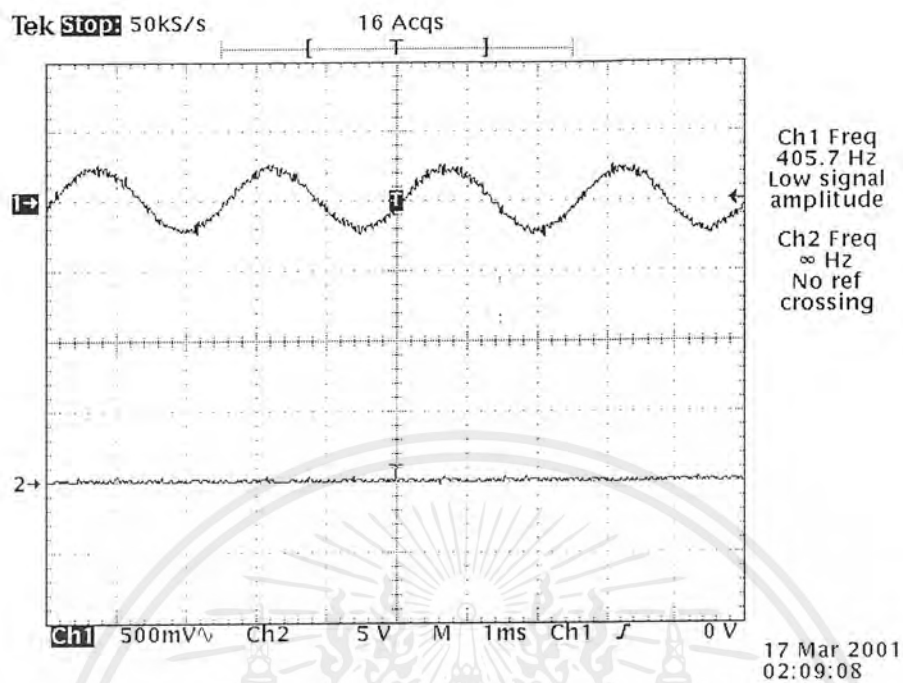


(b)

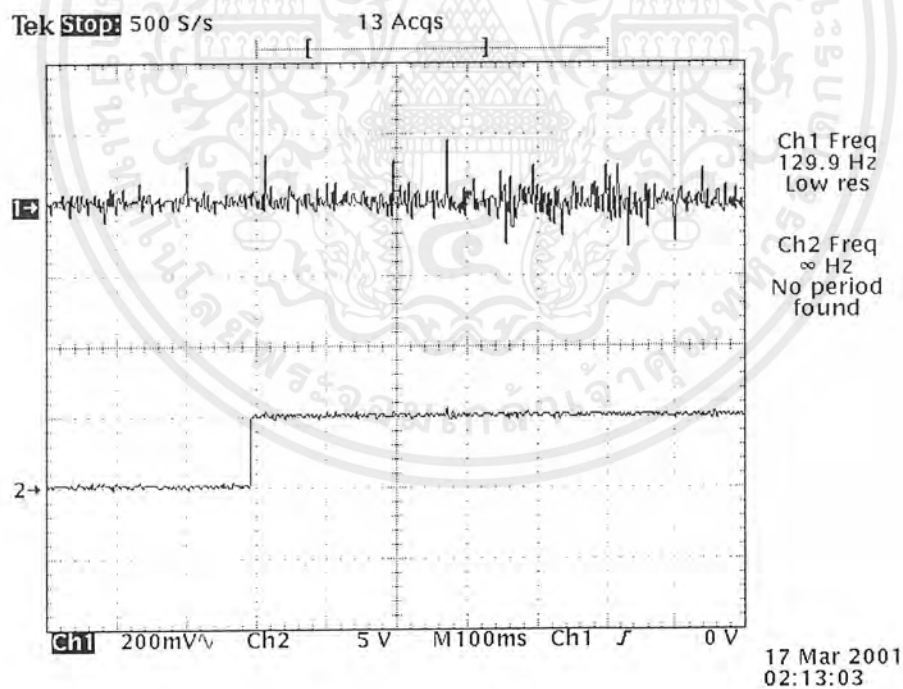
รูปที่ 4.3 แสดงรูปสัญญาณที่ได้จากเอาต์พุตของ IC 741

รูป (a) สัญญาณเมื่อปลายทางยังไม่รับสาย

รูป (b) สัญญาณเมื่อปลายทางรับสาย



(a)



(b)

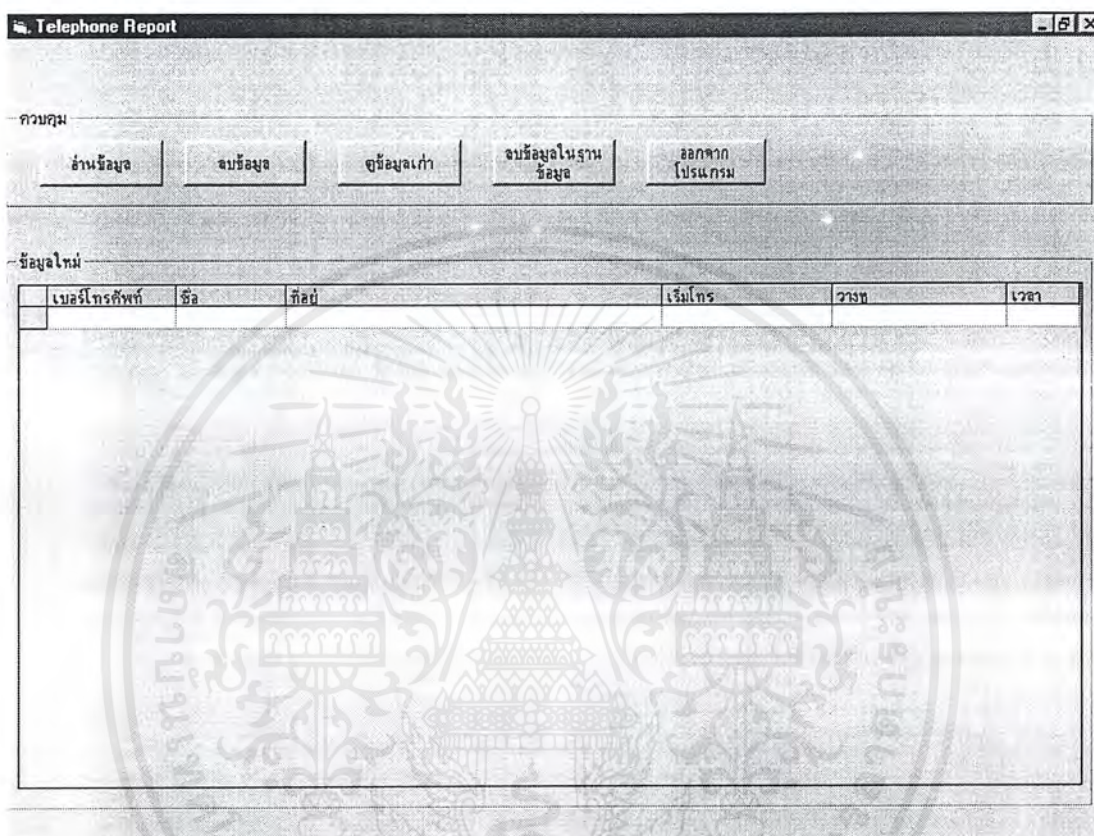
รูปที่ 4.4 แสดงรูปสัญญาณที่ได้จากเอาต์พุตของวงจรตรวจสอบสัญญาณเรียกกลับ

รูป (a) สัญญาณเมื่อปลายทางยังไม่รับสาย

รูป (b) สัญญาณเมื่อปลายทางรับสาย

4.2 การทดลองและผลการทดลองในส่วนแสดงผลและประมวลผล

ในส่วนแสดงผลนี้จะทำการทดลองแสดงค่าที่รับได้จากส่วนวงจรดีเทกเตอร์ นำมาเก็บไว้ในคอมพิวเตอร์ โดยติดต่อผ่าน พอร์ทอนุกรมของ คอมพิวเตอร์ (COM1 หรือ COM2) และเมื่อทำการ RUN โปรแกรม Visual basic จะปรากฏหน้าจอดังรูปที่ 1



รูปที่ 4.5 แสดงรูปแบบเมื่อทำการ RUN โปรแกรม

จากรูปปุ่มคำสั่งต่างๆสามารถอธิบายได้ดังนี้

ส่วน ควบคุม

อ่านข้อมูล : อ่านข้อมูลที่ดึงมาจาก ฐานข้อมูล(Database)

ลบข้อมูล : ลบข้อมูลที่เก็บไว้ใน RAM

เรียกดูข้อมูลเก่า : แสดงข้อมูลเก่าที่เคยอ่านมาแล้วทั้งหมด กรณีที่มีการอ่านข้อมูลซ้ำ(Read Data)

ในส่วนนี้ก็จะแสดงข้อมูลซ้ำด้วยเช่นกัน

ลบข้อมูลในฐานข้อมูล : ลบข้อมูลเก่าที่เคยอ่านมาแล้วทั้งหมด

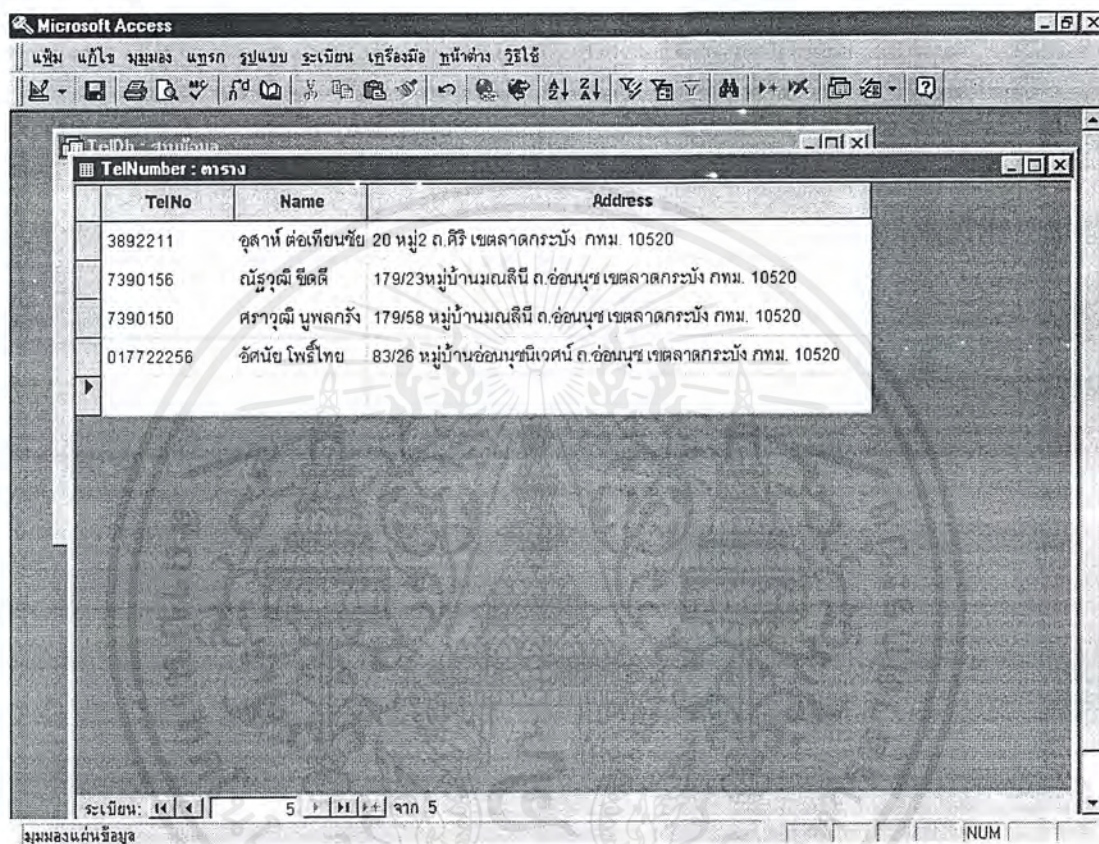
ออกจากโปรแกรม : คำสั่งที่ใช้ในการออกจากโปรแกรม

ส่วน ข้อมูลใหม่

จะเป็นส่วนที่แสดงหมายเลขโทรศัพท์ รวมทั้งชื่อ , ที่อยู่,เวลาที่เริ่มโทร,เวลาที่วางหูโทรศัพท์และเวลาที่ใช้ในการโทรทั้งหมด ซึ่งข้อมูลดังกล่าวจะถูกดึงมาจากฐานข้อมูล (Database)

ในส่วนของฐานข้อมูลจะใช้ Microsoft Access ในการจัดเก็บข้อมูลต่างๆ ในการสร้างข้อมูลและเก็บข้อมูลนั้นจะเก็บเฉพาะเบอร์โทร ชื่อ และที่อยู่ซึ่งเมื่อทำการ อ่านข้อมูล ข้อมูลส่วนดังกล่าวจะถูกเรียกขึ้นมาใช้งาน จากรูปที่ 2 จะแสดงรายละเอียดต่างๆ ที่กล่าวมาแล้ว

โดยในที่นี้จะสมมติเพียง 4 หมายเลขเท่านั้น ซึ่งในการใช้งานจริงเราต้องเพิ่มข้อมูลในส่วนนี้เข้าไปอีก



รูปที่ 4.6 แสดงข้อมูลในฐานข้อมูลของ Microsoft Access

ขั้นตอนการทดลอง

1. จัดเตรียมวงจรดีเทคเตอร์และบอร์ดไมโครคอนโทรลเลอร์ให้อยู่ในสภาวะที่พร้อมจะใช้งาน
2. โทรหมายเลขปลายทาง 4 เลขหมายเรียงตามลำดับข้อมูลในรูปที่ 2
3. จากนั้นทำการโทรไปหมายเลขใดๆ ที่ไม่ใช่หมายเลขในฐานข้อมูล 1 เลขหมาย
4. เมื่อโทรเสร็จทุกเลขหมายแล้วให้ทำการ Run โปรแกรม Visual Basic เพื่ออ่านข้อมูลที่โทรออก

ผลการทดลอง

จากการทดลอง Run โปรแกรมจะได้ผลดังรูปที่ 3 ซึ่งหมายเลข 038422515 นั้น ในส่วนที่แสดง ซึ่งจะปรากฏข้อความว่า “Not found” หมายความว่า หมายเลขดังกล่าวไม่มีอยู่ในฐานข้อมูล (ดังรูปที่ 2) แต่โปรแกรมก็สามารถที่จะบอก เวลาเริ่มโทร เวลาวางหู และระยะเวลาในการโทรได้

ส่วนหมายเลข 7390150 จากการทดลองเมื่อทำการกดหมายเลขดังกล่าว เนื่องจากปลายทางไม่มีผู้รับสาย ดังนั้นจึงไม่แสดงข้อมูลออกมาและจะแสดงเฉพาะข้อมูลของเลขหมายที่มีผู้รับสายเท่านั้น

เบอร์โทรศัพท์	ชื่อ	ที่อยู่	เริ่มโทร	วางท	เวลา
3892211	สุสา ต่อเกียรติ	20 ม.2 ต.ศิริ เขตจตุจักรบึง กทม. 10520	19 March 2001 09:25:20	19 March 2001 09:26:55	1:35
7390156	ฉวีราณี รัตติ	179/23 หมู่บ้านมณีนี ต.อ่อนนุช เขตจตุจักรบึง กทม. 1052	19 March 2001 09:27:39	19 March 2001 09:28:18	39
017722256	อัศนัย โพธิ์ไทย	83/26 หมู่บ้านอ่อนนุชนิเวศน์ ต.อ่อนนุช เขตจตุจักรบึง กทม.	19 March 2001 09:29:51	19 March 2001 09:30:53	1:02
▶ 038422515	Not found		19 March 2001 09:31:31	19 March 2001 09:33:17	1:46

รูปที่ 4.7 แสดงข้อมูลต่างๆ ที่ได้จากการทดลอง

บทที่ 5

บทสรุปและวิจารณ์

5.1 ปัญหาและแนวทางการแก้ไข

ในการปฏิบัติและทดลองวงจรตรวจสอบการยกหูโทรศัพท์จะพบว่า ค่าของ R1 จะมีผลกระทบต่อวงจรดีเทคเตอร์เพราะถ้าหากกำหนดค่า R1 ไม่เหมาะสม จะมีปัญหาทำให้เมื่อมีการยกหูโทรศัพท์แล้วสัญญาณให้กดเลขหมายโทรศัพท์ (Dial Tone) จะขาดหายไปทำให้เราไม่สามารถติดต่อสื่อสารได้ ทั้งนี้ถ้าหากเจอปัญหาดังกล่าวให้เพิ่มค่า R1 ขึ้นจนสัญญาณต่างๆกลับมาเป็นปกติ ซึ่งค่า R1 จะมีค่าเท่าไรนั้นจะต้องทดลองเปลี่ยนค่าไปเรื่อยๆ พร้อมทั้งสังเกตการเปลี่ยนแปลงของเอาต์พุตของวงจรด้วย เพราะถ้าค่า R1 มีค่ามากเกินไป จะทำให้กระแสที่ส่งไปยังอปแอมป์ไอโซเลเตอร์มีค่าน้อยเกินไปทำให้แรงดันที่เอาต์พุตของวงจรลดต่ำลงและในขณะเดียวกันค่าของ R2 จะต้องมีความเหมาะสมด้วย ซึ่งเมื่อเรานำไปต่อกับ MCS 51 จะพบปัญหาคือ แรงดันที่เอาต์พุตจะลดต่ำลง เพราะฉะนั้นเมื่อต้องจรส่วนนี้เข้ากับ MCS 51 จะต้องปรับค่า R2 ให้เหมาะสม (ปรับจนกระทั่งได้เอาต์พุตตามต้องการ)

ในส่วนของวงจรตรวจสอบสัญญาณเรียกกลับนั้น สัญญาณที่ผ่านเมทซิงอิมพีแดนซ์จะมีสัญญาณรบกวนอยู่มาก เพื่อแก้ปัญหาสภาวะดังกล่าวควรต่อคาปาซิเตอร์คร่อมทางขดขั้วขั้วมูเอาไว้จะแก้ปัญหาดังนี้ได้ และ VR1 ควรใช้ค่าที่มีการปรับค่าได้ละเอียดมาก ส่วนค่าของ C16 และ R11 นั้นจะต้องกำหนดให้เหมาะสมเพราะค่าของอุปกรณ์ทั้งสองจะเป็นตัวกำหนดค่าไทม์คอนสแตนต์ในการเปรียบเทียบระดับแรงดันของอปแอมป์ 741 ถ้าค่าของไทม์คอนสแตนต์น้อยเกินไป จะทำให้แรงดันที่ขา 3 ในขณะที่ยังไม่มีการรับสายมีค่าน้อยกว่าแรงดันที่ขา 2 ทำให้เอาต์พุตที่ขา 6 ไม่ได้ค่าตามที่เรากำลังต้องการ

ในส่วนในส่วนควบคุมนั้นถ้าหากไฟดับจะทำให้ข้อมูลที่เก็บไว้ใน RAM นั้นหายไป และในขณะเดียวกันข้อมูลของ RTC ก็หายไปด้วย เพราะฉะนั้นจะต้องมีการสร้างวงจรไฟสำรองข้อมูลเอาไว้ด้วย ในที่นี้จะใช้ IC MAX 691 ในการสร้างวงจรดังกล่าว ในขณะเดียวกันก็จะใช้แบตเตอรี่สำรองไฟให้กับ RTC ด้วย แต่ในที่นี้ RAM ไม่สามารถเก็บข้อมูลได้เพราะในสถานะแรกเมื่อเปิดเครื่องขึ้นมาโปรแกรมจะสั่งให้ไมโครคอนโทรลเลอร์ลบข้อมูลทุกอย่างที่อยู่ใน RTC และ RAM เพราะฉะนั้นสำหรับโครงการนี้นี้หากจะให้ RTC และ RAM เก็บข้อมูลได้จะต้องไปแก้ไขในส่วนของโปรแกรมที่ควบคุมไมโครคอนโทรลเลอร์ สำหรับ RAM นั้นใช้ 2 ตัวมีความจุรวมกันขนาด 16 กิโลไบต์ ซึ่งสามารถเก็บข้อมูลการใช้งานโทรศัพท์ได้มากกว่า 1000 ครั้ง ในส่วนของ RTC นั้น PC จะติดต่อกับ RTC แบบ Serial ซึ่งเราต้องเขียนโปรแกรมที่ควบคุมสายนำสัญญาณสองเส้นขึ้นมาเอง และต้องให้ตรงตามเงื่อนไขของการสื่อสาร ส่วนสัญญาณที่ได้จากวงจรถอดรหัสนั้นไม่แน่นอนเพราะเกิดสัญญาณรบกวนต่างๆในขณะที่ใช้งานทำให้การตรวจสอบสัญญาณทำได้ยาก และระดับของสัญญาณที่ได้จากวงจรดีเทคเตอร์นั้นไม่คงที่ในส่วนของการแสดงและประมวลผลนั้นเราจะต้องนำเอาข้อมูลที่อยู่ในรูปของรหัสแอสกีมาแปลงให้เป็นข้อมูลแบบ String แล้วนำเอา String เหล่านั้นมาทำการตัดแบ่งและแปลความหมายของข้อมูลเพื่อที่จะบันทึกข้อมูลเหล่านั้นลงใน Database ในการเก็บข้อมูลลงใน Database นั้นเราจะเลือกใช้ Database ที่สร้างบน Microsoft Access ซึ่งสนับสนุนการทำงานบน Visual Basic อย่างเต็มที่

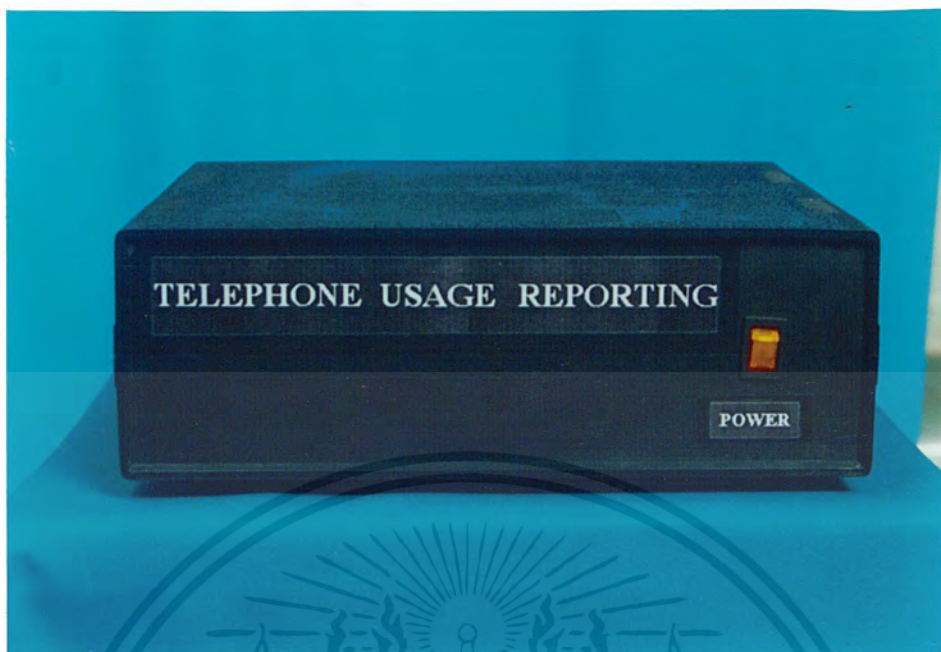
หนังสืออ้างอิง

- [1] กิตติ ภัคดีวัฒนะกุล,จำลอง ครูอุตสาหะ.Visual Basic 6 ฉบับโปรแกรมเมอร์.กรุงเทพมหานคร:ไทยเจริญการพิมพ์, 2542
- [2] แผนกหนังสือพิเศษด้านอิเล็กทรอนิกส์.รวมโครงการอิเล็กทรอนิกส์ ไมโครคอมพิวเตอร์. กรุงเทพมหานคร :ซีเอ็ดยุคชั่น, 2539
- [3] ธวัชชัย เลื่อนจวิ.เทคโนโลยีโทรศัพท์.กรุงเทพมหานคร :2536.
- [4] กฤษดา ใจเย็น, อรรถพล บุญโยภา, ชัยวัฒน์ ลิ้มพรจิตรวิไล.เรียนรู้และปฏิบัติการเชื่อมต่อคอมพิวเตอร์กับอุปกรณ์ภายนอกผ่านพอร์ตอนุกรม.กรุงเทพมหานคร:อิน โนเวตีฟ เอ็กเพอริเมนต์, 2542



ภาคผนวก

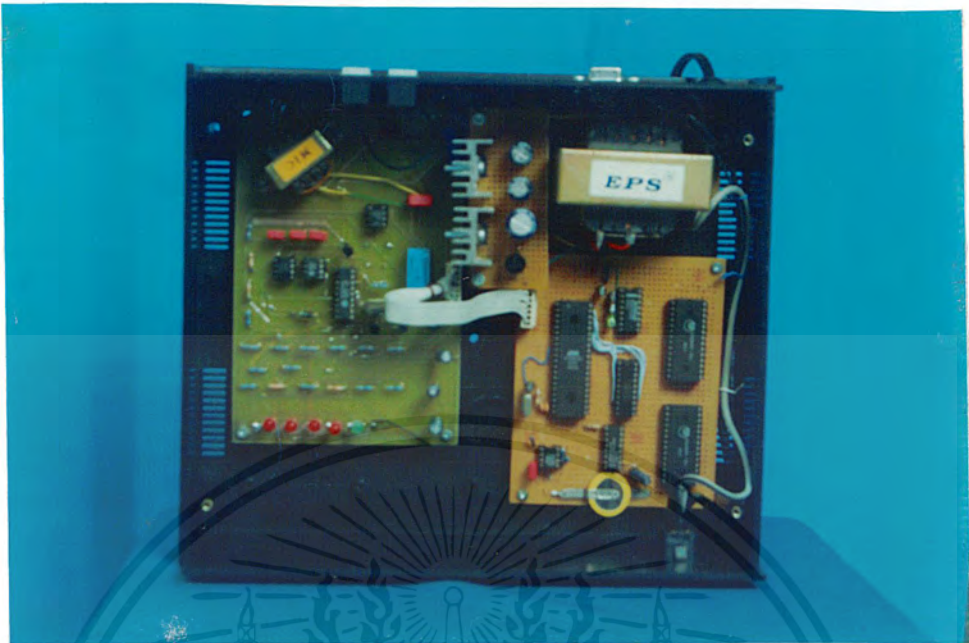




รูปแสดงด้านหน้าของเครื่องแสดงข้อมูลการใช้งานโทรศัพท์



รูปแสดงด้านหลังของเครื่องแสดงข้อมูลการใช้งานโทรศัพท์



รูปแสดงด้านบน(ภายใน)ของเครื่องแสดงข้อมูลการใช้งานโทรศัพท์



ISO²-CMOS MT8870D/MT8870D-1 Integrated DTMF Receiver

ISSUE 5

March 1997

Features

- Complete DTMF Receiver
- Low power consumption
- Internal gain setting amplifier
- Adjustable guard time
- Central office quality
- Power-down mode
- Inhibit mode
- Backward compatible with MT8870C/MT8870C-1

Ordering Information

MT8870DE/DE-1	18 Pin Plastic DIP
MT8870DS/DS-1	18 Pin SOIC
MT8870DN/DN-1	20 Pin SSOP
-40 °C to +85 °C	

Description

The MT8870D/MT8870D-1 is a complete DTMF receiver integrating both the bandsplit filter and digital decoder functions. The filter section uses switched capacitor techniques for high and low group filters; the decoder uses digital counting techniques to detect and decode all 16 DTMF tone-pairs into a 4-bit code. External component count is minimized by on chip provision of a differential input amplifier, clock oscillator and latched three-state bus interface.

Applications

- Receiver system for British Telecom (BT) or CEPT Spec (MT8870D-1)
- Paging systems
- Repeater systems/mobile radio
- Credit card systems
- Remote control
- Personal computers
- Telephone answering machine

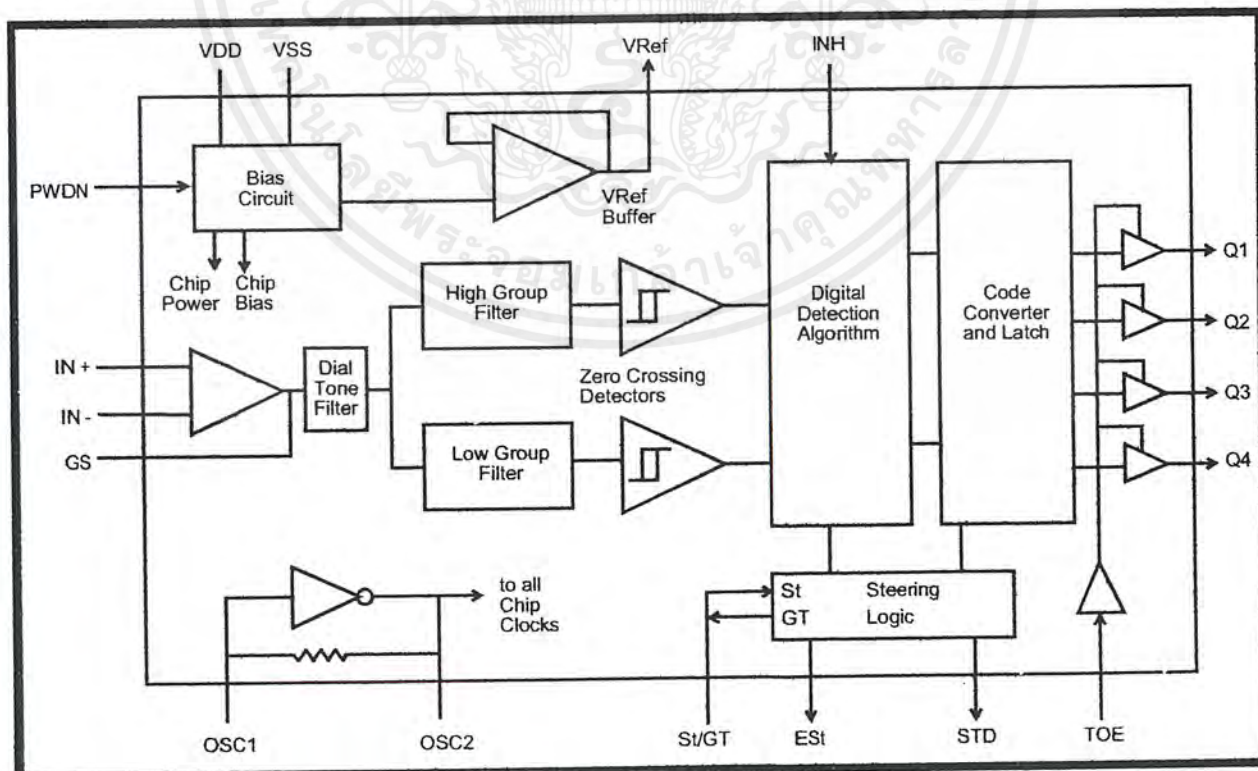


Figure 1 - Functional Block Diagram

MT8870D/MT8870D-1 ISO²-CMOS

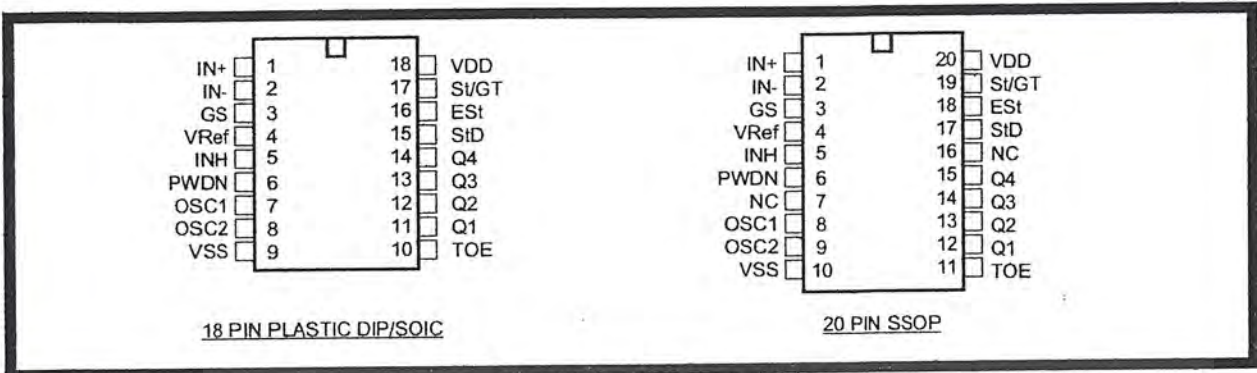


Figure 2 - Pin Connections

Pin Description

Pin #		Name	Description
18	20		
1	1	IN+	Non-Inverting Op-Amp (Input).
2	2	IN-	Inverting Op-Amp (Input).
3	3	GS	Gain Select. Gives access to output of front end differential amplifier for connection of feedback resistor.
4	4	V _{Ref}	Reference Voltage (Output). Nominally V _{DD} /2 is used to bias inputs at mid-rail (see Fig. 6 and Fig. 10).
5	5	INH	Inhibit (Input). Logic high inhibits the detection of tones representing characters A, B, C and D. This pin input is internally pulled down.
6	6	PWDN	Power Down (Input). Active high. Powers down the device and inhibits the oscillator. This pin input is internally pulled down.
7	8	OSC1	Clock (Input).
8	9	OSC2	Clock (Output). A 3.579545 MHz crystal connected between pins OSC1 and OSC2 completes the internal oscillator circuit.
9	10	V _{SS}	Ground (Input). 0V typical.
10	11	TOE	Three State Output Enable (Input). Logic high enables the outputs Q1-Q4. This pin is pulled up internally.
11-14	12-15	Q1-Q4	Three State Data (Output). When enabled by TOE, provide the code corresponding to the last valid tone-pair received (see Table 1). When TOE is logic low, the data outputs are high impedance.
15	17	StD	Delayed Steering (Output). Presents a logic high when a received tone-pair has been registered and the output latch updated; returns to logic low when the voltage on St/GT falls below V _{TSt} .
16	18	EST	Early Steering (Output). Presents a logic high once the digital algorithm has detected a valid tone pair (signal condition). Any momentary loss of signal condition will cause EST to return to a logic low.
17	19	St/GT	Steering Input/Guard time (Output) Bidirectional. A voltage greater than V _{TSt} detected at St causes the device to register the detected tone pair and update the output latch. A voltage less than V _{TSt} frees the device to accept a new tone pair. The GT output acts to reset the external steering time-constant; its state is a function of EST and the voltage on St.
18	20	V _{DD}	Positive power supply (Input). +5V typical.
	7, 16	NC	No Connection.

Functional Description

The MT8870D/MT8870D-1 monolithic DTMF receiver offers small size, low power consumption and high performance. Its architecture consists of a bandsplit filter section, which separates the high and low group tones, followed by a digital counting section which verifies the frequency and duration of the received tones before passing the corresponding code to the output bus.

Filter Section

Separation of the low-group and high group tones is achieved by applying the DTMF signal to the inputs of two sixth-order switched capacitor bandpass filters, the bandwidths of which correspond to the low and high group frequencies. The filter section also incorporates notches at 350 and 440 Hz for exceptional dial tone rejection (see Figure 3). Each filter output is followed by a single order switched capacitor filter section which smooths the signals prior to limiting. Limiting is performed by high-gain comparators which are provided with hysteresis to prevent detection of unwanted low-level signals. The outputs of the comparators provide full rail logic swings at the frequencies of the incoming DTMF signals.

Decoder Section

Following the filter section is a decoder employing digital counting techniques to determine the frequencies of the incoming tones and to verify that they correspond to standard DTMF frequencies. A complex averaging algorithm protects against tone simulation by extraneous signals such as voice while

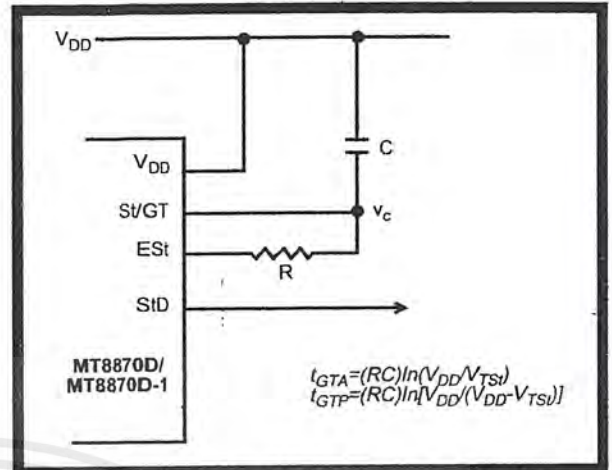


Figure 4 - Basic Steering Circuit

providing tolerance to small frequency deviations and variations. This averaging algorithm has been developed to ensure an optimum combination of immunity to talk-off and tolerance to the presence of interfering frequencies (third tones) and noise. When the detector recognizes the presence of two valid tones (this is referred to as the "signal condition" in some industry specifications) the "Early Steering" (EST) output will go to an active state. Any subsequent loss of signal condition will cause EST to assume an inactive state (see "Steering Circuit").

Steering Circuit

Before registration of a decoded tone pair, the receiver checks for a valid signal duration (referred to as character recognition condition). This check is performed by an external RC time constant driven by EST. A logic high on EST causes v_c (see Figure 4) to rise as the capacitor discharges. Provided signal

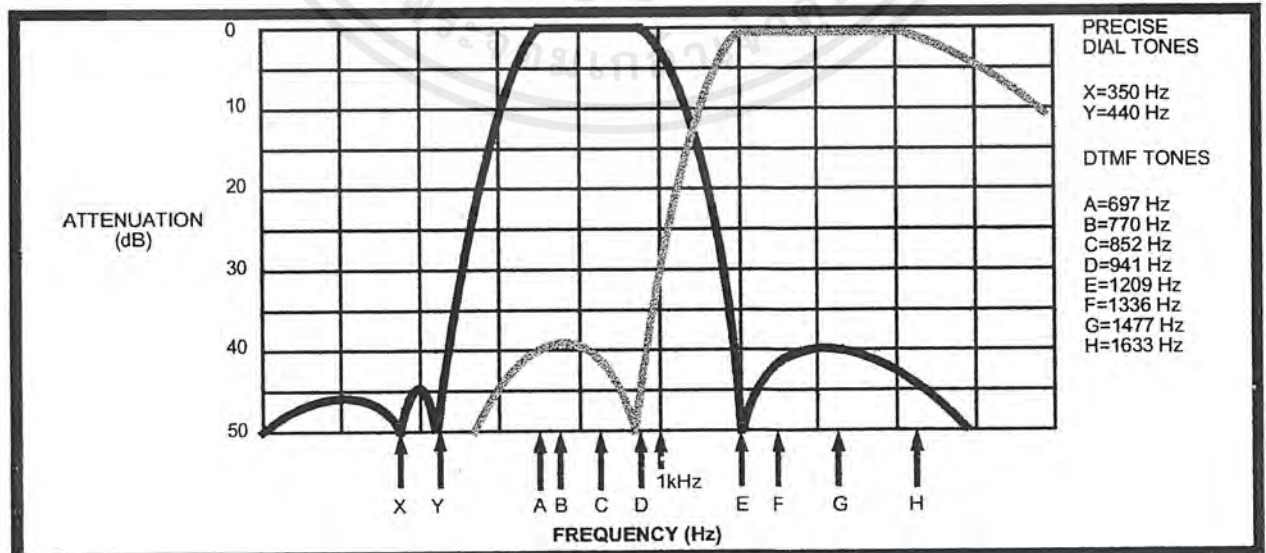


Figure 3 - Filter Response

condition is maintained (EST remains high) for the validation period (t_{GTP}), v_c reaches the threshold (V_{TSI}) of the steering logic to register the tone pair, latching its corresponding 4-bit code (see Table 1) into the output latch. At this point the GT output is activated and drives v_c to V_{DD} . GT continues to drive high as long as EST remains high. Finally, after a short delay to allow the output latch to settle, the delayed steering output flag (StD) goes high, signalling that a received tone pair has been registered. The contents of the output latch are made available on the 4-bit output bus by raising the three state control input (TOE) to a logic high. The steering circuit works in reverse to validate the interdigit pause between signals. Thus, as well as rejecting signals too short to be considered valid, the receiver will tolerate signal interruptions (dropout) too short to be considered a valid pause. This facility, together with the capability of selecting the steering time constants externally, allows the designer to tailor performance to meet a wide variety of system requirements.

Guard Time Adjustment

In many situations not requiring selection of tone duration and interdigital pause, the simple steering circuit shown in Figure 4 is applicable. Component values are chosen according to the formula:

$$t_{REC} = t_{DP} + t_{GTP}$$
$$t_{ID} = t_{DA} + t_{GTA}$$

The value of t_{DP} is a device parameter (see Figure 11) and t_{REC} is the minimum signal duration to be recognized by the receiver. A value for C of 0.1 μF is

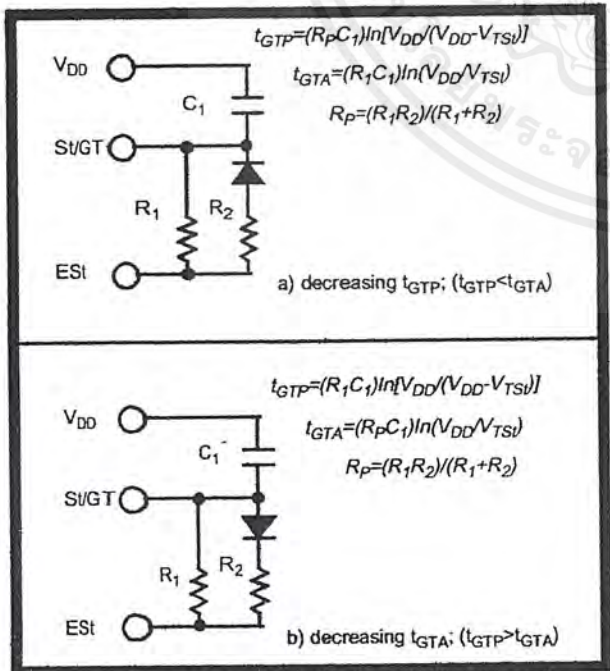


Figure 5 - Guard Time Adjustment

Digit	TOE	INH	ES _t	Q ₄	Q ₃	Q ₂	Q ₁
ANY	L	X	H	Z	Z	Z	Z
1	H	X	H	0	0	0	1
2	H	X	H	0	0	1	0
3	H	X	H	0	0	1	1
4	H	X	H	0	1	0	0
5	H	X	H	0	1	0	1
6	H	X	H	0	1	1	0
7	H	X	H	0	1	1	1
8	H	X	H	1	0	0	0
9	H	X	H	1	0	0	1
0	H	X	H	1	0	1	0
*	H	X	H	1	0	1	1
#	H	X	H	1	1	0	0
A	H	L	H	1	1	0	1
B	H	L	H	1	1	1	0
C	H	L	H	1	1	1	1
D	H	L	H	0	0	0	0
A	H	H	L	undetected, the output code will remain the same as the previous detected code			
B	H	H	L				
C	H	H	L				
D	H	H	L				

Table 1. Functional Decode Table

L=LOGIC LOW, H=LOGIC HIGH, Z=HIGH IMPEDANCE
X = DON'T CARE

recommended for most applications, leaving R to be selected by the designer.

Different steering arrangements may be used to select independently the guard times for tone present (t_{GTP}) and tone absent (t_{GTA}). This may be necessary to meet system specifications which place both accept and reject limits on both tone duration and interdigital pause. Guard time adjustment also allows the designer to tailor system parameters such as talk off and noise immunity. Increasing t_{REC} improves talk-off performance since it reduces the probability that tones simulated by speech will maintain signal condition long enough to be registered. Alternatively, a relatively short t_{REC} with a long t_{DO} would be appropriate for extremely noisy environments where fast acquisition time and immunity to tone drop-outs are required. Design information for guard time adjustment is shown in Figure 5.

Power-down and Inhibit Mode

A logic high applied to pin 6 (PWDN) will power down the device to minimize the power consumption in a standby mode. It stops the oscillator and the functions of the filters.

Inhibit mode is enabled by a logic high input to the pin 5 (INH). It inhibits the detection of tones representing characters A, B, C, and D. The output code will remain the same as the previous detected code (see Table 1).

Differential Input Configuration

The input arrangement of the MT8870D/MT8870D-1 provides a differential-input operational amplifier as well as a bias source (V_{Ref}) which is used to bias the inputs at mid-rail. Provision is made for connection of a feedback resistor to the op-amp output (GS) for adjustment of gain. In a single-ended configuration, the input pins are connected as shown in Figure 10 with the op-amp connected for unity gain and V_{Ref} biasing the input at $\frac{1}{2}V_{DD}$. Figure 6 shows the differential configuration, which permits the adjustment of gain with the feedback resistor R_5 .

Crystal Oscillator

The internal clock circuit is completed with the addition of an external 3.579545 MHz crystal and is normally connected as shown in Figure 10 (Single-Ended Input Configuration). However, it is possible to configure several MT8870D/MT8870D-1 devices employing only a single oscillator crystal. The oscillator output of the first device in the chain is coupled through a 30 pF capacitor to the oscillator input (OSC1) of the next device. Subsequent devices are connected in a similar fashion. Refer to Figure 7 for details. The problems associated with unbalanced loading are not a concern with the arrangement shown, i.e., precision balancing capacitors are not required.

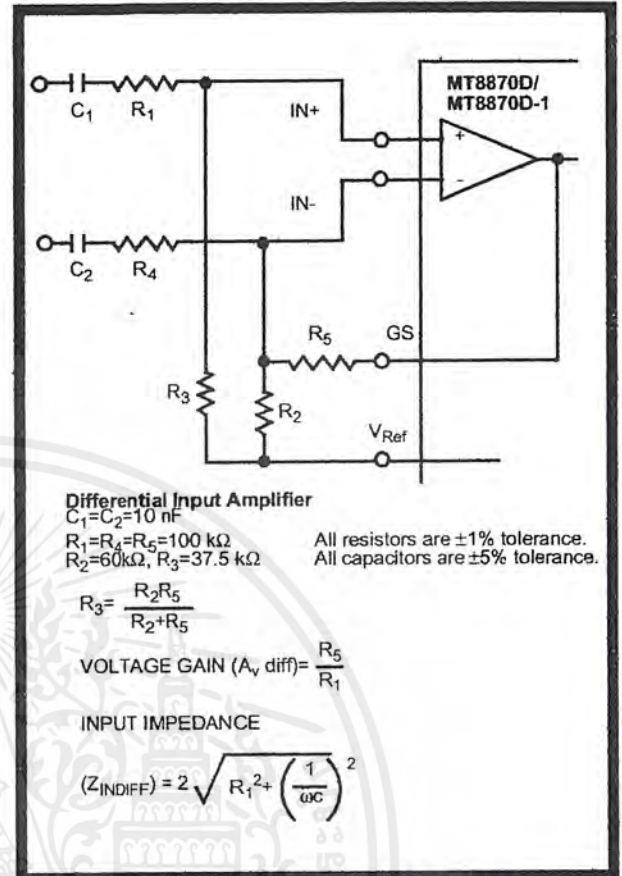


Figure 6 - Differential Input Configuration

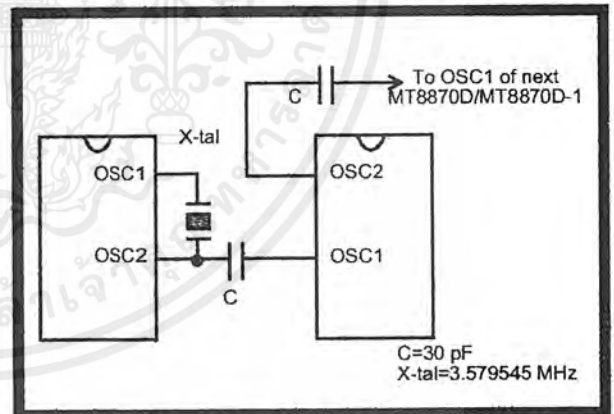


Figure 7 - Oscillator Connection

Parameter	Unit	Resonator
R1	Ohms	10.752
L1	mH	.432
C1	pF	4.984
C0	pF	37.915
Qm	-	896.37
Δf	%	±0.2%

Table 2. Recommended Resonator Specifications

Note: Qm=quality factor of RLC model, i.e., $1/2\pi\sqrt{R1C1}$.

Applications

RECEIVER SYSTEM FOR BRITISH TELECOM
SPEC POR 1151

The circuit shown in Fig. 9 illustrates the use of MT8870D-1 device in a typical receiver system. BT Spec defines the input signals less than -34 dBm as the non-operate level. This condition can be attained by choosing a suitable values of R₁ and R₂ to provide 3 dB attenuation, such that -34 dBm input signal will correspond to -37 dBm at the gain setting pin GS of MT8870D-1. As shown in the diagram, the component values of R₃ and C₂ are the guard time requirements when the total component tolerance is 6%. For better performance, it is recommended to use the non-symmetric guard time circuit in Fig. 8.

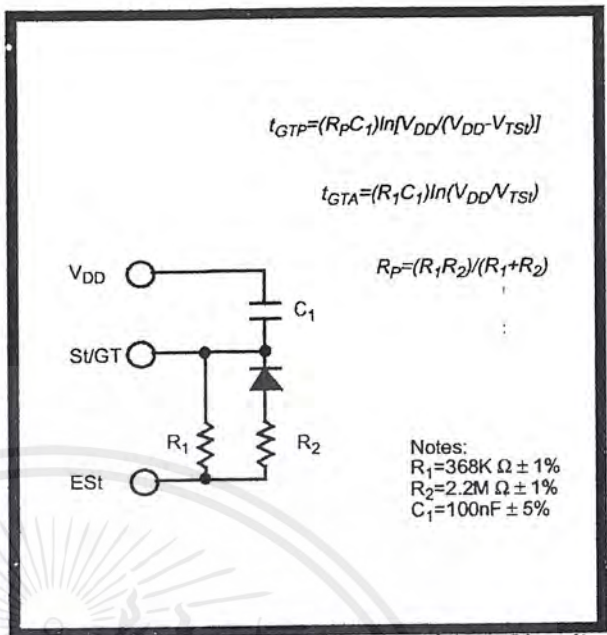


Figure 8 - Non-Symmetric Guard Time Circuit

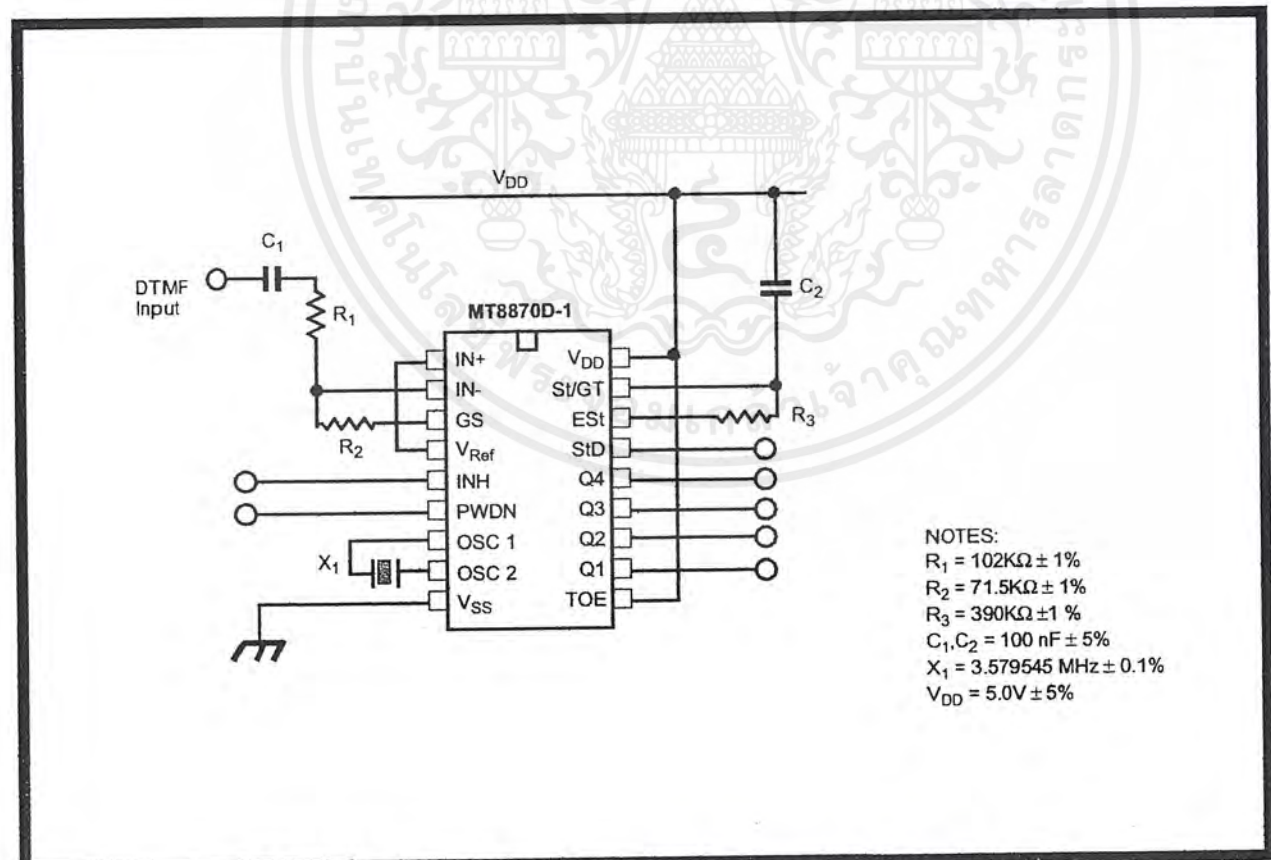


Figure 9 - Single-Ended Input Configuration for BT or CEPT Spec

Absolute Maximum Ratings[†]

	Parameter	Symbol	Min	Max	Units
1	DC Power Supply Voltage	V_{DD}		7	V
2	Voltage on any pin	V_I	$V_{SS}-0.3$	$V_{DD}+0.3$	V
3	Current at any pin (other than supply)	I_I		10	mA
4	Storage temperature	T_{STG}	-65	+150	°C
5	Package power dissipation	P_D		500	mW

[†] Exceeding these values may cause permanent damage. Functional operation under these conditions is not implied. Derate above 75 °C at 16 mW / °C. All leads soldered to board.

Recommended Operating Conditions - Voltages are with respect to ground (V_{SS}) unless otherwise stated.

	Parameter	Sym	Min	Typ [‡]	Max	Units	Test Conditions
1	DC Power Supply Voltage	V_{DD}	4.75	5.0	5.25	V	
2	Operating Temperature	T_O	-40		+85	°C	
3	Crystal/Clock Frequency	fc		3.579545		MHz	
4	Crystal/Clock Freq. Tolerance	Δfc		± 0.1		%	

[‡] Typical figures are at 25°C and are for design aid only: not guaranteed and not subject to production testing.

DC Electrical Characteristics - $V_{DD}=5.0V \pm 5\%$, $V_{SS}=0V$, $-40^\circ C \leq T_O \leq +85^\circ C$, unless otherwise stated.

		Characteristics	Sym	Min	Typ [‡]	Max	Units	Test Conditions
1	S U P P L Y	Standby supply current	I_{DDQ}		10	25	μA	PWDN= V_{DD}
2		Operating supply current	I_{DD}		3.0	9.0	mA	
3		Power consumption	P_O		15		mW	fc=3.579545 MHz
4	I N P U T S	High level input	V_{IH}	3.5			V	$V_{DD}=5.0V$
5		Low level input voltage	V_{IL}			1.5	V	$V_{DD}=5.0V$
6		Input leakage current	I_{IH}/I_{IL}		0.1		μA	$V_{IN}=V_{SS}$ or V_{DD}
7		Pull up (source) current	I_{SO}		7.5	20	μA	TOE (pin 10)=0, $V_{DD}=5.0V$
8		Pull down (sink) current	I_{SI}		15	45	μA	INH=5.0V, PWDN=5.0V, $V_{DD}=5.0V$
9		Input impedance (IN+, IN-)	R_{IN}		10		M Ω	@ 1 kHz
10		Steering threshold voltage	V_{TSt}	2.2	2.4	2.5	V	$V_{DD} = 5.0V$
11	O U T P U T S	Low level output voltage	V_{OL}			$V_{SS}+0.03$	V	No load
12		High level output voltage	V_{OH}	$V_{DD}-0.03$			V	No load
13		Output low (sink) current	I_{OL}	1.0	2.5		mA	$V_{OUT}=0.4 V$
14		Output high (source) current	I_{OH}	0.4	0.8		mA	$V_{OUT}=4.6 V$
15		V_{Ref} output voltage	V_{Ref}	2.3	2.5	2.7	V	No load, $V_{DD} = 5.0V$
16		V_{Ref} output resistance	R_{OR}		1		k Ω	

[‡] Typical figures are at 25°C and are for design aid only: not guaranteed and not subject to production testing.

MT8870D/MT8870D-1 ISO²-CMOS

Operating Characteristics - $V_{DD}=5.0V\pm5\%$, $V_{SS}=0V$, $-40^{\circ}C \leq T_O \leq +85^{\circ}C$, unless otherwise stated.
Gain Setting Amplifier

	Characteristics	Sym	Min	Typ [‡]	Max	Units	Test Conditions
1	Input leakage current	I_{IN}			100	nA	$V_{SS} \leq V_{IN} \leq V_{DD}$
2	Input resistance	R_{IN}	10			M Ω	
3	Input offset voltage	V_{OS}			25	mV	
4	Power supply rejection	PSRR	50			dB	1 kHz
5	Common mode rejection	CMRR	40			dB	$0.75 V \leq V_{IN} \leq 4.25 V$ biased at $V_{Ref}=2.5 V$
6	DC open loop voltage gain	A_{VOL}	32			dB	
7	Unity gain bandwidth	f_C	0.30			MHz	
8	Output voltage swing	V_O	4.0			V_{pp}	Load $\geq 100 k\Omega$ to V_{SS} @ GS
9	Maximum capacitive load (GS)	C_L			100	pF	
10	Resistive load (GS)	R_L			50	k Ω	
11	Common mode range	V_{CM}	2.5			V_{pp}	No Load

MT8870D AC Electrical Characteristics - $V_{DD}=5.0V \pm 5\%$, $V_{SS}=0V$, $-40^{\circ}C \leq T_O \leq +85^{\circ}C$, using Test Circuit shown in Figure 10.

	Characteristics	Sym	Min	Typ [‡]	Max	Units	Notes*
1	Valid input signal levels (each tone of composite signal)		-29		+1	dBm	1,2,3,5,6,9
			27.5		869	mV _{RMS}	1,2,3,5,6,9
2	Negative twist accept				8	dB	2,3,6,9,12
3	Positive twist accept				8	dB	2,3,6,9,12
4	Frequency deviation accept		$\pm 1.5\% \pm 2 \text{ Hz}$				2,3,5,9
5	Frequency deviation reject		$\pm 3.5\%$				2,3,5,9
6	Third tone tolerance			-16		dB	2,3,4,5,9,10
7	Noise tolerance			-12		dB	2,3,4,5,7,9,10
8	Dial tone tolerance			+22		dB	2,3,4,5,8,9,11

[‡] Typical figures are at 25 °C and are for design aid only: not guaranteed and not subject to production testing.

*NOTES

1. dBm= decibels above or below a reference power of 1 mW into a 600 ohm load.
2. Digit sequence consists of all DTMF tones.
3. Tone duration= 40 ms, tone pause= 40 ms.
4. Signal condition consists of nominal DTMF frequencies.
5. Both tones in composite signal have an equal amplitude.
6. Tone pair is deviated by $\pm 1.5\% \pm 2 \text{ Hz}$.
7. Bandwidth limited (3 kHz) Gaussian noise.
8. The precise dial tone frequencies are (350 Hz and 440 Hz) $\pm 2\%$.
9. For an error rate of better than 1 in 10,000.
10. Referenced to lowest level frequency component in DTMF signal.
11. Referenced to the minimum valid accept level.
12. Guaranteed by design and characterization.

MT8870D-1 AC Electrical Characteristics - $V_{DD}=5.0V\pm5\%$, $V_{SS}=0V$, $-40^{\circ}C \leq T_O \leq +85^{\circ}C$, using Test Circuit shown in Figure 10.

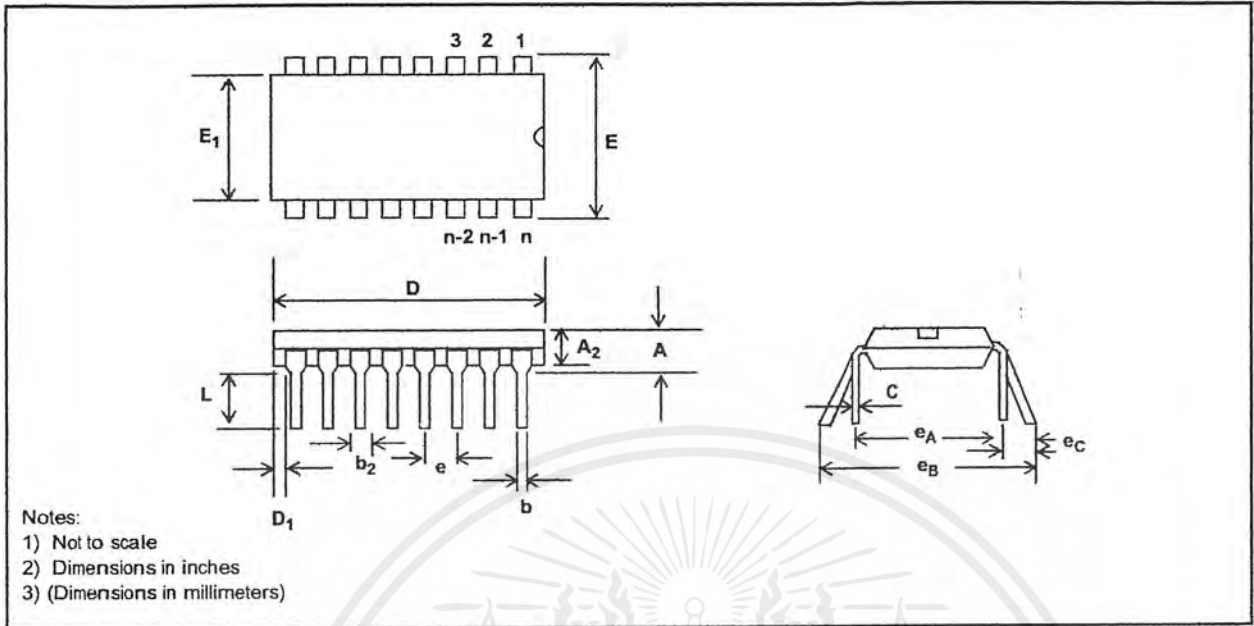
	Characteristics	Sym	Min	Typ [‡]	Max	Units	Notes*
1	Valid input signal levels (each tone of composite signal)		-31		+1	dBm	Tested at $V_{DD}=5.0V$ 1,2,3,5,6,9
			21.8		869	mV _{RMS}	
2	Input Signal Level Reject		-37			dBm	Tested at $V_{DD}=5.0V$ 1,2,3,5,6,9
			10.9			mV _{RMS}	
3	Negative twist accept				8	dB	2,3,6,9,13
4	Positive twist accept				8	dB	2,3,6,9,13
5	Frequency deviation accept		$\pm 1.5\% \pm 2$ Hz				2,3,5,9
6	Frequency deviation reject		$\pm 3.5\%$				2,3,5,9
7	Third zone tolerance			-18.5		dB	2,3,4,5,9,12
8	Noise tolerance			-12		dB	2,3,4,5,7,9,10
9	Dial tone tolerance			+22		dB	2,3,4,5,8,9,11

[‡] Typical figures are at 25 °C and are for design aid only: not guaranteed and not subject to production testing.

***NOTES**

1. dBm= decibels above or below a reference power of 1 mW into a 600 ohm load.
2. Digit sequence consists of all DTMF tones.
3. Tone duration= 40 ms, tone pause= 40 ms.
4. Signal condition consists of nominal DTMF frequencies.
5. Both tones in composite signal have an equal amplitude.
6. Tone pair is deviated by $\pm 1.5\% \pm 2$ Hz.
7. Bandwidth limited (3 kHz) Gaussian noise.
8. The precise dial tone frequencies are (350 Hz and 440 Hz) $\pm 2\%$.
9. For an error rate of better than 1 in 10,000.
10. Referenced to lowest level frequency component in DTMF signal.
11. Referenced to the minimum valid accept level.
12. Referenced to Fig. 10 input DTMF tone level at -25dBm (-28dBm at GS Pin) interference frequency range between 480-3400Hz.
13. Guaranteed by design and characterization.

Package Outlines

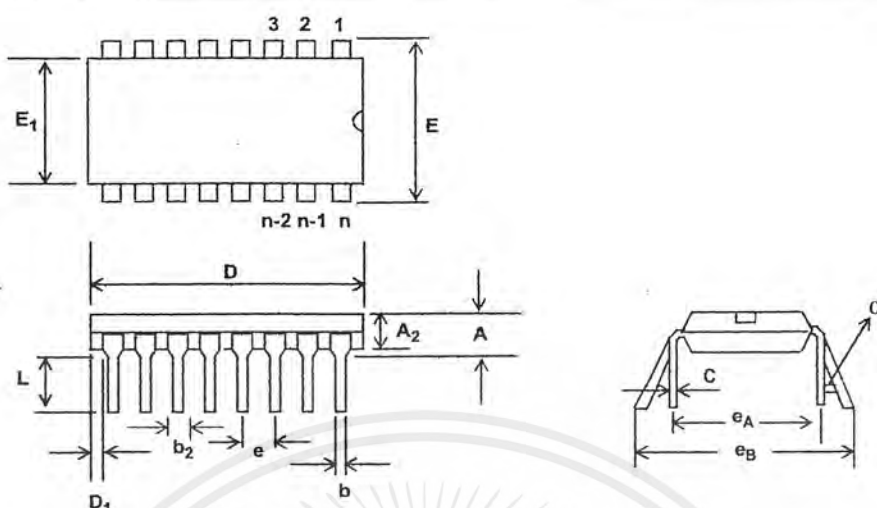


Plastic Dual-In-Line Packages (PDIP) - E Suffix

DIM	8-Pin		16-Pin		18-Pin		20-Pin	
	Plastic		Plastic		Plastic		Plastic	
	Min	Max	Min	Max	Min	Max	Min	Max
A		0.210 (5.33)		0.210 (5.33)		0.210 (5.33)		0.210 (5.33)
A ₂	0.115 (2.92)	0.195 (4.95)	0.115 (2.92)	0.195 (4.95)	0.115 (2.92)	0.195 (4.95)	0.115 (2.92)	0.195 (4.95)
b	0.014 (0.356)	0.022 (0.558)	0.014 (0.356)	0.022 (0.558)	0.014 (0.356)	0.022 (0.558)	0.014 (0.356)	0.022 (0.558)
b ₂	0.045 (1.14)	0.070 (1.77)	0.045 (1.14)	0.070 (1.77)	0.045 (1.14)	0.070 (1.77)	0.045 (1.14)	0.070 (1.77)
C	0.008 (0.203)	0.014 (0.356)	0.008 (0.203)	0.014 (0.356)	0.008 (0.203)	0.014 (0.356)	0.008 (0.203)	0.014 (0.356)
D	0.355 (9.02)	0.400 (10.16)	0.780 (19.81)	0.800 (20.32)	0.880 (22.35)	0.920 (23.37)	0.980 (24.89)	1.060 (26.9)
D ₁	0.005 (0.13)		0.005 (0.13)		0.005 (0.13)		0.005 (0.13)	
E	0.300 (7.62)	0.325 (8.26)	0.300 (7.62)	0.325 (8.26)	0.300 (7.62)	0.325 (8.26)	0.300 (7.62)	0.325 (8.26)
E ₁	0.240 (6.10)	0.280 (7.11)	0.240 (6.10)	0.280 (7.11)	0.240 (6.10)	0.280 (7.11)	0.240 (6.10)	0.280 (7.11)
e	0.100 BSC (2.54)		0.100 BSC (2.54)		0.100 BSC (2.54)		0.100 BSC (2.54)	
e _A	0.300 BSC (7.62)		0.300 BSC (7.62)		0.300 BSC (7.62)		0.300 BSC (7.62)	
L	0.115 (2.92)	0.150 (3.81)	0.115 (2.92)	0.150 (3.81)	0.115 (2.92)	0.150 (3.81)	0.115 (2.92)	0.150 (3.81)
e _B		0.430 (10.92)		0.430 (10.92)		0.430 (10.92)		0.430 (10.92)
e _C	0	0.060 (1.52)	0	0.060 (1.52)	0	0.060 (1.52)	0	0.060 (1.52)

NOTE: Controlling dimensions in parenthesis () are in millimeters.

Package Outlines



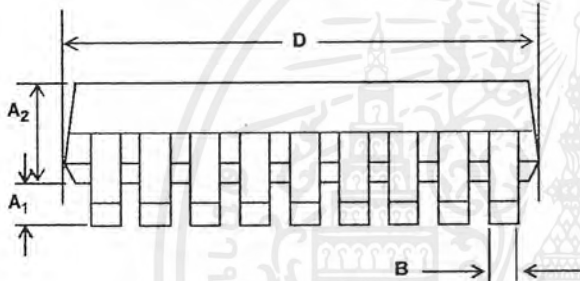
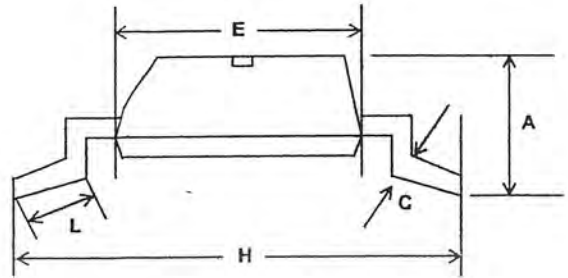
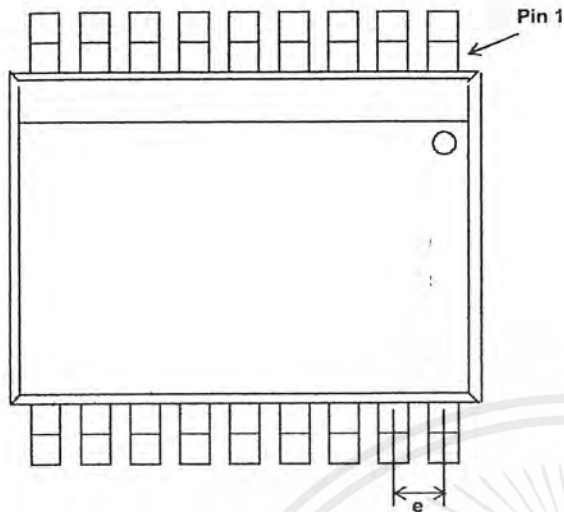
Notes:

- 1) Not to scale
- 2) Dimensions in inches
- 3) (Dimensions in millimeters)

Plastic Dual-In-Line Packages (PDIP) - E Suffix

DIM	22-Pin		24-Pin		28-Pin		40-Pin	
	Plastic		Plastic		Plastic		Plastic	
	Min	Max	Min	Max	Min	Max	Min	Max
A		0.210 (5.33)		0.250 (6.35)		0.250 (6.35)		0.250 (6.35)
A ₂	0.125 (3.18)	0.195 (4.95)	0.125 (3.18)	0.195 (4.95)	0.125 (3.18)	0.195 (4.95)	0.125 (3.18)	0.195 (4.95)
b	0.014 (0.356)	0.022 (0.558)	0.014 (0.356)	0.022 (0.558)	0.014 (0.356)	0.022 (0.558)	0.014 (0.356)	0.022 (0.558)
b ₂	0.045 (1.15)	0.070 (1.77)	0.030 (0.77)	0.070 (1.77)	0.030 (0.77)	0.070 (1.77)	0.030 (0.77)	0.070 (1.77)
C	0.008 (0.204)	0.015 (0.381)	0.008 (0.204)	0.015 (0.381)	0.008 (0.204)	0.015 (0.381)	0.008 (0.204)	0.015 (0.381)
D	1.050 (26.67)	1.120 (28.44)	1.150 (29.3)	1.290 (32.7)	1.380 (35.1)	1.565 (39.7)	1.980 (50.3)	2.095 (53.2)
D ₁	0.005 (0.13)		0.005 (0.13)		0.005 (0.13)		0.005 (0.13)	
E	0.390 (9.91)	0.430 (10.92)	0.600 (15.24)	0.670 (17.02)	0.600 (15.24)	0.670 (17.02)	0.600 (15.24)	0.670 (17.02)
E ₁	0.330 (8.39)	0.380 (9.65)	0.485 (12.32)	0.580 (14.73)	0.485 (12.32)	0.580 (14.73)	0.485 (12.32)	0.580 (14.73)
E ₁			0.246 (6.25)	0.254 (6.45)				
e	0.100 BSC (2.54)		0.100 BSC (2.54)		0.100 BSC (2.54)		0.100 BSC (2.54)	
e _A	0.400 BSC (10.16)		0.600 BSC (15.24)		0.600 BSC (15.24)		0.600 BSC (15.24)	
e _A			0.300 BSC (7.62)					
e _B				0.430 (10.92)				
L	0.115 (2.93)	0.160 (4.06)	0.115 (2.93)	0.200 (5.08)	0.115 (2.93)	0.200 (5.08)	0.115 (2.93)	0.200 (5.08)
α		15°		15°		15°		15°

Shaded areas for 300 Mil Body Width 24 PDIP only

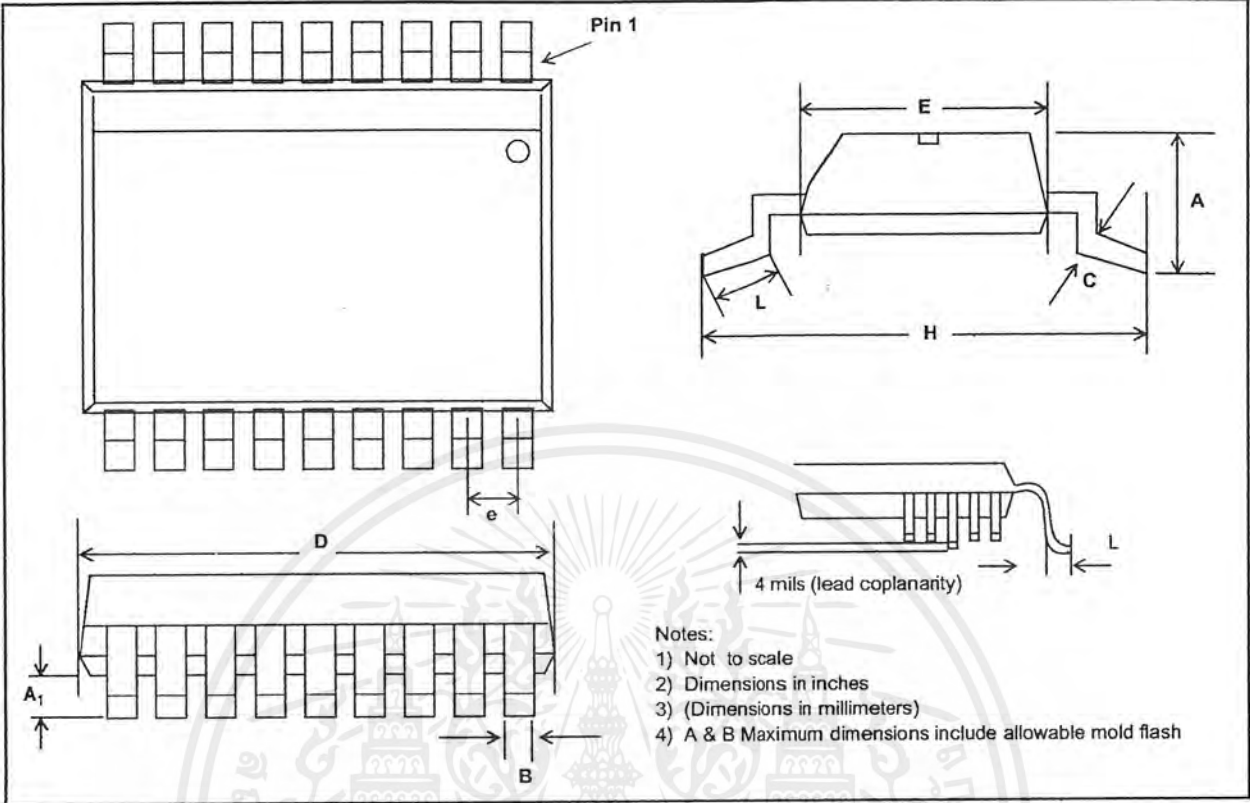


Notes:

- 1) Not to scale
- 2) Dimensions in inches
- 3) (Dimensions in millimeters)
- 4) Ref. JEDEC Standard M0-150/M0118 for 48 Pin
- 5) A & B Maximum dimensions include allowable mold flash

Dim	20-Pin		24-Pin		28-Pin		48-Pin	
	Min	Max	Min	Max	Min	Max	Min	Max
A		0.079 (2)		0.079 (2)		0.079 (2)	0.095 (2.41)	0.110 (2.79)
A ₁	0.002 (0.05)		0.002 (0.05)		0.002 (0.05)		0.008 (0.2)	0.016 (0.406)
B	0.0087 (0.22)	0.013 (0.33)	0.0087 (0.22)	0.013 (0.33)	0.0087 (0.22)	0.013 (0.33)	0.008 (0.2)	0.0135 (0.342)
C		0.008 (0.21)		0.008 (0.21)		0.008 (0.21)		0.010 (0.25)
D	0.27 (6.9)	0.295 (7.5)	0.31 (7.9)	0.33 (8.5)	0.39 (9.9)	0.42 (10.5)	0.62 (15.75)	0.63 (16.00)
E	0.2 (5.0)	0.22 (5.6)	0.2 (5.0)	0.22 (5.6)	0.2 (5.0)	0.22 (5.6)	0.291 (7.39)	0.299 (7.59)
e	0.025 BSC (0.635 BSC)		0.025 BSC (0.635 BSC)		0.025 BSC (0.635 BSC)		0.025 BSC (0.635 BSC)	
A ₂	0.065 (1.65)	0.073 (1.85)	0.065 (1.65)	0.073 (1.85)	0.065 (1.65)	0.073 (1.85)	0.089 (2.26)	0.099 (2.52)
H	0.29 (7.4)	0.32 (8.2)	0.29 (7.4)	0.32 (8.2)	0.29 (7.4)	0.32 (8.2)	0.395 (10.03)	0.42 (10.67)
L	0.022 (0.55)	0.037 (0.95)	0.022 (0.55)	0.037 (0.95)	0.022 (0.55)	0.037 (0.95)	0.02 (0.51)	0.04 (1.02)

Small Shrink Outline Package (SSOP) - N Suffix



DIM	16-Pin		18-Pin		20-Pin		24-Pin		28-Pin	
	Min	Max	Min	Max	Min	Max	Min	Max	Min	Max
A	0.093 (2.35)	0.104 (2.65)	0.093 (2.35)	0.104 (2.65)	0.093 (2.35)	0.104 (2.65)	0.093 (2.35)	0.104 (2.65)	0.093 (2.35)	0.104 (2.65)
A ₁	0.004 (0.10)	0.012 (0.30)	0.004 (0.10)	0.012 (0.30)	0.004 (0.10)	0.012 (0.30)	0.004 (0.10)	0.012 (0.30)	0.004 (0.10)	0.012 (0.30)
B	0.013 (0.33)	0.020 (0.51)	0.013 (0.33)	0.030 (0.51)	0.013 (0.33)	0.020 (0.51)	0.013 (0.33)	0.020 (0.51)	0.013 (0.33)	0.020 (0.51)
C	0.009 (0.231)	0.013 (0.318)	0.009 (0.231)	0.013 (0.318)	0.009 (0.231)	0.013 (0.318)	0.009 (0.231)	0.013 (0.318)	0.009 (0.231)	0.013 (0.318)
D	0.398 (10.1)	0.413 (10.5)	0.447 (11.35)	0.4625 (11.75)	0.496 (12.60)	0.512 (13.00)	0.5985 (15.2)	0.614 (15.6)	0.697 (17.7)	0.7125 (18.1)
E	0.291 (7.40)	0.299 (7.40)	0.291 (7.40)	0.299 (7.40)	0.291 (7.40)	0.299 (7.40)	0.291 (7.40)	0.299 (7.40)	0.291 (7.40)	0.299 (7.40)
e	0.050 BSC (1.27 BSC)		0.050 BSC (1.27 BSC)		0.050 BSC (1.27 BSC)		0.050 BSC (1.27 BSC)		0.050 BSC (1.27 BSC)	
H	0.394 (10.00)	0.419 (10.65)	0.394 (10.00)	0.419 (10.65)	0.394 (10.00)	0.419 (10.65)	0.394 (10.00)	0.419 (10.65)	0.394 (10.00)	0.419 (10.65)
L	0.016 (0.40)	0.050 (1.27)	0.016 (0.40)	0.050 (1.27)	0.016 (0.40)	0.050 (1.27)	0.016 (0.40)	0.050 (1.27)	0.016 (0.40)	0.050 (1.27)

Lead SOIC Package - S Suffix

NOTES: 1. Controlling dimensions in parenthesis () are in millimeters.
2. Converted inch dimensions are not necessarily exact.

MAXIM

Microprocessor Supervisory Circuits

General Description

The MAX690 Family of supervisory circuits reduce the complexity and number of components required for power supply monitoring and battery control functions in microprocessor systems. These include μ P reset and backup-battery switchover, watchdog timer, CMOS RAM write protection, and power-failure warning. The MAX690 Family significantly improves system reliability and accuracy compared to that obtainable with separate ICs or discrete components.

The MAX690, MAX692 and MAX694 are supplied in 8-pin packages and provide four functions:

- 1) A Reset output during power-up, power-down and brownout conditions.
- 2) Battery backup switching for CMOS RAM, CMOS microprocessor or other low power logic.
- 3) A Reset pulse if the optional watchdog timer has not been toggled within a specified time.
- 4) A 1.3V threshold detector for power fail warning, low battery detection, or to monitor a power supply other than +5V.

The MAX691, MAX693 and MAX695 are supplied in 16-pin packages and perform all MAX690/692/694 functions, plus:

- 1) Write protection of CMOS RAM or EEPROM.
- 2) Adjustable reset and watchdog timeout periods.
- 3) Separate outputs for indicating a watchdog timeout, backup-battery switchover, and low V_{CC} .

Applications

Computers
Controllers
Intelligent Instruments
Automotive Systems
Critical μ P Power Monitoring

Features

- ◆ Precision Voltage Monitor
4.65V in MAX690, MAX691, MAX694 and MAX695
4.40V in MAX692 and MAX693
- ◆ Power OK/Reset Time Delay – 50, 200ms, or adjustable
- ◆ Watchdog Timer – 100ms, 1.6 sec, or adjustable
- ◆ Minimum Component Count
- ◆ 1 μ A Standby Current
- ◆ Battery Backup Power Switching
- ◆ Onboard Gating of Chip Enable Signals
- ◆ Voltage Monitor for Power Fail or Low Battery Warning

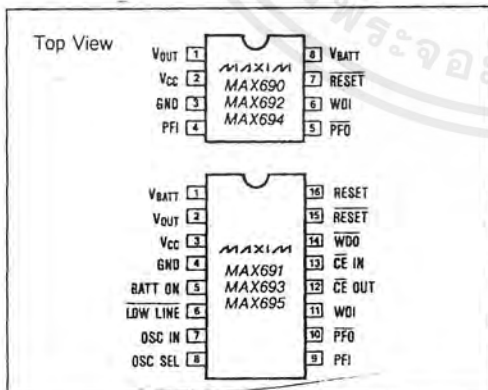
Ordering Information

PART	TEMP. RANGE	PIN-PACKAGE
MAX690CPA	0°C to +70°C	8 Lead Plastic DIP
MAX690C/D	0°C to +70°C	Dice*
MAX690EPA	-40°C to +85°C	8 Lead Plastic DIP
MAX690EJA	-40°C to +85°C	8 Lead CERDIP
MAX690MJA	-55°C to +125°C	8 Lead CERDIP
MAX691CPE	0°C to +70°C	16 Lead Plastic DIP
MAX691CWE	0°C to +70°C	16 Lead Wide SO
MAX69C/D	0°C to +70°C	Dice*
MAX691EPE	-40°C to +85°C	16 Lead Plastic DIP
MAX691EWE	-40°C to +85°C	16 Lead Wide SO
MAX691EJE	-40°C to +85°C	16 Lead CERDIP
MAX691MJE	-55°C to +125°C	16 Lead CERDIP

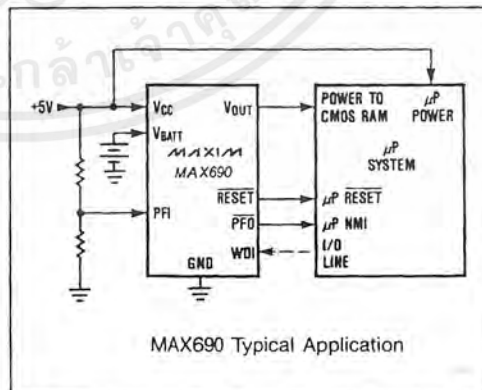
Ordering information continued on last page.

*Contact factory for dice specifications.

Pin Configuration



Typical Operating Circuit



Microprocessor Supervisory Circuits

ABSOLUTE MAXIMUM RATINGS

Terminal Voltage (with respect to GND)

V_{CC} -0.3V to 6.0V
 V_{BATT} -0.3V to 6.0V
 All Other Inputs (Note 1) .. -0.3V to ($V_{OUT} + 0.5V$)

Input Current

V_{CC} 200mA
 V_{BATT} 50mA
 GND 20mA

Output Current

V_{OUT} short circuit protected
 All Other Outputs 20mA

Rate-of-Rise, V_{BATT} , V_{CC} 100V/ μ s

Operating Temperature Range

C suffix 0°C to +70°C
 E suffix -40°C to +85°C
 M suffix -55°C to +125°C

Power Dissipation

8 Pin Plastic DIP
 (Derate 5mW/°C above +70°C) 400mW

8 Pin CERDIP
 (Derate 8mW/°C above +85°C) 500mW

16 Pin Plastic DIP
 (Derate 7mW/°C above +70°C) 600mW

16 Pin Small Outline
 (Derate 7mW/°C above +70°C) 600mW

16 Pin CERDIP
 (Derate 10mW/°C above +85°C) 600mW

Storage Temperature Range -65°C to +160°C

Lead Temperature (Soldering, 10 seconds) 300°C

Stresses above those listed under "Absolute Maximum Ratings" may cause permanent damage to the device. These are stress ratings only, and functional operation of the device at these or any other conditions above those indicated in the operational sections of the specifications is not implied. Exposure to absolute maximum ratings conditions for extended periods may affect device reliability.

ELECTRICAL CHARACTERISTICS

(V_{CC} = full operating range, V_{BATT} = 2.8V, T_A = 25°C, unless otherwise noted.)

PARAMETER	CONDITIONS	MIN	TYP	MAX	UNITS
BATTERY BACKUP SWITCHING					
Operating Voltage Range MAX690, MAX691, MAX694, MAX695 V_{CC} MAX690, MAX691, MAX694, MAX695 V_{BATT} MAX692, MAX693 V_{CC} MAX692, MAX693 V_{BATT}		4.75 2.0 4.5 2.0		5.5 4.25 5.5 4.0	V
V_{OUT} Output Voltage	$I_{OUT} = 1mA$ $I_{OUT} = 50mA$	$V_{CC} - 0.3$ $V_{CC} - 0.5$	$V_{CC} - 0.1$ $V_{CC} - 0.25$		V
V_{OUT} in Battery Backup Mode	$I_{OUT} = 250\mu A$, $V_{CC} = V_{BATT} - 0.2V$	$V_{BATT} - 0.1$	$V_{BATT} - 0.02$		V
Supply Current (excludes I_{OUT})	$I_{OUT} = 1mA$ $I_{OUT} = 50mA$		2 3.5	5 10	mA
Supply Current in Battery Backup Mode	$V_{CC} = 0V$, $V_{BATT} = 2.8V$		0.6	1	μA
Battery Standby Current (+ = Discharge, - = Charge)	$5.5V > V_{CC} > V_{BATT} + 1V$ $T_A = 25^\circ C$ T_A = Full Operating Range	-0.1 -1.0		+0.02 +0.02	μA
Battery Switchover Threshold $V_{CC} - V_{BATT}$	Power Up Power Down		70 50		mV
Battery Switchover Hysteresis			20		mV
BATT ON Output Voltage	$I_{SINK} = 3.2mA$			0.4	V
BATT ON Output Short Circuit Current	BATT ON = $V_{OUT} = 4.5V$ Sink Current BATT ON = 0V Source Current	0.5	25 1	25	mA μA
RESET AND WATCHDOG TIMER					
Reset Voltage Threshold	T_A = Full Operating Range MAX690, MAX691, MAX694, MAX695 MAX692, MAX693	4.5 4.25	4.65 4.4	4.75 4.5	V

Note 1: The input voltage limits on PFI and WDI may be exceeded provided the input current is limited to less than 10mA.

Microprocessor Supervisory Circuits

MAX690/91/92/93/94/95

ELECTRICAL CHARACTERISTICS (continued)

(V_{CC} = full operating range, $V_{BATT} = 2.8V$, $T_A = 25^\circ C$, unless otherwise noted.)

PARAMETER	CONDITIONS	MIN	TYP	MAX	UNITS
Reset Threshold Hysteresis			40		mV
Reset Timeout Delay (MAX690/91/92/93)	Figure 6. OSC SEL HIGH, $V_{CC} = 5V$	35	50	70	ms
Reset Timeout Delay (MAX694/95)	Figure 6. OSC SEL HIGH, $V_{CC} = 5V$	140	200	280	ms
Watchdog Timeout Period, Internal Oscillator	Long Period, $V_{CC} = 5V$	1.0	1.6	2.25	sec
	Short Period, $V_{CC} = 5V$	70	100	140	ms
Watchdog Timeout Period, External Clock	Long Period	3840		4097	Clock Cycles
	Short Period	768		1025	
Minimum WDI Input Pulse Width	$V_{IL} = 0.4$, $V_{IH} = 0.8V_{CC}$	200			ns
RESET and LOW LINE Output Voltage	$I_{SINK} = 1.6mA$, $V_{CC} = 4.25V$ $I_{SOURCE} = 1\mu A$, $V_{CC} = 5V$	3.5		0.4	V
RESET and WDO Output Voltage	$I_{SINK} = 1.6mA$ $I_{SOURCE} = 1\mu A$, $V_{CC} = 5V$	3.5		0.4	V
Output Short Circuit Current	RESET, RESET, WDO, LOW LINE	1	3	25	μA
WDI Input Threshold Logic Low Logic High	$V_{CC} = 5V$ (Note 2)			0.8	V
		3.5			
WDI Input current	$WDI = V_{OUT}$		20	50	μA
	$WDI = 0V$	-50	-15		
POWER FAIL DETECTOR					
PFI Input Threshold	$V_{CC} = +5V$, $T_A = \text{Full}$	1.2	1.3	1.4	V
PFI Input Current			± 0.01	± 25	nA
PFO Output Voltage	$I_{SINK} = 3.2mA$ $I_{SOURCE} = 1\mu A$	3.5		0.4	V
PFO Short Circuit Source Current	$PFI = V_{IH}$, $PFO = 0V$	1	3	25	μA
CHIP ENABLE GATING					
CE IN Thresholds	V_{IL} V_{IH}	3.0		0.8	V
CE IN Pullup Current			3		μA
CE OUT Output Voltage	$I_{SINK} = 3.2mA$ $I_{SOURCE} = 3.0mA$ $I_{SOURCE} = 1\mu A$, $V_{CC} = 0V$		$V_{OUT} = 1.5$ $V_{OUT} = 0.05$	0.4	V
CE Propagation Delay	$V_{CC} = 5V$		50	200	ns
OSCILLATOR					
OSC IN Input Current			± 2		μA
OSC SEL Input Pullup Current			5		μA
OSC IN Frequency Range	OSC SEL = 0V	0		250	kHz
OSC IN Frequency with External Capacitor	OSC SEL = 0V $C_{OSC} = 47pF$		4		kHz

Note 1: The input voltage limits on PFI and WDI may be exceeded provided the input current is limited to less than 10mA.

Note 2: WDI is guaranteed to be in the mid-level (inactive) state if WDI is floating and V_{CC} is in the operating voltage range. WDI is internally biased to 38% of V_{CC} with an impedance of approximately 125 kilohms.

Microprocessor Supervisory Circuits

Pin Description

NAME	PIN		FUNCTION
	MAX690/ 692/694	MAX691/ 693/695	
V _{CC}	2	3	The +5V input.
V _{BATT}	8	1	Backup battery input. Connect to Ground if a backup battery is not used.
V _{OUT}	1	2	The higher of V _{CC} or V _{BATT} is internally switched to V _{OUT} . Connect V _{OUT} to V _{CC} if V _{OUT} and V _{BATT} are not used. Connect a 0.1µF or larger bypass capacitor to V _{OUT} .
GND	3	4	0V Ground reference for all signals.
RESET	7	15	RESET goes low whenever V _{CC} falls below either the reset voltage threshold or the V _{BATT} input voltage. The reset threshold is typically 4.65V for the MAX690/691/694/695, and 4.4V for the MAX692 and MAX693. RESET remains low for 50ms after V _{CC} returns to 5V. (except 200ms in MAX694/695). RESET also goes low for 50ms if the Watchdog Timer is enabled but not serviced within its timeout period. The RESET pulse width can be adjusted as shown in Table 1.
WDI	6	11	The watchdog input, WDI, is a three level input. If WDI remains either high or low for longer than the watchdog timeout period, RESET pulses low and WDO goes low. The Watchdog Timer is disabled when WDI is left floating or is driven to mid-supply. The timer resets with each transition at the Watchdog Timer Input.
PFI	4	9	PFI is the non-inverting input to the Power Fail Comparator. When PFI is less than 1.3V, PFO goes low. Connect PFI to GND or V _{OUT} when not used. See Figure 1.
PFO	5	10	PFO is the output of the Power Fail Comparator. It goes low when PFI is less than 1.3V. The comparator is turned off and PFO goes low when V _{CC} is below V _{BATT} .
CE IN	—	13	The input to the CE gating circuit. Connect to GND or V _{OUT} if not used.
CE OUT	—	12	CE OUT goes low only when CE IN is low and V _{CC} is above the reset threshold (4.65V for MAX691 and MAX695, 4.4V for MAX693). See Figure 6.
BATT ON	—	5	BATT ON goes high when V _{OUT} is internally switched to the V _{BATT} input. It goes low when V _{OUT} is internally switched to V _{CC} . The output typically sinks 25mA and can directly drive the base of an external PNP transistor to increase the output current above the 50mA rating of V _{OUT} .
LOW LINE	—	6	LOW LINE goes low when V _{CC} falls below the reset threshold. It returns high as soon as V _{CC} rises above the reset threshold. See Figure 6, Reset Timing.
RESET	—	16	RESET is an active high output. It is the inverse of RESET.
OSC SEL	—	8	When OSC SEL is unconnected or driven high, the internal oscillator sets the reset time delay and watchdog timeout period. When OSC SEL is low, the external oscillator input, OSC IN, is enabled. OSC SEL has a 3µA internal pullup. See Table 1.
OSC IN	—	7	When OSC SEL is low, OSC IN can be driven by an external clock to adjust both the reset delay and the watchdog timeout period. The timing can also be adjusted by connecting an external capacitor to this pin. See Figure 8. When OSC SEL is high or floating, OSC IN selects between fast and slow Watchdog timeout periods.
WDO	—	14	The Watchdog Output, WDO, goes low if WDI remains either high or low for longer than the Watchdog timeout period. WDO is set high by the next transition at WDI. If WDI is unconnected or at mid-supply, WDO remains high. WDO also goes high when LOW LINE goes low.

Microprocessor Supervisory Circuits

Typical Applications

MAX691, MAX693 and MAX695

A typical connection for the MAX691/693/695 is shown in Figure 1. CMOS RAM is powered from V_{OUT} . V_{OUT} is internally connected to V_{CC} when 5V power is present, or to V_{BATT} when V_{CC} is less than the battery voltage. V_{OUT} can supply 50mA from V_{CC} , but if more current is required, an external PNP transistor can be added. When V_{CC} is higher than V_{BATT} , the BATT ON output goes low, providing 25mA of base drive for the external transistor. When V_{CC} is lower than V_{BATT} , an internal 200Ω MOSFET connects the backup battery to V_{OUT} . The quiescent current in the battery backup mode is 1μA maximum when V_{CC} is between 0V and V_{BATT} -700mV.

Reset Output

A voltage detector monitors V_{CC} and generates a RESET output to hold the microprocessor's Reset line low when V_{CC} is below 4.65V (4.4V for MAX693). An internal monostable holds RESET low for 50ms* after V_{CC} rises above 4.65V (4.4V for MAX693). This prevents repeated toggling of RESET even if the 5V power drops out and recovers with each power line cycle.

The crystal oscillator normally used to generate the clock for microprocessors takes several milliseconds to start. Since most microprocessors need several clock cycles to reset, RESET must be held low until the microprocessor clock oscillator has started. The

MAX690 Family power-up RESET pulse lasts 50ms* to allow for this oscillator start-up time. The manual reset switch and the 0.1μF capacitor connected to the reset bus can be omitted if manual reset is not needed. An inverted, active high, RESET output is also supplied.

Power Fail Detector

The MAX691/93/95 issues a non-maskable interrupt (NMI) to the microprocessor when a power failure occurs. The +5V power line is monitored via two external resistors connected to the Power Fail Input (PFI). When the voltage at PFI falls below 1.3V, the Power Fail Output (PFO) drives the processor's NMI input low. If a Power Fail threshold of 4.8V is chosen, the microprocessor will have the time when V_{CC} fails from 4.8V to 4.65V to save data into RAM. An earlier power fail warning can be generated if the unregulated DC input of the 5V regulator is available for monitoring.

RAM Write Protection

The MAX691/93/95 CE OUT line drives the Chip Select inputs of the CMOS RAM. CE OUT follows CE IN as long as V_{CC} is above the 4.65V (4.4V for MAX693) reset threshold. If V_{CC} falls below the reset threshold, CE OUT goes high, independent of the logic level at CE IN. This prevents the microprocessor from writing erroneous data into RAM during power-up, power-down, brownouts, and momentary power interruptions. The LOW LINE output goes low when V_{CC} falls below 4.65V (4.4V for MAX693).

*200ms for MAX695

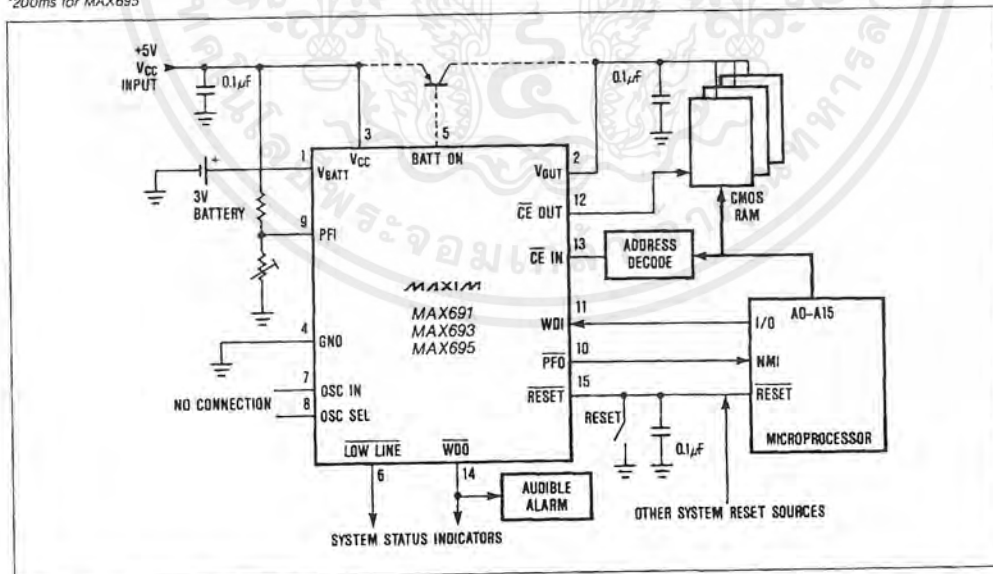


Figure 1. MAX691/693/695 Typical Application

Microprocessor Supervisory Circuits

Watchdog Timer

The microprocessor drives the WATCHDOG INPUT (WDI) with an I/O line. When OSC IN and OSC SEL are unconnected, the microprocessor must toggle the WDI pin once every 1.6 seconds to verify proper software execution. If a hardware or software failure occurs such that WDI not toggled, the MAX691/93 will issue a 50ms* RESET pulse after 1.6 seconds. This typically restarts the microprocessor's power-up routine. A new RESET pulse is issued every 1.6 seconds until WDI is again strobed.

The WATCHDOG OUTPUT (WDO) goes low if the watchdog timer is not serviced within its timeout period. Once WDO goes low it remains low until a transition occurs at WDI. The watchdog timer feature can be disabled by leaving WDI unconnected. OSC IN and OSC SEL also allow other watchdog timing options, as shown in Table 1 and Figure 8.

MAX690, MAX692 and MAX694

The 8 pin MAX690, MAX692 and MAX694 have most of the features of the MAX691, MAX693 and MAX695.

*200ms for MAX695

Figure 2 shows the MAX690/692/694 in a typical application. Operation is much the same as with the MAX691/693/695 (Figure 1) but in this case the Power Fail Input (PFI) monitors the unregulated input to the 7805 regulator. The MAX690/694 RESET output goes low when V_{CC} falls below 4.65V. The RESET output of the MAX692 goes low when V_{CC} drops below 4.4V.

The current consumption of the battery-backed-up power bus must be less than 50mA. The MAX690/692/694 does not have a BATT ON output to drive an external transistor. The MAX690/92/94 also does not include chip enable gating circuitry that is available on the MAX691/93/95. In many systems though, CE gating is not needed since a low input to the microprocessor RESET line prevents the processor from writing to RAM during power-up and power-down transients.

The MAX690/92/94 watchdog timer has a fixed 1.6 second timeout period. If WDI remains either low or high for more than 1.6 seconds, a RESET pulse is sent to the microprocessor. The watchdog timer is disabled if WDI is left floating.

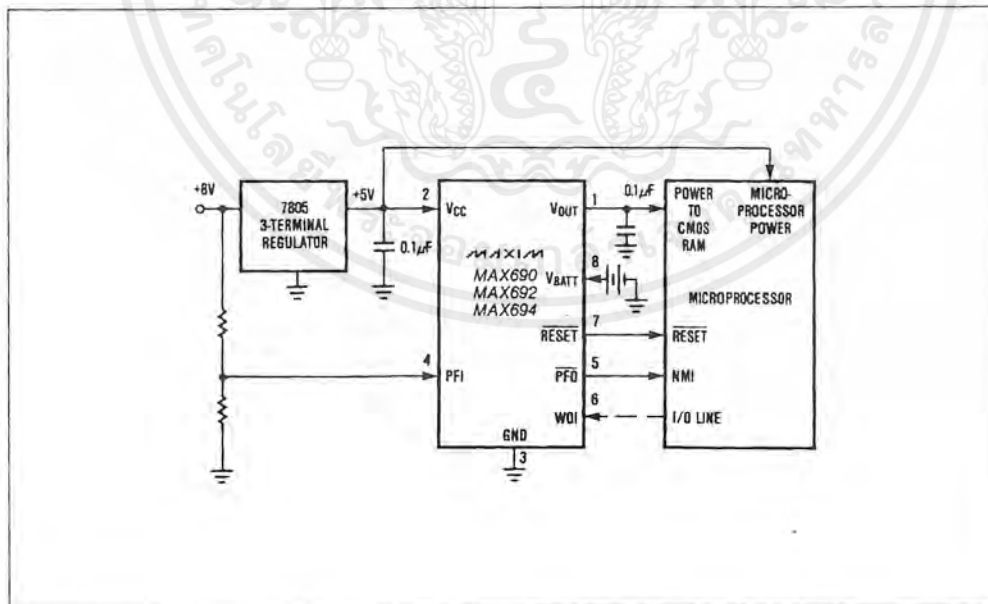


Figure 2. MAX690/692/694 Typical Application

Microprocessor Supervisory Circuits

MAX690/91/92/93/94/95

Detailed Description

Battery-Switchover and V_{OUT}

The battery switchover circuit compares V_{CC} to the V_{BATT} input, and connects V_{OUT} to whichever is higher. Switchover occurs when V_{CC} is 50mV greater than V_{BATT} as V_{CC} falls, and when V_{CC} is 70mV more than V_{BATT} as V_{CC} rises (see Figure 4). The switchover comparator has 20mV of hysteresis to prevent repeated, rapid switching if V_{CC} falls very slowly or remains nearly equal to the battery voltage.

When V_{CC} is higher than V_{BATT} , V_{CC} is internally switched to V_{OUT} via a low saturation PNP transistor. V_{OUT} has 50mA output current capability. Use an external PNP pass transistor in parallel with internal transistor if the output current requirement at V_{OUT} exceeds 50mA or if a lower V_{CC} - V_{OUT} voltage differential is desired. The BATT ON output (MAX691/693/695 only) can directly drive the base of the external transistor.

It should be noted that the MAX690/91/92/93/94/95 need only supply the average current drawn by the CMOS RAM if there is adequate filtering. Many RAM data sheets specify a 75mA maximum supply current, but this peak current spike lasts only 100ns. A 0.1 μ F bypass capacitor at V_{OUT} supplies the high instantaneous current, while V_{OUT} need only supply the average load current, which is much less. A capacitance of 0.1 μ F or greater must be connected to the V_{OUT} terminal to ensure stability.

A 200 Ω MOSFET connects the V_{BATT} input to V_{OUT}

during battery backup. This MOSFET has very low input-to-output differential (dropout voltage) at the low current levels required for battery backup of CMOS RAM or other low power CMOS circuitry. When V_{CC} equals V_{BATT} the supply current is typically 12 μ A. When V_{CC} is between 0V and (V_{BATT} - 700mV) the typical supply current is only 600nA typical, 1 μ A maximum.

The MAX690/MAX691/MAX694/MAX695 operate with battery voltages from 2.0V to 4.25V while the MAX692/MAX693 operate with battery voltages from 2.0V to 4.0V. High value capacitors can also be used for short-term memory backup. External circuitry is required to ensure that the capacitor voltage does not rise above the reset threshold, and that the charging resistor does not discharge the capacitor when in backup mode. The MAX691A and the MAX791 provide solutions requiring fewer external components.

A small charging current of typically 10nA (0.1 μ A max) flows out of the V_{BATT} terminal. This current varies with the amount of current that is drawn from V_{OUT} but its polarity is such that the backup battery is always slightly charged, and is never discharged while V_{CC} is in its operating voltage range. This extends the shelf life of the backup battery by compensating for its self-discharge current. Also note that this current poses no problem when lithium batteries are used for backup since the maximum charging current (0.1 μ A) is safe for even the smallest lithium cells.

If the battery-switchover section is not used, connect V_{BATT} to GND and connect V_{OUT} to V_{CC} . Table 2 shows the state of the inputs and output in the low power battery backup mode.

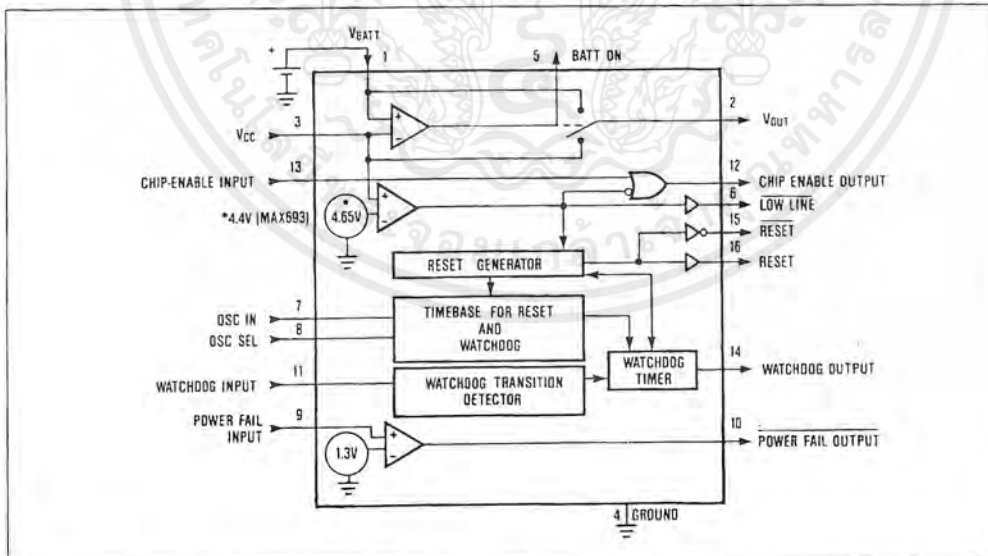


Figure 3. MAX691/693/695 Block Diagram

Microprocessor Supervisory Circuits

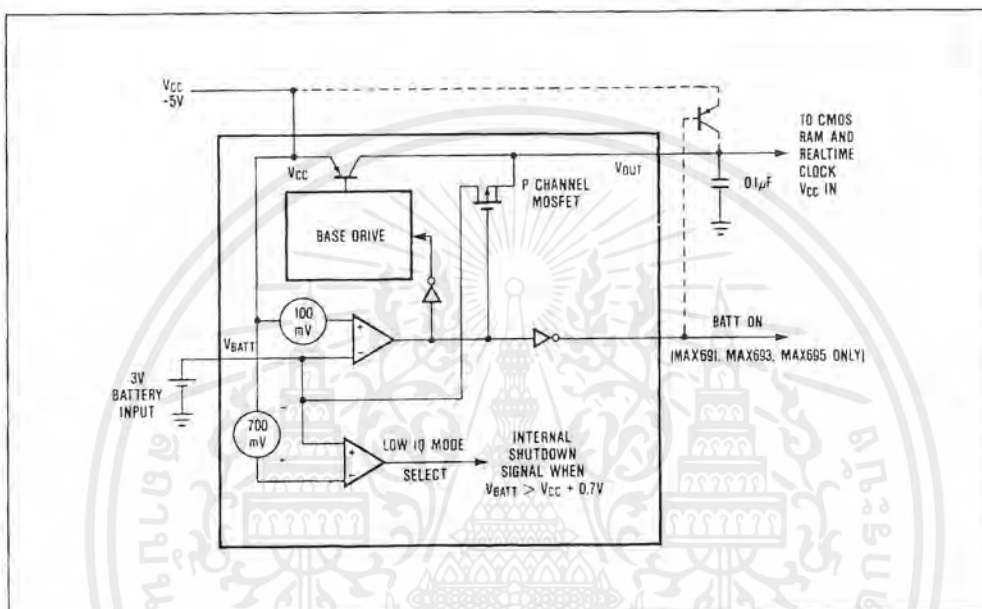


Figure 4. Battery-Switchover Block Diagram

Reset Output

RESET is an active low output which goes low whenever V_{CC} falls below 4.5V (MAX690/691/694/695) or 4.25V (MAX692/693). It will remain low until V_{CC} rises above 4.75V (MAX690/691/694/695) or 4.5V (MAX692/693) for 50 milliseconds*. See Figures 5 and 6.

The guaranteed minimum and maximum thresholds of the MAX690/691/694/695 are 4.5V and 4.75V, while the guaranteed thresholds of the MAX692/693 are 4.25V and 4.5V. The MAX690/691/694/695 is compatible with 5V supplies with a +10%, -5% tolerance while the MAX692/693 is compatible with $5V \pm 10\%$ supplies. The reset threshold comparator has approximately 50mV of hysteresis, with a nominal threshold of 4.65V in the MAX690/691/694/695, and 4.4V in the MAX692/693.

The response time of the reset voltage comparator is about 100µs. V_{CC} should be bypassed to ensure that glitches do not activate the **RESET** output.

RESET also goes low if the Watchdog Timer is enabled and WDI remains either high or low longer than the watchdog timeout period. **RESET** has an internal 3µA pullup, and can either connect to an open collector Reset bus or directly drive a CMOS gate without an external pullup resistor.

*200ms for MAX694 and MAX695

CE Gating and RAM Write Protection

The MAX691, MAX693 and MAX695 use two pins to control the Chip Enable or Write inputs of CMOS RAMs. When V_{CC} is +5V, **CE OUT** is a buffered replica of **CE IN**, with a 50ns propagation delay. If V_{CC} input falls below 4.65V (4.5V min, 4.75V max) an internal gate forces **CE OUT** high, independent of **CE IN**. The MAX693 **CE OUT** goes high whenever V_{CC} is below 4.4V (4.25V min, 4.5V max). The **CE** output of both devices is also forced high when V_{CC} is less than V_{BATT} . (See Figure 5.)

CE OUT typically drives the **CE**, **CS**, or **Write** input of battery backed up CMOS RAM. This ensures the integrity of the data in memory by preventing write operations when V_{CC} is at an invalid level. Similar protection of EEPROMs can be achieved by using the **CE OUT** to drive the Store or Write inputs of an EEPROM, EAROM, or NOVROM.

If the 50ns typical propagation delay of **CE OUT** is too long, connect **CE IN** to GND and use the resulting **CE OUT** to control a high speed external logic gate. A second alternative is to AND the **LOW LINE** output with the **CE** or **WR** signal. An external logic gate and the **RESET** output of the MAX690/692/694 can also be used for CMOS RAM write protection.

Microprocessor Supervisory Circuits

MAX690/91/92/93/94/95

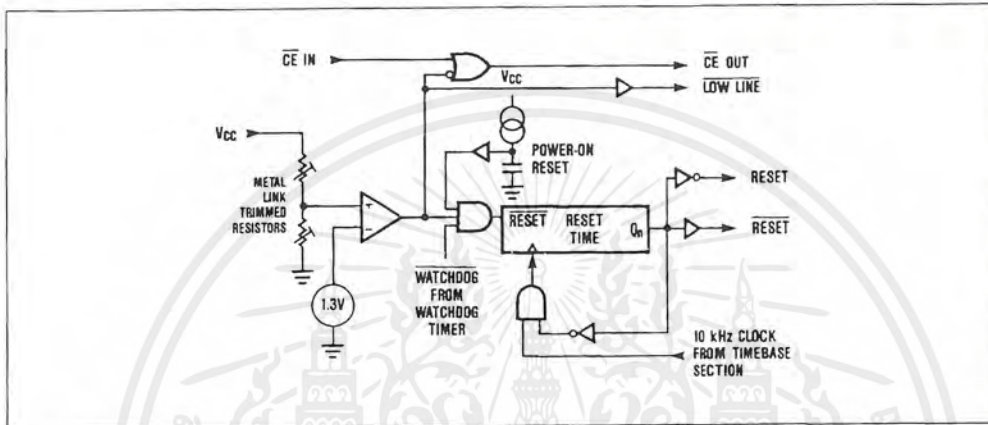


Figure 5. Reset Block Diagram

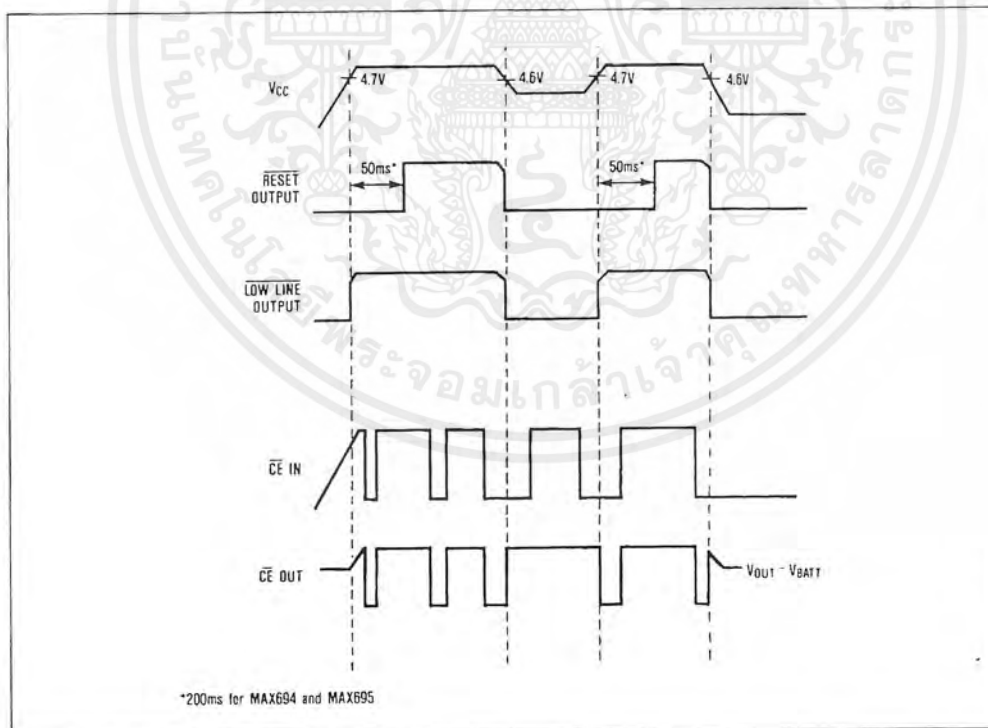


Figure 6. Reset Timing

Microprocessor Supervisory Circuits

1.3V Comparator and Power Fail Warning

The Power Fail Input (PFI) is compared to an internal 1.3V reference. The Power Fail Output (PFO) goes low when the voltage at PFI is less than 1.3V. Typically PFI is driven by an external voltage divider which senses either the unregulated DC input to the system's 5V regulator or the regulated 5V output. The voltage divider ratio can be chosen such that the voltage at PFI falls below 1.3V several milliseconds before the +5V supply falls below 4.75V. PFO is normally used to interrupt the microprocessor so that data can be stored in RAM before V_{CC} falls below 4.75V and the RESET output goes low (4.5V for MAX692/93).

The Power Fail Detector can also monitor the backup battery to warn of a low battery condition. To conserve battery power, the Power Fail Detector comparator is turned off and PFO is forced low when V_{CC} is lower than the V_{BATT} input voltage.

Watchdog Timer and Oscillator

The watchdog circuit monitors the activity of the microprocessor. If the microprocessor does not toggle the Watchdog Input (WDI) within the selected timeout period, a 50 millisecond* RESET pulse is generated. Since many systems cannot service the watchdog timer immediately after a reset, the MAX691/693/695 has a longer timeout period after a reset is issued. The normal timeout period becomes effective following the first transition of WDI after RESET has gone high. The watchdog timer is restarted at

the end of Reset, whether the Reset was caused by lack of activity on WDI or by V_{CC} falling below the reset threshold. If WDI remains either high or low, reset pulses will be issued every 1.6 seconds. The watchdog monitor can be deactivated by floating the Watchdog Input (WDI).

The Watchdog Output (WDO, MAX691/693/695 only) goes low if the watchdog timer "times out" and remains low until set high by the next transition on the watchdog input. WDO is also set high when V_{CC} goes below the reset threshold.

The watchdog timeout period is fixed at 1.6 seconds and the reset pulse width is fixed at 50ms* on the 8-pin MAX690, MAX692 and MAX694. The MAX691, MAX693 and MAX695 allow these times to be adjusted per Table 1. Figure 8 shows various oscillator configurations.

The internal oscillator is enabled when OSC SEL is floating. In this mode, OSC IN selects between the 1.6 second and 100ms watchdog timeout periods. In either case, immediately after a reset the timeout period is 1.6 seconds. This gives the microprocessor time to re-initialize the system. If OSC IN is low, then the 100ms watchdog period becomes effective after the first transition of WDI. The software should be written such that the I/O port driving WDI is left in its power-up reset state until the initialization routines are completed and the microprocessor is able to toggle WDI at the minimum watchdog timeout period of 70ms.

*200ms for MAX694

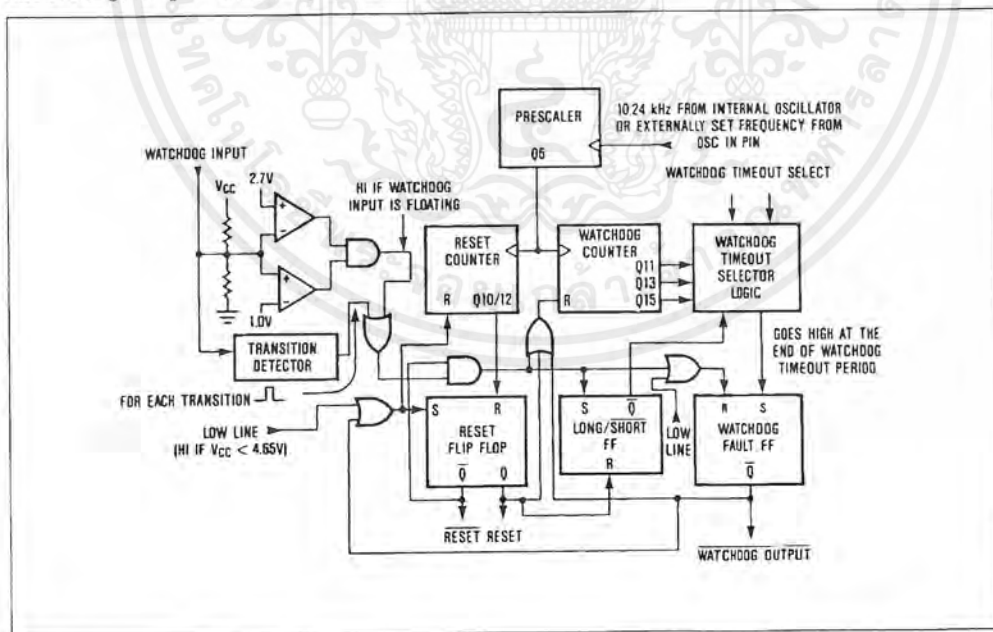


Figure 7. Watchdog Timer Block Diagram

Microprocessor Supervisory Circuits

MAX690/91/92/93/94/95

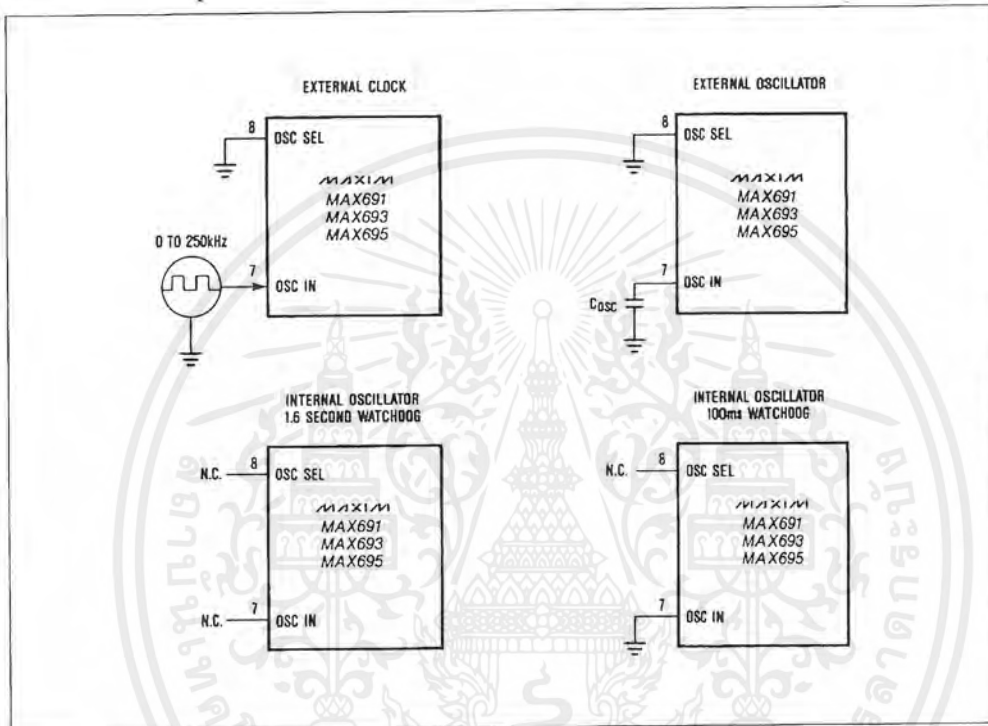


Figure 8. Oscillator Circuits

Table 1. MAX691, MAX693 and MAX695 Reset Pulse Width and Watchdog Timeout Selections

OSC SEL	OSC IN	Watchdog Timeout Period		Reset Timeout Period	
		Normal	Immediately After Reset	MAX691/93	MAX695
Low	External Clock Input	1024 clks	4096 clks	512 clks	2048 clks
Low	External Capacitor	$\frac{400\text{ms}}{47\text{pF}} \times C$	$\frac{1.6\text{sec}}{47\text{pF}} \times C$	$\frac{200\text{ms}}{47\text{pF}} \times C$	$\frac{800\text{ms}}{47\text{pF}} \times C$
Floating	Low	100ms	1.6 sec	50ms	200ms
Floating	Floating	1.6 sec	1.6 sec	50ms	200ms

Note 1: The MAX690/692/694 watchdog timeout period is fixed at 1.6 seconds nominal, the MAX690/692 reset pulse width is fixed at 50ms nominal and the MAX694 is 200ms nominal.

Note 2: When the MAX691 OSC SEL pin is low, OSC IN can be driven by an external clock signal or an external capacitor can be connected between OSC IN and GND. The nominal internal oscillator frequency is 6.55kHz. The nominal oscillator frequency with capacitor is:

$$F_{\text{osc}}(\text{Hz}) = \frac{120,000}{C(\text{pF})}$$

Note 3: See Electrical Characteristics Table for minimum and maximum timing values.

Microprocessor Supervisory Circuits

Application Hints

Other Uses of the Power Fail Detector

In Figure 9 the Power Fail Detector is used to initiate a system reset when V_{CC} falls to 4.95V. Since the threshold of the Power Fail Detector is not as accurate as the onboard Reset voltage detector, a trimpot must be used to adjust the voltage detection threshold. Both the PFO and RESET outputs have high sink current capability and only 10 μ A of source current drive. This allows the two outputs to be connected directly to each other in a "wired or" fashion.

The overvoltage detector circuit in Figure 10 resets the microprocessor whenever the nominal 5V V_{CC} is above 5.5V. The battery monitor circuit (Figure 11) shows the status of the memory backup battery. If desired, the CE OUT can be used to apply a test load to the battery. Since CE OUT is forced high during the battery backup mode, the test load will not be applied to the battery while it is in use, even if the microprocessor is not powered.

Adding Hysteresis to the Power Fail Comparator

Since the power fail comparator circuit is non-inverting, hysteresis can be added by connecting a resistor between the PFO output and the PFI input as shown in Figure 12. When PFO is low, resistor R3 sinks current from the summing junction at the PFI pin. When PFO is high, the series combination of R3 and R4 source current into the PFI summing junction.

Alternate Watchdog Input Drive Circuits

The Watchdog feature can be enabled and disabled under program control by driving WDI with a 3-state buffer (Figure 13). The drawback to this circuit is that a software fault may erroneously 3-state the buffer, thereby preventing the MAX690 from detecting that the microprocessor is no longer working. In most cases a better method is to extend the watchdog period rather than disabling the watchdog. See Figure 14. When the control input is high, the OSC SEL pin is low and the watchdog timeout is set by the external capacitor. A 0.01 μ F capacitor sets a watchdog timeout delay of 100 seconds. When the control input is low, the OSC SEL pin is driven high, selecting the internal oscillator. The 100ms or the 1.6 sec period is chosen, depending on which diode in Figure 14 is used.

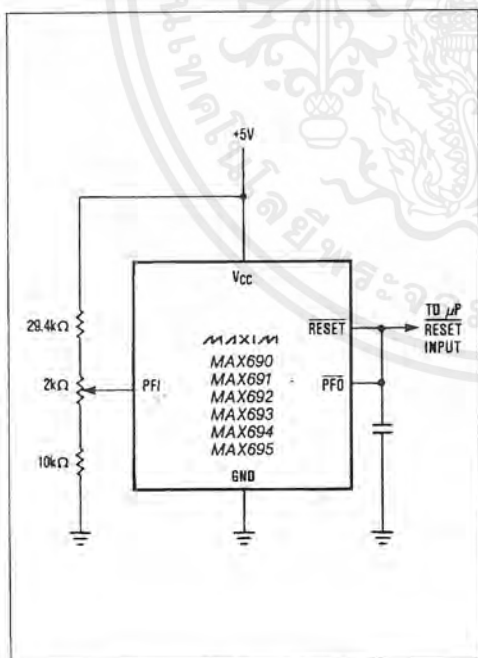


Figure 9. Externally Adjustable V_{CC} Reset Threshold

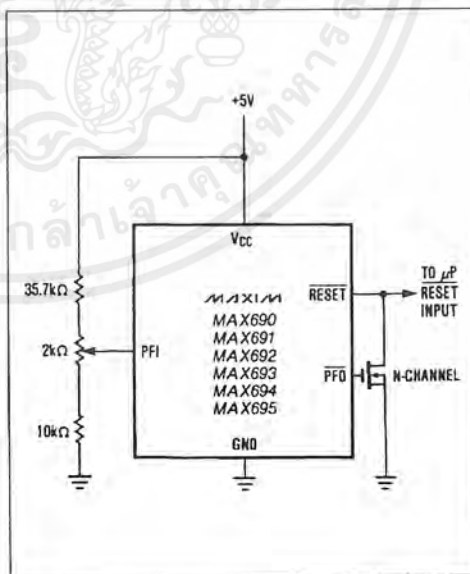


Figure 10. Reset on Overvoltage or Undervoltage

Microprocessor Supervisory Circuits

MAX690/91/92/93/94/95

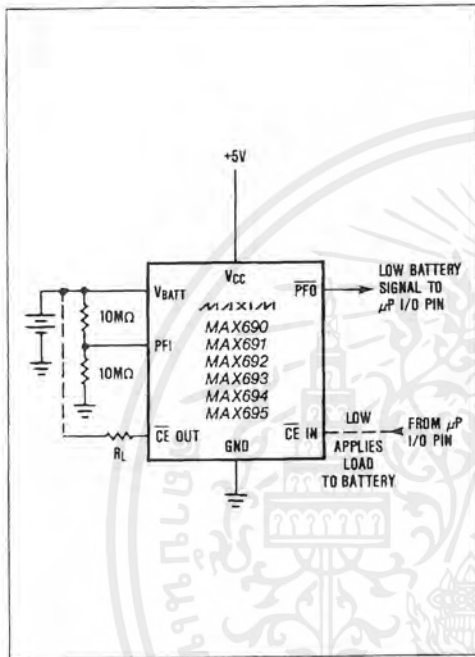


Figure 11. Backup Battery Monitor with Optional Test Load

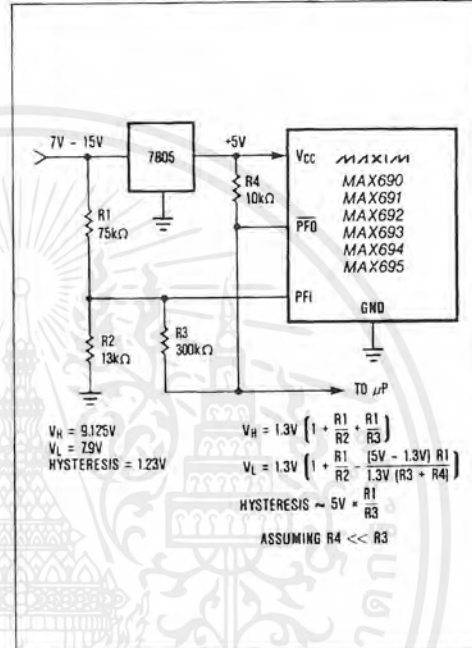


Figure 12. Adding Hysteresis to the Power Fail Voltage Comparator

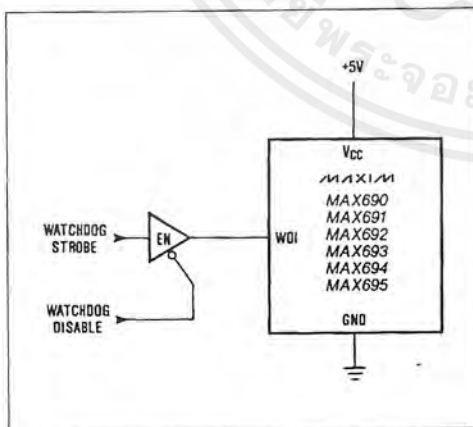


Figure 13. Disabling the Watchdog Under Program Control

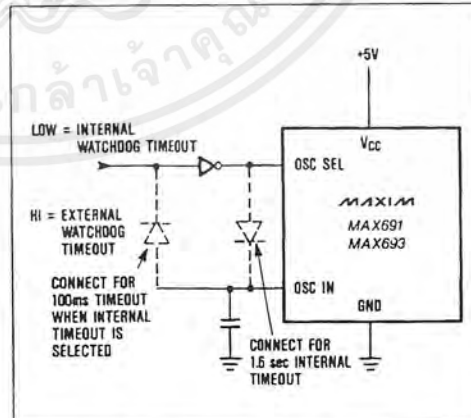


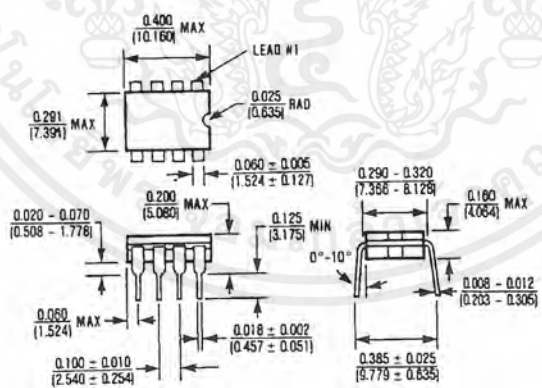
Figure 14. Selecting Internal or External Watchdog Timeout

Microprocessor Supervisory Circuits

Table 2. Input and Output Status In Battery Backup Mode

V _{BATT} , V _{OUT}	V _{BATT} is connected to V _{OUT} via internal MOSFET.
RESET	Logic low
RESET	Logic high. The open circuit output voltage is equal to V _{OUT} .
LOW LINE	Logic low
BATT ON	Logic high
WDI	WDI is internally disconnected from its internal pullup and does not source or sink current as long as its input voltage is between GND and V _{OUT} . The input voltage does not affect supply current.
WDO	Logic high
PFI	The Power Fail Comparator is turned off and the Power Fail Input voltage has no effect on the Power Fail Output.
PFO	Logic low
CE IN	CE IN is internally disconnected from its internal pullup and does not source or sink current as long as its input voltage is between GND and V _{OUT} . The input voltage does not affect supply current.
CE OUT	Logic high
OSC IN	OSC IN is ignored.
OSC SEL	OSC SEL is ignored.
V _{CC}	Approximately 12μA is drawn from the V _{BATT} input when V _{CC} is between V _{BATT} + 100mV and V _{BATT} - 700mV. The supply current is 1μA maximum when V _{CC} is less than V _{BATT} - 700mV.

Package Information

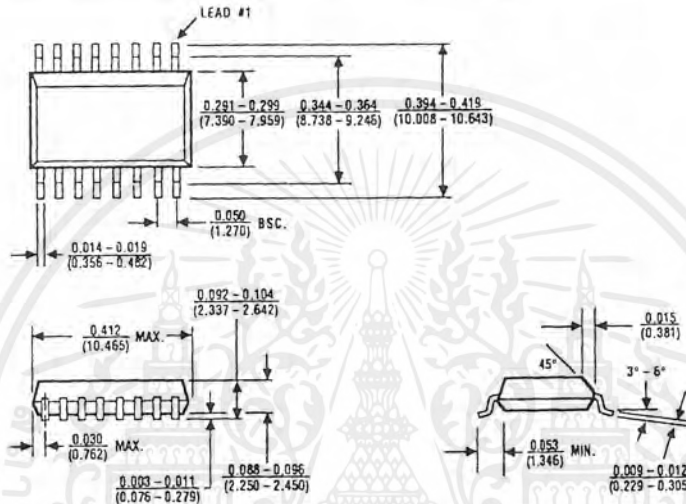


8 Lead Cerdip (JA)
 $\theta_{JA} = 125^{\circ}\text{C/W}$
 $\theta_{JC} = 55^{\circ}\text{C/W}$

Microprocessor Supervisory Circuits

Package Information (continued)

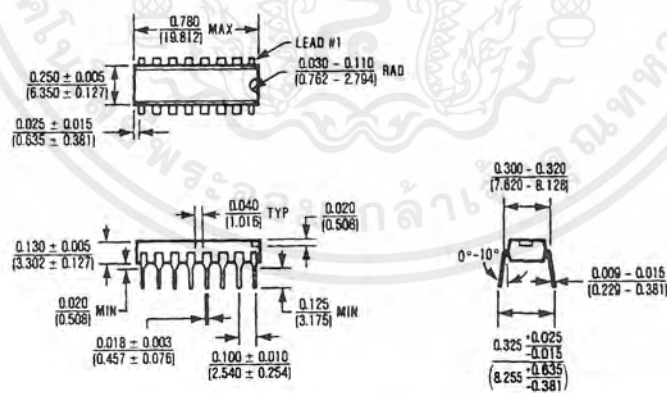
MAX690/91/92/93/94/95



16 Lead Small Outline, Wide (WE)

$\theta_{JA} = 105^{\circ}\text{C/W}$

$\theta_{JC} = 60^{\circ}\text{C/W}$



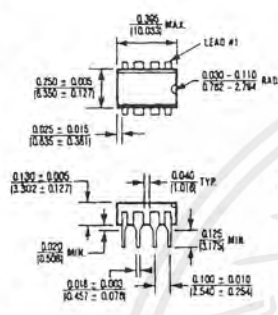
16 Lead Plastic DIP (PE)

$\theta_{JA} = 135^{\circ}\text{C/W}$

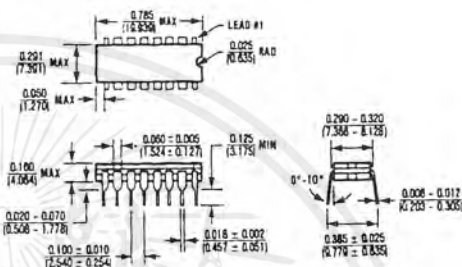
$\theta_{JC} = 65^{\circ}\text{C/W}$

Microprocessor Supervisory Circuits

Package Information (continued)



8 Lead Plastic DIP (PA)
 $\theta_{JA} = 160^{\circ}\text{C/W}$
 $\theta_{JC} = 75^{\circ}\text{C/W}$

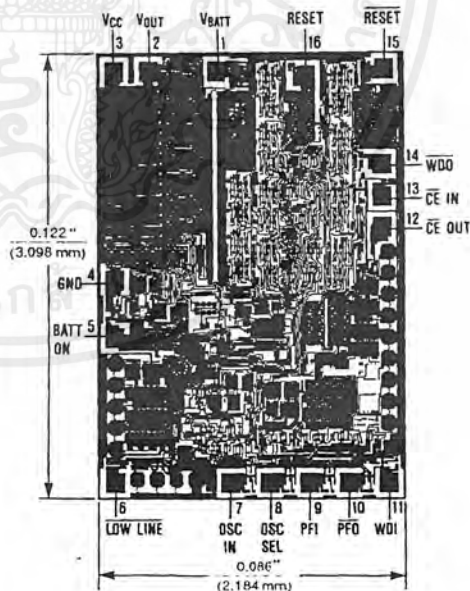


16 Lead Cerdip (JE)
 $\theta_{JA} = 100^{\circ}\text{C/W}$
 $\theta_{JC} = 50^{\circ}\text{C/W}$

Ordering Information (continued)

Chip Topography

PART	TEMP. RANGE	PIN-PACKAGE
MAX692C/D	0°C to +70°C	Dice
MAX692CPA	0°C to +70°C	8 Lead Plastic DIP
MAX692EPA	-40°C to +85°C	8 Lead Plastic DIP
MAX692EJA	-40°C to +85°C	8 Lead Cerdip
MAX692MJA	-55°C to +125°C	8 Lead Cerdip
MAX693C/D	0°C to +70°C	Dice
MAX693CPE	0°C to +70°C	16 Lead Plastic DIP
MAX693CWE	0°C to +70°C	16 Lead Wide SO
MAX693EPE	-40°C to +85°C	16 Lead Plastic DIP
MAX693EJE	-40°C to +85°C	16 Lead Cerdip
MAX693EWE	-40°C to +85°C	16 Lead Wide SO
MAX693MJE	-55°C to +125°C	16 Lead Cerdip
MAX694C/D	0°C to +70°C	Dice
MAX694CPA	0°C to +70°C	8 Lead Plastic DIP
MAX694EPA	-40°C to +85°C	8 Lead Plastic DIP
MAX694EJA	-40°C to +85°C	8 Lead Cerdip
MAX694MJA	-55°C to +125°C	8 Lead Cerdip
MAX695C/D	0°C to +70°C	Dice
MAX695CPE	0°C to +70°C	16 Lead Plastic DIP
MAX695CWE	0°C to +70°C	16 Lead Wide SO
MAX695EPE	-40°C to +85°C	16 Lead Plastic DIP
MAX695EJE	-40°C to +85°C	16 Lead Cerdip
MAX695EWE	-40°C to +85°C	16 Lead Wide SO
MAX695MJE	-55°C to +125°C	16 Lead Cerdip



Maxim cannot assume responsibility for use of any circuitry other than circuitry entirely embodied in a Maxim product. No circuit patent licenses are implied. Maxim reserves the right to change the circuitry and specifications without notice at any time.

16 Maxim Integrated Products, 120 San Gabriel Drive, Sunnyvale, CA 94086 (408) 737-7600

© 1995 Maxim Integrated Products

Printed USA

MAXIM is a registered trademark of Maxim Integrated Products.

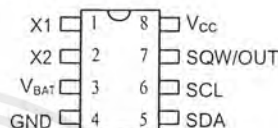
FEATURES

- Real time clock counts seconds, minutes, hours, date of the month, month, day of the week, and year with leap year compensation valid up to 2100
- 56 byte nonvolatile RAM for data storage
- 2-wire serial interface
- Programmable squarewave output signal
- Automatic power-fail detect and switch circuitry
- Consumes less than 500 nA in battery backup mode with oscillator running
- Optional industrial temperature range -40°C to +85°C
- Available in 8-pin DIP or SOIC
- Recognized by Underwriters Laboratory

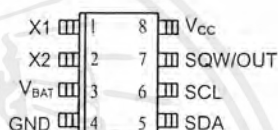
ORDERING INFORMATION

DS1307	8-Pin DIP
DS1307Z	8-Pin SOIC (150 mil)
DS1307N	8-Pin DIP (Industrial)
DS1307ZN	8-Pin SOIC (Industrial)

PIN ASSIGNMENT



DS1307 8-Pin DIP (300 mil)



DS1307Z 8-Pin SOIC (150 mil)

PIN DESCRIPTION

VCC	- Primary Power Supply
X1, X2	- 32.768 kHz Crystal Connection
V _{BAT}	- +3V Battery Input
GND	- Ground
SDA	- Serial Data
SCL	- Serial Clock
SQW/OUT	- Square wave/Output Driver

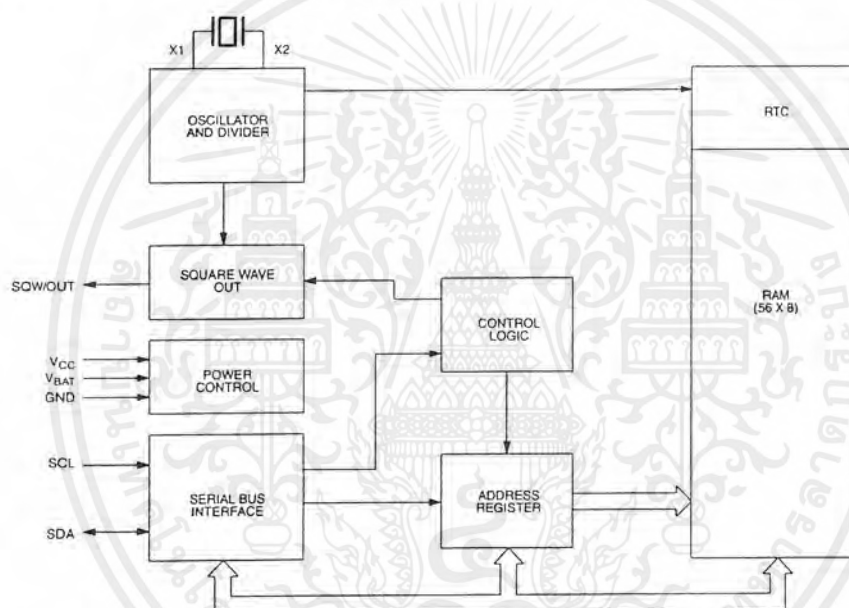
DESCRIPTION

The DS1307 Serial Real Time Clock is a low power, full BCD clock/calendar plus 56 bytes of nonvolatile SRAM. Address and data are transferred serially via a 2-wire bi-directional bus. The clock/calendar provides seconds, minutes, hours, day, date, month, and year information. The end of the month date is automatically adjusted for months with less than 31 days, including corrections for leap year. The clock operates in either the 24-hour or 12-hour format with AM/PM indicator. The DS1307 has a built-in power sense circuit which detects power failures and automatically switches to the battery supply.

OPERATION

The DS1307 operates as a slave device on the serial bus. Access is obtained by implementing a START condition and providing a device identification code followed by a register address. Subsequent registers can be accessed sequentially until a STOP condition is executed. When V_{CC} falls below $1.25 \times V_{BAT}$ the device terminates an access in progress and resets the device address counter. Inputs to the device will not be recognized at this time to prevent erroneous data from being written to the device from an out of tolerance system. When V_{CC} falls below V_{BAT} the device switches into a low current battery backup mode. Upon power up, the device switches from battery to V_{CC} when V_{CC} is greater than $V_{BAT} + 0.2V$ and recognizes inputs when V_{CC} is greater than $1.25 \times V_{BAT}$. The block diagram in Figure 1 shows the main elements of the Serial Real Time Clock.

DS1307 BLOCK DIAGRAM Figure 1



SIGNAL DESCRIPTIONS

V_{CC} , GND - DC power is provided to the device on these pins. V_{CC} is the +5 volt input. When 5 volts is applied within normal limits, the device is fully accessible and data can be written and read. When a 3-volt battery is connected to the device and V_{CC} is below $1.25 \times V_{BAT}$, reads and writes are inhibited. However, the Timekeeping function continues unaffected by the lower input voltage. As V_{CC} falls below V_{BAT} the RAM and timekeeper are switched over to the external power supply (nominal 3.0V DC) at V_{BAT} .

V_{BAT} - Battery input for any standard 3-volt lithium cell or other energy source. Battery voltage must be held between 2.0 and 3.5 volts for proper operation. The nominal write protect trip point voltage at which access to the real time clock and user RAM is denied is set by the internal circuitry as $1.25 \times V_{BAT}$ nominal. A lithium battery with 48 mAh or greater will back up the DS1307 for more than 10 years in the absence of power at 25 degrees C.

SCL (Serial Clock Input) - SCL is used to synchronize data movement on the serial interface.

SDA (Serial Data Input/Output) - SDA is the input/output pin for the 2-wire serial interface. The SDA pin is open drain which requires an external pullup resistor.

SQW/OUT (Square Wave/ Output Driver) - When enabled, the SQWE bit set to 1, the SQW/OUT pin outputs one of four square wave frequencies (1 Hz, 4 kHz, 8 kHz, 32 kHz). The SQW/OUT pin is open drain which requires an external pullup resistor. SQW/OUT will operate with either Vcc or Vbat applied.

X1, X2 - Connections for a standard 32.768 kHz quartz crystal. The internal oscillator circuitry is designed for operation with a crystal having a specified load capacitance (CL) of 12.5 pF.

For more information on crystal selection and crystal layout considerations, please consult Application Note 58, “Crystal Considerations with Dallas Real Time Clocks.” The DS1307 can also be driven by an external 32.768 kHz oscillator. In this configuration, the X1 pin is connected to the external oscillator signal and the X2 pin is floated.

Please review Application Note 95, “Interfacing the DS1307 with a 8051-Compatible Microcontroller” for additional information.

RTC AND RAM ADDRESS MAP

The address map for the RTC and RAM registers of the DS1307 is shown in Figure 2. The real time clock registers are located in address locations 00h to 07h. The RAM registers are located in address locations 08h to 3Fh. During a multi-byte access, when the address pointer reaches 3Fh, the end of RAM space, it wraps around to location 00h, the beginning of the clock space.

DS1307 ADDRESS MAP Figure 2

00H	SECONDS
	MINUTES
	HOURS
	DAY
	DATE
	MONTH
	YEAR
07H	CONTROL
08H	RAM 56 x 8
3FH	

CLOCK AND CALENDAR

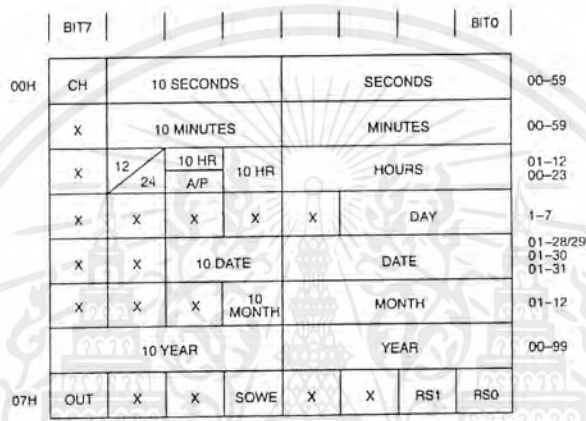
The time and calendar information is obtained by reading the appropriate register bytes. The real time clock registers are illustrated in Figure 3. The time and calendar are set or initialized by writing the appropriate register bytes. The contents of the time and calendar registers are in the Binary-Coded Decimal (BCD) format. Bit 7 of Register 0 is the Clock Halt (CH) bit. When this bit is set to a 1, the oscillator is disabled. When cleared to a 0, the oscillator is enabled.

Please note that the initial power on state of all registers is not defined. Therefore it is important to enable the oscillator (CH bit=0) during initial configuration.

The DS1307 can be run in either 12-hour or 24-hour mode. Bit 6 of the hours register is defined as the 12- or 24-hour mode select bit. When high, the 12-hour mode is selected. In the 12-hour mode, bit 5 is the AM/PM bit with logic high being PM. In the 24-hour mode, bit 5 is the second 10 hour bit (20-23 hours).

On a 2-wire START, the current time is transferred to a second set of registers. The time information is read from these secondary registers, while the clock may continue to run. This eliminates the need to re-read the registers in case of an update of the main registers during a read.

DS1307 TIMEKEEPER REGISTERS Figure 3



CONTROL REGISTER

The DS1307 Control Register is used to control the operation of the SQW/OUT pin.

BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
OUT	X	X	SQWE	X	X	RS1	RS0

OUT (Output control): This bit controls the output level of the SQW/OUT pin when the square wave output is disabled. If SQWE=0, the logic level on the SQW/OUT pin is 1 if OUT=1 and is 0 if OUT=0.

SQWE (Square Wave Enable): This bit, when set to a logic 1, will enable the oscillator output. The frequency of the square wave output depends upon the value of the RS0 and RS1 bits.

RS (Rate Select): These bits control the frequency of the square wave output when the square wave output has been enabled. Table 1 lists the square wave frequencies that can be selected with the RS bits.

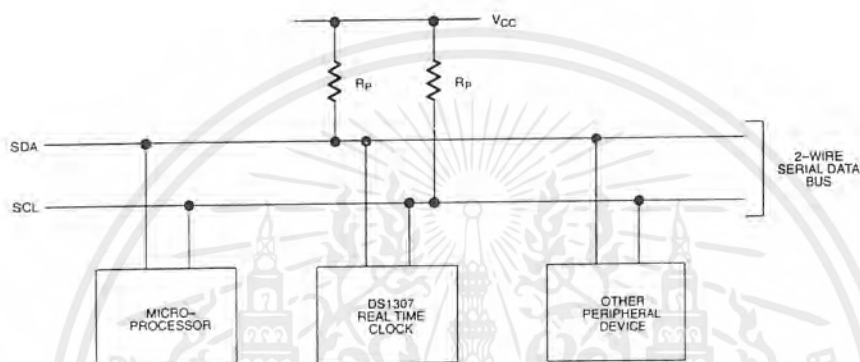
SQUAREWAVE OUTPUT FREQUENCY Table 1

RS1	RS0	SQW OUTPUT FREQUENCY
0	0	1 Hz
0	1	4.096 kHz
1	0	8.192 kHz
1	1	32.768 kHz

2-WIRE SERIAL DATA BUS

The DS1307 supports a bi-directional 2-wire bus and data transmission protocol. A device that sends data onto the bus is defined as a transmitter and a device receiving data as a receiver. The device that controls the message is called a master. The devices that are controlled by the master are referred to as slaves. The bus must be controlled by a master device which generates the serial clock (SCL), controls the bus access, and generates the START and STOP conditions. The DS1307 operates as a slave on the 2-wire bus. A typical bus configuration using this 2-wire protocol is shown in Figure 4.

TYPICAL 2-WIRE BUS CONFIGURATION Figure 4



Figures 5, 6, and 7 detail how data is transferred on the 2-wire bus.

- Data transfer may be initiated only when the bus is not busy.
- During data transfer, the data line must remain stable whenever the clock line is HIGH. Changes in the data line while the clock line is high will be interpreted as control signals.

Accordingly, the following bus conditions have been defined:

Bus not busy: Both data and clock lines remain HIGH.

Start data transfer: A change in the state of the data line, from HIGH to LOW, while the clock is HIGH, defines a START condition.

Stop data transfer: A change in the state of the data line, from LOW to HIGH, while the clock line is HIGH, defines the STOP condition.

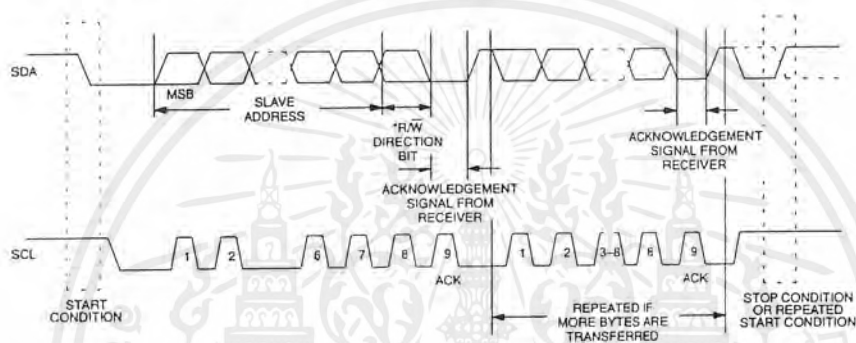
Data valid: The state of the data line represents valid data when, after a START condition, the data line is stable for the duration of the HIGH period of the clock signal. The data on the line must be changed during the LOW period of the clock signal. There is one clock pulse per bit of data.

Each data transfer is initiated with a START condition and terminated with a STOP condition. The number of data bytes transferred between START and STOP conditions is not limited, and is determined by the master device. The information is transferred byte-wise and each receiver acknowledges with a ninth bit. Within the 2-wire bus specifications a regular mode (100 kHz clock rate) and a fast mode (400 kHz clock rate) are defined. The DS1307 operates in the regular mode (100 kHz) only.

Acknowledge: Each receiving device, when addressed, is obliged to generate an acknowledge after the reception of each byte. The master device must generate an extra clock pulse which is associated with this acknowledge bit.

A device that acknowledges must pull down the SDA line during the acknowledge clock pulse in such a way that the SDA line is stable LOW during the HIGH period of the acknowledge related clock pulse. Of course, setup and hold times must be taken into account. A master must signal an end of data to the slave by not generating an acknowledge bit on the last byte that has been clocked out of the slave. In this case, the slave must leave the data line HIGH to enable the master to generate the STOP condition.

DATA TRANSFER ON 2-WIRE SERIAL BUS Figure 5



Depending upon the state of the R/W bit, two types of data transfer are possible:

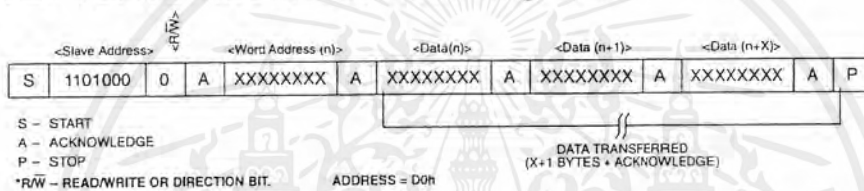
1. **Data transfer from a master transmitter to a slave receiver.** The first byte transmitted by the master is the slave address. Next follows a number of data bytes. The slave returns an acknowledge bit after each received byte. Data is transferred with the most significant bit (MSB) first.
2. **Data transfer from a slave transmitter to a master receiver.** The first byte (the slave address) is transmitted by the master. The slave then returns an acknowledge bit. This is followed by the slave transmitting a number of data bytes. The master returns an acknowledge bit after all received bytes other than the last byte. At the end of the last received byte, a 'not acknowledge' is returned.

The master device generates all of the serial clock pulses and the START and STOP conditions. A transfer is ended with a STOP condition or with a repeated START condition. Since a repeated START condition is also the beginning of the next serial transfer, the bus will not be released. Data is transferred with the most significant bit (MSB) first.

The DS1307 may operate in the following two modes:

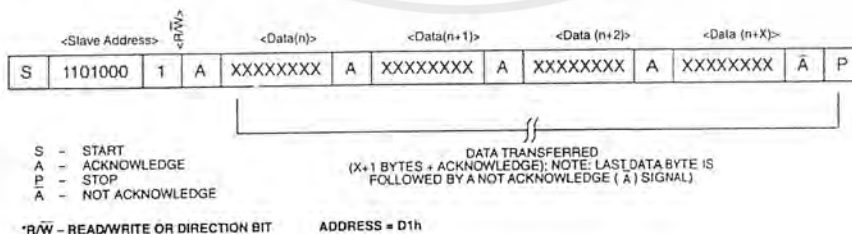
1. **Slave receiver mode (DS1307 write mode):** Serial data and clock are received through SDA and SCL. After each byte is received an acknowledge bit is transmitted. START and STOP conditions are recognized as the beginning and end of a serial transfer. Address recognition is performed by hardware after reception of the slave address and *direction bit (See Figure 6). The address byte is the first byte received after the start condition is generated by the master. The address byte contains the 7 bit DS1307 address, which is 1101000, followed by the *direction bit ($R/\bar{W}$) which, for a write, is a 0. After receiving and decoding the address byte the device outputs an acknowledge on the SDA line. After the DS1307 acknowledges the slave address + write bit, the master transmits a register address to the DS1307. This will set the register pointer on the DS1307. The master will then begin transmitting each byte of data with the DS1307 acknowledging each byte received. The master will generate a stop condition to terminate the data write.

DATA WRITE - SLAVE RECEIVER MODE Figure 6



2. **Slave transmitter mode (DS1307 read mode):** The first byte is received and handled as in the slave receiver mode. However, in this mode, the *direction bit will indicate that the transfer direction is reversed. Serial data is transmitted on SDA by the DS1307 while the serial clock is input on SCL. START and STOP conditions are recognized as the beginning and end of a serial transfer (See Figure 7). The address byte is the first byte received after the start condition is generated by the master. The address byte contains the 7-bit DS1307 address, which is 1101000, followed by the *direction bit ($R/\bar{W}$) which, for a read, is a 1. After receiving and decoding the address byte the device inputs an acknowledge on the SDA line. The DS1307 then begins to transmit data starting with the register address pointed to by the register pointer. If the register pointer is not written to before the initiation of a read mode the first address that is read is the last one stored in the register pointer. The DS1307 must receive a Not Acknowledge to end a read.

DATA READ - SLAVE TRANSMITTER MODE Figure 7



ABSOLUTE MAXIMUM RATINGS*

Voltage on Any Pin Relative to Ground	-0.5V to +7.0V
Operating Temperature	0°C to 70°C (-40°C to 85°C for industrial)
Storage Temperature	-55°C to +125°C
Soldering Temperature	260°C for 10 seconds DIP See JPC/JEDEC Standard J-STD-020A for Surface Mount Devices

* This is a stress rating only and functional operation of the device at these or any other conditions above those indicated in the operation sections of this specification is not implied. Exposure to absolute maximum rating conditions for extended periods of time may affect reliability.

RECOMMENDED DC OPERATING CONDITIONS

(0°C to 70°C or -40°C to +85°C)

PARAMETER	SYMBOL	MIN	TYP	MAX	UNITS	NOTES
Supply Voltage	V_{CC}	4.5	5.0	5.5	V	1
Logic 1	V_{IH}	2.2		$V_{CC}+0.3$	V	1
Logic 0	V_{IL}	-0.3		+0.8	V	1
V_{BAT} Battery Voltage	V_{BAT}	2.0		3.5	V	1

DC ELECTRICAL CHARACTERISTICS(0°C to 70°C or -40°C to +85°C; $V_{CC}=4.5V$ to 5.5V)

PARAMETER	SYMBOL	MIN	TYP	MAX	UNITS	NOTES
Input Leakage	I_{LI}			1	μA	10
I/O Leakage	I_{LO}			1	μA	11
Logic 0 Output	V_{OL}			0.4	V	2
Active Supply Current	I_{CCA}			1.5	mA	9
Standby Current	I_{CCS}			200	μA	3
Battery Current (OSC ON); SQW/OUT OFF	I_{BAT1}		300	500	nA	4
Battery Current (OSC ON); SQW/OUT ON (32 kHz)	I_{BAT2}		480	800	nA	4

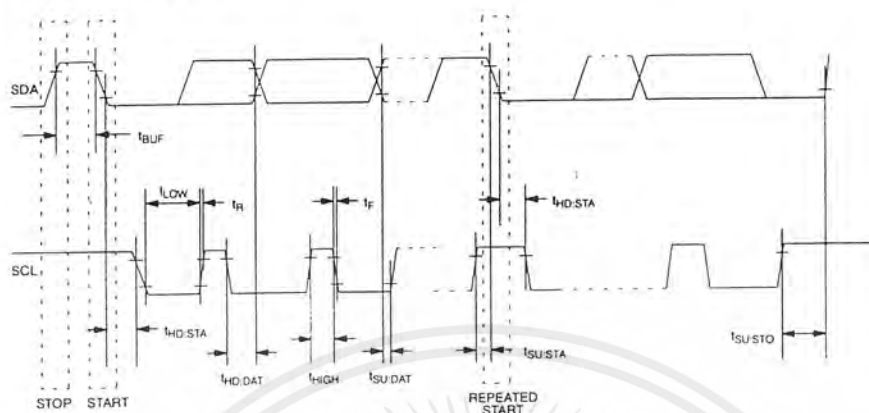
AC ELECTRICAL CHARACTERISTICS(0°C to 70°C or -40°C to +85°C; $V_{CC}=4.5V$ to 5.5V)

PARAMETER	SYMBOL	MIN	TYP	MAX	UNITS	NOTES
SCL Clock Frequency	f_{SCL}	0		100	kHz	
Bus Free Time Between a STOP and START Condition	t_{BUF}	4.7			μs	
Hold Time (Repeated) START Condition	$t_{HD:STA}$	4.0			μs	5
LOW Period of SCL Clock	t_{LOW}	4.7			μs	
HIGH Period of SCL Clock	t_{HIGH}	4.0			μs	
Set-up Time for a Repeated START Condition	$t_{SU:STA}$	4.7			μs	
Data Hold Time	$t_{HD:DAT}$	0			μs	6, 7
Data Set-up Time	$t_{SU:DAT}$	250			ns	
Rise Time of Both SDA and SCL Signals	t_R			1000	ns	
Fall Time of Both SDA and SCL Signals	t_F			300	ns	
Set-up Time for STOP Condition	$t_{SU:STO}$	4.7			μs	
Capacitive Load for each Bus Line	C_B			400	pF	8
I/O Capacitance	$C_{I/O}$		10		pF	
Crystal Specified Load Capacitance			12.5		pF	

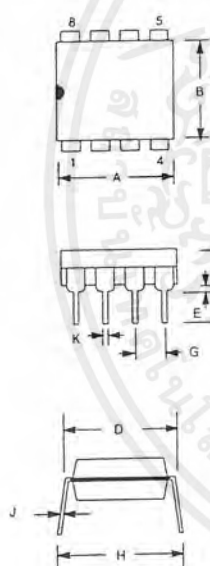
NOTES:

1. All voltages are referenced to ground.
2. Logic zero voltages are specified at a sink current of 5 mA at $V_{CC}=4.5V$, $V_{OL}=GND$ for capacitive loads.
3. I_{CCS} specified with $V_{CC}=5.0V$ and SDA, SCL=5.0V.
4. $V_{CC}=0V$, $V_{BAT}=3V$.
5. After this period, the first clock pulse is generated.
6. A device must internally provide a hold time of at least 300 ns for the SDA signal (referred to the V_{IHMIN} of the SCL signal) in order to bridge the undefined region of the falling edge of SCL.
7. The maximum $t_{HD:DAT}$ has only to be met if the device does not stretch the LOW period (t_{LOW}) of the SCL signal.
8. C_B - total capacitance of one bus line in pF.
9. I_{CCA} - SCL clocking at max frequency = 100 kHz.
10. SCL only.
11. SDA and SQW/OUT

TIMING DIAGRAM Figure 8

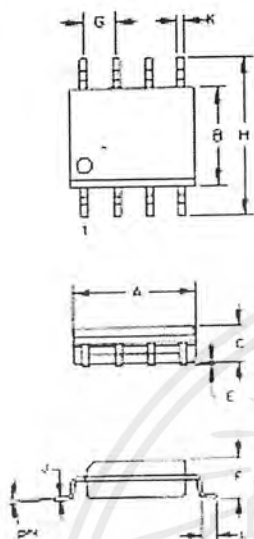


DS1307 64 X 8 SERIAL REAL TIME CLOCK 8-PIN DIP MECHANICAL DIMENSIONS



PKG	8-PIN	
DIM	MIN	MAX
A IN.	0.360	0.400
MM	9.14	10.16
B IN.	0.240	0.260
MM	6.10	6.60
C IN.	0.120	0.140
MM	3.05	3.56
D IN.	0.300	0.325
MM	7.62	8.26
E IN.	0.015	0.040
MM	0.38	1.02
F IN.	0.120	0.140
MM	3.04	3.56
G IN.	0.090	0.110
MM	2.29	2.79
H IN.	0.320	0.370
MM	8.13	9.40
J IN.	0.008	0.012
MM	0.20	0.30
K IN.	0.015	0.021
MM	0.38	0.53

DS1307Z 64 X 8 SERIAL REAL TIME CLOCK 8-PIN SOIC (150-MIL) MECHANICAL DIMENSIONS



PKG	8-PIN (150 MIL)	
DIM	MIN	MAX
A IN.	0.188	0.196
MM	4.78	4.98
B IN.	0.150	0.158
MM	3.81	4.01
C IN.	0.048	0.062
MM	1.22	1.57
E IN.	0.004	0.010
MM	0.10	0.25
F IN.	0.053	0.069
MM	1.35	1.75
G IN.	0.050 BSC	
MM	1.27 BSC	
H IN.	0.230	0.244
MM	5.84	6.20
J IN.	0.007	0.011
MM	0.18	0.28
K IN.	0.012	0.020
MM	0.30	0.51
L IN.	0.016	0.050
MM	0.41	1.27
phi	0°	8°

56-G2008-001