สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

เครื่องบันทึกการใช้งานโทรศัพท์อัตโนมัติ

AUTOMATIC TELEPHONE RECORDER

โดย

นางสาวธิดารัตน์ ทับเงิน

ปน.
ว582ค
2549

เลขหมู่.....
เลขทะเบียน 72983
วัน,เดือน,ปี 26 มิ.ย. 2550

b 11775932
i.....

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต

สาขาวิชาวิศวกรรมโทรคมนาคม

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2549

ภาควิชา
วิศวกรรมโทรคมนาคม

ผ่านการตรวจชิ้นงานแล้ว

(ลงชื่อ).....ผู้ตรวจ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เครื่องบันทึกการใช้งานโทรศัพท์อัตโนมัติ

AUTOMATIC TELEPHONE RECORDER

โดย

นางสาวธิดารัตน์ ทับเงิน 46012011

อาจารย์ที่ปรึกษา

รศ.ดร.ปราโมทย์ วาดเขียน

ผศ.ดร.จีรสุดา โกษียาภรณ์

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต

สาขาวิชาวิศวกรรมโทรคมนาคม

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2549

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ปริญญานิพนธ์ปีการศึกษา 2549

ภาควิชาวิศวกรรมโทรคมนาคม

คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เรื่อง เครื่องบันทึกการใช้งานโทรศัพท์อัตโนมัติ

AUTOMATIC TELEPHONE RECORDER

ผู้จัดทำ

1. น.ส.ธิดารัตน์ ทับเงิน 46012011

..... อาจารย์ที่ปรึกษา

(รศ.ดร.ปราโมย์ วาดเขียน)

..... อาจารย์ที่ปรึกษา

(ผศ.ดร.จิรฐดา โกษียาภรณ์)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เครื่องบันทึกการใช้งานโทรศัพท์อัตโนมัติ

AUTOMATIC TELEPHONE RECORDER

โดย 1. น.ส.ธิดารัตน์ ทับเงิน 46012011

อาจารย์ที่ปรึกษา รศ.ดร.ปราโมทย์ วาดเขียน

ผศ.ดร.จีรสุดา โกษียาภรณ์

บทคัดย่อ

โครงงานนี้ได้นำเสนอการจัดทำเครื่องบันทึกการใช้งานโทรศัพท์อัตโนมัติ ซึ่งมีหน่วยความจำสำหรับบันทึกเบอร์โทรศัพท์ที่โทรเข้า – ออก วันและเวลาที่ใช้งาน ระยะเวลาที่สนทนา และสามารถพิมพ์ออกทางเครื่องพิมพ์ได้ โดยในแต่ละสถานะการทำงานจะมีการแสดงผลผ่านจอ LCD ซึ่งเครื่องบันทึกการใช้งานโทรศัพท์นี้จะต่อขนานกับเครื่องโทรศัพท์และทำงานเป็นอิสระต่อกัน

ABTRACT

This project presents the automatic telephone recorder witch has memory for recording the incoming /outgoing telephone number, date/time and the duration of conversation. In addition, this information can be printed out. Moreover, each status of operation will be shown on LCD screen. This recording system is parallelly connected with telephone which they are independently operat

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กิตติกรรมประกาศ

ขอขอบคุณ คุณพ่อคุณแม่ที่ให้กำลังใจในการเรียนเสมอมา ขอขอบคุณอาจารย์ที่ปรึกษา รศ.ดร.ปราโมทย์ วาดเขียน และ ผศ.ดร.จีรสุดา โกษียาภรณ์ ที่ให้คำแนะนำในการทำงานตลอดจนช่วยติดตามตรวจสอบความถูกต้องความเรียบร้อยของโครงงานชิ้นนี้ และช่วยอำนวยความสะดวกในการทำงาน ขอขอบคุณ คุณปานวิทย์ ธุวะนุติ ที่อำนวยความสะดวกในการทำงานและขอขอบคุณเพื่อน ๆ ทุกคนที่ให้ความช่วยเหลือในการทำงานครั้งนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ

		หน้า
	บทคัดย่อ	
	สารบัญ	
	สารบัญภาพ	
	สารบัญตาราง	
บทที่ 1	บทนำ	1
	1.1 รายละเอียดโดยย่อของโครงงาน	1
	1.2 วัตถุประสงค์โครงงาน	1
	1.3 ประโยชน์และผลที่คาดว่าจะได้รับ	2
บทที่ 2	ทฤษฎี	3
	2.1 ทฤษฎีเบื้องต้นเกี่ยวกับโทรศัพท์	3
	2.1.1 เครื่องโทรศัพท์	3
	2.1.2 หน้าที่ของเครื่องโทรศัพท์	3
	2.1.3 ส่วนประกอบของเครื่องโทรศัพท์	4
	2.1.4 สัญญาณ (Signaling)	5
	2.1.4.1 สัญญาณระหว่างชุมสายกับชุมสาย (Inter Exchange Signaling)	5
	2.1.4.2 สัญญาณระหว่างผู้เช่ากับชุมสาย (Subscriber Signaling)	6
	2.1.5 Caller ID	8
	2.1.5.1 เกี่ยวกับ Caller ID	8
	2.1.5.2 รูปแบบของข้อความ	8
	2.1.6 โทรศัพท์ระบบปุ่มกด (DTMF)	10
	2.1.7 วงจรถอดรหัสความถี่โทรศัพท์	11
	2.1.7.1 คุณสมบัติของ MT8870	11
	2.1.7.2 โครงสร้างไอซี MT8870	11
	2.1.7.3 ฟังก์ชั่นการทำงานภายใน MT8870	12
	2.1.7.4 การนำ MT8870 ไปใช้งาน	15
	2.2 ไมโครคอนโทรลเลอร์ตระกูล MCS-51	15
	2.2.1 คุณสมบัติทั่วไปของไมโครคอนโทรลเลอร์ตระกูล MCS-51	16
	2.2.2 การจัดหน่วยความจำของไมโครคอนโทรลเลอร์	17
	2.2.2.1 หน่วยความจำสำหรับโปรแกรม (Program Memory)	17
	2.2.2.2 หน่วยความจำสำหรับเก็บข้อมูล (Data Memory)	17

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.2.2.3 รีจิสเตอร์ทำหน้าที่เฉพาะ (Special Function Register) 18
2.2.3 ไทม์เมอร์ / เคาน์เตอร์ 19
2.3 การเชื่อมต่อระหว่างคอมพิวเตอร์กับพรินเตอร์ 20
2.3.1 ลักษณะของการสื่อสารข้อมูล 20
2.3.2 การอินเตอร์เฟสกับพรินเตอร์แบบขนานและแบบเซ็นโทรนิกส์ 21
2.3.3 สายสัญญาณและคอนเนคเตอร์ 23
2.4 หน่วยแสดงผลแบบ LCD 25
2.4.1 ส่วนประกอบของ LCD 25
2.4.2 โครงสร้างภายในของตัวควบคุมโมดูล 25
2.4.3 โครงสร้างและขาของ LCD display 26
2.4.4 การเขียนคำสั่งและข้อมูลให้แก่โมดูล LCD 27
2.5 การเชื่อมต่ออุปกรณ์ระบบบัส I^2C 28
2.5.1 คุณสมบัติโดยทั่วไปของบัส I^2C 28
2.5.2 หลักการทำงานของบัส I^2C 29
2.5.3 สภาวะที่เกิดขึ้นบนบัส I^2C 29
2.5.4 การทำงานบนบัส I^2C 31
2.5.4.1 การอ้างอิงแบบ 7 บิต (7-bit addressing) 31
2.5.4.2 การอ้างอิงถึงแบบ 10 บิต 31
2.6 DS1307 ไอซีสร้างฐานเวลาจริงหรือไทม์คล็อก (RTC) 32
2.6.1 รายละเอียดขาการต่อใช้งานของ DS1307 32
2.6.2 การจัดสรรหน่วยความจำใน DS1307 33
2.6.3 รีจิสเตอร์ควบคุม 34
2.6.4 โหมดการทำงานของ DS1307 35
2.6.4.1 โหมดการเขียนข้อมูล 35
2.6.4.2 โหมดการอ่านข้อมูล 35
บทที่ 3 การออกแบบและการสร้าง 36
3.1 การทำงานโดยรวมของระบบ 36
3.2 วงจรดีเทคเตอร์ 36
3.2.1 การออกแบบวงจรตรวจสอบการวางหูและยกหูโทรศัพท์ 36
3.2.2 วงจรตรวจจับสัญญาณเรียก (Ringing Detector) 37
3.2.3 วงจรถอดรหัสหมายเลขโทรศัพท์ (DTMF Decoder) 38
3.2.4 วงจรตรวจจับสัญญาณเรียกกลับและสัญญาณไม่ว่าง 39
(Ringback, Busy Tone Ddetector)
3.2.5 วงจร Caller ID 41

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.2.5.1 การทำงานโดยทั่วไป 41
3.2.5.2 ภาคอินพุทและการเข้าถึงสัญญาณอนาลอกโดยตรง 42
(Direct Analog Access: DAA)
3.2.5.3 วงจรบริบและขยายแรงดัน 43
3.2.5.4 ภาคตัวกรองความถี่ 43
3.2.5.5 ภาคขยายแถบความถี่กว้าง 45
3.2.5.6 วงจรเฟสล็อคลูปและวงจรถอดรหัส FSK 45
3.3 Microcontroller 45
3.2.1 ส่วนควบคุมการรับข้อมูลจากวงจรดีเทคเตอร์ 46
3.2.2 การเชื่อมต่อจอแสดงผล LCD เข้ากับไมโครคอนโทรลเลอร์ 46
3.2.3 การเชื่อมต่อ PCF8574 เข้ากับไมโครคอนโทรลเลอร์ 47
3.2.4 การเชื่อมต่อ DS1307 เข้ากับไมโครคอนโทรลเลอร์ 48
3.2.5 โครงสร้างทาง Solfware 49
3.2.5.1 โปรแกรมควบคุมการบันทึกการใช้งานโทรศัพท์ 49
บทที่ 4 ผลการทดลอง 53
4.1 การทดลองวงจรตรวจจับสัญญาณยกหู / วางหูโทรศัพท์ 53
4.2 การทดลองวงจรตรวจจับสัญญาณเรียก (Ringing Detector) 54
4.3 การทดลองวงจรถอดรหัสหมายเลขโทรศัพท์ (DTMF Decoder) 54
4.4 การทดลองวงจรตรวจจับสัญญาณเรียกกลับและสัญญาณไม่ว่าง 55
(Ringback, Busy Tone Ddetector)
4.5 การทดลองวงจรกรองความถี่และวงจรขยาย 56
4.5.1 วงจรบริบและขยายตามแรงดัน 56
4.5.2 วงจร High Pass Filter 57
4.5.3 วงจร Low Pass Filter 58
4.5.4 การทดลองวงจรขยายแถบความถี่กว้าง 59
4.6 การทดลองวงจรถอดรหัส FSK 60
4.7 ผลการทดลองโปรแกรมส่วนแสดงเบอร์โทรออก 61
บทที่ 5 สรุปและวิจารณ์ผลการทดลอง 64
ภาคผนวก
กิตติกรรมประกาศ
เอกสารอ้างอิง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญภาพ

	หน้า
รูปที่ 1.1 องค์ประกอบของเครื่องบันทึกการใช้งานโทรศัพท์	1
รูปที่ 2.1 แสดงสัญญาณติดต่อระหว่างชุมสายกับชุมสาย	5
รูปที่ 2.2 สัญญาณให้หมุน	6
รูปที่ 2.3 สัญญาณไม่ว่าง	6
รูปที่ 2.4 สัญญาณเรียกกลับ	7
รูปที่ 2.5 สัญญาณกริ่งเรียก	7
รูปที่ 2.6 แสดงสัญญาณติดต่อระหว่างชุมสายกับชุมสาย (Subscriber signaling)	7
รูปที่ 2.7 รูปแบบข้อมูลแบบ Single Message	10
รูปที่ 2.8 รูปแบบข้อมูลแบบ Multiple Message	10
รูปที่ 2.9 แสดงหน้าปัทปุ่มกดโทรศัพท์แบบกดและความถี่ที่ใช้	11
รูปที่ 2.10 โครงสร้างภายในของ MT8870	12
รูปที่ 2.11 ความถี่ที่ได้จากภาคกรองความถี่	12
รูปที่ 2.12 วงจรตรวจสอบสัญญาณอย่างง่าย	13
รูปที่ 2.13 การต่อวงจรภาคอินพุท	14
รูปที่ 2.14 การต่อวงจรผลิตความถี่	15
รูปที่ 2.15 วงจรใช้งานเบื้องต้นของไอซี MT8870	15
รูปที่ 2.16 แสดงการติดต่อระหว่างคอมพิวเตอร์กับพรินเตอร์แบบขนาน	20
รูปที่ 2.17 แสดงการติดต่อระหว่างคอมพิวเตอร์กับพรินเตอร์แบบอนุกรม	21
รูปที่ 2.18 แสดงขาของคอนเนคเตอร์แบบแอมฟีนอล (Amphenal)	22
รูปที่ 2.19 แสดงช่วงเวลาของรูปคลื่นสัญญาณที่จะส่งข้อมูลอักขระไปยังพรินเตอร์แบบขนาน	23
รูปที่ 2.20 แสดงขาของคอนเนคเตอร์แบบ DB-25	23
รูปที่ 2.21 ลักษณะการต่อพอร์ทขนาน (D-TYPE)	23
รูปที่ 2.22 แสดงการต่อใช้งาน Parallel Port และ Printer Port	24
รูปที่ 2.23 รูปร่างและการจัดขาใช้งานของโมดูล LCD แบบอักขระ	27
รูปที่ 2.24 การต่อตัวต้านทานพูลอัปบนสายสัญญาณในระบบบัส I^2C	28
รูปที่ 2.25 การต่อตัวต้านทาน RS เพื่อลดสัญญาณรบกวนขนาดใหญ่ที่อาจเข้ามาในบัส I^2C	29
รูปที่ 2.26 ไดอะแกรมเวลาแสดงสถานะต่าง ๆ ในบัส I^2C	30
รูปที่ 2.27 รูปแบบของข้อมูลอนุกรมที่ใช้ติดต่อกับอุปกรณ์บัส I^2C เมื่อใช้การอ้างอิงแบบ 7 บิต	31
รูปที่ 2.28 รูปแบบของข้อมูลอนุกรมที่ใช้ติดต่อกับอุปกรณ์บัส I^2C เมื่อใช้การอ้างอิงถึงแบบ 10 บิต	32
รูปที่ 2.29 การจัดขาของไอซี DS1307 ไอซีสร้างฐานเวลาจริง (RTC)	33
รูปที่ 2.30 การจัดสรรหน่วยความจำแรมภายในพร้อมกับรายละเอียดของรีจิสเตอร์เก็บค่าเวลา และรีจิสเตอร์ควบคุมของ DS1307	34

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 2.31 รูปแบบของข้อมูลสำหรับติดต่อกับ DS1307 ในโหมดการเขียนข้อมูล 35
รูปที่ 2.32 รูปแบบของข้อมูลสำหรับติดต่อกับ DS1307 ในโหมดการอ่านข้อมูล 35
รูปที่ 3.1 แสดง Block diagram รวมของระบบ 36
รูปที่ 3.2 วงจรตรวจจับการยกหู / วางหู 37
รูปที่ 3.3 วงจรตรวจจับสัญญาณกระดิ่ง 38
รูปที่ 3.4 วงจรถอดรหัสหมายเลขโทรศัพท์ 38
รูปที่ 3.5 วงจรตรวจจับสัญญาณเรียกกลับและสัญญาณสายไม่ว่าง 39
รูปที่ 3.6 บล็อกไดอะแกรมวงจร CID 41
รูปที่ 3.7 วงจร Direct Analog Access 42
รูปที่ 3.8 วงจรกรองความถี่และวงจรขยาย 43
รูปที่ 3.9 แสดงรูปคลื่นเอาท์พุทของวงจรขริบ 43
รูปที่ 3.10 กราฟแสดงผลตอบสนองความถี่ของอินพุทฟิลเตอร์ 44
รูปที่ 3.11 วงจรการถอดรหัส FSK 46
รูปที่ 3.12 แสดงการต่อ LCD มอดูลกับพอร์ท P0 ของ MCS-51 47
รูปที่ 3.13 การกำหนดแอดเดรสของชิป PCF8574 47
รูปที่ 3.14 แสดงการเชื่อมต่อไมโครคอนโทรลเลอรกับไอซี PCF8574 48
รูปที่ 3.15 แสดการเชื่อมต่อไมโครคอนโทรลเลอร์ MCS-51 กับไอซี Real Time Clock DS1307 48
รูปที่ 3.16 โฟว์ชาร์ทของโปรแกรมแสดงข้อมูลที่บันทึกไว้ออกจอ LCD 49
รูปที่ 3.17 โฟว์ชาร์ทของโปรแกรมบันทึกการใช้งานในส่วนของการโทรเข้า 50
รูปที่ 3.18 โฟว์ชาร์ทของโปรแกรมบันทึกการใช้งานในส่วนของการโทรออก 51
รูปที่ 3.17 โฟว์ชาร์ทของโปรแกรมติดต่อพริ้นเตอร์ 52
รูปที่ 4.1 แสดงสัญญาณขณะวางหู 53
รูปที่ 4.2 แสดงสัญญาณเมื่อทำการยกหูโทรศัพท์ 53
รูปที่ 4.3 แสดงสัญญาณเรียก (Ringing) 54
รูปที่ 4.4 แสดงสัญญาณ DTMF เมื่อมีการกดหมายเลข 1,2,3 54
รูปที่ 4.5 แสดงสัญญาณ DTMF เมื่อมีการกดหมายเลข 4,5,6 55
รูปที่ 4.6 แสดงผลสัญญาณเรียกกลับ 56
รูปที่ 4.7 แสดงผลการทดลองวงจรขริบและขยายสัญญาณ 56
รูปที่ 4.8 กราฟแสดงผลตอบสนองความถี่ของ High Pass Filter 58
รูปที่ 4.9 กราฟแสดงผลตอบสนองความถี่ของ Low Pass Filter 59
รูปที่ 4.10 แสดงผลการทดลองวงจรขยายแถบความถี่กว้าง 59
รูปที่ 4.11 แสดงสัญญาณเอาท์พุทมีค่าลอจิกเป็น "1" เมื่อสัญญาณอินพุทมีค่าเป็น 1,200 Hz 60
รูปที่ 4.12 แสดงสัญญาณเอาท์พุทมีค่าลอจิกเป็น "0" เมื่อสัญญาณอินพุทมีค่าเป็น 2,200 Hz 60
รูปที่ 4.13 แสดงรูปจอ LCD ที่แสดงวันที่และเวลา 61

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 4.14 แสดงรูปจอ LCD ที่แสดงรายการการใช้งาน 61
รูปที่ 4.15 แสดงรูปจอ LCD ที่แสดง 61
รูปที่ 4.16 แสดงรูปจอ LCD ที่แสดงการยกหูและเลขหมายที่ถูกกด 61
รูปที่ 4.17 แสดงรูปจอ LCD เมื่อทำการกดเลขหมายไม่เสร็จภายในเวลา 30 วินาที 62
รูปที่ 4.18 แสดงรูปจอ LCD เมื่อทางด้านรับไม่รับสายภายในเวลา 20 วินาที 62
รูปที่ 4.19 แสดงรูปจอ LCD เมื่อด้านรับทำการรับสายภายในเวลา 20 วินาที 62
รูปที่ 4.20 แสดงรูปจอ LCD ที่แสดงข้อความให้เลือกดูเบอร์ที่โทรออกที่ทำการบันทึกเอาไว้ 62
รูปที่ 4.21 แสดงรูปจอ LCD ที่แสดงหมายเลขโทรศัพท์ ระยะเวลาที่โทร เวลา และวันที่ที่ทำการโทรออก 63
รูปที่ 4.22 แสดงรูปจอ LCD เมื่อยังไม่มีการบันทึกข้อมูล 63

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญตาราง

ตารางที่ 2.1 ตัวอย่าง Caller Identification Coding 9

ตารางที่ 2.2 รายละเอียดของ Bell 202A 9

ตารางที่ 2.3 แสดงค่าถอดรหัสที่ได้จากความถี่ต่าง ๆ 13

ตารางที่ 2.4 แสดงตำแหน่งของหน่วยความจำสำหรับเก็บโปรแกรม 17

ตารางที่ 2.5 ตารางตำแหน่งแอดเดรสของหน่วยความจำข้อมูลภายใน (Internal data memory) 18

ตารางที่ 2.6 แสดงรีจิสเตอร์ที่ใช้งานเฉพาะ 19

ตารางที่ 2.7 แสดงการเชื่อมโยงขาสำหรับการต่อขาระหว่าง D-TYPE 25 กับ Centronics 24

ตารางที่ 2.8 แสดงความสัมพันธ์ในการทำงานของขา RS, R/W และ E ของโมดูล LCD 27

ตารางที่ 3.1 แสดงสถานะของระดับสัญญาณเมื่อยกหูโทรศัพท์ 37

ตารางที่ 4.1 แสดงผลการทดลองวงจรถอดรหัสเลขหมายโทรศัพท์ 55

ตารางที่ 4.2 ผลการทดลอง High Pass Filter 57

ตารางที่ 4.3 ผลการทดลอง Low Pass Filter 58

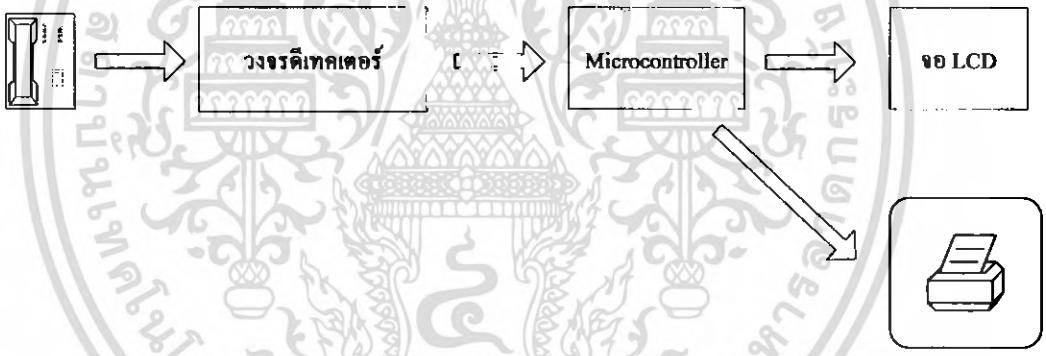
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 1
บทนำ

รายละเอียดโดยย่อของโครงงาน

ในปัจจุบันคงปฏิเสธไม่ได้ว่าโทรศัพท์เป็นเทคโนโลยีด้านการสื่อสารที่มีความสำคัญอย่างหนึ่งที่ช่วยเพิ่มความสะดวกสบายในด้านต่าง ๆ ไม่ว่าจะเป็นด้านธุรกิจ ด้านการบริหาร บริการและอื่น ๆ อีกมากมาย โดยปัจจุบันโทรศัพท์ที่มีการใช้งานอยู่ทั่วไปจะแบ่งเป็นโทรศัพท์เคลื่อนที่และโทรศัพท์ที่ใช้งานตามบ้านและสำนักงานซึ่งเป็นที่ทราบกันดีแล้วว่าโทรศัพท์เคลื่อนที่นั้นมีการพัฒนารูปแบบการใช้งานเพื่อเพิ่มความสะดวกสบายอยู่ตลอดเวลาและส่วนใหญ่จะให้ความสนใจในการพัฒนาการใช้งานบนโทรศัพท์เคลื่อนที่มากกว่าโทรศัพท์ตามบ้านและสำนักงาน

โครงงานนี้จึงจัดทำขึ้นเพื่อเพิ่มความสะดวกสบายในการใช้โทรศัพท์ตามบ้านและตามสำนักงานมากยิ่งขึ้น โดยเพิ่มรูปแบบการใช้งานโทรศัพท์ให้สามารถบันทึกข้อมูลการใช้งาน เช่น การบันทึกเลขหมายโทรเข้า – ออก วันเดือนปีที่ใช้งาน และระยะเวลาการใช้งาน โดยข้อมูลทั้งหมดจะถูกบันทึกไว้ในหน่วยความจำและสามารถดูข้อมูลในขณะที่ใช้งานอยู่ได้ทางจอ LCD และถ้าต้องการดูข้อมูลทั้งหมดที่อยู่ในหน่วยความจำก็สามารถพิมพ์ข้อมูลออกมาดูได้ โดยที่เครื่องบันทึกการใช้งานโทรศัพท์นี้จะต่อขนานกับเครื่องโทรศัพท์และทำงานเป็นอิสระต่อกัน โดยมีองค์ประกอบดังต่อไปนี้



รูปที่ 1.1 องค์ประกอบของเครื่องบันทึกการใช้งานโทรศัพท์

1.2 วัตถุประสงค์โครงงาน

1.1.1 เพื่อศึกษาและเรียนรู้เกี่ยวกับระบบการทำงานของโทรศัพท์

1.1.2 สามารถแสดงหมายเลขที่โทรออกและหมายเลขที่โทรเข้าได้

1.1.3 สามารถออกแบบและสร้างวงจรเพื่อดีเทคสัญญาณโทรศัพท์พร้อมกับสามารถนำสัญญาณต่าง ๆ มาประยุกต์ใช้งานได้

1.1.4 เพื่อศึกษาและเรียนรู้ การเชื่อมต่อไมโครคอนโทรลเลอร์กับอุปกรณ์ภายนอกที่ทำการติดต่อด้วย

1.1.5 เขียนโปรแกรมเพื่อรับข้อมูลที่ได้มาจากวงจรดีเทคได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1.1.6 เขียนโปรแกรมเพื่อควบคุมการทำงานทั้งหมด

1.1.7 จัดทำในส่วนของการบันทึกข้อมูลรายละเอียดของการใช้โทรศัพท์

1.3 ประโยชน์และผลที่คาดว่าจะได้รับ

ในปัจจุบัน เทคโนโลยีด้านการใช้งานโทรศัพท์มีการพัฒนาอย่างมาก แต่ยังมีข้อเสียคือ เรื่องของค่าใช้จ่าย ซึ่งยังสูงอยู่มาก และยังไม่มีการพัฒนาถึงขั้นที่ผู้ใช้บริการสามารถที่จะเก็บและพิมพ์ข้อมูลการใช้งานไว้ตรวจสอบความถูกต้องในการใช้บริการ ซึ่งเครื่องบันทึกการใช้งานโทรศัพท์นี้สามารถนำมาต่อใช้คู่กับโทรศัพท์เดิมที่มีอยู่ได้ทันทีโดยไม่ต้องเสียค่าใช้จ่ายในการซื้อโทรศัพท์ใหม่และข้อมูลที่ถูกเก็บไว้ยังสามารถพิมพ์ออกมาใช้เป็นข้อมูลอ้างอิงในการตรวจสอบความถูกต้องในการใช้งานโทรศัพท์ได้อีกด้วย

ในรายงานเล่มนี้ จะนำเสนอเกี่ยวกับทฤษฎีและหลักการเบื้องต้นของการส่งสัญญาณและข้อมูลผ่านสายโทรศัพท์ ตลอดจน รายละเอียดในการพัฒนาเครื่องบันทึกการใช้งานโทรศัพท์ให้มีประสิทธิภาพและมีรูปแบบการใช้งานให้มากขึ้น และ ผลการทดลอง ทั้งจาก ส่วนที่เป็นการโทรเข้า ส่วนที่เป็นการโทรออก และ ส่วนของวงจรรวมทั้งหมด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 2
ทฤษฎี

2.1 ทฤษฎีเบื้องต้นเกี่ยวกับโทรศัพท์

2.1.1 เครื่องโทรศัพท์

ความหมายของ Local loop คือ สายส่ง 2 wire จากเครื่องโทรศัพท์ ไปชุมสายปลายทางและมีค่าอิมพีแดนซ์ของสายประมาณ 500-1000 โอห์ม แต่ค่าที่ใช้โดยทั่วไปคือ 600 โอห์ม ถ้าในชุมสายปลายทางมีการติดตั้งแหล่งจ่ายไฟร่วม DC ขนาด 48 โวลต์ ให้แต่ละลูปของผู้ใช้โทรศัพท์ ลวดตัวนำ 2 เส้น มีชื่อว่า ทิป (Tip) และริง (Ring) โดยริงจะต่อกับสัญญาณไฟ -48 VDC ทิปจะต่อกับกราวด์ เมื่อผู้ใช้ยกหูโทรศัพท์มีผลทำให้ฮุคสวิทช์ปิดลง (Hook off) จากนั้นกระแสไฟฟ้าตรงขนาด 20 mA ไหลวนอยู่ในลูป ซึ่งสภาวะยกหูโทรศัพท์นี้ระดับแรงดันไฟฟ้าทิปและริงจะมีค่าลดลงเหลือ 4 โวลต์เปลี่ยนแปลงเล็กน้อยภายในลูป (20 mA) ซึ่งเกิดจากสัญญาณ AC ทับบนกระแสลูป DC

เครื่องโทรศัพท์จะติดต่อกับชุมสายโทรศัพท์ด้วยสายสัญญาณ 2 สาย คือ สายทิป (Tip) และสายริง (Ring) ปกติเมื่อไม่มีการใช้งานโทรศัพท์วงจรเครื่องรับโทรศัพท์ จะถูกตัดออกจากคู่สายของโทรศัพท์ จะคงเหลือแต่วงจรกำเนิดเสียง (Ringing) หรือวงจรกระดิ่งต่อกับชุมสายโทรศัพท์เท่านั้น เพื่อส่งสัญญาณเรียก เมื่อมีการติดต่อจากผู้อื่นเข้ามา ทำให้ในขณะที่โทรศัพท์ไม่ถูกใช้งาน จะไม่มีกระแสไหลผ่านเครื่องรับโทรศัพท์แต่เมื่อเรายกหูโทรศัพท์ก็จะมีการเชื่อมต่อเครื่องโทรศัพท์เข้ากับชุมสายโทรศัพท์ ทำให้เกิดการไหลของกระแสขึ้นในวงจร โดยกระแสนี้จะมาจากแบตเตอรี่ในชุมสายโทรศัพท์และ เมื่อชุมสายโทรศัพท์เลือกคู่สายที่ต้องการจะต่อด้วยได้แล้วก็จะทำการส่งสัญญาณกระดิ่ง ซึ่งเป็นสัญญาณ AC ออกไปยังเครื่องรับโทรศัพท์ของผู้ถูกเรียก เพื่อทำการสั่นกระดิ่งให้ดังขึ้น เมื่อผู้ถูกเรียกยกหูโทรศัพท์ทำการรับสายก็จะเกิดกระแส DC ไหล เมื่อชุมสายตรวจพบก็จะหยุดส่งสัญญาณกระดิ่ง ดังนั้นจะสามารถทำการสนทนากันได้ โดยในส่วนที่เชื่อมต่อระหว่างปากพูดและหูฟังกับสายโทรศัพท์จะต้องมีหม้อแปลงอัตโนมัติทำหน้าที่ในการปรับอิมพีแดนซ์ของหูฟังและสายโทรศัพท์ให้สมดุลกัน เพื่อให้การรับและการส่งสัญญาณมีประสิทธิภาพมากที่สุดรวมทั้งทำให้ผู้พูดสามารถได้ยินเสียงตนเองออกไปด้วย เพื่อที่จะปรับระดับการพูดของตนเองไม่ให้ดังเกินไป

2.1.2 หน้าที่ของเครื่องโทรศัพท์

1. เครื่องโทรศัพท์ทำหน้าที่บอกรับในกรณีที่มีการเรียกเข้ามา โดยใช้สัญญาณ Ringing
2. เครื่องโทรศัพท์จะส่งสัญญาณที่เรียกว่า "สัญญาณหมุน" บอกว่าพร้อมที่จะทำการหมุนหมายเลขที่จะติดต่อได้ ซึ่งก็คือ เสียงที่ได้ยินเวลายกหู เป็นสัญญาณเสียงที่มีความถี่ 350 Hz กับ 400 Hz มามอดูเลตกัน
3. เครื่องโทรศัพท์จะส่งสัญญาณบอกผู้เรียกว่าหมายเลขที่ต้องการติดต่อด้วยว่าว่างหรือไม่ ถ้าว่างก็จะส่งสัญญาณตอบกลับเป็นความถี่ 480 Hz กับ 620 Hz มอดูเลตรวมกันมา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4. เครื่องโทรศัพท์จะส่งสัญญาณไปยังชุมสายเพื่อแจ้งให้ทราบว่า สิ้นสุดการใช้งานแล้ว และให้ชุมสายทำการเลิกติดต่อกับอีกฝ่ายได้

5. เครื่องโทรศัพท์ทำหน้าที่แปลงสัญญาณเสียงของผู้พูดเป็นสัญญาณไฟฟ้า และในทางกลับกันก็จะแปลงสัญญาณไฟฟ้าที่ถูกส่งมาให้เป็นสัญญาณเสียง

6. รูปแบบการส่งหมายเลขของผู้ใช้โทรศัพท์ เพื่อส่งให้กับ Central office สามารถแบ่งออกได้เป็น 2 แบบคือ Pulse กับ Tone

7. เครื่องโทรศัพท์จะปรับระดับสัญญาณเสียงให้มีความดังที่เหมาะสม โดยชดเชยความสูญเสียอันเกิดจากระยะทางระหว่างเครื่องโทรศัพท์และ Central office ซึ่งมีความใกล้และไกลต่างกัน

8. เครื่องโทรศัพท์จะติดต่อกับ Central office เมื่อผู้ใช้โทรศัพท์ต้องการใช้บริการ โดยทำการยก Handset (Off Hook)

9. เครื่องโทรศัพท์จะทำให้มี Feedback จากไมโครโฟนไปสู่หูของผู้ฟัง ผู้พูดก็จะได้ยินเสียงตนเอง (Side Tone) และสามารถควบคุมระดับความดังของ Side Tone ได้

10. เมื่อเครื่องโทรศัพท์ยังไม่มีการใช้งาน (On Hook) วงจรOn – Hook / Off – Hook จะทำการแยกเครื่องโทรศัพท์ออกจาก Central Office โดยการใช้ Switch Hook

2.1.3 ส่วนประกอบของเครื่องโทรศัพท์

1. วงจรกำเนิดสัญญาณหมายเลข

จะทำหน้าที่ในการสร้างสัญญาณหมายเลขโทรศัพท์ ซึ่งอาจจะเป็นสัญญาณพัลส์ หรืออาจจะเป็นสัญญาณความถี่คู่ผสมก็แล้วแต่โทรศัพท์ที่ใช้งาน

2. วงจรกระดิ่ง

ชุมสายโทรศัพท์จะส่งสัญญาณเรียกเข้ามาเป็นกระแสสลับประมาณ 100 – 120 โวลต์ ความถี่ 25 Hz ส่งเข้ามา วงจรกระดิ่งจะดังขึ้น เพื่อแจ้งให้เจ้าของเครื่องรับทราบว่ามีการเรียกเข้ามา

3. วงจรตัดเสียง

จะทำการตัดเสียงต่าง ๆ ไม่ให้เข้ามาในชุมสายโทรศัพท์เพื่อป้องกันการผิดพลาด ถ้าหากบังเอิญเกิดมีสัญญาณ เสียงที่มีความถี่ใกล้เคียงกับสัญญาณความถี่ของหมายเลข ก็จะทำให้การส่งสัญญาณหมายเลขเกิดความผิดพลาดขึ้นได้

4. วงจรไฮบริด (Hybrid)

เนื่องจากวงจรในเครื่องรับโทรศัพท์จะใช้ระบบโทรศัพท์ 4 สาย ก็คือใช้ 2 สาย ใช้สำหรับการส่งสัญญาณเสียงพูดและอีก 2 สาย สำหรับนำเสียงพูดจากอีกด้านหนึ่งมายังหูฟัง แต่ในการเชื่อมต่อระหว่างเครื่องโทรศัพท์กับชุมสายโทรศัพท์จะเป็นระบบ 2 สาย ซึ่งจะติดต่อโดยใช้สายสัญญาณเพียง 2 เส้น เท่านั้นเอง จึงต้องใช้วงจรไฮบริดมาทำการเชื่อมต่อระบบ 2 สาย กับ ระบบ 4 สาย ให้สามารถติดต่อกันได้

5. วงจรปากพูดหูฟัง

ในส่วนนี้จะทำหน้าที่เหมือนวงจรเครื่องรับ – เครื่องส่ง ที่ใช้ในการติดต่อสื่อสารกันระหว่างเครื่องโทรศัพท์ของผู้เรียกและผู้ถูกเรียกเพื่อให้สามารถสนทนากันได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.1.4 สัญญาณ (Signaling)

สัญญาณ คือ ข่าวสารที่ใช้ติดต่อระหว่างเครื่องโทรศัพท์กับชุมสายโทรศัพท์หรือข่าวสารที่ติดต่อระหว่างชุมสายหนึ่งไปยังอีกชุมสายหนึ่ง

หน้าที่ทั่ว ๆ ไปของสัญญาณโทรศัพท์ในปัจจุบันมี 4 หน้าที่ คือ

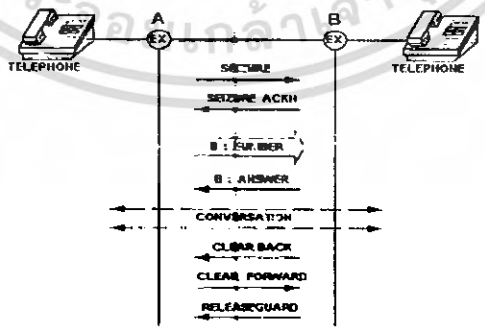
1. การเตรียมพร้อม (Alerting)
2. การส่งที่อยู่ของข่าวสาร (Transmission Address Information)
3. การตรวจตรา (Supervising)
4. การส่งสัญญาณข่าวสาร (Transmission Information Signaling)

จากข้อมูลข้างต้นเราสามารถแบ่งสัญญาณที่ใช้ในการติดต่อสื่อสารผ่านสายโทรศัพท์ออกเป็น 2 ประเภท คือ

2.1.4.1 สัญญาณระหว่างชุมสายกับชุมสาย (Inter Exchange Signaling)

สัญญาณพื้นฐานมีอยู่ 5 ประเภท คือ

1. Seizure คือ สัญญาณจับวงจร เป็นสัญญาณให้ชุมสายปลายทางทราบว่าคู่สายขณะนี้ถูกใช้งานอยู่ ชุมสายปลายทางจะทำการจัดเตรียมอุปกรณ์ที่ตัวรับเลขหมายของผู้ถูกเรียกจะส่งมา
2. Address Information เป็นสัญญาณบอกหมายเลขหรือประเภทของผู้เช่า
3. Answer Signal คือ สัญญาณตอบรับ สัญญาณนี้จะถูกส่งเมื่อผู้เช่าฝ่ายถูกเรียกยกหูรับ หน้าที่หลักของสัญญาณนี้มีดังนี้
   1) เริ่มต้นคิดค่าบริการ
   2) ส่งสัญญาณคิดค่าบริการ
   3) ตัดวงจรการจับเวลาการใช้อุปกรณ์
4. Clear – Forward คือ สัญญาณยกเลิกการต่อตรง สัญญาณนี้จะถูกส่งเมื่อฝ่ายเรียกวางหู ผลของสัญญาณนี้จะทำให้วงจรทางปลายทางทำการยกเลิกการต่อวงจรต่าง ๆ
5. Clear – Back คือ สัญญาณยกเลิกการต่อกลับ สัญญาณนี้จะถูกส่งเมื่อผู้เช่าฝ่ายถูกเรียกวางหู ผลของสัญญาณนี้จะทำให้ชุมสายต้นทางเริ่มจับเวลาเมื่อเวลาผ่านไป 90 – 120 วินาที ชุมสายต้นทางจะยกเลิกการติดต่อพร้อมกับส่งสัญญาณ Clear – Forward



รูปที่ 2.1 แสดงสัญญาณติดต่อระหว่างชุมสายกับชุมสาย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

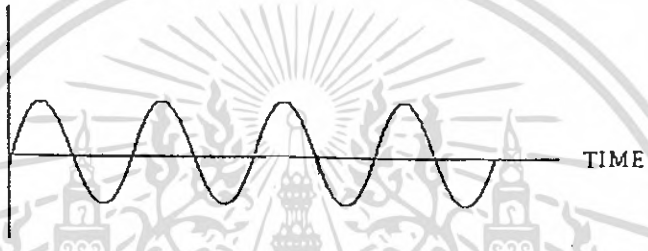
2.1.4.2 สัญญาณระหว่างผู้เช่ากับชุมสาย (Subscriber Signaling)

1. On Hook หมายถึงสภาพผู้เช่าวางหูหรือสภาพว่าง (Idle) ลักษณะของวงจรเป็นแบบ Open loop ซึ่งจะทำให้มีค่าอิพีแดนซ์สูง

2. Off Hook หมายถึงสภาพผู้เช่ายกหู ลักษณะของวงจรเป็นแบบ Close loop ซึ่งจะทำให้มีค่าอิมพีแดนซ์ต่ำ

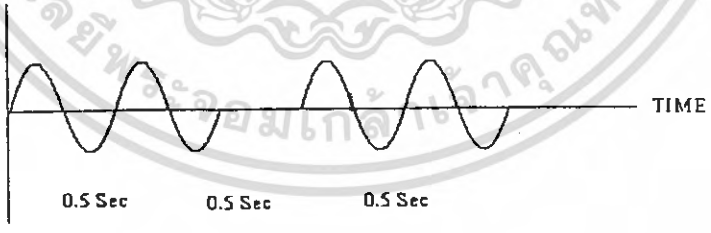
3. Dialing หมายถึง สภาพผู้เช่าทำการหมุนหมายเลข โดยเครื่องแบบหมุนสัญญาณเป็นพัลส์ (Pulse) ค่า Impedance จะสูงต่ำสลับกันตามรหัสที่หมุน ถ้าเป็นแบบกดปุ่ม Touch – Tone สัญญาณออกจะเป็นความถี่ DTMF ส่งออกไปชุมสาย

4. Dial Tone คือ สัญญาณที่บอกสภาพการว่างของชุมสายพร้อมที่จะรับเลขหมาย สัญญาณ Dial Tone นี้จะเป็นสัญญาณแบบคลื่นซายน์ความถี่ 400 ถึง 450 Hz ส่งมาอย่างต่อเนื่องและมีระดับขนาด 5 โวลต์ พีคทูพีค ดังต่อเนื่องนานประมาณ 30 วินาทีดังรูปที่ 2.2



รูปที่ 2.2 สัญญาณให้หมุน

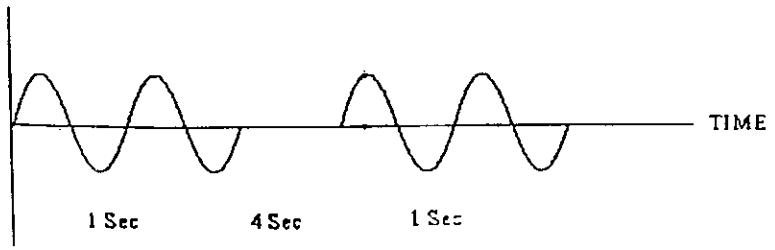
5. Busy Tone คือ สัญญาณที่บอกให้ทราบว่าอุปกรณ์ที่ชุมสายไม่ว่าง ถ้าได้ยินเสียงตอนยกหูแสดงว่าอุปกรณ์ในการติดต่อชุมสายไม่ว่าง และถ้าได้ยินสัญญาณนี้หลังจากหมุนเลขหมายไปแล้วแสดงว่าผู้เช่าฝ่ายถูกเรียกสายไม่ว่าง ผู้เรียกจึงควรวางหูโทรศัพท์สักระยะหนึ่งแล้วจึงทำการเรียกใหม่ ลักษณะสัญญาณที่ส่งไปจะเป็นสัญญาณที่ขาดตอนเป็นช่วง ๆ ส่ง 0.5 วินาที หยุด 0.5 วินาที ความถี่ของสัญญาณ 400 ถึง 450 Hz และมีขนาด 5 โวลต์พีคทูพีค ดังรูปที่ 2.3



รูปที่ 2.3 สัญญาณไม่ว่าง

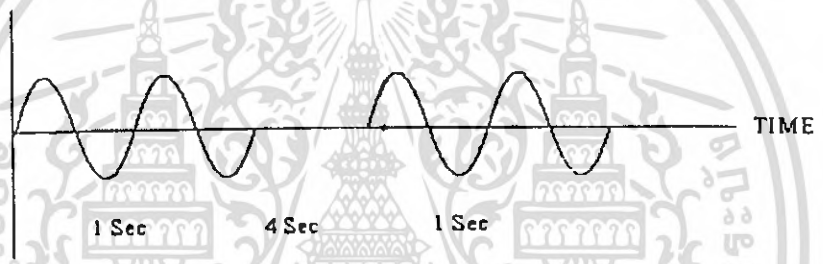
6. Ring Back Tone คือ สัญญาณที่ผู้เรียกได้ยินหลังจากหมุนเลขครบแล้ว เพื่อแสดงว่าการต่อถูกขั้นตอนตามความต้องการของผู้เรียกไปยังผู้รับ และส่งสัญญาณกริ่งให้กับผู้รับ เพื่อตอบรับการเรียกสัญญาณกริ่งเป็นสัญญาณคลื่นซายน์ที่มีความถี่ 25 Hz มาเป็นช่วง ๆ โดยมีจังหวะดัง 1 วินาที และหยุด 4 วินาที ระดับขนาด 90 โวลต์ ดังรูปที่ 2.4

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



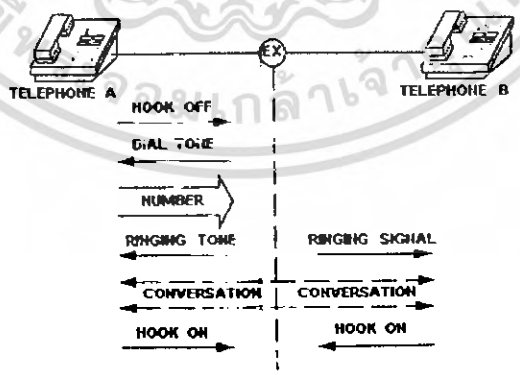
รูปที่ 2.4 สัญญาณเรียกกลับ

7. Ringing Tone คือ สัญญาณที่ผู้ถูกเรียกได้ยิน เพื่อแสดงว่าการต่อทุกขั้นตอนเป็นไปตามความต้องการของผู้เรียกไปยังผู้รับ เครื่องชุมสายโทรศัพท์ สามารถดำเนินการต่อได้สำเร็จและส่งสัญญาณกริ่งมาให้ผู้รับตอบรับการเรียก สัญญาณกริ่งเป็นสัญญาณคลื่นซายน์ มีความถี่ 400 ถึง 450 Hz ส่งมาเป็นช่วงโดยมีจังหวะดัง 1 วินาที หยุด 4 วินาที ดังรูปที่ 2.5



รูปที่ 2.5 สัญญาณกริ่งเรียก

8. สัญญาณโทนอื่น ๆ เช่น Nu Tone (Number Unobtainable Tone) เป็นเสียงที่บอกให้ผู้เรียกทราบว่าเบอร์ที่เรียกไปนั้นยังไม่ได้ติดตั้ง ลักษณะของสัญญาณ Nu Tone จะเป็นความถี่ 400 ถึง 450 Hz ดังเป็นจังหวะ 0.1 วินาทีต่อเนื่องกันไป



รูปที่ 2.6 แสดงสัญญาณติดต่อระหว่างชุมสายกับชุมสาย (Subscriber signaling)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.1.5 Caller ID

2.1.5.1 เกี่ยวกับ Caller ID

จุดเด่นของ Caller ID คือ เมื่อยังไม่มีการใช้งานโทรศัพท์ (On Hook) จะสามารถเรียมข้อมูลผู้ใช้เกี่ยวกับผู้เรียกก่อนตอบรับการเรียกจริง ข้อมูลที่แสดงนั้นเป็นข้อมูลข่าวสารที่ส่งจาก Central Office ไปยัง CPE โดยใช้การรับส่งข้อมูลทิศทางเดียวแบบ VDI – 1 (Voice Band Digital Interface) ระหว่างช่องว่างที่มีการเว้นช่วงของสัญญาณและหลังการปรากฏสัญญาณ Ringing ความถี่ 20 Hz สัญญาณแรก ข้อมูลที่ส่งมาจะมีวันที่ (วันและเดือน), เวลา (ชั่วโมงและนาที) และข้อมูลของผู้เช่าที่ทำการเรียกซึ่งจะมีรูปแบบการแสดงข้อมูลอย่างใดอย่างหนึ่งใน 3 รูปแบบต่อไปนี้

1) แสดงหมายเลข 2 – 10 Digit
2) แสดงเครื่องหมายที่แสดงความเป็นส่วนตัวกรณีผู้เช่าที่ทำการเรียกไม่ต้องการแสดงหมายเลข
3) แสดงเครื่องหมายที่แสดงว่าอยู่นอกเขตพื้นที่หมายเลขผู้เรียกก็จะไม่สามารถแสดงบนจอได้

VDI – 1 เป็นรูปแบบจำเพาะของ 3 layer คือ Physical Layer, Data link Layer และ Presentation Layer

2.1.5.2 รูปแบบของข้อความ

ข้อมูล Caller ID จะถูกส่งไปยัง CPE ในช่วงสัญญาณว่างหลังจากสัญญาณ Ringing สัญญาณแรก สำนักงานกลางจะรอครึ่งวินาทีหลังจากสัญญาณ Ringing ก่อนจะเริ่มส่งข้อมูลและตลอดการส่งข้อมูลครึ่งวินาทีก่อนสัญญาณ Ringing สัญญาณต่อไป ข้อมูล FSK ที่ถูกส่งไปยัง CPE เป็นแบบ Single message หรือ Multiple message (ดูรูปที่ 2.7 และ 2.8) ข้อมูล Caller ID ทั้งหมดจะต้องมีช่องสัญญาณ Seizure ส่งมาก่อน (มีรูปแบบ 01010101) เป็นเวลา 250 msec สัญญาณนี้จะถูกส่งออกมาก่อนเพื่อเป็นการเตือนให้ CPE ทราบว่าจะมีข้อความมาถึง ข้อความที่ถูกส่งตามมานั้นจะใช้เวลา 150 msec มีความถี่ 1,200 Hz ซึ่งจะช่วยในการปรับเปลี่ยนสภาวะของเครื่องรับข้อมูล ข้อความที่จะส่งจะเริ่มต้นจาก ชนิดของข้อความ (Message type) ซึ่งมีนาด 1 ไบต์ ตามตารางที่ 2.1 โดยที่ความยาวข้อความ (Message length) ข้อมูล (Data word) นั้นจะมีค่าจาก 9 จนถึง 18 ซึ่งจะระบุหมายเลขของข้อมูลที่กำลังจะส่งตามเวิร์ดนี้มา หมายเลขนี้จะไม่รวมกับ Check sum ที่จะส่งตามมาตอนท้ายสุด

บิตข้อมูล Caller ID จะแบ่งเป็นกลุ่มละ 8 บิตตัวอักษร ซึ่งจะนำหน้าด้วยบิตเริ่มต้น (Start bit) มีลอจิกเป็น "0" และจะปิดท้ายด้วยบิตบิตหยุด (Stop bit) มีลอจิก "1" ตามรูปที่ 2.7 ข้อมูล (Data word) จะถูกส่งเป็นอักษร ASCII (ASCII characters) ซึ่งไม่มีแพริตี้ ส่วนแรกของข้อมูลจะเป็นวันที่ (วันและเดือน) และเวลา (ชั่วโมงและวินาที) อย่างละ 2 ตัวอักษร เวิร์ดที่ 11 ถึง 20 จะเป็นหมายเลขของผู้เช่า โดยหมายเลขของผู้เช่าจะมีตั้งแต่ 2 ถึง 10 ดิจิตหรือเป็นพยัญชนะตัวอักษร ASCII แสดงให้ทราบเช่น "P" แสดงว่าเป็นข้อมูลส่วนตัวไม่ต้องการให้ผู้อื่นทราบหรือ "O" แสดงถึงพื้นที่นอกเขตบริการ ไบต์สุดท้ายคือ Check sum จะถูกใช้โดย CPE เพื่อรับรองความสมบูรณ์ของข้อมูลที่ได้รับ ส่วนของ Check sum จะประกอบด้วย 2's complement ของจำนวน 256 ของทุกส่วนที่ส่งจาก CO ซึ่งจะรวมชนิดข้อมูล (Message type) ความยาวข้อมูล (Message length) และข้อมูล (data word) เมื่อ CPE ได้รับจำนวนและบวกจำนวนนี้เข้ากับ Check sum แล้วได้ผลลัพธ์ไม่เท่ากับศูนย์แสดงว่าข้อมูลที่ได้รับไม่ถูกต้อง (ดูตารางที่ 2.1)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ส่วนข้อมูลแบบ Multiple message จะประกอบด้วยข้อมูลพารามิเตอร์ที่เพิ่มเข้าไป พารามิเตอร์แต่ละตัวเป็นข้อมูลอนุกรมของข้อมูล (Data word) ที่ระบุชนิดของพารามิเตอร์ (Parameter type) ความยาวพารามิเตอร์ (Parameter length) และข้อมูลพารามิเตอร์ (Parameter data) ตามรูปที่ 2.8

Word #	Signification	Binary Contents 7 6 5 4 3 2 1 0	Description	Dec. Value	Hex Value	Mod. 256 in Hex
1	Msg. Type	0 0 0 0 0 1 0 0	CND[1]	04	04	04
2	Length	0 0 0 1 0 0 1 0	18	18	12	16
3	Month	0 0 1 1 0 0 0 0	0	48	30	46
4		0 0 1 1 0 1 0 0	4	52	34	7A
5	Day	0 0 1 1 0 0 1 0	2	50	32	AC
6		0 0 1 1 1 0 0 0	8	56	38	E4
7	Hour	0 0 1 1 0 0 0 1	1	49	31	15
8		0 0 1 1 0 0 1 1	3	51	33	48
9	Minutes	0 0 1 1 0 0 1 0	2	50	32	7A
10		0 0 1 1 0 0 0 0	0	48	30	AA
11	Calling Number	0 0 1 1 0 1 0 0	4	52	34	DE
12		0 0 1 1 0 0 0 0	0	48	30	OE
13		0 0 1 1 1 0 0 0	8	56	38	46
14		0 0 1 1 0 1 0 0	4	52	34	7A
15		0 0 1 1 0 0 1 1	3	51	33	AD
16		0 0 1 1 0 1 0 0	4	52	34	E1
17		0 0 1 1 0 1 1 0	6	54	36	17
18		0 0 1 1 0 1 0 0	4	52	34	4B
19		0 0 1 1 0 0 0 0	0	48	30	7B
20		0 0 1 1 0 0 0 0	0	48	30	AB
21	Checksum	0 1 0 1 0 1 0 1	Checksum[2]	85	55	55

Notes
[1] CND = Calling Number Delivery
[2] Calculated Checksum + Received Checksum = 0 AB + 55 = 0 Mod 256

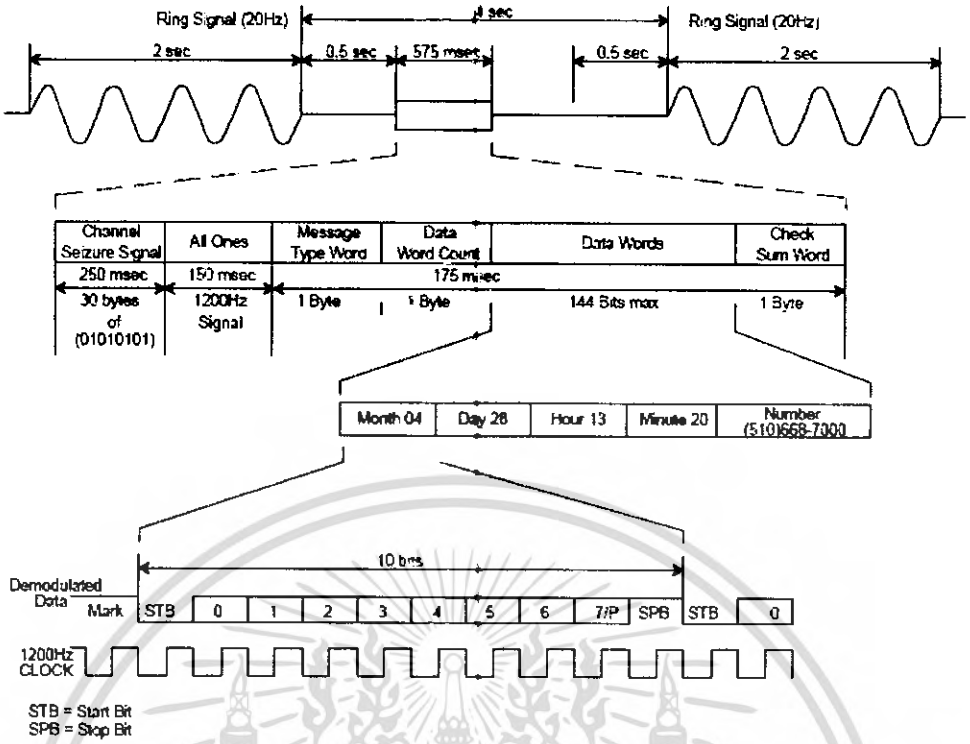
ตารางที่ 2.1 ตัวอย่าง Caller Identification Coding

การดีมอดูเลชั่นของสัญญาณ FSK ทำได้ตามรายละเอียดของ Bell 202A ดังนี้

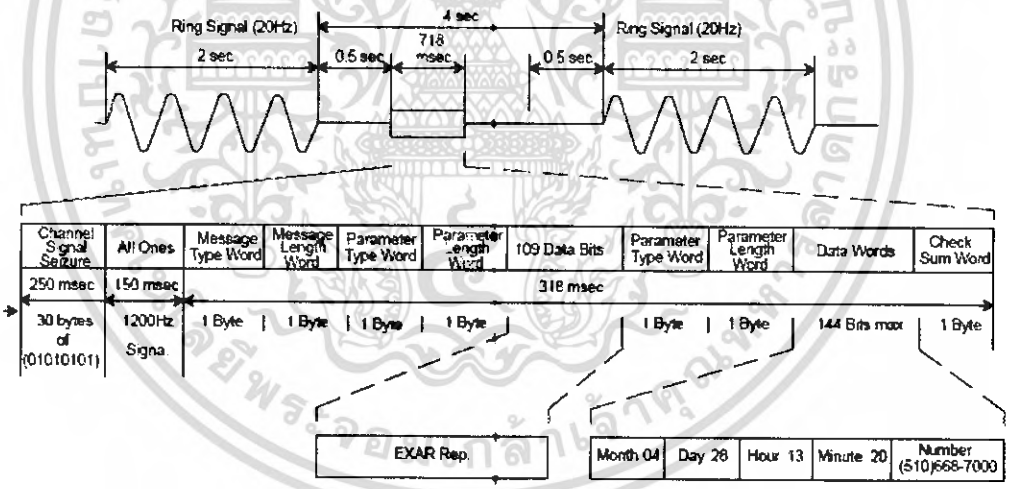
Link Type:	Simplex
Modulation Scheme:	Phase Coherent Frequency Shift Keying
Logical 1 (Mark):	1200+/-12Hz
Logical 0 (Space):	2200+/-22Hz
Transmission rate:	1200 bits per second
Data:	Serial, Binary, Asynchronous
Transmission Level:	-13,5+/-1dBm into 900Ω

ตารางที่ 2.2 รายละเอียดของ Bell 202A

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.7 รูปแบบข้อมูลแบบ Single Message



รูปที่ 2.8 รูปแบบข้อมูลแบบ Multiple Message

2.1.6 โทรศัพท์ระบบปุ่มกด (DTMF)

เครื่องโทรศัพท์ระบบกดปุ่มจะใช้วิธีของ Dual Tone Multi Frequency (DTMF) ในการส่งหมายเลขโทรศัพท์ โดยทั่ว ๆ ไปหน้าปัดของเครื่องจะมี 12 ปุ่ม แบ่งเป็น 4 แถว และ 3 หลัก และบางรุ่นจะมี 16 ปุ่ม โดยการเพิ่มหลักที่ 4 ขึ้นมาอีกดังที่แสดงในรูปที่ 2.9

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

	column 1 1209 Hz	column 2 1336 Hz	column 3 1477 Hz	column 4 1633 Hz
row 1 697 Hz	1	2	3	A
row 2 770 Hz	4	5	6	B
row 3 852 Hz	7	8	9	C
row 4 941 Hz	*	0	#	D

รูปที่ 2.9 แสดงหน้าปัทปุ่มกดโทรศัพท์แบบกดและความถี่ที่ใช้

การทำงานเมื่อกดปุ่มหมายเลขใดหมายเลขหนึ่งจะประกอบด้วย โทนเสียง 2 ความถี่ คือ ความถี่ของทั้ง 4 แถว เรียกว่า กลุ่มความถี่ต่ำ (Low Group Frequency) และความถี่ของทั้ง 3 และ 4 หลัก เรียกว่า กลุ่มความถี่สูง (High Group Frequency) ซึ่งแต่ละหมายเลขจะให้ค่าความถี่คู่ต่างกัน การกดปุ่มที่หมายเลขใด ๆ จะทำให้วงจรภายในของเครื่องผลิตความถี่ออกมา 2 ความถี่ เช่น เมื่อกดปุ่ม เลข "9" จะอยู่ในหลักของ 1447 Hz และแถว 852 Hz เอาท์พุท ที่ผลิตออกมาเรียกว่า ความถี่ DTMF

2.1.7 วงจรถอดรหัสความถี่โทรศัพท์

จะใช้ไอซี MT8870 ถอดรหัสสัญญาณโทรศัพท์ชนิดกดปุ่ม (DTMF) ให้เป็นตัวเลข BCD ขนาด 4 บิต โดยใช้งานร่วมกับคริสตอล 3.579 MHz เท่านั้น

การถอดรหัสความถี่โทรศัพท์ หมายถึง การแปลงสัญญาณความถี่ซึ่งเกิดจากการกดปุ่มตัวเลขของโทรศัพท์ชนิดกดปุ่ม (ชนิด Tone หรือ DTMF) ให้เป็นระบบตัวเลขทางดิจิตอล ซึ่งไอซี MT8870 ใช้แปลงความถี่โทรศัพท์ให้เป็นเลขฐานสองขนาด 4 บิต

ในยุคก่อน การออกแบบวงจรถอดรหัสความถี่โทรศัพท์มักใช้ไอซีจำพวกเฟสล็อคลูป ซึ่งสร้างปัญญาหาสารพัด ไม่ว่าเรื่องขนาดความถี่ที่เปลี่ยนแปลงไป การปรับแต่งวงจรขนาดวงจรที่ใหญ่เพราะต้องใช้ไอซีจำนวนมาก

2.1.7.1 คุณสมบัติของ MT8870

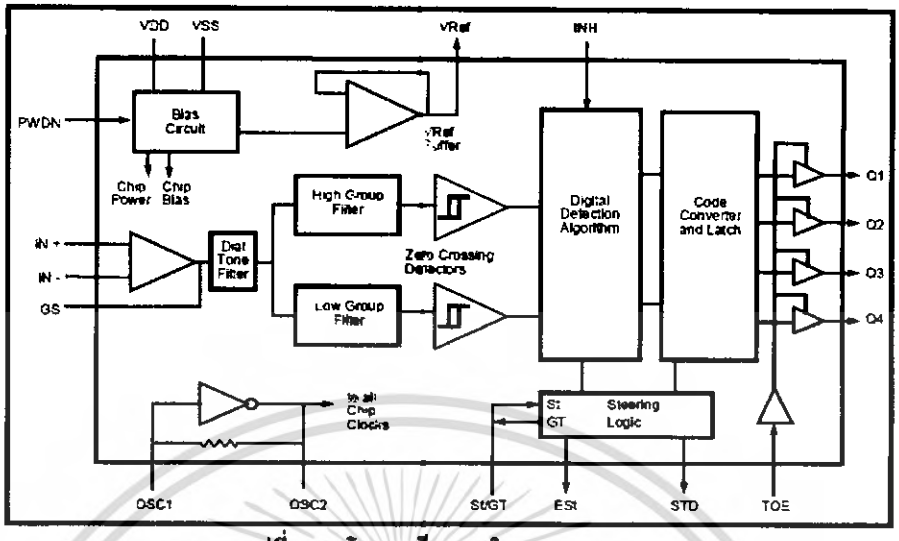
- เป็นตัวรับและถอดรหัสความถี่ (DTMF)
- กินไฟน้อย ใช้ไฟเลี้ยงระดับเดียวกันกับ TTL
- สามารถตั้งอัตราขยายภายในตัวไอซีได้
- สามารถปรับการ์ดไทม์ (Guard time)
- เป็นไอซีคุณภาพสูง

2.1.7.2 โครงสร้างไอซี MT8870

โครงสร้างภายในของ MT8870 ประกอบด้วยวงจรความถี่ และวงจรถอดรหัสทางดิจิตอล เป็นไอซีที่สร้างโดยใช้เทคโนโลยี ISO2-CMOS ในส่วนของวงจรกรองความถี่ใช้เทคนิคของ Switch Capacitor สำหรับการกรองความถี่สูงและกรองความถี่ต่ำ ส่วนวงจรถอดรหัสใช้เทคนิคการนับทางดิจิตอลเพื่อ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตรวจจับและถอดรหัสทั้ง 16 ความถี่ออกเป็นเลขฐานสองขนาด 4 บิต และตรวจสอบช่วงเวลาที่สัญญาณเข้ามา ส่วน Output เป็นวงจรค้าง 3 สถานะ

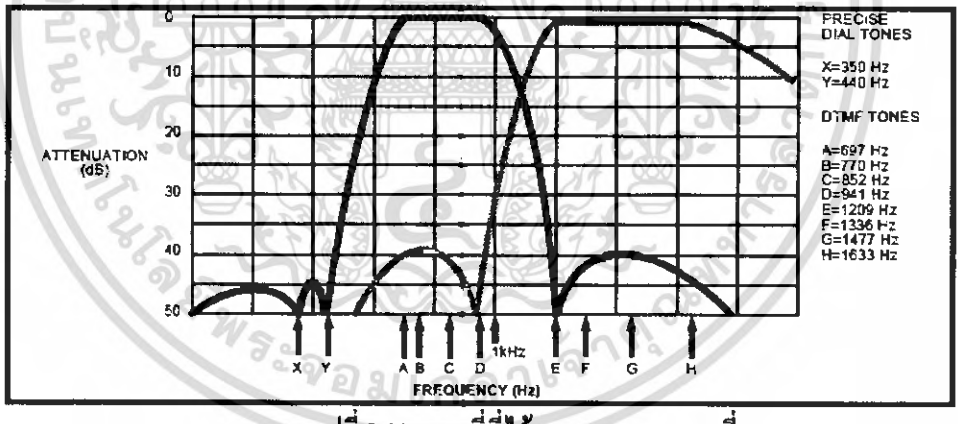


รูปที่ 2.10 โครงสร้างภายในของ MT8870

2.1.7.3 ฟังก์ชั่นการทำงานภายใน MT8870

1. ภาคกรองความถี่

ในส่วนนี้จะแยกสัญญาณ DTMF ที่เข้ามาออกเป็นสองกลุ่มความถี่ คือ ช่วงความถี่สูงและช่วงความถี่ต่ำ โดยใช้วงจรความถี่อันดับ 6 ชนิด ช่วงความถี่ คือ ช่วงความถี่สูง (High Frequency) และช่วงความถี่ต่ำ (Low Frequency)



รูปที่ 2.11 ความถี่ที่ได้จากภาคกรองความถี่

2. ภาคถอดรหัส

ความถี่ DTMF ที่ถูกกรองเรียบร้อยจะผ่านเข้าวงจรถอดรหัสความถี่ออกเป็นตัวเลขโดยใช้เทคนิคการนับแบบดิจิตอล และตรวจสอบความถี่ที่เข้ามาว่าเป็น ความถี่มาตรฐาน DTMF หรือไม่ เพื่อป้องกันความถี่อื่นเข้ามาปน เมื่อตรวจสอบว่าความถี่ถูกต้อง สัญญาณที่เข้าขา EAT (Early Streering) ก็จะทำงาน สำหรับค่าที่ได้จากการถอดรหัสสัญญาณความถี่ต่าง ๆ นั้นแสดงในตารางที่ 2.3

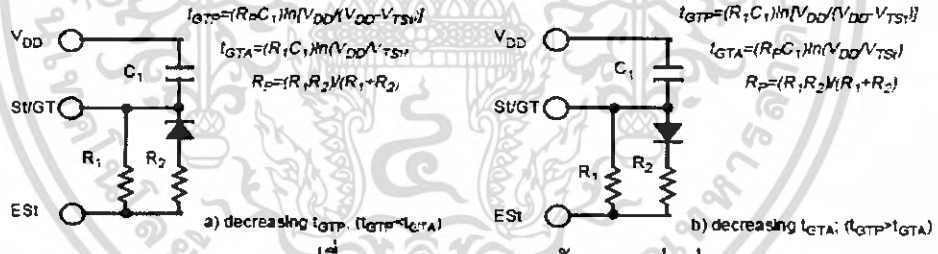
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Digit	TOE	INH	EST	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				

ตารางที่ 2.3 แสดงค่าถอดรหัสที่ได้จากความถี่ต่าง ๆ

3. ภาคตรวจสอบสัญญาณ

ก่อนที่จะมีการถอดรหัสความถี่ไปที่ Output จะมีการตรวจสอบช่วงความถี่ที่เข้ามามีระยะเวลาตามที่กำหนดไว้หรือไม่ โดยสังเกตจากระยะเวลาการกดปุ่มโทรศัพท์ ซึ่งต้องกดปุ่มให้มีความถี่ออกมาให้เป็นเวลาพอสมควร มิฉะนั้นวงจรส่วนนี้จะไม่รับ โดยถือว่าสัญญาณนั้นไม่ถูกต้อง ส่วนช่วงเวลายาวเท่าใดสามารถตั้งได้โดยใช้ RC ภายนอก สัญญาณที่เข้าขา EST จะเป็น 1 นานใกล้เคียงกับระยะเวลาที่มีความถี่ DTMF เข้ามา จากรูปที่ 2.12 เมื่อขา EST เป็น 1 ทำให้ VC สูงขึ้นจนถึงค่าเทรสโฮลด์ วงจรถอดรหัสจึงจะสามารถถอดรหัสออกเป็นตัวเลขขนาด 4 บิต



รูปที่ 2.12 วงจรตรวจสอบสัญญาณอย่างง่าย

อธิบายขั้นตอนการทำงาน

A– ตรวจสอบความถี่ที่เข้ามา แต่คาบเวลาไม่ถูกต้อง Output ไม่เปลี่ยนแปลง

B– ความถี่ # n ถูกตรวจสอบพบ และมีคาบเวลาที่ถูกต้อง ความถี่ที่ถูกถอดรหัสและค้างไว้ที่ Output

C– จบความถี่ # n ช่วงห่างถูกต้อง Output ยังคงค้างจนกว่าจะได้รับความถี่ที่ถูกต้องใหม่

D– Output เปลี่ยนเป็น อิมพีแดนซ์สูง

E– ความถี่ # n+1 ถูกตรวจพบช่วงเวลาที่ถูกต้อง ความถี่ถูกถอดรหัสและค้างไว้

I – ความถี่ # n+1 หายไป ช่วงห่างไม่ถูกต้อง Output ยังคงค้างอยู่

G– ความถี่ # n+1 ช่วงห่างถูกต้อง Output ยังคงค้างอยู่จนถึงความถี่ใหม่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

อธิบายคำศัพท์

Vin – สัญลักษณ์ความถี่ DTMF ที่เข้ามา

Est – Early Streering Output ให้แสดงความถี่ที่ถูกต้อง

St/Gt – Streeing input /Guard Time Output สำหรับความถี่ต่ำกับ RC ภายนอก

StD – Delayed Streeing output ใช้แสดงว่าความถี่ที่ได้รับหรือที่หายไปมีค่าเวลาตามที่กำหนดเพื่อแสดงความถูกต้องของสัญญาณ

TOE – Tone Output Enable (input) ใช้ควบคุม Q1 – Q2 ให้เป็นอิมพีแดนซ์สูง

Tree – คาบเวลาที่นานที่สุดที่ตรวจพบความถี่ DTMF ที่ถูกต้อง 2 สัญญาณ

Tid – เวลาสั้นที่สุดระหว่างสัญญาณ DTMF ที่ถูกต้อง 2 สัญญาณ

Tdo – เวลานานที่สุดที่ยอมให้สัญญาณหายไปในคาบเวลาความถี่ที่ถูกต้อง

Tdp – เวลาที่ใช้ในการตรวจพบความถี่ DTMF ที่ถูกต้อง

Tda – เวลาที่ใช้ในการตรวจการหายไปของสัญญาณความถี่ DTMF ที่ถูกต้อง

Tgtp – ช่วงคาบเวลาของความถี่ของการปรากฏความถี่ DTMF

Tgta – ช่วงคาบเวลาของความถี่การหายไปของความถี่ DTMF

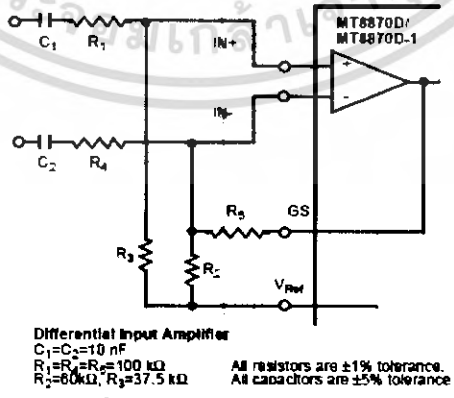
สำหรับคำว่า Guard Time หมายถึง ช่วงคาบเวลาความถี่ที่เข้ามาซึ่งจะต้องนานเท่ากับหรือมากกว่าช่วงเวลาที่ตั้งใจไว้ จึงจะได้การยอมรับว่าสัญญาณความถี่นั้นถูกต้อง หรือพูดว่า เวลาที่เราตั้งไว้โดย RC ก็คือ Guard Time นั่นเอง เมื่อสัญญาณความถี่เข้ามานานเท่ากับหรือมากกว่าเวลาที่ตั้งไว้จึงสามารถแปลงเป็นตัวเลขได้ ถ้าสัญญาณความถี่ที่เข้ามาสั้นกว่าก็จะไม่มีการถอดรหัสเป็นตัวเลขออกไป การตั้งเวลาและการคำนวณดูได้จากรูปที่ 2.12

4. ภาคขยายสัญญาณความแตกต่าง

วงจรส่วน Input ของ MT8870 เป็นภาคขยายสัญญาณจากออปแอมป์ ที่สามารถปรับอัตราการขยายโดยตรงต่อวงจรภายนอกเพิ่มเข้าไป และ แสดงการต่อวงจรภายนอกเข้ากับ Input ซึ่งสามารถคำนวนอัตราการขยายความแตกต่างของ Input และ อิมพีแดนซ์ดังนี้

อัตราการขยาย (Av diff) = R5/R1

อิมพุทอิมพีแดนซ์ (Zin diff) = 2/RI^2 + (1/WC)^2

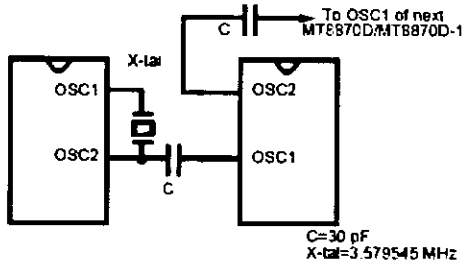


รูปที่ 2.13 การต่อวงจรภาคอินพุท

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5. ภาคกำหนดความถี่

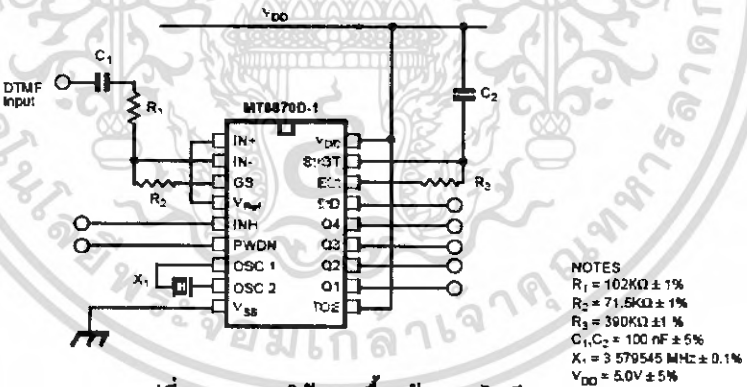
ในภาคนี้จะมีวงจรเวลาอยู่ภายในเพียงต่อแค่ X-TAL ที่มีความถี่ 3.579545 MHz ก็จะสามารถใช้งานได้เลย



รูปที่ 2.14 การต่อวงจรผลิตความถี่

2.1.7.4 การนำ MT8870 ไปใช้งาน

- **นำไปใช้งานด้านรีโมทคอนโทรล**
- **เครื่องป้องกันโทรศัพท์ทางไกล**
- **ใช้งานเกี่ยวกับเครดิตการ์ด**
- **ใช้งานร่วมกับคอมพิวเตอร์**
- **ใช้ในเครื่องชุมสายขนาดย่อม หรือ PABX**
- **ใช้กับงานทางด้านโทรศัพท์ทั่วไป**
- **เครื่องป้องกันขโมย**
- **การควบคุมอุปกรณ์ทางโทรศัพท์**
- **ใช้ทำเครื่องสอบถามทางโทรศัพท์**



รูปที่ 2.15 วงจรใช้งานเบื้องต้นของไอซี MT8870

2.2 ไมโครคอนโทรลเลอร์ตระกูล MCS-51

ปัจจุบันการพัฒนาและการแข่งขันทางด้านเทคโนโลยีผลิตชิ้นส่วนสารกึ่งตัวนำ ที่นำไปสร้างเป็นไอซี มีประสิทธิภาพสูงมากขึ้น และมีเทคโนโลยีที่เกิดจากการผลิตของบริษัทต่าง ๆ ซึ่งส่งผลให้การผลิตชิปมีขนาดเล็กลง แต่มีประสิทธิภาพและคุณสมบัติต่าง ๆ มากขึ้น ไอซีที่ถูกสร้างเป็นแบบ LSI (Large Scale Integrate Circuit) เป็นเทคโนโลยีการสร้างโดยการนำเอาทรานซิสเตอร์จำนวนมากมาสร้างเป็นไอซีดิจิตอลที่ซับซ้อน โดยทำขึ้นเพื่อทำหน้าที่เป็นหน่วยประมวลผลข้อมูลหรือเรียกว่าไมโครโพรเซสเซอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

(Microprocessor) ที่มีคุณสมบัติหลัก คือ การประมวลผลข้อมูลการคำนวณทางคณิตศาสตร์ และลอจิก ถ้าหากมีการติดต่อกับหน่วยความจำที่เป็นแบบรอม หรืออุปกรณ์ภายนอกที่เป็นอินพุท-เอาท์พุท ต้องมีการต่ออุปกรณ์อื่น ๆ ร่วมด้วย เพื่อทำหน้าที่เลือกอุปกรณ์ในการติดต่อหรือวงจรถอดรหัส (Decoder) ซึ่งสามารถทำงานได้ภายใต้การควบคุมของโปรแกรมและในการที่นำไมโครโพรเซสเซอร์มาเป็นตัวประมวลผลกลางมีหน่วยความจำแบบแรม พอร์ตอินพุท และเอาท์พุท เรียกว่าไมโครคอมพิวเตอร์ เป็นความเสี่ยงไม่คุ้มกับการลงทุนหากนำมาใช้ในงานควบคุมขนาดเล็ก และอาจต้องใช้เนื้อที่มาก ในการออกแบบ ดังนั้นการพัฒนาด้านเทคโนโลยีในการสร้างชิป จึงมีการรวบรวมคุณสมบัติที่ต้องการใช้งานมาอยู่ในตัวเดียวกัน คือ มีองค์ประกอบเกือบทุกอย่างของคอมพิวเตอร์อยู่ภายในตัวไอซีที่เรียกว่า ไมโครคอมพิวเตอร์แบบชิปเดียวซึ่งประกอบด้วยอุปกรณ์พื้นฐานเหมือนไมโครคอมพิวเตอร์เช่นหน่วยประมวลผลกลางขนาดเล็ก (8 บิต - 16 บิต) และหน่วยประมวลผล ที่สามารถเข้าข้อมูลแบบบิตหน่วยความจำข้อมูลพื้นฐานแบบแรมขนาด 128 ไบต์ และบรรจุหน่วยความจำโปรแกรมประเภทรอม (บางเบอร์) สามารถใช้งานให้เป็นได้ทั้งอินพุทและเอาท์พุท มีวงจรสื่อสารอนุกรมแบบฟูลดูเพล็กซ์ วงจร Counter/Timer ที่อยู่ภายใน สามารถต่ออุปกรณ์ที่ใช้ในการสร้างวงจรกำเนิดสัญญาณนาฬิกา เช่น คริสตอล (Crystal)และตัวเก็บประจุก็สามารถใช้งานได้เป็นต้น เรียกกันทั่ว ๆ ไปว่า ไอซีไมโครคอนโทรลเลอร์(Microcontroller)

2.2.1 คุณสมบัติทั่วไปของไมโครคอนโทรลเลอร์ตระกูล MCS-51

1. เป็นไมโครคอนโทรลเลอร์ขนาด 8 bit
2. มีวงจรออสซิลเลเตอร์และวงจรผลิตเวลาภายในตัว
3. มีขาสัญญาณ อินพุท เอาท์พุท จำนวน 32 bit
4. สามารถเชื่อมต่อหน่วยความจำข้อมูลภายนอก (External program memory) และสามารถเชื่อมต่อหน่วยความจำโปรแกรมภายนอก (External program memory)
5. มีหน่วยความจำโปรแกรมภายในตัว (On-chip program memory) ขนาด 4 Kbyte โดยเฉพาะเบอร์ 8052 จะมีหน่วยความจำในส่วนนี้ถึง 8 Kbyte สำหรับเบอร์ 8031 และ 8032 จะไม่มีหน่วยความจำในส่วนนี้
6. มีหน่วยความจำข้อมูลภายใน (On-chip data memory) ขนาด 128 byte โดยเฉพาะเบอร์ 8032 และ 8052 จะมีหน่วยความจำในส่วนนี้ถึง 256 byte
7. หน่วยความจำข้อมูลภายในบางส่วนสามารถเข้าถึงข้อมูลระดับบิตได้ด้วย ทำให้การควบคุมหรือการตรวจสอบสถานะบิตได้ง่ายส่งผลให้การเขียนโปรแกรมทำได้ง่ายมากขึ้น
8. มีไทม์เมอร์/เคาท์เตอร์ (Timer / Counter) ขนาด 16 บิต จำนวน 2 ตัว โดยเฉพาะเบอร์ 8032 หรือ 8052 จะมีไทม์เมอร์/เคาท์เตอร์จำนวน 3 ตัว
9. การอินเตอร์รัปต์สามารถทำได้จาก 6 แหล่งกำเนิด โดยการอินเตอร์รัปต์ยังสามารถจัดลำดับความสำคัญได้เป็น 2 ระดับ
10. มีพอร์ทสื่อสารอนุกรมได้ภายในตัวเอง ซึ่งทำงานเป็นแบบฟูลดูเพล็กซ์ (Full Duplex) เลือกรูปแบบได้ 4 โหมด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

11. มีคำสั่งในการคำนวณทางคณิตศาสตร์ และทางตรรกศาสตร์

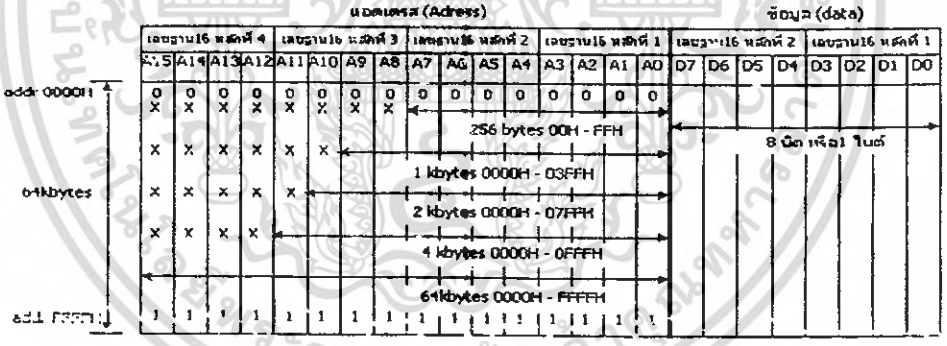
12. คำสั่งส่วนใหญ่ใช้เวลาในการทำงานเพียง 1 ไมโครวินาที เมื่อใช้คริสตอลความถี่ 12 MHz

2.2.2 การจัดหน่วยความจำของไมโครคอนโทรลเลอร์

หน้าที่การทำงานของหน่วยความจำคือ ทำหน้าที่เก็บโปรแกรมคำสั่ง และข้อมูลที่ใช้ในการกำหนดค่าต่างๆให้กับไมโครคอนโทรลเลอร์ หรือใช้เก็บค่าต่างๆที่ไมโครคอนโทรลเลอร์ ได้กระทำตามคำสั่งแล้ว การจัดหน่วยความจำของไมโครคอนโทรลเลอร์ MCS-51 จะแบ่งหน่วยความจำออกเป็น 3 กลุ่ม คือ

2.2.2.1 หน่วยความจำสำหรับเก็บโปรแกรม (Program Memory)

หน่วยความจำสำหรับเก็บโปรแกรม (ทำหน้าที่เช่นเดียวกับรอม) หรือหน่วยความจำรหัสคำสั่ง (Code Memory)จะทำหน้าที่เก็บชุดคำสั่ง เพื่อให้ไมโครคอนโทรลเลอร์ปฏิบัติตามคำสั่งนั้น ยกตัวอย่างเช่น ในขณะที่เปิดเครื่องไมโครเวฟ จะมีการแสดงผลรายการหลักที่หน้าจอ LCD เพื่อคอยให้ป้อนค่าเวลาที่ต้องการจะอุ่นอาหาร คำสั่งที่จอ LCD เพื่อให้ป้อนข้อมูลนั้น จะเขียนคำสั่งอยู่ในส่วนของหน่วยความจำโปรแกรมนั้นเอง ถึงจะเปิดเครื่องไมโครเวฟกี่ครั้ง ก็จะมีการแสดงผลที่ LCD ให้ป้อนค่าเวลาที่ต้องการเหมือนเดิม หน่วยความจำสำหรับเก็บโปรแกรมใน MCS-51 จะแบ่งออกเป็นสองส่วน คือ หน่วยความจำสำหรับโปรแกรมภายในชิป (Internal Program Memory) และหน่วยความจำสำหรับเก็บโปรแกรมภายนอกชิป (External Program Memory) ขนาดของหน่วยความจำสำหรับเก็บโปรแกรมภายในชิปมีได้ตั้งแต่ 0,4,8,16 ถึง 64 กิโลไบต์ขึ้นอยู่กับเบอร์ของชิป



ตารางที่ 2.4 แสดงตำแหน่งของหน่วยความจำสำหรับเก็บโปรแกรม

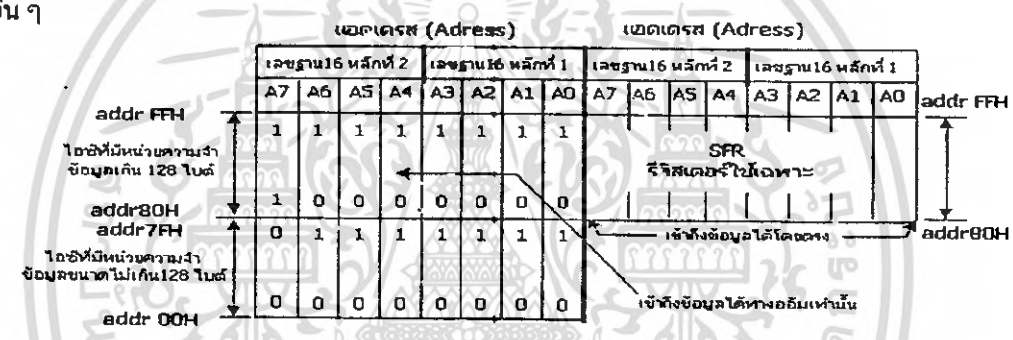
2.2.2.2 หน่วยความจำสำหรับเก็บข้อมูล (Data Memory)

หน่วยความจำข้อมูล (RAM) จะทำหน้าที่เก็บรักษาข้อมูล โดยข้อมูลอาจจะเป็นค่าหลังจากไมโครคอนโทรลเลอร์ ทำการประมวลผล หรือเก็บค่าข้อมูลที่จะให้กับไมโครคอนโทรลเลอร์ประมวลผลในขณะนั้น และจะทำหน้าที่เป็น สแตก (Stack)ยกตัวอย่างเช่น ถ้าเป็นเครื่องไมโครเวฟ ที่ใช้สำหรับอุ่นอาหาร ก็คือ ส่วนที่ป้อนข้อมูล เช่น เวลา หรือ อุณหภูมิที่เป็นปัจจุบันหลังจากหน่วยความจำโปรแกรมแสดงรายการหลักที่ LCD นั่นเอง สังเกตว่าหากปิดเครื่องแล้วเปิดเครื่องใหม่อีกครั้งหนึ่ง ค่าข้อมูลที่เป็นเวลา และ อุณหภูมิเดิมที่กำหนดไว้ในครั้งแรกก็จะหายไป และจะให้ป้อนค่าข้อมูลใหม่อีกครั้งดังนั้นการที่

72983
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จะรักษาข้อมูลเดิมไว้ได้ จะต้องมีแหล่งจ่ายไฟสำรองไว้สำหรับเพื่อเลี้ยงให้กับตัวไอซี ตลอดเวลา หรือ ที่เรียกว่า Battery จะมีหน่วยความจำสำหรับเก็บข้อมูลของ MCS-51 จะแบ่งออกเป็น 2 ส่วน คือ หน่วยความจำสำหรับเก็บข้อมูลภายในชิป และหน่วยความจำสำหรับเก็บข้อมูลนอกชิป ซึ่งแบ่งเป็นส่วนย่อย ๆ ได้ดังนี้

1. พื้นที่ในหน่วยความจำข้อมูล (แรม) ตำแหน่งที่ 00H-1FH จำนวน 32 ไบต์ จะถูกแบ่งออกเป็น 4 กลุ่ม เรียกว่า แบงค์ (Bank) และในแต่ละแบงค์จะมี 8 ไบต์ ดังแสดงในตารางที่ 2.6 พื้นที่ในแต่ละแบงค์จะถูกใช้งานเป็นรีจิสเตอร์ที่ใช้งานทั่วไป (รีจิสเตอร์ R0-R7 เป็นรีจิสเตอร์ที่มีขนาด 8 บิต หรือ 1 ไบต์) โดยที่รีจิสเตอร์ R0 จะอยู่ในตำแหน่งแรกของแต่ละแบงค์และรีจิสเตอร์ R7 จะอยู่ในตำแหน่งสุดท้ายของแต่ละแบงค์ ในการนำไปใช้งานจะเลือกใช้รีจิสเตอร์ R0-R7 ได้เพียงแบงค์เดียวและเลือกใช้พื้นที่ของรีจิสเตอร์ R0-R7 ในแบงค์ใด ๆ ก็ได้ โดยการกำหนดค่าข้อมูลที่รีจิสเตอร์ PSW ในเป็นส่วนของรีจิสเตอร์เฉพาะ (Special Function Register) หากไม่ได้กำหนดค่าใด ๆ เลย เมื่อทำการรีเซตให้กับ ไอซีไมโครคอนโทรลเลอร์ จะถูกกำหนดให้เริ่มต้นใช้งานที่รีจิสเตอร์ R0-R7 ในหน่วยความจำแบงค์ 0 ให้เอง ดังนั้นในการทดลองเริ่มต้นในส่วนแรก ๆ เราจะยังไม่กำหนดค่าใด ๆ ในการเลือกใช้งานรีจิสเตอร์แบงค์อื่น ๆ



ตารางที่ 2.5 ตารางตำแหน่งแอดเดรสของหน่วยความจำข้อมูลภายใน (Internal data memory)

2. พื้นที่ในหน่วยความจำข้อมูลภายใน (แรม) ตำแหน่งแอดเดรสที่ 20H-2FH จำนวน 16 ไบต์ เป็นส่วนที่สามารถใช้งานในลักษณะการเข้าข้อมูลแบบไบต์หรือแบบบิตได้ และสามารถอ้างตำแหน่งแบบบิตได้โดยตรง เพียงแต่ระบุตำแหน่งหรือชื่อของบิตนั้น ๆ ได้ ซึ่งจะมีอยู่ด้วยกันจำนวนทั้งหมด 128 บิต แต่ละบิตจะมีหมายเลขตำแหน่งของบิตคือ 00H-7FH โดยตำแหน่งบิตที่ 00H ก็คือข้อมูลบิตต่ำสุดในตำแหน่งแอดเดรสที่ 20H หรือเราอาจเรียกว่า (20H.1) และตำแหน่งของบิตที่ 7FH คือ ข้อมูลบิตสูงสุดในตำแหน่งแอดเดรสที่ 2FH หรือ เราอาจเรียกว่า (20H.7) การอ้างตำแหน่งแบบบิตจะทำให้โปรแกรมทำงานได้รวดเร็วขึ้น

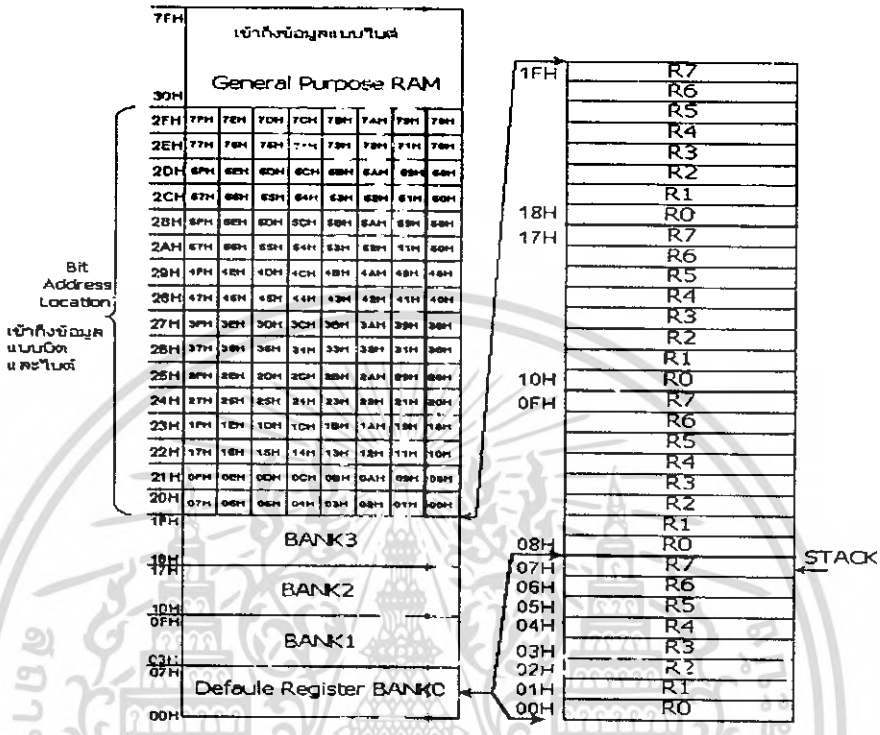
3. พื้นที่บริเวณหน่วยความจำข้อมูลในตำแหน่งที่ 30H-7FH จะเป็นพื้นที่ของหน่วยความจำใช้งานทั่วไป และการติดต่อกับข้อมูลในตำแหน่งต่าง ๆ ของหน่วยความจำส่วนนี้จะอ้างตำแหน่งข้อมูลได้ในลักษณะแบบไบต์เท่านั้น และพื้นที่ส่วนนี้เราอาจจะใช้เป็นสแตกได้

2.2.2.3 รีจิสเตอร์ทำหน้าที่เฉพาะ (Special Function Register)

เนื่องจาก MCS-51 ถูกออกแบบไว้สำหรับใช้ควบคุมระบบโดยเฉพาะ จึงทำให้มีความสามารถเฉพาะตัวหลายอย่าง ซึ่งจำเป็นต้องอาศัยวงจรภายในชิปที่มีเพิ่มขึ้นจากไมโครโปรเซสเซอร์ทั่วไป การ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ควบคุมการทำงานของวงจรภายในไมโครคอนโทรลเลอร์จะกระทำผ่านรีจิสเตอร์ที่ถูกกำหนดหน้าที่ไว้แล้ว ดังนั้นหากต้องการใช้ MCS-51 ให้มีประสิทธิภาพ จำเป็นต้องทราบหน้าที่ของรีจิสเตอร์ใช้งานเฉพาะแต่ละตัวให้ละเอียด รีจิสเตอร์ใช้งานเฉพาะทั้งหมดจะอยู่ในหน่วยความจำสำหรับเก็บข้อมูลภายในชิป บริเวณที่ใช้เป็นรีจิสเตอร์ใช้งานเฉพาะ ดังแสดงในตารางที่ 2.6



ตารางที่ 2.6 แสดงรีจิสเตอร์ที่ใช้งานเฉพาะ

2.2.3 ไทม์เมอร์ / เคาน์เตอร์

ใน MCS-51 มีรีจิสเตอร์ใช้งานเฉพาะที่สามารถนับจำนวนสัญญาณนาฬิกาหรือแมชชีน ไซเคิลของวงจรออสซิเลเตอร์ภายใน (ทำงานเป็นไทม์เมอร์) หรือนับจำนวนครั้งของการเปลี่ยนสถานะของสัญญาณภายนอก (นับจำนวนพัลส์ภายนอก) ที่ขา T0, T1 ของพอร์ทสาม (ทำงานเป็นเคาเตอร์) รีจิสเตอร์ที่ใช้เป็นไทม์เมอร์หรือเคาน์เตอร์ขนาด 16 บิต จำนวนสองตัวคือ รีจิสเตอร์ไทม์เมอร์ 0 และรีจิสเตอร์ ไทม์เมอร์ 1 ตามลำดับ เมื่อต้องการใช้ไทม์เมอร์ 0 หรือ ไทม์เมอร์ 1 และเมื่อนับได้ครบจำนวนที่ตั้งไว้จะมีสัญญาณอินเตอร์รัปต์เพื่อบอกให้ซีพียูทราบ

การควบคุมการทำงานของไทม์เมอร์ หรือเคาน์เตอร์สามารถควบคุมได้จากวงจรภายนอก (ควบคุมด้วยสัญญาณที่ขา INT0, INT1) หรือควบคุมจากคำสั่งในโปรแกรม ดังนั้นรีจิสเตอร์ที่ใช้เป็นไทม์เมอร์ใน MCS-51 จะสามารถจะวัดช่วงห่างของเวลา วัดความกว้างของพัลส์ หรือนับจำนวนครั้งของเหตุการณ์ที่เกิดขึ้นภายนอกที่เปลี่ยนให้อยู่ในรูปของสัญญาณไฟฟ้าแล้ว รวมทั้งใช้กำเนิดสัญญาณอินเตอร์รัปต์ที่มีคาบเวลาที่แน่นอนได้

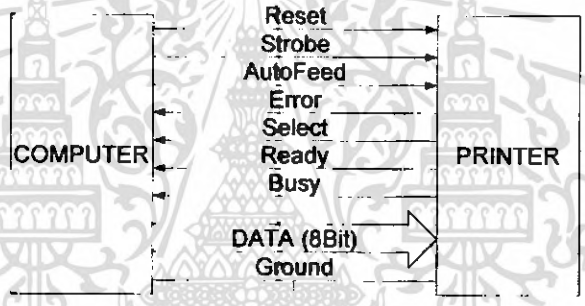
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.3 การเชื่อมต่อระหว่างคอมพิวเตอร์กับพรินเตอร์

การสื่อสารข้อมูลกับพรินเตอร์นั้น คอมพิวเตอร์สามารถส่งข้อมูล 3 ชนิด ไปที่พรินเตอร์ คือ ข้อมูลตัวอักขระ รหัสควบคุม และข้อมูลกราฟฟิก ข้อมูลตัวอักขระแสดงได้ในรูปของตัวอักขระ ตัวเลข เครื่องหมายวรรคตอน สัญลักษณ์ต่าง ๆ รหัสควบคุม จะถูกใช้เพื่อส่งคำสั่งไปยังพรินเตอร์ รหัสเหล่านี้จะกำหนดโหมดการทำงาน เช่น รูปแบบตัวพิมพ์ (Font Style) ขนาดตัวอักษร หรือการควบคุมที่สามารถกระทำได้โดยตรง เช่น การเลื่อนกระดาษทีละบรรทัด หรือ ทีละหน้า การใช้รหัสควบคุม แยกแยะมีความจำเป็นต่อการทำงานขณะที่พิมพ์เอกสาร ซึ่งรหัสควบคุมอื่น ๆ จะทำให้พรินเตอร์ทำงานในโหมดกราฟิกต่าง ๆ ได้

2.3.1 ลักษณะของการสื่อสารข้อมูล

ข้อมูลที่อยู่ในรูปแบบไบนารี่ (Binary) ที่ใช้ถ่ายเทระหว่างคอมพิวเตอร์ และพรินเตอร์ ต้องใช้ตัวกลางในการสื่อสาร คือ สายเคเบิล การผลิตและออกแบบคุณสมบัติของสายเคเบิลจะต้องขึ้นอยู่กับวิธีที่ใช้ติดต่อ ซึ่งมักนิยมใช้อยู่ 2 แบบ คือ แบบอนุกรม และแบบขนาน การติดต่อแบบขนาน (Parallel) เป็นวิธีที่ง่ายที่สุด เนื่องจากทำการส่งข้อมูลตรง ๆ เลย ดังแสดงในรูปที่ 2.16



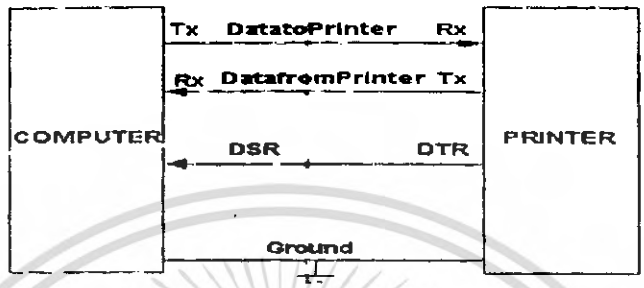
รูปที่ 2.16 แสดงการติดต่อระหว่างคอมพิวเตอร์กับพรินเตอร์แบบขนาน

สังเกตได้ว่า ข้อมูล 8 บิต ส่งแยกบิต D0 กับบิต D7 พร้อม ๆ กัน แต่ถ้าจะส่งได้อย่างถูกต้อง คอมพิวเตอร์และพรินเตอร์ ต้องซิงโครนัส (Synchronous) กัน คือพรินเตอร์จะรับข้อมูลเมื่อมันขอหรือพรินเตอร์สั่งให้คอมพิวเตอร์รอจนกว่าพรินเตอร์จะพร้อมรับส่งข้อมูล ดังนั้น การซิงโครนัส ของการสื่อสารข้อมูลแบบขนาน จำเป็นต้องมีสายควบคุมจำนวนมาก สำหรับสัญญาณพรินเตอร์และสัญญาณคอมพิวเตอร์ การทำงานสัมพันธ์กันดังกล่าว เรียกว่า การแฮนด์เชค (Handshaking) การส่งแบบขนานมีความเร็วพอใช้ได้ โดยพรินเตอร์จะรับส่งข้อมูลเร็วเท่าๆกับที่คอมพิวเตอร์ส่งไป ซึ่งมีความเร็วพอประมาณ 1,000 cps. เมื่อข้อมูลตัวอักขระมี 8 บิต ความเร็วจะมากกว่า 8,000 บิตต่อวินาที บัส (Bus) ข้อมูลภายในพรินเตอร์มี 8 บิต จะสามารถรับข้อมูลในบัฟเฟอร์ (Buffer) ได้โดยเพิ่มในวงจรเล็กน้อยการส่งข้อมูลแบบขนานเป็นการติดต่ออย่างง่าย แต่มีข้อเสียจุดใหญ่ คือ ใช้สายเคเบิล มากแต่มีความเร็วในการส่ง และมีสัญญาณรบกวนทางอิเล็กทรอนิกส์ ซึ่งทำให้เกิดความผิดเพี้ยนของข้อมูลได้

การติดต่อแบบอนุกรมสามารถต่อได้ง่าย เพราะมีจำนวนสายน้อยลง ดังรูปที่ 2.17 แต่การทำงานจะยุ่งยากมาก จะพบว่ารูปที่ 2.17 มีสายสองเส้นที่ใช้ในการส่งข่าวสาร หนึ่งเส้นสำหรับส่งข้อมูลจาก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

คอมพิวเตอร์ไปยังพรินเตอร์ และอีกหนึ่งเส้นสำหรับส่งข้อมูล จากพรินเตอร์ไปยังคอมพิวเตอร์ เพราะว่าข้อมูลสามารถส่งไปส่งมาได้ทั้งสองทิศทาง จึงเรียกการติดต่อสื่อสารแบบนี้ว่าการสื่อสารข้อมูลแบบสองทาง (Bidirectional Data Link) ถ้ามีสายเส้นเดียวจะสามารถส่งหรือรับข้อมูลได้เพียงอย่างใดอย่างหนึ่งในเวลาหนึ่งเท่านั้น ข้อมูลแบบอนุกรมต้องซิงโครนัสระหว่างคอมพิวเตอร์และพรินเตอร์ ดังนั้นต้องเพิ่มซิงโครไนซ์บิตที่จุดเริ่มต้นและจุดสิ้นสุดของแต่ละข้อมูลอักขระที่ส่ง และยังมีพาริตี้บิต (Parity Bit) ที่ต้องรวมอยู่ด้วยสำหรับตรวจสอบบิตผิดพลาด



รูปที่ 2.17 แสดงการติดต่อระหว่างคอมพิวเตอร์กับพรินเตอร์แบบอนุกรม

การทำงานแฮนด์เชคแบบอนุกรมนั้น แบ่งเป็นทั้ง ซอร์ฟแวร์และฮาร์ดแวร์ โดยซอร์ฟแวร์ใช้ในการติดต่อแบบสองทิศทาง เช่น ใช้สั่งพรินเตอร์ให้ส่งรหัสควบคุมไปยังคอมพิวเตอร์ เช่น รหัสสำหรับการแฮนด์เชค "XON" และ "XOFF" ในส่วนของฮาร์ดแวร์ของการแฮนด์เชคนั้น ไม่ได้ใช้ในการส่งข้อมูลจากพรินเตอร์ไปยังคอมพิวเตอร์ สัญญาณแฮนด์เชคที่แจ้งให้คอมพิวเตอร์ที่เพิ่มขึ้น คือ สัญญาณพรินเตอร์ไม่ว่าง ดังนั้น การติดต่อจึงมีมากกว่าหนึ่งสายแฮนด์เชค โดยทั่วไปเรามักจะให้มีการแฮนด์เชคระหว่างพรินเตอร์ค่อนข้างมาก แม้ว่าการทำงานของการสื่อสารข้อมูลแบบอนุกรมจะยุ่งยาก แต่การติดต่อแบบอนุกรมก็เป็นที่นิยมกันมาก เนื่องจากสามารถในการติดต่อแบบสองทิศทาง และสามารถใช้งานได้ในระยะไกล

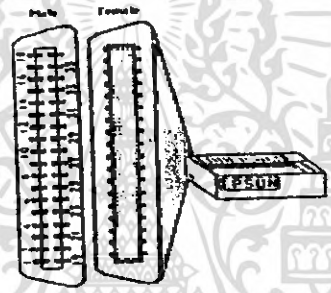
2.3.2 การอินเตอร์เฟสกับพรินเตอร์แบบขนานและแบบเซ็นโทรนิกส์

ข้อมูลที่จะส่งไปยังพรินเตอร์จะส่งในลักษณะของรหัสแอสกี (ASCII) ผ่านสายแบบขนาน 8 สาย พรินเตอร์รับตัวอักษรเพื่อจะพิมพ์และเก็บข้อมูลไว้ในบัฟเฟอร์แรมภายใน เมื่อพรินเตอร์ตรวจพบอักขระแคริเอจรีเทิร์น (Carriage Return: 0DH) มันจะพิมพ์อักขระแถวแรกจากบัฟเฟอร์ เมื่อพรินเตอร์ตรวจพบแคริเอจรีเทิร์น ตัวที่สองมันจะพิมพ์อักขระแถวที่สองออกมา ขบวนการต่าง ๆ จะดำเนินต่อไปจนกระทั่งตัวอักษรที่ต้องการทั้งหมดถูกพิมพ์

การแปลงรหัสแอสกีจากไมโครคอมพิวเตอร์ไปยังพรินเตอร์จะต้องทำบนพื้นฐานการแฮนด์เชค เพราะไมโครคอมพิวเตอร์สามารถส่งตัวอักษรเร็วกว่าความสามารถในการพิมพ์ของพรินเตอร์มาก พรินเตอร์จะต้องมีแนวทางเพื่อที่จะบอกให้ไมโครคอมพิวเตอร์รู้ว่า ขณะนี้บัฟเฟอร์เต็มแล้ว และไม่สามารถรับข้อมูลอักขระได้อีกจนกระทั่งมันพิมพ์ออกมาแล้ว มาตรฐานการอินเตอร์เฟสกับพอร์ทขนานของพรินเตอร์เป็นมาตรฐานเซ็นโทรนิกส์ (Centronics parallel standard) พรินเตอร์ ชนิดเซ็นโทรนิกส์ โดยปกติแล้วจะมี 36 ขา ในการอินเตอร์เฟส (Interface) ตามตารางที่ 2.7 แสดงการกำหนดขาและ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

คำอธิบายสำหรับขาคอนเนคเตอร์ (Connector) การที่มีขาสำหรับการเชื่อมต่อกับพรินเตอร์มากถึง 36 ขา นั้น เนื่องจากสายของแต่ละสัญญาณข้อมูลจะมีสายกราวด์ (Ground) ของมันแยกออกจากกัน เช่น จากตารางที่ 2.7 ขา 2 คือ บิตข้อมูลที่มีนัยสำคัญต่ำสุด (LSB) ของข้อมูลที่จะส่งไปยังพรินเตอร์ และขา 20 จะเป็นกราวด์สำหรับสายสัญญาณนี้ เหตุผลสำหรับแยกสัญญาณกราวด์นี้ เพื่อลดสัญญาณรบกวนในสาย ถ้าจะทำการต่อสายเคเบิลแบบขนานกับพรินเตอร์ สายกราวนด์ของคอมพิวเตอร์ที่ปลายของสายเคเบิล โดยขา 16 เรียกว่า ลอจิกกราวด์ (Logic Ground) และขา 17 เรียกว่า แชสซิสกราวด์ (Chassis Ground) เพื่อที่จะป้องกันกระแสสัญญาณรบกวนที่มาจากสายลอจิกกราวด์ ดังนั้นจึงต้องเชื่อมต่อสายเหล่านี้เข้าด้วยกันกับไมโครคอมพิวเตอร์ ส่วนขาอื่น ๆ ในจำนวน 36 ขาของคอนเนคเตอร์นั้น แบ่งออกได้เป็น 2 ประเภท คือ สัญญาณที่ส่งไปยังพรินเตอร์ เพื่อที่จะบอกว่าอะไรกำลังทำงาน และสัญญาณจากพรินเตอร์ เพื่อที่จะแสดงสถานะของมัน สัญญาณควบคุมหลักไปยังพรินเตอร์ คือ INIT ที่ขา 31 ซึ่งจะบอกพรินเตอร์ เพื่อให้ทำการกำหนดค่าเริ่มต้นภายในลำดับ และ STROBE ที่ขา 1 ซึ่งจะบอกพรินเตอร์ว่าขณะนี้มีข้อมูลส่งมาแล้ว ส่วนอีกสองขาอินพุท คือ ขา 14 และ ขา 36 โดยปกติจะใช้สำหรับดูแลภายในพรินเตอร์โดยทั่วไปคอนเนคเตอร์จะมีลักษณะดังรูปที่ 2.18

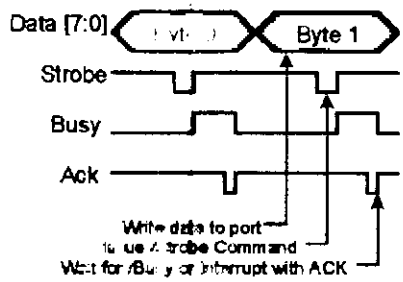


รูปที่ 2.18 แสดงขาของคอนเนคเตอร์แบบแอมฟีนอล (Amphenal)

โดยใช้พื้นฐานของสัญญาณแฮนด์เชค (Handshake Signals) สมมติว่าพรินเตอร์ได้ถูกกำหนดค่าเริ่มต้นแล้ว ขั้นแรกต้องทำการตรวจสอบสัญญาณที่ขา BUSY เพื่อที่จะดูว่าพรินเตอร์พร้อมที่จะรับข้อมูลหรือไม่ ถ้าสัญญาณอยู่ในสถานะ "ต่ำ" แสดงว่าพรินเตอร์พร้อมที่จะรับข้อมูล จึงสามารถส่งรหัสแอสกีบนสายข้อมูลขนานทั้ง 8 สายได้ หลังจากอย่างน้อย 0.5 ไมโครวินาที การแสดงสัญญาณ STROBE จะเป็นสถานะ "ต่ำ" เพราะพรินเตอร์แสดงสัญญาณ BUSY ออกมาเป็นสถานะ "สูง" ได้อีกครั้งหนึ่ง โดยข้อมูลข่าวสารต้องคงสถานะการใช้งานได้บนสายข้อมูลอย่างน้อย 0.5 ไมโครวินาที หลังจากสัญญาณ STROBE สามารถขึ้นอยู่ในสถานะ "สูง"

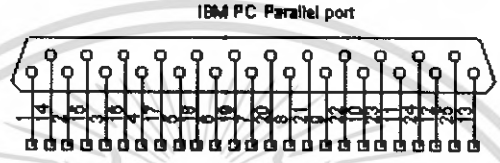
เมื่อพรินเตอร์พร้อมจะรับข้อมูลอักขระถัดไปมันจะแสดงสัญญาณ ACKNLG ในสถานะ "ต่ำ" เป็นเวลา 5 ไมโครวินาที ขอบขาขึ้นของสัญญาณ ACKNLG จะบอกไมโครคอมพิวเตอร์ว่ามันสามารถส่งข้อมูลอักขระไปได้แล้วขอบขาขึ้นของสัญญาณ ACKNLG จะทำการรีเซ็ต สัญญาณ BUSY จะเป็นสถานะ "ต่ำ" เพื่อแสดงว่าพรินเตอร์พร้อมที่จะรับข้อมูลอักขระถัดไป บางระบบจะใช้สัญญาณ ACKNLG สำหรับการแฮนด์เชคและระบบจะใช้สัญญาณ BUSY

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.19 แสดงช่วงเวลาของรูปคลื่นสัญญาณที่จะส่งข้อมูลอักขระไปยังพรินเตอร์แบบขนาน

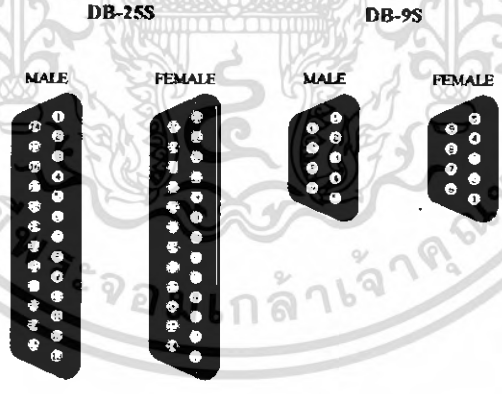
ส่วนทางฝั่งคอมพิวเตอร์ โดยปกติแล้วจะใช้คอนเนคเตอร์ของการต่อขาแบบ DB-25 ซึ่งเป็นตัวผู้ และพอร์ทของคอมพิวเตอร์จะเป็นตัวเมีย ดังนั้น ในการต่อออกจากคอมพิวเตอร์จะต้องคิดด้วยว่าจะต่อแบบใดดังแสดงในรูปที่ 2.20



รูปที่ 2.20 แสดงขาของคอนเนคเตอร์แบบ DB-25

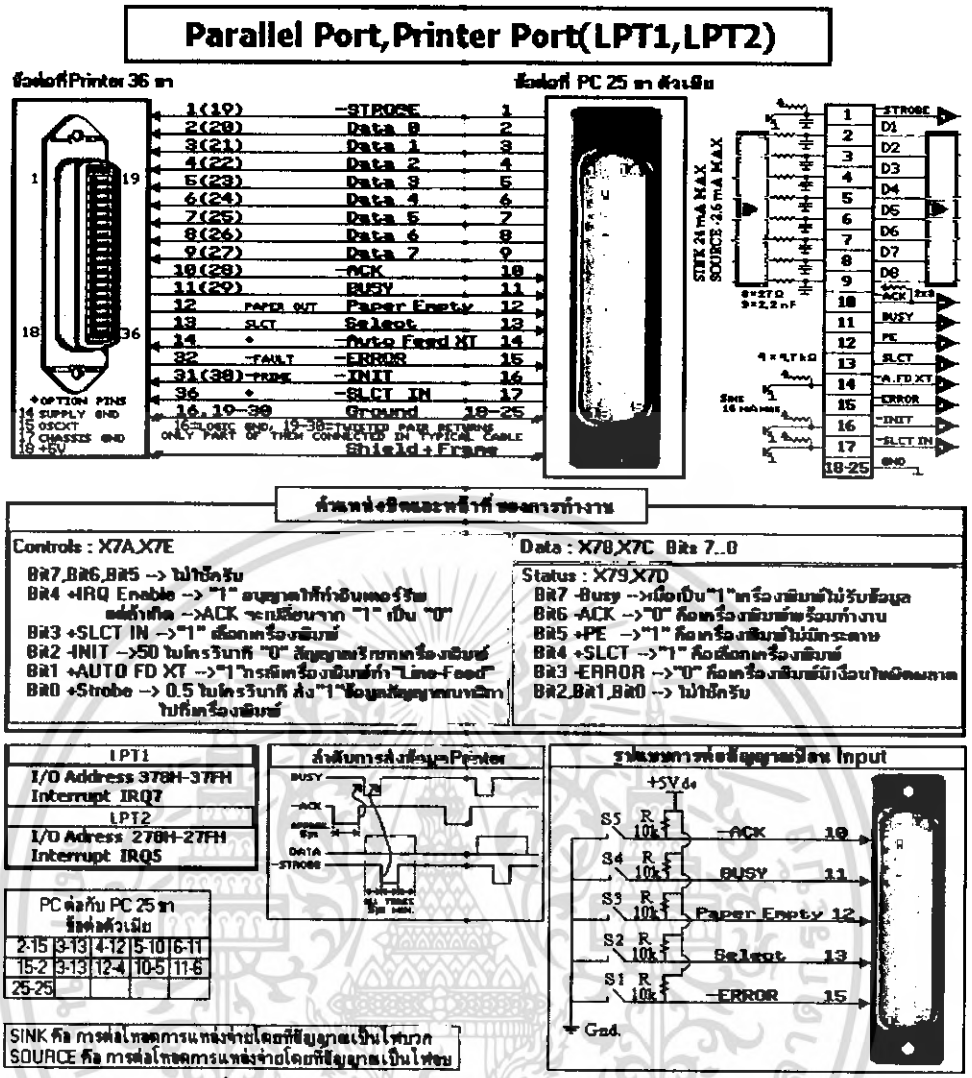
2.3.3 สายสัญญาณและคอนเนคเตอร์

สายสัญญาณเชื่อมโยงระหว่างพอร์ทเครื่องพิมพ์แบบขนานจากคอมพิวเตอร์ไปยังอุปกรณ์ภายนอกด้วยเครื่องพิมพ์แบบขนานจะต่อเป็นคอนเนคเตอร์แบบ D-TYPE 25 และด้านเครื่องพิมพ์หรืออุปกรณ์ภายนอกจะเป็นคอนเนคเตอร์แบบ Centronics ดังแสดงในรูปที่ 2.21



รูปที่ 2.21 ลักษณะการต่อพอร์ทขนาน (D-TYPE)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.22 แสดงการต่อใช้งาน Parallel Port และ Printer Port

โดยรายละเอียดขาของ DB-25 แสดงไว้ในตารางที่ 2.7 แสดงการปรับให้การต่อของ DB-25 กับแอมฟีนอลสัมพันธ์กัน

Pin NO. (D-TYPE 25)	Pin No. (D-TYPE 25)	SPP signal	Direction In / Out	Register
1	1	NStrobe	In/Out	Control
2	2	Data 0	Out	Data
3	3	Data 1	Out	Data
4	4	Data 2	Out	Data
5	5	Data 3	Out	Data
6	6	Data 4	Out	Data

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

7	7	Data 5	Out	Data
8	8	Data 6	Out	Data
9	9	Data 7	Out	Data
10	10	NAck	In	Status
11	11	Busy	In	Status
12	12	Paper-Out / Paper-End	In	Status
13	13	Select	In	Status
14	14	nAuto-Line feed	In/Out	Control
15	32	nError-nFault	In	Status
16	31	nInitialize	In/Out	Control
17	136	nSelect-printer / nSelect-In	In/Out	Control
18-28	19-30	Ground	GND	

ตารางที่ 2.7 แสดงการเชื่อมโยงขาสำหรับการต่อขนานระหว่าง D-TYPE 25 กับ Centronics

2.4 หน่วยแสดงผลแบบ LCD

จากการแสดงผลแบบผลึกเหลว (Liquid Crystal Display: LCD) เป็นอุปกรณ์แสดงผลที่นิยมใช้ในปัจจุบัน เนื่องจากมีความเหมาะสมหลาย ๆ ด้าน เช่น การใช้กระแสไฟฟ้าต่ำ, สามารถแสดงผลอักขระและตัวเลข หรือแสดงเป็น กราฟฟิก (เฉพาะรุ่น) ได้

2.4.1 ส่วนประกอบของ LCD

ในโมดูล LCD จะมีส่วนประกอบหลัก ๆ 3 ส่วน ดังนี้

- ตัวแสดงผล (Display) ภายในเป็นผลึกเหลวที่สามารถแสดงผลให้เห็น โดยอาศัยแสงจากภายนอก ดังนั้น จึงต้องมีมุมในการมองข้อมูลที่แสดงผลบนจอ LCD
- ตัวควบคุม (Controller) เป็นตัวรับข้อมูลจากอุปกรณ์ภายนอกมาควบคุมการทำงานของโมดูล LCD เช่น ลบจอภาพ แสดงตัวอักษรหรือเลื่อนเคอร์เซอร์ เป็นต้น ตัวควบคุมนี้ใช้ชิปควบคุม โดยเฉพาะชิปที่นิยมใช้ คือ เบอร์ HD44780 และ HD61830 โดย HD44780 จะใช้ควบคุม LCD แบบอักขระ ส่วน HD61830 ใช้ควบคุม LCD แบบกราฟิก
- ตัวขับ (Driver) เป็นตัวรับสัญญาณจากตัวควบคุมมาขับให้ตัวแสดงผลแสดงข้อมูลตามที่กำหนด ชิปที่ใช้ทำหน้าที่เป็นตัวขับนี้ ได้แก่ เบอร์ HD4100H และ MSM5259 เป็นต้น

2.4.2 โครงสร้างภายในของตัวควบคุมโมดูล

ในการใช้งานโมดูล LCD จำเป็นต้องทำความเข้าใจเกี่ยวกับโครงสร้างและคำสั่งที่ใช้ในการควบคุมให้ดีเสียก่อน โดยขอยกตัวอย่างโมดูล LCD แบบอักขระ เพราะสามารถเข้าใจได้ง่ายโดยให้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บล็อกไดอะแกรมภายในของชิปควบคุม LCD เบอร์ HD44780 ซึ่งใช้ในโมดูล LCD แบบอักขระ ประกอบด้วย

1. บัฟเฟอร์อินพุท เอาท์พุท เป็นส่วนที่ใช้ในการติดต่อรับส่งข้อมูลกับอุปกรณ์ภายนอก เพื่อที่จะถ่ายทอดข้อมูลเข้าออกภายในตัวควบคุม

2. รีจิสเตอร์คำสั่ง (Instruction Register: IR) เป็นรีจิสเตอร์ใช้รับข้อมูลคำสั่งจากอุปกรณ์ภายนอก เพื่อนำไปควบคุมการแสดงผล

3. รีจิสเตอร์ข้อมูล (Data Register: DR) เป็นรีจิสเตอร์ใช้รับข้อมูลจากอุปกรณ์ภายนอก เพื่อถ่ายทอดไปยังหน่วยความจำที่ทำหน้าที่เก็บข้อมูลแสดงผล หรือนำข้อมูลไปสร้างตัวอักษรเพิ่มเติมในแรมสร้างตัวอักษร

4. แรมเก็บข้อมูลแสดงผล (Display Data RAM: DDRAM) เป็นหน่วยความจำแรม ทำหน้าที่เก็บข้อมูลที่มาจากรีจิสเตอร์ DR ตัวควบคุมจะนำข้อมูลใน DDRAM นี้ไปเปิดตาราง (Look up-table) ของตัวอักษรที่เก็บไว้ในหน่วยความจำรอม และแรมเก็บตัวอักษร เพื่อนำไปแสดงที่ตัวแสดงผล

5. รอมเก็บตัวอักษร (Character Generator ROM: CGROM) เป็นหน่วยความจำรอม ที่เก็บข้อมูลตัวอักษรหรือสัญลักษณ์ที่สามารถแสดงอ่านออกไปตัวแสดงผลมีขนาด 7,200 บิต โดยจะถูกอ่านด้วยค่าของข้อมูลใน DDRAM

6. แรมเก็บตัวอักษร (Character Generator RAM: CGRAM) เป็นหน่วยความจำแรม ที่ใช้เก็บอักษรที่มีการสร้างเพิ่มเติมขึ้นใหม่ ในกรณีที่ตัวอักษรใน CGROM ไม่เพียงพอ มีขนาด 512 บิต การเขียนและการอ่านค่าไปใช้นั้น ทำได้เช่นเดียวกับ CGROM คือ เขียนข้อมูลลงใน DDRAM แล้วตัวควบคุมจะมาอ่านค่าจาก CGRAM เอง

7. แฟลก BUSY เป็นส่วนที่ทำหน้าที่แจ้งสถานะ การทำงานของตัวควบคุมให้อุปกรณ์ภายนอกทราบว่า ตัวควบคุมพร้อมที่จะรับข้อมูลหรือคำสั่งหรือไม่ ดังนั้น ก่อนการส่งข้อมูลหรือคำสั่งมายังตัวควบคุมต้องตรวจสอบสถานะของแฟลก BUSY นี้เสียก่อน

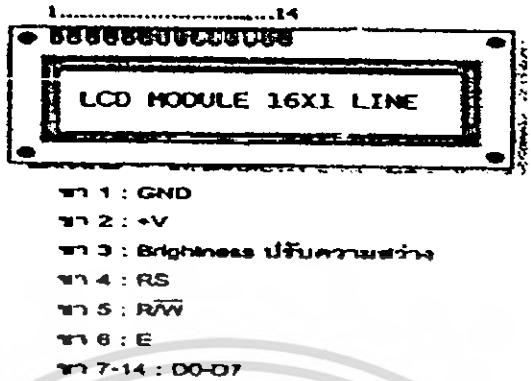
2.4.3 โครงสร้างและขาของ LCD display

โมดูล LCD มีการต่อใช้งานทั้งสิ้น 14 ขา สำหรับรายละเอียดการทำงานของแต่ละขามีดังนี้

- VSS (ขา 1) ต่อกราวด์
- VDD (ขา 2) ต่อไฟเลี้ยง +5V
- VO (ขา 3) เป็นขาอินพุทรับแรงดัน เพื่อปรับความเข้มของการแสดงผล
- RS (ขา 4) เป็นขาอินพุท ใช้ในการแยกข้อมูลที่ทำการประมวลผลในขณะนั้น ว่าเป็นคำสั่งสำหรับรีจิสเตอร์ IR หรือ รีจิสเตอร์ DR โดยถ้าขานี้เป็น "0" ข้อมูลที่ส่งมาจะเป็นคำสั่ง แต่ถ้าขานี้เป็น "1" จะเป็นข้อมูลสำหรับการแสดงผล
- R/W(ขา 5) เป็นขาที่ใช้เลือกการอ่านหรือเขียนข้อมูลกับโมดูล LCD ถ้าเป็น 0 เป็นการกำหนดให้เขียนข้อมูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- E (ขา 6) เป็นขาสำหรับรับสัญญาณพัลส์เอ็นเอเบิลโมดูล LCD ให้ทำงาน
- D0-D7 (ขา 7- ขา 14) เป็นขาที่ใช้เป็นทางผ่านของข้อมูลขนาด 8 บิต



รูปที่ 2.23 รูปร่างและการจัดขาใช้งานของโมดูล LCD แบบอักขระ

RS	R/W	E	การทำงาน
0	0	↓	เขียนคำสั่ง
0	1	_┌┐_	อ่านสถานะของโมดูล LCD
1	0	↓	เขียนข้อมูล
1	1	_┌┐_	อ่านข้อมูล

ตารางที่ 2.8 แสดงความสัมพันธ์ในการทำงานของขา RS, R/W และ E ของโมดูล LCD

2.4.4 การเขียนคำสั่งและข้อมูลให้แก่โมดูล LCD

การควบคุมการทำงานของ LCD โมดูลอาศัยสัญญาณ Enable ซึ่งจะถูกส่งไปยัง LCD โมดูล หลังจากที่มีการกำหนดการทำงานว่าอยู่ในโหมดการอ่านหรือการเขียน ซึ่งการกำหนดโหมดการอ่านการเขียนอยู่ในรูปแบบข้อมูลของตัวอักขระ หรือข้อมูลคำสั่งการกำหนดรูปแบบของข้อมูลทำได้ที่ขาสัญญาณ RS (Register Select) LCD โมดูล โดยถ้า RS = 1 จะเป็นโหมดการอ่านเขียนข้อมูลอักขระ และถ้า RS = 0 จะเป็นโหมดการอ่านเขียนข้อมูลคำสั่งควบคุมภายใน LCD โมดูล

การส่งผ่านข้อมูลทั้งตัวอักขระและคำสั่งควบคุมถูกป้อนเข้ากับบัสข้อมูล 2 ทิศทางของระบบในการใช้งาน LCD โมดูล สามาถใช้งานได้ทั้งแบบ 4 บิต และ 8 บิต ในการเชื่อมต่อแบบ 8 บิต จึงสามารถรับส่งข้อมูลกับบัสของ MCS-51 ได้พอดีและสามารถเขียนโปรแกรมให้ทำงาน รับ-ส่ง ข้อมูลแบบ 8 บิต ได้ แต่ในการใช้งานในรูปแบบ 4 บิต ก็เป็นการประหยัด และสามารถนำพอร์ทที่เหลืออีก 4 พอร์ทไปใช้งานควบคุมอื่น ๆ ได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.5 การเชื่อมต่ออุปกรณ์ระบบบัส I^2C

I^2C ย่อมาจาก Inter-IC Communication หมายถึง การติดต่อสื่อสารระหว่างไอซี โดยบัส I^2C ได้รับการพัฒนาขึ้นโดยฟิลิปส์ (Philips) ด้วยจุดมุ่งหมายหลักคือ ต้องการให้ไอซีหรือโมดูลสามารถติดต่อสั่งงาน และควบคุมภายได้สายสัญญาณเพียง 2 เส้น เส้นหนึ่งคือ สายข้อมูล อีกเส้นหนึ่งคือ สายสัญญาณนาฬิกาที่ใช้ในการกำหนดจังหวะการทำงาน การต่อร่วมกันของอุปกรณ์บนบัส I^2C ทำได้ง่ายมาก เพียงแต่ต่อสายข้อมูลและสายสัญญาณนาฬิกาของอุปกรณ์แต่ละตัวขนานหรือพ่วงกันไป ส่วนการกำหนดแอดเดรสหรือตำแหน่งสำหรับติดต่ออุปกรณ์แต่ละตัว จะใช้รหัสข้อมูลและการกำหนดสภาวะลอจิกที่ขาแอดเดรส ของอุปกรณ์แต่ละตัว

สายข้อมูลบนบัส I^2C มีชื่อเรียกอย่างเป็นทางการว่า สายข้อมูลอนุกรมหรือ SDA (Serial Data Line) ส่วนสายสัญญาณนาฬิกามีชื่อเรียกว่า สายสัญญาณนาฬิกาอนุกรมหรือ SCL (Serial Clock Line)

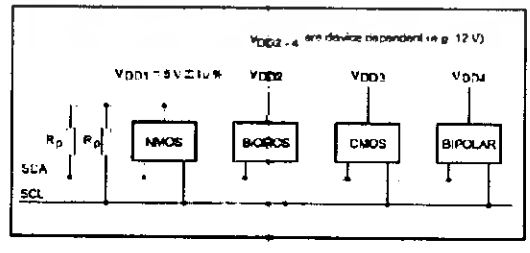
อุปกรณ์ที่ทำการเชื่อมต่อบนบัส I^2C มีหลากหลายไม่ว่าจะเป็น ไอซีขยายพอร์ทอินพุท / เอาท์พุท (I/O Expander) ไอซีแปลงสัญญาณอนาลอกเป็นดิจิตอล (ADC) และแปลงสัญญาณดิจิตอลเป็นอนาลอก (DAC) ไอซีขับโมดูล LCD หน่วยความจำอีพรอม และไมโครคอนโทรลเลอร์

2.5.1 คุณสมบัติโดยทั่วไปของบัส I^2C

สาย SDA และ SCL เป็นสายสัญญาณ 2 ทิศทาง (Bi-directional Line) ต้องมีการต่อตัวต้านทานพูลอัปกับแรงดัน +5V ไว้ตลอดเวลาเพื่อให้สายสัญญาณมีสถานะลอจิกสูงในขณะที่ไม่มีการติดต่อใช้งาน ทั้งยังช่วยในการป้องกันสัญญาณรบกวนที่อาจมีเข้ามาในสายสัญญาณทั้งสอง วงจรเอาท์พุทของอุปกรณ์ที่ต่ออยู่บนบัส I^2C ต้องมีลักษณะเป็นวงจรเดรนเปิด (Open-drain) หรือ คอลเลคเตอร์เปิด (Open-collector)

อัตราการถ่ายทอดข้อมูลลงบนบัสสูงถึง 100 กิโลบิตต่อวินาที ในโหมดปกติ (Standard mode) และสูงถึง 400 กิโลบิตต่อวินาที ในโหมดความเร็วสูง (Fast mode) อุปกรณ์ที่ต่อร่วมอยู่บนบัส I^2C จะต้องมีความจุไฟฟ้าที่เกิดขึ้นระหว่างสาย SDA และ SCL ไม่เกิน 400 pF การเข้าถึงอุปกรณ์ I^2C ใช้ข้อมูลสำหรับการเข้าถึง 2 ค่า คือ 7 บิต (7-bit addressing) หรือ 10 บิต (10-bit addressing)

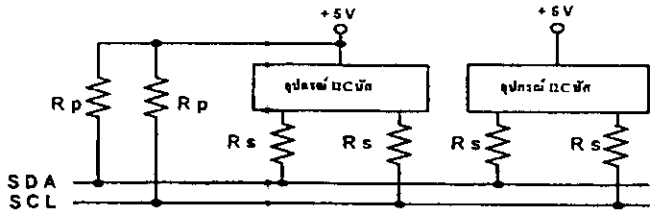
ข้อเด่นอีกประการหนึ่งของบัส I^2C คือ สามารถเชื่อมต่ออุปกรณ์ที่ใช้ไฟเลี้ยงไม่เท่ากันให้สามารถติดต่อสื่อสารกันได้ โดยอุปกรณ์ระบบบัส I^2C ตัวหนึ่งอาจใช้ไฟเลี้ยง +5 V ในขณะที่อีกตัวหนึ่งใช้ไฟเลี้ยง +12 V การต่อร่วมกันบนบัส I^2C สามารถกระทำได้ในลักษณะเดียวกับกรณีที่อุปกรณ์ทั้งสองใช้ไฟเลี้ยงเท่ากัน กล่าวคือ ให้ต่อสาย SDA และ SCL ของอุปกรณ์แต่ละตัวเข้าด้วยกัน และต้องต่อตัวต้านทานพูลอัป (R_p) เข้ากับแรงดัน +5 V ไว้เสมอดังรูปที่ 2.24



รูปที่ 2.24 การต่อตัวต้านทานพูลอัปบนสายสัญญาณในระบบบัส I^2C

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ในกรณีที่อาจมีแรงดันไฟฟ้ากระชากขนาดใหญ่ปะปนเข้ามาในบัส I^2C ที่ขา SDA และ SCL ของอุปกรณ์แต่ละตัวต้องต่อตัวต้านทานอนุกรมกับขา SDA และ SCL เรียกว่า R_S ก่อนต่อเข้าสู่บัส I^2C ดังในรูปที่ 2.25



รูปที่ 2.25 การต่อตัวต้านทาน RS เพื่อลดสัญญาณรบกวนขนาดใหญ่ที่อาจเข้ามาในบัส I^2C

2.5.2 หลักการทำงานของบัส I^2C

บัส I^2C ประกอบด้วยสายสัญญาณ 2 เส้น ดังที่ได้กล่าวมาแล้วคือ SDA และ SCL อุปกรณ์ที่ต่อพ่วงบนบัสสามารถมีได้มากมาย ดังนั้นจึงต้องมีการกำหนดรูปแบบการติดต่อบนบัสหรือเรียกว่าโปรโตคอล (Protocol) เพื่อให้ทราบว่า ขณะนี้ได้ติดต่อกันอยู่ และอุปกรณ์ตัวใดเป็นตัวรับหรือตัวส่ง ต่อไปนี้จะอธิบายลักษณะ หน้าที่ และนิยามอุปกรณ์ที่ต่ออยู่บนบัส I^2C เพื่อเป็นข้อตกลงพื้นฐานการต่อก่อนที่จะอธิบายหลักการทำงานของบัส I^2C ต่อไป

- อุปกรณ์ที่เป็นผู้รับข้อมูล เรียกว่า ตัวรับ (Receiver) อุปกรณ์บนบัส I^2C สามารถเป็นได้ทั้งตัวรับและตัวส่ง บางอุปกรณ์ทำหน้าที่เป็นตัวรับอย่างเดียว จะไม่มีอุปกรณ์ใดบนบัส I^2C ทำหน้าที่เป็นตัวส่งอย่างเดียว
- อุปกรณ์ที่เป็นผู้สร้างข้อมูลหรือส่งข้อมูล เรียกว่า ตัวส่ง (Transmitter)
- อุปกรณ์ที่ทำหน้าที่ควบคุมจังหวะการติดต่อบนบัส I^2C เรียกว่า มาสเตอร์ (Master)
- อุปกรณ์ที่ทำหน้าที่ควบคุมหรืออุปกรณ์ที่ต่อพ่วงเข้าไปบนบัส I^2C เรียกว่า สเลฟ (Slave)

ข้อกำหนด 2 ประการสำคัญของการติดต่อบนบัส I^2C คือ

1. การถ่ายทอดข้อมูลจะเกิดขึ้นได้เมื่อบัสว่าง
2. ในระหว่างการถ่ายทอดข้อมูล เมื่อใดก็ตามที่มีสาย SCL มีสถานะเป็นลอจิกสูง สายข้อมูลต้องรักษาข้อมูลไว้ อย่าให้เกิดการเปลี่ยนแปลงขึ้นเด็ดขาด มิฉะนั้น สัญญาณที่เกิดขึ้นจะได้รับการแปลความหมายเป็นสัญญาณควบคุมแทน

2.5.3 สภาวะที่เกิดขึ้นบนบัส I^2C

มีด้วยกัน 5 สภาวะ ดังนี้

1. **บัสว่าง (Bus not busy)** สภาวะนี้เกิดขึ้นเมื่อสถานะลอจิกบนสาย SDA และ SCL เป็นลอจิกสูงทั้งคู่ นั่นหมายความว่า การถ่ายทอดข้อมูลสามารถเริ่มต้นขึ้นได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

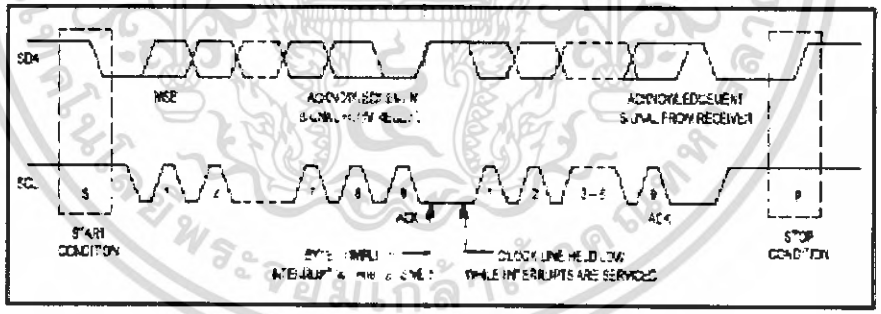
2. **เริ่มต้นการถ่ายทอดข้อมูล (Start data transfer)** เกิดขึ้นเมื่อสาย SDA มีการเปลี่ยนแปลงระดับลอจิกจากสูงไปต่ำ ในขณะที่สาย SCL มีสถานะลอจิกสูง เรียกสภาวะที่เกิดขึ้นนี้ว่า สภาวะเริ่มต้น (START)

3. **หยุดการถ่ายทอดข้อมูล (Stop data transfer)** เกิดขึ้นเมื่อสาย SDA มีการเปลี่ยนแปลงระดับลอจิกต่ำไปสูง ในขณะที่สาย SCL มีสถานะลอจิกสูง เรียกสภาวะที่เกิดขึ้นนี้ว่า สภาวะหยุด (STOP)

4. **ข้อมูลคำรงอยู่บนบัส (Data valid)** สภาวะนี้จะเกิดขึ้นถัดจากสภาวะเริ่มต้น โดยสภาวะลอจิกที่เกิดขึ้นบนสาย SDA ก็คือข้อมูลที่ทำการถ่ายทอด เมื่อสาย SCL เป็นลอจิกสูง สถานะที่สาย SDA ต้องคงที่ เพื่อให้อุปกรณ์รับข้อมูลในจังหวะนั้นว่าเป็น "0" หรือ "1" ข้อมูลอาจเกิดเปลี่ยนแปลงได้ในขณะที่สาย SCL เป็นลอจิกต่ำ แต่เมื่อใดก็ตามที่ต้องการให้เกิดการถ่ายทอดข้อมูลอย่างสมบูรณ์ สถานะลอจิกที่ขา SDA ต้องคงที่ตลอดช่วงเวลาที่สาย SCL มีสถานะลอจิกสูง หากเกิดการเปลี่ยนแปลงสถานะลอจิกในขณะที่สาย SCL มีลอจิกสูงอยู่นั้น อุปกรณ์มาสเตอร์ที่ทำการควบคุมการถ่ายทอดข้อมูลจะแปลความหมายเป็นการหยุดหรือสภาวะเริ่มต้นก็ได้ ทำให้ข้อมูลที่ทำการถ่ายทอดนั้นเกิดความผิดพลาดขึ้น

5. **รับรู้ข้อมูล (Acknowledge)** เกิดขึ้นหลังจากที่การถ่ายทอดข้อมูลจากตัวส่งมายังตัวรับเกิดขึ้นอย่างสมบูรณ์ โดยตัวส่งจะทำการส่งข้อมูลมา 1 **บิต** เรียกว่า บิตรับรู้ (Acknowledge bit) มีสถานะเป็นลอจิกสูง หลังจากส่งข้อมูลครบถ้วน ส่วนอุปกรณ์มาสเตอร์จะทำการส่งสัญญาณรับรู้พิเศษซึ่งสัมพันธ์กับสัญญาณนาฬิกา เพื่อตอบสนองบิตรับรู้ที่ส่งมาจากตัวส่ง ทางด้านรับจะส่งบิตรับรู้ที่มีสถานะลอจิกต่ำลงบนบัส อุปกรณ์สเลฟที่ถูกอ้างอิงถึงในการติดต่อหรือกำลังติดต่ออยู่ในขณะนั้นก็จะกำเนิดบิตรับรู้ เพื่อตอบสนองให้ทราบว่าได้รับข้อมูลในแต่ละไบต์เรียบร้อยแล้ว

ในรูปที่ 2.26 เป็นไดอะแกรมเวลาที่แสดงถึงการเกิดสภาวะต่าง ๆ บนบัส I^2C ไม่ว่าจะเป็นสภาวะบัสว่าง, เริ่มต้น, ถ่ายทอดข้อมูล, รับรู้ และหยุดการถ่ายทอดข้อมูล



รูปที่ 2.26 ไดอะแกรมเวลาแสดงสถานะต่าง ๆ ในบัส I^2C

2.5.4 การทำงานบนบัส I^2C

ก่อนที่จะเริ่มต้นการถ่ายทอดข้อมูลระหว่างอุปกรณ์ต่าง ๆ ที่ต่ออยู่บนบัส ต้องมีการอ้างอิงถึงเสียก่อน โดยการอ้างอิงถึงอุปกรณ์บนบัส I^2C นั้นจะใช้การอ้างอิงถึงแบบ 7 บิต หรือ 10 บิต ในกรณีที่มีอุปกรณ์ต่ออยู่บนบัสไม่มากใช้การอ้างอิงถึงแบบ 7 บิต ก็เพียงพอ แต่ถ้ามีอุปกรณ์ต่ออยู่บนบัสมากกว่า

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

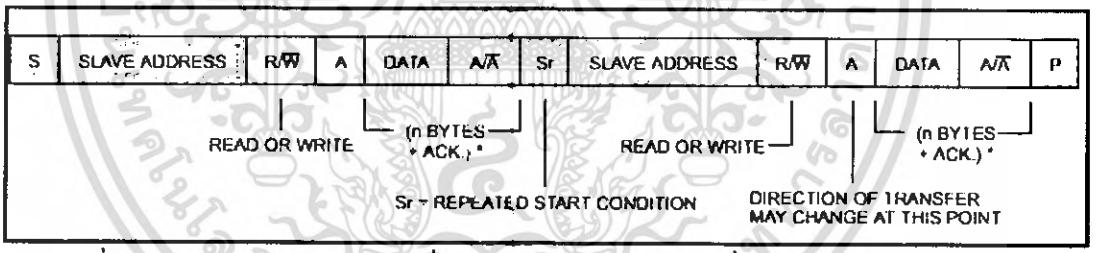
127 แอดเดรส จำเป็นต้องใช้การอ้างอิงแบบ 10 บิต หลังจากที่ติดต่ออุปกรณ์แต่ละตัวได้เรียบร้อยแล้ว ก็จะเริ่มต้นการถ่ายทอดกันต่อไป

2.5.4.1 การอ้างอิงแบบ 7 บิต (7-bit addressing)

ข้อมูลไบต์แรกหลังจากที่สภาวะเริ่มต้น ข้อมูลที่ใช้ในการอ้างอิงอุปกรณ์ที่ต้องการติดต่อ หรือข้อมูลกำหนดแอดเดรส โดยมีรูปแบบแสดงในรูปที่ 2.27 ใน 7 บิตบนรวมทั้งบิต MSB ด้วยจะเป็นข้อมูลแอดเดรสของอุปกรณ์สเลฟที่ต้องการติดต่อ โดยแบ่งเป็น บิตกำหนดแอดเดรสคงที่ (Fixed Address bit) จำนวน 4 บิต ซึ่งข้อมูลนี้อุปกรณ์แต่ละตัวจะถูกกำหนดมาจากผู้ผลิต ไม่สามารถเปลี่ยนแปลงแก้ไขได้ ถัดมาอีก 3 บิต เป็นบิตกำหนดแอดเดรสที่สามารถโปรแกรมได้ (Programmable Address bit) โดยผู้ใช้งานต้องกำหนดสถานะลอจิกให้แก่ขา A0-A2 ของอุปกรณ์ที่มีการเชื่อมต่อแบบบัส I^2C ส่วนในบิต LSB เป็นบิตที่กำหนดการอ่านหรือเขียนข้อมูลกับอุปกรณ์สเลฟตัวนั้น ๆ หากบิต LSB เป็น "0" หมายถึงต้องการเขียนข้อมูลลงไปยังอุปกรณ์นั้น ถ้าเป็น "1" จะเป็นการอ่านข้อมูลจากอุปกรณ์สเลฟ

ข้อมูลในไบต์ต่อมาคือ ข้อมูลควบคุม (Control Byte) ในอุปกรณ์แต่ละตัวมีการกำหนดข้อมูลควบคุมที่แตกต่างกันไป ยกตัวอย่าง ไอซีขยายพอร์ทจะมีข้อมูลควบคุมที่ใช้กำหนดว่า บิตใดเป็นอินพุทบิตใดเป็นเอาท์พุท ในขณะที่ไอซี ACD/DAC ต้องการข้อมูลควบคุมเพื่อกำหนดให้ทำงานเป็นวงจร ADC หรือ DAC เป็นต้น

ข้อมูลไบต์ต่อมาคือ ข้อมูลที่ทำการถ่ายทอดจริง (Data) หลังจากที่มีการถ่ายทอดข้อมูลในแต่ละไบต์ อุปกรณ์สเลฟที่ได้รับการติดต่อต้องส่งสัญญาณรับรู้กลับมาด้วยทุกครั้ง เพื่อให้กระบวนการถ่ายทอดข้อมูลสามารถดำเนินการต่อไปได้ ในรูปที่ 2.27 แสดงรูปแบบข้อมูลอนุกรมที่เกิดขึ้นในการติดต่อบนบัส I^2C ของการอ้างอิงแบบ 7 บิต



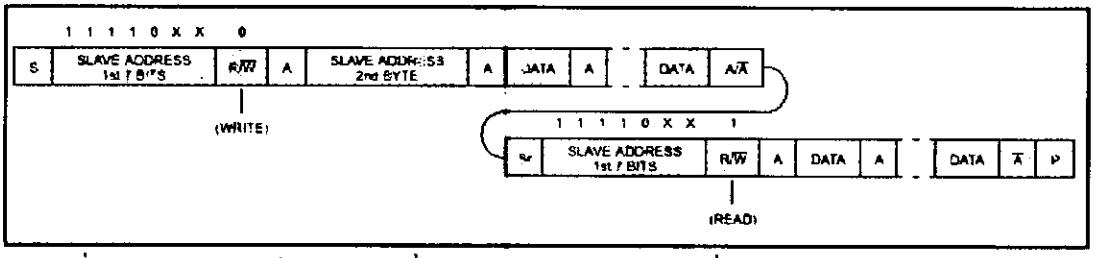
รูปที่ 2.27 รูปแบบของข้อมูลอนุกรมที่ใช้ติดต่อกับอุปกรณ์บัส I^2C เมื่อใช้การอ้างอิงแบบ 7 บิต

2.5.4.2 การอ้างอิงถึงแบบ 10 บิต

ในการอ้างอิงแบบนี้ ยังคงใช้รูปแบบข้อมูลอนุกรมที่เหมือนกับแบบ 7 บิต หากแต่จะมีข้อมูลเพิ่มเติมขึ้นมาเล็กน้อย โดยข้อมูลไบต์แรกหลังจากเกิดสภาวะเริ่มต้น ต้องกำหนดให้ 5 บิต บนเป็นข้อมูล 11110 ส่วนอีก 2 บิต ถัดมาเป็นบิตแอดเดรสของอุปกรณ์สเลฟตัวที่ต้องการติดต่อในบิต LSB ของข้อมูลไบต์แรกยังคงเป็นการกำหนดว่า ต้องการอ่านหรือเขียนข้อมูลกับอุปกรณ์สเลฟตัวที่ต้องการติดต่อด้วย ข้อมูลไบต์ต่อมาเป็นข้อมูลแอดเดรสในไบต์ที่ 2 ของอุปกรณ์ที่ต้องการติดต่อด้วย ข้อมูลไบต์ถัดไปจึงเป็นข้อมูลควบคุม ข้อมูลหลังจากนั้นจะเป็นข้อมูลจริงที่ใช้ในการติดต่อ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เช่นเดียวกับการอ้างอิงแบบ 7 บิตหลังจากถ่ายทอดข้อมูลครบทุกไบต์ ต้องมีสภาวะรับรู้เกิดขึ้น เพื่อให้กระบวนการถ่ายทอดข้อมูลสามารถดำเนินต่อไปได้ ในรูปที่ 2.28 แสดงรูปแบบข้อมูลของการอ้างถึงแบบ 10 บิต



รูปที่ 2.28 รูปแบบของข้อมูลอนุกรมที่ใช้ติดต่อกับอุปกรณ์บัส I^2C เมื่อใช้การอ้างอิงถึงแบบ 10 บิต

2.6 DS1307 ไอซีสร้างฐานเวลาจริงหรือเรียลไทม์คล็อก (RTC)

ผู้ผลิตคือ ดัลลัสเซมิคอนดักเตอร์ (Dallas semiconductor) มีหน้าที่สร้างฐานเวลาจริงให้แก่ระบบไมโครคอนโทรลเลอร์ โดย DS1307 จะใช้ข้อมูลเกี่ยวกับเวลาทั้งหมด ไม่ว่าจะเป็นค่าของเวลาที่ละเอียดของหลักวินาที, นาที, ชั่วโมง, วันที่ (date), วันสัปดาห์ (day), เดือนและปี โดยสามารถปรับวัน เดือน ปีให้ตรงตามปฏิทินได้อย่างถูกต้อง รวมถึงการกำหนดวันในปีอธิกสุรทินด้วย คุณสมบัติทางเทคนิคที่สำคัญมีดังนี้

- เป็นไอซีเรียลไทม์คล็อกให้ข้อมูลตั้งแต่วินาที จนถึงปี รวมถึงการจัดวันในปีอธิกสุรทินด้วย สามารถให้ข้อมูลเวลาได้อย่างเที่ยงตรงถึงปีคริสตศักราช 2100
- มีหน่วยความจำ Nonvolative แรม 56 ไบต์ อยู่ภายในสามารถเก็บข้อมูลทั่วไปได้
- ใช้การเชื่อมต่อแบบระบบบัส I^2C
- มีวงจรตรวจจับไฟเลี้ยงต่ำหรือหายไปอย่างอัตโนมัติ และสามารถรักษาข้อมูลเวลาไว้ได้แม้ไม่มีไฟเลี้ยงไอซี

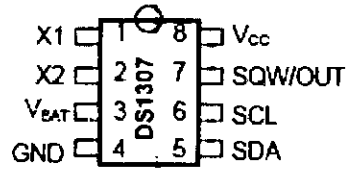
2.6.1 รายละเอียดขาการต่อใช้งานของ DS1307

ในรูปที่ 2.29 แสดงการจัดขาของ DS1307 แต่ละขามีหน้าที่และการใช้งานดังนี้

- Vcc, GND (ขา 8, 4) ต่อไฟเลี้ยง +5 V
- V_{BAT} (ขา 3) ใช้ต่อกับแบตเตอรี่ 3 V เพื่อรักษาการทำงานของวงจรสร้างฐานเวลาของ DS1307 ให้คงอยู่ต่อไปแม้ว่าไม่มีไฟเลี้ยงจ่ายให้แก่ DS1307 ชนิดของแบตเตอรี่ที่เหมาะสมคือ แบตเตอรี่แบบลิเธียม ซึ่งมีความจุ mAhr หรือมากกว่าจะสามารถรักษาข้อมูลได้นาน 10 ปีที่อุณหภูมิ 25 องศาเซลเซียส
- SDA, SCL (ขา 5 และขา 6) เป็นขาสำหรับเชื่อมต่อกับไมโครคอนโทรลเลอร์บนบัส I^2C
- SQW OUT (ขา 7) ที่ขานี้จะมีสัญญาณรูปสี่เหลี่ยมส่งออกมา โดยสามารถเลือกความถี่ได้ 1Hz, 4.096 Hz และ 32 kHz ในการใช้งานต้องต่อตัวต้านทาน Pull up ที่ขานี้ด้วย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

● X1, X2 (ขา 1 และ ขา 2) จะใช้ต่อกับคริสตอลความถี่มาตราฐาน 32.768 kHz เพื่อใช้เป็นฐานเวลาในการสร้างค่าเวลาจริง ในการใช้งานต้องต่อคริสตอลเข้ากับขาทั้งสองนี้และที่แต่ละขาต้องต่อตัวเก็บประจุค่าต่ำ ๆ ประมาณ 15 pF คร่อมกับขากราวนด์ด้วย



รูปที่ 2.29 การจัดขาของไอซี DS1307 ไอซีสร้างฐานเวลาจริง (RTC)

ไอซี DS1307 จัดการเชื่อมต่อในแบบ I^2C โดยจะทำงานเป็นอุปกรณ์สเลฟเสมอ ดังนั้นการติดต่อเพื่อใช้งานจึงต้องกำหนดรูปแบบตามที่กำหนดไว้ในการติดต่อแบบ I^2C ส่วนประกอบหลักที่สำคัญและไดอะแกรมการทำงานของ DS1307 วงจรออสซิเลเตอร์ถือเป็นหัวใจหลักของไอซีเนื่องจากเป็นจุดเริ่มต้นของการสร้างข้อมูลจริงในขณะที่ DS1307 ทำงานที่ขา SQW / OUT จะมีสัญญาณพัลส์สี่เหลี่ยมส่งออกมาตลอดเวลา ในกรณีที่มีการอินาเบิลวงจรกำเนิดสัญญาณพัลส์ที่มีรีจิสเตอร์ควบคุมค่าความถี่ของสัญญาณนี้สามารถเลือกได้ 4 ค่า คือได้ 1 Hz, 4.096 kHz, 8.192 kHz และ 32 kHz พร้อมกันนั้นจะมีการเก็บค่าของเวลาในหน่วยความจำ Nonvolatime แรมซึ่งมีขนาดรวม 64 ไบต์ แต่จัดสรรให้ใช้เก็บข้อมูลเวลา 8 ไบต์ และเป็นหน่วยความจำสำหรับเก็บข้อมูลทั่วไปสำหรับผู้ใช้งานอีก 56 ไบต์

วงจรควบคุมกำลังงานไฟฟ้าจะตรวจสอบสภาวะของไฟเลี้ยงไอซีหากไฟเลี้ยงต่ำกว่า $1.25 \times V_{BAT}$ ก็จะควบคุมให้ DS1307 หยุดการทำงานรีเซ็ตค่าตัวนับแอดเดรสภายใน ทำให้ไม่สามารถติดต่อกับ DS1307 ได้ ดังนั้น ในการใช้งาน DS1307 ต้องระมัดระวังอย่าให้ไฟเลี้ยงต่ำกว่า $1.25 \times V_{BAT}$ หรือประมาณ 3.75 V ในกรณีที่ใช้ V_{BAT} เท่ากับ 3 V ถ้าหากไฟเลี้ยงมีค่าต่ำกว่า $1.25 \times V_{BAT}$ ไอซี DS1307 จะเข้าสู่โหมดสำรองข้อมูลกระแสต่ำทันทีจะไม่มีการส่งสัญญาณพัลส์ออกมาที่ขา SQW / OUT แต่วงจรสร้างฐานเวลายังคงทำงานเพื่อให้ค่าของเวลาเดินไปอย่างไม่ผิดพลาด เมื่อมีไฟเลี้ยงปรากฏขึ้นอีกครั้ง DS1307 ก็จะสามารถให้ค่าเวลาที่แท้จริงแก่ผู้ใช้งานได้ต่อไป

วงจรสื่อสารอนุกรมภายใน DS1307 ได้กำหนดให้ทำงานในรูปแบบของบัส I^2C เป็นช่องทางการสื่อสารระหว่าง DS1307 กับอุปกรณ์มาสเตอร์ผู้ใช้งานสามารถเข้าสู่หน่วยความจำที่ใช้เก็บค่าเวลาและหน่วยความจำใช้งานทั่วไปได้โดยการเขียนข้อมูลตามรูปแบบที่กำหนดในระบบบัส I^2C

2.6.2 การจัดสรรหน่วยความจำใน DS1307

ในรูปที่ 2.30 แสดงการจัดสรรพื้นที่ของหน่วยความจำภายใน DS1307 พื้นที่ 7 ไบต์แรกตั้งแต่แอดเดรส 00H-06H เป็นพื้นที่ของรีจิสเตอร์ค่าเวลาในการเก็บข้อมูลเกี่ยวกับเวลา ไบต์ต่อมาที่แอดเดรส 07H เป็นพื้นที่ของรีจิสเตอร์ควบคุมการทำงานของ DS1307 ในรูปที่ 2.30 แสดงรายละเอียดของรีจิสเตอร์ค่าเวลาและรีจิสเตอร์ควบคุมของ DS1307

ด้วยการจัดสรรพื้นที่แบบนี้ ทำให้ผู้ใช้งานสามารถเรียกข้อมูลเวลาออกมาได้ตามที่ต้องการโดยไม่จำเป็นต้องอ่านออกมาได้ทั้งหมดได้ก็ได้ ค่าของเวลาทั้งหมดจะอยู่ในรูปเลขฐานสิบ สำหรับการแสดง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ในรูปของชั่วโมงสามารถเลือกได้ว่าต้องการแบบ 12 หรือ 24 ชั่วโมง โดยกำหนดบิตที่ 6 ของแอดเดรส 02H และเมื่อเลือก 12 ชั่วโมงที่บิต 5 ในแอดเดรสเดียวกันใช้ในการแสดงค่า AM/PM โดยถ้าบิตนี้เป็น "1" หมายถึง ค่าชั่วโมงในขณะนี้เป็นช่วงเวลาหลังเที่ยงวัน ในกรณีที่เป็น 24 ชั่วโมง บิตนี้จะใช้ในการแสดงค่า 2 ของหลักสิบในหน่วยชั่วโมง

		บิต 7	บิต 6	บิต 5	บิต 4	บิต 3	บิต 2	บิต 1	บิต 0	ค่าของข้อมูล
00H	วินาที	CH	ข้อมูลวินาที (หลักสิบ)			ข้อมูลวินาที (หลักหน่วย)				00-59
	นาที	X	ข้อมูลนาที (หลักสิบ)			ข้อมูลนาที (หลักหน่วย)				00-59
	ชั่วโมง	X	12 ชม./ 24 ชม.	ค่าของ(หลักสิบ) AM/PM	ข้อมูลชั่วโมง (หลักสิบ)	ข้อมูลชั่วโมง (หลักหน่วย)				01-12 00-23
	วัน	X	X	X	X	X	ข้อมูลวันในสัปดาห์			1-7.
	วันที่	X	X	ข้อมูลวันที่ (หลักสิบ)		ข้อมูลวันที่ (หลักหน่วย)				01-28/29 01-30 01-31
	เดือน	X			ข้อมูลเดือน (หลักสิบ)	ข้อมูลเดือน(หลักหน่วย)				01-12.
	ปี	ข้อมูลปี (หลักสิบ)				ข้อมูลปี (หลักหน่วย)				00-99
07H	รีจิสเตอร์ควบคุม	OUT	X	X	SQWE	X	X	RS1	RS0	
08H	แรม 56 ไบต์									
3FH										

รูปที่ 2.30 การจัดสรรหน่วยความจำแรมภายในพร้อมกับรายละเอียดของรีจิสเตอร์เก็บค่าเวลาและรีจิสเตอร์ควบคุมของ DS1307

2.6.3 รีจิสเตอร์ควบคุม

มีแอดเดรสอยู่ที่ 07H โดยมีรายละเอียดของบิตดังนี้

OUT (Output control): ใช้ในการควบคุมระดับลอจิกที่ขา SQW / OUT ในกรณีที่ดิสเอเบิลการกำเนิดสัญญาณสี่เหลี่ยม โดยถ้าบิตนี้เป็น "1" ที่ขา SQW / OUT ก็จะเป็น "1" ถ้าบิตนี้เป็น "0" ที่ขา SQW / OUT ก็จะเป็น "0"

SQWE (Square Wave Enable): ใช้ในการเอ็นเอเบิลวงจรกำเนิดสัญญาณสี่เหลี่ยมที่ขา SQW / OUT ถ้าต้องการให้มีสัญญาณสี่เหลี่ยมออกมาให้กำหนดบิตเป็น "1"

RS1, RS0 (Rate Select): ใช้ในการเลือกสัญญาณความถี่ของสัญญาณสี่เหลี่ยมที่ออกจากขา SQW / OUT มีรายละเอียดดังต่อไปนี้

RS1	RS0	ค่าความถี่ของสัญญาณสี่เหลี่ยม
0	0	1 Hz
0	1	4.096 Hz
1	0	8.192 Hz
1	1	32.768 Hz

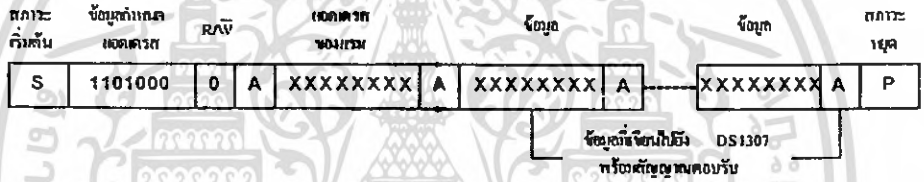
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.6.4 โหมดการทำงานของ DS1307

มีด้วยกัน 2 โหมด คือ โหมดเขียนข้อมูลและโหมดอ่านข้อมูล ในการใช้งาน DS1307 ตามปกติจะใช้งานเฉพาะโหมดอ่านข้อมูลเท่านั้น เนื่องจากไมโครคอนโทรลเลอร์จะติดต่อกับ DS1307 เพื่ออ่านข้อมูลของเวลาไปใช้งาน โหมดการเขียนข้อมูลจะถูกใช้งานก็ต่อเมื่อต้องการตั้งค่าเวลาใหม่และต้องการเขียนข้อมูลลงไปในหน่วยความจำใช้งานทั่วไป อย่างไรก็ตามเมื่อเริ่มต้นติดต่อกับ DS1307 จำเป็นอย่างยิ่งที่จะต้องเข้าสู่โหมดการเขียนข้อมูลก่อน เพื่อกำหนดแอดเดรสที่ต้องการอ่านข้อมูลจากนั้นจึงเปลี่ยนโหมดการทำงานมาเป็นโหมดการอ่านข้อมูล

2.6.4.1 โหมดการเขียนข้อมูล

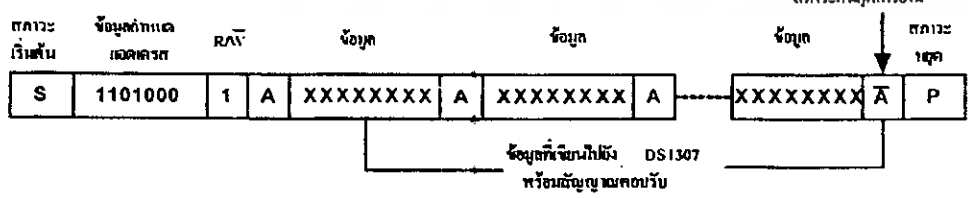
มีรูปแบบดังรูปที่ 2.31 เริ่มต้นเมื่อไมโครคอนโทรลเลอร์จะกำหนดสภาวะเริ่มต้น (START:S) จากนั้นส่งข้อมูลกำหนดแอดเดรส 1101000 ตามด้วยข้อมูลเลือกการเขียน นั่นคือค่า "0" จากนั้นจะรอการตอบรับจาก DS1307 ขั้นตอนต่อมาคือ ส่งข้อมูลเพื่อเลือกแอดเดรสที่ต้องการเขียนจากนั้นรอการตอบรับจาก DS1307 เมื่อมีการตอบรับมาเรียบร้อยก็เริ่มทยอยเขียนข้อมูลลงไปครั้งละแอดเดรสหลังจากเขียนข้อมูลในแต่ละแอดเดรส จะต้องหยุดรอการตอบรับจาก DS1307 ทุกครั้งจึงสามารถเขียนข้อมูลต่อไปได้ เมื่อเขียนเรียบร้อยแล้วให้ส่งสภาวะหยุด (STOP:P) เป็นอันสิ้นสุดกระบวนการเขียนข้อมูล



รูปที่ 2.31 รูปแบบของข้อมูลสำหรับติดต่อกับ DS1307 ในโหมดการเขียนข้อมูล

2.6.4.2 โหมดการอ่านข้อมูล

มีรูปแบบการทำงานแสดงดังในรูปที่ 2.32 โดยเริ่มต้นการทำงานเหมือนกับโหมดการเขียนข้อมูล คือ ไมโครคอนโทรลเลอร์กำหนดสภาวะเริ่มต้นแล้วส่งข้อมูลกำหนดแอดเดรส ตามด้วยข้อมูลเลือกการอ่านซึ่งเท่ากับ "1" จากนั้นรอการตอบรับจาก DS1307 เมื่อตอบรับเรียบร้อย DS1307 จะทยอยส่งข้อมูลออกมาให้กับไมโครคอนโทรลเลอร์ คราวละ 1 แอดเดรส หรือ 1 ไบต์ โดยแอดเดรสที่เลือกอ่านข้อมูลจะต้องมีการกำหนดมาก่อนล่วงหน้าด้วยโหมดการเขียนข้อมูล วิธีการง่าย ๆ คือ เข้าสู่โหมดการเขียนข้อมูลก่อน เมื่อถึงจังหวะที่ต้องเขียนข้อมูลให้ทำการสร้างสภาวะเริ่มต้น และส่งข้อมูลกำหนดแอดเดรสใหม่อีกครั้ง ตามด้วยโหมดการเลือกการอ่านข้อมูล ข้อมูลที่ออกมาจาก DS1307 จะเป็นข้อมูลจากแอดเดรสที่กำหนดไว้ก่อนหน้านี้

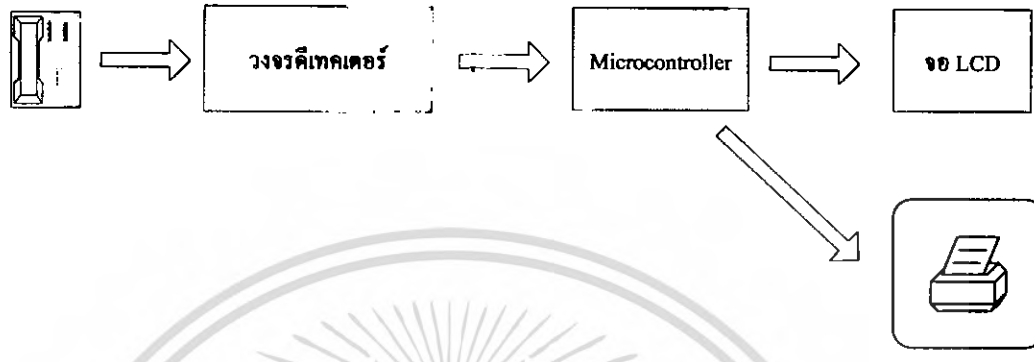


รูปที่ 2.32 รูปแบบของข้อมูลสำหรับติดต่อกับ DS1307 ในโหมดการอ่านข้อมูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 3
การออกแบบและการสร้าง

3.1 การทำงานโดยรวมของระบบ



รูปที่ 3.1 แสดง Block Diagram รวมของระบบ

จากรูปที่ 3.1 เป็น Block Diagram อธิบายการทำงานโดยรวมในโครงงานนี้ เริ่มต้น จากวงจรดีเทคเตอร์ซึ่งจะทำการดีเทคสัญญาณต่าง ๆ ในสายโทรศัพท์ เมื่อได้สัญญาณที่ต้องการนำไปใช้งานแล้วเราก็จะทำการส่งสัญญาณเหล่านี้ไปยังวงจรควบคุมเพื่อทำการประมวลผลและแสดงผลที่จอ LCD หรือในการสั่งพิมพ์ข้อมูลด้วยพรินเตอร์

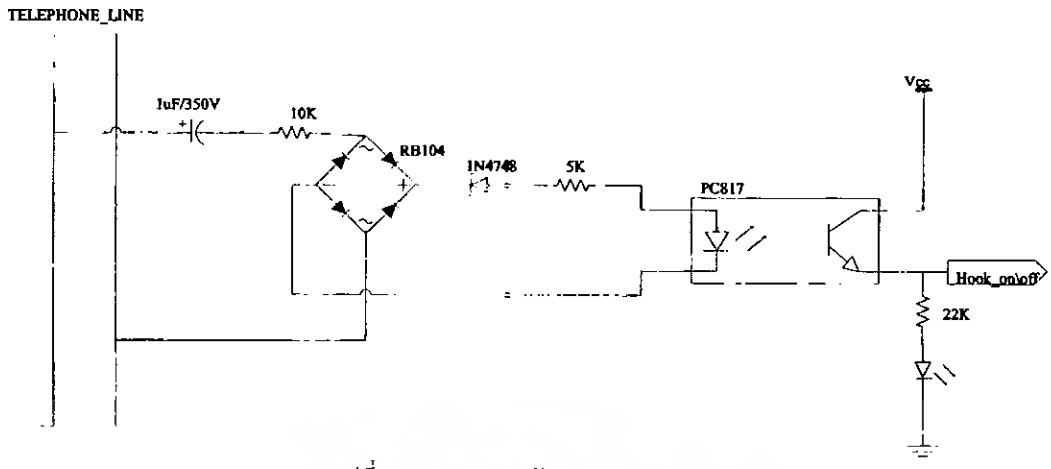
3.2 วงจรดีเทคเตอร์

ในส่วนของวงจรดีเทคเตอร์ประกอบด้วยวงจรย่อย 5 วงจร ได้แก่ วงจรตรวจจับการวางหูและการยกหูโทรศัพท์, วงจรตรวจจับสัญญาณเรียก, วงจรถอดรหัสหมายเลขโทรศัพท์ (DTMF Decoder), วงจรตรวจจับการเรียกกลับและสัญญาณไม่ว่าง และวงจร Caller ID ซึ่งวงจรเหล่านี้จะต่ออยู่กับคู่สายโทรศัพท์ของเครื่องที่ต้องการแสดงข้อมูลการใช้งานโทรศัพท์ เพื่อรับสัญญาณอินพุทจากคู่สายโทรศัพท์นั้น และเอาท์พุทของวงจรจะส่งไปยังส่วนควบคุมการทำงาน

3.2.1 การออกแบบวงจรตรวจสอบการวางหูและยกหูโทรศัพท์

การทำงานของวงจรในสภาวะปกติสายโทรศัพท์จะมีแรงดันประมาณ 48 โวลต์ เมื่อยกหูจะมีแรงดันลดลงเหลือประมาณ 6-8 โวลต์ ทั้งนี้ขนาดของแรงดันจะขึ้นอยู่กับอิมพีแดนซ์ของเครื่องโทรศัพท์ ซึ่งวงจรที่ได้ทำการทดลองตรวจจับแรงดันของการยกหู/วางหู แสดงดังรูปที่ 3.2

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.2 วงจรตรวจจับการยกหู / วางหู

จากวงจรในรูปที่ 3.2 ใช้ในการปรียบเทียบแรงดันระหว่างการยกหูและการวางหู ซึ่งในขณะที่วางหูอยู่นั้นจะมีแรงดันตกคร่อมคู่สายโทรศัพท์ประมาณ 48 โวลท์ ซึ่งก็แล้วแต่ค่าความต้านทานของเครื่องโทรศัพท์แต่ละเครื่อง พิจารณาในขณะทำการวางหู แรงดันไฟ 48 โวลท์ จะผ่านวงจรบริดจ์เรคติไฟเออร์ซึ่งจะให้ขั้วแรงดันแน่นอน แรงดันดังล่าวมีค่ามากกว่าแรงดัน Zener ของซีเนอร์ไดโอดขนาด 24 โวลท์ เป็นผลให้มีกระแสไหลผ่านความต้านทาน 5 K จึงทำให้มีกระแสไหลผ่าน LED ในออปโตไอโซเลเตอร์และไปกระตุ้นให้ทรานซิสเตอร์นำกระแสตามไปด้วย เกิดแรงดันตกคร่อมตัวต้านทาน 1 K ที่เอาท์พุตเป็น 5 โวลท์ ซึ่งแทนด้วยระดับลอจิก "1" พิจารณาในขณะที่ทำการยกหู แรงดันจะตกลงเหลือประมาณ 6-9 โวลท์ แรงดันดังกล่าวมีค่าน้อยกว่าแรงดันซีเนอร์ จึงไม่มีกระแสไหลผ่านซีเนอร์ไปได้ เป็นผลให้ LED และทรานซิสเตอร์ในออปโตไอโซเลเตอร์ไม่นำกระแส แรงดันเอาท์พุตจึงตกลงเป็น 0 โวลท์ ซึ่งแทนด้วยระดับลอจิก "1" ซึ่งผลการวัดแรงดันเป็นดังตาราง 3.1

สถานะ	แรงดันที่ขา C ของ Q1
ก่อนยกหู	5 V
หลังยกหู	0 V

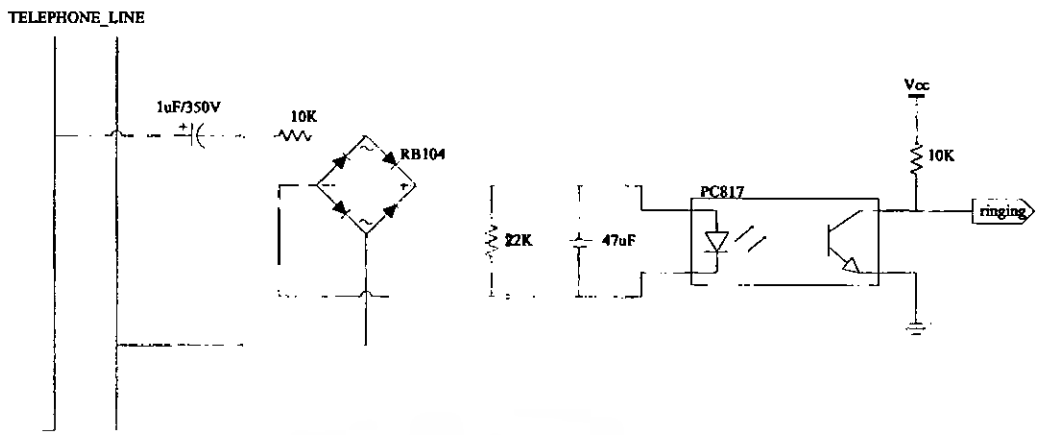
ตารางที่ 3.1 แสดงสถานะของระดับสัญญาณเมื่อยกหูโทรศัพท์

3.2.2 วงจรตรวจจับสัญญาณเรียก (Ringing Detector)

เป็นสัญญาณกระดิ่งหรือสัญญาณเรียกเข้าดังประมาณ 1 ถึง 2 วินาที หยุดประมาณ 3 ถึง 4 วินาที ซึ่งเป็นแรงดันไฟสลับประมาณ 40 ถึง 150 โวลต์ ความถี่ 20 ถึง 40 เฮิรตซ์ สำหรับแถบอเมริกาและ 15 ถึง 68 เฮิรตซ์ สำหรับแถบยุโรป สำหรับความถี่มาตรฐานที่ใช้งานกันก็คือ 20 เฮิรตซ์ สำหรับแถบยุโรป

การตรวจสอบสัญญาณชนิดนี้สามารถทำได้ โดยการตรวจจับการเกิดแรงดันไฟสลับที่เกิดขึ้นชั่วขณะ หนึ่ง ซึ่งก็คือสัญญาณกระดิ่งนั่นเอง โดยวงจรที่แสดงดังรูปที่ 3.3

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

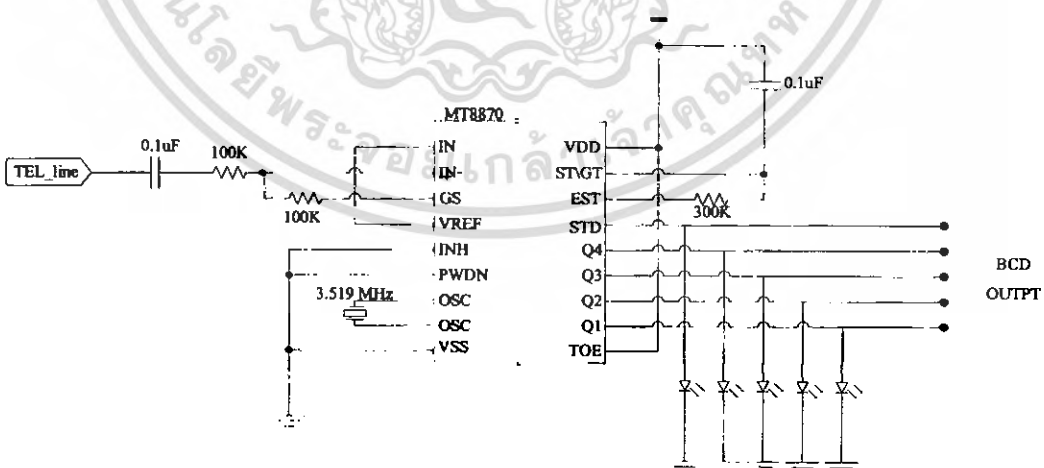


รูปที่ 3.3 วงจรตรวจจับสัญญาณกระดิ่ง

จากรูปที่ 3.3 ตัวเก็บประจุ C1 ทำหน้าที่กั้นแรงดันไฟตรงจากคู่สายโทรศัพท์ไม่ให้ผ่านเข้าไปสู่วงจรได้ R1 ทำหน้าที่ลดขนาดของแรงดันของสัญญาณกระดิ่งที่เข้ามา R2 ร่วมกันทำหน้าที่เป็นวงจร Divider หรือลดขนาดแรงดันไฟตรงที่เข้ามาทาง BD1 ให้ต่ำลง จากนั้น C2 ก็จะทำการกรองให้แรงดันไฟตรงนั้นเรียบยิ่งขึ้น ซึ่งสามารถจะนำไปประยุกต์ใช้งานได้ทั้ง 2 ชุด

ขณะที่มีสัญญาณกระดิ่งเข้ามา แรงดันที่เข้ามาก็จะถูกลดทอนโดย C1, R1 ผ่าน BD1 เพื่อเปลี่ยนเป็นแรงดันไฟตรง จากนั้นก็จะถูกลดโดย อีกครั้ง แรงดันไฟตรงที่ถูกลดแล้วก็จะป้อนเข้าขาแอโนดของ LED ภายในออปโตไอโซเลเตอร์ ที่ขา 1 เมื่อ LED ภายในทำงานก็จะส่งผลให้ทรานซิสเตอร์ภายในทำงานด้วยเช่นกัน ทำให้ขา 5 ของออปโตไอโซเลเตอร์หรือ ขา C ของทรานซิสเตอร์ภายในถูกต่อลงกราวด์ แรงดันที่ขา 5 ก็จะลดลงเกือบ 0 โวลต์ เกิดเป็นพัลส์ขอบขาลดลง ส่งผลให้ขา B ของ U1 ถูกต่อลงกราวด์ด้วยเช่นกัน U1 จึงทำงาน ที่ขา U1 จะมีแรงดันตกคร่อม R6 เกิดเป็นพัลส์ขอบขาขึ้น เมื่อมีสัญญาณเรียกเข้ามาที่ขา 5 ของออปโตไอโซเลเตอร์จะมีสถานะ "0" 1 วินาที และ "1" 4 วินาทีสลับกันไป

3.2.3 วงจรถอดรหัสหมายเลขโทรศัพท์ (DTMF Decoder)



รูปที่ 3.4 วงจรถอดรหัสหมายเลขโทรศัพท์

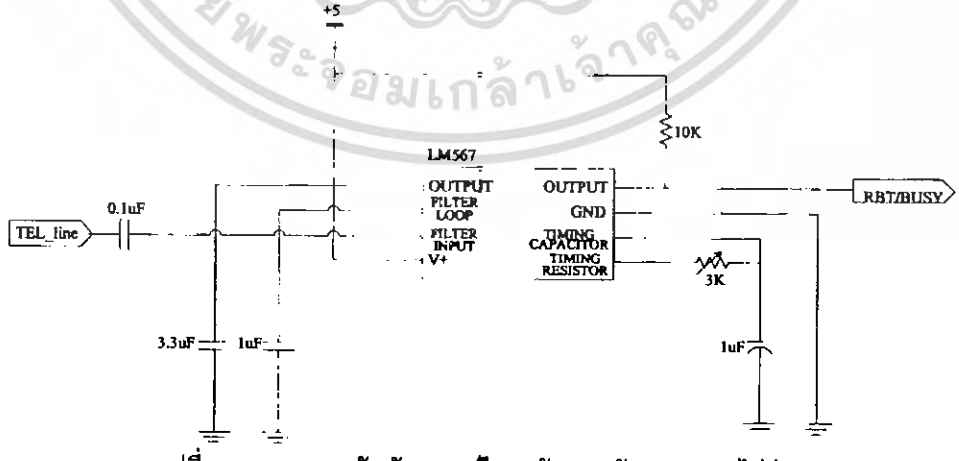
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

วงจรถอดรหัสหมายเลขโทรศัพท์จะใช้ไอซีสำเร็จรูปเบอร์ MT8870 ซึ่งจะทำหน้าที่ในการถอดรหัสความถี่ DTMF ออกมาเป็นรหัสไบนารี่ 4 บิต ภายใน MT8870 ซึ่งจะประกอบไปด้วยภาคกรองความถี่, ภาคถอดรหัส, ภาคตรวจสัญญาณ, ภาคขยายสัญญาณความแตกต่าง และภาคกำเนิดความถี่ โดยจะใช้คริสตอล 3.576 เมกกะเฮิรตซ์เป็นมาตรฐาน จากวงจรจะใช้แมชชิ่ง (Matching) 600 โอห์ม มาเป็นตัวคัปปลิ้ง (Coupling) สัญญาณจากคู่สายโทรศัพท์เข้ามายังวงจรเฉพาะสัญญาณไฟสลับเท่านั้น และยังทำหน้าที่ในการแยกกราวด์ (Ground) ของวงจรอีกด้วย เพราะในคู่สายโทรศัพท์จะมีสัญญาณรบกวนต่าง ๆ มากมายซึ่งอาจจะมีผลต่อการดีโค้ดสัญญาณได้ หลังจากที่ทำการกดหมายเลขและไอซีทำการดีโค้ดรหัสแล้ว ก็จะส่งไบนารี่มาเป็นเอาท์พุท Q1-Q4 จากนั้นไอซีก็จะทำการแลตช์ค่าสัญญาณนั้นเอาไว้จนกว่าจะได้รับสัญญาณใหม่เข้ามา แล้วค่อยเปลี่ยนเป็นรหัสตัวใหม่ โดยในการเขียนโปรแกรมควบคุมนั้นจะใช้สัญญาณจากขา STD และ Q1-Q4 เป็นตัวเช็คสถานะในการเขียนโปรแกรมควบคุม

3.2.4 วงจรตรวจจับสัญญาณเรียกกลับและสัญญาณไม่ว่าง (Ringback, Busy Tone Ddetector)

เมื่อมีการยกหูโทรศัพท์จะได้ยินสัญญาณ Dial Tone เอาท์พุทของ LM567 ที่ขา 8 จะได้ลอจิก "0" เมื่อมีการกดหมายเลขจะทำให้สัญญาณ Dial Tone เงียบหายไป ซึ่งจะทำให้เอาท์พุทของ LM567 นี้จะเปลี่ยนจากลอจิก "1" เป็น "0" นั้นได้แก่สัญญาณเรียกกลับ (Ringback Tone) ซึ่งสัญญาณนี้จะมีความถี่ 440 เฮิรตซ์ และ 480 เฮิรตซ์ ติด 1 วินาที คับ 4 วินาที สลับกัน ซึ่งจะทำให้เอาท์พุทที่ได้มีค่า "0" และ "1" สลับกัน และอีกสัญญาณหนึ่งคือ สัญญาณสายไม่ว่าง (Busy Tone) ซึ่งสัญญาณนี้จะมีความถี่ 480 เฮิรตซ์ กับ 620 เฮิรตซ์ ซึ่งจะติด 0.5 วินาที คับ 0.5 วินาที ซึ่งก็จะทำให้เอาท์พุทมีค่าเป็น "0" และ "1" สลับกัน ซึ่งสัญญาณเอาท์พุทเหล่านี้จะถูกส่งเข้าไมโครคอนโทรลเลอร์เพื่อทำการเปรียบเทียบระดับสัญญาณ โดยจะเลือกเอาเฉพาะสัญญาณที่เป็น "1" นานกว่า 4 วินาที มาใช้ในการเริ่มต้นการจับเวลาการใช้โทรศัพท์ และจะสิ้นสุดการจับเวลาเมื่อเครื่องโทรศัพท์ถูกวางหูลง

การตรวจจับสัญญาณ Ring Back Tone สามารถทำได้โดยตรวจจับความถี่ 400 เฮิรตซ์ ซึ่งวงจรที่ทำหน้าที่นี้ก็คือ เฟสล็อคลูป (PLL: Phase Lock Loop) และในส่วนของอุปกรณ์ที่นำมาใช้งานได้แก่ ไอซีเบอร์ 567 วงจรเฟสล็อคลูปที่ใช้ทดลองตรวจจับความถี่ 400 เฮิรตซ์ แสดงดังรูปที่ 3.5



รูปที่ 3.5 วงจรตรวจจับสัญญาณเรียกกลับและสัญญาณสายไม่ว่าง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากวงจรในรูปที่ 3.5 เมื่อทำการวัดสัญญาณเอาท์พุทจะพบว่า สัญญาณเอาท์พุทจะมีลักษณะเป็นพัลส์สแควร์เวฟ ตามสัญญาณด้านอินพุท คือ

1. ถ้าเป็นสัญญาณ Dial Tone เอาท์พุทที่ได้จะเป็นลอจิก "0" ยาวติดกันตลอดเวลา
2. ถ้าเป็นสัญญาณสาญไม่ว่าง เอาท์พุทที่ออกมาจะมีลักษณะเป็นลอจิก "0" และลอจิก "1" สลับกันไป โดยจะมีช่วงเวลาห่างกัน 0.5 วินาที
3. ถ้าเป็นสัญญาณเรียกกลับ เอาท์พุทที่ออกมาจะมีลักษณะเป็นลอจิก "0" วินาที และเป็นลอจิก "1" 4 วินาที สลับกันไป

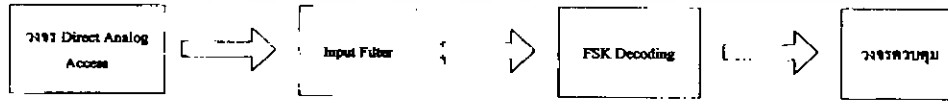
3.2.5 วงจร Caller ID

การออกแบบวงจร Caller ID ที่อธิบายในที่นี้คือ "อธิบายวิธีการทำ" ตัวอย่างในการสร้างส่วนประกอบพื้นฐานที่ต้องการเชื่อมต่อไปยังสายโทรศัพท์และที่ได้มาจาก CO (Central Office) ที่จัดส่งข้อมูล CID (Caller ID) แต่ละชุดส่วนประกอบด้วยชุดของการบรรยายแบบแผนการอธิบายการเชื่อมต่อสายโทรศัพท์และที่ได้ข้อมูล CID ประกอบด้วยโปรแกรมทำการเล็ก ๆ ซึ่งทำการรับในเวลาที่ CO ได้เตรียมข้อมูล CID เปลี่ยนแปลงข้อมูลนี้ให้กลายเป็นรูปแบบที่สามารถแสดงไปยัง CRT ของ PC โปรแกรมนี้จะถูกใช้ร่วมกันกับเกณฑ์ Bellcore TA-NWT-000030 ที่ใช้อ้างอิงเมื่อมีการออกแบบการเชื่อมต่อ User ที่คุณเป็นเจ้าของ แบบแผนและซอฟแวร์ที่ถูกสาธยายในที่นี้คือ การสร้าง ทดสอบ และการทดสอบก่อนที่จะนำไปใช้งานจริง

3.2.5.1 การทำงานโดยทั่วไป

ข้อมูล CID เตรียมการโดย CO ไปยังวงจร (หลังจากทำการถอดรหัสด้วยบอร์ดสาธิต) ที่ถูกกำหนดเส้นทางโดยตรงไปยัง PC ทางพอร์ต RS-232 ไม่ว่า PC จะถูกใช้ในการควบคุมหรือไม่กำลังงานก็จะถูกนำไปประยุกต์ใช้กับวงจร เช่นเดียวกับที่แสดงข้อมูล CID

ในขณะที่กำลังรอสัญญาณ CID ส่วนมากวงจรจะหมดกำลัง เหตุการณ์แรกในลำดับนี้ที่เกิดขึ้นเป็นการชี้บอกสัญญาณกระดิ่ง (Ring Indication) ซึ่งจะเป็นอินพุทให้กับเหตุการณ์ที่สองด้วยรูปแบบของโปรแกรมซอร์ฟแวร์แล้วทำการเปิดสวิทช์ไฟเพื่อให้กระแสไฟเข้าวงจรน้อยที่สุด (ซึ่งต้องการให้โปรแกรมซอร์ฟแวร์นี้ได้ทำงาน) วงจรตอนนี้พร้อมที่จะรับการเข้ารหัสข้อมูล FSK ที่ส่งโดย CO ทันทีที่ข้อมูลถูก ดีมอดูเลต ข้อมูลที่ส่งจากวงจรผ่านทางสายแคเบิ้ลไปยังพอร์ต RS-232 ของ PC โปรแกรมที่จับได้อย่างแรกและเมื่อแสดงข้อมูลบน CRT ของ PC หลังจากข้อมูล CID ถูกแสดงแล้ว (ถึงแม้ว่าจะยังคงอยู่ภายใต้การควบคุมของซอร์ฟแวร์) วงจรก็จะย้อนกลับไปยังส่วนของการปิดสวิทช์ไฟไม่ให้กระแสไฟเข้าเครื่องคอมพิวเตอร์

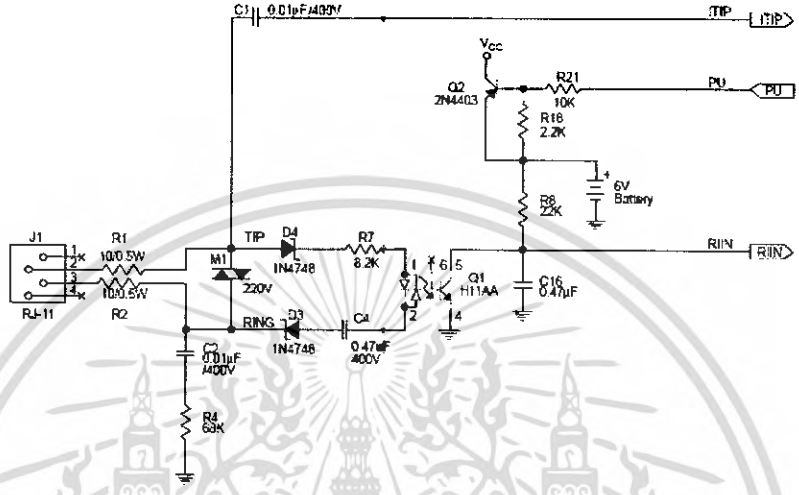


รูปที่ 3.6 บล็อกไดอะแกรมวงจร CID

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.2.5.2 ภาคอินพุทและการเข้าถึงสัญญาณอนาลอกโดยตรง (Direct Analog Access: DAA)

ในภาคแรก (ดูรูปที่ 3.7) ของวงจรที่ออกแบบเป็นภาคอินพุท ภาคนี้ประกอบไปด้วยฟังก์ชั่นการเข้าถึงสัญญาณอนาลอกโดยตรง (Direct Analog Access) และเครื่องตรวจจับสัญญาณ Ring Indicator การเข้าถึงสัญญาณอนาลอกโดยตรงต้องมีการแยกระหว่างวงจรและสายโทรศัพท์ เพื่อให้สามารถถอดข้อมูลที่ส่งโดย CO ออกมาได้ การเข้าถึงสัญญาณอนาลอกโดยตรงที่ดีที่สุดที่สายโทรศัพท์จะสิ้นสุดถ้าหากมีอิมพีแดนซ์ทิปและริงที่เหมาะสม



รูปที่ 3.7 วงจร Direct Analog Access

การแยกออกจะกำหนดโดย การเข้าถึงสัญญาณอนาลอกโดยตรงจะถูกเรียกร้องให้ป้องกันแรงดัน full Ring Indicator (สูงสุด 300 V Peak-to-peak บนยอดของแรงดันสูงสุด 48 V ที่ส่งมาจากแบตเตอรี่ของ CO) ซึ่งทำให้ส่วนประกอบแรงดันต่ำของวงจรเสียหาย ในเวลาเดียวกันการเข้าถึงสัญญาณอนาลอกโดยตรงจะต้องทิ้งแรงดันบางค่าที่น้อยกว่าแรงดันกระดิ่ง 26 V ที่ไม่ผลต่อสัญญาณกระดิ่ง ตัวต้านทานฟิวส์ซึ่งไม่ไวไฟ (ค่า 10Ω) ซึ่งเป็นส่วนประกอบของวงจรส่วนแรกที่กลายมาเป็นจุดเชื่อมต่อกับสายโทรศัพท์ ให้ฟิวส์เป็นตัวป้องกันในกรณีของแรงดันเกิน.

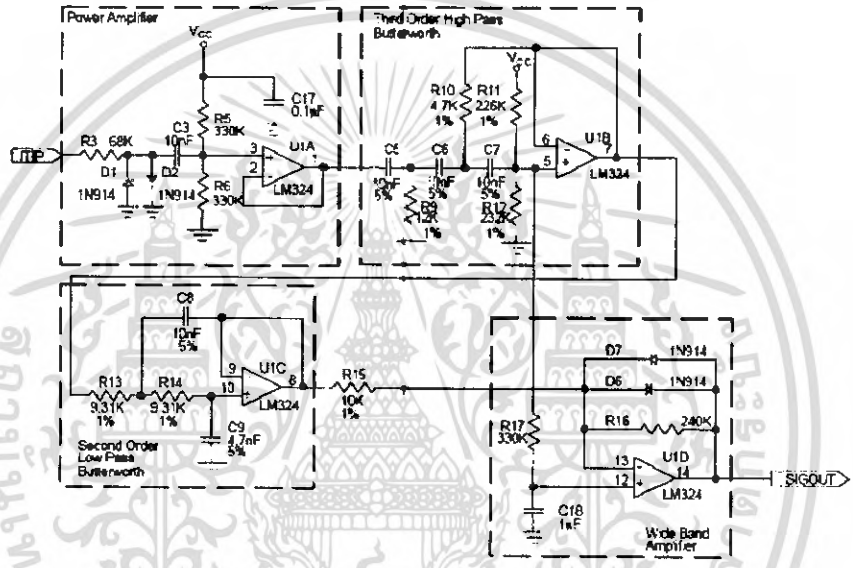
ในการเตรียมที่จะส่งข้อมูล CID (CO ส่งสัญญาณกระดิ่งสัญญาณแรก) สิ่งที่ใส่ในวงจรจะแจ้งให้ทราบเกี่ยวกับข้อมูล CID ที่ได้รับ Ring Indicator ถูกใช้แบ่งส่วนเปิด-ปิดสวิทซ์ไฟไม่ให้กระแสไฟเข้าตัววงจร

ในภาคอินพุทจะมีตัวกรองความถี่สูง RC ซึ่งไม่มีอิทธิพลมากพอที่จะสังเกตเห็นได้ในแบนด์วิดท์ของภาคตัวกรองความถี่ วงจรมีความต้านทานต่อไฟสลับ AC เช่นเดียวกับที่ติดต่อกับ CO ซึ่งมากกว่า 7,000Ω ความต้านทานอินพุท DC เท่านั้นที่เกิดขึ้นจากการรั่วของตัวเก็บประจุอินพุทซึ่งมีผลน้อยกว่า 5 μA (กำหนดเฉพาะตัวสั่นกระดิ่งที่มีค่าเท่ากัน 1 ตัว) ความต้านทานอินพุท DC ที่น้อยมากเกินไปจะส่งผลให้เกิดสัญญาณรบกวนขึ้นที่ความถี่ต่ำที่วงจรทำให้เกิดการเปิดสวิตช์ไม่ให้กระแสไหลเข้าวงจรได้เอง การปฏิบัติหน้าที่ภาคอินพุทในส่วนที่เป็นภาคของกลุ่ม DC สังเกตได้ว่าอุปกรณ์ในภาคอินพุทจะต้องสามารถทนต่อศักย์ทางไฟฟ้าสูงสุดประมาณ 348 V

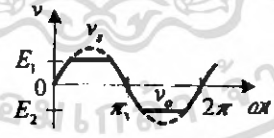
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.2.5.3 วงจรขริบและขยายแรงดัน

ภาคที่สอง (ดูรูปที่ 3.8) ของวงจรที่ออกแบบเป็นภาคการขริบและขยายแรงดันอินพุทซึ่งประกอบด้วยไดโอดต่อขนานกับโหลด ทำให้กระแสไหลผ่านวงจรทั้งสองไม่เท่ากัน เมื่อไดโอดนำกระแสใช้ความต้านทาน 68 kΩ สำหรับจำกัดกระแสที่ไหลผ่านไดโอด เมื่อแรงดันอินพุทเท่ากับแรงดันเอาท์พุทจะทำให้ไดโอด D1 หยุดนำกระแส เมื่อแรงดันเอาท์พุทสูงกว่า 0V จะทำให้ไดโอด D1 หยุดนำกระแส และแรงดันเอาท์พุทเท่ากับ 0V ในทางตรงกันข้ามเมื่อแรงดันอินพุทต่ำกว่า 0V จะทำให้ไดโอด D1 นำกระแส และแรงดันเอาท์พุทเท่ากับแรงดันอินพุท ในทำนองเดียวกัน เมื่อแรงดันอินพุทสูงกว่า 0V ไดโอด D2 จะนำกระแส และแรงดันเอาท์พุทเท่ากับแรงดัน 0V ในทางตรงข้ามเมื่อแรงดันอินพุทต่ำกว่า 0V ไดโอด D2 จะหยุดนำกระแส และแรงดันเอาท์พุทเท่ากับแรงดันอินพุท รูปที่ 3.9 แสดงให้เห็นรูปคลื่นเอาท์พุทของวงจรขริบ โดยที่เอาท์พุทที่ได้จะนำไปขยายแรงดันด้วยวงจรขยายตามแรงดัน ซึ่งประกอบด้วยออปแอมป์



รูปที่ 3.8 วงจรกรองความถี่และวงจรขยาย



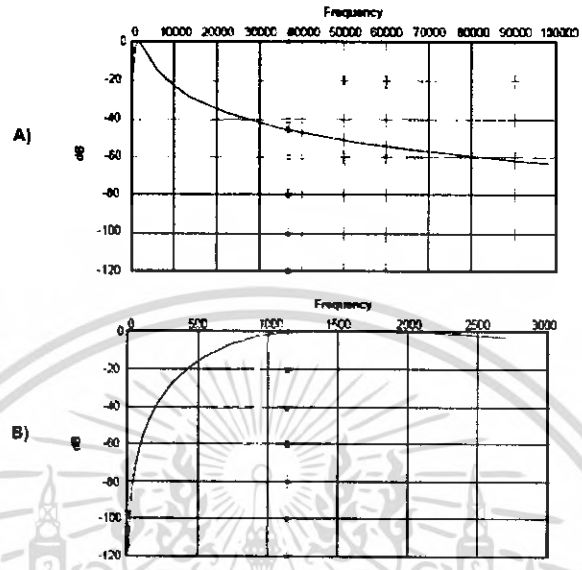
รูปที่ 3.9 แสดงรูปคลื่นเอาท์พุทของวงจรขริบ

3.2.5.4 ภาคตัวกรองความถี่

ภาคที่สาม (ดูรูปที่ 3.8) ของวงจรที่ออกแบบเป็นภาคการกรองความถี่ซึ่งประกอบด้วยวงจร Band Pass Filter และวงจรขยาย ฟังก์ชันแบนด์พาสจะประกอบไปด้วยวงจรกรองความถี่ 2^{nd} order Low Pass Active Butterworth และ 3^{rd} order High Pass Active Butterworth ซึ่งมีประสิทธิภาพ -3 dB ในช่วงความถี่ Band Pass ตั้งแต่ 960 – 2,850 Hz ในขณะที่ LM-324 ถูกใช้ให้เป็นประโยชน์ในฐานะที่เป็นอัตราขยายขององค์ประกอบในวงจรไฟฟ้า มักจะเป็นที่สังเกตว่าวงจรขยายเกือบทั้งหมดประกอบด้วยอัตราขยายที่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สมเหตุสมผลอย่างมากเมื่อเปรียบเทียบกับวงจรอื่นจะมีอินพุทอิมพีแดนซ์สูง และมีแบนด์วิดท์กว้างพอ ที่จะนำไปใช้งาน กำลังขับเคลื่อนเอาท์พุทจะมากพอ ๆ กับที่ขับโหลดอิมพีแดนซ์ของฟิลเตอร์ ผลตอบสนองแบนด์พาสและอัตราขยายจะได้มาจากการเปลี่ยนแปลงของฟิลเตอร์จากสมการที่จะกล่าวถึงต่อไป ผลบวกอัตราขยายเทียบกับเส้นกราฟความถี่ของ Low Pass Filter และ High Pass Filter ในรูปที่ 3.10



รูปที่ 3.10 กราฟแสดงผลตอบสนองความถี่ของอินพุทฟิลเตอร์ A) Low Pass Filter B) High Pass Filter

ลำดับขั้นของฟิลเตอร์คำนวณได้จาก

$$N = INT\left(Log\frac{10\,\hat{}\,(AMAX/10-1)}{2((10\,\hat{}\,0.3)-1)Log(Wn)}\right)+1$$

AMAX: การลดทอนที่ความถี่ Stop band

$Wn = F_1/F_2$ สำหรับการคำนวณ Low Pass Filter

$WN = F_C/F_1$ สำหรับ High Pass Filter

F_1 = ความถี่ Stop band

F_C = ความถี่ Cutoff

ผลลัพธ์ทั้งหมดสามารถตรวจสอบได้ว่าเป็น Band Pass Filter ซึ่งมีโพลโรลล์ออฟในด้านความถี่ต่ำ 3 ค่า (60 dB/decade) และมีโพลโรลล์ออฟด้านความถี่สูง 2 ค่า (40 dB/decade) สิ่งที่ต้องการเพิ่มเติมเข้าไปในส่วนแรกและส่วนที่สอง คือ วงจรกรองสัญญาณกระดิ่ง 20 Hz และสัญญาณไฟฟ้ารบกวน 60 Hz หรือ 50 Hz ออกวงจรนี้จึงได้ออกแบบด้วยการลดทอนสัญญาณ 60 Hz ด้วยค่าที่น้อยที่สุด 70 dB ลักษณะของฟิลเตอร์ที่ดีมักจะใช้ตัวต้านทาน 1% และตัวเก็บประจุ 5% ถ้าภาคอินพุทถูกใช้ประโยชน์สำหรับลักษณะจำเพาะ High Pass ของมัน มันมักจะถูกควบคุมด้วยค่าตัวต้านทานและค่าตัวเก็บประจุด้วยเช่นกัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.2.5.5 ภาคขยายแถบความถี่กว้าง

ภาคที่สี่ (ดูรูปที่ 3.8) ของวงจรที่ออกแบบเป็นวงจรขยายแถบความถี่กว้าง อัตราขยายจะเลือกใช้ในกรณีสัญญาณแย่ที่สุด (3.0 mVrms หรือ -48 dB) ตัวถอดรหัสวงจรเฟสล็อคลูปจะยังคงทำงานอยู่และระบบจะสามารถดึงข้อมูล CID ออกมาได้ ในภาคนี้จะถูกนำมาใช้ประโยชน์ให้กับไอซี LM-324 เป็นองค์ประกอบในวงจรขยาย สมการการควบคุมสำหรับภาคขยายมีดังนี้

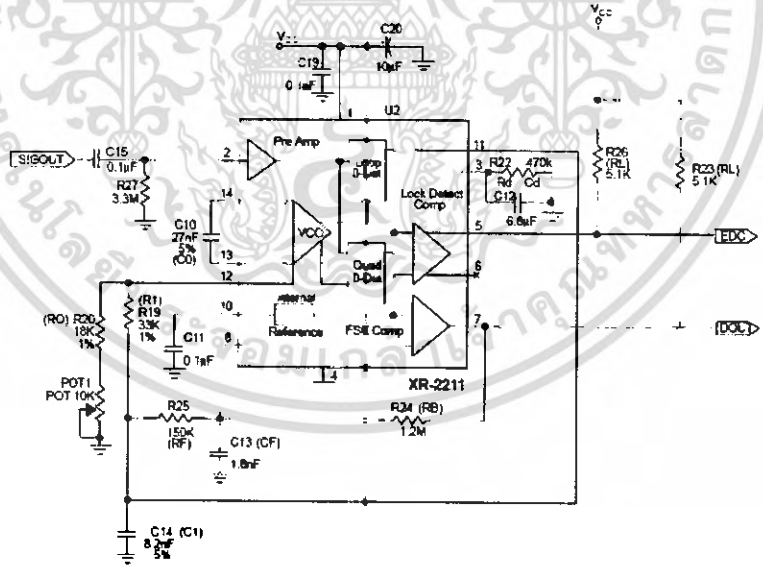
$$Gain = \frac{Rfb}{Rin}$$

Rfb คือ ตัวต้านทานที่เชื่อมต่อจากเอาท์พุทที่วงจรกลับขั้วสัญญาณอินพุทของวงจรขยายเชิงดำเนินการ

Rin คือ ตัวต้านทานจากแหล่งกำเนิดสัญญาณที่วงจรกลับขั้วสัญญาณอินพุทของวงจรขยาย

3.2.5.6 วงจรเฟสล็อคลูปและวงจรถอดรหัส FSK

ภาคที่ห้า (รูปที่ 3.11) ของวงจรที่ออกแบบเป็นวงจรถอดรหัส FSK และภาคตรวจจับสัญญาณ Carrier เส้นทางของสัญญาณในสายโทรศัพท์ของภาคนี้จะส่งผ่านมาจากภาค Band Pass Filter ในภาคนี้มีหน้าที่การทำงาน 2 หน้าที่ หน้าที่แรกเป็นเพียงการตรวจจับสัญญาณถ้ามีความถี่อยู่ในแถบความถี่จำเพาะสัญญาณการตรวจจับพลังงานจะกลายเป็นแอกทีฟ หน้าที่ที่สองเป็นการดีมอดูเลตสัญญาณที่ถูกมอดูเลตแบบ FSK ซึ่งมีอัตราการส่งสัญญาณ 1200 โบด์ ของความถี่ในแถบความถี่ตั้งแต่ 1200 Hz ถึง 2200 Hz ข้อมูลที่ถูกดีมอดูเลตนี้จะประกอบกันขึ้นเป็นข้อมูล CID ที่ถูกมอดูเลตโดย CO สังเกตว่าการตรวจจับพลังงานจะต้องให้ผลที่ต้องการก่อนข้อมูล CID สามารถให้ผลที่ต้องการซึ่งได้รับการพิจารณาอย่างถี่ถ้วนแล้ว



รูปที่ 3.11 **วงจรการถอดรหัส** FSK

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ซึ่งภาคนี้ได้ใช้ประโยชน์วงจรเฟสล็อคลูปในไอซี XR-2211 ที่ทำงานในฟังก์ชั่นนี้ XR-2211 ที่รวมความถี่ VCO มักจะถูกปรับค่าด้วยตัวปรับค่าความต้านทานที่มีความถี่กลางเชิงเรขาคณิต 1,625 Hz ที่รับรองค่า Duty Cycle 50% ที่ขา 7 ของไอซี XR-2211

หมายเหตุ ในขณะที่มันไม่ได้ทำการออกแบบวงจร มันอาจจะเป็นได้ว่าวงจรขยายสัญญาณในภาควงจรกรองสัญญาณถูกตัดทิ้งและใช้ประโยชน์ XR-2211 เช่นเดียวกับหลักการของภาควงจรขยาย ซึ่งอาจจะต้องการถอดออกมากกว่าขยายสัญญาณจากภาควงจรกรองสัญญาณ หรือการเคลื่อนที่ที่เสี่ยงต่อการตอบสนองที่ไม่มีความไวมากพอในกระบวนการระดับต่ำ ๆ (-48 dB) สมการสำหรับเฟสล็อคลูปคำนวณได้ดังต่อไปนี้

$$C0 \quad \frac{1}{f0 * R0} \qquad f0 = \frac{f1 + f2}{2}$$

f1, f2: คือ เครื่องกำบังและความถี่อวกาศ

R0: คือ ความถี่ที่ควบคุมตัวต้านทานที่ต่อที่ขา 12 ของไอซี XR-2211

$$R1 \quad \frac{f0 * R0 * 2}{f2 - f1}$$

R1: คือ ตัวต้านทานที่ต่อจากขา 12 ไปขา 11

$$\varsigma \quad \sqrt{\frac{1.25 * C0}{R1 * C1}}$$

ς : คือ Damping Factor R1 มีหน่วยเป็น kΩ

$$V_{REF} \quad \frac{V_{CC}}{2} = .650$$

V_{REF}: คือ ค่าแรงดันอ้างอิงที่ขา 10

$$Ko \quad \frac{2 * \pi}{V_{REF} * C0 * R1}$$

K_0: ตัวประกอบการเปลี่ยนแปลง VCO มีหน่วยเป็น Radian / sec / volt

$$Kd = \frac{V_{REF} * R1}{10 * \pi}$$

Kd: Phase Detector Gain มีหน่วยเป็น Volt / Radian

3.3 Microcontroller

ในโครงงานนี้จะใช้ไมโครคอนโทรลเลอร์ AT89C51 ทั้งนี้เพื่อความสะดวกในการทดลองและใช้งานจริงได้ ซึ่งส่วนของวงจรดีเทคเตอร์จะทำการดีเทคสัญญาณในสภาวะต่าง ๆ บนคู่สายโทศัพท์เข้ามาในส่วนควบคุมการทำงาน โดยไมโครคอนโทรลเลอร์ AT89C51 จะเป็นตัวควบคุมการทำงานทั้งหมด ซึ่งในส่วนควบคุมนี้จะมีวงจร Real Time Clock ทำหน้าที่ในการสร้างฐานเวลาให้กับ AT89C51

3.2.1 ส่วนควบคุมการรับข้อมูลจากวงจรดีเทคเตอร์

สัญญาณจากคู่สายโทรศัพท์จะผ่านวงจรดีเทคเตอร์เพื่อแปลงให้เป็นสัญญาณดิจิตอล แล้วป้อนให้กับพอร์ทของ AT89C51 ซึ่งสัญญาณจากวงจรดีเทคเตอร์มีดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

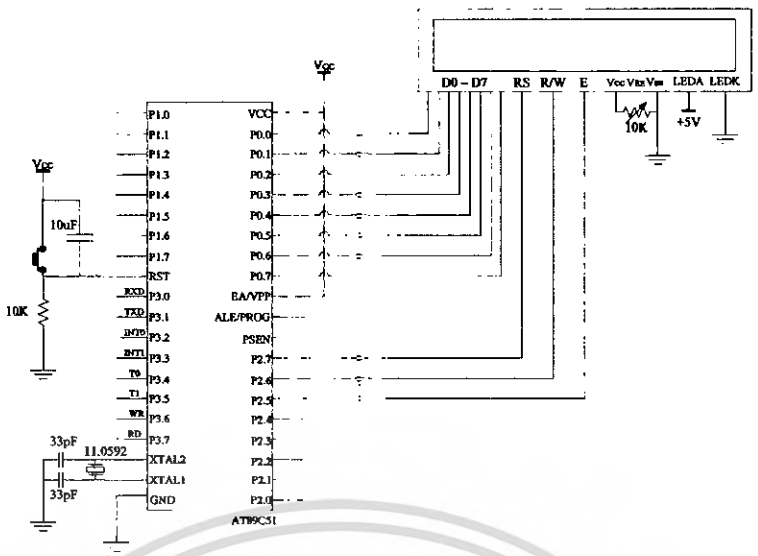
1. สัญญาณตรวจสอบการยกหูโทรศัพท์ เป็นสัญญาณข้อมูล 1 บิต ขนาดประมาณ 5 โวลท์ ซึ่งแทนด้วยระดับลอจิก "1" เมื่อมีการวางหูโทรศัพท์และสัญญาณมีขนาดประมาณ 0 โวลท์ ซึ่งแทนด้วยระดับลอจิก "0" เมื่อมีการยกหูสัญญาณดังกล่าวจะถูกส่งเข้าพอร์ท 1 (ขา 6) ของ AT89C51
2. สัญญาณตรวจสอบการกดปุ่มเลขหมายโทรศัพท์ เป็นสัญญาณข้อมูล 1 บิต ได้จากขา 15 ของ IC MT8870 ของวงจรถอดรหัสเลขหมายโทรศัพท์ ซึ่งแทนด้วยระดับลอจิก "1" ขณะกดปุ่มหมายเลขโทรศัพท์ และแทนด้วยระดับลอจิก "0" เมื่อไม่มีการกดปุ่มหมายเลขโทรศัพท์ สัญญาณดังกล่าวจะส่งเข้าพอร์ท 1 (ขา 1) ของ AT89C51
3. สัญญาณเลขหมายโทรศัพท์ เมื่อการกดหมายเลขโทรศัพท์ 1 ตัว IC MT8870 ของวงจรถอดรหัสหมายเลขโทรศัพท์จะทำการถอดรหัสออกเป็นเลขไบนารี่ 4 บิต สัญญาณจากขา 14 จะเป็นบิตที่มีนัยสำคัญสูงสุด แล้วลดลงมาจนถึงบิตที่มีนัยสำคัญต่ำสุดที่ขา 11 สัญญาณดังกล่าวจะส่งเข้าพอร์ท 1 (ขา2 -5) AT89C51
4. สัญญาณตรวจสอบการรับสายของปลายทาง สัญญาณนี้ได้จากวงจรตรวจสอบสัญญาณเรียกกลับเป็นสัญญาณลอจิก "0" 4 วินาที และลอจิก "1" 1 วินาที เมื่อปลายทางยังไม่ได้รับสาย และเป็นลอจิก "0" ตลอด เมื่อปลายทางรับสายแล้ว สัญญาณนี้จะถูกส่งเข้าพอร์ท 2 (ขา 23) ของ AT89C51
5. สัญญาณตรวจสอบการโทรเข้า สัญญาณนี้จะได้จากวงจรตรวจจับสัญญาณกระดิ่ง ซึ่งเป็นสัญญาณลอจิก "0" 1 วินาที และลอจิก "1" 4 วินาทีและเป็นลอจิก "1" ตลอดเมื่อเราทำการรับสายแล้ว สัญญาณนี้จะถูกส่งเข้าพอร์ท 2 (ขา 24) ของ AT89C51
6. สัญญาณข้อมูล Caller ID ที่ถูกมอดูเลตแบบ FSK สัญญาณนี้เราจะได้จากการดีมอดูเลตสัญญาณ FSK ที่ถูกส่งมาระหว่างสัญญาณกระดิ่งสัญญาณแรกกับสัญญาณกระดิ่งสัญญาณที่สอง สัญญาณดังกล่าวจะถูกส่งเข้าพอร์ท 2 (ขา 22) ของ AT89C51

3.2.2 การเชื่อมต่อจอแสดงผล LCD เข้ากับไมโครคอนโทรลเลอร์

การเชื่อมต่อ LCD โมดูลกับ MCS-51 นั้นสามารถทำได้หลายวิธี โดยอาจต่อตรงกับ MCS-51 แบบ Memory Map ก็ได้ หรืออาจต่อกับพอร์ตอินพุท - เอาต์พุทที่มีอยู่แล้วก็ได้ การต่อ LCD โมดูลในรูปที่ 3.12 ขาอินาเบิลของ LCD จะถูกถอดรหัสไว้ที่แอดเดรส FA00H ดังนั้นแอดเดรสต่าง ๆ ของ LCD โมดูลจะเป็นดังนี้

ตำแหน่งแอดเดรส	การใช้งาน
FA00H	เขียนคำสั่ง
FA01H	อ่านคำสั่ง
FA02H	เขียนข้อมูล
FA03H	อ่านข้อมูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.12 แสดงการต่อ LCD มอดูลกับพอร์ท P0 ของ MCS-51

3.2.3 การเชื่อมต่อ PCF8574 เข้ากับไมโครคอนโทรลเลอร์

ไอซีเบอร์ PCF8574 เป็นไอซีวงจรรวมที่ใช้ในการขยายพอร์ท ซึ่งเป็นพอร์ทขนานแบบ 8 บิต และสามารถเชื่อมต่อระบบไมโครคอนโทรลเลอร์โดยใช้บัสแบบ I^2C ซึ่งมีคุณสมบัติหลัก ๆ ดังนี้

1. ใช้แรงดันไฟต่ำตั้งแต่ 2.5 โวลต์ ถึง 6 โวลต์
2. สิ้นเปลืองกระแสขณะสแตนด์บายสูงสุดเพียง 10 ไมโครแอมป์
3. อินพุตเอาต์พุตพอร์ทเป็นแบบ 8 บิต
4. เอาท์พุทสามารถแลตช์ข้อมูลได้ และสามารถขับกระแสสำหรับหลอด LED ได้
5. มีขาอินเทอร์รัปต์ที่สามารถต่อกับไมโครคอนโทรลเลอร์ได้ โดยเมื่อมีข้อมูลจากภายนอกเข้ามาทางอินพุท ตัวมันจะสร้างสัญญาณอินเทอร์รัปต์ออกมา
6. สามารถต่อไอซีเบอร์นี้ได้ถึง 8 ตัว โดยการกำหนดบิตแอดเดรสที่สามารถโปรแกรมได้ (A0-A2)

ถ้าหากต้องการติดต่อไมโครคอนโทรลเลอร์กับชิปตัวนี้ จะต้องส่งค่าแอดเดรสออกไปก่อน ชิป PCF8574 มีการกำหนดแอดเดรสดังรูปที่ 3.13

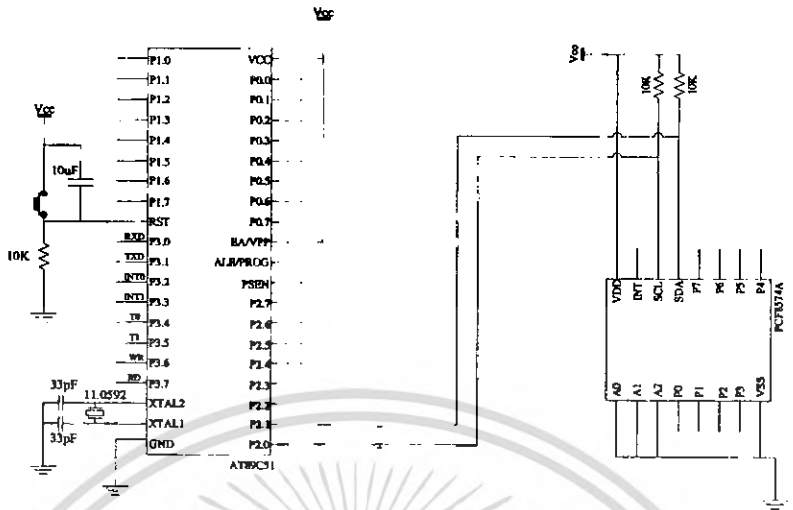
0	1	1	1	A2	A1	A0	$\overline{R/W}$

รูปที่ 3.13 การกำหนดแอดเดรสของชิป PCF8574

โดยค่า 0111 เป็นบิตแอดเดรสแบบคงที่ ที่ถูกโปรแกรมมาจากโรงงานผู้ผลิต ส่วน A2 A1 A0 เป็นบิตแอดเดรสที่สามารถโปรแกรมเองได้ สำหรับบิตต่ำสุดถ้าเป็นลอจิก "0" หมายความว่าใช้เป็นเอาท์พุทพอร์ต ถ้าลอจิก "1" หมายความว่าเป็นอินพุทพอร์ต สำหรับโครงงานนี้ได้ออกแบบการใช้งาน PCF8574 สำหรับการส่งข้อมูลไปยังพรินเตอร์ดังรูปที่ 3.14 โดยกำหนดให้ P0-P7 ของ PCF8574 เป็น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บัสข้อมูลของเครื่องพรินเตอร์ และ ให้ P1.6 และ P1.7 ใช้สำหรับสัญญาณ STROBE และสัญญาณ BUSY ติดต่อเครื่องพรินเตอร์ตามลำดับ

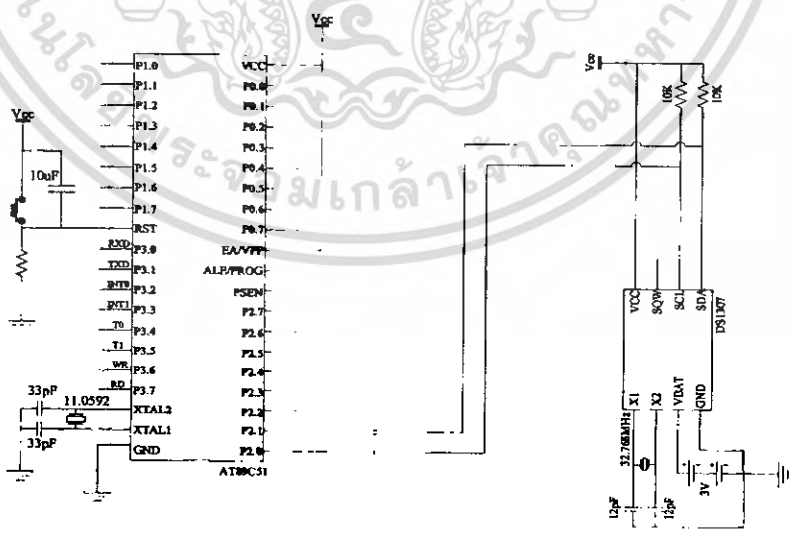


รูปที่ 3.14 แสดงการเชื่อมต่อไมโครคอนโทรลเลอรกับไอซี PCF8574

3.2.4 การเชื่อมต่อ DS1307 เข้ากับไมโครคอนโทรลเลอร์

จากวงจรแสดงในรูปที่ 3.15 จะเห็นได้ว่ามีลักษณะการต่อเหมือนกับอุปกรณ์บนระบบบัส I^2C ตัวอื่นทุกประการ และสามรถที่จะต่อไอซีทั้งหมดร่วมกันบนสาย SDA และ SCL ได้เป็นการย้ำให้เห็นถึงความสามารถพิเศษของระบบบัส I^2C ที่ผู้ใช้งานสามารถเชื่อมต่ออุปกรณ์ที่มีความแตกต่างกันในหน้าที่การทำงานบนสายสัญญาณเส้นเดียวกันได้

จากวงจรในรูปที่ 3.15 ไอซี DS1307 จำเป็นต้องต่อแบตเตอรี่ไว้ตลอดเวลาไม่ว่าจะใช้งานหรือไม่ ทั้งนี้เพื่อรักษาการทำงานของวงจรภายใน DS1307 ให้ยังคงทำงานต่อเนื่อง เมื่อใดที่ไมโครคอนโทรลเลอร์ต้องการอ่านข้อมูลก็จะได้ข้อมูลเวลาที่เป็นจริงตลอดเวลา



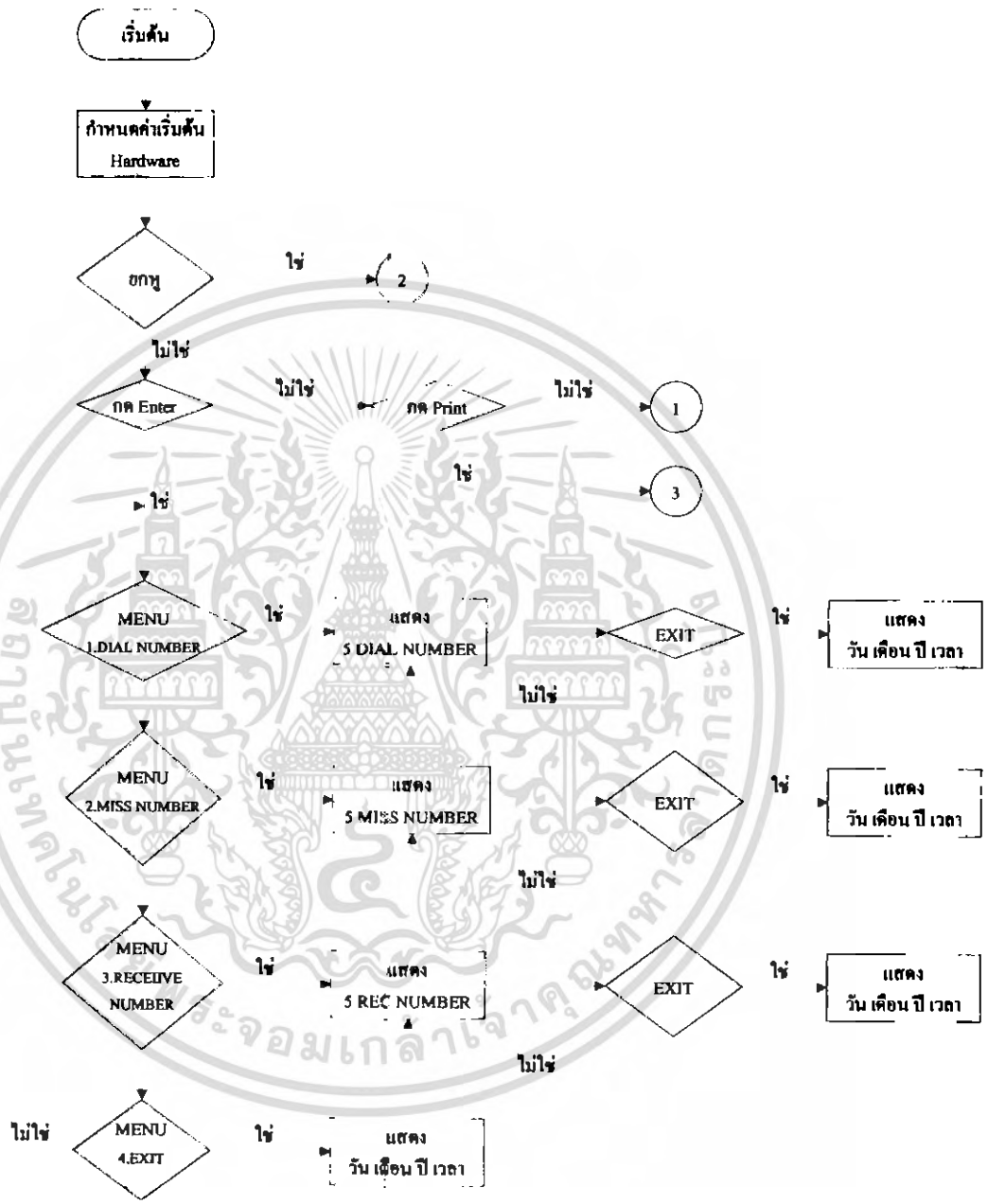
รูปที่ 3.15 แสดการเชื่อมต่อไมโครคอน โทรลเลอร์ MCS-51 กับไอซี Real Time Clock DS1307

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.2.5 โครงสร้างทาง Solfware

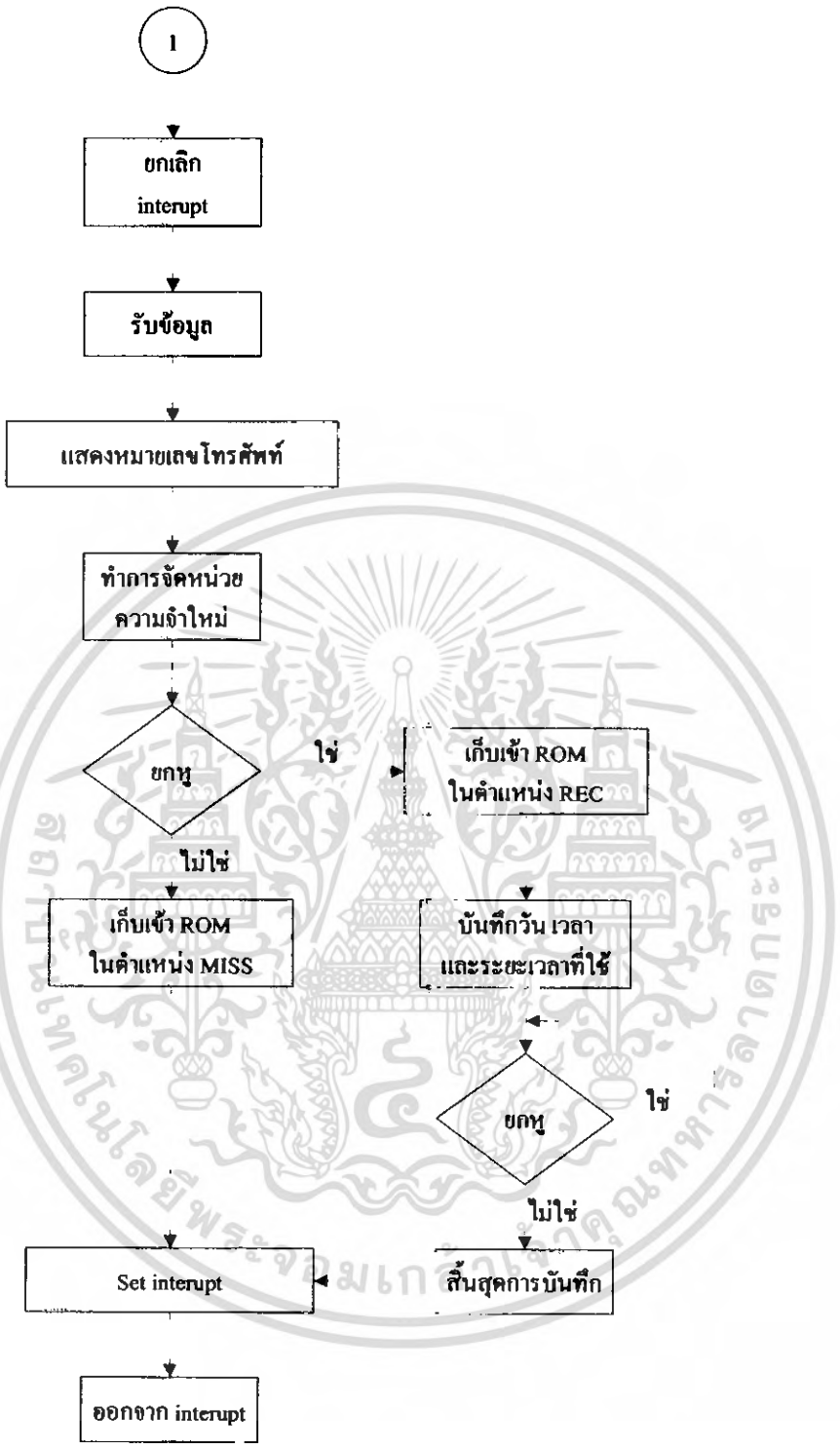
ทำการเขียนโปรแกรมควบคุมการทำงานตาม Flow Chart ต่อไปนี้

3.2.5.1 โปรแกรมควบคุมการบันทึกการใช้งานโทรศัพท์



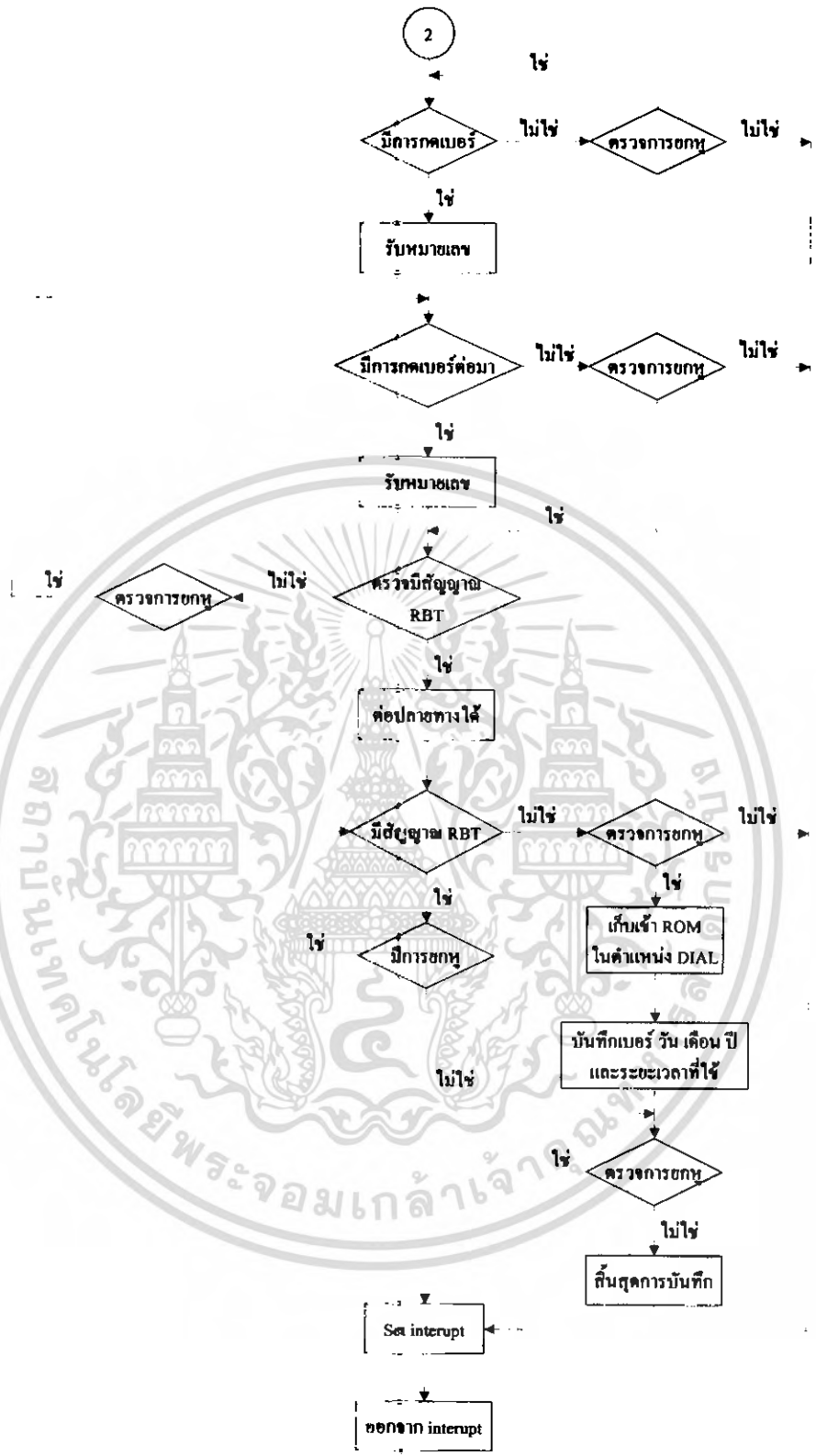
รูปที่ 3.16 โฟว์ชาร์ทของโปรแกรมแสดงข้อมูลที่บันทึกไว้ออกจอ LCD

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



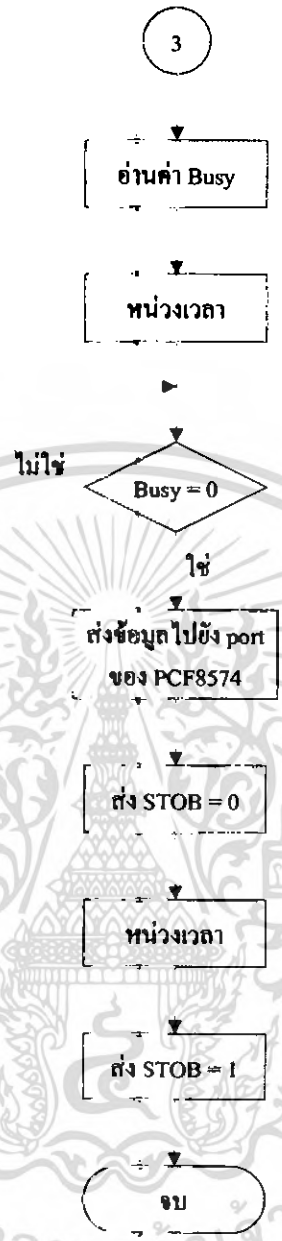
รูปที่ 3.17 โฟว์ชาร์ทของโปรแกรมบันทึกการใช้งานในส่วนของการโทรเข้า

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.18 โฟว์ชาร์ทของโปรแกรมบันทึกการใช้งานในส่วนของการโทรออก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.17 โฟว์ชาร์ทของโปรแกรมติดต่อพริ้นเตอร์

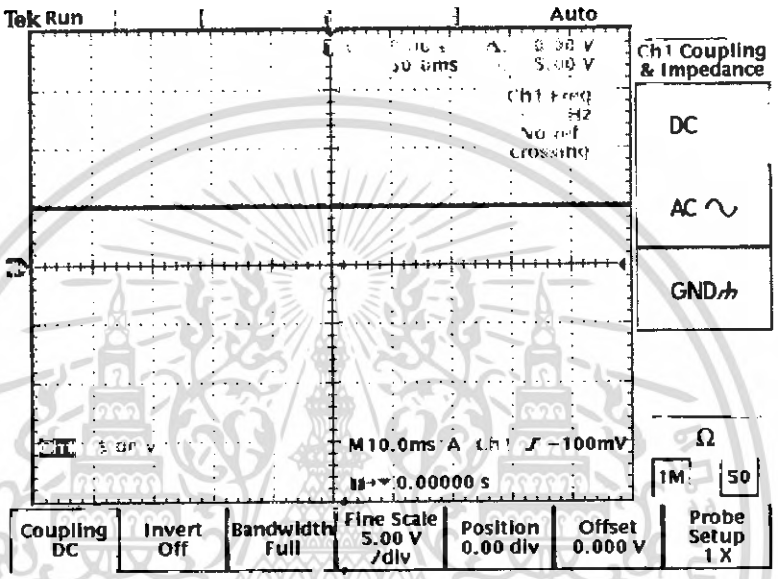
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 4

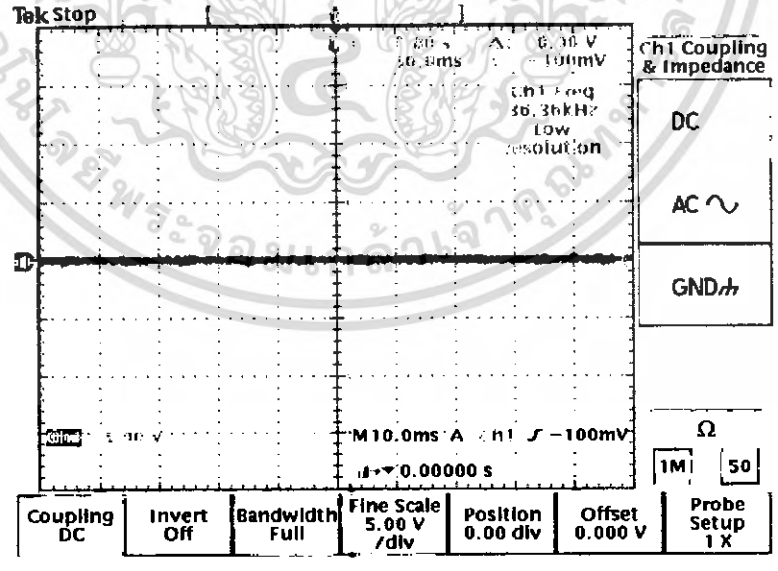
การทดลองและผลการทดลอง

4.1 การทดลองวงจรตรวจจับสัญญาณยกหู / วางหูโทรศัพท์

ทำการทดลองโดยการต่อวงจรตามรูปที่ 3.2 เมื่อการวัดสัญญาณจะได้ดังรูปที่ 4.1 และ รูปที่ 4.2 ว่า ในขณะที่วางหูโทรศัพท์สัญญาณที่ได้จะเป็นสัญญาณที่เป็นลอจิก "1" และเมื่อเราทำการยกหูโทรศัพท์ สัญญาณที่วางได้จะกลายเป็นลอจิก "0"



รูปที่ 4.1 แสดงสัญญาณขณะวางหู

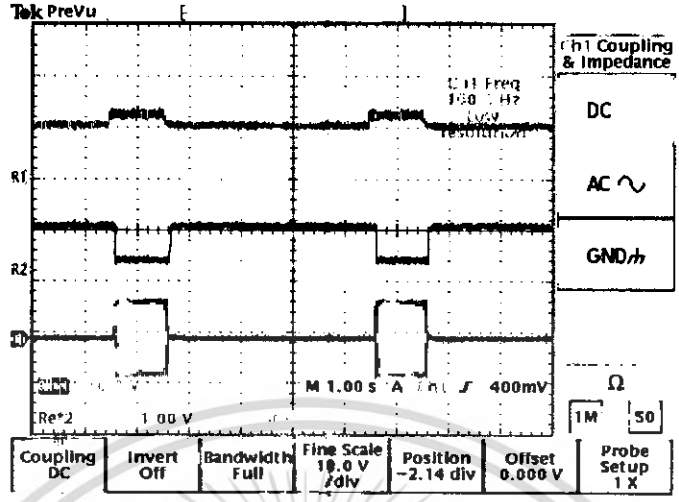


รูปที่ 4.2 แสดงสัญญาณเมื่อทำการยกหูโทรศัพท์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.2 การทดลองวงจรตรวจจับสัญญาณเรียก (Ringing Detector)

ทำการต่อวงจรตามรูปที่ 3.3 จะได้ผลการทดลองดังนี้



รูปที่ 4.3 แสดงสัญญาณเรียก (Ringing)

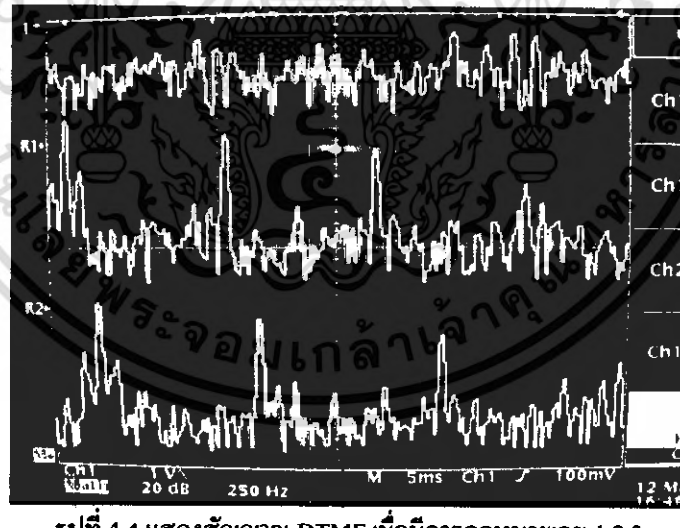
R1: สัญญาณเรียกเมื่อทำการวัดสัญญาณคร่อมระหว่างขา 1 และขา 2 ของออปโตไอโซเลเตอร์

R2: สัญญาณเอาท์พุทที่ได้จากการทดลอง

Ch1: สัญญาณอินพุทจากสายโทรศัพท์

4.3 การทดลองวงจรถอดรหัสหมายเลขโทรศัพท์ (DTMF Decoder)

ทำการต่อวงจรตามรูปที่ 3.4 ทำการวัดสัญญาณอินพุทได้สัญญาณดังนี้



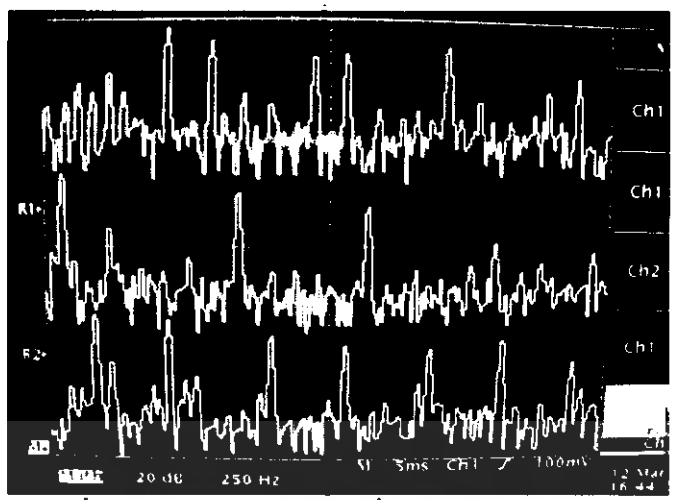
รูปที่ 4.4 แสดงสัญญาณ DTMF เมื่อมีการกดหมายเลข 1,2,3

R1: แสดงสัญญาณ DTMF เมื่อทำการกดเลข 1

R2: แสดงสัญญาณ DTMF เมื่อทำการกดเลข 2

M : แสดงสัญญาณ DTMF เมื่อทำการกดเลข 3

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.5 แสดงสัญญาณ DTMF เมื่อมีการกดหมายเลข 4,5,6

R1: แสดงสัญญาณ DTMF เมื่อทำการกดเลข 4

R2: แสดงสัญญาณ DTMF เมื่อทำการกดเลข 5

M : แสดงสัญญาณ DTMF เมื่อทำการกดเลข 6

ซึ่งจากการทดลองจะได้เอาท์พุทออกมาเป็นดังตารางต่อไปนี้

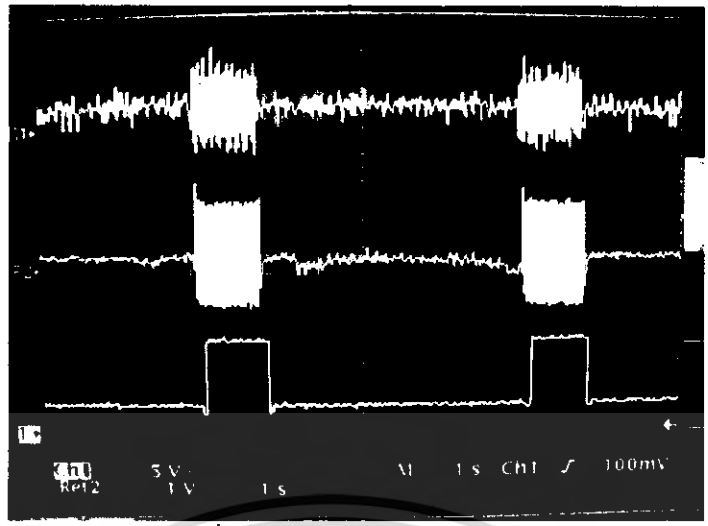
ปุ่ม	สัญญาณ STD ที่ขา 5		DTMF Decoder			
หมายเลข	กดปุ่ม	ไม่กดปุ่ม	Q4	Q3	Q2	Q1
1	1	0	0	0	0	1
2	1	0	0	0	1	0
3	1	0	0	0	1	1
4	1	0	0	1	0	0
5	1	0	0	1	0	1
6	1	0	0	1	1	0
7	1	0	0	1	1	1
8	1	0	1	0	0	0
9	1	0	1	0	0	1
0	1	0	1	0	1	0
*	1	0	1	0	1	1
#	1	0	1	5	0	0

ตารางที่ 4.1 แสดงผลการทดลองวงจรถอดรหัสเลขหมายโทรศัพท์

4.4 การทดลองวงจรตรวจจับสัญญาณเรียกกลับและสัญญาณไม่ว่าง (Ringback, Busy Tone Ddetector)

ทำการต่อวงจรดังรูปที่ 3.5 จะได้สัญญาณอินพุทเทียบกับสัญญาณเอาท์พุท โดยที่สัญญาณเอาท์พุทที่ได้จะเป็นลอจิก "1" นานประมาณ 1 วินาที และลอจิก "0" นานประมาณ 4 วินาที ดังรูปที่ 4.6

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.6 แสดงผลสัญญาณเรียกกลับ

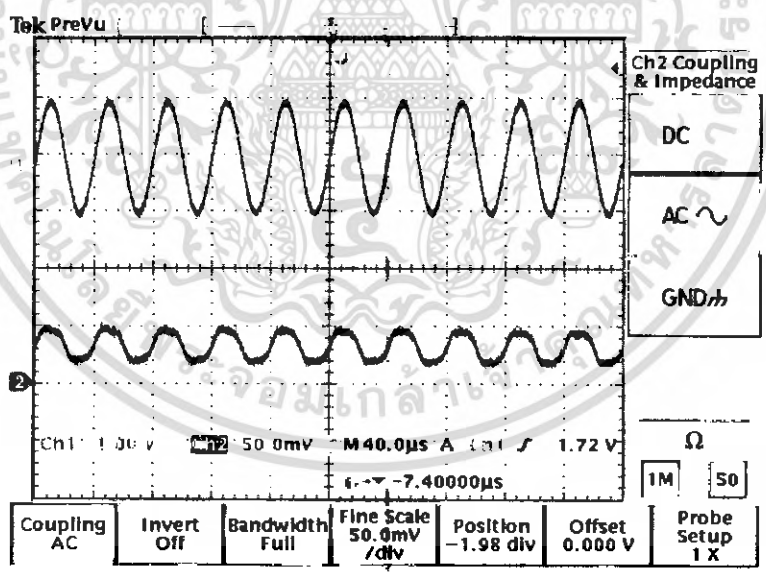
R1: แสดงเอาท์พุทของสัญญาณเรียกกลับ

R2: แสดงอินพุทของสัญญาณเรียกกลับ

4.5 การทดลองวงจรกรองความถี่และวงจรขยาย

4.5.1 วงจรบริบและขยายตามแรงดัน

ทำการต่อวงจรบริบและขยายตามแรงตามรูปที่ 3.8 ในส่วนของวงจรขยายแรงดัน โดยกำหนดให้อินพุทมีค่าแอมปลิจูด 1Vp มีความถี่ 25 kHz จะได้ผลการทดลองดังรูปที่ 4.7



รูปที่ 4.7 แสดงผลการทดลองวงจรบริบและขยายสัญญาณ

Ch1: สัญญาณอินพุท

Ch2: สัญญาณเอาท์พุทของวงจรบริบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากรูปที่ 4.7 จะเห็นว่าผลการทดลองจะได้สัญญาณเอาท์พุทเป็นไปตามทฤษฎีหัวข้อที่ 3.2.5.3 และได้สัญญาณเอาท์พุทเช่นเดียวกับรูปที่ 3.9

4.5.2 วงจร High Pass Filter

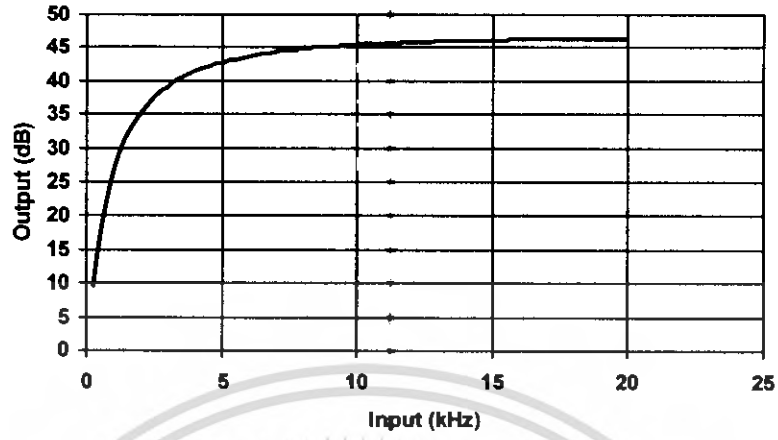
ทำการต่อวงจรดังรูปที่ 3.8 ในส่วนที่เป็น High Pass Filter โดยใช้ฟังก์ชั่นเจนเนอร์เรเตอร์เป็นตัวป้อนสัญญาณอินพุท โดยในการทดลองนี้จะกำหนดอินพุทให้มีแอมปลิจูด 100 mV ความถี่ 250 Hz จากนั้นทำการเพิ่มความถี่อินพุททีละ 250 Hz จนความถี่อินพุทมีค่าเป็น 20 kHz โดยให้แอมปลิจูดมีค่าคงที่ จะได้ผลการทดลองดังตารางต่อไปนี้

Input (kHz)	Output (mV)	Output (dB)	Input (kHz)	Output (mV)	Output (dB)	Input (kHz)	Output (mV)	Output (dB)	Input (kHz)	Output (mV)	Output (dB)
0.25	3	9.54	5.25	142	43.045	10.25	190	45.575	15.25	200	46.02
0.5	7	16.9	5.5	145	43.23	10.5	190	45.575	15.5	200	46.02
0.75	13	22.28	5.75	148	43.405	10.75	190	45.575	15.75	205	46.235
1	21	26.44	6	152	43.64	11	190	45.575	16	205	46.235
1.25	30	29.54	6.25	156	43.86	11.25	192	45.66	16.25	205	46.235
1.5	39	31.82	6.5	160	44.08	11.5	192	45.66	16.5	205	46.235
1.75	48	33.62	6.75	162	44.19	11.75	195	45.8	16.75	205	46.235
2	57	35.12	7	165	44.35	12	195	45.8	17	205	46.235
2.25	66	36.39	7.25	170	44.61	12.25	195	45.8	17.25	205	46.235
2.5	74	37.38	7.5	170	44.61	12.5	195	45.8	17.5	205	46.235
2.75	84	38.485	7.75	170	44.61	12.75	195	45.8	17.75	205	46.235
3	90	39.08	8	175	44.86	13	200	46.02	18	205	46.235
3.25	100	40	8.25	175	44.86	13.25	200	46.02	18.25	205	46.235
3.5	104	40.34	8.5	180	45.105	13.5	200	46.02	18.5	205	46.235
3.75	110	40.83	8.75	180	45.105	13.75	200	46.02	18.75	205	46.235
4	116	41.3	9	180	45.105	14	200	46.02	19	205	46.235
4.25	124	41.87	9.25	180	45.105	14.25	200	46.02	19.25	210	46.44
4.5	128	42.14	9.5	185	45.34	14.5	200	46.02	19.5	210	46.44
4.75	134	42.54	9.75	185	45.34	14.75	200	46.02	19.75	210	46.44
5	136	42.67	10	185	45.34	15	200	46.02	20	210	46.44

ตารางที่ 4.2 ผลการทดลอง High Pass Filter

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เมื่อนำผลการทดลองในตารางที่ 4.2 มาเขียนกราฟจะได้กราฟแสดงผลตอบสนองความถี่ของ High Pass Filter ดังรูปที่ 4.8



รูปที่ 4.8 กราฟแสดงผลตอบสนองความถี่ของ High Pass Filter

4.5.3 วงจร Low Pass Filter

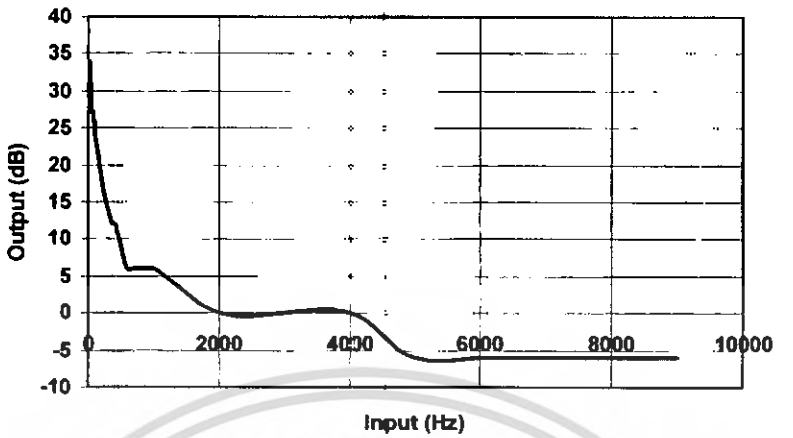
ทำการต่อวงจรดังรูปที่ 3.8 ในส่วนที่เป็น Low Pass Filter โดยใช้ฟังก์ชั่นเจนเนอร์เรเตอร์เป็นตัวป้อนสัญญาณอินพุท โดยในการทดลองนี้จะกำหนดอินพุทให้มีแอมปลิจูด 50 mV ความถี่ 10 Hz จากนั้นทำการเพิ่มความถี่อินพุทตามค่าอินพุทในตารางที่ 4.3 จนอินพุทมีความถี่เป็น 9 kHz โดยให้แอมปลิจูดมีค่าคงที่ จะได้ผลการทดลองดังตารางต่อไปนี้

Input (Hz)	Output (mV)	Output (dB)	Input (Hz)	Output (mV)	Output (dB)	Input (Hz)	Output (mV)	Output (dB)
10	50	33.98	200	8	18.06	2000	1	0
20	50	33.98	250	6	15.56	3000	1	0
30	40	32.04	300	5	13.98	4000	1	0
40	35	30.88	350	4	12.04	5000	0.5	-6.02
50	30	29.54	400	4	12.04	6000	0.5	-6.02
60	25	26.96	500	3	9.54	7000	0.5	-6.02
70	23	27.23	600	2	6.02	8000	0.5	-6.02
80	20	26.02	700	2	6.02	9000	0.5	-6.02
90	20	26.02	800	2	6.02			
100	16	24.08	900	2	6.02			
150	12	21.58	1000	2	6.02			

ตารางที่ 4.3 ผลการทดลอง Low Pass Filter

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

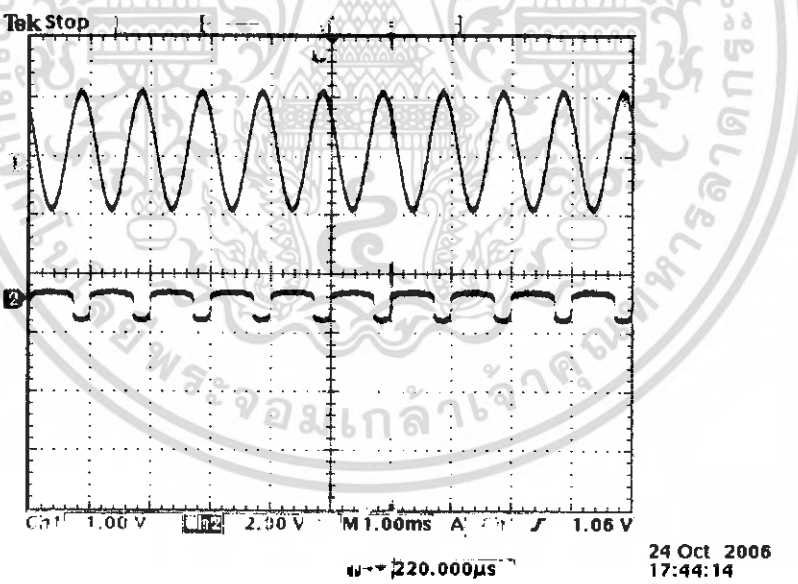
เมื่อนำค่าในตารางที่ 4.3 มาเขียนกราฟแสดงผลตอบสนองความถี่ของวงจร Low Pass Filter จะได้กราฟดังรูปที่ 4.9



รูปที่ 4.9 กราฟแสดงผลตอบสนองความถี่ของ Low Pass Filter

4.5.4 การทดลองวงจรขยายแถบความถี่กว้าง

ทำการต่อวงจรขยายแถบความถี่กว้างตามรูปที่ 3.8 ในส่วนของวงจร Wide Band Amplifier โดยกำหนดให้สัญญาณอินพุทมีความถี่ 1 kHz และมีแรงดันประมาณ 1.6V ทำการวัดสัญญาณอินพุทจะได้สัญญาณอินพุทในชาแนลที่ 1 และจะได้เอาท์พุทในชาแนลที่ 2 ดังรูปที่ 4.10



รูปที่ 4.10 แสดงผลการทดลองวงจรขยายแถบความถี่กว้าง

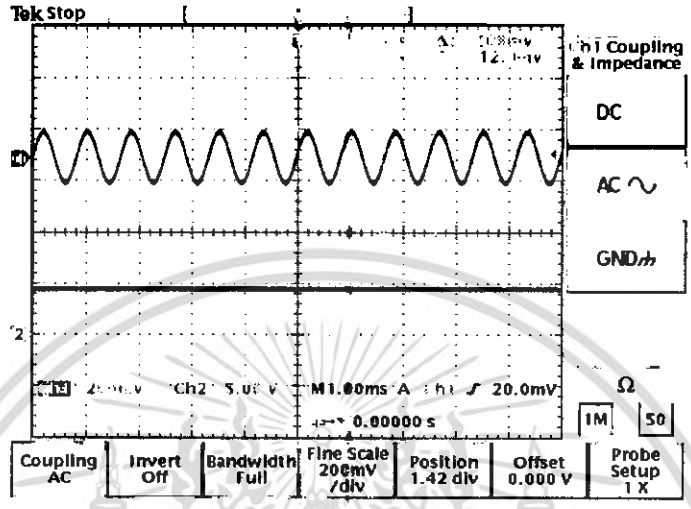
Ch1: สัญญาณอินพุท

Ch2: สัญญาณเอาท์พุทของวงจรขยายแถบความถี่กว้าง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.6 การทดลองวงจรถอดรหัส FSK

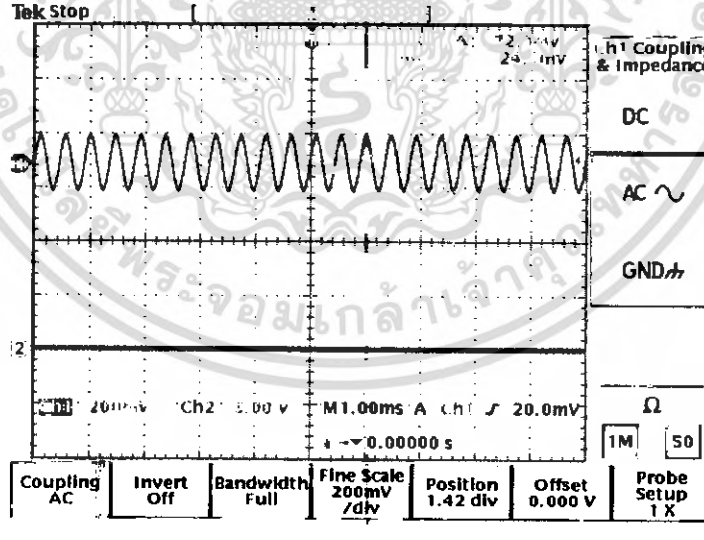
ทำการทดลองโดยการต่อวงจรตามรูปที่ 3.10 ทำการใส่สัญญาณอินพุทความถี่ 1,200 Hz จะได้สัญญาณเอาท์พุทมีค่าเป็น 5V หรือมีค่าลอจิกเป็น "1" ดังรูปที่ 4.11 และเมื่อทำการเพิ่มความถี่ของสัญญาณอินพุทจนมีค่าเป็น 2,200 Hz จะได้สัญญาณเอาท์พุทมีค่าเป็น 0V หรือมีค่าลอจิกเป็น "0" ดังรูปที่ 4.12



รูปที่ 4.11 แสดงสัญญาณเอาท์พุทมีค่าลอจิกเป็น "1" เมื่อสัญญาณอินพุทมีค่าเป็น 1,200 Hz

Ch1: สัญญาณอินพุท

Ch2: สัญญาณเอาท์พุทของวงจรถอดรหัส FSK เมื่ออินพุทมีค่าเป็น 1,200 Hz



รูปที่ 4.12 แสดงสัญญาณเอาท์พุทมีค่าลอจิกเป็น "0" เมื่อสัญญาณอินพุทมีค่าเป็น 2,200 Hz

Ch1: สัญญาณอินพุท

Ch2: สัญญาณเอาท์พุทของวงจรถอดรหัส FSK เมื่ออินพุทมีค่าเป็น 2,200 Hz

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.7 ผลการทดลองโปรแกรมส่วนแสดงเบอร์โทรออก

1. เมื่อเปิดเครื่องหน้าจอ LCD จะแสดงวันที่และเวลา ดังรูปที่ 4.13

รูปที่ 4.13 แสดงรูปจอ LCD ที่แสดงวันที่และเวลา

2. เมื่อเราทำการกด Keypad ปุ่มใดปุ่มหนึ่งหน้าจอ LCD จะแสดงเมนู 4 รายการ คือ

1) ตั้งเวลาและวันที่

2) แสดงเบอร์ที่โทรออก วัน เวลา และระยะเวลาการใช้งาน

3) แสดงเบอร์โทรเข้าที่ได้รับสาย วัน เวลา และระยะเวลาการใช้งาน

4) แสดงเบอร์โทรเข้าที่ไม่ได้รับสาย วัน และเวลา

รูปที่ 4.14 แสดงรูปจอ LCD ที่แสดงรายการการใช้งาน

3. ขณะที่อยู่หน้าเมนูเมื่อทำการกดหมายเลข 1 จะเข้าสู่การตั้งค่าเวลาและวันที่

รูปที่ 4.15 แสดงรูปจอ LCD ที่แสดง

4. เมื่อทำการยกหูโทรศัพท์หน้าจอจะแสดงคำว่า Pick up และแสดงเลขหมายที่เราทำการกด

รูปที่ 4.16 แสดงรูปจอ LCD ที่แสดงการยกหูและเลขหมายที่ถูกกด

5. ถ้าเราทำการกดหมายเลขโทรศัพท์ไม่เสร็จภายในเวลา 30 วินาที หน้าจอ LCD จะแสดงคำว่า Put Down Please เพื่อให้เราทำการวางหูแล้วกดเลขหมายโทรศัพท์อีกครั้ง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 4.17 แสดงรูปจอ LCD เมื่อทำการกดเลขหมายไม่เสร็จภายในเวลา 30 วินาที

6. ถ้าเราทำการกดเลขหมายจนครบ แต่ทางด้านรับไม่รับสายภายในเวลา 20 วินาที หน้าจอ LCD จะแสดงคำว่า Put Down Please เพื่อให้เราทำการวางหูแล้วทำการโทรออกอีกครั้ง ซึ่งการโทรออกครั้งนี้จะไม่มีการบันทึกการใช้งานเอาไว้

รูปที่ 4.18 แสดงรูปจอ LCD เมื่อทางด้านรับไม่รับสายภายในเวลา 20 วินาที

7. ถ้าด้านรับทำการรับสายภายในเวลา 20 วินาที หน้าจอ LCD จะแสดงคำว่า Pick up Received เพื่อให้เราทราบว่ามีการรับสายและเครื่องบันทึกการใช้งานโทรศัพท์ได้เริ่มทำการบันทึกการใช้งานแล้ว

รูปที่ 4.19 แสดงรูปจอ LCD เมื่อด้านรับทำการรับสายภายในเวลา 20 วินาที

8. เมื่อเราต้องการดูการใช้งานที่เราบันทึกไว้ เราก็ไปที่เมนูกด 2 หน้าจอ LCD จะแสดงคำว่า Select Record เพื่อเราเลือกดูเบอร์ที่เราโทรออก เช่นถ้าเราต้องการดู Record ที่ 3 เราก็ทำการกดเลข 3 เพื่อดู record ที่ 3 ที่ถูกบันทึกเอาไว้

รูปที่ 4.20 แสดงรูปจอ LCD ที่แสดงข้อความให้เลือกดูเบอร์ที่โทรออกที่ทำการบันทึกเอาไว้

9. เมื่อเราทำการเลือก Record ที่ต้องการได้แล้วหน้าจอ LCD จะแสดงเบอร์ที่โทรออก ระยะเวลาการใช้งาน (หน่วยเป็นวินาที) เวลาที่ทำการโทรออก และวันที่โทรออก เมื่อทำการบันทึกครบ 5 หมายเลข เครื่องบันทึกการใช้งานโทรศัพท์จะทำการบันทึกซ้ำที่ตำแหน่งที่ 1

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 4.21 แสดงรูปจอ LCD ที่แสดงหมายเลขโทรศัพท์ ระยะเวลาที่โทร เวลา และวันที่ที่ทำการโทรออก

10. ในกรณีที่ยังไม่มีการบันทึกข้อมูลหน้าจอ LCD จะแสดงคำว่า No Data

รูปที่ 4.22 แสดงรูปจอ LCD เมื่อยังไม่มีการบันทึกข้อมูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 5
สรุปและวิจารณ์ผลการทดลอง

จากการทดลองวงจรดีเทคสัญญาณต่าง ๆ จะเห็นได้ว่าสัญญาณที่ได้นั้นถูกต้องเป็นไปตามทฤษฎี แต่มีปัญหาที่วงจรเรียกกลับ คือ ถ้าเราโทรเข้าโทรศัพท์เคลื่อนที่เครื่องที่ใช้เสียงเพลงรอสาย (Calling Melody) จะไม่สามารถจับสัญญาณเรียกกลับได้

จากการทดลองวงจรขริบและขยายตามแรงดันจะเห็นได้ว่าสัญญาณเอาท์พุทที่ได้มีลักษณะเช่นเดียวกันกับสัญญาณเอาท์พุทตามรูปที่ 3.8

ในการทดลองส่วนของวงจรHigh Pass Filter จะเห็นได้ว่าเมื่อเราทำการเพิ่มความถี่สูงขึ้นเรื่อย ๆ สัญญาณก็จะมีแรงดันเพิ่มมากขึ้นจนถึงความถี่ที่สูงมากพอที่จะสามารถผ่านวงจรนี้ไปได้ ค่าแรงดันก็จะมีการเปลี่ยนแปลงน้อยลงจนเกือบจะมีค่าคงที่เมื่อนำค่าแรงดันของสัญญาณเอาท์พุทคิดค่ากำลังงานในหน่วย dB เพื่อหากราฟผลตอบสนองทางความถี่จะได้กราฟรูปเอ็กซ์โพแนนเชียล เช่นเดียวกันรูปที่ 3.10 B)

และในส่วนของวงจร Low Pass Filter ก็จะได้กราฟเอ็กซ์โพแนนเชียล แต่เนื่องจากในการทดลองได้ทำการวัดค่าไม่ละเอียดพอจึงทำให้กราฟไม่เป็นเส้นโค้งเช่นเดียวกับรูปที่ 3.10A) ดังนั้น ถ้าต้องการให้กราฟออกมาเป็นเส้นโค้งเช่นเดียวกันรูปที่ 3.10 A) จะต้องทำการวัดค่าให้ละเอียดเช่นเดียวกับการทดลองวงจร High Pass Filter

ในการทดลองวงจรกลับขั้วสัญญาณจะเห็นว่าเมื่อสัญญาณอินพุทผ่านไดโอดสัญญาณจะโดนขริบออกและเมื่อผ่านออปแอมป์จะได้สัญญาณกลับเฟสดังรูปที่ 4.11 โดยสัญญาณเอาท์พุทจะความถี่เท่ากับสัญญาณอินพุท

ในส่วนของการทดลองวงจรถอดรหัส FSK จะเห็นได้ว่าเมื่อความถี่มีค่า 1,200 Hz จะได้เอาท์พุทเป็นลอจิก "1" และเมื่อความถี่มีค่าเป็น 2,200 Hz จะได้เอาท์พุทเป็นลอจิก "0" ซึ่งในความเป็นจริงแล้วสัญญาณ FSK จะมีความถี่สอง 2 ค่าส่งมาพร้อมกัน ดังนั้น สัญญาณเอาท์พุทจริง ๆ ที่ออกมาจะเป็นสัญญาณพัลส์ที่มีลอจิกเป็น "0" และ "1"

แนวทางในการพัฒนาควรจะมีการเชื่อมต่อกับคอมพิวเตอร์เพื่อเก็บข้อมูลไว้ในคอมพิวเตอร์และสามารถพิมพ์ข้อมูลทั้งหมดที่ต้องการออกมาได้ด้วย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เอกสารอ้างอิง

[1] ธวัชชัย เลื่อนฉวี.เทคโนโลยีโทรศัพท์.พิมพ์ครั้งที่ 3. กรุงเทพฯ: สำนักพิมพ์ศุภลัย มีเดีย, 2533

[2] อุดม รานอก. ภาษา C สำหรับงานควบคุมมไมโครคอนโทรลเลอร์ MCS-51.นนทบุรี: ไอดีซี,2548

[3] ผศ.ธีรวัฒน์ ประกอบผล. การพัฒนาไมโครคอนโทรลเลอร์ด้วยภาษาซี.พิมพ์ครั้งที่ 6. กรุงเทพฯ: สมาคมส่งเสริมเทคโนโลยี (ไทย - ญี่ปุ่น), 2545

[4] สักรียา ชิตวงศ์. วิศวกรรมอิเล็กทรอนิก.พิมพ์ครั้งที่ 1. กรุงเทพฯ: ห้างหุ้นส่วนจำกัด วี.เจ.พริ้นติ้ง, 2544

[5] http://www.exar.com/products/tan_008.pdf

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1. โปรแกรมเครื่องบันทึกการใช้งานโทรศัพท์ส่วนที่บันทึกการโทรออก

```
#include "reg52.h"
#include "absacc.h"
unsigned char  TempP2;
#include "Function.c"
#include "LCD.c"
#include "MT8870.c"
#include "I2C.C"
#include "DS1307.c"
unsigned int Times,Time;
unsigned char Tel[10],RecNo;
unsigned char idata Rec[5][20];
main()
{ unsigned  char i,Data,Key,Max;
  bit  Flag;
   Init_LCD();
   Init_Timer0(15);
   TempP2=0xFF;
   RecNo = 0;
   Init_MT8870();
   Rx_Mode();
   Wr_String("Time =          ",1);
   Wr_String("Date =          ",2);
  while(1)
  {   while(Scan_Key()==0xFF)
      {
          Rd_Time();
          ShowTime(1,9);
          ShowDate(2,9);
          if(PickUp==0)
          {
            Hook_On();
            Wr_String("    Pick up     ",1);
            Wr_String("No.=            ",2);
            Goto_XY(2,7);
            //------Check  KeyPress -------//
            Flag = 1;
            i = 0; Max =9;
            Time = 0;
            while((i<Max)&&(Flag==1))
            { if(Time >=  10){Flag=0;}
              if((RD_Status() & 0X04)==0X04)
                {Sound();
                  Data  =     RD_DTMF();
                   if(Data==10){Data=0;}
                   {Wr_Char(Data+0x30);}
                   Tel[i] = Data;
                   i++;
                   if((i==2)&&(Data>=8)){Max = 10;}
                }
          }
            //------Check  Ringback -------//
            if(Flag==1)
            {
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        Init_MT8870();
     Tx_Mode();
         Time = 0;
        while((Time < 30)&&(Check_Tone()!=0)){}
        if(Time < 30)
        { Time =0;
          Hook_Off();
          Wr_String("    Recieved    ",2);
          Sound();
          while(PickUp==0){}
          if(RecNo==5){RecNo=0;}
          for(i=0;i<Max;i++){Rec[RecNo][i] = Tel[i]+0X30;}
          if(Max==9){Rec[RecNo][9]=0X20;}
          Rec[RecNo][10]=Time>>8;    //  Time od used
          Rec[RecNo][11]=Time&0xFF;
          Rec[RecNo][12]=Hour;
          Rec[RecNo][13]=Min;
          Rec[RecNo][14]=Sec;
          Rec[RecNo][15]=Date;
          Rec[RecNo][16]=Mon;
          Rec[RecNo][17]=Year;
          Rec[RecNo][18]='V';
          RecNo++;
        }
       }
       Hook_Off();
       Wr_String("Put Down  Please",1);
       while(PickUp==0){}
       Init_MT8870();
    Rx_Mode();
       Wr_String("Time =          ",1);
     Wr_String("Date =           ",2);
    }
   }
   Sound();
Key_Off();
   Wr_String("1)Time  3)Revc",1);
   Wr_String("2)Call  4)Miss",2);
Key =KeyPress();
   Wr_String("                ",2);
   if(Key==1)
   {Wr_String("Time =          ",1);
    Wr_String("Date =          ",2);
    GetTime(1,9);
    GetDate(2,9);
    Wr_Time();
   }else
   if(Key==2)
   { Wr_String(" Select  Record ",1);
     Wr_String("                ",2);
     Key = KeyPress();
     Wr_String("                ",1);
     Goto_XY(1,1);
     if(Rec[Key-1][18]=='V')
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
            {
            for(i=0;i<10;i++){Wr_Char(Rec[Key-1][i]);}
            Goto_XY(1,12);
            ShowNum(((Rec[Key-1][10]*256)+Rec[Key-1][11]),5);
            Hour = Rec[Key-1][12];
            Min  = Rec[Key-1][13];
            Sec  = Rec[Key-1][14];
            Date = Rec[Key-1][15];
            Mon  = Rec[Key-1][16];
            Year = Rec[Key-1][17];
            ShowTime(2,1);
            Goto_XY(2,6);Wr_Char(' ');Wr_Char(' ');Wr_Char(' ');
            ShowDate(2,9);
            KeyPress();
            }else{Wr_String("    No  Data     ",1);KeyPress();}
          }else{Wr_String("    No  Data     ",1);KeyPress();}
           Wr_String("Time =          ",1);
           Wr_String("Date =          ",2);
    }
}
void  Timer0_Int() interrupt 1
{TH0=_TH0;
 TL0=_TL0;
 Times++;
 if(Times == 15)
 { Times=0;
   Time++;
 }
}
*****************************Function************************
#include "string.h"
sbit Row0          =     P1^0;
sbit Row1          =     P1^1;
sbit Row2          =     P1^2;
sbit Row3          =     P1^3;
unsigned  char  data BufDisp[7];
void Wait(unsigned int x)
{unsigned int i;
      for(i=0;i<x;i++) {}
}
sbit Piezo  = P3^6;
void Sound(void)
{unsigned char i;
      for (i=0;i<50;i++)
        {
            Piezo =     ~Piezo;
          Wait(50);
        }
}
sbit Piezo2  = P2^5;
void SoundSmart(void)
{unsigned char i;
      for (i=0;i<50;i++)
        {
            Piezo2      =     ~Piezo2;
          Wait(50);
        }
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
}
/* ------- Scan  Key Sw. ------------ */
unsigned  char  Scan_Key(void)
{
      P1    =     0xEF;
      if (Row0 == 0) {return(11);}
      if (Row1 == 0) {return(9);}
      if (Row2 == 0) {return(6);}
      if (Row3 == 0) {return(3);}

      P1    =     0xDF;
      if (Row0 == 0) {return(0);}
      if (Row1 == 0) {return(8);}
      if (Row2 == 0) {return(5);}
      if (Row3 == 0) {return(2);}
      P1    =     0xBF;
      if (Row0 == 0) {return(10);}
      if (Row1 == 0) {return(7);}
      if (Row2 == 0) {return(4);}
      if (Row3 == 0) {return(1);}
      return(0xff);
      P1    =     0x7F;
      if (Row0 == 0) {return(15);}
      if (Row1 == 0) {return(14);}
      if (Row2 == 0) {return(13);}
      if (Row3 == 0) {return(12);}
      return(0xff);
}
/* ---   Delay ----------- */
void  Delay(unsigned int x )
{ unsigned int   i,j;
      for (i=0;i<x;i++)
       {
          for (j=0;j<115;j++) { }
       }
}
void   Key_Off(void)
{
      while (Scan_Key() != 0xFF) { }
      Delay(50);
}
unsigned char KeyPress(void)
{unsigned char Key;
      while (Scan_Key() == 0xFF) { }
    Key = Scan_Key();
      Sound();
    Key_Off();
      return(Key);
}
code unsigned char BufSEG[] = {0xC0,0xF9,0xA4,0xB0,0x99,0x92,
                               0x82,0xF8,0x80,0x90,0xFF,0xC6};
void  Display (void)
{
          P2 = 0xFF;          /* Display off */
        P0 = BufSEG[BufDisp[3]];
        P2 = 0xEF;            /*  Scan   7 - Segment  */
          Delay(4);
          P2 = 0xFF;          /* Display off */
        P0 = BufSEG[BufDisp[2]];
        P2 = 0xDF;            /*  Scan   7 - Segment  */
          Delay(4);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
          P2 = 0xFF;          /* Display off */
        P0 = BufSEG[BufDisp[1]];
        P2 = 0xBF;            /*  Scan   7 - Segment  */
          Delay(4);
          P2 = 0xFF;          /* Display off */
        P0 = BufSEG[BufDisp[0]];
        P2 = 0x7F;      /*  Scan   7 - Segment  */
          Delay(4);
            P2 = 0xFF;
}
void  CountToDisp( unsigned  long  Count)
{
            BufDisp[6] =      Count / 1000000;
            Count    =  Count % 1000000;
            BufDisp[5] =      Count / 100000;
            Count    =  Count % 100000;
            BufDisp[4] =      Count / 10000;
            Count    =  Count % 10000;
            BufDisp[3] =      Count / 1000;
            Count    =  Count % 1000;
            BufDisp[2] =      Count / 100;
            Count    =  Count % 100;
            BufDisp[1] =      Count / 10;
            BufDisp[0] =      Count % 10;
}
void  CountToDisp2( unsigned  long  Count)
{
    BufDisp[6] =  Count / 1000000;
      if(BufDisp[6] == 0) BufDisp[6] = 10;
      Count    =  Count % 1000000;
      BufDisp[5] =      Count / 100000;
      if((BufDisp[6] == 10) && (BufDisp[5] == 0)) BufDisp[5] = 10;

      Count    =  Count % 100000;
      BufDisp[4] =      Count / 10000;
      if((BufDisp[5] == 10) && (BufDisp[4] == 0)) BufDisp[4] = 10;

    Count      =  Count % 10000;
      BufDisp[3] =      Count / 1000;
      if((BufDisp[4] == 10) && (BufDisp[3] == 0)) BufDisp[3] = 10;

      Count    =  Count % 1000;
      BufDisp[2] =      Count / 100;
      if((BufDisp[3] == 10) && (BufDisp[2] == 0)) BufDisp[2] = 10;

      Count    =  Count % 100;
      BufDisp[1] =      Count / 10;
      if((BufDisp[2] == 10) && (BufDisp[1] == 0)) BufDisp[1] = 10;

      Count    =  Count % 10;
      BufDisp[0] =      Count;
}
void ShowVal(unsigned int Num)
{
  CountToDisp(Num);
  Display();
}
void  Tx_Byte(unsigned char Data)
{
      TI    =     1;
      SBUF  =     Data;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
      TI    =     0;
      while (!TI) {}
      TI    =     0;
}
unsigned char Rx_Byte(void)
{unsigned char Data;
      while (!RI) {}
      RI    =     0;
      Data  =     SBUF;
      return(Data);
}
void Init_Serial(unsigned  char  br)
{
      SCON  =     0x50;
    TMOD    =   (TMOD & 0x0F)+0x20;
      TH1         =     0xFD;
      TR1         =   1;
      EA          =     1;
    ES      =   1;
    if (br == 48)     TH1  =  0xFA;
    if (br == 96)     TH1  =  0xFD;
    if (br == 192)    {PCON = 0x80;} else {PCON = 0x00;}
}
void   Tx_String(unsigned char  *dptr)
{unsigned char i,Count;
    Count = strlen(dptr);
    for(i=0;i<Count;i++)
     {
       Tx_Byte(*dptr);
       dptr++;
     }
}
unsigned char _TH0,_TL0;
void Init_Timer0(unsigned int Freq)
{unsigned int Count;
   Count =  65536 - (921600 / Freq);
   TH0   =  Count >> 8;
   TL0   =  Count & 0xFF;
   _TH0  =  TH0;
   _TL0  =  TL0;
   TMOD  =  (TMOD & 0xF0)+0x01;
   TR0   =  1;
   ET0   =  1;
   EA    =  1;
}
void  Timer0_IntOn(void)
{
   ET0 = 1;
}
void  Timer0_IntOff(void)
{
   ET0 = 0;
}
void  Timer0_On(void)
{
   TR0 = 1;
}
void  Timer0_Off(void)
{
   TR0 = 0;
}
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
void  Serial_IntOn(void)
{
   ES = 1;
}
void  Serial_IntOff(void)
{
   ES = 0;
}
**********************************LCD************************
sbit  EN_LCD      =     P0^3;
sbit  RS_LCD      =     P0^2;
sbit    LIGHT     =   P0^1;
bit     Light;
void  Wr_Register(unsigned char Data)
{unsigned char Temp;
    Temp    =     Data;
      Temp  =     Temp | 0x0F;
      P0            =     Temp;
    if (Light) { LIGHT = 0;}
      RS_LCD        =     0;
      EN_LCD        =     0;
      Delay(1);
      EN_LCD        =     1;
      Temp  =     Data;
      Temp  =   Temp << 4;
      Temp  =     Temp | 0x0F;
      P0            =     Temp;
    if (Light) { LIGHT = 0;}
      RS_LCD        =     0;
      EN_LCD        =     0;
      Delay(1);
      EN_LCD        =     1;
}
void  Wr_Char(unsigned char Data)
{unsigned char Temp;
    Temp    =     Data;
      Temp  =     Temp | 0x0F;
      P0            =     Temp;
    if (Light) { LIGHT = 0;}
      RS_LCD        =     1;
      EN_LCD        =     0;
      Delay(1);
      EN_LCD        =     1;
      Temp  =     Data;
      Temp  =   Temp << 4;
      Temp  =     Temp | 0x0F;
      P0            =     Temp;
      if (Light) { LIGHT = 0;}
      RS_LCD        =     1;
      EN_LCD        =     0;
      Delay(1);
      EN_LCD        =     1;
}
void  Init_LCD(void)
{
      Delay(200);
      Wr_Register(0x33);
      Wr_Register(0x32);
      Wr_Register(0x28);
      Wr_Register(0x0C);
      Wr_Register(0x06);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
      Wr_Register(0x01);
      Delay(100);
}
void        Goto_XY(unsigned char x,y)
{
      Wr_Register(0x80+(0x40 *(x-1))+(y-1));
}
void   Wr_String(unsigned char  *dptr,Row)
{unsigned  char   i;
      Goto_XY(Row,1);
           for (i=0;i<16;i++)
             {    Wr_Char(*dptr);
                        dptr++;
             }
 }
void  Cursor_Off(void)
{
      Wr_Register(0x0C);
}
void  Cursor_On(void)
{
      Wr_Register(0x0F);
}
unsigned  BufKey[4];
void  ReadKey(unsigned  Count,Row,Col)
{unsigned char i;
   Key_Off(); Delay(100);
   Cursor_On();
   Goto_XY(Row,Col);
   i = 0;
   while(i<Count)
    {
       if(Key_Press() < 10)
        {
          BufKey[i] = Key_Press();
          Wr_Char(BufKey[i]+0x30);
          Sound(); Key_Off();
          i++;
        }
    }
}
unsigned int Key2Num(void)
{
  return( (BufKey[0]*1000)+(BufKey[1]*100)+(BufKey[2]*10)+BufKey[3]
);
}
void  ShowNum(unsigned int Count,unsigned char Row,Col)
{
   CountToLCD(Count);
   Goto_XY(Row,Col);
   Wr_Char(BufDisp[0]);
   Wr_Char(BufDisp[1]);
   Wr_Char(BufDisp[2]);
   Wr_Char(BufDisp[3]);
}
*****************************MT887C*****************************
/*  P2.4 - P2.7 -> D0 - D3  */
sbit  RS_Tel  =   P3^0;
sbit  WR_Tel      =     P3^1;
sbit  RD_Tel      =     P3^2;
sbit  CS_Tel      =     P3^5;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
sbit  Tone  =     P2^0;
sbit  Ring  =     P2^1;
sbit  Hook  =     P2^2;
sbit  PickUp      =     P2^3;
void CheckRing(unsigned char Count)
{unsigned i;
      i     =     0;
      while (i < Count)
       {
            while(Ring) {}
            i++;
            Delay(1000);
      }
}
/* ---------------------------- */
unsigned  char    RD_Status(void)
{unsigned char  Data;
      P2          =     0xFF & TempP2;
      RS_Tel      =     1;
      RD_Tel      =     1;
      WR_Tel      =   1;
      CS_Tel      =     0;Wait(50);
      RD_Tel      =     0;Wait(50);
      Data  =     P2;
      Data  =     (Data >> 4) & 0x0F;
      RD_Tel      =     1;Wait(50);
      CS_Tel      =     1;Wait(50);
      return(Data);
}
/* ---------------------------- */
void  WR_RegA(unsigned char Data)
{
      RS_Tel      =     1;
      RD_Tel      =     1;
      WR_Tel      =     1;
      CS_Tel      =   0;Wait(50);

      WR_Tel      =     0;Wait(50);
      P2          =     ((Data << 4) | 0x0F)& TempP2;
      WR_Tel      =     1;Wait(50);
      CS_Tel      =     1;Wait(50);
}
/* ---------------------------- */
unsigned  char    RD_DTMF(void)
{unsigned char  Data;
      P2          =     0xFF & TempP2;;
      RS_Tel      =     0;
      RD_Tel      =     1;
      WR_Tel      =     1;
      CS_Tel      =   0;Wait(50);
      RD_Tel      =     0;Wait(50);
      Data  =     P2;
      Data  =     (Data >> 4) & 0x0F;
      RD_Tel      =     1;Wait(50);
      CS_Tel      =     1;Wait(50);
      return(Data);
}
/* ---------------------------- */
void WR_DTMF(unsigned char Data)
{
    if(Data == 0){Data = 10;}
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
      RS_Tel      =     0;
      RD_Tel      =     1;
      WR_Tel      =     1;
      CS_Tel      =   0;Wait(50);

      WR_Tel      =     0;Wait(50);
      P2          =     ((Data << 4) | 0x0F)& TempP2;
      WR_Tel      =     1;Wait(50);
      CS_Tel      =     1;Wait(50);
}
/* ---------------------------- */
void  Init_MT8870(void)
{
      Delay(100);
      RD_Status();
      WR_RegA(0);
      WR_RegA(0);
      WR_RegA(0x0C);
      WR_RegA(0);
      RD_Status();
    WR_DTMF(0);
      Delay(100);
}
/* ---------------------------- */
void  Rx_Mode(void)
{
      WR_RegA(0x04);
}
/* ---------------------------- */
void  Tx_Mode(void)
{
      WR_RegA(0x0F);
      WR_RegA(0x00);
}
/* check Tone number for Busy Tone & Dail Tone */
unsigned char Check_Tone(void)
{unsigned char  Count,Index;
      Count = 0;
      Index = 0;
      Tone  = 1;
      while(Index<50)
      {
            Index++;
            Delay(100);
            if(Tone==1)
            {
                  Count++;
                  //WR_DTMF(0);
                  Delay(500);
                  Index= Index+5;
            }
      }

     return(Count);
}
void Hook_On(void)
{
      Hook  =     0;
      TempP2  = 0xFB;
}
void Hook_Off(void)
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
{
      Hook  =     1;
      TempP2  = 0xFF;
}
*********************************I2C*****************************
sbit SDA  = P3^4;
sbit SCL  = P3^3;
void  I2c_Wait(void)
{ unsigned  char  i;
          for (i=0;i<60;i++) { }
}
void  I2c_Clk(void)
{     SCL   =     1;
      I2c_Wait();
      SCL   =     0;
      I2c_Wait();
}
void  I2c_Ack(void)
{     SDA   = 0;
      I2c_Clk();
}
void  I2c_Nack(void)
{     SDA   =  1;
      I2c_Clk();
}
void  I2c_Start(void)
{     SCL   =     0;
      SDA   =     1;
      SCL   =     1;
      I2c_Wait();
      SDA   =     0;
      I2c_Wait();
}
void  I2c_Stop(void)
{     SCL   =     0;
      SDA   =     0;
      SCL   =     1;
      I2c_Wait();
      SDA   =     1;
      I2c_Wait();
}
void  I2c_Write(unsigned char Data )
{ unsigned char  i;
  bit out;
        SCL =     0;
      for (i=0;i<8;i++)
      {
         out      =     Data & 0x80;
         Data     =     Data << 1;
         SDA      =     out;
         I2c_Clk();
      }
        SDA      =     1;
        I2c_Clk();
}
unsigned  char  I2c_Read(void)
{ unsigned char   Data,i;
  bit out;
      SCL   =     0;
      SDA   =     1;
      I2c_Wait();
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
      for (i=0;i<8;i++)
         {
            SCL   =     1;
            I2c_Wait();
            out   =     SDA;
            Data  =     Data << 1;
            Data  =     Data |  out;
            SCL   =     0;
            I2c_Wait();
         }
          return(Data);
}
*********************************DS1307*****************************
unsigned char     S_Sec,S_Min,S_Hour,S_Day,S_Date,S_Mon,S_Year;
unsigned char     Sec,Min,Hour,Day,Date,Mon,Year;
void  Wr_Time(void)
{
      I2c_Start();
      I2c_Write(0xD0);
      I2c_Write(0x00);
      I2c_Write(S_Sec);
      I2c_Write(S_Min);
      I2c_Write(S_Hour);
      I2c_Write(S_Day);
      I2c_Write(S_Date);
      I2c_Write(S_Mon);
      I2c_Write(S_Year);
      I2c_Stop();
}
void  Rd_Time(void)
{     I2c_Start();
      I2c_Write(0xD0);
      I2c_Write(0x00);
      I2c_Start();
      I2c_Write(0xD1);
      Sec  =      I2c_Read();  I2c_Ack();
      Min  =  I2c_Read();  I2c_Ack();
      Hour =  I2c_Read();  I2c_Ack();
      Day  =  I2c_Read();  I2c_Ack();
      Date =  I2c_Read();  I2c_Ack();
      Mon  =  I2c_Read();  I2c_Ack();
      Year =  I2c_Read();  I2c_Nack();
      I2c_Stop();
}
void GetTime(unsigned char X,Y)
{unsigned char i,Key;
 Cursor_On();
 Goto_XY(X,Y);
 Wr_Char(' ');Wr_Char(' ');
 Wr_Char(':');
 Wr_Char(' ');Wr_Char(' ');
 Wr_Char(':');
 Wr_Char(' ');Wr_Char(' ');
 Goto_XY(X,Y);
 i = 0;
 Key = 0;
 while(Key != 11)
  {
   Cursor_On();
   while (Scan_Key() == 0xFF) { }
   Key = Scan_Key();
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    Sound();
    if((Key < 10)&&(i<6))
     {   BufDisp[i] = Key;
         Wr_Char(Key+0x30);
         i++;
         if(i==2)
           {
               Wr_Char(':');
               S_Hour = (BufDisp[0]<<4) + BufDisp[1];
               if(S_Hour > 0x23){goto Retry;}
           }
         if(i==4)
           {
               Wr_Char(':');
               S_Min = (BufDisp[2]<<4) + BufDisp[3];
               if(S_Min > 0x59)
                 {
                     Goto_XY(X,Y+3);
                             Wr_Char(' ');Wr_Char(' ');
                     Goto_XY(X,Y+3);
                     i = 2;
                 }
           }
          if(i==6)
           {
               Wr_Char(':');
               S_Sec = (BufDisp[4]<<4) + BufDisp[5];
               if(S_Sec > 0x59)
                 {
                     Goto_XY(X,Y+6);
                             Wr_Char(' ');Wr_Char(' ');
                     Goto_XY(X,Y+6);
                     i = 4;
                 }
           }
     }
    if(Key == 10)
     { Retry:
        Goto_XY(X,Y);
        Wr_Char(' ');Wr_Char(' ');
        Wr_Char(':');
        Wr_Char(' ');Wr_Char(' ');
        Wr_Char(':');
        Wr_Char(' ');Wr_Char(' ');
        i=0;
        Goto_XY(X,Y);
     }
     Key_Off();
   }
  }
void  ShowTime(unsigned X,Y)
{
  Goto_XY(X,Y);
  Wr_Char((Hour>>4)+0x30);
  Wr_Char((Hour & 0x0F)+0x30);
  Wr_Char(':');
  Wr_Char((Min>>4)+0x30);
  Wr_Char((Min & 0x0F)+0x30);
  Wr_Char(':');
  Wr_Char((Sec>>4)+0x30);
  Wr_Char((Sec & 0x0F)+0x30);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
}
void GetDate(unsigned char X,Y)
{unsigned char i,Key;
 Cursor_On();
 Goto_XY(X,Y);
 Wr_Char(' ');Wr_Char(' ');
 Wr_Char('/');
 Wr_Char(' ');Wr_Char(' ');
 Wr_Char('/');
 Wr_Char(' ');Wr_Char(' ');
 Goto_XY(X,Y);
 i = 0;
 Key = 0;
 while(Key != 11)
  {
   Cursor_On();
   while (Scan_Key() == 0xFF) { }
   Key = Scan_Key();
   Sound();
   if((Key < 10)&&(i<6))
    {   BufDisp[i] = Key;
        Wr_Char(Key+0x30);
        i++;
        if(i==2)
          {
              Wr_Char('/');
              S_Date = (BufDisp[0]<<4) + BufDisp[1];
              if(S_Date > 0x31){goto Retry;}
          }
        if(i==4)
          {
              Wr_Char('/');
              S_Mon = (BufDisp[2]<<4) + BufDisp[3];
              if(S_Mon > 0x12)
                {
                   Goto_XY(X,Y+3);
                              Wr_Char(' ');Wr_Char(' ');
                   Goto_XY(X,Y+3);
                   i = 2;
                }
          }
         if(i==6)
          {
              S_Year = (BufDisp[4]<<4) + BufDisp[5];
          }
    }
   if(Key == 10)
    { Retry:
       Goto_XY(X,Y);
       Wr_Char(' ');Wr_Char(' ');
       Wr_Char('/');
       Wr_Char(' ');Wr_Char(' ');
       Wr_Char('/');
       Wr_Char(' ');Wr_Char(' ');
       i=0;
       Goto_XY(X,Y);
    }
    Key_Off();
  }
 }
void  ShowDate(unsigned X,Y)
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
{
  Goto_XY(X,Y);
  Wr_Char((Date>>4)+0x30);
  Wr_Char((Date & 0x0F)+0x30);
  Wr_Char('/');
  Wr_Char((Mon>>4)+0x30);
  Wr_Char((Mon & 0x0F)+0x30);
  Wr_Char('/');
  Wr_Char((Year>>4)+0x30);
  Wr_Char((Year & 0x0F)+0x30);
}
```


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้