

สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

เครื่องสลับพริ้นเตอร์ไร้สาย 1:2

1:2 AUTOCROSS WIRELESS PRINTER

โดย

นางสาว กฤตพร	ชুমหิวิวโฉลก	36014009
นางสาว อูติพร	ตันชนะ	36014131
นางสาว สายใจ	จันทรแจ่ม	36014482

อาจารย์ที่ปรึกษา

ผศ.ดร. ปราโมทย์ วาดเขียน

ปริญญานิพนธ์ฉบับนี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาศาสตรบัณฑิต

สาขาวิชาวิศวกรรมโทรคมนาคม

คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2539

เลขหมู่.....

เลขทะเบียน.....27881

วัน, เดือน, ปี 2.6.ค.ย. 2540

เอกสารนี้เป็นทรัพย์สินของสถาบันฯ ใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ปริญญาโทปีการศึกษา 2539

ภาควิชาวิศวกรรมโทรคมนาคม

คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เรื่อง เครื่องสลับพรีนเตอร์ไร้สาย 1:2

1:2 AUTOCROSS WIRELESS PRINTER

ผู้จัดทำ

1. นางสาว กฤตพร ชุณหะวัณโลก 36014009
2. นางสาว สฐิติพร ตันธนะ 36014131
3. นางสาว สายใจ จันทร์แจ่ม 36014482

..... ปรีชิต กาทิโกน

อาจารย์ที่ปรึกษา

(ผศ.ดร. ปราโมทย์ วาดเขียน)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เครื่องสลับพริ้นเตอร์ไร้สาย 1:2
1:2 AUTOCROSS WIRELESS PRINTER

โดย

นางสาว กฤตพร ชุณหะวัณ 36014009

นางสาว สุทธิพร ตันธนะ 36014131

นางสาว สายใจ จันทร์แจ่ม 36014482

อาจารย์ที่ปรึกษา

ผศ.ดร. ปราโมทย์ วาดเขียน

บทคัดย่อ

โครงการนี้เป็นการออกแบบและสร้างเครื่องสลับพริ้นเตอร์ไร้สาย ซึ่งมีหลักการทำงานคือ คอมพิวเตอร์หลายเครื่องสามารถที่จะส่งข้อมูลมาพิมพ์ยังเครื่องพิมพ์ได้ โดยใช้คลื่นวิทยุความถี่สูงผสมกับ ข้อมูลแล้วส่งออกไปโดยผ่านอากาศเป็นตัวกลาง โดยที่เครื่องพิมพ์จะเป็นฝ่ายเรียกขอการใช้บริการจากเครื่อง คอมพิวเตอร์แต่ละเครื่อง จากนั้นเครื่องพิมพ์จะทำการแยกข้อมูลออกจากคลื่นวิทยุแล้วทำการแปลงสัญญาณ ข้อมูลจากอนุกรมเป็นสัญญาณแบบขนาน จากนั้นจึงพิมพ์ออกมาทางเครื่องพิมพ์ต่อไป

Abstract

This project is concerns an autocross wireless printer that can transfer the data from computers to a printer. The data was modulated with high frequency radio wave and transmit it. A printer will ask each computer to send data. The computer sends the data to this device then it demodulates the data from serial to parallel and pass it to printer.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ

บทที่	หน้า
1. บทนำ	1
2. ทฤษฎีและหลักการ	2
2.1 การเชื่อมต่อระหว่างคอมพิวเตอร์กับพรินเตอร์	2
2.2 โครงสร้างของไมโครคอนโทรลเลอร์ตระกูล MCS-51	5
2.3 การใช้งานพอร์ตขนาน	19
2.4 การอินเทอร์เฟซกับพรินเตอร์แบบขนานและขาการเชื่อมต่อแบบ เซ็นโทรนิกส์	24
2.5 การต่อแบบขนานแบบไอบีเอ็ม	29
2.6 ดิจิตอลโมดูลเซ็น	31
2.7 วงจรออสซิลเลเตอร์	34
2.8 หลักการส่งข้อมูลสลับสายอัตโนมัติ	38
3. วงจรการทำงานของพรินเตอร์ไร้สาย	39
3.1 ภาคส่ง	39
3.2 ภาครับ	58
3.3 ส่วนเชื่อมต่อกับคอมพิวเตอร์และพรินเตอร์	70
3.4 ภาคควบคุม	72
4. ผลการทดลอง	74
4.1 ภาคส่งและภาครับ	74
4.2 ส่วนเชื่อมต่อระหว่างเครื่องคอมพิวเตอร์และพรินเตอร์	78
4.3 ส่วนแสดงสัญญาณต่าง ๆ ของเครื่องสลับพรินเตอร์ไร้สาย	79
5. สรุปและวิเคราะห์ผลการดำเนินงาน	84

ภาคผนวก

เอกสารอ้างอิง

กิตติกรรมประกาศ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญญรูปภาพ

รูปที่	หน้า
2.1 การติดต่อระหว่างคอมพิวเตอร์และพรินเตอร์แบบขนาน	2
2.2 การติดต่อระหว่างคอมพิวเตอร์และพรินเตอร์แบบอนุกรม	3
2.3 แสดงสถานะขาของชิปไมโครคอนโทรลเลอร์ตระกูล MCS-51	5
2.4 แสดงวงจรสำหรับรีเซตชิปไมโครคอนโทรลเลอร์ MCS-51	7
2.5 แสดงโครงสร้างภายในของชิปไมโครคอนโทรลเลอร์ MCS-51	8
2.6 แสดงโครงสร้างหน่วยความจำทั้งหมดของ MCS-51	9
2.7 แสดงการใช้หน่วยจำสำหรับเก็บโปรแกรมภายนอกชิป	10
2.8 แสดงโครงสร้างและตำแหน่งของรีจิสเตอร์ใช้งานเฉพาะใน MCS-51	11
2.9 รีจิสเตอร์ใช้งานเฉพาะ PSW	12
2.10 รีจิสเตอร์ใช้งานเฉพาะ PCON	13
2.11 รีจิสเตอร์ใช้งานเฉพาะ IE	14
2.12 รีจิสเตอร์ใช้งานเฉพาะ TCON	15
2.13 รีจิสเตอร์ใช้งานเฉพาะ TMOD	16
2.14 รีจิสเตอร์ใช้งานเฉพาะ SCON	17
2.15 แผนผังโครงสร้างของไอซี 8255	19
2.16 แผนผังวงจรภายในและการจัดขาของไอซี 8255	20
2.17 ความหมายของบิตต่าง ๆ ในรหัสควบคุม	21
2.18 โครงสร้างตรวจสอบสัญญาณของพอร์ตอินพุตและเอาต์พุต	22
2.19 วงจรการต่อ 8255 ในโหมด 1	23
2.20 แสดงขาของคอนเนคเตอร์	25
2.21 แสดงช่วงเวลาของรูปคลื่นสัญญาณที่จะส่งข้อมูลอักขระไปยังพรินเตอร์แบบขนาน	28
2.22 รูปคลื่นของดิจิตอลโมดูเลชันที่ใช้ในการส่งข้อมูลไปนารี	32
2.23 วงจรฮอสซิลเลเตอร์ที่ใช้การป้อนกลับแบบบวก	34
2.24 วงจรฮอสซิลเลเตอร์แบบฮาร์ทเลย์	36
2.25 เฟสของแรงดันไฟฟ้าในการแท็ปคอปยล์	37
2.26 แสดงหลักการโพลลิงระหว่างเครื่องคอมพิวเตอร์กับพรินเตอร์	38
3.1 บล็อกไดอะแกรมของภาคส่ง	39
3.2 วงจรเครื่องส่ง	41
3.3 บล็อกไดอะแกรมของภาครับ	58
3.4 วงจรเครื่องรับ	61

3.5	แสดงการเชื่อมต่อแบบ DB-25 กับส่วนเชื่อมต่อ	70
3.6	แสดงการเชื่อมต่อแบบแอมป์อินอลกับส่วนเชื่อมต่อ	71
3.7	วงจรเครื่องควบคุม	73
4.1	ลักษณะของสัญญาณที่ผลิตออกมาจากเครื่องส่ง	74
4.2	สัญญาณอินพุทที่เครื่องส่ง และสัญญาณเอาต์พุทที่เครื่องรับ	75
4.3	รูปสัญญาณของเครื่องส่งเมื่อป้อนความถี่ 5 kHz	75
4.4	แสดงความถี่ของเครื่องส่งเมื่อป้อนความถี่ 5 kHz	76
4.5	รูปสัญญาณเมื่อป้อนความถี่ 2.5 kHz	76
4.6	รูปสัญญาณเมื่อป้อนความถี่ 5 kHz	77
4.7	แสดงสัญญาณระหว่างเครื่องคอมพิวเตอร์กับพริ้นเตอร์	78
4.8	แสดงสัญญาณโพลลิ่ง ' 1AA '	79
4.9	แสดงสัญญาณโพลลิ่ง ' 1 AB '	79
4.10	สัญญาณสโตรป (strobe) ที่ส่งให้พริ้นเตอร์	80
4.11	สัญญาณบีซี (busy) จากพริ้นเตอร์แสดงสถานะไม่ว่าง	80
4.12	สัญญาณบีซี (busy) จากพริ้นเตอร์แสดงสถานะว่าง	81
4.13	สัญญาณแอกโนเลจ (acknowledge) จากพริ้นเตอร์	81
4.14	สัญญาณจากส่วนของคอมพิวเตอร์เมื่อไม่มีข้อมูลจะพิมพ์	82
4.15	สัญญาณข้อมูลจากส่วนของคอมพิวเตอร์เมื่อมีข้อมูลจะพิมพ์	82
4.16	สัญญาณสโตรป (strobe) จากเครื่องคอมพิวเตอร์	83

สารบัญตาราง

ตารางที่	หน้า
2.1 หน้าที่ของสัญญาณต่าง ๆ ของพอร์ท C เมื่อทำงานในโหมด 1	23
2.2 รายละเอียดการเชื่อมต่อขาแบบเซ็นโทรนิคส์	25
2.3 แสดงขาพอร์ทแบบขนานของไอบีเอ็ม	29
2.4 แสดงการปรับต่อขาของ DB-25 และ แอมฟินอล	30



บทที่ 1

บทนำ

ในปัจจุบันคอมพิวเตอร์ได้เข้ามามีบทบาทเป็นอย่างมากในชีวิตของคนเรา ไม่ว่าจะเป็นทางด้านการศึกษา การวิจัย การสื่อสาร การคมนาคม ข้อมูลที่คอมพิวเตอร์แสดงแก่ผู้ใช้มักจะอยู่ในรูปของการแสดงผลทางจอภาพ และการแสดงผลทางเครื่องพิมพ์เป็นหลัก โดยเฉพาะในกรณีที่เราต้องการข้อมูลที่เป็นรายงานทางอักษรเราจำเป็นต้องอย่างมากที่จะให้คอมพิวเตอร์แสดงผลออกทางเครื่องพิมพ์ ในบางครั้งพบว่าภายในห้องหนึ่งมีคอมพิวเตอร์หลายเครื่องแต่มีเครื่องพริ้นเตอร์เพียงเครื่องเดียว ในกรณีเช่นนี้นอกจากจะใช้วิธีถอดสายนำสัญญาณออกจากคอมพิวเตอร์ตัวเดิมแล้วนำไปติดกับคอมพิวเตอร์ตัวใหม่ที่มีเครื่องพิมพ์อยู่แล้ว ซึ่งจะเป็นการเสียเวลาทำให้สิ้นเปลือง ดังนั้นจึงได้คิดค้นเครื่องสลับสายพริ้นเตอร์นี้ขึ้นมา

เครื่องสลับพริ้นเตอร์ไร้สาย (AUTOCROSS WIRELESS PRINTER) เป็นเครื่องที่ทำหน้าที่ส่งข้อมูลระหว่างคอมพิวเตอร์ (COMPUTER) กับเครื่องพิมพ์ (PRINTER) สามารถส่งข้อมูลแบบไร้สายออกทางอากาศโดยไม่ต้องต่อสายให้ยุ่งยาก นอกจากนี้ยังสามารถเพิ่มประสิทธิภาพได้โดยสามารถส่งข้อมูลจากคอมพิวเตอร์หลายๆเครื่องโดยใช้เครื่องพิมพ์เครื่องเดียวกัน เหมาะสำหรับการใช้ในสำนักงานที่ต้องใช้คอมพิวเตอร์หลายเครื่อง แต่มีเครื่องพิมพ์จำนวนจำกัด ทำให้การพิมพ์สะดวกรวดเร็วยิ่งขึ้น

เครื่องสลับพริ้นเตอร์ไร้สายนี้ ยังมีความสามารถยิ่งกว่านั้นคือ สามารถพักข้อมูลที่ส่งมาจากคอมพิวเตอร์เนื่องจากเครื่องพิมพ์เป็นอุปกรณ์ที่ทำงานช้ากว่าคอมพิวเตอร์มาก ในการส่งพิมพ์แต่ละครั้ง ถ้าข้อมูลที่จะพิมพ์ยังถูกส่งไปไม่หมด คอมพิวเตอร์จะต้องส่งข้อมูลให้หมดเสียก่อนจึงจะสามารถไปทำงานอื่นได้ ซึ่งถ้าหากว่าข้อมูลมีขนาดใหญ่ คอมพิวเตอร์จะต้องเสียเวลาในการรอการส่งข้อมูลนาน ดังนั้นเครื่องสลับพริ้นเตอร์ไร้สายนี้จะช่วยแก้ปัญหาได้ โดยจะทำหน้าที่รับข้อมูลมาจากคอมพิวเตอร์ก่อน โดยที่เครื่องสลับพริ้นเตอร์ไร้สายจะสามารถรับข้อมูลได้รวดเร็วเท่ากับความเร็วที่คอมพิวเตอร์ส่งมาให้ ทำให้คอมพิวเตอร์ไม่ต้องเสียเวลาในการรอส่งข้อมูล เมื่อส่งข้อมูลครบเรียบร้อยแล้วคอมพิวเตอร์สามารถไปทำงานอื่นต่อไปได้อีกแล้วเครื่องสลับพริ้นเตอร์ไร้สายก็จะทำการส่งข้อมูลต่อไปยังเครื่องพิมพ์ในอัตราเร็วที่เหมาะสมกับเครื่องพิมพ์นั้นๆ

บทที่ 2

ทฤษฎีและหลักการ

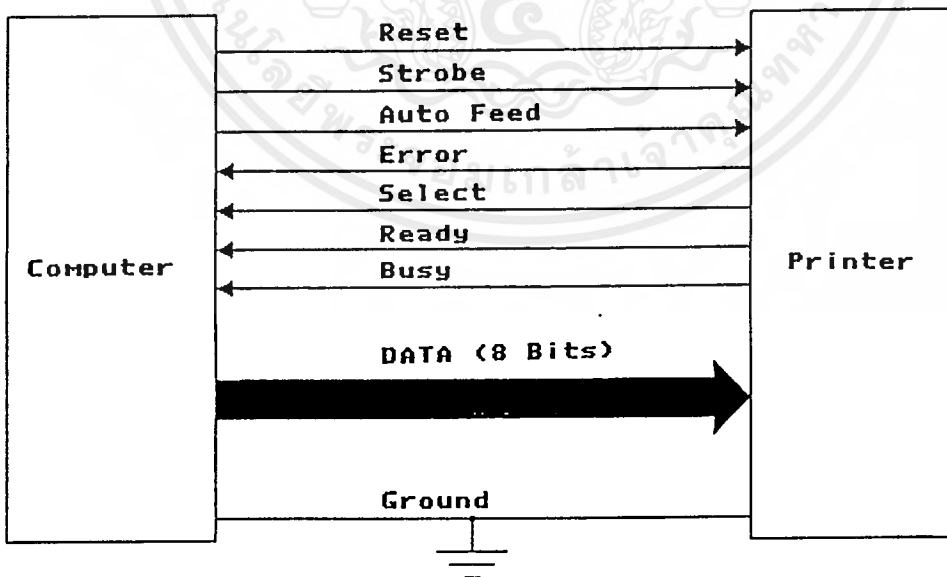
2.1 การเชื่อมต่อระหว่างคอมพิวเตอร์กับพรินเตอร์

การสื่อสารข้อมูลกับพรินเตอร์นั้น คอมพิวเตอร์สามารถส่งข้อมูล 3 ชนิด ไปที่พรินเตอร์ คือ ข้อมูลตัวอักษร รหัสควบคุม และ ข้อมูลกราฟฟิก ข้อมูลอักขระแสดงได้ในรูปของตัวอักษร ตัวเลข เครื่องหมายวรรคตอน สัญลักษณ์ต่าง ๆ รหัสควบคุมจะถูกใช้เพื่อส่งคำสั่งไปยังพรินเตอร์ รหัสเหล่านี้จะกำหนดโหมดการทำงาน เช่น รูปแบบตัวพิมพ์ (Font Style) ขนาดตัวอักษร หรือ การควบคุมที่สามารถกระทำได้โดยตรง เช่น การเลื่อนกระดาษที่ละบรรทัด หรือ ทีละหน้า การใช้รหัสควบคุมแยกแยะมีความจำเป็นต่อการทำงานขณะที่พิมพ์เอกสาร ซึ่งรหัสควบคุมอื่น ๆ จะทำให้พรินเตอร์ทำงานในโหมดกราฟฟิกต่าง ๆ ได้

ลักษณะของการสื่อสารข้อมูล

ข้อมูลที่อยู่ในรูปแบบไบนารี (Binary) ที่จะใช้ถ่ายทอดระหว่างคอมพิวเตอร์ และ พรินเตอร์ ต้องใช้ตัวกลางในการสื่อสารคือ สายเคเบิล การผลิตและออกแบบคุณสมบัติของสายเคเบิลจะต้องขึ้นอยู่กับวิธีที่เราใช้ติดต่อ ซึ่งมักนิยมใช้อยู่ 2 แบบ คือ แบบอนุกรม และ แบบขนาน

การติดต่อแบบขนาน (Parallel) เป็นวิธีที่ง่ายที่สุด เนื่องจาก เราส่งข้อมูลตรง ๆ เลย ดังแสดงในรูปที่ 2.1



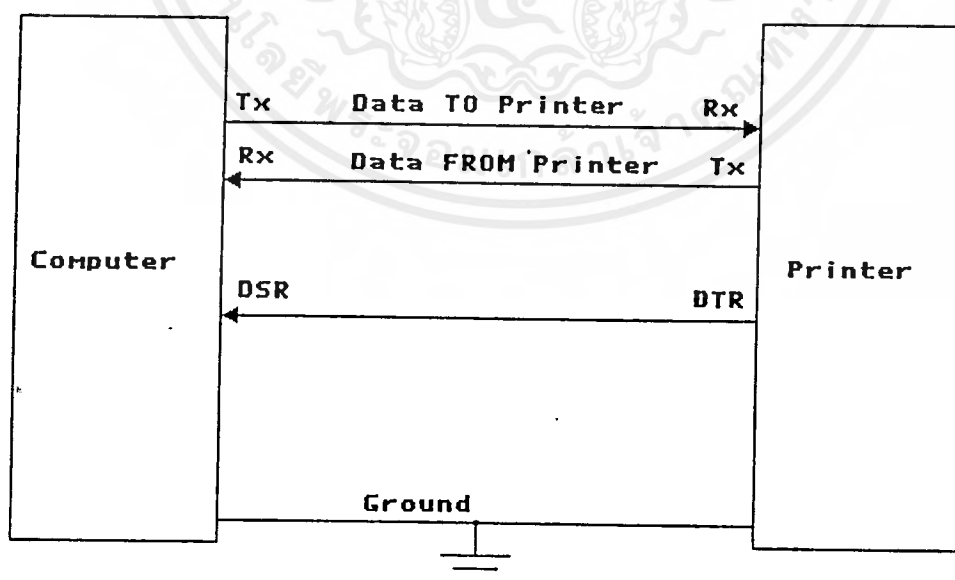
รูปที่ 2.1 แสดงการติดต่อระหว่างคอมพิวเตอร์และพรินเตอร์แบบขนาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สังเกตได้ว่า ข้อมูล 8 บิต ส่งแยก บิต D_0 ถึง บิต D_7 พร้อม ๆ กัน แต่ถ้าจะให้ส่งได้อย่างถูกต้อง คอมพิวเตอร์ และ พริ้นเตอร์ต้องซิงโครนัส (Synchronous) กัน คือ พริ้นเตอร์จะรับข้อมูลเมื่อมันขอ หรือ พริ้นเตอร์สั่งให้คอมพิวเตอร์รอจนกว่าพริ้นเตอร์จะพร้อมรับข้อมูล ดังนั้นการซิงโครนัสของการสื่อสารข้อมูล แบบขนานจำเป็นต้องมีสายควบคุมจำนวนมากสำหรับ สัญญาณพริ้นเตอร์ และ สัญญาณคอมพิวเตอร์ การทำงานอย่างสัมพันธ์กันดังกล่าวจะเรียกว่า การแฮนด์เชค (Handshaking)

การส่งแบบขนานมีความเร็วพอใช้ได้ โดยพริ้นเตอร์จะรับข้อมูลเร็วเท่า ๆ กับที่คอมพิวเตอร์ส่งไปซึ่ง มีความเร็วประมาณ 1,000 cps. เมื่อข้อมูลหนึ่งตัวอักษรจะมี 8 บิต ความเร็วจะมากกว่า 8,000 บิตต่อวินาที บัส (Bus) ข้อมูลภายในพริ้นเตอร์มี 8 บิต จะสามารถรับข้อมูลในบัฟเฟอร์ (Buffer) ได้โดยเพิ่มวงจรเพียงเล็กน้อย การส่งข้อมูลแบบขนานเป็นการติดต่ออย่างง่าย แต่มีข้อเสียจุดใหญ่คือ ใช้สายเคเบิลมาก แต่จะมีความเร็วในการส่งสูง และมีสัญญาณรบกวนทางอิเล็กทรอนิกส์ ซึ่งทำให้เกิดความผิดพลาดของข้อมูลได้

การติดต่อแบบอนุกรม สามารถต่อได้ง่าย เพราะมีจำนวนสายน้อยลง ดังรูปที่ 2.2 แต่การทำงานจะยุ่งยากกว่า เราจะพบว่าจากรูปที่ 2.2 มีสายสองเส้นที่ใช้ในการส่งข่าวสาร เส้นหนึ่งสำหรับส่งข้อมูลจากคอมพิวเตอร์ไปยังพริ้นเตอร์ และ อีกหนึ่งเส้นสำหรับส่งข้อมูลจากพริ้นเตอร์ไปยังคอมพิวเตอร์ เพราะว่าข้อมูลสามารถส่งไปมาได้ทั้งสองทิศทาง จึงเรียกการ ติดต่อสื่อสารแบบนี้ว่า การสื่อสารข้อมูลแบบสองทาง (Bidirectional Data Link) ถ้ามีสายเส้นเดียวจะสามารถส่งหรือรับข้อมูลได้เพียงอย่างเดียวอย่างใดอย่างหนึ่งในเวลาหนึ่งเท่านั้น ข้อมูลแบบอนุกรมต้องซิงโครนัสกันระหว่างคอมพิวเตอร์และพริ้นเตอร์ ดังนั้นเราต้องเพิ่มซิงโครไนซ์บิต ที่จุดเริ่มต้น และ จุดสิ้นสุด ของแต่ละข้อมูลอักขระที่ส่ง และ ยังมีพาริตีบิต (Parity Bit) ที่ต้องรวมอยู่ด้วยสำหรับตรวจสอบบิตผิดพลาด



รูปที่ 2.2 แสดงการติดต่อระหว่างคอมพิวเตอร์และพริ้นเตอร์แบบอนุกรม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การทำกรแฮนด์เซคแบบอนุกรมนั้น แบ่งเป็นทั้ง ซอฟต์แวร์ และ ฮาร์ดแวร์ โดยซอฟต์แวร์ใช้ในการติดต่ออนุกรมแบบสองทิศทาง เช่น ใช้ส่งพรินเตอร์ให้ส่งรหัสควบคุมไปยังคอมพิวเตอร์ เช่น รหัสสำหรับการแฮนด์เซค “XON” และ “XOFF” ในส่วนของฮาร์ดแวร์ของการแฮนด์เซคนั้นไม่ได้ใช้ในการส่งข้อมูลจากพรินเตอร์ไปยังคอมพิวเตอร์ สัญญาณแฮนด์เซคที่แจ้งให้คอมพิวเตอร์ที่เพิ่มขึ้นคือสัญญาณพรินเตอร์ไม่ว่าง ดังนั้นการติดต่อจึงจะมีมากกว่าหนึ่งสายแฮนด์เซค โดยทั่วไปเรามักจะให้มีการแฮนด์เซคระหว่างพรินเตอร์ค่อนข้างมาก แม้ว่าการทำงานของกรสื่อสารข้อมูลแบบอนุกรมจะยุ่งยากแต่การติดต่อแบบอนุกรมก็เป็นที่นิยมกันมาก เนื่องจากความสามารถในการติดต่อแบบสองทิศทาง และ สามารถใช้งานในระยะไกล



- **ขาพอร์ต 1 (ขา 1-8)** มี 8 ขา ใช้เป็นขาสำหรับพอร์ต 1 ขนาด 8 บิต (P1.0-P1.7) พอร์ตนี้สามารถใช้งานเป็นอินพุตเอาต์พุตพอร์ตทั่วไปได้ โดยหากใช้งานเป็นอินพุตพอร์ตต้องโหลดค่า 1 ไปยังแต่ละบิตของพอร์ตนี้ เพื่อบังคับให้ขาอยู่ในสถานะถูกปล่อยลอย (มีสถานะ high-impedance) โดยมีวงจรถูกอ์ภายใน

- **ขาพอร์ต 2 (ขา 21-28)** มี 8 ขา ใช้เป็นขาสำหรับพอร์ต 2 ขนาด 8 บิต (P2.0-P2.7) พอร์ตนี้สามารถใช้งานเป็นอินพุตเอาต์พุตพอร์ตทั่วไปได้ โดยหากใช้งานเป็นอินพุตพอร์ตต้องโหลดค่า 1 ไปยังแต่ละบิตของพอร์ตนี้ เพื่อบังคับให้ขาอยู่ในสถานะถูกปล่อยลอย (มีสถานะ high-impedance) โดยมีวงจรถูกอ์ภายใน นอกจากนี้ใช้งานเป็นอินพุตเอาต์พุตพอร์ตทั่วไปแล้วพอร์ต 2 ยังใช้ในการติดต่อรวมและแรมภายนอกด้วย โดยใช้สำหรับส่งค่าแอดเดรสไบต์สูง (A8-A15)

- **ขาพอร์ต 3 (ขา 10-17)** มี 8 ขา ใช้เป็นขาสำหรับพอร์ต 3 ขนาด 8 บิต (P3.0-P3.7) พอร์ตนี้สามารถใช้งานเป็นอินพุตเอาต์พุตพอร์ตทั่วไปได้ โดยหากใช้งานเป็นอินพุตพอร์ตต้องโหลดค่า 1 ไปยังแต่ละบิตของพอร์ตนี้ เพื่อบังคับให้ขาอยู่ในสถานะถูกปล่อยลอย (มีสถานะ high-impedance) โดยมีวงจรถูกอ์ภายใน นอกจากนี้ยังใช้งานในหน้าที่พิเศษต่างๆดังนี้

ขา P3.0 ใช้รับข้อมูลจากภายนอกแบบอนุกรม

ขา P3.1 ใช้ส่งข้อมูลออกไปภายนอกแบบอนุกรม

ขา P3.2 ใช้เป็นอินพุตเพื่อรับสัญญาณอินเตอร์รัปต์ชนิดที่ 0

ขา P3.3 ใช้เป็นอินพุตเพื่อรับสัญญาณอินเตอร์รัปต์ชนิดที่ 1

ขา P3.4 ใช้เป็นสัญญาณอินพุตให้เคาต์เตอร์ของไทม์เมอร์ 0

ขา P3.5 ใช้เป็นสัญญาณอินพุตให้เคาต์เตอร์ของไทม์เมอร์ 1

ขา P3.6 ใช้เป็นสัญญาณควบคุมการเขียนข้อมูลสำหรับเก็บข้อมูลภายนอกชิป

ขา P3.7 ใช้เป็นสัญญาณควบคุมการอ่านข้อมูลสำหรับเก็บข้อมูลภายนอกชิป

การใช้งานพอร์ต 3 ในหน้าที่พิเศษนี้จะต้องโหลดค่า 1 ไปยังแต่ละบิตที่ต้องการใช้ก่อนทุกครั้ง

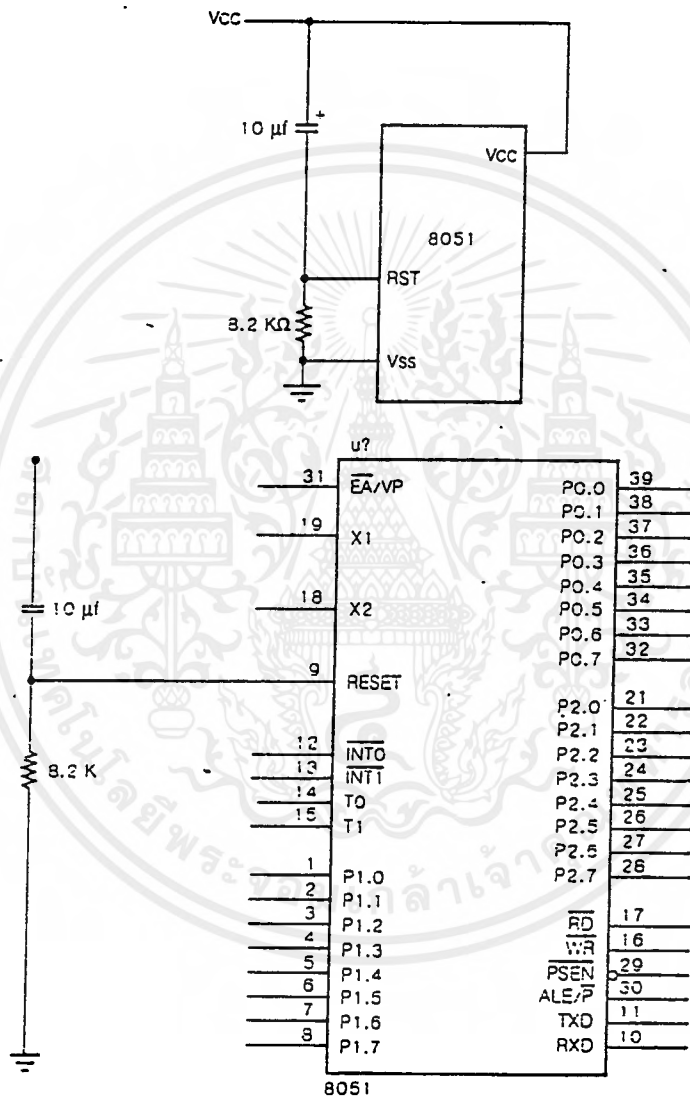
- **ขา RST (ขา 9)** ใช้สำหรับการรีเซตวงจรถูกอ์ภายในชิปเพื่อเริ่มต้นทำงานใหม่ โดยมีการต่อวงจรถูกอ์ที่ 2.4

- **ขา ALE / PROG (ขา 30)** ใช้สำหรับส่งสัญญาณออกไปภายนอกเพื่อควบคุมการแลตช์ค่าแอดเดรสไบต์ต่ำ (Address Latch Enable) จากพอร์ต 0 ในระหว่างการติดต่อรวมหรือ แรมภายนอก

- **ขา PSEN (ขา 29)** ใช้ส่งสัญญาณลตรบเพื่ออ่านคำสั่งจากโปรแกรมที่เก็บไว้ในรอมภายนอก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

-ขา EA / V_{pp} (ขา 31) เป็นขาสำหรับใช้เลือกให้ MCS-51 ทำงานจากโปรแกรมที่อยู่ภายในหรือภายนอกชิป โดยขาขานี้มีสถานะเป็น 0 ให้ใช้โปรแกรมจากภายนอกรวมภายใน ถ้าขาขานี้มีสถานะเป็น 1 หมายถึงบังคับให้ MCS-51 ใช้โปรแกรมจากภายในชิป ส่วนใน MCS-51 ที่ไม่มีรวมภายในให้ต่อขานี้ลงกราวด์เสมอ



รูปที่ 2.4 แสดงวงจรสำหรับรีเซ็ตชิปไมโครคอนโทรลเลอร์ MCS-51 เมื่อเริ่มจ่ายพลังงานโดยอัตโนมัติ

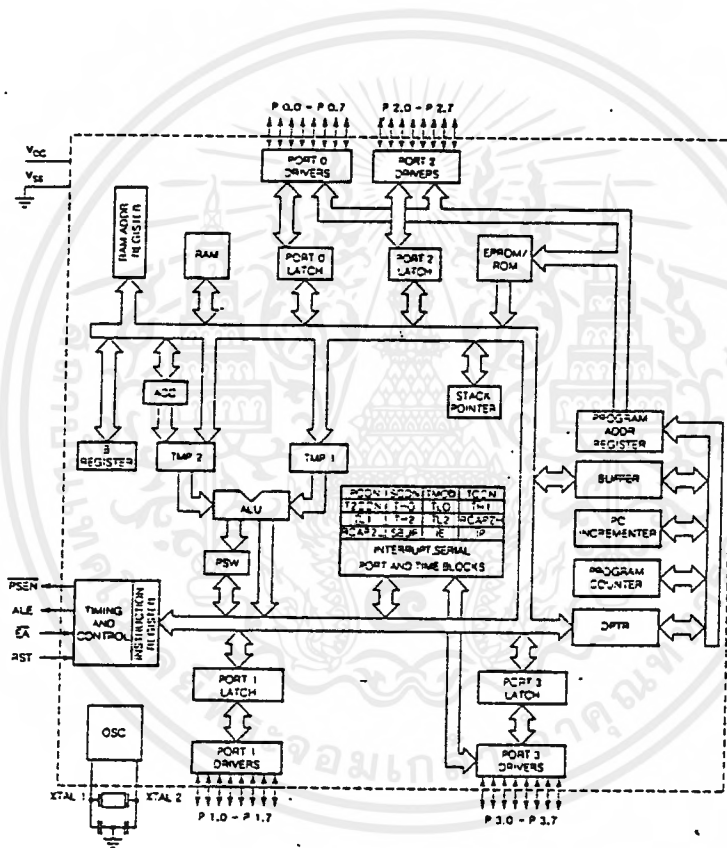
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

-ขา XTAL 1 (ขา 19) ใช้ต่อคริสตอลภายนอก โดยเป็นอินพุตเข้าสู่วงจรรอสซิลเลเตอร์

-ขา XTAL 2 (ขา 18) ใช้ต่อคริสตอลภายนอก โดยเป็นเอาต์พุตออกจากวงจรรอสซิลเลเตอร์

2.2.2 โครงสร้างภายในของ MCS-51

โครงสร้างภายในของชิปไมโครคอนโทรลเลอร์ตระกูล MCS-51 แสดงดังในรูปที่ 2.5

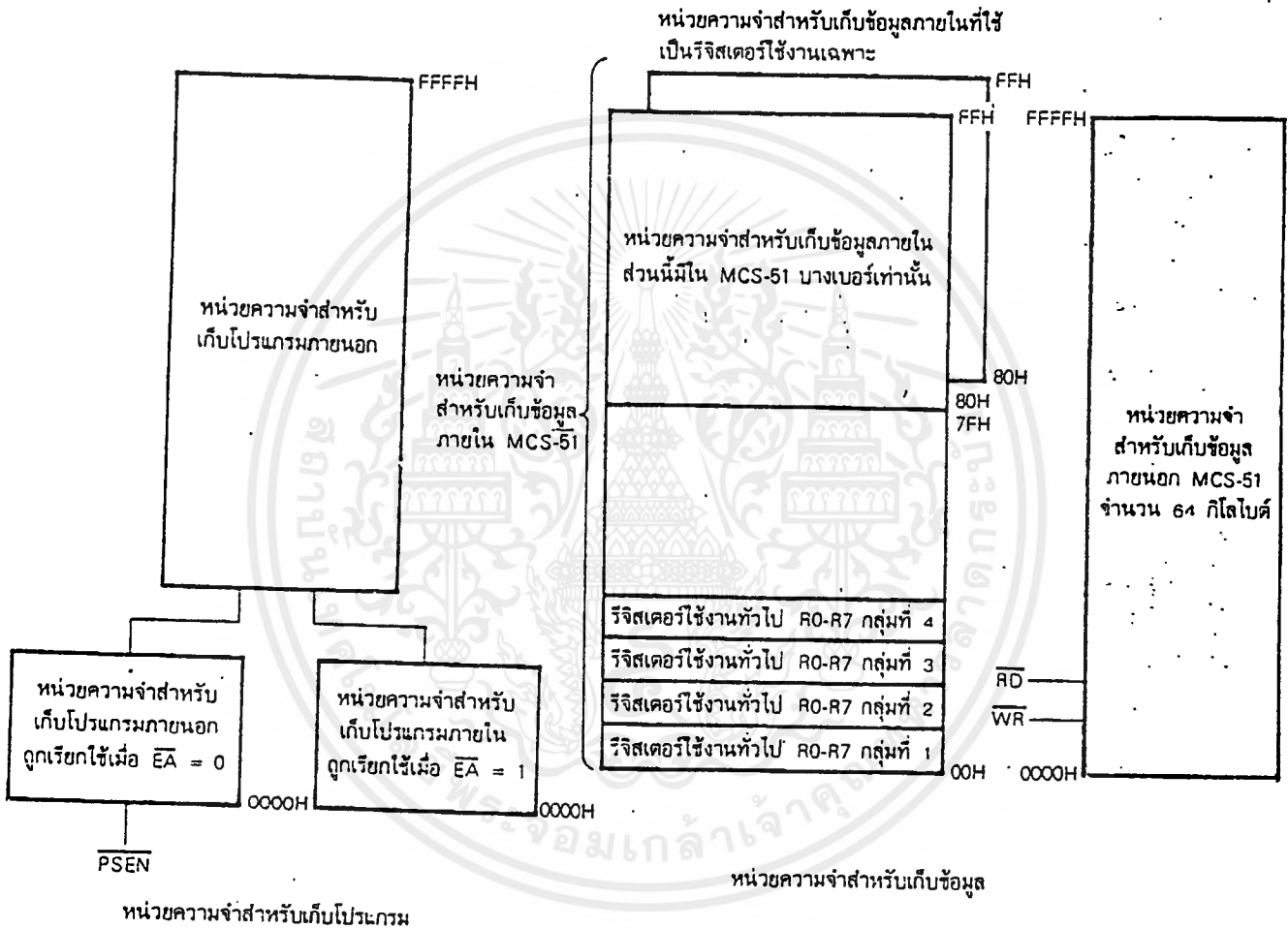


รูปที่ 2.5 แสดงโครงสร้างภายในของชิปไมโครคอนโทรลเลอร์ MCS-51

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.2.3 โครงสร้างหน่วยความจำภายในของ MCS-51

โครงสร้างหน่วยความจำทั้งหมดของ MCS-51 แสดงดังในรูปที่ 2.6



รูปที่ 2.6 แสดงโครงสร้างหน่วยความจำทั้งหมดของ MCS-51

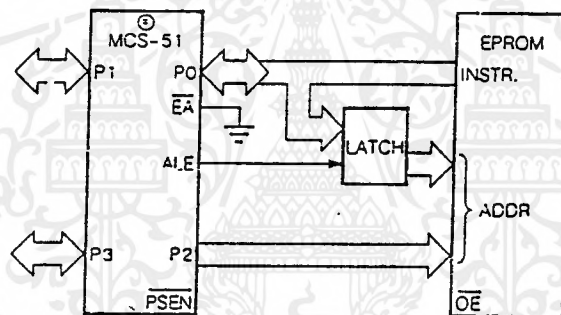
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ไมโครคอนโทรลเลอร์ตระกูล MCS-51 ทุกเบอร์จะแบ่งหน่วยความจำออกเป็น 2 ส่วน คือ

- หน่วยความจำสำหรับเก็บโปรแกรม (program memory)

หน่วยความจำสำหรับเก็บโปรแกรมควบคุมการทำงานของชิป MCS-51 บางเบอร์จะมีหน่วยความจำนี้อยู่ภายในชิป (internal program memory) ซึ่งมีขนาดได้ตั้งแต่ 0 , 4 , 8 , 16 กิโลไบต์ แต่บางเบอร์จะไม่มี ทำให้ต้องเก็บโปรแกรมไว้ในหน่วยความจำภายนอก (external program memory) ทั้งหมด

สัญญาณควบคุมการเฟลซ์คำสั่งกับโปรแกรมที่เก็บไว้ในหน่วยความจำสำหรับเก็บข้อมูลภายนอกชิป (read strobe) คือสัญญาณ PSEN (Program Strobe Enable) จากขา 29 ซึ่งนำไปใช้ต่อกับขา RD ของหน่วยความจำสำหรับเก็บโปรแกรมภายนอกแต่สัญญาณ PSEN จะไม่ถูกใช้งาน เมื่อ MCS-51 ทำงานจากโปรแกรมซึ่งอยู่ในหน่วยความจำสำหรับเก็บโปรแกรมที่อยู่ภายในชิป การใช้โปรแกรมจากหน่วยความจำภายนอกชิปแสดงดังในรูปที่ 2.7



รูปที่ 2.7 แสดงการใช้หน่วยความจำสำหรับเก็บโปรแกรมภายนอกชิป

-หน่วยความจำสำหรับเก็บข้อมูล (data memory)

หน่วยความจำสำหรับเก็บข้อมูล ซึ่งใช้สำหรับเก็บข้อมูลระหว่างการทำงาน MCS-51 ทุกเบอร์จะมีหน่วยความจำนี้อยู่ภายในชิปจำนวนหนึ่ง แต่จะมากหรือน้อยเท่าไรขึ้นอยู่กับเบอร์ของชิป แผนภาพแสดงหน่วยความจำสำหรับเก็บข้อมูลภายในชิปแสดงดังในรูปที่ 2.6 โดยแบ่งออกเป็น 2 ส่วน คือ หน่วยความจำสำหรับเก็บข้อมูลภายในชิป และหน่วยความจำสำหรับเก็บข้อมูลภายนอกชิป หน่วยความจำสำหรับเก็บข้อมูลภายในชิปยังแบ่งออกได้เป็น 2 ส่วนย่อย คือ

- ส่วนที่ใช้เก็บข้อมูลทั่วไป (internal ram) อยู่ภายในชิปบริเวณ 128 ไบต์ แรก
- ส่วนที่ใช้เป็นรีจิสเตอร์ใช้งานเฉพาะ (special function register:SFR) อยู่ภายในชิปบริเวณ 128 ไบต์ หลังซึ่งเป็นส่วนที่มีเพิ่มขึ้นมาใน MCS-51 บางเบอร์ดังแสดงในรูปที่ 2.8

เข้าถึงได้ในระดับบิต

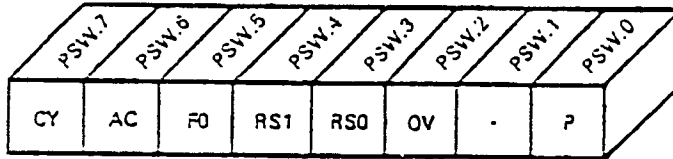
8 ไบต์

F8									FF
F0	8								F7
E8									EF
E0	ACC								E7
D8									DF
D0	PSW								D7
C8	T2CON		RCAP2L	RCAP2H	TL2	TH2			CF
C0									C7
B8	IP								BF
B0	P3								B7
A8	IE								AF
A0	P2								A7
98	SCON	SBUF							9F
90	P1								97
88	TCON	TMOD	TLO	TL1	TH0	TH1			8F
80	P0	SP	DPL	DPH				PCON	87

รูปที่ 2.8 แสดงโครงสร้างและตำแหน่งของรีจิสเตอร์ใช้งานเฉพาะใน MCS-51

2.2.4 รีจิสเตอร์ที่ใช้ควบคุม MCS-51

- 1). รีจิสเตอร์ที่ใช้งานเฉพาะ PSW (Program Status Word)
เข้าถึงข้อมูลได้ในระดับบิตดังแสดงในรูปที่ 2.9

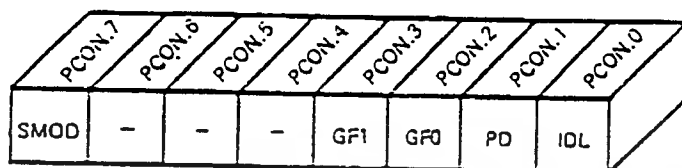


รูปที่ 2.9 รีจิสเตอร์ใช้งานเฉพาะ PSW

- PSW.7 CY carry flag
- PSW.6 AC auxiliary carry flag
- PSW.5 FO flag 0 เป็นบิตบอกสถานะที่ผู้ใช้สามารถกำหนดการใช้งานเองได้
- PSW.4 RS1 ใช้เลือกกลุ่มรีจิสเตอร์ R0-R7
- PSW.3 RS0 ใช้เลือกกลุ่มรีจิสเตอร์ R0-R7
- 0 0 เลือกรีจิสเตอร์ใช้งานทั่วไปกลุ่มที่ 0
- 0 1 เลือกรีจิสเตอร์ใช้งานทั่วไปกลุ่มที่ 1
- 1 0 เลือกรีจิสเตอร์ใช้งานทั่วไปกลุ่มที่ 2
- 1 1 เลือกรีจิสเตอร์ใช้งานทั่วไปกลุ่มที่ 3
- PSW.2 OV บิตแสดงการเกิด over flow
- PSW.1 - บิตที่ผู้ใช้กำหนดการใช้งานเองได้
- PSW.0 P parity flag ถูกเซตหรือเคลียร์โดยวงจรภายใน MCS-51 ในแต่ละไซเคิลของคำสั่งแต่ละคำสั่ง ใช้เป็นตัวบอกให้ทราบว่าในรีจิสเตอร์ ACC มีข้อมูลที่เป็น 1 อยู่เป็นจำนวนคู่หรือคี่

2) รีจิสเตอร์ใช้งานเฉพาะ PCON (Power Control Register)

ไม่สามารถเข้าถึงข้อมูลในระดับบิตได้ ดังแผนภาพที่แสดงในรูปที่ 2.10



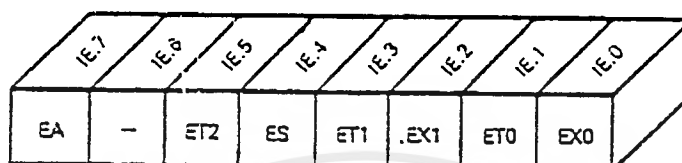
รูปที่ 2.10 รีจิสเตอร์ใช้งานเฉพาะ PCON

- PCON.7 SMOD เมื่อใช้ไทม์เมอร์ 1 เป็นตัวกำหนด baud rate หากบิตนี้มีค่าเป็น 1 ในการใช้งานพอร์ตสื่อสารอนุกรมโหมด 1,2 และ 3 ค่า baud rate จะเพิ่มขึ้นเป็น 2 เท่า
- PCON.6 - ไม่ถูกกำหนดการใช้งาน (สำรองไว้ใช้ใน MCS-51 เบอร์ใหม่ๆในอนาคต)
- PCON.5 - ไม่ถูกกำหนดการใช้งาน (สำรองไว้ใช้ใน MCS-51 เบอร์ใหม่ๆในอนาคต)
- PCON.4 - ไม่ถูกกำหนดการใช้งาน (สำรองไว้ใช้ใน MCS-51 เบอร์ใหม่ๆในอนาคต)
- PCON.3 GF1 บิตที่ผู้ใช้สามารถกำหนดการใช้งานได้เอง
- PCON.3 P0 บิตควบคุมการให้พลังงานแบบ Power Down Mode
- 1 : MCS-51 จะอยู่ในสถานะ Power Down Mode (หยุดทำงานจนกว่าจะมีการรีเซ็ต) ใช้ได้เฉพาะ MCS-51 ที่ผลิตโดยใช้เทคโนโลยี CHMOS เท่านั้น
- 0 : MCS-51 จะอยู่ในสถานะการทำงานปกติ
- PCON.0 IDL บิตควบคุมการทำงานแบบพลังงานต่ำแบบ idle mode
- 1 : MCS-51 อยู่ในสถานะ idle mode คือหยุดการทำงานชั่วคราวจนกว่าจะมีสัญญาณอินเตอร์รัปต์ ในระหว่างที่รอสัญญาณอินเตอร์รัปต์ MCS-51 จะใช้พลังงานน้อยมาก
- 0 : MCS-51 อยู่ในสถานะการทำงานปกติ (บิต PD ต้องเป็น 1)

2.2.5 รีจิสเตอร์สำหรับใช้ควบคุมการทำงานเกี่ยวกับอินเทอร์รัปต์

1) รีจิสเตอร์ใช้งานเฉพาะ IE (Interrupt Enable-Register)

เข้าถึงข้อมูลได้ในระดับบิตดังแสดงในรูปที่ 2.11



รูปที่ 2.11 รีจิสเตอร์ใช้งานเฉพาะ IE

IE.7	EA	ใช้ควบคุมตอบสนองต่อสัญญาณอินเทอร์รัปต์ทั้งหมด 0 : MCS-51 จะไม่ตอบสนองต่อสัญญาณอินเทอร์รัปต์ใด ๆ ทั้งสิ้น 1 : อินเทอร์รัปต์แต่ละชนิดถูกควบคุมการตอบสนองจากบิตในรีจิสเตอร์นี้
IE.6	-	ไม่ถูกกำหนดการใช้งาน (สำรองไว้ใช้ใน MCS-51 เบอร์ใหม่ ๆ ในอนาคต)
IE.5	ET2	ควบคุมการตอบสนองต่อสัญญาณอินเทอร์รัปต์ของไทม์เมอร์ 2 เมื่อเกิดโอเวอร์โฟลหรือเกิดอินเทอร์รัปต์ของไทม์เมอร์ 2 (มีเฉพาะ MCS-51 บางเบอร์ที่มีไทม์เมอร์ 2 เช่น 8052)
IE.3	ET1	กำหนดลำดับความสำคัญในการตอบสนองต่อสัญญาณอินเทอร์รัปต์ของไทม์เมอร์ 1
IE.2	EX1	กำหนดลำดับความสำคัญในการตอบสนองต่อสัญญาณอินเทอร์รัปต์ภายนอกชนิด 1
IE.1	ET0	กำหนดลำดับความสำคัญในการตอบสนองต่อสัญญาณอินเทอร์รัปต์ของไทม์เมอร์ 0
IE.0	EX0	กำหนดลำดับความสำคัญในการตอบสนองต่อสัญญาณอินเทอร์รัปต์ภายนอกชนิด 1

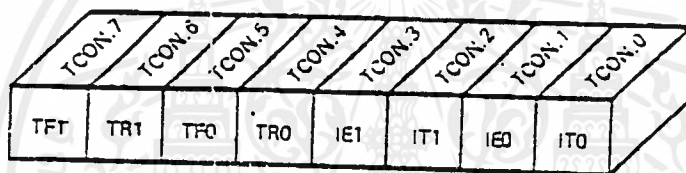
การให้บิตกำหนดความสำคัญของอินเทอร์รัปต์เป็น 0 หมายถึงให้อินเทอร์รัปต์ชนิดนั้นมีความสำคัญต่ำ ส่วนการให้บิตกำหนดความสำคัญของการอินเทอร์รัปต์เป็น 1 หมายถึงให้อินเทอร์รัปต์ชนิดนั้นมีความสำคัญสูง

รีจิสเตอร์ที่ควบคุมการทำงานของไทม์เมอร์/เคาน์เตอร์ รีจิสเตอร์ที่ควบคุมการทำงานของไทม์เมอร์หรือเคาน์เตอร์ใน MCS-51 ประกอบด้วยรีจิสเตอร์ใช้งานเฉพาะ TCON, TMOD และ T2CON (มีใน MCS-51 ที่มีไทม์เมอร์ 2 เท่านั้น)

2.2.6 รีจิสเตอร์ที่ควบคุมการทำงานของไทม์เมอร์ / เคาน์เตอร์

1) รีจิสเตอร์ใช้งานเฉพาะ TCON (Timer / Counter Control Register)

เข้าถึงข้อมูลได้ในระดับบิต ดังแสดงในรูปที่ 2.12



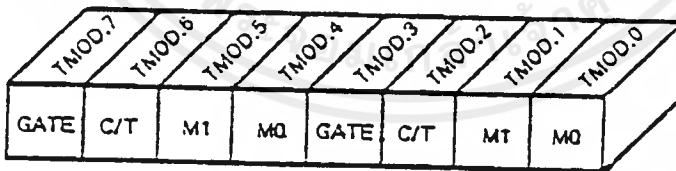
รูปที่ 2.12 รีจิสเตอร์ใช้งานเฉพาะ TCON

TCON.7	TF1	บิตแสดงการเกิด โอเวอร์โฟล ของไทม์เมอร์ 1 จะถูกเซตเองเมื่อไทม์เมอร์ 1 เกิดโอเวอร์โฟล และจะถูกเคลียร์เมื่อซีพียูย้ายไปทำงานที่โปรแกรมบริการการอินเทอร์รัปต์
TCON.6	TR1	บิตควบคุมการนับของไทม์เมอร์ 1 (ควบคุมการทำงานจากโปรแกรม) 1 : ไทม์เมอร์ 1 เริ่มทำงานต่อ (นับต่อ) 0 : ไทม์เมอร์ 1 หยุดการทำงาน (หยุดนับสัญญาณนาฬิกาภายใน หรือ จำนวนพัลส์ภายนอกที่ขา INT1)
TCON.5	TF0	บิตแสดงการเกิดโอเวอร์โฟลของไทม์เมอร์ 0 จะถูกเซตเองเมื่อไทม์เมอร์ 0 เกิดโอเวอร์โฟล และจะถูกเคลียร์เองเมื่อซีพียูย้ายไปทำงานที่โปรแกรมบริการการอินเทอร์รัปต์
TCON.4	TRO	บิตควบคุมการนับของไทม์เมอร์ 0 (ควบคุมการทำงานจากโปรแกรม) 1 : ไทม์เมอร์ 0 เริ่มทำงานต่อ (นับต่อ)

- 0 : ไทม์เมอร์ 0 หยุดการทำงาน (หยุดนับสัญญาณนาฬิกาภายใน หรือ จำนวนพัลส์ภายนอกที่ขา INTO)
- TCON.3 IE1 บิตแสดงสถานะสัญญาณอินเตอร์รัปต์ภายนอกชนิดที่ 1 ซึ่งจะถูกเซตเองโดยฮาร์ดแวร์เมื่อมีสัญญาณอินเตอร์รัปต์ที่ได้จากการตรวจสอบการเปลี่ยนสถานะและจะถูกเคลียร์เองโดยคำสั่ง RETI ที่อยู่ในโปรแกรมส่วนบริการอินเตอร์รัปต์
- TCON.2 IT1 บิตเลือกประเภทการตรวจสอบสัญญาณอินเตอร์รัปต์ที่เกิดขึ้นที่ขา INT1 โดย
1 : ตรวจสอบการเปลี่ยนระดับสัญญาณจาก 1 เป็น 0 ที่ขา INT1
0 : ตรวจสอบระดับของสัญญาณที่ขา INT1
- TCON.1 IE0 บิตแสดงสถานะสัญญาณอินเตอร์รัปต์ภายนอกชนิดที่ 0 ซึ่งจะถูกเซตเองโดยฮาร์ดแวร์เมื่อมีสัญญาณอินเตอร์รัปต์ที่ได้จากการตรวจสอบการเปลี่ยนสถานะและจะถูกเคลียร์เองโดยคำสั่ง RETI ที่อยู่ในโปรแกรมส่วนบริการอินเตอร์รัปต์
- TCON.0 ITO บิตเลือกประเภทการตรวจสอบสัญญาณอินเตอร์รัปต์ที่เกิดขึ้นที่ขา INT1 เหมือนบิต IT1

2) รีจิสเตอร์ใช้งานเฉพาะ TMOD (Timer/Counter Mode Control Register)

ไม่สามารถเข้าถึงข้อมูลในระดับบิตได้ ดังแสดงในรูปที่ 2.13



รูปที่ 2.13 รีจิสเตอร์ใช้งานเฉพาะ TMOD

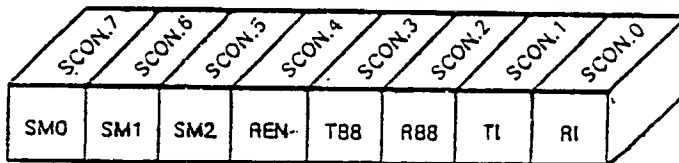
GATE บิตเลือกการควบคุมให้รีจิสเตอร์สำหรับใช้เป็นไทม์เมอร์หรือเคาน์เตอร์ทำงานโดยการควบคุมจากฮาร์ดแวร์หรือซอฟต์แวร์ดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- เมื่อเปิด TRx (TR0 , TR1) และ GATE ถูกเซต ไทม์เมอร์หรือเคาน์เตอร์จะทำงานต่อ เมื่อสถานะที่ขา INTx (INTO , INT1) มีค่าเป็น 1 (ควบคุมจากฮาร์ดแวร์)
- เมื่อเปิด GATE ถูกเคลียร์ ไทม์เมอร์หรือเคาน์เตอร์จะทำงานก็ต่อเมื่อเปิด TRx ถูกเซต โดยไม่ขึ้นกับสถานะที่ขา INTx (ควบคุมจากซอฟต์แวร์)

- C/T บิตเลือกการทำงานของรีจิสเตอร์สำหรับใช้เป็นไทม์เมอร์หรือเคาน์เตอร์ดังนี้
- 0 : หมายถึงทำงานเป็นไทม์เมอร์ (นับจำนวนแมชชีนไซเคิล)
 - 1 : หมายถึงทำงานเป็นเคาน์เตอร์ (นับจำนวนพัลส์ภายนอกที่ขา Tx)
- M1 บิตสำหรับเลือกโหมดการทำงานของไทม์เมอร์ 0 หรือ ไทม์เมอร์ 1
- M0 บิตสำหรับเลือกโหมดการทำงานของไทม์เมอร์ 0 หรือ ไทม์เมอร์ 1
- 0 0 โหมด 0 : ไทม์เมอร์หรือเคาน์เตอร์ขนาด 13 บิต
 - 0 1 โหมด 1 : ไทม์เมอร์หรือเคาน์เตอร์ขนาด 16 บิต
 - 1 0 โหมด 2 : ไทม์เมอร์หรือเคาน์เตอร์ขนาด 8 บิต ที่มีการไหลดค่าเองเมื่อมี overflow
 - 1 1 โหมด 3 : ไทม์เมอร์ 0
 - รีจิสเตอร์ TLO ใช้เป็นไทม์เมอร์หรือเคาน์เตอร์ขนาด 8 บิตที่ควบคุมการทำงานได้จากบิตของไทม์เมอร์ 0 เอง
 - รีจิสเตอร์ TH0 ใช้เป็นไทม์เมอร์ขนาด 8 บิตที่ควบคุมการทำงานได้จากบิตของไทม์เมอร์ 1
 - 1 1 โหมด 3 : ไทม์เมอร์ 1 หยุดการทำงาน

3) รีจิสเตอร์ใช้งานเฉพาะ SCON (Serial Port Control Register) สามารถเข้าถึงข้อมูลได้ในระดับบิต ดังแสดงในรูปที่ 2.14



รูปที่ 2.14 รีจิสเตอร์ใช้งานเฉพาะ SCON

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

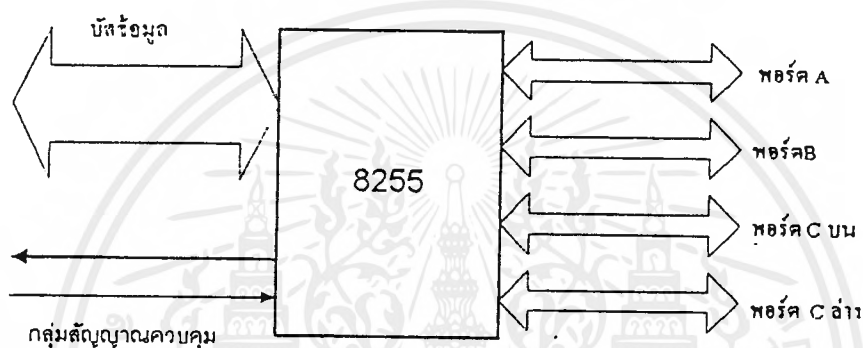
- SCON.7 SM0 บิตเลือกการทำงานของพอร์ตสื่อสารอนุกรมในโหมดต่าง ๆ
- SCON.6 SM1 บิตเลือกการทำงานของพอร์ตสื่อสารอนุกรมในโหมดต่าง ๆ
- 0 0 โหมด 0 : ทำงานเป็นซีพียูจัสเตอร์มี บอดเรท = $1/12$ ของความถี่
ออสซิลเลเตอร์
- 0 1 โหมด 1 : 8 บิต UART บอดเรทกำหนดเองได้
- 1 0 โหมด 2 : 9 บิต UART บอดเรท = $1/32$ หรือ $1/64$ ของความถี่
ออสซิลเลเตอร์
- 1 1 โหมด 3 : 9 บิต UART บอดเรทกำหนดเองได้
- SCON.5 SM2 บิตเลือกการใช้งานพอร์ตสื่อสารอนุกรมในโหมด 2 และ 3 เพื่อใช้ติดต่อ
ระหว่างซีพียูด้วยกันเอง
- 1 : ใช้พอร์ตสื่อสารอนุกรมในการติดต่อระหว่างซีพียูด้วยกันเอง
- 0 : ใช้พอร์ตสื่อสารอนุกรมตามปกติ
- SCON.4 REN บิตควบคุมการอนุญาตให้มีการรับข้อมูล ดังนี้
- 1 : อนุญาตให้มีการรับข้อมูลจากภายนอกได้
- 0 : ไม่อนุญาตให้มีการรับข้อมูลจากภายนอก
- SCON.3 TB8 บิตข้อมูลบิตที่ 9 ซึ่งจะถูกส่งออกไปในการทำงานของพอร์ตสื่อสารอนุกรม
โหมด 2 และ 3 การเซตหรือเคลียร์จะกระทำด้วยคำสั่งในโปรแกรมเท่านั้น
- SCON.2 RB8 บิตข้อมูลบิตที่ 9 ได้รับมาจากภายนอกในการทำงานของพอร์ตสื่อสารอนุกรม
โหมด 2 และ 3 ส่วนในการทำงานโหมด 1 ถ้าหากบิต SM2 = 0 บิตนี้จะเป็น
บิตสิ้นสุดของข้อมูลที่ได้รับเข้ามาได้ และไม่ถูกกำหนดการใช้งานในโหมด 0
- SCON.1 TI บิตบอกสถานะสัญญาณอินเตอร์รัปต์ที่เกิดจากการส่งข้อมูลถูกเซตโดย
ฮาร์ดแวร์ เมื่อข้อมูลบิตที่ 8 ถูกส่งออกไปแล้วในการทำงานโหมด 0 ส่วนใน
การทำงานในโหมดอื่น ๆ จะถูกเซตโดยฮาร์ดแวร์ เมื่อเริ่มส่งบิตสิ้นสุดของ
ข้อมูลออกไปและจะต้องถูกเคลียร์โดยคำสั่งในโปรแกรมเท่านั้น
- SCON.0 RI บิตบอกสถานะสัญญาณอินเตอร์รัปต์ที่เกิดจากการรับข้อมูลถูกเซตโดย
ฮาร์ดแวร์ เมื่อรับข้อมูลบิตที่ 8 เรียบร้อยแล้วในการทำงานโหมด 0 หรือที่จุด
ครึ่งทางของช่วงรับบิตสิ้นสุดของการทำงานในโหมดอื่น (มีข้อยกเว้นในกรณี
ใช้พอร์ตสื่อสารอนุกรมติดต่อระหว่างซีพียูด้วยกันเอง) และจะต้องถูกเคลียร์
โดยคำสั่งในโปรแกรมเท่านั้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.3 การใช้งานพอร์ตขนาน

2.3.1 ทฤษฎี 8255 พอร์ตข้อมูลแบบขนาน

8255 เป็นไอซีที่มี 40 ขา สามารถที่จะต่อเป็นพอร์ตให้ไมโครโปรเซสเซอร์ได้ 3 พอร์ต โดยมีโครงสร้างพื้นฐานแสดงได้ดังรูป 2.15

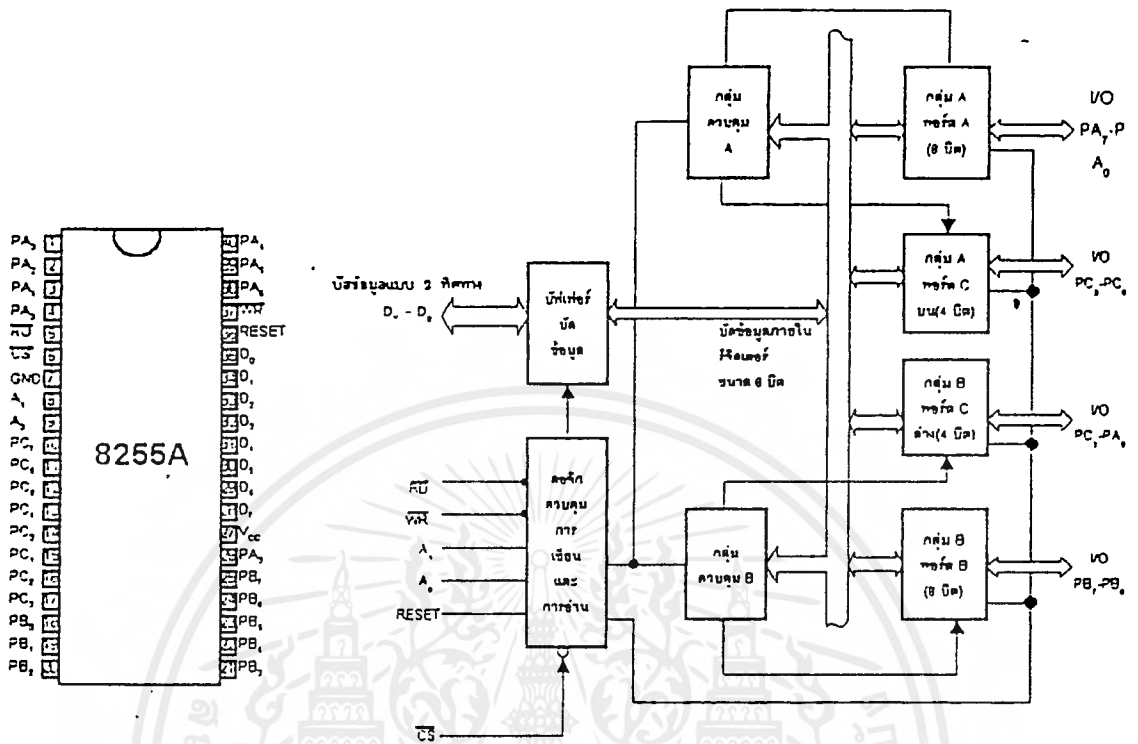


รูปที่ 2.15 แผนผังโครงสร้างของไอซี 8255

การเรียกพอร์ตของ 8255 จะเรียกพอร์ตต่าง ๆ ว่า พอร์ต A พอร์ต B และพอร์ต C โดยพอร์ต C แยกเป็น 2 ส่วนคือ พอร์ต C ล่างหรือตั้งแต่ $PC_0 - PC_3$ มีจำนวน 4 บิต และ พอร์ต C บนหรือตั้งแต่ $PC_4 - PC_7$ ที่พิเศษคือ พอร์ตทุกพอร์ตเป็นได้ทั้งพอร์ตอินพุตและพอร์ตเอาต์พุต

แผนผังภายในของไอซีและการจัดวางขาของไอซี 8255 แสดงดังในรูปที่ 2.16 การทำงานของวงจรจะใช้สัญญาณควบคุมจากไมโครโปรเซสเซอร์มาควบคุม การทำงาน โดยไมโครโปรเซสเซอร์จะส่งคำสั่งมาโปรแกรมการทำงาน หรือ กำหนดรูปแบบของพอร์ตให้เป็นอินพุตหรือเอาต์พุตได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่นอนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.16 แผนผังวงจรภายในและการจัดขาของไอซี 8255

2.3.2 ขาต่าง ๆ ของ 8255

เพื่อให้เข้าใจวิธีในการต่อใช้งานจึงจำเป็นต้องเข้าใจในความหมายและตำแหน่งของขาต่าง ๆ เสียก่อน ขาทั้ง 40 ขาของไอซีประกอบด้วย

D₀-D₇ เป็นขาที่ข้อมูลอินพุต เอาต์พุตจะต้องผ่านเข้าออกจากส่วนนี้ D₀-D₇ จึงต่อเข้ากับระบบบัสของไมโครโปรเซสเซอร์ เพื่อให้ไมโครโปรเซสเซอร์สามารถอ่านหรือเขียนข้อมูลจากพอร์ตผ่านทางบัสนี้

CS (สัญญาณเลือกชิป) ขานี้เป็นขาอินพุตที่จะรับสัญญาณจากภายนอกเพื่อเลือกชิป 8255 โดยเมื่อขานี้เป็น "0" จะควบคุมให้ 8255 ต่อเข้ากับระบบบัสของไมโครโปรเซสเซอร์ เพื่อให้ไมโครโปรเซสเซอร์เขียนหรืออ่านข้อมูลจากพอร์ตได้

RD (สัญญาณการอ่าน) เป็นสัญญาณอินพุตที่ต้องส่งมาจากซีพียู เมื่อสัญญาณที่ขานี้เป็น "0" และสัญญาณ CS เป็น "0" ด้วย ไอซี 8255 จะทำตัวให้ซีพียูอ่านข้อมูลจากบัสในขณะที่เป็นพอร์ตอินพุต

WR เป็นสัญญาณการเขียน จะแอกทีฟเมื่อสัญญาณ WR และสัญญาณ CS เป็น "0" สัญญาณนี้จะมาจากซีพียูเมื่อต้องการเขียนข้อมูลลงบนพอร์ตที่กำหนด

A₀-A₇ (สัญญาณแอดเดรส) ลอจิกของสัญญาณทั้งสองจะถอดรหัสออกเป็น 4 รหัส เพื่อกำหนดรีจิสเตอร์ภายในที่เชื่อมต่อกับพอร์ตอินพุตเอาต์พุตของ 8255

RESET (สัญญาณรีเซ็ต) เป็นสัญญาณที่ส่งมาจากภายนอกเข้ามาทำการรีเซ็ต 8255 เพื่อเคลียร์สถานะต่าง ๆ ของ 8255 เมื่อ 8255 ได้รับการรีเซ็ต ก็จะถูกเข้าสู่โหมดอินพุตหรือทุกพอร์ตเป็นพอร์ตอินพุต

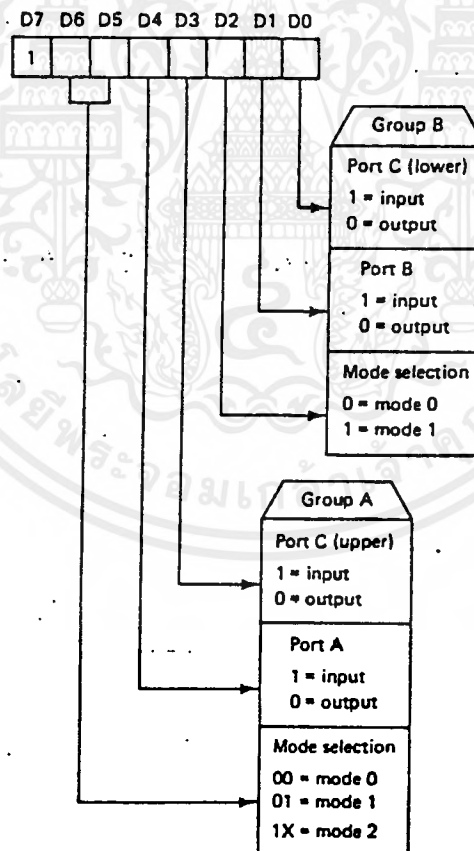
$PA_0 - PA_7$ เป็นสายสัญญาณที่เป็นพอร์ตของ 8255 ที่ชื่อพอร์ต A การเลือกพอร์ตจะเลือกโดยสัญญาณแอดเดรส $A_0 - A_1$

$PB_0 - PB_7$ เป็นสายสัญญาณที่เป็นพอร์ต B ของ 8255 ซึ่ง ถูกเลือกโดยสัญญาณแอดเดรส $A_0 - A_1$

$PC_0 - PC_7$ เป็นสายสัญญาณที่เป็นพอร์ต C ของ 8255 การกำหนดพอร์ตนี้จะได้รับการกำหนดโดยสัญญาณแอดเดรส $A_0 - A_1$ พอร์ต C นี้แบ่งออกเป็น 2 กลุ่ม คือกลุ่ม $PC_0 - PC_3$ และกลุ่ม $PC_4 - PC_7$

2.3.3 โหมด 0 หรือ อินพุต-เอาต์พุตแบบพื้นฐาน

การกำหนดโหมดการทำงาน จะต้องส่งข้อมูลคำสั่งเข้าไปโปรแกรมในพอร์ตควบคุมของ 8255 แต่ละบิตของข้อมูลที่ส่งไปจะมีความหมายในตัวเอง ลักษณะความหมายของแต่ละบิตในรหัสควบคุมแสดงได้ดังรูปที่ 2.17



รูปที่ 2.17 ความหมายของบิตต่าง ๆ ในรหัสควบคุม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การโปรแกรม 8255 คือการให้ค่ารหัสบิตต่าง ๆ เข้าไปในรหัสควบคุมแล้วส่งไปยังรีจิสเตอร์ของพอร์ตควบคุม ความหมายของบิตต่าง ๆ มีดังนี้

บิต D_7 เป็นบิตที่แสดงรหัสคำสั่งควบคุม ถ้าบิตนี้เป็น "1" หมายถึงรหัสควบคุมนี้จะมีผลต่อการเปลี่ยนแปลงการเซตโหมดต่าง ๆ ของ 8255

บิต D_6 และ D_5 เป็นการเลือกโหมดของพอร์ต A ซึ่งมี 3 โหมด คือ โหมด 0, โหมด 1 และโหมด 2 ดังแสดงในรูปที่ 2.6

บิต D_4 ถ้ามีค่าเป็น "0" หมายถึงการกำหนดพอร์ต A เป็นเอาต์พุต ถ้ามีค่าเป็น "1" จะหมายถึงการกำหนดให้พอร์ต A เป็นอินพุต

บิต D_3 เป็นบิตที่บอกถึง การเซตของพอร์ต C ถ้าเป็น "0" จะทำให้พอร์ต C บน เป็นเอาต์พุต

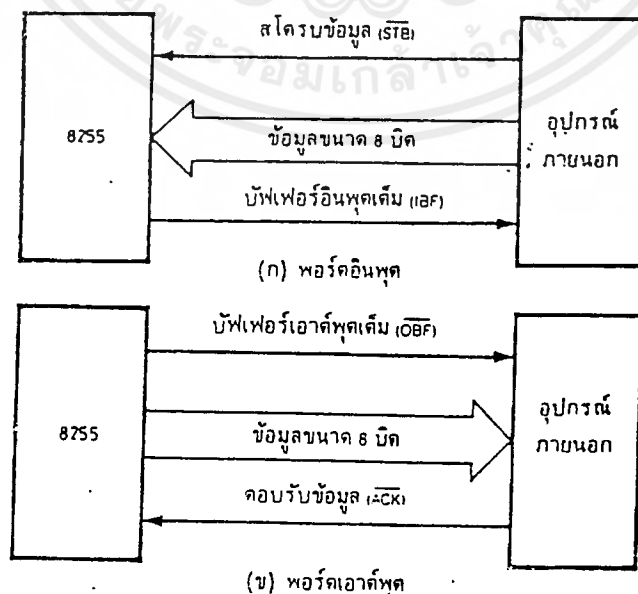
บิต D_2 เป็นบิตที่บอกถึงการเซตโหมดของพอร์ต B ถ้าเป็น "0" หมายถึง การเลือกพอร์ต B เป็นโหมด 0 และถ้าเป็น "1" หมายถึง การเลือกโหมด 1

บิต D_1 เป็นการกำหนดอินพุตเอาต์พุตของพอร์ต B ถ้าเป็น "0" หมายถึงเอาต์พุต ถ้าเป็น "1" หมายถึงอินพุต

บิต D_0 เป็นการกำหนดอินพุตเอาต์พุตของพอร์ต C ถ้าเป็น "0" หมายถึงเอาต์พุต ถ้าเป็น "1" หมายถึง อินพุต

2.3.4 การทำงานของ 8255 ในโหมด 1

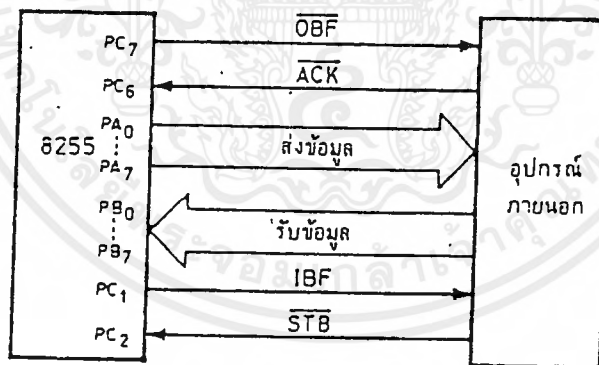
การทำงานของ 8255 ในโหมด 1 เป็นโหมดที่ทำให้อินพุตเอาต์พุต มีการตรวจสอบสัญญาณ (handshaking) ของพอร์ต A และพอร์ต B เป็นหลัก และใช้พอร์ต C บนเป็นตัวตรวจสอบสัญญาณ (handshake) ของพอร์ต A ส่วนพอร์ต C ล่างเป็นตัวตรวจสอบสัญญาณของพอร์ต B การจัดสัญญาณต่าง ๆ เหล่านี้ แสดงได้ดังรูปที่ 2.18



รูปที่ 2.18 โครงสร้างตัวตรวจสอบสัญญาณของพอร์ตอินพุตและเอาต์พุต

ขา	กรณีอินพุต	กรณีเอาต์พุต
PC ₀	INTR _B	INTR _B
PC ₁	IBF _B	$\overline{\text{OBF}}_B$
PC ₂	$\overline{\text{STB}}_B$	$\overline{\text{ACK}}_B$
PC ₃	INTR _A	INTR _A
PC ₄	$\overline{\text{STB}}_A$	I/O
PC ₅	IBF _A	I/O
PC ₆	I/O	$\overline{\text{ACK}}_A$
PC ₇	I/O	$\overline{\text{OBF}}_A$

ตารางที่ 2.1 หน้าที่ของสัญญาณต่าง ๆ ของพอร์ต C เมื่อทำงานในโหมด 1



รูปที่ 2.19 วงจรการต่อ 8255 ในโหมด 1

เมื่อโปรแกรม 8255 เป็นโหมด 1 แล้ว ตัว 8255 จะให้พอร์ต C เป็นสัญญาณควบคุม โดยแต่ละบิตของพอร์ต C เป็นไปตามที่กำหนดไว้ ดังตารางที่ 2.1

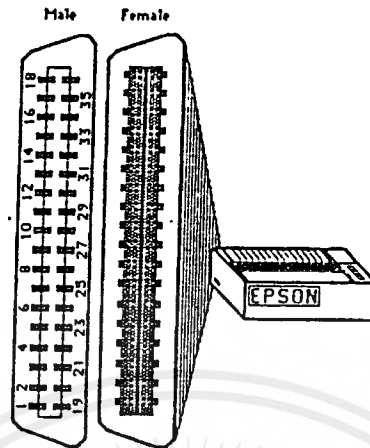
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.4 การอินเตอร์เฟสกับพริ้นเตอร์แบบขนานและขบวนการเชื่อมต่อแบบเซ็นโทรนิคส์

ข้อมูลที่จะส่งไปพิมพ์ที่พริ้นเตอร์จะส่งในลักษณะของรหัสแอสกี (ASCII) ผ่านสายแบบขนาน 8 สาย พริ้นเตอร์รับตัวอักษรเพื่อจะพิมพ์ และ เก็บข้อมูลไว้ในบัฟเฟอร์แรมภายใน เมื่อพริ้นเตอร์ตรวจพบอักขระแคริเอจ รีเทิร์น (Carriage Return ; CR) มันจะพิมพ์อักขระแถวแรกจากบัฟเฟอร์ เมื่อพริ้นเตอร์ตรวจพบแคริเอจ รีเทิร์น ตัวที่สอง มันจะพิมพ์อักขระแถวที่สองออกมา ขบวนการต่าง ๆ จะดำเนินต่อไปจนกระทั่งตัวอักษรที่ต้องการทั้งหมดถูกพิมพ์

การแปลงรหัสแอสกีจากไมโครคอมพิวเตอร์ส่งไปยังพริ้นเตอร์จะต้องทำบนพื้นฐานการแฮนด์เชค เพราะ ไมโครคอมพิวเตอร์สามารถส่งตัวอักษรเร็วกว่าความสามารถในการพิมพ์ของพริ้นเตอร์มาก พริ้นเตอร์จะต้องมีแนวทางเพื่อที่จะบอกให้ไมโครคอมพิวเตอร์รู้ว่าบัฟเฟอร์ของมันเต็มแล้ว และ ไม่สามารถรับข้อมูลอักขระได้อีกจนกระทั่งมันพิมพ์ออกมาแล้ว มาตรฐานสำหรับการอินเตอร์เฟสกับพริ้นเตอร์ขนานของพริ้นเตอร์เป็นมาตรฐานของ เซ็นโทรนิคส์ (Centronics Parallel Standard)

พริ้นเตอร์ชนิดเซ็นโทรนิคส์ โดยปกติแล้วจะมี 36 ขา ในการอินเตอร์เฟส (Interface) ตารางที่ 2.2 แสดงการกำหนดขา และ คำอธิบายสำหรับสำหรับขาคอนเนคเตอร์ (Connector) การที่มีขาสำหรับการเชื่อมต่อกับพริ้นเตอร์มากถึง 36 ขา นั้นเนื่องจากสายของแต่ละสัญญาณข้อมูลจะมีสายกราวนด์ (Ground) ของมันแยกออกจากกัน เช่น จากตารางที่ 2.2 ขา 2 คือ บิตข้อมูลที่มีนัยสำคัญต่ำสุด (LSB) ของข้อมูลที่จะส่งไปยังพริ้นเตอร์ และ ขา 20 จะเป็นกราวนด์สำหรับสายสัญญาณนี้ เหตุผลสำหรับการแยกสัญญาณกราวนด์นี้ เพื่อลดสัญญาณรบกวนในสาย ถ้าจะทำการต่อสายเคเบิลแบบขนานกับพริ้นเตอร์ สายกราวนด์ทั้งหมดจะต้องถูกต่อเข้าด้วยกันเพื่อต่อกับกราวนด์ของไมโครคอมพิวเตอร์ที่ปลายของสายเคเบิล โดยขา 16 เรียกว่า ลอจิกกราวนด์ (Logic Ground) และ ขา 17 เรียกว่า แชสสิสกราวนด์ (Chassis Ground) เพื่อที่จะป้องกันกระแสสัญญาณรบกวนที่มาจากสายของลอจิกกราวนด์ เราจึงต้องเชื่อมต่อสายเหล่านี้เข้าด้วยกันกับไมโครคอมพิวเตอร์ ส่วนขาอื่น ๆ ในจำนวน 36 ขา ของคอนเนคเตอร์นั้น แบ่งออกได้เป็น 2 ประเภท คือ สัญญาณที่ส่งไปยังพริ้นเตอร์ เพื่อที่จะบอกว่าอะไรกำลังทำงาน และ สัญญาณจากพริ้นเตอร์เพื่อที่จะแสดงสถานะของมัน สัญญาณควบคุมหลักไปยังพริ้นเตอร์ คือ \overline{INIT} ที่ขา 31 ซึ่งจะบอกพริ้นเตอร์เพื่อให้ทำการกำหนดค่าเริ่มต้นภายในของลำดับ และ \overline{STROBE} ที่ขา 1 ซึ่งจะบอกพริ้นเตอร์ว่าขณะนี้ข้อมูลส่งมาแล้ว ส่วนอีก 2 ขาอินพุท (Input) คือ ขา 14 และ ขา 36 โดยปกติจะใช้สำหรับดูแลภายในพริ้นเตอร์ โดยตัวคอนเนคเตอร์จะมีลักษณะดังรูปที่ 2.20



รูปที่ 2.20 แสดงขาของคอนเน็คเตอร์

ตารางที่ 2.2 แสดงการเชื่อมต่อขาและคำอธิบายสำหรับการเชื่อมต่อแบบเซนโทรนิคส์ขนาน

ขา สัญญาณ ที่	ขา สัญญาณ ตอกลับ	สัญญาณ	ทิศทาง	คำอธิบาย
1	19	STROBE	เข้า	พัลส์ (PULSE) STROBE เพื่อจะอ่านข้อมูลเข้ามา ความกว้างของพัลส์จะต้องมากกว่า 0.5 μ s ที่เทอร์มินัล ด้านรับ ระดับสัญญาณปกติจะอยู่ในสภาวะ “ สูง ” การอ่านข้อมูลเข้ามากกระทำที่ระดับสัญญาณ “ ต่ำ ”
2	20	DATA 1	เข้า	สัญญาณเหล่านี้ จะแสดงข่าวสารข้อมูลของบิทที่ 1 - 8 ของข้อมูลขนานตามลำดับ แต่ละสัญญาณจะมีระดับ “ สูง ” เมื่อมีข้อมูลเป็นลอจิก “ 1 ” และมีระดับ “ ต่ำ ” เมื่อมีข้อมูลเป็นลอจิก “ 0 ”
3	21	DATA 2	เข้า	
4	22	DATA 3	เข้า	
5	23	DATA 4	เข้า	
6	24	DATA 5	เข้า	
7	25	DATA 6	เข้า	
8	26	DATA 7	เข้า	
9	27	DATA 8	เข้า	

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไมออนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ขา สัญญาณ ที่	ขา สัญญาณ ตอกลับ	สัญญาณ	ทิศทาง	คำอธิบาย
10	28	\overline{ACKNLG}	ออก	พัลส์ประมาณ 5 μ s สถานะ " ต่ำ " แสดงว่าข้อมูล นั้นถูกรับโดยพรีนเตอร์แล้วและพรีอัมที่จะรับข้อมูลอื่นต่อไป
11	29	BUSY	ออก	สัญญาณ " สูง " แสดงว่าพรีนเตอร์ไม่สามารถรับข้อมูล ถ้าสัญญาณกลายเป็นสถานะ " สูง " จะมีกรณีต่าง ๆ ดังนี้ 1. อยู่ในระหว่างข้อมูลเข้า 2. อยู่ในระหว่างกระทำการพิมพ์ 3. อยู่ในสถานะออฟไลน์ (offline) 4. อยู่ในสภาวะเกิดเหตุผิดพลาดที่พรีนเตอร์
12	30	PE	ออก	สัญญาณ " สูง " แสดงว่าพรีนเตอร์อยู่ในสภาวะไม่มีกระดาษ
13	-	SLCT	ออก	สัญญาณนี้แสดงว่าพรีนเตอร์อยู่ในสภาวะถูกเลือก
14	-	\overline{AUTO} \overline{FEEDXT}	เข้า	ถ้าสัญญาณนี้อยู่ที่ระดับ " ต่ำ " กระดาษจะถูกเลื่อนโดยอัตโนมัติ 1 บรรทัด หลังจากการพิมพ์ (ระดับสัญญาณสามารถกำหนดที่ระดับ " ต่ำ " กับดิฟเฟอเรนเชียลขา 2-3 เตรียมไว้สำหรับควบคุมแผงวงจร)
15	-	NC	-	ไม่ใช้
16	-	OV	-	ระดับลอจิกกราวด์
17	-	CHASSIS - GND	-	พรีนเตอร์ แชนลิสกราวด์ ในพรีนเตอร์ แชนลิสกราวด์ และลอจิกกราวด์จะแยกออกจากกันและกัน
18	-	NC	-	ไม่ใช้
19 - 30	-	GND	-	สัญญาณทวิสต์แพร์รีเทิร์น (Twisted Pair Return) ระดับกราวด์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ขา สัญญาณ ที่	ขา สัญญาณ ตอบกลับ	สัญญาณ	ทิศทาง	คำอธิบาย
31	-	\overline{INIT}	เข้า	เมื่อระดับของสัญญาณนี้กลายเป็นสถานะ " ต่ำ " ส่วนควบคุมพริ้นเตอร์จะถูกรีเซตเพื่อไปที่ค่าสถานะเริ่มต้นของมัน (Initial State) และพริ้นท์บัฟเฟอร์ (Print Buffer) จะถูกเคลียร์ สัญญาณปกติจะอยู่ที่ระดับ " สูง " และ ความกว้างของพัลส์จะต้องมากกว่า 50 μ s ที่ เทอร์มินอลด้านรับ
32	-	\overline{ERROR}	ออก	ระดับของสัญญาณนี้จะกลายเป็นสถานะ " ต่ำ " เมื่อพริ้นเตอร์อยู่ในสถานะเปเปอร์เอนด์ (Paper End) ออฟไลน์ หรือ เออเรอร์ (Error)
33	-	GND	-	เหมือนกับขาที่ 19 - 30
34	-	NC	-	ไม่ใช่
35	-			ต่อกับ 5 Vdc ผ่านตัวต้านทาน 4.7 K-ohms
36	-	\overline{SLCTIN}	เข้า	ข้อมูลเข้ามาที่พริ้นเตอร์อาจเป็นไปไม่ได้เมื่อระดับของสัญญาณนี้อยู่ในสถานะ " ต่ำ " (การกำหนดภายในสามารถทำได้ที่ดิฟเฟอเรนเชียล 1 - 8 เงื่อนไขที่เวลาของการส่งจะถูกกำหนดให้อยู่ในสถานะ " ต่ำ " สำหรับสัญญาณนี้)

หมายเหตุ

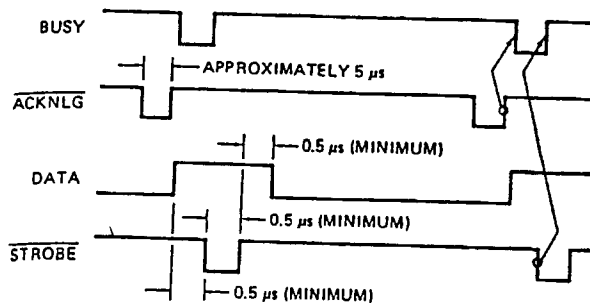
1. ทิศทาง หมายถึง ทิศทางการไหลของสัญญาณที่มองจากพริ้นเตอร์
2. สัญญาณตอบกลับ หรือ ทวิสท์แพร์รีเทิร์น จะถูกต่อกับสัญญาณที่ระดับกราวด์ เมื่อมีการอินเตอร์เฟส ต้องแน่ใจว่าใช้สายเคเบิลแบบทวิสท์แพร์ สำหรับแต่ละสัญญาณ และ ต้องเชื่อมต่ออย่างสมบูรณ์ที่ด้านตอบกลับ
3. เงื่อนไขทั้งหมดของการอินเตอร์เฟสมีพื้นฐานบนระดับของ TTL ทั้งเวลาขาขึ้น และ ขาลงของแต่ละสัญญาณต้องน้อยกว่า 0.2 μ s
4. ข้อมูลที่ส่งต้องมีสัญญาณ \overline{ACKNLG} หรือ สัญญาณ BUSY ด้วย (ข้อมูลที่จะส่งไปที่พริ้นเตอร์นี้สามารถส่งเดี่ยว ๆ หลังจากมีการยืนยันสัญญาณ \overline{ACKNLG} หรือ เมื่อระดับของสัญญาณ BUSY เป็นสถานะ " ต่ำ ")

สัญญาณแสดงสถานะเอาต์พุต (Output) จากพรีนเตอร์มีดังนี้

1. สัญญาณ \overline{ACKNLG} ที่ขา 10 ซึ่งจะมีสถานะ “ ต่ำ ” (Low) จะแสดงว่า ข้อมูลอักขระได้ถูก
รับโดยพรีนเตอร์แล้ว และพร้อมที่จะรับตัวอักขระต่อไป
2. สัญญาณ BUSY ที่ขา 11 ซึ่งจะมีสถานะ “ สูง ” (High) ในกรณีที่ไม่มีกระดาษ หรือพรีนเตอร์ยังไม่พร้อมที่จะรับข้อมูลอักขระ
3. สัญญาณ PE ที่ขา 12 ซึ่งจะมีสถานะ “ สูง ” ถ้าสวิทช์ของการไม่มีกระดาษในพรีนเตอร์ถูก
กระตุ้น
4. สัญญาณ SLCT ที่ขา 13 ซึ่งจะมีสถานะ “ สูง ” ถ้าพรีนเตอร์ถูกเลือกสำหรับการรับข้อมูล
5. สัญญาณ \overline{ERROR} ที่ขา 32 ซึ่งจะมีสถานะ “ ต่ำ ” เมื่อเกิดปัญหาในเงื่อนไขต่าง ๆ ของพรีน
เตอร์

รูปที่ 2.21 แสดงช่วงเวลาของรูปคลื่นสัญญาณสำหรับการส่งข้อมูลตัวอักขระไปที่พรีนเตอร์โดยใช้พื้นฐานของสัญญาณแฮนด์เชค (Handshake Signals) สมมติว่าพรีนเตอร์ได้ถูกกำหนดค่าเริ่มต้นแล้ว ขั้นแรกต้องทำการตรวจสอบสัญญาณที่ขา BUSY เพื่อจะดูว่าพรีนเตอร์พร้อมที่จะรับข้อมูลหรือไม่ ถ้าสัญญาณอยู่ในสถานะ “ ต่ำ ” แสดงว่าพรีนเตอร์พร้อมจะรับข้อมูล เราจะสามารถส่งรหัสแอสกีบนสายข้อมูลขนานทั้ง 8 สายได้ หลังจากอย่างน้อย $0.5 \mu\text{s}$ เราแสดงสัญญาณ \overline{STROBE} ที่สถานะ “ ต่ำ ” เพื่อจะบอกพรีนเตอร์ว่าข้อมูลตัวอักขระถูกส่งไปแล้ว สัญญาณ \overline{STROBE} จะเป็นสถานะ “ ต่ำ ” เพราะพรีนเตอร์แสดงสัญญาณ BUSY ของมันเป็นสถานะ “ สูง ” หลังจากเวลาอย่างน้อยที่สุด $0.5 \mu\text{s}$ แล้วสัญญาณ \overline{STROBE} สามารถขึ้นเป็นสถานะ “ สูง ” ได้อีกครั้งหนึ่ง โดยข้อมูลข่าวสารต้องคงสภาพการใช้งานได้บนสายข้อมูลอย่างน้อย $0.5 \mu\text{s}$ หลังจากสัญญาณ \overline{STROBE} ถูกทำให้อยู่ในสถานะ “ สูง ”

เมื่อพรีนเตอร์พร้อมจะรับข้อมูลอักขระถัดไป มันจะแสดงสัญญาณ \overline{ACKNLG} ในสถานะ “ ต่ำ ” เป็นเวลา $5 \mu\text{s}$ ขอบขาขึ้นของสัญญาณ \overline{ACKNLG} จะบอกไมโครคอมพิวเตอร์ว่ามันสามารถส่งข้อมูลอักขระถัดไปได้แล้ว ขอบขาขึ้นของสัญญาณ \overline{ACKNLG} จะทำการรีเซ็ตสัญญาณ BUSY จากพรีนเตอร์ สัญญาณ BUSY จะเป็นสถานะ “ ต่ำ ” เพื่อจะแสดงว่าพรีนเตอร์พร้อมจะรับข้อมูลอักขระถัดไป บางระบบจะใช้สัญญาณ \overline{ACKNLG} สำหรับการแฮนด์เชค และ ระบบจะใช้สัญญาณ BUSY



รูปที่ 2.21 แสดงช่วงเวลาของรูปคลื่นสัญญาณที่จะส่งข้อมูลอักขระไปยังพรีนเตอร์แบบขนาน

2.5 การต่อแบบขนานแบบไอบีเอ็ม (IBM)

เวลานี้เราใช้การส่งข้อมูลแบบขนานเป็นมาตรฐาน โดยไอบีเอ็มได้เป็นผู้ออกแบบไว้ซึ่งใช้การเชื่อมต่อโดย DB-25 มี 25 ขาสำหรับการติดต่อแบบขนาน หมายความว่าขา 36 ขานั้นไม่ได้ใช้ทั้งหมด เครื่องพิมพ์ส่วนใหญ่ที่ติดต่อแบบขนานจะใช้แอมพินอลเพราะ DB-25 ไม่สามารถใช้แทนแอมพินอลได้

การต่อขาของ DB-25 จะเป็นตัวผู้และพอร์ตของคอมพิวเตอร์จะเป็นตัวเมีย ดังนั้นในการต่อออกจากคอมพิวเตอร์จะต้องดูด้วยว่าจะต่อแบบใด ฉะนั้นตัวเมียจะเป็นพอร์ตแบบขนาน ส่วนตัวผู้จะเป็นพอร์ตอนุกรมโดยขาของ DB-25 แสดงไว้ในตารางที่ 2.3 ส่วนตารางที่ 2.4 เป็นตัวปรับให้การต่อของ DB-25 กับแอมพินอลเหมือนกัน

ตารางที่ 2.3 แสดงขาพอร์ตแบบขนานของไอบีเอ็ม

DB25 Pin	Function
1	Strobe
2	Data bit 0
3	Data bit 1
4	Data bit 2
5	Data bit 3
6	Data bit 4
7	Data bit 5
8	Data bit 6
9	Data bit 7
10	Acknowledge
11	Busy
12	Paper end (out of paper)
13	Select
14	Auto feed
15	Error
16	Initialize printer (reset)
17	Select input
18-25	Ground

ตารางที่ 2.4 แสดงการปรับต่อขาของ DB-25 และ แอมฟีนอล

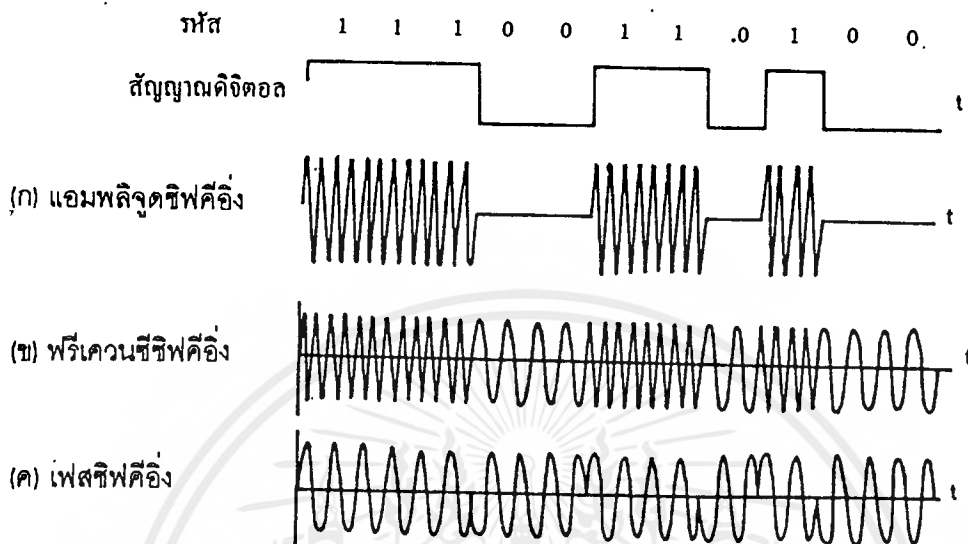
DB25	Amphenon
1	1
2	2
3	3
4	4
5	5
6	6
7	7
8	8
9	9
10	10
11	11
12	12
13	13
14	14
15	32
16	31
17	36
18	33
19	19
20	21
22	25
23	27
24	29
25	30

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.6 ดิจิตอลโมดูเลชัน

ปัจจุบันการสื่อสารนิยมใช้ดิจิตอลโมดูเลชันมากขึ้น เพราะระบบดิจิตอลโมดูเลชันมีความเชื่อถือสูง และมีราคาถูกลงเนื่องจากได้มีการพัฒนาทางดิจิตอลอิเล็กทรอนิกส์และไอซีอย่างรวดเร็วทำให้ต้นทุนการผลิตถูกลงนอกจากนี้ดิจิตอลโมดูเลชันยังสามารถใช้วิธีการลงรหัสพัลส์ก่อนโมดูเลชันเพื่อลดพหุคูณบิตลดความผิดพลาดให้ต่ำลงการโมดูเลชันตัวพาด้วยข่าวสารดิจิตอลนี้ทำได้หลายวิธีด้วยกันในที่นี้เราจะศึกษาเทคนิคการโมดูเลชันที่ข่าวสารดิจิตอลเปลี่ยนแปลงขนาดความถี่และเฟสของตัวพา รูปที่ 2.22 แสดงรูปคลื่นของดิจิตอลโมดูเลชันวิธีต่าง ๆ ที่ใช้ในการส่งข่าวสารไบนารีผ่านช่องสื่อสาร (Communication Channel) ซึ่งโดยมากจะมีการตอบสนองไม่ถี่ บริเวณความถี่ใกล้ศูนย์จึงคำนึงได้ว่าเป็นช่องความถี่ผ่านแคบ (Bandpass Channel) ในรูปที่ 2.22 ก. สัญญาณดิจิตอล 1 และ 0 จะสวิตช์ขนาดของตัวพาให้มียาค่าสองค่าคือ เปิดและปิดรูปคลื่นที่ประกอบด้วยพัลส์เปิดหรือมาร์ค (Mark) จะแทนเลขไบนารี "1" และพัลส์ปิดหรือสเปซ (Space) จะแทนเลขไบนารี " 0 " ดิจิตอลโมดูเลชันนี้เรียกว่า แอมพลิจูดชิฟคีย์อิง (Amplitude Shift Keying หรือ ASK) รูปที่ 2.22 ข. แสดงรูปคลื่นของฟริควีนซีชิฟคีย์อิง (Frequency Shift Keying หรือ FSK) ที่มีการเปลี่ยนความถี่ของตัวพาหระหว่างสองความถี่ตามสัญญาณดิจิตอล 1 และ 0 แต่สัญญาณมาร์คที่มีความถี่สูงจะแทนพัลส์ 1 และสัญญาณสเปซที่มีความถี่ต่ำจะแทนด้วยพัลส์ 0 ส่วนรูปที่ 2.22 ค. แสดงรูปคลื่นของเฟสชิฟคีย์อิง (Phase Shift Keying หรือ PSK) ที่มีการเปลี่ยนเฟสของตัวพาหระหว่างสองเฟสคือ 0 องศา และ 180 องศา ตามสัญญาณดิจิตอล 1 และ 0 เฟสชิฟคีย์อิงชนิดนี้เรียกว่า สองเฟสพีเอสเค (2 PSK) มีข้อสังเกตว่าในกรณีของเอฟเอสเค และพีเอสเค ขนาดของตัวพาหจะมีค่าคงที่เสมอ และรูปคลื่นที่ถูกโมดูเลชันแล้วของดิจิตอลโมดูเลชันทั้งสามวิธีมีค่าต่อเนื่องตลอดเวลา

ในบทนี้เราจะศึกษาวิสัยความสามารถของระบบดิจิตอลโมดูเลชันทั้งสามระบบ ในขณะที่มีเสียงรบกวน วิสัยความสามารถของระบบดิจิตอลโมดูเลชันมักจะวัดในรูปของ พรอบบะบิวลิตีความผิดพลาด เพื่อให้เกิดความเข้าใจง่ายขึ้นเราจะศึกษาเครื่องกรองความถี่แบบออปติมัม (Optimum Filter) ซึ่งเป็นเครื่องกรองความถี่แบบแมช (Matched Filter) หรือเครื่องรับแบบคอร์รีเลชัน (Correlation Receiver) ที่ใช้ในการดีโมดูเลชันเอฟเอสเค เอฟเอสเค และพีเอสเค



รูปที่ 2.22 รูปคลื่นของดิจิทัลโมดูเลชันที่ใช้ในการส่งข้อมูลไบนารี

2.6.1 เอเอสเค (ASK)

การจัดสัญญาณเอเอสเค (ASK Signalling)

สัญญาณเอเอสเคเป็นสัญญาณดิจิทัลโมดูเลชันที่ค้นพบก่อน ระบบดิจิทัลโมดูเลชันอื่น โดยอาศัยหลักการของแอมพลิจูดโมดูเลชันสัญญาณเอเอสเคนี้ เวลาที่มีสัญญาณมาร์ค หรือ สัญญาณเปิด จะส่งสัญญาณพัลส์ เวลาที่มีสัญญาณสเปสหรือสัญญาณปิดจะไม่ส่งสัญญาณ ซึ่งมีรูปคลื่นดังนี้

$$\begin{aligned}
 S1(t) &= A \cos wct && \text{สำหรับสัญญาณมาร์ค} \\
 S2(t) &= 0 && \text{สำหรับสัญญาณสเปส.....(2.4)}
 \end{aligned}$$

สัญญาณมาร์คส่งไม่จำเป็นต้องเป็นพัลส์สี่เหลี่ยมเสมอไปอาจเป็นพัลส์ที่ผ่านเครื่องกรองความถี่ต่ำผ่านแล้วเป็นพัลส์ที่จัดรูปร่าง (Shaped pulse) ก็ได้สัญญาณเอเอสเค อาจใช้เป็นสัญญาณโทรเลข หรือเป็นสัญญาณดิจิทัลโมดูเลชันแบบต่าง ๆ ที่ไม่ต้องการความเร็วในการส่ง (bit rate) สูง สัญญาณเอเอสเคเมื่อส่งผ่านช่องสื่อสารจะถูกรบกวนโดยเสียงรบกวนเกาส์เซียนจะได้สัญญาณรวมเป็น

$$V_i(t) = \begin{cases} A \cos wct + n(t) & \text{สำหรับสัญญาณมาร์ค} \\ n(t) & \text{สำหรับสัญญาณสเปส.....(2.5)} \end{cases}$$

2.6.2 การเปรียบเทียบระบบดิจิทัลโมดูเลชัน

ในการเปรียบเทียบวิสัยความสามารถของระบบดิจิทัลโมดูเลชัน เราต้องพิจารณาในแง่ทฤษฎี หัวข้อต่อไปนี้

1. แแถบความถี่ (bandwidth)
2. พหุขบเบบิวลิตีความผิดพลาด (Probability of Error)

แต่ในแง่ปฏิบัติเวลาเราเลือกระบบดิจิทัลโมดูเลชันเพื่อใช้งานนั้น เราควรคำนึงถึงหัวข้ออื่นนอกเหนือจากหัวข้อ ที่ได้กล่าวข้างต้นด้วย เช่น อัตราความเร็วในการส่งข้อมูล ราคา ความยากง่าย ของอุปกรณ์ในการใช้ และการซ่อมบำรุงรักษา ความเข้ากันได้ของอุปกรณ์ที่จะจัดซื้อกับระบบเก่าที่มีอยู่ ภูมิคุ้มกัน (Immunity) ของระบบดิจิทัลโมดูเลชันที่มีต่อเสียงรบกวน และความเสื่อมโทรมของช่องสัญญาณ (Channel Impairment) เช่น ความไม่เป็นเชิงเส้นการสั่นของเฟส (Phase Jitter) เฟดดิ้ง (Fading) และความเสถียรภาพของความถี่ (Frequency Stability)

หัวข้อเหล่านี้ มีความสำคัญต่อการเลือกซื้อระบบดิจิทัลโมดูเลชันมาก เช่น ถ้าช่องสัญญาณ (Channel) เกิดเฟดดิ้งอยู่เสมอ ถ้าเราเลือกส่งข้อมูลด้วยสัญญาณที่เฮสเคจจะได้ค่าพหุขบเบบิวลิตีความผิดพลาดมากกว่าการเลือกส่งข้อมูลด้วยสัญญาณนอนโคฮีเร้นท์เอฟเอสเค เพราะในขณะที่เกิดเฟดดิ้งสัญญาณอ้างอิงโคฮีเร้นท์สำหรับพีเอสเคจะเพี้ยนไป ทำให้เกิดความผิดพลาดมาก เป็นต้น ในกรณีที่มีข้อจำกัดทางด้านกำลังของสัญญาณ (Power Limitation) เช่น การรับส่งสัญญาณควมเทียมควรเลือกแบบโคฮีเร้นท์ เพราะแบบโคฮีเร้นท์ที่ใช้กำลังของสัญญาณน้อยกว่าแบบนอนโคฮีเร้นท์ที่อัตราการส่ง และ ค่าพหุขบเบบิวลิตีความผิดพลาดเดียวกัน

ความต้องการแถบความถี่ (Bandwidth Requirements) สัญญาณเอเอสเค ดีพีเอสเค และพีเอสเคจะมีความต้องการความถี่ประมาณ 2 เท่าของอัตราความเร็วในการส่งข้อมูล (Data Rate) สัญญาณเอฟเอสเคต้องการแถบความถี่มากกว่า 2 เท่าของอัตราความเร็วในการส่งข้อมูล ส่วนสัญญาณดิจิทัลเอฟเอ็มที่นิยมใช้กันส่วนมากเป็นแบบแถบความถี่แคบ (Narrowband) นั้น ต้องการแถบความถี่อยู่ระหว่าง 2-3 เท่าของอัตราความเร็วในการส่งข้อมูล ฉะนั้นถ้าต้องการเลือกใช้ระบบดิจิทัลโมดูเลชันในงานที่มีข้อจำกัดทางด้านแถบความถี่ (Bandwidth Limitation) ก็ไม่ควรเลือกระบบเอฟเอสเค

2.7 วงจรออสซิลเลเตอร์

ระบบการสื่อสารโดยทั่ว ๆ ไปมีความจำเป็นต้องใช้คลื่นรูปไซน์ในการทำงานเป็นอย่างมาก หรืออาจพูดอีกนัยหนึ่งได้ว่าระบบการสื่อสารจะทำงานไม่ได้ถ้าขาดแหล่งคลื่นรูปไซน์ มีวงจรหลายชนิดที่ใช้ผลิตคลื่นรูปไซน์เหล่านี้ เช่น วงจรออสซิลเลเตอร์แบบป้อนกลับ (Feedback Oscillator) วงจร RC และวงจร LC ความถี่ที่ผลิตนี้เริ่มตั้งแต่ความถี่เสียงถึงความถี่วีเอชเอฟ (VHF)

2.6.1 วงจรออสซิลเลเตอร์ป้อนกลับแบบบวก

การป้อนกลับแบบบวก (Positive Feedback) เป็นพื้นฐานของวงจรออสซิลเลเตอร์ โดยทั่วไปที่ใช้กันอยู่ดังรูปที่ 2.23 แสดงวงจรรขยายแบบป้อนกลับ (Feedback Amplifier) ภายใต้สถานะอันหนึ่งสามารถทำให้วงจรนี้ มีการป้อนกลับแบบบวกและกลายเป็นวงจรมผลิตความถี่ (Oscillator Circuit) ขึ้นมา ข้อแม้ของวงจรที่จะทำให้เกิดการออสซิลเลตได้นั้นจะต้องมีอัตราขยายในลูปป้อนกลับจะต้องมากกว่า 1 เฟสที่เปลี่ยน 90 องศา (Phase Shift) ในลูปนี้จะต้องเป็นเฟสบวก คูณด้วย 2 เรเดียน หรือ 360 องศา จะต้องเกิดขึ้นในเวลาเดียวกัน จึงจะทำให้เกิดการออสซิลเลตได้

$$\text{อัตราขยายในลูป (Loop Gain)} = |GH| \theta \dots \dots \dots (2.6)$$

กำหนดให้

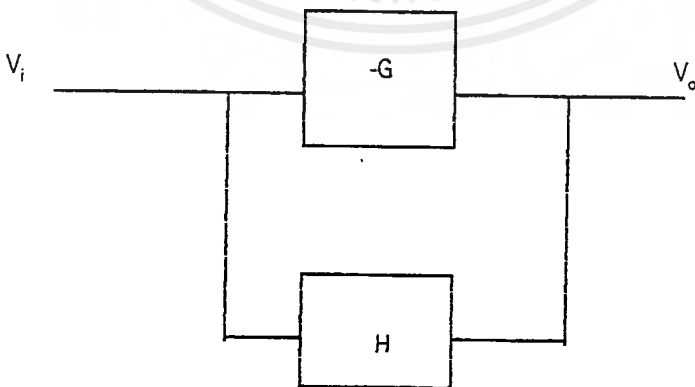
$$n = 0, 1, 2, 3, \dots$$

$$\theta = n360 \text{ องศา}$$

และ $|GH| \geq 1$

วงจรรขยายโดยทั่ว ๆ ไปจะมีเฟสของสัญญาณเอาต์พุตตรงข้ามกับสัญญาณอินพุต 180 องศา และมีอัตราขยายมากกว่า 1 สมมติให้ G ตามรูปที่ 2.23 คือการขยายวงจร และ H คือการป้อนกลับของลูป ดังนั้นอัตราขยายของวงจรภายในลูปนี้คือ ผลคูณของ G และ H และ การขยายทั้งหมดของวงจร (Overall Gain) ที่มีการป้อนกลับคือ

$$A_v = G / (1-GH) \dots \dots \dots (2.7)$$



รูปที่ 2.23 วงจรออสซิลเลเตอร์ที่ใช้การป้อนกลับแบบบวก

กำหนดให้

A_v = การขยายทั้งหมดของวงจร

G = ส่วนการขยายของวงจร

H = ส่วนการขยายของการป้อนกลับ

GH = การขยายของลูป

ค่าที่กำหนดทั้งหมดนี้เป็นค่าคอมเพล็กซ์ (Complex) ซึ่งหมายถึงขนาด และ เฟส (Magnitude and Phase Angle) รวมอยู่ด้วย

ความถี่ของออสซิลเลเตอร์นี้ถูกกำหนดโดยส่วนประกอบของ H ซึ่งทำให้เฟสเปลี่ยนไป 180 องศา การเลือกค่าขององค์ประกอบเหล่านี้ต้องเลือกด้วยความระมัดระวัง เพื่อให้เกิดเฟสชิฟ 180 องศา ที่ความถี่เดียวกันนี้ และ ความถี่นี้ คือ ความถี่ของออสซิลเลเตอร์ที่นำไปใช้ วงจรออสซิลเลเตอร์แบ่งออกได้เป็น 3 ชนิด คือ

1. วงจรออสซิลเลเตอร์ที่ใช้ LC
2. วงจรออสซิลเลเตอร์ที่ใช้ RC
3. วงจรออสซิลเลเตอร์ที่ใช้ก้อนผลึก (Crystal)

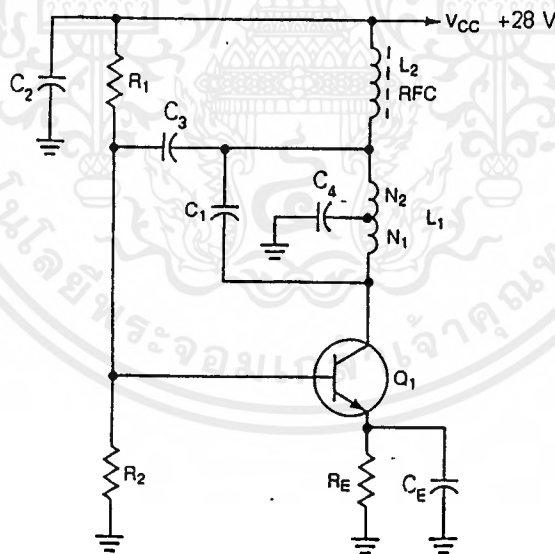
2.7.2 วงจรออสซิลเลเตอร์แบบฮาร์ทลีย์

จุดสังเกตของวงจรแบบนี้อยู่ที่วงจรจูน LC ที่มีการแท็ปกคอยล์สำหรับเป็น วงจรคอยล์ป้อนกลับ (inductive feedback) แทนที่จะเป็นคอยล์ทริกเกอร์แบบแยก จากรูป C_1 และ L_1 ประกอบกันเป็นวงจรจูน การแท็ปลัญญานจากคอยล์ L_1 ที่จุด G ก็เพื่อเป็นการจ่ายแรงดันคอลเลคเตอร์ L_2 ในวงจรคือ RF ช็อค (Choke) จุดแท็ปลัญญาน G จะต่ออยู่กับกราวด์โดยมี C_2 เป็นบายพาสคาปาซิเตอร์อยู่ สัญญาณเอาท์พุทของออสซิลเลเตอร์จะจ่ายออกที่ขาคอลเลคเตอร์ซึ่งมีระดับแรงดันไฟฟ้าเท่ากับ V_{AG} ซึ่งเป็นความต่างศักย์ระหว่างจุด A บนคอยล์ L_1 เทียบกับจุด G ส่วนในด้านตรงข้ามกับจุดแท็ปลัญญานป้อนกลับเท่ากับ V_{BG} ซึ่งถูกคัปปลิ่งโดย C_2 ไปยังขาเบสของ Q_1 การป้อนกลับสัญญาณในลักษณะนี้เป็นแบบบวก เพราะ จะมีความต่างเฟส 180 องศา เมื่อเทียบกับ V_{AG} ซึ่งผลลัพธ์ที่เกิดขึ้นจะก่อให้เกิดการออสซิลเลทผลผลิตสัญญาณไฟสลัปจ่ายออกมาที่เอาท์พุทด้วยความถี่รีโซแนนท์ของวงจร LC

พิจารณาที่ระดับแรงดันไฟตรง V_C มีค่าเท่ากับ 28 V เพราะความต้านทานไฟตรงของคอยล์ RF L_1 และ L_2 มีค่าน้อยมากไม่นำมาคำนวณก็ได้ ซาอิมิเตอร์มีแรงดันไฟไบอัสตนเองเท่ากับ 1 V จาก R_E โดยมี C_E เป็นตัวรักษาเสถียรภาพของการไบอัสแรงดันไฟฟ้าฟอร์เวิร์ดที่ขาเบสจ่ายผ่าน R_1 , R_2 ซึ่งแบ่งมาจากแหล่งจ่ายไฟ +28 V ดังนั้นค่า $V_{BE} = 1.4 - 1.0 = 0.4$ V ซึ่งน้อยกว่าค่าแรงดันไฟฟ้าคัทออฟ 0.5 V แต่ค่าแรงดันยอดด้านบวกของแรงดันไฟฟ้าป้อนกลับจะขับให้ขาเบสมีระดับแรงดัน

ไฟฟ้าเป็นบวก ซึ่งสามารถทำให้ Q_1 นำกระแสไฟฟ้า และเกิดการออสซิลเลตได้ หน้าที่ของอุปกรณ์แต่ละตัวในวงจรรูปที่ 2.24 สามารถสรุปได้ดังนี้

- L_1 : อินดักแตนซ์สำหรับวงจรจูน มีการเทปเพื่อป้องกันสัญญาณกลับ
- C_1 : คาปาซิเตอร์สำหรับวงจรจูน เปลี่ยนแปลงค่าได้
- L_2 : RF ใช้ทำหน้าที่แยกสัญญาณออสซิลเลเตอร์ออกจากแหล่งจ่ายไฟ
- C_4 : บายพาสสำหรับต่อจุดเทปจาก L_1 เพื่อดึงสัญญาณไฟสลับลงกราวนด์
- R_1 : ร่วมกับ R_2 ป้องกันแรงดันไบอัสตรงให้กับขาเบส
- C_2 : คัปปลิ่งสัญญาณป้อนกลับเข้าไปยังขาเบสพร้อมกับ C_3 เป็นวงจร RC คัปปลิ่งสัญญาณร่วมกับ R_1 เป็นวงจรแบ่งแรงดันสำหรับไบอัสขาเบสพร้อมกับ C_2 เป็นวงจรกรองแบบ RC สำหรับแปลงไฟของสัญญาณไบอัส

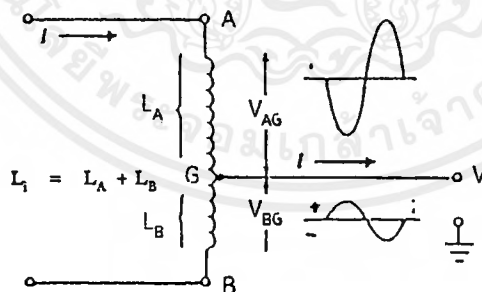


รูปที่ 2.24 วงจรออสซิลเลเตอร์แบบฮาร์ทเลย์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.7.3 การกลับเฟสของสัญญาณแท็ปคอยล์

เหตุผลที่ใช้อธิบายว่าเพราะอะไรการแท็ปสัญญาณของ L_1 จึงให้เกิดการป้อนกลับแบบบวก ก่อนอื่นพิจารณาส่วนของคอยล์ L_1 ซึ่งแบ่งออกได้เป็น 2 ส่วนคือ L_A และ L_B วิเคราะห์การไหลของกระแสอิเล็กทรอนิกส์เข้าไปยังจุด A จะเห็นว่าทิศทางการไหลผ่านคอยล์ L_A ระหว่างจุด A กับจุด G แล้วไหลไปสู่แหล่งจ่ายไฟ V_+ ซึ่งในกรณีนี้คอยล์ L_B จะไม่มีส่วนเกี่ยวข้องกับใด ๆ กับทิศทางการไหลของกระแส แต่อย่างไรก็ตามคอยล์ทั้งสองส่วนก็ต่อเนื่องกันอยู่ ดังนั้น L_B จึงเป็นตัวหม้อแปลงคัปปลิงสัญญาณไปสู่ L_A ได้ ในการแปรผันของแรงดันไฟสลับ สมมติให้ I มีค่าเพิ่มขึ้นของเลขชี้ (Lenz Law) จะได้ว่าเกิดการเหนี่ยวนำด้วยตัวเองขึ้น (Self Induced) เกิดแรงดันไฟฟ้า V_{AG} ซึ่งมีขั้วเป็นลบที่จุด A เพื่อต่อต้านการเพิ่มขึ้นของ I ยิ่งกว่านั้นแรงดันที่เหนี่ยวนำขึ้นมาจะส่งผลให้คอยล์ทั้งหมดมีแรงดันไฟฟ้าเป็นลบ และเนื่องจากลักษณะของการพันคอยล์เป็นแบบในทิศทางเดียวกัน ดังนั้นจึงมีสนามแม่เหล็กเหมือนกันตลอดทั้งคอยล์ จุด A ถือว่าเป็นจุดปลายสุดของแรงดันไฟลบที่เหนี่ยวนำขึ้นมาเมื่อเปรียบเทียบกับจุดอื่น ๆ หรือขดลวดถัดมาด้านล่างตามรูป ส่วนจุด B เมื่อพิจารณาตามแรงดันไฟฟ้าที่เหนี่ยวนำขึ้นมา จุด B จะมีแรงดันเป็นบวกเมื่อเทียบกับขดลวดที่อยู่เหนือขึ้นไปตามรูป ดังนั้นทั้งจุด A และจุด B จึงมีขั้วตรงกันข้ามเสมอ เมื่อเปรียบเทียบกับแท็ป นั่นคือ V_{AG} และ V_{BG} จะมีเฟสของสัญญาณต่างกัน 180 องศาเสมอ ในขณะที่จุดหนึ่งเป็นลบมากที่สุดอีกจุดหนึ่งก็จะมีเฟสเป็นบวกมากที่สุด เนื่องจากจุดแท็ป G ต่ออยู่กับกราวด์ เพราะฉะนั้น V_{AG} และ V_{BG} จึงเป็นสัญญาณกลับที่มีขั้วตรงกันข้ามกันเสมอเมื่อเทียบกับจุดกราวด์ อ้างถึงวงจรรูปที่ 2.25 V_{AG} จะถูกป้อนกลับแบบบวกไปเข้ายังอินพุทของขาเบส โดยทั่ว ๆ ไปจุดแท็ปจะป้อนกลับแรงดันไฟฟ้าประมาณ 1 ใน 3 ของแรงดันไฟฟ้าที่ตกคร่อมคอยล์ทั้งหมด



รูปที่ 2.25 เฟสของแรงดันไฟฟ้าในการแท็ปคอยล์ L_B จะต่างเฟส 180 องศา
เทียบกับแรงดันไฟฟ้าในคอยล์ L_A

2.8 หลักการส่งข้อมูลสลับสายอัตโนมัติ

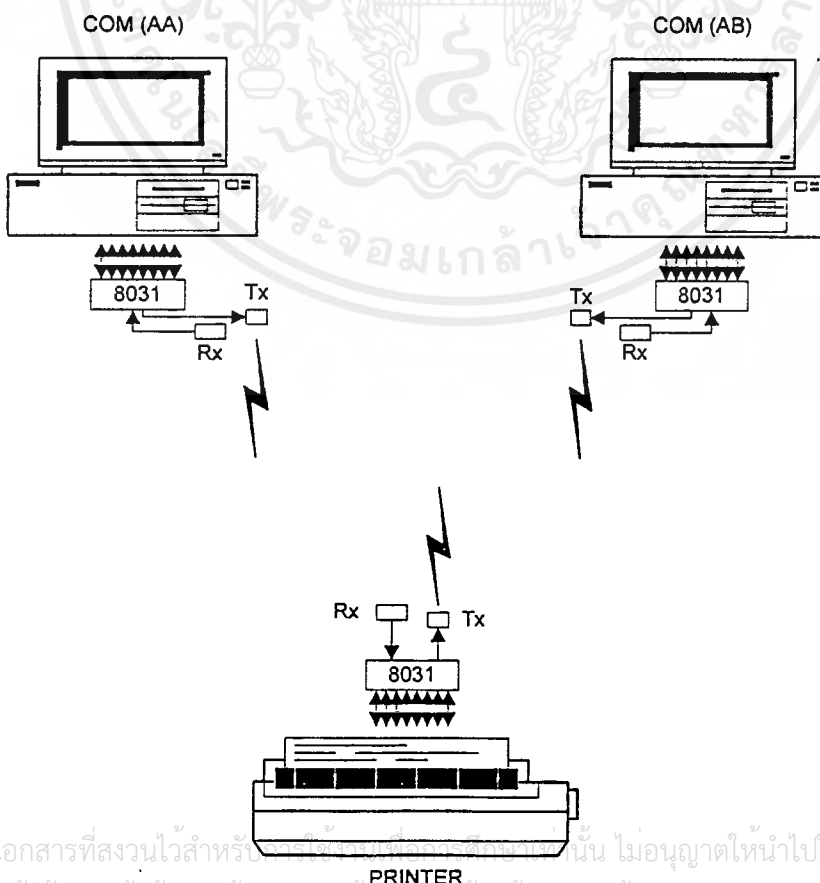
การติดต่อระหว่างเครื่องคอมพิวเตอร์หลายเครื่องกับพรินเตอร์เครื่องเดียวนั้นจะใช้หลักการโพลลิง (Polling) โดยที่เครื่องพรินเตอร์จะทำหน้าที่เป็นสถานีปฐมภูมิ (Primary) เป็นตัวส่งการร้องขอข้อมูลจากเครื่องคอมพิวเตอร์ซึ่งก็คือ สถานีทุติยภูมิ (Secondary) ในการติดต่อระหว่างเครื่องคอมพิวเตอร์กับพรินเตอร์นั้นจะต้องกำหนดหมายเลขสถานีให้แก่เครื่องคอมพิวเตอร์แต่ละเครื่อง

เครื่องพีวเตอร์เครื่องที่ 1 หมายเลขสถานีคือ AA

เครื่องพีวเตอร์เครื่องที่ 2 หมายเลขสถานีคือ AB

และในการติดต่อจะใช้การติดต่อในลักษณะ 9 บิต โดยที่ 8 บิต เป็นข้อมูลและอีก 1 บิต ใช้เป็นบิตควบคุมเพื่อแสดงว่าเป็นคำสั่งหรือข้อมูล คือถ้าเป็น “ 1 ” แสดงว่าเป็นคำสั่ง และถ้าเป็น “ 0 ” แสดงว่าเป็นข้อมูล

ในสภาวะปกติเครื่องสลับสายจะส่งสัญญาณบิตที่เป็นสัญญาณลวงคือ “0” ไปให้แก่เครื่องคอมพิวเตอร์ทั้ง 2 เครื่อง ในการเริ่มต้นการติดต่อเครื่องพรินเตอร์จะเป็นฝ่ายร้องขอโดยส่ง “ 1AA ” ไปยังเครื่องคอมพิวเตอร์เครื่องที่ 1 หากเครื่องคอมพิวเตอร์เครื่องที่ 1 นี้มีข้อมูลต้องการพิมพ์ มันจะส่งข้อมูลเหล่านี้มาให้เครื่องพรินเตอร์ โดยส่งบิตควบคุมเป็น 0 หลังจากการติดต่อสิ้นสุดลง เครื่องพรินเตอร์จะร้องขอไปยังเครื่องคอมพิวเตอร์เครื่องที่ 2 ต่อไป โดยส่ง “ 1AB ” ไป หากเครื่องคอมพิวเตอร์เครื่องที่ 2 นี้มีข้อมูลต้องการจะพิมพ์ก็จะทำเช่นเดียวกัน แต่ถ้าหากไม่มีข้อมูลจะส่งพิมพ์ เครื่องพรินเตอร์ก็จะทำการร้องขอสถานีถัดไปเรื่อย ๆ จนครบทุกสถานีจึงกลับมาร้องขอจากสถานีที่ 1 ใหม่อีกรอบ เป็นเช่นนี้ตลอดไป แต่ถ้าเครื่องพรินเตอร์ไม่ว่างขึ้นมากก็จะทำการส่ง “ 1BB ” ไปให้เครื่องคอมพิวเตอร์ทุกเครื่อง



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับใช้เฉพาะภายในเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 2.26 แสดงหลักการโพลลิงระหว่างเครื่องคอมพิวเตอร์กับพรินเตอร์

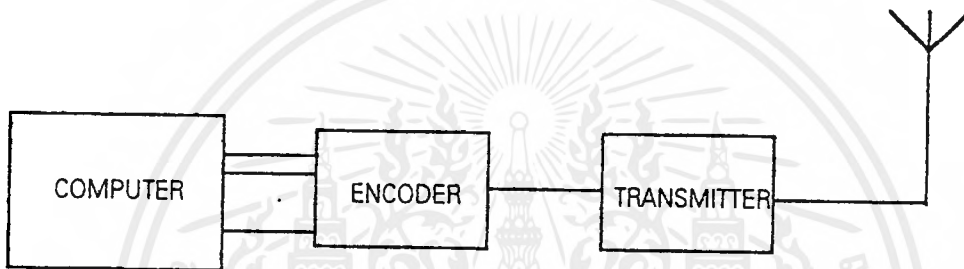
บทที่ 3

วงจรการทำงานของพรีนเตอร์ไร้สาย

3.1 ภาคส่ง

3.1.1 รายละเอียดของบล็อกไดอะแกรมทางภาคส่ง

จากรูปที่ 3.1 จะเป็นบล็อกไดอะแกรมของทางภาคส่ง ซึ่งส่วนแรกเป็นบล็อกของสัญญาณข้อมูลแบบขนาน ซึ่งส่งออกมาจากตัวคอมพิวเตอร์ ส่วนที่สองจะเป็นส่วนการเข้ารหัสซึ่งแปลงจากสัญญาณแบบขนานเป็นสัญญาณแบบอนุกรมออกมาให้เป็นแบบอะซิงโครนัล โดยใช้ซีพียู 8031 เป็นหัวใจหลักในการจัดการส่วนนี้



รูปที่ 3.1 แสดงบล็อกไดอะแกรมของภาคส่ง

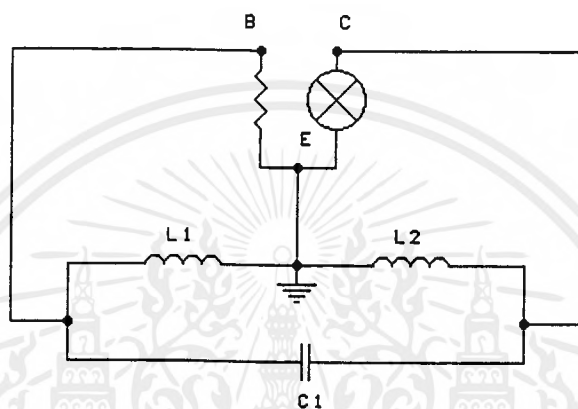
3.1.2 ส่วน HARDWARE

การทำงานของเครื่องส่ง

ในขณะที่คอมพิวเตอร์ส่งสถานะที่เป็นลอจิก 0 แล้วผ่านวงจรรีจิสเตอร์ เป็นลอจิก 1 มากจะทำให้ทรานซิสเตอร์เบอร์ C9013 ทำงาน เนื่องจากมีระดับแรงดันไบอัสสูงพอ ซึ่งทำหน้าที่เสมือนเป็น สวิตช์เปิดให้ทรานซิสเตอร์เบอร์ MPSH 10 ทำงานเกิดการออสซิลเลชัน เนื่องจากการเกิดการครบวงจรขึ้น เพราะเปรียบเสมือนขาคีมิตเตอร์ของทรานซิสเตอร์เบอร์ MPSH 10 ถูกต่อลงกราวด์ ดังนั้นจึงเกิดการออสซิลเลชันได้ที่ย่านความถี่เรโซแนนท์ ย่านยูเอชเอฟ ประมาณ 500 MHz และมีขนาด (Amplitude) สูงสุดที่ความถี่ 549 MHz โดยการจูนที่ C_1 แล้วแพร่กระจายคลื่นออกไปจาก C_1 ดูจาก รูปที่ 3.2 การออกแบบวงจรในภาคส่งนี้จะใช้หลักการจากวงจรออสซิลเลเตอร์แบบฮาร์ทลีย์ จุดสังเกตของวงจรแบบนี้จะอยู่ที่วงจรรูน LC ซึ่งการที่ปคอยล์ เป็น วงจรคอยล์ป้อนกลับ (Inductive Feedback) จากรูปวงจรจะเห็นว่าที่แท็ปคอยล์ที่จุด G ก็เพื่อจ่ายแรงดันคอลเลคเตอร์ L_2 ในวงจร เป็น RF ใช้ค ทำหน้าที่แยกสัญญาณออสซิลเลเตอร์ออกจากแหล่งจ่ายไฟ สัญญาณเอาท์พุทของออสซิลเลเตอร์ จะจ่ายออกที่ขาคอลเลคเตอร์ ซึ่งมีระดับแรงดันไฟฟ้า

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เท่ากับ V_{AG} ซึ่งเป็นความต่างศักย์ระหว่างจุด A บนคอยล์ L_1 เทียบกับ จุด G ส่วนในด้านตรงกันข้ามกับจุด
 เทียบ แรงดันไฟสลับบ็อนกลับ เท่ากับ V_{BG} ซึ่งถูกคัปปลิงโดย C_2 ไปยังขาเบสของ Q_1 การบ็อนกลับ
 สัญญาณนี้เป็นบวก เพราะมีความต่างเฟส 180 องศา เมื่อเทียบกับ V_{AG} ซึ่งผลลัพธ์ที่เกิดขึ้นก่อให้เกิดการ
 ออกซิเลทผลผลิตสัญญาณไฟสลับจ่ายออกมาที่เอาต์พุท ด้วยความถี่รีโซแนนท์ของวงจร LC ซึ่งคำนวณได้
 จากสูตรดังต่อไปนี้



$$r_{in} = \frac{v_{be}}{i_b} = \frac{(\beta + 1)v_{be}}{i_e} = (\beta + 1)r'_e = h_{ie}$$

$$(i_b = \frac{i_e}{\beta + 1})$$

$$i_b = \frac{j\omega L_1}{h_{ie} - j\omega L_1} * i_L$$

$$i_b = \frac{1}{\frac{h_{ie}}{j\omega L_1} - 1} * i_L$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
 ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$$i_L = \left(\frac{h_{ie}}{j\omega L_1} - 1 \right) * i_b$$

$$i_L = \frac{j\omega L_2}{j\omega L_2 + \left(\frac{-j}{\omega c_1} \right) + \frac{h_{ie}(j\omega L_1)}{h_{ie} + j\omega L_1}} * i_2$$

$$\left(\frac{h_{ie}}{j\omega L_1} - 1 \right) i_b = \frac{j\omega L_2 \cdot \omega c_1 \cdot (h_{ie} + j\omega L_1)}{h_{ie} + j\omega L_1} * i_2$$

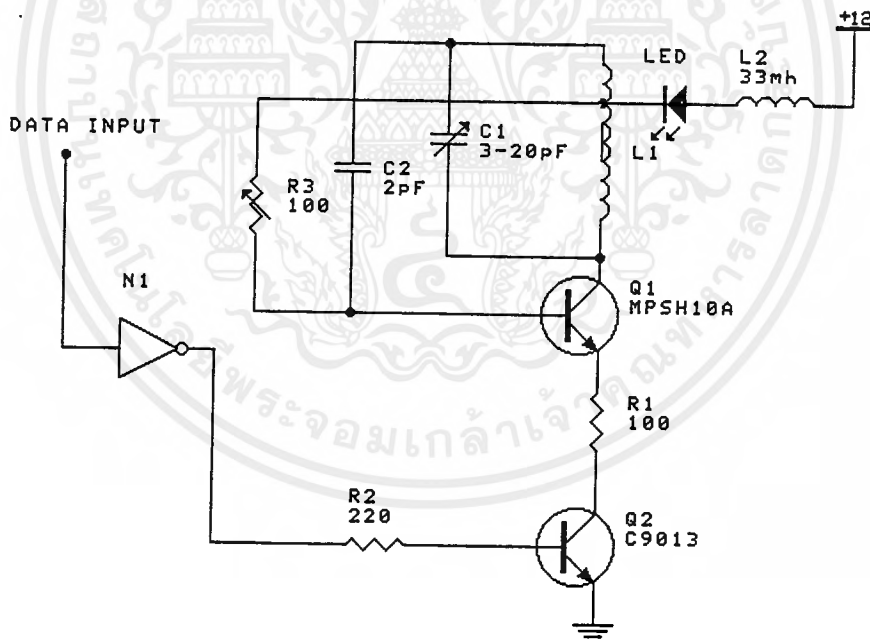
$$\frac{i_c}{i_b} = \left(\frac{j\omega^2 c_1 L_2 - j}{\omega^3 L_1 L_2 c_1} (h_{ie} + j\omega L_1) + h_{ie} (j\omega^2 L_1 c_1) \right)$$

$$\frac{i_c}{i_b} = \frac{h_{ie} j\omega^2 c_1 L_2 - \omega^3 L_1^2 c_1 - j h_{ie} + \omega L_1 h_{ie} + j\omega^2 L_1 c_1}{\omega^3 L_1 L_2 c_1}$$

$$h_{ie} \omega^2 c_1 L_2 - h_{ie} + h_{ie} \omega^2 L_1 c_1 = 0$$

$$\omega = \frac{1}{\sqrt{c_1 (L_1 + L_2)}}$$

$$f = \frac{1}{2\pi \sqrt{c_1 (L_1 + L_2)}}$$



รูปที่ 3.2 วงจรเครื่องส่ง

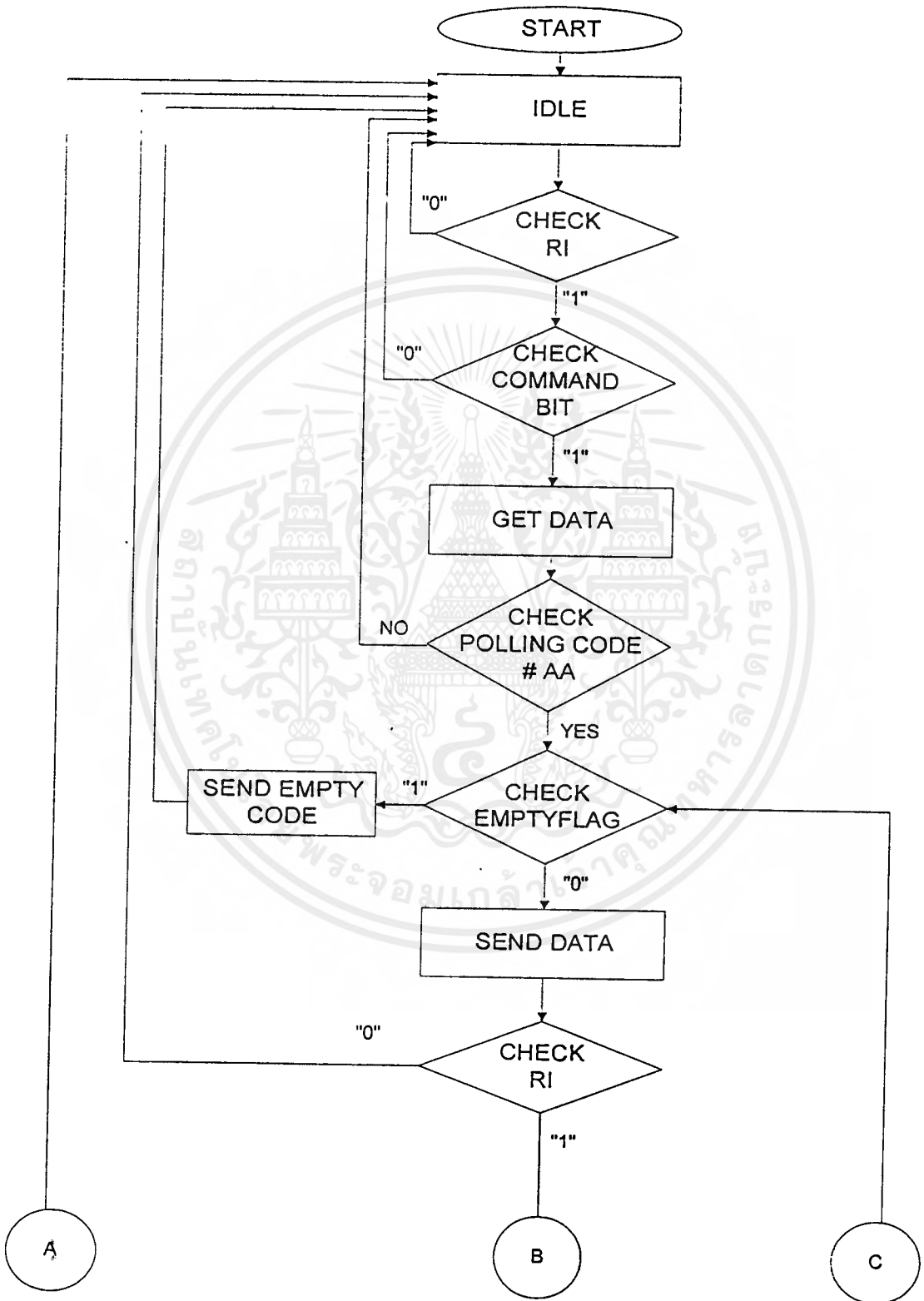
จากการออกแบบจะเห็นว่าค่า L น้อยมาก ดังนั้นเราจึงใช้ลวดทองแดงบนแผ่นปริ้นท์ทำให้เป็นค่า L ตัวนี้ แล้วใช้ C ปรับค่าได้จนเพื่อให้ได้ความถี่ที่ต้องการ

ความถี่ที่ใช้งาน จะเห็นว่า เราใช้กำลังส่งไม่มากนัก เราจึงเลือกใช้ทรานซิสเตอร์เบอร์ MPSH 10 ซึ่งมีความถี่ที่ใช้งานที่ 472.50 MHz ซึ่งเหมาะสมที่จะใช้งานมากที่สุด

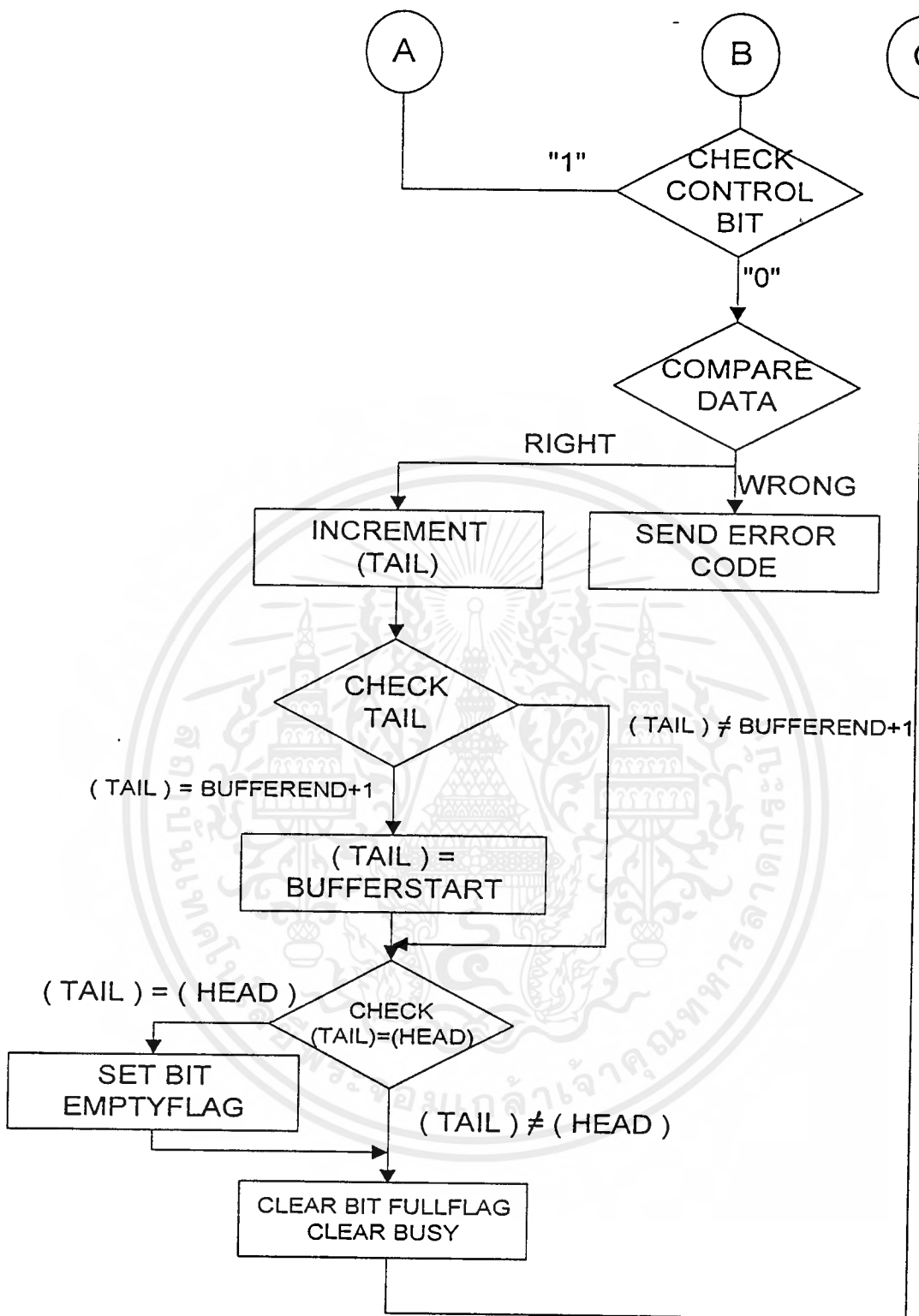
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.1.3 ส่วน SOFTWARE

ไฟลชาร์ตแสดงการทำงานทางด้านส่งของเครื่องคอมพิวเตอร์เครื่องที่ 1

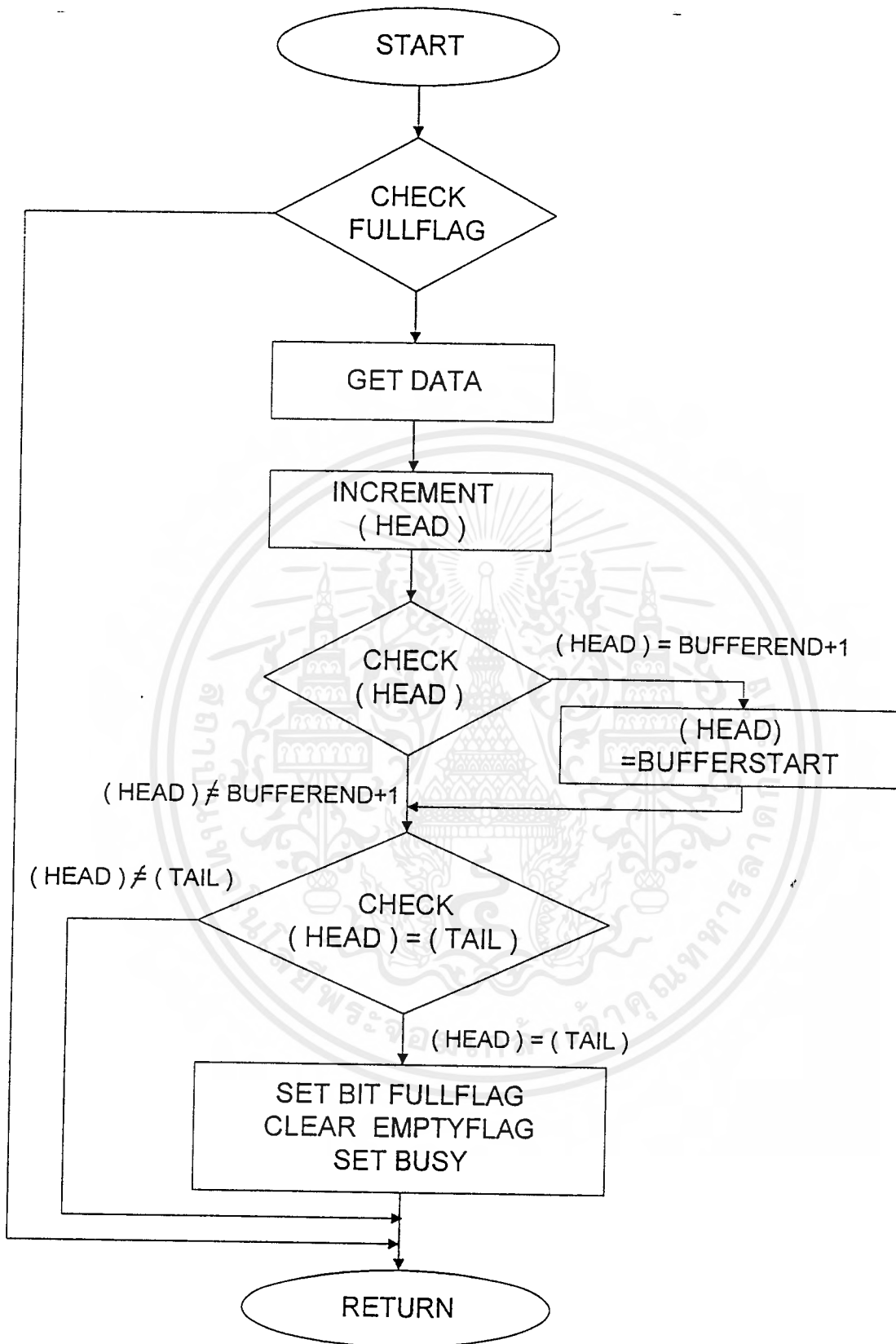


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



COMPUTER 1 PROGRAM

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



INTERRUPT PROGRAM (1)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โปรแกรมทางด้านส่ง เครื่องคอมพิวเตอร์เครื่องที่ 1

```

PCA EQU 4000H
PCB EQU 4001H
PCC EQU 4002H
PCWC EQU 4003H

```

```

TIBUAD9600 EQU 0FDH
BO EQU 21H
BUFFST EQU 22H
BUFFEND EQU 4FH
EMPTYFLAG EQU 0H
FULLFLAG EQU 1H

```

```

;
ORG 0000H
SJMP START
ORG 0003H ; INT DATA SERVICE INTO
LCALL GETPARALLEL
RETI
;

```

```

ORG 0050H
START: MOV R2,#0H
ST: MOV R3,#0H
DJNZ R3,$
DJNZ R2,ST
MOV DPTR,#PCWC
MOV A,#10000010B ; MODE 0 A OUT B IN C UP,LOW OUT
MOVX @DPTR,A
LCALL DELAY
MOV DPTR,#PCC

```

เอกสารนี้เป็นเอกสารสงวนไว้สำหรับงานเพื่อ; SET BUSY เท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

MOVX    @DPTR,A
LCALL   DELAY

MOV     SP,#07H
MOV     TL1,#T1BUAD9600
MOV     TH1,#T1BUAD9600
MOV     SCON,#11010000B           ; MODE 3 , REN = 1
MOV     TMOD,#10100000B         ; TIMER 1 MODE 2 ( AUTORELOAD )
SETB    TR1                       ; START TIMER 1
MOV     IP,#1                     ; HI PRIORITY - EXT0
MOV     IE,#81H                  ; ENABLE EXT0
MOV     R0,#BUFFST               ; R0 * HEAD *
MOV     R1,#BUFFST               ; R1 * TAIL *
SETB    EMPTYFLAG
CLR     FULLFLAG
MOV     B0,#00H
MOV     DPTR,#PCC
CLR     A                          ; CLEAR BUSY
MOVX    @DPTR,A
CLR     RB8
CLR     TB8
CLR     TI
CLR     RI
; WAIT FOR * 1AA * ( POLLING CODE )

IDLE:   JNB     RI,IDLE              ; IF NO DATA , WAIT
CLR     RI

ID1:    JNB     RB8,IDLE            ; IF BIT #8 NOT = 1 GOTO IDLE STATE
CLR     RB8
; YES , CHECK IF POLL
MOV     A,SBUF
CJNE    A,#0AAH,IDLE              ; NOT A POLLNG CODE ,IDLE
; YES , DO SEND DATA TO HOST ( PRINTER )

```

```

SEND:      ;IF THERE 'S NOT DATA TO SEND ,SEND 100H
           ;ELSE ,SEND DATA IN BUFFER
           JNB      EMPTYFLAG,SEND1
           ; REPLY WITH EMPTY ( 100H )
           SETB     TB8                ; SET 8TH BIT
           CLR      A
           CLR      TI
           MOV      SBUF,A            ; SEND 100H
           JNB      TI,$
           CLR      TI
           CLR      TB8
           MOV      P1,#0
           SJMP     IDLE
SEND1:     MOV      A,@R1             ; SEND DATA
           CLR      TB8
           CLR      TI
           MOV      SBUF,A
           JNB      TI,$             ; WAIT FOR END OF TX
           CLR      TI
WAIT:      MOV      B0,#0
WAIT1:     JB       RI,WAIT2
           NOP
           NOP
           NOP
           NOP
           NOP
           NOP
           NOP
           DJNZ     B0,WAIT1         ; COUNT DOWN ( APPROX 2.5 mS )
           ; TIME 'S UP ,NOT REPLIED , IDLE
           SJMP     IDLE
WAIT2:     CLR      RI
           JB       RB8,WAIT3
           MOV      B,SBUF

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับบริการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

MOV     A,@R1
CJNE   A,B,ERROR      ; COMPARE SENT DATA WITH RECEIVED DATA
                                ; EQUAL ,SEND NEXT BYTE
                                ; ELSE ,SEND ERROR COMMAND
; NEXT
INC     R1              ; ADVANCE TAIL POINTER
CJNE   R1,#BUFFEND+1,CHKT ; CHECK OVERFLOW
MOV     R1,#BUFFST     ; WRAP
CHKT:  MOV     A,R0      ; CHECK TAIL = HEAD ?
MOV     B,R1
CJNE   A,B,CHKT1
SETB   EMPTYFLAG
CHKT1: CLR     FULLFLAG
MOV     DPTR,#PCC      ; CLEAR BUSY
CLR     A
MOVX   @DPTR,A
SJMP   SEND
WAIT3: CLR     RB8
SJMP   IDLE
ERROR: ; SEND COMMAND * ERROR * ( 1FFH )
SETB   TB8
CLR     TI
MOV     SBUF,#0FFH
JNB    TI,$
CLR     TI
CLR     TB8
SJMP   SEND

```

 INTO PROGRAM

```

GETPARALLEL:  PUSH      ACC
               PUSH      B
               PUSH      DPH
               PUSH      DPL
               PUSH      PSW
               JB         FULLFLAG,RETURN      ; FULLFLAG =1 RETURN , =0 GET DATA
               MOV       DPTR,#PCB           ; GET DATA
               MOVX      A,@DPTR
               CLR       EMPTYFLAG
               MOV       @R0,A
               INC       R0                  ; INC HEAD
               CJNE     R0,#BUFFEND+1,CHKH    ; CHECK HEAD
               MOV       R0,#BUFFST
CHKH:          MOV       A,R0                ;CHECK HEAD = TAIL ?
               MOV       B,R1
               CJNE     A,B,RETURN
               MOV       DPTR,#PCC          ; SET BUSY
               MOV       A,#0FFH
               MOVX     @DPTR,A
               SETB     FULLFLAG
RETURN:        POP      PSW
               POP      DPL
               POP      DPH
               POP      B
               POP      ACC
               RET
  
```

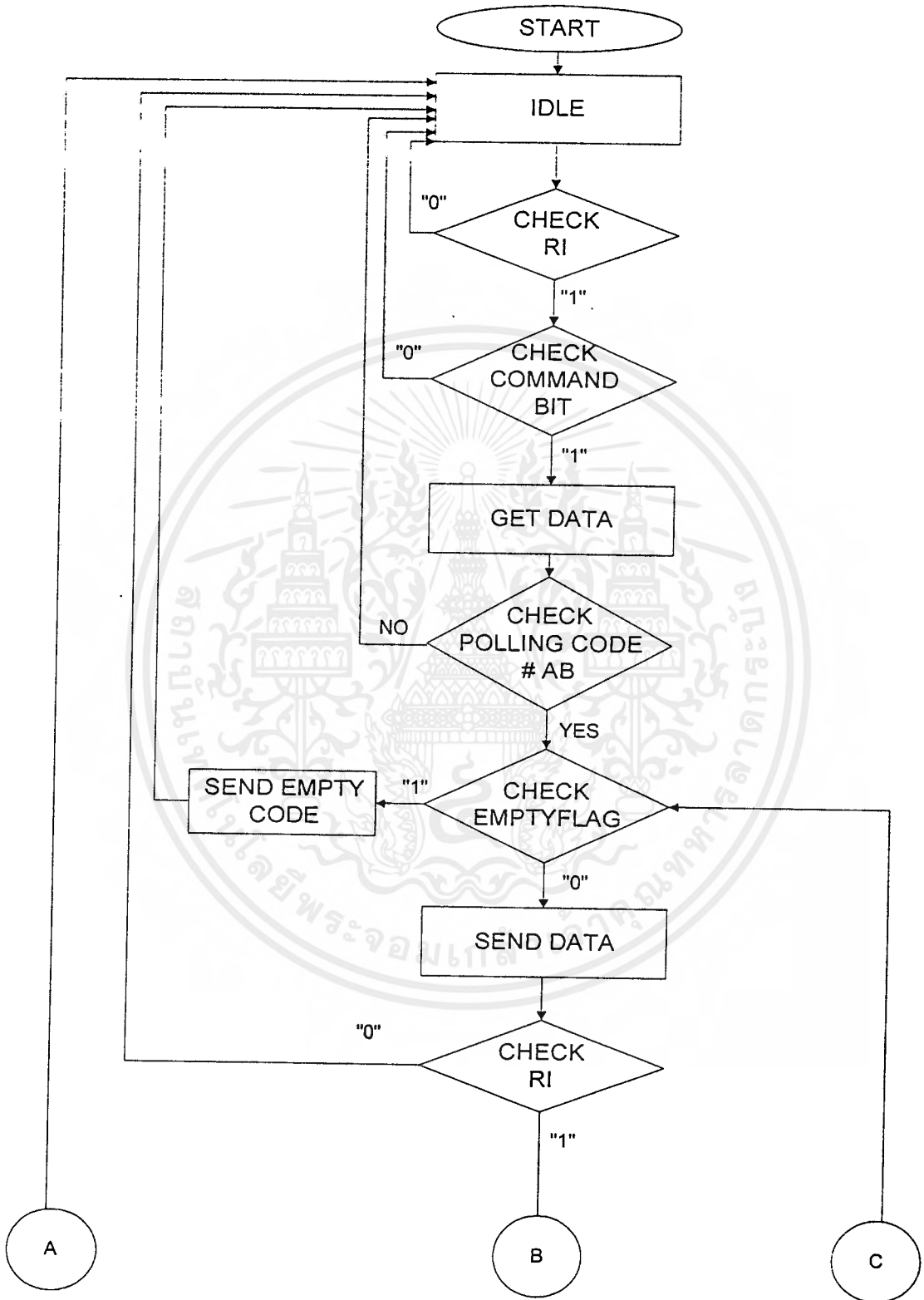
```

DELAY:        MOV       R2,#10H
RES:          MOV       R3,#10H
               DJNZ     R3,$
               DJNZ     R2,RES
               RET
  
```

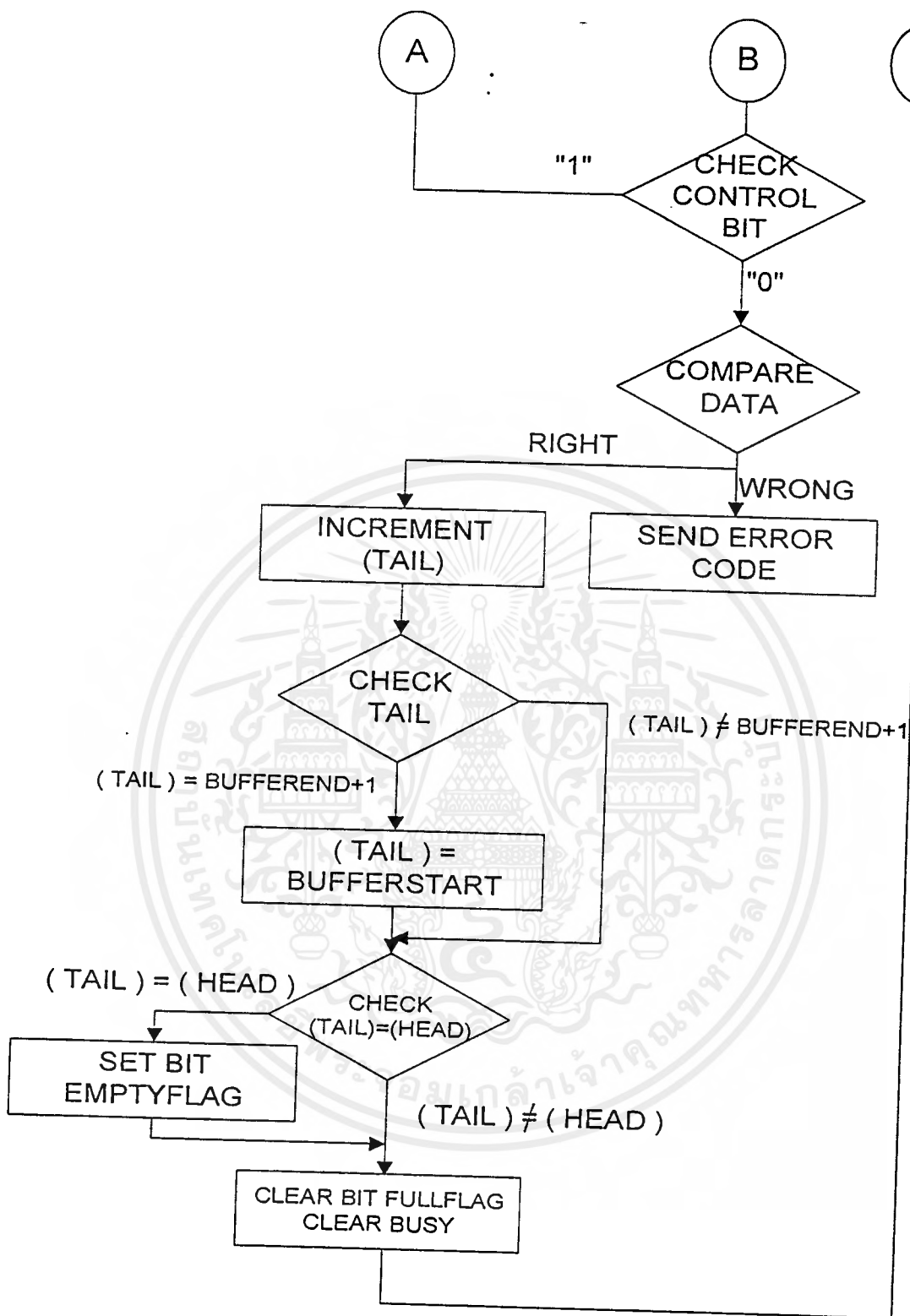
END

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โฟลชาร์ตแสดงการทำงานทางด้านส่งของเครื่องคอมพิวเตอร์เครื่องที่ 2

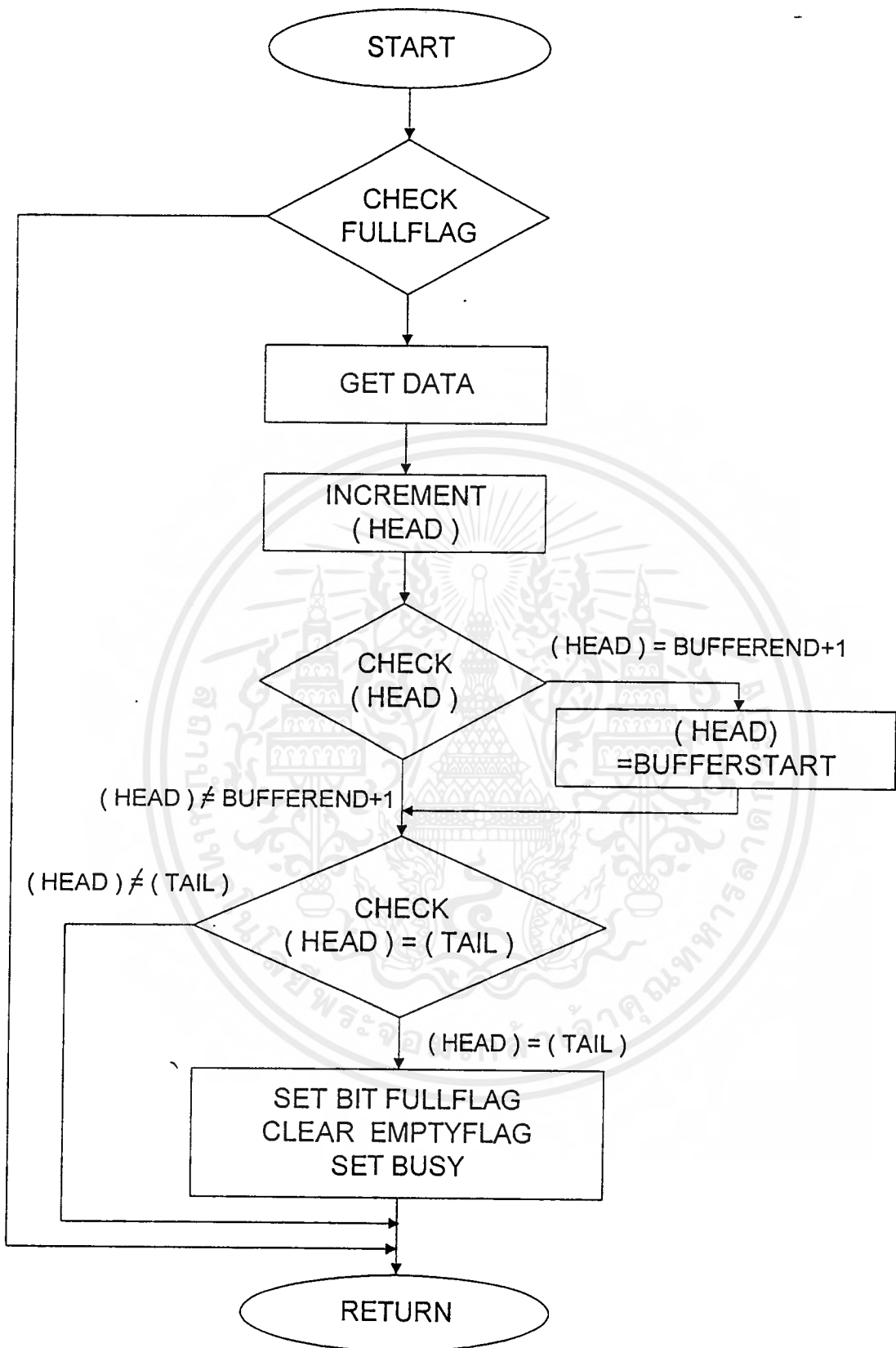


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



COMPUTER 2 PROGRAM

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



INTERRUPT PROGRAM (2)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โปรแกรมทางด้านส่ง เครื่องคอมพิวเตอร์เครื่องที่ 2

```

PCA EQU 4000H
PCB EQU 4001H
PCC EQU 4002H
PCWC EQU 4003H

```

```

TIBUAD9600 EQU 0FDH
B0 EQU 21H
BUFFST EQU 22H
BUFFEND EQU 4FH
EMPTYFLAG EQU 0H
FULLFLAG EQU 1H

```

```

ORG 0000H
SJMP START
ORG 0003H ; INT DATA SERVICE INTO
LCALL GETPARALLEL
RETI

```

```

ORG 0050H
START: MOV R2,#0H
ST: MOV R3,#0H
DJNZ R3,$
DJNZ R2,ST

MOV DPTR,#PCWC
MOV A,#10000010B ; MODE 0 A OUT B IN C UP,LOW OUT
MOVX @DPTR,A
LCALL DELAY

MOV DPTR,#PCC
MOV A,#0FFH ; SET BUSY

```

เอกสารนี้เป็นเอกสารสงวนไว้สำหรับใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

MOVX    @DPTR,A
LCALL   DELAY

MOV     SP,#07H
MOV     TL1,#T1BUAD9600
MOV     TH1,#T1BUAD9600
MOV     SCON,#11010000B           ; MODE 3 , REN = 1
MOV     TMOD,#10100000B          ; TIMER 1 MODE 2 ( AUTORELOAD )
SETB    TR1                       ; START TIMER 1
MOV     IP,#1                     ; HI PRIORITY - EXT0
MOV     IE,#81H                   ; ENABLE EXT0
MOV     R0,#BUFFST                ; R0 * HEAD *
MOV     R1,#BUFFST                ; R1 * TAIL *
SETB    EMPTYFLAG
CLR     FULLFLAG
MOV     B0,#00H

MOV     DPTR,#PCC
CLR     A                          ; CLEAR BUSY
MOVX    @DPTR,A
CLR     RB8
CLR     TB8
CLR     TI
CLR     RI

; WAIT FOR ' 1AB ' ( POLLING CODE )

```

```

IDLE:   JNB     RI,IDLE              ; IF NO DATA , WAIT
        CLR     RI
ID1:    JNB     RB8,IDLE            ; IF BIT #8 NOT = 1 GOTO IDLE STATE
        CLR     RB8
        ; YES , CHECK IF POLL
        MOV     A,SBUF
        CJNE    A,#0ABH,IDLE       ; NOT A POLLNG CODE ,IDLE

```

เอกสารนี้เป็นเอกสารของมหาวิทยาลัยเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
 YES: DO SEND DATA TO HOST (PRINTER) เท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
 ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

SEND:      ;IF THERE 'S NOT DATA TO SEND ,SEND 100H
           ;ELSE ,SEND DATA IN BUFFER
           JNB      EMPTYFLAG,SEND1
           ; REPLY WITH EMPTY ( 100H )
           SETB     TB8                ; SET 8TH BIT
           CLR      A
           CLR      TI
           MOV      SBUF,A            ; SEND 100H
           JNB      TI,$
           CLR      TI
           CLR      TB8
           MOV      P1,#0
           SJMP     IDLE
SEND1:     MOV      A,@R1            ; SEND DATA
           CLR      TB8
           CLR      TI
           MOV      SBUF,A
           JNB      TI,$            ; WAIT FOR END OF TX
           CLR      TI
WAIT:      MOV      B0,#0
WAIT1:     JB       RI,WAIT2
           NOP
           NOP
           NOP
           NOP
           NOP
           NOP
           NOP
           DJNZ     B0,WAIT1        ; COUNT DOWN ( APPROX 2.5 mS )
           ; TIME 'S UP ,NOT REPLIED , IDLE
           SJMP     IDLE
WAIT2:     CLR      RI
           JB       RB8,WAIT3
           MOV      B,SBUF

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

MOV      A,@R1
CJNE    A,B,ERROR      ; COMPARE SENT DATA WITH RECEIVED DATA
                                ; EQUAL ,SEND NEXT BYTE
                                ; ELSE ,SEND ERROR COMMAND
; NEXT
INC      R1              ; ADVANCE TAIL POINTER
CJNE    R1,#BUFFEND+1,CHKT ; CHECK OVERFLOW
MOV      R1,#BUFFST     ; WRAP
CHKT:   MOV      A,R0    ; CHECK TAIL = HEAD ?
        MOV      B,R1
        CJNE    A,B,CHKT1
        SETB    EMPTYFLAG
CHKT1:  CLR      FULLFLAG
        MOV      DPTR,#PCC ; CLEAR BUSY
        CLR     A
        MOVX   @DPTR,A
        SJMP   SEND
WAIT3:  CLR      RB8
        SJMP   IDLE
ERROR:  ; SEND COMMAND * ERROR * ( 1FFH )
        SETB   TB8
        CLR    TI
        MOV    SBUF,#0FFH
        JNB   TI,$
        CLR   TI
        CLR   TB8
        SJMP  SEND

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

 INTO PROGRAM

```

GETPARALLEL:  PUSH      ACC
               PUSH      B
               PUSH      DPH
               PUSH      DPL
               PUSH      PSW
               JB         FULLFLAG,RETURN      ; FULLFLAG =1,RETURN , =0 GET DATA
               MOV       DPTR,#PCB           ; GET DATA
               MOVX      A,@DPTR
               CLR       EMPTYFLAG
               MOV       @R0,A
               INC       R0                  ; INC HEAD
               CJNE      R0,#BUFFEND+1,CHKH   ; CHECK HEAD
               MOV       R0,#BUFFST
CHKH:          MOV       A,R0                ;CHECK HEAD = TAIL ?
               MOV       B,R1
               CJNE      A,B,RETURN
               MOV       DPTR,#PCC           ; SET BUSY
               MOV       A,#0FFH
               MOVX      @DPTR,A
               SETB      FULLFLAG
RETURN:        POP       PSW
               POP       DPL
               POP       DPH
               POP       B
               POP       ACC
               RET
  
```

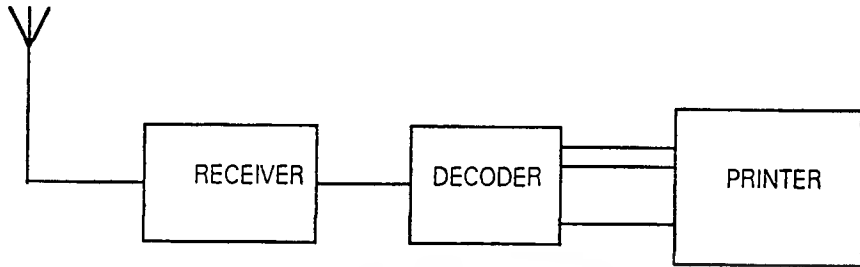
```

DELAY:        MOV       R2,#10H
RES:          MOV       R3,#10H
               DJNZ      R3,$
               DJNZ      R2,RES
               RET
  
```

เอกสารนี้เป็นเอกสารสงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.2 ภาครับ

3.2.1 รายละเอียดของบล็อกไดอะแกรมทางด้านรับ

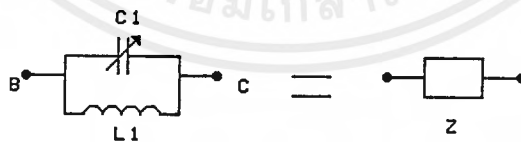


รูปที่ 3.3 แสดงบล็อกไดอะแกรมทางภาครับ

3.2.2 ส่วน HARDWARE

สัญญาณความถี่สูง ย่านยูเอชเอฟ ที่ส่งแพร่กระจายมาในอากาศจากทางภาคส่งนั้น ทางภาครับ จะสามารถรับสัญญาณความถี่นี้ได้ เนื่องจากภาครับมีวงจรจูนที่ดึงความถี่ที่ส่งมา โดยที่ ทรานซิสเตอร์ Q_2 จะเป็นภาคฟรอนเอนด์ (Front End) เพื่อขยายสัญญาณให้แรงขึ้น จากนั้นก็ส่งต่อไปให้ทรานซิสเตอร์ Q_1 ขยายความถี่ที่จูนเอาไว้ โดยวงจรทั้งคู่จะขยายเฉพาะความถี่ที่จูนไว้เท่านั้น

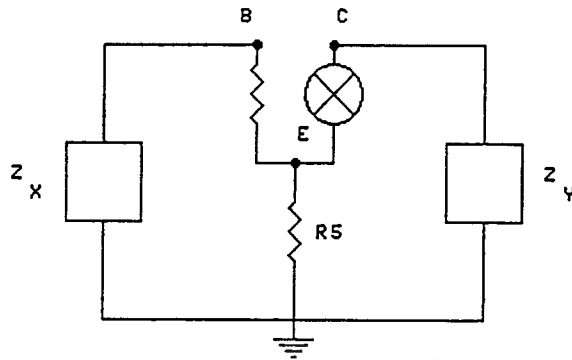
เราสามารถคำนวณหาทรานเฟอร์ฟังก์ชัน (Transfer function) ของทั้งสองวงจรได้ ดังนี้



$$Z = X_c // X_L$$

$$= \frac{wL}{j(w^2 Lc - 1)}$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



$$Z_1 = \frac{Z}{1-A}$$

$$Z_2 = \frac{Z}{1-\frac{1}{A}}$$

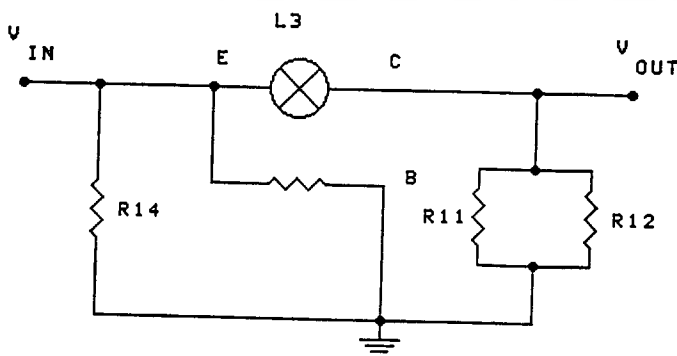
$$Z_X = Z_1 // R_A // R_4$$

$$Z_Y = Z_C // Z_2$$

$$\text{node B: } \left(\frac{1}{Z_X} + \frac{1}{(\beta+1)r'_{be}} \right) V_B - \frac{1}{(\beta+1)r'_{be}} V_E = 0$$

$$\text{node E: } \frac{-1}{(\beta+1)r'_{be}} V_B + \left(\frac{1}{(\beta+1)r'_{be}} + \frac{1}{R_5} \right) V_E = i_B \beta$$

$$\text{node C: } \frac{1}{Z_Y} V_C = -i_B \beta$$



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$$R_A = R_{11} // R_{12}$$

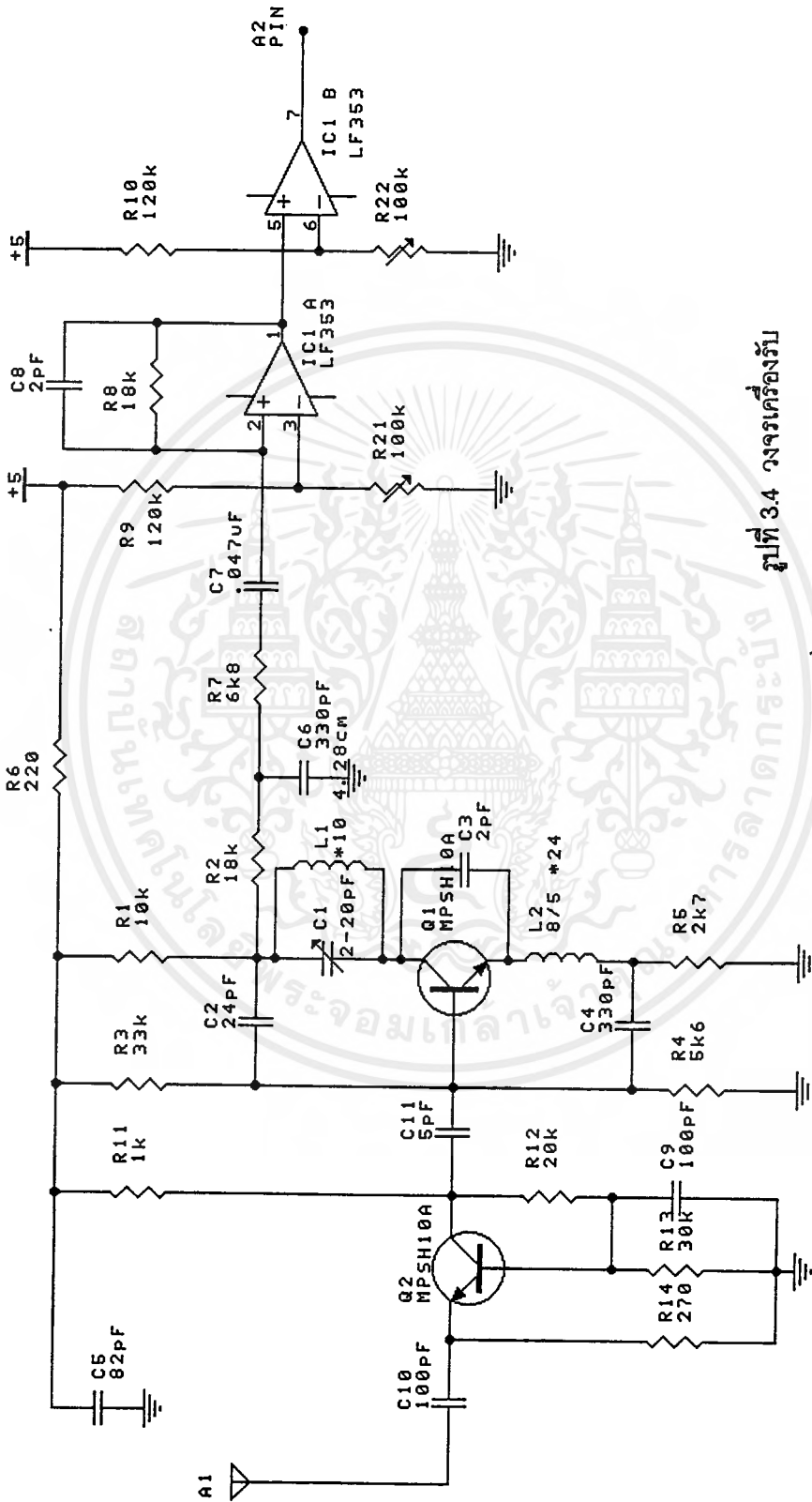
$$(\beta + 1)i_B = i_c$$

$$\frac{V_{in}}{r_{bb'} + r_{b'e}} \cdot (\beta + 1) \cdot (R_{11} // R_{12})$$

$$\frac{V_o}{V_{in}} = \frac{(\beta + 1) \cdot R_A}{r_{bb'} + r_{b'e}}$$

จากนั้นก็จะผ่านวงจร ดิฟเฟอเรนเชียล คอมพาราเตอร์ และที่วงจรคอมพาราเตอร์ จาก ออปแอมป์ เบอร์ LF 353 จะได้เอาท์พุทลอจิก 1 ที่แรงดัน 12 โวลท์ และ ลอจิก 0 ที่ แรงดัน 1.3 โวลท์ ที่สัญญาณตรงนี้ยังไม่สามารถนำไปต่อกับภาคถอดรหัส (Decoder) ได้ จึงต้องผ่านวงจรปรับระดับแรงดัน ให้ได้ลอจิก 1 ที่ 5 โวลท์ และ ที่ลอจิก 0 ที่ 0 โวลท์ และ เป็นวงจรอินเวอร์เตอร์ เพื่อปรับสัญญาณให้อยู่ในรูปแบบเดิม ดังวงจร รูปที่ 3.4 วงจรเครื่องรับ

จากรูปที่ 3.3 ส่วนที่สองจะเป็นภาคถอดรหัส ซึ่งจะทำหน้าที่แปลงสัญญาณอนุกรมแบบอะซิงโครนัส ให้เป็นสัญญาณแบบขนานตามเดิม โดยใช้ซีพียู 8031 เป็นตัวจัดการในส่วนนี้

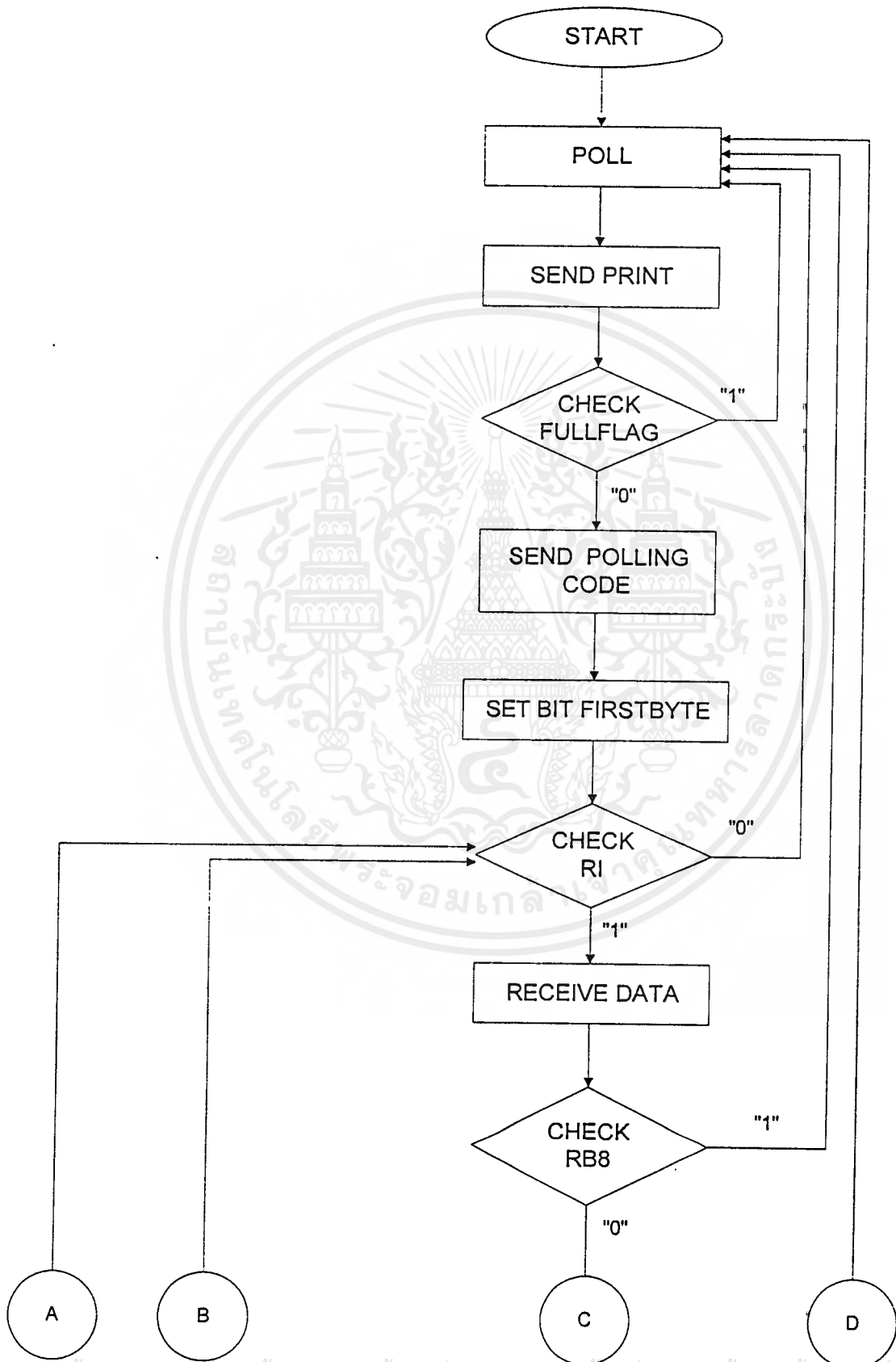


รูปที่ 3.4 วงจรเครื่องรับ

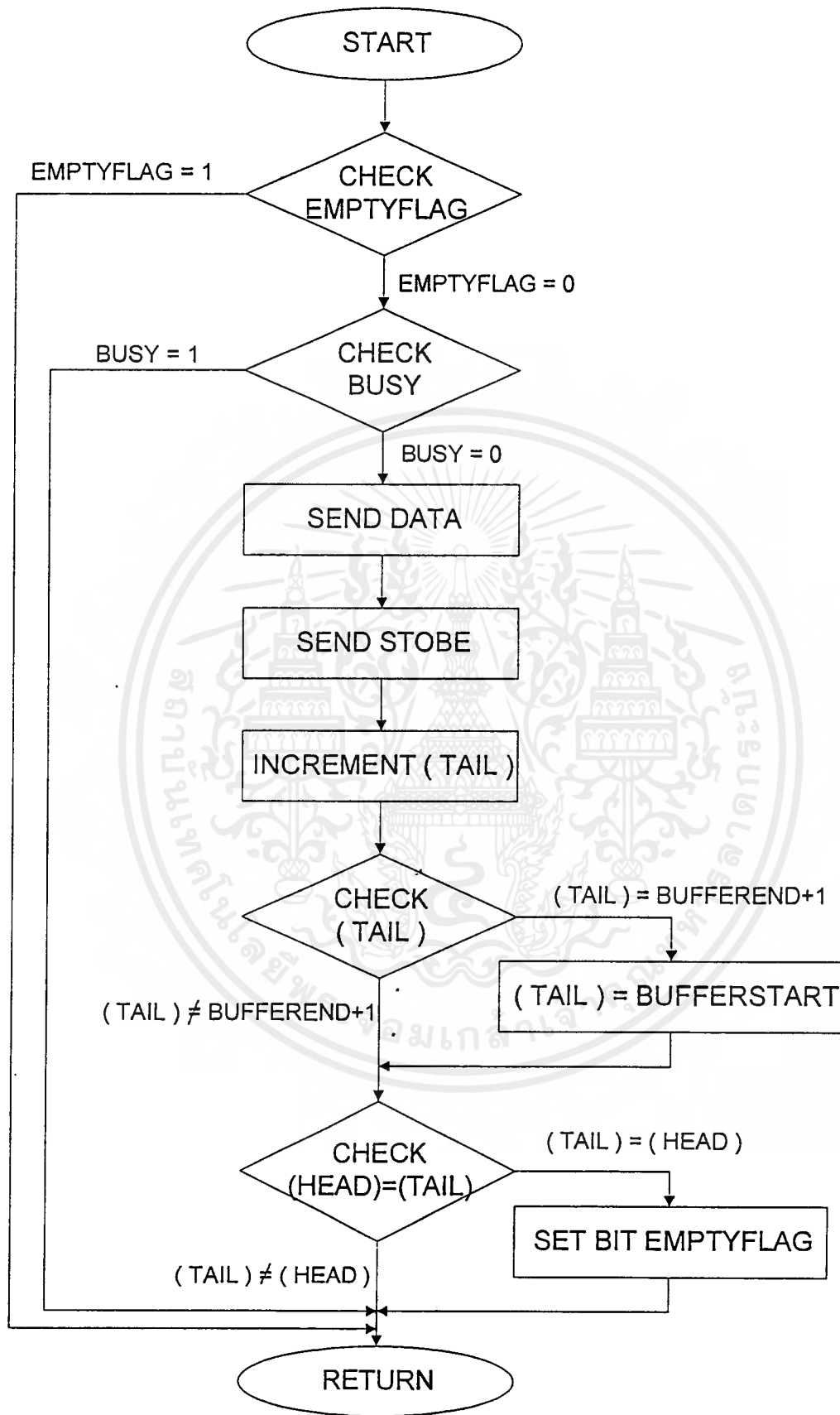
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.2.3 ส่วน SOFTWARE

โฟลชาร์ตแสดงการทำงานทางด้านรับของพริ้นเตอร์



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โปรแกรมทางด้านรับ เครื่องพรีนเตอร์

```

PCA EQU 4000H
PCB EQU 4001H
PCC EQU 4002H
PCWC EQU 4003H

```

```

TIBUAD9600 EQU 0FDH
B0 EQU 21H
B1 EQU 22H
B2 EQU 23H
PV EQU 24H
BUFFST EQU 26H
BUFFEND EQU 4FH
EMPTYFLAG EQU 0H
FULLFLAG EQU 1H
FIRSTBYTE EQU 2H

```

```

ORG 0000H
START: MOV R2,#0H
ST: MOV R3,#0H
DJNZ R3,$
DJNZ R2,ST

MOV DPTR,#PCWC
MOV A,#10010000B ; MODE 0 A IN B OUT C UP,LOW OUT
MOVX @DPTR,A
LCALL DELAY

MOV DPTR,#PCC
MOV A,#11111111B ; SET STROBE
MOVX @DPTR,A
LCALL DELAY

```

```

MOV      SP,#07H
MOV      TL1,#T1BUAD9600
MOV      TH1,#T1BUAD9600
MOV      SCON,#1100000B      ; MODE 3 , REN = 0
MOV      TMOD,#1010000B     ; TIMER 1 MODE 2 ( AUTORELOAD )
SETB     TR1                  ; START TIMER 1
SETB     EMPTYFLAG
CLR      FULLFLAG
CLR      TI
CLR      RI
SETB     REN
MOV      PV,#0A9H

```

```

-----
POLLNEXT: MOV      R0,#BUFFST      ; R0 ' HEAD '
          MOV      R1,#BUFFST      ; R1 ' TAIL '
          SETB     EMPTYFLAG
          CLR      FULLFLAG
          INC      PV
          MOV      A,PV
          CJNE    A,#0ACH,POLL
          MOV      PV,#0AAH

```

```

POLL:     ACALL    SENDPRN
          JB       FULLFLAG,POLL
          CLR      TI
          SETB     TB8
          MOV      SBUF,PV
          JNB     TI,$
          CLR      TB8
          CLR      TI
          SETB     FIRSTBYTE

```

```

WAIT:     MOV      B2,#10          ; WAIT FOR A BYTE

```

```

WAIT0:    MOV      B1,#200

```

```

WAIT00:   MOV      B0,#250

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับกร้ใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

WAIT1:    JB          RI,WAIT2
          NOP
          NOP
          NOP
          NOP
          NOP
          NOP
          DJNZ       B0,WAIT1          ; COUNT DOWN ( APPROX 5 S )
          DJNZ       B1,WAIT0
          DJNZ       B2,WAIT0
          ; TIME ' S UP ,NOT REPLIED ,POLL NEXT STATION
          SJMP      POLLNEXT
WAIT2:    CLR          RI
          MOV        A,SBUF
          JNB       RB8,REPLY        ; CHECK IF COMMAND
          JZ         POLLNEXT
          CJNE      A,#0FFH,WAIT     ; IF = 1FFH ,WAIT
          SJMP      WAIT             ; WAIT FOR A NEW BYTE
REPLY:    JNB       FIRSTBYTE,REP1
          CLR        FIRSTBYTE
          MOV        OLDBYTE,A
          CLR        TI
          CLR        TB8
          MOV        SBUF,A          ; REPLY
          JNB       TI,$
          CLR        TI
          SJMP      WAIT

```

```

REP1:    MOV        A,OLDBYTE                ; NEXT BYTE RECEIVED ,PUT OLDBYTE
                                                INTO BUFFER

        MOV        @R0,A
        INC        R0
        CLR        EMPTYFLAG
        CJNE       R0,#BUFFEND+1,CHKH       ; CHECK HEAD
        MOV        R0,#BUFFST

CHKH:    MOV        A,R0
        MOV        B,R1
        CJNE       A,B,RETURN              ; CHECK HEAD = TAIL ?
        SETB       FULLFLAG
        SETB       TB8
        CLR        TI
        MOV        SBUF,#0BBH
        JNB        TI,$
        CLR        TI
        CLR        TB8
        SJMP       POLL

RETURN:  MOV        A,SBUF
        CLR        RB8
        CLR        TI
        CLR        TB8
        MOV        SBUF,OLDBYTE
        JNB        TI,$
        CLR        TI
        MOV        OLDBYTE,A
        SJMP       WAIT

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

 PARALLEL PROGRAM

```

SENDPRN:  JB      EMPTYFLAG,RETURN1
           MOV     DPTR,#PCA           ; CHECK BUSY
           MOVX   A,@DPTR
           JB     ACC.0,RETURN1
           MOV     DPTR,#PCB           ; SEND DATA
           MOV     A,@R1
           MOVX   @DPTR,A
           MOV     DPTR,#PCC           ; STB ' LOW '
           CLR    A
           MOVX   @DPTR,A
           NOP
           NOP
           NOP
           NOP
           MOV     DPTR,#PCC           ; STB ' HIGH '
           MOV     A,#0FFH
           MOVX   @DPTR,A
           CLR    FULLFLAG
           INC    R1
           CJNE   R1,#BUFFEND+1,CHKT  ; CHECK TAIL
           MOV     R1,#BUFFST
CHKT:     MOV     A,R0                 ; CHECK HEAD = TAIL ?
           MOV     B,R1
           CJNE   A,B,SENDPRN
           SETB   EMPTYFLAG
RETURN1:  RET
  
```

```

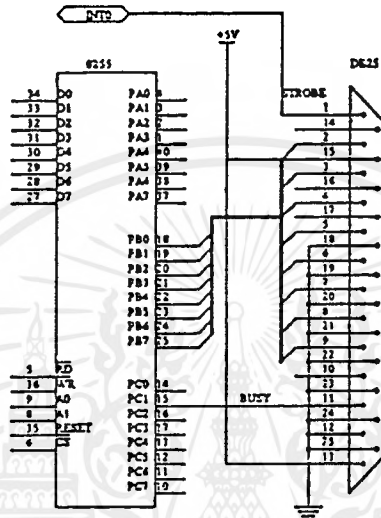
DELAY:   MOV     R2,#10H
RES:     MOV     R3,#10H
           DJNZ   R3,$
           DJNZ   R2,RES
           RET
  
```

เอกสารนี้เป็นเอกสารสงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.3 ส่วนเชื่อมต่อกับคอมพิวเตอร์และพริ้นเตอร์

3.3.1 ส่วนเชื่อมต่อกับคอมพิวเตอร์

จากรูปที่ 3.5 จะเป็นส่วนของการแปลงสัญญาณข้อมูลแบบขนานที่รับมาจากคอมพิวเตอร์ เพื่อส่งออกเป็นแบบอนุกรมอะซิงโครนัส เพื่อจะได้ลดจำนวนช่องสัญญาณในการส่งข้อมูลให้เหลือเพียงช่องเดียว



รูปที่ 3.5 แสดงการเชื่อมต่อแบบ DB-25 กับส่วนเชื่อมต่อ

จากรูปที่ 3.5 เป็นส่วนแปลงสัญญาณขนานให้เป็นแบบอนุกรมโดยรับข้อมูลจากคอมพิวเตอร์เข้ามาทางพอร์ต B ของ 8255 ในโหมด 1 โดยจะมีสัญญาณ STROBE เข้ามาเป็นอินเดอริบต์เพื่อให้เกิดการรับข้อมูลจากนั้น ซีพียู 8031 ก็จะเก็บข้อมูลไว้ในหน่วยความจำ และข้อมูลในหน่วยความจำนี้จะถูกแปลงไปเป็นข้อมูลแบบอนุกรม โดยกำหนดให้ส่งที่บอดเรท 9600 ซึ่งออกมาทางพอร์ตอนุกรม Tx

3.4 ภาคควบคุม

ภาคควบคุมนี้จะทำการควบคุมสัญญาณระหว่างเครื่องคอมพิวเตอร์กับเครื่องส่ง และระหว่างเครื่องรับกับเครื่องพรีนเตอร์ โดยการแปลงสัญญาณข้อมูลขนานเป็นอนุกรม และ อนุกรมเป็นขนานตามลำดับ รูปวงจรภาคควบคุมนี้แสดงในรูปที่ 3.7

วงจรภาคควบคุมในรูปที่ 3.7 8031 เป็นซีพียู ใช้คริสตอลขนาด 11.059 Mhz โดยมี คาปาซิเตอร์ (20 pF) ต่ออยู่ที่ขาทั้งสองเพื่อทำให้วงจรสร้างความถี่ใน ซีพียู มีเสถียรภาพ ขา PSEN และ ขา READ ของ 8031 เข้าแอนด์เกต (AND GATE : ไอซีเบอร์ 74 LS 08) เอาท์พุทที่ได้จะเข้าที่ขาเอาท์พุทเอนาเบิล (ENABLE) ของรอม และ แรม (ไอซีเบอร์ 2764 และ 6264 ตามลำดับ) ซึ่งมีขนาดอย่างละ 8 K ทำให้ รอม และ แรม เป็นได้ทั้งโค้ดเมมโมรี (Code memory) และ ดาต้าเมมโมรี (Data memory)

ไอซีเบอร์ 74LS373 ต่ออยู่กับขา ALE ของซีพียู CPU 8031 เพราะทั้งดาต้าเมมโมรี (Data memory) และ แอดเดรสเมมโมรี (Address memory) ถูกมัลติเพล็กซ์มาด้วยกันเพื่อลดจำนวนขาของซีพียู ตัว ไอซีเบอร์ 74LS373 ทำหน้าที่เป็นตัว แลทช์ ซึ่งจะแยก ดาต้า และ แอดเดรส เมมโมรีออกจากกัน

ขา 0000 - 1FFF ต่อกับ อีพรอม (EPROM)

ขา 2000 - 3FFF ต่อกับ แรม (RAM)

ขา 4000 - 4003 ต่อกับ 8255

ในการรับสัญญาณแบบขนานจากเครื่องคอมพิวเตอร์จะทำโดย 8255 หลังจากนั้นซีพียู 8031 จะเปลี่ยนสัญญาณขนานนี้ให้เป็นสัญญาณอนุกรม เพื่อส่งให้เครื่องส่งผ่านทางขา Tx และซีพียู 8031 ทางด้านรับจะรับสัญญาณที่ส่งมานี้ทางขา RX หลังจากนั้นจะทำการเปลี่ยนสัญญาณให้เป็นสัญญาณแบบขนาน โดย 8255 ทางด้านรับเป็นตัวส่งสัญญาณแบบขนานให้กับพรีนเตอร์

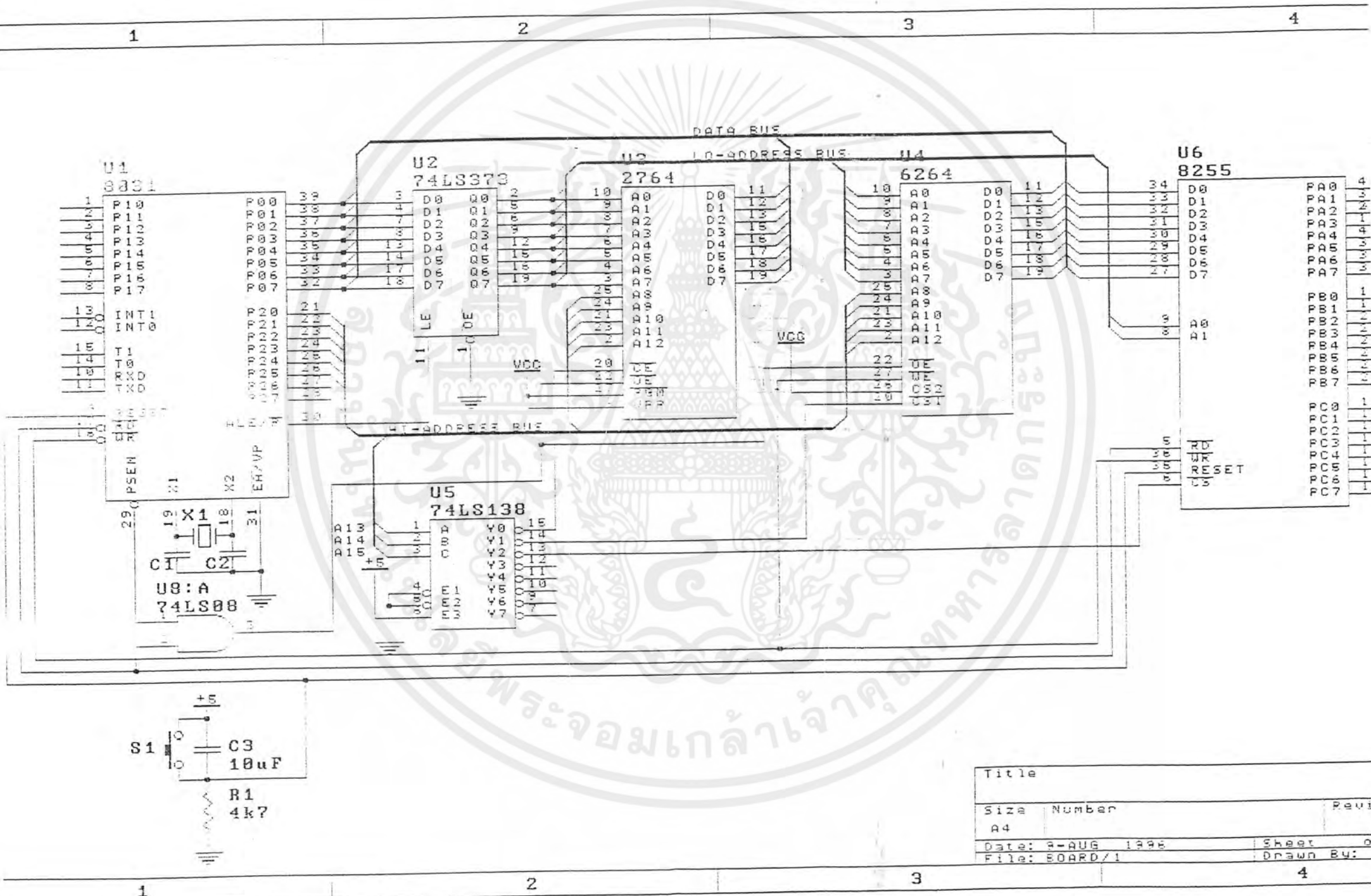
ขา A13 , A14 , A15 ต่อเข้าขา A, B, C ของ ไอซี เบอร์ 74LS138 เพื่อแบ่งความจำของ รอม และ แรม เป็น 8 เพจ เพจละ 8 K ทำให้ รอม และ แรม มีขนาดตัวละ 8 K นั่นเอง โดย

ขา Y0 ของ ไอซีเบอร์ 74LS138 ต่อกับ รอม

ขา Y1 ของ ไอซีเบอร์ 74LS138 ต่อกับ แรม

ขา Y2 ของ ไอซีเบอร์ 74LS138 ต่อกับ 8255

ในแผ่นวงจรควบคุม ตัวรีเซต คือ R1 (4.7 K) และ C1 (100 uF) ทำหน้าที่รีเซตซีพียูทุกครั้งที่มีกระแสไฟเลี้ยงเข้ามาหรืออาจใช้ปุ่มรีเซตก็ได้



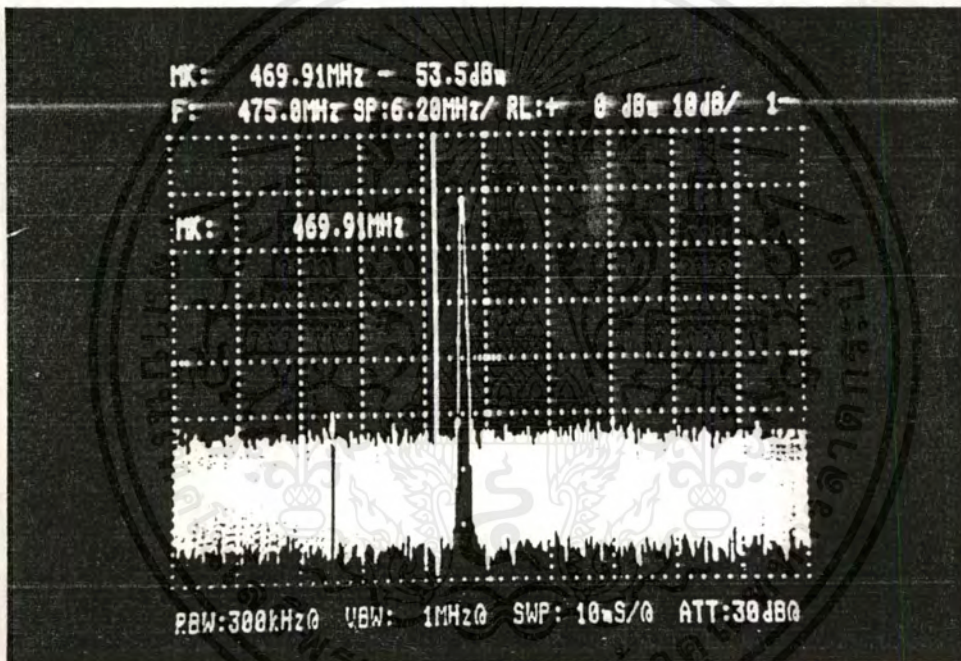
รูปที่ 3.7 แสดงวงจรภาคควบคุมที่ส่งงานไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 4

ผลการทดลอง

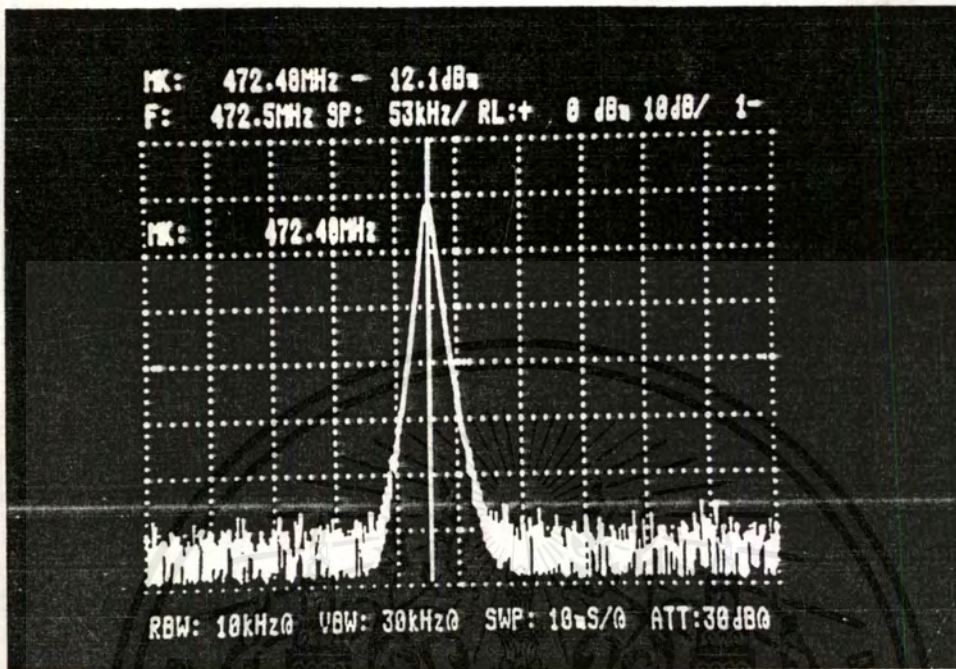
4.1 ภาคส่งและภาครับ

เมื่อทดลองเครื่องส่ง โดยป้อนไฟกระแสตรง + 5 V เข้าที่อินพุท (โดยไม่ผ่านอินเวอร์เตอร์) แล้วนำเครื่องส่งไปวัดความถี่ที่ผลิตออกมา โดยใช้สเปคตรัมอานาไลเซอร์ จะวัดสัญญาณได้ดังใน รูปที่ 4.1 และมีความถี่ขนาดสูงสุด 472.5 MHz ดังในรูปที่ 4.2

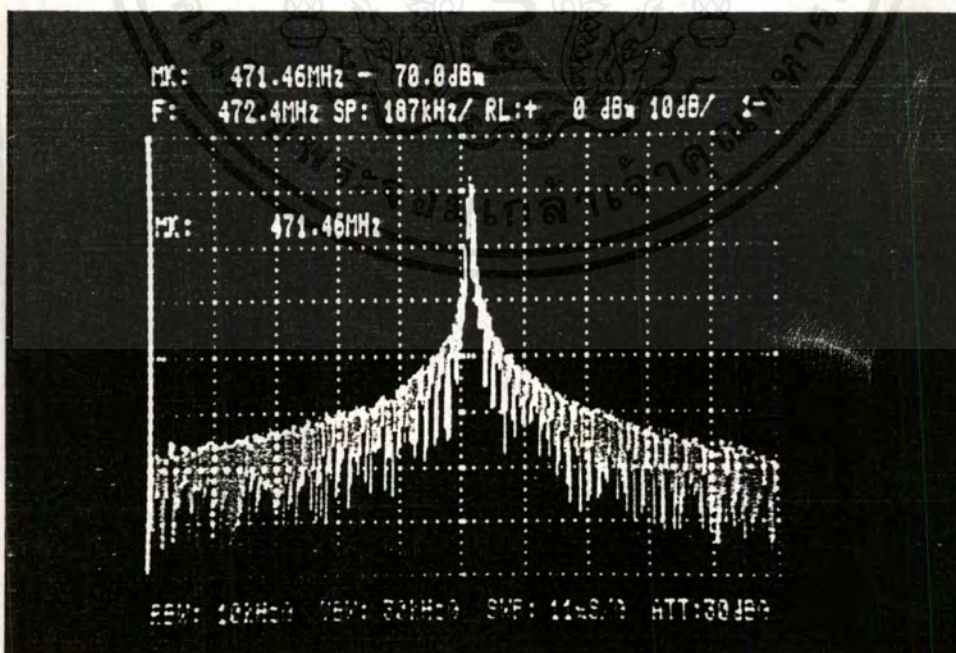


รูปที่ 4.1 ลักษณะของสัญญาณที่ผลิตออกมาจากเครื่องส่ง

เมื่อป้อนสัญญาณรูปสี่เหลี่ยมความถี่ 5 กิโลเฮิรตซ์ เข้าที่เครื่องส่ง แล้วนำเครื่องส่งไปวัดสัญญาณที่ผลิตออกมา โดยใช้สเปคตรัมอานาไลเซอร์จะได้สัญญาณในลักษณะดังรูปที่ 4.3 และมีขนาดความถี่สูงสุดประมาณ 472.5 Mhz ดังรูปที่ 4.4

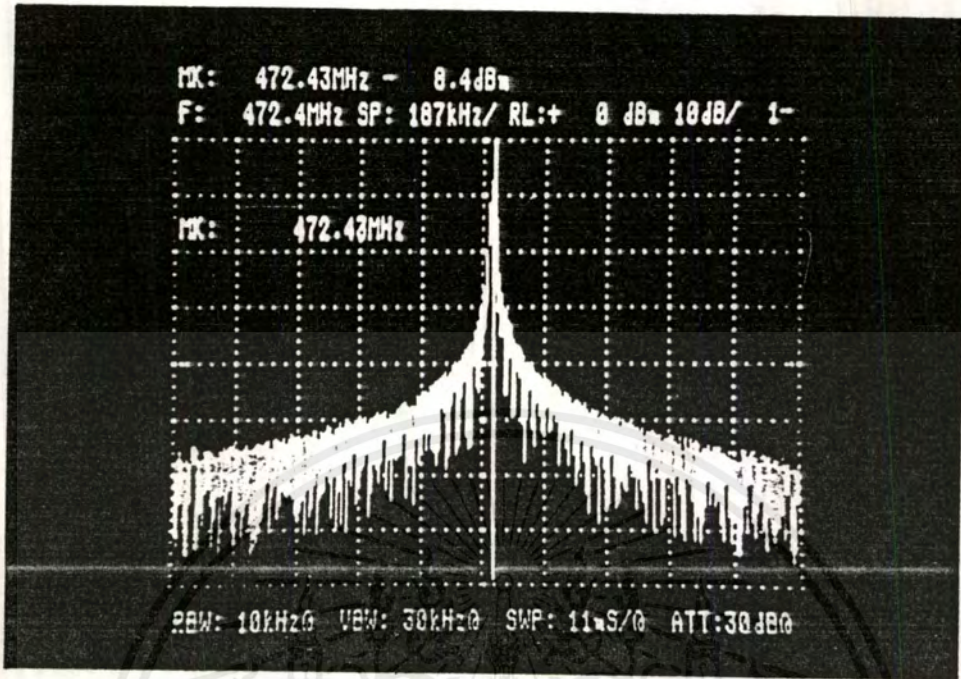


รูปที่ 4.2 รูปสัญญาณที่สเปกตรัมออกมา มีความถี่ของเครื่องส่งที่ 472.5 MHz

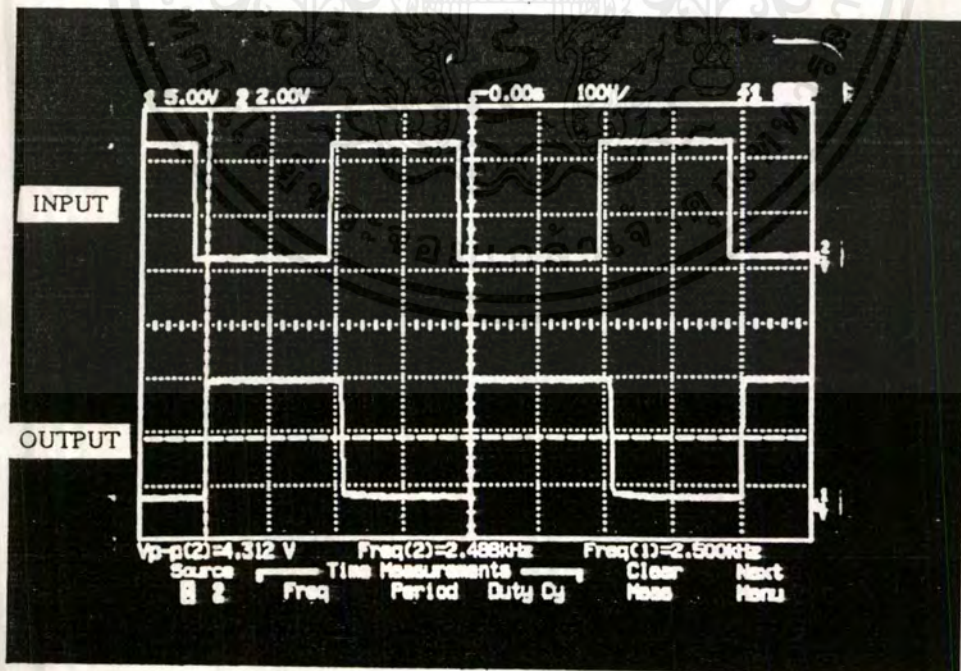


รูปที่ 4.3 รูปสัญญาณของเครื่องส่งเมื่อป้อนสัญญาณความถี่ 5 kHz

เอกสารนี้เป็นเอกสารสงวนลิขสิทธิ์ไว้เพื่อใช้ในการศึกษาเท่านั้น ไม่อนุญาตให้เผยแพร่ไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



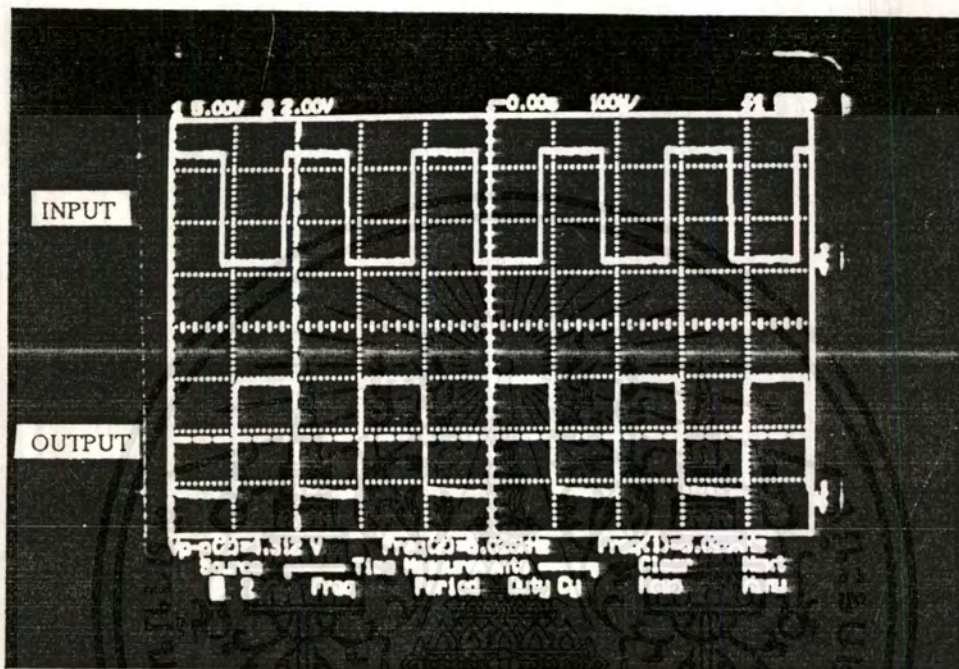
รูปที่ 4.4 แสดงความถี่ของเครื่องส่งเมื่อป้อนสัญญาณความถี่ 5 kHz



รูปที่ 4.5 รูปสัญญาณเมื่อป้อนความถี่ 2.5 kHz

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ในการปรับแต่งเครื่องรับนั้นเราจะป้อนสัญญาณรูปสี่เหลี่ยมเข้าที่เครื่องส่ง แล้วทำการปรับค่าตัวเก็บประจุที่ปรับค่าได้ (C_1) จนเครื่องรับสามารถรับสัญญาณจากเครื่องส่งได้ เมื่อทำการป้อนสัญญาณรูปสี่เหลี่ยมที่ความถี่ 2.5 kHz และ 5 kHz เข้าที่เครื่องส่ง แล้วรับสัญญาณมายังเครื่องรับ จะได้ลักษณะของสัญญาณอินพุต และ เอาท์พุท ดังรูปที่ 4.5 และ 4.6 ตามลำดับ

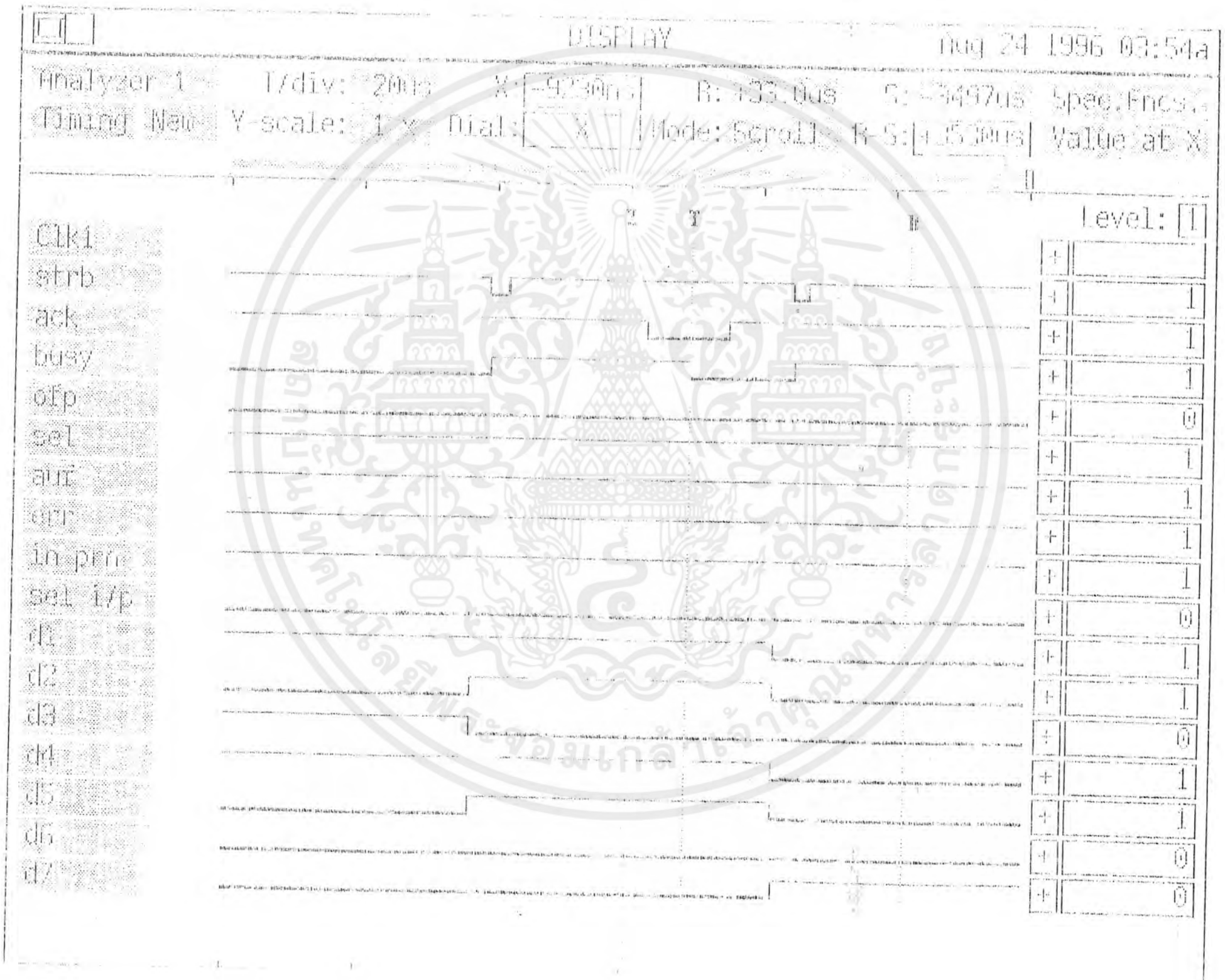


รูปที่ 4.6 รูปสัญญาณเมื่อป้อนความถี่ 5 kHz

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

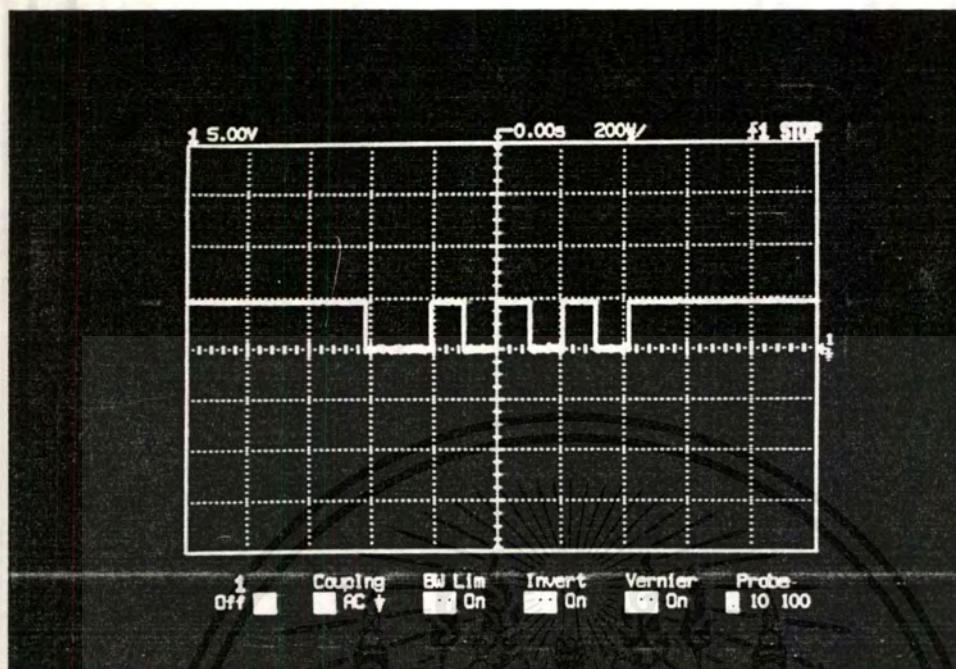
4.2 ส่วนเชื่อมต่อระหว่างคอมพิวเตอร์ และ พรินเตอร์

เมื่อทดลองใช้เครื่องลอจิกแอนาไลเซอร์ (Logic Analyzer) ได้ผลสัญญาณระหว่างคอมพิวเตอร์กับพรินเตอร์ออกมาดังในรูปที่ 4.7

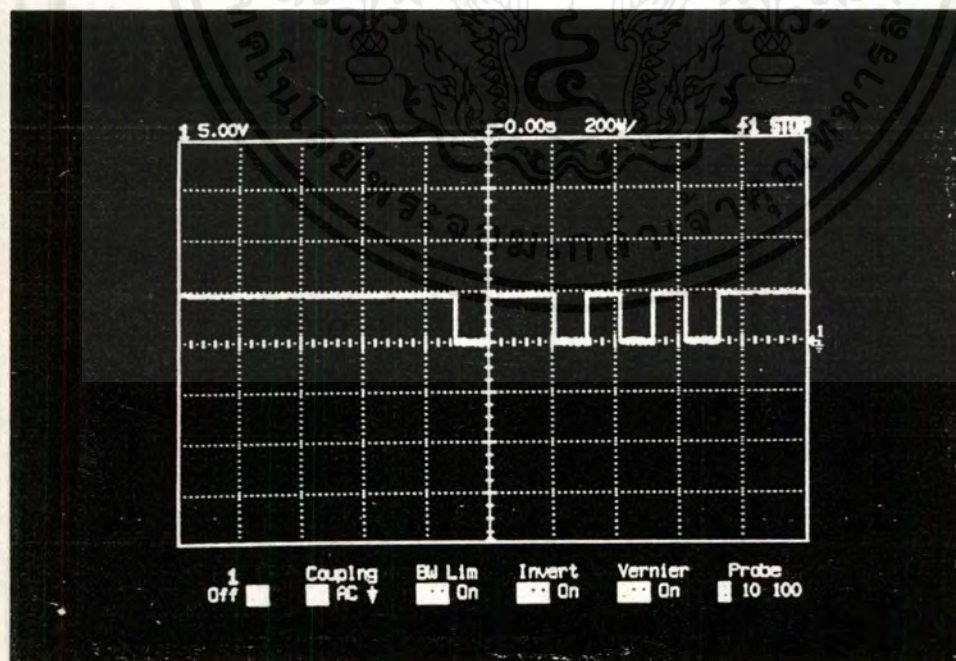


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
รูปที่ 4.7 แสดงสัญญาณระหว่างเครื่องคอมพิวเตอร์กับพรินเตอร์

4.3 ส่วนแสดงสัญญาณต่างๆ ของเครื่องสลับพรีนเตอร์ไร้สาย

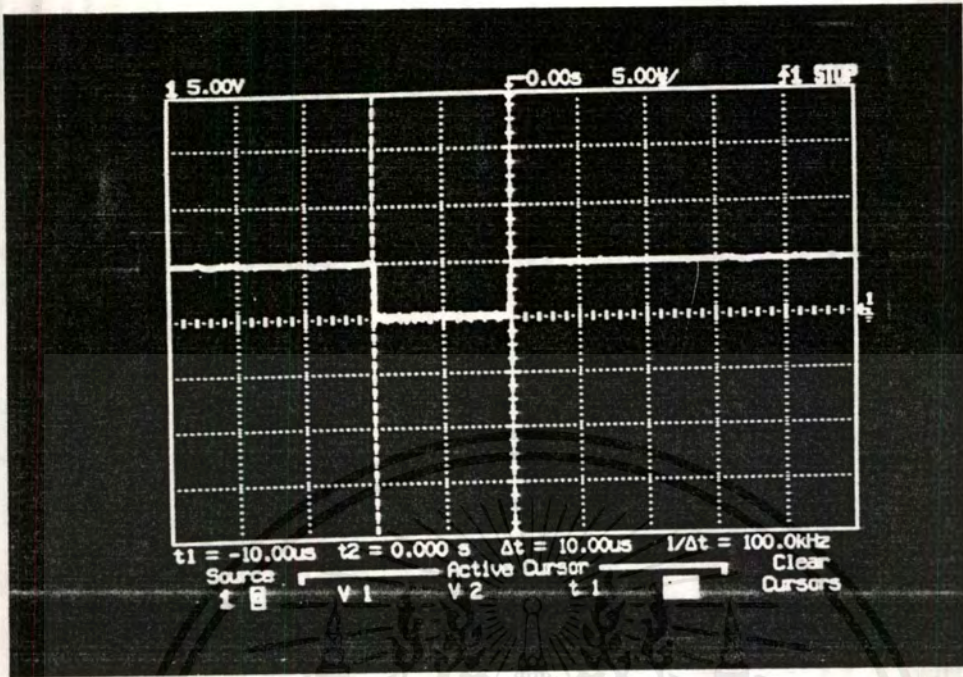


รูปที่ 4.8 แสดงสัญญาณโพลลิ่ง * 1AA *

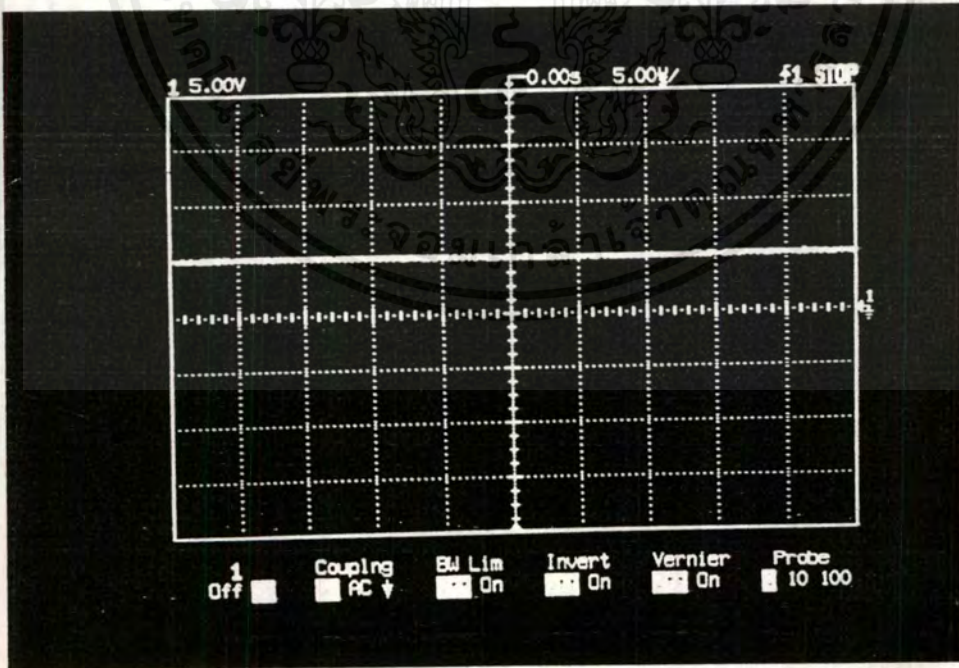


รูปที่ 4.9 แสดงสัญญาณโพลลิ่ง * 1AB *

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

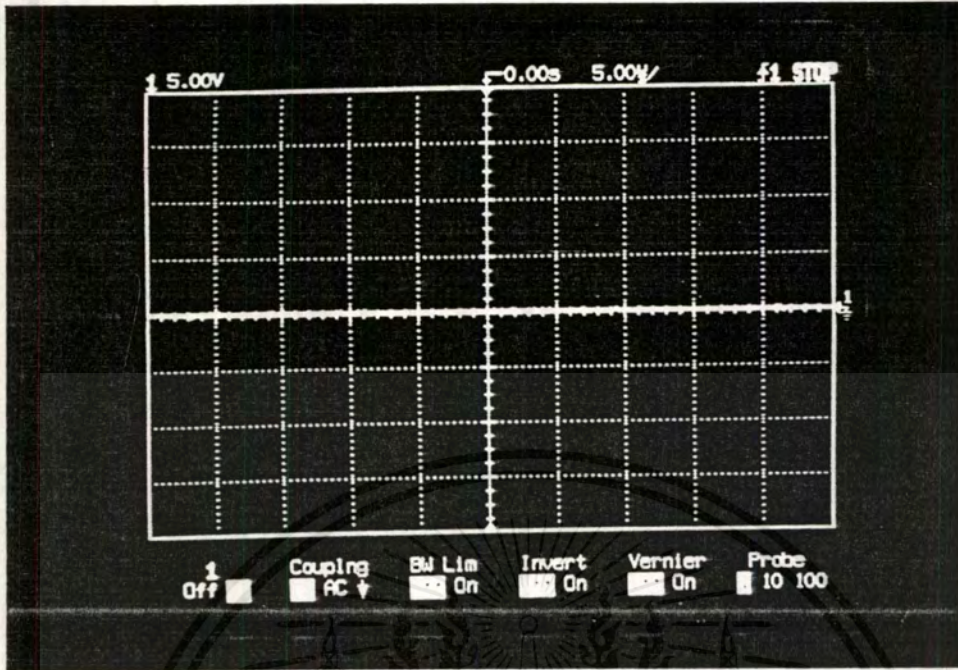


รูปที่ 4.10 สัญญาณสโตรป (STROBE) ที่ส่งให้พริบเตอร์

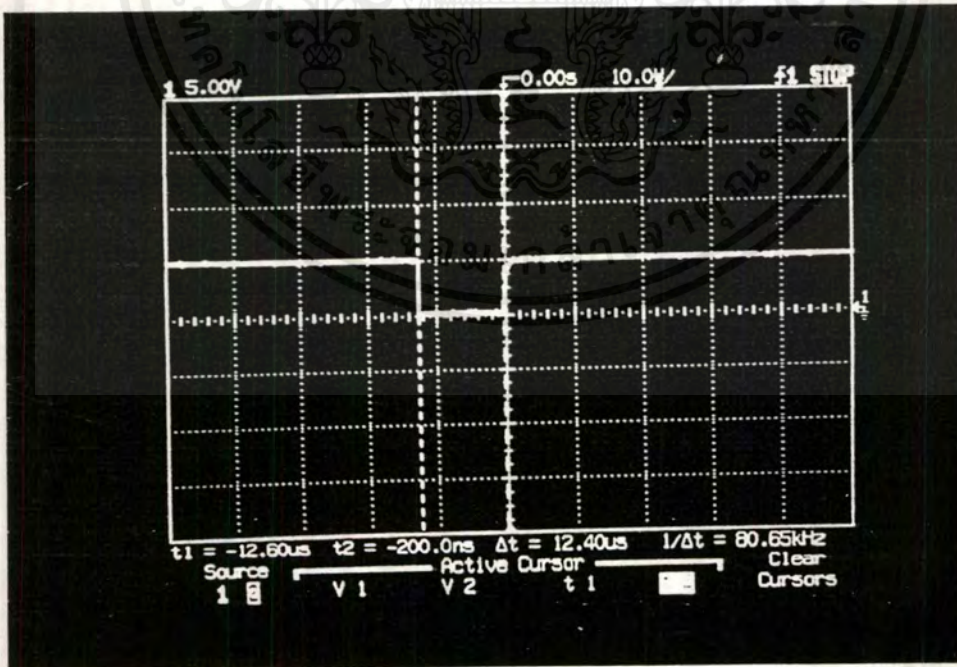


รูปที่ 4.11 สัญญาณบีซี (BUSY) จากพริบเตอร์แสดงสถานะไม่ว่าง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

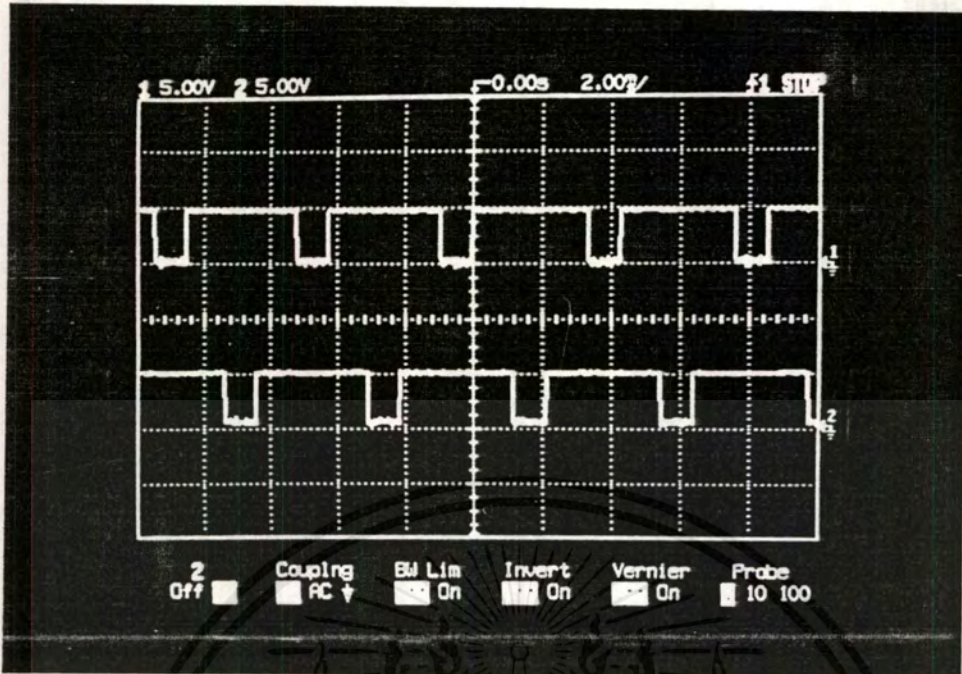


รูปที่ 4.12 สัญญาณบิต (BUSY) จากพริินเตอร์แสดงสถานะว่าง

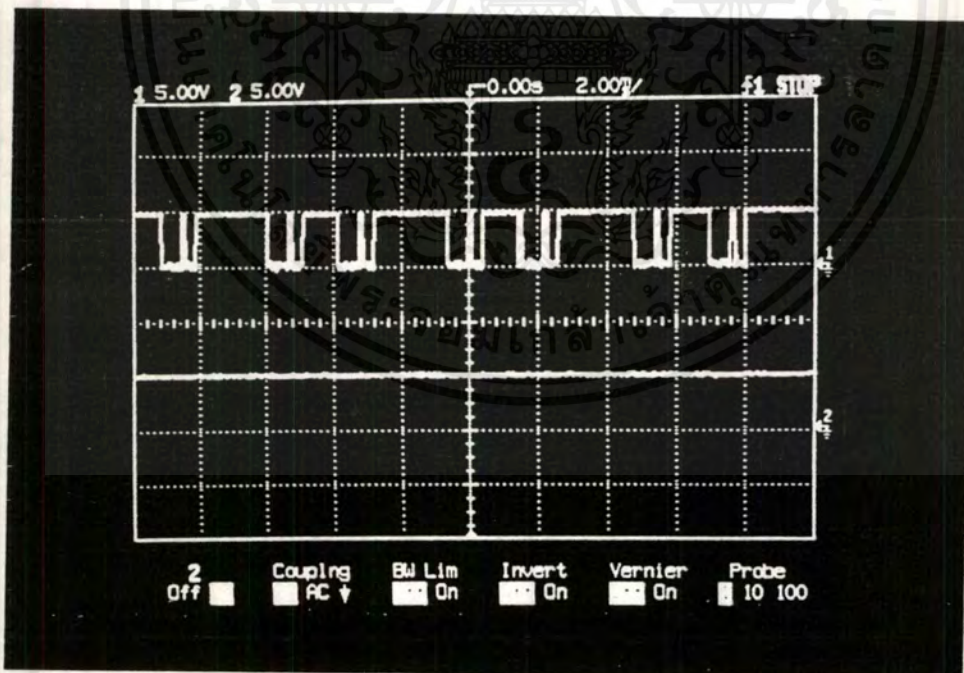


รูปที่ 4.13 สัญญาณแอดโนเลข (ACKNOWLEDGE) จากพริินเตอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

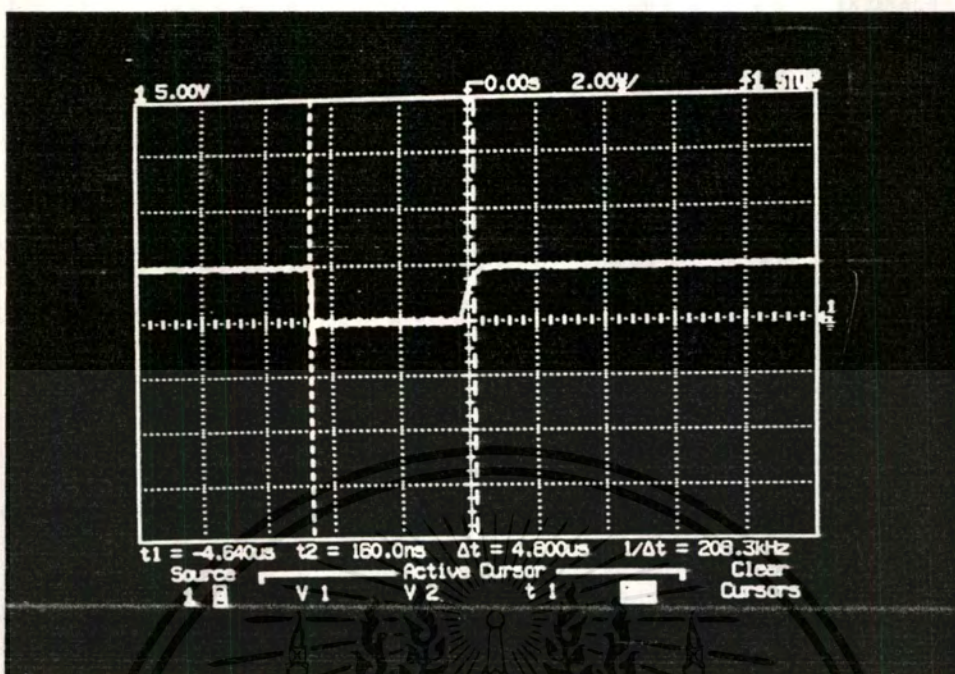


รูปที่ 4.14 สัญญาณจากส่วนของเครื่องคอมพิวเตอร์เมื่อไม่มีข้อมูลจะพิมพ์



รูปที่ 4.15 สัญญาณข้อมูลจากส่วนของเครื่องคอมพิวเตอร์เมื่อมีข้อมูลจะพิมพ์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.16 สัญญาณสโตรป (strobe) จากเครื่องคอมพิวเตอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 5

สรุปผลและวิเคราะห์ผลการดำเนินงาน

จากการทำโครงการที่ผ่านมาทำให้ได้รับความรู้เพิ่มมากขึ้นในด้านการส่งข้อมูลแบบดิจิทัล ตลอดจนการปรับแต่งสัญญาณของเครื่องรับและเครื่องส่ง การปรับแต่งเครื่องรับ เครื่องส่งให้รับส่งกันได้ โดยการปรับค่าตัวเก็บประจุที่ปรับค่าได้ที่วงจรรุ่นของทั้งเครื่องรับ และ เครื่องส่ง ซึ่งจะต้องปรับค่าตัวเก็บประจุที่เครื่องส่งให้ได้ความถี่ที่ต้องการก่อน

โดยเครื่องส่งที่จะออกซิลเลทียนความถี่เฮซเฟอออกมา นั้น จะต้องระวังในการต่อตัวคาปาซิเตอร์ที่ขนานกับ L ซึ่งใช้ไลน์ทองแดงบนแผ่นปริ้นท์แทน เพราะถ้าตำแหน่งการวางตัวคาปาซิเตอร์คลาดเคลื่อน จะทำให้ความถี่ที่ผลิตออกมาได้มีค่าผิดพลาดไปจากที่ต้องการมาก

ในส่วนของเครื่องรับ เราจะปรับจูนให้รับสัญญาณจากเครื่องส่งได้ โดยทำการปรับสัญญาณรูปสี่เหลี่ยมเข้าที่เครื่องส่ง แล้ว ปรับค่าตัวเก็บประจุที่ปรับค่าได้ที่วงจรรุ่น ซึ่งอยู่ต่อจากวงจรรีเฟรอนเอน (Front End) ของเครื่องรับ





เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก

MAXIMUM RATINGS

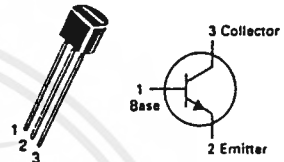
Rating	Symbol	Value	Unit
Collector-Emitter Voltage	V_{CEO}	25	Vdc
Collector-Base Voltage	V_{CBO}	30	Vdc
Emitter-Base Voltage	V_{EBO}	3.0	Vdc
Total Device Dissipation @ $T_A = 25^\circ\text{C}$ Derate above 25°C	P_D	350 2.8	mW mW/°C
Total Device Dissipation @ $T_C = 25^\circ\text{C}$ Derate above 25°C	P_D	1.0 8.0	Watt mW/°C
Operating and Storage Junction Temperature Range	T_J, T_{stg}	-55 to +150	°C

THERMAL CHARACTERISTICS

Characteristic	Symbol	Max	Unit
Thermal Resistance, Junction to Ambient	$R_{\theta JA}$	357	°C/W
Thermal Resistance, Junction to Case	$R_{\theta JC}$	125	°C/W

MPSH10*
MPSH11*

CASE 29-04, STYLE 2
TO-92 (TO-226AA)



VHF/UHF TRANSISTORS

NPN SILICON

*These are Motorola
designated preferred devices.

ELECTRICAL CHARACTERISTICS ($T_A = 25^\circ\text{C}$ unless otherwise noted.)

Characteristic	Symbol	Min	Max	Unit
OFF CHARACTERISTICS				
Collector-Emitter Breakdown Voltage ($I_C = 1.0 \text{ mA}, I_E = 0$)	$V_{(BR)CEO}$	25	—	Vdc
Collector-Base Breakdown Voltage ($I_C = 100 \mu\text{A}, I_E = 0$)	$V_{(BR)CBO}$	30	—	Vdc
Emitter-Base Breakdown Voltage ($I_E = 10 \mu\text{A}, I_C = 0$)	$V_{(BR)EBO}$	3.0	—	Vdc
Collector Cutoff Current ($V_{CB} = 25 \text{ Vdc}, I_E = 0$)	I_{CBO}	—	100	nA
Emitter Cutoff Current ($V_{EB} = 2.0 \text{ Vdc}, I_C = 0$)	I_{EBO}	—	100	nA
ON CHARACTERISTICS				
DC Current Gain ($I_C = 4.0 \text{ mA}, V_{CE} = 10 \text{ Vdc}$)	h_{FE}	60	—	—
Collector-Emitter Saturation Voltage ($I_C = 4.0 \text{ mA}, I_B = 0.4 \text{ mA}$)	$V_{CE(sat)}$	—	0.5	Vdc
Base-Emitter On Voltage ($I_C = 4.0 \text{ mA}, V_{CE} = 10 \text{ Vdc}$)	V_{BE}	—	0.95	Vdc
SMALL-SIGNAL CHARACTERISTICS				
Current-Gain — Bandwidth Product ($I_C = 4.0 \text{ mA}, V_{CE} = 10 \text{ Vdc}, f = 100 \text{ MHz}$)	f_T	650	—	MHz
Collector-Base Capacitance ($V_{CB} = 10 \text{ Vdc}, I_E = 0, f = 1.0 \text{ MHz}$)	C_{cb}	—	0.7	pF
Common-Base Feedback Capacitance ($V_{CB} = 10 \text{ Vdc}, I_E = 0, f = 1.0 \text{ MHz}$)	C_{rb}	0.35 0.6	0.65 0.9	pF
Collector Base Time Constant ($I_C = 4.0 \text{ mA}, V_{CB} = 10 \text{ Vdc}, f = 31.8 \text{ MHz}$)	$rb'C_c$	—	9.0	ps

MPSH10, MPSH11

COMMON-BASE γ PARAMETERS versus FREQUENCY
($V_{CB} = 10 \text{ Vdc}$, $I_C = 4.0 \text{ mAdc}$, $T_A = 25^\circ\text{C}$)

Y_{ib} , INPUT ADMITTANCE

FIGURE 1 - RECTANGULAR FORM

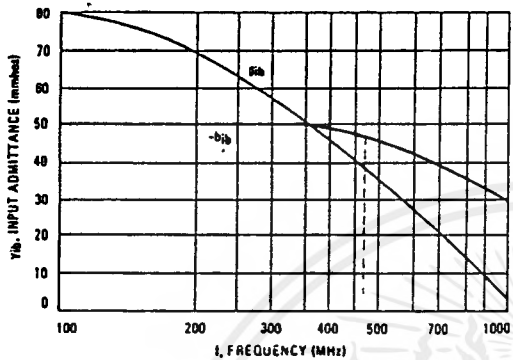
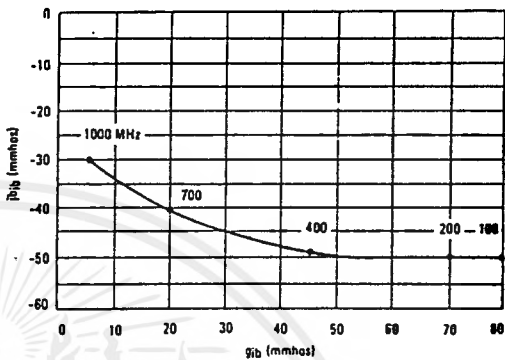


FIGURE 2 - POLAR FORM



COMMON-BASE γ PARAMETERS versus FREQUENCY
($V_{CB} = 10 \text{ Vdc}$, $I_C = 4.0 \text{ mAdc}$, $T_A = 25^\circ\text{C}$)

Y_{fb} , FORWARD TRANSFER ADMITTANCE

FIGURE 3 - RECTANGULAR FORM

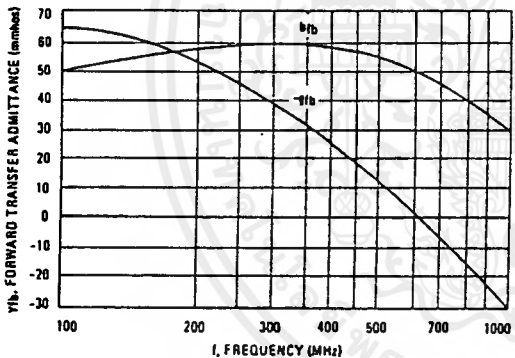
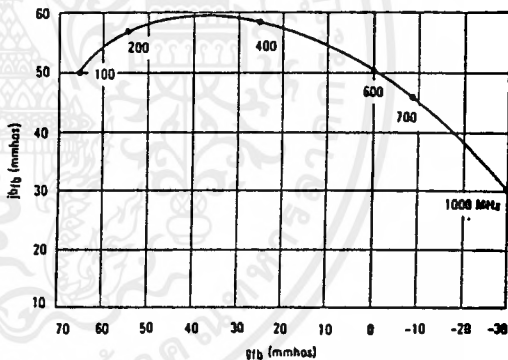


FIGURE 4 - POLAR FORM



Y_{rb} , REVERSE TRANSFER ADMITTANCE

FIGURE 5 - RECTANGULAR FORM

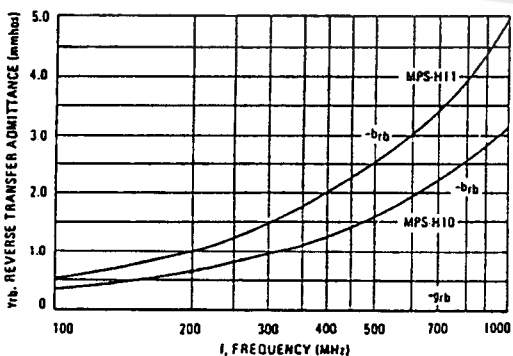
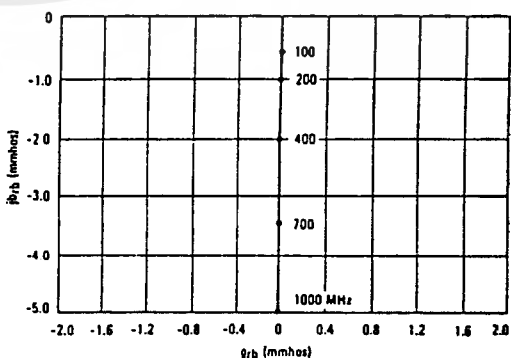


FIGURE 6 - POLAR FORM



MOTOROLA SMALL-SIGNAL TRANSISTORS, FETs AND DIODES

MPSH10, MPSH11

Y_{ob} , OUTPUT ADMITTANCE

FIGURE 7 - RECTANGULAR FORM

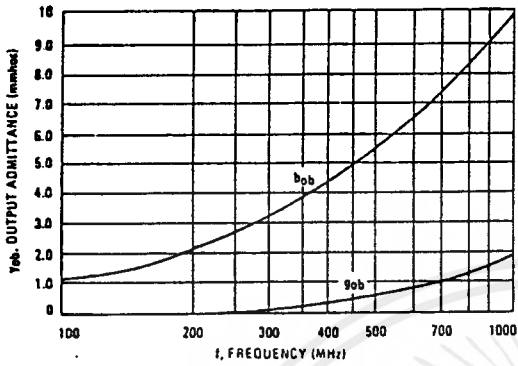
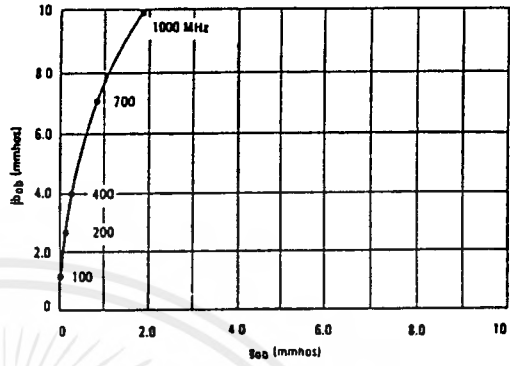


FIGURE 8 - POLAR FORM



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



MOTOROLA

**LF347
LF351
LF353**

JFET INPUT OPERATIONAL AMPLIFIERS

These low cost JFET input operational amplifiers combine two state-of-the-art linear technologies on a single monolithic integrated circuit. Each internally compensated operational amplifier has well matched high voltage JFET input devices for low input offset voltage. The BIFET technology provides wide bandwidths and fast slew rates with low input bias currents, input offset currents, and supply currents.

These devices are available in single, dual and quad operational amplifiers which are pin-compatible with the industry standard MC1741, MC1458, and the MC3403/LM324 bipolar devices.

- Input Offset Voltage of 5.0 mV Max (LF347B)
- Low Input Bias Current - 50 pA
- Low Input Noise Voltage - 16 nV/√Hz
- Wide Gain Bandwidth - 4.0 MHz
- High Slew Rate - 13 V/μs
- Low Supply Current - 1.8 mA per Amplifier
- High Input Impedance - 10¹² Ω
- High Common-Mode and Supply Voltage Rejection Ratios - 100 dB

MAXIMUM RATINGS

Rating	Symbol	Value	Unit
Supply Voltage	V _{CC} V _{EE}	+18 -18	V
Differential Input Voltage	V _{ID}	±30	V
Input Voltage Range (Note 1)	V _{IDR}	±15	V
Output Short Circuit Duration (Note 2)	t _S	Continuous	
Power Dissipation at T _A = +25°C	P _D	900	mW
Derate above T _A = +25°C	W/°C	10	mW/°C
Operating Ambient Temperature Range	T _A	0 to +70	°C
Operating Junction Temperature Range	T _J	115	°C
Storage Temperature Range	T _{stg}	-65 to +150	°C

NOTES:

1. Unless otherwise specified, the absolute maximum negative input voltage is limited to the negative power supply.
2. Any amplifier output can be shorted to ground indefinitely. However, if more than one amplifier output is shorted simultaneously, maximum junction temperature ratings may be exceeded.

**FAMILY OF BIFET OPERATIONAL AMPLIFIERS
SILICON MONOLITHIC INTEGRATED CIRCUITS**



**N SUFFIX
PLASTIC PACKAGE
CASE 626-05
(LF351, LF353 Only)**



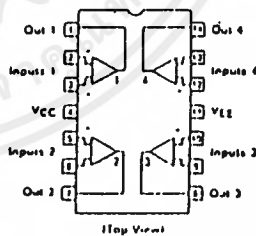
**D SUFFIX
PLASTIC PACKAGE
CASE 751-02
SO-8
(LF351, LF353 Only)**



**N SUFFIX
PLASTIC PACKAGE
CASE 646-08
(LF347 Only)**



**D SUFFIX
PLASTIC PACKAGE
CASE 751A-02
SO-14**



ORDERING INFORMATION

Function	Device	Package
Single	LF351D	SO-8
Single	LF351N	Plastic DIP
Dual	LF353D	SO-8
Dual	LF353N	Plastic DIP
Quad	LF347D	SO-14
Quad	LF347DN	Plastic DIP
Quad	LF347N	Plastic DIP

MOTOROLA LINEAR/INTERFACE DEVICES

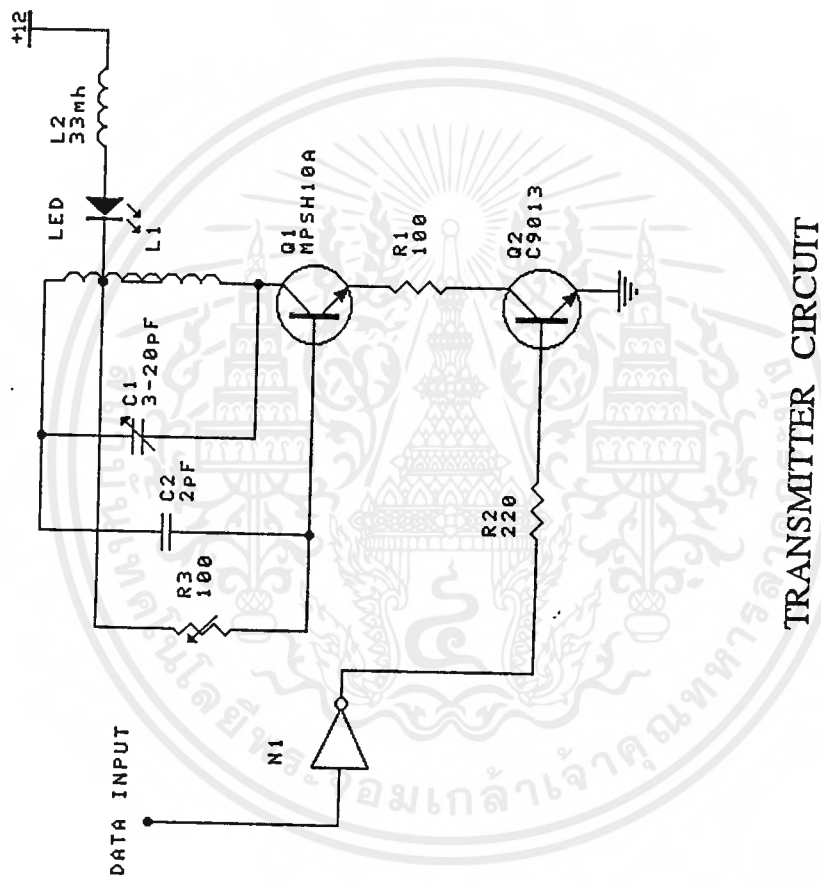
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

7.086. PRINTER ADAPTER I/O PORT USAGE

Port	Bit Number								Adapter	Direction	Function
	7	6	5	4	3	2	1	0			
378	✓								Printer	Output	Controls pin 9 (data bit 7)
		✓							Printer	Output	Controls pin 8 (data bit 6)
			✓						Printer	Output	Controls pin 7 (data bit 5)
				✓					Printer	Output	Controls pin 6 (data bit 4)
					✓				Printer	Output	Controls pin 5 (data bit 3)
						✓			Printer	Output	Controls pin 4 (data bit 2)
							✓		Printer	Output	Controls pin 3 (data bit 1)
							✓	Printer	Output	Controls pin 2 (data bit 0)	
379	✓								Printer	Input	Status of pin 11 (busy)
		✓							Printer	Input	Status of pin 10 (acknowledge)
			✓						Printer	Input	Status of pin 12 (out of paper)
				✓					Printer	Input	Status of pin 13 (select)
					✓				Printer	Input	Status of pin 15 (error)
37A						✓	✓	✓	Printer	Input	NOT USED
	✓	✓	✓						Printer	Input	Status of IRQ Enable
				✓					Printer	Input	Inverted status of pin 17 (select input)
					✓				Printer	Input	Status of pin 16 (initialize printer)
						✓			Printer	Input	Inverted status of pin 14 (auto feed)
							✓		Printer	Input	Inverted status of pin 1 (strobe)
	✓	✓	✓	✓					Printer	Output	NOT USED
					✓				Printer	Output	Inverted status of pin 17 (select input)
						✓			Printer	Output	Status of pin 16 (initialize printer)
							✓		Printer	Output	Inverted status of pin 14 (auto feed)
3BC	✓								MDA	Output	Controls pin 9 (data bit 7)
		✓							MDA	Output	Controls pin 8 (data bit 6)
			✓						MDA	Output	Controls pin 7 (data bit 5)
				✓					MDA	Output	Controls pin 6 (data bit 4)
					✓				MDA	Output	Controls pin 5 (data bit 3)
						✓			MDA	Output	Controls pin 4 (data bit 2)
							✓		MDA	Output	Controls pin 3 (data bit 1)
							✓	MDA	Output	Controls pin 2 (data bit 0)	
3BD	✓								MDA	Input	Status of pin 11 (busy)
		✓							MDA	Input	Status of pin 10 (acknowledge)
			✓						MDA	Input	Status of pin 12 (out of paper)
				✓					MDA	Input	Status of pin 13 (select)
					✓				MDA	Input	Status of pin 15 (error)
						✓	✓	✓	MDA	Input	NOT USED
3BE	✓	✓	✓						MDA	Input	NOT USED
				✓					MDA	Input	Status of IRQ enable
					✓				MDA	Input	Inverted status of pin 17 (select input)
						✓			MDA	Input	Status of pin 16 (initialize printer)
							✓		MDA	Input	Inverted status of pin 14 (auto feed)
								✓	MDA	Input	Inverted status of pin 1 (strobe)
	✓	✓	✓	✓					MDA	Output	NOT USED
					✓				MDA	Output	Inverted status of pin 17 (select input)
						✓			MDA	Output	Status of pin 16 (initialize printer)
						✓		MDA	Output	Inverted status of pin 14 (auto feed)	
							✓	MDA	Output	Inverted status of pin 1 (strobe)	

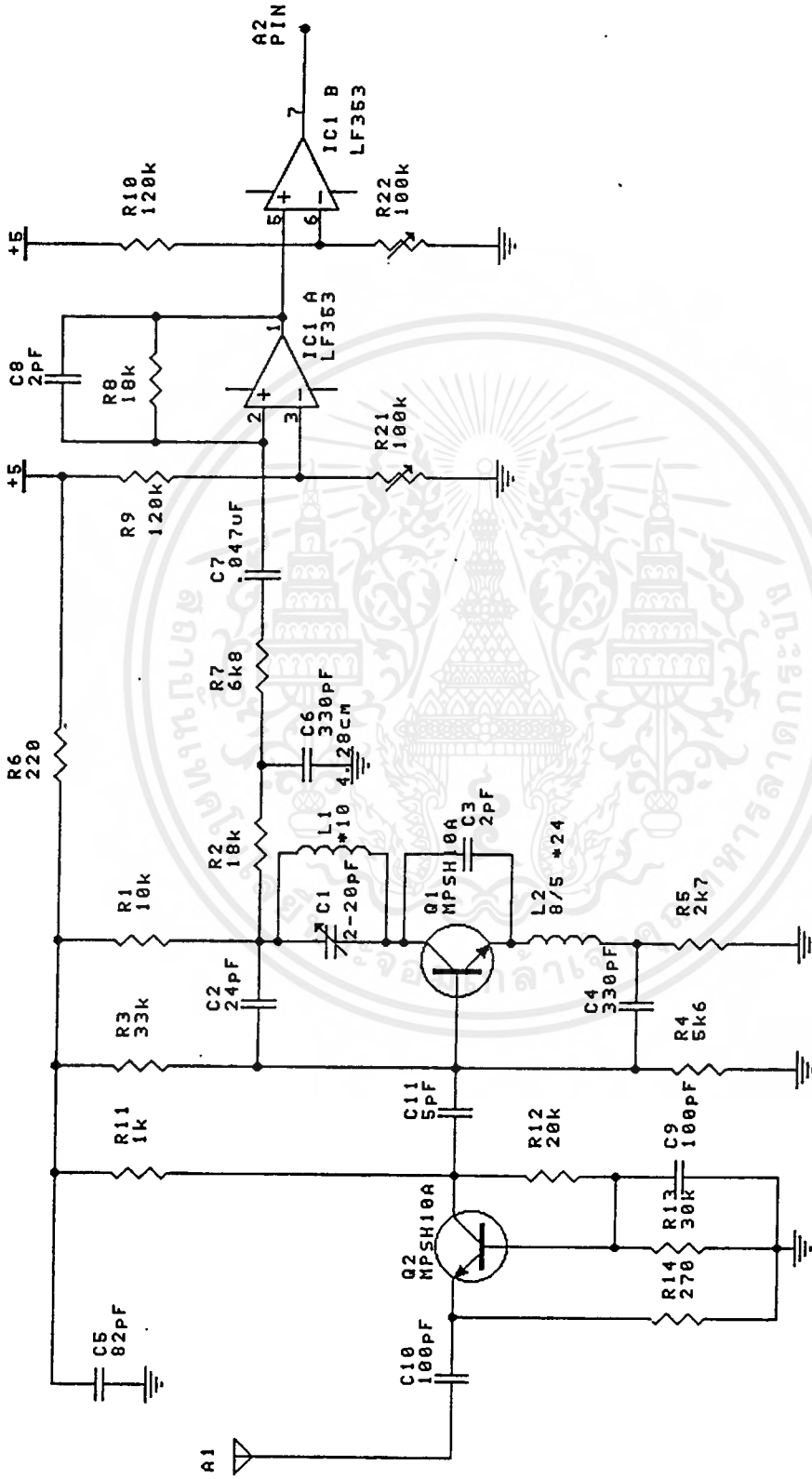
Note: • Although the printer adapter and MDA printer ports work identically, they appear at different port addresses.
 • The source contains incomplete material.

Source: IBM Options and Adapters Technical Reference, Vol. 2, pages Printer Adapter 3 through 7 and Monochrome Adapter 13 through 17



TRANSMITTER CIRCUIT

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



RECEIVER CIRCUIT

กิตติกรรมประกาศ

ขอขอบพระคุณ ผศ. ปราโมทย์ วาดเขียน ที่กรุณาให้ความช่วยเหลือ และให้คำปรึกษาในการแก้ปัญหาต่าง ๆ ทั้งยังสนับสนุนการทำปฏิญานีพนธ์นี้

ขอขอบคุณ คุณ ศิริศักดิ์ จังคศิริ , คุณพีเบียง , คุณพีบิล ที่ให้คำแนะนำเกี่ยวกับเทคนิค และให้ยืมอุปกรณ์ในการทำปฏิญานีพนธ์นี้เป็นอย่างดี



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เอกสารอ้างอิง

- [1] ชูชัย ธารสารตั้งเจริญ , ทินกร ดุ๊ก , “ การสื่อสารข้อมูล ” , สำนักพิมพ์ฟิสิกส์เซ็นเตอร์
- [2] ถวิล พึ่งมา รศ. , “ การออกแบบวงจรโทรคมนาคมความถี่สูง “ , ลาดกระบัง , พ.ศ. 2534.
- [3] ประสิทธิ์ ประพัฒมงคลการ ดร. , “ หลักการระบบสื่อสาร “ , ซีเอ็ดดูเคชั่น , พ.ศ. 2521.
- [4] Douglas V. Hall , “ Microprocessors and Interfacing “ , McGraw-Hill , 1986.
- [5] Fredick F. Driscoll , “ Data Communication “ , Saunders College Publishing , 1992.
- [6] Kim G. House , “ Printer Connection Bible “ , Howard W. Sam , 1989.
- [7] Stephen J. Bigelow , “ Computer Printers “ , McGraw-Hill , 1992.