ปีการศึกษา 2532

ปริญญานิพนธ์เรื่อง โมเด็ม

โดย

นาย ณัฏฐ์ คงสำราญ

นาย สุมลชัย รอดแฟง

นาย สาธิต พันธ์ไพศาล

อาจารย์ที่ปรึกษา

รศ.ประกิจ ตังติสานนท์

026996

22 พ.ย. 25..

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ปริญญานิพนธ์ ปีการศึกษา.. 2532

เรื่อง.. โมเด็ม

ผู้จัดทำ..

1.นาย ณัฏฐ์ คงสำราญ เลขที่ 29.1054

2.นาย สุมลชัย รอดแฟง เลขที่ 29.1252

3.นาย สาธิต พันธ์ไพศาล เลขที่ 29.1254

อาจารย์ที่ปรึกษา

รศ.ประกิจ ตังติสานนท์

(.............................)

026998

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โมเด็ม

ณัฏฐ์ คงสำราญ 29.1054
สุมลชัย รอดแฟง 29.1252
สาธิต พันธ์ไพศาล 29.1254
อาจารย์ที่ปรึกษา
รศ.ประชกิจ ตังติสานนท์
ปีการศึกษา 2532

บทคัดย่อ

ในยุคปัจจุบัน การสื่อสารข้อมูลของคอมพิวเตอร์ ได้ใช้งานกันอย่างแพร่หลายมาก อุปกรณ์สำคัญชิ้นหนึ่งที่สามารถทำให้ คอมพิวเตอร์ติดต่อสื่อสารข้อมูลกันได้ในระยะไกลก็คือโมเด็ม สามารถทำงานได้โดย โมเด็มจะเปลี่ยนสัญญาณดิจิตอลที่ส่งมาแบบอนุกรมให้เป็นสัญญาณซายน์ ที่มีความถี่เหมาะสมสำหรับตัวกลาง ซึ่งวิธีที่นิยมใช้กันทั่วไปรวมทั้งสำหรับโครงงานนี้ คือการสื่อสารข้อมูลโดยผ่านสายโทรศัพท์ โมเด็มที่ทำขึ้นสามารถส่งข้อมูลได้ด้วยอัตราเร็ว 300 บิตต่อวินาที ด้วยมาตราฐาน BELL 103 หรือ C.C.I.T.T. V21 มีการรับโทรศัพท์ได้โดยอัตโนมัติ สามารถส่งข้อมูลถึงกันได้ทั้งในลักษณะเป็นแฟ้มข้อมูลและพิมพ์จากแป้นพิมพ์ โมเด็มแต่ละตัวสามารถทำงานได้ทั้งในโหมดเริ่มติดต่อ (Original Mode) และโหมดตอบรับ (Answer Mode)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

MODEM

NAT KONGSAMRAN
SUMONCHAI RODFANG
SATHIT PANPAISAN
ADVISOR
ASSOCIATE PROFESSOR
PRAKIT TANGTISANON
1989

ABSTRACT

In this current era,Data Communication of Computer is used in many spread areas. The important equipment that be used is MODEM. MODEM will convert Digital Signal to Sine Signal that appropriate for media. The popular method is using telephone line.

This MODEM can send data at rate 300 bps by standard BELL 103 or C.C.I.T.T. V21. It has auto Telephone-Answer. It can send data in both file from diskette and character from keyboard. Each MODEM can work in Original Mode and Answer Mode also.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ

หน้าที่

บทที่ 1 บทนำ 1

บทที่ 2 ทฤษฎีพื้นฐาน 3

2.1 โมเด็ม 3

2.2 วิธีของการถ่ายโอนข้อมูล 12

2.3 พอร์ท RS 232C 23

2.4 การควบคุมการสื่อสารของคอมพิวเตอร์ 28

บทที่ 3 วงจรและการออกแบบ 31

3.1 บล็อคไดอะแกรมของโครงงาน 31

3.2 ดูเพล็กเซอร์ 32

3.3 วงจรกำเนิดสัญญาณนาฬิกา 34

3.4 ภาคควบคุมโหมดของฟิลเตอร์ 35

3.5 วงจรกรองสัญญาณ 36

3.6 ลิมิตเตอร์ 46

3.7 ภาคแคเรียร์ดีเทค 47

3.8 ภาคมอดดูเลทและดีมอดดูเลท 47

3.9 ภาครับโทรศัพท์อัตโนมัติ 53

3.10 ภาคแปลงระดับแรงดันสำหรับติดต่อกับ RS 232C PORT 55

3.11 ความรู้พื้นฐานสำหรับการเขียนโปรแกรมภาษาเบสิค ในการติดต่อกับพอร์ทสื่อสาร 56

บทที่ 4 การทดลองและผลการทดลอง 64

บทที่ 5 สรุปผลและวิจารณ์ 70

กิตติกรรมประกาศ

หนังสืออ้างอิง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 1

บทนำ

การติดต่อสื่อสารในปัจจุบัน คอมพิวเตอร์มีบทบาทสำคัญเป็นอย่างมาก เนื่องจากมีการใช้งานคอมพิวเตอร์กันอย่างแพร่หลาย รวมทั้งนับวันก็มีราคาถูกลงเรื่อยๆ ซึ่งเป็นสัดส่วนกลับกันกับประสิทธิภาพ ดังนั้นจะเป็นการเพิ่มประสิทธิภาพของการสื่อสารขึ้น ถ้าสามารถให้คอมพิวเตอร์ติดต่อสื่อสารหรือเชื่อมโยงข้อมูลกันได้ เรียกโดยทั่วไปว่า DATA COMMUNICATION และถ้าปรับปรุงต่อไปให้เป็นระบบเครือข่ายยิ่งจะมีประโยชน์สูงมาก แต่จะต้องใช้อุปกรณ์ที่มีประสิทธิภาพสูง และยังต้องมีการออกแบบอย่างสลับซับซ้อน

เราสามารถให้คอมพิวเตอร์เชื่อมโยงข้อมูลกันได้ ทำได้ 2 วิธีคือ 1.เชื่อมโยงกันโดยตรงส่งเป็นสัญญาณดิจิตอลออกไป แต่ว่าวิธีนี้ ถ้าต้องการส่งในระยะไกล จะต้องใช้สายส่งพิเศษซึ่งมีราคาสูงมาก 2. ส่งโดยแปลงเป็นสัญญาณอนาลอกก่อน โดยจะเปลี่ยนเป็นสัญญาณที่เหมาะสมกับตัวกลาง จากวิธีนี้ ถ้าเราสามารถนำไปส่งผ่านสายโทรศัพท์ ซึ่งมีระบบโครงข่ายอยู่อย่างกว้างขวางแล้วละก็ จะเป็นการลดค่าใช้จ่ายในการติดตั้ง และยังสามารถใช้งานในระยะทางไกลๆได้

การเพี้ยนของสัญญาณดิจิตอล เมื่อส่งผ่านสายโทรศัพท์

ระบบโทรศัพท์โดยทั่วไปนี้จะมีข้อจำกัดสำคัญ คือ สายโทรศัพท์จะถูกออกแบบขึ้นมาใช้สำหรับช่วงความถี่เสียง สำหรับสายโทรศัพท์จะมีช่วงความถี่ระหว่าง 300-3 KHz คือ จะไม่สามารถตอบสนองต่อสัญญาณ ความถี่สูงและความถี่ต่ำมากๆได้

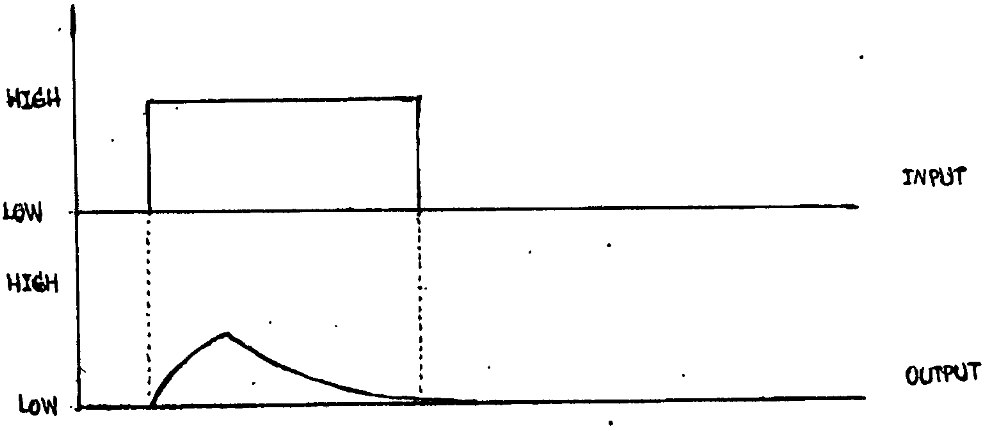
ผลทางด้านความถี่สูง

เมื่อมีการเปลี่ยนแปลงสัญญาณดิจิตอล จากศูนย์ไปหนึ่ง หรือ หนึ่งไปศูนย์ก็ตาม เนื่องจากเป็นการเปลี่ยนแปลงอย่างรวดเร็ว ทำให้ตอบสนองไม่ทัน ดังนั้นผลที่เกิดขึ้น จะเป็นดังรูป 1.1

ผลทางด้านความถี่ต่ำ

เมื่อข้อมูลมีการคงสภาวะนานๆโดยเฉพาะอย่างยิ่งเมื่อเป็น "high" ดังนั้นกล่าวได้ว่า สัญญาณขณะนี้มีความถี่เท่ากับ ศูนย์ แต่ว่าสายโทรศัพท์ที่ใช้กันทั่วไป จะไม่สามารถตอบสนองต่อสัญญาณไฟตรงได้ ผลที่เกิดขึ้นจะเป็นดังรูป 1.1 ทำให้เกิดการผิดพลาดในการรับส่งได้

เอกสารนี้ถือเป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 1.1 แสดงลักษณะสัญญาณที่เพี้ยนไป เมื่อผ่านสายโทรศัพท์

ในการใช้งานจะทำการเปลี่ยนสัญญาณดิจิตอล เป็นสัญญาณซายน์ที่มีความถี่ อยู่ในช่วงแบนวิดท์ของสายโทรศัพท์ เรียกว่าการมอดดูเลท (MODULATION) และฝ่ายรับจะแปลงสัญญาณนั้นกลับมาเป็นสัญญาณดิจิตอลดังเดิม เรียกว่าการดีมอดดูเลท (DEMODULATION) อุปกรณ์ชิ้นนี้เรียกว่า "MODEM" ซึ่งเป็นคำที่นำส่วนหนึ่งของ 2 คำมาประกอบด้วยกันคือ "MO" จาก "MODULATION" และ "DEM" จาก "DEMODULATION"

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 2
ทฤษฎีพื้นฐาน

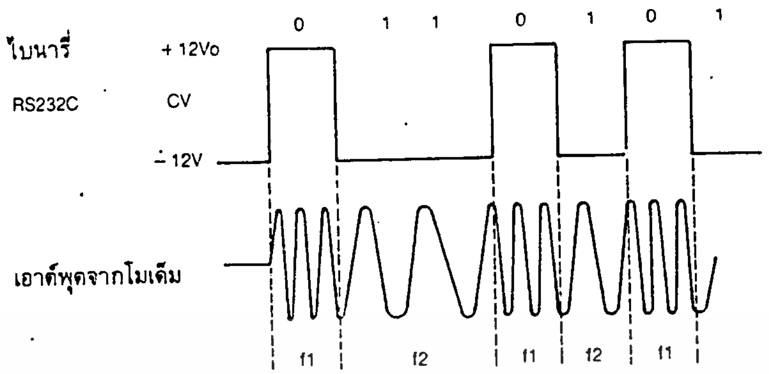
2.1 โมเด็ม

โมเด็มย่อมาจาก Modulator Demodulator ใช้ในการแปลงสัญญาณทางลอจิกให้เหมาะสมก่อนที่จะส่งผ่านตัวกลางที่มีความกว้างของแถบคลื่นต่ำๆ อย่างเช่น สายโทรศัพท์ ทำไมสัญญาณทางลอจิกจึงส่งออกไปโดยตรงไม่ได้ อย่าลืมว่าสัญญาณลอจิกมีลักษณะเป็นคลื่นสี่เหลี่ยม (Square wave) "0" และ "1" ซึ่งอาจจะแทนด้วยค่าของแรงดันสองค่า คลื่นรูปสี่เหลี่ยมประกอบด้วยรูปคลื่นรูปซายน์หลายความถี่ เป็นทวีคูณของความถี่พื้นฐาน หากผ่านตัวกลางที่มีแถบความกว้างของคลื่นต่ำแล้วความถี่สูงๆ ก็จะหายไป เหลือสัญญาณที่ปลายทางผิดเพี้ยนไปจากเดิม โดยเฉพาะโทรศัพท์เขาออกแบบให้ใช้กับการสื่อสารที่เป็นเสียงมนุษย์เท่านั้น แถบความกว้างของคลื่นแค่ 3 กิโลเฮิรตซ์เท่านั้นจำเป็นที่เราจะต้องเปลี่ยนสัญญาณลอจิกให้อยู่ในรูปแบบที่เหมาะสมก่อนที่จะส่งออกไป (ผจญกับโลกภายนอก) ข้างฝ่ายรับก็จำเป็นต้องเปลี่ยนสัญญาณที่ถูกแปลงมานี้กลับให้เป็นสัญญาณทางลอจิก และก็จะต้องมีขบวนการที่ตรงกันข้ามกับการฝ่ายส่ง อุปกรณ์ที่ทำหน้าที่ทั้งสองอันนี้เรียกว่า โมเด็ม (MODEM)

2.1.1 การทำงานของโมเด็ม

แรกเริ่มทีเดียวการแปลงสัญญาณลอจิก ให้เหมาะสมกับการส่งผ่านไปในสายโทรศัพท์ใช้วิธีการที่เรียกว่า Frequency Shift keying คือ ใช้ความถี่ของเสียงสองความถี่สำหรับแทนสัญญาณลอจิก "0" และ "1" ฝ่ายรับก็พยายามจับเอาสองความถี่ที่ว่านี้มาแปลงเป็นสัญญาณลอจิกกลับคืน ความถี่ของเสียงทั้งสองเสียงต้องห่างกันพอที่จะแยกออกจากกันได้โดยวงจรอิเล็กทรอนิคส์ และก็จะต้องไม่ห่างเกินจนตกขอบของความสามารถของสายโทรศัพท์จะนำพาไปได้ รูปที่ 2.1 แสดงถึงหลักการทำงานของ Frequency shift Keying เรียกย่อๆ ว่า FSK

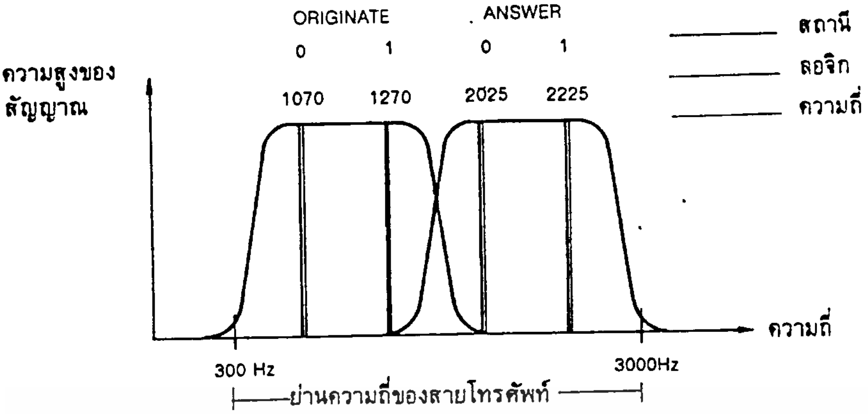
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



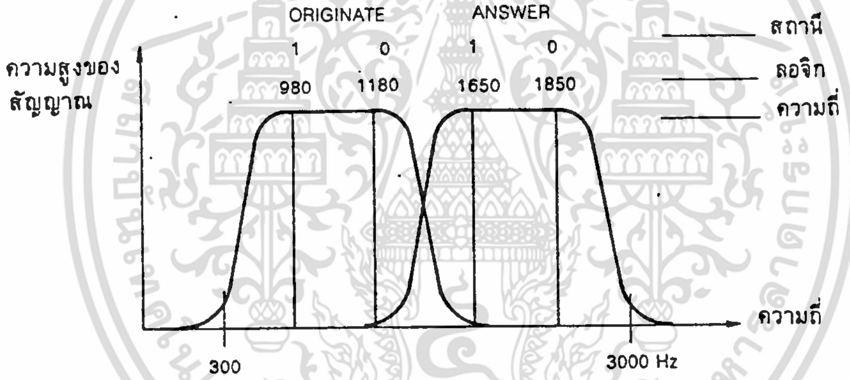
รูปที่ 2.1 การแปลงสัญญาณของโมเด็ม

เนื่องจากแถบความถี่คลื่นที่สายโทรศัพท์ยอมให้ผ่านไปได้อยู่ในช่วง 300 Hz ถึง 3000 Hz เราสามารถแบ่งความถี่ในย่านนั้นออกเป็น 4 คลื่นเสียงที่สำคัญ สำหรับสถานีส่งสองเสียงและสถานีรับสองเสียง เนื่องจากเราต้องการให้การติดต่อเป็นฟูลดูเพล็กซ์ คือ ทั้งรับและส่งได้ในเวลาเดียวกัน จำเป็นจะต้องแยกสถานีออกเป็นสองฝ่าย ฝ่ายหนึ่งเรียกว่า originate หรือฝ่ายเริ่มการติดต่อ และอีกฝ่ายเรียกว่า answer ฝ่าย originate จะใช้ความถี่สำหรับส่งสองความถี่สำหรับสัญญาณลอจิก "0" และ "1" ฝ่าย answer จะต้องใช้ความถี่อีกสองความถี่ที่แตกต่างไปจากฝ่ายส่ง (เพื่อป้องกันการรบกวนกันเอง) สำหรับแทนสัญญาณลอจิก "0" และ "1" จะได้รับและส่งในเวลาเดียวกันเป็นฟูลดูเพล็กซ์ได้ คราวนี้ก็มาถึงปัญหาว่า ความถี่ไหนล่ะที่ใช้กันอยู่ ก็มีมาตรฐานอยู่สองแห่งที่ใช้กัน คือ ของระบบ CCITT และของบริษัทเบลเทลีโฟนสำหรับโมเด็มที่มีความเร็วไม่เกิน 300 บอด ห้องปฏิบัติการวิจัยเบลใช้มาตรฐาน 103 ส่วน CCITT ใช้มาตรฐานที่ชื่อว่า V.21 ดังแสดงในรูปที่ 2.2 และ 2.3

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.2 มาตรฐานของโมเด็ม 103 โมเด็มแบบฟูลดูเพล็กซ์ อัตราส่งต่ำ



รูปที่ 2.3 มาตรฐานของ CCITT V.21 โมเด็มแบบฟูลดูเพล็กซ์ อัตราส่งต่ำ

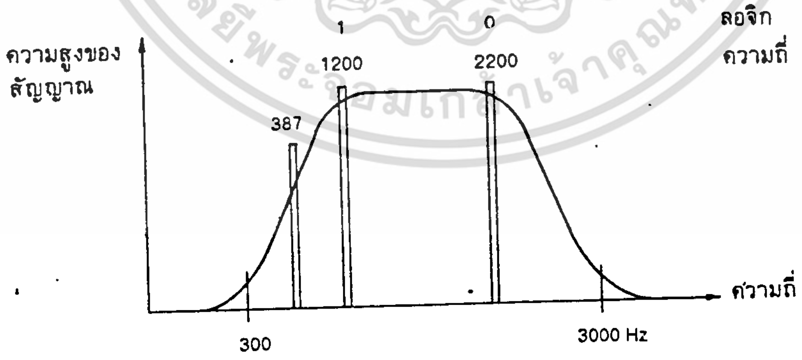
จากรูปที่ 2.2 และ 2.3 จะพบว่า สถานีรับและสถานีส่งใช้ความถี่ต่างกันใน การมอดดูเลต สัญญาณลอจิก "0" และ"1" การดีมอดดูเลตก็จะต้องให้ตรงกับความถี่ของฝ่ายตรงกันข้ามส่งมา ยกตัวอย่างเช่น โมเด็มชนิด 103 ถ้าหากใช้เป็นผู้ริเริ่มการติดต่อ (ซึ่งโดยมากผู้ใช้ไมโครคอมพิวเตอร์เวลาติดต่อกับฝ่ายคอมพิวเตอร์ขนาดใหญ่ จะเป็นฝ่ายที่เรียกว่า originate) จะส่งสัญญาณลอจิก "1" ด้วยความถี่ 1270 Hz ลอจิก "0" ด้วยความถี่ 1070 Hz ขณะเดียวกันจะต้องรับด้วยความถี่ 2025 Hzและ 2225 Hz ทั้งรับและส่งของโมเด็มก็จำเป็นต้องทีวงจรกรองความถี่ เพื่อป้องกันความถี่อื่นหลงเข้ามารบกวนเครื่องรับ วงจรกรองความถี่ที่จะต้องแยกแยะความถี่ของฝ่ายรับและส่งออกจากกัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โมเด็มชนิด 103 เป็นมาตรฐานที่ใช้กันอยู่ในอเมริกา ส่วน CCITT ใช้กันเกือบทั่วโลก รวมทั้งเมืองไทยด้วย

เนื่องจาก ความถี่ของเสียงที่ใช้มีความถี่ต่ำการมอดูเลตแบบ FSK ย่อมทำให้การถ่ายโอนข้อมูลเร็วกว่าความถี่นั้นไม่ได้แน่นอน เนื่องจากวงจรรับจะต้องดีเทกให้ได้ว่ามีความถี่เปลี่ยนแปลงเกิดขึ้น อย่างน้อยความถี่จะต้องปรากฏให้เห็น 2 ถึง 3 ไซเคิลเป็นอย่างน้อย ลองคำนวณแต่ง่ายๆ ความถี่ต่ำสุดที่ใช้ในโมเด็มชนิด 103 คือ 1070 Hz ต้องใช้อย่างน้อย 2 ไซเคิล ต่อการมอดูเลต 1 บิต จะเห็นว่าการถ่ายโอนข้อมูลจะไปกว่า 600 บิตต่อวินาทีได้ยาก

ถ้าเราใช้เทคนิค FSK เหมือนเดิมแต่แยกความถี่ของสองเสียงที่ใช้แทน "0" และ "1" ให้ห่างกัน จำนวนไซเคิลที่ใช้มอดูเลตก็จะน้อยลง เราจะแยกความถี่ให้ห่างกันได้ก็ต้องส่งได้ทีละข้างหรือเป็นแบบฮาล์ฟดูเพล็กซ์ ระบบเบล 202 ใช้เทคนิคอันนี้ในการส่งข้อมูลด้วยความเร็ว 1200 บอด แบบฮาล์ฟดูเพล็กซ์โดยใช้ความถี่ 1200 แทนมาร์ก และ 2200 แทนสเปซ และเพื่อเป็นการประกันว่าฝ่ายรับกำลังรับอยู่ ฝ่ายรับจะส่งความถี่ 387 Hz ควบกลับมาให้รู้ว่า "ฉันกำลังฟังอยู่" บางครั้งความถี่ 387 Hz นี้อาจจะใช้ในการบอกฝ่ายส่งว่า ข้อความที่ส่งมามีข้อผิดพลาดอยู่กรุณาส่งมาใหม่ รูปที่ 2.4 แสดงสเปกตรัมของโมเด็มชนิด 202



รูปที่ 2.4 โมเด็มชนิดเบล 202 ฮาล์ฟดูเพล็กซ์

สำหรับ CCITT หรือที่ใช้กันในประเทศสากล มาตรฐานจะเป็นโมเด็มแบบ V.23 ซึ่งจำลองมาจากเบล 202 แต่จะต่างกันตรงที่มีโหมดให้เลือก 2 โหมด คือ 600 บอด และ 1200 บอด โดยทั้งสองโหมดใช้ความถี่ต่างกัน คือ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

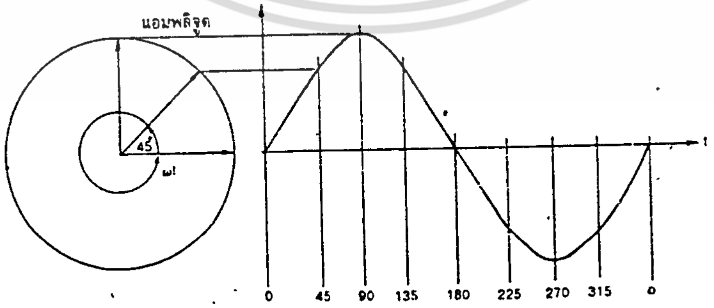
	มาร์ก	สเปซ
โหมด 1 (600 บอด)	1300 Hz	1700 Hz
โหมด 2 (1200 บอด)	1300 Hz	2100 Hz

นอกเหนือไปจากนั้น V.23 ยังสามารถให้ฝ่ายรับส่งข้อมูลกลับมาได้ด้วยความเร็ว 75 บอด โดยใช้ FSK จริงๆ 390 Hz แทนมาร์ก และ 450 Hz แทนสเปซ ในกรณีเช่นนี้เหมาะสำหรับการติดต่อกับเทอร์มินัลที่ผู้ใช้ป้อนข้อมูลทางคีย์บอร์ดเนื่องจากความเร็วของการพิมพ์ของมนุษย์คงไม่มีใครทำได้เกิน 100 คำต่อนาที เป็นแน่ (ลองคำนวณดูคร่าวๆ) 1 คำใช้ ใช้ 4 ตัวอักษร อักษร 1 ตัว ใช้ 8 บิต บวกสตาร์ตบิตอีก 2 บิต รวมแล้ว 200 คำต่อนาที เทียบเท่ากับ 40 x 100 หาร 60 = 66 บิตต่อวินาที

2.1.2 โมเด็มความเร็วสูง

ในการถ่ายโอนที่ต้องใช้ความเร็วสูง การมอดูเลตโดย FSK เห็นจะไม่ไหวเลย เปลี่ยนมาใช้วิธีการที่เรียก PSK หรือ Phase Shift Keying แทนที่จะใช้ความถี่ในการแทนสัญญาณลอจิก กลับใช้สัญญาณเสียงความถี่เดียว แต่ใช้เฟสต่างกันออกไปสำหรับแทนสัญญาณลอจิก

การใช้เฟสในการมอดูเลตเป็นอย่างไร มาลองดูรูปร่างของสัญญาณอย่างง่ายๆ ตามรูปที่ 2.5

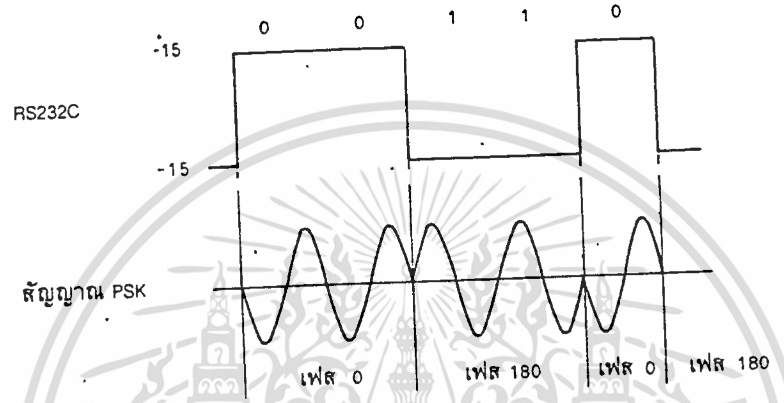


รูปที่ 2.5 การกำหนดคลื่นรูปซายน์และเฟสต่างๆ

การกำหนดของคลื่นรูปซายน์ก็เหมือนกับการหมุนของเข็มนาฬิกา ไปเป็นเส้นรอบวง ถ้าเราวัดความสูงของเข็มนาฬิกาเทียบกับแนวนอนขณะใดขณะหนึ่ง แล้วนำมาพล็อตเทียบกับ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

แกนเวลาเราจะได้รูปร่างของคลื่นรูปซายน์ เข็มที่เราใช้หมุนเรียกว่าเวกเตอร์ มุมที่หมุนไปเรียกว่า เฟส ฉะนั้นเฟสของสัญญาณรูปคลื่นซายน์จะมีตั้งแต่ 0 ถึง 360 องศา ถ้าหากเราจะเลือกใช้เฟสในการมอดูเลตสัญญาณลอจิกเราก็จะแบ่งเฟสที่เราจะใช้ออกเป็น 2 เฟส ในการมอดูเลตสัญญาณลอจิก "0" และ "1" คือใช้ เฟส 0 แทน 0 และ เฟส 180 องศา แทน 1 ลักษณะของสัญญาณจากโมเด็มก็จะเป็นดังรูปที่ 2.6



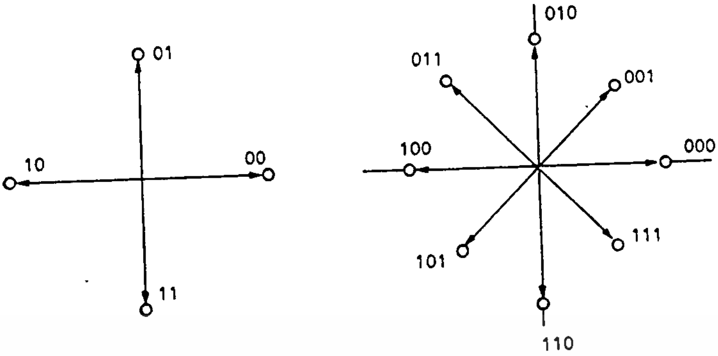
รูปที่ 2.6 สัญญาณ PSK

ถ้าหากเราแบ่งสัญญาณ PSK ออกเป็น 4 เฟส คือ 0, 90, 180 และ 270 องศา โดยเราแทนเฟสทั้ง 4 เฟสด้วยเลขฐานสอง สองหลักหรือสองบิตในกรณีเช่นนี้การเปลี่ยนเฟสครั้งหนึ่งเท่ากับว่าเราได้ข้อมูล 2 บิตเข้าไปแล้ว ในลักษณะนี้อัตราบิตจะเป็น 2 เท่าของอัตราบอด เพราะอัตราบอด คืออัตราการเปลี่ยนแปลงสัญญาณใน 1 วินาที แต่การเปลี่ยนแปลงสัญญาณ 1 ครั้ง ข้อมูลเปลี่ยนแปลง 2 บิต ความเร็วในการถ่ายโอน จึงเป็น 2 เท่าของอัตราบอด

โมเด็มชนิด 201 B ของเบล ใช้ Phase Shift Keying โดยการแบ่งเฟสเป็น 4 เฟสดังกล่าว อัตราในการส่ง 1200 บอด เท่ากับได้ความเร็วในการถ่ายโอนข้อมูล 2400 บิตต่อวินาที

ถ้าเราแบ่งเฟสของสัญญาณออกเป็น 8 คือ 0, 45, 90, 125, 180, 225, 270 และ 315 โดยแต่ละเฟสแทนได้ด้วยข้อมูล 3 บิต จะเห็นว่าความเร็วในการถ่ายโอนข้อมูลจะกลายเป็น 3 เท่าของอัตราบอด ถ้าใช้อัตราบอด 1600 ก็จะได้ความเร็วของการถ่ายโอนข้อมูล 4800 บอดต่อวินาที โมเด็มชนิด 201C ของบริษัทเบลล์ใช้เทคนิคทำการถ่ายโอนข้อมูลทำได้เร็วถึง 4800 บิตต่อวินาที

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.7 การมอดูเลตโดยวิธี PSK ของ Bell 201

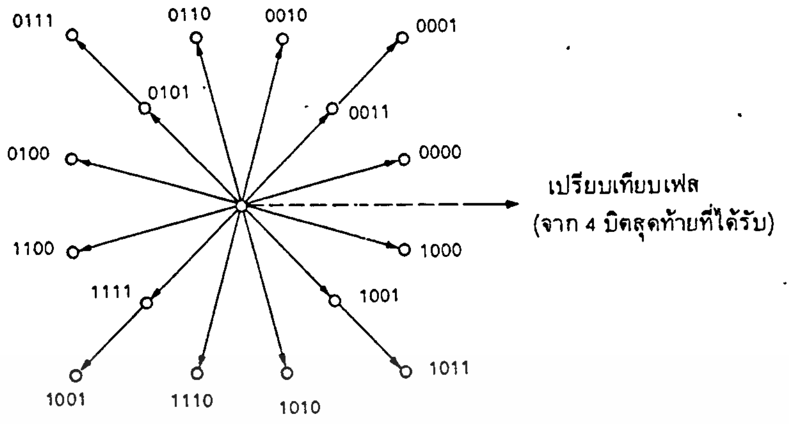
คิดดูง่ายก็จะเห็นว่าถ้าเราแบ่งเฟสของสัญญาณออกเป็น 2^N เฟส เราจะได้ความเร็วในการถ่ายโอนข้อมูลเป็น N เท่าของอัตราบอด ความจริงไม่ง่ายอย่างนั้น เนื่องจากถ้าเฟสเข้ามาใกล้กันมาก การแยกสัญญาณออกจะทำได้ยากมาก N=3 หรือ 8 เฟส ก็นับว่าเต็มกลืน ถ้า N=4 ก็คงจะเป็น 16 เฟส โอกาสที่จะแยกสัญญาณอย่างผิดพลาดคงจะมีแน่ๆ

วิธีการที่จะเพิ่มความเร็วในการถ่ายโอนให้สูงขึ้นก็โดยการเอาความสูงหรือ แอมปลิจูดของสัญญาณเข้ามามอดูเลตด้วยเรียกว่า Quardrature Modulation โมเด็มที่ส่งด้วยความเร็ว 9600 บิตต่อวินาที เขาแบ่งเฟสออกเป็น 12 เฟส มีอยู่ 4 เฟส ที่มีโอกาสมีแอมพลิจูดได้สองค่ารวมแล้วทั้งหมดสามารถใช้เลขฐานสอง 4 บิตแทนเฟสและแอมพลิจูดทั้ง 16 สถานภาพสัญญาณในสายใช้ความเร็ว 2400 บอดก็จะสามารถให้ความเร็วในการถ่ายโอนได้ถึง 9600 บิตต่อวินาที รูปที่ 2.8 แสดงไดอะแกรมของสัญญาณ

12 พ.ย. 535

026998

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.8 เฟสของสัญญาณในการส่งสัญญาณด้วยความเร็ว 9600 บิตต่อวินาที

นอกจากมาตรฐานของโมเด็มที่ได้กล่าวมายังมีมาตรฐานออกมาใหม่ๆ อีกหลายอย่าง หากท่านต้องการจะซื้อโมเด็มใช้ อย่าลืมถามฝ่ายตรงข้ามที่ท่านจะติดต่อสื่อสารข้อมูลก่อนว่าเขาใช้โมเด็มชนิดใด ท่านจะไม่สามารถติดต่อกับเขาได้อย่างแน่นอนถ้าหากเป็นโมเด็มชนิดที่ต่างกัน

ตารางที่ 2.1 ชนิดของโมเด็มและวิธีการมอดุเลต

ชนิด	ความเร็ว	วิธีการมอดุเลต	อัตราบอด	ดูเพล็กซ์
103	300 bps	FSK	300 baud	Full/FDM
202	1200 bps	FSK	300 baud	HALF
212	1200 bps	DPSK	600 baud	Full/FDM
V.22	1200 bps	DPSK	600 baud	Full/FDM
201	2400 bps	DPSK	1200 baud	Half
V.22bis	2400 bps	QAM	600 baud	Full/FDM
V.26ter	2400 bps	DPSK	1200 baud	Full/ECT
208	4800 bps	DPM	1600 baud	Half

ตารางที่ 2.1 (ต่อ)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ชนิด	ความเร็ว	วิธีการมอดูเลต	อัตราบอด	ดูเพล็กซ์
209	9600 bps	QAM	2400 baud	Half
V.29	9600 bps	MQAM	2400 baud	HALF
V.32	4800 bps	QAM	2400 baud	Full/ECT
V.32	9600 bps	MQAM	2400 baud	Full/ECT

DPM = Differential phase Modulation

FSK = Frequency Shift Keying

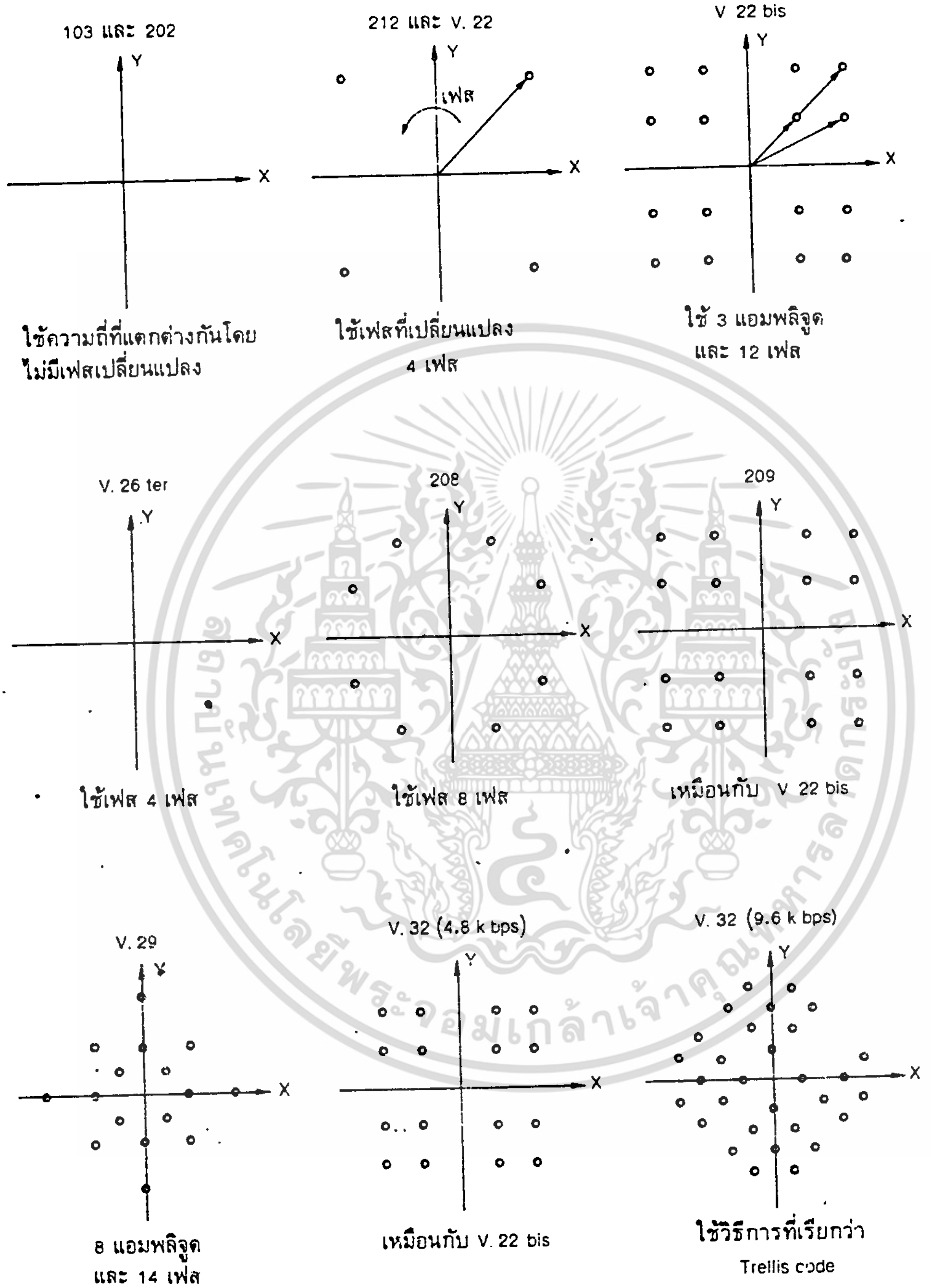
FDM = Frequency Division multiplex

DPSK = Different Phase Shift Keying

QAM = Quadrature Amplitude Modulation

ECT = Echo Cancellation Technique

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.9 วิธีการมอดุเลตแบบต่างๆ

2.2 วิธีของการถ่ายโอนข้อมูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

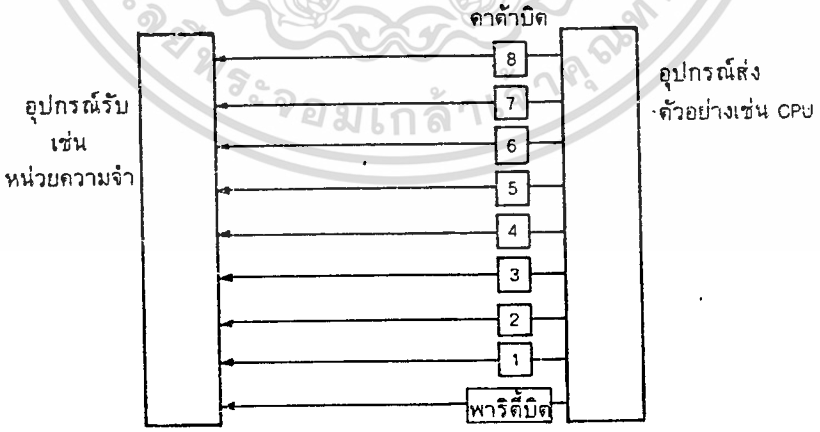
ก่อนที่เราจะกล่าวถึงการสื่อสารระหว่างเครื่อง เรามามองดูก่อนว่าสัญญาณที่ส่งออกมาจากเครื่องและรับเข้าไปในเครื่องเป็นอย่างไร

1. การถ่ายโอนข้อมูลแบบขนาน

ลักษณะของการส่งข้อมูลแบบขนาน ทำได้โดยการส่งข้อมูลออกมาทีละ 1 ไบต์ คือ 8 บิต จากอุปกรณ์ส่งไปยังอุปกรณ์รับ ตัวกลางระหว่าง 2 เครื่องจะต้องมีช่องทางให้ข้อมูลเดินทางอย่างน้อย 8 ช่องทาง โดยมากจะเป็นสายขนานให้กระแสไฟฟ้าวิ่งมากกว่าจะเป็นตัวกลางชนิดอื่น เนื่องจากมีสัญญาณสูญหายไปกับความต้านทางของสาย ระยะทางระหว่าง 2 เครื่องไม่ควรจะเกิน 100 ฟุต ปัญหาที่เกิดขึ้นหากระยะทางของสายมากกว่านี้ก็คือ ระดับของกราวนด์ในทางไฟฟ้าที่จุดรับผิดไปจากจุดส่ง ทำให้เกิดการผิดพลาดในการรับสัญญาณลอจิกทางฝ่ายรับ

นอกจากสายที่เป็นทางเดินของข้อมูลแล้วอาจจะมีทางเดินของสัญญาณควบคุมอื่นๆ อีก เป็นต้นว่า บิตที่บอกพาริตี้ของสัญญาณ เพื่อเป็นการตรวจสอบความผิดพลาดของการรับสัญญาณที่ปลายทาง หรือสายที่ควบคุมการโต้ตอบ (Hand-shake) ดังที่กล่าวมาแล้ว

จะเห็นว่าการส่งแบบขนานส่วนมากจะทำในระยะใกล้ๆ เนื่องจากจะต้องมีช่องทางเดินของสัญญาณมากกว่า 8 สาย และอุปกรณ์ที่ติดต่อแบบขนานกับคอมพิวเตอร์ก็เห็นจะได้แก่เครื่องพิมพ์ดังกล่าวมาแล้ว



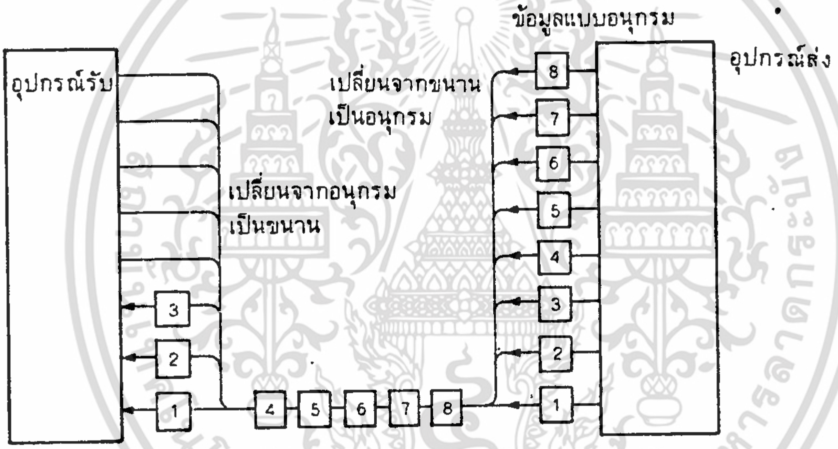
รูปที่ 2.10 การส่งข้อมูลแบบขนาน

2. การถ่ายโอนข้อมูลแบบอนุกรม

ในการถ่ายโอนข้อมูลแบบอนุกรม ข้อมูลถูกส่งออกมาทีละบิต ระหว่างจุดส่งและจุดรับ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จะเห็นว่าการส่งข้อมูลแบบนี้จะช้ากว่าแบบขนานที่กล่าวมาแล้วแน่นอน แล้วทำไมต้องส่งแบบนี้ละ คำตอบก็คือ ตัวกลางการสื่อสารต้องการเพียงช่องเดียวหรือสายเพียงคู่เดียว ค่าใช้จ่ายในสื่อกลางจะต้องถูกกว่าแบบขนานอย่างแน่นอน สำหรับการส่งระยะทางไกลๆ โดยเฉพาะเมื่อเรามีระบบการสื่อสารทางโทรศัพท์ไว้ใช้งานอยู่แล้วย่อมจะเป็นการประหยัดกว่าที่จะทำการติดต่อสื่อสารทีละ 8 ช่อง เพื่อการถ่ายโอนข้อมูลแบบขนานอย่างแน่นอน



รูปที่ 2.11 การส่งข้อมูลแบบอนุกรม

รูปที่ 2.11 แสดงให้เห็นการส่งข้อมูลแบบอนุกรม ข้อมูลจากจุดส่งจะถูกเปลี่ยนให้เป็นอนุกรมเสียก่อน แล้วค่อยทยอยส่งออกทีละบิตไปยังจุดรับ ณ ที่จุดรับจะต้องมีกลไกในการเปลี่ยนข้อมูลที่ส่งมาทีละบิต ให้เป็นสัญญาณแบบขนานซึ่งลงตัวพอดี นั่นคือ บิต 1 ลงที่บัสข้อมูลเส้นที่ 1 พอดี การที่จะทำให้การแปลงสัญญาณจากอนุกรมทีละบิตให้ลงพอดีนั้นจำเป็นจะต้องมีกลไกลที่เหมาะสมเพื่อป้องกันการผิดพลาดในการรับ กลไกที่ว่านี้แบ่งเป็น 2 แบบ คือ

1. การสื่อสารแบบซิงโครนัส

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2. การสื่อสารแบบอะซิงโครนัส

รายละเอียดของทั้งสองแบบจะกล่าวในหัวข้อถัดไป

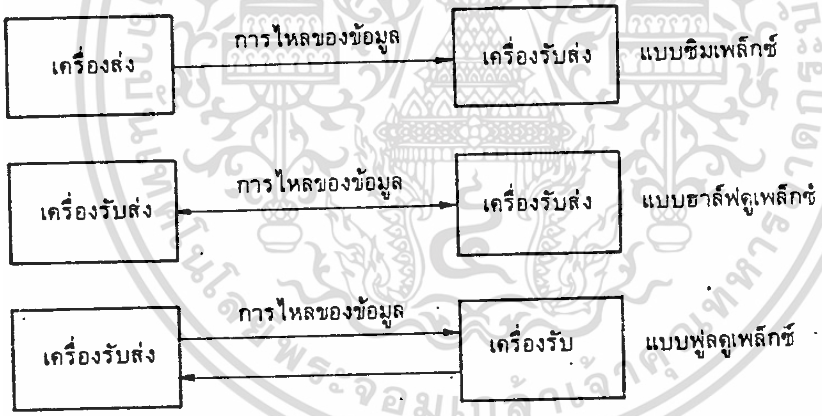
2.2.1 รูปแบบของการติดต่อสื่อสารแบบอนุกรม

การติดต่อแบบอนุกรมอาจจะแบ่งตามรูปลักษณะได้ 3 แบบ ตามรูปที่ 2.11

1. แบบซิมเพล็กซ์ (simplex) ข้อมูลส่งได้ในทางเดียวเท่านั้น บางครั้งก็ เรียก ว่า การส่งทิศทางเดียว (Unidirectional data bus)

2. แบบฮาล์ฟดูเพล็กซ์ (half duplex) ข้อมูลสามารถส่งได้ทั้งสองสถานี แต่จะต้องผลัดกันส่งและผลัดกันรับ จะส่งและรับพร้อมกันไม่ได้

3. แบบฟูลดูเพล็กซ์ (full duplex) ทั้งสองสถานีสามารถรับและส่งได้ในเวลาเดียวกัน



รูปที่ 2.12 รูปแบบของการติดต่อสื่อสารข้อมูลแบบอนุกรม

การส่งแบบฟูลดูเพล็กซ์และฮาล์ฟดูเพล็กซ์ ไม่ขึ้นอยู่กับจำนวนของสายในการติดต่อ บางครั้งคำว่า ทูไวร์ (two wire) หรือสองเส้น และโฟร์ไวร์ (four wire) หรือ 4 สี่เส้น ใช้ในการบรรยายถึงลักษณะการสื่อสารข้อมูลซึ่งอาจจะทำให้เข้าใจและฮาล์ฟดูเพล็กซ์สายโทรศัพท์ทั่วไปเป็นแบบ 2 เส้น ส่วนในสายที่เป็นแบบเช่า (lease line) นั้นส่วนมากจะเป็น 4 เส้น

2.2.2 ความเร็วในการถ่ายโอนข้อมูลแบบอนุกรม

ความเร็วของการถ่ายโอนข้อมูลแบบอนุกรม หน่วยวัดเป็นบิตต่อวินาที (bps)

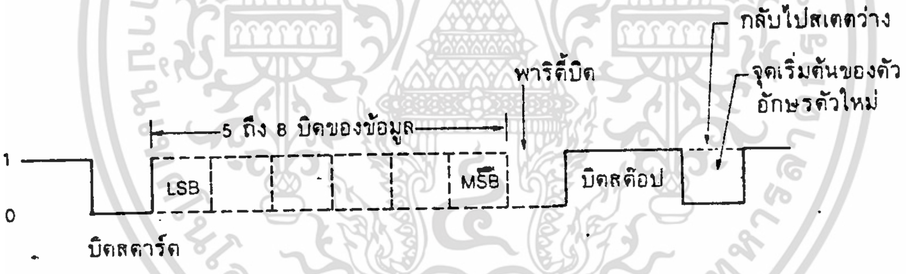
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หน่วยที่บรรยายถึงการเปลี่ยนแปลงของสัญญาณใน 1 วินาที เรียกว่าบอดเรต (baud rate) หรืออัตราบอด หลายคนยังเข้าใจสับสนระหว่างอัตราบอด และอัตราบิต (bit rate) การเปลี่ยนแปลงของสัญญาณ 1 ครั้ง อาจจะแสดงถึงการส่งข้อมูลแบบอนุกรมมากกว่า 1 บิต (รายละเอียดเกี่ยวกับเรื่องนี้จะได้กล่าวถึงอีกครั้งในเรื่องของโมเด็ม) ถ้าเขียนในรูปของสมการทางคณิตศาสตร์เราก็จะได้

อัตราบิต (bit rate) = อัตราบอด (baud rate) x บิตใน 1 บอด

2.2.3 การสื่อสารแบบอะซิงโครนัส

การส่งแบบอะซิงโครนัสนี้ พัฒนามาจากการส่งโทรพิมพ์ในสมัยก่อน ลักษณะของสัญญาณแสดงไว้ในรูปที่ 2.13 เพื่อเพิ่มกลไกในการรับส่งอย่างถูกต้อง สัญญาณอะซิงโครนัส จะประกอบด้วยบิตเริ่มต้นหรือบิตสตาร์ต (start) และบิตสิ้นสุดหรือบิตสต๊อป (stop bit)



รูปที่ 2.13 ฟอร์แมตการสื่อสารแบบอะซิงโครนัส

ขณะที่สถานะของการส่งเป็นแบบว่าง (Idle) คือ ยังไม่มีสัญญาณส่งออกมา จะมีสัญญาณหรือมีแรงดัน (หรือกระแส) ตลอดเวลา เพื่อความแน่ใจว่าฝ่ายรับยังติดต่ออยู่กับฝ่ายส่ง เมื่อเริ่มจะส่งข้อมูล สัญญาณของอะซิงโครนัสจะเป็น 0 หนึ่งช่วงสัญญาณนาฬิกา บิตนี้เรียกว่าสตาร์ทบิต ตามหลังของสตาร์ทบิตก็จะเป็นข้อมูลสำหรับ 1 ตัวอักษร ซึ่งอาจจะมีขนาดตั้งแต่ 5 บิต จนถึง 8 บิต โดยบิตที่มีค่าน้อยที่สุด (LSB) จะถูกส่งออกมาก่อนไล่ไปจนถึงบิตที่มีค่ามากที่สุด (MSB) การเข้ารหัสอักขระนี้ส่วนมากจะนิยมใช้รหัส ASCII แรกเริ่มที่เดียวในงานของโทรพิมพ์เขาใช้รหัส Baudot ซึ่งใช้ 5 บิต ในการแทนอักขระ 1 ตัว ตามหลังข้อมูลก็จะเป็นพาริตี้บิต ซึ่งอาจจะใช้หรือไม่ใช้ก็ได้ พาริตี้บิตทำหน้าที่เป็นตัวตรวจสอบความถูกต้องของสัญญาณที่ได้รับ พาริตี้บิตอาจจะเป็นแบบ คู่ (Even) หรือแบบ คี่ (Odd) หมายความว่าถ้าหากเป็นพาริตี้คู่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่าการนำไปใช้ทั้งหมดหรือบางส่วน อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จำนวนบิตที่เป็น 1 ในช่วงบิตข้อมูลกับบิตพาริตี้รวมแล้วจะต้องเป็นจำนวนคู่ ผู้ส่งจะต้องทำหน้าที่ตรวจสอบข้อมูลแล้วใส่พาริตี้บิตเอง ฝ่ายรับเมื่อรับแล้วก็จะต้องตรวจสอบดูว่าเป็นจริงดังสถานการณ์ที่ตั้งเอาไว้หรือไม่ หากผิดพลาดก็หมายความว่าสัญญาณที่รับนั้นผิดพลาดไปจากสถานีส่งส่งออกมา ทั้งนี้ทั้งนั้นจะต้องผิดเป็นจำนวนคี่เท่านั้น คือผิดไป 1 บิต 3 บิต หรือ 5 บิต พร้อมกันจึงจะตรวจสอบได้ว่าผิด มองเห็นง่ายๆ ว่าถ้าผิดเป็นจำนวนคู่ ผลรวมของจำนวนหนึ่งก็ยังเป็นคู่อยู่ดี ทั้งนี้ทั้งนั้นไม่ได้หมายความว่าพาริตี้คี่ (Odd parity) จะตรวจสอบการผิดพลาดเป็นจำนวนคี่ความจริงแล้วตรวจสอบดูว่าสัญญาณที่รับเข้ามีจำนวนคู่ ก็ตรวจสอบดูว่ามีจำนวนคี่หรือเปล่าอย่างไรก็ตามโอกาสที่จะผิดพลาด 2 บิตพร้อมกันมีน้อยมาก

ย้อนกลับมาดูสัญญาณอะซิงโครนัสใหม่ หลังจากบิตพาริตี้แล้วก็จะต้องมีสต๊อปบิตซึ่งเป็น 1 ความกว้างของสต๊อปบิตอาจจะเป็น 1, 1.5 หรือ 2 พัลส์ของสัญญาณนาฬิกา แล้วแต่ผู้รับและผู้ส่งจะตกลงใช้กันเอง การเริ่มใช้พอร์ตอนุกรม (ทางออกอนุกรม) จึงจำเป็นจะต้องตั้งค่าต่างๆ สำหรับเป็นการส่งแบบอนุกรมอันได้แก่

1. ความเร็วในการส่ง
2. ความยาวรหัส 1 อักขระ
3. บิตตรวจสอบ
4. จำนวนสต๊อปบิต

ในการส่งโทรพิมพ์หรือโทรเลขเมื่อก่อนนี้ใช้ความเร็วแค่ 70 บอด และ 110 บอด สำหรับคอมพิวเตอร์ความเร็วในการส่งมีให้เลือกตั้งแต่ 110, 200, 300, 1200, 2400, 4800, 9600 บอด และสูงไปกว่านั้น เนื่องจากมี IC หลายเบอร์ทำหน้าที่รับส่งแบบอะซิงโครนัสให้ใช้ การส่งแบบอนุกรมจึงสะดวกสบายสำหรับคนออกแบบพอร์ตอนุกรม

จะเห็นว่ากลไกในการวิงโครนัสของการสื่อสารอะซิงโครนัส มีลักษณะเป็นไปทีละตัวอักขระ จำนวนพัลส์ของสัญญาณที่ส่งออกยังมีบางส่วนใช้ในการควบคุมการส่งอยู่อันได้แก่ บิตสตาร์ต บิตสต๊อป และบิตพาริตี้ ทำให้ความเร็วการส่งอักขระต่อวินาทีน้อยลงไป การส่งสัญญาณด้วยความเร็ว 300 บอด สำหรับการเข้ารหัส 7 บิต ไม่ได้หมายความว่าส่งได้ 300 หารด้วย 7 อักขระต่อวินาที

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

2.2.4 การสื่อสารข้อมูลแบบอะซิงโครนัสที่มีการแมตช์ความเร็ว

ไม่ว่ากรณีใดๆทั้งสิน อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เราได้กล่าวถึงการถ่ายโอนข้อมูลจากแบบอนุกรมจากอุปกรณ์เครื่องหนึ่งไปยังอีกเครื่องหนึ่งซึ่งอาจจะเป็นคอมพิวเตอร์หรืออุปกรณ์สื่อสารชนิดอื่น โดยสมมติฐานว่าความเร็วในการเปลี่ยนสัญญาณจากขนานเป็นอนุกรมได้เร็วพอ และฝ่ายรับเปลี่ยนจากอนุกรมเป็นขนานแล้วนำไปแสดงบนจอ พิมพ์ออกที่เครื่องพิมพ์หรือเก็บไว้ในดิสก์ทันทีได้ทันเวลาด้วยความเร็วในแต่ละการทำงานเท่ากันทั้งฝ่ายรับและฝ่ายส่ง ไม่มีการหน่วงเวลาหรือการอินเตอร์รัพต์ระหว่างกลาง อย่างไรก็ตามสมมติฐานนี้ย่อมไม่เป็นความจริง ฝ่ายส่งทำหน้าที่ส่งอย่างเดียวแต่ฝ่ายรับอาจต้องทำหน้าที่หลายอย่าง เช่น รับ แสดงผล เก็บ พิมพ์ เป็นต้น ความเร็วของฝ่ายรับหากไม่เพียงพอที่จะทำหลายอย่างให้ทันกับฝ่ายส่ง (แน่นอนย่อมขึ้นอยู่กับความเร็วในการส่ง) ก็จำเป็นจะต้องมีกลไกในการควบคุมการถ่ายโอน เทคนิคในการควบคุมความเร็วในการส่งมีอยู่หลายรูปแบบซึ่งอาจจะแบ่งออกได้เป็น 2 ลักษณะคือ ส่งข้อมูลบอกการออนออฟการทำงาน (on-off data flow toggle) และหาที่เก็บข้อมูลชั่วคราวหรือสร้างบัฟเฟอร์ (Temporary data storage mechanism)

2.2.5 การควบคุมการส่งเมื่อความเร็วในการทำงานของฝ่ายรับและฝ่ายส่งไม่เท่ากัน

เนื่องจากการใช้ภาษาในระดับสูงเขียนเป็นโปรแกรมสำหรับการควบคุมการทำงานของการถ่ายโอนข้อมูลแบบอนุกรม อาจจะใช้เวลามากกว่าที่จะรับข้อมูลเข้ามาได้ทันกับสถานีส่งข้อมูลมา จำเป็นจะต้องมีวิธีการควบคุมไม่ให้เกิดการสูญหายของข้อมูลที่สถานีส่งส่งมา วิธีการดังกล่าวนี้มีอยู่หลายวิธีเช่น

1. การมีบัฟเฟอร์ในการสื่อสารข้อมูล

บัฟเฟอร์สำหรับการสื่อสารก็มีหน่วยความจำในคอมพิวเตอร์ซึ่งแบ่งแยกออกมาจากหน่วยความจำหลักสำหรับเก็บพักข้อมูลในการติดตั้งชั่วคราวบัฟเฟอร์สำหรับการสื่อสารนี้ส่วนมากใช้สำหรับฝ่ายรับเท่านั้น เนื่องจากฝ่ายรับจำเป็นจะต้องตามฝ่ายส่งให้ทัน ถ้าหากฝ่ายรับใช้ภาษาแอสเซมบลี้ควบคุม มีความเร็วพออาจไม่จำเป็นต้องใช้บัฟเฟอร์สำหรับการสื่อสารเนื่องจากภาษาแอสเซมบลี้มีความเร็วสูง

ข้อมูลที่จัดส่งให้คอมพิวเตอร์ที่เป็นฝ่ายรับ ส่วนมากจะอ่านมาจากแฟ้มที่บันทึกไว้ในดิสต์ หากพิจารณาระหว่างการส่งข้อมูลออก ข้อมูลที่อ่านมาจากดิสต์จะมีลักษณะเป็นกลุ่มได้รับการนำมาสู่บัฟเฟอร์ การอ่านแต่ละกลุ่มดำเนินไปจนกระทั่งบัฟเฟอร์เต็ม การอ่านจะหยุดลงจนกระทั่งบัฟเฟอร์ถูกส่งออกไปหมดในลักษณะของเข้าก่อนออกก่อนข้อมูลก็จะถูกอ่านออกมาใส่บัฟเฟอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ส่งอีกครั้ง โดยปกติบัฟเฟอร์ส่งจะมีขนาด 255 ตัวอักษรหรือประมาณ 3 บรรทัดของ 80 อักษร

บัฟเฟอร์รับของฝ่ายรับมีผลกระทบต่อการรับ-ส่งข้อมูลมากกว่าบัฟเฟอร์ส่งบัฟเฟอร์รับทำหน้าที่เช่นเดียวกับบัฟเฟอร์ส่ง แต่ทิศทางของการไหลของข้อมูลอยู่ในทางตรงกันข้าม ฝ่ายรับรับข้อมูลเข้ามาเก็บไว้ในบัฟเฟอร์รับก่อนจนกว่าโปรแกรมควบคุมการสื่อสารจะนำข้อมูลออกไปจากบัฟเฟอร์รับเพื่อไปแสดงหรือพิมพ์หรือเก็บไว้ในแฟ้มก็แล้วแต่ ขอเพิ่มเติมอีกนิดว่าในระบบควบคุมการทำงานของไมโครคอมพิวเตอร์อย่างเช่น IBM PC มีกลไกบัฟเฟอร์รับส่งนี้ไว้อยู่ โปรแกรมในระดับสูงจึงเพียงแต่ทำหน้าที่ดึงเอาข้อมูลจากบัฟเฟอร์นี้ไปใช้ เราจะเห็นได้ชัดถึงความจำเป็นในการใช้บัฟเฟอร์เมื่อความเร็วในการส่งสูงเกินกว่า 600 บอด ภาษาในระดับสูง เช่น ภาษาเบสิกไม่สามารถที่จะรับข้อมูลจากพอร์ตอนุกรมได้ทันแน่ๆ ระบบควบคุมการทำงานจึงถูกออกแบบมาเพื่อการสื่อสารข้อมูล โดยการใช้อินเตอร์รัพต์เข้ามาช่วยเมื่อมีข้อมูลเข้ามาที่พอร์ตอนุกรมเมื่อไร ระบบควบคุมจะอินเตอร์รัพต์การทำงาน เพื่อดึงข้อมูลไปใส่ในบัฟเฟอร์รับทันที เพื่อไม่ให้ข้อมูลที่รับหายไปก่อนเมื่อมีตัวใหม่ส่งมาทับที่พอร์ตอนุกรม

หน้าที่ของโปรแกรมควบคุมการรับส่งก็คือ การอ่านข้อมูลจากบัฟเฟอร์รับไปใช้เมื่อถูกอ่านจากบัฟเฟอร์รับไปแล้ว ตัวที่อ่านออกไปก็จะหายไปจากบัฟเฟอร์ ลองนึกภาพดูจะเห็นว่าฝ่ายหนึ่งคือระบบควบคุมการทำงาน (OS) รับข้อมูลจากพอร์ตอนุกรมใส่บัฟเฟอร์ อีกฝ่ายหนึ่งคือโปรแกรมควบคุมการรับส่งดึงข้อมูลออกจากบัฟเฟอร์ เปรียบเสมือนคนหนึ่งตักน้ำใส่ตุ่มอีกคนหนึ่งตักออกจากตุ่ม ถ้าฝ่ายที่ตักออกมีความเร็วมากกว่าตุ่มก็จะมีโอกาสแห้ง ในทางตรงกันข้ามถ้าฝ่ายตักออกช้ากว่าฝ่ายตักเข้าโอกาสที่จะล้นตุ่มก็ย่อมจะมี ในทางสื่อสารเรียกว่า บัฟเฟอร์รับโอเวอร์โฟล์ (receive buffer overflow) การไหลล้นดังกล่าวทำให้ข้อมูลที่ได้รับหายไป

โปรแกรมที่เขียนด้วยภาษาแอสเซมบลี้สามารถทำงานได้เร็ว และอาจไม่จำเป็นต้องใช้บัฟเฟอร์สำหรับการรับส่งเลยก็ได้ แต่ถ้าหากเขียนด้วยภาษาเบสิกจำเป็นจะต้องมีบัฟเฟอร์อย่างน้อย 1024 ไบต์ สำหรับการสื่อสารที่ความเร็วไม่เกิน 600 บอด

ในเครื่อง IBM PC เราสามารถกำหนดบัฟเฟอร์สำหรับการสื่อสารได้เมื่อเราเรียกใช้ BASIC หรือ BASICA เช่น

```
A > BASICA/C:4096
```

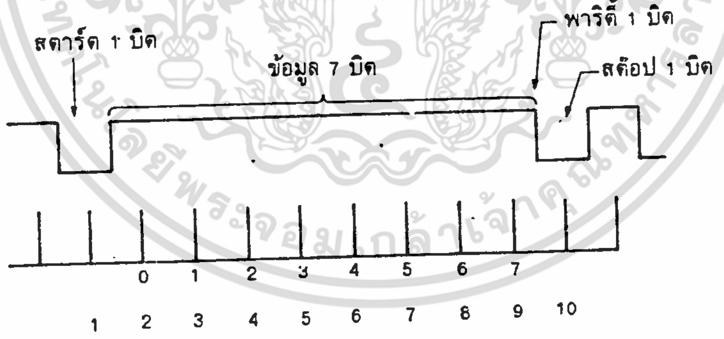
เป็นการกำหนดบัฟเฟอร์การสื่อสารด้วยขนาด 4 กิโลไบต์ ถ้าหากเราไม่กำหนด

เอกสารนี้เป็นเอกสารเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ทางการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ค่าบัฟเฟอร์สื่อสารเอาไว้ DOS จะตั้งค่าให้เท่ากับ 256 ไบต์ โดยการใส่ / C ทำให้เรา สามารถกำหนดบัฟเฟอร์ได้จาก 0 ถึง 32,767 ไบต์ อย่างไรก็ตาม ภาษาเบสิกใช้หน่วยความจำได้ 64 กิโลไบต์ หากเรากำหนดค่าบัฟเฟอร์สื่อสารเอาไว้มากย่อมจะมีหน่วยความจำเหลือสำหรับใช้อย่างอื่นน้อยลง เมื่อเข้าสู่ภาษาเบสิก แล้วแรกเริ่มภาษาเบสิกจะบอกจำนวนหน่วยความจำเหลือใช้งานจริงๆ ในขณะที่โหลดโปรแกรมภาษาเบสิกเข้ามา ขนาดของโปรแกรมย่อมจะเข้าไปอาศัยในหน่วยความจำที่เหลือ ทำให้หน่วยความจำที่จะใช้งานสำหรับเก็บตัวแปรต่างๆ น้อยลงไปอีก เราสามารถใช้คำสั่งโดยตรงสำหรับการตรวจดูขนาดของหน่วยความจำที่ว่างโดยการพิมพ์คำสั่ง PRINT FRE (0)

2. การควบคุมโดยใช้ XON/XOFF

ถึงแม้ว่าเราจะมีบัฟเฟอร์สำหรับการสื่อสารแล้วก็ตาม ในบางครั้งการถ่ายโอนข้อมูลด้วยความเร็วสูงและด้วยขนาดของแฟ้มที่จะทำการถ่ายโอนมีขนาดใหญ่กว่าบัฟเฟอร์สื่อสาร โอกาสที่ข้อมูลจะหายไปมีอยู่มาก ลองมาคำนวณดูง่ายๆ สมมุติว่า เราใช้ความเร็วในการถ่ายโอนข้อมูล 9600 บิตต่อวินาที สต๊อปบิตเป็น 1 บิต ข้อมูล 7 บิต และพาริตี้เป็นคู่ ใน 1 ตัวอักษร จะต้องใช้ 11 บิต



รูปที่ 2.14 รูปแบบของข้อมูล 1 ตัวอักษร

เพราะฉะนั้นฝ่ายรับจะต้องอ่านข้อมูลจากพอร์ตอนุกรมทุก 11/9600 ได้ผลประมาณ .001 วินาที หรือ 1 มิลลิวินาที ถ้าเปรียบเป็นพัลส์ ได้ประมาณ 5000 พัลส์นาฬิกา (ความเร็วนาฬิกาของ IBM PC = 4.77 MHz หรือประมาณ .2 ไมโครวินาทีต่อหนึ่งพัลส์ 1000/.2 = 5000) ในเมื่อบัฟเฟอร์ไม่เพียงพอ เราก็จำเป็นจะต้องควบคุมการรับส่ง โดยการบอกให้ฝ่ายส่งหยุดส่งชั่วคราว (XOFF) จนกว่าฝ่ายรับจะจัดการเอาข้อมูลออกจากบัฟเฟอร์สื่อสารหมดเสียก่อน จึงบอกให้ฝ่ายจัดการส่งต่อไป (XON) ในรหัส ASCII XON มีค่าเท่ากับ 17 XOFF

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

มีค่าเท่ากับ 19 เป็นหน้าที่ของนักเขียนโปรแกรมที่จะต้องจัดการส่ง XOFF ออกไปให้ฝ่ายส่งได้รู้ก่อนที่บัฟเฟอร์สื่อสารจะเต็มเสียก่อน

คอมพิวเตอร์เมนเฟรมส่วนมากจะมีระบบ XON/XOFF ให้ สำหรับการเชื่อมต่อทางด้านความเร็ว (Speed matching) แต่โปรแกรมสื่อสารที่มีขายสำหรับ IBM PC ไม่มี XON/XOFF ทุกตัว โดยมากโปรแกรมที่เขียนโดยภาษาแอสเซมบลี้จะมี XON/XOFF ให้ แต่โปรแกรมที่เขียนโดยภาษาเบสิกจะมีเพียงบางโปรแกรมเท่านั้น โปรแกรมที่เขียนโดยภาษาแอสเซมบลี้มีความเร็วพอที่จะคอยตรวจสอบดู XOFF ที่ส่งมาจากฝ่ายรับ แต่ถ้าเป็นภาษาเบสิก ความเร็วอาจจะไม่เพียงพอต่อการตรวจสอบ XOFF ที่ฝ่ายส่ง

3. การใช้โปรโตคอล

อีกเทคนิคหนึ่ง ในการควบคุมการรับส่งก็คือ การส่งโปรโตคอล (Protocol Transfer) เทคนิคนี้จำเป็นจะต้องมีเหมือนกันทั้งฝ่ายรับและฝ่ายส่ง โดยการใช้อักขระควบคุมในตารางของแอสกี้ สำหรับควบคุมการส่งข้อมูลออกมาเป็นกลุ่มที่มีขนาดคงที่

การใช้โปรโตคอลอาจจะใช้อักขระต่อไปนี้ ในการส่งข้อมูลเป็นกลุ่มๆ

ETB End of Transsion Block

ETB มีค่าเท่ากับ 23 ในตารางแอสกี้ เป็นการบอกฝ่ายรับว่าขณะนี้สิ้นสุดการส่งข้อมูลกลุ่มหนึ่งแล้ว

ETX End of Text

ETX มีค่าเท่ากับ 03 ในตารางแอสกี้ เป็นการบอกฝ่ายรับว่าขณะนี้สิ้นสุดการส่งแล้ว

ENQ Enquiry

ENQ มีค่าเท่ากับ 05 ในตารางแอสกี้ เป็นอักขระที่ส่งมาจากฝ่ายรับ ขอให้ฝ่ายส่ง ส่งข้อมูลมา

NAK Negative Acknowledge

ในตารางระหัสแอสกี้ มีค่าเท่ากับ 021 เป็นการบอกฝ่ายส่งว่าข้อมูลที่ได้รับนั้นผิดพลาด

ACK Acknowledge

เป็นการบอกฝ่ายส่งว่า ข้อมูลที่ได้รับนั้นถูกต้อง แล้วเป็นระหัสแอสกี้มีค่า

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับ... ได้รับอนุญาตให้ใช้... ในการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เท่ากับ 06

โปรโตคอลที่ใช้ใน IBM PC ส่วนมากจะเป็นโปรโตคอลที่มีชื่อว่า XMODEM ซึ่งมีลักษณะการทำงานดังนี้

ฝ่ายส่งจะไม่ส่งข้อมูลจนกว่าจะได้รับ NAK จากฝ่ายรับ ฝ่ายส่งจะส่งข้อมูลออกไป โดยมีรูปแบบเริ่มด้วย SOH ตามด้วยอักขระ 2 ตัว สำหรับบอกกลุ่มของข้อมูลที่ส่งและตามด้วยส่วนเติมเต็ม (1's complement) ของกลุ่มต่อไปที่จะส่ง ต่อจากนั้นจะเป็นข้อมูล 128 ไบต์แล้วหารด้วย 255 เศษที่เหลือก็คือค่า checksum ข้างฝ่ายรับเมื่อแยกเอา SOH และหมายเลขบล็อก (Block Number) ทั้งสองออกไปแล้วก็จะเอาข้อมูล 128 ไบต์มารวมกันเพื่อหาค่า checksum เอา checksum ที่หาได้เปรียบเทียบกับค่าที่ได้รับ หากตรงกันถือว่าข้อมูลที่ได้รับถูกต้อง จึงส่งสัญญาณ ACK ไปให้ฝ่ายส่งได้รู้ว่าขณะนี้ได้รับข้อมูลไว้ถูกต้อง แล้วส่งกลุ่มของข้อมูลต่อมาได้ ถ้าหากค่า checksum ไม่ถูกต้อง ฝ่ายรับจะส่ง NAK ให้ฝ่ายส่ง เพื่อเป็นการบอกว่าข้อมูลที่ได้รับผิดพลาด ช่วยส่งกลุ่มข้อมูลอันเก่ามาให้ที ฝ่ายส่งก็จะส่งข้อมูลมาให้ใหม่ การส่งใหม่จะดำเนินไป 9 ครั้ง หากยังคงได้รับแต่สัญญาณ NAK ฝ่ายส่งจะหยุดทำงาน แสดงว่าตัวกลางการสื่อสารแย่มาก

การที่ XMODEM ใช้เลขบอกกลุ่ม (Block Number) 2 ตัว (ตัวหนึ่งบอกกลุ่มที่ส่งขณะนี้ อีกส่วนหนึ่งเป็นส่วนเติมเต็มหนึ่ง ของกลุ่มต่อไป) เพื่อความแน่นอนว่ากลุ่มเดียวกันจะไม่ถูกส่งออกไปสองครั้ง หากอักขระควบคุมการส่งเกิดสูญหายไประหว่างการส่ง ฝ่ายรับจะจะตรวจสอบดูว่ากลุ่มของข้อมูลที่ส่งมา เป็นกลุ่มที่ฝ่ายรับต้องการหรือไม่ หากกลุ่มเก่าถูกส่งมาอีกด้วยความผิดพลาด จาก ACK เป็น NACK ฝ่ายรับก็จะจับข้อมูลที่รับมาโยนทิ้งไป เมื่อดำเนินไปจนสิ้นสุดแฟ้มที่จะส่ง ฝ่ายส่งก็จะส่ง ETX เป็นการบอกฝ่ายรับว่าหมดข้อความที่จะส่งแล้ว

XMODEM เหมาะสำหรับ IBM PC เหนือโปรโตคอลชนิดอื่น 3 ข้อคือ

1.ใช้อักขระควบคุมที่มีอยู่แล้วในระหัสแอสกี้

2.สามาถใช้ภาษาในระดับสูงควบคุมได้ เช่น BASIC , PASCAL

3.ต้องการบัฟเฟอร์สื่อสารแค่ 256 ไบต์

4.ระบบบริการข่าวสารด้วยคอมพิวเตอร์ โดยทั่วไปใช้โปรโตคอล XMODEM

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ข้อดีของระบบควบคุมการรับส่งแบบอนุกรมโดยการใช้ระบบการใช้โปรโตคอลก็คือ

1.โปรโตคอลบางชนิดสามารถเลือกขนาดของกลุ่มข้อมูลได้

2.สามาถส่งข้อมูลที่ไม่ใช่แอสกี้ได้ โดยไม่ต้องกลัวว่ารหัสนั้นจะไปทับกับ รหัสควบคุมของแอสกี้

3.การตรวจสอบโดยวิธี checksum มีความสามารถตรวจสอบความผิดพลาดได้ดีกว่าใช้บิตพาริตี้ในอะซิงโครนัสในขณะที่บิตพาริตี้สามารถให้ประสิทธิผลได้ 95% แต่ checksum สามารถให้ประสิทธิผลถึง 99.5% หากพบการผิดพลาดด้วยบิตพาริตี้ ไม่ทำให้เกิดการส่งใหม่ แต่ข้อผิดพลาดจาก checksum ทำให้เกิดการส่งข้อมูลมาใหม่

2.3 **พอร์ต** RS 232C

โดยปกติไมโครคอมพิวเตอร์จะมีพอร์ตที่เป็นแบบอนุกรม เรียกชื่อกันว่า RS 232C อยู่ในตัวเองอยู่แล้ว หลายเครื่องไม่มีมากับเครื่อง อย่างเช่น IBM PC จำเป็นจะต้องมีการ์ดที่เรียกว่าอะซิงโครนัสอะแดบเตอร์ (Asynchronous Communication Adapter) มาใส่

พอร์ต RS 232 นี้ทำหน้าที่รับและส่งข้อมูลในแบบอนุกรมเรียกว่า Universal Asynchronous Adapter เหตุที่มีชื่อเรียกว่า RS 232C ก็เนื่องจากสมาคมผู้ผลิตอุปกรณ์อิเล็กทรอนิกส์ของอเมริกาหรือ EIA ได้กำหนดมาตรฐานของอุปกรณ์การสื่อสารแบบอนุกรมเอาไว้ภายใต้ชื่อว่า RS 232C ความจริงมาตรฐานของการส่งข้อมูลแบบอนุกรมมีหลายมาตรฐาน แต่ที่นิยมกันมากที่สุดสำหรับไมโครคอมพิวเตอร์ก็คือ RS 232C

หน้าที่สำคัญของการสื่อสารแบบอะซิงโครนัสก็คือ

รับสัญญาณ

1. เปลี่ยนสัญญาณเข้ามาแบบอนุกรมให้เป็นแบบขนาน
2. ตรวจสอบความผิดพลาดของสัญญาณที่รับ
3. ตัดสต๊อปบิตและพาริตี้บิตออก
4. ส่งสัญญาณให้ซีพียูรู้ว่ารับสัญญาณไว้แล้ว

ส่งสัญญาณ

1. เปลี่ยนสัญญาณแบบขนานจากซีพียูค่อยทยอยส่งออกเป็นแบบอนุกรม

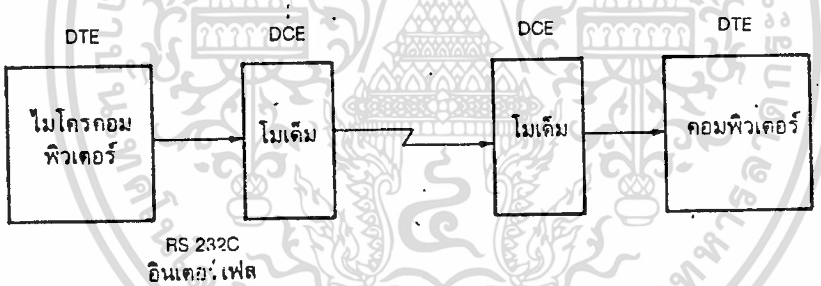
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษา ไม่ใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2. เพิ่มสต๊อปบิตและพาริตี้บิต

3. เพิ่มสัญญาณควบคุมโมเด็มที่ต่อเชื่อม (ถ้ามี)

มาตรฐาน RS 232C

มาตรฐาน RS 232C ได้จัดพิมพ์ขึ้นเมื่อ ปี ค.ศ. 1969 โดยสมาคมผู้ผลิตอุปกรณ์อิเล็กทรอนิกส์แห่งสหรัฐอเมริกา RS ย่อมาจาก Recommened Standard ส่วน 232 เป็นหมายเลขบ่งบอกของมาตรฐานตัวนี้ "C" เป็นหมายเลขของฉบับท้ายสุดของมาตรฐานตัวนี้ จุดประสงค์ของมาตรฐานตัวนี้ก็เพื่อบรรยายคุณลักษณะของการเชื่อมต่ออุปกรณ์รับส่งข้อมูลปลายทาง (Data Terminal Equipment DTE) กับอุปกรณ์สื่อสารข้อมูล (Data Communication Equipment DCE) สำหรับผู้ใช้ไมโครคอมพิวเตอร์ DTE ก็หมายถึงตัวไมโครคอมพิวเตอร์และ DCE ก็หมายถึงโมเด็ม อุปกรณ์อื่นๆ เช่น เครื่องพิมพ์ที่รับสัญญาณแบบอนุกรมอาจจะเป็นได้ทั้ง DTE



รูปที่ 2.15 การใช้ RS 232C เชื่อมต่ออุปกรณ์

อาจจะใช้ต่อชนิดอื่นที่นอกเหนือไปจาก DB-25 ยกตัวอย่างเช่น Fujitsu F IBM AT, IBM Jr เป็นต้น ตัวเมียของข้อต่อควรจะอยู่ที่ตัวโมเด็ม ขณะที่ตัวผู้ควรจะอยู่ที่ asynchronous communication adapter หรือที่ตัวไมโครคอมพิวเตอร์เอง อย่างไรก็ตามผู้ผลิตหลายรายไม่ได้ทำตามกฎเกณฑ์ที่ว่านี้

สัญญาณต่างๆ ถูกมอบหมายให้ทำหน้าที่ดังนี้

Transmit Data (TD ขาที่2)

เป็นสัญญาณที่ส่งออกจาก DTE (หรือตัวไมโครคอมพิวเตอร์) ไปยังโมเด็มหรือต่อเข้าโดยตรงกับไมโครคอมพิวเตอร์ตัวอื่น หรือเครื่องพิมพ์ เมื่อไม่มีสัญญาณส่งออกสถานภาพของลอจิกเท่า

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้เพื่อการศึกษา ห้ามมิให้นำไปใช้ประโยชน์ทางการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ขานี้จะมีค่าเท่ากัน "1" หรือเทียบเท่ากับสต๊อบบิต

Receive Data (RD ขาที่ 3)

เป็นทางของสัญญาณเข้าไปยัง DTE หรือไมโครคอมพิวเตอร์เมื่อไม่มีสัญญาณรับเข้ามา ขานี้จะมีสถานภาพทางลอจิก เป็น "1"

Request To send (RTS ขาที่ 4)

ใช้สำหรับส่งสัญญาณไปยังโมเด็มหรือเครื่องพิมพ์เป็นการเรียกร้องที่จะส่งสัญญาณมาทาง ขา 2 สัญญาณนี้ใช้คู่กับ CTS หรือ Clear to send อุปกรณ์รับหาได้รับสัญญาณ RTS จะตรวจสอบตัวเองว่าพร้อมจะรับสัญญาณได้หรือยัง หากพร้อมจะรับก็ส่งสัญญาณออกไปที่สาย CTS

Clear To Send(CTS ขาที่ 5)

ดังอธิบายไว้ใน RTS เมื่อสัญญาณนี้อยู่ในสถานะออฟ(negative voltage หรือลอจิก "1") หมายความว่า อุปกรณ์รับกำลังบอว่าพร้อมที่จะรับข้อมูลแล้ว

Data Set Ready (DSR ขาที่ 6)

เมื่อสัญญาณสายนี้อยู่ในสถานะออน (หรือลอจิก 0) เป็นการบอกไมโครคอมพิวเตอร์หรือฝ่ายส่งว่า โมเด็มต่อเข้ากับสายโทรศัพท์เรียบร้อยแล้วและพร้อมที่จะได้แล้ว โมเด็มที่มีการหมุนหมายเลขอัตโนมัติจะส่งสัญญาณนี้ไปบอกคอมพิวเตอร์รู้ว่าต่อโทรศัพท์ได้สำเร็จแล้ว

SignalGround (SG ขาที่ 7)

SG ทำหน้าที่เป็นระดับแรงดันอ้างอิงสำหรับทุกๆ สายของสัญญาณ จะมีแรงดันเป็น "0" เมื่อเทียบกับสัญญาณตัวอื่น

Carrier Detect (CD ขาที่ 8)

โมเด็มจะส่งสัญญาณที่อยู่ในสถานะออน (ลอจิก "0") ไปบอกไมโครคอมพิวเตอร์ เมื่อได้รับสัญญาณจากโมเด็มของอีกฝ่ายหนึ่ง สัญญาณนี้จะนำไปจุด LED บอกว่าได้รับสัญญาณจากโมเด็มอีกฝ่ายหนึ่งแล้ว ไฟ LED จะอยู่บนหน้าปัดของโมเด็มเอง

เอกสารนี้เป็นเอกสารเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

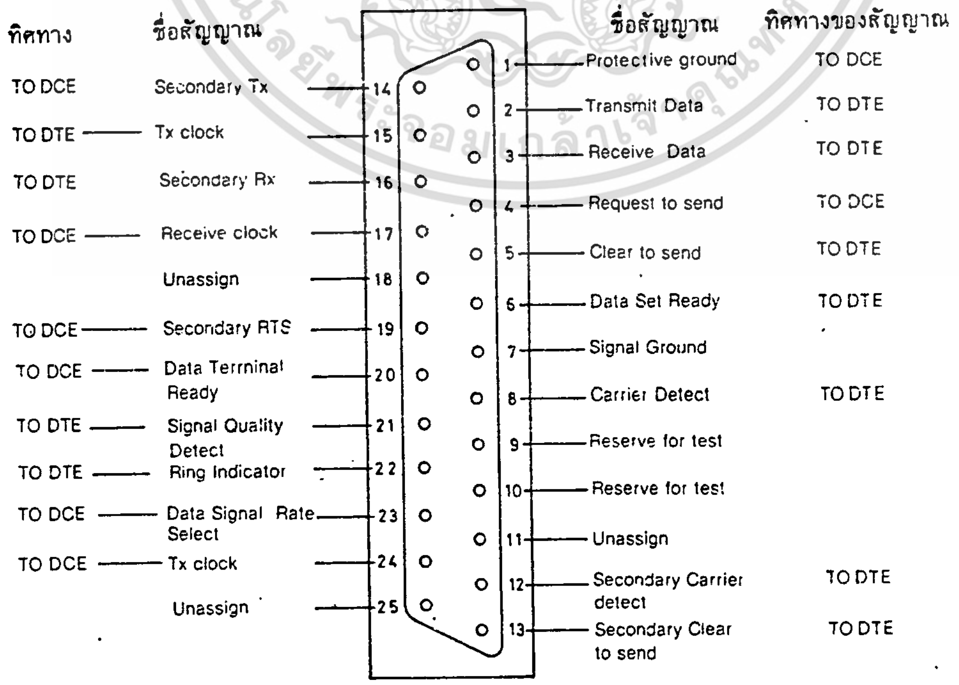
Data Terminal Ready (DTR ขาที่ 20)

คอมพิวเตอร์เปิดสัญญาณสายนี้ให้ออน (ลอจิก "0") เมื่อพร้อมที่จะติดต่อกับโมเด็ม โมเด็มส่วนมากจะไม่รายงานสถานภาพของตัวเอง (CD, DSR และ CTS) ให้คอมพิวเตอร์รู้ หากคอมพิวเตอร์ไม่เปิดสัญญาณ DTR

Ring Indicator (RI ขาที่ 22)

สัญญาณนี้ใช้ในโมเด็มที่เป็นระบบตอบได้อัตโนมัติ (Auto-answer) สัญญาณนี้จะออน เมื่อมีสัญญาณกระดิ่งมา และจะออฟระหว่างเสียงดังของกระดิ่ง

อาจสับสนได้ว่า ระหว่างสถานภาพของลอจิก กับสถานภาพของสัญญาณ โดยปกติเรา จะคุ้นเคยอยู่กับความรู้สึกที่ว่า เมื่อแรงดันเป็นบวก หรือสัญญาณออนลอจิกน่าจะเป็น "1" สำหรับ สัญญาณต่างๆ ที่กล่าวมานี้จะมีลักษณะตรงกันข้าม ทำไมเขากำหนดกฎเกณฑ์ออกมาอย่างนี้ ก็เพราะ ว่าแต่เดิมนั้นการติดต่อกันทางโทรเลข การทำงานของสัญญาณจะต้องครบวงจรทั้งฝ่ายส่งและฝ่าย รับ เมื่อลอจิกเป็น "0" หรือขณะที่ไม่มีอะไรส่งควรจะมีสัญญาณทางไฟฟ้าครบวงจรอยู่ตลอดเวลา จะได้รู้ว่าวงจรไม่ขาดระหว่างทางตรงไหน ควรจะรู้ว่าวงจรครบอยู่ตลอดเวลาก็โดยการให้ค่า แรงดันที่ฝ่ายส่ง ดังนั้นจึงถือกันว่าสัญญาณไฟบวกใช้เป็นลอจิก "0"



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 2.2 คุณลักษณะโดยย่อของสัญญาณ RS 232C

Driver output logic levels with	15V>O_h >5V
3K to 7k load	-5>O_L >-15 โวลต์
Driver output voltage when open	V_o <25 โวลต์
circuit	
Driver output impedance with Power off	R_o>300 Ohms
Output short circuit current	I_o < 0.5 A
Driver slew rate	dv/dt < 30 V/s
Receiver input impedance	7k >R_{in} >3k
Receiver input voltage	+15 compatible
	with driver
Receiver output with open	MARK
circuit input	
Receiver output with +3V input	SPACE
Receiver output with -3V input	MARK
+ 15	LOGIC 0 =SPACE=
+ 5	CONTROL ON
+ 5	Noise margin
+ 3	
+ 3	Transition Region
- 3	
- 3	Noise margin
- 5	
- 5	LOGIC 1 = MARK =
- 15	CONTROL OFF

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.4 การควบคุมการสื่อสาร ของคอมพิวเตอร์

หลังจากเราได้รู้ส่วนประกอบที่เป็นฮาร์ดแวร์ ที่จำเป็นสำหรับการสื่อสารข้อมูลแล้วมีสิ่งที่เราจะต้องทำความเข้าใจและบวกเข้ากับ PC ก่อนที่จะทำให้ PC สื่อสารข้อมูลได้ นั่นก็คือซอฟต์แวร์สำหรับควบคุมการสื่อสาร ก่อนอื่นลองมาทำความเข้าใจการทำงานระหว่างฮาร์ดแวร์และซอฟต์แวร์สำหรับการควบคุมการสื่อสารของไมโครคอมพิวเตอร์ โดยจะกล่าวโดยอ้างถึง IBM PC เป็นหลัก

ตามรูปที่ 2.16 จะเห็นว่าหน่วยประมวลผลกลางหรือซีพียู(CPU)เป็นส่วนของฮาร์ดแวร์ที่สำคัญในการที่จะทำให้การสื่อสารดำเนินไปในเครื่องคำสั่งที่ใช้ในการควบคุมมาจากส่วนหนึ่งของ OS หรือโปรแกรมแม่บ้าน OS ก็คือโปรแกรมที่ทำหน้าที่ควบคุมการทำงานภายในของอุปกรณ์ต่างๆ ที่ประกอบเป็นไมโครคอมพิวเตอร์ หน้าที่สำคัญก็คือ การจัดตารางการทำงานของอุปกรณ์ประกอบต่างๆ

IBM PC มีโปรแกรมทำหน้าที่แม่บ้านพื้นฐานที่บรรจุลงใน ROM โดยทำหน้าที่พื้นฐานสำหรับควบคุมการทำงานของอุปกรณ์ต่างๆ ตั้งแต่แรกเริ่มเปิดเครื่องขึ้นมาโปรแกรมส่วนนี้จะทำหน้าที่เริ่มทำงานก่อน โปรแกรมส่วนนี้เรียกว่า BIOS ซึ่งย่อมาจาก Basic Input Output System (BIOS) จะประกอบด้วยส่วนโปรแกรมสำหรับทำหน้าที่ควบคุมอุปกรณ์ต่างๆ เป็นต้นว่าการติดต่อกับคีย์บอร์ด การติดต่อกับเครื่องพิมพ์ การติดต่อกับดิสต์ และการติดต่อกับมอนิเตอร์ จนกระทั่งการติดต่อกับอะแดปเตอร์อะซิงโครนัส หรือ RS 232C แรกเริ่มที่เดียวที่เปิดเครื่องหรือรีเซตเครื่องใหม่ โปรแกรมใน BIOS ส่วนที่ทำหน้าที่ Bootrap จะเริ่มทำงานโดยการตรวจสอบความเรียบร้อยของหน่วยความจำ ตรวจสอบอุปกรณ์ที่ประกอบเข้ากับระบบว่ามีอะไรบ้างมอนิเตอร์เป็นสีหรือขาวดำ เครื่องที่พอร์ตไหน ดิสต์มีกี่ไดรฟ์ และอื่นๆ รูปลักษณ์เหล่านี้จะถูกบันทึกเอาไว้เพื่อให้โปรแกรมที่จะนำมาใช้งานได้ใช้รูปลักษณ์เหล่านี้ได้อย่างถูกต้อง ท่านที่ใช้ IBM PC อาจจะเคยนึกสงสัยว่าโปรแกรมสำเร็จที่เขาเขียนมา รู้ได้อย่างไรว่าขณะนี้ใช้อยู่กับจอมอนิเตอร์สีหรือมอนิเตอร์ขาวดำ (แอดเดรสสำหรับหน่วยความจำแสดงผลของทั้งสีและขาวสำหรับ IBM PC อยู่คนละที่) โปรแกรมสำเร็จเหล่านั้นเพียงแต่ตรวจดูผลงานที่ BIOS ทำไว้เท่านั้นเอง นอกจากนี้ BIOS จะเริ่มดึงเอา OS จริงๆ มาจากดิสต์ (ถ้าหากมีดิสต์อยู่) โดยที่ OS โปรแกรมจะกำหนดไว้ในแทร็กและเซกเตอร์ที่แน่นอนของดิสต์เกตต์ ถ้าหากหาไม่เจอ BIOS จะแสดงความผิดพลาดบนจอว่า "Non System Disk" หรือคล้ายกับเป็นการบอกผู้ใช้ให้รู้ว่าหา OS โปรแกรมไม่พบนั่นเอง ถ้าหากพบ BIOS จะดึงเอาโปรแกรมนี้มาเก็บไว้ใน RAM และเริ่มใช้โปรแกรมที่ว่านี้ทำงาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ทันที OS ที่นิยมใช้กันอย่างมากสำหรับ IBM PC ก็คือ PCDOS หรือ MSDOS นั่นเอง

เมื่อ DOS เริ่มงาน เราก็จะมีแม่บ้านที่สูงขึ้นมากกว่า BIOS นั่นคือสามารถใช้ DOS ทำงานได้มากกว่า BIOS ว่ากันที่จริง DOS ก็เป็นโปรแกรมที่เขียนขึ้นโดยใช้ความสามารถที่มีอยู่ของ BIOS เป็นหลัก ถ้าจะถามว่าทำไมไม่ทำ BIOS ซึ่งมีความสามารถเท่ากับ DOS เลยจะได้ไม่ต้องมีหลายขั้นตอน เหตุผลก็คือว่า โดยวิธีการนี้จะเป็นการแยกซอฟต์แวร์ให้เป็นอิสระจากฮาร์ดแวร์ OS อาจจะเปลี่ยนแปลงได้ เช่น เปลี่ยนเป็น CPM/86 หรือ P-System เป็นต้น อีกประการหนึ่งก็คือ สามารถพัฒนา OS ให้มีความสามารถยิ่งขึ้นโดยไม่ต้องเปลี่ยนฮาร์ดแวร์ใหม่ เช่น มีเวอร์ชันใหม่ของ DOS ออกมา เครื่องตัวเดิมก็ยังสามารถใช้ DOS ตัวใหม่ได้

เมื่อ DOS เริ่มเป็นขั้นที่ 1 ของโปรแกรม ยังมีอีกหลายชั้นที่ตามมา เนื่องจาก DOS เพียงแต่ทำหน้าที่แม่บ้าน เครื่องคอมพิวเตอร์จะยังใช้งานที่เป็นประโยชน์ไม่ได้ นอกจากจะมีโปรแกรมทำหน้าที่ควบคุมให้คอมพิวเตอร์ทำงานเฉพาะอย่างอีกทีหนึ่ง ในกรณีของเราโปรแกรมที่ว่าก็คือโปรแกรมสำหรับการสื่อสารข้อมูล

ถ้าหากโปรแกรมที่ใช้งานเป็นโปรแกรมที่เขียนโดยภาษาแอสเซมบลี้ หรือเป็นโปรแกรมที่ถูกคอมไพล์ให้เป็นภาษาเครื่องเรียบร้อยแล้ว การเรียกมาใช้งานก็เพียงแต่ใส่ชื่อของโปรแกรมนั้นเข้าไป เมื่อ DOS รอคำสั่งอยู่ โปรแกรมที่ถูกเรียกเข้ามาจะเป็นชั้นที่ 2 ของโปรแกรมต่อจาก DOS โปรแกรมที่ถูกเรียกมาใช้งานบางครั้งจะใช้คำสั่งหรือฟังก์ชันที่มีพร้อมให้ใช้ใน DOS

ถ้าหากเป็นโปรแกรมที่เขียนโดยภาษาระดับสูงอย่างเช่น ภาษาเบสิกหรือปาสคาล จำเป็นจะต้องโหลดตัวภาษานั้นๆ เข้ามาก่อน เช่น ถ้าโปรแกรมเขียนเป็นภาษาเบสิกก็จะต้องโหลดภาษาเบสิกลงมาในหน่วยความจำ เป็นซอฟต์แวร์ชั้นที่ 3 เสียก่อน ก่อนที่จะดึงโปรแกรมที่เขียนในภาษาเบสิกมาใช้งานได้

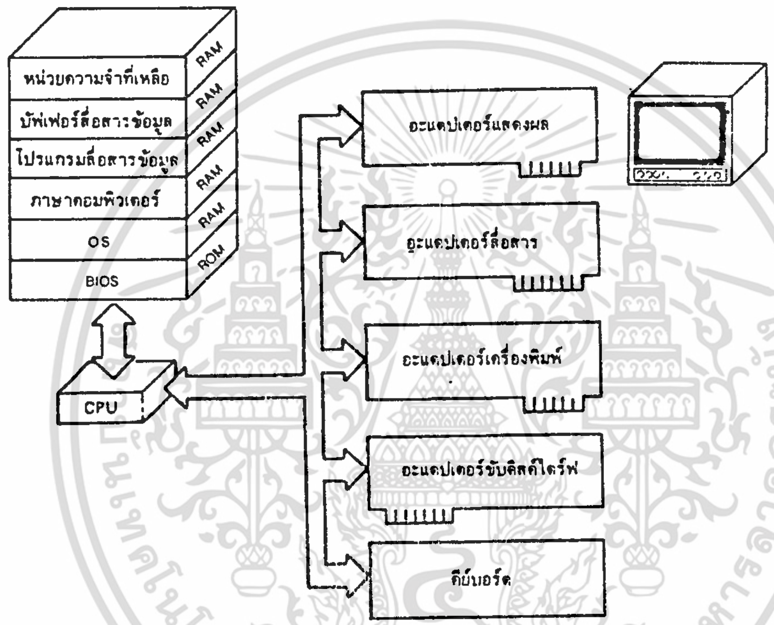
เมื่อโปรแกรมการสื่อสารเริ่มทำงาน จะมีหน่วยความจำส่วนหนึ่งถูกจองที่เอาไว้สำหรับเป็นพื้นที่สำหรับการสื่อสาร (communication buffer) สำหรับภาษาเบสิกนั้นที่สำหรับการสื่อสารนี้จะถูกกำหนดค่า เมื่อแรกเริ่มดึงเอาภาษาเบสิกมาใช้งาน โดยปกติภาษาเบสิกจะกำหนดพื้นที่สื่อสารได้แค่ 256 ไบต์ ถ้าหากเราต้องการพื้นที่มากกว่านั้น เราจะต้องบอกให้ OS รู้ ตอนที่เรียกภาษาเบสิกมาใช้โดยการใส่ C ตามหลังภาษาเบสิก เช่น

```
A > BASIC/C:4096
```

เป็นการเรียกภาษาเบสิกมาใช้โดยให้มีพื้นที่สำหรับการสื่อสาร 4096 ไบต์ ภาษาเบสิกของ IBM PC สามารถกำหนดได้ตั้งแต่ 0 ไบต์ ถึง 32 กิโลไบต์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เหตุผลสำคัญของการแบ่งซอฟต์แวร์ออกเป็นชั้นๆตามรูปที่ 2.16 ก็เพื่อความสามารถต่อการใช้และการเขียนโปรแกรม ตัวภาษาเองและโปรแกรมที่เขียนในภาษานั้นจะต้องออกแบบโดยกำหนด OS ที่จะใช้เอาไว้ โปรแกรมที่เขียนควบคุมการสื่อสารจะต้องสามารถเข้าใจและนำไปปฏิบัติงานได้สำหรับโอเปอเรติ้งซิสเต็มนั้น

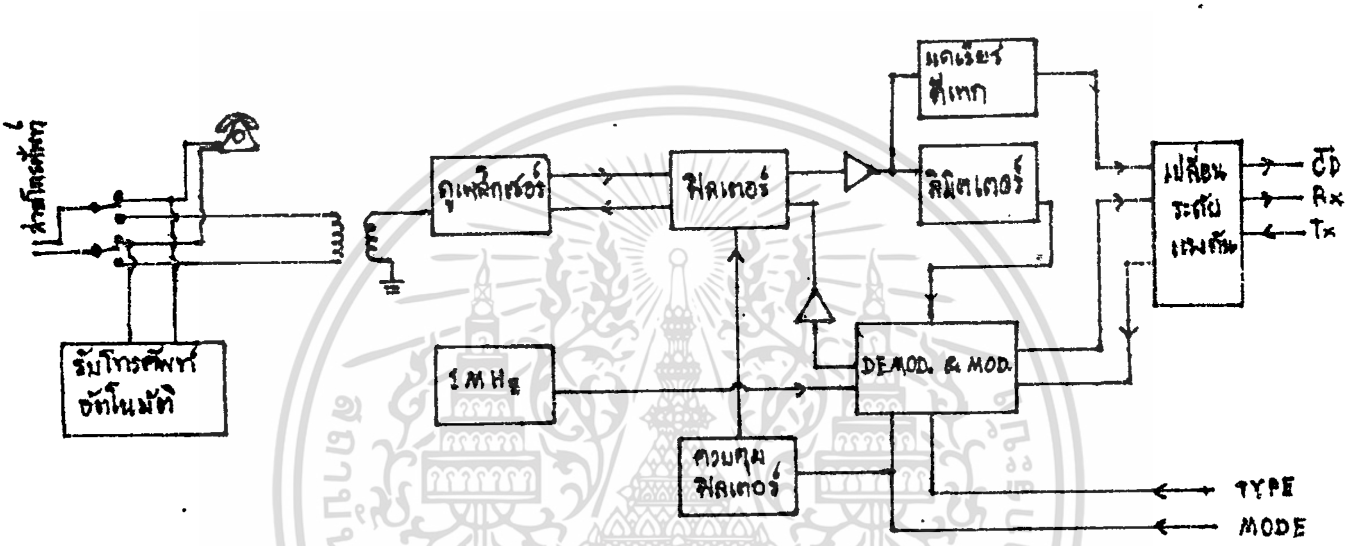


รูปที่ 2.16 แผนภูมิการทำงานของ IBM PC

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 3 วงจรและการออกแบบ

3.1 บล็อกไดอะแกรมของโครงงาน



รูปที่ 3.2.1 บล็อคไดอะแกรมการทำงาน

การทำงานขณะส่งสัญญาณ

ภาคมอดดูเลชั่น จะทำการตรวจสัญญาณข้อมูลที่จะส่ง แล้วทำการมอดดูเลทออกมาเป็นสัญญาณซายน์ ซึ่งมีฮาร์โมนิกปนอยู่ด้วย จะถูกปรับขนาดแล้วส่งไปที่ วงจรกรองสัญญาณแบบแบนด์พาส ซึ่งมีวงจรควบคุมเพื่อที่จะส่งสัญญาณ เข้าวงจรกรองสัญญาณที่ถูกต้อง จากนั้นเข้าวงจรดูเพล็กเซอร์ ซึ่งทำหน้าที่เสมือนเปลี่ยนการส่งแบบ 2 สาย เป็นการส่งแบบ 4 สาย จากนั้นผ่านหม้อแปลงออกไป

การทำงานขณะรับสัญญาณ

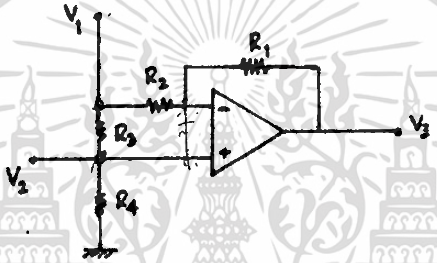
สัญญาณจากฝ่ายส่งผ่านเข้ามาทางหม้อแปลง ผ่านดูเพล็กเซอร์ ซึ่งจะแยกสัญญาณส่งที่ปนอยู่ด้วยออกไป จากนั้นทำการลดสัญญาณให้เหมาะสม แล้วเข้าวงจรกรองสัญญาณเอาที่พท

เอกสารนี้ใช้เพื่อประโยชน์ส่วนบุคคลหรือการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ที่ออกมาจะถูกส่งเข้าภาค แคเรียร์ดีเทก (CARRIER DETECT) ซึ่งเป็นตัวตรวจจับสัญญาณที่รับมา และภาคลิมิตเตอร์ เอาท์พุทจากลิมิตเตอร์ จะเป็นสัญญาณสี่เหลี่ยมมีความถี่เท่ากับสัญญาณซายน์จากอินพุท ต่อจากนั้นไปเข้าภาคดีมอดดูเลชั่นได้เป็นสัญญาณ ดิจิตอล

สัญญาณ Rx,Tx,CD ซึ่งเป็นสัญญาณที่ส่งระหว่าง คอมพิวเตอร์กับโมเด็ม โดยผ่าน RS 232C พอร์ท ซึ่งมีระดับแรงดัน ±12V. ดังนั้นจะต้องมีวงจรแปลงระดับสัญญาณ เพื่อที่จะเปลี่ยนระดับสัญญาณให้เหมาะสม

3.2 ดูเพล็กเซอร์ (Duplexer)



รูปที่ 3.2.1 วงจรดูเพล็กเซอร์

V1 = สัญญาณส่ง

V2 = สัญญาณที่สายโทรศัพท์

V3 = สัญญาณที่รับมาจากอีกฝ่าย ซึ่งจะไปเข้าวงจรกรองสัญญาณ

A1 = V2/V1

= อัตราขยายของสัญญาณจากมอดดูเลเตอร์ออกไปสายโทรศัพท์ จะได้

A1 = R4/(R3+R4)

A2 = V3/V2

= อัตราการขยายของสัญญาณจากสายโทรศัพท์ออกไป เข้าวงจรกรองสัญญาณ จะได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

A2 = 1+R2/R1

A3 = V3/V1

= อัตราการขยาย ของสัญญาณจากมอดดูเลเตอร์ออกไป เข้าวงจรกรองสัญญาณ จะได้

$-1 + (1+1)\cdot\frac{1}{2} = 0$

A3 = -R1/R2 + (1+R2/R1)R3/(R3+R4)

ในการใช้งาน เราไม่ต้องการให้สัญญาณจากมอดดูเลเตอร์ ไปปนกับสัญญาณที่จะไปเข้าวงจรกรองสัญญาณ ดังนั้นเราจะต้องออกแบบให้ A3 มีค่าน้อยที่สุด โดยเราให้

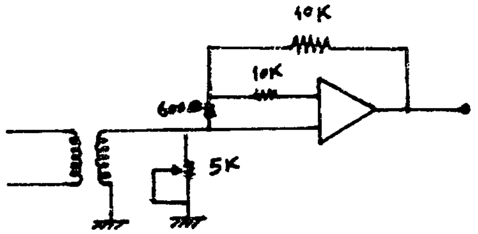
R1 = R2 และ R3 = R4

จะได้ A3 = 0 และ A1 = 1/2 และ A2 = 2

เราให้ R1 = R2 = 10K, R3 = 600 โอห์ม

R4 เป็น Line Impedance มีค่าประมาณ 600 โอห์ม

แต่จากวงจรที่ใช้จริงนั้น เป็นไปได้ยากที่จะหาหม้อแปลงที่มี Line Impedcence เท่ากับ 600 โอห์ม พอดี ดังนั้นเรากำหนด R3 = 600 โอห์ม แล้วเราต่อตัวต้านทานปรับค่าได้ VR1 5K โอห์ม ขนานกับหม้อแปลง เราสามารถปรับค่าความต้านทานให้เหมาะสมได้โดยป้อน V1 เข้าไปจับสัญญาณ ที่ V3 ปรับค่า VR1 จนกระทั่ง V3 มีค่าน้อยที่สุด นั่นหมายถึง ค่า อิมพิแดนซ์ รวมมีค่าประมาณ 600 โอห์ม พอดี

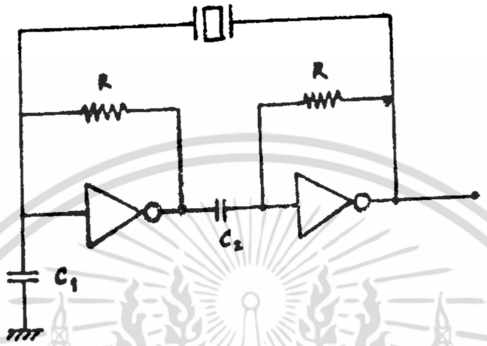


รูปที่ 3.2.2 วงจรที่ใช้งาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

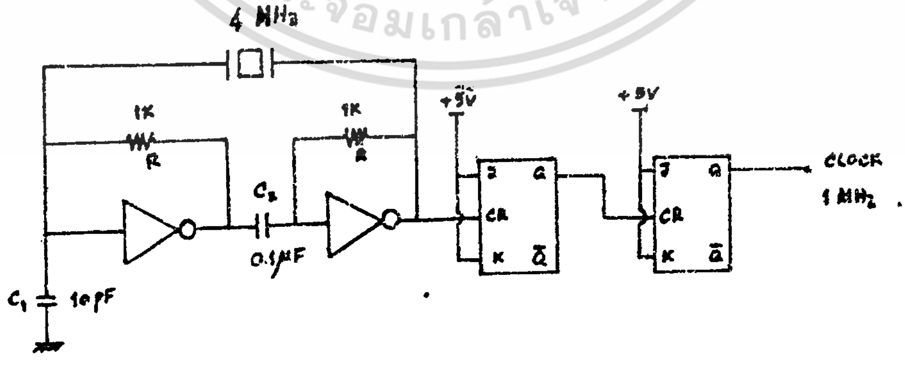
3.3 วงจรกำเนิดสัญญาณนาฬิกา

เราสามารถสร้างสัญญาณนาฬิกาที่มีความถี่ที่เที่ยงตรงได้ โดยใช้วงจรออสซิลเลเตอร์แบบใช้คริสตอล มีคริสตอลเป็นตัวป้อนกลับสัญญาณ จากเอาท์พุทสู่อินพุทแบบไม่กลับเฟส และนำนำเกทแบบอินเวทเตอร์ มาต่อเป็นวงจรขยายแบบลบ ที่มีอัตราการขยายสูงพอสมควรเมื่อต่อคาสเคดกันจะได้วงจรขยายแบบไม่กลับเฟส จะได้วงจรดังรูป 3.3.1



รูปที่ 3.3.1 วงจรสร้างสัญญาณนาฬิกา

สำหรับโครงงานนี้ ต้องการสัญญาณนาฬิกาความถี่ 1 MHz เนื่องจากทางผู้ทำโครงงานไม่สามารถหาคริสตอลความถี่ 1 MHz ได้ ดังนั้นจึงใช้คริสตอล 4 MHz สร้างสัญญาณความถี่ 4 MHz แล้วทำการหารความถี่ลง 4 เท่า โดยใช้วงจรหารความถี่ประกอบด้วย JK FLIP FLOP 2 ตัว ต่อเป็น T FLIP FLOP

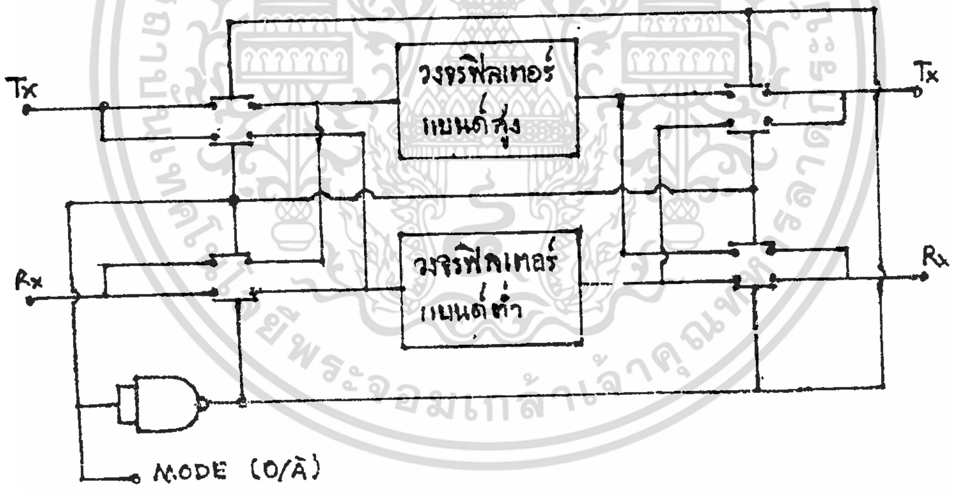


รูปที่ 3.3.2 วงจรสร้างสัญญาณนาฬิกา 1 MHz

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.4 ภาคควบคุมโหมดของฟิลเตอร์

เนื่องจากการติดต่อส่งข้อมูลกันนั้นมีการทำงานอยู่ 2 โหมดด้วยกันคือ โหมดเริ่มติดต่อ (Originate Mode) และโหมดตอบรับ (Answer Mode) ซึ่งกรณีที่ทำงานอยู่ในโหมดเริ่มติดต่อ โมเด็มจะส่งสัญญาณออกไปด้วยความถี่ในช่วงแถบความถี่ด้านต่ำ และรับสัญญาณเข้ามาในช่วงแถบความถี่ด้านสูง และในโหมดตอบรับก็จะทำในทางกลับกัน เพื่อที่จะให้โมเด็มสามารถทำงานได้ทั้งสองโหมด จึงต้องมีส่วนควบคุมเพื่อให้สัญญาณรับส่งนั้นเข้าสู่วงจรกรองความถี่ที่เหมาะสม โดยนำเอา IC 4066 ซึ่งเป็นอนาล็อกสวิทช์ (Analog Switch) ซึ่งสามารถควบคุมสวิทช์ให้ เปิด ปิด ได้โดยป้อน "1" หรือ "0" แก่ขาควบคุม และสามารถสร้างวงจรควบคุมโหมดของฟิลเตอร์ได้ดังนี้

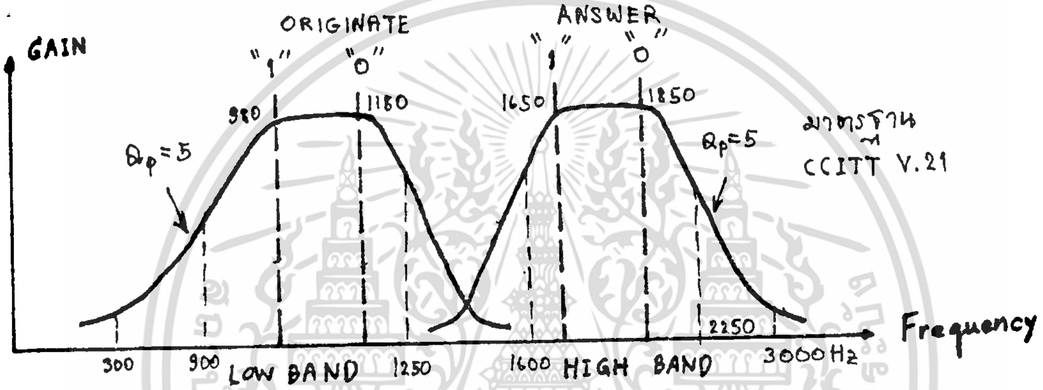


รูปที่ 3.4.1 วงจรควบคุมโหมดของฟิลเตอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.5 วงจรกรองสัญญาน

เพื่อให้โมเด็มในโครงงานนี้สื่อสารกันได้แบบสองทาง (Full Duplex) จึงได้จึงได้ทำการแบ่งช่วงพาสแบนด์ของโทรศัพท์ออกเป็นสองแบนด์คือ ช่วงแบนด์ต่ำและช่วงแบนด์สูง ดังกราฟในรูปที่ 3.5.1 ฝ่ายริ่มทำการติดต่อ จะส่งความถี่ 2 ความถี่ สำหรับแทนสัญญาณลอจิก "0" และ "1" ซึ่งจะอยู่ในช่วงแบนด์ต่ำ อีกฝ่ายหนึ่งคือ ฝ่ายตอบรับ จะต้องส่งความถี่อีกสอง ความถี่ที่แตกต่างจากฝ่ายเริ่มติดต่อ สำหรับแทนสัญญาณลอจิก "0" และ "1" โดยจะอยู่ในช่วงแบนด์สูง



รูปที่ 3.5.1 แสดงการแบ่งช่วงความถี่ของการติดต่อ

การเลือกชนิดของวงจรกรองสัญญาณ

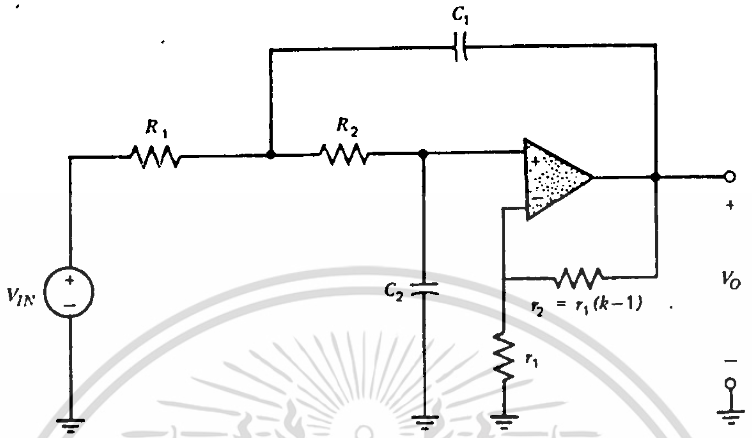
จากรูปที่ 3.5.1 จะเห็นว่า ความถี่คัทออฟทางด้านสูง (Higgh Frequency Cutoff) ของช่วงแบนด์ต่ำ กับความถี่คัทออฟทางด้านต่ำ (Low Frequency Cutoff) ของช่วงแแบนด์สูงมีค่าใกล้เคียงกัน ทำให้ต้องเลือกใช้วงจรแบนด์พาสฟิลเตอร์ โดยชุดของช่วงแบนด์ต่ำต้องออกแบบให้ ค่าควอลิตี้แฟคเตอร์ (Quality Factor-Q) ด้านความถี่สูงมีค่าสูง ขณะที่ชุดของช่วงแบนด์สูง ต้องออกแบบให้ค่า ควอลิตี้แฟคเตอร์ด้านความถี่ต่ำมีค่าสูง

ในโครงงานนี้เลือกใช้วงจรซาเลนแอนด์คีย์ (Sallen and Key's Circuit) แบบโลว์พาส และไฮพาส ต่อกันแบบคาสเคด (Cascade) โดยมีช่วงตอบสนองทางความถี่ที่ซ้อนกัน ทำให้ได้วงจรแบนด์พาสฟิลเตอร์ ที่มีผลตอบสนองทางความถี่อยู่ระหว่างช่วงที่เกิดการซ้อนกันของผลตอบสนองทางความถี่นั่นเอง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สูตรการคำนวณของวงจรซาลเลนแอนด์คีย์

1. วงจรซาลเลนแอนด์คีย์แบบโลว์พาส (Sallen and Key Low-pass Circuit)



รูปที่ 3.5.2 Low-pass Sallen and Key Circuit

ทรานส์เฟอร์ฟังชั่นของวงจรโลว์พาสคือ

$$T_{lp}(s) = \frac{K}{s^2 + (w_o/Q_o)s + w_o^2} \qquad \ldots(3.5.1)$$

จากรูปที่ 3.5.2 จะได้

$$V_o/V_{in} = \frac{k/R_1R_2C_1C_2}{s^2 + s[1/R_1R_2 + 1/R_2C_1 + (1-k)/R_2C_2] + 1/R_1R_2C_1C_2} \qquad \ldots(3.5.2)$$

ดังนั้น

$$w_o = [1/R_1R_2C_1C_2]^{1/2} \qquad \ldots(3.5.3)$$

$$Q_o = w_o/(bw)_o$$

$$= \frac{[1/R_1R_2C_1C_2]^{1/2}}{1/R_1R_2 + 1/R_2C_1 + (1-k)/R_2C_2} \qquad \ldots(3.5.4)$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$$K = k/R_1R_2C_1C_2 \quad ...(3.5.5)$$

2. <u>วงจรซาลเลนแอนด์คีย์ไฮพาส</u> (Sellen and Key High-pass circuit)

โดยการใช้การแปลง RC --> CR เพื่อแปลงโลว์พาสฟังก์ชั่น $T_{LP}(s)$ ที่มีขอบของช่วงพาสแบนด์อยู่ที่ $w = w_o$ ให้เป็นไฮพาสฟังก์ชั่น $T_{HP}(s)$ ซึ่งมีขอบของพาสแบนด์อยู่ที่ $w = w_o$ เช่นเดียวกัน

จะได้

$$T_{HP}(s) = T_{LP}(S)\Big|_{S = w_o^2/s} \quad ...(3.5.6)$$

จาก (3.5.1) ได้

$$T_{HP}(s) = \frac{s^2}{s^2 + (w_o/Q_o)s + w_o^2} \quad ...(3.5.7)$$

และเนื่องจากวงจรทั้งสองมีค่าความถี่คัทออฟเท่ากัน จากสมการ (3.5.1) และ (3.5.7) จะทำให้อัตราส่วนของทรานส์ซิชั่นแบนด์ (Transition Band) เท่ากัน นั่นคือ

$$\frac{w_{STOP}}{w_{PASS}}\Big|_{LP} = \frac{w_{PASS}}{w_{STOP}}\Big|_{HP} \quad ...(3.5.8)$$

จาก (3.5.1) เขียนใหม่ในรูป

$$T_{LP}(S) = \frac{1}{(1/w_o)^2S^2 + (1/w_oQ_o)S + 1} \quad ...(3.5.9)$$

ค่ามิติของความถี่ (Dimension of Frequency) ของวงจร RC คือ

$$\dim(w) = [R]^{-1}[C]^{-1} \quad ...(3.5.10)$$

$$\dim(w_o)^2 = [R]^{-2}[C]^{-2} \quad ...(3.5.11)$$

$$\dim(w_oQ_o) = [R]^{-1}[C]^{-2} \quad ...(3.5.12)$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ดังนั้นสมการ (3.5.9) เขียนใหม่ได้เป็น

$$T_{LP}(S) = \frac{1}{([R]^2[C]^2)S^2 + ([R][C])S + 1}$$

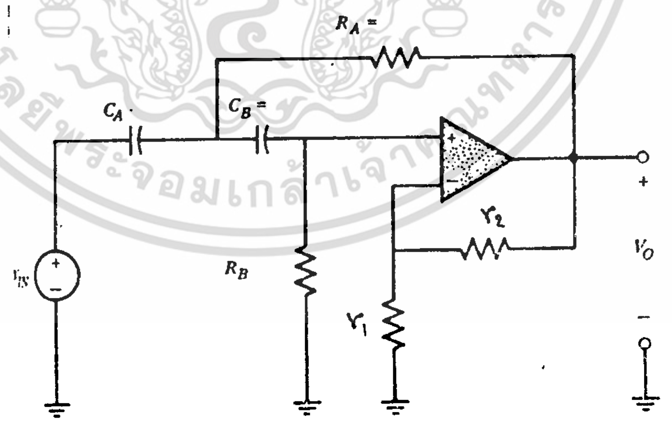
$$= \frac{1}{[R]^2([C]S)^2 + [R]([C]S) + 1} \qquad \ldots(3.5.13)$$

จากสมการ (3.5.6) การแปลง LP --> HP ต้องแทน S ด้วย w_o/s จึงทำให้สามารถแทน

$$R \;--->\; Rw_o/S \qquad \ldots(3.5.14)$$

$$C \;--->\; Cw_o/S \qquad \ldots(3.5.15)$$

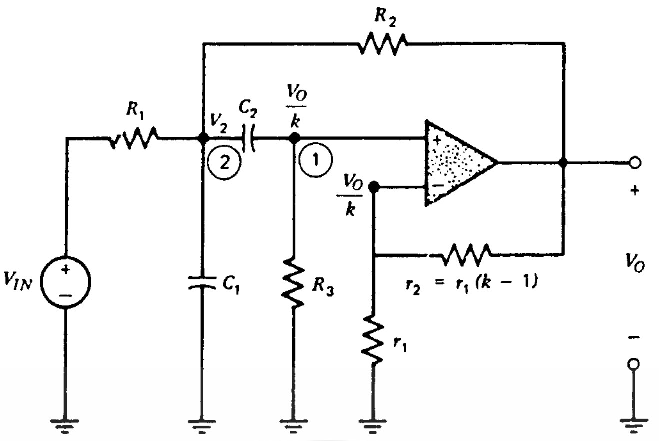
นั่นคือ ในวงจรโลว์พาสรูปที่ 3.5.2 สามารถทำให้เป็นวงจรไฮพาสได้ด้วยการแทนตัวความต้านทาน R ในวงจรรูปที่ 3.5.2 ด้วยตัวเก็บประจุค่า $1/Rw_o$ และแทนตัวเก็บประจุ C ด้วยตัวต้านทานค่า $1/Cw_o$ ซึ่งจะเป็นวงจรไฮพาสฟิลเตอร์ในรูปที่ 3.5.3



รูปที่ 3.5.3 High-pass Sallen and Key Circuit

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.วงจรซาลเลนแอนด์คีย์แบนด์พาส (Sallen and Key Band-pass Circuit)



รูปที่ 3.5.4 Sallen and Key Band-pass Circuit

ทรานสเฟอร์ฟังก์ชั่น ของวงจรแบนด์พาสฟิลเตอร์ คือ

$$T_{sr} = \frac{s}{s^2 + (w_o/Q_o)s + w_o^2} \qquad \ldots(3.5.16)$$

จากรูปที่ 3.5.4 ได้

$$V_o/V_{in} = \frac{ks/R_1C_1}{s^2+s(1/R_1C_1+1/R_3C_2+1/R_3C_1+(1-k)/R_2C_1)+(R_1+R_2)/R_1R_2R_3C_1C_2} \qquad \ldots(3.5.17)$$

ถ้าให้ $C_1 = C_2 = 1$, $R_1 = R_2 = R_3 = R$

เปรียบเทียบสมการ (3.5.16) กับสมการ (3.5.17) ได้

$$R = R_1 = R_2 = R_3 = (2)^{1/2}/w_o \qquad \ldots(3.5.18)$$

$$k = 1 + r_2/r_1 \qquad \ldots(3.5.19)$$

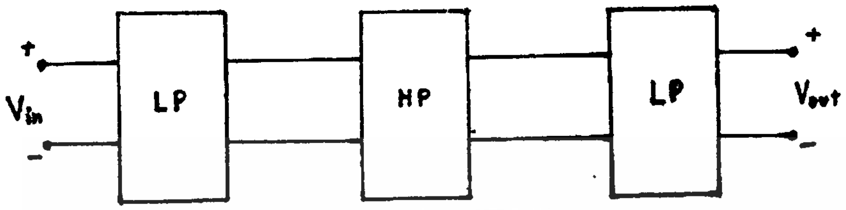
$$K = k/R_1C_1$$

$$= w_o[2(2)^{1/2} - 1/Q_o] \qquad \ldots(3.5.20)$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การออกแบบวงจรกรองสัญญาณที่ใช้ในโครงงาน

1.วงจรกรองสัญญาณชุดช่วงแบนด์ต่ำ



รูปที่ 3.5.5 วงจรกรองสัญญาณของชุดช่วงแบนด์ต่ำ

จากรูปที่ 3.5.5 เป็นวงจรที่ต่อให้ เป็นวงจรแบนด์พาส โดยการนำเอาวงจรโลว์พาสที่มีคัทออฟ เท่ากับไฮคัทออฟของวงจรแบนด์พาส มาต่อแบบคาสเคด (Cascade) กับวงจรไฮพาส ที่มีคัทออฟเท่ากับโลว์คัทออฟของวงจรแบนด์พาส เพื่อให้ค่าควอลิตี้แฟคเตอร์ ของแต่ละด้านมีค่าสูงพอประมาณ และสามารถออกแบบโดยให้มีอิสระต่อกัน นอกจากนี้ยังเพิ่มวงจรโลว์พาสซึ่งมีคุณสมบัติเหมือนกันต่อแบบคาสเคดเข้าไปอีกชุดหนึ่ง เพื่อเพิ่มค่า Q ด้านไฮคัทออฟให้มีค่าสูงขึ้นไปอีก

วงจรโลว์พาส 2 ชุด

คำนวณค่า $R_1, R_2, C_1, C_2, r_1, r_2$ ของรูปที่ 3.5.2

จากสมการ (3.5.3) , (3.5.4) และ (3.5.5) กำหนดให้

$$k=1 \;,\; R_1=R_2=1 \qquad \ldots(3.5.21)$$

และจะได้

$$C_1 = 2Q_o/w_o \qquad \ldots(3.5.22)$$

$$C_2 = 1/2w_oQ_o \qquad \ldots(3.5.23)$$

$$r_2 = 0 \text{ โอห์ม} \qquad \ldots(3.5.24)$$

จากการกำหนดเงื่อนไข ตามสมการ (3.5.21) จะทำให้วงจร รูปที่ 3.5.2 มีค่า

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เซนซิติวิตี้ (Sensitivity) ต่ำ และคงที่ไม่ขึ้นกับพารามิเตอร์ w_o และ Q_o

จากกราฟรูปที่ 3.5.1 ได้กำหนด $w_o = 2*\pi*1250$ rad/sec. , $Q_o = 5$

จาก (3.5.22) $C_1 = 2Q_o/w_o \quad = 2*5/(2*\pi*1250)$

$= 1.27324*10^{-3}$

จาก (3.5.23) $C_2 = 1/2Q_ow_o \quad = 1/(2*2*\pi*1250*5)$

$= 1.27324*10^{-5}$

สเกลล์ด้วยค่า $2.7*10^4$

$R_1 = R_2 = 27.1$ K โอห์ม ใช้ 27 K โอห์ม

$C_1 = 47$ nF

$C_2 = 0.47$ nF = 470 pF

วงจรไอพาส 1 ชุด

จากกราฟรูปที่ 3.5.1 ได้กำหนด $W_o = 2*\pi*900$ rad/sec. , $Q_o = 5$

หาวงจรโลว์พาสตามรูปที่ 3.5.2 ก่อน ได้

$R_{LP1} = R_{LP2} = 1$

$C_{LP1} = 2Q_o/w_o = 2*5/(2*\pi*900)$

$= 1.7684*10^{-3}$

$C_{LP2} = 1/2w_oQ_o = 1/(2*2*\pi*900*5)$

$= 1.7684*10^{-5}$

แปลง LP --> HP ด้วยสมการ (3.5.14) , (3.5.15) เพื่อหาค่า C_A , C_B , R_A , R_B ตามรูปที่ 3.5.3

$C_A = C_B = 1/(w_oR_{LP1}) \quad = 1/(2*\pi*900*1)$

$= 1.7684*10^{-4}$

$R_A = 1/(w_oC_{LP1}) \quad = 1/(2*\pi*900*1.7684*10^{-3})$

$= 0.1$

$R_B = 1/(w_oC_{LP2}) \quad = 1/(2*\pi*900*1.7684*10^{-5})$

$= 10$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สเกลด้วย $1.474 * 10^4$

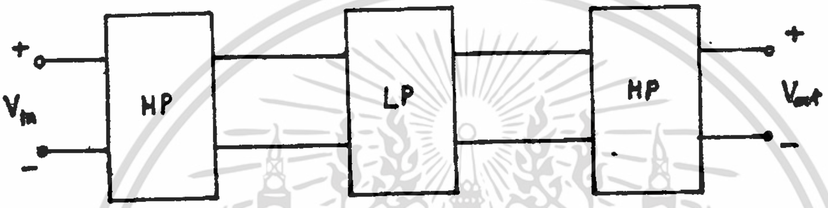
$C_A = C_B = 12nF$

$R_A = 1.47 * 10^3$ โอห์ม ใช้ 1.5 K โอห์ม

$R_B = 1.47 * 10^5$ โอห์ม ใช้ 150 K โอห์ม

ซึ่งจะได้วงจรชุดช่วงแบนด์ต่ำ ดังรูปที่ 3.5.7

<u>2.วงจรกรองสัญญาณชุดช่วงแบนด์สูง</u>



รูปที่ 3.5.6 วงจรกรองสัญญาณของชุดช่วงแบนด์สูง

จากรูปที่ 3.5.6 เป็นวงจรที่ต่อให้เป็นวงจรแบนด์พาส มีหลักการเหมือนชุดช่วงแบนด์ต่ำ แต่เพิ่มวงจรไฮพาสขึ้นอีกชุดเพื่อเพิ่มค่า Q ด้าน Low cutoff ขึ้นอีก

วงจรโลว์พาส 1 ชุด

จากกราฟรูปที่ 3.5.1 ได้กำหนด $w_o = 2*\pi*2250$ rad/sec. , $Q_o = 5$

จากรูปที่ 3.5.2 ได้ $R_1 = R_2 = 1$

$C_1 = 2Q_o/w_o = 2*5/(2*\pi*2250)$

$= 7.0736*10^{-4}$

$C_2 = 1/2w_oQ_o = 1/(2*2*\pi*2250*5)$

$= 7.0736*10^{-6}$

สเกลด้วย $1.505 * 10^4$

ได้ $R_1 = R_2 = 15.05$ K โอห์ม ใช้ 15 K โอห์ม

$C_1 = 47$ nF

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$C_2 = 0.47\ nF$

$= 470\ pF$

วงจรไฮพาส 2 ชุด

จากกราฟรูปที่ 3.5.1 $w_o = 2*\pi*1650$ rad/sec. , $Q_o = 5$

หาวงจรโลว์พาสตามรูปที่ 3.5.2 ก่อน

$R_{LP1} = R_{LP2} = 1$

$C_{LP1} = 2Q_o/w_o = 2*5/(2*\pi*1650)$

$= 9.646*10^{-4}$

$C_{LP2} = 1/2Q_o w_o = 1/(2*2*\pi*1650*5)$

$= 9.646*10^{-6}$

แปลง LP --> HP ได้

$C_A = C_B = 1/w_o R_{LP1} = 1/(2*\pi*1650*1)$

$= 9.646*10^{-5}$

$R_A = 1/w_o C_{LP1} = 1/(2*\pi*1650*9.646*10^{-4})$

$= 0.1$

$R_B = 1/w_o C_{LP2} = 1/(2*\pi*1650*9.646*10^{-6})$

$= 10$

สเกลด้วย $2.05*10^4$

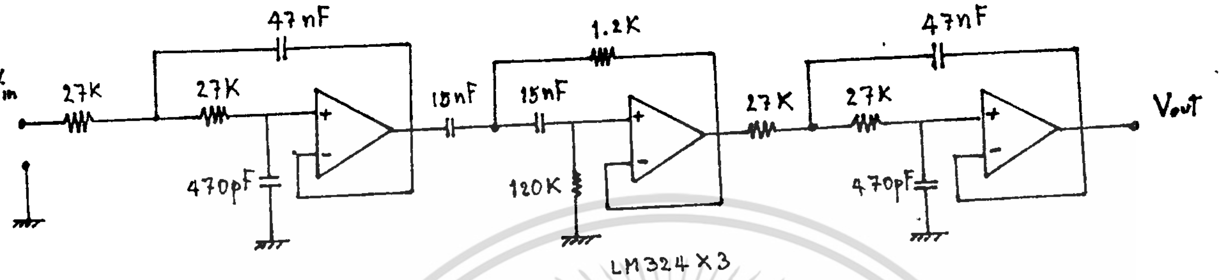
$C_A = C_B = 4.7\ nF$

$R_A = 2.05\ K$ โอห์ม

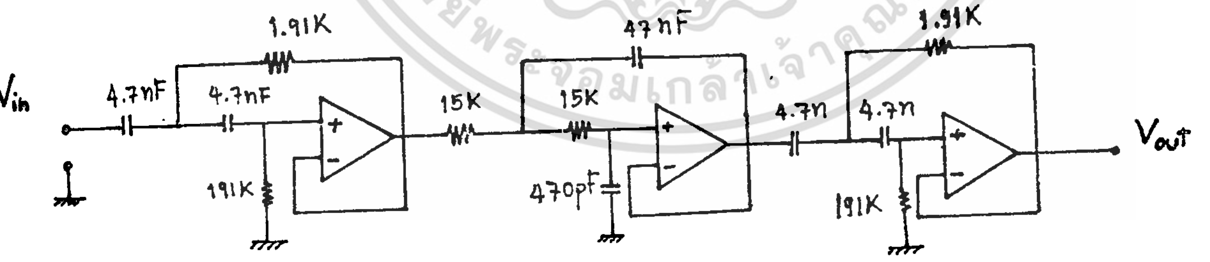
$R_B = 205\ K$ โอห์ม

ซึ่งจะได้วงจรตามรูปที่ 3.5.8

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.5.7 วงจรกรองสัญญาณชุดช่วงความถี่ต่ำ

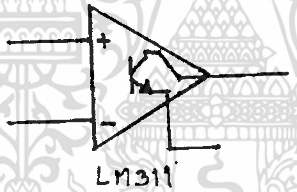


รูปที่ 3.5.8 วงจรกรองสัญญาณชุดช่วงความถี่สูง

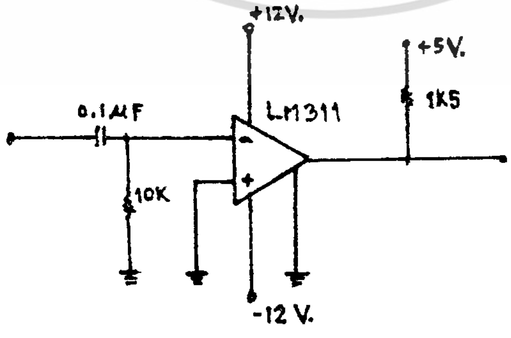
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.6 ลิมิตเตอร์ (LIMITTER)

วงจรลิมิตเตอร์เป็นตัวแปลงสัญญาณที่ได้รับมาจากอีกฝ่าย มาเป็นสัญญาณสี่เหลี่ยมมีความถี่เท่ากันกับอินพุท ซึ่งจะเป็นรูปแบบที่ส่วนดีมอดดูเลท สามารถดีมอดออกมาเป็นสัญญาณทางดิจิตอลได้ จากวงจรใช้ LM 311 ซึ่งเป็นวงจรเปรียบเทียบ (COMPARATER) ข้อดีของไอซีเบอร์นี้ก็คือ เราสามารถควบคุมระดับของสัญญาณเอาท์พุทให้สวิงอยู่ในค่าที่ต้องการได้ เนื่องจากภาคเอาท์พุทของ LM 311 จะมีทรานซิสเตอร์ที่ขา คอลเลคเตอร์ และอิมิตเตอร์ ถูกปล่อยลอยไว้ ส่วนขาเบสจะเป็นตัวรับสัญญาณจากภาคขยาย ดังนั้น ถ้าต้องการให้เอาท์พุทสวิงอยู่ในช่วง +5 Volts กับ 0 Volt จึงทำการต่อขาอิมิตเตอร์ลงกราวน์ และต่อตัวต้านทานจากขาคอลเลคเตอร์ไป +5 Volts แต่ค่าของตัวต้านทานนี้ต้องไม่มีค่าน้อยเกินไป มิฉะนั้นขณะที่ทรานซิสเตอร์อิ่มตัวจะมีกระแสไหลมากเกินไป อาจเกิดการเสียหายขึ้นได้ ดังรูปที่ 3.6.1 และจะได้วงจรลิมิเตอร์ดังรูปที่ 3.6.2



รูปที่ 3.6.1 แสดงภาคเอาท์พุทของ LM 311



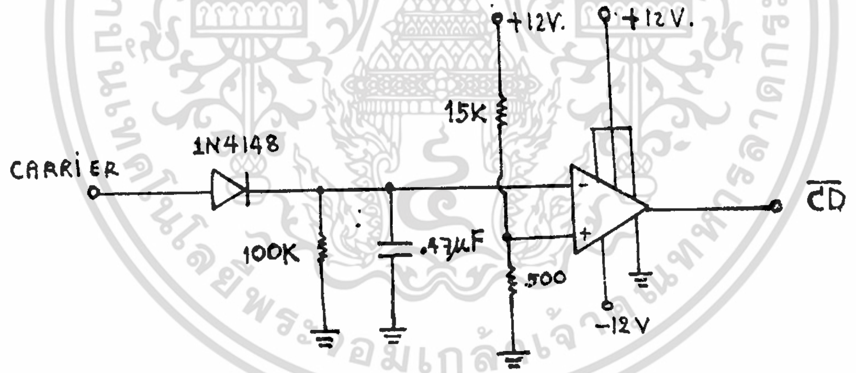
รูปที่ 3.6.2 วงจรลิมิตตอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.7 ภาคแคเรียร์ดีเทค (CARRIER DETECT)

วงจรภาคนี้จะทำหน้าที่คอยตรวจว่ามีสัญญาณส่งมาจากอีกฝ่ายหรือไม่ ถ้าตรวจจับได้ จะให้สัญญาณออกมาเป็น "0" และจะเรียกเอาท์พุทจากภาคนี้ว่าสัญญาณ "CD" ซึ่งจะมีการนำเอาสัญญาณนี้ไปเข้า พอร์ท RS 232C เพื่อใช้ในการควบคุมการรับส่งของข้อมูล เราสามารถออกแบบวงจรได้โดย ป้อนสัญญาณอินพุทเข้าสู่วงจรตัดสัญญาณแบบครึ่งคลื่น (Half Wave Rectifier) จากนั้นต่อเข้าวงจรกรองความถี่แบบโลว์พาส (Low-pas Filter) จะได้เอาท์พุทออกมาเป็นระดับแรงดันไฟตรงค่าหนึ่ง

แรงดันค่านี้จะถูกนำไปเปรียบกับระดับแรงดันอ้างอิงค่าหนึ่ง วงจรเปรียบเทียบจะทำการเปรียบเทียบสัญญาณทั้งสอง ถ้าระดับสัญญาณจาก วงจรตัดสัญญาณมีค่ามากกว่า ระดับสัญญาณอ้างอิง วงจรเปรียบเทียบจะให้เอาท์พุทเป็น "0" แต่ถ้าน้อยกว่าจะเป็น "1" ในการใช้งานต้องออกแบบให้แรงดันอ้างอิง มีค่ามีค่าต่ำกว่าแรงดันจากวงตัดสัญญาณ เมื่อมีสัญญาณส่งมาจากอีกฝ่าย และมีค่าสูงกว่าเมื่อ ไม่มีสัญญาณส่งเข้ามา จะได้วงจรดังรูปที่ 3.7



รูปที่ 3.7 วงจรแคเรียร์ดีเทค

3.8 ภาคมอดดูเลทและดีมอดดูเลท

ส่วนนับว่าเป็นหัวใจสำคัญของโมเด็ม สำหรับโครงงานนี้ใช้ไอซีเบอร์ MC 14412 ทำการมอดดูเลทแบบ FSK (Frequency Shift Keying) ใช้ความเร็วในการติดต่อ 300 บิตต่อวินาที

ลักษณะทั่วไป

-มีออสซิเลเตอร์ภายใน ซึ่งจะต่อกับคริสตอล 1 MHz ภายนอก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

-ทำงานได้ทั้งแบบฝ่ายเริ่มติดต่อ และฝ่ายตอบรับ

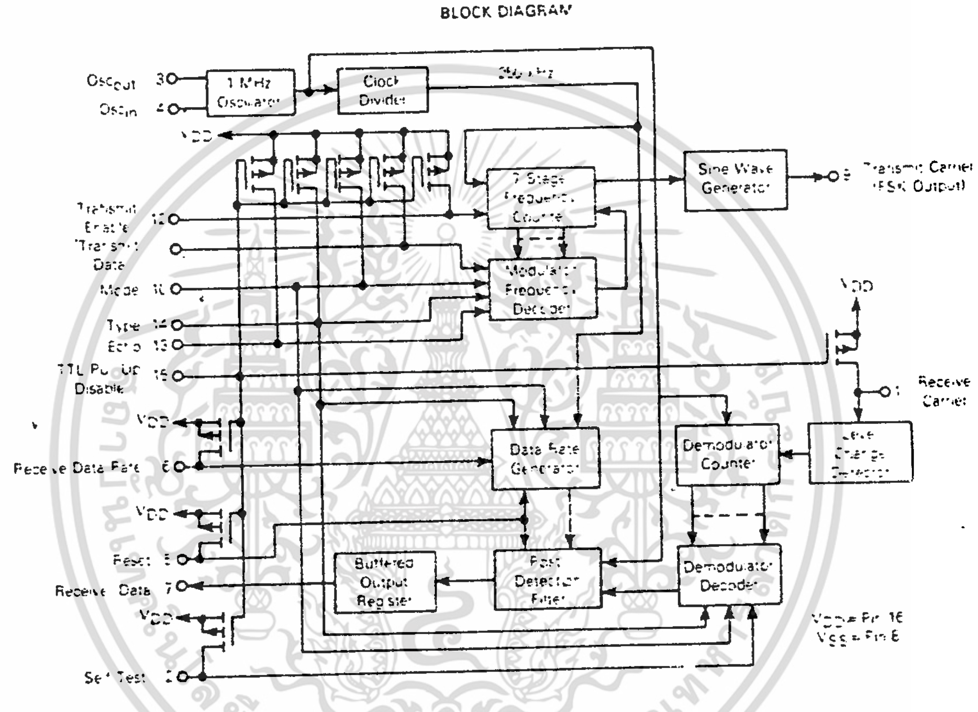
-มีตัวกำเนิดความถี่สัญญาณซายน์ภายใน

-มีระบบทดสอบตัวเอง

-สามารถรับส่งได้ทั้งแบบทางเดียว กึ่งสองทาง และ สองทาง

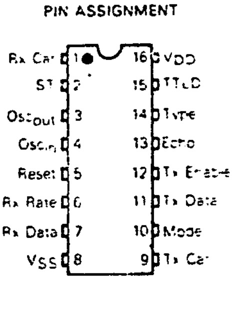
-เลือกอัตราเร็วของการส่งได้ 0-300 BPS และ 0-600 BPS

-ใช้ได้กับทั้งไอซีแบบ TTL และ CMOS



รูปที่ 3.6.1 บล็อกไดอะแกรมของส่วนประกอบภายใน MC 14412

ลักษณะของขาไอซี MC 14412

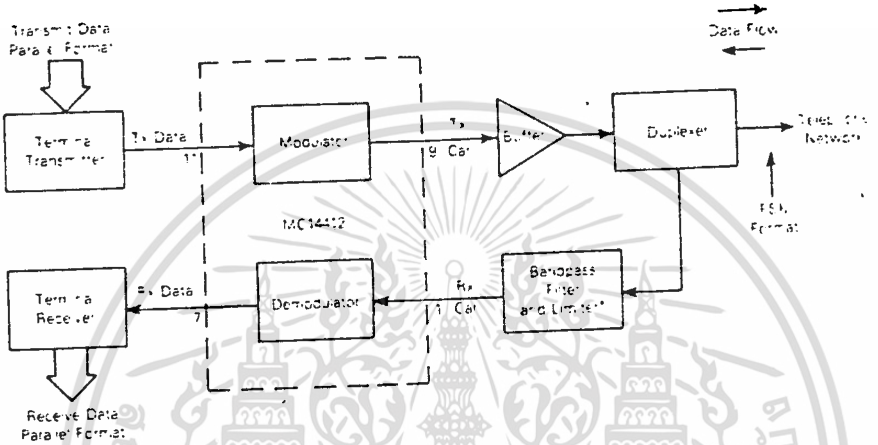


รูปที่ 3.6.2 ลักษณะของขา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

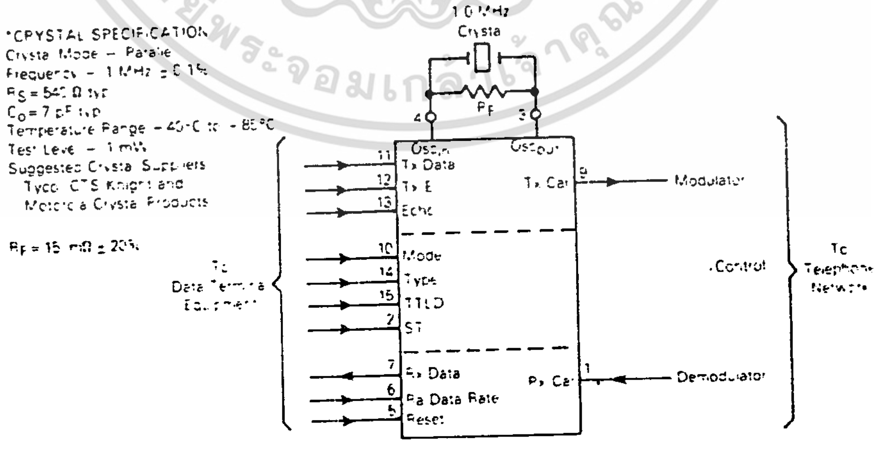
การทำงานของ MC 14412

จากรูปที่ 3.8.3 แสดงการใช้งานไอซีโมเด็มเบอร์นี้ ข้อมูลจะถูกส่งแบบอนุกรมไป ยังมอดดูเลเตอร์ เพื่อเปลี่ยนเป็นสัญญาณซายน์ที่มีความถี่ สำหรับส่งไปตามสายโทรศัพท์ได้ ที่เอาท์พุทของของตัวมอดดูเลเตอร์ จะมีบัฟเฟอร์และวงจรขยาย ก่อนที่จะส่งไปยังสายโทรศัพท์ ซึ่งมีอิมพิแดนซ์ 600 โอห์ม



รูปที่ 3.8.3 ลักษณะการใช้งาน

อินพุท/เอาท์พุทฟังชั่น



รูปที่ 3.8.4 สัญญาณอินพุท/เอาท์พุท

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

<u>สัญญาณ TYPE (ขา 14)</u>

เป็นสัญญาณที่ใช้ในการเลือกระบบ U.S.A. (BELL) หรือ C.C.I.T.T. โดยเมื่อ

TYPE = "1" จะเลือกระบบ U.S.A.

TYPE = "0" จะเลือกระบบ C.C.I.T.T.

<u>สัญญาณ TRANSMIT DATA (Tx data , ขา 11)</u>

เป็นสัญญาณอินพุทแบบไบนารี่ (BINARY) ซึ่งจะถูกส่งไปมอดดูเลทให้เป็นสัญญาณ ซายน์ (ที่มีฮาโมนิกปนอยู่) ด้วยวิธี FSK

Tx = "1" แทน Mark

Tx = "0" แทน Space

<u>สัญญาณ TRANSMIT CARRIER (Tx Car , ขา 9)</u>

คือสัญญาณ Tx data ที่ถูกมอดดูเลทมีลักษณะเป็นสัญญาณซายน์ แบบดิจิตอลซินธีไซส์ (Digital-synthesized sine wave) จากออสซิลเลเตอร์ 1 MHz ที่ขานี้จะมีเอาท์พุท อิมพิแดนซ์ 5 โอห์ม

ความถี่ของสัญญาณ Tx Car แสดงได้ดังนี้

TYPE = "0" (U.S.A)

ECHO = "0"

MODE		Tx data		Tx Car
ORIGINAL	"1"	MARK	"1"	1270 Hz
ORIGINAL	"1"	SPACE	"0"	1070 Hz
ANSWER	"0"	MARK	"1"	2225 Hz
ANSWER	"0"	SPACE	"0"	2025 Hz

ตารางที่ 3.8.1 ความถี่ของสัญญาณ Tx Car เมื่อใช้มาตราฐาน BELL 103

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

TYPE = "0" (C.C.I.T.T.)

ECHO = "0"

MODE		Tx data		Tx Car
CHANNEL	"1"	MARK	"1"	980 Hz
No.1	"1"	SPACE	"0"	1180 Hz
CHANNEL	"0"	MARK	"1"	1650 Hz
No.2	"0"	SPACE	"0"	1850 Hz

ตารางที่ 3.8.2 ความถี่ของสัญญาณ Tx Car เมื่อใช้มาตราฐาน C.C.I.T.T. V.21

TYPE = "0"

ECHO = "1"

MODE		Tx data	Tx Car
CHANNEL No.2	"0"	"1"	2100 Hz

ตารางที่ 3.8.3 ความถี่ของสัญญาณ Tx Car เมื่อมีการใช้สัญญาณ ECHO

สัญญาณ TRANSMIT ENABLE (Tx ENABLE , ขา 12)

เป็นขาที่ใช้ควบคุมว่าจะให้มีสัญญาณ Tx Car ออกมาหรือไม่

Tx ENABLE = "1" จะมีสัญญาณ Tx Car ออกมา

Tx ENABLE = "0" จะไม่มีสัญญาณ Tx Car ออมา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สัญญาณ MODE (ขา 10)

ใช้กำหนดการทำงานของโมเด็มว่าจะทำงานอยู่ในโหมดใด

MODE = "1" จะทำงานในโหมดเริ่มติดต่อ

MODE = "0" จะทำงานในโหมดตอบรับ

สัญญาณ ECHO (ขา 13)

เมื่อเราให้ ECHO = "1" [TYPE = "0", MODE = "0", Tx data = "1"] มอดดูเลเตอร์จะส่งสัญญาณความถี่ 2100 Hz ออกมา

ในการใช้งานปกติ เราให้ ECHO = "0"

สัญญาณ RECEIVE DATA (Rx Data , ขา 7)

เป็นสัญญาณดิจิตอลที่ทำการดีมอดมาจากสัญญาณ Receive Carrier

สัญญาณ RECEIVE CARRIER (Rx Car , ขา 1)

เป็นสัญญาณอินพุทเข้าสู่ภาคดีมอดดูเลเตอร์ สามารถใช้ได้ทั้งระดับสัญญาณจาก TTL และ CMOS จะต้องมีดิวตี้ไซเคิล 50% ±2% มีลักษณะเป็นสัญญาณสี่เหลี่ยมได้มาจากวงจรลิมิตเตอร์

สัญญาณ RECEIVE DATA RATE (Rx rate , ขา 5)

ใช้กำหนดอัตราเร็วในการรับส่ง

Data Rate	Rx Rate
0-300 bps	"1"
0-600 bps	"0"

สัญญาณ SELF TEST (ST , ขา 2)

เมื่อ ST = "1" ดีมอดดูเลเตอร์จะถูกสวิทช์ความถี่ในการดีมอด ไปเท่ากับความถี่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ของการมอดดูเลทของตัวเอง และทำการดีมอดดูเลทสัญญาณแบบ FSK

สัญญาณ RESET (ขา 5)

ถ้าสัญญาณขานี้เท่ากับ "1" โมเด็มจะไม่ทำการดีมอดดูเลทสัญญาณ

สัญญาณ CRYSTAL (OSCin,OSCout , ขา 4 และขา 3 ตามลำดับ)

เราใช้คริสตอล 1 MHz สำหรับออสซิเลเตอร์ในไอซีตัวนี้ หรืออาจใช้สัญญาณนาฬิกา ความถี่ 1 MHz จากภายนอกป้อนเข้าขา OSCin โดยปล่อยขา OSCout ลอยไว้ก็ได้

สัญญาณ TTL PULL-UP DISABLE (TTLD , ขา 15)

เพื่อเพิ่มความสามารถในการต่อกับไอซี TTL จะต้องมีการ Pull-up แต่เมื่อเรา ต่อกับไอซีพวก CMOS ก็ไม่จำเป็นต้องมีการ Pull-up เพื่อลดการสูญเสียของพลังงาน

เมื่อ TTLD = "1" จะไม่มีการ Pull-up

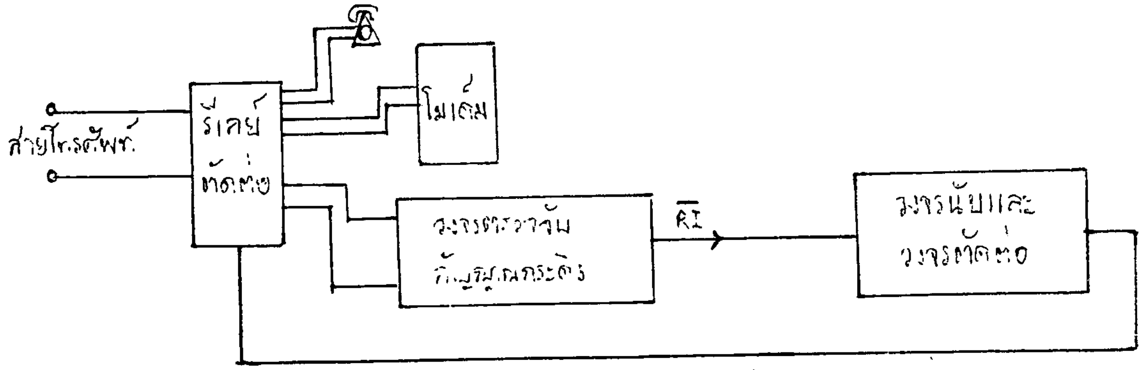
TTLD = "0" ตัวความต้านทาน Pull-up จะถูกใส่เพื่อช่วยในการจ่ายกระแส

3.9 ภาครับโทรศัพท์อัตโนมัติ

วงจรในส่วนนี้จะทำให้การติดต่อรับส่งข้อมูลมีประสิทธิภาพ และความสะดวกสบาย เพิ่มขึ้น เมื่อฝ่ายเริ่มทำการติดต่อออกไป ฝ่ายตอบรับจะสามารถสื่อสารข้อมูลกันได้โดยไม่จำเป็น ต้องมีคนคอยรับโทรศัพท์ ทำให้สามารถส่งข้อมูลไปยังฝ่ายรับได้ทั้งๆที่ อีกฝ่ายไม่มีคนอยู่ ดังนั้นการ ทำงานจึงสะดวกขึ้นมาก แต่อย่างไรก็ตาม ฝ่ายที่ทำการรับโทรศัพท์อัตโนมัตินั้น จะต้องรันโปร แกรมสำหรับการรับส่งข้อมูล ไว้เรียบร้อยแล้ว

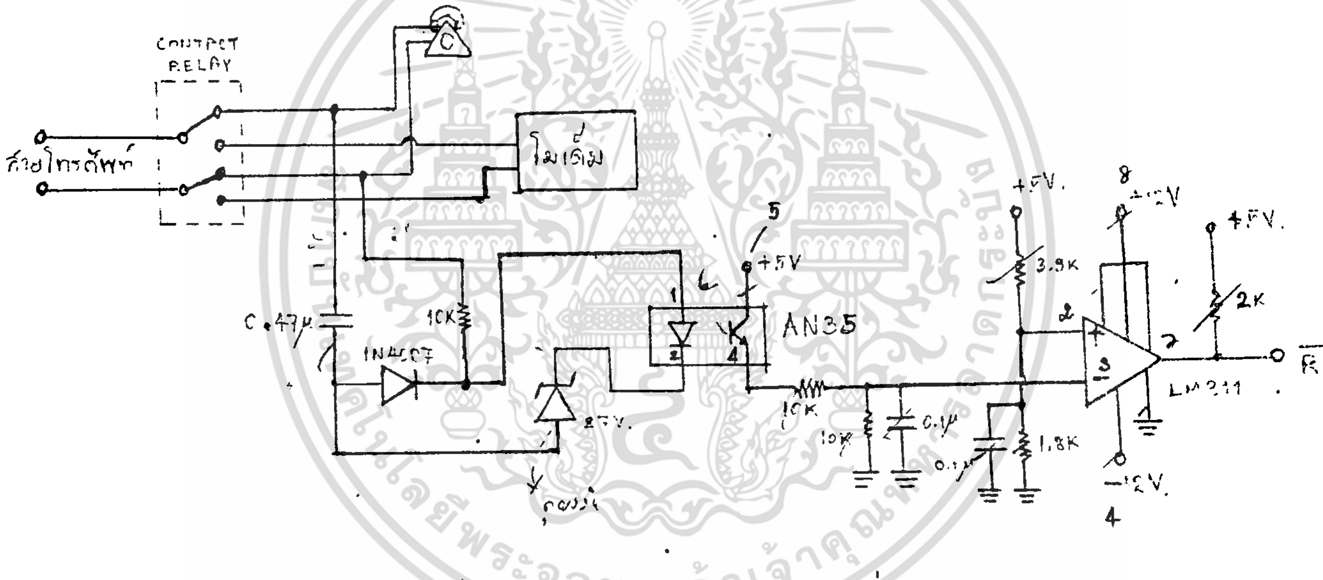
การออกแบบวงจรทำได้โดย ใช้วงจรตรวจจับสัญญาณกระดิ่ง เมื่อมีสัญญาณกระดิ่ง เข้ามา (นั่นคือได้รับการติดต่อจากอีกฝ่าย) จะมีเอาท์พุทเป็นสัญญาณพัลส์สี่เหลี่ยมขึ้น ส่งไปยัง วงจรนับ เมื่อนับถึงค่าที่ต้องการแล้วก็จะทำการตัดต่อให้สายโทรศัพท์ มาเชื่อมเข้ากับโมเด็ม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.9.1 บล็อคไดอะแกรมของภาครับโทรศัพท์อัตโนมัติ

3.9.1 วงจรตรวจจับสัญญาณกระดิ่ง



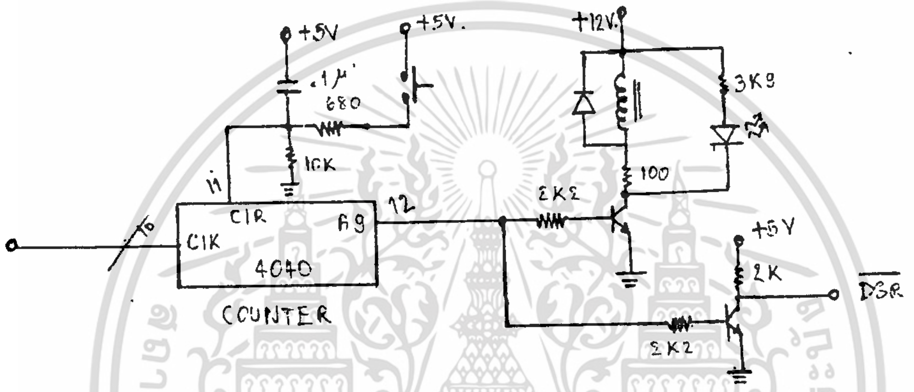
รูปที่ 3.9.2 วงจรตรวจจับสัญญาณกระดิ่ง

เมื่อมีเสียงกระดิ่งดังขึ้นนั่นคือ เกิดสัญญาณกระดิ่งแรงดันที่สายโทรศัพท์จะเป็นสัญญาณพัลซ์ มีค่าอยู่ในช่วงระหว่าง 40 ถึง 100 โวลต์ ตามจังหวะของเสียงกระดิ่ง และเพื่อความปลอดภัยของวงจรจึงทำการแยกกราวน์ของส่วนนี้ออกไป โดยใช้ ออปโต คัปเปลอร์ (Opto Coupler) ถ้ามีสัญญาณพัลซ์เข้ามาจะทำให้ มีกระแสไหลผ่านทรานซิสเตอร์ภายในของออปโตคัปเปลอร์ ดังนั้นที่ขาลบของออปแอมป์จะสูงกว่าที่ขาบวก ทำให้ได้เอาท์พุทเป็นศูนย์โวลต์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.9.2 วงจรนับและวงจรตัดต่อ

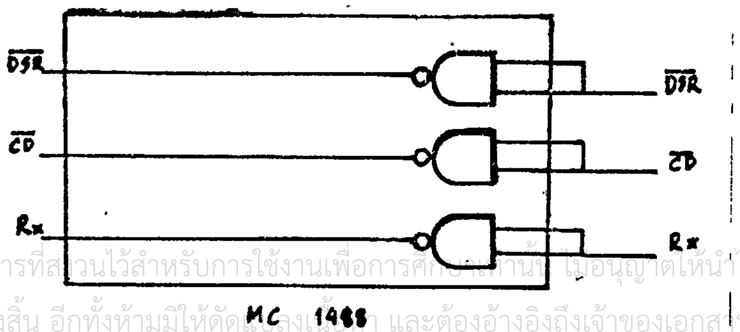
เราใช้ไอซีเบอร์ 4040 โดยป้อนสัญญาณ RI จากวงจรที่แล้วป้อนเข้าขาอินพุท เมื่อนับถึงค่าที่ต้องการ เอาพุทก็จะมีค่าเท่ากับลอจิก "1" ทำให้ทรานซิสเตอร์ทั้ง 2 ตัวอิ่มตัว โดยตัวแรกจะทำหน้าที่ขับรีเลย์ ซึ่งขณะนี้หน้าสัมผัสกับรีเลย์ จะเด้งไปอยู่ที่ด้านโมเด็ม ทำให้โมเด็มพร้อมที่จะติดต่อได้ และทรานซิสเตอร์อีกตัวจะใช้เป็นตัวสร้างสัญญาณ DSR (Data Set Ready) ซึ่งจะเป็นสัญญาณที่บอกคอมพิวเตอร์ว่า อุปกรณ์โมเด็มพร้อมที่จะติดต่อรับส่งข้อมูล



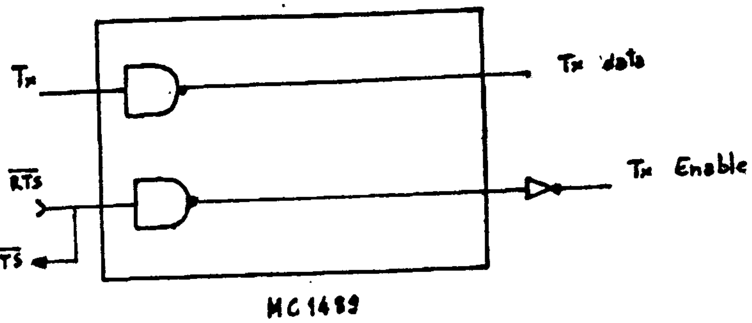
รูปที่ 3.9.3 วงจรนับและวงจรตัดต่อ

3.10 ภาคแปลงระดับสัญญาณสำหรับติดต่อกับ RS 232C PORT

ในการติดต่อกับคอมพิวเตอร์ต้องทำการติดต่อผ่าน พอร์ท RS 232C ซึ่งจะมีระดับสัญญาณเป็น ± 12 โวลต์ แต่ว่าสัญญาณติดต่อต่างๆ จากโมเด็มนั้นมีขนาดระหว่าง ศูนย์ถึงห้าโวลต์เท่านั้น ดังนั้นเราต้องทำการแปลงระดับสัญญาณเพื่อให้สามารถใช้งานกันได้ สำหรับโครงงานนี้ใช้ไอซี MC 1488 และ MC 1489 ดังรูปที่ 3.10.1



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.10.1 แสดงการติดต่อกับ RS 232C PORT

ดังนั้นในการใช้งานจะมีการใช้ขาของ RS 232C จำนวน 7 ขาด้วยกันคือ Tx,Rx, RTS,CTS,DSR, Signal Ground และ CD ซึ่งสำหรับความหมายของขาต่างๆ ได้กล่าวไว้แล้วในบทที่ 2

3.11 ความรู้พื้นฐานสำหรับการเขียนโปรแกรมภาษาเบสิค ในการติดต่อของพอร์ตสื่อสาร

บนเครื่องไมโครคอมพิวเตอร์ 16 บิท จะมีส่วนที่เรียกว่า อะซิงโครนัสอะแดปเตอร์ ซึ่งจะมีพอร์ตต่อมาใช้งานโดย ไอบีเอ็ม พีซี กำหนดเป็น com1 และ com2 โปรแกรมในไบออสจะเขียนสนับสนุนการใช้พอร์ตสื่อสารนี้แล้ว แต่ถ้าเราไม่ต้องการติดต่อกับพอร์ตสื่อสารในลักษณะเชื่อมกับไบออส เราก็อาจใช้ภาษาชั้นสูงในการใช้งาน ซึ่งภาษาที่ใช้สะดวก และมีประสิทธิภาพสูงก็ได้แก่ภาษาเบสิค

<u>การทำงานของโปรแกรม</u>

ในการเขียนโปรแกรมส่งข้อมูลทางพอร์ตสื่อสารนี้ สามารถส่งข้อมูลไปได้ทั้งแบบอีมูเลชั่นเทอร์มินัล (Emulation terminal) และส่งข้อมูลเป็นไฟล์ (Data File) โดยมีรายละเอียดการทำงานดังนี้

<u>โปรแกรมอีมูเลชั่นเทอร์มินัล</u>

เมื่อเราต้องการให้คอมพิวเตอร์ทำงานเป็นเทอร์มินัล เราก็เขียนโปรแกรมควบคุมพอร์ตสื่อสาร ให้มีการทำงานเป็นเหมือนเทอร์มินัล โดยให้โปรแกรมอ่านค่าจากคีย์บอร์ด ถ้ามีการกดคีย์บอร์ดก็จะส่งข้อมูลออกมาทาง RS-232C ถ้าไม่การกดก็จะวนดู ในขณะที่วนดูว่ามีข้อมูลจากพอร์ตสื่อสารหรือไม่ ถ้ามีก็เอามาตรวจสอบดู หากเป็นตัวอักษรที่ไม่ต้องการก็จะเปลี่ยนเป็นช่องว่าง แต่ถ้าเป็นตัวอักษรที่ต้องการก็ส่งออกทางจอภาพ

ในการรับส่งข้อมูลจะมีการตรวจสอบ XON/XOFF โดยให้จำนวนตัวอักษรส่งสูงสุด

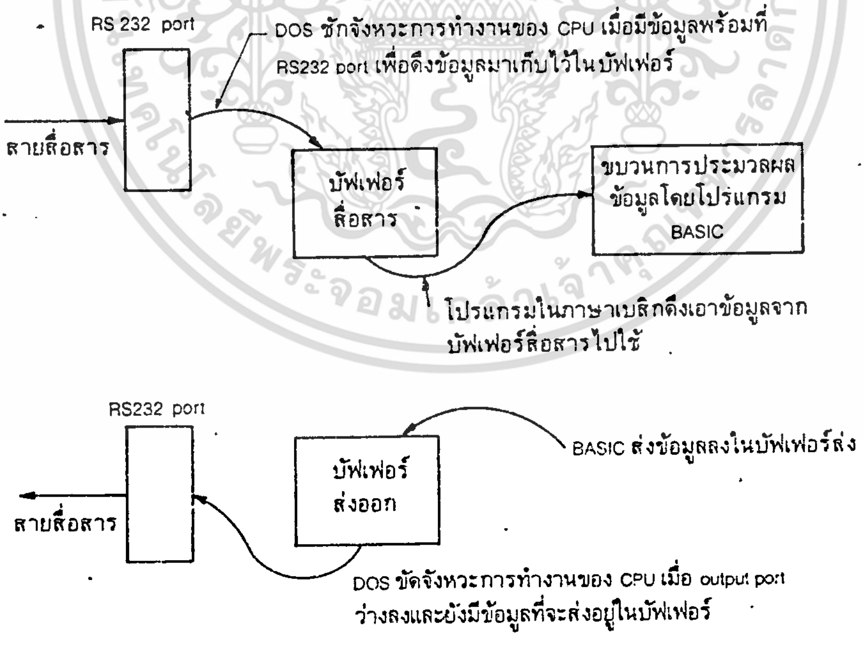
เอกสารนี้เป็นเอกสารที่ใช้เพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ไม่เกิน 225 ตัว เพราะบัฟเฟอร์การสื่อสารที่ตั้งโดย เอ็มเอสดอส กำหนดค่าไว้ 256 ตัว เพื่อให้ไม่เกิดการโอเวอร์โฟลว์ของข้อมูลในการรับ

<u>โปรแกรมการรับส่งไฟล์ข้อมูล</u>

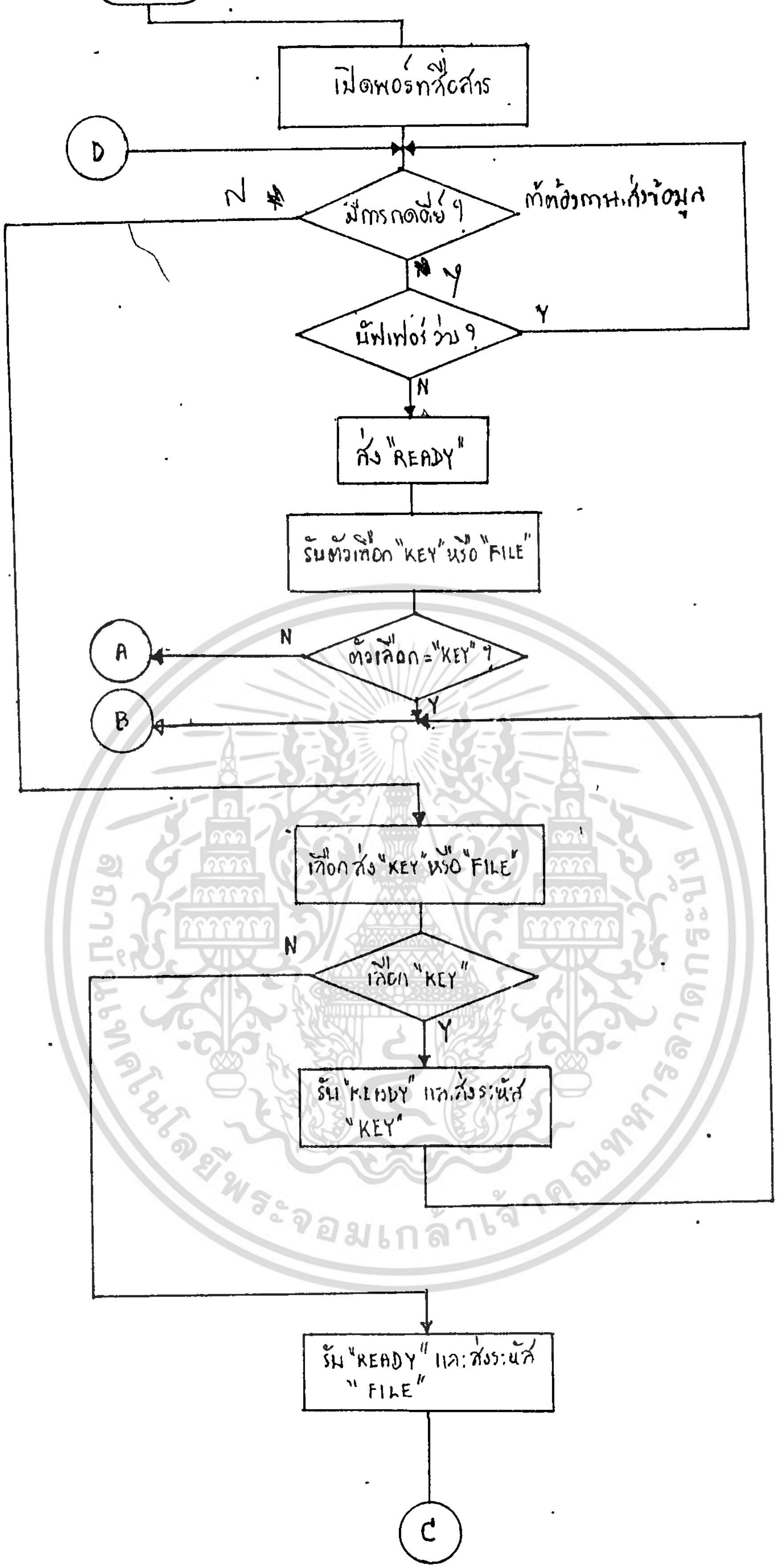
โปรแกรมการรับส่งไฟล์ข้อมูลนี้ เป็นการนำแฟ้มข้อมูลจาก ดิสต์ของเครื่องหนึ่งส่งถ่ายไปเก็บเป็นไฟล์ไว้ในดิสต์ของอีกเครื่องหนึ่งซึ่งอยู่ปลายทาง

วิธีการส่งข้อมูล เราจะต้องให้คอมพิวเตอร์เครื่องหนึ่งเป็นตัวส่ง และอีกเครื่องหนึ่งเป็นตัวรับ โดยสามารถจะเลือกส่งหรือรับจากโปรแกรม ซึ่งจะถามความต้องการของผู้ใช้ แล้วคอมพิวเตอร์ที่เป็นตัวส่งก็จะรันโปรแกรมส่งข้อมูล และคอมพิวเตอร์ตัวรับก็จะรันโปรแกรมรับข้อมูล ซึ่งในการตั้งพารามิเตอร์การสื่อสารข้อมูลของการส่งไฟล์ข้อมูลนี้ ต้องเหมือนกันทั้งด้านส่งและด้านรับ

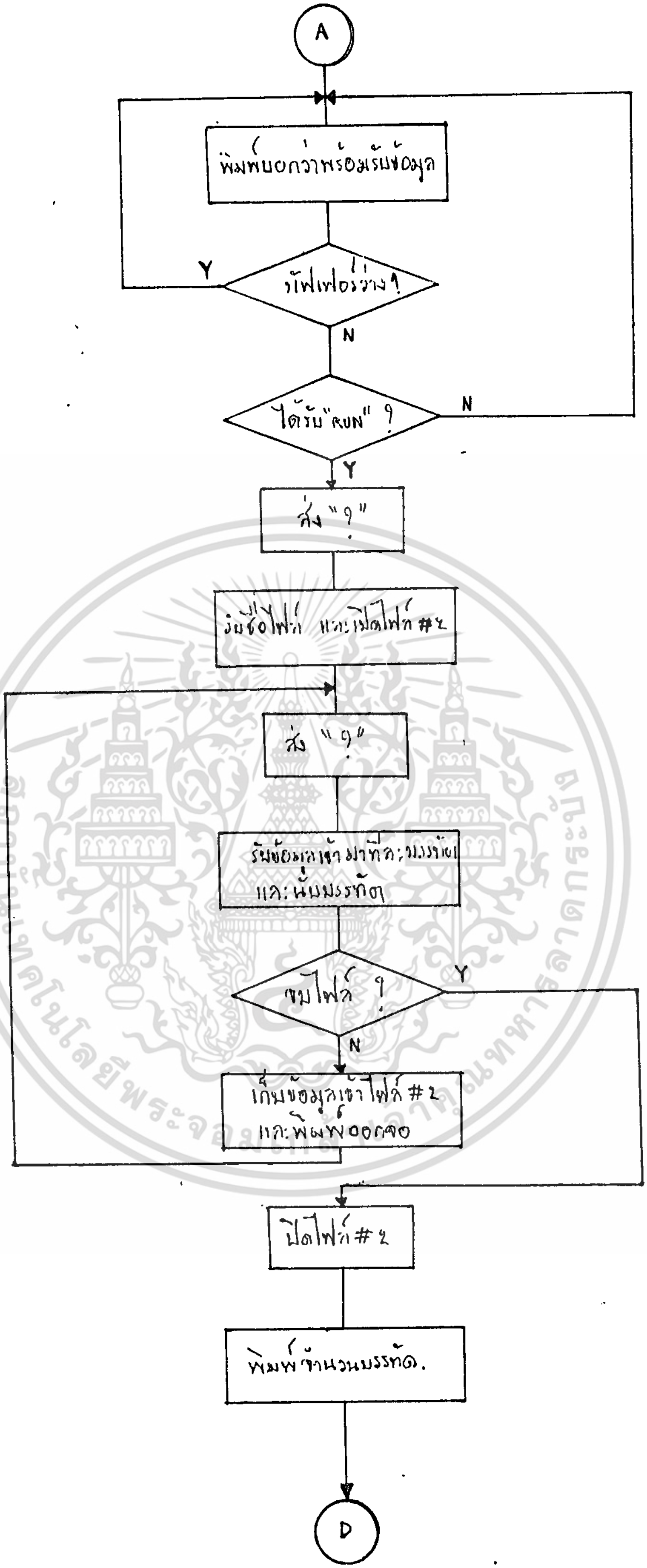


รูปที่ 3.11.1 แสดงการติดต่อระหว่างพอร์ต RS-232C กับภาษาเบสิค

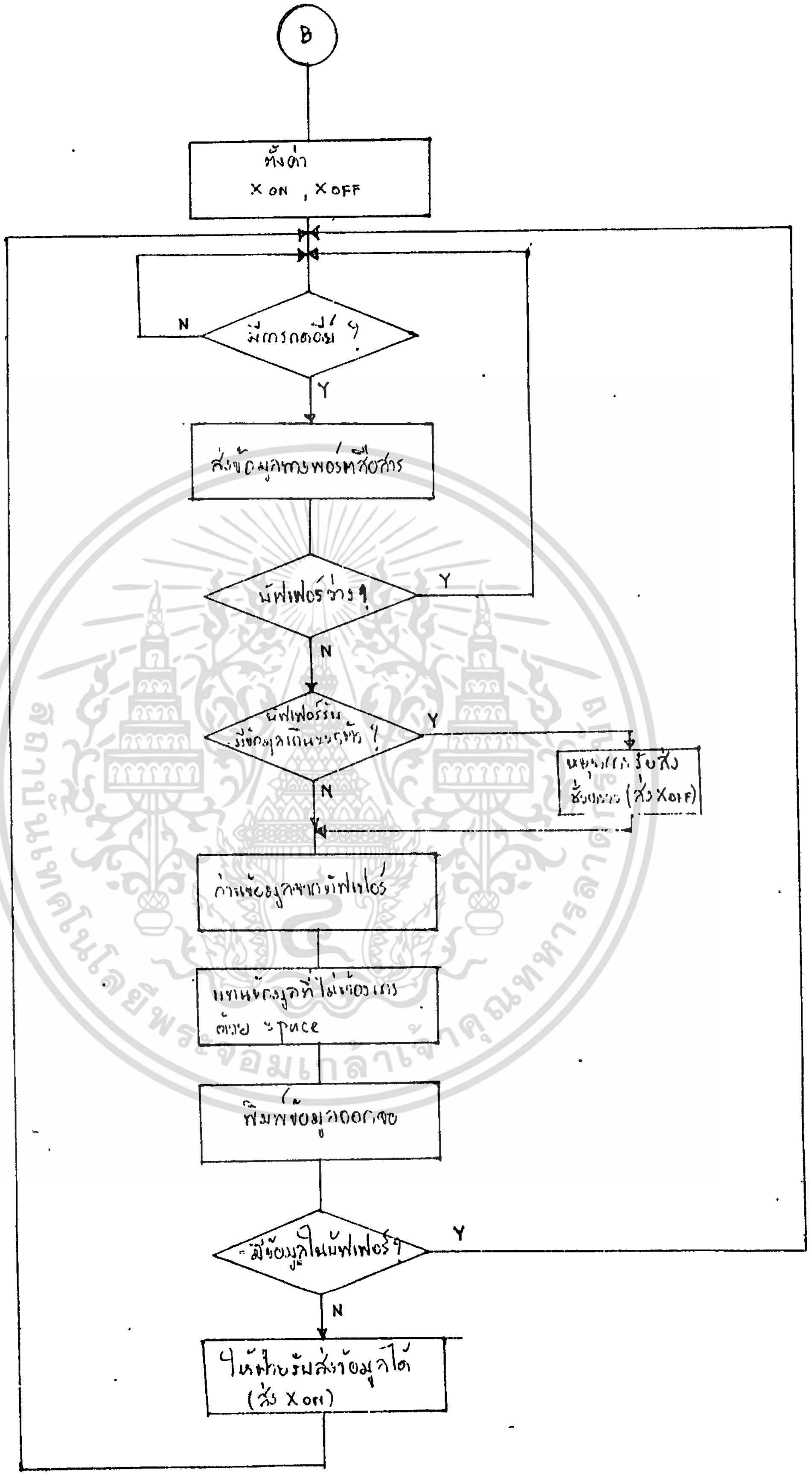
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



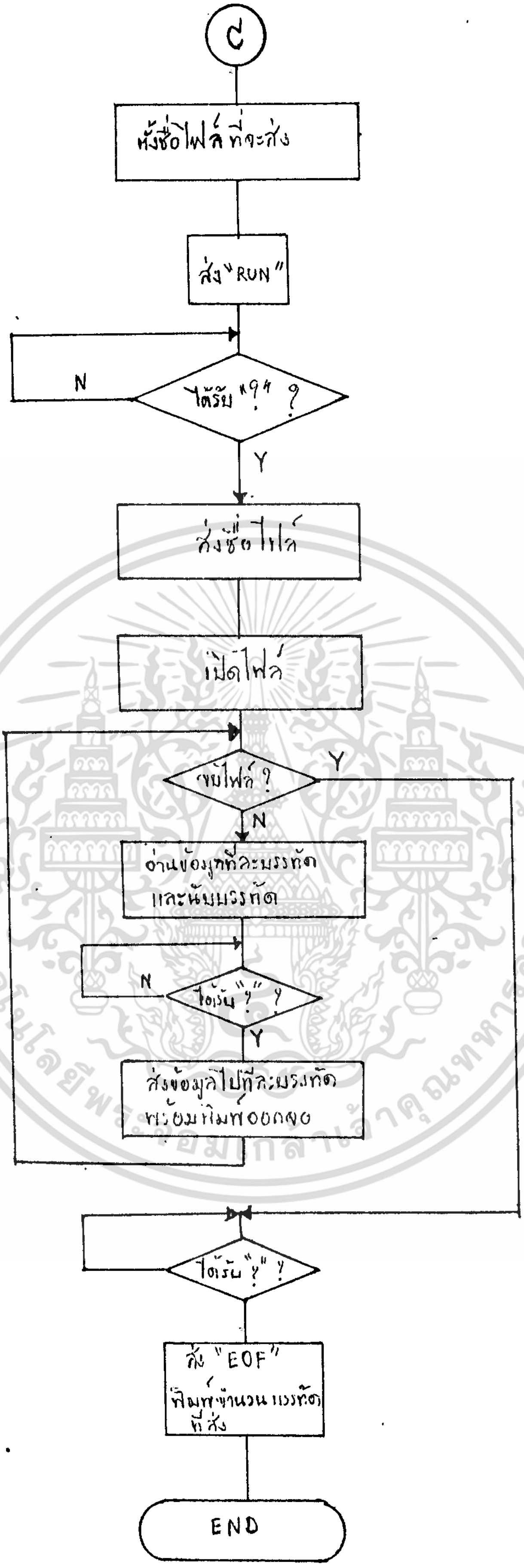
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



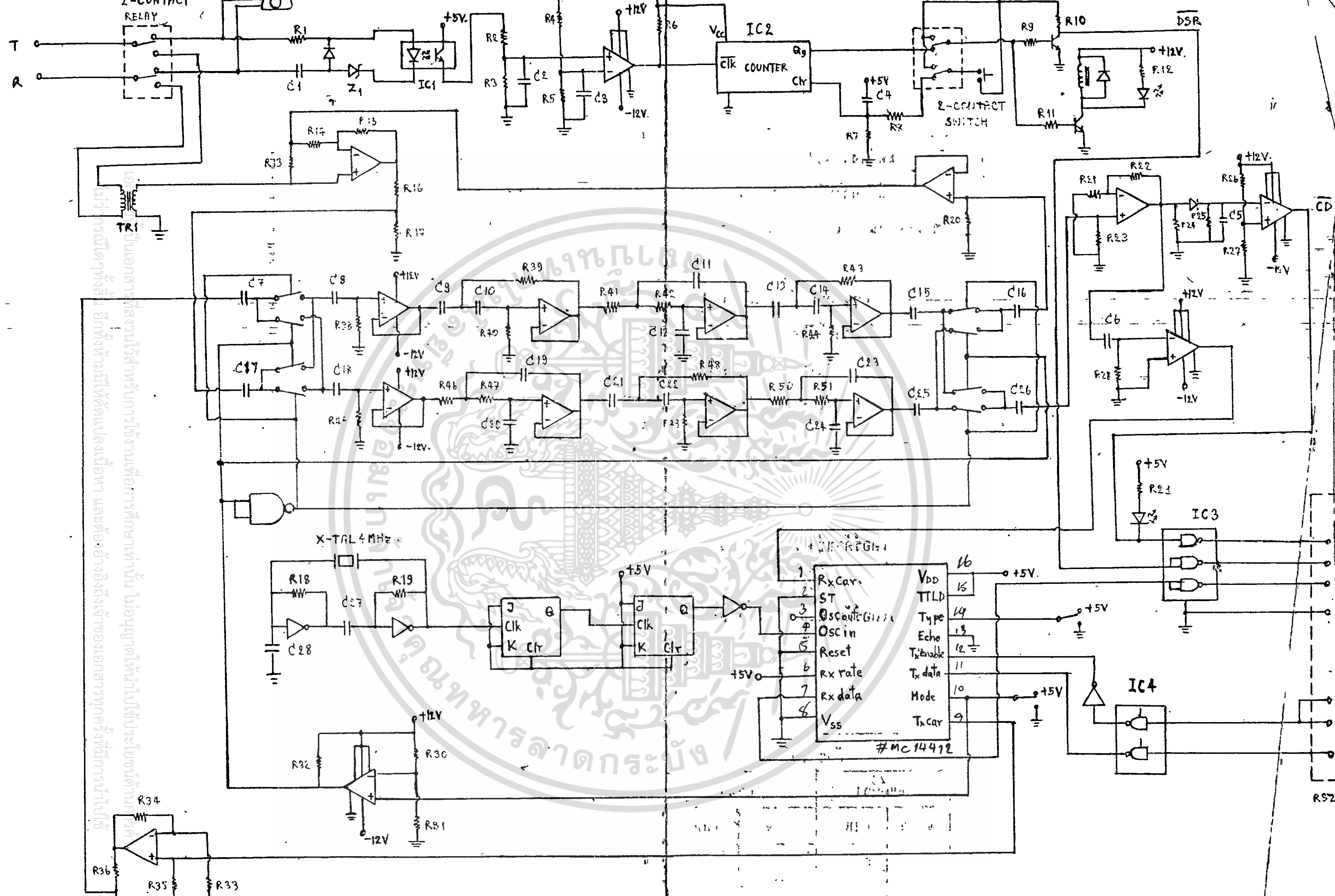
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
10  OPEN "com1:300,e,7" AS#1
20  CLS
30  PRINT "Press any keys ,if you want to send data."
40  Z$ = INKEY$
50  IF Z$ <> "" GOTO 70
60  IF EOF(1) THEN 40 ELSE 100
70  PRINT "are you sure? (y/n)"
80  INPUT X$
90  IF X$ = "y" OR X$ = "Y" GOTO 360 ELSE 20
100 PRINT #1,"ready"
110 INPUT #1,V$
120 IF LEFT$(V$,3) <> "key" THEN 700 ELSE 160
130 ON ERROR GOTO 930
140 PRINT #1,"key"
150 IF EOF(1) THEN 150
160 CLS
170 WAITING = 0
180 CLS
190 XON$ = CHR$(17)
200 XOFF$ =CHR$(19)
210 B$ = INKEY$
220 IF B$ <> "" THEN PRINT #1,B$;
230 IF EOF(1)THEN 210 : REM EOF(1)means receive buffer is empty
240 REM check to see if recieve is more than 225 characters
250 IF LOC(1)>225 THEN WAITING = 1:PRINT #1,XOFF$;
260 A$ = INPUT$(LOC(1),#1)
270 REM Check incoming data for unwanted characters
280 FOR I = 1 TO LEN(A$)
290 D = ASC(MID$(A$,I,1))
300 IF (D= 17 OR D = 19 OR D = 127) THEN MID$(A$,I,1) = ""
310 NEXT I
320 PRINT A$;
330 IF LOC(1)>0 THEN 210
340 IF WAITING THEN WAITING = 0:PRINT #1,XON$;
350 GOTO 210
360 CLS
370 PRINT "1 key":PRINT"2 file"
380 INPUT "choose = ",ANS$
390 IF ANS$ = "1" THEN GOTO 130 ELSE 400
400 PRINT #1,"start"
410 IF EOF(1) THEN 410
420 PRINT #1 ,"file"
430 IF EOF(1) THEN 430
440 FILES
450 INPUT "TYPE FILENAME TO BE TRANSFERED";NM1$
460 INPUT "TYPE NAME AT  DESTINATION COMPUTER(IF DIFFERENTS)";NM2$
470 IF NM2$ = "" THEN NM2$ = NM1$
480 REM Transmit RUN command to receiving program
490 PRINT#1,"RUN"
500 A$ = INPUT$(1,#1)
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
510 IF A$<> "?" THEN 500
520 REM Transmit the name of the file to the receiving program
530 PRINT#1,CHR$(34);NM2$;CHR$(34)
540 REM Begin file-transmission phase
550 OPEN NM1$ FOR INPUT AS #2
560 IF EOF(2) THEN 640
570 LINE INPUT #2,A$
580 SUM = SUM + 1
590 V$ = INPUT$(1,#1)
600 IF V$<> "?" THEN 590
610 PRINT #1,A$
620 PRINT A$
630 GOTO 560
640 IF A$ = "EOF" THEN SUM = SUM-1: GOTO 680
650 V$ = INPUT$(1,#1)
660 IF V$<> "?" THEN 650
670 PRINT #1,"EOF"
680 PRINT:PRINT"TRANSFER COMPLETE.....NUMBER OF RECORDS SENT WAS'
690 END
700 PRINT #1,"begin"
710 PRINT "STANDING BY TO RECEIVE"
720 IF EOF(1) THEN 710
730 INPUT #1,A$
740 IF LEFT$(A$,3)<>"RUN" THEN 710
750 REM Get the name of the file from sending program
760 PRINT #1,"?"
770 INPUT #1,NM$
780 PRINT"THE FILE WILL BE SAVED UNDER THE NAME";NM$
790 PRINT :PRINT"WAITING FOR DATA TRANSFER TO BEGIN":PRINT
800 OPEN NM$ FOR OUTPUT AS #2
810 PRINT#1,"?"
820 LINE INPUT #1,A$
830 IF A$ = "EOF" THEN 880
840 SUM = SUM+1
850 PRINT A$
860 PRINT #2,A$
870 GOTO 810
880 CLOSE #2
890 PRINT:PRINT"THE NUMBER OF RECORDS RECEIVED WAS";SUM
900 FOR J = 1 TO 1500
910 NEXT J
920 GOTO 20
930 RESUME
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



IC1 - 4N35 IC2 - 4040 IC3 - MC1488 IC4 - MC1489 TR1 - 1:1 600Ω TRANSFORMER

C1 - .47μ 100V. C2, C3, C6, C27 - .1μ C5 - .47μ C7, C8, C15, C16, C17, C18, C25, C26 - 10μ C9, C10, C13, C14 - 4.7n

C11, C19, C23 - 47n C12, C20, C24 - 470p C21, C22 - 15n. C28 - 10p

R1, R2, R3, R7, R14, R15, R16, R22, R28, R30, R34, R36 - 10K R4, R12 - 3K9 R5 - 1K8 R6, R10, R17 - 2K R8 - 680Ω R9, R11 - 2K2, R13 - 600

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 4

การทดลองและผลการทดลอง

4.1 การทดลองผลตอบสนองทางความถี่ของแบนด์พาสฟิวเตอร์

วงจรกรองความถี่ที่ใช้งาน เป็นการต่อคาสเคด ระหว่างวงจรโลว์พาสฟิวเตอร์กับวงจรไฮพาสฟิวเตอร์ ซึ่งมีการซ้อนกันของช่วงตอบสนองทางความถี่ ทำให้ได้วงจรแบนด์พาสฟิวเตอร์ขึ้นมา ดังนั้นเป็นการสะดวกในการกำหนดแบนด์วิดท์ทั้งสองด้าน เพราะวงจรโลว์พาสและไฮพาสฟิวเตอร์สามารถคำนวณและใช้งานได้ง่าย

วงจรที่ใช้งานมีการเพิ่มค่าควอลิตี้แฟคเตอร์ (Quality Factor or Q Factor) โดยใช้วงจรฟิวเตอร์ที่มีคุณสมบัติเช่นเดียวกันซ้อนเข้าไปอีกที ในด้านที่ต้องการให้มีการลดทอนสูง

การทดลอง ทำการวัดผลตอบสนองทางความถี่ โดยการป้อนสัญญาณซายน์สังเคราะห์จากตัวมอดดูเลเตอร์ และเปรียบเทียบระหว่างการมีวงจรฟิวเตอร์อีกวงจรซ้อนเข้าไป กับการมีวงจรเพียง 2 วงจรซ้อนกันอยู่

กำหนด

- วงจรฟิวเตอร์แบนด์ต่ำ1 และแบนด์สูง1 คือวงจรแบนด์พาสฟิวเตอร์ด้านช่วงความถี่ต่ำและความถี่สูงตามลำดับ ที่คาสเคดกัน 2 ตัว
- วงจรฟิวเตอร์แบนด์ต่ำ2 และแบนด์สูง2 คือวงจรแบนด์พาสฟิวเตอร์ด้านช่วงความถี่ต่ำและความถี่สูงตามลำดับ ที่คาสเคดกัน 3 ตัว เพื่อเพิ่มค่า Q Factor

		สัญญาณเอาท์พุท		อัตราขยาย			
				แบนด์ต่ำ1		แบนด์ต่ำ2	
ความถี่อินพุท (Hz)	ขนาดสัญญาณอินพุท (V_{p-p})	แบนด์ต่ำ1 (V_{p-p})	แบนด์ต่ำ2 (V_{p-p})	(V/V)	(dB)	(V/V)	(dB)
980	0.13	0.28	0.80	2.15	6.67	6.15	15.78
1180	0.13	0.50	2.60	3.85	11.70	4.62	13.28
1650	0.13	0.15	0.15	1.15	1.24	1.15	1.24
1850	0.13	0.10	0.06	0.77	-2.27	0.46	-6.75

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 4.1 (ต่อ)

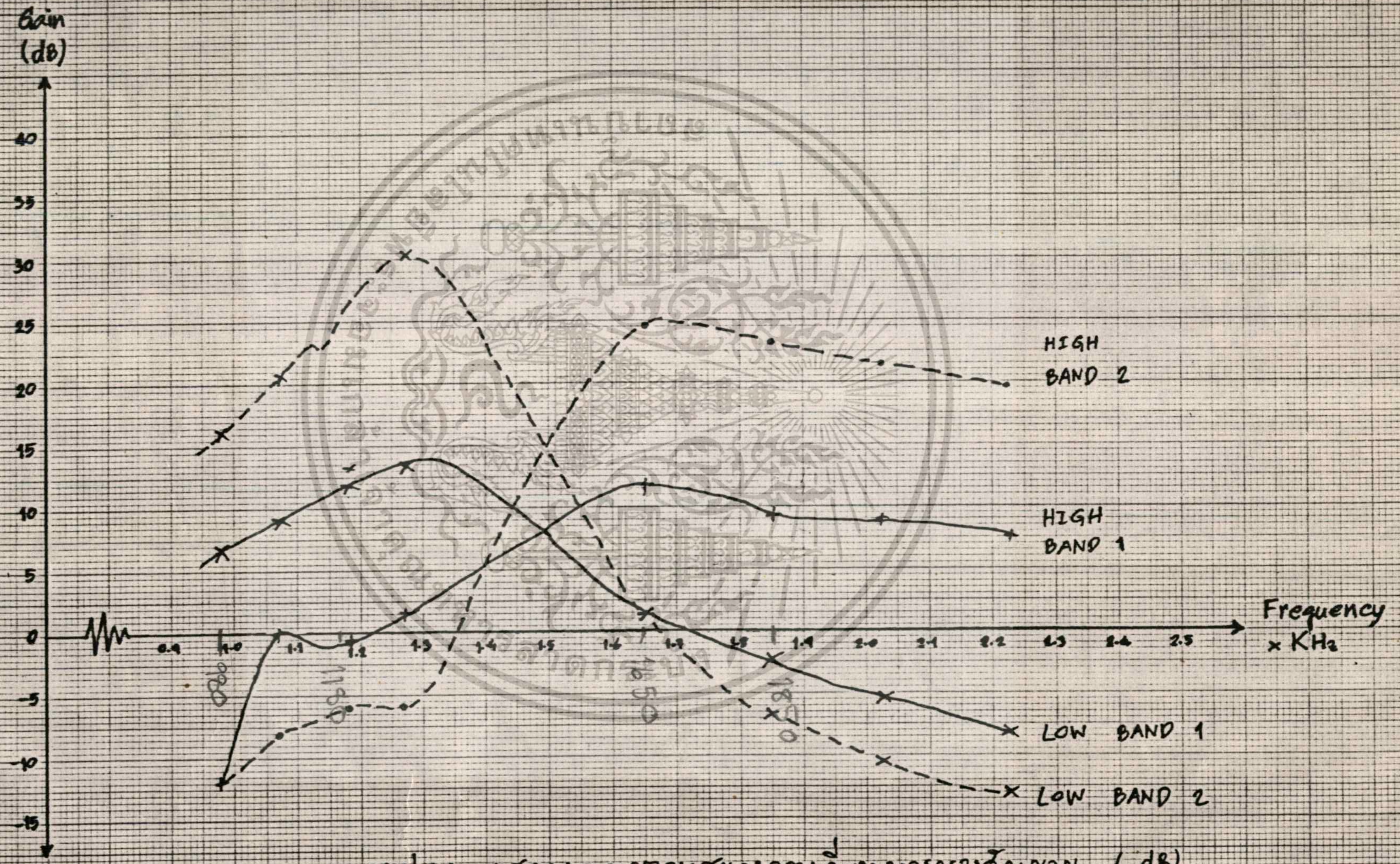
1270	0.13	0.60	4.20	4.62	13.28	32.30	30.18
1070	0.13	0.36	1.40	2.77	8.85	10.77	20.64
2225	0.13	0.05	0.03	0.39	-8.29	0.23	-12.70
2025	0.13	0.07	0.04	0.54	-5.35	0.30	-10.46

ตารางที่ 4.1 ผลการทดลองของวงจรแบนด์พาสช่วงต่ำ

		สัญญาณเอาท์พุท		อัตราขยาย			
ความถี่อินพุท (Hz)	ขนาดสัญญาณอินพุท (V_{p-p})	แบนด์สูง1 (V_{p-p})	แบนด์สูง2 (V_{p-p})	แบนด์สูง1 (V/V)	แบนด์สูง1 (dB)	แบนด์สูง2 (V/V)	แบนด์สูง2 (dB)
980	0.20	0.05	0.05	0.25	-12.04	0.25	-12.04
1180	0.20	0.10	0.10	0.50	-6.02	0.50	-6.02
1650	0.34	1.30	5.80	3.82	11.64	17.06	24.64
1850	0.34	1.00	5.00	2.94	9.36	14.70	23.35
1270	0.20	0.30	0.10	1.50	3.52	0.50	-6.02
1070	0.20	0.20	0.08	1.00	0.00	0.40	-7.96
2225	0.42	1.00	4.00	2.28	7.54	9.52	19.56
2025	0.42	1.20	4.50	2.86	9.12	11.90	21.51

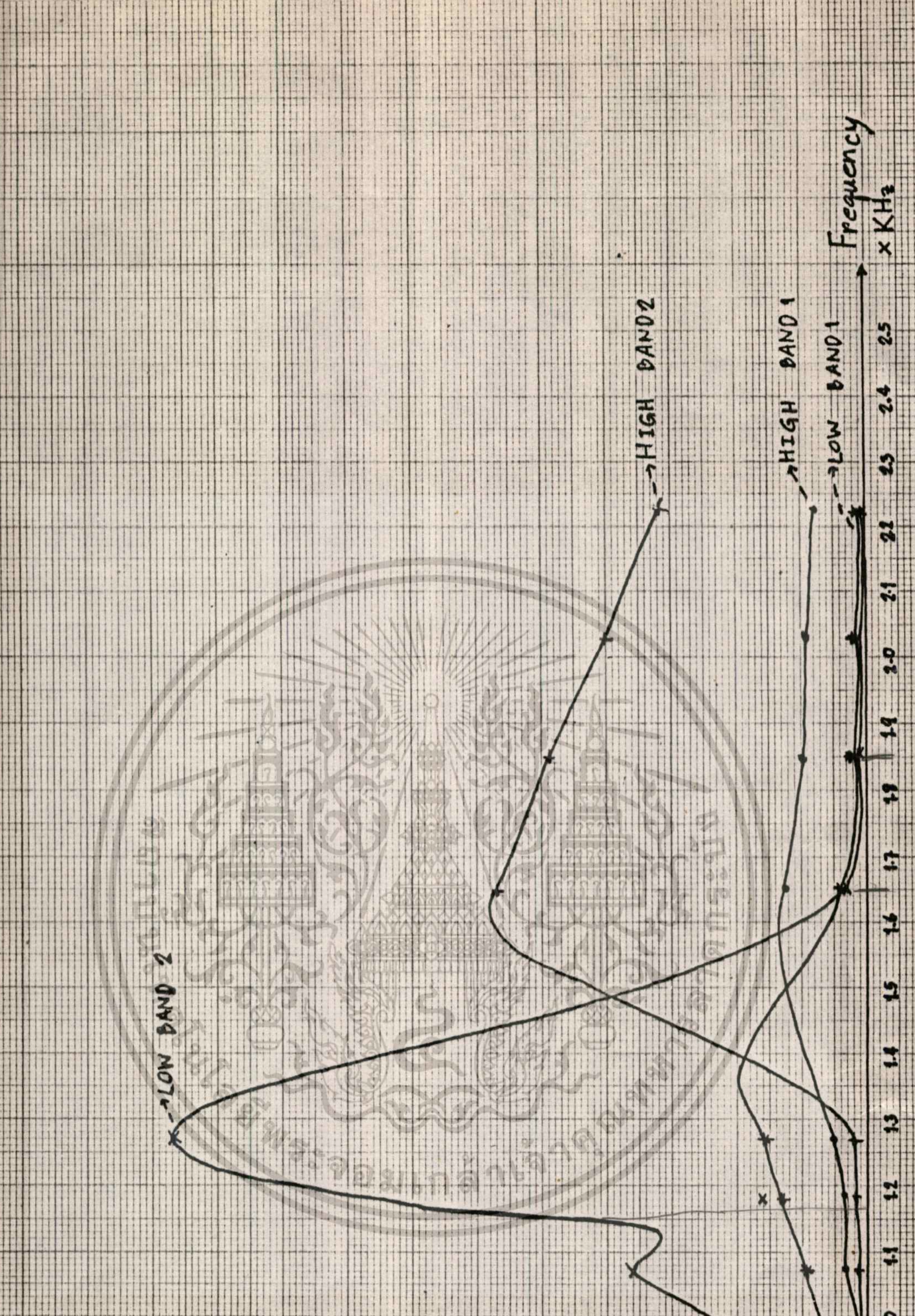
ตารางที่ 4.2 ผลการทดลองวงจรแบนด์พาสช่วงความถี่สูง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



กราฟ 4.1 แสดงผลการตอบสนองความถี่ของวงจรกรองสัญญาณ (dB)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



กราฟ(4.2) แสดงผลการตอบสนองความถี่ ของวงจรกรองสัญญาณ (V/V)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

<u>การทดลองการส่งไฟล์</u>

ABCDEFGHIJKLMNOPQRSTUVWXYZ

ABCDEFGHIJKLMNOPQRSTUVWXYZ

ABCDEFGHIJKLMNOPQRSTUVWXYZ

1234567890 1234567890

1234567890 1234567890

0987654321 0987654321

0987654321 0987654321

"HELLO" "HELLO" "HELLO"

"HELLO" "HELLO" "HELLO"

" ไฟล์ด้านส่ง "

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

aBCDEFGH*JKLMNOP@RSTUVWXYZ

ABCDEfGHIJKLmNOPQRSTUVWXYz

ABCDEFGHIJKLXNOPQRSTUVWXYZ

12Z4567890 1234567890

1234567890 1234567890

098765n321 0987654321

0987654321 0987654321

"HELLO" "HELLO" "HELLO"

"hELLO" "HELLO" "HELLO"

" ไฟล์ที่รับได้ "

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 5
สรุปและวิจารณ์

โครงงานเรื่องโมเด็มนี้ เป็นการศึกษาระบบการส่งข้อมูลระหว่างคอมพิวเตอร์ผ่านสายโทรศัพท์ ด้วยวิธี FSK ใช้ไอซีโมเด็มเบอร์ MC 14412 จากการทดลองสามารถส่งได้ทั้งระบบ BELL 103 และ CCITT V.21ใช้โปรแกรมภาษาเบสิกในการควบคุมการสื่อสาร สามารถรับส่งข้อมูลได้ทั้งแบบตัวอักษรจากคีย์บอร์ด และไฟล์ข้อมูลจากดิสค์ โดยใช้งานเป็นแบบโปรโตคอล

ในการทดลองรับส่งสัญญาณทางสายโทรศัพท์ สามารถรับส่งข้อมูลได้ แต่มีความผิดพลาดของข้อมูล เนื่องมาจากการแยกความถี่ของวงจรกรองสัญญาณไม่ดีพอ แบนด์วิทของช่วงความถี่ต่ำและช่วงความถี่สูงใกล้กันมากทำให้มีการกวนกัน ข้อมูลจึงมีความผิดพลาดบ้างเล็กน้อย การปรับปรุงแก้ไขควรพัฒนาทางด้านฮาร์ดแวร์ให้มีประสิทธิภาพมากขึ้น ส่วนทางด้านซอฟท์แวร์นั้นสามารถใช้ได้ดีอยู่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กิตติกรรมประกาศ

โครงงานและปริญญานิพนธ์นี้สำเร็จลุล่วงไปด้วยดี ก็เนื่องด้วยความอนุเคราะห์ อุปกรณ์เครื่องมือ สถานที่ที่ทำการวิจัย และการให้คำปรึกษาอย่างดียิ่ง จากรองศาสตราจารย์ประกิจ ตังติสานนท์ อาจารย์ที่ปรึกษา ตลอดจนคณะอาจารย์ทุกๆท่าน ที่ให้ความรู้พื้นฐานในด้านต่างๆแก่ผู้จัดทำ รวมทั้งเพื่อนๆทุกคนที่ได้ให้คำแนะนำอันมีค่า และให้กำลังใจด้วยดีตลอดมา จึงใคร่ขอขอบพระคุณอย่างสูงมา ณ ที่นี้

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เอกสารอ้างอิง

ก. เอกสารอ้างอิงที่เป็นภาษาไทย

1. น.ต.ดร.ไพศาล สงวนหมู่ และ รศ.ยืน ภู่วรวรรณ ,"การสื่อสารข้อมูลและไมโครคอมพิวเตอร์เนตเวอร์ค" , ซีเอ็ดยูเคชั่น , 2531

2. รศ.ยืน ภู่วรวรรณ , "พื้นฐานการสื่อสารข้อมูลบนไมโครคอมพิวเตอร์ฯ , วารสารไมโครคอมพิวเตอร์ ฉบับที่ 35 , หน้า 130-136 , มกราคม 2531

3. น.ต.ดร.ไพศาล สงวนหมู่,"มาตราฐานในการสื่อสารข้อมูลสำหรับไมโครคอมพิวเตอร์และการพัฒนายลิตี้" ,วารสารไมโครคอมพิวเตอร์ ฉบับที่ 35 ,หน้า 147-157 ,มกราคม 2531

4. พีรศักดิ์ ศรีกังวาล ,"เรียนภาษาเบสิก/เบสิกขั้นสูงสำหรับเครื่อง IBM-PC" ,ซีเอ็ดยูเคชั่น , 2531

ข. เอกสารอ้างอิงที่เป็นภาษาอังกฤษ

1.PAUL BATES , "Practical Digital Data Communications with LSI Applications" ,PRENTICE-HALL ,Inc. ,1987

2.JOHN UFFENBACK ,"Microcomputers and Microprocessors" ,PRENTICE HALL ,Inc. ,1985

3.GOBIND DARYANANI , "Priciple of Active Network Synthesis and Design" ,BELL Laboratories ,Inc. ,1987

4.MICROSOFT QUICK BASIC 4.0 ,"Basic Language Reference",MICROSOFT CORPERATION ,1987

5.BARRY J. ARNOW ,"Writing Communications in BASIC" ,BYTE ,DEC. 1984 ,Vol.9 No.13 pp.187

6."MOTOROLA TELECOMMUNICATIONS DEVICE DATA" ,MOTOROLA INC. ,1985

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้