



ปีการศึกษา 2532

โมเด็มไร้สาย

(WIRELESS MODEM)

โดย

นาย ประพจน์ เจตนมิ่งคลรัตน์

นาย พงษ์พันธ์ เมฆรุ่งเรืองไกร

อาจารย์ที่ปรึกษา

ผ.ศ. ณรงค์ เหมกรณ์

ปริญญาโท ประจำปีการศึกษา 2532

ภาควิชา วิศวกรรมโทรคมนาคม

คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
เรื่อง โฉมเติมใบสไลด์

ผู้จัดทำ

1. นาย ประพจน์ เจตน์มงคลรัตน์ 29.1107
2. นาย พงษ์พันธ์ เมฆรุ่งเรืองไกร 29.1141



..... อาจารย์ที่ปรึกษา

(ผ.ศ. ประจักษ์ เหมกর্ণ)



40 CH WIRELESS MODEM

Prapoj Jatmongkolrat

Pongpan Makrungruengkrai

Asst. Prof. Narong Haemakorn, Advisor

1989

Abstract

"40 CH WIRELESS MODEM" is one of Data Communication Projects. The objective of this project is to change transmission media of computer network from cable to radio link. This technic will help to solve the problem of cable lying between personal computer.

Data will be converted from analog signals to digital signals. Then, they will be modulated with radio wave carrier which is generated from frequency synthesizer circuit. The range of carrier frequency is in Citizen Band Class D (26.965 - 27.405 MHz.) that is divided into 40 channels.

Because of having 40 channels, this system can transmit 20 informations between personal computer at the same time. The typical of signal transmission is simplex.

โมเด็มไร้สาย

ประพนธ์ เจตน์มิ่งครรัตน์
พงษ์พันธ์ เมฆรุ่งเรืองไกร
ผ.ศ.ณรงค์ เหมกรณ์ อาจารย์ที่ปรึกษา
ปีการศึกษา 2532

บทคัดย่อ

"โมเด็มไร้สาย" เป็นหนึ่งในโครงการที่เกี่ยวกับการสื่อสารข้อมูลโดยมีจุดประสงค์ที่จะเปลี่ยนตัวกลางที่ใช้ในการเชื่อมต่อคอมพิวเตอร์เน็ตเวิร์ค จากเดิมที่ใช้สายเคเบิลมาเป็นการเชื่อมต่อด้วยคลื่นวิทยุ ซึ่งจะช่วยลดข้อยุ่งยากในการวางสายระหว่างคอมพิวเตอร์ลงได้

การมีอัตราเลขที่ข้อมูลเข้ากับคลื่นวิทยุสามารถทำได้โดยการแปลงสัญญาณดิจิทัลให้เป็นอนาล็อกโดยโมเด็มเสียก่อน จากนั้นจึงทำการมอดดูเลขสัญญาณอนาล็อกเข้ากับคลื่นวิทยุแบบ FM ความถี่คลื่นพาหะที่ใช้จะถูกลังเคราะห์ขึ้นจากวงจรรังเคราะห์ความถี่ในช่วง Citizen Band Class D (26.965 - 27.405 MHz) จำนวน 40 ช่องสัญญาณ จากการรังเคราะห์ความถี่จำนวน 40 ช่องสัญญาณจะทำให้สามารถส่งข่าวสารผ่านคลื่นวิทยุระหว่างคอมพิวเตอร์ได้เป็นจำนวน 20 ข่าวสารพร้อมกันได้ โดยมีลักษณะการสื่อสารเป็นแบบซิมเพล็กซ์ (Simplex)

สารบัญ

	หน้า
บทที่ 1 บทนำ	1
บทที่ 2 ทฤษฎีและหลักการเบื้องต้น	
2.1 การสื่อสารข้อมูล	2
2.2 การส่งข้อมูลแบบขนานและอนุกรม	3
2.3 ทิศทางการสื่อสารข้อมูล	5
2.4 โมเด็ม	6
2.5 เทคนิคการมอดูเลตสัญญาณดิจิทัล	6
2.6 อินเตอร์เฟส EIA RS-232C	9
2.7 หลักการทำงานเบื้องต้นของโครงการ	13
บทที่ 3 การออกแบบและการสร้าง	
3.1 การออกแบบโมเด็มความเร็วต่ำ	17
3.2 การออกแบบเครื่องส่งวิทยุ 27 Mhz	30
3.3 การออกแบบโปรแกรมการสื่อสารข้อมูล	38
บทที่ 4 การทดลองและผลการทดลอง	
4.1 การทดลองวัดค่าต่างๆ ของโมเด็ม 600 บอด	44
4.2 การวัดคุณสมบัติต่างๆ ของเครื่องส่ง	53
บทที่ 5 บทวิจารณ์และสรุป	64
ภาคผนวก	
กิตติกรรมประกาศ	
หนังสืออ้างอิง	

ปัจจุบันเทคโนโลยีด้านการสื่อสารข้อมูลได้เข้ามามีบทบาทอย่างมากในการติดต่อสื่อสารระหว่างคอมพิวเตอร์ เมื่อก่อนเราสามารถใส่คอมพิวเตอร์ในแบบ Stand Alone ได้ในระนาบข้อมูลที่ใช้ในการประมวลผลยังเล็กอยู่ แต่ในปัจจุบันการประมวลผลข้อมูลแต่ละครั้งจะต้องมีการเรียกใช้ฐานข้อมูลจากคอมพิวเตอร์เครื่องอื่นร่วมด้วยอยู่เสมอ โดยปกติการสื่อสารข้อมูลระหว่างคอมพิวเตอร์จะทำได้โดยการเชื่อมโยงสายเข้าด้วยกัน ทำให้เกิดลักษณะของโครงข่ายคอมพิวเตอร์ (Computer Network) ขึ้นและสามารถแบ่งลักษณะของโครงข่ายคอมพิวเตอร์ได้เป็น 2 ลักษณะใหญ่ๆคือ

1) Local Area Network ระบบนี้เป็นการสื่อสารข้อมูลระหว่างคอมพิวเตอร์ในระยะทางไม่เกิน 2 กิโลเมตร โดยใช้สายซึ่งตอบสนองความถี่ได้สูง อย่างเช่นสาย Coaxial cable ต่อระหว่างคอมพิวเตอร์เข้าด้วยกัน และจะได้โครงข่ายคอมพิวเตอร์ในรูปแบบการวางตัว (Topology) ที่แตกต่างกันเป็น Star, Bus หรือ Tree Topology เป็นต้น

2) Wide Area Network ระบบนี้จะใช้ในการสื่อสารข้อมูลระหว่างคอมพิวเตอร์ที่อยู่ห่างกันมากกว่า 2 กิโลเมตรเป็นต้นไป โดยจะอาศัยโครงข่ายโทรศัพท์ซึ่งวางอยู่โดยทั่วไปอยู่แล้วในการส่งข้อมูลไปยังปลายทาง แต่ช่องสัญญาณโทรศัพท์เป็นช่องสัญญาณที่ใช้ในย่านความถี่เสียงไม่เกิน 4Khz เท่านั้นซึ่งไม่เพียงพอในการตอบสนองสัญญาณดิจิทัล จึงต้องใช้อุปกรณ์ในการแปลงสัญญาณดิจิทัลให้เป็นอนาล็อกในย่านความถี่เสียงเสียก่อน จึงจะสามารถส่งข้อมูลเข้าไปในโครงข่ายโทรศัพท์ได้ ซึ่งอุปกรณ์ที่ทำหน้าที่ดังกล่าวก็คือโมเด็ม (MODEM)

อย่างไรก็ตามการใช้สายในการสื่อสารข้อมูลยังมีข้อจำกัดอยู่บ้าง นั่นคือเมื่อคอมพิวเตอร์ที่จะทำการส่งข้อมูลเพิ่มมากขึ้น จะทำให้การเดินทางยุ่งยาก และการเคลื่อนย้ายจะทำได้ลำบาก นอกจากนี้ในบางพื้นที่ที่การวางข่ายโทรศัพท์เข้าไม่ถึงก็จะไม่สามารถสื่อสารข้อมูลได้

โครงงานนี้เป็นโครงงานเกี่ยวกับการสื่อสารข้อมูลผ่านระบบวิทยุ เพื่อแก้ปัญหาดังกล่าวโดยจะเน้นไปที่การใช้งานในแบบ Local Area และคอมพิวเตอร์ที่อยู่ในพื้นที่เดียวกันที่จะทำการส่งข้อมูลสามารถส่งข้อมูลออกไปพร้อมๆกันได้

ทฤษฎีและหลักการเบื้องต้น

2.1 การสื่อสารข้อมูล (Data Communication)

การสื่อสารข้อมูลเป็นการส่งข่าวสารดิจิทัล (Digital Information) ซึ่งส่วนมากอยู่ในรูปเลขฐานสองจากแหล่งกำเนิดไปยังจุดหมายปลายทาง ข่าวสารจากแหล่งกำเนิดอาจจะเป็นรหัสของตัวอักษร ตัวเลข หรือ เครื่องหมายที่อยู่ในรูปเลขฐานสอง เช่น รหัสแอสกี (ASCII), รหัส EBCDIC, รหัสควบคุมไมโครโปรเซสเซอร์ หรือ opcode ก็ได้

การสื่อสารข้อมูลอาจจะเป็นการส่ง-รับระหว่าง 2 จุดหมาย เช่น การเชื่อมโยงระหว่างคอมพิวเตอร์ส่วนบุคคล (Personal Computer) 2 เครื่องในระยะทางไกลๆ ด้วยสายธรรมดาก็ทำได้ แต่ในกรณีที่มีการแลกเปลี่ยนข่าวสารข้อมูลมีมากกว่า 2 จุดหมายขึ้นไปจะทำให้เกิดการสื่อสารข้อมูลในลักษณะที่เป็นเน็ตเวิร์คขึ้น ซึ่งถ้าเป็นเน็ตเวิร์คที่อยู่ในอาณาบริเวณเดียวกันก็จะเรียกว่าเป็น โลคัลแอเรียเน็ตเวิร์ค (Local Area Network) หรือเรียกว่า LAN เน็ตเวิร์คแบบโลคัลแอเรียเน็ตเวิร์คสามารถที่จะต่อสายเข้าด้วยกันระหว่างเทอร์มินัล (terminal) ได้ง่าย โดยมีรูปแบบของเน็ตเวิร์คหรือ topology แตกต่างกันไปหลายชนิดตามความเหมาะสมของพื้นที่นั้นๆ เช่น โทโปโลยี, แบบสตาร์, แบบบัส และแบบริง เป็นต้น

ในกรณีที่มีการสื่อสารข้อมูลด้วยระยะทางไกลๆ หรือที่เรียกว่า Wide Area Network การเดินสายเชื่อมโยงระหว่างผู้ใช้ไม่สามารถทำได้ เพราะจะสิ้นเปลืองค่าใช้จ่ายในการวางสาย สามารถแก้ไขได้โดยการใช้องค์ข่ายชนิดอื่นที่มีอยู่แล้ว เช่น องค์กรข่ายโทรศัทพ์ และ องค์กรข่ายวิทยุ มาทำการส่งข้อมูลแทนได้ เน็ตเวิร์คที่เป็นแบบ Wide Area Network ได้แก่ การส่งข่าวสารระหว่าง Mainframe Computer ที่สำนักงานใหญ่ กระจายไปตามเทอร์มินัลที่สาขาต่างๆ โดยผ่านองค์กรข่ายโทรศัทพ์ หรือการใช้ Microwave Link เป็นต้น

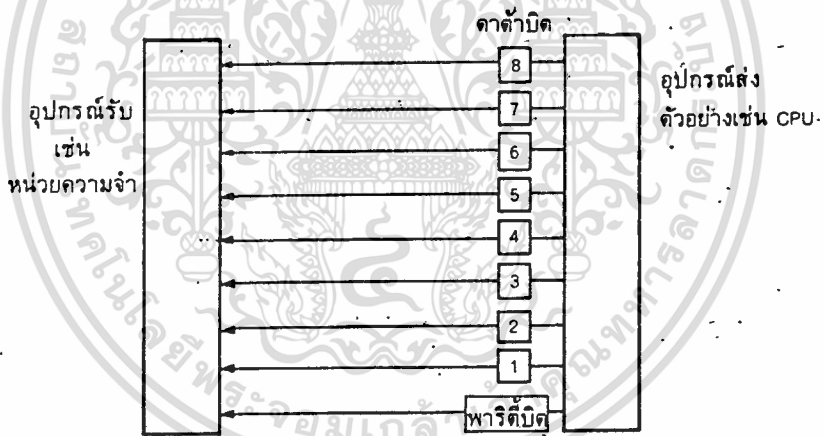
2.2 การส่งข้อมูลแบบขนาน และแบบอนุกรม

ในการสื่อสารข้อมูลนั้น ลักษณะการส่งสัญญาณข้อมูล มีได้ 2 ลักษณะคือ

2.2.1 การส่งข้อมูลแบบขนาน

ลักษณะของการส่งข้อมูลแบบขนาน ทำได้โดยการส่งข้อมูลออกมาทีละ 1

ไบท์ ซึ่งถ้าใน 1 ไบท์มี 8 บิต ทั้ง 8 บิตจะถูกส่งจากอุปกรณ์ส่งไปยังอุปกรณ์รับพร้อมกัน และช่องสัญญาณในการส่งข้อมูลจะต้องมีอย่างน้อย 8 ช่องสัญญาณ โดยมากช่องสัญญาณในการส่งจะใช้สายเคเบิลแบบที่มีตัวนำหลายเส้น อาจจะเป็นสายเคเบิลชนิดแบนหรือกลมก็ได้ โดยที่ระยะทางระหว่าง 2 เครื่องไม่ควรไกลเกินไป เนื่องจากสาเหตุต่างๆหลายสาเหตุ เช่น การที่สัญญาณถูกลดทอนไปกับความต้านทานของสาย การผิดเพี้ยนของสัญญาณเพราะสภาพความเป็นตัวเก็บประจุในสาย การเดินทางมาถึงอุปกรณ์รับไม่พร้อมกันของแต่ละบิตเพราะสภาพความไม่สมดุลย์ของตัวนำแต่ละเส้นในสายเคเบิล และการที่ระดับของกราวด์ทางไฟฟ้าที่อุปกรณ์รับผิดไปจากอุปกรณ์ส่ง สาเหตุเหล่านี้อาจจะทำให้เกิดการผิดพลาดของข้อมูลขึ้นได้นอกจากสายที่ที่เป็นทางเดินของข้อมูลแล้ว อาจจะมีทางเดินของสัญญาณควบคุมต่างๆอีก เช่นสายที่ใช้ควบคุมการโต้ตอบ (Hand Shake) ซึ่งการส่งข้อมูลแบบขนานจะเห็นได้จากการส่งข้อมูลจากคอมพิวเตอร์ไปยังเครื่องพิมพ์เป็นส่วนใหญ่ลักษณะการส่งข้อมูลแบบขนานเป็นดังรูป 2.1

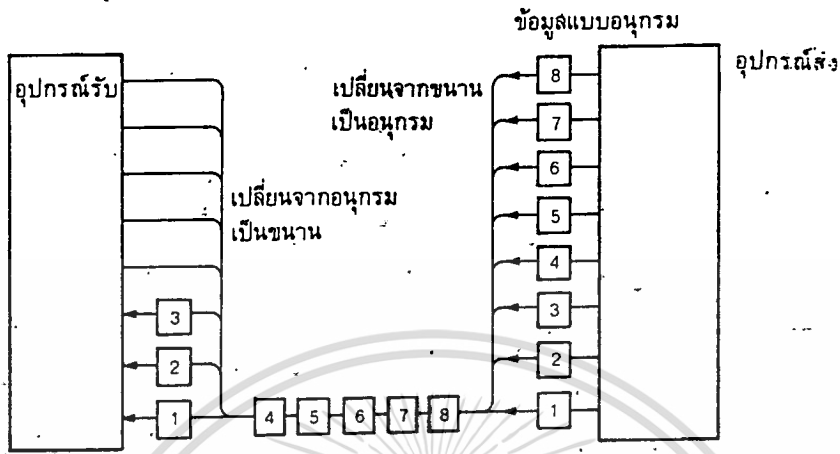


รูปที่ 2.1 การส่งข้อมูลแบบขนาน

2.2.2 การส่งข้อมูลแบบอนุกรม

ลักษณะของการส่งข้อมูลแบบอนุกรมนั้นข้อมูลจะถูกส่งออกมาทีละบิตจากอุปกรณ์ส่งไปยังอุปกรณ์รับ ช่องสัญญาณในการส่งข้อมูลอาจจะใช้เพียง 1 หรือ 2 ช่องสัญญาณเท่านั้น ทำให้ค่าใช้จ่ายในการสื่อสารถูกกว่าแบบขนาน แต่อัตราการส่งข้อมูลจะล่าช้ากว่า การส่งแบบขนาน การส่งข้อมูลแบบอนุกรม ข้อมูลที่ต้องการส่งซึ่งจะอยู่ในลักษณะเป็นไบท์จะถูกทยอยส่งทีละบิตและทางอุปกรณ์รับจะต้องรับข้อมูลเข้ามาทีละบิตแล้วมารวมกันเป็นไบท์ซึ่ง

ทางด้านอุปกรณ์รับจะต้องคอยตรวจสอบว่า บิตใดเป็นบิตเริ่มแรกของไบนารีนั้น การตรวจสอบขึ้นอยู่กับรูปแบบของบิตที่ทยอยส่งจากอุปกรณ์ส่ง การส่งข้อมูลแบบอนุกรมแสดงได้ดังรูป 2.2



รูปที่ 2.2 การส่งข้อมูลแบบอนุกรม

การส่งข้อมูลแบบอนุกรมมี 2 แบบ คือ

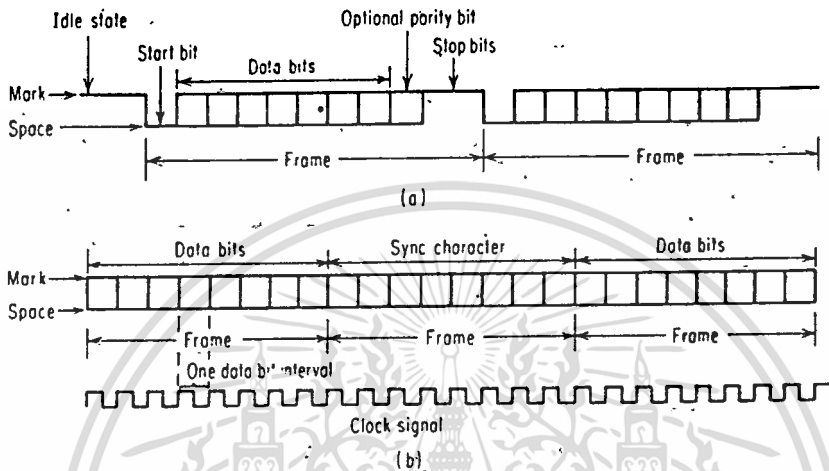
ก.) การส่งแบบอะซิงโครนัส (Asynchronous Transmission)

ในการส่งแบบอะซิงโครนัส ข้อมูลที่ต้องการส่งแต่ละตัวหรือแต่ละไบนารีนั้นจะมีสัญญาณสำหรับตรวจสอบบิตแรกภายในตัวของมันเอง โดยแต่ละไบนารีจะถูกเพิ่มด้วยบิตเริ่มต้น (Start Bit) นำหน้าไบนารีนั้นและบิตสิ้นสุด (Stop Bit) ตามหลังไบนารีนั้น ซึ่งอาจจะมีการเพิ่มบิตพาริตีก่อนบิตสิ้นสุดก็ได้ ระยะเวลาระหว่างข้อมูลแต่ละไบนารีไม่จำเป็นต้องแน่นอน เพราะที่อุปกรณ์รับจะตรวจสอบบิตแรกที่ละไบนารีเท่านั้น โดยขณะไม่มีการส่งข้อมูลสภาพลอจิกจะเป็น "1" อุปกรณ์รับจะคอยตรวจสอบจากการเปลี่ยนลอจิก "1" เป็นลอจิก "0" (บิตเริ่มต้นมีลอจิกเป็น "0") ซึ่งหมายถึงบิตที่ตามมาหลังจากบิตเริ่มต้นก็คือบิตแรกของไบนารีนั้น รูปแบบการจัดเรียงบิตในการส่งแบบอะซิงโครนัสแสดงดังรูปที่ 2.3ก

ข.) การส่งแบบซิงโครนัส (Synchronous Transmission)

การส่งข้อมูลแบบซิงโครนัส เป็นการส่งข้อมูลที่มีการกำหนดจำนวนอักขระที่จะส่งในแฉ่งละครั้งเป็นจำนวนที่แน่นอน เรียกว่า เฟรมข้อมูล ข้อมูลที่อยู่ในเฟรมข้อมูลจะเรียงติดกันไปโดยไม่ต้องมีบิตเริ่มต้น และ บิตสุดท้าย การซิงโครไนซ์

สามารถทำได้โดยการส่งสัญญาณ Sym Character จำนวนหนึ่งนำหน้าเฟรมข้อมูลออกไปยังตัวรับสัญญาณ Syn Character นี้จะมีรูปแบบเฉพาะที่ทางภาครับจะใช้ในการทำ Clock Recovery เพื่อให้อัตราในการส่งข้อมูลและการรับข้อมูลเท่ากัน รูปแบบของการส่งแบบซิงโครนัสเป็นดังรูป 2.3 ข



รูปที่ 2.3 การจัดเรียงบิตในการส่งแบบ ก) อะซิงโครนัส ข) ซิงโครนัส

2.3 ทิศทางการสื่อสารข้อมูล

ช่องสัญญาณ อาจจะตีความหมายได้ว่า ช่องทางหนึ่งช่องทางบนสายที่สัญญาณไฟฟ้าสามารถไหลผ่านได้ ส่วนสายนั้นหมายถึงส่วนประกอบของระบบที่ เชื่อมโยงระหว่างสถานีสื่อสารสองสถานี ทิศทางการสื่อสารข้อมูลในช่องสัญญาณสามารถแบ่งได้เป็น 3 ลักษณะคือ

2.3.1 ซิมเพล็กซ์ (Simplex)

หมายถึง การส่งข้อมูลในทิศทางเดียว ในการสื่อสารแบบนี้อุปกรณ์สื่อสารด้านหนึ่ง จะส่งข้อมูลไปในช่องสัญญาณเท่านั้น และจะไม่รับข้อมูลจากช่องสัญญาณ แต่อุปกรณ์สื่อสารอีกด้านหนึ่งจะรับข้อมูลจากช่องสัญญาณเท่านั้น และจะไม่ส่งข้อมูลไปในช่องสัญญาณเลย

2.3.2 ฮาล์ฟดูเพล็กซ์ (Half Duplex)

หมายถึง การสื่อสารข้อมูลใน 2 ทิศทาง แต่ในช่วงเวลาหนึ่งได้เพียงทิศทางเดียวเท่านั้น อุปกรณ์สื่อสารทั้ง 2 ด้านจะผลัดกันรับผลัดกันส่ง การสื่อสาร

แบบนี้ส่วนใหญ่แล้วจะใช้ระบบสาย 2 เส้น (2W)

2.3.3 ฟูลดเพล็กซ์ (Full Duplex)

หมายถึง การสื่อสารข้อมูลใน 2 ทิศทางพร้อมกัน การสื่อสารแบบนี้ใช้ได้ทั้งระบบสาย 2 เส้นและสาย 4 เส้น (4W) แต่ในระบบสาย 2 เส้น จะต้องอาศัยเทคนิคการแบ่งความถี่เข้าช่วย คือจะส่งในความถี่ช่วงหนึ่ง และจะรับในความถี่อีกช่วงหนึ่ง

2.4 โมเด็ม (Modem)

ในการสื่อสารข้อมูล ลักษณะของสัญญาณไฟฟ้าที่ใช้จะเป็นดิจิทัลที่ต้องการตัวนำสัญญาณที่ตอบสนองความถี่ได้ในช่วงกว้าง จากข้อจำกัดของสายส่งทำให้สามารถส่งสัญญาณดิจิทัลได้ในระยะทางไกลๆเท่านั้น เพราะที่ระยะทางไกลๆ สภาพความเป็นตัวเก็บประจุที่เกิดขึ้นในสาย (Stray Capacitance) จะทำให้ค่า Delay Time ของแต่ละองค์ประกอบความถี่ที่รวมตัวเป็นสัญญาณดิจิทัลมีค่าไม่เท่ากันทำให้เมื่อส่งสัญญาณดิจิทัลที่ความเร็วสูงๆอาจเกิดการผิดพลาดของข้อมูลขึ้นที่ปลายทางได้

ในการสื่อสารข้อมูลระยะทางไกลๆจึงอาศัยโครงข่ายของช่องสัญญาณเสียง (Voice Bandwidth Channel) ที่กระจายครอบคลุมโดยทั่วไปอยู่แล้ว เช่น ระบบโทรศัพท์, ระบบวิทยุมา ใช้ ซึ่งช่องสัญญาณเสียงที่ใช้จะอยู่ในช่วงความถี่ 300-3400 Hz เพราะฉะนั้นการที่จะสื่อสารข้อมูลด้วยสัญญาณดิจิทัล เข้าไปในโครงข่ายดังกล่าวจึงต้องทำการมอดดูเลทสัญญาณดิจิทัลให้กลายเป็นสัญญาณเสียงอนาล็อก แล้วจึงทำการส่งออกไปในโครงข่ายดังกล่าว ในขณะที่เดียวกันเวลารับก็ต้องทำการดีมอดดูเลทสัญญาณเสียงให้กลับมาเป็นสัญญาณดิจิทัลอีกครั้งในภาครับ ซึ่งอุปกรณ์ที่ทำหน้าที่ในการมอดดูเลทและดีมอดดูเลทสัญญาณดังกล่าวนี้เรียกว่า โมเด็ม (Modem)

2.5 เทคนิคในการมอดดูเลทสัญญาณดิจิทัล (Digital Modulation Techniques)

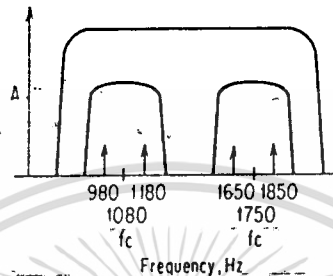
2.5.1 การมอดดูเลทแบบความถี่ (Frequency Shift Keying : FSK)

การมอดดูเลทแบบนี้เมื่อข้อมูลเป็นลอจิก "1" ก็จะได้สัญญาณอนาล็อกความถี่หนึ่ง และเมื่อเป็นลอจิก "0" ก็จะได้สัญญาณอนาล็อกอีกความถี่หนึ่ง โมเด็มแบบ FSK แบ่งออกได้เป็น 2 แบบคือ

ก.) ชนิดที่ใช้ในการส่งข้อมูลในระบบฟูลดเพล็กซ์ (FDX operation)

โมเด็มชนิดนี้จะแบ่งแบนด์วิทของช่องสัญญาณเสียงออกเป็น 2 แบนด์เท่าๆกัน โดยแบนด์

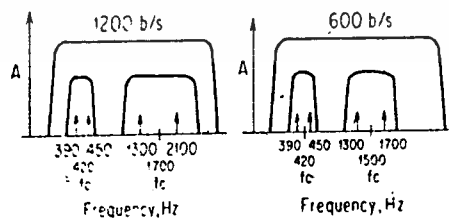
หนึ่งใช้ในการส่งข้อมูล ส่วนอีกแบนด์หนึ่งจะใช้ในการรับข้อมูล ทำให้สามารถส่งข้อมูลเข้าไปในสายพร้อมๆกันได้ ผลตอบสนองความถี่ (Frequency Response) ของโมเด็มแบบนี้เป็นดังรูป 2.4 ตัวอย่างของโมเด็มชนิดนี้ได้แก่ V.24 และ V.22 เป็นต้น



รูปที่ 2.4 ช่องสัญญาณในสายส่ง เมื่อใช้โมเด็ม FSK ฟูลูเพล็กซ์

ข.) ชนิดที่ใช้ในการส่งในระบบฮาล์ฟดูเพล็กซ์ (HDX operation)

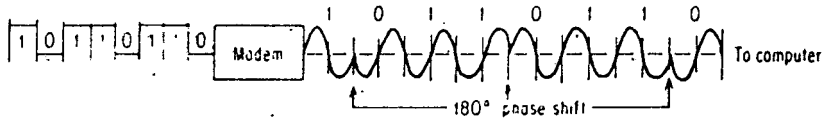
โมเด็มในระบบนี้จะมีแบนด์วิท 2 แบนด์วิทเหมือนกัน แต่ความกว้างของแบนด์วิททั้งสองจะไม่เท่ากัน แบนด์วิทที่กว้างกว่าจะใช้ในการส่งข้อมูล ส่วนแบนด์วิทที่แคบจะใช้ในการส่ง Supervisory Signal ของตัวรับไปยังตัวส่งเพื่อใช้ในการตรวจสอบสภาพการส่งข้อมูลว่าถูกต้องหรือไม่ เรียกแบนด์วิทนี้ว่า Reverse Channel ผลตอบสนองต่อความถี่ของโมเด็มแบบนี้เป็นดังรูป 2.5 ตัวอย่างของโมเด็มชนิดนี้ได้แก่ V.23



รูปที่ 2.5 ช่องสัญญาณในสายส่ง เมื่อใช้โมเด็ม FSK ฮาล์ฟดูเพล็กซ์

2.5.2 การมอดูเลตแบบเฟส (Phase Shift Keying)

การมอดูเลตแบบนี้จะใช้เฟสของสัญญาณอนาล็อกแทนสัญญาณดิจิทัล สัญญาณลอจิก "1" จะให้เฟสของสัญญาณอนาล็อกเฟสหนึ่ง ในขณะที่เดียวกันสัญญาณลอจิก "0" ก็จะให้เฟสของสัญญาณอนาล็อกอีกเฟสหนึ่งดังรูป 2.6



รูปที่ 2.6 การมอดูเลตแบบเฟส

ในกรณีที่ต้องการอัตราในการสื่อสารข้อมูลที่สูง สามารถทำได้โดยการเข้ารหัสสัญญาณดิจิทัลเป็นคู่ๆ เรียกว่า ไตบิต (Dibits) แล้วแทนแต่ละไตบิตด้วยเฟสที่แตกต่างกันดังตารางที่ 2.1 ซึ่งการเข้ารหัสดังกล่าวจะทำให้อัตราในการสื่อสารสูงขึ้น 2 เท่า มิได้มีลักษณะนี้ได้แก่ V.26 ซึ่งมีอัตราการสื่อสาร 1200 บอด แต่มีบิตเรทเป็น 2400 บิต/วินาที

ตารางที่ 2.1 การเข้ารหัสข้อมูลแบบไตบิต ตารางที่ 2.2 การเข้ารหัสข้อมูลแบบไตรบิต

Two Alternative Phase Arrangements for Dibit Coding		
Dibit	Phase change	
	Alternative A	Alternative B
00	0°	+45°
01	+90°	+135°
11	+180°	+225°
10	+270°	+315°

SOURCE: CCITT Rec. V.26.

Phase Arrangements for Tribit Coding		
Tribit values	Phase change	
001		0°
000		45°
010		90°
011		135°
111		180°
110		225°
100		270°
101		315°

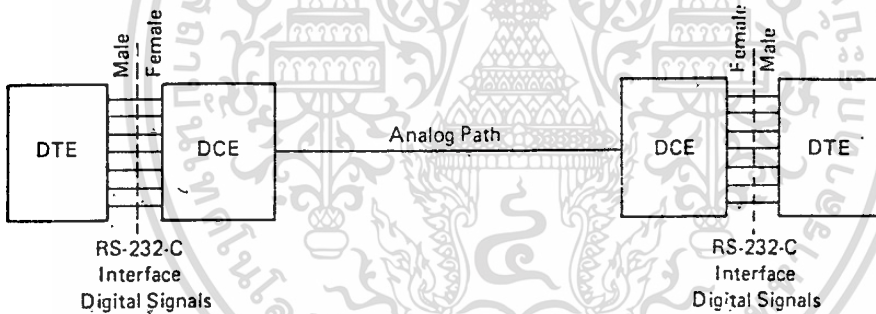
SOURCE: CCITT Rec. V.27.



ในกรณีที่ต้องการความเร็วสูงขึ้นไปอีก ก็ทำได้โดยการรวมสัญญาณดิจิทัล 3 บิต เรียกว่า ไตรบิต (Tribits) แล้วแทนด้วยเฟสต่าง ๆ กัน 8 เฟสก็จะทำให้สามารถเพิ่มอัตราในการสื่อสารเป็น 3 เท่าของบิตเรทได้ ตารางเทียบเฟสแบบ ไตรบิตแสดงได้ดังตารางที่ 2.2 ตัวอย่างของโมเด็มชนิดนี้ ได้แก่ V.27

2.6 อินเทอร์เฟส EIA RS-232C

EIA RS-232C เป็นมาตรฐานที่ใช้ในการอินเทอร์เฟสระหว่างอุปกรณ์ DTE ได้แก่ คอมพิวเตอร์, พรินเตอร์ หรือ teletypewriter เข้ากับอุปกรณ์ DTE ซึ่งโดยทั่วไปก็ได้แก่ โมเด็ม ดังรูป 2.7 มาตรฐาน EIA RS-232C เป็นมาตรฐานที่ได้รับการพัฒนาขึ้นมาโดย Electronics Industries Association ซึ่งจะกล่าวถึงมาตรฐานต่อไปนี้เป็นคือ



รูปที่ 2.7 การต่อ DTE เข้ากับ DCE โดย RS-232C

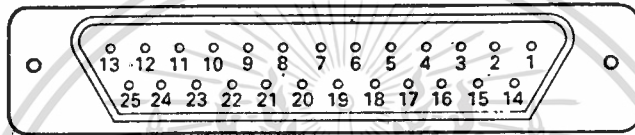
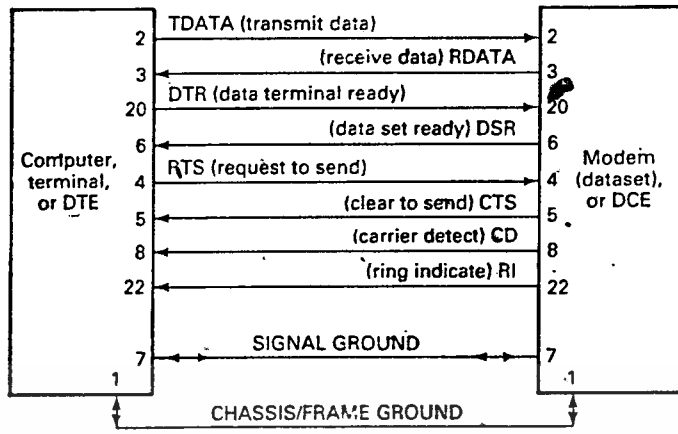
ก.) ระดับศักดาไฟฟ้าของสัญญาณ (Signal Voltage Level)

ข.) ชนิดของหัวต่อ (Connectors) ที่ใช้และการจัดขาสัญญาณ

(Pin Assignment)

ค.) สัญญาณต่างๆที่จำเป็นต้องใช้ในการอินเทอร์เฟส

อินเทอร์เฟส RS-232C จะใช้หัวต่อแบบ DB-25 (D-type 25 pin Connector) เป็นมาตรฐาน โดยหัวต่อบน DTE จะเป็นตัวผู้ดังรูป 2.8 ส่วนหัวต่อที่ DCE จะเป็นตัวเมีย

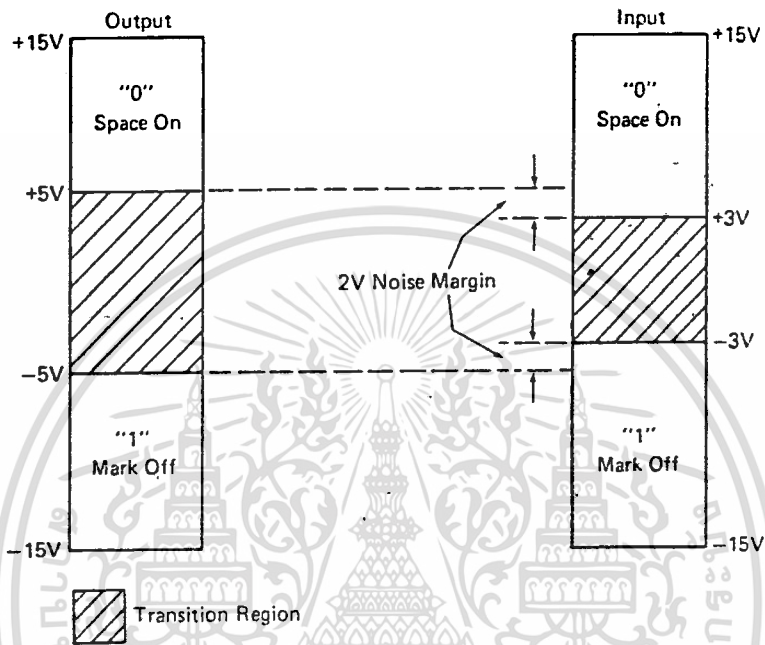


รูปที่ 2.8 หัวต่อ DB-25 Pin ที่ใช้บน RS-232C

ระดับสัญญาณที่ใช้ใน RS-232C จะเป็นแบบ Bipolar Voltage Signals ระดับสัญญาณต่างๆเมื่อเทียบกับกราวด์จะต้องไม่เกิน ± 25 โวลต์ และเมื่อเกิดการช็อตกันระหว่างขาต่างๆ จะต้องไม่เกิดความเสียหายขึ้นกับอุปกรณ์ใดๆที่ต่ออยู่กับ RS-232C รวมทั้งตัวมันเองด้วย สัญญาณเอาต์พุตลอจิก "0" เมื่อผ่าน RS-232C จะมีระดับคิกคาไฟฟ้ายู่ในช่วง $+5$ ถึง $+15$ โวลต์ และ ลอจิก "1" จะมีระดับคิกคาไฟฟ้ายู่ในช่วง -5 ถึง -15 โวลต์ ส่วนที่ตัวรับสัญญาณอินพุตลอจิก "0" จะมีค่าในช่วง $+3$ ถึง $+15$ โวลต์ และสัญญาณอินพุตลอจิก "1" จะมีค่าในช่วง -3 ถึง -15 โวลต์ สำหรับระดับคิกคาในช่วง ± 3 โวลต์จะไม่มีมีความหมายใดๆ ดังรูป 2.9

สัญญาณต่างๆที่ใช้ในมาตรฐาน EIA RS-232C แบ่งออกเป็น 5 กลุ่มคือ

- กลุ่ม A - Ground or Common Return
- กลุ่ม B - Data Circuits
- กลุ่ม C - Controls Circuits
- กลุ่ม D - Timing Circuits
- กลุ่ม S - Secondary Channel Circuits



รูปที่ 2.9 ระดับของสัญญาณที่ใช้บน RS-232C

สัญญาณต่างๆที่ใช้ใน RS-232C รวมทั้งทิศทางของสัญญาณเป็นไปดัง

ตารางที่ 2.3

สัญญาณต่างๆที่ใช้บน RS-232C มีหน้าที่ดังนี้

Transmit Data (TD ขาที่ 2)

เป็นสัญญาณที่ส่งออกจาก DTE (หรือตัวไมโครคอมพิวเตอร์) ไปยังโมเด็มหรือ ต่อเข้าโดยตรงกับคอมพิวเตอร์ตัวอื่นหรือเครื่องพิมพ์ เมื่อไม่มีสัญญาณส่งออก สถานะของลอจิกที่ขานี้จะเหมือนกับ "1" หรือเทียบเท่ากับสตอปบิต

Receive Data (RD ขาที่ 3)

เป็นทางของสัญญาณเข้าไปยัง DTE หรือไมโครคอมพิวเตอร์เมื่อไม่

ตารางที่ 2.3 ชนิดและทิศทางของสัญญาณทั้งหมดที่ใช้บน RS-232C

Interchange Circuit	Pin Assignment	Description	Mnemonic	Data		Control		Timing	
				From DCE	To DCE	From DCE	To DCE	From DCE	To DCE
AA BB	1 7	Protective Ground Signal Ground/Common Return	PG SG						
BA BB	2 3	Transmitted Data Received Data	TD RD	X	X				
CA CB CC CD CE CF CG CH CI	4 5 6 20 22 8 21 23	Request to Send Clear to Send Data Set Ready Data Terminal Ready Ring Indicator Received Line Signal Detector Signal Quality Detector Data Signal Rate Selector (DTE) Data Signal Rate Selector (DCE)	RTS CTS DSR DTR RI DCD SQ				X X X X X		
DA DB DD	24 15 17	Transmitter Signal Element Timing (DTE) Transmitter Signal Element Timing (DCE) Receiver Signal Element Timing (DCE)	TSET TSET RSET					X X	X
SBA SBB	14 16	Secondary Transmitted Data Secondary Received Data	(S)TD (S)RD	X	X				
SCA SCB SCF	19 13 12	Secondary Request to Send Secondary Clear to Send Secondary Rec'd Line Signal Detector	(S)RTS (S)CTS (S)DCD				X X X		

มีสัญญาณรับเข้ามา ขานี้จะมีสถานะภาพทางลอจิกเป็น "1"

Request To Send (RTS ขาที่ 4)

ใช้สำหรับส่งสัญญาณไปยังโมเด็มหรือเครื่องพิมพ์เป็นการเรียกร้องที่จะส่งสัญญาณมาทางขา 2 สัญญาณนี้จะใช้คู่กับ CTS หรือ Clear to send อุปกรณ์รับ หากได้รับสัญญาณ RTS จะตรวจสอบตัวเองว่าพร้อมจะรับสัญญาณได้หรือยัง หากพร้อมที่จะรับก็ส่งสัญญาณออกไปที่สาย CTS

Clear To Send (CTS ขาที่ 5)

ตั้งอธิบายไว้ใน RTS เมื่อสัญญาณนี้อยู่ในสถานะภาพออฟ (negative voltage หรือลอจิก "1") หมายความว่า อุปกรณ์รับกำลังบอกว่าพร้อมที่จะรับข้อมูลแล้ว

Data Set Ready (DSR ขาที่ 6)

เมื่อสัญญาณสายนี้อยู่ในสถานะออน (หรือลอจิก 0) เป็นการบอกไมโครคอมพิวเตอร์หรือฝ่ายส่งว่า โมเด็มต่อเข้ากับสายโทรศัพท์เรียบร้อยแล้วและพร้อมที่จะส่งได้แล้ว โมเด็มที่มีการหมุนเวียนเลขอัตโนมัติจะส่งสัญญาณสายนี้ไปบอกให้คอมพิวเตอร์รู้ว่าต่อโทรศัพท์ได้สำเร็จแล้ว

Signal Ground (SG ขาที่ 7)

SG ทำหน้าที่เป็นระดับแรงดันอ้างอิงสำหรับทุกสายของสัญญาณ จะมีแรงดันเป็น "0" เมื่อเทียบกับสัญญาณตัวอื่น

Carrier Detect (CD ขาที่ 8)

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

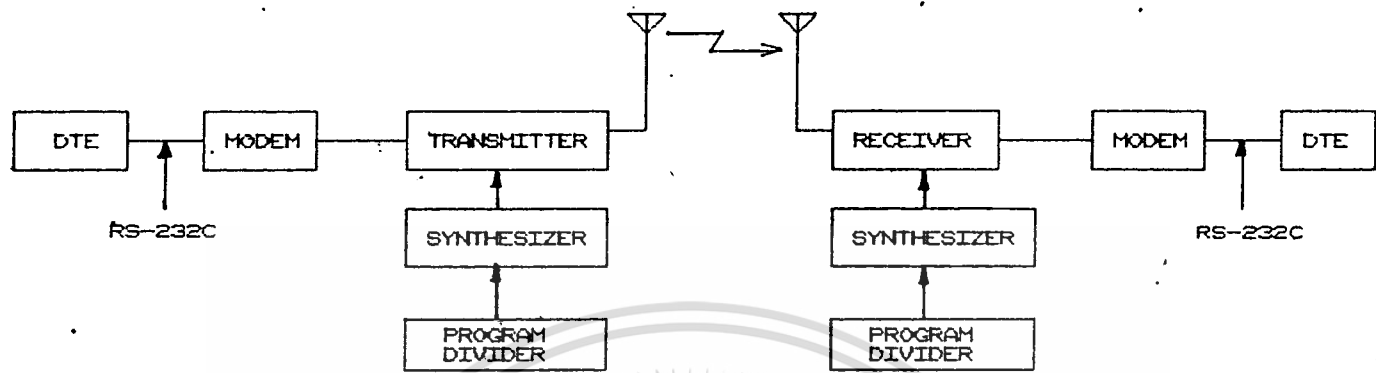
Data Terminal Ready (DTR ขาที่ 20)

คอมพิวเตอร์เปิดสัญญาณสายนี้ให้ออน (ลอจิก "0") เมื่อพร้อมที่จะติดต่อกับโมเด็ม โมเด็มส่วนมากจะไม่รายงานสถานะภาพของตัวเอง (CD, USR และ CTS) ให้คอมพิวเตอร์รู้ หากคอมพิวเตอร์ไม่เปิดสัญญาณ DTR

๒) 7 หลักการทำงานของโครงงานเบื้องต้น

การสื่อสารข้อมูลผ่านคลื่นวิทยุด้วยระบบ 40 CHANNEL WIRELESS MODEM เป็นการส่งข่าวสารระหว่างอุปกรณ์ปลายทางข้อมูล Data Terminal Equipment (DTE) ซึ่งในโครงงานนี้คือ Personal Computer โดยมีจุดมุ่งหมายเพื่อให้ Personal Computer จำนวนมากที่อยู่ในบริเวณไม่ไกลกันมากนัก (Local Area) สามารถที่จะส่งข่าวสารจากต้นทางไปยังปลายทางพร้อมๆกันได้โดยผ่านตัวกลาง ซึ่งก็คืออากาศร่วมกันในเวลาเดียวกันได้

การทำงานของระบบในโครงงาน จะอาศัยการมอดูเลตสัญญาณข่าวสารเข้ากับคลื่นพาห์ (Carrier) ด้วยความถี่ที่แตกต่างกันไป ข่าวสารจาก Personal Computer หนึ่งจะถูกมอดูเลตด้วยคลื่นพาห์ความถี่หนึ่ง ส่วนข่าวสารจาก Personal Computer อีกเครื่องหนึ่งก็จะถูกมอดูเลตด้วยคลื่นพาห์อีกความถี่หนึ่ง จากหลักการดังกล่าวสามารถเขียนเป็นบล็อกไดอะแกรมง่ายๆได้ดังรูปที่ 2.10

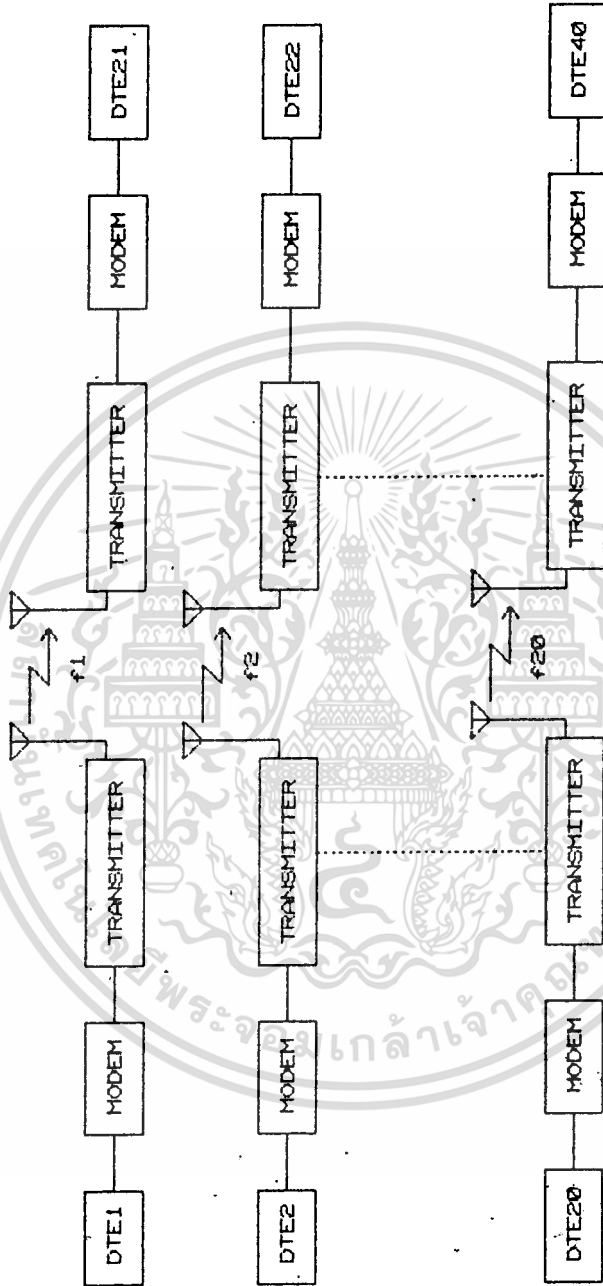


รูป 2.10 บล็อกไดอะแกรมอย่างง่ายของโครงการ

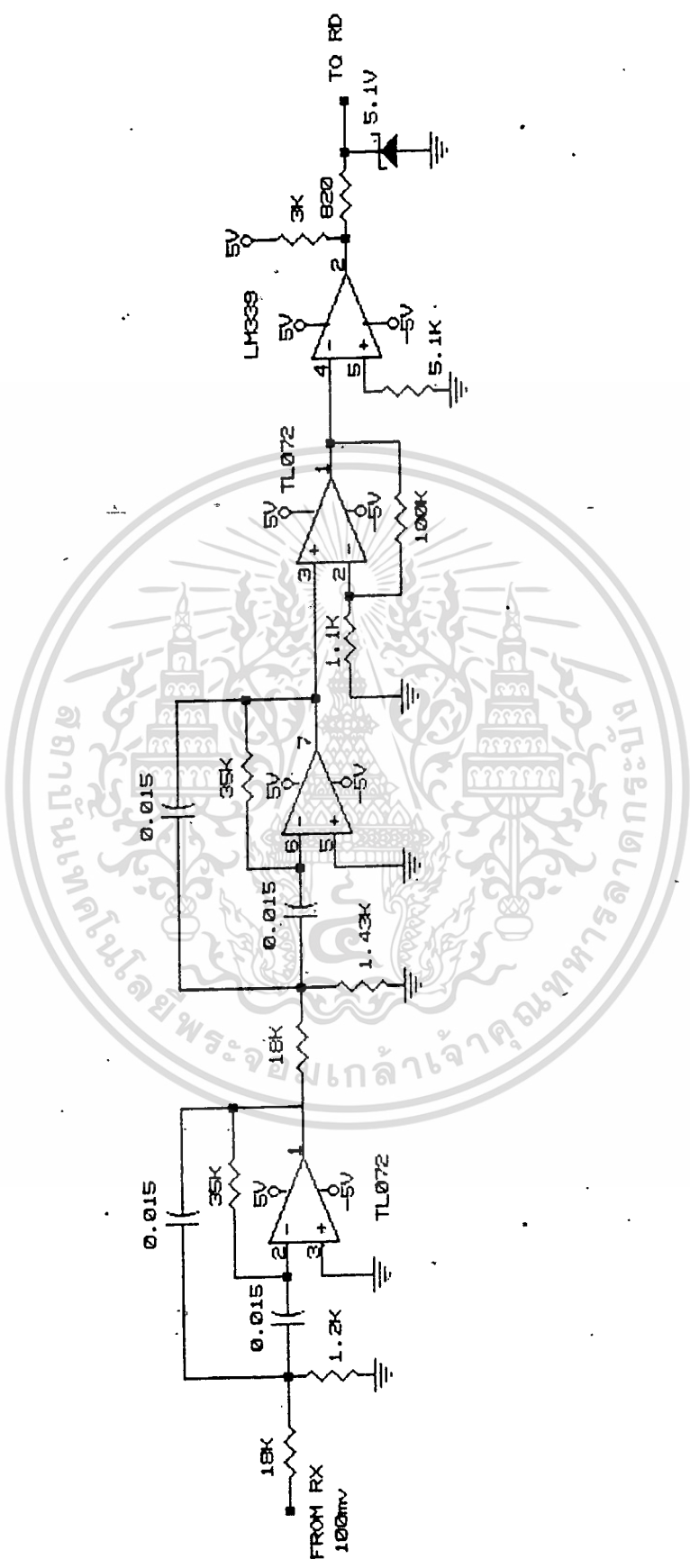
จากบล็อกไดอะแกรม โมเด็มจะทำหน้าที่ในการอินเตอร์เฟสกับ Personal Computer ผ่านทาง RS-232C ข่าวดิจิทัลที่ส่งจากคอมพิวเตอร์จะถูกแปลงเป็นสัญญาณอนาล็อก โดยการเข้ารหัสความถี่ (Frequency Shift Keying) แล้วทำการมอดูเลตกับคลื่นพาห์ ซึ่งถูกสังเคราะห์ขึ้นด้วยความถี่ต่างๆโดยวงจรสังเคราะห์ความถี่ (Frequency Synthesizer) ความถี่ที่ใช้จะใช้ความถี่เท่าใดก็ขึ้นอยู่กับทางสถานีภาครับว่าได้ตั้งความถี่ในการรับเอาไว้เท่าไร

สำหรับในโครงการนี้ได้เลือกความถี่ที่ใช้ในการสื่อสารในย่าน Citizen Band Class D ซึ่งมีความถี่ในช่วง 26.965 - 27.465 Mhz จำนวน 40 ช่องสัญญาณมาใช้ โดยกำหนดให้สถานีภาครับแต่ละสถานีมีความถี่ในการรับเป็นไปตามตารางที่ 2.4

จากลักษณะดังกล่าวจะทำให้ได้โครงข่ายข้อมูลวิทยุที่สามารถส่งข่าวสารผ่านคลื่นวิทยุได้พร้อมๆกันเป็นจำนวน 20 ข่าวสาร โดยมีลักษณะการส่งเป็นแบบซิมเพล็กซ์ (Simplex) ดังรูป 2.11



รูปที่ 2.11 ลักษณะการสื่อสารด้วยระบบ โมเด็มไร้สาย



รูปที่ 1.5 วงจร E negative data

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
 ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 3

การออกแบบและการสร้าง

จากหลักการที่แสดงโดยบล็อกไดอะแกรมที่ 3.1 จะแบ่งการออกแบบและการสร้างออกเป็น 2 ส่วนคือ ในส่วนของโมเด็ม (Modem) , ส่วนของเครื่องส่งวิทยุ (Transmitter) และ ในส่วนของโปรแกรมที่ใช้ในการสื่อสาร สำหรับเครื่องรับวิทยุได้นำเอาเครื่องรับวิทยุที่มีอยู่แล้วมาใช้

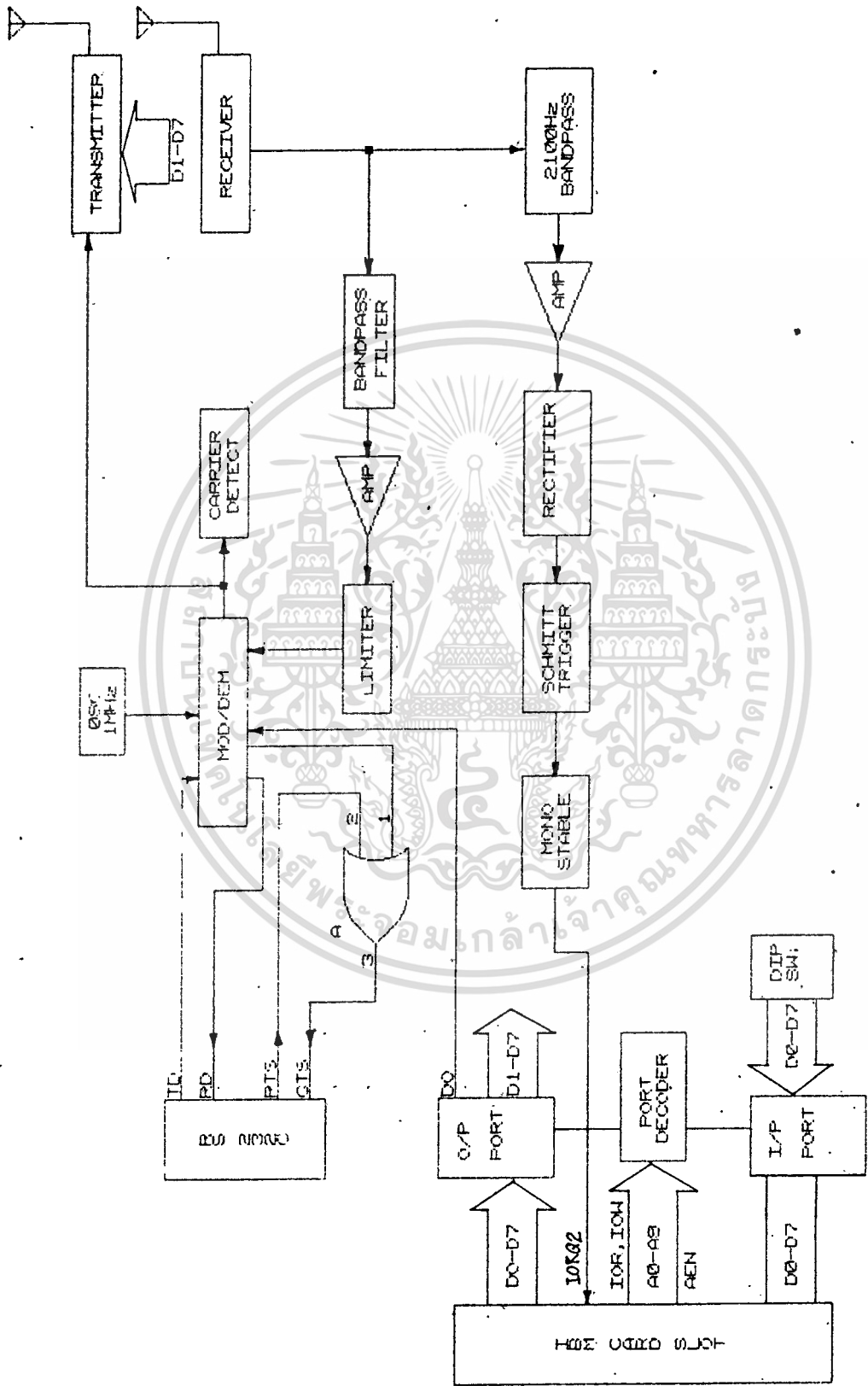
3.1 การออกแบบโมเด็มความเร็วต่ำ

รูปที่ 3.1 แสดงบล็อกไดอะแกรมของโมเด็มที่ใช้ในโครงการโดยมีหลักการการทำงานดังนี้ เมื่อเริ่มการทำงานโมเด็มจะทำการส่งสัญญาณควบคุมผ่านทางเอาต์พุตพอร์ต เพื่อไปควบคุมภาค Mod/Dem เพื่อสร้างสัญญาณ Ringing ความถี่ 2100Hz ในขณะที่เดียวกันก็จะส่งสัญญาณควบคุมอีกชุดหนึ่งไปควบคุมภาคส่งเคราะห์ความถี่ ให้ผลิความถี่คลื่นพาห้ให้มีความถี่ตามสถานีภาครับ แล้วเครื่องส่งวิทยุจะทำการมอดดูเลทสัญญาณความถี่ 2100 Hz ส่งออกอากาศไปยังสถานีภาครับ

ทางสถานีรับจะทำการกรองความถี่ เอาเฉพาะความถี่ 2100 Hz แล้วทำการเรคตีไฟร์ให้เป็นสัญญาณไฟ dc เพื่อไปทริกโมโนสเตเบิลให้ส่งสัญญาณ IORQ2 (Interrupt Request 2) ไปหยุดการทำงานของคอมพิวเตอร์ทางภาครับ คอมพิวเตอร์ภาคส่งจะรอเวลาด้วยระยะเวลาค่าหนึ่งเพื่อให้คอมพิวเตอร์ทางภาครับรับข้อมูลเสร็จเสียก่อน ต่อจากนั้นจึงทำการส่งข้อมูลโดยการโปรแกรมตัวอะแดปเตอร์การสื่อสารแบบอะซิงโครนัสผ่านอินเตอร์เฟส RS-232C ไปยังโมเด็ม โมเด็มจะทำการเข้ารหัสแถบความถี่โดยลอจิก "1" จะมีความถี่เป็น 1680 Hz ลอจิก "0" จะมีความถี่เป็น 1850 Hz แล้วส่งไปยังเครื่องรับวิทยุเพื่อส่งข่าวสารออกอากาศไปยังสถานีภาครับ

ทางสถานีภาครับเมื่อได้รับข้อมูลที่ส่งจากสถานีภาคส่ง ก็จะกรองความถี่เอาเฉพาะความถี่ 1680, 1850 Hz ผ่านเข้าวงจรขยายและลิมิตเตอร์เพื่อแปลงสัญญาณชายน์เวฟให้เป็นสัญญาณแควร์เวฟความถี่ 1680, 1850 Hz แล้วทำการดีโมดูเลทก็จะได้อข้อมูลซึ่งจะถูกส่งไปยังคอมพิวเตอร์โดยผ่าน อินเตอร์เฟส RS-232C

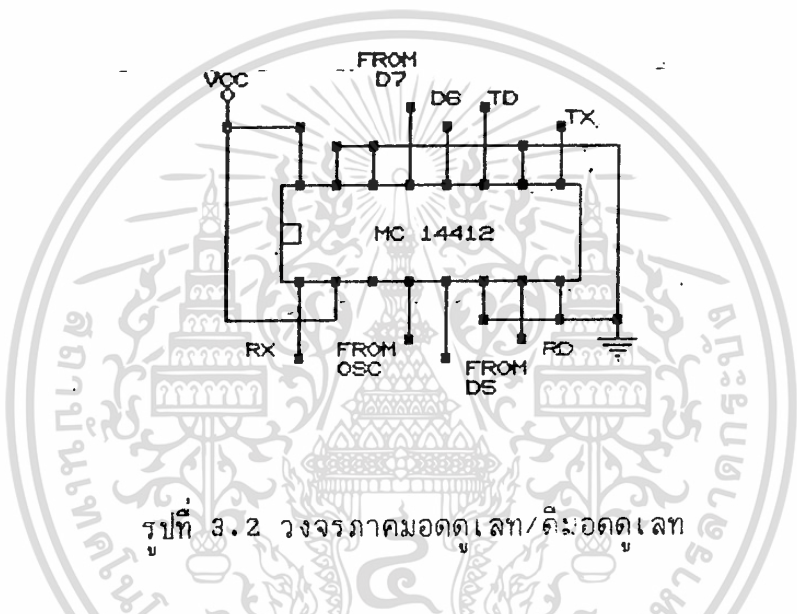
จากหลักการทำงานดังกล่าว สามารถนำมาออกแบบและสร้างส่วนต่างๆของโมเด็มได้ดังนี้



รูปที่ 3.1 นวัตกรรมของโมเด็มที่ใช้ในโครงการ

3.1.1 ภาค MOD/DEM

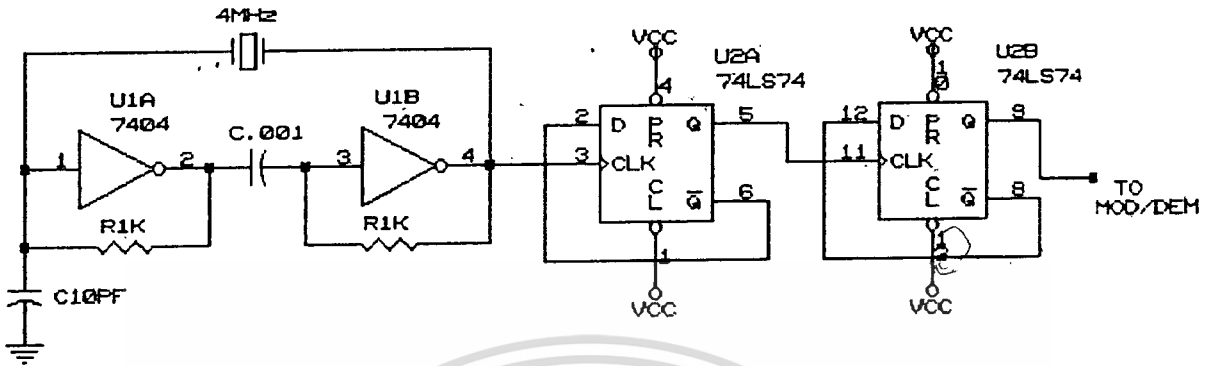
ในภาคนี้จะใช้ไอซี MC14412 ซึ่งจะทำหน้าที่ในการมอดดูเลขสัญญาณดิจิทัล โดยการเข้ารหัสแถบความถี่ FSK (Frequency Shift Keying) โดยมีอัตราการสื่อสาร 600 บอด ลอจิก "1" จะถูกแทนด้วยสัญญาณอนาล็อกความถี่ 1680 Hz และ ลอจิก "0" จะถูกแทนด้วยความถี่ 1850 Hz ในขณะที่เดียวกันไอซีตัวนี้จะทำหน้าที่ในการตีโมดเลข เพื่อให้ได้ข้อมูลดั้งเดิมด้วย โดยที่สัญญาณที่จะนำมาตีโมดเลขจะต้องเป็นสัญญาณสแควร์เวฟ ซึ่งมี Duty Cycle 50% ความถี่ 1680 และ 1850 Hz



รูปที่ 3.2 วงจรภาคมอดดูเลข/ตีมอดดูเลข

3.1.2 ภาค OSCILLATOR

ภาคนี้จะทำหน้าที่ในการกำเนิดสัญญาณนาฬิกาความถี่ 1 MHz เพื่อเป็นความถี่พื้นฐานให้กับไอซี MC14412 ภาคนี้จะใช้วงจร Inverter Oscillator โดยใช้ไอซี TTL อินเวอร์ตเตอร์ 74LS04 ร่วมกับค่าความต้านทานและตัวเก็บประจุ และใช้คริสตัล ความถี่ 4 MHz ความถี่จะถูกหาร 4 โดยใช้ ดีฟลิป-ฟลอป 74LS74 เป็นตัวหารเพื่อให้ได้ความถี่ 1 MHz ดังแสดงดังรูป 3.3

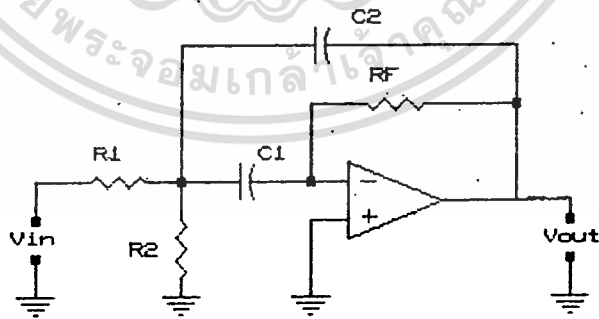


รูป 3.3 วงจรภาค OSCILLATOR

3.1.3 ภาค RECEIVE DATA

ลิมิตเตอร์

ภาคนี้จะประกอบด้วย 1080, 1850 แบนด์พาสฟิลเตอร์, แอมป์ลิฟายด์ และฟิลเตอร์ที่ใช้จะเป็นแบบ Multiple Feedback Bandpass ดังรูป 3.4



รูป 3.4 วงจรฟิลเตอร์แบบ Multiple Feedback Bandpass Filter

โดยมีสมการในการคำนวณดังนี้

$$Q = F_c / BW \quad ; \quad F_c = \text{Cut Off Frequency} \quad (3.1)$$

$$AV_c < 2Q^2 \quad (3.2)$$

$$C_1 = C_2 = C$$

$$R_1 = Q / (2 * F_c * AV_c * C) \quad (3.3)$$

$$R_2 = Q / [2 * F_c * (2Q^2 - AV_c)] \quad (3.4)$$

$$R_f = Q / F_c * C \quad (3.5)$$

$$F_c = (1/2) * (R_1 + R_2) / R_1 R_2 R_f C^2 \quad (3.6)$$

จากสมการดังกล่าวเรานำเอาออกมาออกแบบภาค 1680, 1850 แบนด์พาส โดยใช้โอปแอมป์ TL 072 โดยกำหนดให้ $F_c = 1750$, $BW = 600$ Hz เพราะฉะนั้นจะได้

$$Q = 2.81$$

ใช้ค่า $C_1 = C_2 = 0.015$ F

$$AV_c = 1$$

จะได้ $R_1 = 18K$

$$R_2 = 1.2K$$

$$R_f = 35K$$

เอาที่พ่วงจากภาคแบนด์พาสฟิลเตอร์จะถูกขยายด้วยวงจรขยายแบบไม่กลับเฟส (Noninverting Amplifier) โดยใช้โอปแอมป์ TL072 ซึ่งค่า Close Loop Gain จะหาได้จาก

$$AV_c = 1 + (R_f / R_{in}) \quad (3.7)$$

ในที่นี้ใช้ค่า

$$R_f = 100K$$

$$R_{in} = 1.1K$$

เพื่อขยายอินพุต 100 mV ให้ได้โวลเตจ 3.4 โวลต์ จากนั้นจะต่อเข้ากับภาค ลิมิตเตอร์ที่ใช้ไอซี LM339 ทำหน้าที่เป็น Inverting Comparator โดยมีระดับคัทดา ไฟฟ้าอ้างอิงเป็น 0 โวลต์ เพื่อให้ได้เอาต์พุตเป็นสัญญาณสแควร์เวฟ ซิเนอร์ไดโอดจะช่วย รักษากระดับแรงดันให้อยู่ในช่วง 0-4 โวลต์ โดยมีความต้านทาน 820 โอห์มเป็นตัวจำกัด กระแส ซึ่งจะได้วงจรที่สมบูรณ์ของการทำงานในส่วน Receive Data ดังแสดงในรูป 3.5

ตาราง 2.4 แสดงช่องสัญญาณ (HF) Class, D ที่อนุญาตให้ใช้.

Channel	Frequency (MHz)	Channel	Frequency (MHz)
1	26.965	21	27.215
2	26.975	22	27.225
3	26.985	23	27.235
4	27.005	24	27.245
5	27.015	25	27.255
6	27.025	26	27.265
7	27.035	27	27.275
8	27.055	28	27.285
9	27.065	29	27.295
10	27.075	30	27.305
11	27.085	31	27.315
12	27.105	32	27.325
13	27.115	33	27.335
14	27.125	34	27.345
15	27.135	35	27.355
16	27.155	36	27.365
17	27.165	37	27.375
18	27.175	38	27.385
19	27.185	39	27.395
20	27.195	40	27.405

3.1.4 ภาคอินเทอร์รัพท์

ภาคอินเทอร์รัพท์จะเริ่มจากการกรองความถี่ Ringing 2100 Hz โดยวงจร Multiple Feedback Bandpass Filter ซึ่งใช้อุปกรณ์ TL072 ซึ่งกำหนดให้

$$F_c = 2100 \text{ Hz}$$

$$Q = 10$$

$$A_{Vc} = 2$$

$$C_1 = C_2 = 0.015 \text{ F}$$

จากสมการ 3.1 - 3.6 จะได้

$$R_1 = 25K$$

$$R_2 = 320K$$

$$R_f = 100K$$

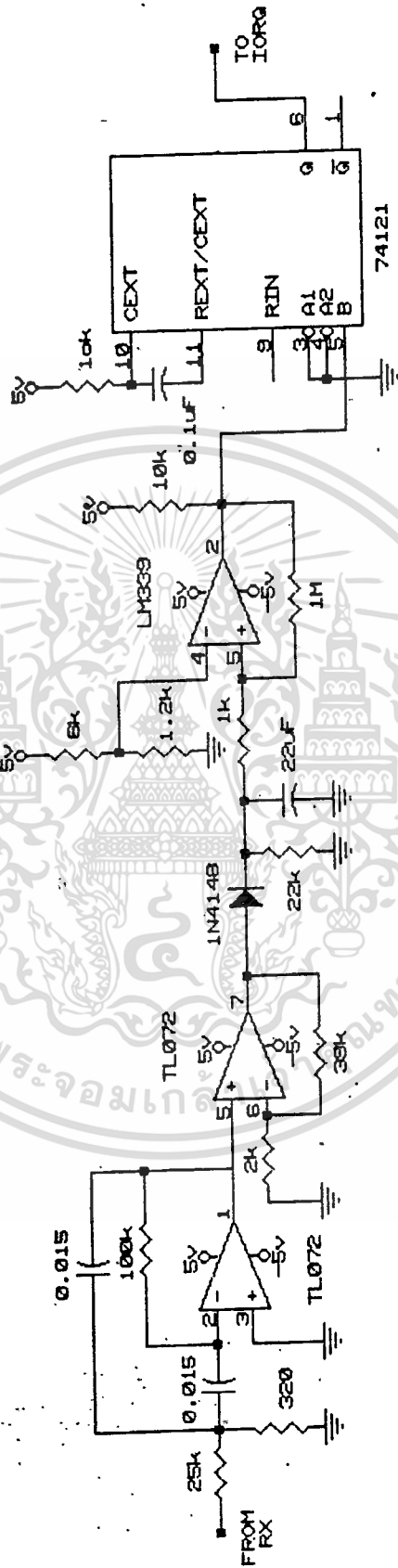
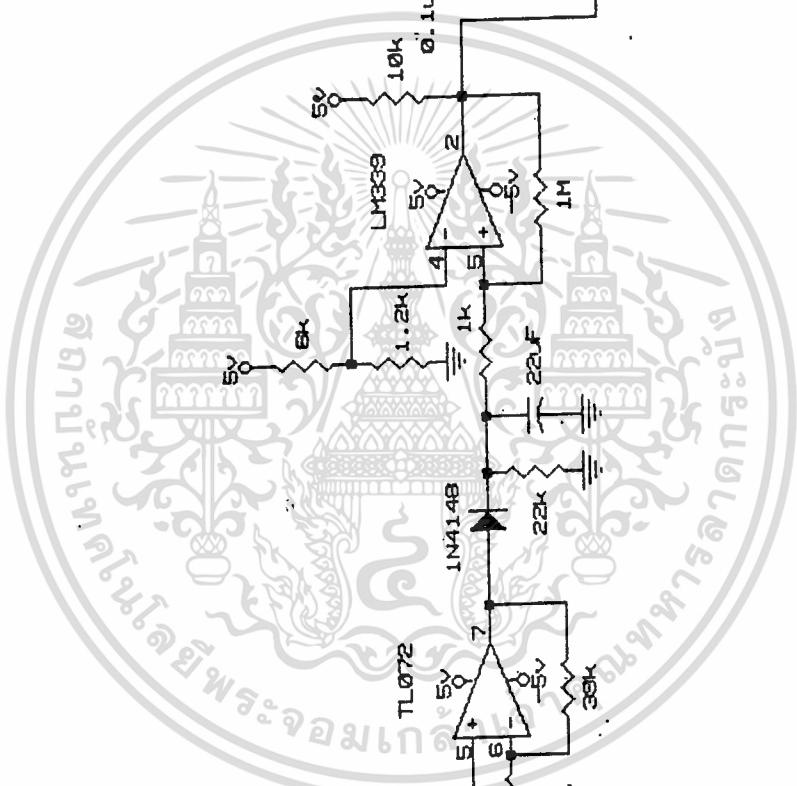
จากอินพุต 100 mV เมื่อผ่านวงจรฟิลเตอร์จะได้ระดับสัญญาณเป็น AC 0.15 V_{rms} ทำการขยายด้วยวงจรขยายแบบไม่กลับเฟส โดยใช้อุปกรณ์ TL072 ค่า Close Loop Gain = 20 ก็จะได้เอาต์พุต 3 V_{rms} จากนั้นทำการเรกซ์ไฟล์ เพื่อให้ได้สัญญาณไฟ DC แล้วเปรียบเทียบกับวงจรถ่าย Noninverting Comparator ซึ่งใช้อุปกรณ์ LM339 โดยตั้งระดับแรงดันอ้างอิงเป็น 0.9 โวลท์

ในกรณีที่สัญญาณ Ringing 2100 Hz เข้ามาในระดับสัญญาณไฟ DC จะมีค่าเป็น 1 โวลท์ และทำให้เอาต์พุตของ Comparator มีค่าเป็น 5 โวลท์ เพื่อไปทริกการทำงานของ Monostable Vibrator ส่วนที่ความถี่อื่นๆระดับ DC จะต่ำกว่าระดับแรงดันอ้างอิงและจะได้เอาต์พุตเป็น 0 โวลท์ วงจรของภาคอินเทอร์รัพท์แสดงได้ดังรูป 3.6

3.1.5 วงจรกรองความถี่ต่ำ

วงจรกรองความถี่ต่ำจะใช้เพื่อทำให้สัญญาณเอาต์พุตของตัวโมเด็ม (IC 14412) ซึ่งเป็นขั้วบันไดให้กลายเป็นสัญญาณชานส์เวฟ

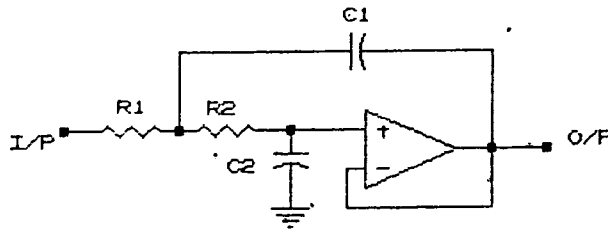
วงจรกรองความถี่ต่ำที่ใช้จะใช้วงจรกรองความถี่อันดับ 2 บัตเตอร์เวิร์ท (Second Order Butterworth Low Pass Filter) โดยมีวงจรแสดงได้ดังรูป 3.7



รูปที่ 3.6 วงจรเกิด interrupt

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.7 วงจรกรองความถี่ต่ำแบบบัตเตอร์เวิร์ท

จากรูป

ค่า R_1, R_2, C_1, C_2 หาได้จากสมการต่อไปนี้

$$C_1 F = 1.414$$

$$C_2 F = 0.707$$

$$F_{cutoff} = \text{cut Off Frequency}$$

$$C_1 = C_1 / 2 F_{cutoff}$$

$$= 1.414 / 2 \cdot 4000 = 5.626E^{-5}$$

$$C_2 = 0.707 / 2 F_{cutoff}$$

$$= 0.707 / 2 \cdot 4000 = 2.81E^{-5}$$

ทำ Impedance Scaling โดยการเลือกค่า

$$C_1 \text{ Max} = 0.01 \text{ F}$$

$$R_1 = R_2 = R$$

จาก

$$C_1 = C_1 / R$$

$$R = C / C$$

$$= 5.626E^{-5} / 10^{-8} = 5.626K$$

เลือกใช้ค่า

$$R = 5.6K$$

$$C_2 = C_2 / R$$

$$= 2.813E^{-5} / 5.626 = 0.005 \text{ F}$$

3.1.6 ภาคอินพุท - เอาท์พุทพอร์ท

การอินเตอร์เฟสกับเครื่อง IBM Pc โดยผ่านทาง Card Slot บน Main board ของ IBM Pc สามารถทำได้โดยสัญญาณต่างๆที่ใช้ในการอินเตอร์เฟสจะประกอบด้วย

1) $D_0 - D_7$

Data bus เป็นสายสัญญาณสองทิศทาง (Bidirectional) ที่ใช้ในการส่งข้อมูลระหว่าง 8088 ไมโครโปรเซสเซอร์ กับหน่วยความจำและอินพุทเอาท์พุทพอร์ท การใช้งาน Data bus ในภาวะปกติจะอยู่ภายใต้การควบคุมของ 8288 Bus Controller แต่ในภาวะ Direct Memory Access (DMA) ดาต้าบัสจะถูกตัดออกจากตัวไมโครโปรเซสเซอร์และการใช้งานของดาต้าบัสจะถูกควบคุมโดย 8337-5 DMA Controller

2) $A_0 - A_7$

แอดเดรสบัส เป็นสายสัญญาณที่มีทิศทางออกจากไมโครโปรเซสเซอร์เพียงทิศทางเดียว สัญญาณทั้ง 20 สัญญาณจะใช้ในการบ่งตำแหน่งต่างๆบนหน่วยความจำและอินพุท-เอาท์พุทพอร์ท ซึ่งการเลือกตำแหน่งต่างๆบนหน่วยความจำจะถูกกำหนดโดยตัว 8088 ซึ่งจะให้ System Bus Memory & Write Cycle เป็นตัวกำหนดจังหวะในการทำงาน ส่วนการใช้แอดเดรสบัสในการบ่งตำแหน่งของอินพุท-เอาท์พุทพอร์ท จะทำโดยผ่าน Input/Output Port Read & Write Cycle โดยคำสั่ง IN และ OUT

3) IOR (I/O Read)

สัญญาณนี้เป็นสัญญาณที่ถูกขับออกมาจาก 8288 Bus Controller เพื่อแสดงให้ I/O Port รู้ว่าเป็น I/O Port Read Cycle และแอดเดรสบนแอดเดรสบัสคือ อินพุทพอร์ทแอดเดรส สัญญาณนี้จะแอกทีฟที่ Low ในช่วงที่สัญญาณนี้แอกทีฟอินพุทพอร์ทจะต้องส่งดาต้าจำนวน 8 บิต ไปบนดาต้าบัสก่อนช่วงขอบขาขึ้นของสัญญาณนี้อย่างน้อย 30 nSec

4) IOW (I/O Write)

สัญญาณนี้เป็นสัญญาณที่แอกทีฟ Low ซึ่งถูกขับออกมาจาก 8288 Bus Controller เพื่อแสดงให้อาท์พุทพอร์ทรู้ว่าเป็น I/O Port Write Cycle และแอดเดรสที่อยู่บนแอดเดรสบัสก็คือ แอดเดรสของเอาท์พุทพอร์ทที่จะต้องรับข้อมูลจำนวน 8 บิต ที่อยู่บนดาต้าบัส

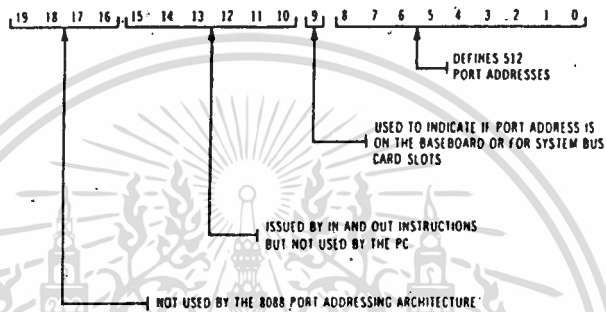
5) AEN (Address Enable)

สัญญาณนี้เป็นสัญญาณที่แอกทีฟ High ที่ถูกขับออกมาโดย 8239-5 DMA Controller เพื่อหยุดการติดต่อระหว่าง 8088 ไมโครโปรเซสเซอร์กับหน่วยความจำและอินพุท-เอาท์พุทพอร์ทโดยจะถือว่าเป็นการเริ่มต้นของกระบวนการ Direct Memory Access

3.1.6.1 การออกแบบ I/O Port Decoder

พอร์ทต่างๆที่ใช้ใน IBM Pc จะประกอบด้วย 2 ส่วนใหญ่ๆ คือ พอร์ทที่ใช้ออยู่บน Mainboard ซึ่งมีหมายเลขพอร์ทตั้งแต่ 0000H-01FFH จำนวน 512 พอร์ท และ พอร์ทที่อยู่บน Card Slot ซึ่งมีหมายเลขพอร์ทตั้งแต่ 0200H-03FFH อีกจำนวน 512 พอร์ท

หมายเลขของ I/O Port Address บน IBM Pc มีลักษณะดังรูป 3.8

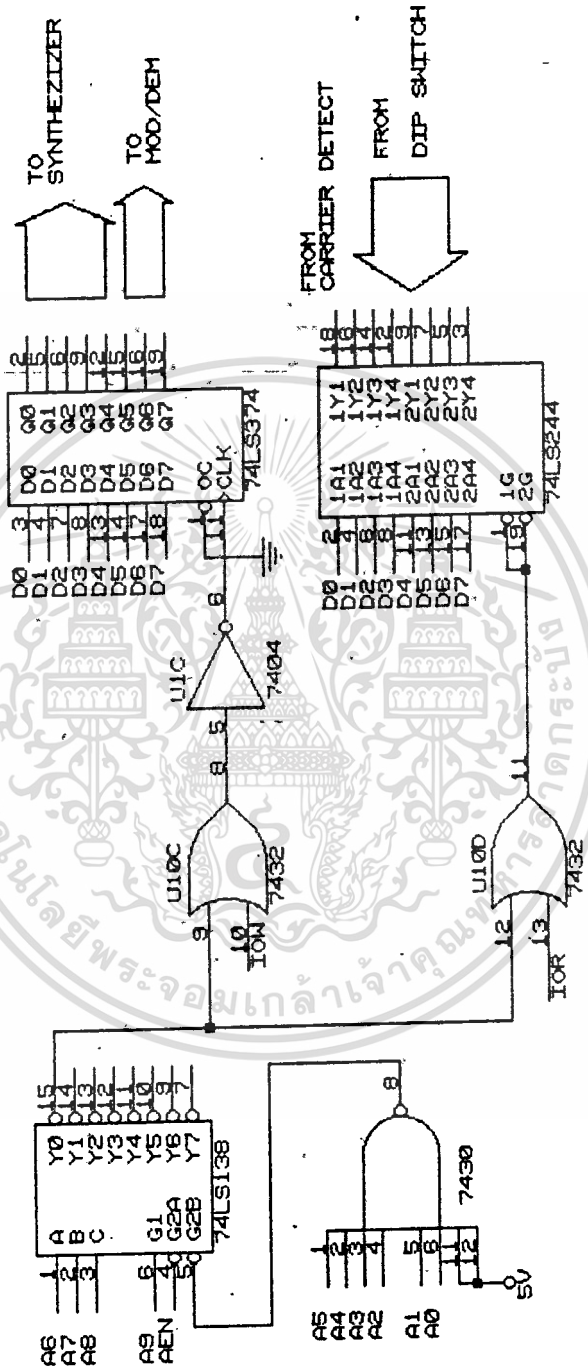


รูปที่ 3.8 อินพุท-เอาต์พุทพอร์ท แอดเดรสบน IBM Pc

จากรูป 3.8 จะเห็นว่าแอดเดรสที่ใช้ในการ Decode พอร์ทคือ $A_0 - A_{16}$ เท่านั้นโดย A_{17} จะเป็นขาที่บอกว่าพอร์ทที่ใช้งานอยู่บน Mainboard หรืออยู่บน Card Slot โดยที่ถ้า A_{17} เป็นลอจิก "1" จะหมายถึงตำแหน่งพอร์ทที่อยู่บน Card Slot และถ้าเป็นลอจิก "0" จะหมายถึงตำแหน่งพอร์ทซึ่งอยู่บน Mainboard

การเลือกหมายเลขพอร์ทที่จะใช้บน Card Slot สามารถดูได้จาก รูป 3.9 ซึ่งจะเห็นว่าหมายเลขพอร์ทจำนวน 512 พอร์ทไม่ได้ถูกใช้งานหมด การที่จะเลือกว่าเราจะใช้หมายเลขพอร์ทใดจะต้องพิจารณาตำแหน่งของพอร์ทที่ว่างรวมทั้งตรวจสอบ Card ที่เสียบบน Slot ต่างๆที่ใช้งานจริงว่าหมายเลขใดถูกใช้ไปบ้าง แล้วจึงเลือกหมายเลขพอร์ทที่ว่างมาใช้งาน ซึ่งในโครงการนี้ได้เลือกหมายเลขพอร์ทตำแหน่งที่ 023FH มาใช้

เมื่อพิจารณาจากหมายเลขพอร์ทที่ใช้หมายเลข 023FH กับ I/O Port Read & Write Cycle จะได้ว่าวงจร I/O Port Decoder หมายเลข 023FH ดังรูป 3.10 IC 74LS374 จะทำหน้าที่ Latch Data- ($D_0 - D_7$) ให้ค้างค่าไว้และ IC 74LS244 จะทำหน้าที่เป็นบัฟเฟอร์ในการปล่อยค่าเข้าสู่ดาต้าบัส



รูปที่ 3.10 วงจรภาพ I/O port decoder

3.2 การออกแบบเครื่องส่งวิทยุ

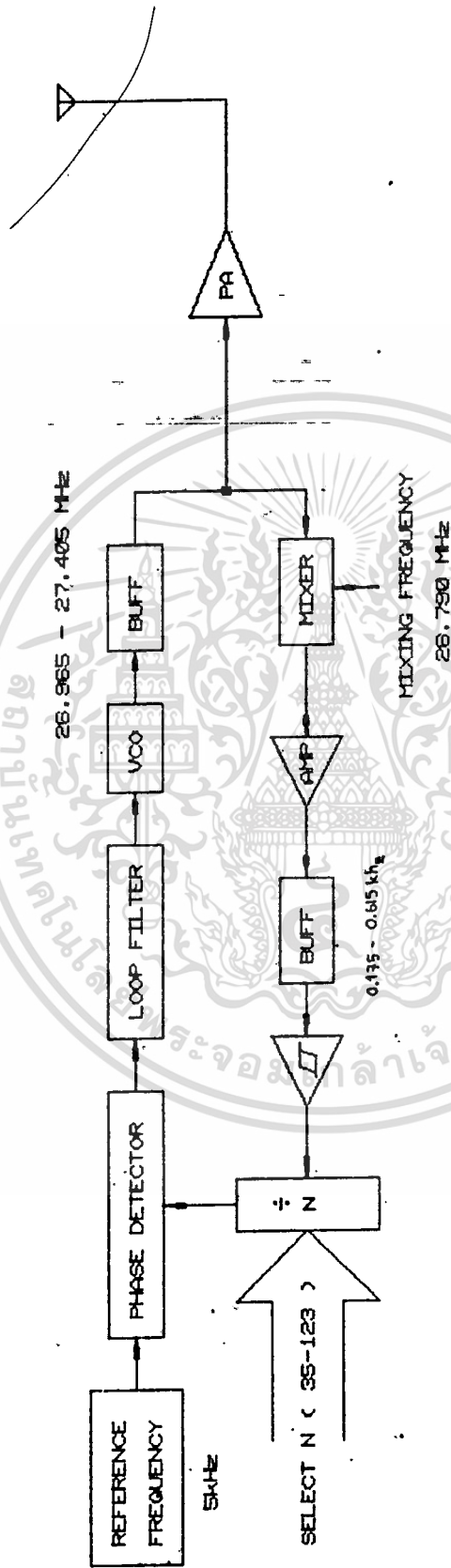
ในการออกแบบเครื่องส่งวิทยุจะแบ่งการออกแบบออกเป็น 2 ส่วน คือ

- 1) การออกแบบวงจรส่งเคราะห์ความถี่ ในช่วงความถี่ citizen band class D ความถี่ 26.965 - 27.405 Mhz จำนวน 40 ช่องสัญญาณ
- 2) การออกแบบวงจรขยายกำลัง (Power Amplifier)

3.2.1 การออกแบบวงจรส่งเคราะห์ความถี่

ในโครงงานนี้ได้เลือกใช้วิธีส่งเคราะห์ความถี่แบบ mixing technics วิธีนี้ แตกต่างจากวิธี PLL แบบหาร N เล็กน้อย ตรงที่ vco จะผ่านการ mix กับ สัญญาณจาก osc ก่อนที่จะป้อนให้วงจรหาร N ความถี่ของ vco ในสภาวะรับและส่งอาจจะไม่เท่ากันก็ได้ (เพราะว่าจะอาจจะใช้วิธีการเลื่อนความถี่ให้ห่างกันเท่ากับ ความถี่ IF) block diagram ที่สมบูรณ์และที่ใช้ในโครงงานจริงๆ ดังแสดงในรูปที่ 3.11

วงจรจะถูกออกแบบให้ใช้ crystal กำเนิดความถี่หลัก เพียงตัวเดียวที่ ความถี่ 10.24 Mhz แล้วเข้าวงจรหารด้วยค่า 2¹¹ ได้ความถี่ reference 5 khz ความถี่ reference กับ output ของ vco ที่ผ่านวงจรหารมาแล้วจะเทียบ phase กัน ผลลัพธ์ที่ได้จะส่งไปยัง loop filter เพื่อควบคุมการ oscillate ของ vco ซึ่ง vco ตัวนี้ จะออกแบบให้ LOCK ความถี่ในสภาวะส่งที่ความถี่ 26.965 - 27.405 Mhz ด้วยการกำหนดค่า N ที่เหมาะสมในวงจรหาร N จะสังเกตว่า vco จะ mix ใน loop กับ PLL osc (ความถี่ 26.790 Mhz ซึ่งได้จากวงจร osc ที่สร้างขึ้นเพื่อการ mix down ความถี่ลงมา) ความถี่ที่ผ่านการ mix จะอยู่ในย่าน 0.175 - 0.675 Mhz บางทีก็เรียกเป็นความถี่ IF ของ PLL (PLL-IF) จากนั้นป้อนความถี่ IF ของ PLL เข้าวงจรหาร N ซึ่งมีค่าเลือกได้จาก 0 - 511 ผ่านทาง programe ก่อนที่จะส่งเข้าวงจร PHASE DETECTOR การลดความถี่ vco ลงเป็นความถี่ PLL-IF ก็เพื่อให้วงจรหาร N ทำงานได้ที่ความถี่ต่ำลง และสามารถกำหนดค่า N เพื่อควบคุม vco ให้ กำเนิดความถี่ได้ตรงช่องต่าง ๆ ที่กำหนดไว้ตลอดย่าน 27 Mhz ทั้ง 40 ช่อง หลังจาก สัญญาณผ่านวงจรหาร N แล้ว จะถูกส่งต่อไปเทียบ PHASE กับความถี่ REFERENCE เพื่อหาผลต่างของสัญญาณความถี่คลาดเคลื่อน แล้วผ่านวงจร LOOP FILTER เพื่อเปลี่ยนเป็นแรงดัน ไปควบคุมการ osc ของ vco ต่อไป

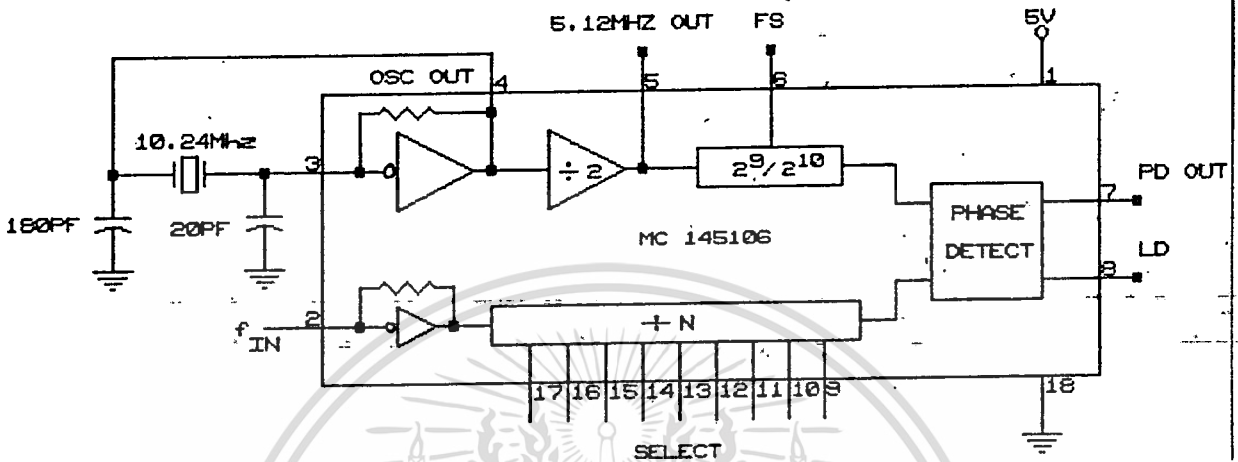


รูปที่ 3.11 แสดง BLOCK DIAGRAM ภาคส่งโครงข่ายความถี่

KINGMONKUT INSTITUTE OF TECHNOLOGY LACKRABANG	
Title MAIN BLOCK DIAGRAM	
Size A	Document Number REV BLK
Date: February 9, 1990 Sheet 1 of 1	

วงจรต่าง ๆ ที่ใช้ในภาคสังเคราะห์ความถี่

3.2.1.1 Phase lock loop block



รูปที่ 3.12 แสดง block ต่างๆ ภายในตัว MC 145106

ใช้ตัว IC เบอร์ MC 145106 ซึ่งประกอบด้วยวงจร phase detector, programmable divider, และ prescaler divider เป็นต้น c_1 และ c_2 ใช้เพื่อปรับความถี่ของ crystal ให้ตรงกับ resonance ที่ 10.24 Mhz และ logic ที่ขา 6 จะให้เป็น low เพื่อเลือกความถี่ reference เป็น 5 khz input ของวงจรหาร N ป้อนเข้าที่ขา 2 โดยค่าของ N จะ program ผ่านทาง output port ของ computer ส่วน output ของวงจร phase detector จะออกที่ขา 7 (ข้อควรระวังคือ ถ้า output lock detector ขา 8 ออกมาเป็นสถานะ high แสดงว่า loop อยู่ในภาวะ lock แต่ถ้าเป็น low แล้ว loop จะไม่อยู่ในภาวะ lock นั่นคือ วงจร PLL BLOCK นี้จะใช้งานไม่ได้)

3.2.1.2 วงจร Loop Filter

ใช้วงจรแบบ PASSIVE RC FILTER กรองเอาสัญญาณความถี่ต่ำ ซึ่งเราสามารถกำหนดคุณสมบัติของ loop filter ได้ดังนี้

$$K_{vco} = \frac{F_{vco}}{V_{vco}} \text{ rad/s/volt} \quad (3.8)$$

$$K_{\phi} = V_{cc} / 4\pi \quad (3.9)$$

$$K = K_{vco} K_{\phi} / N \quad (3.10)$$

$$\omega_n^2 = K / (T_1 + T_2) \quad (3.11)$$

$$\alpha = \omega_n [T_2 + 1/K] \quad (3.12)$$

โดยที่ $T_1 = R_1 C$

$$T_2 = R_2 C$$

$$N = \text{ค่าตัวหารที่มากที่สุด}$$



รูปที่ 3.13 แสดงกราฟความสัมพันธ์ระหว่าง V_{in} และความถี่ vco จากความสัมพันธ์ระหว่าง V_{in} และความถี่ vco ดังผลการทดลองในบทที่ 4 และจากความสัมพันธ์ตามสมการ (3.8) - (3.12) จะหาค่า R_1, R_2, C ได้ดังนี้

$$\begin{aligned} K_{vco} &= 2\pi [27.56 - 26.95] / 0.4 \\ &= 9.58 \times 10^6 \quad \text{rad/s/volt} \end{aligned}$$

$$K_{\phi} = 5 / 4\pi = 0.3978$$

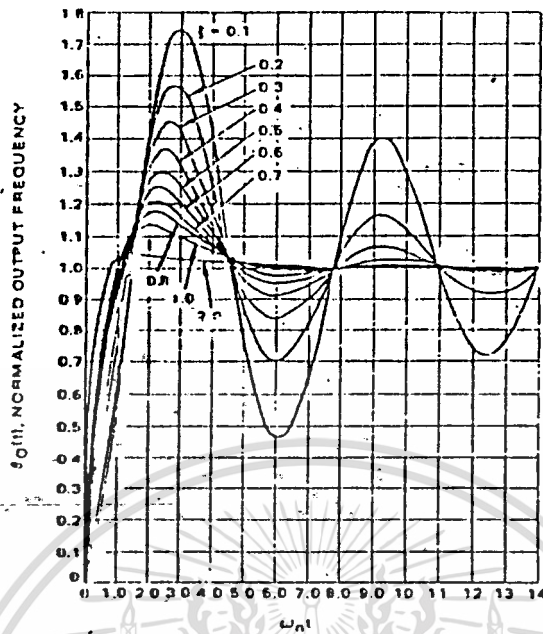
$$N = 123$$

$$K = 0.3978 \times 9.56 \times 10^6 / 123$$

$$= 30983.12$$

กำหนดค่า damping factor ที่ทำให้ระบบเข้าสู่สภาวะได้เร็วที่สุด คือค่า damping $\alpha = 0.8$ ซึ่งให้ค่า ω_n ประมาณ $4.5 \text{ Hz} \cdot \text{sec}$ โดยหาจากความสัมพันธ์ในเอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

รูปที่ 3.14



รูปที่ 3.14 แสดงความสัมพันธ์ระหว่าง $\omega_n t$ กับ normalized output frequency

โดยที่ $\omega_n t =$ ความถี่ cutoff

$t =$ setting time

เลือกค่า $t = 10$ ms

ฉะนั้น $\omega_n = 450$

จากสมการ (3.11) แทนค่า ω_n และ K จะได้

$$T_1 + T_2 = 0.153$$

จากสมการ (3.11) แทนค่า ω_n , α , และ K จะได้

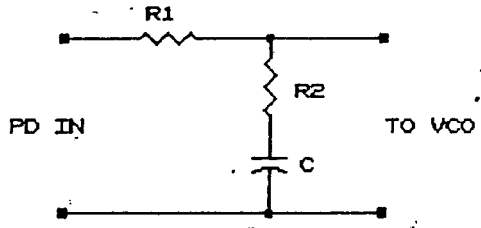
$$T_2 = 3.52 \times 10^{-3}$$

เพราะฉะนั้น $T_1 = 0.1494$

เลือกค่า $C = 2.2 \mu F$

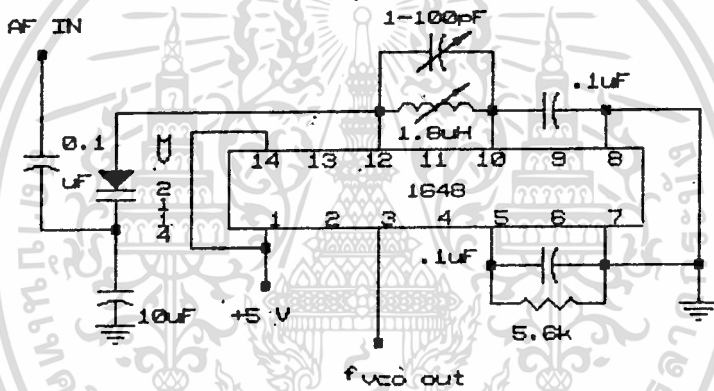
จะได้ค่า $R_1 = 0.1494 / 2.2 \mu F = 68 k$

$R_2 = 3.52 \times 10^{-3} / 2.2 \mu F = 1.6 k$



รูปที่ 3.15 แสดงวงจร loop filter ที่ใช้

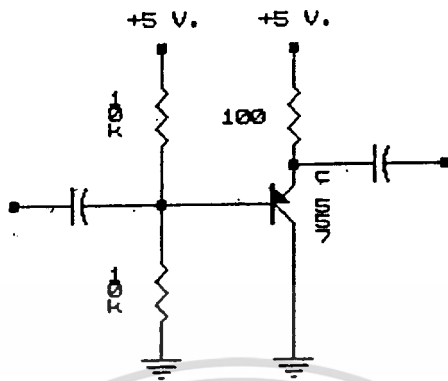
3.2.1.3 Voltage Controlled Oscillator



รูปที่ 3.16 แสดง วงจรของ vco

ใช้ IC เบอร์ MC 1648 เป็นวงจร osc แบบ ecl output ประมาณ 400 mV แต่คุณสมบัติทางด้าน noise อาจสู้แบบ FET ไม่ได้ ความถี่ของ vco ถูกควบคุมด้วยแรงดันจาก loop filter ที่จะมา bias ตัว varactor diode ในวงจร tune ถ้าแรงดัน bias เพิ่มขึ้น vco จะมีความถี่ output สูงขึ้น (ดูจากกราฟ 4.15 ความสัมพันธ์ระหว่าง V_{in} กับความถี่ของ vco) varactor diode ในวงจรใช้เบอร์ MV 2114 ซึ่งมีค่า nominal capacitance = 82 pF ที่ 4 V และที่ 4-60 V ค่า capacitance ratio เป็น 2.6 วงจร resonance จะ tune ไว้ที่ 27 Mhz โดยพัน L = 1.8 μ H และใช้ trimmer capacitor ในการจูนละเอียดที่ 27 Mhz

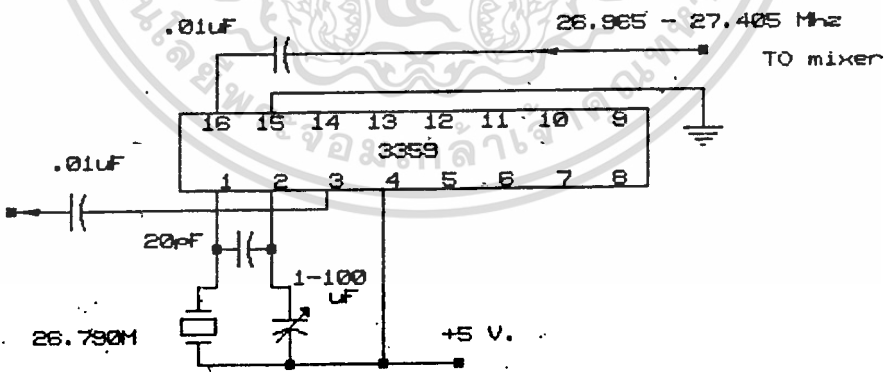
3.2.1.4 Buffer



รูปที่ 3.17 แสดงวงจร buffer

ใช้ transistor PNP เบอร์ C 557 ต่อเป็นวงจร common emitter เพื่อใช้เป็น buffer และ driver ป้องกันไม่ให้วงจรในภาค mixer และ power amp. มา load วงจร vco โดยตรง จึงต้องใช้วงจร buffer คั่นกลางไว้

3.2.1.5 วงจร Mixer and Mixing Oscillator



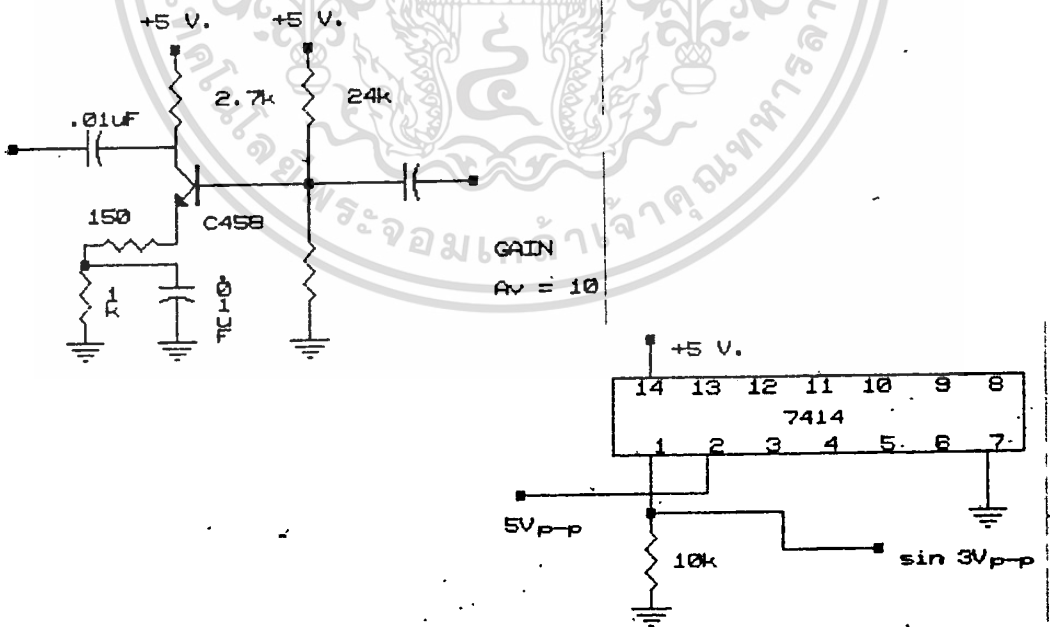
รูปที่ 3.18 แสดงวงจรการต่อภาค mixer ของตัว MC 3359

เพื่อให้วงจรมีขนาดเล็ก ประสิทธิภาพดี โดยที่จะต้องมิตค่า high order intermodulation ต่ำ ในขณะที่มี gain และ AM rejection สูง จึงเลือกใช้ เฉพาะภาค mixer และ oscillator ภายใน IC เบอร์ MC 3359 ซึ่งออกแบบไว้

สำหรับเป็น second IF down conversion. โดยที่ในวงจร เราต้องการให้ IF อยู่ในย่านความถี่ 0.175 - 0.675 Mhz ซึ่งมีความถี่ input ที่ขา 18 เป็น 26.965 - 27.405 Mhz จึงต้องใช้ความถี่ 26.790 Mhz มา mix รวมกัน เนื่องจากวงจร osc มีอยู่ในตัว IC เพียงแต่เพิ่ม crystal 26.790 Mhz และใส่ trimmer capacitor เข้าไปเพื่อปรับความถี่ให้ตรง ก็จะได้ output ที่ขา 3 ในย่าน 0.175 - 0.675 Mhz ตามต้องการ

3.2.1.6 วงจร IF Amp. และ วงจร schmitt trigger

เนื่องจาก output จาก mixer เป็นสัญญาณขนาดเล็ก ไม่สามารถป้อนเข้าสู่วงจรหารได้โดยตรง จึงต้องทำการขยายสัญญาณให้มีขนาดใหญ่ขึ้น โดยผ่านวงจรขยายสัญญาณ ให้ได้ขนาดประมาณ 3V ในการป้อนเป็น input ให้กับ inverter ภายใน IC 7414 ให้ทำหน้าที่เป็น schmitt trigger โดยเลือกค่า R ที่เหมาะสม เพื่อปรับค่า duty cycle ให้ได้ 50% ก่อนจะป้อนเข้าวงจรหาร N ภายใน IC MC 145106 ที่ขา 2 ต่อไป



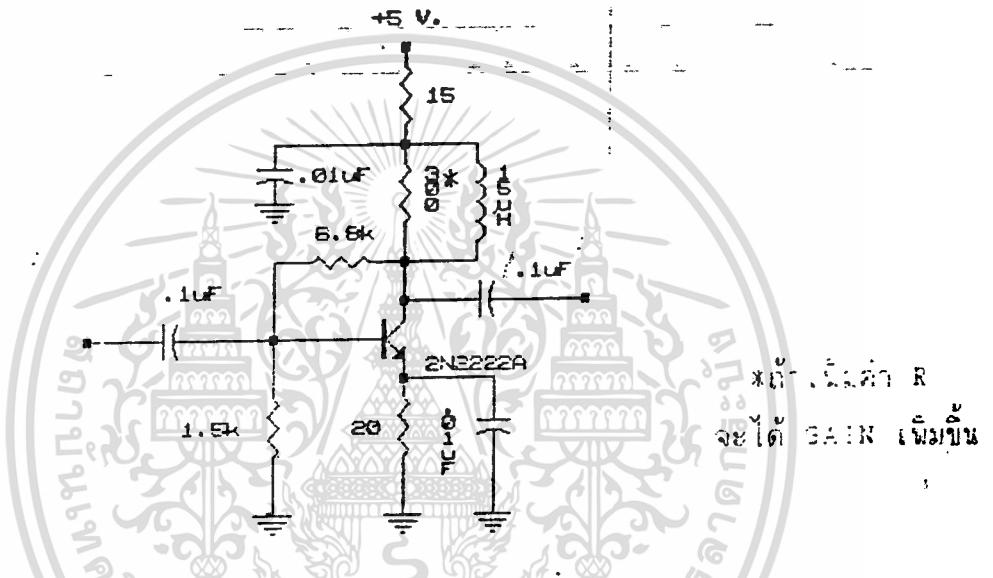
รูปที่ 3.19 แสดงวงจร IF amp และวงจร schmitt trigger

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานที่ ๒๓๗-ศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.2.1.7 วงจร N - divider

จะใช้วงจรภายในตัว MC 145106 ซึ่งสามารถ program ค่าได้ตั้งแต่ 0 - 511 กำหนดในรูปของ binary บิตทางขา 9 - 17 โดยขา 9 เป็นขา MSB และขา 17 เป็นขา LSB โดยที่ในวงจรสามารถ program ผ่านทาง output ของ computer

3.2.2 วงจร Power Amplifier



รูปที่ 3.20 แสดงวงจรมายกำลัง

เนื่องจากลักษณะความถี่ที่ vco ให้ออกมานั้น ยังมีขนาดเล็ก คือวัดได้เพียง 300mV เท่านั้น กำลังส่งออกอากาศก็มีค่าที่น้อยมาก ส่งออกไปได้ไกล ๆ เท่านั้น จึงต้องใช้วิธีเพิ่มวงจรมายกำลังให้มากขึ้น แต่เนื่องจากแหล่งจ่ายไฟที่ใช้เฉพาะไฟ + 5VDC กระแส 1 A เท่านั้น การออกแบบวงจรเพิ่มกำลังส่งสูง ๆ จึงทำได้จำกัด ในวงจรใช้งานจริงจะใช้ transistor ต่อเป็นวงจร common emitter โดยตั้งอัตราขยายของวงจรผ่านทางตัวต้านทานที่ขา collector ถ้าค่าตัวต้านทานมาก ก็จะได้ gain สูงขึ้น ในวงจรจะตั้ง gain เอาไว้ ประมาณ 15 dB โดยทำเป็น 2 วงจรนำมาต่อ cascade กัน จะได้ gain รวมประมาณ 30 dB

3.3 การออกแบบโปรแกรมการสื่อสารข้อมูล

การสื่อสารข้อมูลระหว่าง Personal Computer ในโครงงานนี้จะเป็นแบบอะซิงโครนัส (Asynchronous) โดย ผ่านการอินเตอร์เฟสตามมาตรฐาน EIA RS-232C อุปกรณ์ที่จะทำหน้าที่ในการแปลงข้อมูลบนดาต้าบัส ให้เป็นไปตามมาตรฐานดังกล่าวคือ ตัวอะแดปเตอร์สื่อสารข้อมูลอะซิงโครนัส (Asynchronous Communication Adapter)

หัวใจในการทำงานของอะแดปเตอร์ตัวนี้คือ IC INS 8250 LSI ซึ่งการที่จะใช้งานไอซีตัวนี้จะต้องทำการโปรแกรมรีจิสเตอร์ต่างๆบน 8250 โดยผ่านทาง อินพุต-เอาต์พุตพอร์ท ตำแหน่งที่ 03F8H ถึง 03FFH สำหรับอะแดปเตอร์ตัวที่ 1 และทาง อินพุต-เอาต์พุตพอร์ท ตำแหน่งที่ 02F8H ถึง 02FFH สำหรับอะแดปเตอร์ตัวที่ 2

การโปรแกรมดังกล่าว ก็เพื่อที่จะเห็นพารามิเตอร์ต่างๆในการสื่อสารข้อมูลแบบอะซิงโครนัส ได้แก่ บิตเริ่มต้น, บิตสิ้นสุด, บิตพาริตี, อัตราเร็วในการสื่อสาร, จำนวนบิตใน 1 คาแรคเตอร์ รวมทั้งการโปรแกรมควบคุมลำดับของสัญญาณ Handshake ต่างๆบน RS-232C อีกด้วย ซึ่งรายละเอียดของการทำงานของอะแดปเตอร์ 8250 สามารถหาอ่านได้จากหนังสือ IBM Technical Reference

สำหรับภาษาที่ใช้ในการเขียนโปรแกรมใช้ ล็อกใช้ภาษา Assembly 8088 ซึ่งสามารถควบคุมฮาร์ดแวร์ได้ดี โดยจะแบ่งการเขียนโปรแกรมออกเป็น 2 ส่วนใหญ่ๆ คือ

1. โปรแกรมที่ใช้ในการส่งข้อมูล
2. โปรแกรมที่ใช้ในการรับข้อมูล

3.3.1 โปรแกรมการส่งข้อมูล

เพื่อให้สะดวกในการใช้งานสำหรับผู้ใช้โดยทั่วไป จึงได้ออกแบบเมนู (Menu) ต่างๆขึ้นเพื่อให้ผู้ใช้ได้เลือกพารามิเตอร์ต่างๆโดยสะดวก คือ

Select Transmission Frequency Menu จะเป็นเมนูที่ให้ผู้ใช้เลือกความถี่คลื่นพาห้ที่จะใช้ในการส่งข้อมูล โดยความถี่ต่างๆจะมีให้เลือก 40 ความถี่ (ตั้งแต่ 01-40) ความถี่จะอยู่ในย่าน Citizen Band Class D (26.965-27.405 MHz)

การเขียนโปรแกรมเมนูทั้ง 2 เมนูจะเป็นโปรซีเจอร์ (Procedure) ซึ่งจะถูกเรียกใช้โดยโปรแกรมหลัก (Main Program) การใส่ค่าพารามิเตอร์ต่างๆในเมนู ถ้าผิดโปรแกรมหลักจะเรียก Error Message Display Procedure เพื่อแสดงต่อผู้ใช้ให้ใส่ค่าพารามิเตอร์ใหม่

นอกจากนี้ในส่วนของโปรแกรมหลักจะทำหน้าที่ในการเช็คพารามิเตอร์ การสื่อสารแบบอะซิงโครนัสให้มีค่าต่างๆ ดังนี้

1. บิตล้นสุด = 1 บิต
2. จำนวนบิตต่ออักขระ = 8 บิต
3. อัตราเร็วในการสื่อสาร = 600 บอด
4. ให้มีการตรวจสอบสัญญาณ Handshake ระหว่างสัญญาณ RTS และ CTS

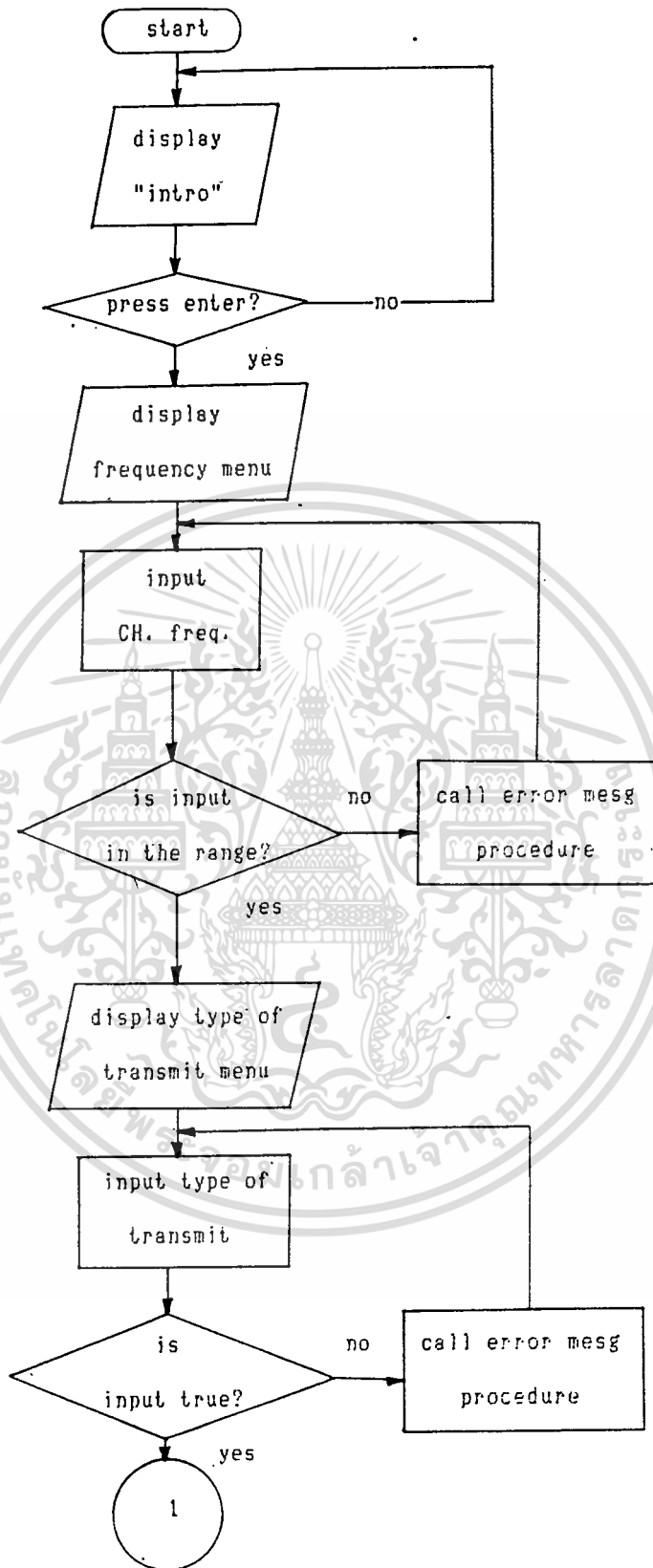
รายละเอียดของโฟลว์ชาร์ต (Flowchart) แสดงได้ดังรูป 3.21 ส่วนตัวโปรแกรมสามารถดูได้จากภาคผนวก ก.

3.3.2 โปรแกรมการรับข้อมูล

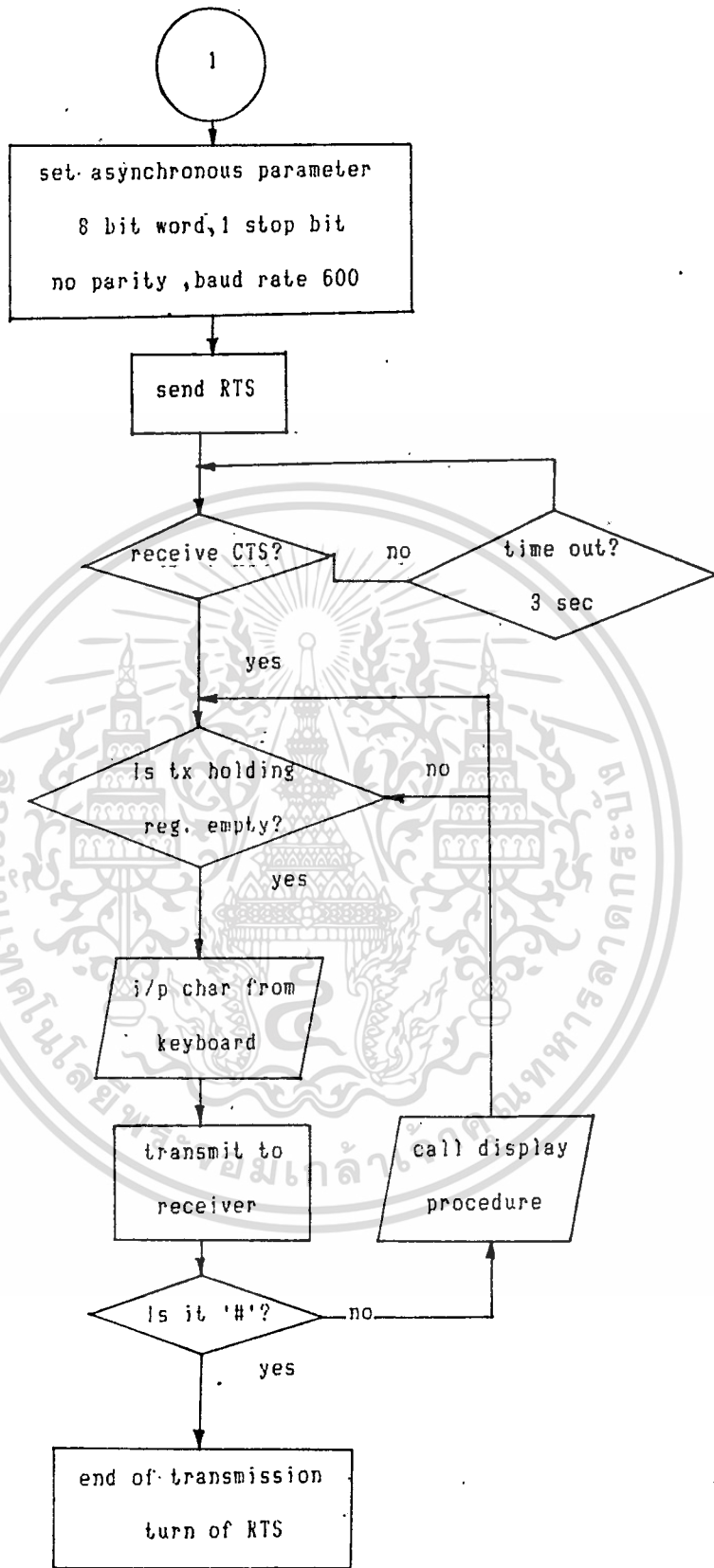
หลักสำคัญในการเขียนการรับข้อมูล แบบอะซิงโครนัสโดยผ่าน RS-232C พอร์ต คือ การเช็คค่าพารามิเตอร์ในการรับข้อมูลแบบอะซิงโครนัสให้ตรงกับพารามิเตอร์ที่ใช้ในการส่งข้อมูลแบบอะซิงโครนัสเสียก่อน พารามิเตอร์ที่ใช้ก็คือ

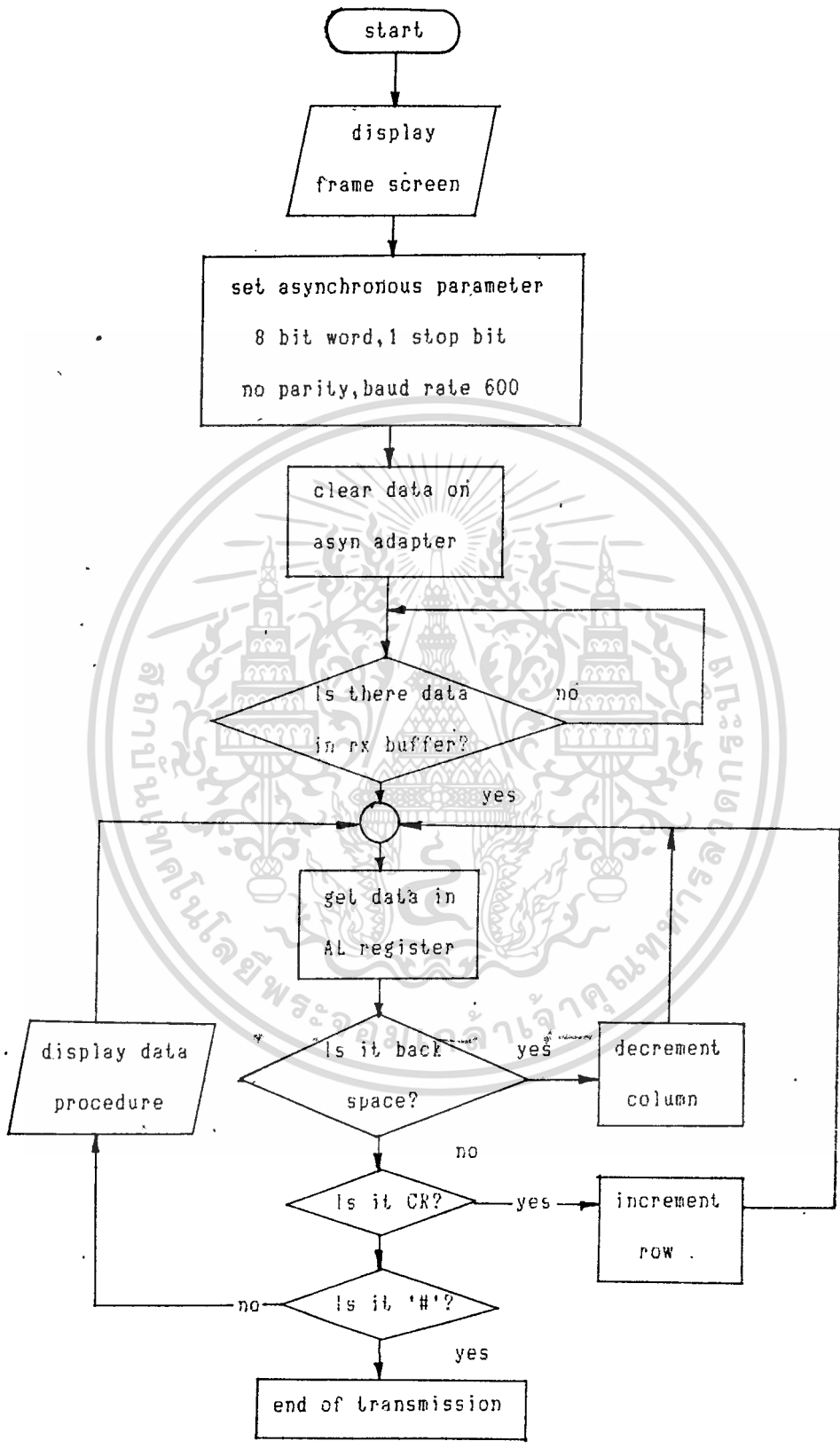
1. บิตล้นสุด = 1
2. จำนวนบิตต่ออักขระ = 8 บิต
3. อัตราเร็วในการสื่อสาร = 600 บอด

หลังจากเช็คค่าพารามิเตอร์ต่างๆ เรียบร้อย จึงทำการตรวจบัฟเฟอร์ข้อมูลรับ ถ้ามีข้อมูลส่งเข้ามาก็นำขึ้นแสดงบนจอภาพ รายละเอียดโฟลว์ชาร์ตของโปรแกรมการรับข้อมูลแสดงได้ดังรูป 3.22 ส่วนตัวโปรแกรมสามารถดูได้จาก ภาคผนวก ก.



รูปที่ 3.21 ฟลิวชาร์ตโปรแกรมการส่งข้อมูล





รูปที่ 3.22 โฟล์ชาร์ตโปรแกรมการรับข้อมูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการเรียนเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

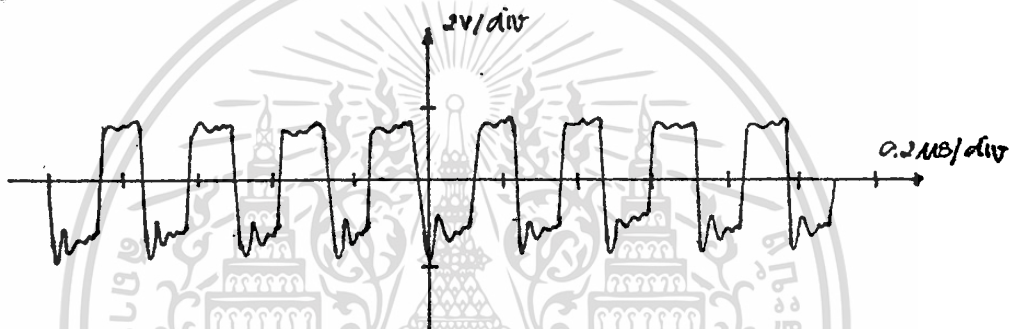
บทที่ 4

การทดลองและผลการทดลอง

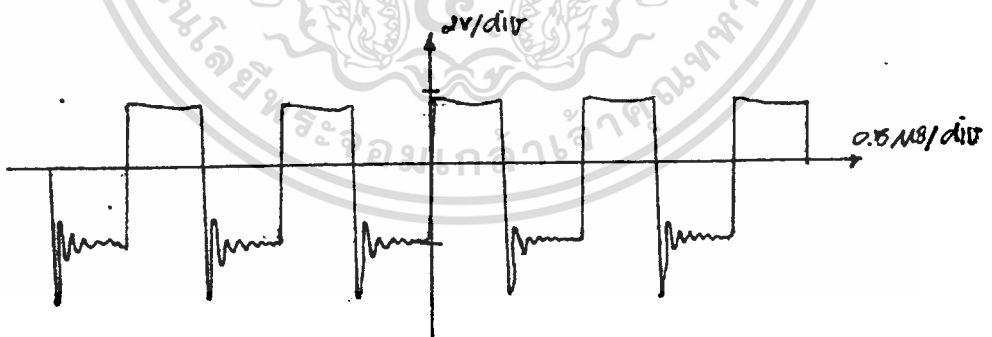
4.1 การทดลองวัดค่าต่างๆของวงจรโมเด็ม 600 ขอด

4.1.1 การวัดความถี่ของวงจรออสซิลเลเตอร์

ความถี่ของวงจรออสซิลเลเตอร์ได้จากการนำเอา คริสตอลออสซิลเลเตอร์ ความถี่ 4 MHz แล้วมาทำการหาร 4 ด้วย D ฟลิป-ฟลอป 2 ตัว เพื่อให้ได้ความถี่ 1 MHz รูปสัญญาณของสัญญาณที่ได้จากวงจรคริสตอลออสซิลเลเตอร์ และรูปสัญญาณที่ได้หลังจากทำการหารด้วย 4 แล้วเป็นดังรูป 4.1 และ 4.2 ตามลำดับ



รูปที่ 4.1 รูปสัญญาณของวงจรคริสตอลออสซิลเลเตอร์ 4 MHz



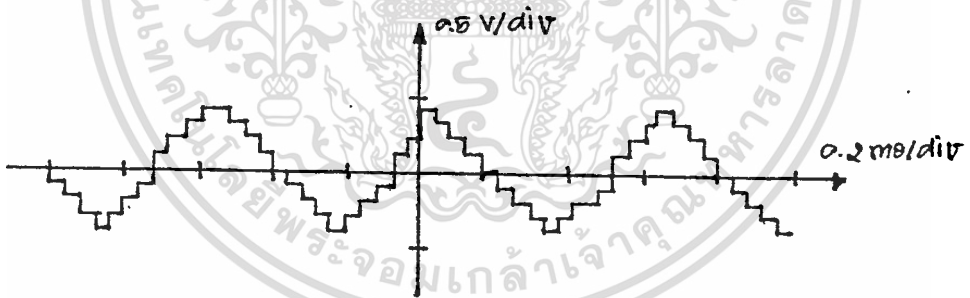
รูปที่ 4.2 รูปสัญญาณ 1 MHz ที่ได้จากการหารด้วย D ฟลิป-ฟลอป

4.1.2 การวัดความถี่และรูปสัญญาณของสัญญาณเสียงที่ได้จากวงจรมอดคูเลเตอร์
ความถี่จากการมอดคูเลทสัญญาณดิจิทัลเป็นไปตามตารางที่ 4.1

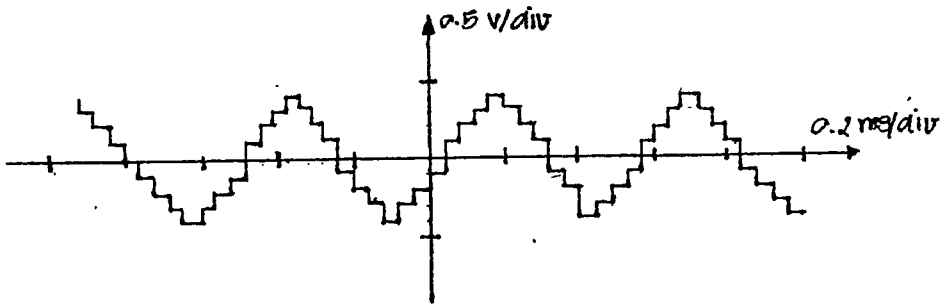
ตารางที่ 4.1 ความถี่จากการมอดคูเลทสัญญาณดิจิทัล

สัญญาณข้อมูล	ความถี่สัญญาณแอนะล็อก
1	1650
0	1850
ECHO	2100

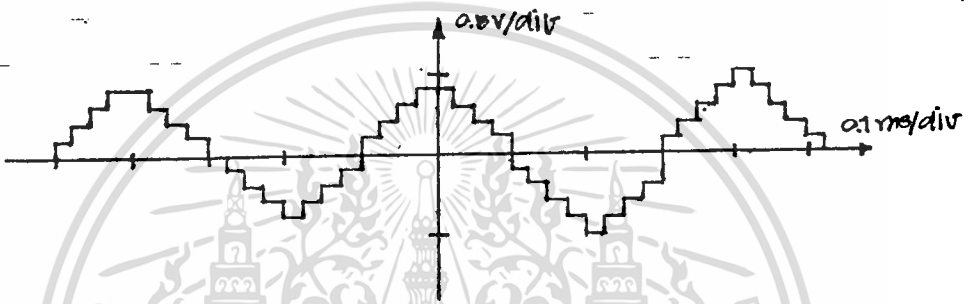
รูปสัญญาณที่ได้จากการมอดคูเลทจะได้จากการทำ DIA คอนเวอร์เตอร์จะเป็น
รูปขั้นบันได 8 บิต โดยมีระดับสัญญาณ 800 mV ดังรูป 4.3, 4.4, 4.5



รูปที่ 4.3 สัญญาณลอจิก "1" ที่ผ่านการมอดคูเลท



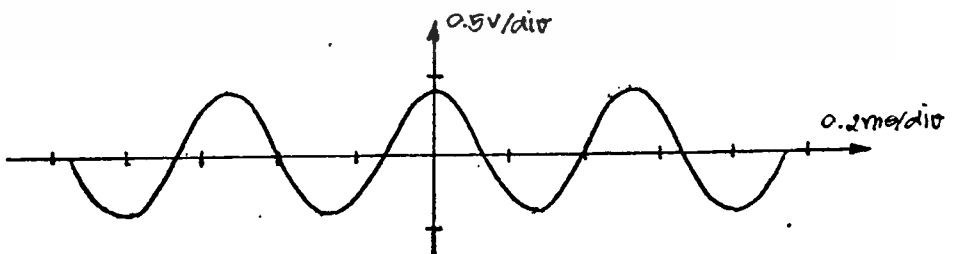
รูปที่ 4.4 สัญญาณลอจิก "0" ที่ผ่านการมอดูเลต



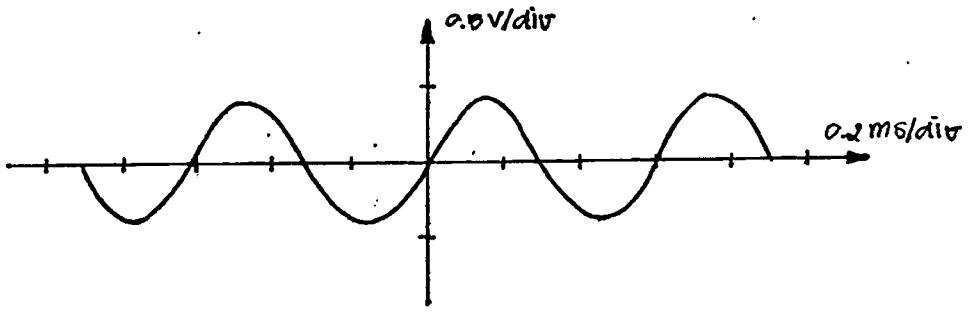
รูปที่ 4.5 สัญญาณ Echo ที่ผ่านการมอดูเลต

4.2.3 การวัดผลตอบสนองความถี่วงจรรองความถี่ต่ำ

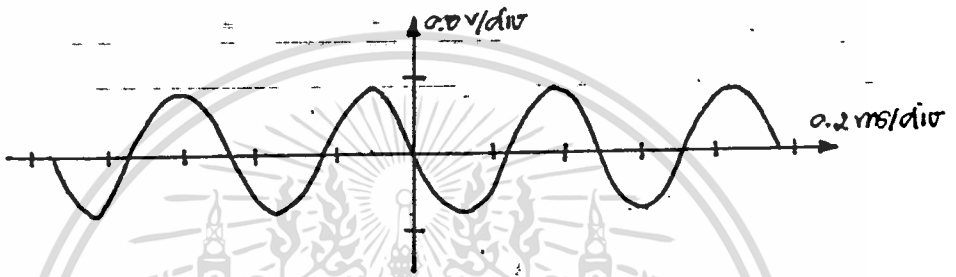
วงจรรวมความถี่ต่ำที่ใช้เป็นแบบ 2^{nd} บัต์เตอร์เวอร์ชันฟิลเตอร์ ทำหน้าที่จัดรูป
 เวกเตอร์ของเอาต์พุตภาคมอดูเลตให้เรียบ ดังรูป 4.6; 4.7, 4.8 โดยมีผลตอบ
 สอนงต่อสัญญาณความถี่ต่างๆตามตารางที่ 4.2



รูปที่ 4.6 สัญญาณความถี่ 1850 Hz เมื่อผ่านวงจรรองความถี่ต่ำ



รูปที่ 4.7 สัญญาณความถี่ 1650 Hz เมื่อผ่านวงจรกรองความถี่ต่ำ

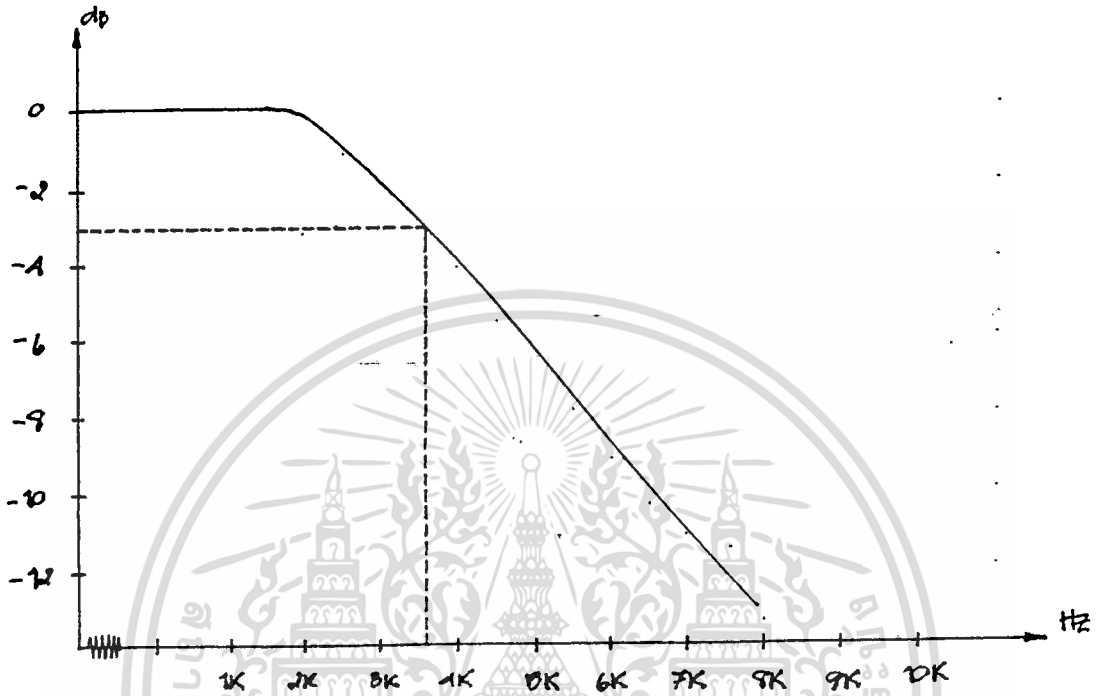


รูปที่ 4.8 สัญญาณ Echo เมื่อผ่านวงจรกรองความถี่ต่ำ

ตารางที่ 4.2 ผลตอบสนองต่อความถี่สัญญาณเมื่อผ่านวงจรกรองความถี่ต่ำ

f_{in} (kHz)	V_{out}	db	f_{in} (kHz)	V_{out}	db
1.0	2.00	0.0	5.5	0.8	-7.95
1.5	2.00	0.0	6.0	0.7	-9.12
2.0	1.90	-0.44	6.5	0.6	-10.5
2.5	1.75	-1.16	7.0	0.5	-11.2
3.0	1.60	-1.94	7.5	0.5	-12.1
3.5	1.40	-3.09	8.0	0.4	-13.5
4.0	1.25	-4.08	8.5	0.4	-14.0
4.5	1.05	-5.60	9.0	0.4	-15.2
5.0	0.95	-6.46	9.5	0.3	-15.8

จากตาราง 4.2 นำมาพล็อตผลตอบสนองต่อความถี่สัญญาณได้ดังรูป 4.9 โดยมี ความถี่คัทออฟอยู่ที่ 4000 Hz



รูปที่ 4.9 เส้นตอบสนองความถี่สัญญาณวงจรรองความถี่ต่ำ

4.1.4 การวัดผลตอบสนองต่อความถี่วงจรรวมแบนด์พาสฟิลเตอร์ 1650, 1850 Hz

เมื่อป้อนสัญญาณเข้าขั้วอินพุต ความถี่ต่างๆที่ระดับสัญญาณ 2 V_{pp} เป็นอินพุตของวงจรรวมแบนด์พาสฟิลเตอร์ และทำการวัดระดับสัญญาณเอาต์พุตของฟิลเตอร์ จะได้ผลตอบสนองความถี่ดังตารางที่ 4.3 และพล็อตหาเส้นตอบสนองความถี่ได้ดังรูป 4.10

4.1.5 การวัดผลตอบสนองต่อความถี่วงจรรวมแบนด์พาสฟิลเตอร์ 2100 Hz

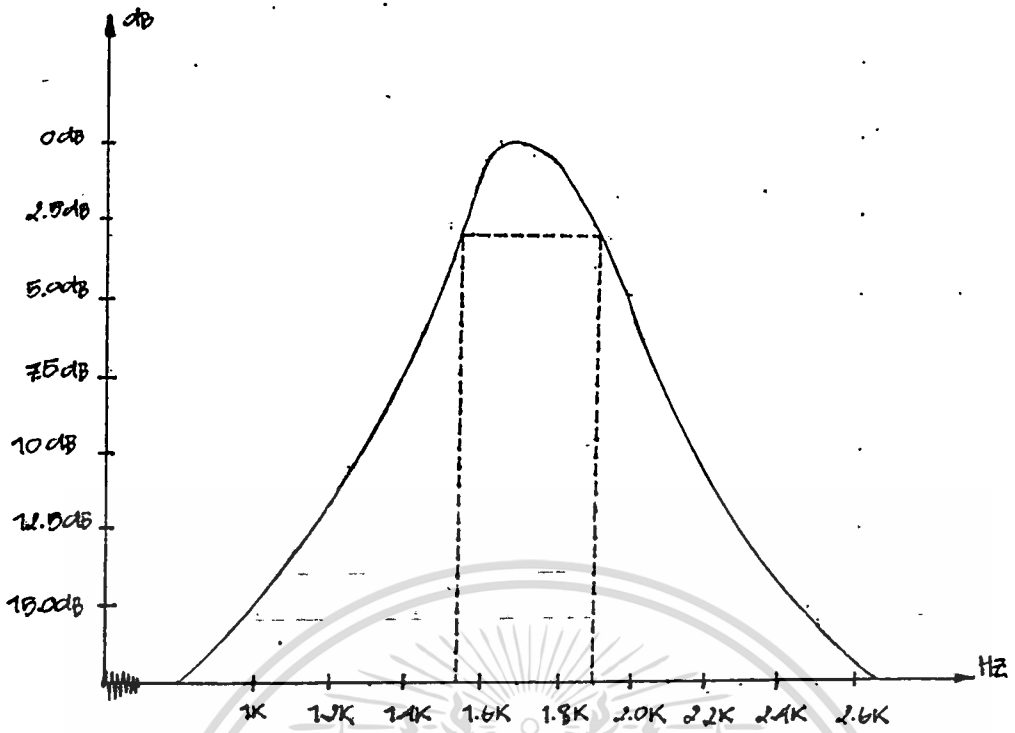
ผลตอบสนองต่อสัญญาณเข้าขั้วอินพุต 2 V_{pp} ที่ความถี่ต่างๆเป็นไปดังตารางที่ 4.4 และเส้นตอบสนองความถี่ของสัญญาณเป็นดังรูป 4.11

ตารางที่ 4.3 ผลตอบสนองความถี่สัญญาณเมื่อผ่านวงจรแบนด์พาสฟิลเตอร์ 1650, 1850 Hz

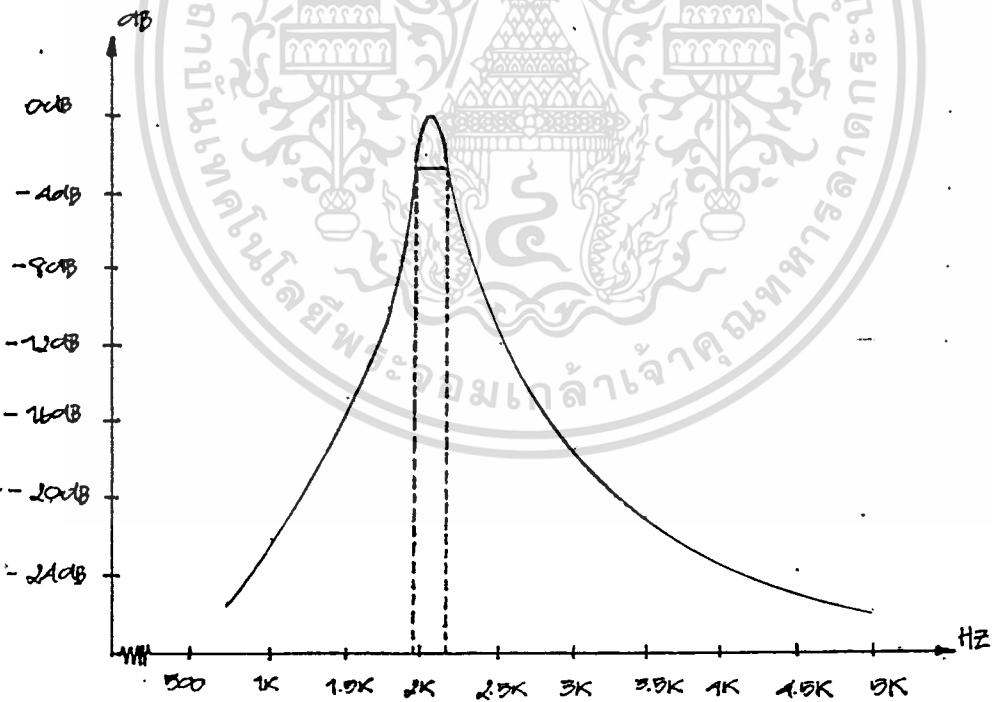
f_{in} (kHz)	V_o	db	db(normalize)
0.75	0.08	-27.95	-25.5
1.00	0.18	-20.91	-18.4
1.25	0.45	-12.95	-10.5
1.50	1.10	- 5.19	- 2.7
1.60	1.40	- 3.09	- 0.6
1.65	1.50	- 2.49	0.0
1.70	1.50	- 2.49	0.0
1.75	1.45	- 2.79	- 0.3
1.80	1.35	- 3.41	- 0.9
1.90	1.10	- 5.19	- 2.7
2.00	0.85	- 7.43	- 4.9
2.50	0.25	-18.06	-15.6
3.00	0.12	-24.43	-27.9

ตารางที่ 4.4 ผลตอบสนองความถี่สัญญาณเมื่อผ่านวงจรแบนด์พาสฟิลเตอร์ 2100 Hz

f_{in} (kHz)	V_o	db	db(normalize)
0.75	0.20	-20.00	-25.57
1.00	0.30	-16.47	-22.04
1.25	0.40	-13.98	-19.55
1.50	0.68	-9.43	-15.00
1.75	1.20	-4.44	-10.01
2.00	3.00	3.52	-2.05
2.10	3.80	5.57	0.00
2.25	1.90	-0.44	-6.01
2.50	0.90	-6.93	-12.50
3.00	0.50	-12.04	-17.51
3.50	0.36	-14.89	-20.46
4.00	0.28	-17.07	-22.64



รูปที่ 4.10 เส้นตอบสนองความถี่สัญญาณวงจรแบนด์พาสฟิลเตอร์ 1650, 1850 Hz



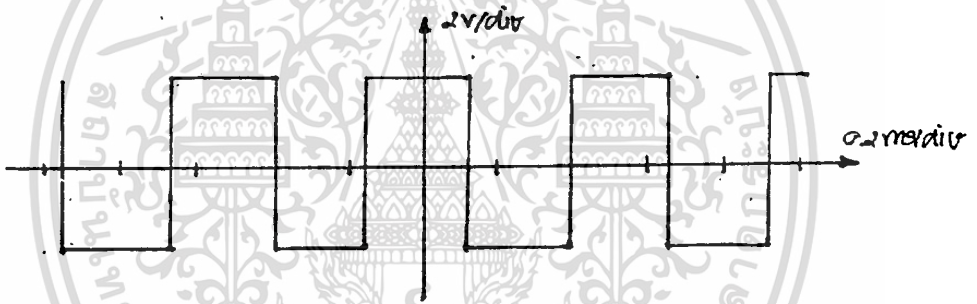
รูปที่ 4.11 เส้นตอบสนองความถี่สัญญาณวงจรแบนด์พาสฟิลเตอร์ 2100 Hz

4.1.6 การวัดความถี่และรูปสัญญาณจากวงจรติมอดดูละเตอร์

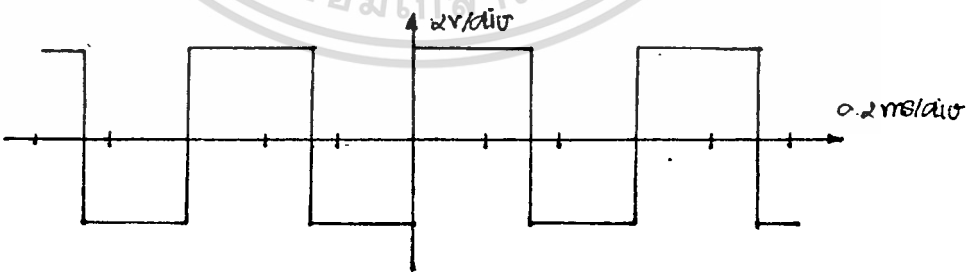
สัญญาณที่ใช้ในการติมอดดูละเตอร์จะเป็นสัญญาณแอสแควร์เวฟ โดยมี Duty Cycle 50% ดังรูป 4.12, 4.13 โดยความสัมพันธ์ระหว่างอินพุทและเอาต์พุทเป็นไปดังตารางที่ 4.5

ตารางที่ 4.5 สัญญาณที่ได้จากวงจรติมอดดูละเตอร์

อินพุท	เอาต์พุท
1850 Hz	0
1650 Hz	1



รูปที่ 4.12 สัญญาณ 1650 ที่ใช้ในการติมอดดูละเตอร์

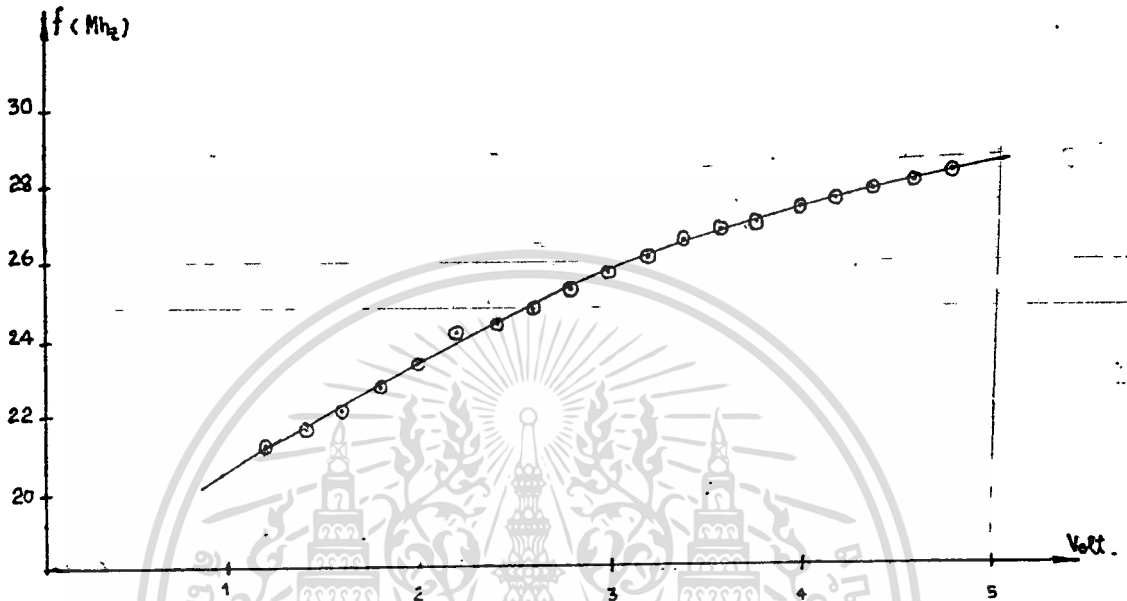


รูปที่ 4.13 สัญญาณ 1850 ที่ใช้ในการติมอดดูละเตอร์

4.2 การวัดคุณสมบัติต่าง ๆ ของวงจรเครื่องส่ง

4.2.1 ผลการทดลองวัดคุณสมบัติของวงจร vco

โดยการเปลี่ยนแปลงแรงดันควบคุม varactor diode แล้ววัด output ของ vco จะได้ผลดังตารางที่ 4.6 และเขียนกราฟได้ดังรูปที่ 4.14



รูปกราฟที่ 4.14 แสดงกราฟการทำงานของ vco ที่วัดได้

ตารางที่ 4.6 แสดงผลการวัดความถี่ของ vco เมื่อควบคุมแรงดันที่ varactor diode

Volt	Frequency (Mhz)	Volt	Frequency (Mhz)
1.2	21.22	3.2	25.90
1.4	21.64	3.4	26.28
1.6	21.96	3.6	26.60
1.8	22.58	3.8	26.95
2.0	23.22	4.0	27.27
2.2	24.00	4.2	27.56
2.4	24.12	4.4	27.82
2.6	24.60	4.6	28.14
2.8	25.08	4.8	28.42

4.2.2. ผลการทดลองความสัมพันธ์ระหว่างค่า N และ ความถี่ของ vco

จากวงจร PLL ที่ได้สร้างขึ้น จะหาค่า f_{vco} ได้จากสูตร

$$f_{vco} = (5 \times 10^3 \times N) + 26.790 \times 10^6$$

เมื่อทำการทดลองวัดค่าความถี่ vco โดยป้อนค่า N จะได้ผลดังนี้

ตารางที่ 4.7 แสดงความสัมพันธ์ระหว่างค่า N และความถี่ของ vco

ch	N	frequency (Mhz)	ch	N	frequency (Mhz)
1	35	26.9650	21	85	27.2149
2	37	26.9749	22	87	27.2250
3	39	26.9850	23	93	27.2550
4	43	27.0050	24	89	27.2350
5	45	27.0150	25	91	27.2449
6	47	27.0250	26	95	27.2650
7	49	27.0349	27	97	27.2750
8	53	27.0550	28	99	27.2850
9	55	27.0650	29	101	27.2950
10	57	27.0750	30	103	27.3050
11	59	27.0850	31	105	27.3150
12	63	27.1049	32	107	27.3249
13	65	27.1149	33	109	27.3350
14	67	27.1250	34	111	27.3450
15	69	27.1350	35	113	27.3550
16	73	27.1550	36	115	27.3650
17	75	27.1650	37	117	27.3749
18	77	27.1750	38	119	27.3850
19	79	27.1850	39	121	27.3950
20	83	27.2050	40	123	27.4050

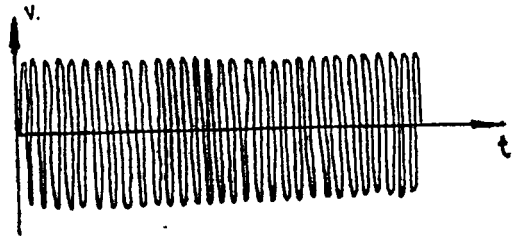
4.2.3 ผลการทดลองวัดสัญญาณภาคสังเคราะห์ความถี่โดยให้ค่า $N = 35$

1) เมื่อทดลองเลือก ch ที่ 1 โดยป้อนค่า

$$N = (00100011)_2 = (35)_{10}$$

$$\text{จะได้ความถี่ } v_{co} = 26.965 \text{ Mhz}$$

$$\text{ขนาดสัญญาณ} = 250 \text{ mV}$$



รูปที่ 4.15 แสดงสัญญาณ เมื่อ $N = 35$

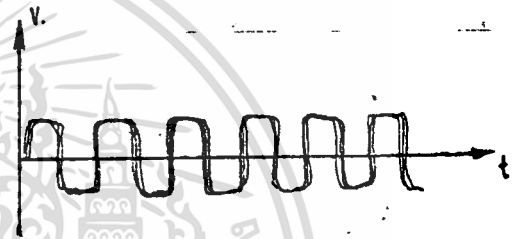
2) วัดสัญญาณ mixer output ที่

ขา 3 ของ IC MC 3359 จะได้

$$\text{ความถี่หลังผ่าน mixer} = 175 \text{ khz}$$

$$\text{ขนาดสัญญาณ} = 0.2 \text{ V}$$

$$\text{ขนาดสัญญาณเมื่อผ่านการขยาย} = 2.8 \text{ V}$$



รูปที่ 4.16 แสดงสัญญาณของ mixer output

3) เมื่อสัญญาณในข้อ 2) ที่ขยายแล้ว

ผ่านวงจร schmitt trigger 7414 จะได้

$$\text{สัญญาณรูปสี่เหลี่ยมขนาด} = 5 \text{ V}$$

$$\text{duty cycle} = 47 \%$$

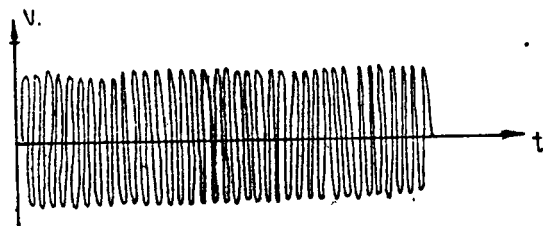


รูปที่ 4.17 แสดงสัญญาณเมื่อผ่าน schmitt trigger

4) สัญญาณของ v_{co} เมื่อผ่านวงจร

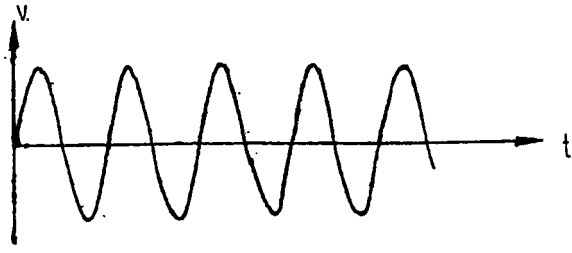
power amplify จะได้

$$\text{ขนาดสัญญาณ} = 7.1 \text{ V}$$



รูปที่ 4.18 แสดงสัญญาณ v_{co} เมื่อผ่านวงจร power amplify

5) สัญญาณ AF ที่นำมาทดลอง mod
 สัญญาณรูป sine ความถี่ = 4 kHz
 ขนาดสัญญาณ = 0.1 V



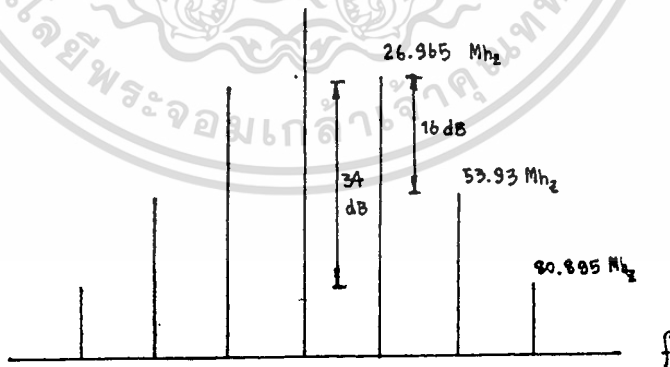
รูปที่ 4.19 แสดงสัญญาณ AF ที่ใช้ mod

6) spectrum ของ vco เมื่อยังไม่ mod

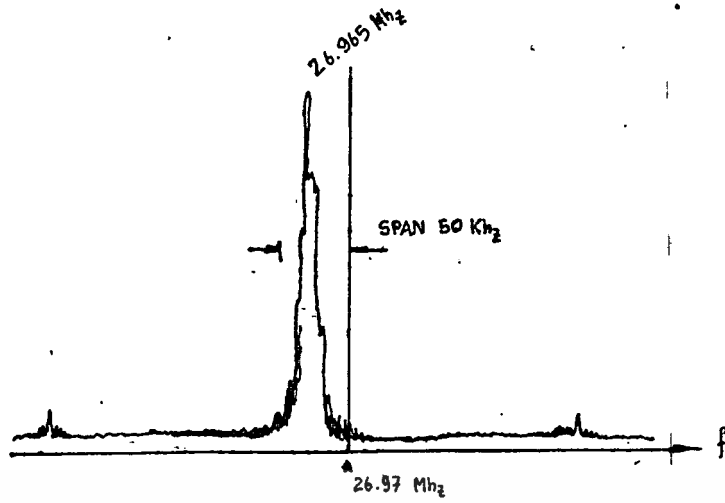


รูปที่ 4.20 แสดง spectrum ของ vco เมื่อยังไม่ mod

7) spectrum ของ vco เมื่อผ่าน power amplify โดยยังไม่ mod

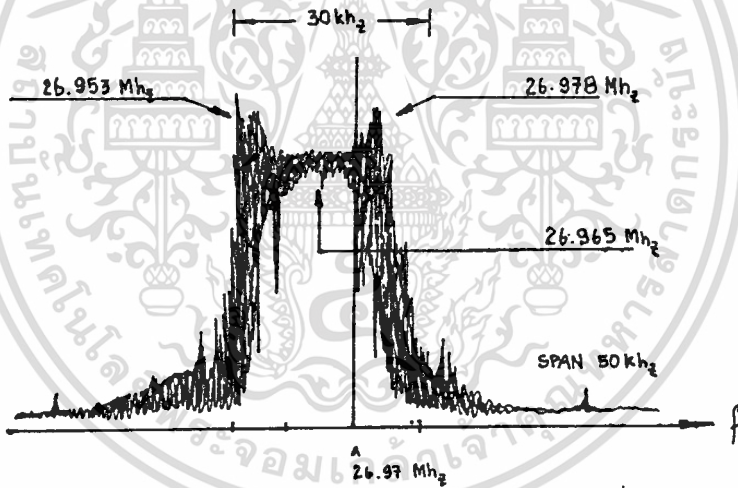


รูปที่ 4.21 แสดง spectrum เมื่อผ่าน power amplify



รูปที่ 4.22 แสดง spectrum ที่ย่านความถี่ 27 Mhz

8) spectrum ของ vco เมื่อผ่าน power amplify และ mod ด้วยสัญญาณในข้อ 5)



รูปที่ 4.23 แสดง spectrum เมื่อผ่าน power amplify และ mod แล้ว

9) สัญญาณเมื่อ modulate แล้วจะหาค่า

$$\begin{aligned} \text{Mod index } m_f &= \Delta f / f_c = (25 \text{ kHz} / 2) / 4 \text{ kHz} = 3.125 \\ \text{Percent Mod } M &= \Delta f_{\text{actual}} \times 100 / \Delta f_{\text{max}} \\ &= 12.5 \text{ kHz} \times 100 / 10 \text{ kHz} \\ &= 125 \% \end{aligned}$$

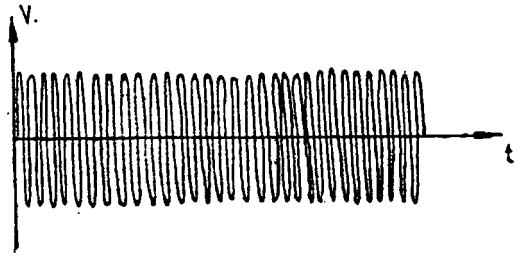
4.2.4 ผลการทดลองวัดสัญญาณภาคสังเคราะห์ความถี่โดยให้ค่า $N = 83$

1) เมื่อทดลองเลือก ch ที่ 20 โดยป้อนค่า

$$N = (01010011)_2 = (83)_{10}$$

$$\text{จะได้ความถี่ } v_{co} = 27.205 \text{ Mhz}$$

$$\text{ขนาดสัญญาณ} = 235 \text{ mV}$$



รูปที่ 4.24 แสดงสัญญาณ เมื่อ $N = 83$

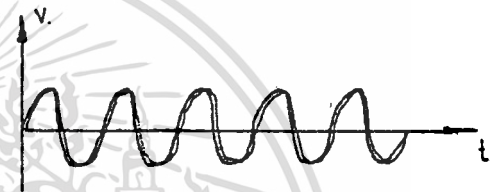
2) วัดสัญญาณ mixer output ที่

ขา 3 ของ IC MC 3359 จะได้

$$\text{ความถี่หลังผ่าน mixer} = 415 \text{ khz}$$

$$\text{ขนาดสัญญาณ} = 0.36 \text{ V}$$

$$\text{ขนาดสัญญาณเมื่อผ่านวงจรถยาย} = 2.5 \text{ V}$$



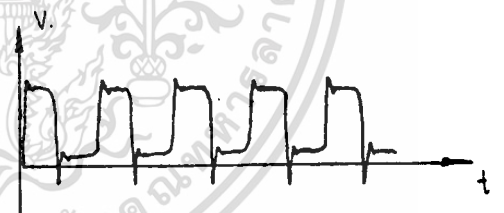
รูปที่ 4.25 แสดงสัญญาณของ mixer output

3) เมื่อสัญญาณในข้อ 2) ที่ขยายแล้ว

ผ่านวงจรถ scmitt trigger 7414 จะได้

$$\text{สัญญาณรูปสี่เหลี่ยมขนาด} = 5 \text{ V}$$

$$\text{duty cycle} = 48 \%$$

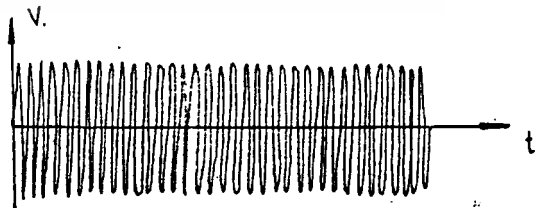


รูปที่ 4.26 แสดงสัญญาณเมื่อผ่าน scmitt trigger

4) สัญญาณของ v_{co} เมื่อผ่าน

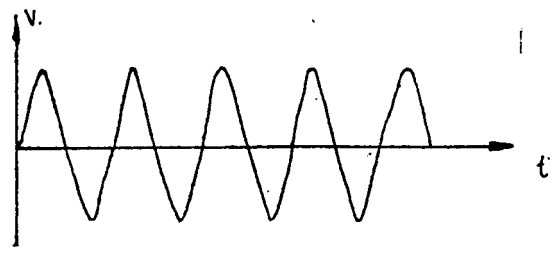
power amplify จะได้

$$\text{ขนาดสัญญาณ} = 6.8 \text{ V}$$



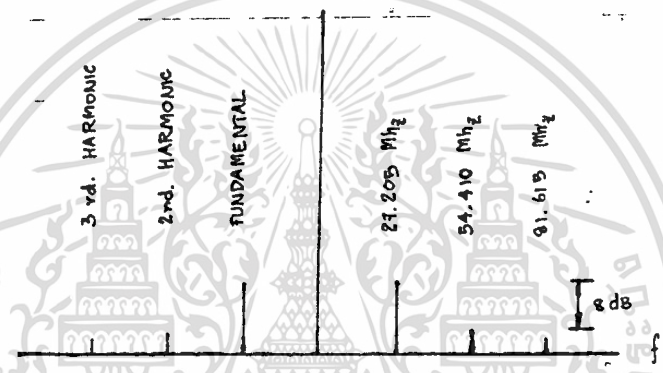
รูปที่ 4.27 แสดงสัญญาณ v_{co} เมื่อผ่านวงจรถ power amplify

5) สัญญาณ AF ที่นำมาทดลอง mod
 สัญญาณรูป sine ความถี่ = 4 khz
 ขนาดสัญญาณ = 0.1 V



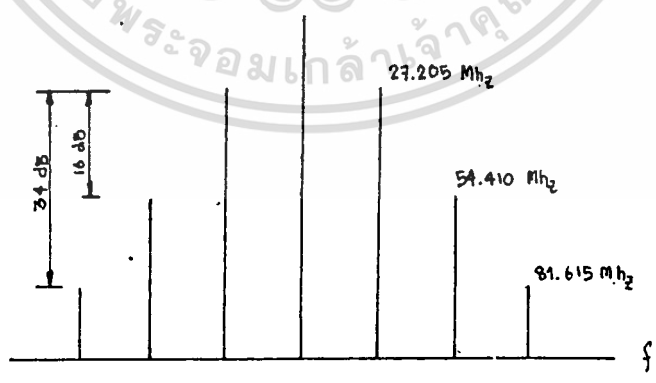
รูปที่ 4.28 แสดงสัญญาณ AF ที่ใช้ mod

6) spectrum ของ vco เมื่อยังไม่ mod

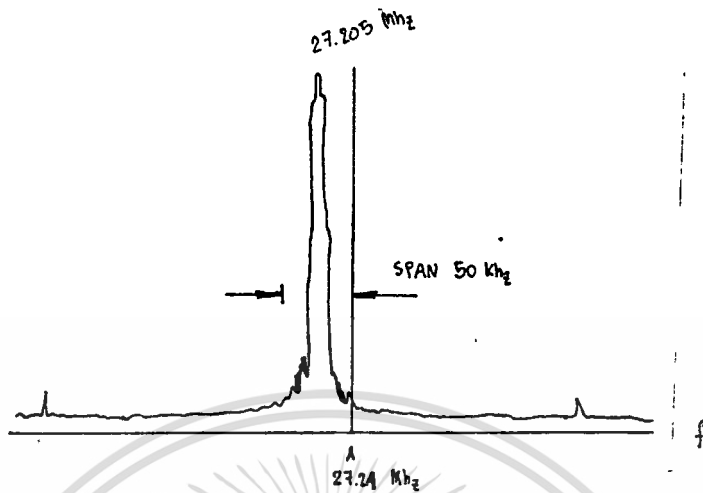


รูปที่ 4.29 แสดง spectrum ของ vco เมื่อยังไม่ mod

7) spectrum ของ vco เมื่อผ่าน power amplify โดยยังไม่ mod

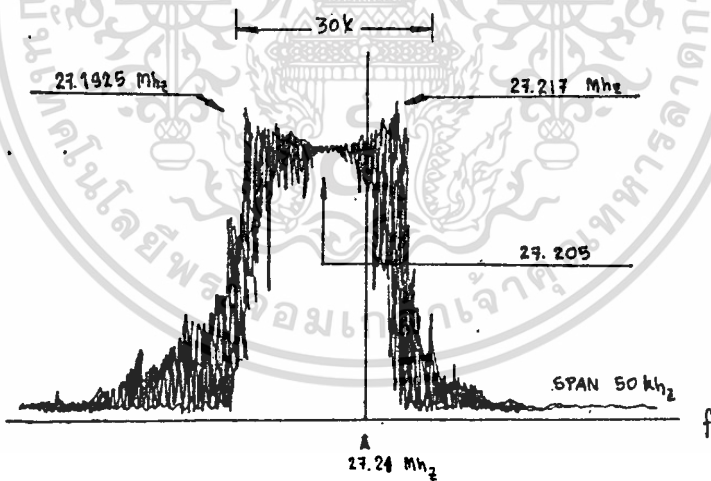


รูปที่ 4.30 แสดง spectrum เมื่อผ่าน power amplify



รูปที่ 4.31 แสดง spectrum ที่ย่านความถี่ 27 Mhz

8) spectrum ของ vco เมื่อผ่าน power amplify และ mod ด้วยสัญญาณในข้อ 5)



รูปที่ 4.32 แสดง spectrum เมื่อผ่าน power amplify และ mod แล้ว

9) สัญญาณเมื่อ modulate แล้วจะหาค่า

$$\text{Mod index } m_f = \frac{\Delta f}{f_c} = \left(\frac{25 \text{ kHz}}{2} \right) / 4 \text{ kHz} = 3.125$$

$$\begin{aligned} \text{Percent Mod } M &= \frac{\Delta f_{\text{actual}}}{\Delta f_{\text{max}}} \times 100 \\ &= \frac{12.5 \text{ kHz}}{10 \text{ kHz}} \times 100 \\ &= 125 \% \end{aligned}$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา -60- ต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

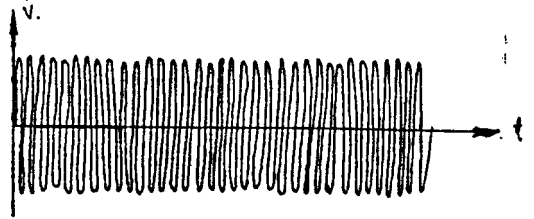
4.2.5 ผลการทดลองวัดสัญญาณภาคสังเคราะห์ความถี่โดยให้ค่า $N = 123$

1) เมื่อทดลองเลือก ch ที่ 40 โดยป้อนค่า

$$N = (01111011)_2 = (123)_{10}$$

$$\text{จะได้ความถี่ } v_{co} = 27.405 \text{ Mhz}$$

$$\text{ขนาดสัญญาณ} = 280 \text{ mV}$$



รูปที่ 4.33 แสดงสัญญาณ เมื่อ $N = 123$

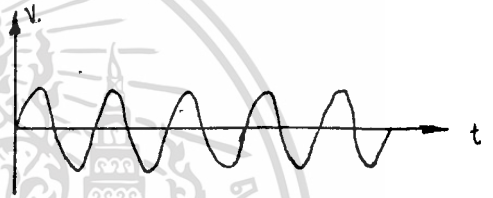
2) วัดสัญญาณ mixer output ที่

ขา 3 ของ IC MC 3359 จะได้

$$\text{ความถี่หลังผ่าน mixer} = 615 \text{ khz}$$

$$\text{ขนาดสัญญาณ} = 0.34 \text{ V}$$

$$\text{ขนาดสัญญาณเมื่อผ่านวงจรขยาย} = 2.6 \text{ V}$$



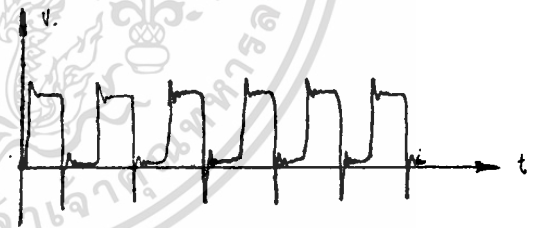
รูปที่ 4.34 แสดงสัญญาณของ mixer output

3) เมื่อสัญญาณในข้อ 2) ที่ขยายแล้ว

ผ่านวงจร schmitt trigger 7414 จะได้

$$\text{สัญญาณรูปสี่เหลี่ยมขนาด} = 5 \text{ V}$$

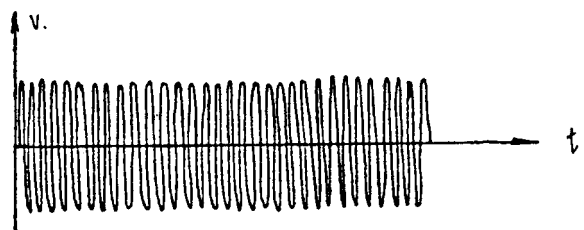
$$\text{duty cycle} = 48 \%$$



รูปที่ 4.35 แสดงสัญญาณเมื่อผ่าน schmitt trigger

4) สัญญาณของ vco เมื่อผ่าน power amplify จะได้

$$\text{ขนาดสัญญาณ} = 6.6 \text{ V}$$

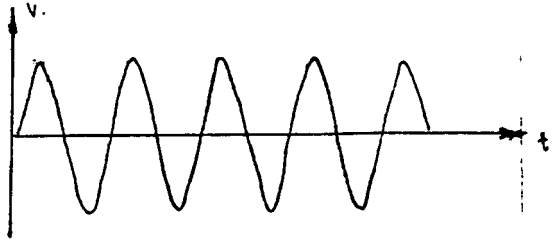


รูปที่ 4.36 แสดงสัญญาณ vco เมื่อผ่านวงจร power amplify

5) สัญญาณ AF จากส่วน modem ที่เข้ามา mod

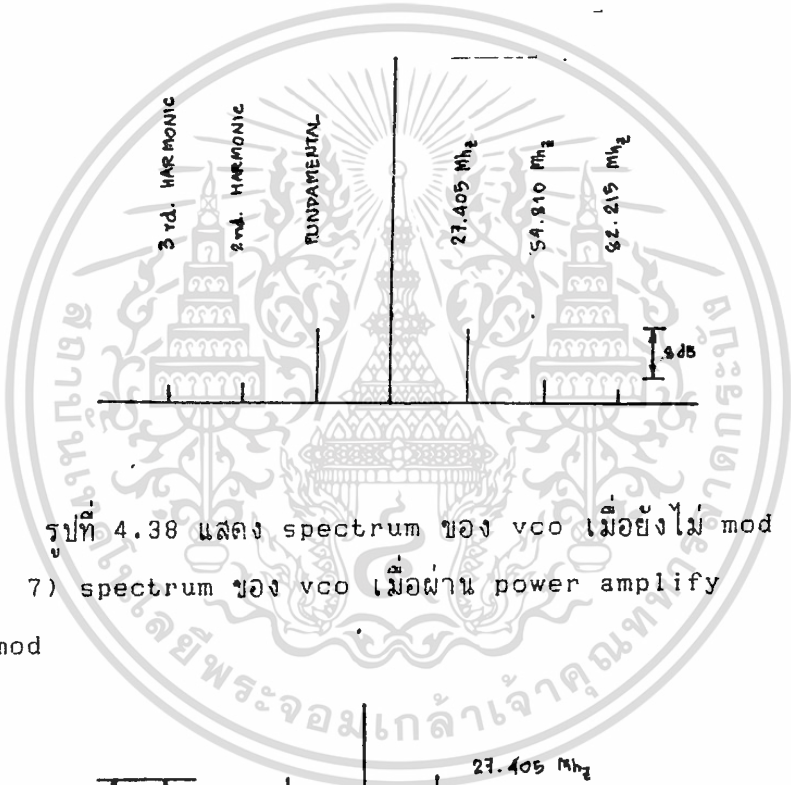
สัญญาณรูป sine ความถี่ = 4 khz

ขนาดสัญญาณ = 0.1 V



รูปที่ 4.37 แสดงสัญญาณ AF ที่ใช้ mod

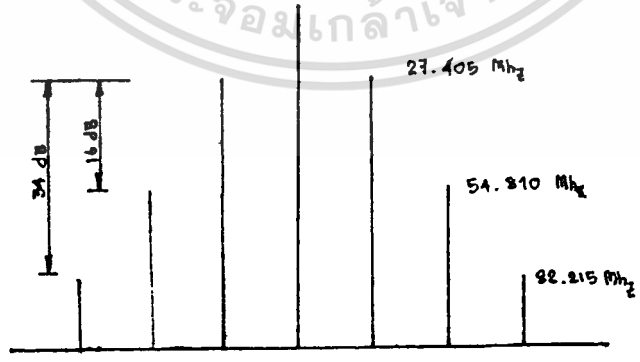
6) spectrum ของ vco เมื่อยังไม่ mod



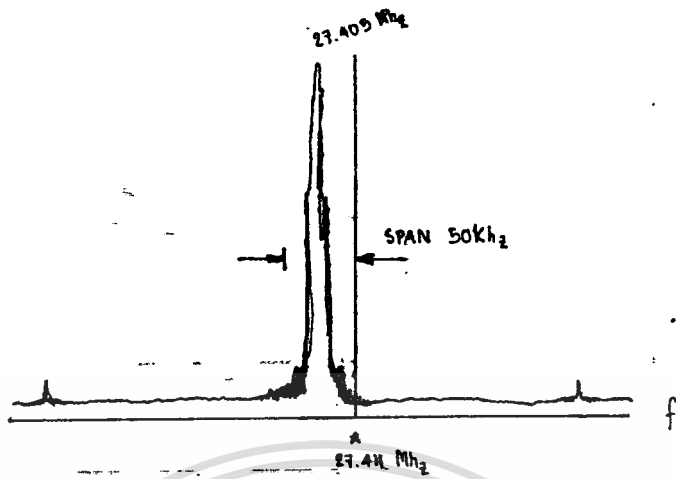
รูปที่ 4.38 แสดง spectrum ของ vco เมื่อยังไม่ mod

7) spectrum ของ vco เมื่อผ่าน power amplify

โดยยังไม่มี mod

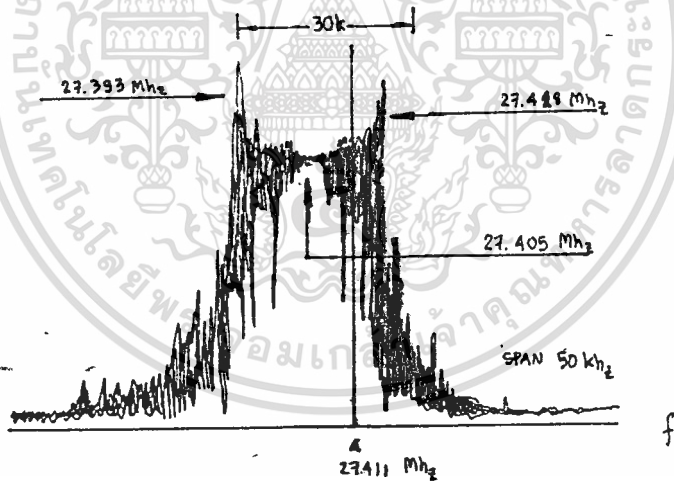


รูปที่ 4.39 แสดง spectrum เมื่อผ่าน power amplify



รูปที่ 4.40 แสดง spectrum ที่ย่านความถี่ 27 Mhz

8) spectrum ของ vco เมื่อผ่าน power amplify และ mod ด้วยสัญญาณในข้อ 5)



รูปที่ 4.41 แสดง spectrum เมื่อผ่าน power amplify และ mod แล้ว

9) สัญญาณเมื่อ modulate แล้วจะหาค่า

$$\begin{aligned} \text{Mod index } m_f &= \Delta f / f_c = (25 \text{ kHz} / 2) / 4 \text{ kHz} = 3.125 \\ \text{Percent Mod } M &= \Delta f_{\text{actual}} \times 100 / \Delta f_{\text{max}} \\ &= 12.5 \text{ kHz} \times 100 / 10 \text{ kHz} \\ &= 125 \% \end{aligned}$$

บทที่ 5

บทวิจารณ์และสรุป

งานโครงการงาน "ระดมใจรั้วสาย" ได้แบ่งการทดลองส่งข้อมูลออกเป็น 2 ลักษณะ และสรุปผลการทดลองได้ดังนี้

1) การทดลองส่งข้อมูลจาก คีย์บอร์ดของคอมพิวเตอร์ฝั่งหนึ่ง ภายผ่านโมเด็ม แล้วต่อเข้ากับโมเด็มของคอมพิวเตอร์อีกฝั่งหนึ่ง ภายใช้สายโทรศัพท์แบบ Lease Line เป็นระยะทาง 20 เมตร สามารถรับข้อมูลได้โดยไม่มีเกิดความผิดพลาด และการลดหย่อนของสัญญาณนั้นๆมีน้อยมาก แสดงว่าการทำงานในส่วนของโมเด็มสามารถทำงานได้ดี

2) การทดลองส่งข้อมูลโดยผ่าน RF Carrier จะพบว่ามีเกิด Character Error ประมาณ 10% ของจำนวนอักขระทั้งหมดที่ส่งออกไป ซึ่งเกิดจากสาเหตุต่างๆดังต่อไปนี้

ก. ช่องสัญญาณที่ใช้นั้นย่าน Citizen Band Class D (26.965-27.405 MHz) เป็นช่องสัญญาณที่มีการใช้งานอยู่มาก เพราะฉะนั้นในขณะที่ไม่ทำการส่งข้อมูล สถานะของช่องสัญญาณจะไม่ว่าง (Idle) โดยตลอด ทำให้เครื่องรับสามารถรับข้อมูลได้ทั้งที่ยังมีการส่งข้อมูลออกไป ทั้งนี้ก็เพราะว่าการสื่อสารแบบอะซิงโครนัส ในขณะที่ไม่ทำการส่งข้อมูล สถานะทางลอจิกจะต้องเป็น High โดยตลอด

ข. กำลังส่งของเครื่องส่งต่ำเกินไป ประมาณ 250 mW ทำให้สัญญาณข้อมูลมีกำลังส่งแรงไม่พอ

ค. เครื่องคอมพิวเตอร์ที่ใช้ มีความถี่ของ Oscillator ในตัวเอง ใกล้เคียงกับความถี่คลื่นนำที่ใช้อ้างอิง ทำให้เกิดสัญญาณรบกวนจากคอมพิวเตอร์ตัวรับไปยังเครื่องรับได้

ข้อ เสนอแนะและปรับปรุง

ก. Resolution ของความถี่ของวงจรสังเคราะห์ความถี่ยังไม่ละเอียดเท่าที่ควรควรมี Resolution 5 KHz ถ้าจะรับข้อมูลได้ก็ควรปรับปรุงให้มี Resolution ที่ 1 KHz จะทำให้ความผิดพลาดลดลง

ข. วงจรเพาเวอร์แอมป์ที่ใช้อ้างอิงให้ระดับเพาเวอร์เอาต์พุตที่ต่ำ ควรปรับปรุงให้ได้กำลังส่งมากขึ้น เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่นอนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ค. แบนด์วิธของการมอดดูเลทกว้างเกินกว่าที่ CCITT กำหนดไว้คือ 10 KHz ต่อหนึ่งช่องสัญญาณ คือมีแบนด์วิธประมาณ 25% และมีเบอร์เซ็นต์มอดดูเลท 125% เพราะฉะนั้นถ้ามีการสื่อสารข้อมูลหรือมีภาระทุกช่องสัญญาณ ก็จะทำให้เกิดการรบกวนกันระหว่างช่องได้

ง. ระบบการสื่อสารข้อมูลยังเป็นแบบ Simplex อยู่ ควรที่จะปรับปรุงให้สามารถส่งข้อมูลแบบ Half Duplex ได้ โดยการสร้างเครื่องรับ - ส่งวิทยุขึ้น พร้อมทั้งเขียนซอฟต์แวร์ที่จะอำนวยความสะดวก Hand Shake

จ. รมเดิมที่ช้ายังมีความเร็วต่ำอยู่ ควรเปลี่ยนตัวรมเดิมให้มีความเร็วที่สูงขึ้น พร้อมทั้งเขียนโปรแกรมเพื่อทำ Error Detection ขึ้นมา



ภาคผนวก ก.

โปรแกรมการส่งข้อมูล

```

;-----
; MAIN MODULE FOR TRANSMIT /
; PROGRAM /
;-----

extrn screen1:far
extrn screen2:far
extrn screen3:far
extrn screen4:far

main_data segment
    num db 2
    false db 22,10,'YOU PRESS WRONG KEY','
           db 23,10,' TRY AGAIN ','
    div db 00,35,37,39,43,45,47,49,53,55
         db 59,63,65,67,69,73,75,77,79,83
         db 87,93,89,91,95,97,99,101,103,
         db 107,109,111,113,115,117,119,1
    failure db 0dh,0ah,31 dup(' ')
           db 'TRANSMISSION FAILURE','$'
main_data ends
main_code segment public 'code'
    assume cs:main_code,ds:main_data
read_key macro
    mov ah,08h
    int 21h
endm

;-----
; MAIN PROGRAM FOR MAIN /
; MODULE /
;-----

start: mov ax,main_data
       mov ds,ax ;set ds regist
       call screen1
ski: read_key
     cmp al,0dh

```

```

jne    sk1
call   screen2
sk2:   read_key
       mov    ch,al        ;input channel
       read_key
       mov    cl,al        ;between 01 -
       cmp    cx,3031h     ;is input data
       jb    sk4
       cmp    cx,3430h     ;is input data
       ja    sk4
       sub    ch,30h
       sub    cl,30h
       mov    al,ch
       mov    dl,10
       mul   dl
       sub    ch,ch
       add    ax,cx
       mov    di,ax
       mov    dl,div[di]
       add    dl,128
       mov    al,dl
       mov    dx,023fh
       out   dx,al
       mov    bl,90
       call  delay
       mov    dl,div[di]
       mov    al,dl
       mov    dx,023fh
       out   dx,al
       call  screen3
sk3:   read_key
       cmp    al,31h
       jb    sk5
       cmp    al,32h
       ja    sk5
       jmp   transmission
sk4:   call   error_mesg
       jmp   sk2

```

```

                sk5:   call   error_mesg
                   jmp    sk3
transmission:  call   tranemit
                   mov    ah,4ch
                   int    21h

```

```

;-----
; ERROR MESSAGE PROCEDURE /
;   FOR MAIN MODULE      /
;-----

```

```

error_mesg     proc
                lea    di,num
                sub    ch,ch
                mov    cl,[di]
                lea    di,false
sk8:           mov    dh,[di]
                mov    dl,[di+1]
                push   cx
                add    di,2
                mov    cx,1
sk6:           push   bx
                mov    ah,2
                int    10h
                mov    al,[di]
                cmp    al,04h
                je     sk7
                mov    bl,8fh
                mov    ah,9
                int    10h
                inc    dl
                inc    di
                pop    bx
                jmp    sk6
sk7:           inc    di
                pop    bx
                pop    cx
                loop   sk8
                ret
error_mesg     endp

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลง-67- และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

;-----
; DELAY PROCEDURE FOR /
;   MAIN MODULE   /
;-----

```

```

delay proc
    push cx
    push dx
    sub cx,cx
    sub dx,dx
    mov ah,1
    int lah

chk_5sec: mov ah,0
          int lah
          cmp dl,bl
          jb  chk_5sec
          pop dx
          pop cx
          ret
delay endp

```

```

;-----
; CLEAR SCREEN PROCEDURE /
;   FOR MAIN MODULE   /
;-----

```

```

clrscr proc
    mov ch,0
    mov dh,24
    int 10h

    mov ah,6
    mov al,0
    mov cx,0
    mov dh,24
    mov dl,79
    mov bh,7
    int 10h
    ret
clrscr endp

```

```

;-----
; TRANSMIT PROCEDURE /
; FOR MAIN MODULE /
;-----

```

```

transmit proc
    mov dx,03fbh ;set 8bit word,1 stop bit
    mov al,83h ;no parity bit,DLAB = 1
    out dx,al
    mov dx,03f9h ;set baud rate at 600 baud
    mov al,00h
    out dx,al
    mov dx,03f8h
    mov al,0c0h
    out dx,al
    mov dx,03fbh ;set DLAB = 0
    mov al,03h
    out dx,al
    mov dx,03fch
    mov al,02h
    out dx,al ;out RTS signal
chk_CTS: sub cx,cx
        sub dx,dx
        mov ah,1
        int 1ah
chk_3sec: mov dx,03feh
        in al,dx ;read modem status
        and al,10h ;receive CTS signal
        jnz send_data ;if yes send data
        mov ah,0
        int 1ah
        cmp dl,90 ;if no wait until 3 sec
        je stop1
        jmp chk_3sec
send_data: call screen4
        mov ah,15
        int 10h
        mov dh,1

```

```

inc_line:  mov  dl,3
           inc  dh
           mov  cx,0
sk9:      push dx
           mov  ah,2
           int  10h
chk_reg:  mov  dx,03fdh
           in   al,dx      ;read line status register
           jz   chk_reg    ;is transmit holding register empty
           mov  ah,08h    ;if yes read character to send
           int  21h      ;from keyboard
           cmp  al,08h
           je   sk11
           mov  dx,03f8h
           out  dx,al     ;transmit data to RS-232c port
           pop  dx
           cmp  al,05h
           jz   sk10
           cmp  al,23h    ;is it '#' character
           je   stop2     ;if yes end transmission
           push cx        ;if no display character on
           mov  bl,0fh    ;screen
           mov  cx,1
           mov  ah,9
           int  10h
           pop  cx
           inc  cx
           cmp  cx,75
           jz   sk10
           inc  dl
           jmp  sk9
sk10:     jmp  inc_line
sk11:     pop  dx
           dec  dl
           dec  cx
           jmp  sk9
stop1:    mov  dx,offset failure
           mov  ah,09h

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
 ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลง-70- และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

int 21h
stop2: mov dx,03fch
mov al,00h
out dx,al
ret
transmit endp
main_code ends
end start
end

```



```

;-----
;PROCEDURE FAR SCREEN 1 MODULE /
; TO DISPLAY MENU1 /
;-----

```

```

public screen1
menudata segment public 'data'
count1 db 9
data1 db 5,34,0b0h,'PROJECT',0b0h,'$'
db 6,28,0b0h,'40 CH. WIRELESS MODEM',0b0h,'$'
db 8,38,'BY','$'
db 10,20,31h,'.', '$'
db 12,20,32h,'.', '$'
db 14,35,'ADVISOR','$'
db 18,24,0b0h,'TELECOMMUNICATION DEPARTMENT',0b0
db 19,34,0b0h,'KMIT_L',0b0h,'$'
db 23,52,20h,'PRESS ENTER TO CONTINUE',20h,'$'
count2 db 3
data2 db 10,24,'Prapoj Jalemongkolrat 291107','$'
db 12,24,'Pongpan Makrungruengkrai 291141','$'
db 16,25,'Asst.Prof. Narong Haemakorn','$'
menudata ends
menucode segment public 'code'
screen1 proc far
assume cs:menucode,ds:menudata

```

```

;-----
;MAIN PROGRAM FOR SCREEN1 /
; MODULE /
;-----

```

```

begin: push ds
sub ax,ax
push ax
mov ax,menudata ;set ds register
mov ds,ax
mov ah,0 ;set 80x25 black and white
int 10h ;monitor
mov ah,15 ;read active display page
int 10h
mov dh,2 ;set the first row and column

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา -72- ต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

mov     dl,9           ;for the top of horizontal line
        call horizon
        mov     dh,21      ;set the first row and column
        mov     dl,9       ;for the bottom of horizontal
        call horizon
        mov     dh,2       ;set the first row and column
        mov     dl,9       ;for the left of vertical line
        call vertical
        mov     dh,2       ;set the first row and column
        mov     dl,69      ;for the right of vertical lin
        call vertical
        call rev_char
        call normal_char
        pop     ax
        pop     ds
        ret

```

```

;-----
;HORIZONTAL LINE PLOT PROCEDURE /
;   FOR SCREEN1 /
;-----

```

```

horizon proc
        push    bx
        mov     cx,1
repl:   mov     ah,2
        int     10h
        mov     bl,70h
        mov     al,0cdh
        mov     ah,9
        int     10h
        inc     dl
        cmp     dl,70
        jnb    repl
        pop     bx
        ret
horizon endp

```

```

;-----
;VERTICAL LINE PLOT PROCEDURE /
;   FOR SCREEN1           /
;-----

```

```

vertical proc
push bx
mov cx,1
rep2: mov ah,2
int 10h
mov bl,70h
mov al,0bah
mov ah,9
int 10h
inc dh
cmp dh,22
jnb rep2
pop bx
ret
vertical endp

```

```

;-----
;REVERSE CHARACTER DISPLAY /
;  PROCEDURE FOR SCREEN1 /
;-----

```

```

rev_char proc
lea di,count1
sub ch,ch
mov cl,ldi1
lea di,data1
rep3: mov dh,ldi1
mov dl,ldi+11
push cx
mov cx,1
add di,2
rep4: push bx
mov ah,2
int 10h
mov al,[di]
cmp al,24h

```

```

je rep5
mov bl,70h
mov ah,9
int 10h
inc dl
inc di
pop bx
jmp rep4
rep5: inc di
pop bx
pop cx
loop rep3
ret
rev_char endp

```

;-----
;NORMAL CHARACTER DISPLAY /
; PROCEDURE FOR SCREEN1 /
;-----

```

normal_char proc
lea di,count2
sub ch,ch
mov cl,[di]
lea di,data2
rep6: mov dh,[di]
mov dl,[di+1]
push cx
mov cx,1
add di,2
rep7: push bx
mov ah,2
int 10h
mov al,[di]
cmp al,24h
je rep8
mov bl,0fh
mov ah,9
int 10h
inc di

```

```

inc di
pop bx
jmp rep7
rep8: inc di
pop bx
pop cx
loop rep6
ret
normal_char endp
screen1 endp
menucode ends
end begin
end

```



```

;-----
;PROCEDURE FAR SCREEN2 MODULE /
;   TO DISPLAY MENU2      /
;-----

```

```

                public  screen2
menudata       segment public 'data'
                count1  db    6
                data1   db    6,29,0b1h,'CHANNEL FREQUENCY',0b1h,'$'
                db    8,32,0b1h,'TRANSMISSON',0b1h,'$'
                db    12,27,' CH1 = ','$'
                db    16,27,' CH40 = ','$'
                db    18,24,' BANDWIDTH ','$'
                db    23,49,' PRESS NUMBER 1 - 40 ','$'
                count2  db    5
                data2   db    10,21,'Select Frequency To Transmit Data',
                db    12,38,'26.965 MHz','$'
                db    14,21,'To','$'
                db    16,38,'27.405 MHz','$'
                db    18,42,'10 KHz','$'
menudata       ends
menucode       segment public 'code'
screen2        proc    far
                assume  cs:menucode,ds:menudata

```

```

;-----
;MAIN PROGRAM FOR SCREEN2 /
;   MODULE                /
;-----

```

```

begin:  push  ds
        sub   ax,ax
        push ax
        mov  ax,menudata
        mov  ds,ax      ;set ds register
        mov  ah,0       ;set 80x25 black and white
        int  10h        ;monitor
        mov  ah,15      ;read active display page
        int  10h
        mov  dh,3       ;set the first row and column
        mov  dl,11      ;for the top of horizontal lin

```

```

call horizon
mov dh,21 ;set the first row and column
mov dl,11 ;for the bottom of horizontal
call horizon
mov dh,3 ;set the first row and column
mov dl,11 ;for the left of vertical lin
call vertical
mov dh,3 ;set the first row and column
mov dl,67 ;for the right of vertical li
call vertical
call rev_char
call normal_char
pop ax
pop ds
ret
;-----
; HORIZONTAL LINE PLOT /
; PROCEDURE FOR SCREEN2 /
;-----
horizon proc
push bx
mov cx,1
repl: mov ah,2
int 10h
mov bl,70h
mov al,0cdh
mov ah,9
int 10h
inc dl
cmp dl,67
jb repl
pop bx
ret
horizon endp

```

```
;-----  
; VERTICAL LINE PLOT /  
;PROCEDURE FOR SCREEN2 /  
;-----
```

```
vertical proc  
    push    bx  
    mov     cx,1  
rep2:  mov     ah,2  
    int     10h  
    mov     bl,70h  
    mov     al,0bah  
    mov     ah,9  
    int     10h  
    inc     dh  
    cmp     dh,22  
    jb     rep2  
    pop     bx  
    ret  
vertical endp
```

```
;-----  
; REVERSE CHARACTER DISPLAY /  
; PROCEDURE FOR SCREEN2 /  
;-----
```

```
rev_char proc  
    lea     di,count1  
    sub     ch,ch  
    mov     cl,ldi1  
    lea     di,data1  
rep3:  mov     dh,[di]  
    mov     di,[di+1]  
    push   cx  
    mov     cx,1  
    add     di,2  
rep4:  push   bx  
    mov     ah,2  
    int     10h  
    mov     al,[di]  
    cmp     al,24h
```

```

je rep5
mov bl,70h
mov ah,9
int 10h
inc dl
inc di
pop bx
jmp rep4
rep5: inc di
pop bx
pop cx

```

```

loop rep3
ret

```

rev_char endp

```

;-----
;NORMAL CHARACTER DISPLAY /
;PROCEDURE FOR SCREEN2 /
;-----

```

normal_char proc

```

lea di,count2
sub ch,ch
mov cl,[di]
lea di,data2
rep6: mov dh,[di]
mov dl,[di+1]
push cx
mov cx,1
add di,2

```

```

rep7: push bx
mov ah,2
int 10h
mov al,[di]
cmp al,24h
je rep8
mov bl,7fh
mov ah,9
int 10h
inc di

```

```
inc di
pop bx
jmp rep7
rep8: inc di
pop bx
pop cx
loop rep6
ret
normal_char endp
screen2 endp
menucode ends
end begin
end
```



```

-----
; PROCEDURE FAR SCREEN3 MODULE /
; TO DISPLAY MENU3 /
-----

```

```

public screen3
menudata segment public 'data'
count1 db 4
data1 db 8,28,0b1h,'METHOD OF TRANSMISSION',0b1h,'$'
db 10,23,' PRESS 1 FOR ','$'
db 12,23,' PRESS 2 FOR ','$'
db 14,32,20h,'SET UP ALREADY',20h,'$'
count2 db 2
data2 db 10,39,'Keyboard Transfer','$'
db 12,39,'File Transfer','$'
data3 db 5,16,0c9h,5,62,0bbh,18,16,0c8h,18,62,0bch,
menudata ends
menucode segment public 'code'
screen3 proc far
assume cs:menucode,d:menudata

```

```

-----
; MAIN PROGRAM FOR SCREEN3 /
; MODULE /
-----

```

```

begin: push ds
sub ax,ax
push ax
mov ax,menudata
mov ds,ax ;set ds register
mov ah,0 ;set 80x25 black and white
int 10h ;monitor
mov ah,15 ;read active display page
int 10h
mov dh,5 ;set the first row and column
mov dl,17 ;for the top of horizontal lin
call horizon
mov dh,18 ;set the first row and column
mov dl,17 ;for the bottom of horizontal
call horizon

```

```

mov dh,6 ;set the first row and column
mov dl,16 ;for the left of vertical line
call vertical
mov dh,6 ;set the first row and column
mov dl,62 ;for the right of vertical line
call vertical
call corner
call rev_char
call normal_char
pop ax
pop ds
ret

```

```

;-----
; HORIZONTAL LINE PLOT PROCEDURE /
; FOR SCREEN3 /
;-----

```

```

horizon proc
push bx
mov cx,i
repl: mov ah,2
int 10h
mov bl,0fh
mov al,0cdh
mov ah,9
int 10h
inc dl
cmp dl,62
jb repl
pop bx
ret
horizon endp

```

```

;-----
; VERTICAL LINE PLOT PROCEDURE /
; FOR SCREEN3 /
;-----

```

```

vertical proc
push bx
mov cx,1

```

```

rep2: mov    ah,2
      int    10h
      mov    bl,0fh
      mov    al,0bah
      mov    ah,9
      int    10h
      inc    dh
      cmp    dh,18
      jb     rep2
      pop    bx
      ret

```

```

-- vertical endp

```

```

; -----
; CORNER PLOT PROCEDURE /
;   FOR SCREEN3       /
; -----

```

```

corner proc
      mov    cx,1
      lea    di,data3
k1:   push   bx
      mov    dh,[di]
      cmp    dh,24h
      je     k2
      mov    dl,[di+1]
      mov    ah,2
      int    10h
      mov    bl,0fh
      mov    al,[di+2]
      mov    ah,9
      int    10h
      add    di,3
      pop    bx
      jmp    k1
k2:   pop    bx
      ret
corner endp

```

```

vertical proc
    push bx
    mov cx,1
rep2: mov ah,2
    int 10h
    mov bl,07h
    mov al,0bah
    mov ah,9
    int 10h
    inc dh
    cmp dh,24
    jb rep2
    pop bx
    ret
vertical endp

```

; CORNER PLOT PROCEDURE /

```

corner proc
    mov cx,1
    lea di,data2
    push bx
rep3: mov dh,[di]
    cmp dh,24h
    je rep4
    mov di,[di+1]
    mov ah,2
    int 10h
    mov bi,07h
    mov al,[di+2]
    mov ah,9
    int 10h
    add di,3
    jmp rep3
rep4: pop bx
    ret
corner endp

```

```

lea di,count2
sub ch,ch
mov cl,[di]
lea di,data2
rep6: mov dh,[di]
mov dl,[di+1]
push cx
mov cx,1
add di,2
rep7: push bx
mov ah,2
int 10h
mov al,[di]
cmp al,24h
je rep8
mov bl,7fh
mov ah,9
int 10h
inc dl
inc di
pop bx
jmp rep7
rep8: inc di
pop bx
pop cx
loop rep6
ret

normal_char endp
screen3 endp
menucode ends

end begin
end

```

```

        lea    di,count2
        sub    ch,ch
        mov    cl,[di]
        lea    di,data2
rep6:   mov    dh,[di]
        mov    dl,[di+1]
        push  cx
        mov    cx,1
        add    di,2
rep7:   push  bx
        mov    ah,2
        int    10h
        mov    al,[di]
        cmp    al,24h
        je     rep8
        mov    bl,7fh
        mov    ah,9
        int    10h
        inc    dl
        inc    di
        pop    bx
        jmp    rep7
rep8:   inc    di
        pop    bx
        pop    cx
        loop  rep6
        ret

normal_char  endp
screen3      endp
menucode     ends
end          begin
end

```

; CLEAR SCREEN PROCEDURE /

```
    clrscr    proc  
            mov    ch,0  
            mov    dh,24  
            mov    ah,0  
            int    10h  
            mov    ah,6  
            mov    al,0  
            mov    cx,0  
            mov    dh,24  
            mov    dl,79  
            mov    bh,7  
            int    10h  
            ret  
    clrscr    endp
```

; HORIZONTAL PLOT PROCEDURE /

```
    horizon  proc  
            push   bx  
            mov    cx,1  
repl:      mov    ah,2  
            int    10h  
            mov    bh,07h  
            mov    al,0dbh  
            mov    ah,9  
            int    10h  
            inc    dl  
            cmp    dl,79  
            jb    repl  
            pop    bx  
            ret  
    horizon  endp
```

; VERTICAL PLOT PROCEDURE /

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีก๊อมนำไปใช้

```

rep2: mov    ah,2
      int    10h
      mov    bl,0fh
      mov    al,0bah
      mov    ah,9
      int    10h
      inc    dh
      cmp    dh,18
      jnb   rep2
      pop    bx
      ret

```

```
vertical endp
```

```

; CORNER PLOT PROCEDURE /
;   FOR SCREEN3 /
;-----

```

```

corner proc
      mov    cx,1
      lea   di,data3
      k1:  push  bx
      mov   dh,[di]
      cmp   dh,24h
      je    k2
      mov   dl,[di+1]
      mov   ah,2
      int   10h
      mov   bl,0fh
      mov   al,[di+2]
      mov   ah,9
      int   10h
      add   di,3
      pop   bx
      jmp   k1
      k2:  pop   bx
      ret
corner endp

```

```

;-----
; NORMAL CHARACTER DISPLAY /
; PROCEDURE /
;-----

```

```

normal_char proc
    lea di,data1
    mov dh,fdil
    mov dl,(di+11)
    push bx
    add di,2
rep5:    mov cx,1
        mov ah,2
        int 10h
        mov al,(di)
        cmp al,24h
        je rep6
        mov bl,70h
        mov ah,9
        int 10h
        inc dl
        inc di
        jmp rep5
rep6:    pop bx
        ret
normal_char endp

```

```

;-----
; RECEIVE DATA PROCEDURE /
;-----

```

```

receive proc
    mov dx,03fbh ;set 8 bit word,1stop bit
    mov al,83h ;no parity
    out dx,al
    mov dx,03f9h ;set baud rate to 600
    mov al,00h
    out dx,al
    mov dx,03f8h
    mov al,0c0h
    out dx,al

```

```

mov dx,03fbh ;set DLAB = 0
mov al,03h
out dx,al
mov dx,03f8h ;clear any data receive
in al,dx ;before
mov ah,15
int 10h
mov dh,1
inc_line: mov dl,3
inc dh
mov cx,0
rep7: push dx
mov ah,2
int 10h
chk_reg: mov dx,03fdh ;read line status register
in al,dx
and al,01h ;if data in receive buffer
jz chk_reg ;no check again
mov dx,03f8h ;yes get data to display
in al,dx ;on screen
pop dx
cmp al,08h ;is data backspace?
je rep9
cmp al,0dh ;is data return?
jz rep8
cmp al,23h ;is data #
je stop
push cx
mov bl,0fh
mov cx,1
mov ah,9 ;display data at position
int 10h
pop cx
inc cx
cmp cx,75 ;is character is 75?
jz rep8
inc dl
jmp rep7

```

```
rep8: jmp inc_line ;point to first row and column
rep9: pop dx
      dec dl ;yes point column back
      dec cx ;decrement counter
      jmp rep7

stop: ret ;end of transmission

receive endp
rx_code ends

end start
end
```



กิตติกรรมประกาศ

ปริญญาโทฉบับนี้ สำเร็จลงได้ด้วยความช่วยเหลือจากบุคคลหลาย ๆ ฝ่าย ขอขอบพระคุณ ผศ. ณรงค์ เหมกรณ์ อาจารย์ที่ปรึกษามา ณ ที่นี้ ที่ได้ให้คำปรึกษาและสนับสนุนการทำงานด้านต่าง ๆ เป็นอย่างดีตลอดเวลา ขอขอบพระคุณ อ. นิภา ลีลารุจิ ที่ได้กรุณาให้คำแนะนำแนวทางการค้นคว้าหาข้อมูล ในการทำปริญญาโทฉบับนี้ ขอขอบคุณเพื่อน ๆ ทุกคน ที่ช่วยให้ความคิดเห็นและกำลังใจ สุดท้ายนี้ ขอขอบคุณ คุณ นุสรวิรุฬห์วิ ที่ให้กำลังใจและพิมพ์ต้นฉบับจนแล้วเสร็จ

ผู้จัดทำ

นาย ประพนธ์ เจตน์มงคลรัตน์
นาย พงษ์พันธ์ เมฆรุ่งเรืองไกร



หนังสืออ้างอิง

[1] " The 1988 ARRL HANDBOOK "

The American Radio Relay League 1988

[2] George B. Rutkowski ; "Integrate Circuit Operation Amplifier

Prentice Hall ; 1985

[3] Lewis C. Eggebrecht ; "Interface to the IBM PC"

Howard W. Sams & Co., Inc ; 1983

[4] IBM CORP., "TECHNICAL REFERENCE" IBM Personal computer

Hardware Reference Library , p., 1983.

[5] "Linear & Interface Integrated Circuit"

Motorola Inc.

[6] "Telecommunications Device Data"

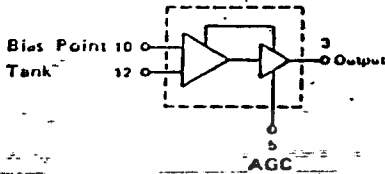
Motorola Inc. DL 136 1984

[7] "Mos Databook"

National Semiconductor

MC1648/MC1648M

VOLTAGE-CONTROLLED OSCILLATOR



Input Capacitance = 6 pF typ
 Maximum Series Resistance for L (External Inductance) = 50 Ω typ
 Power Dissipation = 150 mW typ/pkg
 (+5.0 Vdc Supply)
 Maximum Output Frequency = 225 MHz typ

The MC1648 requires an external parallel tank circuit consisting of the inductor (L) and capacitor (C).

A varactor diode may be incorporated into the tank circuit to provide a voltage variable input for the oscillator (VCO). The MC1648 was designed for use in the Motorola Phase-Locked Loop shown in Figure 9. This device may also be used in many other applications requiring a fixed or variable frequency clock source of high spectral purity. (See Figure 2.)

The MC1648 may be operated from a +5.0 Vdc supply or a -5.2 Vdc supply, depending upon system requirements.

Supply Voltage	Gnd Pins	Supply Pins
+5.0 Vdc	7, 8	1, 14
-5.2 Vdc	1, 14	7, 8



L SUFFIX
 CERAMIC PACKAGE
 CASE 632



P SUFFIX
 PLASTIC PACKAGE
 CASE 646



F SUFFIX
 CERAMIC PACKAGE
 CASE 607

FIGURE 1 - CIRCUIT SCHEMATIC

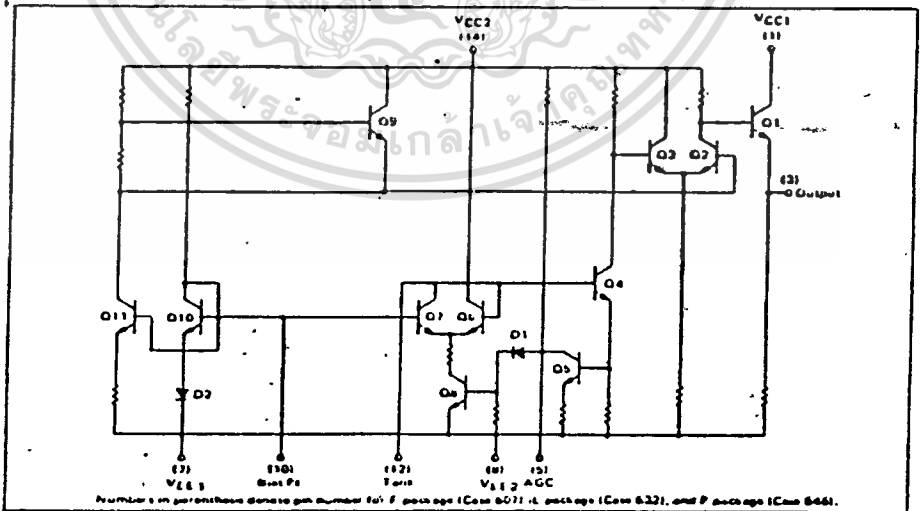
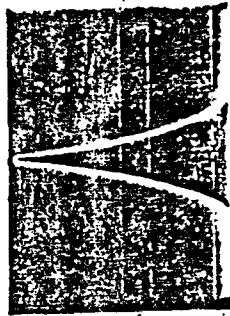
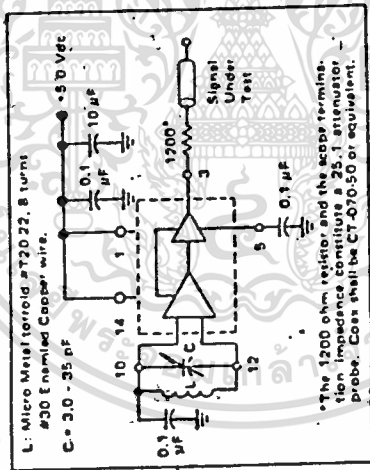


FIGURE 2 - SPECTRAL PURITY OF SIGNAL AT OUTPUT



B.W. = 10 kHz
 Center Frequency = 100 MHz
 Scan Width = 50 kHz/div
 Vertical Scale = 10 dB/div



TEST VOLTAGE/CURRENT VALUES		
	(Volts)	mA dc
V _{IHmax}		V _{CC}
MC1648		
	+1.50	5.0
	+1.35	5.0
	+1.20	5.0
MC1648M		
	+1.57	5.0
	+1.35	5.0
	+1.10	5.0

ELECTRICAL CHARACTERISTICS

Supply Voltage = +5.0 Volts

Characteristic	Symbol	-55°C		-30°C		+25°C		+85°C		+125°C		Unit	Conditions	
		Min	Max	Min	Max	Min	Max	Min	Max	Min	Max			
Power Supply Drain Current	I _E	-	-	-	4.1	-	-	-	-	-	-	mA dc	Inputs and outputs open.	
Logic "1" Output Voltage	V _{OH}	3.92	4.13	3.955	4.185	4.04	4.25	4.11	4.30	4.16	4.40	V _{dc}	V _{IHmin} to Pin 12, I _L @ P ₁ .	
Logic "0" Output Voltage	V _{OL}	3.13	3.38	3.16	3.40	3.20	3.43	3.22	3.475	3.23	3.51	V _{dc}	V _{IHmax} to Pin 12, I _L @ P ₁ .	
Bias Voltage	V _{Bias} *	1.67	1.97	1.60	1.90	1.45	1.75	1.30	1.60	1.20	1.50	V _{pc}	V _{IHmin} to Pin 12.	
		Min	Typ	Max	Min	Typ	Max	Min	Typ	Max	Min	Typ	Max	
		-	-	-	-	400	-	-	-	-	-	-	mV	
Peak-to-Peak Tank Voltage	V _{p-p}	-	-	-	-	-	-	-	-	-	-	%		
Output Duty Cycle	V _{DC}	-	-	-	-	-	-	-	-	-	-	%		
Oscillation Frequency	f _{max} **	-	725	-	-	200	225	-	225	-	225	MHz	See Figure 3.	

*This measurement guarantees the dc potential at the bias point for purposes of incorporating a varactor turning diode at this point.

** Frequency variation over temperature is a direct function of the ΔC/Δ Temperature and ΔL/Δ Temperature.

TEST VOLTAGE/CURRENT VALUES			
(Volts)		mAdc	
V _{IH} max	V _{IL} min	V _{CC}	I _L
MC1648			
-3.20	-3.70	-5.2	-5.0
-3.35	-3.85	-5.2	-5.0
-3.50	-4.00	-5.2	-5.0
MC1648M			
-3.13	-3.63	-5.2	-5.0
-3.35	-3.85	-5.2	-5.0
-3.60	-4.10	-5.2	-5.0



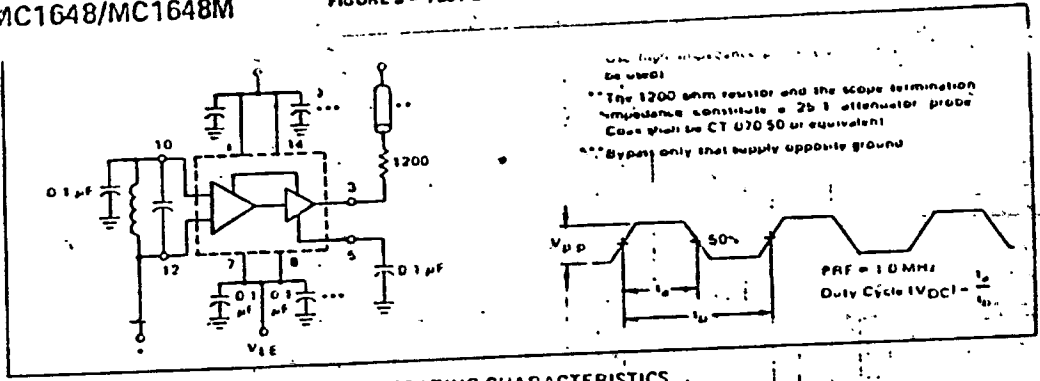
ELECTRICAL CHARACTERISTICS

Supply Voltage = -5.2 Volts

Characteristic	Symbol	-55°C		-30°C		+25°C		+85°C		+125°C		Unit	Conditions
		Min	Max	Min	Max	Min	Max	Min	Max	Min	Max		
Power Supply Drain Current	I _E	-	-	-	-	-	41	-	-	-	-	mAdc	Inputs and outputs open.
Logic "1" Output Voltage	V _{OH}	-1.080	-0.870	-1.045	-0.815	-0.960	-0.750	-0.890	-0.640	-0.840	-0.600	Vdc	V _{IL} min to Pin 12, I _L @ Pin
Logic "0" Output Voltage	V _{OL}	-1.920	-1.670	-1.890	-1.650	-1.850	-1.620	-1.830	-1.575	-1.820	-1.540	Vdc	V _{IH} max to Pin 12, I _L @ Pin
Bias Voltage	V _{Bias} *	-3.53	-3.23	-3.60	-3.30	-3.75	-3.45	-3.90	-3.60	-4.00	-3.70	Vdc	V _{IL} min to Pin 12.
Peak-to-Peak Tank Voltage	V _{p-p}	-	-	-	-	-	400	-	-	-	-	mV	
Output Duty Cycle	V _{DC}	-	-	-	-	-	50	-	-	-	-	%	See Figure 3.
Oscillation Frequency	f _{max} **	-	-	-	225	-	200	275	-	-	225	MHz	

*This measurement guarantees the dc potential at the bias point for purposes of incorporating a varactor turning diode at this point.

**Frequency variation over temperature is a direct function of the ΔC/A Temperature and ΔL/Δ Temperature.



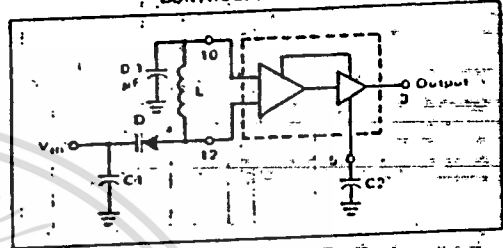
OPERATING CHARACTERISTICS

Figure 1 illustrates the circuit schematic for the MC1648. The oscillator incorporates positive feedback by coupling the base of transistor Q6 to the collector of Q7. An automatic gain control (AGC) is incorporated to limit the current through the emitter-coupled pair of transistors (Q7 and Q6) and allow optimum frequency response of the oscillator.

In order to maintain the high Q of the oscillator, and provide high spectral purity at the output, transistor Q4 is used to translate the oscillator signal to the output differential pair Q2 and Q3. Q2 and Q3, in conjunction with output transistor Q1, provides a highly buffered output which produces a square wave. Transistors Q9 and Q11 provide the bias drive for the oscillator and output buffer. Figure 2 indicates the high spectral purity of the oscillator output (pin 3).

When operating the oscillator in the voltage controlled mode (Figure 4), it should be noted that

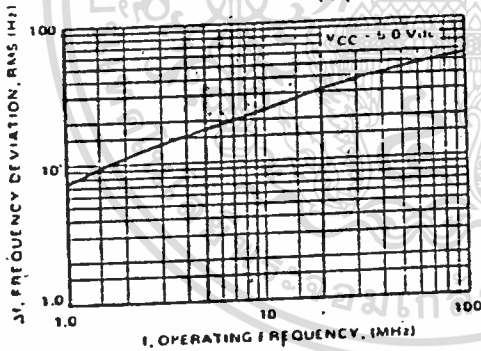
FIGURE 4 - THE MC1648 OPERATING IN THE VOLTAGE CONTROLLED MODE



the cathode of the varactor diode (D) should be biased at least 2 VBE above VEE (≈ 1.4 V for positive supply operation).

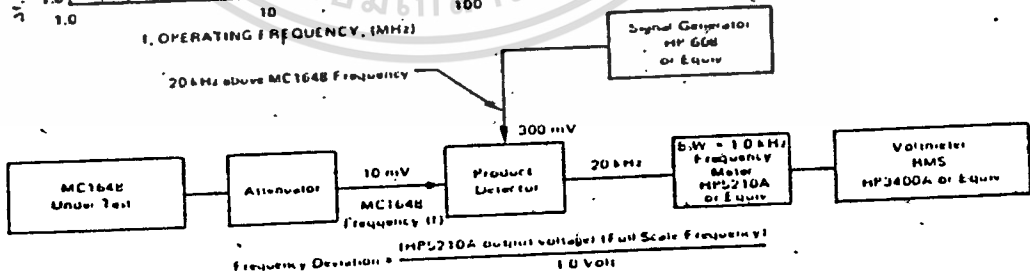
When the MC1648 is used with a constant dc voltage to the varactor diode, the output frequency will vary slightly because of internal noise. This variation is plotted versus operating frequency in Figure 5.

FIGURE 5 - NOISE DEVIATION TEST CIRCUIT AND WAVEFORM



Oscillator Tank Components (Circuit of Figure 4)

f MHz	-D	L μH
10.10	MV2115	100
10.60	MV2115	2.3
60.100	MV2106	0.15



NOTE: Any frequency deviation caused by the signal generator and MC1648 power supply should be determined and minimized prior to testing.

TRANSFER CHARACTERISTICS IN THE VOLTAGE CONTROLLED MODE
USING EXTERNAL VARACTOR DIODE AND COIL. $T_A = 25^\circ\text{C}$

FIGURE 6

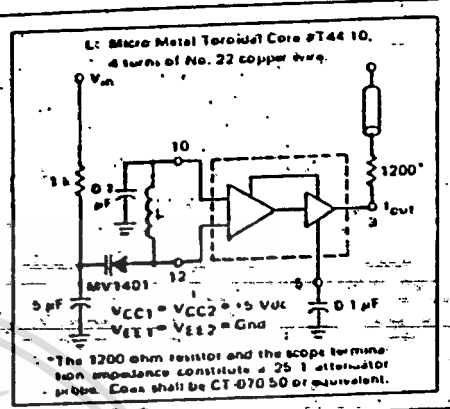
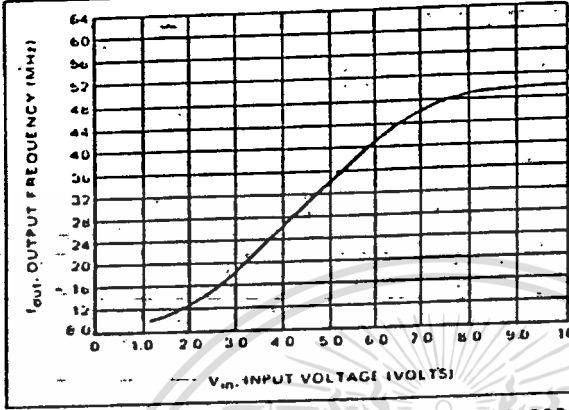


FIGURE 7

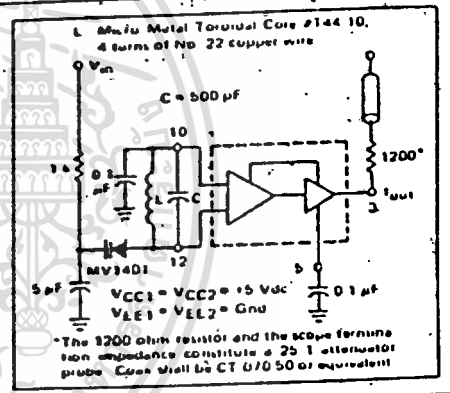
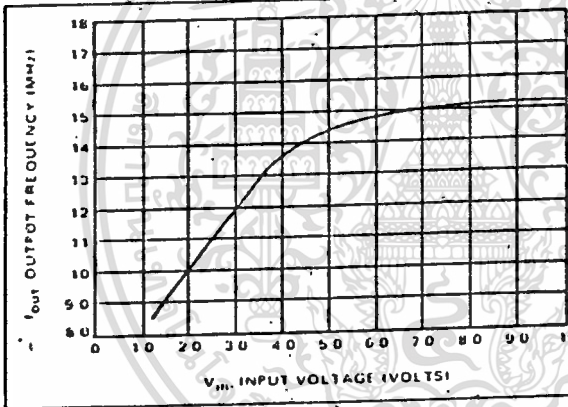
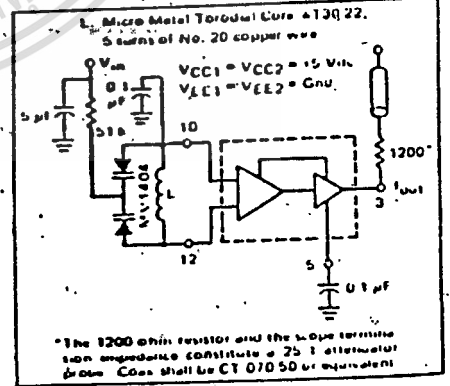
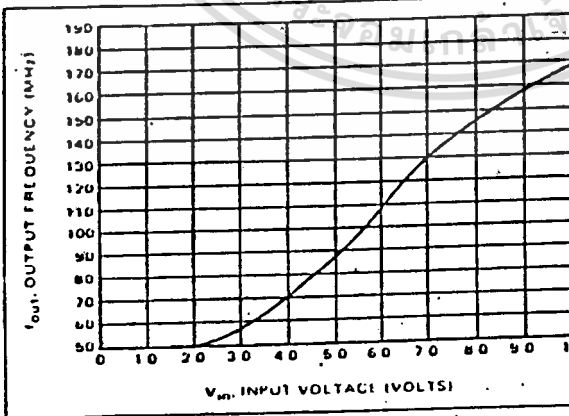


FIGURE 8



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ในการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Typical transfer characteristics for the oscillator in the voltage controlled mode are shown in Figures 6, 7, and 8. Figures 6 and 8 show transfer characteristics employing only the capacitance of the varactor diode (plus the input capacitance of the oscillator, 6 pF typical). Figure 7 illustrates the oscillator operating in a voltage controlled mode with the output frequency range limited. This is achieved by adding a capacitor in parallel with the tank circuit as shown. The 1 kΩ resistor in Figures 6 and 7 is used to protect the varactor diode during testing. It is not necessary as long as the dc input voltage does not cause the diode to become forward biased. The larger-valued resistor (51 kΩ) in Figure 8 is required to provide isolation for the high-impedance junctions of the two varactor diodes.

The tuning range of the oscillator in the voltage controlled mode may be calculated as:

$$f_{\max} = \frac{1}{2\pi\sqrt{L(C_D(\max) + C_S)}}$$

$$f_{\min} = \frac{1}{2\pi\sqrt{L(C_D(\min) + C_S)}}$$

$$\text{where } f_{\min} = \frac{1}{2\pi\sqrt{L(C_D(\max) + C_S)}}$$

C_S = shunt capacitance (input plus external capacitance).

C_D = varactor capacitance as a function of bias voltage.

Good RF and low-frequency bypassing is necessary on the power supply pins. (See Figure 2.)

Capacitors C1 and C2 of Figure 4) should be used to bypass the AGC point and the VCO input (varactor diode), guaranteeing only dc levels at these points.

For output frequency operation between 1 MHz and 50 MHz a 0.1 μF capacitor is sufficient for C1 and C2. At higher frequencies, smaller values of capacitance should be used; at lower frequencies, larger values of capacitance. At high frequencies the value of bypass capacitors depends directly upon the physical layout of the system. All bypassing should be as close to the package pins as possible to minimize unwanted lead inductance.

The peak-to-peak swing of the tank circuit is set internally by the AGC circuitry. Since voltage swing of the tank circuit provides the drive for the output buffer, the AGC potential directly affects the output waveform. If it is desired to have a sine wave at the output of the MC1648, a series resistor is tied from the AGC point to the most negative power potential (ground if +5.0 volt supply is used, -5.2 volts if a negative supply is used) as shown in Figure 10.

At frequencies above 100 MHz (typ), it may be desirable to increase the tank circuit peak-to-peak voltage in order to shape the signal at the output of the MC1648. This is accomplished by tying a series resistor (1 kΩ minimum) from the AGC to the most positive power potential (+5.0 volts if a +5.0 volt supply is used, ground if a -5.2 volt supply is used). Figure 11 illustrates this principle.

APPLICATIONS INFORMATION

The phase locked loop shown in Figure 9 illustrates the use of the MC1648 as a voltage controlled oscillator. The figure illustrates a frequency synthesizer useful in tuners for FM broadcast, general aviation, maritime and landmobile communications, amateur and CB receivers. The system operates from a single +5.0 Vdc supply, and requires no internal transistors, since all components are compatible.

Frequency generation of this type offers the advantages of single crystal operation, simple channel selection, and elimination of special circuitry to prevent harmonic lockup. Additional features include dc digital switching

(preferable over RF switching with a multiple crystal system), and a broad range of tuning (up to 150 MHz, the range being set by the varactor diode).

The output frequency of the synthesizer loop is determined by the reference frequency and the number programmed at the programmable counter, $f_{out} = Nf_{ref}$. The channel spacing is equal to frequency (1_{ref}).

For additional information on applications and designs for phase locked-loops and digital frequency synthesizers, see Motorola Application Notes AN-532A, AN-535, AN-553, AN-564 or AN594.

FIGURE 9 - TYPICAL FREQUENCY SYNTHESIZER APPLICATION

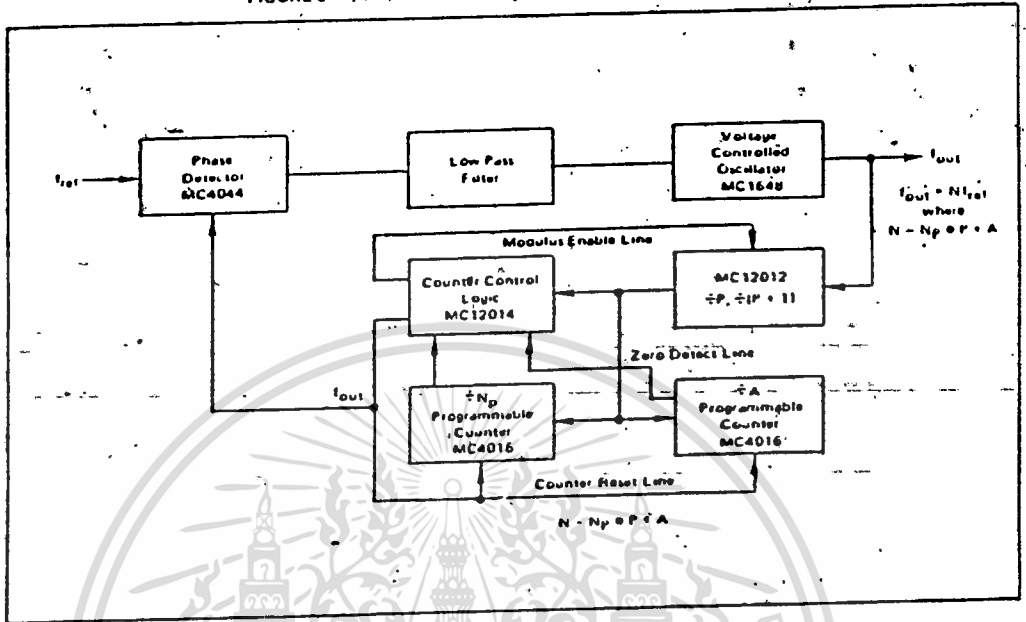


Figure 10 shows the MC1648 in the variable frequency mode operating from a +5.0 Vdc supply. To obtain a sine wave at the output, a resistor is added from the AGC circuit (pin 5) to VEE.

Figure 11 shows the MC1648 in the variable frequency mode operating from a +5.0 Vdc supply. To extend the useful range of the device (maintain a square wave output above 175 MHz), a resistor is added to the AGC circuit at pin 5 (1 k-ohm minimum).

Figure 12 shows the MC1648 operating from +5.0 Vdc and +9.0 Vdc power supplies. This permits a higher voltage swing and higher output power than is possible from the MECL output (pin 3). Plots of output power versus total collector load resistance at pin 1 are given in Figures 13 and 14 for 100 MHz and 10 MHz operation. The total collector load includes R in parallel with Rp of L1 and C1 at resonance. The optimum value for R at 100 MHz is approximately 850 ohms.

FIGURE 10 - METHOD OF OBTAINING A SINE-WAVE OUTPUT.

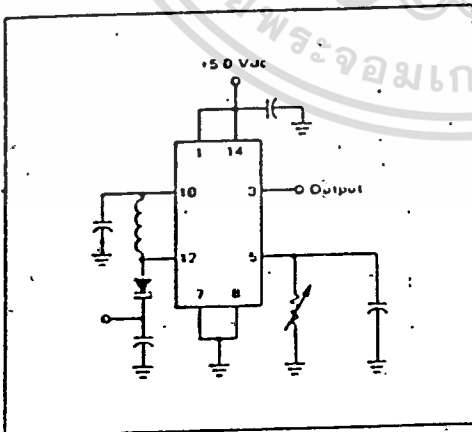


FIGURE 11 - METHOD OF EXTENDING THE USEFUL RANGE OF THE MC1648 (SQUARE WAVE OUTPUT)

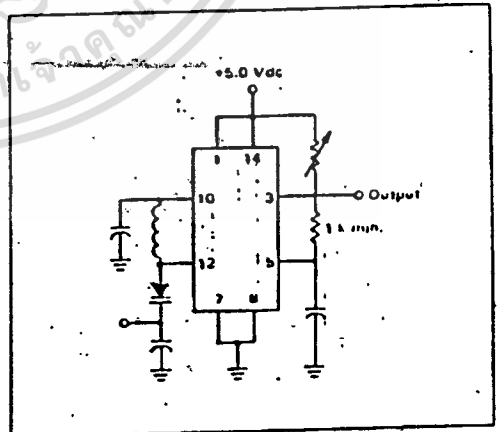


FIGURE 12 - CIRCUIT USED FOR COLLECTOR OUTPUT OPERATION

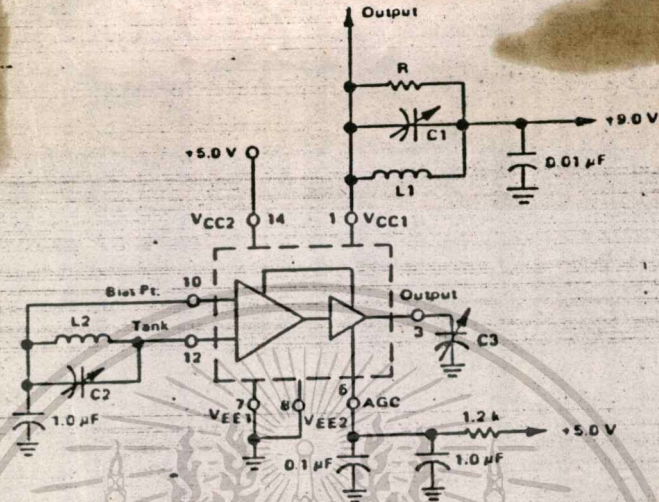


FIGURE 13 - POWER OUTPUT versus COLLECTOR LOAD

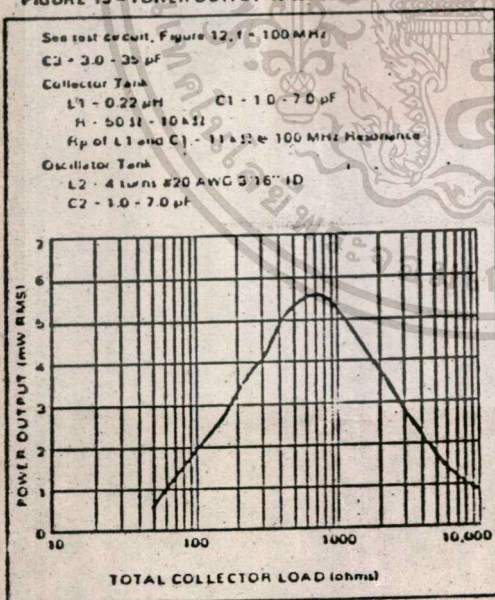


FIGURE 14 - POWER OUTPUT versus COLLECTOR LOAD

