

สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง



ระบบโทรศัพท์ภายในผ่านสายไฟฟ้า AC  
AC-LINE INTERPHONE SYSTEM



โดย  
นายสรศักดิ์ เจริญวัย  
นางสาวสโรชา เกตุศรีเมฆ  
นางสาวสรวล ชวลิตมงคล

เลขหมู่.....  
เลขทะเบียน..... **62015** .....

b..... 11586539 .....

ปริญญาบัตรนี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต

สาขาวิชาวิศวกรรมโทรคมนาคม

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2547

ภาควิชา  
วิศวกรรมโทรคมนาคม

นางสาว.....  
A. S. J.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ระบบโทรศัพท์ภายในผ่านสายไฟฟ้า AC  
AC-LINE INTERPHONE SYSTEM



โดย  
นายสรศักดิ์ เจริญวัย 44010512  
นางสาวสโรชา เกตุศรีเมฆ 44010515  
นางสาวสวรรส ขวลิขิตมงคล 44010516

อาจารย์ที่ปรึกษา  
ดร.พิพัฒน์ พรหมมี

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต  
สาขาวิชาวิศวกรรมโทรคมนาคม  
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง  
ปีการศึกษา 2547

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ปริญญาโทปีการศึกษา 2547

ภาควิชาวิศวกรรมโทรคมนาคม

คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เรื่อง ระบบโทรศัพท์ภายในผ่านสายไฟฟ้า AC

AC-LINE INTERPHONE SYSTEM

ผู้จัดทำ

1. นายสรศักดิ์ เจริญวัย 44010512
2. นางสาวสโรชา เกตุศรีเมฆ 44010515
3. นางสาวสวรส ชวลิตมงคล 44010516



..... อาจารย์ที่ปรึกษา  
(ดร.พิพัฒน์ พรหมมี)



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ระบบโทรศัพท์ภายในผ่านสายไฟฟ้า AC  
AC-LINE INTERPHONE SYSTEM

โดย นายสรศักดิ์ เจริญวัย 44010512  
นางสาวสโรชา เกตุศรีเมฆ 44010515  
นางสาวสรวส ชวลิตมงคล 44010516

อาจารย์ที่ปรึกษา ดร.พิพัฒน์ พรหมมี

**บทคัดย่อ**

โครงการนี้นำเสนอการออกแบบระบบโทรศัพท์ภายใน ซึ่งส่งสัญญาณผ่านสายไฟฟ้า AC โดยใช้ ไมโครคอนโทรลเลอร์ในการควบคุมการทำงาน จะมีส่วนของสัญญาณโทรศัพท์และสัญญาณเสียง ในส่วนของสัญญาณโทรศัพท์นั้นจะถูกสร้างโดยใช้โปรโตคอล ที่กำหนดขึ้นเพื่อใช้ในการติดต่อและสร้างช่องสัญญาณเสียง

**ABSTRACT**

This project presents a design of AC-line interphone system based on microcontroller. It consists of signaling and speech path subsystems. The signaling is designed by proprietary protocol for different signaling communicated and speech path connections.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## สารบัญ

	หน้า
บทที่ 1 บทนำ	1
บทที่ 2 ทฤษฎีและหลักการ	2
2.1 ความรู้เบื้องต้นเกี่ยวกับโทรศัพท์	2
2.1.1 ความหมายของสัญญาณที่ส่งมาจากชุมสาย	2
2.1.2 ระบบสัญญาณควบคุม	4
2.2 การส่งสัญญาณดิจิทัลที่ใช้คลื่นพาห์	6
2.3 Protocol	8
2.3.1 Synchronous Protocol	9
2.3.2 HDLC (High-Level Data Link Control)	9
2.3.2.1 HDLC Control Byte	9
2.4 สถาปัตยกรรมของไมโครคอนโทรลเลอร์แบบชิพเดี่ยวตระกูล MCS-51	11
2.4.1 สถาปัตยกรรมของ MCS-51	12
2.5 การแปลงสัญญาณอนาลอกเป็นสัญญาณดิจิทัลและแปลงสัญญาณดิจิทัลเป็นสัญญาณอนาลอก	14
2.6 สายไฟ AC	14
2.6.1 เอชซีคัปปลิง (AC Coupling)	16
บทที่ 3 วงจรและการทำงานของวงจร	17
3.1 วงจรปรีแอมป์	18
3.2 วงจรแปลงสัญญาณอนาลอกเป็นสัญญาณดิจิทัลและแปลงสัญญาณดิจิทัลเป็นสัญญาณอนาลอก	18
3.3 วงจรสร้างสัญญาณนาฬิกา	19
3.4 วงจรมอดูเลเตอร์และดีมอดูเลเตอร์แบบ FSK	20
3.4.1 การออกแบบวงจรมอดูเลเตอร์แบบ FSK	20
3.4.2 การออกแบบวงจรดีมอดูเลเตอร์แบบ FSK	22
3.5 วงจรกรองแถบความถี่ผ่าน	25
3.6 วงจรเอชซีคัปปลิง	28
3.7 บัฟเฟอร์ 3 สถานะ (Tri-State Buffer)	29
3.8 วงจรสวิตชิง (Switching)	30
3.9 วงจรควบคุม	31
3.9.1 โพรโทคอล HDLC	31
3.9.2 สัญญาณโทรศัพท์	34

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## สารบัญ (ต่อ)

	หน้า
3.10 วงจรคีย์แพด (Key pad)	35
3.11 วงจรภาคจ่ายไฟ (Power Supply)	35
<b>บทที่ 4 การทดลองและผลการทดลอง</b>	<b>37</b>
4.1 วงจรสร้างสัญญาณนาฬิกา	37
4.2 วงจรแปลงสัญญาณอนาลอกเป็นสัญญาณดิจิทัลและแปลงสัญญาณดิจิทัลเป็นสัญญาณอนาลอก	37
4.2.1 วงจรแปลงสัญญาณอนาลอกเป็นสัญญาณดิจิทัล (ด้านส่ง)	37
4.2.2 วงจรแปลงสัญญาณดิจิทัลเป็นสัญญาณอนาลอก (ด้านรับ)	38
4.3 วงจรมอดูเลเตอร์และวงจรมอดูเลเตอร์แบบ FSK	39
4.3.1 วงจรมอดูเลเตอร์แบบ FSK	39
4.3.2 วงจรดีมอดูเลเตอร์แบบ FSK	39
4.4 วงจรกรองแถบความถี่ผ่าน	40
4.5 วงจรเอซซึคัปปลิ่ง	41
4.5.1 วงจรเอซซึคัปปลิ่ง (ด้านส่ง)	41
4.5.2 วงจรเอซซึคัปปลิ่ง (ด้านรับ)	42
<b>บทที่ 5 บทวิจารณ์ และ บทสรุป</b>	<b>43</b>
<b>ภาคผนวก</b>	
<b>บรรณานุกรม</b>	

## สารบัญรูปภาพ

รูปที่	หน้า
รูปที่ 2.1 สัญญาณพื้นฐานของโทรศัพท์	3
รูปที่ 2.2 ลำดับการติดต่อของโทรศัพท์จากคู่สายหนึ่งไปอีกคู่สายหนึ่ง	5
รูปที่ 2.3 สัญญาณเบสแบนด์ดิจิทัล	7
รูปที่ 2.4 Amplitude Shift Keying	7
รูปที่ 2.5 Frequency Shift Keying	7
รูปที่ 2.6 Phase Shift Keying	8
รูปที่ 2.7 HDLC frame	9
รูปที่ 2.8 I-Frame	9
รูปที่ 2.9 S-Frame	10
รูปที่ 2.10 U-Frame	10
รูปที่ 2.11 ไดอะแกรมขาของ MCS-51 แบบ DIP	12
รูปที่ 2.12 สัญญาณ CVSD Waveform	14
รูปที่ 2.13 บล็อกไดอะแกรมของ CVSD	14
รูปที่ 3.1 บล็อกไดอะแกรมของระบบโทรศัพท์ภายในผ่านสายไฟฟ้า AC	17
รูปที่ 3.2 วงจรปริโมค	18
รูปที่ 3.3 วงจรเพาเวอร์แอมป์	18
รูปที่ 3.4 วงจรแปลงสัญญาณอนาลอกเป็นสัญญาณดิจิทัลและแปลงสัญญาณดิจิทัลเป็นสัญญาณอนาลอก	19
รูปที่ 3.5 วงจรสร้างสัญญาณนาฬิกา	19
รูปที่ 3.6 วงจรมอดูเลเตอร์แบบ FSK ของความถี่ $f_{m_1} = 15kHz$ และ $f_{s_1} = 30kHz$	21
รูปที่ 3.7 วงจรมอดูเลเตอร์แบบ FSK ของความถี่ $f_{m_2} = 55kHz$ และ $f_{s_2} = 70kHz$	21
รูปที่ 3.8 วงจรดีมอดูเลเตอร์แบบ FSK ของความถี่ $f_{m_1} = 15kHz$ และ $f_{s_1} = 30kHz$	25
รูปที่ 3.9 วงจรดีมอดูเลเตอร์แบบ FSK ของความถี่ $f_{m_2} = 55kHz$ และ $f_{s_2} = 70kHz$	25
รูปที่ 3.10 วงจรกรองแถบความถี่ผ่านแบบ Sallen and Key	26
รูปที่ 3.11 วงจรกรองแถบความถี่ผ่านแบบ Sallen and Key ลำดับที่ 6 ของความถี่ 15 – 30kHz	28
รูปที่ 3.12 วงจรกรองแถบความถี่ผ่านแบบ Sallen and Key ลำดับที่ 6 ของความถี่ 55 – 70kHz	28
รูปที่ 3.13 วงจรเอซิคัปปลิงสำหรับด้านส่ง	29
รูปที่ 3.14 วงจรเอซิคัปปลิงสำหรับด้านรับ	29
รูปที่ 3.15 ไดอะแกรมขาของบัฟเฟอร์ 3 สถานะ (74 SL244)	30

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## สารบัญรูปภาพ (ต่อ)

รูปที่	หน้า
รูปที่ 3.16 ลอจิกไดอะแกรมของบัพเฟอร์ 3 สถานะ (74 SL244)	30
รูปที่ 3.17 วงจรสวิตชิง	31
รูปที่ 3.18 ขั้นตอนการติดต่อระหว่างผู้เรียกและผู้ถูกเรียก	33
รูปที่ 3.19 ขั้นตอนการยกเลิกการติดต่อระหว่างต้นทางและปลายทาง	34
รูปที่ 3.20 วงจรคีย์แพด	35
รูปที่ 3.21 วงจรภาคจ่ายไฟ	36
รูปที่ 4.1 สัญญาณเอาต์พุตจากวงจรสร้างสัญญาณนาฬิกาที่มีความถี่ 16 kHz	37
รูปที่ 4.2 สัญญาณเอาต์พุต CVSD ด้านส่งเทียบกับอินพุตสัญญาณชาयน์ ความถี่ 1 kHz	38
รูปที่ 4.3 สัญญาณเอาต์พุต CVSD ด้านรับเทียบกับอินพุตสัญญาณชาयน์ ความถี่ 1 kHz ที่ด้านส่ง	38
รูปที่ 4.4 สัญญาณเอาต์พุตของวงจรมอดูเลเตอร์แบบ FSK เทียบกับสัญญาณอินพุต โดยที่อินพุตที่ด้านส่งเป็นสัญญาณชาयน์ ความถี่ 1 kHz	39
รูปที่ 4.5 สัญญาณเอาต์พุตของวงจรมอดูเลเตอร์แบบ FSK เทียบกับสัญญาณอินพุต โดยที่อินพุตที่ด้านส่งเป็นสัญญาณชาयน์ ความถี่ 1 kHz	39
รูปที่ 4.6 สัญญาณเอาต์พุตของวงจรกรองแถบความถี่ผ่านเทียบกับสัญญาณอินพุต โดยที่อินพุตที่ด้านส่งเป็นสัญญาณชาयน์ ความถี่ 1 kHz	40
รูปที่ 4.7 สัญญาณก่อนป้อนอินพุต	41
รูปที่ 4.8 สัญญาณหลังป้อนอินพุต โดยอินพุตที่ด้านส่งเป็นสัญญาณชาयน์ ความถี่ 1 kHz	41
รูปที่ 4.9 สัญญาณก่อนเข้าตัวเก็บประจุของวงจรเอซีคัปปลิง (ด้านรับ) โดยที่อินพุตที่ด้านส่งเป็นสัญญาณชาयน์ ความถี่ 1 kHz	42
รูปที่ 4.10 สัญญาณหลังเข้าตัวเก็บประจุของวงจรเอซีคัปปลิง (ด้านรับ) โดยที่อินพุตที่ด้านส่งเป็นสัญญาณชาयน์ ความถี่ 1 kHz	42

## บทที่ 1

### บทนำ

เทคโนโลยีแห่งโลกาภิวัตน์อย่างในยุคปัจจุบันนี้ มักเปลี่ยนแปลงพัฒนาก้าวไปข้างหน้าอย่างรวดเร็ว นวัตกรรมแต่ละชิ้นล้วนแต่มีแรงดึงดูดให้ค้นหาที่มาของการพัฒนาและประโยชน์ในการใช้งานด้วยกันทั้งสิ้น

ถ้าเรากล่าวถึงเทคโนโลยีเครือข่าย จะพบว่าทุกวันนี้มีสิ่งใหม่ๆ แข่งขันกันออกมาอำนวยความสะดวกให้เราอยู่ตลอดเวลา ไม่ว่าจะเป็นมาตรฐานการเชื่อมต่อที่เร็วขึ้น ปริมาณการรับ - ส่งข้อมูลสูงขึ้น อุปกรณ์แต่ละชิ้นได้รับการออกแบบให้สวยงามและมีขนาดเล็กลง เป็นต้น

สำหรับโครงการนี้ อาจจะไม่ใช้เทคโนโลยีใหม่ล่าสุด เนื่องจากเคยมีการนำเสนอและใช้งานจริงมาก่อนหน้านี้แล้วในต่างประเทศ แต่สำหรับในเมืองไทยแนวโน้มของเทคโนโลยี พาวเวอร์ไลน์ เน็ตเวิร์ก (Power Line Network) หรือการเชื่อมต่อด้วยสายไฟฟ้าในบ้านกำลังได้รับความสนใจจากผู้ใช้งานมากขึ้นทุกวัน พาวเวอร์ไลน์ เน็ตเวิร์ก (Power Line Network) คือ การเชื่อมต่อด้วยสายไฟฟ้าในบ้าน เปลี่ยนระบบไฟฟ้าภายในบ้านหรือสำนักงานขนาดเล็ก ให้สามารถเชื่อมต่อระบบการสื่อสารทางโทรศัพท์ภายใน โดยไม่จำเป็นต้องเดินสายโทรศัพท์ใหม่ เพราะสามารถใช้ปลั๊กไฟที่มีอยู่เป็นจุดเชื่อมต่อได้ ที่สำคัญราคาของโซลูชันลักษณะนี้ไม่แพง ยิ่งระบบเครือข่ายทวีบทบาทในชีวิตประจำวันและการทำงานมากขึ้นทุกวัน ไม่ว่าจะทำอะไรล้วนแล้วแต่ต้องอาศัยเครือข่ายทั้งมีสายและไร้สายทั้งสิ้น โดยขึ้นอยู่กับรูปแบบการใช้งานเป็นหลัก หลายคนอาจคิดว่าเรื่องเกี่ยวกับระบบเครือข่ายน่าจะอยู่ที่ทำงานมากกว่าแต่เดี๋ยวนี้คนไทยจำนวนไม่น้อยได้ติดตั้งระบบเครือข่ายขนาดเล็กอย่างแลน (Local Area Network) ไว้ในบ้านแล้ว ผนวกกับบริการล่าสุดของผู้ให้บริการต่อเชื่อมระบบเครือข่าย และโอเปอเรเตอร์โทรศัพท์ที่มีคุณภาพดีขึ้นกว่าเดิม แคมราคาถูกลงอีก ส่งผลให้เกิดความนิยมในการใช้บริการลักษณะนี้เพิ่มมากขึ้น แต่ปัญหาอย่างหนึ่งของการติดตั้งระบบเครือข่ายภายในบ้านคือ โครงสร้างของระบบฯ ที่ผ่านมายังมีปัญหา การเชื่อมต่ออุปกรณ์ต่างๆ จำเป็นต้องมีสายสัญญาณแยกต่างหากจากสายโทรศัพท์ทำให้ยุ่งยากในการติดตั้ง ผิดกับวันนี้ที่บริษัทผู้ผลิตอุปกรณ์ระบบเครือข่ายชั้นนำทั้งอเมริกาและไต้หวัน ได้พัฒนาอุปกรณ์รุ่นใหม่ๆ ให้สามารถแก้ปัญหาที่เกิดขึ้นได้ ส่วนใหญ่จะออกมาในรูปของการเลือกใช้เทคโนโลยีไร้สาย (Wireless Technology) มาทดแทนซึ่งก็แก้ปัญหาได้ในระดับหนึ่งซึ่งวิธีนี้จะเหมาะกับผู้ที่สามารถรับค่าใช้จ่ายที่เกิดขึ้นจากการใช้งานระบบแลนไร้สายได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## บทที่ 2

### ทฤษฎีและหลักการ

#### 2.1 ความรู้เบื้องต้นเกี่ยวกับโทรศัพท์

โทรศัพท์ที่ใช้กันทั่วๆ ไปมีอยู่ 2 แบบ คือแบบหมุน (Rotating type) ซึ่งเป็นระบบที่ใช้กันดั้งเดิม ตั้งแต่แรกแต่ปัจจุบันกำลังเลิกใช้แล้ว ยังมีใช้งานอยู่บ้างแต่ก็เป็นเพียงส่วนน้อยเท่านั้น และแบบสัญญาณความถี่คู่ (Dual Tone Multi-Frequency type) หรือเรียกอีกอย่างหนึ่งว่าระบบกดปุ่ม ปัจจุบันระบบกดปุ่มเป็นที่นิยมมาก หน้าที่ของโทรศัพท์ทั้ง 2 ระบบจะมีลักษณะเหมือนกัน จะต่างกันตรงที่ระบบกดปุ่มจะส่งสัญญาณออกไปเป็นความถี่ที่แตกต่างกัน 2 ความถี่ ส่วนระบบหมุนจะส่งสัญญาณเป็นจำนวนพัลส์

##### 2.1.1 ความหมายของสัญญาณที่ส่งมาจากชุมสายมอดูเลต ด้วย 50 เฮิรตซ์ ความถี่ 425 เฮิรตซ์

###### - Dial tone

เป็นสัญญาณที่บอกให้ทราบว่า ขณะนี้อุปกรณ์ที่ชุมสายพร้อมจะรับได้คือการหมุนเลขหมายจากผู้เรียก ให้ผู้เรียกทำการส่งเลขหมายได้ สัญญาณ Dial tone เป็นสัญญาณต่อเนื่องความถี่ 425 Hz มอดูเลตด้วย 50 Hz ผู้เช่าจะได้ยินโทนนี้ เมื่อทำการยกหูโทรศัพท์เพื่อทำการเรียก

###### - Busy tone

เป็นสัญญาณที่ส่งมาบอกให้ทราบว่าอุปกรณ์ไม่ว่าง เช่น ผู้เช่ายกหูแล้วได้ยินเสียงนี้แทนที่จะได้ยิน Dial tone แสดงว่าอุปกรณ์ ในชุมสายโทรศัพท์ไม่ว่าง แต่ถ้าได้ยินเสียงนี้หลังจากหมุนเลขหมายไปแล้วแสดงว่าผู้เช่าฝ่ายถูกเรียกไม่ว่างหรืออุปกรณ์สำหรับต่อออกไปยังชุมสายอื่นไม่ว่าง ในกรณีที่ผู้ถูกเรียกอยู่ต่างชุมสายสัญญาณที่ขาดตอนเป็นช่วงๆ ส่ง 0.5 วินาที หยุด 0.5 วินาที ความถี่ของสัญญาณ Sine Wave 425 Hz ภายในระยะเวลา 45 วินาที แล้วเกิดสภาวะ Line – Lock – Line (เจียบ)

###### - Ring back tone

เป็นสัญญาณที่ผู้เรียกได้ยินเสียงจากชุมสายเลขหมายครบแล้ว เพื่อบอกให้ทราบว่าคนที่ได้สำเร็จ ขณะนี้ชุมสายได้ส่งสัญญาณเรียก Ring back tone ไปยังผู้ถูกเรียกสัญญาณใช้ความถี่ 425 Hz Sine Wave โดยจะส่ง 1 วินาที หยุด 4 วินาที

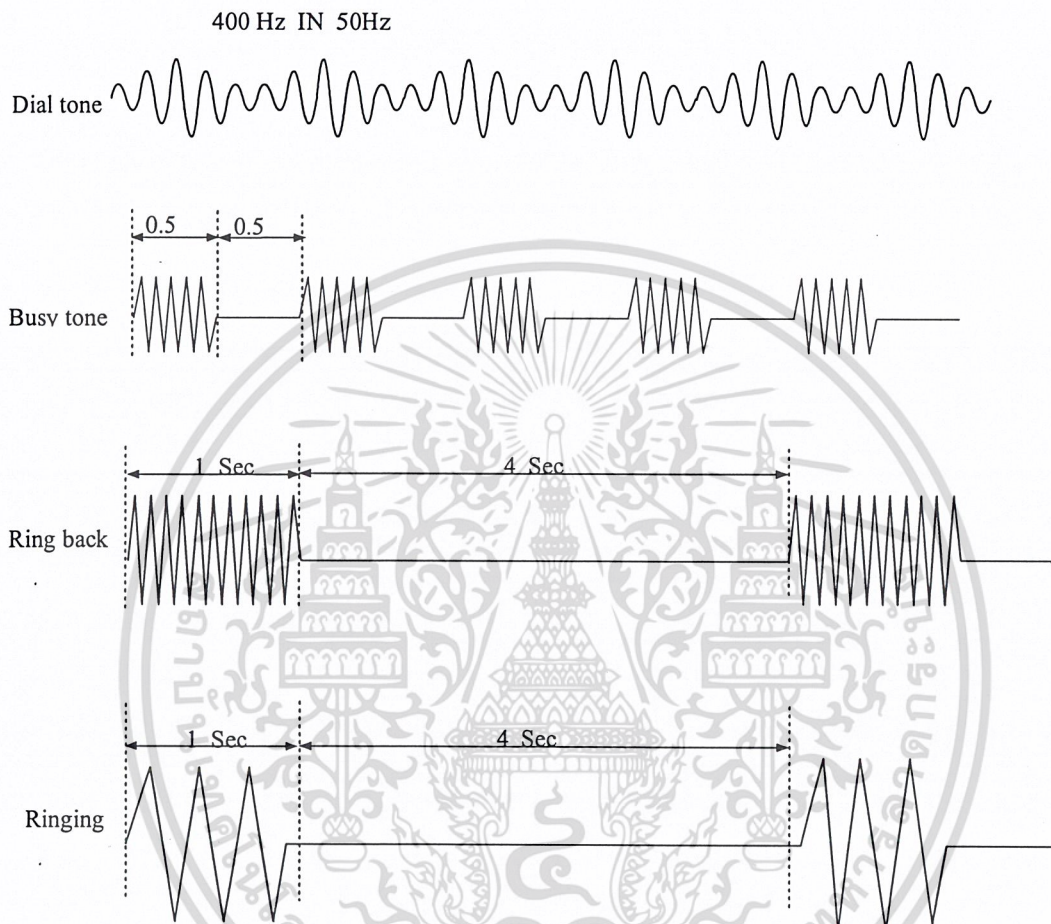
###### - Ringing tone

เป็นสัญญาณเรียกที่ส่งไปยังผู้เช่า ฝ่ายถูกเรียกซึ่งจะได้ยินเสียงกระดิ่งหรือโทน ขึ้นอยู่กับวงจรที่ใช้สัญญาณเป็น Sine Wave 25 Hz ค่าแรงดัน 70-90  $V_{rms}$  ช่วงการส่งเป็นแบบเดียวกับ Ring back tone คือส่ง 1 วินาที หยุด 4 วินาที เป็นระยะเวลายาวนานทั้งสิ้น 70-90 วินาที

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- Nu tone (Number Unobtainable tone)

บอกให้ทราบตัวเลขหมายที่หมุนมาไม่มีใช้งานอยู่ สัญญาณตอบจากเครื่องบันทึกบอกให้ทราบ  
ว่าเลขหมายนี้ได้เปลี่ยนไปแล้ว



รูปที่ 2.1 สัญญาณพื้นฐานของโทรศัพท์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

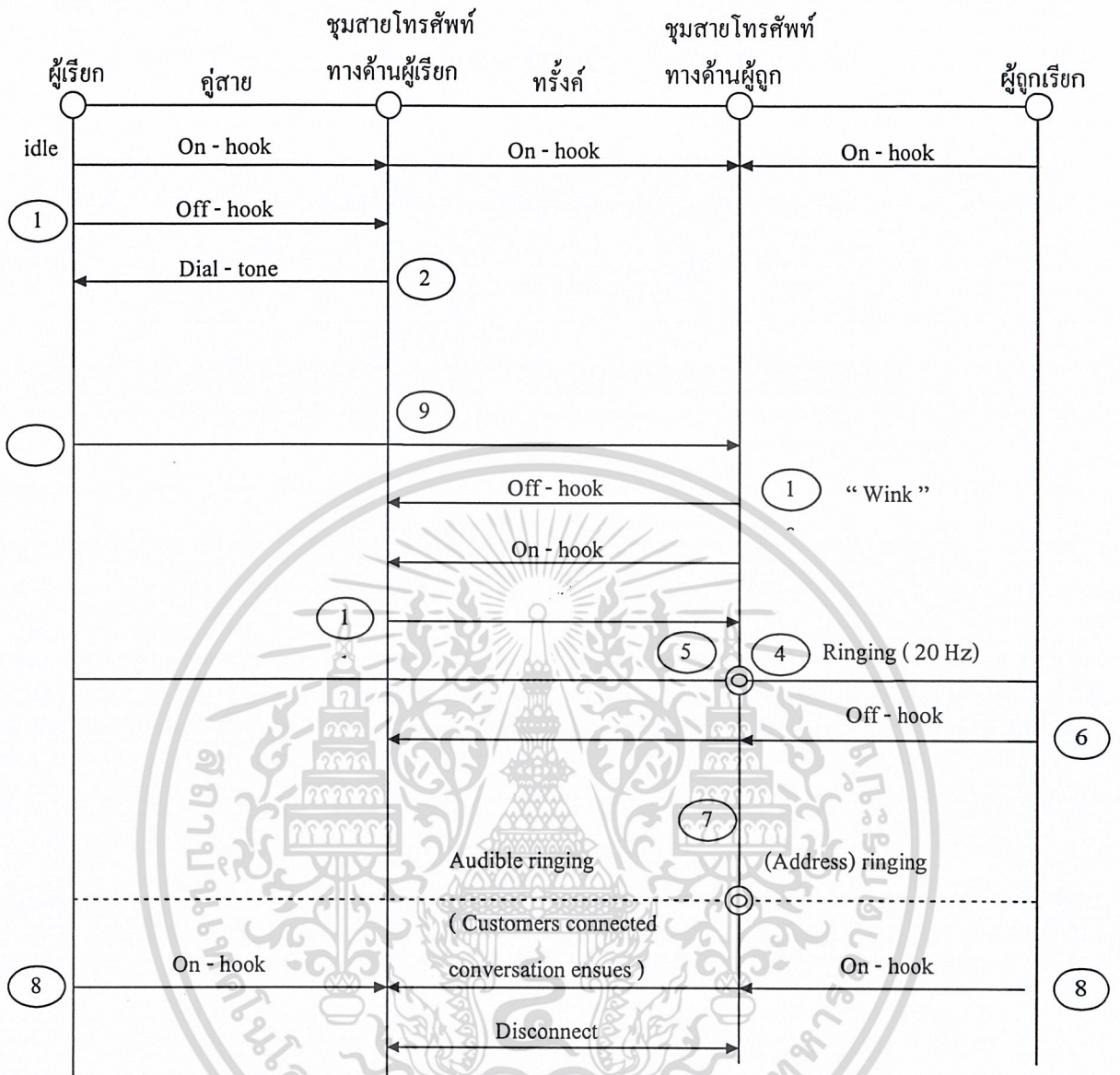
## 2.1.2 ระบบสัญญาณควบคุม

ในโครงข่ายแบบเซอร์กิตสวิตซึ่งสัญญาณควบคุมนั้นหมายถึง การจัดการโครงข่ายและสถาปนา การเรียกจากต้นทางไปยังปลายทาง และจะคงสภาพของเส้นทางนั้นไว้จนกระทั่งสิ้นสุดการติดต่อการสื่อสาร การจัดการโครงข่ายทั้งหมดจะต้องมีการถ่ายโอนข้อมูล จากศูนย์กลางในการจัดการโครงข่ายไปยัง สวิตช์ต่างๆ สำหรับในโครงข่ายโทรคมนาคมขนาดใหญ่ ก็จะต้องมีการใช้สัญญาณควบคุมที่ค่อนข้างจะ ซับซ้อน ซึ่งในโครงข่ายรวมดิจิทัลแบบใหม่จะมีการใช้สัญญาณควบคุมร่วม ( Common Channel Signaling )

### ฟังก์ชันของสัญญาณต่างๆ

สัญญาณควบคุมมีความจำเป็นสำหรับการทำงานของโครงข่ายเซอร์กิตสวิตซึ่ง และการร้องขอ ต่างๆที่เกิดขึ้นในโครงข่ายรวมทั้งการให้บริการในโครงข่ายที่ให้บริการของทั้งสองส่วน(ต้นทางและปลายทาง) ซึ่งจะพิจารณาจากผู้ใช้และพิจารณาจากกลไกภายในโครงข่าย ทำให้โครงข่ายจะมีความซับซ้อนมาก จำนวนของฟังก์ชันที่ดำเนินการ โดยสัญญาณควบคุมจึงต้องมีลักษณะที่มากขึ้นด้วย ซึ่งจะต้องมีฟังก์ชัน ต่างๆ ดังนี้

1. การติดต่อสื่อสารที่ผู้เข้าสามารถรับรู้ได้ ซึ่งจะประกอบด้วยสัญญาณหมุนหมายเลข สัญญาณ ไม่ว่าง และอื่นๆ
2. สัญญาณหมุนหมายเลขที่ส่งไปยังชุมสายโทรศัพท์ ชุมสายโทรศัพท์จะทำหน้าที่ในการเชื่อมต่อ ไปยังผู้ใช้ปลายทางให้สมบูรณ์
3. การส่งข้อมูลระหว่างสวิตช์จะเกิดขึ้น ในกรณีที่ไม่สามารถเชื่อมต่อจากเรียกได้สมบูรณ์เพื่อใช้ในการนำไปในการกำหนดเส้นทางของการส่งข้อมูลใหม่
4. ส่งสัญญาณที่ทำให้กระดิ่งโทรศัพท์ดัง เมื่อมีการเรียกเข้า
5. การส่งข้อมูลที่ใช้ในการคำนวณคิดเงิน
6. การส่งข้อมูลที่ชี้แสดงถึงสถานะของอุปกรณ์หรือในโครงข่าย ข้อมูลนี้จะถูกนำมาใช้เพื่อ จุดประสงค์ในการกำหนดเส้นทางและใช้ในการบำรุงรักษา
7. การส่งข้อมูลที่ใช้ในการวิจัยและแยกแยะระบบที่เสียหายมาก
8. ใช้ในการควบคุมอุปกรณ์ควบคุม จะสามารถแสดงได้ดังรูปที่ 2.2 ซึ่งแสดงลำดับการติดต่อ ของโทรศัพท์จากคู่สายหนึ่งไปยังอีกคู่สายหนึ่งในชุมสายโทรศัพท์เดียวกัน



รูปที่ 2.2 ลำดับการติดต่อของโทรศัพท์จากคู่สายหนึ่งไปอีกคู่สายหนึ่ง

เราสามารถอธิบายลำดับการติดต่อของโทรศัพท์ เป็นขั้นตอนต่างๆ ตามหมายเลขได้ดังนี้

1. ก่อนที่จะมีการเรียกโทรศัพท์ทั้งคู่ยังไม่ถูกใช้งาน (on-hook) การเริ่มต้นการเรียกจะเริ่มต้นเมื่อผู้ใช้รายหนึ่งยกแฮนด์เซตขึ้น (off-hook) ซึ่งจะมีการส่งสัญญาณไปที่ชุมสายโดยอัตโนมัติ
2. ทางชุมสายตอบรับด้วยการส่งสัญญาณ ไดอัล โทน ซึ่งผู้ใช้สามารถได้ยินและเป็นการบอกให้ผู้ใช้ทราบว่าจะสามารถหมุนหมายเลขได้แล้ว
3. ผู้ใช้จะหมุนหมายเลขที่ต้องการจะติดต่อด้วยไปยังชุมสาย
4. ถ้าหมายเลขของผู้ใช้วางชุมสายจะเตือน ไปยังผู้ที่ถูกเรียกโดยการส่งสัญญาณริงกิ้งทำให้กระดิ่งดังขึ้น
5. จะมีการส่งสัญญาณย้อนกลับไปยังผู้เรียก โดยชุมสายดังนี้

5.1 ถ้าผู้ถูกเรียกวาง ชุมสายจะส่งสัญญาณ ริงแบ็ค โทน (Ring back tone) กลับไปยังผู้เรียกในขณะที่ส่งสัญญาณริงกิ้ง ไปยังผู้ถูกเรียก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5.2 ถ้าผู้ถูกเรียกไม่ว่าง หูสายจะส่งสัญญาณไม่ว่าง (Busy tone) กลับไปยังผู้เรียก

5.3 ถ้าการเรียกผ่านทางหูสายไม่สมบูรณ์ หูสายก็จะส่งสัญญาณเสียงที่บันทึกไว้ไปยังผู้เรียก

6. เมื่อผู้ถูกเรียกยกแฮนด์เซตขึ้น (off-hook) จะมีส่งสัญญาณโดยอัตโนมัติไปยังหูสาย

7. หูสายจะหยุดส่งสัญญาณริงกิงและริงแบล็คโทน หลังจากนั้นก็จะทำการเชื่อมต่อระหว่าง 2 ผู้ใช้

8. การเชื่อมต่อจะสิ้นสุดลงเมื่อผู้ใช้รายใดรายหนึ่งวางแฮนด์เซตลง เมื่อผู้ถูกเรียกต่ออยู่กับหูสายต่างกับกับผู้เรียก จึงต้องมีการสวิตช์จากหูสายหนึ่งไปยังหูสายหนึ่ง ซึ่งจะต้องมีการส่งสัญญาณระหว่างทรีนค์

9. หูสายของผู้ถูกเรียกจะส่งสัญญาณไปที่ทรีนค์ระหว่างหูสายที่ว่างจะส่งสัญญาณแสดงการออฟฮุคลงไปยังทรีนค์และร้องขอคิวิตริจิสเตอร์ของปลาย

10. หูสายทางผู้ถูกเรียกจะส่งสัญญาณออฟ-ฮุคหรือที่รู้จักกันในชื่อของวิงก์ (wink) ซึ่งแสดงถึงสถานะของการอ่านค่าในรีจิสเตอร์

11. หูสายของผู้เรียกจะส่งหมายเลขปลายทางที่ต้องการติดต่อไปยังหูสายของผู้ถูกเรียก

## 2.2 การส่งสัญญาณดิจิทัลที่ใช้คลื่นพาห์

เนื่องจากสัญญาณดิจิทัลในแบนด์มูลฐาน มีองค์ประกอบของสัญญาณที่มีความถี่ต่ำอยู่เป็นส่วนใหญ่ จึงเหมาะสมที่จะใช้กับระบบสื่อสารที่ใช้สายโดยตรง แต่ไม่เหมาะสมที่จะใช้ส่งผ่านระบบที่ไม่ยอมให้ความถี่ต่ำผ่านไปได้ เช่น ระบบโทรศัพท์ที่จะต้องผ่านหูสายและไม่เหมาะกับการส่งสัญญาณนั้นโดยตรงด้วยคลื่นในย่านความถี่วิทยุเป็นต้น จึงจำเป็นต้องใช้การมอดูเลตมาช่วยเพื่อย้ายสเปกตรัมของสัญญาณในแบนด์มูลฐานให้สูงขึ้นให้เหมาะสมกับการใช้งานนั้น ดังนั้นจะกล่าวเกี่ยวกับรูปแบบของการส่งสัญญาณดิจิทัลที่ใช้คลื่นพาห์ไว้เป็นสังเขปการส่งสัญญาณดิจิทัล เช่น สัญญาณพีซีเอ็ม โดยคลื่นวิทยุไมโครเวฟจำเป็นที่จะต้องเปลี่ยนสัญญาณดิจิทัลไปเป็นสัญญาณในย่านความถี่วิทยุ

เทคนิคในการมอดูเลตสัญญาณดิจิทัลนี้มี 3 แบบด้วยกันคือ

1. Amplitude Shift Keying (ASK)

2. Frequency Shift Keying (FSK)

3. Phase Shift Keying (PSK)

คลื่นพาห์ในย่านความถี่วิทยุหรือไมโครเวฟสามารถกำหนดได้โดย

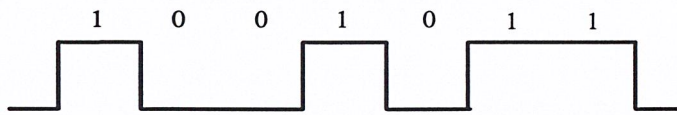
$$\text{คลื่นพาห์} = A \cos(2\pi f_c t + \sigma)$$

ในที่นี้  $A$  : แอมพลิจูดของคลื่นพาห์

$f_c$  : ความถี่คลื่นพาห์

$\sigma$  : Initial phase

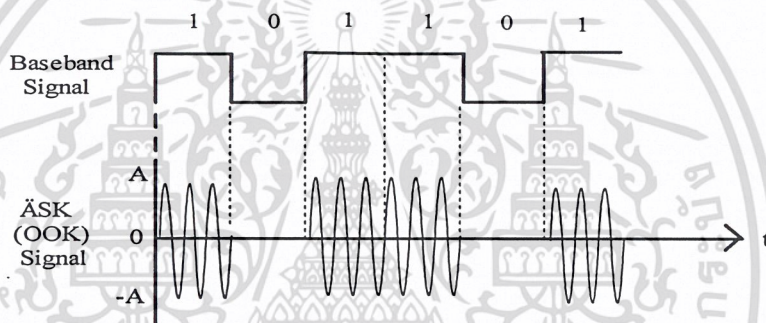
สัญญาณดิจิทัลเบสแบนด์เป็นคลื่นรูปสามเหลี่ยมแสดงรหัสไบนารี 1 และ 0 ในการมอดูเลตสัญญาณดิจิทัลนี้ หนึ่งในสามพารามิเตอร์คือ แอมพลิจูด ความถี่หรือเฟสของคลื่นพาห์จะเปลี่ยนไปตามสถานะ 1 หรือ 0 ของสัญญาณเบสแบนด์



รูปที่ 2.3 สัญญาณเบสแบนด์ดิจิทัล

กรณีของ ASK บางครั้งเรียกว่า OOF (On-Off Keying) เพราะว่าการสลับพลาถูกสวิตซ์ on/off ตามสัญญาณที่เป็น 1 หรือ 0 ถ้าคลื่นพลาที่กำหนดโดย  $A \cos 2P_i f_c t$  สัญญาณ ASK จะกำหนดได้เป็น

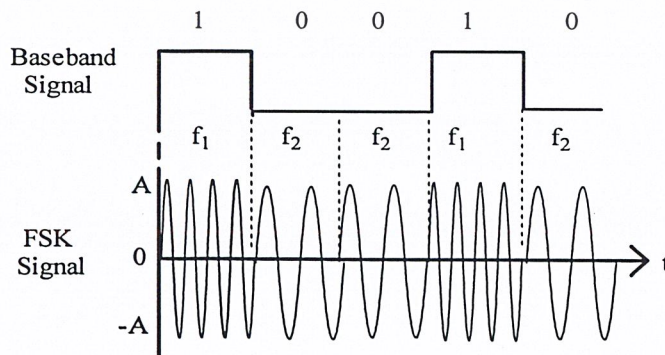
$$S(t) = \begin{cases} A \cos 2P_i f_c t & \text{เมื่อสัญญาณเป็น 1} \\ 0 & \text{เมื่อสัญญาณเป็น 0} \end{cases}$$



รูปที่ 2.4 Amplitude Shift Keying

สำหรับกรณีของ FSK ความถี่ของคลื่นพลาจะมี 2 ความถี่  $f_1$  สำหรับสัญญาณที่เป็น 1 และ ความถี่  $f_2$  สำหรับสัญญาณที่เป็น 0

$$S(t) = \begin{cases} A \cos 2P_i f_1 t & \text{เมื่อสัญญาณเป็น 1} \\ A \cos 2P_i f_2 t & \text{เมื่อสัญญาณเป็น 0} \end{cases}$$



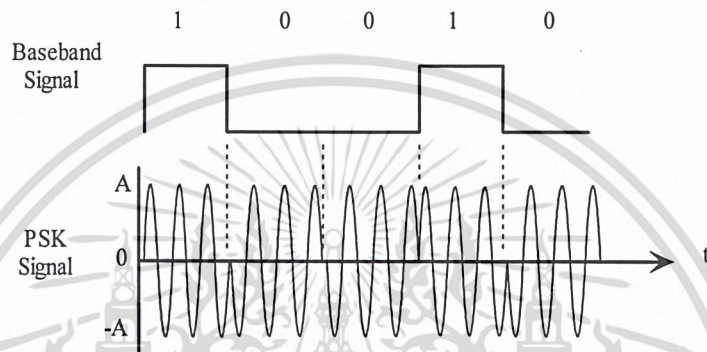
รูปที่ 2.5 Frequency Shift Keying

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ส่วนในกรณีของ PSK แอมพลิจูดและความถี่จะคงที่แต่เฟสเริ่มต้น (initial phase) จะต่างกัน สำหรับสัญญาณที่เป็น 1 หรือ 0 คือเฟสของคลื่นพาห้  $A\cos(2P_i f_c t + \sigma)$  จะเปลี่ยนไปตามสถานะของสัญญาณเช่น เฟส  $\sigma$  เป็น 0 กรณีสัญญาณมีสถานะเป็น 1 และเฟส  $\sigma$  เป็น  $P_i$  กรณีสัญญาณมีสถานะเป็น 0

$$S(t) = A\cos(2P_i f_c t) \quad \text{เมื่อสัญญาณเป็น 1}$$

$$S(t) = A\cos(2P_i f_c t + P_i) \quad \text{เมื่อสัญญาณเป็น 0}$$



รูปที่ 2.6 Phase Shift Keying

สำหรับการสื่อสารผ่านดาวเทียม PSK เป็นแบบที่เหมาะสมที่สุดสำหรับการสื่อสารผ่านดาวเทียมแบบดิจิทัลเมื่อเทียบกับแบบอื่นๆ

### 2.3 Protocol

การติดต่อสื่อสารระหว่างปลายทางทั้ง 2 ด้านนั้น จะต้องมีการโปรโตคอลซึ่งถูกกำหนดขึ้นมาในลักษณะของกฎข้อตกลงที่ใช้ในการโต้ตอบรับ-ส่งข้อมูลระหว่างกัน เพื่อให้ปลายทางทั้ง 2 สามารถติดต่อสื่อสารกันได้และรับ-ส่งข้อมูลได้อย่างถูกต้อง ในระดับ Data link โปรโตคอลจะถูกนำมาใช้เพื่อควบคุมการสื่อสารแบบ point-to-point หรือ point-to-multipoint

องค์ประกอบที่สำคัญของโปรโตคอล คือ

1. Syntax คือ รูปแบบของข้อมูลและระดับสัญญาณ
2. Semantics คือ ข้อมูลที่ใช้ในการควบคุมทั้งทางด้านรับด้านส่งและการจัดการเกี่ยวกับข้อผิดพลาด
3. Timing คือ ความเร็วที่สัมพันธ์กันของด้านรับด้านส่งและลำดับของการจัดการ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 2.3.1 Synchronous Protocol

มีลักษณะดังนี้คือ มีการส่งข้อมูลเป็น Block ต่อเนื่อง ในกรณีทั่วไปจะมีความยาวของ Block ไม่เกิน 1 - 2 กิโลไบต์ มีคุณสมบัติหลักๆ ดังนี้คือ

1. ใช้สัญญาณนาฬิกาพร้อมกัน ทั้งทางด้านส่งด้านรับและด้านส่ง
2. มีการส่งข้อมูลเป็น Block ยาวๆ
3. มีการบอกจุดเริ่มต้นและจุดจบของเฟรม
4. มีการตรวจสอบความผิดพลาด
5. มีการร้องขอให้ส่งข้อมูลใหม่เมื่อเกิดความผิดพลาดใน Block อย่างอัตโนมัติ

### 2.3.2 HDLC (High-Level Data Link Control)

HDLC แสดงไว้ในรูปที่ 2.7 ใช้ในมาตรฐานในการส่งข้อมูลแบบ Synchronous Protocol ซึ่งหมายถึงจะต้องมีการบอกหัวและท้ายของเฟรม (Flag) มีไบนารีของ Address ซึ่งสามารถเพิ่มจำนวนไบต์ข้อมูลในกรณีสื่อสารแบบ Multi-drop มีการร้องขอการส่งต่อแบบอัตโนมัติในกรณี Block ข้อมูลผิดพลาด (Control Byte) Block ข้อมูล (data) แบบโปร่งใสและมีการตรวจสอบความผิดพลาดของข้อมูลโดย Frame Check Sequence (FCS)

Flag	Address	Control	data	FCS	Flag
------	---------	---------	------	-----	------

รูปที่ 2.7 HDLC frame

จากรูปที่ 2.7 ไบนารีของ Flag จะต้องมีค่าคงตัวเท่ากับ “01111110” ซึ่งเป็นตัวบอกจุดเริ่มต้นและจุดสิ้นสุดของเฟรม โดยที่ในส่วนข้อมูลไม่มีลักษณะข้อมูลที่เหมือนกับ Flag เนื่องจากจะมีกระบวนการเติมบิต (Bit Stuffing) ที่ด้านส่งเมื่อพบว่ามีข้อมูลที่เป็น “1” ติดกัน 5 ตัว โดยจะแทรก “0” ลงไป 1 ตัว และทางด้านรับเมื่อพบว่ามีข้อมูลที่เป็น “1” ติดกัน 5 ตัว ก็จะตัดข้อมูลถัดไปออกไป 1 บิต

#### 2.3.2.1 HDLC Control Byte

แบ่งออกเป็น 3 ชนิดคือ

1. Information Frame (I-Frame) ใช้เมื่อมีการส่งข้อมูล

0	NS	P/F	NR
---	----	-----	----

รูปที่ 2.8 I-Frame

บิตแรกจะมีค่าเป็น “0”

NS : Number Sent ประกอบด้วย 3 บิต เพื่อบอกจำนวนเฟรมที่ส่งไปแล้ว (0-7) เมื่อนับถึง 7 ก็  
จะเริ่ม นับ 0 ใหม่

NR : Number Received ประกอบด้วย 3 บิต เพื่อบอกจำนวนเฟรมที่ได้รับมา เช่น ถ้าด้านส่ง  
ส่งเฟรม 0-3 มา ทางด้านรับจะตอบ NR = 4

P/F : Poll / Final Bit เป็นบิตที่ใช้เป็น Poll บิต ซึ่งใช้การร้องขอให้ส่ง Acknowledgment (ACK)  
กลับมา หรือใช้เป็น Final Bit ซึ่งเป็นการบอกว่าไม่มีข้อมูลที่จะส่งแล้ว

2. Number Supervisory Frame (S-Frame) ใช้ในการ ACK ข้อมูลที่ได้รับ

1	0	Cn	P/F	NR
---	---	----	-----	----

รูปที่ 2.9 S-Frame

2 บิตแรกจะมีค่าเป็น “10”

Cn : Numbered Supervisory Command ประกอบด้วย 2 บิต เป็นคำสั่งที่ใช้ในการตอบกลับซึ่งมี  
คำสั่งดังต่อไปนี้

- 1). Receiver Ready (RR) เป็นตัวบอกว่าเฟรมที่ได้รับมานั้นถูกต้อง ไม่มีความผิดพลาดแต่  
ไม่มีข้อมูลตอบกลับ
- 2). Receiver Not Ready (RNR) เป็นตัวบอกว่าเฟรมที่ได้รับมานั้นไม่มีความผิดพลาด แต่เป็น  
การบอกให้ด้านส่งรอซักครู่เนื่องจาก Buffer เต็ม
- 3). Reject (REJ) เป็นการเริ่มต้นกระบวนการส่งใหม่แบบ Go-Back-n เนื่องจากมีความผิด  
พลาดเกิดขึ้น
- 4). Select Reject (SREJ) เป็นการ Reject เฉพาะเฟรมที่ n ให้ส่งเฟรม n ใหม่ เนื่องจากมี  
ความผิดพลาดเกิดขึ้น NR และ P/F มีลักษณะเดียวกับที่ใช้ใน I-Frame

3. Unnumbered Supervisory Frame (U-Frame) ใช้ในกรณีเริ่มต้นและยกเลิกการสื่อสาร

1	1	Cu	P/F	Cu
---	---	----	-----	----

รูปที่ 2.10 U-Frame

2 บิตแรกจะมีค่าเป็น “11”

Cu : Unnumbered Supervisory Command ประกอบด้วย 5 บิต เป็นคำสั่งที่ใช้ในกรณีเริ่มต้นการทำงานหรือฟื้นตัวจากการขัดข้องต่างๆ มีทั้งหมด 32 คำสั่ง

P/F : มีลักษณะเดียวกับที่ใช้ใน I-Frame

## 2.4 สถาปัตยกรรมของไมโครคอนโทรลเลอร์แบบชิพเดี่ยวตระกูล MCS-51

ไมโครคอนโทรลเลอร์แบบชิพเดี่ยว (Signal Chip Microcontroller) คือ ไมโครคอมพิวเตอร์แบบที่มีขนาดเล็กโดยบรรจุไว้ในแผงวงจรรวม (Integrated Circuit) เพียงชิพเดียว เหมาะสำหรับงานควบคุมอุปกรณ์อื่นๆ แบบอัตโนมัติ เพราะผู้ใช้สามารถเขียนโปรแกรมควบคุมการทำงานได้ตามที่ต้องการ

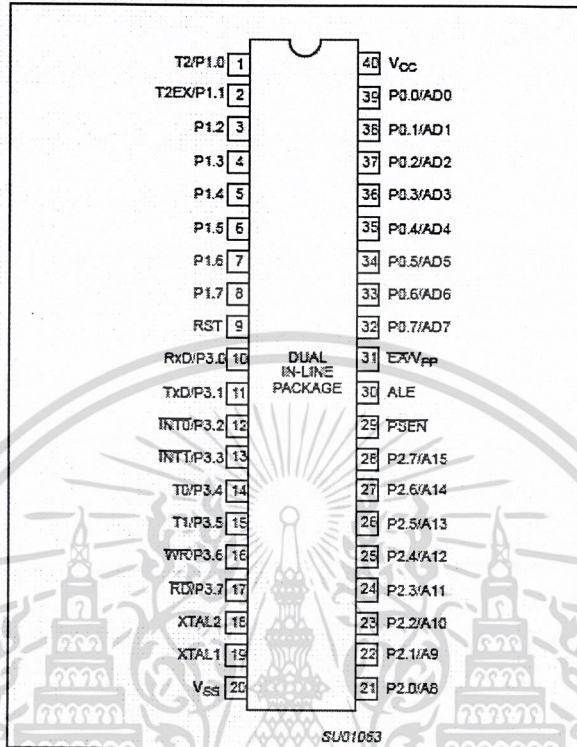
### MCS-51 มีข้อดีดังนี้

1. สามารถนำเอาข้อมูลมา AND, OR หรือทำ Complement ทั้งแบบทีละ 8 บิตและ 1 บิต
2. สามารถใช้กับหน่วยความจำสำหรับโปรแกรม (Program Memory) ซึ่งเป็นหน่วยความจำที่ใช้สำหรับเก็บชุดคำสั่งที่จะให้ MCS-51 ทำงานได้สูงสุด 64 กิโลไบต์ (Kilobyte : 64\*1024 byte) ทำให้สามารถเขียนโปรแกรมควบคุมการทำงานได้มาก
3. สามารถต่อกับหน่วยความจำสำหรับข้อมูล (Data Memory) ซึ่งเป็นหน่วยความจำสำหรับเก็บข้อมูลในระหว่างการทำงานของโปรแกรมได้สูงสุด 64 กิโลไบต์
4. ใน 8051 และ 8751 มีหน่วยความจำสำหรับโปรแกรมจำนวน 4 กิโลไบต์ (ใน 8052 และ 8752 มีหน่วยความจำสำหรับโปรแกรมจำนวน 8 กิโลไบต์) อยู่ในวงจรรวม ทำให้ไม่ต้องต่อหน่วยความจำสำหรับโปรแกรมภายนอก ระบบทั้งหมดจึงมีขนาดเล็กและสัญญาณรบกวนจากภายนอกจะทำให้ MCS-51 ทำงานผิดพลาดได้ยาก
5. มีพอร์ตแบบขนาน (Parallel Port) สำหรับข้อมูลเข้าและออกจำนวน 32 บิต ที่ข้อมูลแต่ละบิตเป็นอิสระต่อกัน
6. มีวงจร Timer / Counter ขนาด 16 บิต 2 ชุด (8032, 8052 มี 3 ชุด) ที่ทำงานในโหมดต่างๆได้ถึง 4 โหมด
7. มี Universal Asynchronous Receiver Transmitter (UART) สำหรับรับ-ส่งข้อมูลแบบอนุกรม (Serial) แบบ Full duplex ที่สามารถเลือกแบบการรับ-ส่งข้อมูลได้ 4 แบบ
8. มีแหล่งกำเนิดสัญญาณขอขัดจังหวะการทำงานของโปรแกรม (Interrupt Request Signal) 6 แหล่งซึ่งสามารถกระโดดไปทำงานตอบสนองการขัดจังหวะ (Interrupt Service Routine) ได้ต่างหากกัน 5 ตำแหน่ง
9. สามารถเลือกการทำงานให้อยู่ในโหมดของ idle และ Power Down ซึ่งจะประหยัดการใช้กำลังไฟในการทำงาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## 2.4.1 สถาปัตยกรรมของ MCS-51

### PIN CONFIGURATIONS



รูปที่ 2.11 ไลออะแกรมขาของ MCS-51 แบบ DIP

ส่วนประกอบของ MCS-51 ดังแสดงในรูปที่ 2.11 ซึ่งจะอธิบายส่วนย่อยๆ ภายในของ MCS-51 เพียงชีพเดียวและสัญญาณภายในจะต่อออกสู่ภายนอกทางขา(pin) ของ MCS-51 ที่มีอยู่ 40 ขาดังรูปที่ 2.11 ซึ่งแต่ละขาจะทำหน้าที่ต่างๆดังนี้

1. VCC ขา 40 เป็นขาป้อนไฟเลี้ยง +5 โวลต์ เข้าไปเพื่อให้วงจรรวมทำงานได้ ระดับแรงดันของลอจิก 0 และ 1 ของ MCS-51 จึงต้องต่อเข้ากับอุปกรณ์ลอจิกแบบ TTL
2. VSS ขา 20 เป็นขาที่ต้องต่อกับกราวด์ (Ground) ของแหล่งจ่ายไฟ การต่ออุปกรณ์ทั้งหมดจะต้องมีกราวด์ของอุปกรณ์ต่อเข้าด้วยกัน
3. พอร์ต 0 (P0.0 - P0.7) มี 8 ขา แต่ละขาสามารถกำหนดให้เป็นได้ทั้งอินพุทและเอาต์พุท สำหรับงานที่ใช้ทั่วไป ถ้าต้องการกำหนดให้พอร์ต 0 ขาใดขาหนึ่งเป็นอินพุทสามารถทำได้โดยการเขียนข้อมูล "1" ไปยังแต่ละบิตของพอร์ตที่ต้องการติดต่อกับ ผลให้ขาพอร์ตนั้นมาสถานะปล่อยลอย (float) จึงมีอินพุทอิมพีแดนซ์สูง สามารถใช้งานเป็นขาพอร์ทอินพุทได้ นอกจากนั้นขาพอร์ตนี้อยู่ถูกใช้งานในการติดต่อกับขาแอดเดรสไบต์ต่ำของหน่วยความจำภายนอก (A0-A7) และขาข้อมูล (D0-D7) โดยใช้กระบวนการมัลติเพล็กซ์เข้าช่วย เพื่อสลับการทำงานเป็นได้ทั้งขาติดต่อกับแอดเดรสและขาข้อมูล

\*\* หมายเหตุ : พอร์ทนี้ไม่มีความต้านทานพูลอัพภายใน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4. พอร์ต 1 (P1.0 - P1.7) มีโครงสร้างคล้ายพอร์ต 0 แต่จะใช้ความต้านทานพูลอัพภายในแทน (Internal Pull-up Register) นอกจากนี้ AT89Sxx จะใช้ขา P1.0 เป็นขาอินพุตสำหรับค่าของไทมเมอร์ 2 และ P1.1 เป็นขาทรiggerเกอร์ของไทมเมอร์ 2 ในขณะที่ขา P1.4 ถึง P1.7 เป็นขาสำหรับเชื่อมต่อแบบ SPI เพื่อทำการโปรแกรมข้อมูลในระบบ

5. พอร์ต 2 (P2.0 - P2.7) มีโครงสร้างคล้ายพอร์ต 0 โดยมี FET ตัวล่างตัวเดียว ส่วนด้านบนใช้ความต้านทานพูลอัพแทน (Internal Pull-up) พอร์ตนี้ทำหน้าที่ 2 อย่างคือ เป็นอินพุตเอาต์พุตพอร์ตใช้งานทั่วไป เมื่อจะใช้งานเป็นอินพุตพอร์ตต้องส่งลอจิก “1” มาที่พอร์ตนี้ก่อนและยังสามารถใช้ในการติดต่อกับขาแอดเดรสไบต์สูงของหน่วยความจำภายนอก (A8-A15)

6. พอร์ต 3 (P3.0 – P3.7) มีโครงสร้างคล้ายพอร์ต 1 ทำงานได้ 2 หน้าที่คือ เป็นอินพุตเอาต์พุตพอร์ต ถ้าจะโปรแกรมให้เป็นอินพุตต้องส่งลอจิก “1” มาที่พอร์ตนี้ก่อนและอีกหน้าที่หนึ่งคือใช้รับส่งสัญญาณควบคุม ซึ่งมีรายละเอียดดังนี้

6.1 P3.0 เป็นขาที่ใช้รับข้อมูลแบบอนุกรม (UART)

6.2 P3.1 เป็นขาที่ใช้ส่งข้อมูลแบบอนุกรม (UART)

6.3 P3.2 ใช้รับสัญญาณการขัดจังหวะจากภายนอกเบอร์ 1

6.4 P3.3 ใช้รับสัญญาณการขัดจังหวะจากภายนอกเบอร์ 2

6.5 P3.4 เป็นขารับสัญญาณพัลส์อินพุตเข้าไปยังวงจรถาน์เตอร์ 0 (เป็นอินพุตโหมดเกตน์เตอร์)

6.6 P3.5 เป็นขารับสัญญาณพัลส์อินพุตเข้าไปยังวงจรถาน์เตอร์ 1 (เป็นอินพุตโหมดเกตน์เตอร์)

6.7 P3.6 ขาสัญญาณที่ใช้ในการเขียนข้อมูลลงหน่วยความจำภายนอก

6.8 P3.7 ขาสัญญาณที่ใช้ในการอ่านข้อมูลลงหน่วยความจำภายนอก

7. ALE (Address Latch Enable) เป็นขาส่งสไตรบสำหรับใช้ในการแลตช์แอดเดรสไบต์ต่ำ (A0-A7) ที่ส่งออกมาจากพอร์ต 0 สัญญาณนี้แอกทีฟทุกๆ 2 ครั้งใน 1 แมกซ์ซินไซเคิล

8. PSEN (Program Store Enable) เป็นขาสไตรบที่ใช้สำหรับอ่านข้อมูลจาก Program Memory ภายนอก สัญญาณนี้จะส่งออกมา 2 ครั้งใน 1 แมกซ์ซินไซเคิล แต่ก่อนเป็นการอ่าน Internal Program Memory จะไม่มีสัญญาณออกที่ขานี้

9. EA (External Access) ใช้เลือกหน่วยความจำโปรแกรมภายนอก

ป้อน “0” จะอ่านโปรแกรมจากภายนอกชิพ

ป้อน “1” จะอ่านโปรแกรมจากภายในชิพ

10. RST (Reset) ขารีเซ็ตจะรีเซ็ตได้ก็ต่อเมื่อป้อนลอจิก “1” เข้าที่ขานี้ นาน 2 แมกซ์ซินไซเคิล

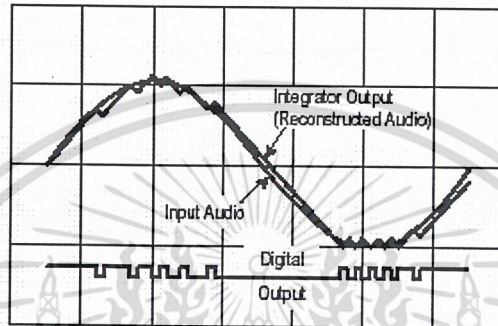
11. XTAL1 ใช้ต่อคริสตัลภายนอก โดยเป็นอินพุตเข้าสู่วงจรรอสซซิลเลเตอร์ภายใน

12. XTAL2 ใช้ต่อคริสตัลภายนอก โดยเป็นเอาต์พุตเข้าสู่วงจรรอสซซิลเลเตอร์ภายใน

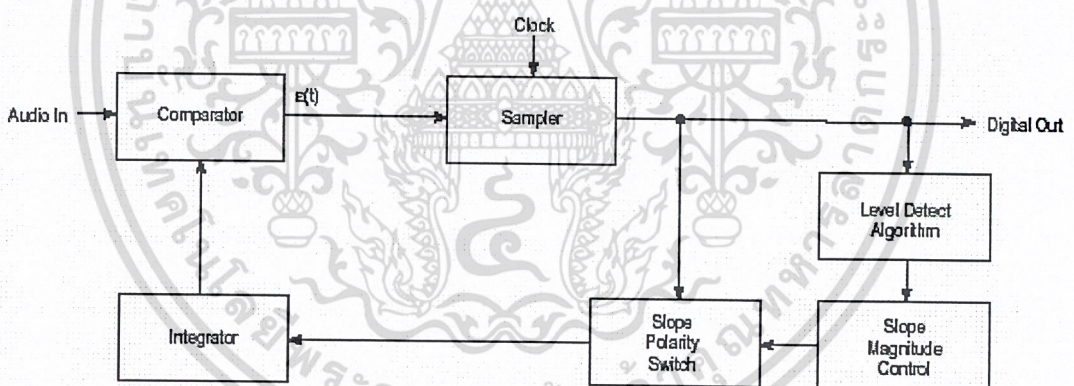
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## 2.5 การแปลงสัญญาณอนาลอกเป็นสัญญาณดิจิทัลและแปลงสัญญาณดิจิทัลเป็นสัญญาณอนาลอก

ส่วนนี้จะทำหน้าที่ แปลงสัญญาณข้อมูลที่เป็นสัญญาณอนาลอกมาเป็นสัญญาณดิจิทัลทางด้านส่งและแปลงสัญญาณดิจิทัลไปเป็นสัญญาณอนาลอกทางด้านรับ ซึ่งสัญญาณดิจิทัลนั้นจะถูกรบกวนได้ยากกว่าสัญญาณอนาลอกธรรมดา ก่อนที่จะนำไปทำการเข้ารหัส โดยในที่นี้จะใช้ไอซีเบอร์ MC34115 ที่ใช้หลักการของ CVSD ในการแปลงสัญญาณอนาลอกเป็นสัญญาณดิจิทัลและมีการมอดูเลตแบบเคลด้า ดังรูปที่ 2.12



รูปที่ 2.12 สัญญาณ CVSD Waveform



รูปที่ 2.13 บล็อกไดอะแกรมของ CVSD

## 2.6 สายไฟ AC

### ลักษณะทั่วไปของสายไฟฟ้า

สายไฟฟ้ามักมี 2 แบบด้วยกันคือ สายแข็ง (Solid wire) และสายตีเกลียว (Stranded wire) วัสดุที่นำมาทำเป็นสายไฟมีทั้งอะลูมิเนียมและทองแดง สายไฟที่ทำด้วยอะลูมิเนียมมักจะทำเป็นสายไฟแรงสูงในระบบสายส่งและเป็นสายเปลือยมากกว่าที่ใช้ทำเป็นสายไฟฟ้าแรงต่ำ สายไฟที่ทำด้วยทองแดงมีใช้งานไฟฟ้าทั่วไป และสายไฟแบ่งออกได้ 2 ชนิดด้วยกัน คือ สายเปลือย (Bare wire) และสายหุ้มฉนวน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

(Insulated wire) สายไฟที่ทำด้วยทองแดงจะต้องมีทองแดงไม่น้อยกว่า 98 % และสายไฟที่ทำด้วยอลูมิเนียม จะต้องมีอะลูมิเนียมไม่ต่ำกว่า 99.3%

### ชนิดของสายไฟหุ้มฉนวน

ฉนวนที่ใช้หุ้มสายแต่ละชนิดไม่เหมือนกัน ดังนั้นสายไฟแต่ละชนิดจึงแบ่งตามชนิดของฉนวนที่นำมาหุ้มเพื่อให้เหมาะสมในแต่ละสภาพที่นำไปใช้งาน

ชนิด IV สายไฟชนิดนี้ตัวนำทำด้วยทองแดงหุ้มด้วยฉนวน PVC ใช้ติดตั้งในบ้านพักอาศัยทั่วไป ทนอุณหภูมิสูงถึง 60 องศาใช้กับแรงดันไฟฟ้าได้ 250 โวลต์และใช้เป็นสายเมนจากตู้ตัดโอห์มมิเตอร์เข้าบ้านได้

ชนิด VCT ตัวนำไฟฟ้าทำด้วยทองแดงหุ้มด้วยฉนวน PVC ใช้ติดตั้งเครื่องจักรกลต่างๆ ทนอุณหภูมิได้สูงสุด 60 องศา ใช้กับแรงดันไฟฟ้าได้ 750 โวลต์

ชนิด VAF ตัวนำไฟฟ้าทำด้วยทองแดงหุ้มด้วยฉนวน PVC ใช้ติดตั้งในบ้านพักอาศัยทั่วไป ทนอุณหภูมิได้ 60 องศา ใช้กับแรงดันไฟฟ้า 250 โวลต์

ชนิด TW ตัวนำไฟฟ้าทำด้วยทองแดงหุ้มด้วยฉนวน PVC ใช้ติดตั้งในบ้านพักอาศัย สำนักงาน และงานอุตสาหกรรมทั่วไป ทนอุณหภูมิได้ 60 องศา ใช้กับแรงดันไฟฟ้า 750 โวลต์

ชนิด THW ตัวนำไฟฟ้าทำด้วยทองแดงหุ้มด้วยฉนวน PVC ใช้งานเดียวกับกรณี TW แต่ทนอุณหภูมิได้ 75 องศาใช้กับแรงดันไฟฟ้า 750 โวลต์

ชนิด NYY ตัวนำไฟฟ้าทำด้วยทองแดงหุ้มฉนวน PVC ใช้ติดตั้งใต้พื้นดิน ทนอุณหภูมิ 60 องศา ใช้กับแรงดันไฟฟ้า 750 โวลต์

ชนิด VFF ตัวนำไฟฟ้าทำด้วยทองแดงชนิดงอได้ (flexible copper wire) หุ้มด้วยฉนวน PVC ใช้กับเครื่องใช้ไฟฟ้าที่เคลื่อนที่ได้ ทนอุณหภูมิได้สูงถึง 60 องศา แรงดันไฟฟ้า 250 โวลต์

ชนิด AV ตัวนำไฟฟ้าทำด้วยทองแดงหุ้มด้วยฉนวน PVC ใช้ติดตั้งกับงานไฟฟ้ารถยนต์ ทนอุณหภูมิ 60 องศาใช้กับแรงดันไฟฟ้าต่ำๆ

### แรงดันไฟฟ้าตกในสายไฟ (Voltage Drop)

สายไฟฟ้าทุกเส้นจะมีความต้านทานภายใน ความต้านทานของสายไฟนี้จะมากหรือน้อยขึ้นอยู่กับความยาว ถ้าสายไฟมีความยาวมาก ความต้านทานก็มากตามด้วย ดังนั้นเมื่อมีกระแสไหลผ่านสายไฟ ก็จะทำให้เกิดแรงดันไฟตกที่สายนั้น เมื่อแรงดันไฟฟ้าที่ป้อนให้กับเครื่องใช้ไฟฟ้าลดลงต่ำกว่าขนาดที่เครื่องใช้ไฟฟ้ากำหนด จะทำให้การทำงานไม่ดีเท่าที่ควร

แรงดันไฟฟ้าที่สายป้อน Feeder ไม่ควรเกิน 2 % และแรงดันไฟฟ้าตกที่วงจรย่อย branch circuit ไม่ควรเกิน 3 % หรือทั้งสายป้อนและวงจรรวมกันแล้ว แรงดันไฟฟ้าตกจะต้องไม่เกิน 5 % ข้อนี้นี้เป็นค่าสูงสุด แต่ถ้าจะให้ดีไม่ควรเกิน 2 %

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 2.6.1 เอซีคัปปลิง (AC Coupling)

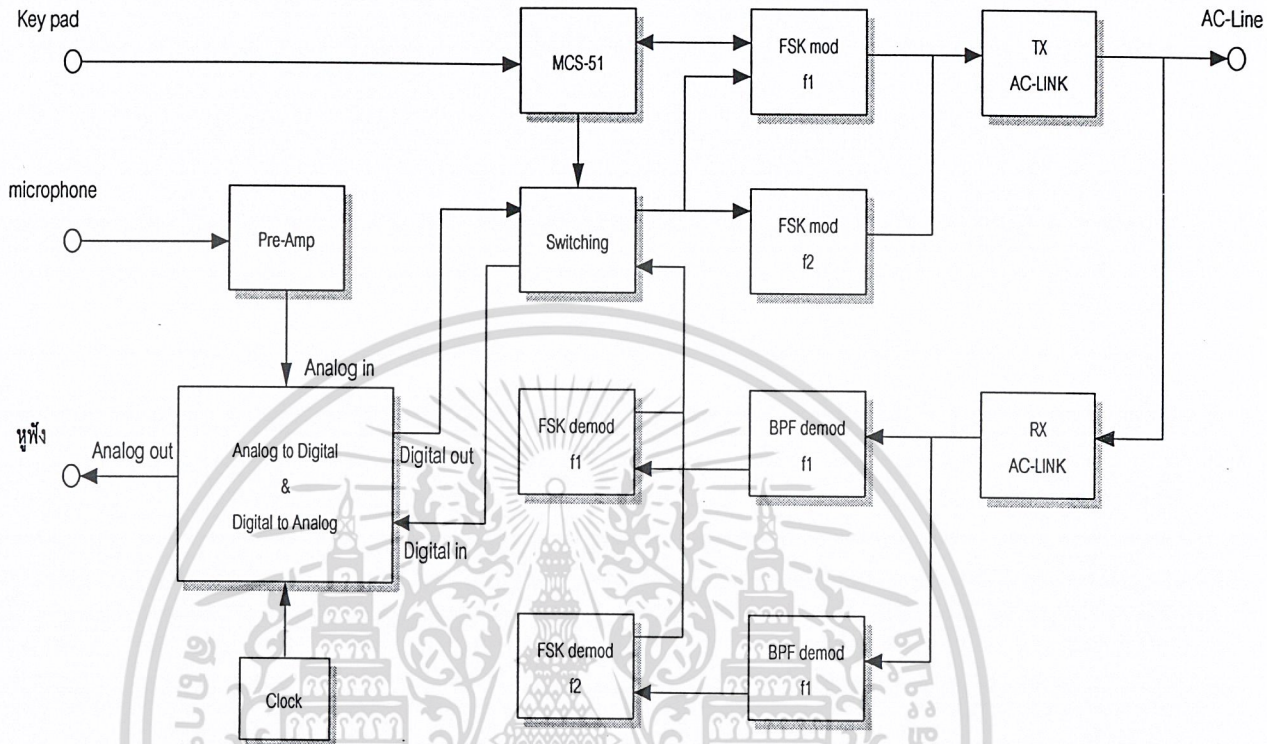
ใช้หลักการของหม้อแปลง โดยใช้ขดลวดของหม้อแปลงและคาปาซิเตอร์เป็นวงจรคัปปลิงในการส่งสัญญาณในสายส่งกำลังไฟฟ้า โดย Induce จากขด Primary ไปยังขด Secondary

อุปกรณ์ต่างๆในวงจรคัปปลิงนี้ นอกจากทำหน้าที่รับสัญญาณแล้ว ในการสร้างวงจรส่วนที่รับสัญญาณจากสายไฟฟ้าส่งกำลังนั้นยังสร้างให้มีหน้าที่คือ ส่วนป้องกันแรงดันสูงจากสายส่งกำลังคือคาปาซิเตอร์ ทำหน้าที่ตัดแรงดันสูงจากสายส่งกำลังส่วนป้องกันสัญญาณ 50 Hz คือหม้อแปลงเนื่องจากแกนของแม่เหล็กเป็นเฟอร์ไรท์จึงตัดสัญญาณ 50 Hz ได้



บทที่ 3

วงจรและการทำงานของวงจร



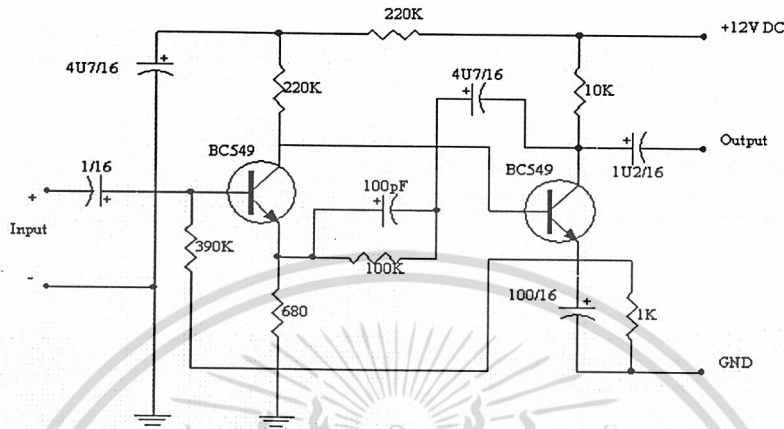
รูปที่ 3.1 บล็อกไดอะแกรมของระบบโทรศัพท์ภายในผ่านสายไฟฟ้า AC

การทำงานของระบบโทรศัพท์ภายในผ่านสายไฟ AC สามารถแบ่งการทำงานออกเป็นส่วนๆ ได้ดังนี้

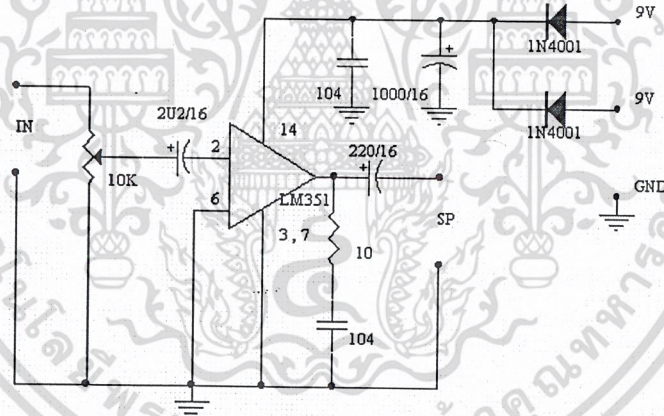
1. วงจรปริแอมป์ (Pre-Amp)
2. วงจรแปลงสัญญาณอนาลอกเป็นสัญญาณดิจิทัลและแปลงสัญญาณดิจิทัลเป็นสัญญาณอนาลอก (Analog to Digital and Digital to Analog Converter)
3. วงจรสร้างสัญญาณนาฬิกา (Clock Generator)
4. วงจรมอดูเลเตอร์และดีมอดูเลเตอร์แบบ FSK (FSK Modulator and FSK Demodulator)
5. วงจรกรองแถบความถี่ผ่าน (Band Pass Filter)
6. วงจรเอซีคัปปลิง (AC Coupling)
7. บัฟเฟอร์ 3 สถานะ (Tri-State Buffer)
8. วงจรสวิตชิง (Switching)
9. วงจรควบคุม
10. วงจรคีย์แพด (Key pad)
11. วงจรภาคจ่ายไฟ (Power Supply)

### 3.1 วงจรปริแอมป์

ประกอบด้วยวงจรปริไมค์ที่ใช้ทรานซิสเตอร์เบอร์ BC549 สำหรับไมโครโฟนที่มีอิมพีแดนซ์ต่ำ (200-600 โอห์ม) และวงจรเพาเวอร์แอมป์ที่ใช้ไอซีเบอร์ LM380N ทำงานด้วยอัตราขยาย 36dB ดังรูปที่ 3.2 และ 3.3 ตามลำดับ



รูปที่ 3.2 วงจรปริไมค์

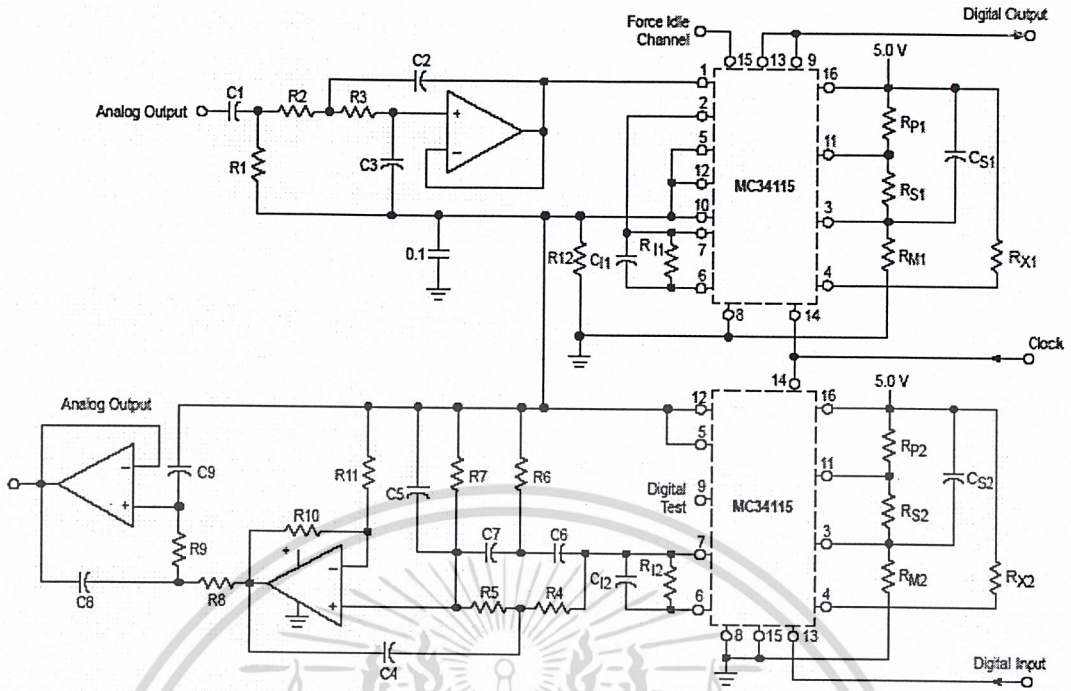


รูปที่ 3.3 วงจรเพาเวอร์แอมป์

### 3.2 วงจรแปลงสัญญาณอนาลอกเป็นสัญญาณดิจิทัลและแปลงสัญญาณดิจิทัลเป็นสัญญาณอนาลอก

ที่ด้านส่ง อินพุทของวงจรก็คือ สัญญาณเสียงที่มาจากวงจรปริแอมป์ ซึ่งเป็นสัญญาณอนาลอก เมื่อนำไปผ่านวงจรแปลงสัญญาณอนาลอกเป็นสัญญาณดิจิทัล และแปลงสัญญาณดิจิทัลเป็นสัญญาณอนาลอก ที่ใช้ไอซีเบอร์ MC34115 ซึ่งทำงานตามหลักการของ CVSD ที่เป็นเคลต้ามอดูเลชัน จะได้เอาท์พุทที่เป็นสัญญาณดิจิทัลออกมา เพื่อส่งไปเป็นอินพุทของวงจรมอดูเลเตอร์แบบ FSK สำหรับทางด้านรับ จะรับอินพุทที่เป็นสัญญาณดิจิทัลที่มาจากวงจรมอดูเลเตอร์แบบ FSK และได้เอาท์พุทออกมาเป็นสัญญาณอนาลอก เพื่อส่งออกไปที่หูฟังต่อไปโดยวงจรจะต้องใช้สัญญาณนาฬิกาที่มีความถี่ 16 kHz ด้วย วงจรแสดงดังรูปที่ 3.4

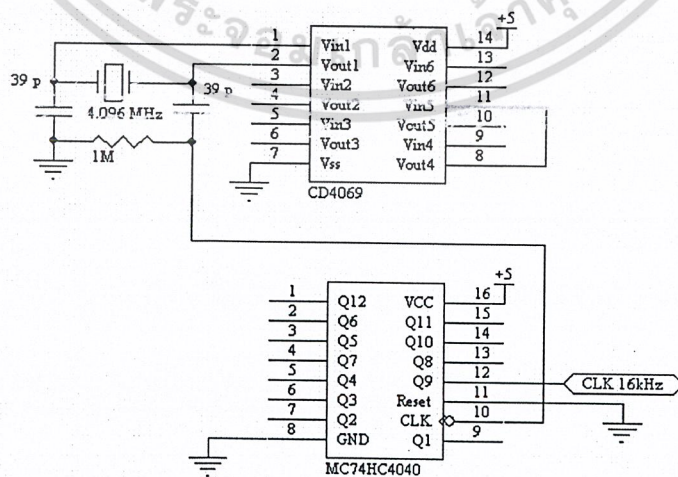
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.4 วงจรแปลงสัญญาณอนาลอกเป็นสัญญาณดิจิทัลและแปลงสัญญาณดิจิทัลเป็นสัญญาณอนาลอก

### 3.3 วงจรสร้างสัญญาณนาฬิกา

ในที่นี้เลือกใช้ออสซิลเลเตอร์ขนาด  $4.096\text{MHz}$  โดยต่อเข้ากับวงจรอินเวอร์เตอร์ ซึ่งในที่นี้เลือกใช้ไอซีเบอร์ CD4069 จะได้สัญญาณนาฬิกาขนาด  $4.096\text{MHz}$  จากนั้นนำไปเข้าวงจรหารความถี่ซึ่งใช้ไอซีเบอร์ 74HC4040 โดยหารเพื่อให้เหลือความถี่สัญญาณนาฬิกาที่ใช้ในวงจร ซึ่งใช้  $16\text{kHz}$  ในการป้อนให้กับวงจรแปลงสัญญาณอนาลอกเป็นดิจิทัล



รูปที่ 3.5 วงจรสร้างสัญญาณนาฬิกา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 3.4 วงจรมอดูเลเตอร์และดีมอดูเลเตอร์แบบ FSK

#### 3.4.1 การออกแบบวงจรมอดูเลเตอร์แบบ FSK

ในการออกแบบวงจรมอดูเลเตอร์แบบ FSK ได้นำเอาไอซีเบอร์ XR2206 ซึ่งเป็นโมโนลิธิค ฟังก์ชันเจนเนอเรเตอร์ กำเนิดรูปคลื่นเอาท์พุทได้ทั้งคลื่นซายน์ คลื่นสามเหลี่ยมและคลื่นสี่เหลี่ยมหรือ แรมป์ (ramp) โดยมีย่านความถี่ตั้งแต่  $0.01\text{kHz}$  ถึง  $1\text{MHz}$

ในกรณีนี้เราจะใช้ XR2206 เป็นตัวกำเนิดคลื่นซายน์ในลักษณะ FSK Generator โดยใช้ไทม์มิง รีซิสเตอร์  $R_1$  และ  $R_2$  ที่ต่อระหว่างขา 7 และ ขา 8 กับกราวด์ตามลำดับ โดยที่สัญญาณดิจิทัลที่ป้อนมายัง ขา 9 ของไอซี เป็นตัวกำหนดสัญญาณเอาท์พุท (ขา 2) ถ้าขา 9 อยู่ในสถานะวงจรเปิดหรือมี  $V_{in} > 2V$  แล้ว  $R_1$  จะเป็นตัวกำหนดไทม์มิงร่วมกับตัวเก็บประจุที่ต่อคร่อมระหว่างขา 5 กับขา 6 (หรือในทางกลับกัน ถ้า ขา 9 มี  $V_{in} < 1V$  แล้ว  $R_2$  จะเป็นตัวกำหนดไทม์มิงร่วมกับตัวเก็บประจุระหว่างขา 5 กับขา 6 เช่นเดียวกัน จึงทำให้ความถี่เอาท์พุทจะอยู่ในช่วงระหว่าง  $f_m$  (mark frequency) กับ  $f_s$  (space frequency) โดยทั้ง  $f_m$  กับ  $f_s$  จะเป็นอิสระต่อกันและสามารถเปลี่ยนแปลงความถี่ได้โดยการเลือกค่า  $R_1$  หรือ  $R_2$  ตามสมการข้างล่างนี้

$$f_m = \frac{1}{R_1 \times C}$$

$$f_s = \frac{1}{R_2 \times C}$$

ตัวเก็บประจุระหว่างขา 5 กับขา 6 จะอยู่ในช่วง  $1000\text{pF} - 100\mu\text{F}$  และตัวต้านทาน  $R_1$  และ  $R_2$  จะอยู่ในช่วงระหว่าง  $4\text{k}\Omega - 200\text{k}\Omega$

ในการออกแบบจากสมการเรากำหนด ค่า  $C = 100\text{nF}$  หากำหนด  $R_1$  และ  $R_2$  เมื่อกำหนด  $f_{m_1} = 15\text{kHz}$  และ  $f_{s_1} = 30\text{kHz}$

$$R_1 = \frac{1}{f_{m_1} \times C} = \frac{1}{15\text{k} \times 100\text{nF}} \approx 670\Omega$$

เลือกใช้โพรโทมิเตอร์  $20\text{k}\Omega$  ต่อกับขา 7 ลงกราวด์

$$R_2 = \frac{1}{f_{s_1} \times C} = \frac{1}{30\text{k} \times 100\text{nF}} \approx 330\Omega$$

เลือกใช้โพรโทมิเตอร์  $20\text{k}\Omega$  ต่อกับขา 8 ลงกราวด์

และกำหนด ค่า  $C = 100\text{nF}$  หากำหนด  $R_1$  และ  $R_2$  เมื่อกำหนด  $f_{m_2} = 55\text{kHz}$  และ  $f_{s_2} = 70\text{kHz}$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

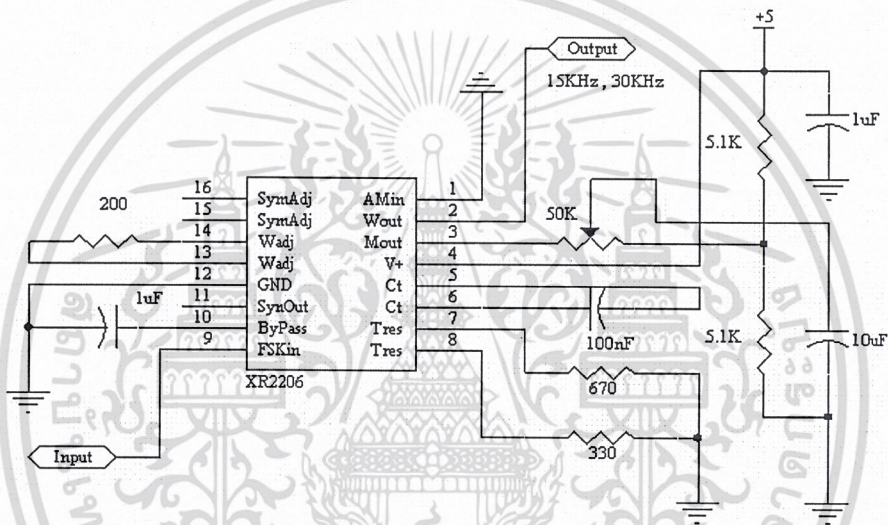
$$R_1 = \frac{1}{f_{m_2} \times C} = \frac{1}{55k \times 100nF} \approx 180\Omega$$

เลือกใช้โพรแท็นทิโอมิเตอร์ 20kΩ ต่อกับขา 7 ลงกราวด์

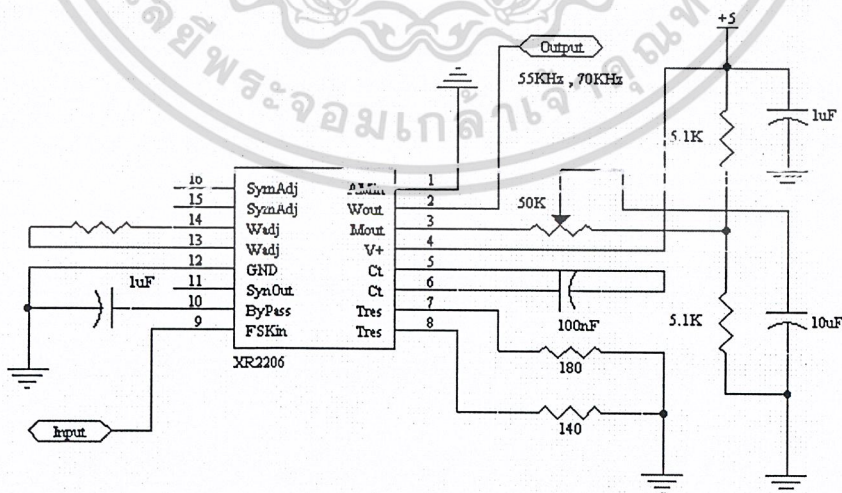
$$R_2 = \frac{1}{f_{s_2} \times C} = \frac{1}{70k \times 100nF} \approx 140\Omega$$

เลือกใช้โพรแท็นทิโอมิเตอร์ 20kΩ ต่อกับขา 8 ลงกราวด์ วงจรแสดงดังรูปที่ 3.6 และ 3.7 ตาม

ลำดับ



รูปที่ 3.6 วงจรมอดูเลเตอร์แบบ FSK ของความถี่  $f_{m_1} = 15kHz$  และ  $f_{s_1} = 30kHz$



รูปที่ 3.7 วงจรมอดูเลเตอร์แบบ FSK ของความถี่  $f_{m_2} = 55kHz$  และ  $f_{s_2} = 70kHz$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ระดับแรงดันไฟตรงที่ขาเอาต์พุต ขา 2 จะมีค่าโดยประมาณเท่ากับแรงดันไบอัสที่ขา 3 จากวงจรจะเห็นว่าแรงดันระหว่างค่าความต้านทาน  $5.1k\Omega$  สองตัวกับแรงดันไฟเลี้ยงประมาณ  $6V$  เมื่อแรงดันไฟเลี้ยงในวงจรเป็น  $12V$  และสามารถปรับโดยไปเทินที่โอมิเตอร์  $100K\Omega$  ที่อนุกรมกับขา 3 ดังนั้นการปรับที่ขา 3 ก็เป็นการปรับระดับของสัญญาณเอาต์พุต ขา 2 ให้อยู่ในระดับที่ต้องการส่ง ส่วนขา 13 และขา 14 มีไว้สำหรับแต่งรูปคลื่นเพื่อปรับการบิดเพี้ยน (Distortion) ของรูปร่างสัญญาณ อันเนื่องมาจากความบิดเพี้ยนฮาร์โมนิก โดยการต่อค่าความต้านทานเข้าไประหว่างขา 13 และขา 14 ในคู่มือ XR2206 แนะนำให้ใช้ค่า  $200\Omega$

### 3.4.2 การออกแบบวงจรดีมอดูเลเตอร์แบบ FSK

ในการออกแบบวงจรดีมอดูเลเตอร์แบบ FSK ได้นำเอาไอซีเบอร์ XR2211 ซึ่งเป็นโมนอลิธิคเฟส ล็อกกลุ๊ปโดยมีย่านความถี่ในการใช้งานตั้งแต่  $0.01Hz$  ถึง  $300kHz$  โครงสร้างภายในของไอซี XR2211 มาประยุกต์วงจรเป็นวงจรดีมอดูเลเตอร์แบบ FSK ส่วนประกอบหลักของเฟสล็อกกลุ๊ป (Phase Lock Loop) ใน XR2211 ประกอบด้วย 2 ส่วนคือ

1. Signal Pre -Amp มีหน้าที่จำกัดขนาดของสัญญาณอินพุต ให้อยู่ในค่าประมาณ  $2mV$  และขยายสัญญาณอีกครั้งหนึ่งเพื่อให้เป็นสัญญาณระดับสูง

2. VCO (Voltage Control Oscillator) มีเฟสดีเทคเตอร์เป็นภาคแรก โดยเอาต์พุตจะได้เป็นความถี่ผลบวกและความถี่ผลต่างคือ  $2f_{IN}$  และ  $0Hz$  ตามลำดับ เมื่อเฟสดีเทคอยู่ในสถานะผลบวก ก็จะได้ DC Error Voltage ออกไปยัง VCO เพื่อควบคุมการผลิตความถี่ออกมาค่าหนึ่ง ในภาวะปกติความถี่ฟรีรันนิ่ง ( $f_o$ ) จะ Set โดยค่าความต้านทาน  $R_o$  และคาปาซิเตอร์  $C_o$  ซึ่งสามารถหา  $f_o$  ได้

$$f_o = \frac{1}{R_o \times C_o}$$

ในการออกแบบอุปกรณ์ต่อภายนอก โดย  $R_o, C_o$  จะเป็นตัวกำหนด Free Running Frequency หรือ Center Frequency ( $f_o$ ) ของเฟสล็อกกลุ๊ป,  $R_1$  จะเป็นตัวกำหนดแบนด์วิดท์,  $C_1$  จะเป็นตัวกำหนด Damping factor หรือ Filter Time Constant,  $C_f$  และ  $R_f$  เป็นตัวกำหนดค่า Data FSK Output ค่าความต้านทาน  $R_B$  ( $510k\Omega$ ) ระหว่างขา 7 กับขา 8 เป็นตัวป้อนกลับทางออก เพื่อให้ transition time ของเอาต์พุต FSK เร็วขึ้น ดังนั้นจึงได้ค่าอุปกรณ์ต่างๆ ดังนี้

## การออกแบบตามการคำนวณใน DATA SHEET

สำหรับที่  $f_{m_1} = 15kHz$  และ  $f_{s_1} = 30kHz$

1. คำนวณค่าความถี่ศูนย์กลาง ( $f_o$ ) ของเฟสล็อกคูลูบ จาก

$$f_o = \frac{f_{m_1} + f_{s_1}}{2} = \frac{15k + 30k}{2} \approx 22504Hz$$

2. กำหนดค่า  $R_o = 20k\Omega$  เลือกใช้โปเท็นทิโอมิเตอร์  $50k\Omega$   
 3. หาค่า  $C_o$  จาก

$$C_o = \frac{1}{f_o R_o} = \frac{1}{22504Hz \times 20k} \approx 2.22 \times 10^{-9} F$$

4. หาค่า  $R_1$  เพื่อตั้งแบนด์วิดท์ จาก

$$R_1 = 20k \left[ \frac{22504Hz}{30kHz - 15kHz} \right] \approx 28.85k\Omega$$

เลือกใช้ โปเท็นทิโอมิเตอร์  $50K\Omega$

5. หาค่า  $C_1$  เพื่อตั้ง Damping factor

$$\text{จาก Damping factor} = \frac{1}{4} \sqrt{\frac{C_o}{C_1}} \text{ โดยกำหนด Damping coefficient} = 0.5$$

$$C_1 = \frac{C_o}{4} = \frac{2.22 \times 10^{-9} F}{4} \approx 0.555 \times 10^{-9} F$$

6. กำหนดให้  $C_f = 1pF$

7. หาค่า  $R_f$  และ  $R_B$  จากความสัมพันธ์

$$R_f \approx 5R_1$$

$$R_f \approx 5 \times 28.85k \approx 144.25k\Omega$$

เลือกใช้ ค่าความต้านทาน  $R_f = 100k\Omega$

และ

$$R_B \approx 5R_f$$

$$R_B \approx 5 \times 100k \approx 500k\Omega$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
 ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เลือกใช้ ค่าความต้านทาน  $R_B = 510k\Omega$

สำหรับที่  $f_{m_2} = 55kHz$  และ  $f_{s_2} = 70kHz$

1. คำนวณค่าความถี่ศูนย์กลาง ( $f_o$ ) ของเฟสล็อกคูลิป จาก

$$f_o = \frac{f_{m_2} + f_{s_2}}{2} = \frac{55k + 70k}{2} \approx 63.5kHz$$

2. กำหนด  $R_o = 20k\Omega$  เลือกใช้ โปเทนทิโอมิเตอร์  $50k\Omega$

3. หาค่า  $C_o$  จาก

$$C_o = \frac{1}{f_o R_o} = \frac{1}{63.5k \times 20k} \approx 780pF$$

4. หาค่า  $R_1$  เพื่อตั้งแบนด์วิดท์ จาก

$$R_1 = 20k \left[ \frac{63.5kHz}{70kHz - 55kHz} \right] \approx 80k\Omega$$

5. หาค่า  $C_1$  เพื่อตั้ง Damping factor

จาก  $Damping\ factor = \frac{1}{4} \sqrt{\frac{C_o}{C_1}}$  โดยกำหนด Damping coefficient = 0.5

$$C_1 = \frac{C_o}{4} = \frac{780pF}{4} \approx 200pF$$

6. กำหนดให้  $C_f = 1pF$

7. หาค่า  $R_f$  และ  $R_B$  จากความสัมพันธ์

$$R_f \approx 5R_1$$

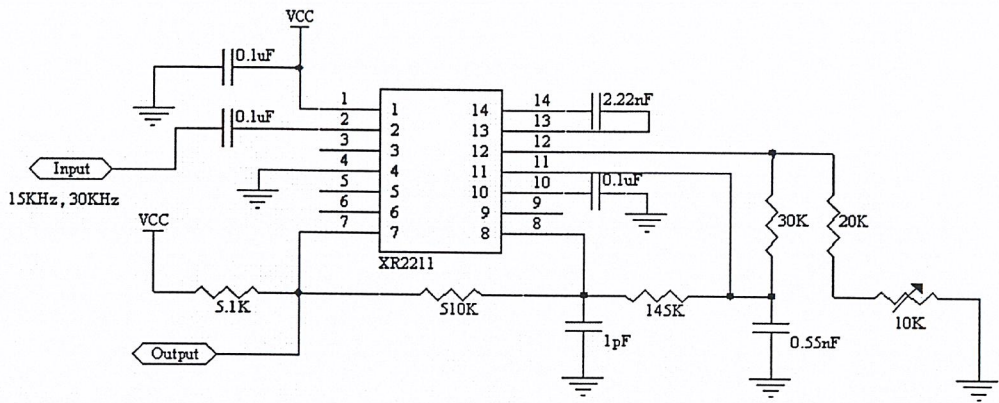
$$R_f \approx 5 \times 80k \approx 400k\Omega$$

$$R_B \approx 5R_f$$

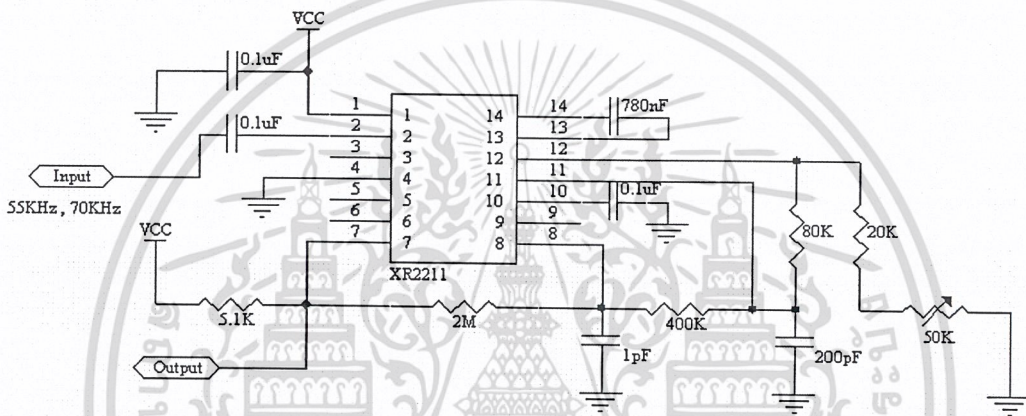
$$R_B \approx 5 \times 400k \approx 2M\Omega$$

เมื่อนำค่าอุปกรณ์ต่างๆ มาต่อได้วงจรดังรูปที่ 3.8 และ 3.9 ตามลำดับ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.8 วงจรดีมอดูเลเตอร์แบบ FSK ของความถี่  $f_{m_1} = 15kHz$  และ  $f_{s_1} = 30kHz$



รูปที่ 3.9 วงจรดีมอดูเลเตอร์แบบ FSK ของความถี่  $f_{m_2} = 55kHz$  และ  $f_{s_2} = 70kHz$

3.5 วงจรกรองแถบความถี่ผ่าน

จากวงจรกรองแถบความถี่ผ่านแบบ Sallen and Key จะได้รูปแบบสมการดังนี้

$$\omega_n = \left( \frac{1 + R_1 / R_2}{R_1 R_4 C_3 C_5} \right)^{1/2}$$

$$\frac{1}{Q} = \frac{[1 + (R_1 / R_2)(1 - K)] \sqrt{R_4 C_3 / R_1 C_5} + \sqrt{R_1 C_3 / R_4 C_5} + \sqrt{R_1 C_5 / R_4 C_3}}{\sqrt{1 + R_1 / R_2}}$$

$$H_0 = \frac{K / R_1 C_5}{1 / R_1 C_5 + 1 / R_2 C_5 + 1 / R_4 C_5 + 1 / R_4 C_3 - K / R_2 C_5}$$

ถ้ากำหนดให้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$$R_1 = R_2 = R_4 = R \quad , \quad C_3 = C_5 = C$$

จะสามารถลดรูปสมการข้างต้นได้เป็น

$$\omega_n = \frac{\sqrt{2}}{RC} \quad Q = \frac{\sqrt{2}}{4-K} \quad H_0 = \frac{K}{4-K}$$

และจะได้ความสัมพันธ์ของค่า  $K$  เป็น

$$K = 4 - \frac{\sqrt{2}}{Q}$$

โดยมีค่า  $K$  เป็นบวกและค่า  $Q$  จะต้องมีค่ามากกว่า  $\frac{\sqrt{2}}{4}$

ซึ่ง  $f_0$  สามารถหาได้จาก

$$f_0 = \sqrt{f_1 \times f_2}$$

และ

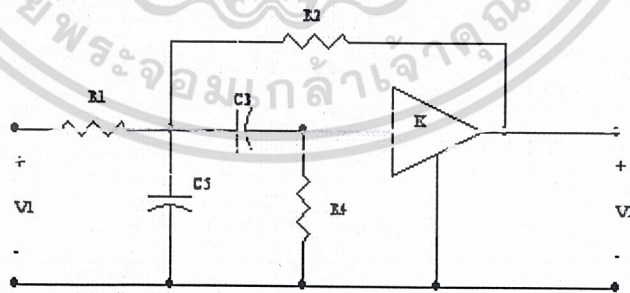
$$f_0 = \frac{\sqrt{2}}{2\pi RC} = \frac{1}{\sqrt{2}\pi RC}$$

เมื่อ  $BW$  คือแถบความถี่ที่ต้องการให้ผ่านมีหน่วยเป็น  $Hz$  หาได้จาก

$$BW = f_2 - f_1$$

โดย

$$f_{BW} = \frac{f_0}{Q}$$



รูปที่ 3.10 วงจรกรองแถบความถี่ผ่านแบบ Sallen and Key

สำหรับที่ช่วงความถี่ 15 – 30kHz จะคำนวณหาค่าต่างๆ ได้ดังนี้

$$f_0 = \sqrt{15k \times 30k} = 21.21kHz$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$$BW = 30k - 15k = 15kHz$$

กำหนดค่า  $C = 0.01\mu F$

$$\therefore R = \frac{1}{\sqrt{2} \times \pi \times 21.21k \times 0.01\mu} \approx 1k\Omega$$

$$Q = \frac{21.21k}{15k} = 1.414$$

$$\therefore K = 4 - \frac{\sqrt{2}}{1.414} \approx 3$$

สำหรับที่ช่วงความถี่  $55 - 70kHz$  จะคำนวณหาค่าต่างๆ ได้ดังนี้

$$f_0 = \sqrt{55k \times 70k} = 62kHz$$

$$BW = 70k - 55k = 15kHz$$

กำหนดค่า  $C = 0.01\mu F$

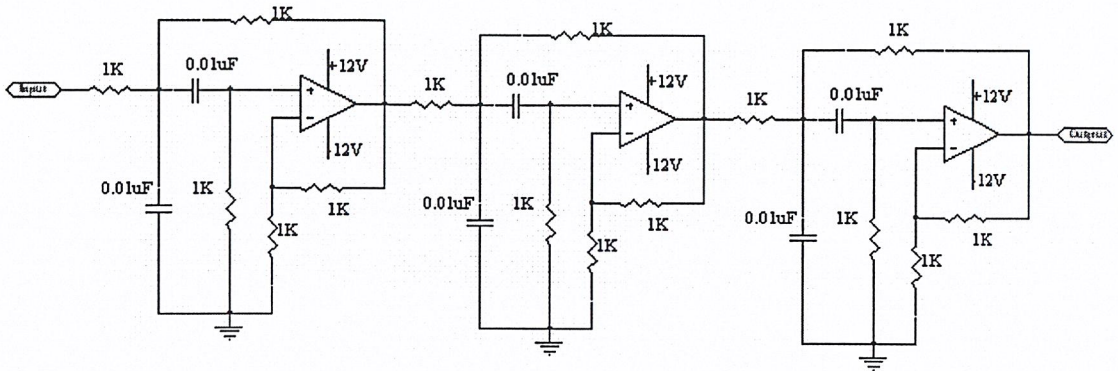
$$\therefore R = \frac{1}{\sqrt{2} \times \pi \times 62k \times 0.01\mu} \approx 363\Omega$$

$$Q = \frac{62k}{15k} = 4.133$$

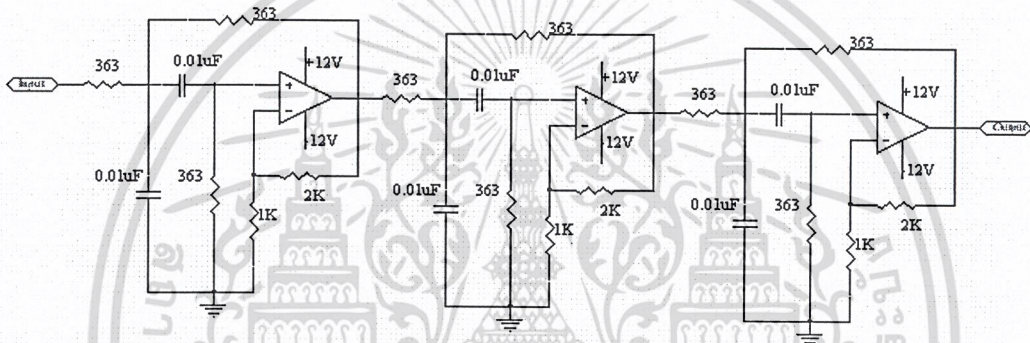
$$\therefore K = 4 - \frac{\sqrt{2}}{4.133} \approx 3.6$$

สำหรับโครงการนี้ ใช้วงจรกรองแถบความถี่ผ่านแบบ Sallen and Key ลำดับที่ 6 (6<sup>th</sup> order filter) โดยการนำวงจรกรองแถบความถี่ผ่านแบบ Sallen and Key ลำดับที่ 2 (2<sup>nd</sup> order filter) มาแคสเคด (cascade) กัน 3 วงจร ดังรูปที่ 3.11 และ 3.12 ตามลำดับ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



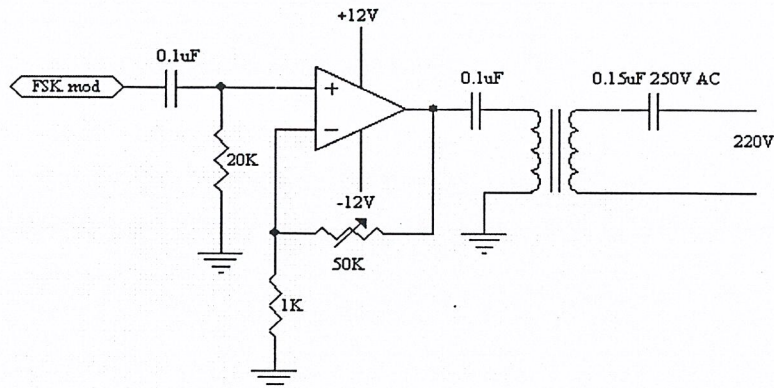
รูปที่ 3.11 วงจรกรองแถบความถี่ผ่านแบบ Sallen and Key ลำดับที่ 6 ของความถี่ 15 – 30kHz



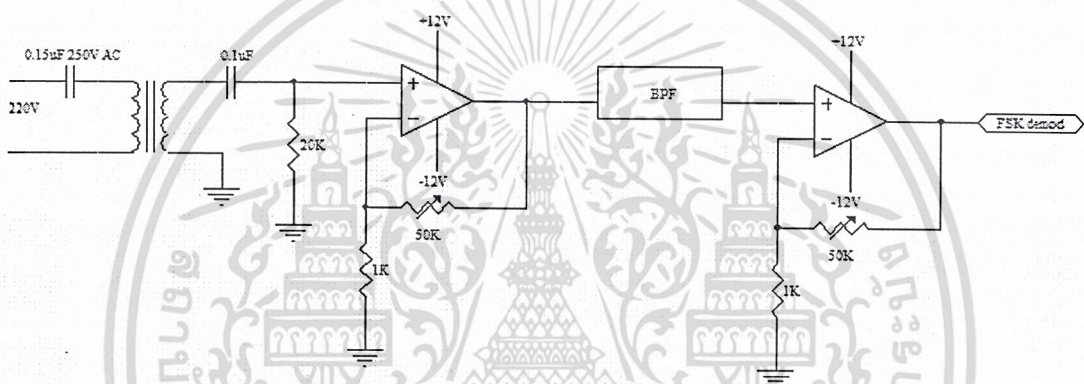
รูปที่ 3.12 วงจรกรองแถบความถี่ผ่านแบบ Sallen and Key ลำดับที่ 6 ของความถี่ 55 – 70kHz

### 3.6 วงจรเอซิกซ์ปลิ่ง

ประกอบด้วยวงจรเอซิกซ์ปลิ่งสำหรับด้านส่ง ที่จะต้องใช้หม้อแปลงที่มีเอาต์พุตอิมพีแดนซ์ต่ำ ใกล้เคียงกับค่าอิมพีแดนซ์ของสายไฟเอซี (AC line) ที่มีค่าประมาณ  $10\Omega$  เพื่อให้วงจรเอซิกซ์ปลิ่งนั้น แมช (match) กับสายไฟและทำให้สามารถส่งสัญญาณผ่านลงไปในสายได้และทางด้านรับก็เช่นเดียวกัน จะต้องใช้หม้อแปลงที่มีอินพุตอิมพีแดนซ์สูง เพื่อที่จะให้มีสัญญาณอินพุตผ่านเข้าวงจรมากที่สุด วงจรเอซิกซ์ปลิ่งสำหรับด้านส่งและด้านรับแสดงดังรูปที่ 3.13 และ 3.14 ตามลำดับ



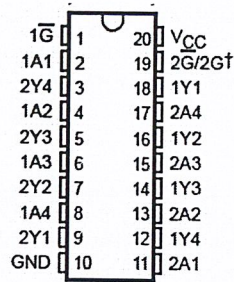
รูปที่ 3.13 วงจรเอชเค็ปปลิงสำหรับด้านส่ง



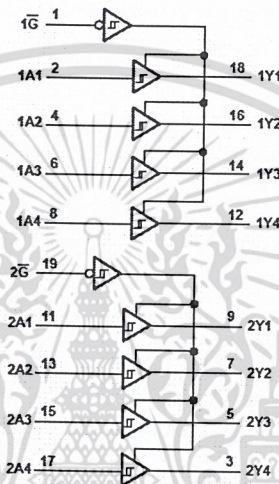
รูปที่ 3.14 วงจรเอชเค็ปปลิงสำหรับด้านรับ

### 3.7 บัฟเฟอร์ 3 สถานะ (Tri-State Buffer)

โดยทั่วไปแล้วเราจะพบบอยคือลอจิกแบบ 2 สถานะคือ Low และ High แต่เมื่อนำอุปกรณ์ภายนอกหลายๆ ตัวไปต่อกับบัสของซีพียูเดียวกัน จึงจำเป็นที่จะต้องควบคุมอุปกรณ์แต่ละตัวและผลัดกันทำงานโดยปลดตัวที่ไม่ใช้งานออกจากระบบ ดังนั้นจึงต้องให้บัฟเฟอร์ 3 สถานะ (74 SL244) มาคั่นระหว่างบัส เอาท์พุทของบัฟเฟอร์ 3 สถานะ มี 3 สถานะคือ Low ,High และ High Impedance หรือบางทีเรียก Floating โดยจะมีขาคอนโทรล (1G และ 2G) ใช้ควบคุมการทำงานของบัฟเฟอร์ 3 สถานะ ดังรูปที่ 3.15



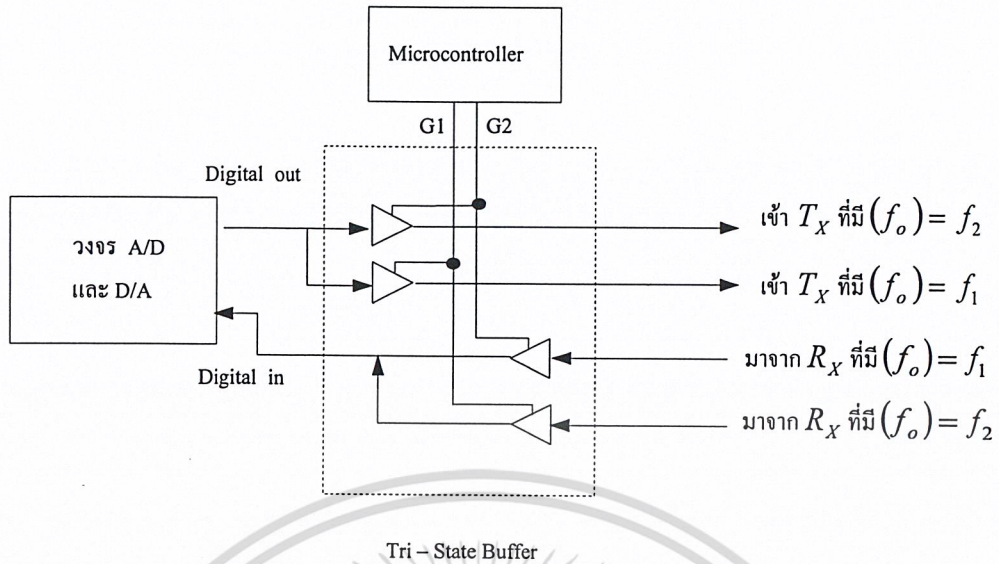
รูปที่ 3.15 ไคอะแกรมขาของบัฟเฟอร์ 3 สถานะ (74 SL244)



รูปที่ 3.16 ลอจิกไคอะแกรมของบัฟเฟอร์ 3 สถานะ (74 SL244)

### 3.8 วงจรสวิตชิง (Switching)

เนื่องจากต้องทำการส่งและรับข้อมูลในเวลาเดียวกัน จึงจำเป็นต้องใช้วงจรมอดูเลเตอร์ 2 ตัว ซึ่งมีค่าความถี่กลางที่ต่างกันออกไปเพื่อใช้ในการส่งที่ความถี่หนึ่งและรับอีกความถี่หนึ่ง ซึ่งถ้าต้นทางใช้ความถี่  $f_1$  ในการส่งและใช้ความถี่  $f_2$  ในการรับข้อมูล ส่วนที่เครื่องรับของปลายทางจะต้องใช้ความถี่  $f_1$  ในการรับข้อมูลและใช้ความถี่  $f_2$  ในการส่งข้อมูล จะเห็นได้ว่าในการติดต่อกันระหว่างต้นทางและปลายทางจะใช้ความถี่สำหรับการรับส่งข้อมูลที่ต่างกัน ดังนั้นเราจึงจำเป็นต้องมีวงจร Switching เพื่อทำการสลับให้ต้นทางและปลายทางใช้ความถี่ที่ในการรับส่งข้อมูลที่สัมพันธ์กัน ซึ่งวงจรนี้จะถูกควบคุมการทำงานโดยไมโครคอนโทรลเลอร์ โดยการส่งลอจิกไปที่ขาคอนโทรลของ Tri-State Buffer



รูปที่ 3.17 วงจรสวิตชิง

จากรูปที่ 3.17 ถ้าเรากำหนดให้ค่า  $G1$  และ  $G2$  มีค่า “01” จะทำให้สัญญาณดิจิทัลที่รับเข้ามาถูกส่งต่อไปยัง  $T_X$  ที่มีค่าความถี่กลางเท่ากับ  $f_2$  และรับสัญญาณเข้ามาจาก  $R_X$  ที่มีค่าความถี่กลางเท่ากับ  $f_1$  ซึ่งค่า  $G1$  และ  $G2$  นี้เราสามารถกำหนดค่าได้โดยไมโครคอนโทรลเลอร์

### 3.9 วงจรควบคุม

ในส่วนของวงจรควบคุมระบบการทำงาน จะใช้ไมโครคอนโทรลเลอร์ในการควบคุม โดยไมโครคอนโทรลเลอร์จะแบ่งการทำงานเป็น 2 ส่วน คือ ส่วนรับส่งโปรโตคอลเพื่อให้ระบบทำการเชื่อมต่อกันได้อย่างเป็นขั้นตอน และส่วนของการสร้างสัญญาณต่างๆที่ใช้ในระบบโทรศัพท์

#### 3.9.1 โปรโตคอล HDLC

ในโครงการนี้ได้ทำการกำหนด HDLC Frame ดังนี้

1. Flag Field มีค่า “01111110”

2. Address Field

2.1. 4 บิตแรก บอกตำแหน่งของด้านส่ง

2.2. 4 บิตหลัง บอกตำแหน่งของด้านรับ

2.3. ค่า “1110” ใช้ในกรณีที่ต้องการส่งไปยังทุกเครื่อง

3. Control Field จะใช้เพียง I-Frame และ S-Frame

3.1. I-Frame จะใช้เพียง P/F เพื่อทำการร้องขอ ACK ส่วน NS และ NR ไม่ใช้

เนื่องจาก เราจะทำการส่งแบบ Stop-and-Wait คือมีการโต้ตอบกันเฟรมต่อเฟรม จึงไม่ จำเป็นต้องทำการนับจำนวนเฟรมที่ทำการรับ-ส่ง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.2. S-Frame จะใช้เพียง Cn และ P/F โดย Cn มีการกำหนดค่าสำหรับคำสั่งต่างๆ ที่ใช้งานดังนี้

- 1). "00" สำหรับคำสั่ง RR
- 2). "01" สำหรับคำสั่ง RNR
- 3). "10" สำหรับคำสั่ง REJ

4. Data Field มีจำนวน 4 บิต กำหนดขึ้นเพื่อใช้บอกว่าเป็นการรับ-ส่ง ACK ใดในขั้นตอนการติดต่อ เพื่อป้องกันการผิดพลาดในการตอบรับ ACK

- 4.1 ค่า "0001" ใช้ในกรณีเรียกหาเครื่องปลายทาง
- 4.2 ค่า "0010" ใช้ในกรณีแสดงการรอรับสาย
- 4.3 ค่า "1010" ใช้ในกรณียกเลิกการติดต่อ

5. FCS Field จำนวน 4 บิต เป็นการตรวจความผิดพลาดที่เกิดขึ้นใน Address Field , Control Field และ Control Field โดยใช้การตรวจสอบแบบ Parity คู่ ตัวอย่างการจัดเฟรมของโปรโตคอลที่ใช้ในโรงงานนี้

1. เฟรมที่ต้นทางส่งไปยังทุกเครื่องเพื่อเรียกหาเครื่องที่มีหมายเลขตรงกับปลายทาง

Flag	ต้นทาง , ปลายทาง	0	000	1	000	0001	FCS	Flag
------	------------------	---	-----	---	-----	------	-----	------

2. เฟรมที่ปลายทางส่ง ACK กลับมายังต้นทางเพื่อบอกว่าตนเป็นเครื่องปลายทาง

Flag	ต้นทาง , ปลายทาง	0	000	1	000	0001	FCS	Flag
------	------------------	---	-----	---	-----	------	-----	------

3. เฟรมที่ส่งไปเพื่อแสดงการรอรับสาย

Flag	ต้นทาง , ปลายทาง	0	000	1	000	0010	FCS	Flag
------	------------------	---	-----	---	-----	------	-----	------

4. เฟรมที่ปลายทางส่ง ACK ของการรอรับสายกลับมายังต้นทาง

Flag	ต้นทาง , ปลายทาง	1	0	RR	1	000	0010	FCS	Flag
------	------------------	---	---	----	---	-----	------	-----	------

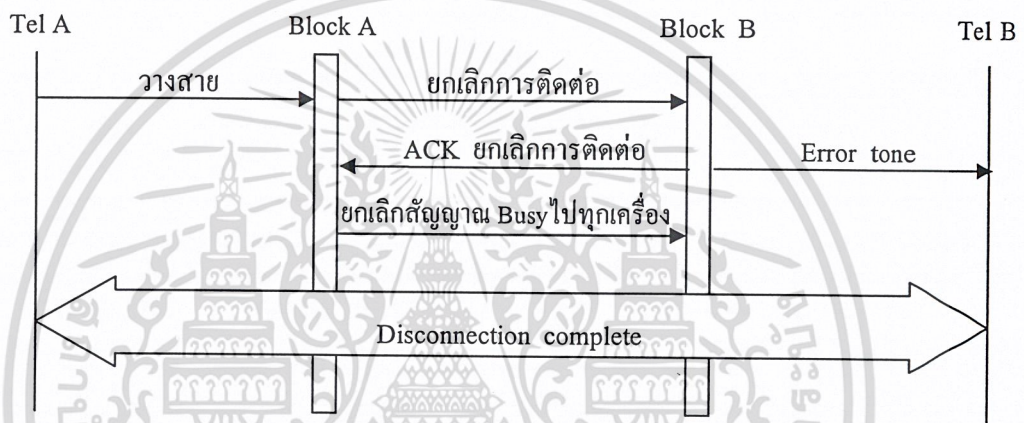
5. เฟรมที่ต้นทางส่งเพื่อทำการยกเลิกการติดต่อ

Flag	ต้นทาง , ปลายทาง	0	000	1	000	1010	FCS	Flag
------	------------------	---	-----	---	-----	------	-----	------

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



- เมื่อได้รับ ACK ก็จะส่งสัญญาณ Ring Back Tone ไปยังโทรศัพท์ของคุณ
5. เมื่อ Block ต้นทางส่งเฟรมที่แสดงการรอรับสายมายังปลายทาง ทาง Block ปลายทางจะสร้างสัญญาณ Ringing Tone ส่งไปยังโทรศัพท์ของคุณ
  6. เมื่อปลายทางทำการรับสาย สัญญาณ Ringing Tone จะถูกตัดไปและ Block ปลายทางจะส่ง ACK กลับไปยังต้นทางว่าได้ทำการรับสายแล้ว ทาง Block ต้นทางเมื่อได้รับ ACK จะตัดสัญญาณ Ring Back Tone
  7. ต้นทางและปลายทางทำการเชื่อมต่อกันได้อย่างสมบูรณ์  
ขั้นตอนการยกเลิกการติดต่อระหว่างต้นทางและปลายทาง



รูปที่ 3.19 ขั้นตอนการยกเลิกการติดต่อระหว่างต้นทางและปลายทาง

จากรูปที่ 3.19 แสดงลำดับขั้นตอนยกเลิกการติดต่อซึ่งสามารถอธิบายได้ดังนี้คือ

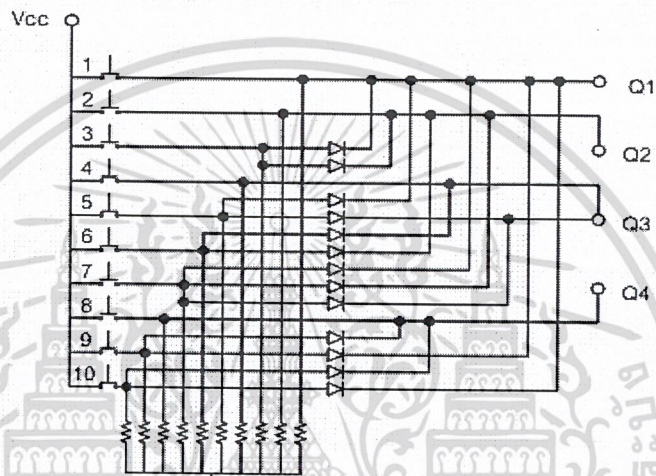
1. เมื่อต้นทางมีการวางสาย ที่ Block ต้นทางจะส่งสัญญาณยกเลิกการติดต่อไปยัง Block ปลายทาง
2. Block ปลายทางจะส่งสัญญาณ Error Tone ไปยังโทรศัพท์ของคุณ และส่ง ACK กลับมายังปลายทางเพื่อยืนยันการยกเลิกการติดต่อ
3. Block ต้นทางจะส่งสัญญาณ ไปยังทุก Block ที่เหลือเพื่อยกเลิกสัญญาณ Busy
4. สิ้นสุดการเชื่อมต่อระหว่างต้นทางและปลายทาง

### 3.9.2 สัญญาณโทรศัพท์

สัญญาณโทรศัพท์จะถูกสร้างจากไมโครคอนโทรลเลอร์ และจะถูกนำไปรวมกับสัญญาณไฟกระแสดังก่อนที่จะถูกป้อนให้คู่สายโทรศัพท์ เพื่อให้สัญญาณมีลักษณะเดียวกันกับสัญญาณที่ชุมสายส่งไปยังโทรศัพท์ ส่วนสัญญาณ Ringing จะใช้การจ่ายไฟให้กับ Buzzer แทน เนื่องจากมีความยุ่งยากในการสร้างสัญญาณ Ringing แบบที่ใช้ในระบบโทรศัพท์ที่ใช้งานจริง

### 3.10 วงจรคีย์แพด (Key pad)

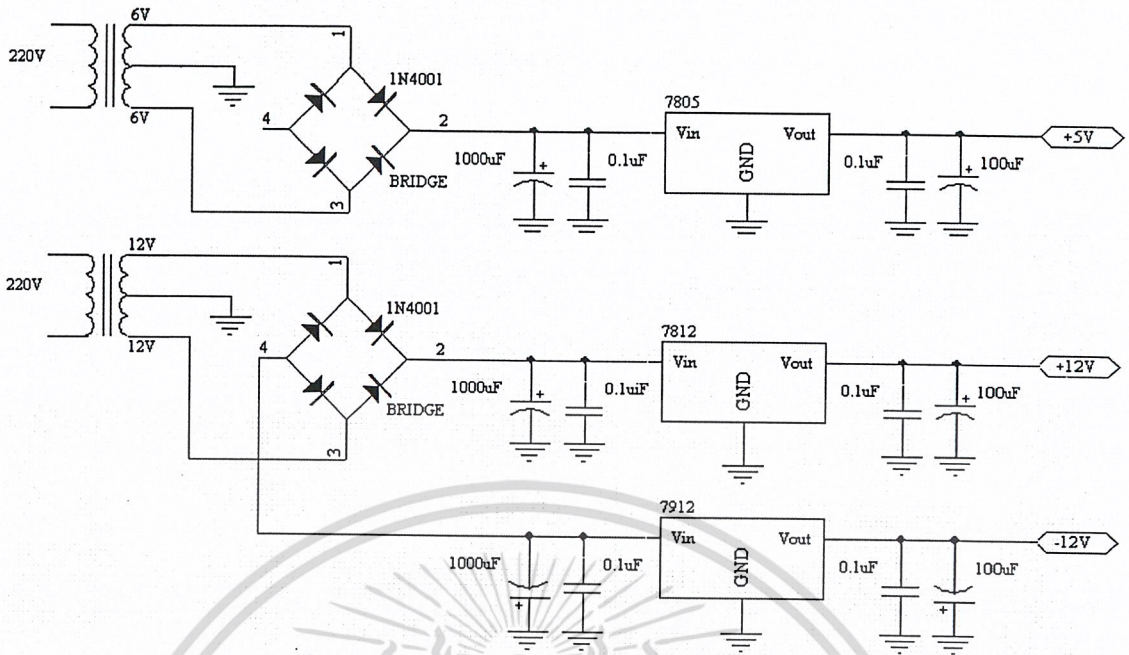
ในโครงงานนี้ใช้วงจรคีย์แพด ในการส่งค่าหมายเลขเครื่องที่ต้องการทำการติดต่อไปยังไมโครคอนโทรลเลอร์ โดยสามารถเลือกกดสวิตได้ตั้งแต่หมายเลข 0 ถึง 9 สวิตทุกตัวด้านหนึ่งจะถูกต่อเข้ากับไฟเลี้ยง เมื่อมีการกดสวิต ระดับไฟเลี้ยงนี้จะถูกส่งผ่านไปยังปลายทาง ส่วนอีกด้านหนึ่งจะถูกต่อเข้ากับตัวต้านทานพูลดาวน์ เพื่อให้ในขณะที่ไม่มีการกดสวิตระดับสัญญาณที่ปลายทางเป็นศูนย์ และมีการต่อเข้ากับไดโอดในลักษณะที่แตกต่างกันออกไป เพื่อให้ได้สัญญาณเอาต์พุตเป็นสัญญาณ BCD แล้วส่งต่อเข้าไปยังไมโครคอนโทรลเลอร์ที่พอร์ท P1.4 - P1.7 โดยที่ตำแหน่ง LSB คือ Q1 และตำแหน่ง MSB คือ Q4 ดังรูปที่ 3.20



รูปที่ 3.20 วงจรคีย์แพด

### 3.11 วงจรภาคจ่ายไฟ (Power Supply)

เป็นภาคแปลงแรงดันไฟกระแสสลับ (220VAC) ให้เป็นไฟกระแสตรง  $\pm 12VDC$  และ  $5VDC$  ออกแบบโดยใช้ไอซีเรกทูเลเตอร์ 7812, 7912 และ 7805 ซึ่งแสดงวงจรได้ดังรูปที่ 3.21



รูปที่ 3.21 วงจรภาคจ่ายไฟ

การทำงานเริ่มจากไฟกระแสสลับ (220VAC) ผ่านหม้อแปลงขนาด 12 โวลต์ แปลงขนาดแรงดันให้เป็น 12VAC ผ่านวงจรเรกติไฟ์กลายเป็นไฟกระแสตรงแบบเต็มคลื่น ผ่านคาปาซิเตอร์ทำการกรองความดันให้เรียบขึ้น ซึ่งจะนำไปเป็นอินพุทให้กับไอซีเรกกูเลเตอร์ 7812, 7912 และ 7805 ทำการแปลงแรงดันจาก 12VAC ให้เป็น +12 โวลต์, -12 โวลต์และ 5 โวลต์ตามลำดับ เพื่อนำไปใช้งานเป็นแหล่งจ่ายไฟให้วงจรต่างๆในเครื่องโทรศัพท์ต่อไป

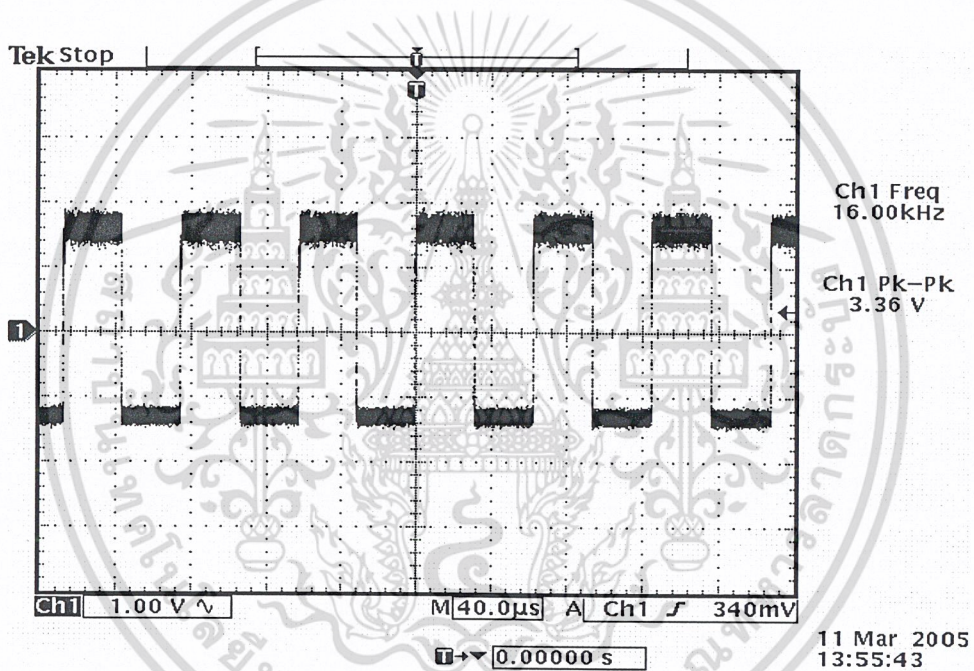
## บทที่ 4

### การทดลองและผลการทดลอง

สำหรับโครงการนี้ เมื่อพิจารณาถึงบล็อกไดอะแกรมที่ได้ทำการทดลองแล้ว จะประกอบด้วย วงจรสร้างสัญญาณนาฬิกา วงจรแปลงสัญญาณอนาลอกเป็นสัญญาณดิจิทัลและแปลงสัญญาณดิจิทัลเป็นสัญญาณอนาลอก วงจรมอดูเลเตอร์และวงจรมอดูเลเตอร์แบบ FSK วงจรกรองแถบความถี่ผ่าน และวงจรเอซีคัปปลิ่ง

#### 4.1 วงจรสร้างสัญญาณนาฬิกา

จากการต่อวงจรจะได้ผลการทดลองเป็นดังรูป

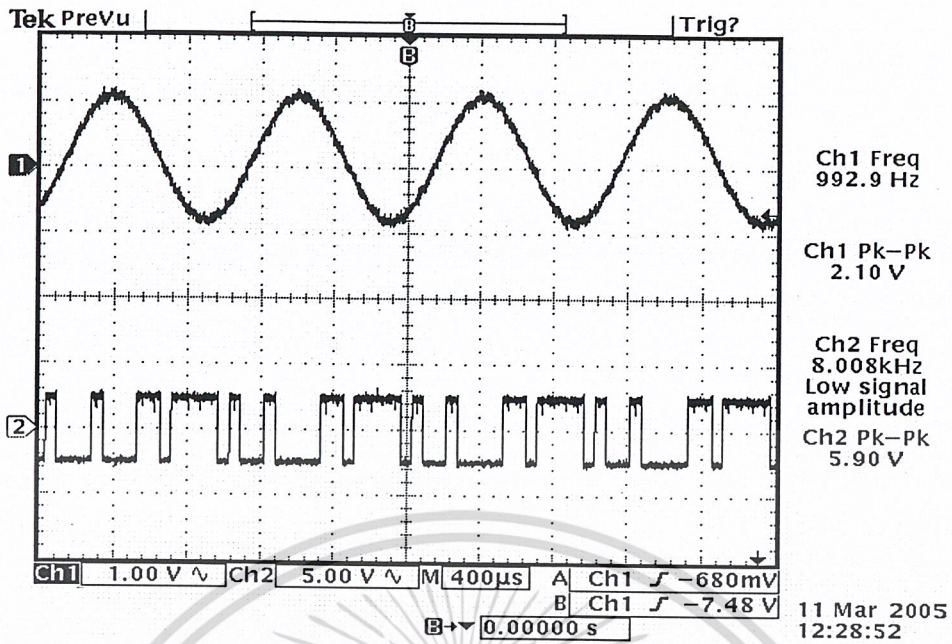


รูปที่ 4.1 สัญญาณเอาต์พุตจากวงจรสร้างสัญญาณนาฬิกาที่มีความถี่ 16 kHz

#### 4.2 วงจรแปลงสัญญาณอนาลอกเป็นสัญญาณดิจิทัลและแปลงสัญญาณดิจิทัลเป็นสัญญาณอนาลอก

##### 4.2.1 วงจรแปลงสัญญาณอนาลอกเป็นสัญญาณดิจิทัล (ด้านส่ง)

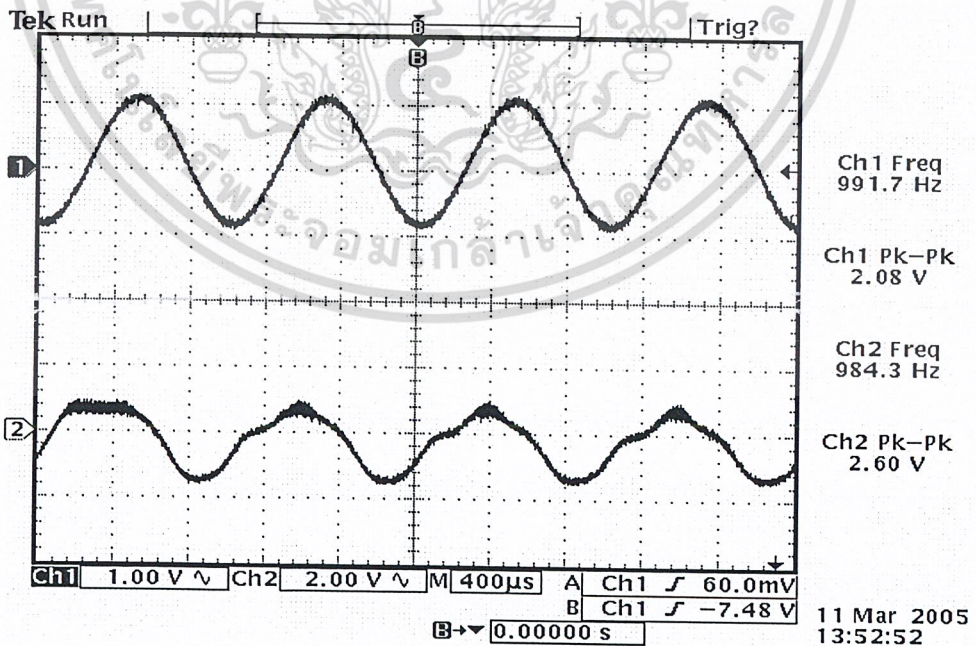
เมื่อทำการต่อวงจรและป้อนสัญญาณนาฬิกาที่มีความถี่ 16 kHz จะได้ผลการทดลองดังรูปที่ 4.2 โดยป้อนอินพุตเป็นสัญญาณไซน์ ความถี่ 1 kHz



รูปที่ 4.2 สัญญาณเอาต์พุต CVSD ด้านส่งเทียบกับ  
อินพุตสัญญาณชายน์ ความถี่ 1 kHz

#### 4.2.2 วงจรแปลงสัญญาณดิจิทัลเป็นสัญญาณอนาลอก (ด้านรับ)

เมื่อทำการต่อวงจรและป้อนสัญญาณนาฬิกาที่มีความถี่ 16 kHz จะได้ผลการทดลองดังรูปที่ 4.3 โดยป้อนอินพุตเป็นสัญญาณชายน์ ความถี่ 1 kHz ที่ด้านส่ง



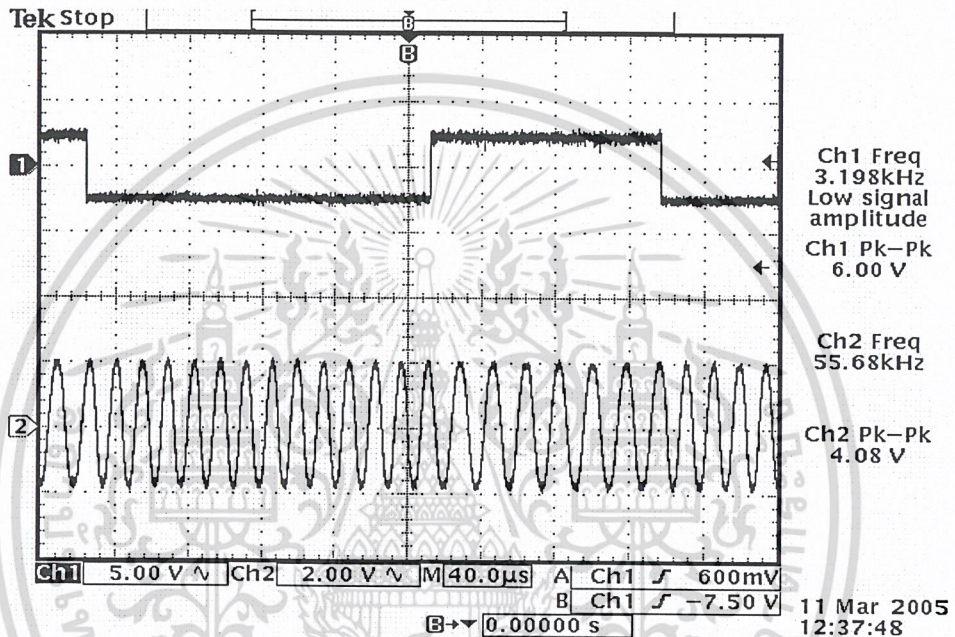
รูปที่ 4.3 สัญญาณเอาต์พุต CVSD ด้านรับเทียบกับ  
อินพุตสัญญาณชายน์ความถี่ 1 kHz ที่ด้านส่ง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 4.3 วงจรมอดูเลเตอร์และวงจรมอดูเลเตอร์แบบ FSK

#### 4.3.1 วงจรมอดูเลเตอร์แบบ FSK

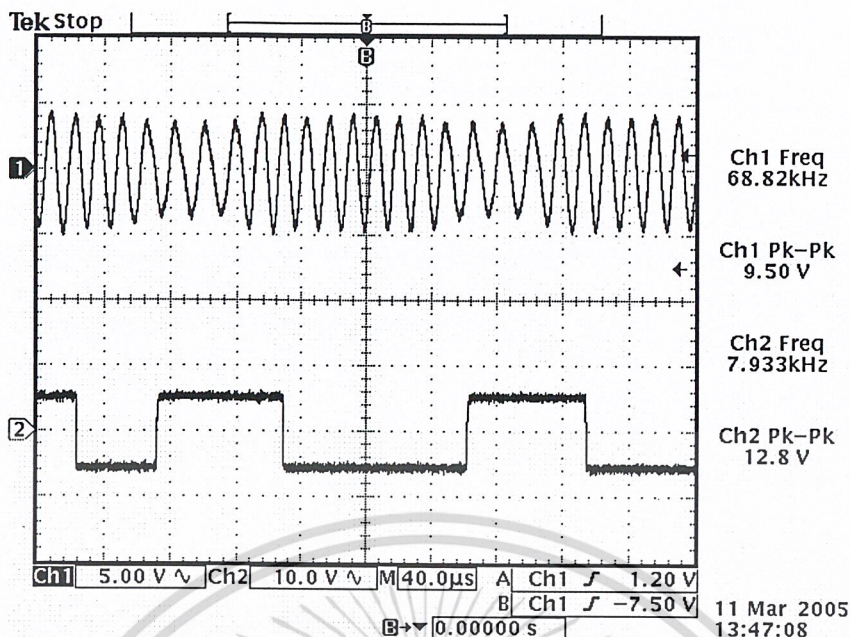
เมื่อทำการต่อวงจรแล้ว จะได้ผลการทดลองดังรูปที่ 4.4 สำหรับช่วงความถี่  $55-70\text{kHz}$  ซึ่งอินพุทของวงจรคือ เอาท์พุทจากวงจรแปลงสัญญาณอนาลอกเป็นสัญญาณดิจิทัล (ด้านส่ง) โดยป้อนอินพุทเป็นสัญญาณไซน์ ความถี่  $1\text{kHz}$  ที่ด้านส่ง



รูปที่ 4.4 สัญญาณเอาท์พุทของวงจรมอดูเลเตอร์แบบ FSK เทียบกับสัญญาณอินพุท โดยที่อินพุทที่ด้านส่งเป็นสัญญาณไซน์ ความถี่  $1\text{kHz}$

#### 4.3.2 วงจรมอดูเลเตอร์แบบ FSK

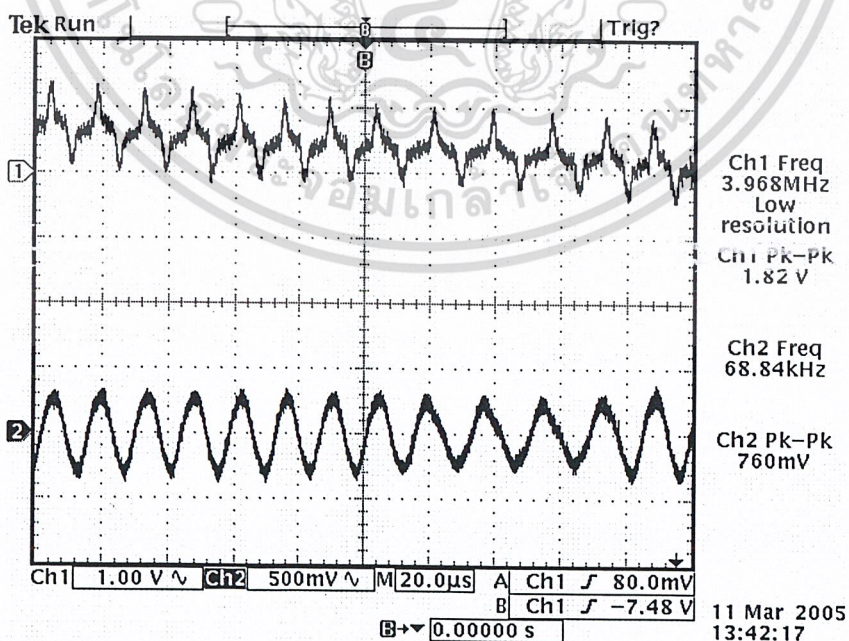
เมื่อทำการต่อวงจรแล้ว จะได้ผลการทดลองดังรูปที่ 4.5 สำหรับช่วงความถี่  $55-70\text{kHz}$  โดยที่อินพุทของวงจรคือ เอาท์พุทจากวงจรกรองแถบความถี่ผ่าน โดยป้อนอินพุทเป็นสัญญาณไซน์ ความถี่  $1\text{kHz}$  ที่ด้านส่ง



รูปที่ 4.5 สัญญาณเอาต์พุตของวงจรมอดูเลเตอร์แบบ FSK เทียบกับสัญญาณอินพุต โดยที่อินพุตที่ด้านส่งเป็นสัญญาณไซน์ ความถี่ 1 kHz

#### 4.4 วงจรกรองแถบความถี่ผ่าน

เมื่อทำการต่อวงจรแล้ว จะได้ผลการทดลองดังรูปที่ 4.6 สำหรับช่วงความถี่ 55–70 kHz โดยที่อินพุตของวงจรคือ เอาต์พุตจากวงจรเอซึคป์ปลิ่ง โดยป้อนอินพุตเป็นสัญญาณไซน์ ความถี่ 1 kHz ที่ด้านส่ง



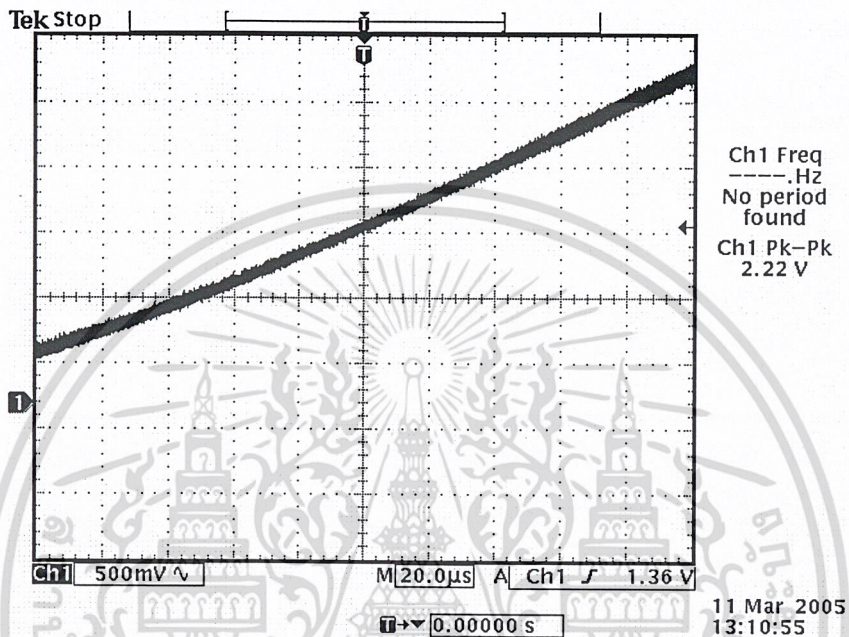
รูปที่ 4.6 สัญญาณเอาต์พุตของวงจรกรองแถบความถี่ผ่านเทียบกับสัญญาณอินพุต โดยที่อินพุตที่ด้านส่งเป็นสัญญาณไซน์ ความถี่ 1 kHz

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

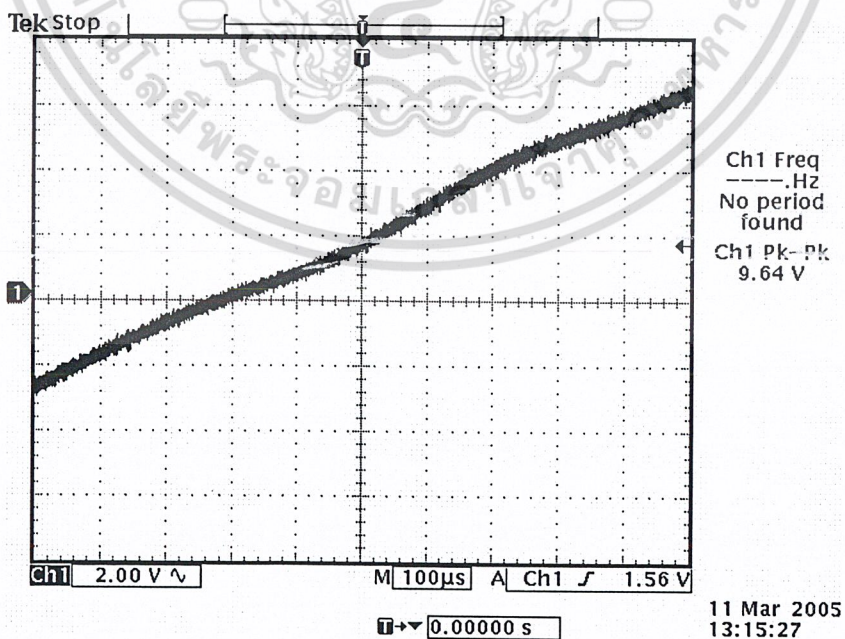
## 4.5 วงจรเอชีกัปปลิ่ง

### 4.5.1 วงจรเอชีกัปปลิ่ง (ด้านส่ง)

เมื่อทำการต่อวงจรแล้ว จะได้ผลการทดลองดังรูปที่ 4.7 ที่อินพุทของวงจรคือ เอาท์พุทจากวงจรมอดูเลเตอร์แบบ FSK โดยป้อนอินพุทเป็นสัญญาณชายน์ ความถี่  $1\text{ kHz}$  ที่ด้านส่ง



รูปที่ 4.7 สัญญาณก่อนป้อนอินพุท

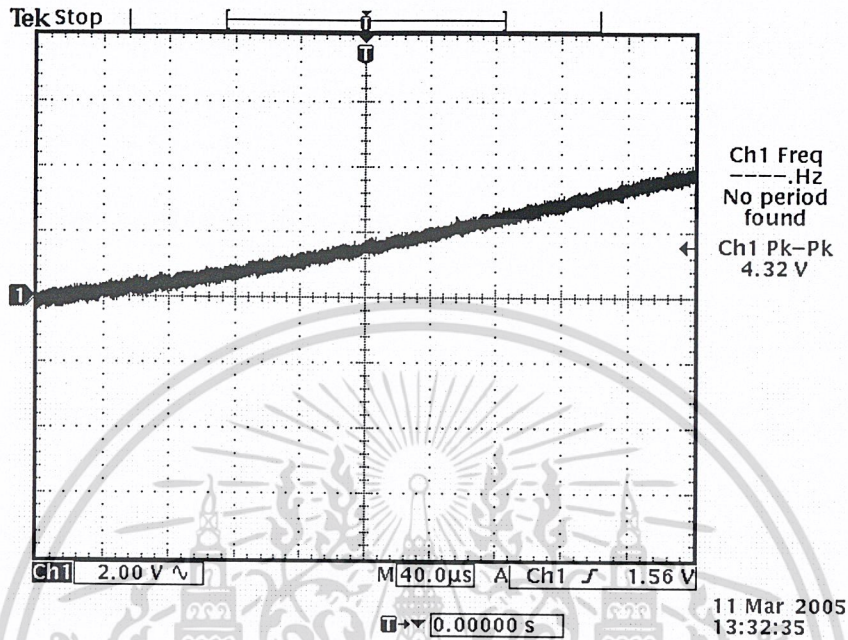


รูปที่ 4.8 สัญญาณหลังป้อนอินพุท โดยอินพุทที่ด้านส่งเป็นสัญญาณชายน์ ความถี่  $1\text{ kHz}$

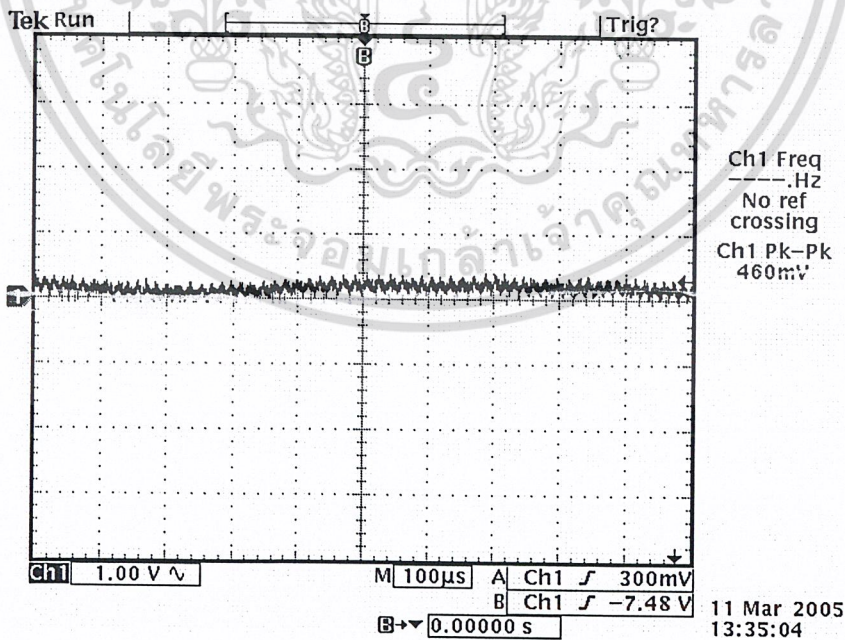
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

#### 4.5.2 วงจรเอชี่คัปปลิ่ง (ด้านรับ)

เมื่อทำการต่อวงจรแล้ว จะได้ผลการทดลองดังรูปที่ 4.9 โดยที่อินพุทของวงจรคือ เอาท์พุทจากสายไฟเอชี่ โดยป้อนอินพุทเป็นสัญญาณชายน้ ความถี่  $1\text{ kHz}$  ที่ด้านส่ง



รูปที่ 4.9 สัญญาณก่อนเข้าตัวเก็บประจุของวงจรเอชี่คัปปลิ่ง (ด้านรับ)  
โดยที่อินพุทที่ด้านส่งเป็นสัญญาณชายน้ ความถี่  $1\text{ kHz}$



รูปที่ 4.10 สัญญาณหลังเข้าตัวเก็บประจุของวงจรเอชี่คัปปลิ่ง (ด้านรับ)  
โดยที่อินพุทที่ด้านส่งเป็นสัญญาณชายน้ ความถี่  $1\text{ kHz}$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## บทที่ 5

### บทวิจารณ์ และ บทสรุป

จากการศึกษาข้อมูลต่างๆที่เกี่ยวกับโครงการ AC-Line Interphone ที่ผ่านมาพบว่า AC-Line หรือ สายไฟบ้านที่กลุ่มของข้าพเจ้าใช้เป็นตัวกลางในการส่งผ่านข้อมูลนั้น มีข้อจำกัดอยู่มากมาย ด้วยข้อเท็จจริงที่ว่า สายไฟบ้านไม่ได้ถูกสร้างมาเพื่อวัตถุประสงค์ในการส่งผ่านข้อมูลโดยเฉพาะ รวมทั้งยังมีเรื่องของสัญญาณรบกวนที่มีอยู่ภายในสายอีกมากมาย ส่งผลทำให้ประสิทธิภาพในการส่งผ่านข้อมูลของชิ้นงานได้ไม่ดีเท่าที่ควรแต่อยู่ในระดับที่ยอมรับได้ คือเครื่องส่งและเครื่องรับสามารถติดต่อสื่อสารกันได้ที่ระยะห่างประมาณ 30 เซนติเมตร จากนั้นได้ทำการส่งผ่านข้อมูลระหว่างเครื่องส่งและเครื่องรับ โดยใช้สายไฟบ้านที่มีขนาดแรงดัน 24 โวลต์ เป็นตัวกลางแทน พบว่าประสิทธิภาพในการทำงานของชิ้นงานดีขึ้นมาก คือที่ระยะห่าง 10 เมตร เครื่องส่งและเครื่องรับยังสามารถติดต่อสื่อสารกันได้อย่างมีประสิทธิภาพ เพราะฉะนั้นผลการทดลองดังที่ได้กล่าวมานี้ จึงเป็นข้อยืนยันว่าชิ้นงานสามารถทำงานได้อย่างมีประสิทธิภาพจริง



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## บรรณานุกรม

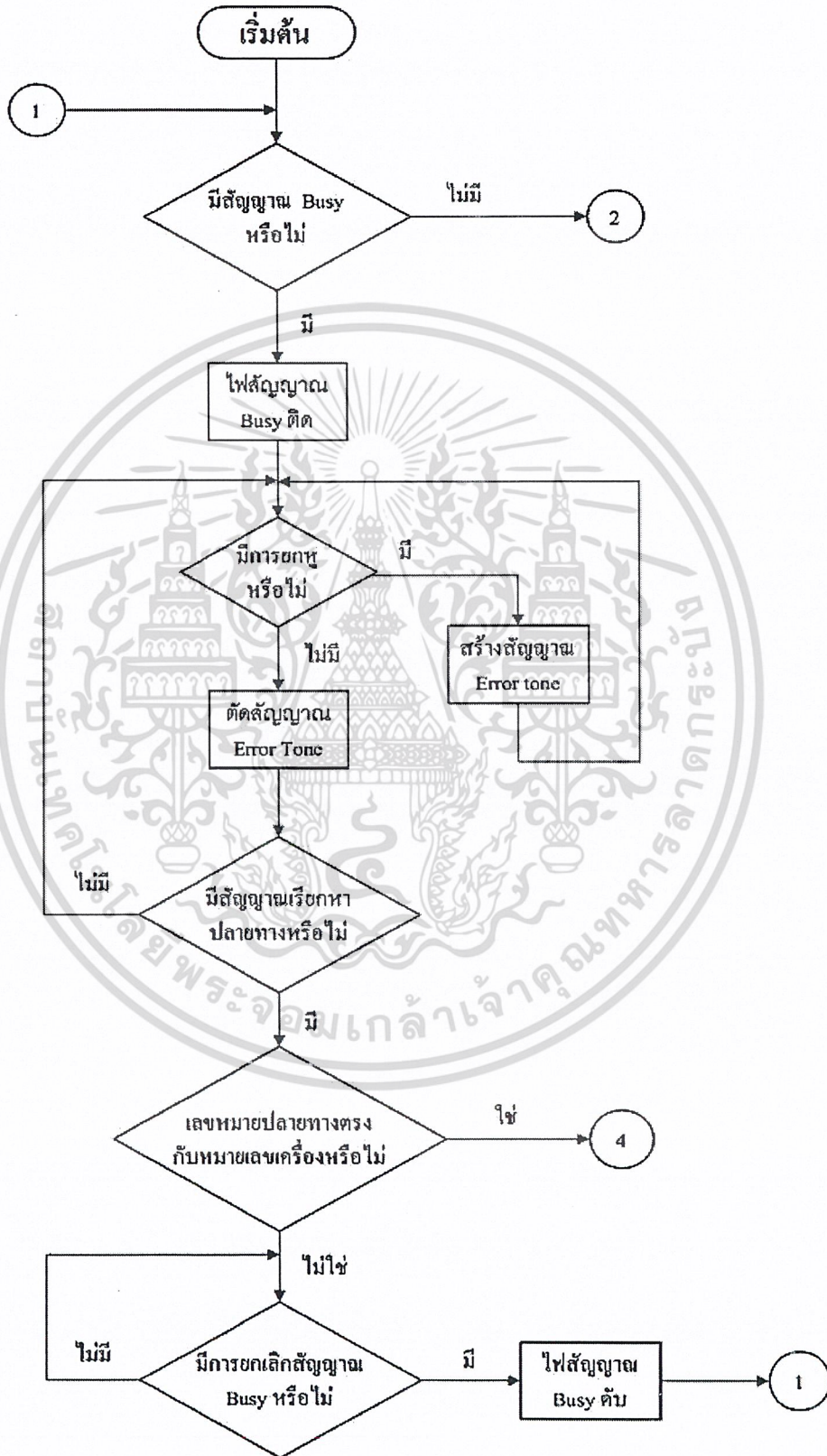
- [1] มนัส นกเฟือก, วันชัย บัวเลิศ, “ระบบสื่อสารข้อมูลผ่านสายไฟ AC โดยเทคนิค FSK = AC line transmission by FSK”, ปรินูญานิพนธ์ (วศ.บ.(วิศวกรรมโทรคมนาคม)), สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง, 2543.
- [2] สมยศ จุณณะปิยะ, “ไมโครคอนโทรลเลอร์ MCS-51“, สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง, 2541.
- [3] สุชิน จำจด, “Telephone Engineering”, ภาควิชาวิศวกรรมโทรคมนาคม, สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
- [4] Wayne Tomasi, “Electronic Communications Systems”, PRENTICE HALL, Englewood Cliffs, New Jersey 07632
- [5] วีระชาติ จันทร์แก้ว, สมเกียรติ เกิดทอง, “ตู้สาขาโทรศัพท์อัตโนมัติ”, ภาควิชาวิศวกรรมโทรคมนาคม, สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง, 2543.
- [6] ไชยยา สุขจันทร์, ณัฐพล ชันเจริญสุข, สุทธนา พรหมสวางค์, “เครื่องชุมสายโทรศัพท์สาขาอัตโนมัติ”, ภาควิชาวิศวกรรมโทรคมนาคม, สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง, 2544.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



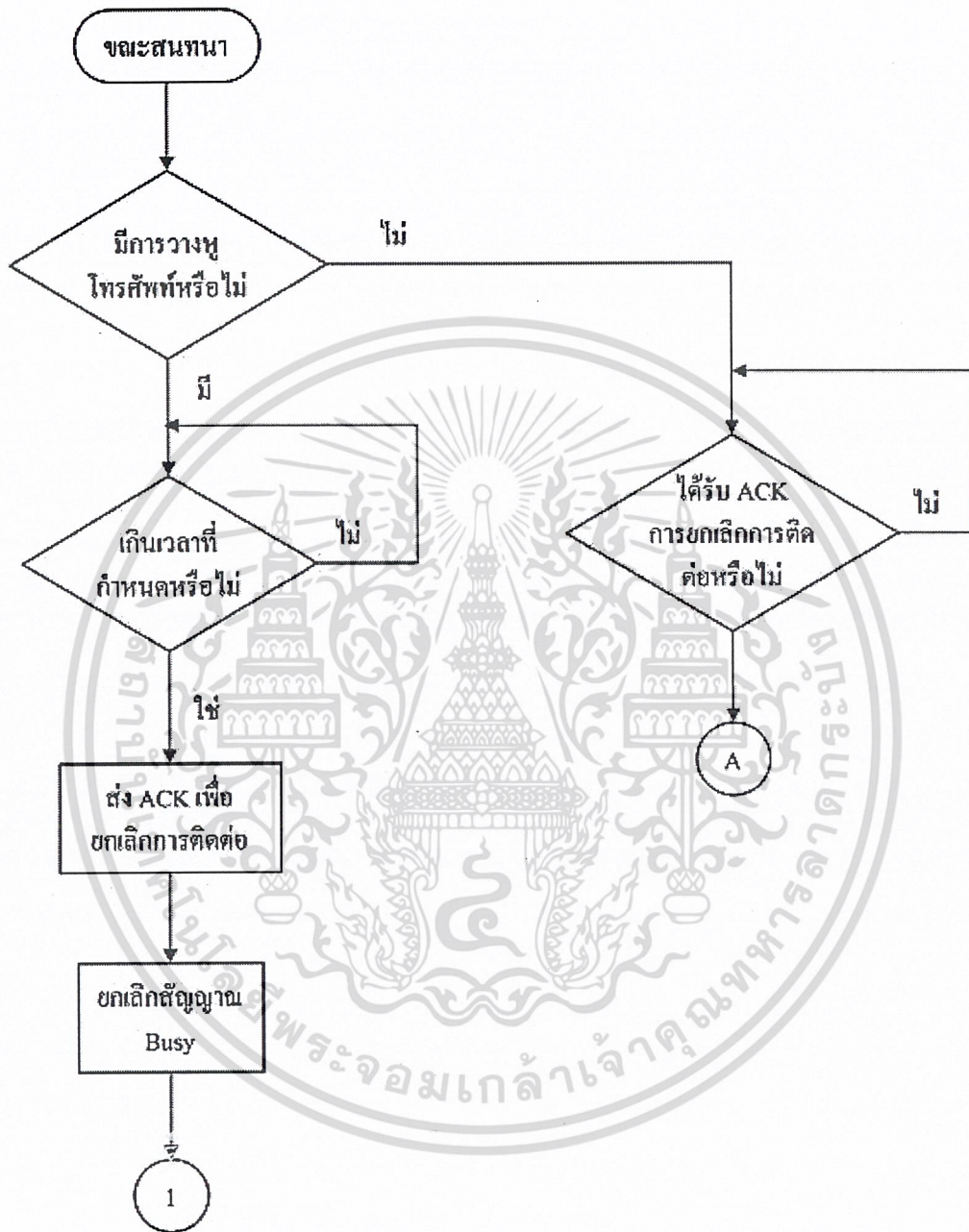
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Flowchart การติดต่อระหว่างผู้เรียกและผู้ถูกเรียก

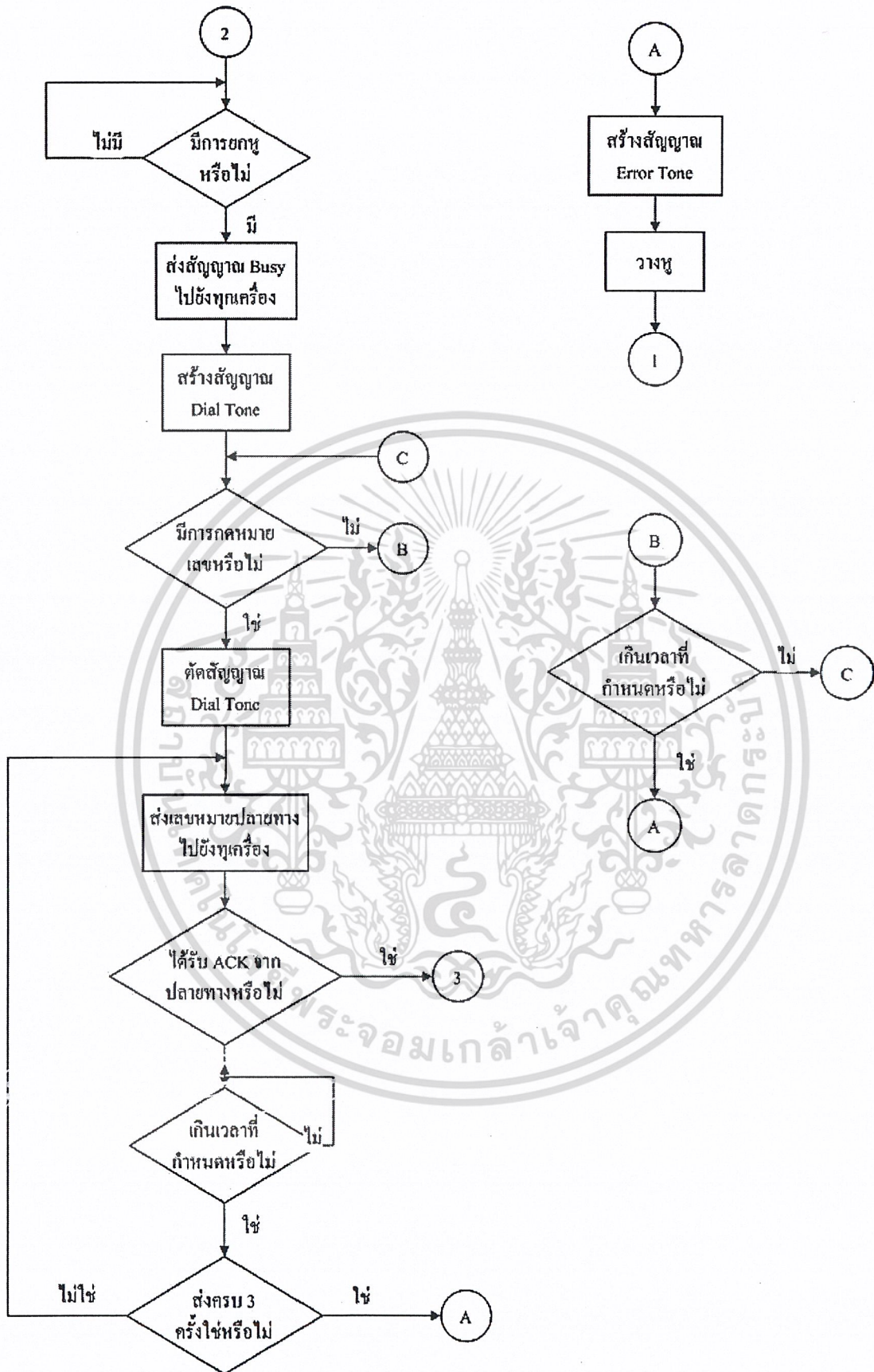


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

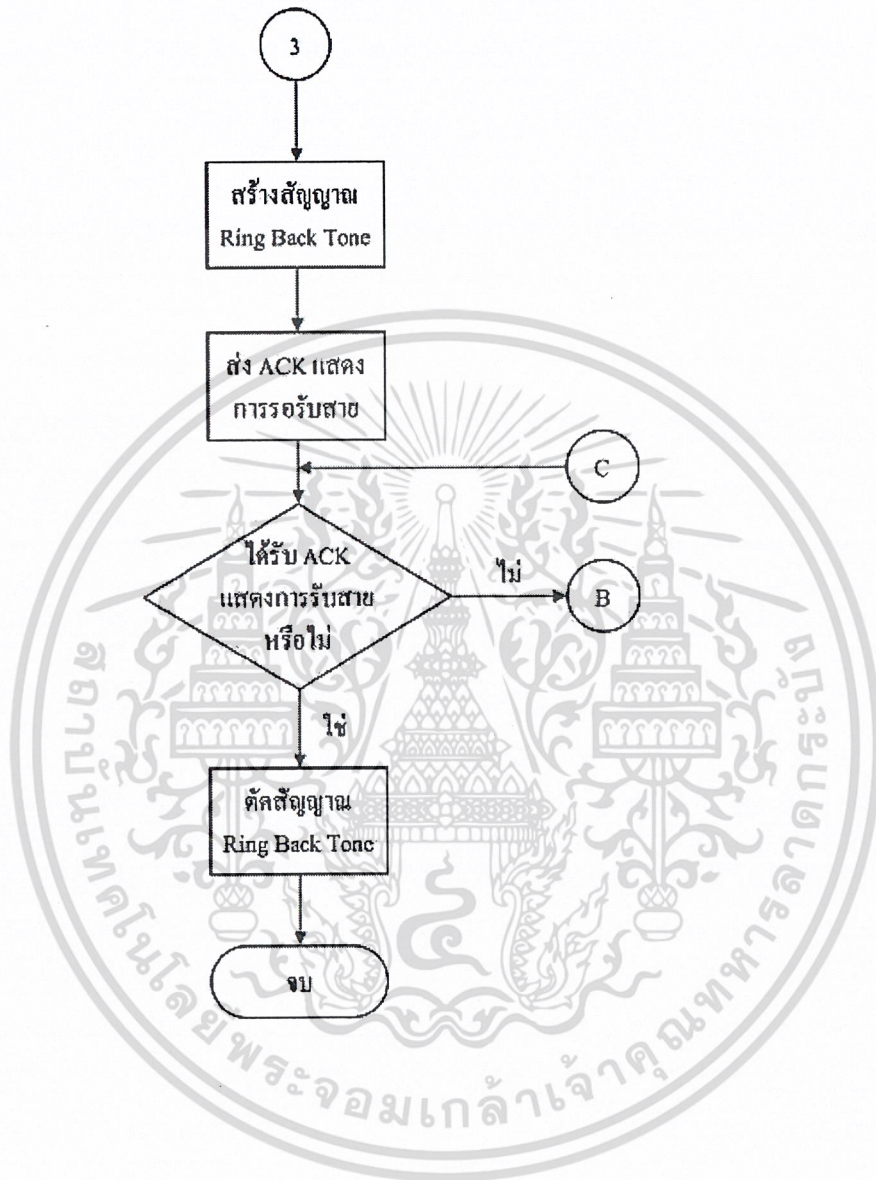
# Flowchart การยกเลิกการติดต่อระหว่างต้นทางและปลายทาง



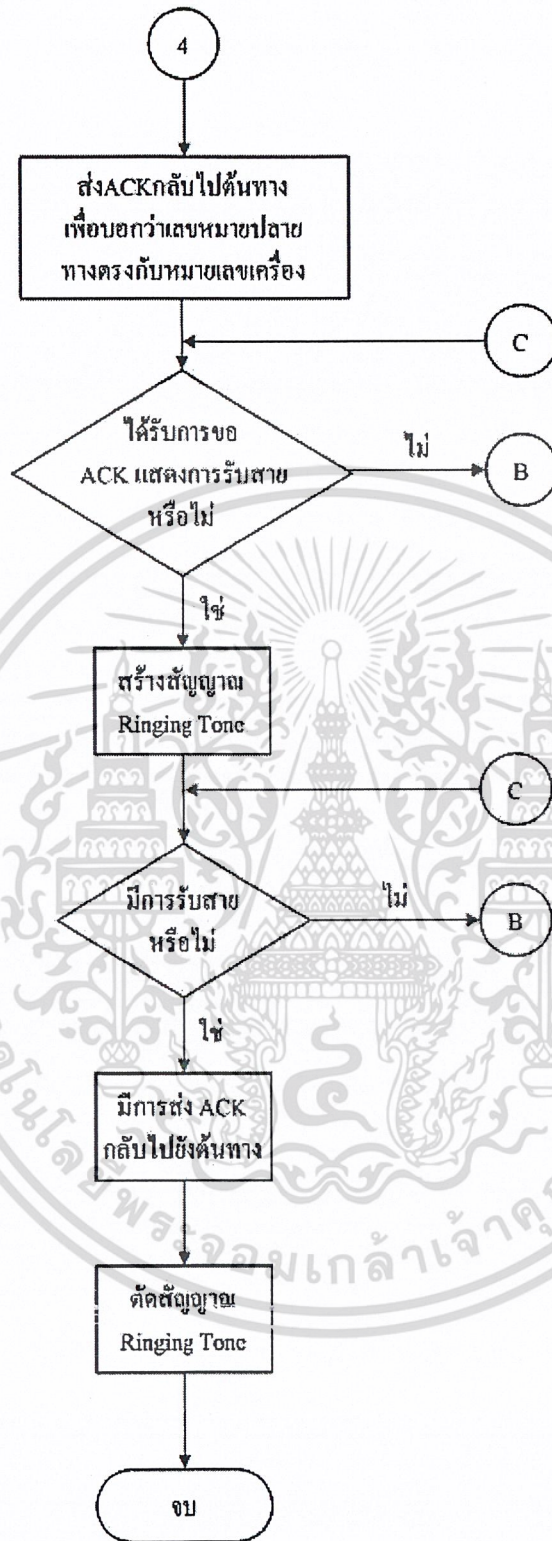
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## โปรแกรมควบคุมการทำงานของระบบโดยไม่โครคอนโทรลเลอร์

```
#pragma code
#include <stdio.h>
#include <reg52.h>
#include <intrins.h>
#define on 1
#define off 0

unsigned int a,b,u,v,q ;
unsigned char m,n ;

sbit status_busy = P0^0 ;
sbit ring = P0^1 ;
sbit tone = P0^2 ;
sbit test1 = P2^4 ;
sbit test2 = P2^5 ;

unsigned char p_send_addrX = 0x01 ;
unsigned char p_ACK_addrX = 0x02 ;
unsigned char p_send_ringback = 0x03 ;
unsigned char p_ACK_ringback = 0x04 ;
unsigned char p_send_disc = 0x05 ;
unsigned char p_ACK_disc = 0x06 ;

unsigned char TX ,RX ;
unsigned char frame[3] ;
unsigned char flag = 0x7E ;
unsigned char frame_add, frame_control, frame_data, frame_FCS ;
unsigned char process ;
unsigned char ID ;
unsigned char err ;
unsigned char _INT1 ;
unsigned char _INT0 ;
unsigned char call ;
unsigned char connect ;
unsigned char back_2_start ;
unsigned char address ;
unsigned char control ;
unsigned char data_FCS ;
bdata unsigned char _data ;
unsigned char _FCS ;

sbit _caller = P1^2 ;
sbit _called = P1^3 ;
sbit Q1 = P1^7 ; /* Q1-Q4 from DTMF decoder */
sbit Q2 = P1^6 ;
sbit Q3 = P1^5 ;
sbit Q4 = P1^4 ;
sbit Id_1 = P2^3 ; /* telephone No. from dip SW. */
sbit Id_2 = P2^2 ;
sbit Id_3 = P2^1 ;
sbit Id_4 = P2^0 ;

void send(unsigned char TX) ;
void parity(unsigned char A,unsigned char C,unsigned char D) ;
void send_frame() ;
void send_busy() ;
void cancel_busy();
void send_addrX();
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

void ACK_addRX()      ;
void send_ringback() ;
void ACK_ringback()  ;
void send_disc()     ;
void ACK_disc()      ;
void delay( unsigned short count ) ;
void dial()          ;
void ringback()      ;
void errortone()     ;
void ringing()       ;
void start()         ;
void caller()        ;
void called()        ;
void chk_frame()     ;
void chk_addRX()     ;
void chk_ringback()  ;
void chk_disc()      ;

void sevice_tell_off(void) interrupt 0
{
    unsigned char i ;
    i = 0 ;
    if (connect == 0xFF)
    {
        while(1)
        {
            send_disc() ;
            u = 0 ;
            v = 20000 ;
            while(1)
            {
                if (back_2_start == 0xFF) return ;
                if (u<2)
                {
                    v-- ;
                    if (v==0)
                    {
                        u++ ;
                        v = 20000 ;
                    }
                    if (u==2) break ;
                }
                if (RI==1) break ;
            }
            chk_frame() ;
            if (SBUF == 0x6B) break ;
            i++ ;
            if (i>5) break ;
        }
        for (i=0; i<5; i++) cancel_busy() ;
        status_busy=0 ;
    }
    else
    {
        for (i=0; i<5; i++) cancel_busy() ;
        for (i=0; i<5; i++) send_disc() ;
    }
    back_2_start = 0xFF ;
    _caller = 1 ;
    _called = 1 ;
    _EXO = 0 ;
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

}

void sevice_DTMF(void) interrupt 2
{
    unsigned char z ;
    z = P1 ;
    m = z & 0xF0 ;

    address = 0x00 ;
    switch (m)
    {
        case 0x10 : address = 0x01 ; break ;
        case 0x20 : address = 0x02 ; break ;
        case 0x30 : address = 0x03 ; break ;
        case 0x40 : address = 0x04 ; break ;
        case 0x50 : address = 0x05 ; break ;
        case 0x60 : address = 0x06 ; break ;
        case 0x70 : address = 0x07 ; break ;
        case 0x80 : address = 0x08 ; break ;
        case 0x90 : address = 0x09 ; break ;
        case 0xA0 : address = 0x0A ; break ;
        default   : while(1) errortone() ;
    }
    address = address + ID ;
    m = address & 0xF0 ;
    n = address & 0x0F ;
    n = n<<4 ;
    _INT1 = 0xFF ;
    EX1   = 0 ; /* disable int. from DTMF decoder */
}

main()
{
    P1 = 0xF0 ;
    P2 = 0xFF ; /* set to input port */
    a = 2500 ; /* set timing of tone gen */
    b = 4*a ;

    /* initial mode & baud rate 1200 */
    PCON = 0x00 ;
    TMOD = 0x20 ;
    TH1  = 0xE8 ;
    TR1  = 1 ;

    /* initial interrupt */
    EA = 1 ;
    EX0 = 0 ;
    EX1 = 0 ;
    IT0 = 0 ;
    IT1 = 0 ;
    status_busy = 0 ;
    while(1) start() ;
}

/* start sub-program */
void start()
{
    unsigned int i,j ;
    unsigned char x ;

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

SCON = 0x50 ;
while (1)
{
    ID = 0x00 ;
    if ( Id_1 == 1 ) ID = ID + 0x80 ;
    if ( Id_2 == 1 ) ID = ID + 0x40 ;
    if ( Id_3 == 1 ) ID = ID + 0x20 ;
    if ( Id_4 == 1 ) ID = ID + 0x10 ;

    back_2_start = 0x00 ;
    tone = 0 ;
    ring = 0 ;
    i = 0 ;
    _caller = 1 ;
    _called = 1 ;
    _INT1 = 0x00;
    process = 0x00 ;
    connect = 0x00 ;
    if ((INT0 == 1)&&(RI != 1)) /* tel on */
    {
        EX0 = 1 ; /* enable int. from tel off */
        if (status_busy == 1)
        {
            while(1)
            {
                errortone() ;
                if (back_2_start == 0xFF ) return ;
            }
        }
        else
        {
            for (j=0; j<5; j++) send_busy() ;
            EX1 = 1 ; /* enable int. from DTMF decoder */
            while ((INT1==1)&&(i<4000))
            {
                dial();
                i++ ;
                if (_INT1 == 0xFF) break ;
                if (back_2_start == 0xFF) return ;
            }
            if ( m == n )
            {
                while(1)
                {
                    errortone() ;
                    if(back_2_start == 0xFF) return ;
                }
            }
            if (INT0 == 0) break ;
            if (i>=4000)
            {
                while(1)
                {
                    errortone() ;
                    if(back_2_start == 0xFF) return ;
                }
            }
            caller() ;
            if (back_2_start == 0xFF ) return ;
        }
    }
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

    }
    if (RI==1) break ;
}
u = 0 ;
if ((SBUF == 0x7E)&&(i==0))
{
    j++ ;
    RI = 0 ;
}
else
{
    frame[i] = SBUF ;
    i++ ;
    RI = 0 ;
}
}
if (frame[3] != 0x7E) break ;
else
{
    while(1)
    {
        frame_add = frame[0] ;
        frame_control = frame[1];
        frame_data = (frame[2]&0xF0) ;
        frame_FCS = (frame[2]&0x0F) ;
        if (call == 0x00)
        {
            x = (frame_add & 0xF0) ;
        }
        if (call == 0xFF)
        {
            x = (frame_add & 0x0F) ;
            x = x<<4 ;
        }
        if (x==ID)
        {
            _FCS = 0x00 ;
            parity(frame_add,frame_control,
            frame_data) ;
            if (frame_FCS == _FCS) err=0x00 ;
            else err = 0xFF ;
        }
        else
        {
            err = 0xFF ;
        }
    }
    if (back_2_start == 0xFF ) return ;
    if (err==0xFF) break ;
    switch (frame_data)
    {
        case 0x10 : chk_addRX() ; break ;
        case 0x20 : chk_ringback() ; break ;
        case 0xA0 : chk_disc() ; break ;
        default : break ;
    }
    break ;
}
}
break ;
}
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

if (INT0 == 0) /* tel off */
{
    call = 0xFF ; /* called */
    if (RI==1)
    {
        x = SBUF ;
        switch (x)
        {
            case 0xBB : status_busy=1 ; break ;
            case 0xCC : status_busy=0 ; break ;
            default : chk_frame() ;
        }
        if (back_2_start == 0xFF ) return ;
        RI = 0 ;
        if ((process == p_send_addRX)&&(INT0 == 0))
            called() ;
        if (back_2_start == 0xFF ) return ;
    }
}
}

void chk_frame()
{
    unsigned int i,j ;
    unsigned char x ;

    if (back_2_start == 0xFF ) return ;
    i = 0 ;
    j = 0 ;
    u = 0 ;
    v = 20000 ;
    process = 0x00 ;
    if (SBUF== 0x7E )
    {SCON = 0x50 ;
    while (1)
    {
        RI = 0 ;
        if (call == 0x00) _INT0 = 0 ; /* caller */
        if (call == 0xFF) /* called */
        {
            if (connect == 0xFF) _INT0 = 0 ;
            else _INT0 = INT0 ;
        }
        if ((call != 0x00)&&(call != 0xFF)) break ;
        i = 0 ;
        j = 0 ;
        while ((_INT0==0)&&(i<4)&&(j<2))
        {
            while(1)
            {
                if (back_2_start == 0xFF ) return ;
                if (u<2)
                {
                    v-- ;
                    if (v==0)
                    {
                        u++ ;
                        v = 20000 ;
                    }
                    if (u==2) return ;
                }
            }
        }
    }
}
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

    }
    else RI = 0 ;
}

void chk_addRX()
{
    switch (frame_control)
    {
        case 0x08 : process = p_send_addRX ; break ;
        case 0x88 : process = p_ACK_addRX ; break ;
        default : break ;
    }
}

void chk_ringback()
{
    switch (frame_control)
    {
        case 0x08 : process = p_send_ringback ; break ;
        case 0x88 : process = p_ACK_ringback ; break ;
        default : break ;
    }
}

void chk_disc()
{
    switch (frame_control)
    {
        case 0x08 : process = p_send_disc ; break ;
        case 0x88 : process = p_ACK_disc ; break ;
        default : break ;
    }
}

void caller()
{
    unsigned int I ;
    unsigned char x ;
    if (back_2_start == 0xFF ) return ;
    call = 0x00 ; /* caller */
    process = 0x00 ;
    send_addRX() ;
    if (back_2_start == 0xFF ) return ;
    for (i=0; i<10; i++)
    {
        u = 0 ;
        v = 20000 ;
        while(1)
        {
            if (back_2_start == 0xFF ) return ;
            if (u<2)
            {
                v-- ;
                if (v==0)
                {
                    u++ ;
                    v = 20000 ;
                }
                if (u==2) break ;
            }
            if (RI==1) break ;

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

}
chk_frame() ;
if (back_2_start == 0xFF ) return ;
if (process == p_ACK_addrX) break ;
else
{
    i++ ;
    send_addrX() ;
    if (back_2_start == 0xFF ) return ;
}
}
if (i>=10)
{
    while(1)
    {
        errortone() ;
        x = SBUF ;
        switch (x)
        {
            case 0x41 : status_busy=1 ; break ;
            case 0x30 : status_busy=0 ; break ;
            default : chk_frame() ;
        }
        if (back_2_start == 0xFF ) return ;
    }
}
send_ringback() ;
if (back_2_start == 0xFF ) return ;
for (i=0; i<20; i++)
{
    ringback() ;
    if (back_2_start == 0xFF ) return ;
    if (process == p_ACK_ringback) break ;
    else
    {
        i++ ;
        send_ringback() ;
        if (back_2_start == 0xFF ) return ;
    }
}
if (i>=20)
while(1)
{
    errortone() ;
    if (back_2_start == 0xFF ) return ;
}
_caller = 0 ;
connect = 0xFF ;
tone = 0 ;
ring = 0 ;
while (1)
{
    while(~RI)
    {
        if (back_2_start == 0xFF ) return ;
    }
    chk_frame() ;
    if (SBUF == 0x61)
    {
        for (i=0; i<20; i++) ACK_disc() ;
        caller = 1 ;

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

        _called = 1 ;
        break ;
    }
    if (back_2_start == 0xFF ) return ;
}
connect = 0x00 ;
status_busy=0 ;
while(1)
{
    errortone() ;
    if (back_2_start == 0xFF ) return ;
}
}

void called()
{
    unsigned int i,q ;

    q = 0x00 ;
    u = 0 ;
    v = 20000 ;
    call = 0xFF ; /* called */
    process = 0x00 ;
    address = frame_add ;
    ACK_addRX() ;
    while(1)
    {
        for (i=0; i<10; i++ )
        {
            u = 0 ;
            v = 20000 ;
            while(1)
            {
                if (back_2_start == 0xFF) return ;
                if (u<2)
                {
                    v-- ;
                    if (v==0)
                    {
                        u++ ;
                        v = 20000 ;
                    }
                    if (u==2) break ;
                }
                if (RI==1) break ;
            }
            chk_frame() ;
            if (process == p_send_ringback) break ;
            else
            {
                i++ ;
                ACK_addRX() ;
            }
        }
        if (i>=10)break ;
        for (i=0; i<10; i++)
        {
            ringing() ;
            if (back_2_start == 0xFF ) return ;
            if (INT0 == 1) break ;
        }
    }
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

if (INT0==0) return ;
EX0 = 1 ; /* enable int. from tel off */
for (i=0; i<5; i++) ACK_ringback() ;
_called = 0 ;
connect = 0xFF ;
back_2_start = 0x00 ;
tone = 0 ;
ring = 0 ;
while (1)
{
    while(~RI) if (back_2_start == 0xFF ) return ;
    chk_frame() ;
    if (SBUF == 0x61)
    {
        for (i=0; i<20; i++) ACK_disc() ;
        _caller = 1 ;
        _called = 1 ;
        break ;
    }
    if (back_2_start == 0xFF ) return ;
}
connect = 0x00 ;
status_busy=0 ;
while(1)
{
    errortone() ;
    if (back_2_start == 0xFF ) return ;
}
}

void send(unsigned char TX)
{
    SCON = 0x40 ;
    SBUF = TX ;
    while(~TI)
    {
        if (back_2_start == 0xFF ) return ;
    }
    TI = 0 ;
}

void parity(unsigned char A,unsigned char C,unsigned char D)
{
    bit aa = 0 ;
    bit bb = 0 ;
    bit cc = 0 ;
    unsigned char i ;
    unsigned char x = 1 ;

    if (back_2_start == 0xFF ) return ;
    for ( i=0 ;i<=7 ;i++)
    {
        if ( i>0 ) x = x*2 ;
        if ( (A & x) != 0 ) aa = ~aa ;
        if ( (C & x) != 0 ) bb = ~bb ;
        if ( (D & x) != 0 ) cc = ~cc ;
    }
    if (cc == 1) _FCS = _FCS + 0x01 ;
    if (bb == 1) _FCS = _FCS + 0x02 ;
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น.อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

        if (aa == 1) _FCS = _FCS + 0x04 ;
    }

void send_frame()
{
    if (back_2_start == 0xFF ) return ;
    _FCS = 0x00 ;
    parity(address,control,_data) ;
    data_FCS = _data + _FCS ;
    send(flag) ;
    send(address) ;
    send(control) ;
    send(data_FCS) ;
    send(flag) ;
    SCON = 0x50 ;
}

void send_busy()
{
    address = 0x41 ;
    control = 0x41 ;
    data_FCS = 0x41 ;
    send(address) ;
    send(control) ;
    send(data_FCS) ;
}

void cancel_busy()
{
    address = 0x30 ;
    control = 0x30 ;
    data_FCS = 0x30 ;
    send(address) ;
    send(control) ;
    send(data_FCS) ;
}

void send_addrX()
{
    if (back_2_start == 0xFF ) return ;
    control = 0x08 ;
    _data = 0x10 ;
    send_frame() ;
}

void ACK_addrX()
{
    if (back_2_start == 0xFF ) return ;
    control = 0x88 ;
    _data = 0x10 ;
    send_frame() ;
}

void send_ringback()
{
    if (back_2_start == 0xFF ) return ;
    control = 0x08 ;
    _data = 0x20 ;
    send_frame() ;
}

void ACK_ringback()
{

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

    if (back_2_start == 0xFF ) return ;
    control = 0x88 ;
    _data = 0x20 ;
    send_frame() ;
}
void send_disc()
{
    if (back_2_start == 0xFF ) return ;
    address = 0x61 ;
    control = 0x61 ;
    data_FCS = 0x61 ;
    send(address) ;
    send(control) ;
    send(data_FCS) ;
}
void ACK_disc()
{
    if (back_2_start == 0xFF ) return ;
    address = 0x6B ;
    control = 0x6B ;
    data_FCS = 0x6B ;
    send(address) ;
    send(control) ;
    send(data_FCS) ;
}
/* tone gen */
void delay( unsigned short count )
{
    unsigned int i,j ;
    for (i=0 ;i<count ;i++)
        for (j=0 ;j<50 ;j++) ;
}
void dial()
{
    tone = 1 ;
    delay(3) ;
    tone = 0 ;
    delay(3) ;
}
void ringback()
{
    unsigned int x,j ;
    for (x=0 ;x<(400) ;x++)
    {
        dial() ;
        if(RI==1)
        {
            chk_frame() ;
            if (process == p_ACK__ringback) return ;
        }
    }
    tone = 0 ;
    for (x=0 ;x<(30000) ;x++)
    {
        for (j=0 ;j<5 ; j++) ;
        if(RI==1)
        {
            chk_frame() ;

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

        if (process == p_ACK_ringback) return ;
    }
}

void errortone()
{
    unsigned int x ;
    for (x=0 ;x<(120) ;x++) dial() ;
    tone = 0 ;
    delay(1000) ;
}

void ringing() /* send to buzzer */
{
    unsigned int i,j ;
    ring = 1 ;
    for (i=0 ;i<a ;i++)
    {
        for (j=0 ;j<50 ;j++) ;
        if (INT0 == 1) return ;
    }
    ring = 0 ;
    for (i=0 ;i<b ;i++)
    {
        for (j=0 ;j<25 ;j++) ;
        if (INT0 == 1) return ;
    }
}

```



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้