สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

ระบบควบคุมอุปกรณ์ไฟฟ้าผ่านสายไฟบ้าน

Data Control Electrical Via AC Line

T055478

โดย

นาย ธนเดช นามโมรา

นาย ธรรมรัตน์ แซ่โค้ว

ปริญญานิพนธ์ฉบับนี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรวิศวกรรมศาสตรบัณฑิต

สาขาวิชาวิศวกรรมอิเล็กทรอนิกส์

คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2546

เลขหมู่
เลขทะเบียน 55478
วัน,เดือน,ปี 10 พ.ค. 2548

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ระบบควบคุมอุปกรณ์ไฟฟ้าผ่านสายไฟบ้าน

Data Control Electrical Via AC Line

โดย

นาย ธเนศร นามโมรา รหัสประจำตัว 43010174

นาย ธรรมรัตน์ แซ่โค้ว รหัสประจำตัว 43010177

อาจารย์ที่ปรึกษา

อ.เทอดศักดิ์ ถิ่วหาทอง

ปริญญานิพนธ์สำหรับปริญญาวิศวกรรมศาสตรบัณฑิต

สาขาวิชาอิเล็กทรอนิกส์

คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2546

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ปริญญานิพนธ์ ปีการศึกษา 2546

ภาควิชา วิศวกรรมอิเล็กทรอนิกส์

คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เรื่อง ระบบควบคุมอุปกรณ์ไฟฟ้าผ่านสายไฟบ้าน

ผู้จัดทำ

1. นาย ธะนตร นามโมรา รหัส 43010174

2. นาย ธรรมรัตน์ แซ่โค้ว รหัส 43010177

.....อาจารย์ที่ปรึกษา

(อ.เทอดศักดิ์ ลิ่วหาทอง)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ระบบควบคุมอุปกรณ์ไฟฟ้าผ่านสายไฟบ้าน

Data Control Electrical Via AC Line

โดย

นาย ธเนตร นามโมรา 43010174

นาย ธรรมรัตน์ แซ่โค้ว 43010177

โครงงานนี้ได้รับการตรวจสอบแล้ว พร้อมที่จะทำการสอบได้

.....อาจารย์ที่ปรึกษา

อ.เทอดศักดิ์ ถิ่วหาทอง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ระบบควบคุมอุปกรณ์ไฟฟ้าผ่านสายไฟบ้าน

นาย ธเนตร นามโมรา รหัส 43010174
นาย ธรรมรัตน์ แซ่โค้ว รหัส 43010177
อ.เทอดศักดิ์ ลิ่วหาทอง
(อาจารย์ที่ปรึกษา)
ปีการศึกษา 2546

บทคัดย่อ

โครงงานนี้นำเสนอการควบคุมอุปกรณ์ไฟฟ้า โดยการส่งสัญญาณควบคุมที่เป็นสัญญาณดิจิตอลผ่านสายไฟบ้าน การทำงานใช้ไมโครคอนโทรลเลอร์ในการรับข้อมูลและสร้างเป็นรหัสควบคุมแล้วทำการส่งออกสายไฟฟ้าผ่านหม้อแปลง ภาครับใช้หม้อแปลงในการรับสัญญาณควบคุมจากสายไฟฟ้าแล้วแปลงรหัสควบคุมให้เป็นข้อมูลเพื่อใช้ในการควบคุมอุปกรณ์ไฟฟ้าต่อไป การรับคำสั่งจากผู้ใช้ทำได้สองทางคือการรับคำสั่งจากคอมพิวเตอร์และทางโทรศัพท์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Data Control Electrical Via AC Line

Thanet Nammora ID 43010174

Thammarat Saekou ID 43010177

Mr. Therdsak Leiwhatong

(Advisor)

Academic Year 2003

ABSTRACT

This project present a system for controlling an electrical appliance. The system comprises two part: a master module and a slave module. The master module consists of two part: 1) an user interface part which receives command to turn on or off the electrical appliance and 2) a transmitting part which uses an AM Modulator for transmitting the control signal through AC-Line. The slave module which connect with the electrical appliance, receive the control signal from AC-Line by using AM Modulator. Then the appliance is turned on or off.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กิตติกรรมประกาศ

ขอกราบขอบพระคุณ คุณพ่อ และคุณแม่ ที่สห้การสนับสนุนด้านการศึกษาคอยดูแลเอาใจใส่และห่วงใยตลอดมาจนการทำงานครั้งนี้สำเร็จลุล่วงไฟได้ด้วยดี

ขอขอบพระคุณ อ. เทอดศักดิ์ ลิ่วหาทอง อาจารย์ที่ปรึกษาของข้าพเจ้า และ อาจารย์หลายๆ ท่านที่ให้ความอนุเคราะห์ทั้งทางด้านความรู้ และอุปกรณ์ในการทำงาน รวมทั้งให้คำปรึกษา และแนะนำแนวทางการทำโครงงานวิทยานิพนธ์นี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ

บทคัดย่อ	I
Abstract	II
กิตติกรรมประกาศ	III
สารบัญ	
บทที่ 1 บทนำ	1
1.1 วัตถุระสงค์ของโครงงาน	1
1.2 ขอบเขตของโครงงาน	1
บทที่ 2 ทฤษฎีและหลักการ	3
2.1 สภาพแวดล้อมของสายไฟฟ้ากำลัง	3
2.2 มาตรฐานที่ต้องคำนึงถึงในการสื่อสารบนสายไฟฟ้ากำลัง	4
2.3 ไมโครคอนโทรลเลอร์ AT89C2051	4
2.4 การใช้งาน IC555 TIMER	5
2.5 วงจรเรโซแนนซ์	6
2.6 วงจรเชื่อมต่อสัญญาณควบคุมกับสายไฟฟ้ากำลัง	7
2.7 วงจรภาครับ	8
2.8 พอร์ตอนุกรม	11
2.9 การเขียนโปรแกรมเพื่อใช้งานพอร์ตอนุกรมด้วยคอนโทรล MSComm	21
บทที่ 3 หลักการโดยรวมของระบบ	29
3.1 รูปแบบการส่งข้อมูล	29
3.2 องค์ประกอบของโมดูล	31
3.3 การทำงานของโปรแกรมเปิดปิดอุปกรณ์ไฟฟ้า	37
3.4 การทำงานของโปรแกรมรับคำสั่งจากโทรศัพท์	39
บทที่ 4 ผลการทดลอง	41
4.1 ทดลองการทำงานของวงจรส่ง	41
4.2 ทดลองการทำงานของวงจรรับ	43
4.3 ทดลองการรับส่งข้อมูลระหว่างโมดูล	44

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.4 ทดลองการรับส่งคำสั่งระหว่างคอมพิวเตอร์และโมดูล 45
4.5 ทดลองการทำงานของโปรแกรมเปิดปิดอุปกรณ์ไฟฟ้า 47
4.6 ทดลองการทำงานของโปรแกรมรับคำสั่งจากโทรศัพท์ 48
บทที่ 5 สรุปและวิจารณ์ผลการทดลอง 50
5.1 สรุปและวิจารณ์ผลการทดลอง 50
5.2 ความสามารถในการทำงาน 50
5.3 ปัญหาของระบบ 51
5.4 การปรับปรุง 51
ภาคผนวก
บรรณานุกรม

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป

	หน้า
รูปที่1.1 Block diagram ของโครงงาน	2
รูปที่ 2.1 สภาพแวดล้อมของสายไฟฟ้ากำลัง	3
รูปที่ 2.2 วงจรภายในของ IC555	5
รูปที่ 2.3 แสดงการต่อ IC555 เป็นวงจร Astable	6
รูปที่ 2.4 วงจรเรโซแนนซ์	6
รูปที่ 2.5 วงจรเชื่อมต่อสัญญาณควบคุมกับสายไฟฟ้ากำลัง	7
รูปที่ 2.6 การจัดรูปวงจรของ Band pass Active Filter	8
รูปที่ 2.7 วงจร Band pass Active Filter ที่ใช้ในโครงงาน	9
รูปที่ 2.8 วงจร Envelope detector และ Low pass	10
รูปที่ 2.9 รูปแบบอย่างง่ายที่สุดของข้อมูลอนุกรม	12
รูปที่ 2.10 รูปแบบอย่างง่ายที่สุดของข้อมูลอนุกรมแบบอะซิงโครนัส	13
รูปที่2.11 การจัดขาของคอนเน็กเตอร์พอร์ตอนุกรมตามมาตรฐาน RS-232	15
รูปที่2.12 การต่ออุปกรณ์ภายนอกเข้ากับคอมพิวเตอร์แบบ Noll modem	16
รูปที่2.13 การต่ออุปกรณ์ภายนอกเข้ากับคอมพิวเตอร์แบบ RS-232 โดยใช้สัญญาณ 3 เส้น	17
รูปที่ 3.1 บล็อกไดอะแกรมโดยรวมของระบบ	29
รูปที่ 3.2 รูปแบบการส่งคำสั่งเปิดปิดอุปกรณ์ไฟฟ้า	30
รูปที่ 3.3 รูปแบบการส่งคำสั่งตรวจดูสถานะ	30
รูปที่3.4 องค์ประกอบของโมดูล	31
รูปที่ 3.5 วงจรตรวจสอบ Zero Crossing	31
รูปที่ 3.6 วงจรส่ง	32
รูปที่ 3.7 วงจรภาครับ	32
รูปที่ 3.8 ไมโครคอนโทรลเลอร์ของ Master Module	33
รูปที่ 3.9 ไมโครคอนโทรลเลอร์ของ Slave Module	34
รูปที่ 3.10 วงจรเปิดปิดอุปกรณ์ไฟฟ้า	35
รูปที่ 3.11 วงจรตรวจสอบอุปกรณ์ไฟฟ้า	35
รูปที่ 3.12 รูปแบบการส่งและรับข้อมูลของโมดูล	36

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 3.13 หน้าต่างตั้งค่าโปรแกรม 37
รูปที่3.14 หน้าต่าง ระบบ เปิด ปิด อุปกรณ์ไฟฟ้า 38
รูปที่ 3.15 หน้าต่างยืนยันการเปิดหรือปิดอุปกรณ์ปลายทางหมายเลข 01 38
รูปที่ 3.16 หน้าต่าง บอกว่าเกิดการผิดพลาดในการส่งข้อมูลถึงอุปกรณ์ปลายทางหมายเลข 01 39
รูปที่ 3.17 หน้าต่างตั้งค่าของโปรแกรมรับคำสั่งจากโทรศัพท์ 39
รูปที่ 3.18 หน้าต่างโปรแกรมรับคำสั่งจากโทรศัพท์ 40
รูปที่ 4.1 สัญญาณที่ ขา 11 ของไมโครคอนโทรลเลอร์ขณะส่งข้อมูล 1111 41
รูปที่ 4.2 สัญญาณที่ ขา 3 ของไอซี555ขณะส่งข้อมูล 1111 42
รูปที่ 4.3 สัญญาณพาร์ที่จุด 0 องศาของไฟฟ้ากำลัง 42
รูปที่ 4.4 สัญญาณพาร์ที่จุด 180 องศาของไฟฟ้ากำลัง 43
รูปที่ 4.5 สัญญาณที่ ขาหม้อแปลง ไอเอฟ ของวงจรรับขณะรับข้อมูล 1111 43
รูปที่ 4.6 สัญญาณที่ ขา 8 ของไมโครคอนโทรลเลอร์ขณะรับข้อมูล 1111 44
รูปที่ 4.7 ผลการทดลองการรับส่งข้อมูลระหว่างโมดูล 44
รูปที่ 4.8 ข้อมูลที่ตอบจากโมดูลเมื่อได้รับคำสั่ง InitialX 45
รูปที่ 4.9 ข้อมูลที่ตอบจากโมดูลเมื่อได้รับคำสั่ง O011X และโมดูลหมายเลข 01 ทำงาน 45
รูปที่ 4.10 ข้อมูลที่ตอบจากโมดูลเมื่อได้รับคำสั่ง O011X และโมดูลหมายเลข 01 ไม่ทำงาน 45
รูปที่ 4.11 ข้อมูลที่ตอบจากโมดูลเมื่อได้รับคำสั่ง StateX
และสถานะโมดูลหมายเลข 01 เป็น อุปกรณ์ปิด(100) 46
รูปที่ 4.12 ข้อมูลที่ตอบจากโมดูลเมื่อได้รับคำสั่ง StateX
และสถานะโมดูลหมายเลข 01 เป็น อุปกรณ์เปิด(110) 46
รูปที่ 4.13 ข้อมูลที่ตอบจากโมดูลเมื่อได้รับคำสั่ง StateX
และสถานะโมดูลหมายเลข 01 เป็น อุปกรณ์เสีย(010) 46
รูปที่ 4.14 ผลการทดลองโปรแกรมเปิดปิดอุปกรณ์ไฟฟ้า 47
รูปที่ 4.15 โปรแกรมรับคำสั่งจากโทรศัพท์ขณะรับสายและรับคำสั่งเปิดอุปกรณ์หมายเลข 01 48

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญตาราง

	หน้า
ตาราง 2.1 ช่วงความถี่ที่ใช้ได้ตามมาตรฐาน EN50065-1	4
ตาราง 2.2 บิตพาริตี้ของข้อมูล	14
ตาราง 2.3 การจัดขา DB-9 และ DB-25	16
ตาราง 2.4 ข้อมูลในแอดเดรส 0000:0411 H ที่ใช้แจ้งจำนวนพอร์ตอนุกรม	21

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่1
บทนำ

ในปัจจุบันมีความคิดมากมายเพื่อให้การดำรงชีวิตประจำวันมีความสะดวกสบายมากขึ้น หนึ่งในนั้นคือ บ้านอัตโนมัติ เป็นบ้านที่ผู้อยู่อาศัยสามารถสั่งปิดเปิดอุปกรณ์ไฟฟ้าภายในบ้านได้จากเครื่องควบคุมเพียงตัวเดียว ซึ่งเครื่องควบคุมนี้อาจจะเป็น เครื่องควบคุมระยะไกล เก้าอี้ที่ใช้นั่งประจำ หรืออาจเป็นจากหน้าจอ PC อันจะทำให้สามารถควบคุมผ่านทาง INTERNET หรือโทรศัพท์มือถือได้ ซึ่งแนวความคิดบ้านอัตโนมัตินี้มีประโยชน์อย่างมากสำหรับผู้อยู่อาศัยที่เป็น ผู้ชรา หรือ ผู้พิการ

โครงงานนี้จะกล่าวถึงการควบคุมอุปกรณ์ไฟฟ้าผ่านสายไฟบ้าน โดยจะสามารถควบคุมการปิดเปิดอุปกรณ์ไฟฟ้าด้วยการตัดหรือต่อไฟฟ้าเข้าอุปกรณ์นั้นๆ การทำงานจะทำงานโดยการส่งข้อมูลดิจิตอลจำนวน 12 บิท ต่อ 1 คำสั่งลงในสายไฟบ้าน ข้อมูล 12 บิทนี้ 4 บิทแรกจะเป็น รหัสเริ่มต้น 6 บิทต่อมาเป็น ข้อมูลที่อยู่ของอุปกรณ์ตัวลูก ส่วนข้อมูล 2 บิทสุดท้ายเป็นคำสั่งเปิดหรือปิดและข้อมูลพาริตี้ อุปกรณ์ตัวลูกจะทำการนำสัญญาณจากสายไฟบ้านมาแปลงเป็นสัญญาณดิจิตอล แล้วตรวจข้อมูลที่อยู่ว่าตรงกับที่อยู่ของตัวเองหรือไม่ ถ้าตรงจะทำการเปิดหรือปิดอุปกรณ์ไฟฟ้าที่ควบคุมอยู่ ซึ่งจะมีตัวลูกได้ทั้งหมด 63 ที่อยู่

รูปแบบการส่งข้อมูลจะใช้ zero crossing ของสัญญาณไฟฟ้ากำลังเป็นจุดเปรียบเทียบ โดยบิทข้อมูลที่เป็น 1 จะใส่สัญญาณความถี่ 98 kHz ขนาด 1 Vp-p เป็นเวลา 1ms และไม่ใส่สัญญาณใดๆเมื่อบิทข้อมูลเป็น 0

1.1 วัตถุประสงค์ของโครงงาน

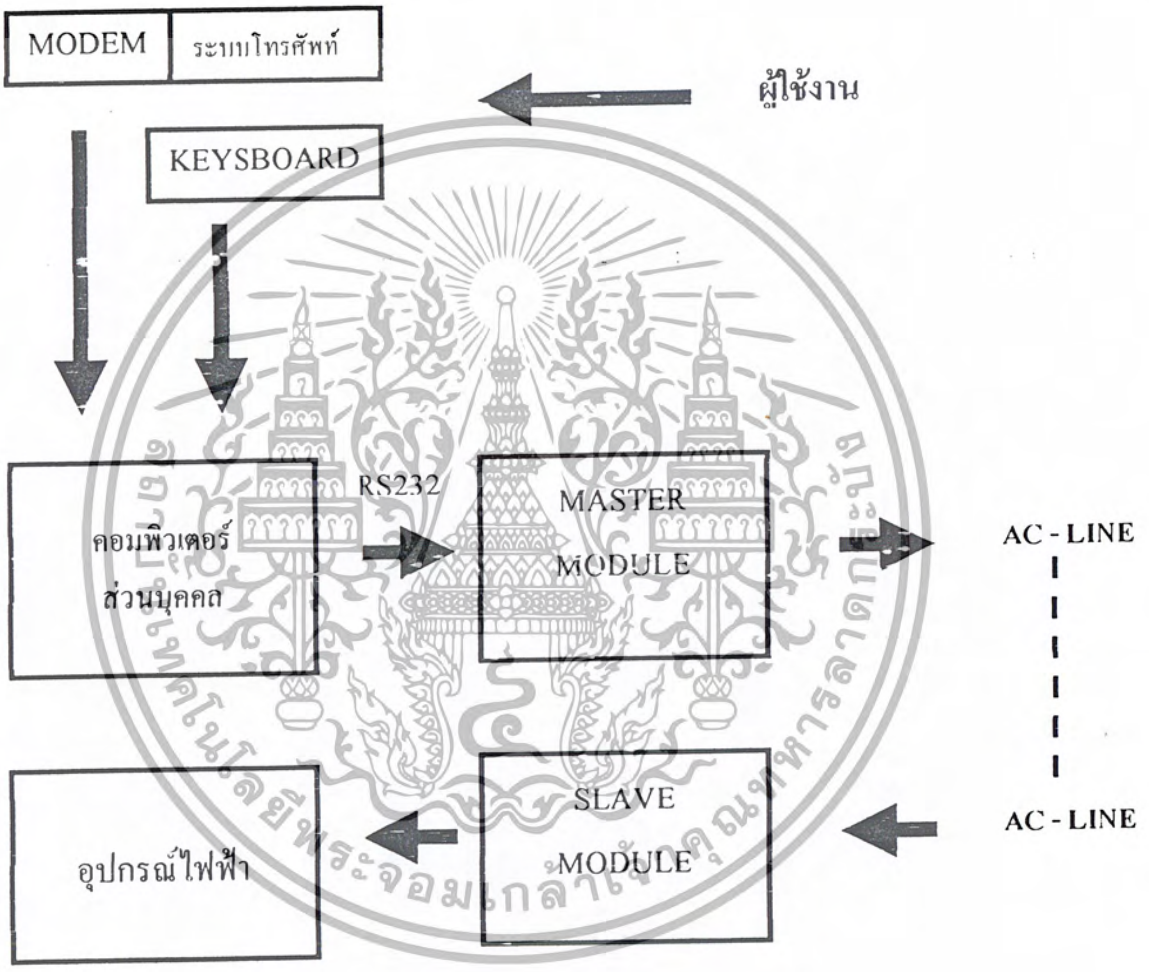
เพื่อศึกษาและออกแบบอุปกรณ์ในการส่งสัญญาณสื่อสารบนระบบไฟฟ้ากำลัง โดยใช้ความรู้ทางด้านอิเล็กทรอนิกส์ โปรแกรมและการประยุกต์ใช้งานไมโครคอนโทรลเลอร์ ศึกษาการสื่อสารผ่านมาตรฐาน RS232 และการสร้างโปรแกรมด้วยภาษา Visual Basic

1.2 ขอบเขตของโครงงาน

เนื้อหาของปริญญานิพนธ์นี้ จะเริ่มจากศึกษาการส่งข้อมูล แล้วทำการออกแบบและสร้างอุปกรณ์ในการรับส่งข้อมูลผ่านระบบไฟฟ้ากำลัง ออกแบบวงจรปิดเปิดและตรวจสอบการกิน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กระแสของอุปกรณ์ไฟฟ้า สุดท้ายคือเขียนโปรแกรมติดต่อกับผู้ใช้งานและโปรแกรมรับคำสั่งจากสายโทรศัพท์ผ่านทางโมเดม



รูปที่1.1 Block diagram ของโครงงาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 2
ทฤษฎีและหลักการ

2.1 สภาพแวดล้อมของสายไฟฟ้ากำลัง

เนื่องจากสายส่งของระบบไฟฟ้ากำลังถูกออกแบบมาเพื่อใช้ในการส่งสัญญาณไฟฟ้ากำลังที่ความถี่ 50 Hz เท่านั้น โดยไม่คำนึงถึงการนำมาส่งสัญญาณสื่อสารความถี่สูงดังนั้นปัญหาสำคัญที่อาจจะเกิดขึ้นกับการส่งสัญญาณสื่อสารดังนี้

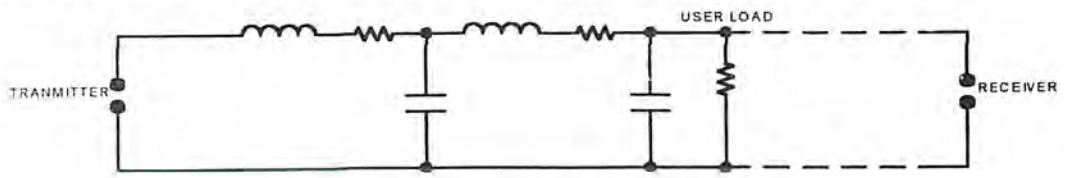
2.1.1 ปัญหาการลดทอนของสัญญาณสื่อสาร

เกิดจากการเปลี่ยนแปลงอิมพีแดนซ์ของสายไฟฟ้า ซึ่งอิมพีแดนซ์จะเปลี่ยนแปลงเนื่องมาจากอุปกรณ์ที่ต่อเข้ากับระบบไฟฟ้า การที่จุดเชื่อมต่อของสายไฟฟ้ามีอิมพีแดนซ์ไม่เท่ากันสัญญาณสื่อสารบางส่วนจะสะท้อนกลับทำให้เกิดการลดทอน จากการติดตั้งสายไฟฟ้าหลายเส้นทาง เนื่องมาจากระบบไฟฟ้ากำลังจะมีการส่งสัญญาณไฟฟ้าไปตามสายไฟฟ้าหลายเส้นทางและความยาวของสายไฟฟ้าก็จะต่างกัน สัญญาณสื่อสารที่ส่งไปในสายไฟฟ้าเส้นต่างๆจะมีเฟสต่างกันหากมาบรรจบกันสัญญาณอาจถูกลดทอน หรืออาจหายไปที่จุดบรรจบนั้นมีสัญญาณสื่อสารจากสองเส้นทางที่มีเฟสต่างกัน 180 องศาพอดี

2.1.2 ปัญหาสัญญาณรบกวนต่างๆในระบบไฟฟ้ากำลัง

ในระบบไฟฟ้ากำลังมีสัญญาณรบกวนมากมายที่สำคัญคือ สัญญาณรบกวนจากการทำงานของวงจร DIMMER ทำให้เกิดสไปรส์ขนาด 20V ถึง 50V ในช่วงความถี่ 100Hz ถึง 120Hz สัญญาณรบกวนจาก SWITCHING POWER SUPPLY จะทำให้เกิดสัญญาณรบกวนในช่วงความถี่ 20kHz ถึง 1MHz สัญญาณจากการส่งของ INTERCOM ซึ่งจะส่งคลื่นพาห์ในช่วงความถี่ 150kHz ถึง 500kHz และจากการทำงานของโหลดประเภทอินดัคทีฟจะเกิดสัญญาณรบกวนอิมพัลซ์ขึ้นในระบบสายส่งระบบไฟฟ้ากำลัง

อิมพีแดนซ์ภายในสายส่งมีค่าประมาณ 10-100 โอห์ม ขนาดของตัวเก็บประจุภายในสายประมาณ 10-1000 พิโคฟารัดต่อ 1000 ฟุต ขนาดความต้านทานในสายประมาณ 0.1-1 โอห์มต่อ 1000ฟุต ความต้านทานในสายยังมีความไม่แน่นอนขึ้นอยู่กับจำนวนอุปกรณ์ไฟฟ้าอีกด้วย



รูปที่ 2.1 สภาพแวดล้อมของสายไฟฟ้ากำลัง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.2 มาตรฐานที่ต้องคำนึงถึงในการสื่อสารบนสายไฟฟ้ากำลัง

มาตรฐาน EN50065-1 เป็นมาตรฐานที่เกี่ยวข้องกับการสื่อสารบนสายไฟฟ้ากำลังมีข้อกำหนดดังนี้

2.2.1 การเลือกความถี่

มาตรฐาน EN50065-1 กำหนดความถี่ของการส่งสัญญาณสื่อสารบนสายไฟฟ้ากำลังไว้ที่ความถี่ 3 kHz ถึง 148 kHz ซึ่งแบ่งออกเป็น 4 ช่วงความถี่ดังนี้

ตาราง 2.1 ช่วงความถี่ที่ใช้ได้ตามมาตรฐาน EN50065-1

Electricity suppliers 3 kHz – 95 kHz	Home systems 95 kHz – 125 kHz	Home system 125 kHz – 140 kHz	Alarm & security 140 kHz – 148 kHz
	Free access	Access rule	

ในโครงงานนี้เลือกใช้ความถี่ 98kHz ซึ่งอยู่ในช่วงที่ใช้ได้อย่างอิสระ

2.2.2 ค่าแรงดันในการส่ง

ค่าแรงดันในการส่งภายในที่อยู่อาศัยกำหนดไว้ไม่เกิน 3.56 Vpp ส่วนในสถานที่พิเศษ เช่น โรงงานอุตสาหกรรม ถูกกำหนดไว้ไม่เกิน 23.8 Vpp

2.3 ไมโครคอนโทรลเลอร์ ATMEL 89C2051

ไมโครคอนโทรลเลอร์ ATMEL 89C2051 เป็นไมโครคอนโทรลเลอร์ขนาดเล็กเป็น ไอซี 20 ขา ที่ใช้งานง่ายมีรูปแบบคำสั่งและสถาปัตยกรรมโดยทั่วไปเหมือนกับไมโครคอนโทรลเลอร์ตระกูล MCS-51 คุณสมบัติที่สำคัญดังนี้

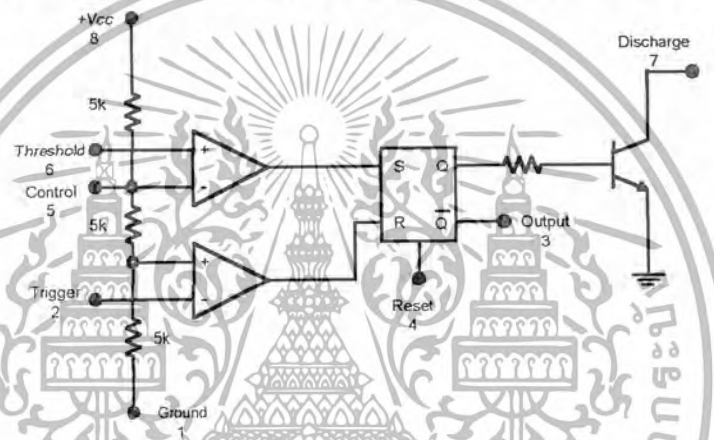
-มีหน่วยความจำโปรแกรมแฟลชขนาด 2 กิโลไบต์สามารถโปรแกรมซ้ำใหม่ได้1,000ครั้ง

- มีหน่วยความจำ RAM 8 บิท ขนาด 128 ไบต์บรรจุอยู่ภายใน
- ทำงานได้ที่ช่วงแรงดันตั้งแต่ 2.7-6 โวลต์
- ใช้ความถี่สัญญาณนาฬิกาได้ตั้งแต่ 0-24 เมกะเฮิรตซ์
- มีพอร์ตอินพุท/เอาท์พุทให้ใช้งาน 15 ตำแหน่ง
- มีระบบป้องกันการคัดลอกโปรแกรม 2 ระดับ
- มีวงจรไทม์เมอร์และวงจรนับ 16 บิท 2 ชุด
- มีวงจรเปรียบเทียบสัญญาณอะนาลอก 1 ช่องสัญญาณ
- มีระบบประหยัดพลังงาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.4 การใช้งาน IC555 TIMER

IC555 TIMER เป็น ไอซี 8 ขา ซึ่งเอาไว้ต่อกับอุปกรณ์ภายนอกเพื่อที่จะให้ ไอซี ทำงานในแบบต่างๆ จากรูปที่ 2.3 แสดงวงจรภายในของ IC555 โดยสัญญาณ Theshold จะเป็นขา 6 สัญญาณ Control เป็นขา 5 ซึ่งการใช้งานทั่วไปนั้นสัญญาณ Control นี้มักจะไม่ได้ใช้ ดังนั้นค่าของ Control voltage จึงเท่ากับ +2Vcc/3 ซึ่งจะได้จากการแบ่งแรงดันของตัวต้านทาน 5kΩ 3 ตัว ขา Collector ของทรานซิสเตอร์ที่จะใช้สำหรับดิสชาร์ตตัวเก็บประจุก็จะเป็นขา 7 ของไอซี สัญญาณที่ได้จากขา Q ของ RS-Flip Flop ก็จะเป็นขา 3 ส่วนสัญญาณ Reset เป็นขา 4 ขา Reset นี้เมื่อถูกต่อเข้ากับ Ground แล้ว IC555 นี้จะไม่ทำงานซึ่งก็เหมือนสวิทช์ ปิดเปิด การทำงานนั้นเอง

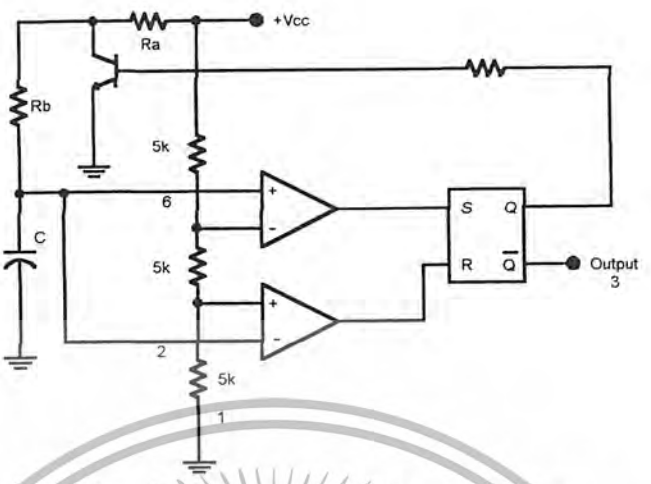


รูปที่ 2.2 วงจรภายในของ IC555

2.4.1 การใช้งาน IC555 เป็นวงจร Astable

จากรูปที่ 2.4 แสดงการต่อ IC555 TIMER ทำงานเป็นแบบ Astable หรือ free-run เอาท์พุทที่ได้ของวงจรจะเป็นสัญญาณคลื่นสี่เหลี่ยมที่ต่อเนื่องกันออกมา เราจะสามารถอธิบายการทำงานได้คือ เมื่อเอาท์พุท Q มีสถานะ Low ทรานซิสเตอร์ก็จะ คัทออฟ ตัวเก็บประจุ C จะถูกชาร์ตผ่านตัวต้านทานรวมของ $R_A + R_B$ ดังนั้นค่า time constant ก็จะเป็น $(R_A + R_B)C$ เมื่อตัวเก็บประจุถูกชาร์ตค่า Theshold Voltage ก็จะมีค่าเพิ่มขึ้น เมื่อค่า Theshold Voltage มีค่าเพิ่มขึ้นจนมากกว่าค่า +2Vcc/3 ออฟแอมตัวบนก็จะมีเอาท์พุทเป็น High ทรานซิสเตอร์ก็จะ Saturate ซึ่งก็เหมือนการชอร์ตขา 7 ลง Ground ขณะนี้ตัวเก็บประจุ C ก็จะถูกดิสชาร์ตโดยผ่านทาง R_B ดังนั้นค่า time constant ของการดิสชาร์ตก็จะเป็น R_B C เมื่อค่าของแรงดันตกคร่อมตัวเก็บประจุ C มีค่าลดลงจนน้อยกว่า +Vcc/3 ออฟแอมตัวล่างก็จะมีเอาท์พุทเป็น High ซึ่งจะ Reset RS-Flip Flop เนื่องจากค่า time constant ของการดิสชาร์ท สัญญาณเอาท์พุทที่ดึ้งไม่สมมาตร ซึ่งค่าคาบเวลาของสัญญาณในขณะที่มีสถานะ High จะยาวนานกว่าในขณะที่มีสถานะ Low

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.3 แสดงการต่อ IC555 เป็นวงจร Astable

สามารถคำนวณค่า duty cycle ได้จาก

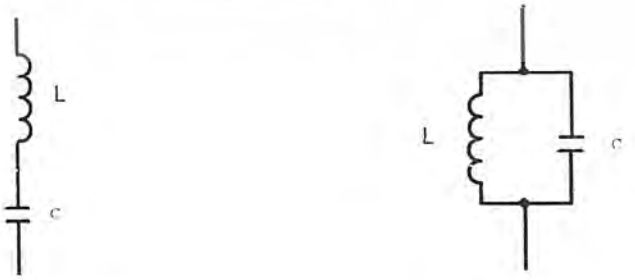
$$D = \frac{R_A + R_B}{R_A + 2R_B} \times 100\% \qquad (2.1)$$

สามารถคำนวณหาค่าความถี่ของสัญญาณเอาท์พุทได้จาก

$$f = \frac{1.44}{(R_A + 2R_B)C} \qquad (2.2)$$

2.5 วงจรเรโซแนนซ์

เป็นวงจรที่ต่อรวมกันระหว่าง ตัวเก็บประจุ กับขดลวดเหนี่ยวนำเพื่อวัตถุประสงค์ให้วงจรเรโซแนนซ์เลือกความถี่



(ก) แบบอนุกรม (ข) แบบขนาน

รูปที่ 2.4 วงจรเรโซแนนซ์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

วงจรเรโซแนนซ์จะยอมให้ความถี่ค่าหนึ่งผ่านได้ดี เรียกว่าความถี่เรโซแนนซ์ ที่ความถี่เรโซแนนซ์วงจรเรโซแนนซ์แบบอนุกรมจะมีกระแสผ่านวงจรมากที่สุดโดยมีแรงดันตกคร่อมต่ำที่สุด ส่วนวงจรแบบขนานกระแสจะไหลผ่านน้อยโดยมีแรงดันตกคร่อมมากที่สุด ซึ่งจะคำนวณหาค่าความถี่เรโซแนนซ์ได้จาก

$$Fr = 1/(2\pi\sqrt{LC}) \qquad (2.3)$$

Fr เป็น ค่าความถี่เรโซแนนซ์(Hz)

L เป็น ค่าความเหนี่ยวนำของขดลวด(H)

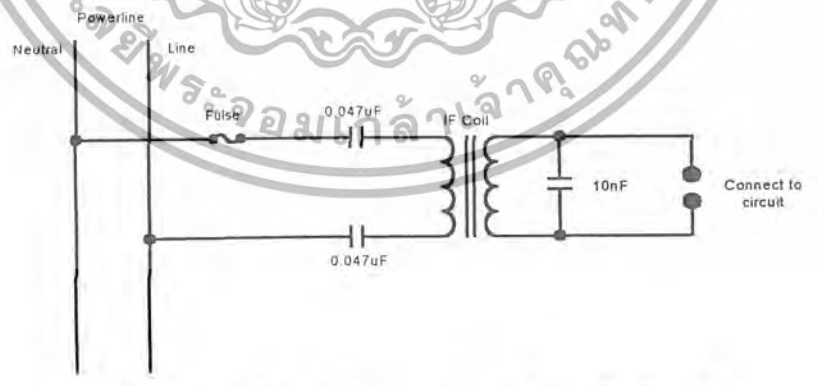
C เป็น ค่าความจุของตัวเก็บประจุ(F)

สมการนี้สามารถใช้คำนวณได้ทั้งแบบขนาน และแบบอนุกรม

2.6 วงจรเชื่อมต่อสัญญาณควบคุมกับสายไฟฟ้ากำลัง

ใช้หม้อแปลงเป็นตัวเชื่อมต่อโดยทำหน้าที่เป็นวงจรเรโซแนนซ์ และกรองความถี่ผ่าน ส่งสัญญาณเข้าไปในสายไฟฟ้ากำลัง และป้องกันอันตรายจากสัญญาณไฟฟ้ากำลังโดยทำหน้าที่แยกกราวด์ของวงจรกับระบบไฟฟ้าออกจากกัน

ในโครงงานนี้เลือกใช้แปลงไอเอฟของวิทยุเอ็มชนิดสีดำมาใช้ ด้านปฐมภูมินำมาต่อขนานกับตัวเก็บประจุเพื่อให้เป็นวงจรเรโซแนนซ์ทำการวัดค่าความเหนี่ยวนำ ได้ค่าความเหนี่ยวนำอยู่ระหว่าง 126uH ถึง 265uH เลือกใช้ที่ค่าความเหนี่ยวนำเท่ากับ 263uH คำนวณหาค่าตัวเก็บประจุจากสมการ (2.3) ได้ค่าเท่ากับ 10nF



รูปที่ 2.5 วงจรเชื่อมต่อสัญญาณควบคุมกับสายไฟฟ้ากำลัง

ส่วนด้านทุติยภูมิของหม้อแปลงไอเอฟ จะเป็นส่วนที่ติดต่อกับด้านไฟฟ้ากำลัง โดยก่อนเข้าสู่สายไฟฟ้าต้องผ่านตัวเก็บประจุเพื่อแยกระหว่างความถี่สูงและความถี่ต่ำออกจากกัน และป้องกันแรงดันสูงจากสายไฟฟ้ากำลัง วัดค่าความเหนี่ยวนำด้านทุติยภูมิตำแหน่งที่ด้านปฐมภูมิมีค่าความ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

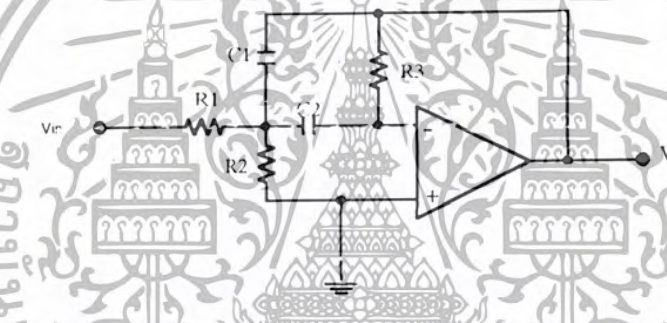
เหนี่ยวนำเท่ากับ 214uH ได้เท่ากับ 75uH คำนวณหาค่าตัวเก็บประจุจากสมการ (2.1) ได้เท่ากับ 23.4nF ดังนั้นใช้ ตัวเก็บประจุขนาด 47nF สองตัวต่อกับหม้อแปลงไอเอฟทางด้านทุติยภูมิ

2.7 วงจรภาครับ

2.7.1 วงจร Active Band Pass Filter แบบ Negative Feedback Topology เป็นวงจรที่เป็นfilter และสามารถขยายสัญญาณได้ด้วย คุณสมบัติของวงจรนี้คือ

- สามารถปรับค่าความถี่กลาง (f_0) ด้วยค่า R_2 เพียงตัวเดียว
- ถ้าค่า Q มีค่าน้อยกว่า 10 sensitivity ของ Q และ f_0 จะเปลี่ยนแปลงน้อยเมื่อค่าของอุปกรณ์เกิดเปลี่ยนแปลง

มีการจัดรูปวงจรเป็น



รูปที่ 2.6 การจัดรูปวงจรของ Active Band Pass Filter

จะได้ทรานสเฟอร์ฟังก์ชันเป็น

$$T = \frac{V_o}{V_{IN}} = \frac{\frac{S}{R_1C_1}}{S^2 + S\left(\frac{1}{R_2C_2} + \frac{1}{R_2C_1}\right) + \frac{1}{R_1R_2C_1C_2} + \frac{1}{R_2R_3C_1C_2}} \qquad (2.4)$$

โดยการกำหนดให้ $C = C_1 = C_2$ จะได้

$$R_1 = \frac{Q}{2\pi f_0 C A_F} \qquad (2.5)$$

$$R_2 = \frac{Q}{2\pi f_0 C (2Q^2 - A_F)} \qquad (2.6)$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$$R_3 = \frac{Q}{\pi f_0 C} \qquad (2.7)$$

ในโครงงานต่อเป็น 2 ภาคต่อกันภาคแรกต้องการ $A_{F1} = 2$, $Q = 5$ และ $f_0 = 98kHz$ ใช้ค่า C เท่ากับ 1.2nF คำนวณค่า R_1, R_2 , R_3 ได้

$R_1 = 5 / (2\pi \times 98k \times 1.2n \times 2) = 3383\ \Omega$ เลือกใช้ค่า $3.3k\Omega$

$R_2 = 5 / (2\pi \times 98k \times 1.2n)(2\times5^2 - 2) = 140\Omega$ เลือกใช้ค่า 150Ω

$R_3 = 5 / (\pi \times 98k \times 1.2n) = 13533\Omega$ เลือกใช้ค่า $14k\Omega$

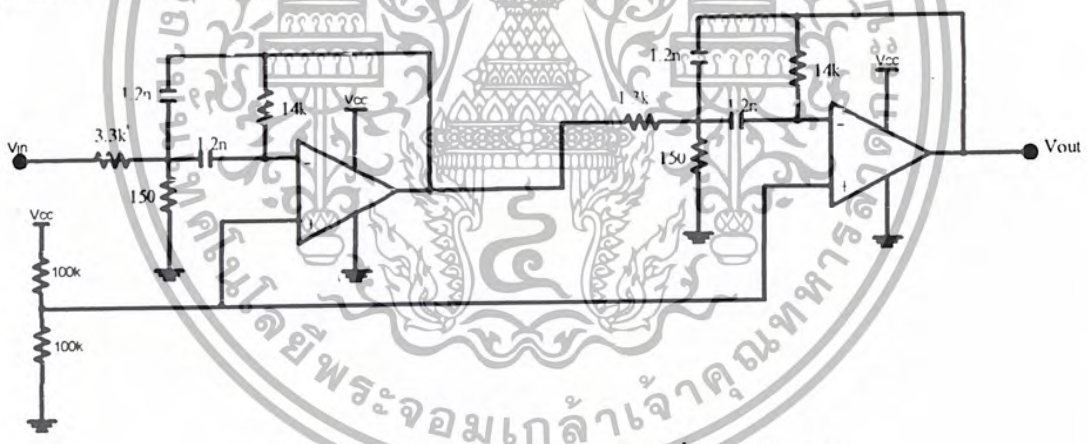
ภาคที่สองต้องการ $A_{F2} = 5$, $Q = 5$ และ $f_0 = 120kHz$ ใช้ค่า C เท่ากับ 1.2nF คำนวณค่า R_1, R_2 , R_3 ได้

$R_1 = 5 / (2\pi \times 98k \times 1.2n \times 5) = 1353\ \Omega$ เลือกใช้ค่า $1.3k\Omega$

$R_2 = 5 / (2\pi \times 98k \times 1.2n)(2\times5^2 - 5) = 150\Omega$ เลือกใช้ค่า 150Ω

$R_3 = 5 / (\pi \times 98k \times 1.2n) = 13533\Omega$ เลือกใช้ค่า $14k\Omega$

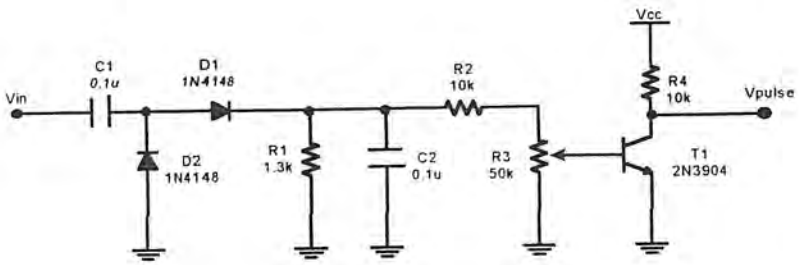
จะได้อัตราขยายรวมเป็น $A_F total = A_{F1} \times A_{F2} = 2 \times 5 = 10$



รูปที่ 2.7 วงจร Active Band Pass Filter ที่ใช้ในโครงงาน

2.7.2 วงจรEnvelope detector และ Low pass เป็นวงจรที่ทำการตัดความถี่คลื่นพาร์ (98kHz)ออกแหลือแค่ส่วนที่เป็นกรอบทางบวก(500Hz)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.8 วงจร Envelope detector และ Low pass

การทำงาน C_1 ทำหน้าที่ตัดแรงดันไฟตรงจากวงจรขยายออก ไดโอด D_1 ยอมให้แรงดันที่เป็นบวกเท่านั้นที่ผ่านได้ เพราะฉะนั้นที่หลัง ไดโอด D_1 จะได้สัญญาณเป็นแรงดันรูปซายน์ซีกบวก R_1, R_2,R_3 และ C_2 ทำงานเป็นวงจร Low pass ซึ่งจะมีความถี่คัทออฟที่

$$F_{cut\,off} = \frac{1}{2\pi C_2 R} \qquad (2.8)$$

โดยที่ $R = R_1 // (R_2 + R_3)$ เนื่องจาก $R_1 << (R_2 + R_3)$ ดังนั้น $R \approx R_1$ คำนวณความถี่คัทออฟได้

$$F_{cut\,off} = 1/(2\pi\ 0.1u\ 1.3k) = 1.2\ kHz$$

สัญญาณที่ผ่านวงจร Low pass แล้วจะเข้าทรานซิสเตอร์ T1 ที่ต่อเป็นสวิทช์ไว้เพื่อแปลงสัญญาณเป็นพัลส์เข้าไมโครคอนทรอลเลอร์ ส่วน D1 ใช้ดิสชาร์ต C1 ในช่วงที่สัญญาณเป็นลบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.8 พอร์ตอนุกรม

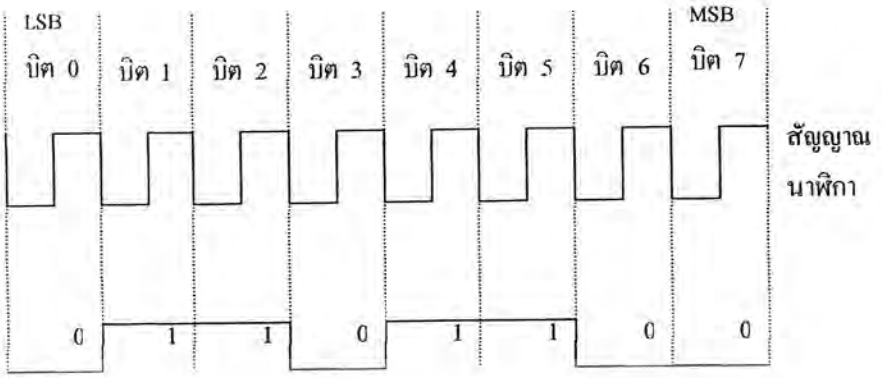
มีทางเลือกอยู่ 2 แนวทางในการที่จะเคลื่อนย้ายข้อมูลจากคอมพิวเตอร์ไปยังอุปกรณ์ต่อพ่วงอื่นๆ หรือคอมพิวเตอร์ด้วยกัน นั่นคือการรับส่งข้อมูลแบบขนานและการรับส่งข้อมูลแบบอนุกรม การรับส่งข้อมูลแบบขนาน จะเป็นการรับส่งข้อมูลคราวละ 4 หรือ 8 บิต ในเวลาเดียวกันซึ่งจะทำให้การรับและส่งข้อมูลทำได้ที่ความเร็วสูง ซึ่งก็หมายความว่าจำนวนของสายที่ใช้ในการส่งจะต้องมีมากเท่ากับจำนวนบิตของข้อมูลที่จะส่งด้วย นอกจากนี้ยังยังจะต้องรวมถึงสายที่ใช้สำหรับการควบคุมและการตรวจสอบการรับส่งข้อมูลด้วย ซึ่งอาจจะต้องสายมากเป็น 2 เท่าของจำนวนบิตข้อมูล ที่จะส่งก็ได้ ซึ่งก็เป็นปัญหาในเรื่องราคาของสายที่ใช้ในการเชื่อมต่อแบบขนานมักจะมีราคาแพง

ในขณะที่การรับส่งข้อมูลแบบอนุกรมเป็นการรับส่งข้อมูลครั้งละ 1 บิต แต่ก็สามารถรับส่งข้อมูลคราวละหลายๆบิตได้ หากแต่จะต้องมีการตกลงกันระหว่างตัวส่งและตัวรับวา จะรับส่งข้อมูลคราวละกี่บิต ตัวรับจะต้องรอข้อมูลมาให้ครบทุกบิตเสียก่อนจึงทำการประมวลผล ส่งผลให้การสื่อสารข้อมูลอนุกรมอาจมีความเร็วต่ำกว่าแบบขนาน ในด้านจำนวนสายสัญญาณการรับส่งข้อมูลแบบอนุกรมจะใช้จำนวนสายที่น้อยกว่ามาก อย่างน้อยที่สุดใช้เพียง 2-3 เส้นเท่านั้น แต่อัตราเร็วในการรับส่งข้อมูลอาจต่ำกว่าแบบขนาน อย่างไรก็ตามการรับส่งข้อมูลแบบอนุกรมสามารถใช้สายสัญญาณที่มีความยาวมากกว่าแบบขนาน ทำให้ระยะทางในการสื่อสารข้อมูลแบบอนุกรมสามารถทำได้มากกว่า

2.8.1 การสื่อสารแบบอนุกรม

การสื่อสารแบบอนุกรมจะแบ่งออกได้เป็น 2 แบบ คือ การสื่อสารอนุกรมแบบซิงโครนัส และการสื่อสารอนุกรมแบบ อะซิงโครนัส การสื่อสารแบบซิงโครนัสจะมีสัญญาณนาฬิการ่วมอยู่กับการรับและส่งสัญญาณด้วย ตัวอย่างการส่งข้อมูลแบบซิงโครนัสก็คือ คีย์บอร์ดของคอมพิวเตอร์ ซึ่งสายเส้นหนึ่งจะเป็นสายสัญญาณของนาฬิกา ส่วนสายอีกเส้นหนึ่งจะเป็นสายของข้อมูล ดังนั้น การติดต่อแบบซิงโครนัสนี้จะต้องใช้สายในการเชื่อมต่ออย่างน้อยที่สุด 3 เส้น คือ สัญญาณนาฬิกา, ข้อมูลและกราวด์ รูปที่ 2.9 แสดงให้เห็นถึงไทมิ่งไดอะแกรมของการส่งข้อมูลแบบซิงโครนัส

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.9 รูปแบบอย่างง่ายที่สุดของข้อมูลอนุกรม

2.8.1.1 การสื่อสารข้อมูลแบบอะซิงโครนัส

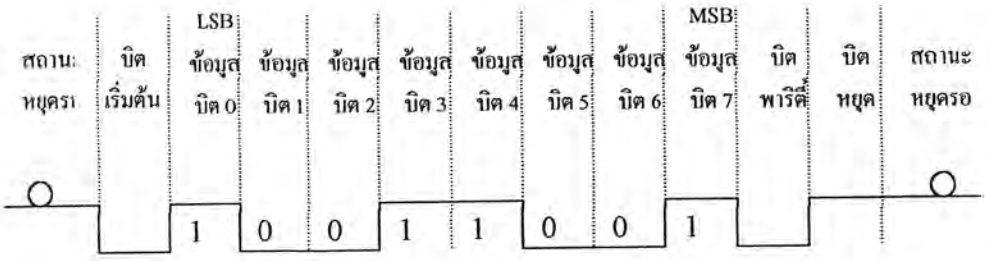
การสื่อสารข้อมูลแบบอะซิงโครนัสคือการรับและส่งข้อมูลไปในสายโดยไม่จำเป็นต้องมีสายสัญญาณนาฬิการ่วมด้วยเหมือนกับการรับส่งข้อมูลแบบซิงโครนัส แต่จะใช้กำหนดค่าสัญญาณนาฬิกาทั้งภาครับและภาคส่งให้มีค่าเท่ากัน ซึ่งเรียกสัญญาณนาฬิกาที่ใช้ในการกำหนดค่าให้ภาครับและภาคส่งนี้ว่า อัตราการถ่ายทอดข้อมูล หรือ บอดเรต (baudrate) มีหน่วยเป็น บิตต่อวินาที (bit per second : bps)

รูปแบบของข้อมูลที่ใช้ในการรับส่งแบบอะซิงโครนัสประกอบด้วย 4 ส่วน คือ

1. บิตเริ่มต้น (Start Bit) ซึ่งจะมีขนาด 1 บิต
2. บิตข้อมูลแบบอนุกรมจะมีขนาด 5,6,7 หรือ 9 บิต
3. บิตตรวจสอบพาริตี้ (Parity Bit) จะมีขนาด 1 บิต หรือ ไม่มี
4. บิตปิดท้าย (Stop Bit) จะมีขนาด 1.1.5 หรือ 2 บิต

รูปที่ 2.10 แสดงรูปแบบของข้อมูลอนุกรมแบบอะซิงโครนัส ซึ่งเมื่อไม่มีข้อมูลที่จะส่งขา Data จะมีสถานะ ลอจิก "1" ซึ่งจะเรียกสถานะนี้ว่า สถานะหยุดรอ (waiting stage) การเริ่มต้นส่งข้อมูลจะเริ่มจากการให้ขา Data มีลอจิก "0" ด้วยช่วงระยะเวลา 1 บิต ซึ่งจะเรียกบิตนี้ว่าบิตเริ่มต้น จากนั้นบิตข้อมูลจะถูกส่งออกไป โดยเริ่มจากบิตที่มีนัยสำคัญต่ำสุด (LSB) ก่อน ซึ่งข้อมูลในไบต์ที่จะส่งอาจมีจำนวนบิต 5,6,7 หรือ 8 บิต ก็ได้ จากนั้นตามด้วยบิตพาริตี้ ซึ่งใช้เพื่อตรวจสอบความผิดพลาดที่เกิดขึ้นจากการส่งข้อมูล บิตสุดท้ายที่จะส่งคือบิตปิดท้าย ซึ่งจะให้ขาดาต้ามีสถานะลอจิก 1 อีกครั้งด้วยระยะเวลาอย่างน้อย 1 บิต 1.5 บิต หรือ 2 บิต เพื่อเป็นการแสดงว่าสิ้นสุดข้อมูลแล้ว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.10 รูปแบบอย่างง่ายที่สุดของข้อมูลอนุกรมแบบอะซิงโครนัส

อุปกรณ์พิเศษที่ได้รับการออกแบบมาสำหรับการรับและการส่งข้อมูลแบบอะซิงโครนัส เรียกว่า Universal Asynchronous Receiver / Transmitter หรือ UART อัตราความเร็วในการรับและส่งข้อมูลของการรับส่งข้อมูลแบบอะซิงโครนัส คือ ค่าบอดเรต ซึ่งก็คือค่าจำนวนบิตต่อวินาทีที่ใช้ในการรับและส่งข้อมูล บอดเรตมาตรฐานที่ใช้สำหรับพอร์ตอนุกรม RS-232 ได้แก่ 110,150,300,600,1200,2400,4800,9600 และ 19200 บิตต่อวินาที และมีค่าเพิ่มมากขึ้นตามเทคโนโลยีของคอมพิวเตอร์ซึ่งการรับส่งแบบอนุกรมโดยไม่ผ่านโมเด็มอาจจะกำหนดค่าบิดเรตได้สูงถึง 115200 บิตต่อวินาที เนื่องจากบิดเรตคือจำนวนบิตข้อมูลที่สามารถถ่ายทอดได้ภายใน 1 วินาที ยกตัวอย่าง ข้อมูลอนุกรมถูกส่งในลักษณะ 8 บิต ไม่มีการตรวจสอบพาริตี้ มีบิตเริ่มต้น 1 บิต และบิตปิดท้าย 1 บิต ความยาวของข้อมูลที่รับส่งนี้เท่ากับ 10บิต ถ้าใช้บิดเรตในการส่งข้อมูลเท่ากับ 9600 บิตต่อวินาที ก็จะสามารถรับส่งข้อมูลได้ด้วยความเร็ว 960 ไบต์ต่อวินาที และถ้ามีการใช้พาริตี้ ความเร็วในการรับส่งข้อมูลจะเหลือเป็น 872 ไบต์ต่อวินาที

การตรวจสอบพาริตี้สามารถกำหนดในเป็นแบบคี่ (odd) , แบบคู่ (even) หรือไม่มีการตรวจสอบพาริตี้ก็ได้ การตรวจสอบพาริตี้เป็นการตรวจสอบจำนวนของบิตที่เป็นลอจิก "1" ภายในข้อมูลที่ส่งไป 1 ไบต์ ว่ามีจำนวนรวมเป็นเลขคู่หรือเลขคี่โดยต้องรวมบิตพาริตี้เข้าไปด้วย ยกตัวอย่างข้อมูลที่จะทำการส่งมีขนาด 8 บิต และมีค่าเท่ากับ 99 ฐานสิบหก หรือ 10011001 ฐานสอง จะเห็นว่าข้อมูลในไบต์นี้มีจำนวนลอจิก "1" จำนวน 4 ตัวซึ่งเป็นเลขคู่ ดังนั้นถ้ากำหนดค่าพาริตี้เป็นคู่ค่าในบิต พาริตี้ จะต้องมีลอจิกเป็น "0" แต่ถ้าพาริตี้เป็นคี่ ค่าบิตพาริตี้จะต้องเป็น "1" เพื่อให้ข้อมูล 1 ไบต์ รวมทั้งพาริตี้มีจำนวนมีจำนวนบิตที่เป็นลอจิก "1" มีจำนวนรวมกันเป็นเลขคี่ ในตารางที่ 4.1 แสดงตัวอย่างของบิตพาริตี้ในการรับส่งข้อมูลอนุกรม

บิตพาริตี้ถูกสร้างขึ้นจากภาคส่งข้อมูลของ UART ซึ่งทางภาครับจะต้องการกำหนดคุณสมบัติการตรวจสอบพาริตี้ให้ตรงกันว่าจะตรวจสอบพาริตี้คี่หรือพาริตี้คู่ จากนั้นภาครับของ UART จะทำการตรวจสอบค่าพาริตี้ที่เกิดขึ้นว่าเป็นคู่หรือเป็นคี่ โดยการนับจำนวนลอจิก "1" ทั้งหมดรวม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ทั้งบิตพาริตี้ด้วย ถ้ากำหนดพาริตี้ไว้เป็นคู่แต่อ่านค่าตัวเลขในการนับออกมาได้ตัวเลขเป็นคี่ทั้งภาครับจะแสดงข้อผิดพลาดออกมาให้ผู้ใช้ทราบ นับเป็นการตรวจสอบความผิดพลาดที่เกิดขึ้นในการถ่ายทอดข้อมูลที่ง่ายที่สุด แต่จะเชื่อถือได้เมื่อมีบิตข้อมูลที่ทำการส่งผิดพลาดเพียงบิตเดียวเท่านั้น ถ้าข้อมูลที่ทำการส่งมีบิตที่ผิดพลาดมากกว่า 1 บิต การตรวจสอบด้วยวิธีนี้จะไม่ได้ผล สำหรับการตั้งพาริตี้บิตเป็น NONE นั้นทั้งภาครับและภาคส่งจะไม่มีการตรวจสอบพาริตี้

ตาราง 2.2 บิตพาริตี้ของข้อมูล

ข้อมูล	บิตพาริตี้คู่	บิตพาริตี้คี่
00000000	0	1
00000001	1	0
00000010	1	0
00000011	0	1
00000100	1	0
11111110	0	1
11111111	1	0

คอมพิวเตอร์ในรุ่น AT เกือบทั้งหมดจะใช้ UART เบอร์ 16450 และ 16550 ส่วนคอมพิวเตอร์ในรุ่น XT ใช้ UART เบอร์ 8250 UART ชิปเหล่านี้มีระดับแรงดันแบบทีทีแอล (1 และ-5 V) แต่เพื่อให้มีแรงดันเป็นไปตามมาตรฐาน RS-232 และเพื่อให้การรับส่งข้อมูลสามารถทำได้ที่ระยะทางไกลมากขึ้น ระดับแรงดันที่ทีแอลจะถูกแปลงเป็นระดับแรงดันที่สูงขึ้น โดยลอจิก "0"มีระดับแรงดัน+3V ถึง +12V ในขณะที่ลอจิก "1"มีระดับแรงดัน -3V จนถึง -12V

2.8.2 มาตรฐานพอร์ตอนุกรมแบบ RS-232

มาจรฐานการเชื่อมต่อแบบอนุกรม RS-232 เป็นมาตรฐานอุตสาหกรรมที่ออกแบบมาเพื่อใช้ในการส่งข้อมูลอนุกรมแบบอะซิงโครนัส 2 ทิศทาง โดยมาตรฐาน RS-232 ในอดีตนั้นถูกออกแบบมาเพื่อการส่งผ่านข้อมูลจากคอมพิวเตอร์ไปยังโมเด็มเพียงอย่างเดียว เพื่อที่จะนำข้อมูลจากโมเด็มนี้สื่อผ่านสายโทรศัพท์ไปยังคอมพิวเตอร์อีกชุดหนึ่งซึ่งอยู่ห่างไกลกัน โดยคณะกรรมการที่เรียกว่า สมาคมอุตสาหกรรมอิเล็กทรอนิกส์ (Electronic Indrustries Association:EIA) ได้วางมาตรฐานที่มีชื่อเรียกกันว่า EIA RS-232 มาตรฐานนี้ในช่วงแรกจะใช้คอนเน็กเตอร์เป็นแบบ DB-25 โดย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กำหนดความยาวสูงสุดของสายสัญญาณไว้ที่ 50 ฟุต มีระดับสัญญาณตั้งแต่ -3 V ถึง -12 V แสดงว่ามีข้อมูล (Mark) และ +3 V ถึง +12 V แสดงว่าเป็นช่องว่าง(Space)

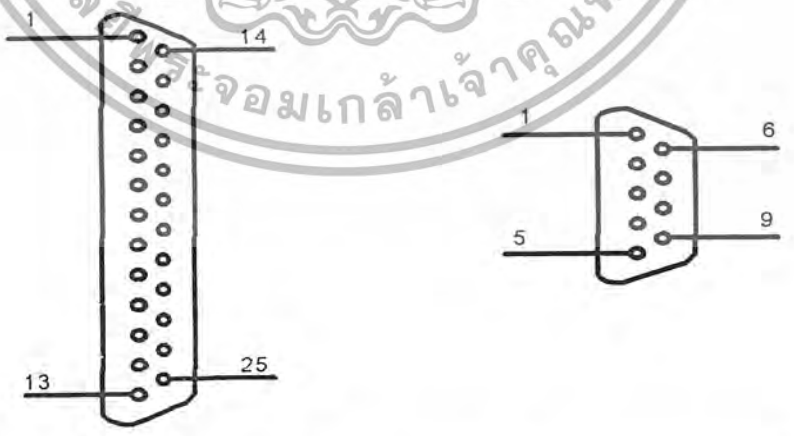
มาตรฐาน RS-232 ได้กำหนดรูปแบบของอุปกรณ์เชื่อมต่อข้อมูล (Data Terminal Equipment : DTE) กับวงจรข้อมูลปลายทาง (Data Cricuit Termination : DCE) ไว้ว่า อุปกรณ์ DTE จะต้องเป็นอุปกรณ์ที่มีการประมวลผลในตัวเช่น ไมโครคอนโทรเลอร์หรือไมโครคอมพิวเตอร์ ซึ่งมีความสามารถในการสร้างบิตข้อมูลแบบอนุกมได้ ส่วนอุปกรณ์ DCE จะทำหน้าที่เป็นเพียงตัวรับข้อมูลที่ส่งมาจาก DTE เท่านั้น โดยการรับส่งข้อมูลระหว่างอุปกรณ์ทั้งสองจะกระทำผ่านมาตรฐาน RS-232

ข้อแตกต่างของอุปกรณ์ DTE และอุปกรณ์ DCE อย่างหนึ่งที่เห็นได้ชัดคือ คอนเน็กเตอร์ของ DTE จะเป็นตัวผู้ ส่วนคอนเน็กเตอร์ของ DCE จะเป็นตัวเมีย ซึ่งพอร์ตอนุกรมของคอมพิวเตอร์ที่ใช้กันอยู่ทั่วไปจะเป็นแบบ DTE ส่วนคอนเน็กเตอร์ที่อยู่ที่โมเด็มจะเป็นแบบ DCE

สำหรับการใช้งานบนคอมพิวเตอร์ พอร์ตอนุกรม RS-232 มักจะถูกใช้เชื่อมต่อกับโมเด็มหรือเม้าท์ โดยสามารถรับส่งข้อมูลได้ที่ความยาวของสายสัญญาณสูงสุดถึง 20 เมตร

2.8.3 คอนเน็กเตอร์สำหรับพอร์ต RS-232 และการเชื่อมต่อ

มาตรฐานการเชื่อมต่อแบบ RS-232 จะใช้คอนเน็กเตอร์แบบ DB-25 ตัวผู้ หรือ DB-9 ตัวผู้ ซึ่งคอนเน็กเตอร์แบบ DB-25 จะมีขาใช้งานเพียง 9 เส้นเช่นเดียวกับคอนเน็กเตอร์แบบ DB-9 เนื่องจาก ขาอื่นๆที่เหลือใช้งานในอดีต ปัจจุบันมีการใช้งานไม่มากนัก จึงถูกยกเลิกไป โดยแสดงรูปร่างและตำแหน่งขาในรูปที่ 2.11



(ก) คอนเน็กเตอร์อนุกรม 25 ขา หรือแบบ DB-25 (มองจากด้านหลังคอมพิวเตอร์)

(ข) คอนเน็กเตอร์อนุกรม 9 ขาหรือแบบ DB-9 (มองจากด้านหลังคอมพิวเตอร์)

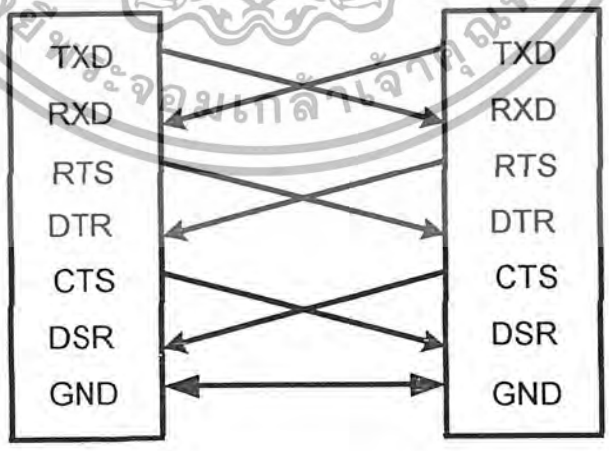
รูปที่2.11 การจัดขาของคอนเน็กเตอร์พอร์ตอนุกรมตามมาตรฐาน RS-232 ทั้งแบบDB-9 และDB-25

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตาราง 2.3 การจัดขา DB-9 และ DB-25

คอนเน็กเตอร์ DB9	คอนเน็กเตอร์DB25	ชื่อของสายสัญญาณ	ชนิดของสายสัญญาณ
1	8	Data Carrier Detect:DCD	อินพุต
2	3	Receiced Data : RxD	อินพุต
3	2	Transmitted Data : TxD	เอาท์พุต
4	20	Data Terminal Teady : DTR	เอาท์พุต
5	7	Signal Ground : GND	-
6	6	Data Set Ready : DSR	อินพุต
7	4	Request To Send : RTS	เอาท์พุต
8	5	Clear To Send : CTS	อินพุต
9	22	Ring Indicator : RI	อินพุต

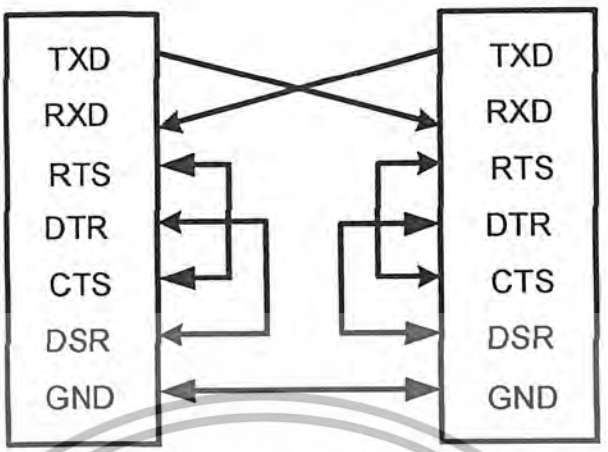
สำหรับการเชื่อมต่อคอมพิวเตอร์กับอุปกรณ์ภายนอกแสดงดังรูป 4.9 ลูกศรในรูปแสดงถึงทิศทางของข้อมูล ในรูปที่ 2.12 เป็นการเชื่อมต่อแบบ Null modem หรือการเชื่อมต่อโดยตรงโดยไม่ต้องผ่านโมเด็ม โดยมีการตรวจสอบหรือแฮนด์เชคเตมรูปแบบ ส่วนในรูปที่ 2.13 เป็นการเชื่อมต่อแบบ Null modem ในลักษณะที่ใช้สายสัญญาณเพียง 3 เส้น โดยเส้นหนึ่งสำหรับส่งข้อมูลอีกเส้นสำหรับรับข้อมูล และเส้นสุดท้ายเป็นกราวด์ สำหรับรายละเอียดหน้าที่การทำงานในแต่ละขาของพอร์ตอนุกรม RS-232 มีดังนี้



รูปที่2.12 การต่ออุปกรณ์ภายนอกเข้ากับคอมพิวเตอร์แบบ Noll modem

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง



คอมพิวเตอร์ อุปกรณ์ภายนอก

รูปที่2.13 การต่ออุปกรณ์ภายนอกเข้ากับคอมพิวเตอร์แบบ RS-232 โดยใช้สัญญาณ 3 เส้น

- Data Carrier Detect : DCD หรืออาจเรียกว่า Carrier Detect : CD ขานี้จะแอกทีฟเมื่อมีการส่งสัญญาณพาห์จากอุปกรณ์สื่อสารข้อมูล เช่น โมเด็ม สำหรับการใช้งานปกติ ขานี้จะไม่ได้ถูกใช้งานมากนัก

- Receive Data : RD หรือ RxD ขานี้ใช้เพื่อรับส่งสัญญาณอนุกรมเข้ามายังคอมพิวเตอร์โดยนำข้อมูลที่อ่านได้เก็บไว้ในรีจิสเตอร์ บัฟเฟอร์

- Transmitted Data : TD หรือ TxD ขานี้ใช้เพื่อส่งข้อมูลออกจากคอมพิวเตอร์โดยนำข้อมูลที่เก็บอยู่ในบัฟเฟอร์สำหรับส่งข้อมูลส่งออกไป

- Data Terminal Ready : DTR เป็นขาสัญญาณที่ส่งออกจากคอมพิวเตอร์เพื่อให้อุปกรณ์ปลายทางรับรู้ว่า ต้องการติดต่อด้วย โดยขา DTR นี้จะต้องเชื่อมต่อกับขา DSR ของอุปกรณ์ปลายทาง และขา DTR ของอุปกรณ์ปลายทางจะต้องเชื่อมต่อเพียง 3 เส้น จะต้องต่อขา DTR และ DSR ของตัวมันเองเข้าด้วยกันและต้องต่อขา DCD ด้วยกรณีที่โปรแกรมสื่อสารที่ใช้มีการตรวจจับสัญญาณพาห์

- Signal Ground : GND ขากราวด์ของระบบ

- Data Set Ready : DSR ขานี้จะใช้คู่กับขา DTR เพื่อตรวจสอบการเชื่อมต่อกันระหว่างคอมพิวเตอร์กับอุปกรณ์ปลายทาง ซึ่งขา DSR นี้จะเป็นขาสำหรับรับข้อมูลจากภายนอกซึ่งถูกส่งมาจากขา DTR

- Request To Send : RTS เป็นขาสำหรับส่งสัญญาณร้องขอให้ทางอุปกรณ์ปลายทางส่งข้อมูลกลับมายังคอมพิวเตอร์ โดยขาที่รับสัญญาณ RTS ก็คือขา CTS ในกรณีที่ใช้การเชื่อมต่อแบบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

55478

Null modem 3 สาย จะต้องเชื่อมต่อขา RTS และ CTS ของตัวมันเองเข้าด้วยกัน เพื่อจะให้การรับและส่งข้อมูลสามารถเกิดขึ้นได้ตลอดเวลา

- Clear To Send : CTS ขานี้จะคอยรับสัญญาณรียกจากสายโทรศัพท์ ปกติในการสื่อสารโดยทั่วไปสายนี้จะไม่ถูกใช้งาน จะใช้งานก็ต่อเมื่อมีการเชื่อมต่อกับโมเด็มและโปรแกรมมีการตรวจสอบสัญญาณนี้เท่านั้น

2.8.4 UART

UART มาจากคำว่า Universal Asynchronous Receiver Transmitter ซึ่งหมายถึงอุปกรณ์ที่ทำหน้าที่รับและส่งข้อมูลแบบอะซิงโครนัสนั่นเอง สำหรับการสื่อสารอนุกรมบนคอมพิวเตอร์แล้ว UART ถือว่าเป็นหัวใจสำคัญของการสื่อสารอนุกรม

หน้าที่หลักของ UART คือ ทำหน้าที่แปลงข้อมูลที่อยู่ในรูปแบบขนานจากคอมพิวเตอร์ให้อยู่ในรูปแบบอนุกรมแบบอะซิงโครนัส แล้วส่งออกไป และทำหน้าที่แปลงสัญญาณอนุกรมแบบอะซิงโครนัสที่ป้อนเข้ามายัง UART ให้เป็นแบบขนานก่อนที่จะส่งข้อมูลเข้าสู่คอมพิวเตอร์ ซึ่งนอกจาก UART จะส่งข้อมูลไปยังคอมพิวเตอร์แล้ว ยังแจ้งข้อมูลอื่นๆ ให้คอมพิวเตอร์รับทราบด้วย เช่น อัตราเร็วในการรับส่งข้อมูล (บอดเรต) , รูปแบบการส่งข้อมูล , ความผิดพลาดที่เกิดขึ้นระหว่างการถ่ายทอดข้อมูล (ผิดพลาดจากพาริตี้ , เฟรมข้อมูล , โอเวอร์รัน)

ภายใน UART จะมีส่วนของวงจรสร้างบอดเรตแบบโปรแกรมได้ (progrannable buadrate generator) โดยการกำหนดค่าตัวหารให้กับสัญญาณนาฬิกาของ UART โดยตัวหารนี้มีขนาด 16 บิต ดังนั้นจึงสามารถกำหนดตัวหารอยู่ในช่วง 1-65,535 UART สามารถรับส่งข้อมูลได้ทั้งแบบ ฮาล์ฟดูเพล็กซ์ (haft duplex) และฟูลดูเพล็กซ์ (full duplex) โดยการส่งแบบฮาล์ฟดูเพล็กซ์เป็นการส่งแบบทิศทางเดียว ส่วนการส่งแบบฟูลดูเพล็กซ์นั้นสามารถรับและส่งข้อมูลได้ในคราวเดียวกัน

2.8.4.1 ชนิดของ UART

ในเครื่องคอมพิวเตอร์ทั่วไปมี UART ที่ใช้งานกันอยู่ 2 เบอร์ คือ 8250 ซึ่งเป็น UART มาตรฐานที่มีใช้กันยาวนาน UART เบอร์นี้จะมีบัฟเฟอร์สำหรับรับและส่งข้อมูลตำแหน่งเดียวกัน ทำให้การรับและการส่งข้อมูลถูกจำกัดความเร็วอยู่ที่ 57.6 กิโลบิตต่อวินาทีเท่านั้น แต่ UART เบอร์นี้ถือว่าเป็นต้นแบบของ UART ที่ใช้ในคอมพิวเตอร์ โดยคอมพิวเตอร์ทุกๆรุ่นจะต้องสนับสนุนการทำงานตามรูปแบบของ UART เบ□ร์นี้

UART อีกเบอร์หนึ่ง คือ 16450 มีความสามารถรับส่งข้อมูลได้ที่ความเร็ว 115,200 บิตต่อวินาที และเพิ่มรีจิสเตอร์สำหรับพักข้อมูลสำหรับ UART นอกจากนั้นยังเพิ่มส่วนของชิฟต์รีจิสเตอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

แบบ FIFO (First In First Out)ขนาด 16 ไบต์เข้าไป ทำให้สามารถสนับสนุนความเร็วในการรับส่งข้อมูลที่ 256 กิโลบิตต่อวินาทีได้ โดยคอมพิวเตอร์ในปัจจุบันใช้ UART เบอร์นี้หรือใหม่กว่า เช่น เบอร์ TL15C750 ซึ่งมีรีจิสเตอร์แบบ FIFO ขนาด 64 ไบต์ ทำงานได้ที่ระดับแรงดัน +5 V และ+3 V มีโหมดการประหยัดพลังงาน สามารถรับส่งข้อมูลได้ที่ความเร็ว 1 เมกะติดต่อวินาทีเมื่อใช้สัญญาณนาฬิกา 16 MHz

อย่างไรก็ตาม ความเร็วในการส่งข้อมูลที่มากมายของ UART เบอร์ใหม่ๆก็ไม่ได้ช่วยให้การรับส่งข้อมูลของคอมพิวเตอร์เร็วขึ้น เนื่องจากคอมพิวเตอร์ยังใช้ความถี่ของสัญญาณนาฬิกาแบบในการแปลงข้อมูลเพียง 1.8432 MHz เท่านั้น

2.8.5 วงจรภายในและรีจิสเตอร์ของพอร์ตอนุกรม RS-232

เครื่องคอมพิวเตอร์โดยทั่วไปสามารถต่อพอร์ตอนุกรม RS-232 สูงสุดได้ถึง 4 พอร์ต ซึ่งจะมีชื่อเรียกเป็น COM 1 , COM 2 , COM 3 , และ COM 4 ซึ่งพอร์ตอนุกรมแต่ละตัวต่างก็ใช้งาน UART ภายในคอมพิวเตอร์ในการติดต่ออุปกรณ์ภายนอกเช่นเดียวกัน

การทำงานภายในพอร์ตอนุกรม ซึ่งประกอบไปด้วยรีจิสเตอร์ขนาด 8 บิต 8 ตัวที่ใช้งานร่วมกับ UART แอดเดรสของรีจิสเตอร์ภายในพอร์ตอนุกรมสามารถคำนวณได้จากค่ารีจิสเตอร์พื้นฐานของพอร์ตอนุกรม ยกตัวอย่าง พอร์ตอนุกรม COM 1 มีแอดเดรสอยู่ที่ 3F8H ตำแหน่งของรีจิสเตอร์ต่างๆจะเป็นตำแหน่งที่บวกเข้าไปกับค่า 3F8H โดยใช้รีจิสเตอร์ที่ใช้งานกับพอร์ตอนุกรมดังนี้

00H เป็นรีจิสเตอร์บัฟเฟอร์สำหรับเก็บข้อมูลที่รับมาหรือเตรียมข้อมูลก่อนที่จะส่งออกไป

01H รีจิสเตอร์อินาเบิลการอินเตอร์รัปต์ ใช้ในการเซตโหมดการอินเตอร์รัปต์ของพอร์ตอนุกรม

02H รีจิสเตอร์แสดงโหมดการอินเตอร์รัปต์ ใช้เพื่อตรวจสอบโหมดการอินเตอร์รัปต์เมื่อมีการอินเตอร์รัปต์เกิดขึ้น

03H รีจิสเตอร์กำหนดรูปแบบของข้อมูล

04H รีจิสเตอร์ควบคุมโมเด็ม ใช้ตรวจสอบบิตสำหรับติดต้อกับโมเด็ม เช่น RTS หรือ DTR

05H รีจิสเตอร์แสดงสถานการณ์รับและการส่งข้อมูลแบบอนุกรม

06H รีจิสเตอร์แสดงสถานะของโมเด็ม ซึ่งจะแสดงสถานะของขา DCD ,RI , DSR และ CTS

07H รีจิสเตอร์สำหรับการเก็บข้อมูลชั่วคราว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รีจิสเตอร์ตำแหน่ง 00H : รีจิสเตอร์บัฟเฟอร์

เป็นรีจิสเตอร์สำหรับเก็บข้อมูลที่รับเข้ามาและข้อมูลที่จะส่งออกไป โดยการติดต่อกับรีจิสเตอร์นี้เพื่อเก็บข้อมูลที่ต้องการจะส่งจะต้องกำหนดให้บิต DLAB ในรีจิสเตอร์กำหนดรูปแบบข้อมูล (03H)จะต้องมีสถานะเป็น 0 ซึ่งการเขียนข้อมูลมายังแอดแดดรสนี้ เป็นการส่งข้อมูลไปยังรีจิสเตอร์ส่งข้อมูลและข้อมูลจะถูกส่งออกไปแบบอนุกรม สำหรับการรับข้อมูล หลังจากมีการอ่านรีจิสเตอร์นี้ออกไปรีจิสเตอร์นี้จะถูกเคลียร์ และเตรียมพร้อมสำหรับการรับข้อมูลในไบต์ต่อไป

2.8.6 ลักษณะสัญญาณอินพุตและเอาท์พุตของพอร์ต RS-232

สัญญาณเอาท์พุตที่ใช้ควบคุม (RTS และ DTR) และสัญญาณแสดงสถานะอินพุต (CTS,DSR และ DCD) ของพอร์ตอนุกม RS-232 จะถูกกลับสถานะภายในตัว UART จะให้ระดับสัญญาณเอาท์พุตออกมาเป็นแบบที่ทีแอลเท่านั้น ดังนั้นเมื่อสัญญาณถูกส่งออกมาจาก UART จึงต้องส่งเข้าสู่วงจรขับเพื่อปรับระดับแรงดันให้ได้ระดับสัญญาณเป็นไปตามมาตรฐาน RS-232 ก่อนส่งออกไปจากคอมพิวเตอร์สำหรับอุปกรณ์ต่อเชื่อมปลายทางก็จะต้องมีวงจรขับในลักษณะนี้เช่นเดียวกัน เพื่อให้ระดับสัญญาณในระดับเดียวกัน แต่วงจรขับที่ใช้ทั้งภายในคอมพิวเตอร์และอุปกรณ์ต่อเชื่อมปลายทางนั้นจะถูกกลับสถานะ

2.8.7 แอสเดรสของพอร์ตอนุกรม

แอสเดรสพื้นฐานของพอร์ตอนุกม มี 4 ตำแหน่งดังนี้ คือ

COM1 : 3E8H

COM2 : 2F8H

COM3 : 3E8H

COM4 : 2E8H

เมื่อเริ่มเปิดเครื่องเพื่อใช้งานคอมพิวเตอร์ ไบออสภายในคอมพิวเตอร์จะทำการตรวจสอบแอดเดรสของพอร์ตอนุกรมทั้งหมด ถ้าไบออสตรวจพบแอดเดรสของพอร์ตอนุกรม ไบออสจะนำแอดเดรสที่ตรวจพบไปเก็บไว้ในหน่วยความจำขนาด 2 ไบต์ สำหรับพอร์ตอนุกรม COM1จะเก็บไว้ที่แอดเดรส 0000 : 0400 H และ 0000 : 0401H ส่วนตำแหน่งอื่นๆมีรายละเอียดดังนี้

COM2 = 0000:0402 H-0000:0403 H

COM3 = 0000:0404 H-0000:0405 H

COM4 = 0000:0406 H-0000:0407 H

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

นอกจากนี้ที่หน่วยความจำแอดเดรส 0000:0411 ยังใช้สำหรับแสดงจำนวนของพอร์ตอนุกรมที่มีใช้อยู่ในคอมพิวเตอร์อีกด้วยโดยมีรายละเอียดดังแสดงในตารางที่ 2.4

ตารางที่ 2.4 ข้อมูลในแอดเดรส 0000:0411 H ที่ใช้แจ้งจำนวนพอร์ตอนุกรม

บิท3	บิท2	บิท1	จำนวนพอร์ต
0	0	0	ไม่มีพอร์ตอนุกรม
0	0	1	มีพอร์ตอนุกรม 1 พอร์ต
0	1	0	มีพอร์ตอนุกรม 2 พอร์ต
0	1	1	มีพอร์ตอนุกรม 3 พอร์ต
1	0	0	มีพอร์ตอนุกรม 4 พอร์ต

2.9 การเขียนโปรแกรมเพื่อใช้งานพอร์ตอนุกรมด้วยคอนโทรล MSComm

สำหรับการใช้งาน Visual BASIC ตั้งแต่เวอร์ชั่น 2 เป็นต้นมา ใน Visual BASIC จะมีคัสตอมคอนโทรลสำหรับการสื่อสารอนุกรมผ่านทางพอร์ตอนุกรมของคอมพิวเตอร์มาให้ โดยใน Visual BASIC เวอร์ชั่น 2 และ เวอร์ชั่น 3 จะใช้ชื่อว่า MSCOMM.VBX ส่วน เวอร์ชั่น 4 ใช้ชื่อว่า *MSCOMM16.OCX สำหรับการทำงานกับระบบปฏิบัติการ 16 บิต และ MACOMM32.OCX* สำหรับการทำงานกับระบบปฏิบัติการ 32 บิต สำหรับใน Visual BASIC เวอร์ชั่น 5 จะมีเพียง MSCOMM32.OCX เท่านั้นเพราะถูกออกแบบมาใช้งานกับระบบปฏิบัติการ 32 บิต

MSComm จัดเตรียมทางเลือกเอาไว้ 2 ทางเพื่อความสะดวกในการสื่อสารข้อมูล ทางแรกคือ การสื่อสารข้อมูลที่กระตุ้นด้วยเหตุการณ์ (event-driven communications) เป็นรูปแบบการใช้งานที่มีประสิทธิภาพมากสำหรับการตอบสนองแบบทันทีทันใด เช่น เมื่อตัวอักษรถูกส่งมาที่พอร์ตอนุกรมหรือเกิดการเปลี่ยนแปลงที่ขา Data Carrier (DCD) หรือขา Request To Send (RTS) เหตุการณ์ Oncomm ของ MSComm จะสามารถจับตรวจสัญญาณนั้นได้ทันที ซึ่งจะกล่าวถึงรายละเอียดในหัวข้อคุณสมบัติ CommEven ต่อไป ส่วนทางเลือกที่สองเป็นการคอยตรวจสอบค่าเหตุการณ์และความผิดพลาดที่เกิดขึ้นด้วยการดูค่าที่เปลี่ยนแปลงภายในคุณสมบัติ ConnEven หลังจากให้โปรแกรมทำงานในฟังก์ชั่นต่างๆไปเรียบร้อยแล้ว ซึ่งวิธีนี้ใช้งานได้ดีในกรณีที่โปรแกรมมีขนาดเล็ก

คอนโทรล MSComm 1 ตัวสามารถควบคุมการทำงานของพอร์ตอนุกรมได้ 1 พอร์ต ถ้าในโปรแกรมที่ใช้งานต้องการติดต่อกับพอร์ตอนุกรมมากกว่า 1 พอร์ต จะต้องใช้คอนโทรล MSComm

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

มากกว่า 1 ตัวเพื่อควบคุมพอร์ตอนุกรมในแต่ละพอร์ต แอดเดรสของพอร์ตอนุกรมและแอดเดรสขิง การเกิดอินเตอร์รัปต์สามารถเปลี่ยนแปลงได้จากการแก้ไขค่าที่ Control Panel

2.9.1 CommPort

ใช้ในการกำหนดและอ่านค่าพอร์ตนุกรมที่ติดต่ออยู่ (COM1,COM2,COM3,COM4)

รูปแบบการใช้งาน

Object.CommPort [= value]

โดย Value เป็นค่าของพอร์ตอนุกรม ชนิดของข้อมูลเป็น Integer ค่า Value สามารถกำหนดได้ในช่วง 1-16 (ค่าเริ่มต้นกำหนดไว้ที่ 1) เมื่อมีการกำหนดค่าแล้วทำการเปิดพอร์ตโดยใช้คุณสมบัติ PortOpen แต่ว่าพอร์ตนั้นไม่มีอยู่ในระบบ MSComm จะสร้างสัญญาณแสดงข้อผิดพลาด error 68 ขึ้นมา ซึ่งหมายถึง อุปกรณ์ตัวนี้ไม่มีอยู่ในระบบ ดังนั้นการเขียนโปรแกรมจึงจำเป็นต้องกำหนดตำแหน่งของพอร์ตอนุกรมก่อนที่ใช้คำสั่ง OpenPort

2.9.2 Setting

ใช้ในการกำหนดและอ่านค่าอัตราบอด , พาริตี้ ,จำนวนบิตข้อมูล ,จำนวนบิตปิดท้าย

รูปแบบการใช้งาน

Object.Setting [= value]

ค่า Value มีชนิดข้อมูลเป็นแบบ String มีรูปแบบเป็น "BBBB,P,D,S" โดย BBBB เป็นค่าอัตราบด , P เป็นค่าพาริตี้ , D เป็นจำนวนของบิตข้อมูล และ S เป็นจำนวนของบิตปิดท้าย ปกติแล้วค่านี้ถูกกำหนดไว้เป็น "9600 ,N,8,1"

ค่าบิดเรตมาตรฐานที่ใช้กับ MSComm มีดังนี้

110 บิตต่อวินาที
300 บิตต่อวินาที
600 บิตต่อวินาที
1,200 บิตต่อวินาที
2,400 บิตต่อวินาที
9,600 บิตต่อวินาที (ค่าปกติ)
14,400 บิตต่อวินาที
19,200 บิตต่อวินาที
28,800 บิตต่อวินาที

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

28,400 บิตต่อวินาที

38,400 บิตต่อวินาที (สงวน)

56,000 บิตต่อวินาที (สงวน)

128,000 บิตต่อวินาที (สงวน)

256,000 บิตต่อวินาที (สงวน)

สำหรับค่ามาตรฐานในการกำหนด ค่าพาริตี้มีดังนี้

สัญลักษณ์	รายละเอียด
E	พาริตี้คู่ (Even)
M	ลอจิก "1" (Mark)
N	ไม่ใช้ (ค่าปกติ)
O	พาริตี้คี่ (Odd)
S	ลอจิก "0" (Space)

ค่าที่ใช้ในการกำหนดจำนวนบิตมี 5 ค่า คือ 4,5,6,7 และ 8 (เป็นค่าปกติ)

ค่าที่ระบุจำนวนบิตปิดท้าย มี 3 ค่า คือ 1 (เป็นค่าปกติ),1.5 และ 2

ตัวอย่างการใช้งานคำสั่ง Setting โดยจะเป็นการกำหนดค่าบิดเรตเท่ากับ 9600 ไม่มีพาริตี้ จำนวนบิตข้อมูล 8 บิต และบิตปิดท้าย 1 บิต สามารถเขียนโปรแกรมได้ดังนี้

MSCONN1.Setting = "9600,N,8,1"

หมายเหตุ เหตุที่ค่าที่กำหนดจะต้องอยู่ภายในเครื่องหมายคำพูด "...." เนื่องจากค่าที่กำหนดนี้อยู่ในรูปตัวแปร String

2.9.3 PortOpen

ใช้ในการกำหนดและอ่านค่าสถานะของพอร์ตอนุกรม เพื่อเปิดและปิดพอร์ตอนุกรม

รูปแบบการใช้งาน

Object.PortOpen [= value]

ค่า Value มีชนิดข้อมูลเป็นแบบบูลีน คือ True กับ False โดย True หมายถึงการเปิดพอร์ตอนุกรมและ Flase หมายถึงการปิดพอร์ตอนุกรม สำหรับการปิดพอร์ตนั้นจะมีการเคลียร์บัฟเฟอร์รับข้อมูลด้วย คอนโทรล MSComm จะปิดพอร์ตอนุกรมโดยอัตโนมัติเมื่อออกจากโปรแกรม ก่อนที่จะใช้คุณสมบัติ CommPort นั้นได้ทำการกำหนดตำแหน่งของพอร์ตอนุกรมไว้ถูกต้องหรือไม่ มิเช่น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

นั้น MSComm จะแสดงข้อผิดพลาด Error 68 แจ้งแก่ผู้ใช้งาน หรือถ้าพอร์ตอนุกรมนั้นถูกเปิดเอาไว้แล้ว โปรแกรมนี้ก็จะแจ้งข้อผิดพลาดออกมาเช่นเดียวกัน

ถ้าคุณสมบัติ DTREnable หรือ RTSEnable ถูกกำหนดให้เป็น True ก่อนที่จะทำการเปิดพอร์ต ค่าคุณสมบัติของ DTREnable จะถูกเซตเป็นFalseหลังจากปิดพอร์ตแต่ถ้าเซตเป็น Flase หลังจากปิดโปรแกรมแล้ว ค่าที่กำหนดไว้จะเป็นค่าเดิม

ตัวอย่างการใช้คำสั่งเปิดพอร์ต เพื่อติดต่อสื่อสารกับพอร์ตอนุกรม COM1 และมีบอดเรต 9,600 บิตต่อวินาที ไม่มีพาริตี้ จำนวนบิตข้อมูล 8 บิต และบิตปิดท้าย 1 บิต มีดังนี้

```
MSComm1.Setting = "9600,n,8,1"
MSComm1.CommPort = 1
MSComm1.PortOpen = True
```

2.9.4 Input

อ่านค่าและลบค่าขบวนข้อมูลจากบัฟเฟอร์การรับ

รูปแบบการใช้งาน

Object.Input

คุณสมบัติ InputLen เป็นตัวกำหนดจำนวนของตัวอักษรที่จะอ่านโดยคุณสมบัติ Input การกำหนดค่าให้ InputLen เท่ากับ 0 เป็นการกำหนดคุณสมบัติ Input ทำให้การอ่านค่าข้อมูลในบัฟเฟอร์รับข้อมูลทั้งหมด

คุณสมบัติ InputMode เป็นตัวกำหนดชนิดของข้อมูลที่มีคุณสมบัติ Input รับเข้ามา ถ้า InputMode ถูกกำหนดเป็น comInputModeText คุณสมบัติ Input จะส่งค่าข้อมูลกลับมาในรูปแบบของข้อความชนิดข้อมูลเป็นแบบ Variant ถ้า InputMode กำหนดเป็น comInputModeBinary คุณสมบัติ Input จะส่งข้อมูลกลับมาในรูปของไบนารี่และชนิดข้อมูลเป็นแบบ Variant

ตัวอย่างโปรแกรมแสดงให้เห็นถึงวิธีในการรับข้อมูลจากบัฟเฟอร์ข้อมูลทั้งหมด

```
Private Sub Command_Click()
Dim InString as String
    MSComm1.InputLen = 0            Retrieve all available data
    If MSComm1. InBufferCont Then   Check for data
    Instring = MSComm1.Input        Read Data
    End If
End Sub
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.9.5 InbufferCount

ส่งค่าจำนวนของตัวอักษรที่อยู่ในบัฟเฟอร์ภาครับ

รูปแบบการใช้งานคำสั่ง

Object.InBufferCount [= value]

คำสั่ง InBufferCount จะแสดงค่าจำนวนของตัวอักษร ซึ่งรับมาจากภายนอกและยังเก็บอยู่ในบัฟเฟอร์ภาครับ เพื่อให้ผู้ใช้งานอ่านค่าออกไป สำหรับการเคลียร์ค่าบัฟเฟอร์ภาครับทำได้โดยกำหนดให้ InBufferCount มีค่าเป็น 0

หมายเหตุ อย่างสับสนระหว่าง คำสั่ง InBufferSize และ InBufferCount คำสั่ง InBufferSize นั้นใช้เพื่อกำหนดขนาดของบัฟเฟอร์ภาครับ

2.9.6 InBufferSize

กำหนดและคืนค่าขนาดของบัฟเฟอร์ภาครับในหน่วยเป็น ไบต์

รูปแบบการใช้งานคำสั่ง

Object.InBufferSize [= value]

คำสั่ง InBufferSize ใช้เพื่อกำหนดขนาดของบัฟเฟอร์ภาครับ โดยค่าเริ่มต้นถูกกำหนดไว้ที่ 1,024 ไบต์

หมายเหตุ การกำหนดค่าบัฟเฟอร์ภาครับขนาดใหญ่จะทำให้ หน่วยความจำที่เหลือสำหรับการใช้งานส่วนอื่นๆจะเหลือน้อย อย่างไรก็ตามการกำหนดค่าบัฟเฟอร์ภาครับที่น้อยเกินไปจะทำให้เกิดการโอเวอร์โฟลวหรือข้อมูลล้นบัฟเฟอร์ เว้นแต่จะมีการใช้แฮนด์เช็ก ดังนั้นค่าปานกลางที่เหมาะสมก็คือ ค่า 1,024 ซึ่งเป็นค่าเริ่มต้นนั่นเอง แต่ถ้าโปรแกรมมีการเกิดโอเวอร์โฟลวแล้วจึงค่อยปรับเพิ่มค่าขนาดของบัฟเฟอร์ให้มีค่าเพิ่มมากขึ้น

```
Private Sub Command1_Click()
Dim Buffer as Variant
Dim Arr () as Byte
MSComm1.CommPort = 1                          'Set and open port
MSComm1.PortOpen = True
MSComm1.InputMode = comInputModeBinary        'Set InputMode  to read binary data
Do Unit MSComm1.InBufferCount < 10            'Wait unit 10 bytes are in the
Input buffer
    DoEvents
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
Loop
Buffer = MSComm1. Input          'Store binary data in buffer
Arr = Buffer                     'Assign to byte array for processing
End Sub
```

2.9.7 Output

ใช้ในการส่งขบวน ของข้อมูลไปยังบัฟเฟอร์ส่งข้อมูล

รูปแบบการใช้งาน

Object.Output [= value]

ค่า value เป็นค่าตัวอักษรที่เขียนไปยังบัฟเฟอร์ส่งข้อมูล คุณสมบัติ Output สามารถใช้ในการส่งข้อมูลตัวอักษรหรือข้อมูลไบนารี่ก็ได้ โดยการส่งข้อมูลเป็นรูปแบบตัวอักษรจะต้องกำหนดข้อมูลเป็นแบบ Variant และมีข้อมูลภายในเป็นแบบ byte

ตัวอย่างโปรแกรมการส่งค่าที่ป้อนจากคีย์บอร์ดไปยังพอร์ตอนุกรม โดยใช้คุณสมบัติ Output

```
Private Sub Form_KeyPress (keyAscii As Integer)
    Dim Buffer as Variant
    MSComm1.ComPort = 1              'Use COM1
    MSComm1.PortOpen = True          'OpenPort
    Buffer = Chr$ (KeyAscii)
    MSComm1.Output = Buffer          'Send DATA
End Sub
```

2.9.8 OutBufferCount

คือค่าจำนวนข้อมูลตัวอักษรที่เก็บอยู่ในบัฟเฟอร์ภาคส่ง และสามารถใช้คำสั่งนี้เพื่อเคลียร์บัฟเฟอร์ภาคส่งได้ด้วย

รูปแบบการใช้งานคำสั่ง

Object.OutBufferCount [= value]

ผู้ใช้งานสามารถเคลียร์บัฟเฟอร์ภาส่งได้โดยกำหนดค่า OutBufferCount เท่ากับ "0"

หมายเหตุ ระวังการสับสนระหว่างคุณสมบัติ OutBufferCount กับ OutBufferSize ซึ่ง OutBufferSize ใช้เพื่อกำหนดขนาดของบัฟเฟอร์ภาคส่ง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.9.9 OutBufferSize

กำหนดค่าและคืนค่าขนาดของบัฟเฟอร์ภาคส่ง ชนิดตัวแปรเป็นไบต์

รูปแบบการใช้งานคำสั่ง

Object.OutBufferSize [=object]

คุณสมบัติ OutBufferSize ใช้สำหรับกำหนดขนาดของบัฟเฟอร์ภาคส่ง โดยปกติที่ใช้งานจะมีค่าเท่ากับ 512 ไบต์

หมายเหตุ การกำหนดค่าบัฟเฟอร์ภาคส่งที่มากเกินไป จะทำให้หน่วยความจำเหลือให้ใช้งานน้อย แต่อย่างไรก็ตามถ้ากำหนดค่าน้อยเกินไป จะทำให้เกิดข้อมูลล้นบัฟเฟอร์ขึ้นได้ ยกเว้นจะมีการใช้ แฮนด์เช็ค วิธีการที่ถูกต้องในการกำหนดค่า คือ ทดลองใช้ค่าเริ่มต้น คือ ค่า512 ไบต์ดูก่อน ถ้าโปรแกรมทำงานแล้วเกิดการล้นของข้อมูลค่อยเพิ่มค่าของ OutBufferSize ให้มากขึ้น

2.9.10 เหตุการณ์ OnComm

เหตุการณ์ OnComm จะถูกสร้างขึ้นเมื่อค่าของคุณสมบัติ Commevent มีการเปลี่ยนแปลงเพื่อแสดงผลการเปลี่ยนแปลงเหล่านั้นแบบทันทีทันใดหรือแสดงข้อผิดพลาดที่เกิดขึ้น ตัวอย่างโปรแกรมย่อย OnComm เพื่อนำเหตุการณ์ CommEvent มาแสดง

```
Private Sub MSComm_OnComm ()
Select CaseMSComm1.CommEvent
    'Handle each event or error by placing
    'code below each case statement
    'Error
            Case comEventBreak          'A Break was received
            Case comEvenCDTO            'CD (RLSD)Timeout
            Case comEvenCTSTO           'CTS Timeout
            Case comEvenDSRTO           'DSR Timeout
            Case comEvenFrame           'Framing Erorr
            Case comEvenOverrun         'Data Lost
            Case comEvenRxOver          'Receive buffer overflow
            Case comEvenRxParity        'Parity Erorr
            Case comEventFull           'Transmit buffer full
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        'Events
        Case comEvCD            'Change in the CD line.
        Case comEvCTS           'Change in the CYS line.
        Case comEvDSR           'Change in the DSR line.
        Case comEvRing          'Chang in the Ring Indicator.
        Case comEvReceive       'Receive Rthreshold # of chars.
        Case comEVSend          'Sthreshold number in the 'transmit buffer.
        Case comEvEof           'An EOF charater was found in the input
stream
        End Select
End Sub
```

2.9.11 การใช้ MSComm เพื่อการติดต่อฮาร์ดแวร์

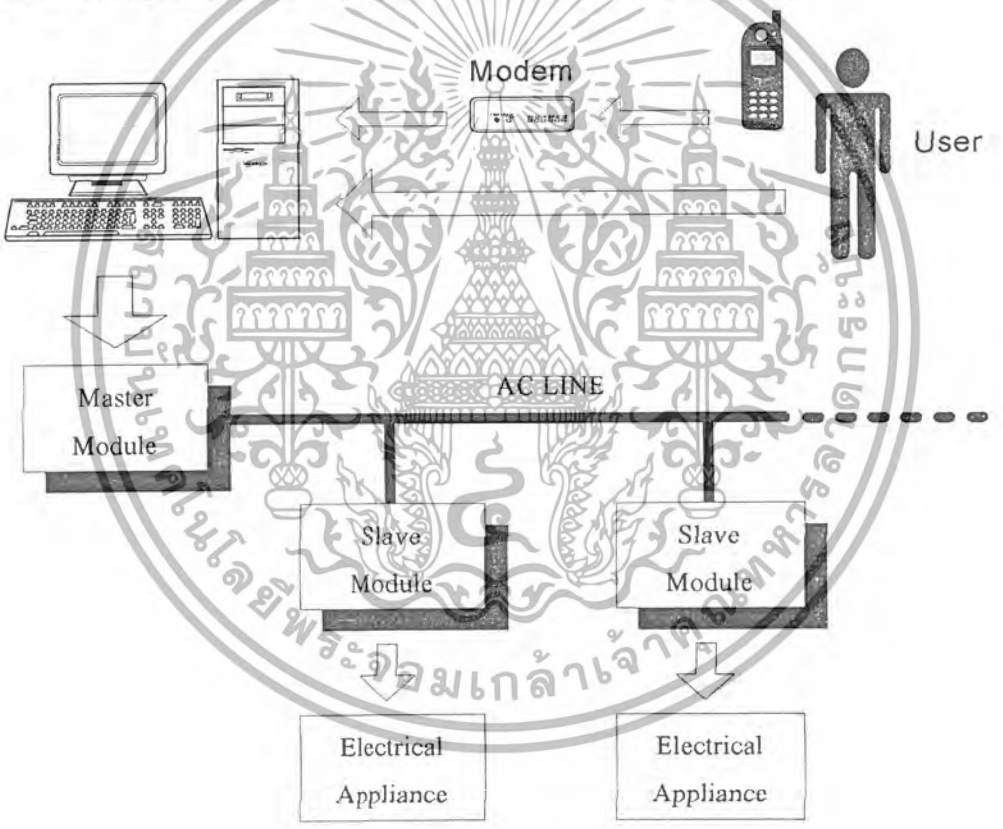
จากรายละเอียดของ MSComm ที่กล่าวไปในตอนต้นนั้น จะเห็นได้ว่าวิธีการที่จะอ่านค่าหรือเขียนค่าไปยังขาสถานะและขาควบคุมของพอร์ตอนุกรมสามารถทำได้ง่ายดายมาก โดยใช้คำสั่งเหล่านี้

DTREnable	สำหรับคำสั่งให้ขาDTR มีลอจิก"0" หรือ"1"
RTSEnable	สำหรับสั่งให้ขา RTS มีลอจิก "0" หรือ "1"
CTSHolding	สำหรับอ่านค่าสถานะจากขาCTS ว่ามีลอจิก "0" หรือ "1"
CDHloding	สำหรับอ่านค่าสถานะจากขา DCD ว่ามีลอจิก "0" หรือ "1"
DSRHolding	สำหรับอ่านค่าสถานะจากขา DSR ว่ามีลอจิก "0" หรือ "1"
Break	สำหรับการสั่งให้ขาTxd มีลอจิก "0" หรือ "1"

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 3
หลักการโดยรวมของระบบ

หลักการทำงานของระบบคือ เครื่องคอมพิวเตอร์รอรับคำสั่งจากผู้ใช้งานอันจะมีได้สองทางคือ ทางคีย์บอร์ดและทางโทรศัพท์ แล้วส่งคำสั่งที่ได้แก่ MASTER MODULE เมื่อได้รับคำสั่ง MASTER MODULE จะแปลงคำสั่งเป็นข้อมูลแล้วส่งผ่านระบบไฟฟ้ากำลังสู่ SLAVE MODULE การส่งข้อมูลจะส่งออกทีละบิทที่จุด Zero Crossing ของไฟฟ้ากำลัง โดยที่รูปแบบของข้อมูลจะมีสองแบบคือ คำสั่งเปิดปิดอุปกรณ์ไฟฟ้า และ คำสั่งตรวจดูสถานะ SLAVE MODULE ตรวจข้อมูลที่รับได้ถ้าข้อมูลที่อยู่ตรงกับที่อยู่ที่ถูกตั้งไว้ จะทำงานตามข้อมูลที่ได้รับ



รูปที่ 3.1 บล็อกไดอะแกรมโดยรวมของระบบ

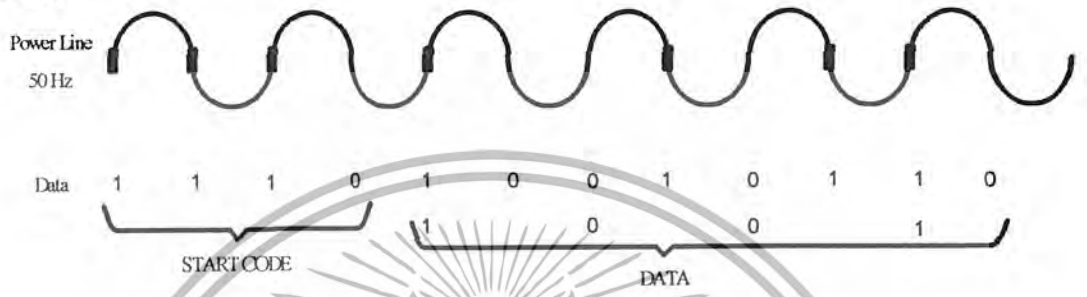
3.1 รูปแบบการส่งข้อมูล

การส่งข้อมูลออกระบบไฟฟ้ากำลัง ข้อมูลที่มีค่า 1 จะปล่อยความถี่ 98 กิโลเฮิรตซ์ออกสายไฟฟ้ากำลังที่จุด Zero Crossing นาน 0.001 วินาที ส่วนข้อมูลค่า 0 จะไม่ส่งสัญญาณใดๆ รูปแบบการส่งข้อมูลมีสองรูปแบบคือ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.1.1 รูปแบบการส่งคำสั่งเปิดปิดอุปกรณ์ไฟฟ้า

ข้อมูลจะประกอบด้วย รหัสเริ่มต้น 1110 ที่อยู่ของปลายทาง(6 บิท)และบิทตรงข้าม(6 บิท) คำสั่งเปิดปิดและบิทตรงข้าม โดยที่ คำสั่งเปิดข้อมูลจะมีค่าเป็น 1 ส่านคำสั่งปิดจะมีค่าเป็น 0 สุดท้าย คือ ค่าพาริตี้ของข้อมูล การที่ต้องใส่บิทตรงข้ามในข้อมูลเพื่อป้องกันรูปแบบของข้อมูลซ้ำกับรหัสเริ่มต้น

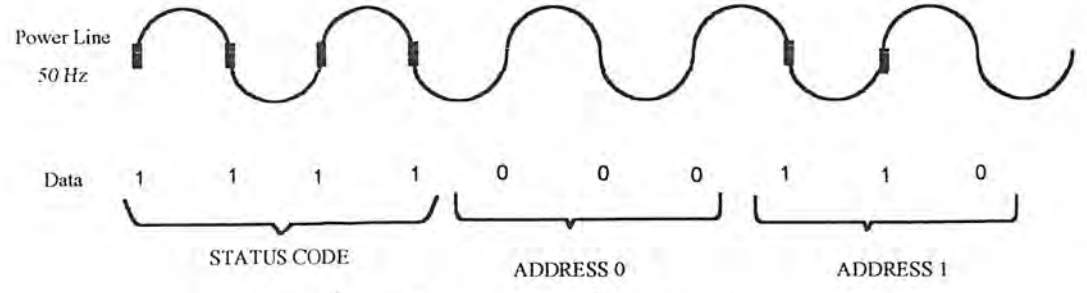


รูปที่ 3.2 รูปแบบการส่งคำสั่งเปิดปิดอุปกรณ์ไฟฟ้า

ตัวอย่างเช่น คำสั่งเปิดอุปกรณ์ไฟฟ้าตัวที่ 1 ข้อมูลที่ส่งจะเป็น รหัสเริ่มต้น 1110 ที่อยู่ 000001 กับบิทตรงข้าม 111110 กลายเป็น 0101010101110 คำสั่งเปิด 1 กับบิทตรงข้าม 0 และสุดท้าย พาริตี้ 0 ดังนั้นข้อมูลที่ส่งจะเป็น 1110 0101010101110 0 รวมทั้งหมด 19 บิท

3.1.2 รูปแบบการส่งคำสั่งตรวจดูสถานะ

เมื่อ MASTER MODULE ต้องการดูสถานะของ SLAVE MODULE จะส่งรหัส 1111 ออกระบบไฟฟ้ากำลังแล้วตรวจดูข้อมูลที่ตอบกลับมา ข้อมูลที่ตอบหลับมาจะมีขนาด 3 บิทนั้นคือมี 4 สถานะที่จะได้คือ *SLAVE MODULE ไม่ตอบสนอง(000)* อุปกรณ์ไฟฟ้าเสียหรือไม่กินกระแส (010) อุปกรณ์ไฟฟ้าปิดอยู่(100) และอุปกรณ์ไฟฟ้าเปิดอยู่ (110) ส่วนการตอบกลับจะตอบในช่องสัญญาณของตัวเองเช่น ที่อยู่เป็น 0 จะตอบค่าสถานะที่ Zero Crossing 1, 2 และ 3 หลังจาก รหัส 1111 ที่อยู่เป็น 1 จะตอบค่าสถานะที่ Zero Crossing 4, 5 และ 6

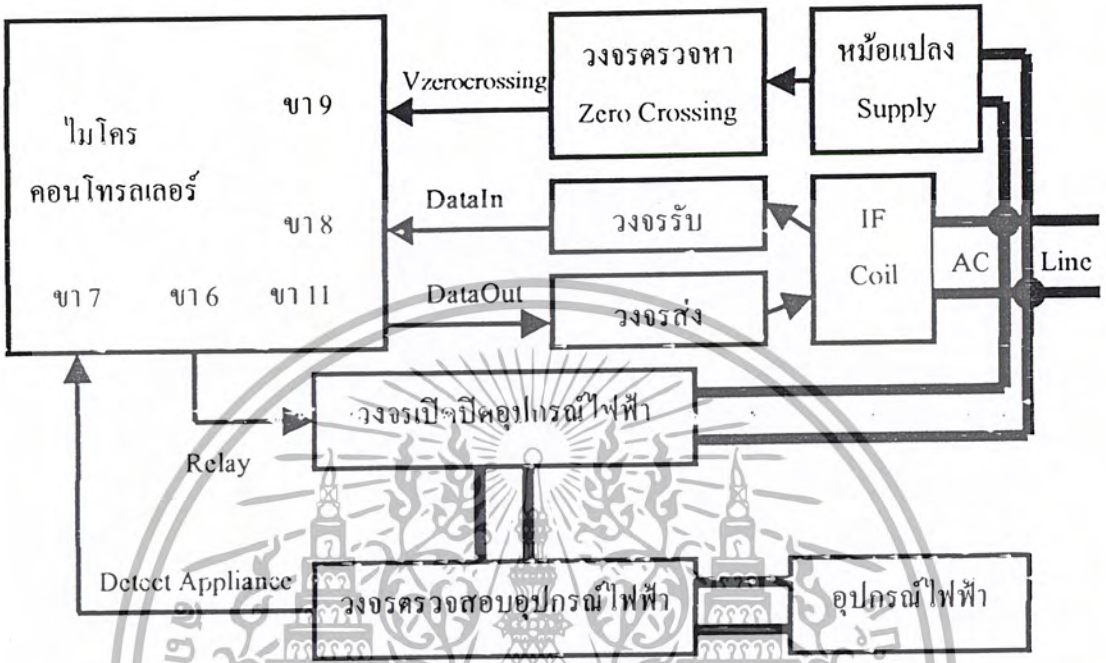


รูปที่ 3.3 รูปแบบการส่งคำสั่งตรวจดูสถานะ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.2 องค์ประกอบของโมดูล

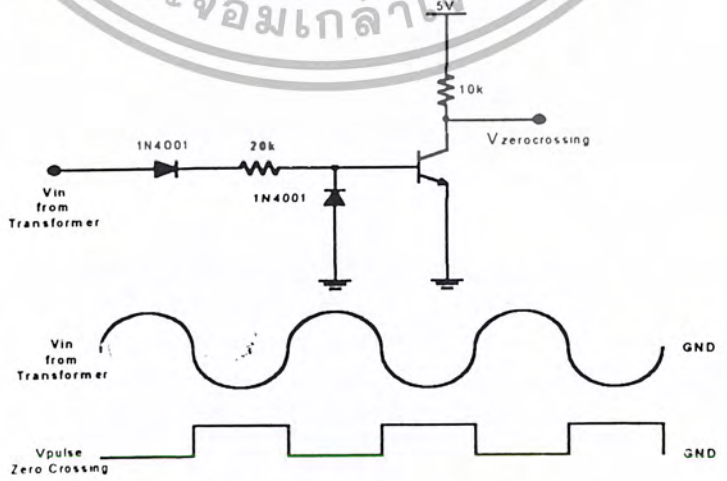
ภายในโมดูลประกอบด้วยส่วนต่างๆดังนี้



รูปที่3.4 องค์ประกอบของโมดูล

3.2.1 วงจรตรวจสอบ Zero Crossing

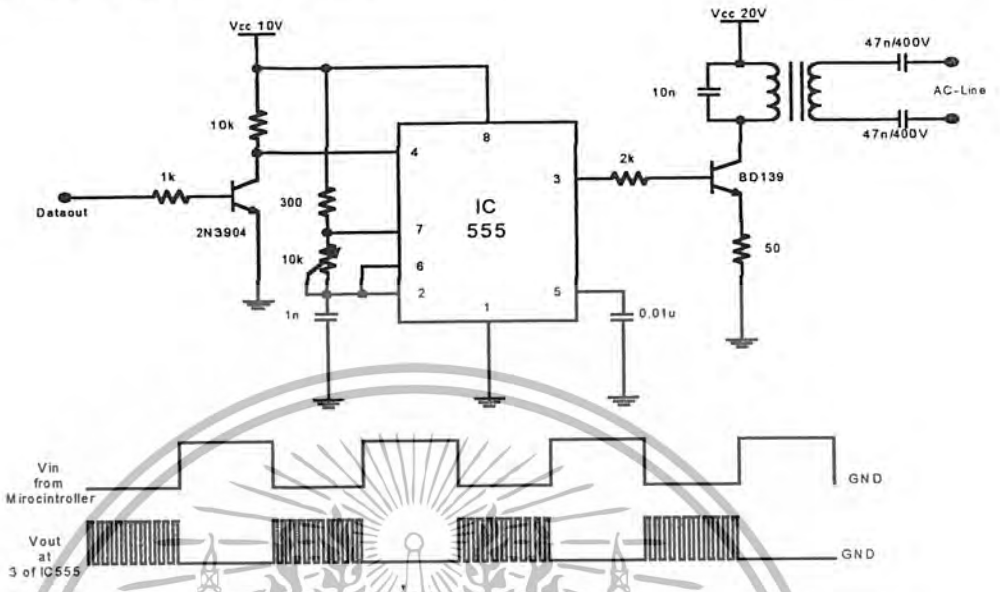
วงจรตรวจ Zero Crossing เป็นวงจรที่แปลงสัญญาณไซน์ 50 Hz เป็นสัญญาณพัลน์ 50 Hz ซึ่งจุด Zero Crossing คือจุดที่สัญญาณพัลน์เปลี่ยนระดับแรงดันจาก 0 เป็น 5 โวลท์ หรือ จาก 5 เป็น 0 โวลท์



รูปที่ 3.5 วงจรตรวจสอบ Zero Crossing

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

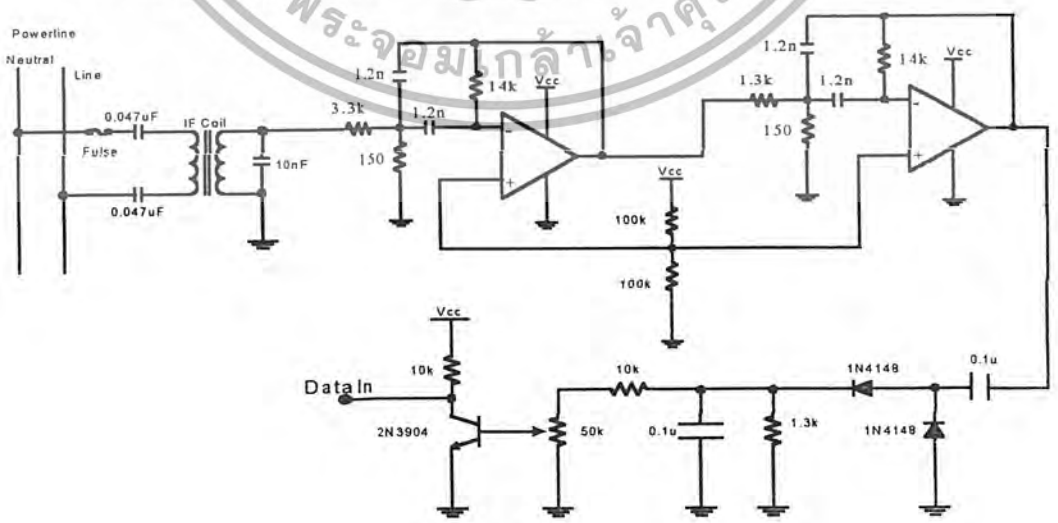
3.2.2 การทำงานของวงจรส่ง(Transmitter)



รูปที่ 3.6 วงจรส่ง

วงจรส่ง ทานซิสเตอร์ 2N3904 ทำงานเป็นสวิทช์ปิดเปิด ไอซี555 คือเมื่อ ทานซิสเตอร์ off แรงดันที่ขา 4 ของไอซี555ซึ่งเป็นขา Reset จะเป็น 12V ไอซี555จะทำงาน แต่เมื่อ ทานซิสเตอร์ on แรงดันที่ขา 4 ของไอซี555 จะเป็น Ground ทำให้ไอซี555 ไม่ทำงาน เมื่อไอซี555ทำงานจะสร้างพัลส์ความถี่ 98kHz ออกมาเข้าทรานซิสเตอร์ T_2 ซึ่งต่อเป็นวงจรขยายคลาส ซี สัญญาณจะถูกส่งออกสายไฟฟ้ากำลังผ่านหม้อแปลง ไอเอฟ

3.2.3 การทำงานของวงจรภาครับ(Receiver)



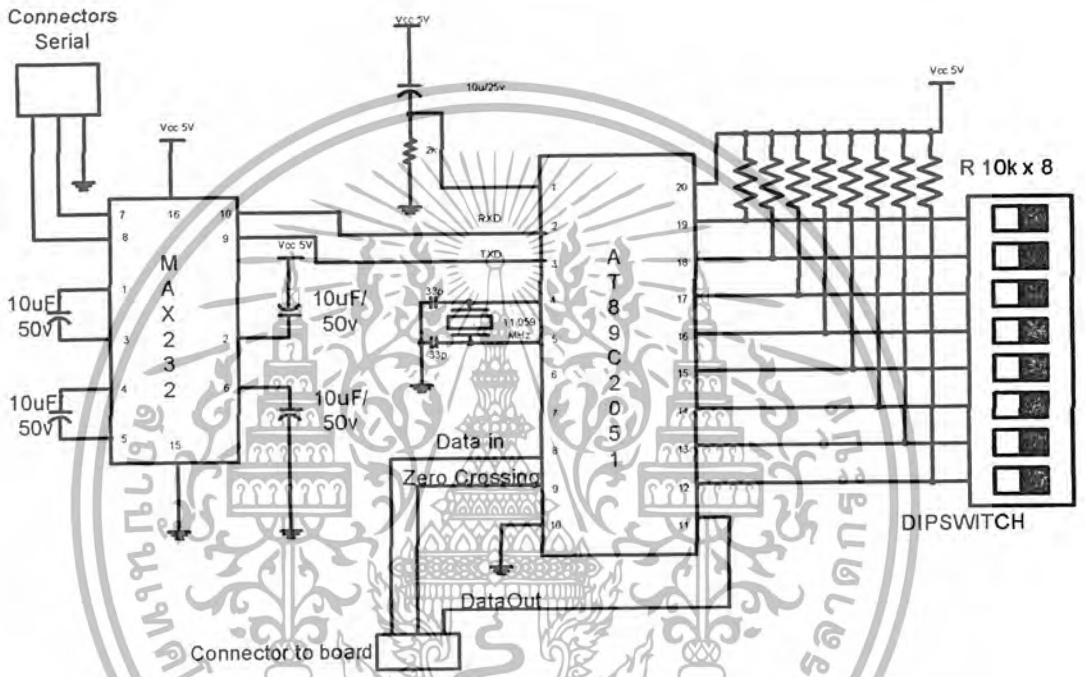
รูปที่ 3.7 วงจรภาครับ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สัญญาณจากสายไฟฟ้ากำลังเข้าวงจรผ่านหม้อแปลง ไอเอฟ ถูกขยายขึ้น 10 เท่าด้วยวงจร Active Filter ที่ความถี่กลาง 98kHz จากนั้นตัดคลื่นพาร์ออกและสร้างเป็นพัลส์เข้าไมโครคอนโทรลเลอร์

3.2.4 การทำงานของส่วนไมโครคอนโทรลเลอร์

3.2.4.1 ไมโครคอนโทรลเลอร์ของ Master Module

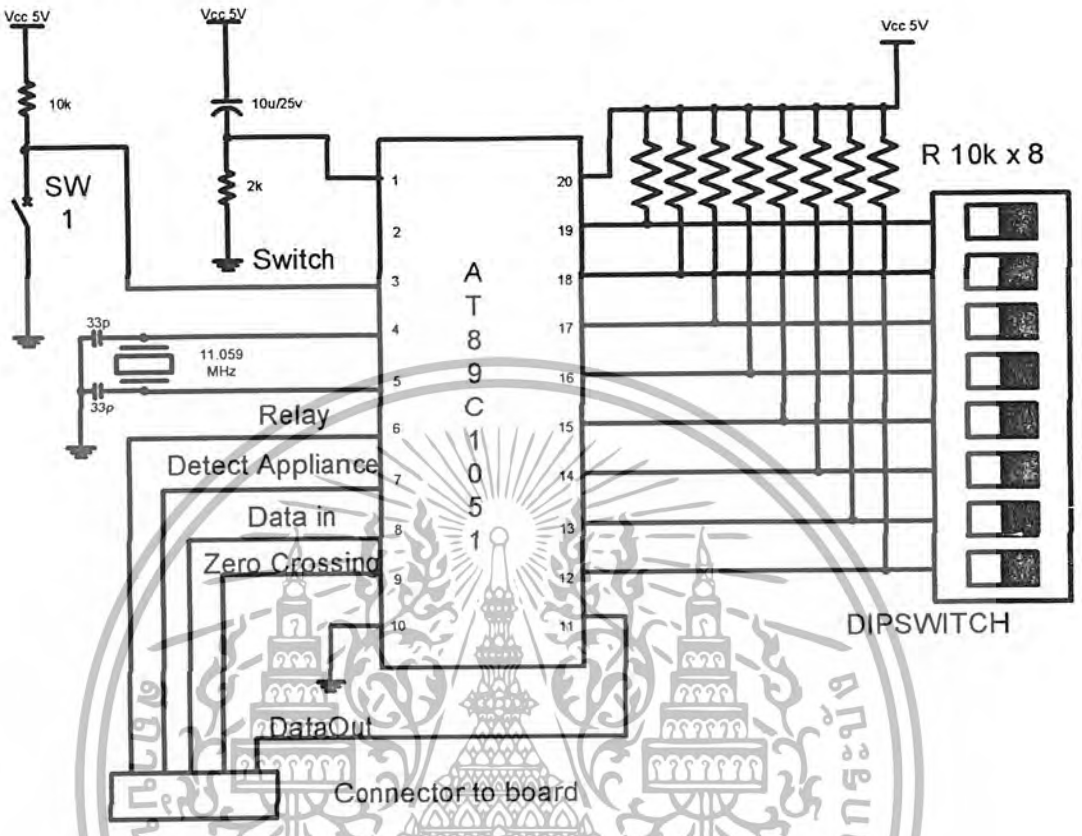


รูปที่ 3.8 ไมโครคอนโทรลเลอร์ของ Master Module

ใช้ไมโครคอนโทรลเลอร์เบอร์ AT89C2051 เนื่องจากต้องการใช้การส่งข้อมูลแบบอนุกรม การทำงานไมโครคอนโทรลเลอร์จะคอยรับคำสั่งจากคอมพิวเตอร์ทางพอร์ทอนุกรมด้วยความเร็ว 9600 บิตต่อวินาที ผ่านไอซี MAX232 คำสั่งจะมี 3 รูปแบบคือ "InitialX" เป็นคำสั่งตรวจสอบว่าโมดูลทำงานหรือไม่ เมื่อโมดูลได้รับคำสั่งจะตอบ "InitialX" แก่คอมพิวเตอร์, "OxxyX" เป็นคำสั่งเปิดหรือปิดอุปกรณ์ไฟฟ้า โดยที่ xx คือหมายเลขปลายทาง ส่วน y เป็นคำสั่ง 1 หมายถึง เปิด 0 หมายถึง ปิด เช่น คำสั่ง เปิดอุปกรณ์หมายเลข 12 คำสั่งจะเป็น "O121X" เป็นต้น เมื่อโมดูลได้รับคำสั่งจะส่งคำสั่งผ่านไปให้โมดูลปลายทางนั้นๆ ถ้าการส่งข้อมูลทำได้สมบูรณ์จะคืนค่า "DonedX" แก่คอมพิวเตอร์แต่ถ้าไม่สมบูรณ์จะคืนค่า "ErrorX" และสุดท้าย "StateX" เป็นคำสั่งขอดูสถานะของอุปกรณ์ไฟฟ้า เมื่อโมดูลได้รับจะคืนสถานะอุปกรณ์ไฟฟ้าตามจำนวนที่ตั้งไว้ที่ดิฟสวิทช์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.2.4.2 ไมโครคอนโทรลเลอร์ของ Slave Module



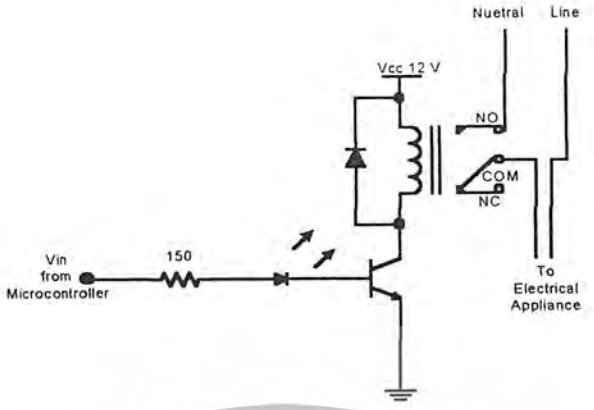
รูปที่ 3.9 ไมโครคอนโทรลเลอร์ของ Slave Module

เลือกใช้ไมโครคอนโทรลเลอร์เบอร์ AT89C1051 ไมโครคอนโทรลเลอร์จะคอยทำการตรวจสอบข้อมูลจากวงจรรับข้อมูลแล้วตอบสนองข้อมูลนั้นถ้าที่อยู่ปลายทางของข้อมูลตรงกับที่อยู่ที่ตั้งไว้ที่ดิพสวิทช์ นอกจากนี้ยังคอยตรวจสอบสถานะของสวิทช์ SW1 ด้วยว่ามีการเปลี่ยนแปลงหรือไม่ ถ้าเปลี่ยนจะทำการเปลี่ยนสถานะการเปิดหรือปิดอุปกรณ์ไฟฟ้าทำให้ผู้ใช้สามารถเปิดปิดอุปกรณ์ไฟฟ้าได้ที่โมดูลปลายทางอีกทาง และทำการตรวจค่าจากวงจรตรวจสอบอุปกรณ์ไฟฟ้าว่าอุปกรณ์ไฟฟ้านั้นๆ ยังทำงานหรือไม่

3.2.5 การทำงานของวงจร เปิดปิดอุปกรณ์ไฟฟ้า

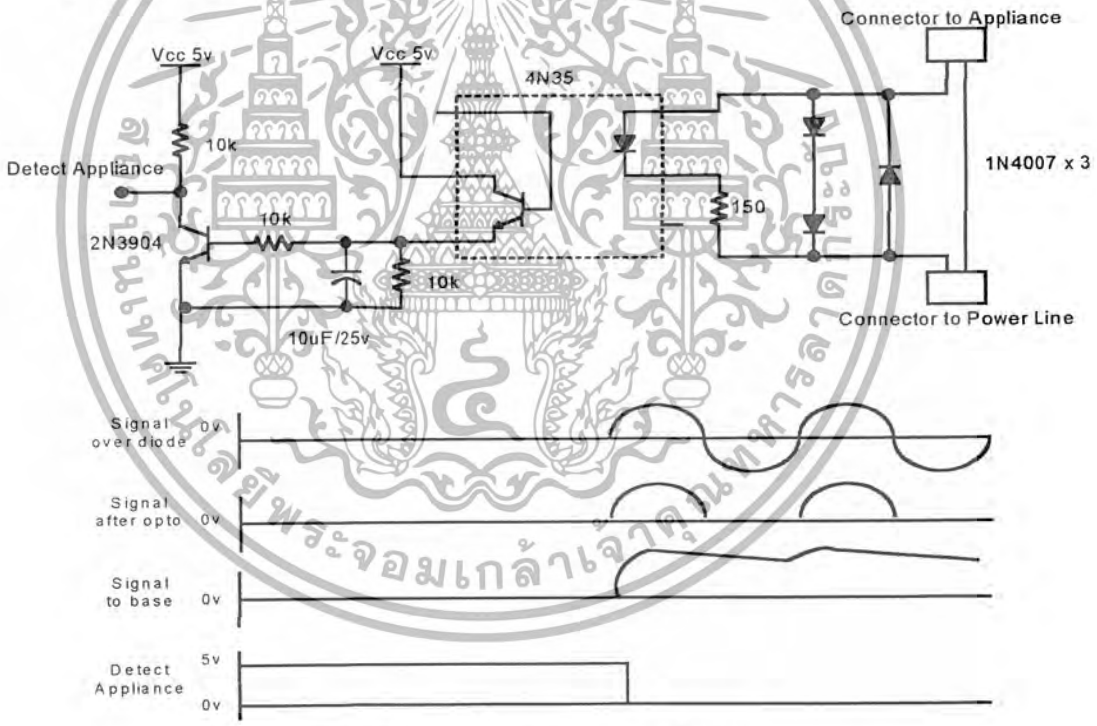
ขาของไมโครคอนโทรลเลอร์ที่ส่งค่าควบคุมอุปกรณ์ไฟฟ้าจะต่ออยู่กับทรานซิสเตอร์ ซึ่งใช้ในการเปิดปิด Relay 5 ขาเพื่อเปิดปิดอุปกรณ์ไฟฟ้า ถ้าค่าจากไมโครคอนโทรลเลอร์เป็น 1 จะเป็นการจ่ายไฟฟ้าให้แก่อุปกรณ์ที่ติดตั้งไว้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.10 วงจรเปิดปิดอุปกรณ์ไฟฟ้า

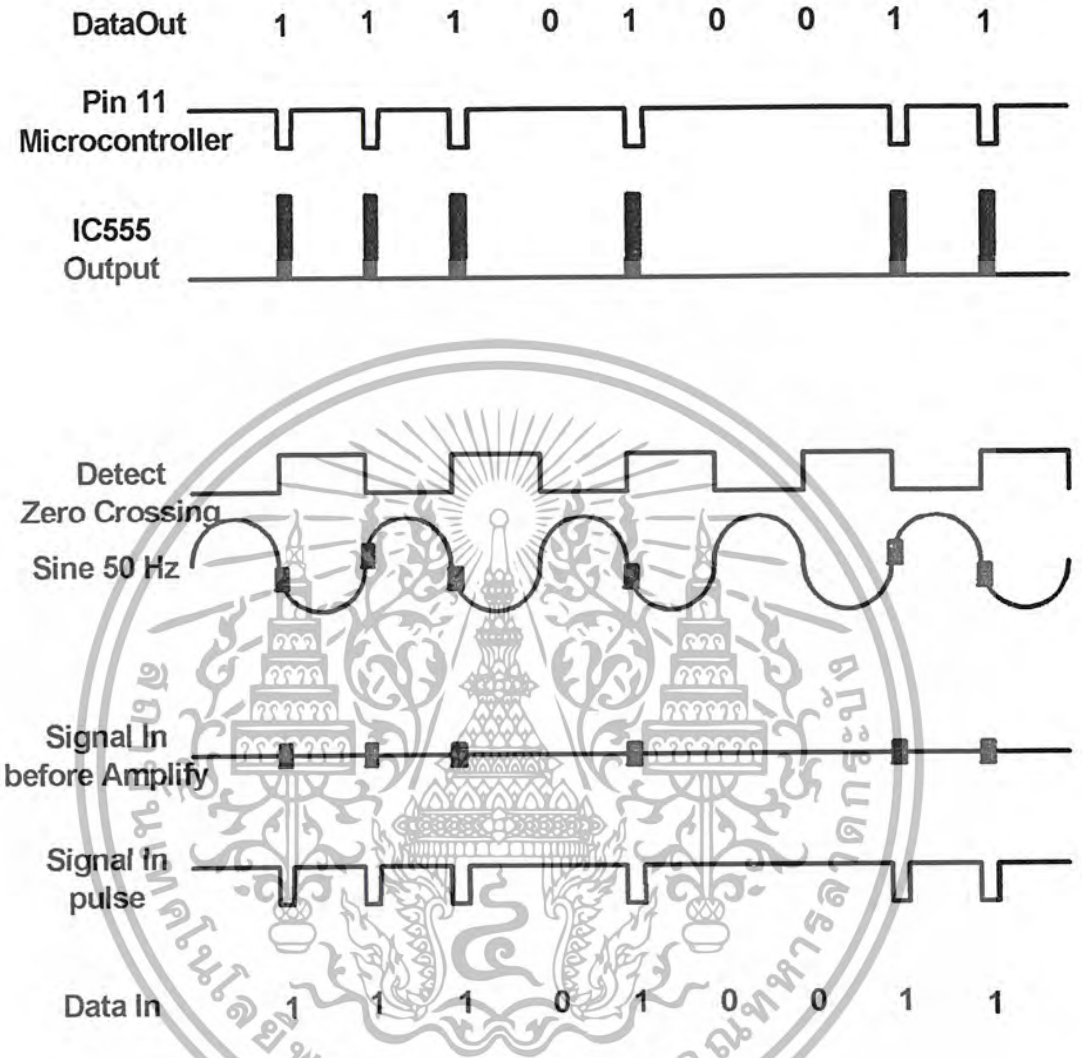
3.2.6 การทำงานของวงจรตรวจสอบอุปกรณ์ไฟฟ้า



รูปที่ 3.11 วงจรตรวจสอบอุปกรณ์ไฟฟ้า

การทำงาน ในขณะที่มีอุปกรณ์ไฟฟ้าต่ออยู่จะมีกระแสไหลผ่านออปโต้ ทำให้มีสัญญาณแรงดันทางด้านขาเบสของทรานซิสเตอร์ส่งผลให้ ค่าที่จุด Detect Appliance มีค่าเป็น 0 (GND) แต่ในขณะที่ไม่มีอุปกรณ์ไฟฟ้าต่ออยู่จะไม่มีกระแสผ่านออปโต้ดังนั้น ค่าที่จุด Detect Appliance มีค่าเป็น 1 (5 V)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.12 รูปแบบการส่งและรับข้อมูลของโมดูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

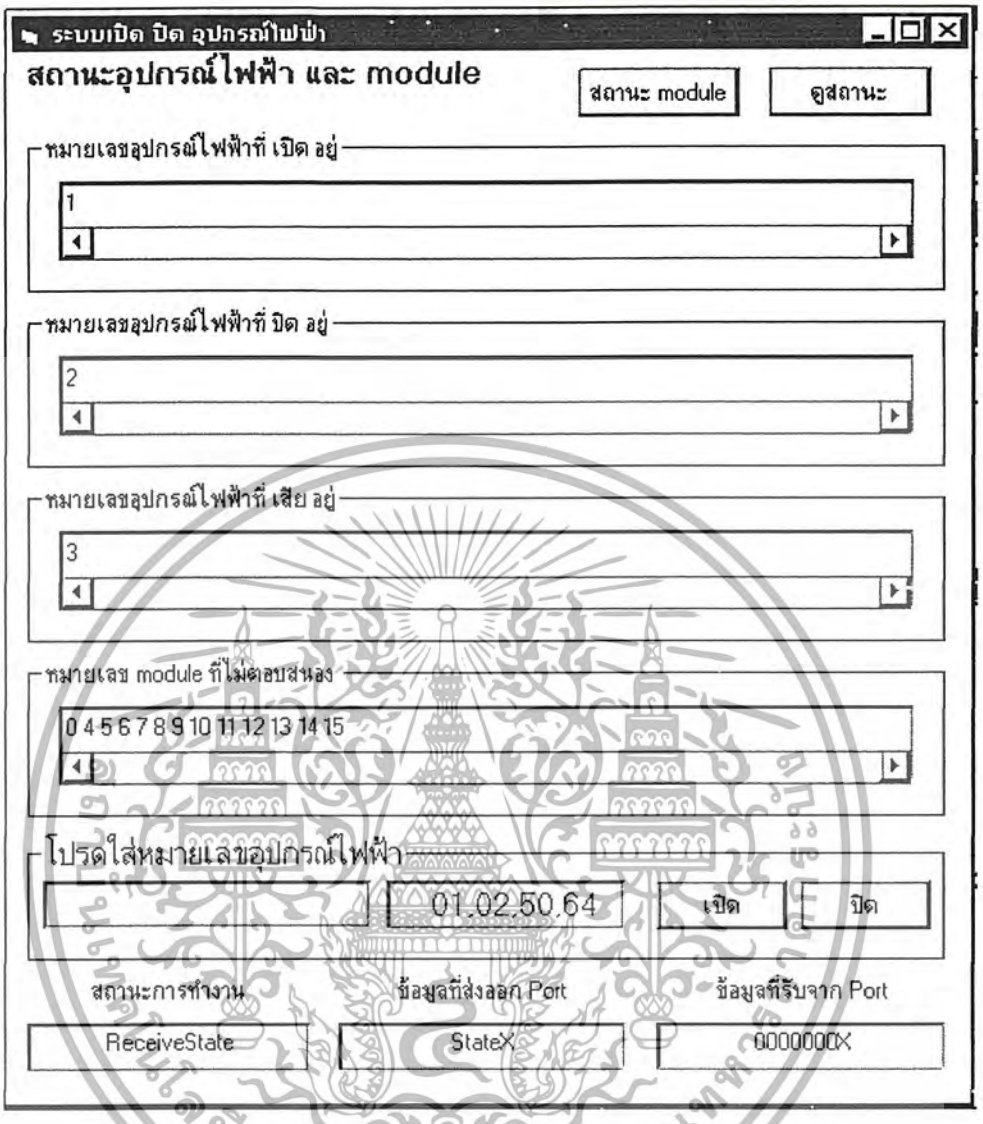
3.3 การทำงานของโปรแกรมเปิดปิดอุปกรณ์ไฟฟ้า

เมื่อเปิดโปรแกรมจะพบกับ หน้าต่าง "ตั้งค่าของโปรแกรม" จะต้องตั้งค่า 2 อย่างให้โปรแกรมคือ ที่อยู่ของ ไฟล์เสียงที่ใช้ และค่าพอร์ทสื่อสารที่ใช้ติดต่อกับโมดูล ประกอบด้วย หมายเลขพอร์ท และค่าตัวแปลอันมีรูปแบบ "บอดเรต, พาริตี้, จำนวนบิตข้อมูล, บิตปิดท้าย"

รูปที่ 3.13 หน้าต่างตั้งค่าโปรแกรม

เมื่อกดปุ่ม "ตกลง" จะเข้าสู่หน้าต่าง "ระบบ เปิด ปิด อุปกรณ์ไฟฟ้า" ภายในหน้าต่างประกอบไปด้วย ส่วนแสดงผลสถานะของอุปกรณ์ไฟฟ้าโดยแสดงเป็นหมายเลขตามที่ตั้งไว้ และส่วนรับค่าอันเป็นการรับค่าแบบข้อความคือ พิมพ์หมายเลขที่ต้องการแล้วกดปุ่มคำสั่ง รูปแบบของหมายเลขกำหนดให้เป็น 2 ตัวอักษรเช่น หมายเลข 1 ให้ใส่เป็น 01 โดยที่ใส่หมายเลขได้ตั้งแต่ 00 ถึง 62 นอกจากนี้ถ้าต้องการใส่มากกว่าหนึ่งหมายเลขให้คั่นด้วยเครื่องหมาย " , " เช่น 01,02,12,62 หลังการกดปุ่มคำสั่งโปรแกรมจะถามการยืนยันการทำงาน และจะตอบว่าการส่งข้อมูลถึงโมดูลหมายเลขใดเกิดความผิดพลาดขึ้น การตรวจสถานะของโมดูลตัวลูกโปรแกรมจะทำการตรวจค่าทุกๆ 5 วินาที

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่3.14 หน้าต่าง ระบบ เปิด ปิด อุปกรณ์ไฟฟ้า

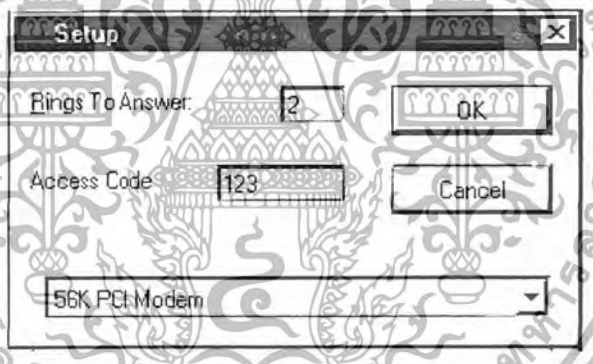
รูปที่ 3.15 หน้าต่างยืนยันการเปิดหรือปิดอุปกรณ์ปลายทางหมายเลข 01

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 3.16 หน้าต่าง บอกว่าเกิดการผิดพลาดในการส่งข้อมูลถึงอุปกรณ์ปลายทางหมายเลข 01

3.4 การทำงานของโปรแกรมรับคำสั่งจากโทรศัพท์

โปรแกรมเขียนโดยใช้ฟังก์ชันจากไฟล์ TAPI.DLL ที่มีอยู่ในไลบรารี ของวินโดว์ ก่อนการใช้งานต้องทำการตั้งค่าให้กับโปรแกรมที่ปุ่ม "SETUP" ค่าที่ต้องใส่ประกอบด้วย จำนวนครั้งริงค์โทนที่ดังก่อนที่โปรแกรมจะรับสาย รหัสผ่าน และค่าอุปกรณ์โมเดมที่ใช้ เมื่อตั้งค่าเสร็จแล้ว ให้กดปุ่ม "ON" เพื่อเริ่มใช้โปรแกรม



รูปที่ 3.17 หน้าต่างตั้งค่าของโปรแกรมรับคำสั่งจากโทรศัพท์

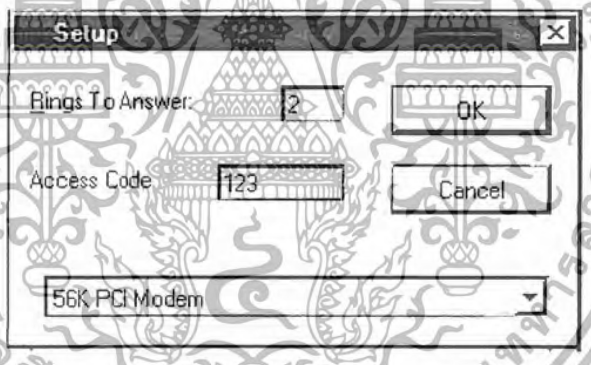
การทำงานเมื่อมีสัญญาณริงค์โทนเข้ามามีจำนวนครั้งมากที่ตั้งไว้ โปรแกรมรับสายแล้วมีเสียงตอนรับและบอกให้ใส่รหัสผ่าน การใส่รหัสให้กดหมายเลขรหัสแล้วตามด้วยเครื่องหมายดอกจัน(*) ถ้ารหัสผิดพลาดจะมีเสียงบอกให้ใส่รหัสอีกครั้ง ถ้ารหัสถูกต้องจะมีเสียงตอนรับแล้วบอกการทำงาน คือ ให้กด หนึ่ง เมื่อต้องการเปิดปิดอุปกรณ์ไฟฟ้า และกด สอง เมื่อต้องการดูสถานะอุปกรณ์ไฟฟ้า หลังจากกดหนึ่ง โปรแกรมมีเสียงบอกวิธีการใส่ข้อมูลคือ ให้กดหมายเลขอุปการณ์ไฟฟ้าเป็นแบบ 2 หลัก ตามด้วยคำสั่ง หนึ่งเพื่อเปิดอุปกรณ์ไฟฟ้าและ สองเพื่อปิด แล้วปิดท้ายด้วยเครื่องหมายดอกจัน (*) เช่น ต้องการเปิดอุปกรณ์ตัวที่ 1 จะกดคำสั่งเป็น 011* หลังจากกดคำสั่งโปรแกรมจะบอก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 3.16 หน้าต่าง บอกว่าเกิดการผิดพลาดในการส่งข้อมูลถึงอุปกรณ์ปลายทางหมายเลข 01

3.4 การทำงานของโปรแกรมรับคำสั่งจากโทรศัพท์

โปรแกรมเขียนโดยใช้ฟังก์ชันจากไฟล์ TAPI.DLL ที่มีอยู่ในไลบรารี ของวินโดว์ ก่อนการใช้งานต้องทำการตั้งค่าให้กับโปรแกรมที่ปุ่ม "SETUP" ค่าที่ต้องใส่ประกอบด้วย จำนวนครั้งริงค์โทนที่ดังก่อนที่โปรแกรมจะรับสาย รหัสผ่าน และค่าอุปกรณ์โมเดมที่ใช้ เมื่อตั้งค่าเสร็จแล้ว ให้กดปุ่ม "ON" เพื่อเริ่มใช้โปรแกรม

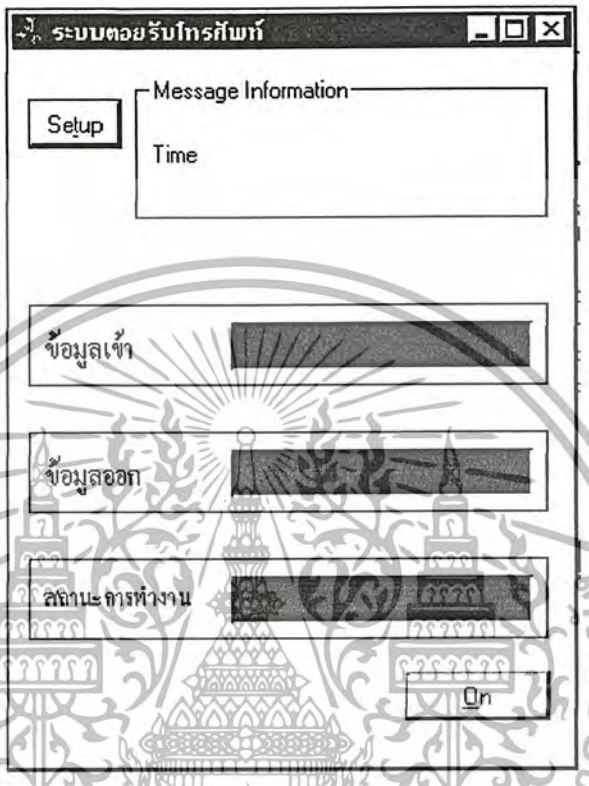


รูปที่ 3.17 หน้าต่างตั้งค่าของโปรแกรมรับคำสั่งจากโทรศัพท์

การทำงานเมื่อมีสัญญาณริงค์โทนเข้ามามีจำนวนครั้งมากที่ตั้งไว้ โปรแกรมรับสายแล้วมีเสียงตอนรับและบอกให้ใส่รหัสผ่าน การใส่รหัสให้กดหมายเลขรหัสแล้วตามด้วยเครื่องหมายดอกจัน(*) ถ้ารหัสผิดพลาดจะมีเสียงบอกให้ใส่รหัสอีกครั้ง ถ้ารหัสถูกต้องจะมีเสียงตอนรับแล้วบอกการทำงาน คือ ให้กด หนึ่ง เมื่อต้องการเปิดปิดอุปกรณ์ไฟฟ้า และกด สอง เมื่อต้องการดูสถานะอุปกรณ์ไฟฟ้า หลังจากกดหนึ่ง โปรแกรมมีเสียงบอกวิธีการใส่ข้อมูลคือ ให้กดหมายเลขอุปการณ์ไฟฟ้าเป็นแบบ 2 หลัก ตามด้วยคำสั่ง หนึ่งเพื่อเปิดอุปกรณ์ไฟฟ้าและ สองเพื่อปิด แล้วปิดท้ายด้วยเครื่องหมายดอกจัน (*) เช่น ต้องการเปิดอุปกรณ์ตัวที่ 1 จะกดคำสั่งเป็น 011* หลังจากกดคำสั่งโปรแกรมจะบอก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ว่าการส่งข้อมูลถึงโมดูลปลายทางนั้นๆถูกต้องหรือไม่ ส่วนการตอบสถานะ จะมีเสียงบอก ส□านะ และตามด้วยเสียงหมายเลขอุปกรณ์ไฟฟ้าจนครบทุกหมายเลขที่ตั้งไว้



รูปที่ 3.18 หน้าต่างโปรแกรมรับคำสั่งจากโทรศัพท์

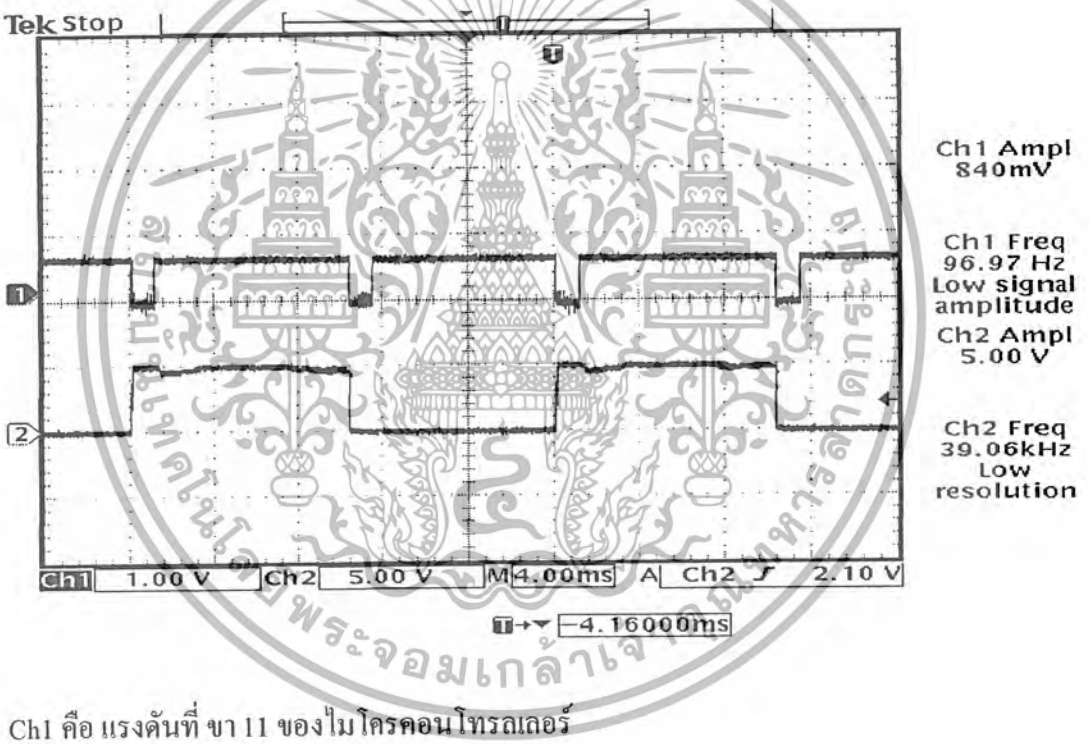
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 4
ผลการทดลอง

4.1 ทดลองการทำงานของวงจรส่ง

วิธีการทดลอง

- โปรแกรมไมโครคอนโทรลเลอร์ให้ทำงานส่งข้อมูลเข้าวงจรส่งด้วยข้อมูล 1 ที่จุด Zero Crossing อย่างต่อเนื่อง
- วัดค่าที่ขา 11 ของไมโครคอนโทรลเลอร์ อันเป็นขาที่ใช้ควบคุมการเปิดปิดไอซี555



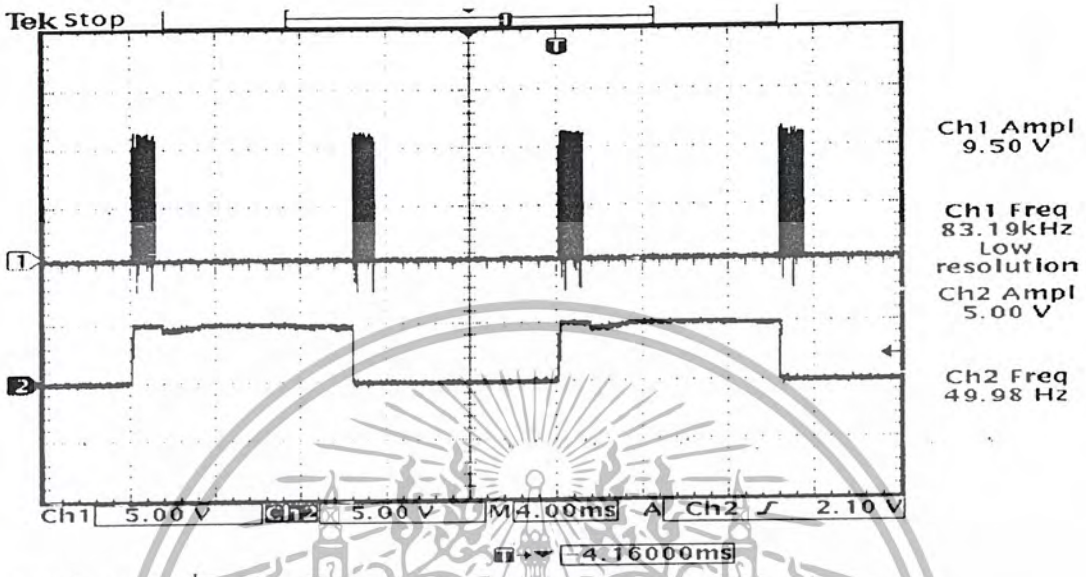
Ch1 คือ แรงดันที่ ขา 11 ของไมโครคอนโทรลเลอร์

Ch2 คือ แรงดันพัลส์จากวงจรตรวจสอบ Zero Crossing

รูปที่ 4.1 สัญญาณที่ ขา 11 ของไมโครคอนโทรลเลอร์ขณะส่งข้อมูล 1111

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- วัดค่าที่ขา 3 ของไอซี555 อันเป็นขาเอาท์พุท

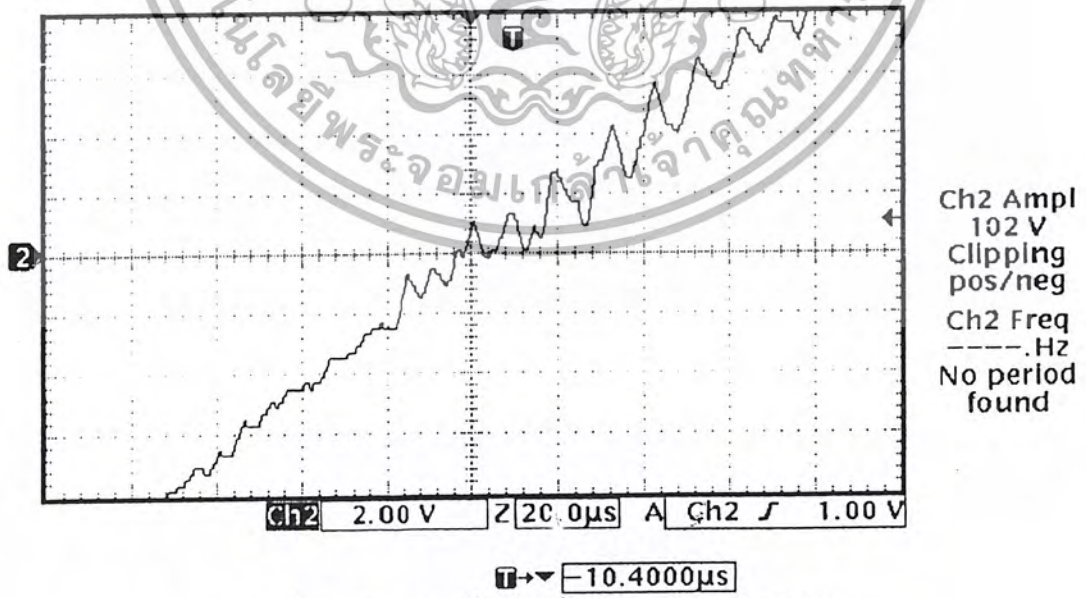


Ch1 คือ แรงดันที่ ขา 3 ของไอซี555

Ch2 คือ แรงดันพัลส์จากวงจรตรวจสอบ Zero Crossing

รูปที่ 4.2 สัญญาณที่ ขา 3 ของไอซี555ขณะส่งข้อมูล 1111

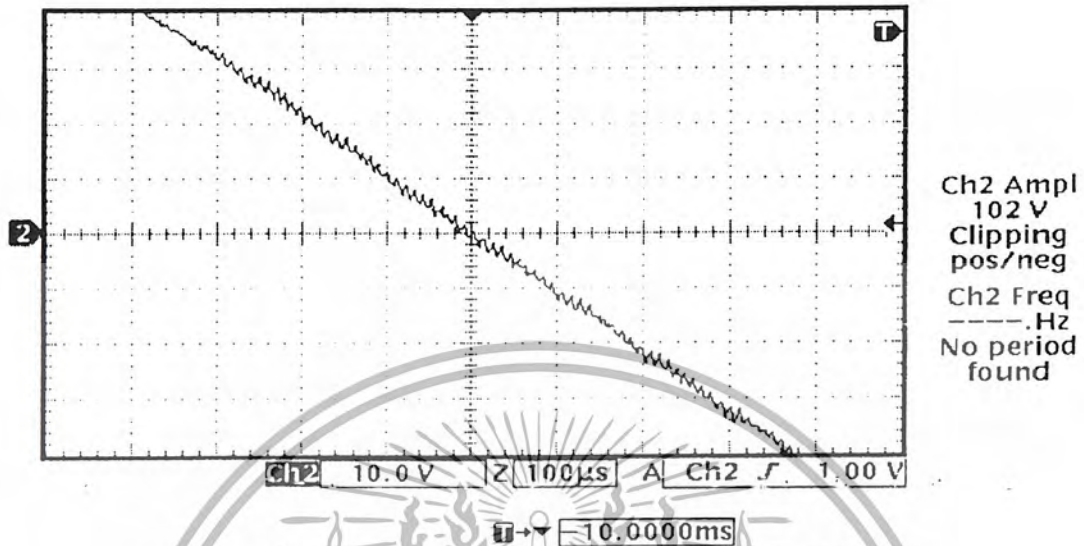
- วัดค่าที่สายไฟฟ้ากำลังที่เฟส 0 องศา



รูปที่ 4.3 สัญญาณคลื่นพาร์ที่จุด 0 องศาของไฟฟ้ากำลัง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- วัดค่าที่สายไฟฟ้ากำลังที่เฟส 180 องศา

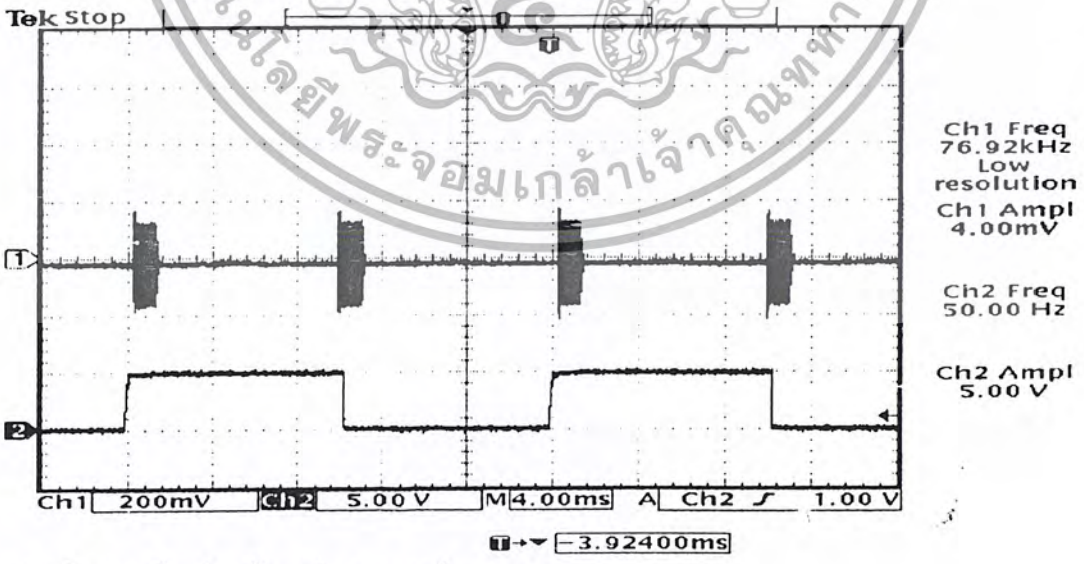


รูปที่ 4.4 สัญญาณคลื่นพาร์ที่จุด 180 องศาของไฟฟ้ากำลัง

4.2 ทดลองการทำงานของวงจรรับ

วิธีการทดลอง

- โปรแกรมไมโครคอนโทรเลอร์ให้ทำงานส่งข้อมูลเข้าวงจรส่งด้วยข้อมูล 1 ที่จุด Zero Crossing อย่างต่อเนื่อง
- วัดค่าที่ขาหม้อแปลง ไอเอฟ ของวงจรรับ



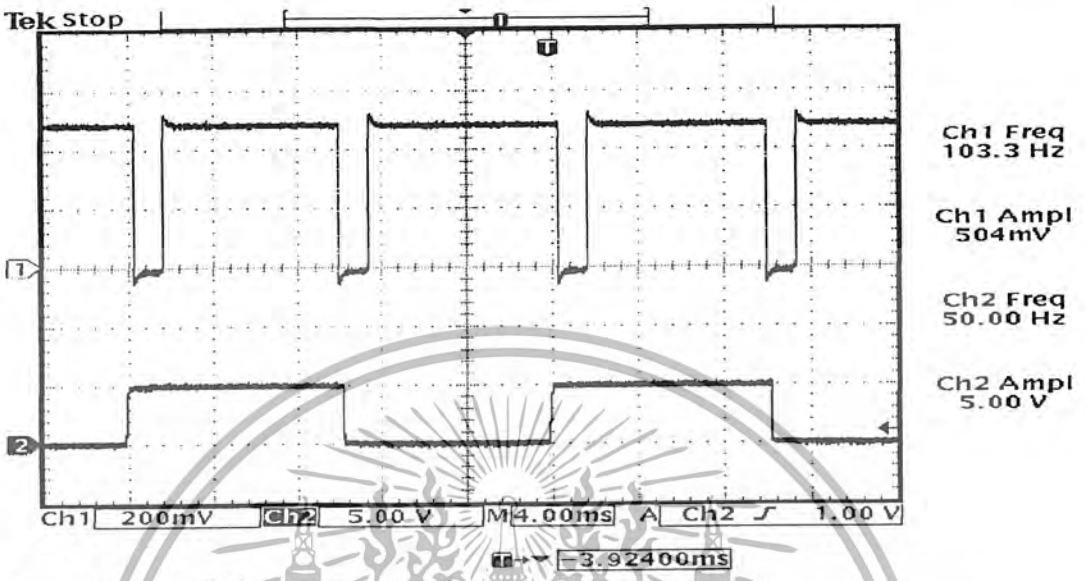
Ch1 คือ ขาหม้อแปลง ไอเอฟ ของวงจรรับ

Ch2 คือ แรงดันพัลส์จากวงจรตรวจสอบ Zero Crossing

รูปที่ 4.5 สัญญาณที่ ขาหม้อแปลง ไอเอฟ ของ วงจรรับขณะรับข้อมูล 1111

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- วัดค่าที่ขา 8 ของไมโครคอนโทรลเลอร์ซึ่งเป็นขาที่รับค่าแรงดันพัลส์จากวงจรภาครับ



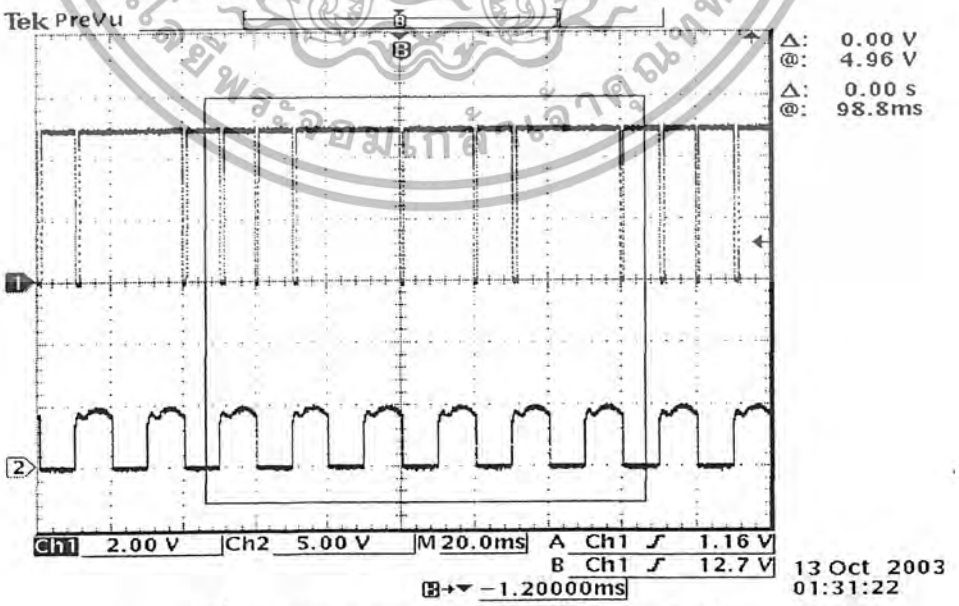
Ch1 คือ ขา 8 ของไมโครคอนโทรลเลอร์ภาครับ

Ch2 คือ แรงดันพัลส์จากวงจรตรวจสอบ Zero Crossing

รูปที่ 4.6 สัญญาณที่ ขา 8 ของไมโครคอนโทรลเลอร์ขณะรับข้อมูล 1111

4.3 ทดลองการรับส่งข้อมูลระหว่างโมดูล

- โปรแกรมไมโครคอนโทรลเลอร์ของโมดูลหลักให้ส่งข้อมูล 1110010111001ซ้ำๆ
- วัดแรงพัลท์เข้าไมโครคอนโทรลเลอร์ของโมดูลปลายทางดูผล



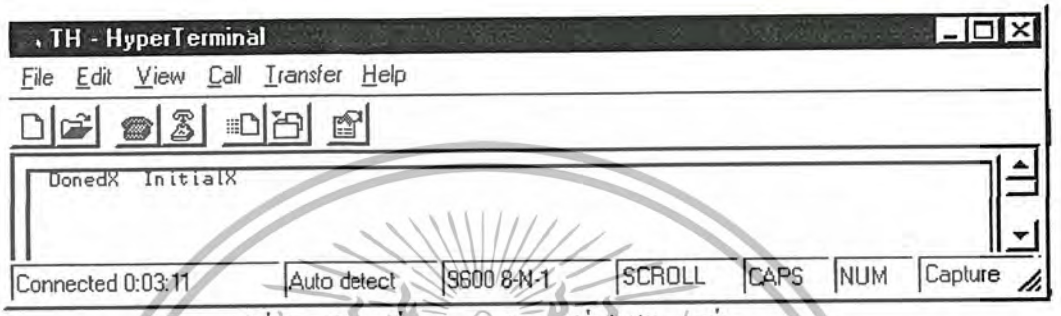
รูปที่ 4.7 ผลการทดลองการรับส่งข้อมูลระหว่างโมดูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.4 ทดลองการรับส่งคำสั่งระหว่างคอมพิวเตอร์และโมดูล

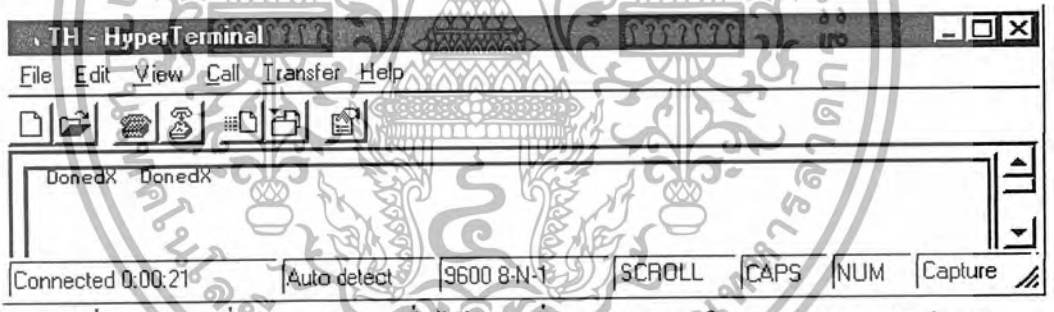
วิธีการทดลอง

- เปิดโปรแกรม Hyper Terminal โดยตั้งค่าความเร็วเป็น 9600 บิตต่อวินาที
- ส่งคำสั่ง InitialX ดูผล

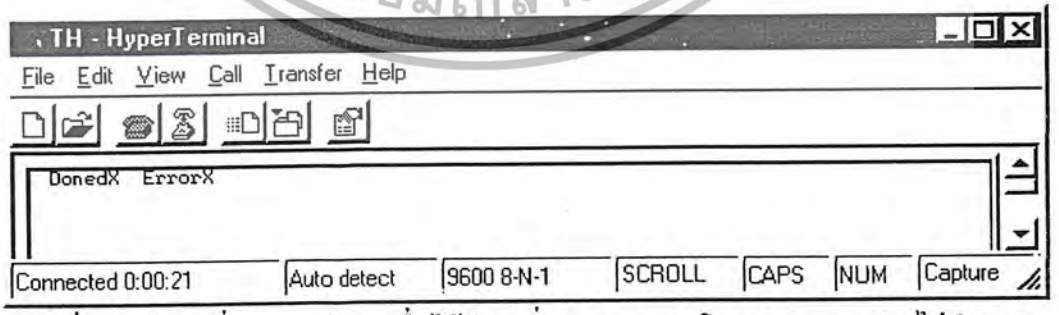


รูปที่ 4.8 ข้อมูลที่ตอบจากโมดูลเมื่อได้รับคำสั่ง InitialX

- ส่งคำสั่ง เปิดปิดอุปกรณ์ ในที่นี้ใช้ O011X คือสั่งเปิดอุปกรณ์หมายเลข 01 ดูผล



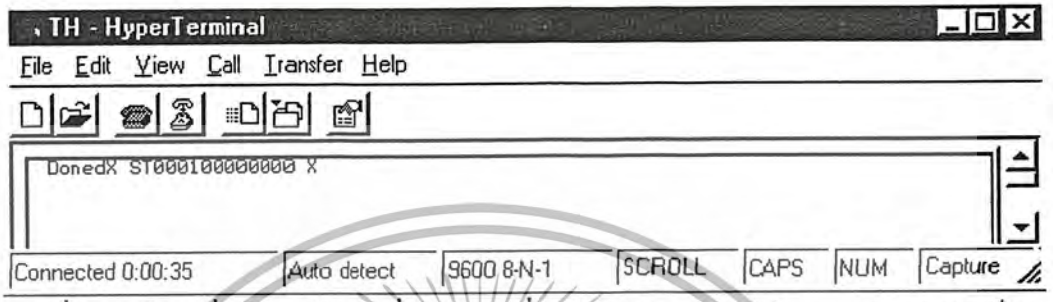
รูปที่ 4.9 ข้อมูลที่ตอบจากโมดูลเมื่อได้รับคำสั่ง O011X และโมดูลหมายเลข 01 ทำงาน



รูปที่ 4.10 ข้อมูลที่ตอบจากโมดูลเมื่อได้รับคำสั่ง O011X และโมดูลหมายเลข 01 ไม่ทำงาน

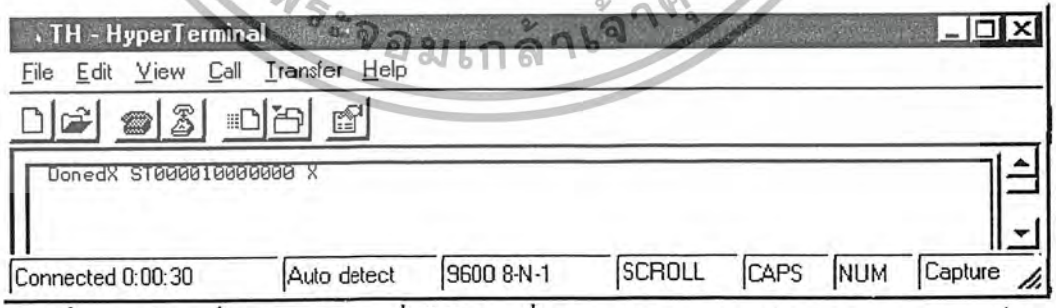
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- ส่งคำสั่ง ขอดูสถานะ คือ StateX ดูผล
- ตั้งโมดูลหลักให้เห็นอุปกรณ์ดังลูกได้ 4 ตัว
- ติดตั้งโมดูลปลายทางหมายเลข 01



รูปที่ 4.11 ข้อมูลที่ตอบจากโมดูลเมื่อได้รับคำสั่ง StateX และสถานะโมดูลหมายเลข 01 เป็นอุปกรณ์ปิด(100)

รูปที่ 4.12 ข้อมูลที่ตอบจากโมดูลเมื่อได้รับคำสั่ง StateX และสถานะโมดูลหมายเลข 01 เป็นอุปกรณ์เปิด(110)



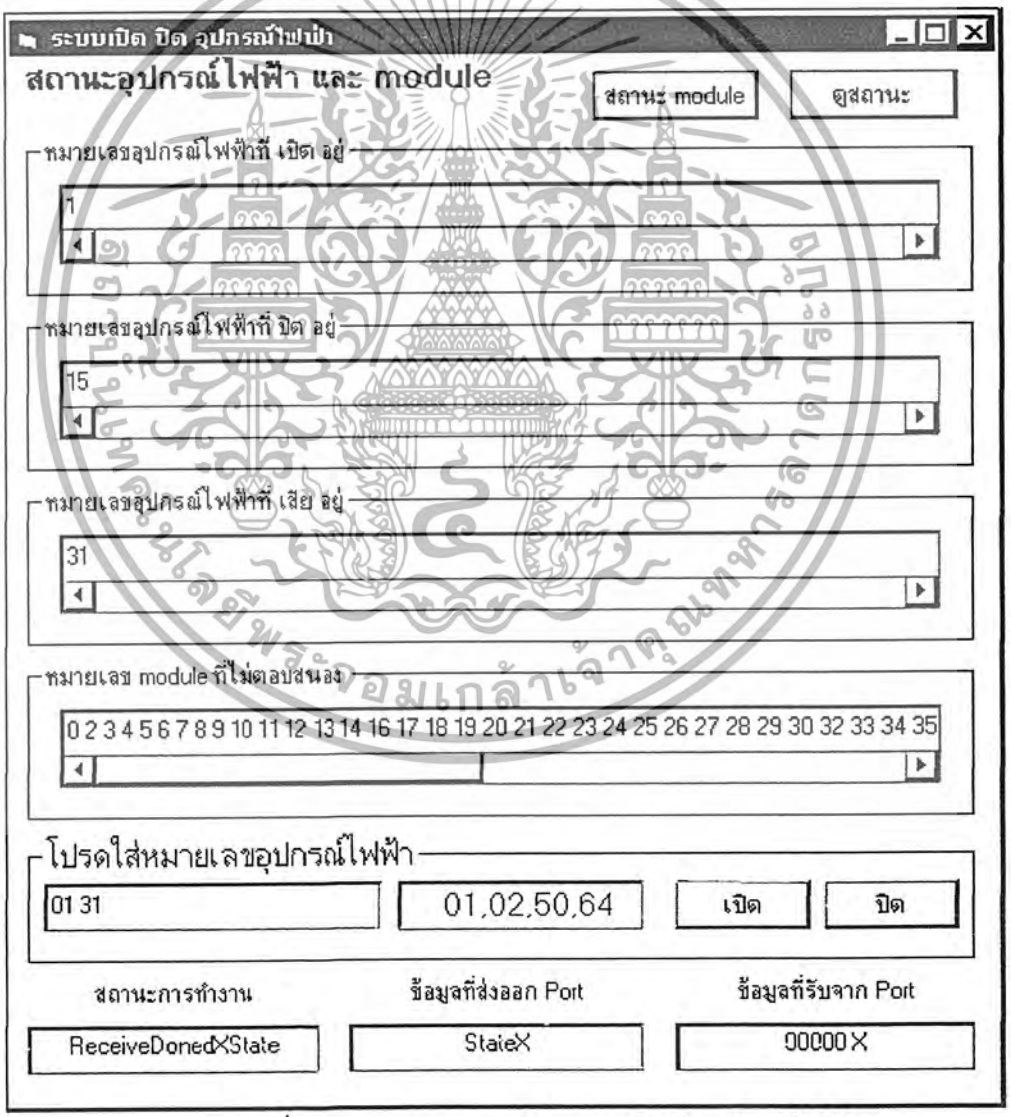
รูปที่ 4.13 ข้อมูลที่ตอบจากโมดูลเมื่อได้รับคำสั่ง StateX และสถานะโมดูลหมายเลข 01 เป็นอุปกรณ์เสีย(010)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.5 ทดลองการทำงานของโปรแกรมเปิดปิดอุปกรณ์ไฟฟ้า

วิธีทดลอง

- เปิดโปรแกรมเปิดปิดอุปกรณ์ไฟฟ้า
- ตั้งโมดูลตัวหลักให้เห็นอุปกรณ์ตัวลูกได้ 62 ตัว
- ตั้งโมดูลปลายทางให้มีหมายเลข 01, 15 และ 31
- ถอดหลอดไฟ ของโมดูลปลายทางหมายเลข 31
- สั่งเปิดอุปกรณ์ไฟฟ้าหมายเลข 01 และ 31
- ดูผล



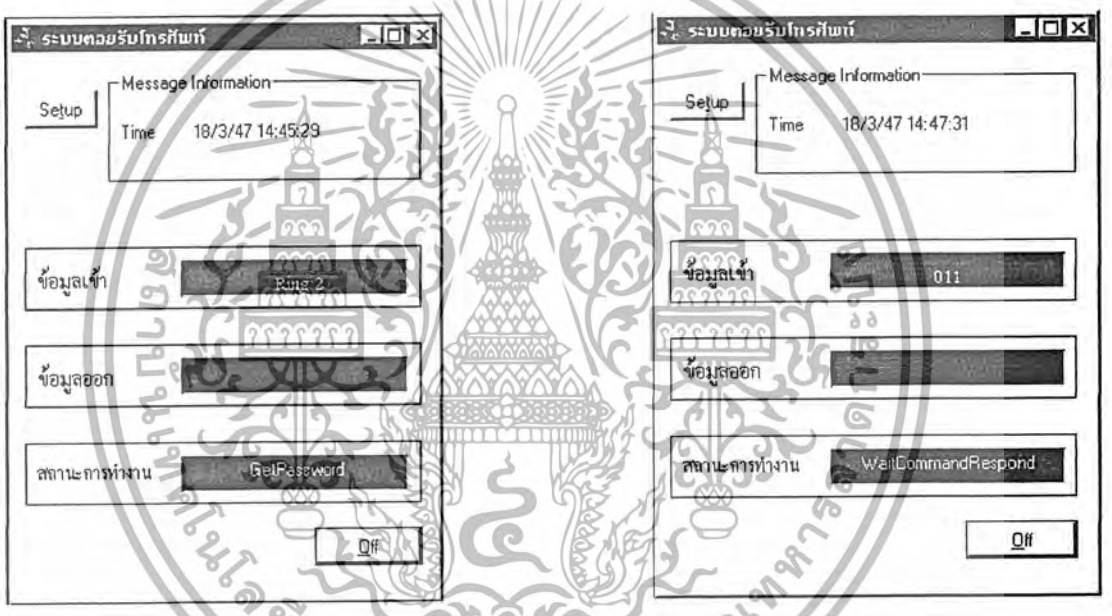
รูปที่ 4.14 ผลการทดลองโปรแกรมเปิดปิดอุปกรณ์ไฟฟ้า

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.6 ทดลองการทำงานของโปรแกรมรับคำสั่งจากโทรศัพท์

วิธีทดลอง

- เปิดโปรแกรมรับคำสั่งจากโทรศัพท์
- ตั้งค่าให้กับระบบ
- ตั่งโมดูลหลักให้เห็นโมดูลปลายทางได้ 4 ที่อยู่
- ติดตั้งโมดูลปลายทางเป็นหมายเลข 01 และ 02
- โทรศัพท์เข้าเพื่อทำการเปิดอุปกรณ์ไฟฟ้าหมายเลข 01 และดูสถานะอุปกรณ์ไฟฟ้า
- ดูผล



รูปที่ 4.15 โปรแกรมรับคำสั่งจากโทรศัพท์ขณะรับสายและรับคำสั่งเปิดอุปกรณ์หมายเลข 01

ผลการทดลอง

- *เมื่อระบบระบบสายมีเสียงตอบรับ* "ยินดีตอนรับสู่ระบบเปิดปิดอุปกรณ์ไฟฟ้า โปรดใส่รหัสผ่าน"
- เมื่อใส่รหัสที่ไม่ถูกต้อง มีเสียงตอบรับ "ขออภัย รหัสผิดพลาด"
- เมื่อใส่รหัสถูกต้อง มีเสียงตอบรับ "ยินดีตอนรับสู่ระบบเปิดปิดอุปกรณ์ไฟฟ้า ต้องการเปิดปิดอุปกรณ์ไฟฟ้ากดหนึ่ง ต้องการดูสถานะอุปกรณ์ไฟฟ้ากดสอง"
- *เมื่อกดหนึ่ง มีเสียงตอบรับ "กดหมายเลขอุปกรณ์ไฟฟ้า ตามด้วยคำสั่งและเครื่องหมายดอกจัน"*

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- เมื่อกดหมายเลข 011* มีเสียงตอบรับ “การส่งข้อมูลสมบูรณ์”

- เมื่อกดสอง มีเสียงตอบรับ “อุปกรณ์ไฟฟ้าที่เปิดอยู่คือหมายเลข หนึ่ง อุปกรณ์ไฟฟ้าที่ปิดอยู่คือหมายเลข สอง โมดูลที่ไม่ตอบสนองคือหมายเลข ศูนย์ สาม”

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 5
สรุปและวิจารณ์ผลการทดลอง

5.1 สรุปและวิจารณ์ผลการทดลอง

5.1.1 การทดลองการทำงานของวงจรส่ง

จากผลการทดลองจะเห็นว่ากราฟแรงดันที่ขา 11 ของไมโครคอนโทรลเลอร์และแรงดันจากขา 3 ของไอซี555เทียบกับแรงดันจากวงจรหาจุด Zero Crossing ที่ได้เป็นไปตามรูปแบบที่ตั้งไว้ ตามรูปที่ 3.12 คือในการส่งข้อมูลที่ เป็น 1 ไมโครคอนโทรลเลอร์จะสั่งให้ค่าที่ขา 11 มีค่าเป็น 0 นาน 1 ms หลังจากจุดที่ แรงดันจากวงจรหา Zero Crossing เปลี่ยนระดับ ส่งผลให้ไอซี555 สร้างสัญญาณความถี่ 98 kHz นาน 1 ms และส่งออกระบบไฟฟ้ากำลัง

แรงดันที่ส่งออกระบบไฟฟ้ากำลังมีขนาดประมาณ 2 Vp-p โดยที่จุดไฟฟ้ากำลังมีเฟส 0 องศา สัญญาณที่ส่งออกจะเริ่มที่ 20 us ก่อนจุด Zero Crossing และที่จุด 180 องศา สัญญาณที่ส่งออกจะเริ่มที่ 300 us ก่อนจุด Zero Crossing

5.1.2 การทดลองการทำงานของวงจรรับ

จากผลการทดลองจะเห็นว่ากราฟแรงดันที่ ได้ก่อนเข้าวงจรขยายและแรงดันพัลท์เข้าไมโครคอนโทรลเลอร์ที่ได้มีรูปแบบเหมือนกับที่ตั้งไว้ตามรูป 3.12 คือจุดที่ข้อมูลมีค่าเป็น 1 แรงดันพัลท์ จะมีค่าเป็น 0 หลังจากจุดที่ แรงดันจากวงจรหา Zero Crossing เปลี่ยนระดับ

5.1.3 การทดลองการส่งข้อมูลระหว่างโมดูล

ข้อมูลที่โมดูลปลายทางรับได้มีค่าตรงกับข้อมูลที่โมดูลต้นทางส่งมาให้

5.1.4 การทดลองการรับส่งข้อมูลระหว่างคอมพิวเตอร์กับโมดูล

การรับส่งข้อมูลเป็นไปตามที่ได้ตั้งไว้

5.1.5 การทดลองทำงานของโปรแกรมรับคำสั่งจากโทรศัพท์

การทำงานของโปรแกรมเป็นไปตามที่ได้ตั้งไว้ทุกประการ

5.2 ความสามารถในการทำงาน

ตัว MASTER มีความสามารถในควบคุมตัว SLAVE ได้ทั้งหมด 64 address ซึ่งน่าจะเพียงพอต่อการใช้ในบ้านจำนวน 1 หลังได้ ในการส่งคำสั่งเปิดหรือปิด 1 คำสั่งจะใช้ทั้งหมด 19 Zero Crossing โดยที่ 1 Zero Crossing ใช้เวลา 0.01 วินาทีดังนั้นคิดเป็นเวลา 0.19 วินาทีต่อ 1 คำสั่ง การส่งและรอรับสถานะจะใช้เวลาขึ้นกับจำนวนของโมดูลปลายทาง โดยจะใช้ไปกับรหัสคำสั่ง (1111)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4 Zero Crossing และโมดูลปลายทางโมดูลละ 3 Zero Crossing เช่น มีโมดูลปลายทาง 10 ที่อยู่จะใช้เวลา 0.04+0.3 วินาที

การสั่งงานสามารถทำได้จากทางโทรศัพท์ทำให้เปิดปิดหรือทราบสถานะอุปกรณ์ไฟฟ้าได้จากระยะไกล

5.3 ปัญหาของระบบ

- มีการส่งข้อมูลที่ช้ามากถ้าคิดในด้านของ Data Communication คือ 50 บิตต่อวินาที
- จะใช้ไม่ได้กับอาคารที่ใช้ไฟฟ้าแบบ 3 เฟส
- การรับทราบสถานะทางโทรศัพท์ถ้ามีอุปกรณ์จำนวนมากจะใช้เวลานานทำให้เสียค่าใช้จ่ายมาก

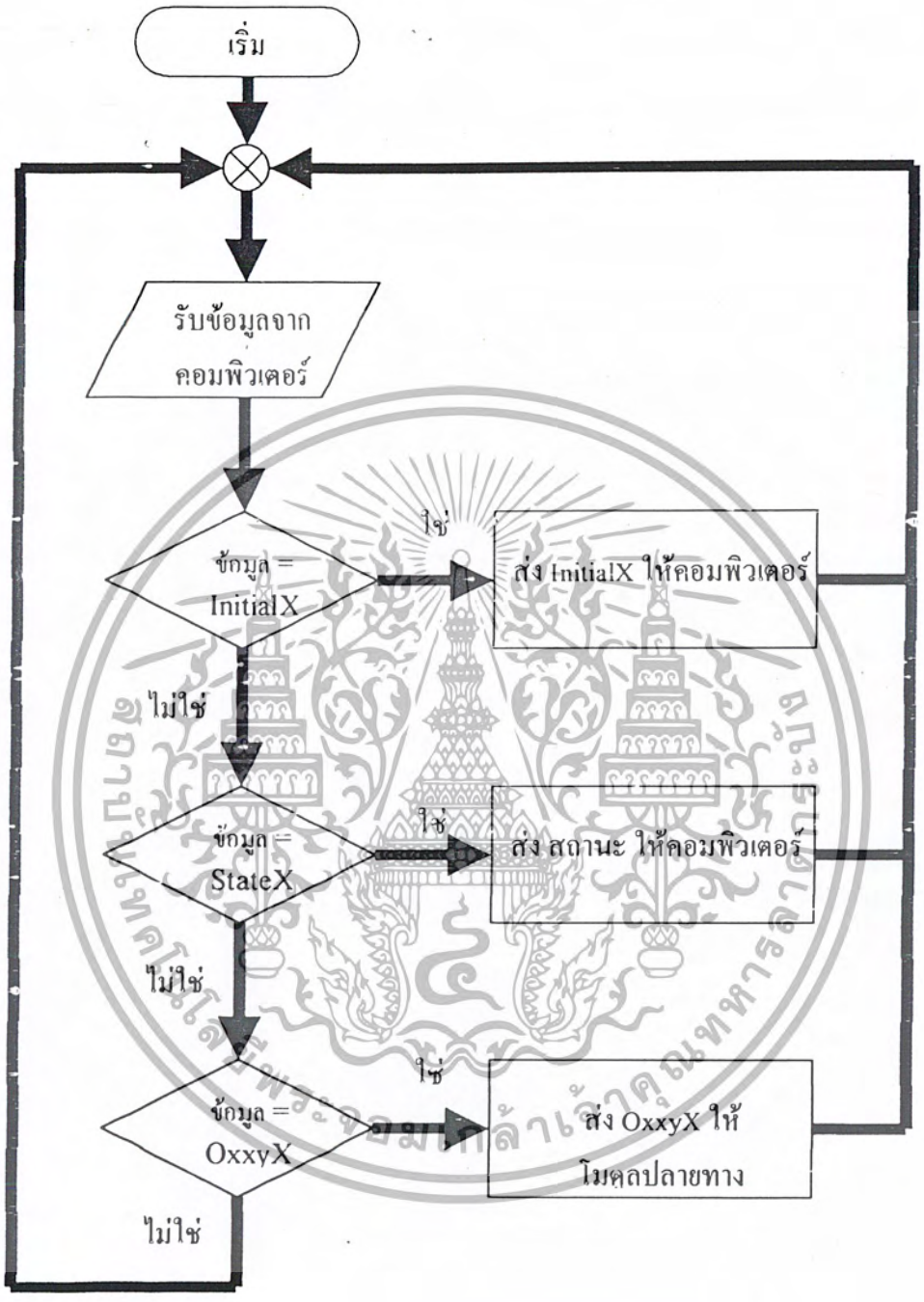
5.4 การปรับปรุง

ปรับปรุงให้มีการส่งสถานะของอุปกรณ์ไฟฟ้าเป็นการส่งทางข้อความผ่านระบบโทรศัพท์เคลื่อนที่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

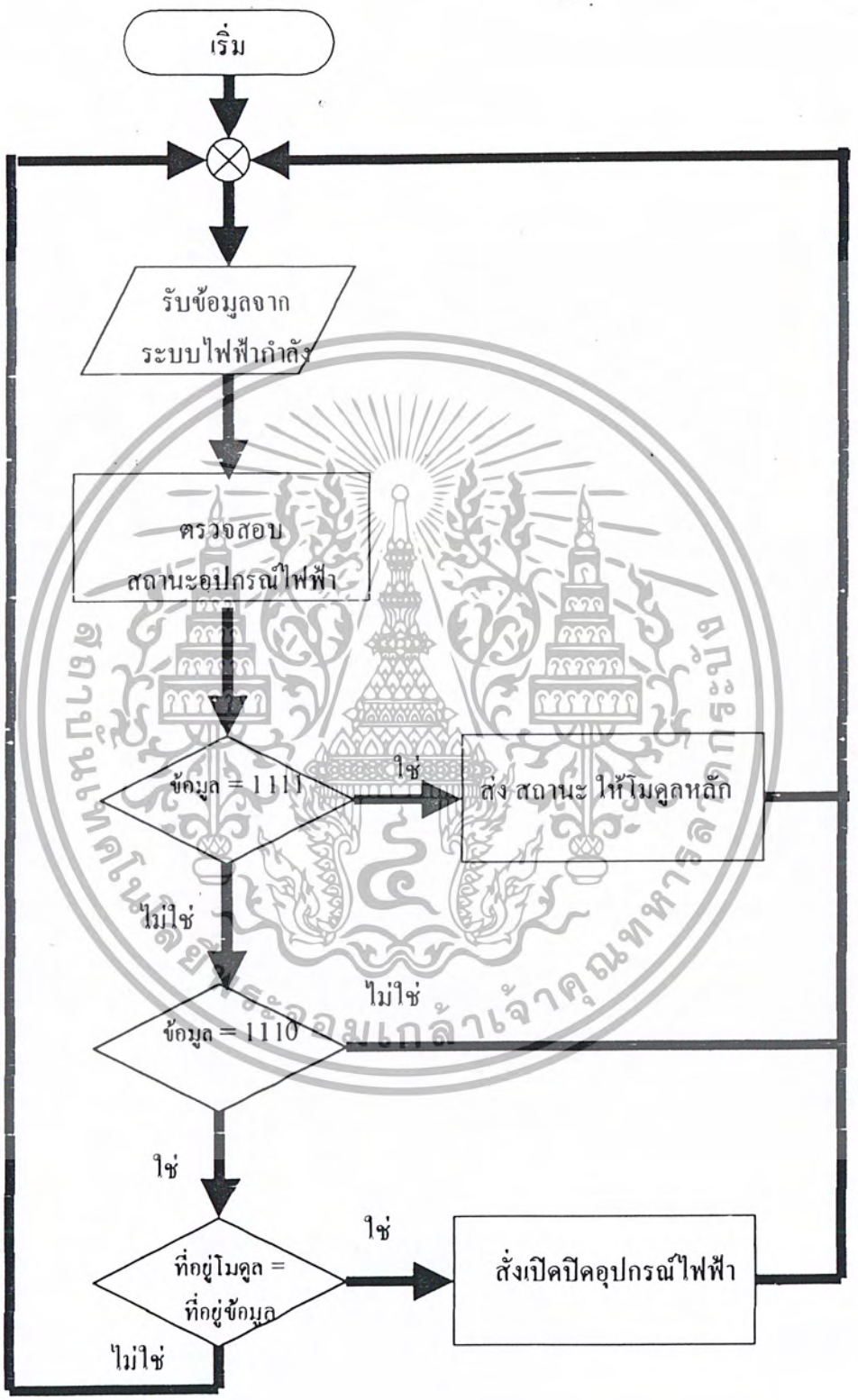
ภาคผนวก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



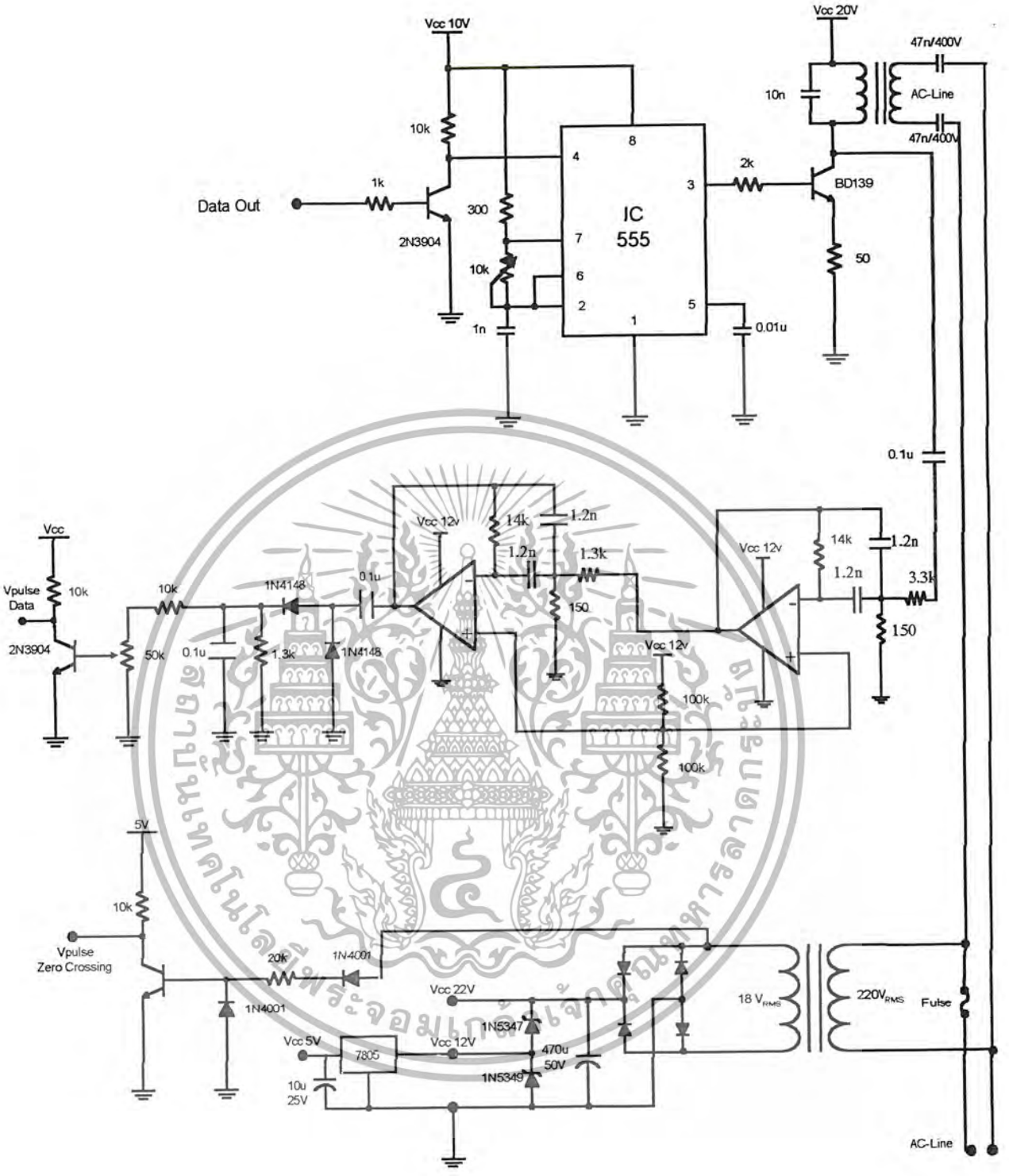
Flow Chart ของ Master Module

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



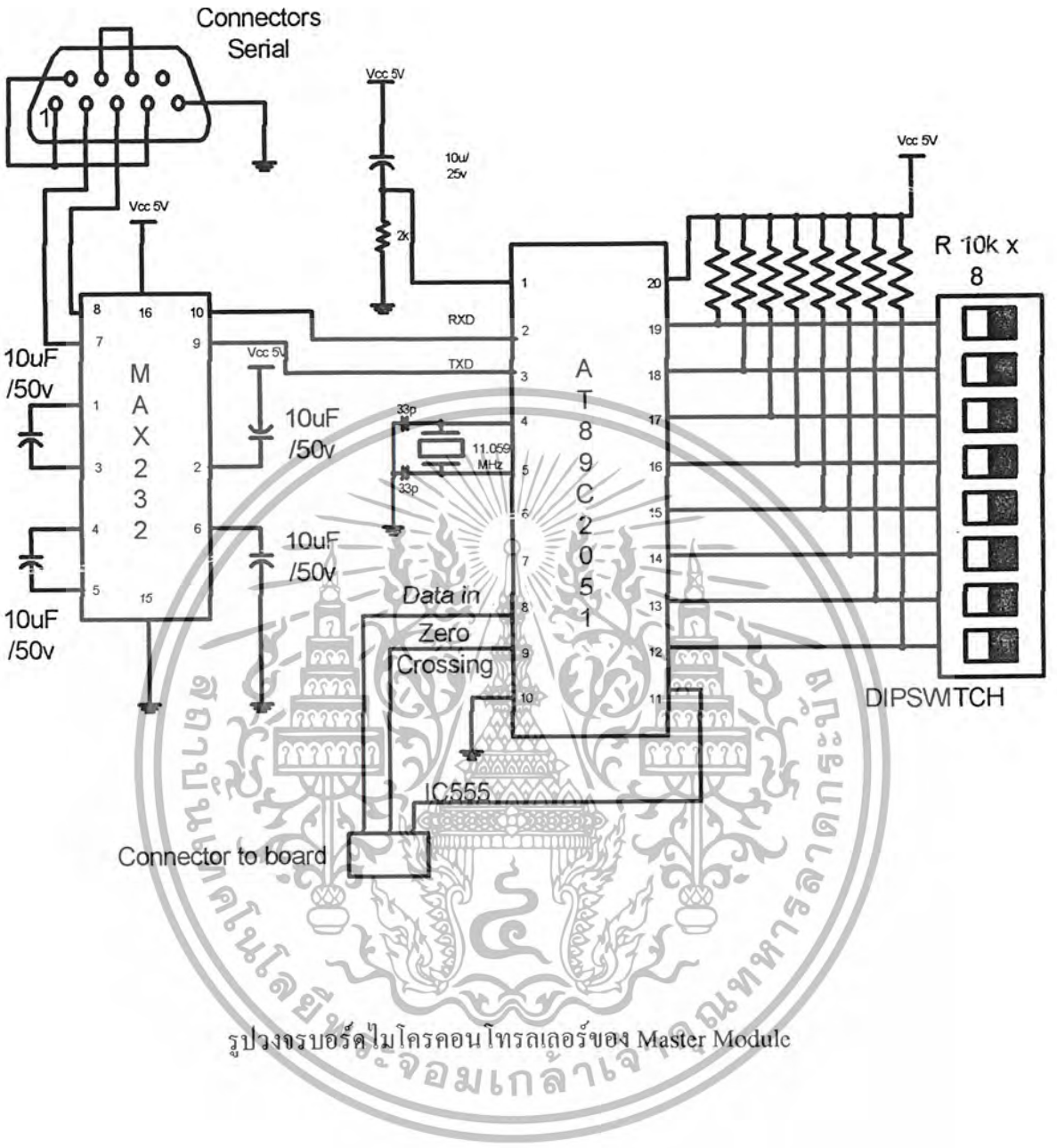
Flow Chart ของ Slave Module

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



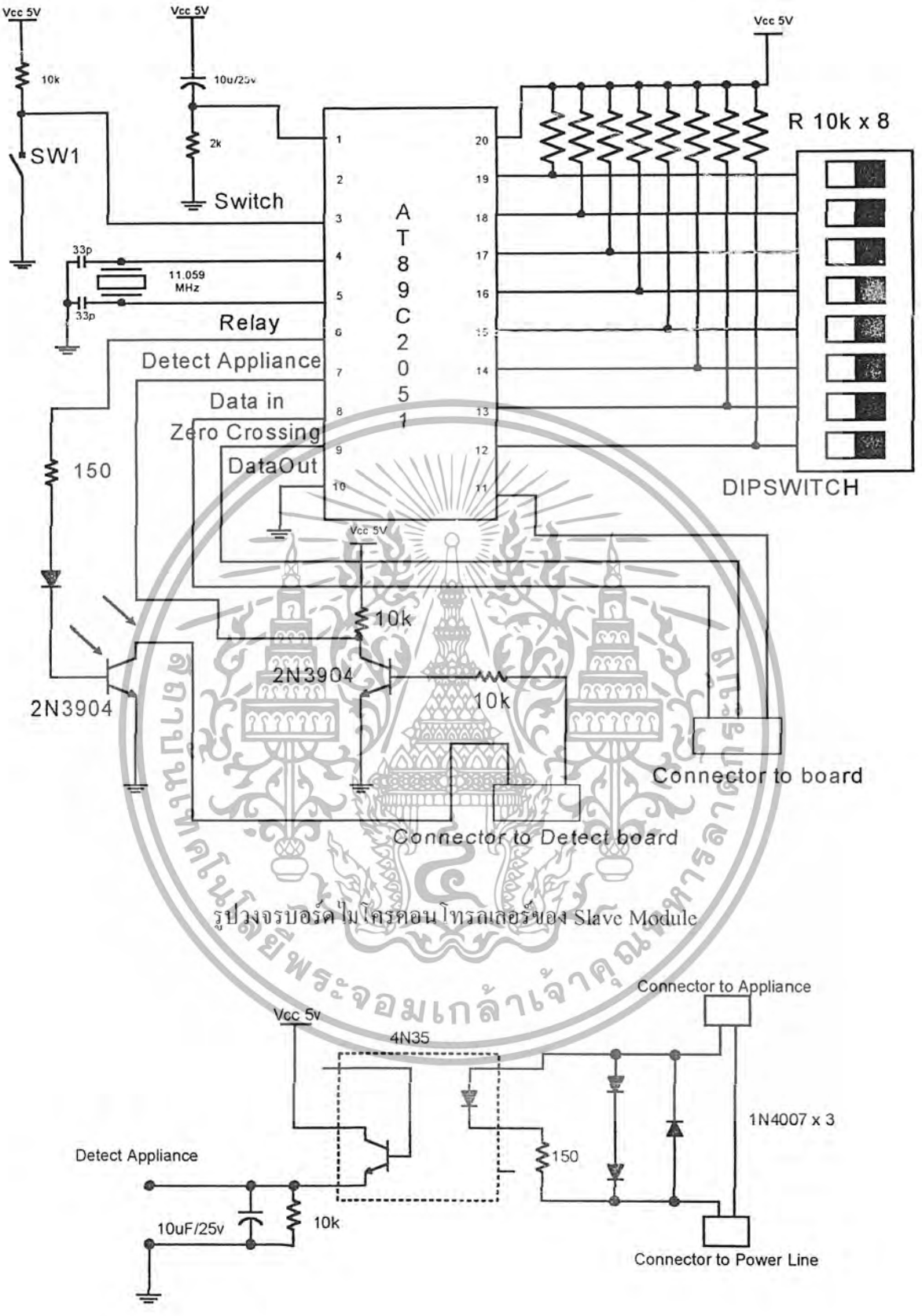
รูปวงจรบอร์ดรับส่งสัญญาณ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปวงจรบอร์ดไมโครคอนโทรลเลอร์ของ Master Module

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปวงจรบอร์ดไมโครคอนโทรเลอร์ของ Slave Module

รูปวงจรบอร์ดตรวจสอบอุปกรณ์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บรรณานุกรม

1. ธีรวัฒน์ ประกอบผล, " การประยุกต์ใช้งาน ไมโครคอนโทรลเลอร์ ", สมาคมส่งสริมเทคโนโลยี(ไทย- ญี่ปุ่น) , 2543
2. " ปฏิบัติการอิเล็กทรอนิกส์3 ", ภาควิชาอิเล็กทรอนิกส์ คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง , 2545

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้