สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

ป้ายรถประจำทางสำหรับผู้พิการทางสายตา

THE INTELLIGENCE BUS STOP FOR BLIND PEOPLE

โดย

นาย ณรงค์ จิรวงศ์บุญรอด

นาย ศักดิ์สิทธิ์ มณีกรณ์

นาย อัจฉริยเทพ ตติยนันทพร

เลขหมู่.....
เลขทะเบียน 62725
วัน,เดือน,ปี 21 ส.ค. 2549

b 11629048
i.....

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต

สาขาวิชาวิศวกรรมโทรคมนาคม

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2548

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ป้ายรถประจำทางสำหรับผู้พิการทางสายตา

THE INTELLIGENCE BUS STOP FOR BLIND PEOPLE

โดย

นาย ณรงค์ จิรวงศ์บุญรอด 45010217

นาย ศักดิ์สิทธิ์ มณีกรณ์ 45010759

นาย อัจฉริยเทพ ตติยนันทพร 45010957

อาจารย์ที่ปรึกษา

ผศ. สุรพล บุญจันทร์

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต

สาขาวิชาวิศวกรรมโทรคมนาคม

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2548

ผ่านการตรวจชิ้นงานแล้ว

(ลงชื่อ).....ผู้ตรวจ

ผ่านการตรวจรูปเล่มแล้ว

(ลงชื่อ).....ผู้ตรวจ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ปริญญานิพนธ์ปีการศึกษา 2548

ภาควิชาวิศวกรรมไทรคมนาคม

คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เรื่อง ป้ายรถประจำทางสำหรับผู้พิการทางสายตา

THE INTELLIGENCE BUS STOP FOR BLIND PEOPLE

ผู้จัดทำ

1: นาย ณรงค์ จิรวงศ์บุญรอด 45010217

2: นาย ศักดิ์สิทธิ์ มณีกรณ์ 45010759

3: นาย อัจฉริยเทพ ศติยนันทพร 45010957

..... อาจารย์ที่ปรึกษา

(ผศ. สุรพล บุญจันทร์)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ป้ายรถประจำทางสำหรับผู้พิการทางสายตา

THE INTELLIGENCE BUS STOP FOR BLIND PEOPLE

โดย นาย ณรงค์ จิรวงศ์บุญรอด 45010217
นาย ศักดิ์สิทธิ์ มณีกรณ์ 45010759
นาย อัจฉริยเทพ ตติยนันทพร 45010957

อาจารย์ที่ปรึกษา ผศ. สุรพล บุญจันทร์

บทคัดย่อ

โครงการนี้เป็นการประดิษฐ์อุปกรณ์เพื่ออำนวยความสะดวกแก่ผู้ที่ใช้บริการรถประจำทาง โดยจัดทำเป็น 2 ส่วน คือ ส่วนของป้ายกับรถ โดยในส่วนของป้ายจะแจ้งว่ารถสายใดกำลังจะมาถึง โดยแสดงผลเป็นตัวเลขและเสียง และในส่วนของรถก็จะแจ้งว่าป้ายที่จะถึงเป็นป้ายใด โดยผู้โดยสารที่อยู่ที่ป้ายสามารถติดต่อไปยังรถได้ว่าต้องการที่จะขึ้นรถสายนั้น โดยการติดต่อระหว่างรถกับป้ายจะใช้ย่านความถี่วิทยุสมัครเล่น (245 MHz) ซึ่งควบคุมโดยไมโครคอนโทรเลอร์

ABSTRACT

In order to provide more convenience to people who usually use public transaction. This project will illustrate some helpful facilities that will encourage those people to use a public bus more easily. The project will mainly have two parts which both focus on the connection between the bus and the bus station. The first part of the project will concentrate on how to make the sign to be easy recognize and use. Some electronic equipment might have to be located on the bus station, such as electric timetable sign. For another part of the project, it will focus on the bus itself. For instance, in every public bus should have some electronic equipment that can connect to the sign in the bus station.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กิตติกรรมประกาศ

ขอขอบพระคุณอาจารย์ สุรพล บุญจันทร์ อาจารย์ที่ปรึกษา ที่ช่วยแนะนำสิ่งต่าง ๆ รวมถึงสถานที่และอุปกรณ์ต่าง ๆ ในการทำการทดลอง ขอขอบคุณชุมนุมวิทยุสมัครเล่น คณะวิทยาศาสตร์ ที่ได้ให้ยืมวิทยุสื่อสารมาใช้ในการทดลอง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ

	หน้า
บทที่ 1 บทนำ	**1**
บทที่ 2 ทฤษฎีและหลักการ	**2**
2.1 ระบบการสื่อสาร	2
2.2 การสื่อสารข้อมูล	2
2.2.1 การรับส่งข้อมูลแบบขนาน	3
2.2.2 การรับส่งข้อมูลแบบอนุกรม	3
2.2.2.1 การส่งข้อมูลแบบซิงโครนัส	4
2.2.2.2 การส่งข้อมูลแบบอะซิงโครนัส	5
2.3 การส่งสัญญาณ	6
2.3.1 การส่งผ่านแบบทิศทางเดียว	6
2.3.2 การส่งผ่านแบบสองทิศทางแต่ต่างเวลากัน	7
2.3.3 การส่งผ่านแบบสองทิศทางที่เวลาเดียวกัน	7
2.4 คุณสมบัติทางเทคนิคที่สำคัญของ AT89Cxx	8
2.4.1 การใช้งานเป็นพอร์ตอินพุท	8
2.4.2 การใช้งานเป็นพอร์ตเอาท์พุท	9
2.4.3 การอ่านค่าลอจิกจากพอร์ท	9
2.4.4 รีจีสเตอร์ที่เกี่ยวข้องในการสื่อสารข้อมูลอนุกรม	9
2.5 การมอดูเลตแบบฟรีเควนซีชิฟคีย์อิ้ง	11
2.6 การมอดูเลทของคลื่น	14
2.6.1 ดัชนีการมอดูเลต	15
2.6.2 ไซด์แบนด์ FM	16
2.6.3 แบนด์วิดท์ของสัญญาณ FM	18
2.7 โพล และ ซีเลคท์	19
2.8 แพ็คเกตเรดิโอ	21
2.8.1 ประวัติความเป็นมาของแพ็คเกตเรดิโอ	21
2.8.2 แพ็คเกตเรดิโอคืออะไร	22
2.8.3 แพ็คเกตโมเด็ม	22
2.9 หลักการของระบบบัส I^2C	22
2.9.1 ความรู้เบื้องต้นเกี่ยวกับ I^2C	22
2.9.2 คุณสมบัติโดยทั่วไปของบัส I^2C	22
2.9.3 หลักการของบัส I^2C	23
2.9.4 สภาวะที่เกิดขึ้นบนบัส I^2C	24

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ(ต่อ)

	หน้า
บทที่ 3 การคำนวณและการสร้าง	**25**
3.1 ส่วนประกอบของโครงงาน	25
3.2 หลักการทำงาน	26
3.3 การแปลงสัญญาณ FSK	27
3.3.1 การออกแบบวงจรเข้ารหัส (FSK Modulation)	27
3.3.2 การออกแบบวงจรถอดรหัส (FSK Demodulation)	28
3.4 การเชื่อมต่ออุปกรณ์ภายนอกต่าง ๆ กับไมโครคอนโทรเลอร์	30
3.5 วงจรขยายเสียง	32
3.6 วงจรจ่ายไฟ	32
3.7 การออกแบบโปรแกรม	33
3.7.1 การทำงานในส่วนของเมนูหลักของป้ายและรถประจำทาง	33
3.7.2 โปรแกรมย่อยในการตั้งค่าให้กับป้ายรถประจำทาง	34
3.7.3 โปรแกรมย่อยในการตั้งค่าให้กับรถประจำทาง	35
3.7.4 โปรแกรมย่อยในการตรวจสอบหมายเลขป้ายรถประจำทางที่ได้บันทึกไว้	36
3.7.5 โปรแกรมย่อยในการตรวจสอบหมายเลขที่รถประจำทางที่ได้บันทึกไว้	37
3.7.6 โปรแกรมหลักการสื่อสารของป้ายรถประจำทาง	38
3.7.7 โปรแกรมย่อยบริการอินเทอร์รัปป์เมื่อมีการตอบกลับจากรถของป้ายรถประจำทาง	39
3.7.8 โปรแกรมการติดต่อกับรถเพื่อเรียกรถ	40
3.7.9 โปรแกรมหลักการสื่อสารของรถประจำทาง	41
3.7.10 โปรแกรมการสื่อสารแบบ Manual ที่รถประจำทาง	42
บทที่ 4 ผลการทดลอง	**43**
4.1 การทดลองวงจรเข้ารหัสและถอดรหัส FSK	43
4.2 การทดลองวัดสัญญาณที่ได้รับเมื่อใช้เครื่องรับ – ส่ง (วิทยุสื่อสาร)	45
4.3 การทดลองวัดสัญญาณที่ได้จากพอร์ตสื่อสารอนุกรม	46
4.4 การทดลองเข้ารหัสสัญญาณที่ได้จากพอร์ตสื่อสารอนุกรม	46
4.5 การทดลองวัดสัญญาณที่ได้รับจากเครื่องรับวิทยุ	47
4.6 การทดลองถอดรหัสสัญญาณที่ได้รับจากเครื่องรับวิทยุ	48
4.7 การทดลองวงจรขยายกำลัง	49
4.8 การทดลองวัดระยะทางที่สามารถติดต่อระหว่างป้ายกับรถ	49
4.9 ขั้นตอนการทำงานในส่วนของป้ายรถประจำทาง	50
4.10 ขั้นตอนการทำงานในส่วนของรถประจำทาง	52

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ(ต่อ)

	หน้า
บทที่ 5 สรุปและวิจารณ์ผลการทดลอง	**55**
ภาคผนวก	
หนังสืออ้างอิง	

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป

	หน้า
รูปที่ 1.1 แสดงการทำงานของโครงงาน	1
รูปที่ 2.1 ระบบการสื่อสาร	2
รูปที่ 2.2 แสดงการส่งข้อมูลผ่านโดยใช้รหัส	2
รูปที่ 2.3 การส่งข้อมูลแบบขนาน	3
รูปที่ 2.4 การสื่อสารข้อมูลแบบอนุกรม	4
รูปที่ 2.5 แสดงการส่งข้อมูลแบบซิงโครนัส	4
รูปที่ 2.6 แสดงการส่งข้อมูลแบบซิงโครนัส	5
รูปที่ 2.7 แสดงการส่งข้อมูลแบบอะซิงโครนัส	6
รูปที่ 2.8 แสดงรูปแบบข้อมูลแบบอะซิงโครนัส	6
รูปที่ 2.9 การส่งผ่านแบบทิศทางเดียว	7
รูปที่ 2.10 การส่งผ่านแบบสองทิศทางแต่ต่างเวลากัน	7
รูปที่ 2.11 การส่งผ่านแบบสองทิศทางที่เวลาเดียวกัน	8
รูปที่ 2.12 ลักษณะของสัญญาณเอฟเอสเค	12
รูปที่ 2.13 อัตราการเปลี่ยนแปลงของสัญญาณอินพุต	12
รูปที่ 2.14 การเบี่ยงเบนความถี่ของเอฟเอสเค	13
รูปที่ 2.15 แสดงสัญญาณความถี่เสียงกับสัญญาณที่ถูกมอดูเลตแบบ FM และ PM	15
รูปที่ 2.16 แสดงการมอดูเลตคลื่น FM ใน Frequency-domain	15
รูปที่ 2.17 กราฟแสดงแอมพลิจูดของพาหะและไซด์แบนด์ในระบบ FM	17
รูปที่ 2.18 รูปคลื่น FM ในเชิงความถี่ที่ค่าดัชนีการมอดูเลตเท่ากับ 0,0.5,1.1.5,2,3	17
รูปที่ 2.19 แสดงการทำงานของโพลและซีเลคท์	21
รูปที่ 3.1 แสดง Block diagram ในส่วนของป้ายรถประจำทาง	25
รูปที่ 3.2 แสดง Block diagram ในส่วนของรถประจำทาง	25
รูปที่ 3.3 วงจรเข้ารหัส (FSK Modulation)	27
รูปที่ 3.4 วงจรถอดรหัส (FSK Modulation)	29
รูปที่ 3.5 การเชื่อมต่ออุปกรณ์ภายนอกต่าง ๆ เข้ากับไมโครคอนโทรเลอร์	30
รูปที่ 3.6 วงจรขยายกำลัง	32
รูปที่ 3.7 วงจรจ่ายไฟ	32
รูปที่ 3.8 Flowchart แสดงการทำงานของเมนูหลัก	33
รูปที่ 3.9 Flowchart แสดงโปรแกรมย่อยในการตั้งค่าให้กับป้ายรถประจำทาง	34
รูปที่ 3.10 Flowchart แสดงโปรแกรมย่อยในการตั้งค่าให้กับรถประจำทาง	35
รูปที่ 3.11 Flowchart แสดงโปรแกรมย่อยในการตรวจสอบหมายเลขที่ป้ายรถประจำทาง	36
รูปที่ 3.12 Flowchart แสดงโปรแกรมย่อยในการตรวจสอบหมายเลขที่รถประจำทาง	37

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป (ต่อ)

	หน้า
รูปที่ 3.13 Flowchart แสดงโปรแกรมหลักการสื่อสารของป้ายรถประจำทาง	38
รูปที่ 3.14 Flowchart แสดงโปรแกรมย่อยบริการอินเทอร์รัปป์ของป้ายรถประจำทาง	39
รูปที่ 3.15 Flowchart แสดงโปรแกรมการติดต่อกับรถเพื่อเรียกรถ	40
รูปที่ 3.16 Flowchart โปรแกรมหลักการสื่อสารของรถประจำทาง	41
รูปที่ 3.17 Flowchart โปรแกรมการสื่อสารแบบ Manual ที่รถประจำทาง	42
รูปที่ 4.1 Block diagram ของการส่งข้อมูล	43
รูปที่ 4.2 Block diagram ของการรับข้อมูล	43
รูปที่ 4.3 วงจรเข้ารหัส (FSK Modulation)	43
รูปที่ 4.4 วงจรถอดรหัส (FSK Modulation)	44
รูปที่ 4.5 แสดงสัญญาณ output ของวงจรเข้ารหัส FSK เทียบกับสัญญาณ input	44
รูปที่ 4.6 แสดงสัญญาณ output ที่ได้จากวงจรถอดรหัส FSK เทียบกับ input	45
รูปที่ 4.7 แสดงสัญญาณ output จากเครื่องรับเทียบกับ input ของเครื่องส่ง	45
รูปที่ 4.8 แสดงสัญญาณ output ที่ได้จากพอร์ตสื่อสารอนุกรม เมื่อทำการตรวจสอบรถหมายเลข 45 ที่วิ่งในทิศทางไปจากป้ายหมายเลข 36	46
รูปที่ 4.9 แสดงสัญญาณ output ที่ได้จากพอร์ตสื่อสารอนุกรม เมื่อทำการตรวจสอบรถหมายเลข 12 ที่วิ่งในทิศทางไปจากป้ายหมายเลข 36	46
รูปที่ 4.10 แสดงสัญญาณ input เทียบกับ output ของวงจรเข้ารหัส เมื่อป้อนสัญญาณจากพอร์ตอนุกรม	47
รูปที่ 4.11 แสดงสัญญาณ input ของเครื่องส่งเทียบกับ output ของเครื่องรับวิทยุ	47
รูปที่ 4.12 แสดงสัญญาณ input เทียบกับ output ของวงจรถอดรหัส	48
รูปที่ 4.13 แสดงสัญญาณ input จากพอร์ตอนุกรม เทียบกับสัญญาณ output ของวงจรถอดรหัส	48
รูปที่ 4.14 วงจรขยายกำลัง	49
รูปที่ 4.15 แสดงสัญญาณ output ของวงจรขยายสัญญาณเทียบกับ input	49
รูปที่ 4.16 หน้าจอแสดงเมนูการทำงานต่าง ๆ ที่สามารถเลือกได้	50
รูปที่ 4.17 หน้าจอแสดงให้เลือกว่าป้ายนั้นอยู่ในเส้นทางใด	50
รูปที่ 4.18 หน้าจอแสดงข้อความให้บันทึกหมายเลขป้าย	50
รูปที่ 4.19 หน้าจอแสดงข้อความให้บันทึกหมายเลขรถที่ผ่านป้าย	50
รูปที่ 4.20 หน้าจอแสดงข้อความให้เลือกว่าจะบันทึกข้อมูลต่อหรือไม่	51
รูปที่ 4.21 หน้าจอแสดงข้อมูลที่บันทึกไว้ว่าอยู่ในเส้นทางใด	51
รูปที่ 4.22 หน้าจอแสดงข้อมูลที่บันทึกไว้ว่าป้ายหมายเลขใด	51
รูปที่ 4.23 หน้าจอแสดงข้อมูลที่ได้บันทึกไว้ว่ามีรถสายใดผ่าน	51
รูปที่ 4.24 รูปแสดงการแสดงผลหมายเลขรถบนบอร์ด LED	52
รูปที่ 4.25 หน้าจอแสดงข้อความเมื่อเข้าสู่โปรแกรมการทำงานหลัก	52

เอกสารนี้ใช้เพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป (ต่อ)

		หน้า
รูปที่ 4.26	หน้าจอแสดงให้เลือกว่ารถนั้นวิ่งอยู่ในเส้นทางใด	52
รูปที่ 4.27	หน้าจอแสดงข้อความให้เลือกว่าจะบันทึกข้อมูลต่อหรือไม่	53
รูปที่ 4.28	หน้าจอแสดงข้อความให้บันทึกหมายเลขรถ	53
รูปที่ 4.29	หน้าจอแสดงข้อความให้บันทึกหมายเลข ID ประจำรถ	53
รูปที่ 4.30	หน้าจอแสดงข้อมูลของหมายเลขและ ID ของรถที่ได้บันทึกไว้	53
รูปที่ 4.31	หน้าจอแสดงข้อความเมื่อเข้าสู่โปรแกรมการทำงานหลักของรถ	54

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญตาราง

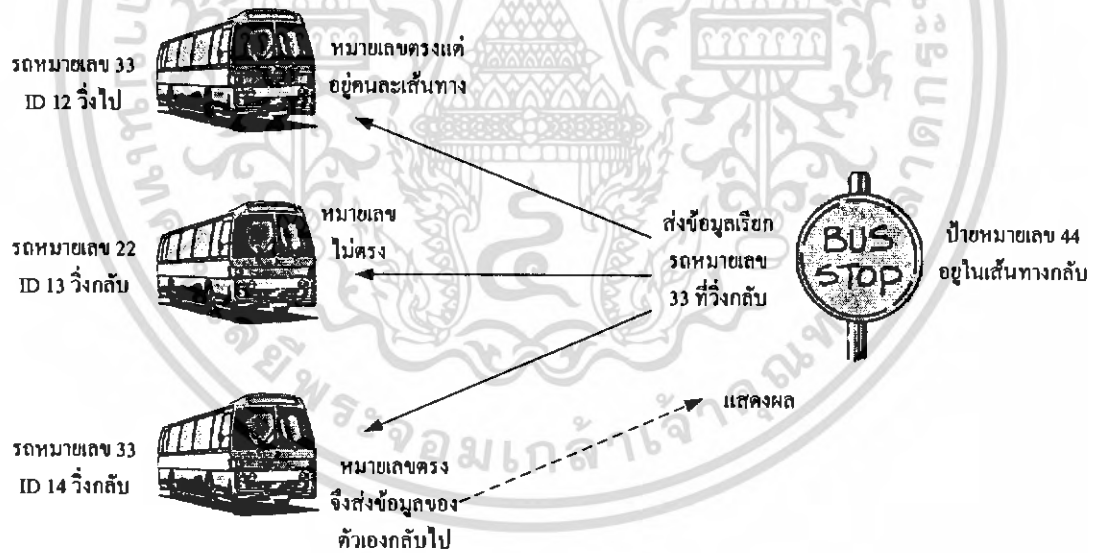
	หน้า
ตารางที่ 2.1 แสดงการกระจายคลื่นพาหะและไซด์แบนด์ที่ดัชนีการมอดูเลตค่าต่างๆ	18
ตารางที่ 3.1 ตารางแสดงคำพูดที่บันทึกในแต่ละแอดเดรส	31

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 1
บทนำ

ในปัจจุบันการเดินทางเข้ามามีความสำคัญกับชีวิตประจำวันของเราทุกคนเป็นอย่างมาก โดยเฉพาะในปัจจุบันซึ่งราคาน้ำมันมีราคาแพงมากขึ้น ดังนั้นการเดินทางโดยรถประจำทางจึงเข้ามามีความสำคัญกับเราทุกคนมากขึ้น ซึ่งรวมไปถึงผู้ที่พิการด้วย เนื่องจากว่าการเดินทางด้วยรถประจำทางเป็นปัญหามากสำหรับผู้พิการทางสายตาโดยเค้าจะไม่รู้เลยว่ารถประจำทางสายที่มาถึงนั้นเป็นสายอะไร หรือถ้าอยู่บนรถก็จะไม่รู้ว่าจะถึงป้ายที่จะลงแล้วหรือยัง ดังนั้นจึงมีแนวความคิดที่จะจัดทำป้ายรถประจำทางที่สามารถใช้ได้ทั้งผู้พิการทางสายตาและบุคคลทั่วไป ซึ่งนอกจากการอำนวยความสะดวกแก่ผู้พิการทางสายตาแล้วยังเป็นการเพิ่มความสะดวกแก่บุคคลทั่วไปอีกด้วย เพื่อให้ประชาชนหันมาใช้การเดินทางโดยรถประจำทางกันมากขึ้น

โดยหลักการทำงานโดยทั่วไปของโครงงานแสดงดังรูปที่ 1.1 คือที่ป้ายรถประจำทางจะทำการบันทึกหมายเลขของรถที่วิ่งผ่านป้ายนั้นไว้ และที่รถก็จะบันทึกหมายเลขและ ID ของรถเอาไว้ โดยที่ป้ายจะทำการส่งสัญญาณเรียกรถแต่ละคันวนไปเรื่อย ๆ และเมื่อมีรถวิ่งเข้ามาในรัศมีของเครื่องส่งก็จะรับข้อมูลนั้นเข้ามาแล้วตรวจสอบดูว่าตรงกับของตัวเองหรือไม่ถ้าตรงก็จะทำการส่งข้อมูลต่าง ๆ ของรถกลับไปยังป้ายเพื่อทำการแสดงผล



รูปที่ 1.1 แสดงการทำงานของโครงงาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 2
ทฤษฎี

2.1 ระบบการสื่อสาร (COMMUNICATION SYSTEM)

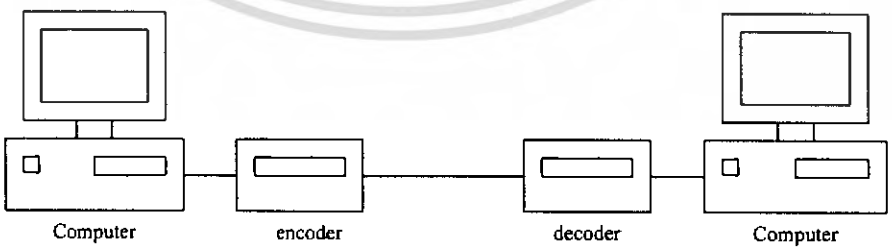
ระบบการสื่อสารมีความหมายกว้างขวางการส่งข่าวสารทางสายก็เป็นชนิดหนึ่งของระบบสื่อสาร โดยพื้นฐานแล้วระบบการสื่อสารจะประกอบด้วยส่วนใหญ่ ๆ 3 ส่วนคือ ตัวส่งข่าวสาร(Transmitter) , ตัวกลางในการส่งข่าวสาร (Medium) และตัวรับข่าวสาร (Receiver) แต่ละส่วนจะมีความสัมพันธ์กัน ดังรูปที่ 2.1

รูปที่ 2.1 ระบบการสื่อสาร

2.2 การสื่อสารข้อมูล

ในการส่งข้อมูลขนาดของข้อมูลหนึ่งตัวอักษรนั้นจะขึ้นอยู่กับอุปกรณ์การสื่อสารที่ใช้ ซึ่งจะมีความยาวอยู่ระหว่าง 7-8 บิท ตัวอักษรนั้นเกิดจากการกำหนดความหมายให้กับกลุ่มของตัวเลขฐานสอง ซึ่งจะมีการแปลงออกมาเป็นอักขระ , ตัวเลข หรือเครื่องหมายวรรคตอนอย่างใดก็ได้ หรือไม่เช่นนั้นก็อาจจะเป็นตัวกำหนดหน้าที่ควบคุมการทำงานของอุปกรณ์ เช่นอาจเป็นคำสั่งให้เครื่องพิมพ์เลื่อนบรรทัดหรือขึ้นหน้าใหม่ ชุดของกลุ่มของเลขฐานสองที่มีการกำหนดความหมายต่าง ๆ มีรหัสจำนวนมากมายที่ใช้กันอยู่ในปัจจุบันในอุปกรณ์การสื่อสารซึ่งจะถูกออกแบบมาให้ใช้ได้เฉพาะกับรหัสชนิดต่าง ๆ ขึ้นอยู่กับวัตถุประสงค์ของผู้ออกแบบและใช้งาน

ประเด็นที่เป็นการสื่อสารระหว่างเครื่องคอมพิวเตอร์จะไม่มีความสามารถเข้าใจถึงความหมายของตัวหนังสือได้ จึงต้องมีการแปลงความหมายให้เป็นแบบที่สามารถตีความได้คือ ในสถานะของเลขฐานสอง ดังนั้นจึงต้องมีอุปกรณ์ทำหน้าที่เข้ารหัส (Encoder) และถอดรหัส (Decoder) มาใช้ในการรับส่งข้อมูลระหว่างเครื่องจักรด้วยกัน แสดงดังรูปที่ 2.2



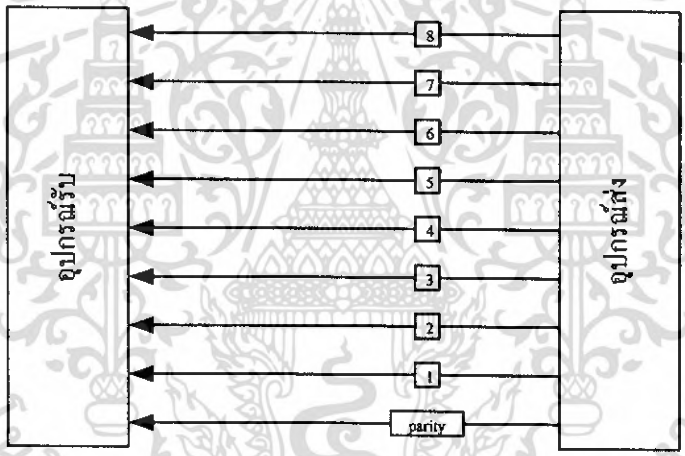
รูปที่ 2.2 แสดงการส่งข้อมูลผ่านโดยใช้รหัส

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.2.1 การรับส่งข้อมูลแบบขนาน (Parallel Transmission)

ลักษณะของการส่งข้อมูลแบบขนาน ทำได้โดยการส่งข้อมูลออกมาทีละ 1 ไบต์ คือ 8 บิท จากอุปกรณ์ส่งไปยังอุปกรณ์รับตัวกลางระหว่าง 2 เครื่องจะต้องมีช่องทางให้ข้อมูลเดินทาง อย่างน้อย 8 ช่องทาง โดยมากจะเป็นสายขนานให้กระแสไฟฟ้าวิ่งมากกว่าจะเป็นตัวกลางชนิดอื่น เนื่องจากมีสัญญาณสูญหายไปกับความต้านทานของสาย ระยะทางระหว่าง 2 เครื่องไม่ควรเกิน 100 ฟุต ปัญหาที่เกิดขึ้นหากระยะทางของสายมากกว่านี้ก็คือ ระดับของกราวน์ในทางไฟฟ้าที่จุดรับผิดไปจากจุดส่ง ทำให้เกิดการผิดพลาดในการรับสัญญาณลอจิกทางฝ่ายรับ

นอกจากสายที่เป็นทางเดินของข้อมูลแล้วอาจจะมีทางเดินของสัญญาณควบคุมอื่น ๆ อีกเป็นต้นว่า บิทที่บอกพาริตี้ของสัญญาณ เพื่อเป็นการตรวจสอบความผิดพลาดของการรับสัญญาณที่ปลายทาง หรือสายที่ควบคุมการโต้ตอบ (Hand shake) จะเห็นได้ว่าการส่งแบบขนานส่วนมากจะทำในระยะใกล้ เนื่องจากจะต้องมีช่องทางเดินของสัญญาณมากกว่า 8 สายและอุปกรณ์ที่ติดต่อแบบขนานกับคอมพิวเตอร์ก็เห็นจะได้แก่เครื่องพิมพ์ดังกล่าวมาแล้ว

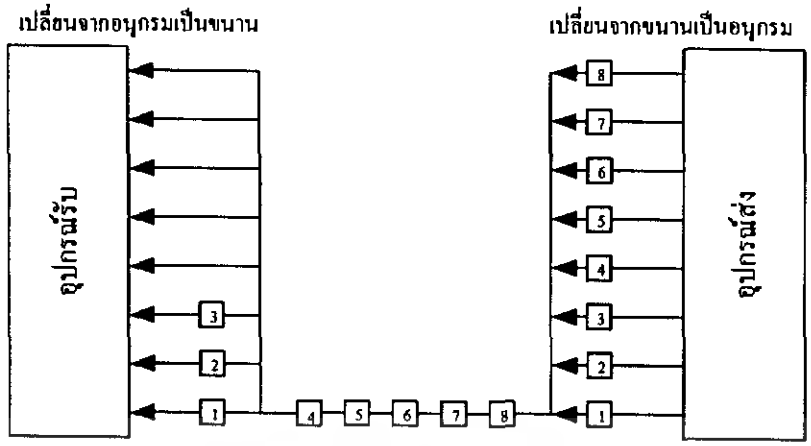


รูปที่ 2.3 การส่งข้อมูลแบบขนาน

2.2.2 การรับส่งข้อมูลแบบอนุกรม (Serial Transmission)

การส่งข้อมูลแบบอนุกรมข้อมูลจะถูกส่งออกมาทีละบิทระหว่างจุดส่งและจุดรับ จะเห็นว่าการส่งข้อมูลแบบนี้จะช้ากว่าแบบขนาน เพราะตัวกลางสื่อสารต้องการเพียงช่องเดียวหรือสายเพียงคู่เดียว ค่าใช้จ่ายในสื่อกลางจะต้องถูกกว่าแบบขนานอย่างแน่นอนสำหรับการส่งระยะทางไกล ๆ โดยเฉพาะเมื่อเรามีระบบสื่อสารทางโทรศัพท์ไว้ใช้งานอยู่แล้วย่อมจะเป็นการประหยัดกว่าที่จะทำการติดต่อสื่อสารทีละ 8 ช่อง แบบขนาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



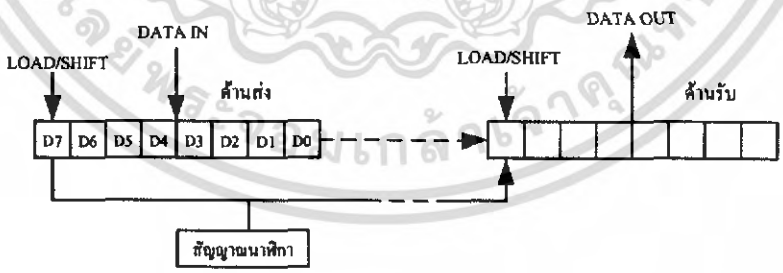
รูปที่ 2.4 การสื่อสารข้อมูลแบบอนุกรม

รูปที่ 2.4 แสดงให้เห็นถึงการส่งข้อมูลแบบอนุกรม ข้อมูลจากจุดส่งจะถูกเปลี่ยนให้เป็นอนุกรมเสียก่อนแล้วค่อยทยอยส่งออกทีละบิทไปยังจุดรับ ณ จุดที่รับจะต้องมีกลไกในการเปลี่ยนข้อมูลที่ส่งมาทีละบิทให้เป็นสัญญาณแบบขนานซึ่งลงตัวพอดี นั่นคือบิท "1" ลงที่บัสข้อมูลเส้นที่ 1 พอดี การที่จะทำให้การแปลงสัญญาณจากอนุกรมทีละบิทให้ลงพอดีนั้นจำเป็นที่จะต้องมีลักษณะการส่งที่เหมาะสม เพื่อป้องกันการผิดพลาดในการรับ ลักษณะการส่งที่เหมาะสมแบ่งเป็น 2 แบบคือ

- การส่งข้อมูลแบบซิงโครนัส
- การส่งข้อมูลแบบอะซิงโครนัส

2.2.2.1 การส่งข้อมูลแบบซิงโครนัส (Synchronous Transmission)

การส่งข้อมูลแบบซิงโครนัสหรือเรียกอีกอย่างหนึ่งว่าการส่งข้อมูลแบบสัมพันธ์ หมายถึง การที่ด้านรับอ่านข้อมูลเข้ามาในจังหวะเดียวกับด้านส่ง โดยใช้สัญญาณนาฬิกาเป็นตัวกำหนดจังหวะการทำงานของรีจีสเตอร์ทั้งสองให้ทำงานสัมพันธ์กัน วงจรกำเนิดสัญญาณนาฬิกาจะติดตั้งภายในด้านส่ง แสดงดังรูปที่ 2.5

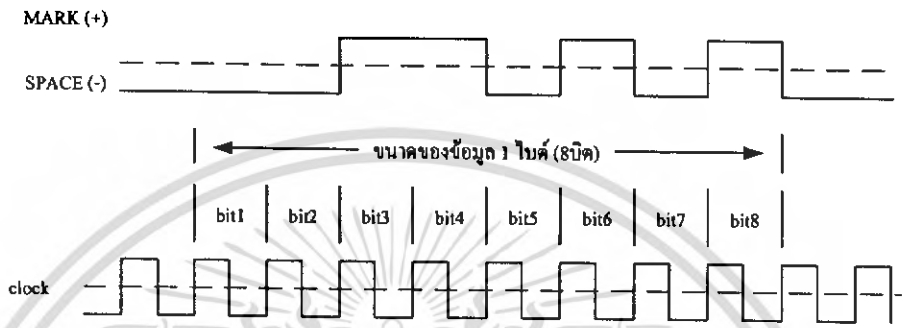


รูปที่ 2.5 แสดงการส่งข้อมูลแบบซิงโครนัส

นอกจากนี้เมื่อจังหวะเวลาถูกตั้งให้ซิ้งกับด้านรับได้แล้ว ข้อมูลจะถูกส่งไปบนทางติดต่อในแบบบิทต่อบิทต่อเนื่องกันไปอาศัยช่วงเวลาระหว่างบิทต่อบิทมีค่าเท่ากัน โดยไม่ต้องมีบิทเริ่มหรือบิทจบคอยกำกับทำให้ความเร็วในการส่งข้อมูลมีสูง นอกจากนี้ทางด้านรับต้องมีวงจรเฟสล็อคลูป ทำหน้าที่รับ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ข้อมูลจังหวะเวลาจากทางด้านส่งและสร้างสัญญาณนาฬิกาขึ้นใหม่ในด้านรับเพื่อให้เกิดการซิ้งขึ้น ข้อมูลแบบสัมพันธ์นี้จะถูกจัดการให้อยู่ในรูปของชุดข้อมูลที่มีลักษณะพิเศษ คือ ช่วงระยะเวลาระหว่างตัวอักษรด้วยกันจะไม่มี ทำให้การส่งข้อมูลเป็นไปอย่างต่อเนื่อง ข้อเสียของการส่งข้อมูลแบบสัมพันธ์คือการที่ต้องมีสัญญาณนาฬิกาขนานไปกับข้อมูลทำให้ต้องการทางติดต่อช่องที่สองเพิ่มขึ้น โดยเฉพาะกรณีระยะทางไกล ๆ เป็นการยากมากที่จะจัดหาทางติดต่อแยกต่างหากสำหรับสัญญาณนาฬิกา แสดงตัวอย่างข้อมูลที่ส่งแบบสัมพันธ์พร้อมสัญญาณนาฬิกาได้ดังรูปที่ 2.6

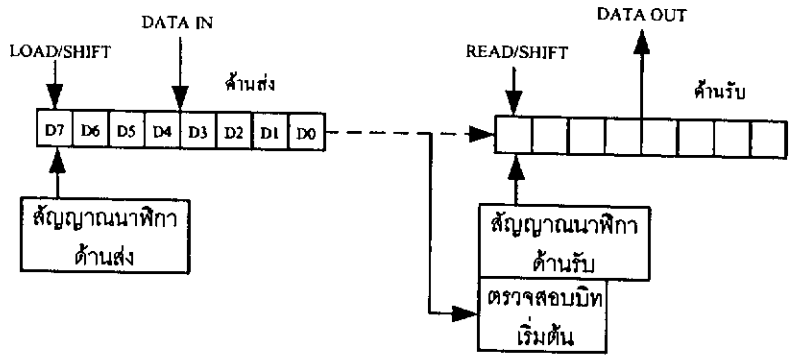


รูปที่ 2.6 แสดงการส่งข้อมูลแบบซิงโครนัส

2.2.2.2 การส่งข้อมูลแบบอะซิงโครนัส (Asynchronous Transmission)

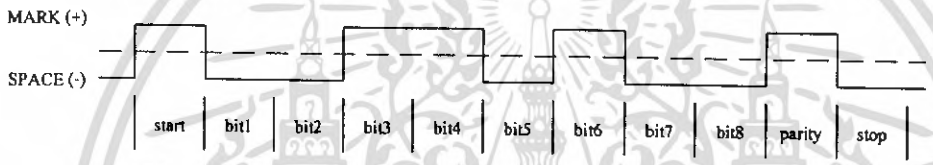
การส่งข้อมูลแบบอะซิงโครนัส หรือเรียกอีกอย่างหนึ่งว่าการส่งข้อมูลแบบไม่สัมพันธ์ จะแตกต่างกับแบบซิงโครนัส ตรงที่ไม่จำเป็นต้องมีการซิงโครนัสกันตลอดเวลาบนทางติดต่อข้อมูลโดยจะซิ้งก็ต่อเมื่อมีข้อมูลที่จะรับหรือส่งเท่านั้น ในการส่งสัญญาณแบบอะซิงโครนัสการส่งตัวอักษรสามารถเริ่มจากเวลาใดก็ได้เมื่อสายว่าง แต่อย่างไรก็ตามเพื่อให้ระบบสามารถที่จะทำงานได้จะต้องมีสภาวะบางอย่างที่ใช้บอกกับเครื่องรับให้รู้ว่า ในช่วงเวลาใดกำลังมีข้อมูลตัวอักษรปรากฏอยู่บนสาย สภาวะที่ใช้บอกนั้นคือ บิทเริ่มต้อน (Start bit) บิทเริ่มต้นไม่ใช่บิทข้อมูลแต่เป็นสัญญาณควบคุมที่จุดเริ่มต้นสถานะของสายส่งข้อมูลจะเปลี่ยนจาก “1” มาเป็น “0” และถัดจากบิทเริ่มต้นก็จะเป็นบิทข้อมูลตัวอักษร ซึ่งจะเป็นเนื้อหาข้อมูลของตัวอักษรแต่ละตัวที่ถูกส่งโดยเริ่มต้นจากบิทที่มีนัยน้อยที่สุด (LSB) จะถูกส่งตามบิทเริ่มต้นออกมา และหลังจากบิทข้อมูลถูกส่งออกมาครบแล้วจะตามด้วย บิทสิ้นสุด (Stop bit) โดยที่สถานะของสายส่งข้อมูลจะถูกบังคับกลับเป็น “1” อีกครั้งหนึ่ง หน้าที่ของบิทสิ้นสุดก็เพื่อบอกการสิ้นสุดของข้อมูลที่รับส่งโดยมีขนาด 1-2 บิท แสดงดังรูปที่ 2.7

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.7 แสดงการส่งข้อมูลแบบอะซิงโครนัส

จากการที่กำหนดให้ช่วงเวลาของบิทสิ้นสุด มีค่า 1-2 บิท ก็เพื่อกรณีบิทสุดท้ายของตัวอักษรตัวเก่า มีสถานะเป็น "0" แล้วสถานะเริ่มต้นของบิทเริ่มต้นที่สังเกตจากการเปลี่ยนแปลงสถานะของสายจาก "0" มาเป็น "1" ก็จะไม่มีความชัดเจน สำหรับรูปสัญญาณในวิธีแบบอะซิงโครนัสแสดงดังรูปที่ 2.8



รูปที่ 2.8 แสดงรูปแบบข้อมูลแบบอะซิงโครนัส

เนื่องจากการส่งข้อมูลแบบอะซิงโครนัสการเพิ่มบิทลงในข้อมูล ทำให้ความเร็วในการส่งข้อมูลช้ากว่าแบบซิงโครนัส การส่งข้อมูลแบบอะซิงโครนัสเหมาะสำหรับงานประเภทการป้อนข้อมูลเข้าเพื่อส่ง มีลักษณะไม่ต่อเนื่องอย่างเช่น จากแป้นพิมพ์ ส่วนในการส่งไฟล์ข้อมูลที่มีขนาดใหญ่มาก ๆ ควรใช้การส่งแบบซิงโครนัส

2.3 การส่งสัญญาณ (Transmission)

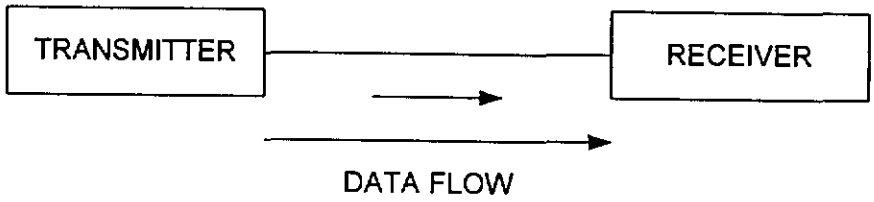
การส่งสัญญาณในที่นี้หมายถึง การนำสัญญาณจากจุดหนึ่งไปยังอีกจุดหนึ่งโดยผ่านสื่อกลางและวิธีการทางไฟฟ้า ในการรับส่งข้อมูลระหว่างกันนั้นก็จะมีความหมายเดียวกับการส่งสัญญาณ อาจแบ่งตามลักษณะและการส่งได้เป็น 3 วิธีใหญ่ ๆ คือ

2.3.1 การส่งผ่านแบบทิศทางเดียว (Simplex)

รูปแบบการส่งสัญญาณเห็นด้านรับได้ฝ่ายเดียว โดยไม่สามารถโต้ตอบผ่านทางการติดต่อได้ ตัวอย่างเช่น การกระจายเสียงของวิทยุหรือสัญญาณโทรทัศน์ ซึ่งทางด้านเครื่องรับวิทยุหรือเครื่องรับโทรทัศน์จะทำหน้าที่รับสัญญาณเพียงอย่างเดียวจะส่งข่าวหรือภาพกลับมายังสถานีส่งไม่ได้เราจึงไม่ค่อยนิยมใช้ในการสื่อสารข้อมูล เนื่องจากว่าเราจำเป็นต้องมีการโต้ตอบระหว่างการรับส่งข้อมูล หรือบางทีก็เปลี่ยนจากผู้รับเป็นผู้ส่งซึ่งจะทำไม่ได้ในการส่งผ่านแบบทิศทางเดียว นอกจากจะใช้สำหรับส่งสัญญาณโทรทัศน์และวิทยุกระจายเสียงแล้ว เครื่องโทรพิมพ์บางสำนักพิมพ์บางชนิดอาจใช้การติดต่อแบบนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

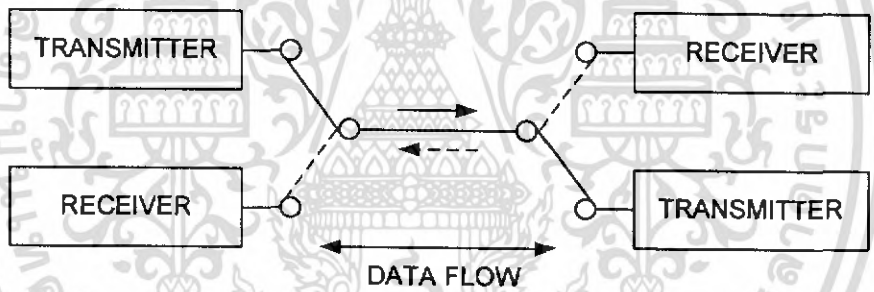
เช่นกันในการรับข่าวสารจากที่อื่น ๆ เพียงอย่างเดียว แสดงตัวอย่างการส่งผ่านแบบทิศทางเดียวได้ดังรูปที่ 2.9



รูปที่ 2.9 การส่งผ่านแบบทิศทางเดียว

2.3.2 การส่งผ่านแบบสองทิศทางแต่ต่างเวลากัน (Half - duplex)

การส่งข้อมูลแบบ Half – duplex สามารถส่งและรับสัญญาณระหว่างกันได้ แต่ว่าต้องสลับกันส่ง โดยฝ่ายหนึ่งเป็นตัวส่งและอีกฝ่ายหนึ่งเป็นตัวรับจะส่งพร้อมกันไม่ได้ ตัวอย่างเช่น การใช้วิทยุสมัครเล่นที่สามารถได้ตอบโต้กันได้ แต่ไม่พร้อมกัน แสดงรูปแบบการส่งผ่านแบบสองทิศทางแต่แตกต่างเวลากัน ดังรูปที่ 2.10

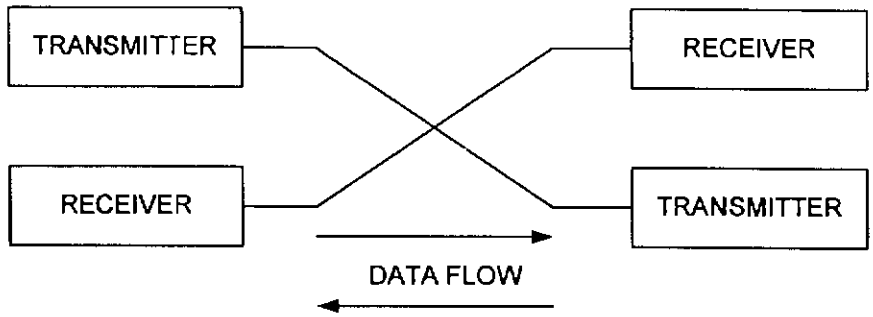


รูปที่ 2.10 การส่งผ่านแบบสองทิศทางแต่ต่างเวลากัน

2.3.3 การส่งผ่านแบบสองทิศทางที่เวลาเดียวกัน (Full - duplex)

การส่งแบบ Full – duplex จะเป็นในลักษณะที่ ผู้รับและส่งสามารถรับและส่งพร้อม ๆ กันในเวลาเดียวกันได้ ไม่จำเป็นต้องรอให้อีกฝ่ายหนึ่งส่งเสร็จเสียก่อน เช่น การสื่อสารทางโทรศัพท์ เราสามารถที่จะพูดโต้ตอบพร้อมกัน แสดงรูปแบบการส่งผ่านแบบสองทิศทางที่เวลาเดียวกันได้ดังรูปที่ 2.11

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.11 การส่งผ่านแบบสองทิศทางที่เวลาเดียวกัน

2.4 คุณสมบัติทางเทคนิคที่สำคัญของ AT89xx

- เป็นไมโครคอนโทรเลอร์ที่ใช้ซีพียูขนาด 8 บิต
- ภายในมีหน่วยความจำโปรแกรมเป็นแบบแฟลชสามารถลบและเขียนใหม่ได้พันครั้ง
- หน่วยความจำข้อมูลพื้นฐานเป็นหน่วยความจำแบบแรม ในบางเบอร์จะมีหน่วยความจำแบบอีอีพรอมเพิ่มเติม
- ขาพอร์ตเป็นแบบสองทิศทาง สามารถใช้งานเป็นได้ทั้งอินพุตและเอาท์พุต
- มีวงจรสื่อสารอนุกรมแบบฟูลดูเพล็กซ์
- ไทเมอร์/เคาน์เตอร์ขนาด 16 บิตอย่างน้อย 2 ตัว
- สามารถรองรับแหล่งกำเนิดอินเทอร์รัปป์ได้ 6 ประเภท
- มีวงจรกำเนิดสัญญาณนาฬิกาอยู่ภายในชิป
- มีวงจรสื่อสารอนุกรมแบบ SPI สำหรับในอนุกรม AT89Sxx
- มีวอตช์ด็อกไทเมอร์ในตัว สำหรับในอนุกรม AT89Sxx

2.4.1 การใช้งานเป็นพอร์ตอินพุท

เนื่องจากพอร์ตทั้งหมดของไมโครคอนโทรลเลอร์ MCS-51 แบบแฟลชสามารถเป็นได้ทั้งอินพุทและเอาท์พุท ดังนั้นจึงจำเป็นอย่างยิ่งต้องทำความเข้าใจถึงการกำหนดลักษณะการทำงานให้แก่พอร์ตของไมโครคอนโทรลเลอร์ MCS-51 แบบแฟลชในการกำหนดให้เป็นพอร์ตอินพุท ต้องเริ่มต้นด้วยการเขียนข้อมูล “1” มาที่แต่ละบิตของพอร์ตที่ต้องการใช้งานเป็นพอร์ตอินพุท เพื่อหยุดการทำงานของเฟตที่ใช้ในการขับสัญญาณของพอร์ตเชื่อมต่อกับวงจรพูลอัพในโดยตรง ส่งผลให้ขาพอร์ตนั้นมีลอจิกเป็น “1” สามารถรับสัญญาณลอจิก “0” จากอุปกรณ์ภายนอกได้ง่าย สัญญาณข้อมูลจากอุปกรณ์ภายนอกจะถูกส่งเข้ามาเก็บไว้ในวงจรบัฟเฟอร์ภายในพอร์ต แล้วรอให้ซีพียูมาอ่านค่าเข้าไป เมื่อเป็นเช่นนี้ อุปกรณ์ภายนอกที่เชื่อมต่อกับพอร์ตอินพุทของไมโครคอนโทรลเลอร์ MCS-51 แบบแฟลช ควรกำหนดให้ทำงานในสภาวะลอจิก “0” จะดีและสะดวกที่สุด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.4.2 การใช้งานเป็นพอร์ตเอาท์พุท

โดยปกติแล้ว ขาพอร์ตจะกำหนดให้มีลักษณะเป็นเอาท์พุทอยู่แล้ว ดังนั้นจึงสามารถส่งข้อมูลออกไปได้อย่างง่ายดายและตรงไปตรงมา กล่าวคือ เมื่อต้องการส่งข้อมูล "0" ออกไปทางเอาท์พุทก็ให้เขียนข้อมูล "0" ไปยังวงจรแลตช์ ซึ่งก็จะส่งผลต่อไปขับเฟต ทำให้เฟตทำงานที่ขาพอร์ตที่กำหนด ให้ทำงานเกิดลอจิก "0" ขึ้น ในทางตรงกันข้าม หากต้องการส่งข้อมูล "1" ออกไปก็ให้เขียนข้อมูล "1" ไปยังวงจรแลตช์ วงจรขับจะหยุดทำงาน ทำให้ที่ขาพอร์ตเชื่อมต่อกับวงจรพูลอัพภายในเกิดเป็นลอจิก "1" ที่ขาพอร์ตนั้น ซึ่งจะคล้ายกับการกำหนดให้เป็นขาอินพุทมาก เพียงแต่แตกต่างกันที่กระบวนการในการเคลื่อนย้ายข้อมูล โดยถ้าเป็นอินพุทจะมีสัญญาณมาอ่านข้อมูลที่บัฟเฟอร์ แต่ถ้าเป็นเอาท์พุทจะไม่มีการอ่านข้อมูลที่บัฟเฟอร์แต่อย่างใด เว้นแต่ในกรณีที่ต้องการตรวจสอบข้อมูลที่ส่งออกมาทางเอาท์พุท เมื่อใช้งานพอร์ตของไมโครคอนโทรลเลอร์ MCS-51 แบบแฟลช เป็นพอร์ตเอาพุทแต่ล่ะขา(หรือแต่ละบิต) ของแต่ละพอร์ตมีความสามารถในการจ่ายกระแสหรือที่เรียกว่า กระแสซอร์ส(Source current) ได้สูงสุด 10 มิลลิแอมแปร์(mA) และทุกขารวมกันในแต่ละพอร์ต(ทั้ง 8 บิต) สูงสุด 26 มิลลิแอมแปร์ สำหรับพอร์ต 0 และ 15 มิลลิแอมแปร์สำหรับพอร์ต 1-3 ในกรณีที่ใช้งานทุกพอร์ตเอาท์พุท จะสามารถจ่ายกระแสรวมกันได้สูงสุด 71 มิลลิแอมแปร์ ดังนั้นในการใช้งานเป็นพอร์ตเอาท์พุทเพื่อไม่ให้เกิดปัญหาเกี่ยวกับ ความสามารถในการจ่ายกระแส จึงควรต่อวงจรบัฟเฟอร์ทางเอาท์พุท เพื่อช่วยในการขับกระแสอีกทางหนึ่ง

2.4.3 การอ่านค่าลอจิกจากพอร์ท

ในไมโครคอนโทรลเลอร์ MCS-51 แบบแฟลช สามารถอ่านค่าลอจิกจากพอร์ตได้ 2 ลักษณะ คืออ่านจากขาพอร์ตโดยตรง และอ่านจากวงจรแลตช์ของแต่ละพอร์ต
ในกรณีที่พอร์ตต่อกับขาเบสทรานซิสเตอร์ชนิด NPN และขาอิมิตเตอร์ของทรานซิสเตอร์ตัวนั้นต่อลงกราวด์ หากมีการส่งข้อมูล "1" ไปยังทรานซิสเตอร์ จะทำให้ทรานซิสเตอร์ทำงานสถานะลอจิกที่ขาพอร์ตจะเป็น "0" เนื่องจาก เมื่อทรานซิสเตอร์ทำงาน จะเสมือนว่าพอร์ตนั้นถูกต่อลงกราวด์ ทำให้อ่านค่าลอจิกที่ขาพอร์ทจะได้ผลตรงข้ามกับที่ส่งออกมา แต่ถ้าหากทำงานอ่านค่าลอจิกที่วงจรแลตช์ จะได้ค่าที่ตรงกับค่าที่ต้องการส่งจริง ดังนั้น ในการอ่านค่าลอจิกพอร์ทจึงต้องเลือกวิธีการที่เหมาะสมกับอุปกรณ์ที่นำมาต่อด้วย

2.4.4 รีจิสเตอร์ที่เกี่ยวข้องในการสื่อสารข้อมูลอนุกรม

รีจิสเตอร์ควบคุมการทำงานของพอร์ตอนุกรมหรือ SCON (Serial port Control Register)
SM0/FE (Serial port mode bit 0/Framing error bit) : ปกติจะใช้กับบิต SM1 เพื่อกำหนดโหมดการทำงานของพอร์ตอนุกรม การเข้าถึงบิตนี้จะเกิดขึ้นได้ก็ต่อเมื่อมีการเคลียร์บิต SMOD ซึ่งก็คือ บิต 6 ของรีจิสเตอร์ PCON ในกรณีที่ใช้ความสามารถตรวจจับความผิดพลาดของเฟรมข้อมูล บิตนี้จะใช้แจ้งความผิดพลาดที่เกิดขึ้น โดยจะเซตเป็น "1" ทันทีเมื่อพบว่า ไม่สามารถตรวจจับบิตหยุดหรือบิตปิดท้ายของ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ข้อมูลของพอร์ตอนุกรมได้ การเอ็นเอเบิลความสามารถนี้ทำได้โดยการเซตบิต SMOD ในรีจีสเตอร์ PCON การเคลียร์บิตนี้ต้องทำทางซอฟต์แวร์เท่านั้น

SM1 (Serial port mode bit 1) : ใช้ร่วมกับบิต SM0 ในการกำหนดโหมดการทำงานของพอร์ตอนุกรมในไมโครคอนโทรเลอร์ ดังมีรายละเอียดดังต่อไปนี้

SM0	SM1	โหมด	รายละเอียด	อัตราบอด
0	0	0	ชิฟต์รีจีสเตอร์	ความถี่สัญญาณนาฬิกา/6 (ในโหมด 6 ไซเกิลสัญญาณนาฬิกา)
0	1	1	UART 8 บิต	ปรับค่าได้
1	0	2	UART 9 บิต	ความถี่สัญญาณนาฬิกา/32 (ในโหมด 6 ไซเกิลสัญญาณนาฬิกา)
1	1	3	UART 9 บิต	ปรับค่าได้

SM2 (Serial port mode bit 2) : ใช้ในการเอ็นเอเบิลความสามารถการรับรู้แอดเดรสในการติดต่ออัตโนมัติ เมื่อมีการเชื่อมต่อไมโครคอนโทรเลอร์หลายตัวเข้าด้วยกัน โดยความสามารถนี้จะใช้ได้กี่ต่อเมื่อวงจรพอร์ตอนุกรมของไมโครคอนโทรเลอร์ทำงานในโหมด 2 หรือ 3 ถ้าบิต SM2 เป็น "1" บิต RI จะไม่เซต เว้นแต่ข้อมูลบิตที่ 9 ที่รับเข้ามาเป็น "1" เป็นแจ้งว่า สามารถติดต่อได้และข้อมูลที่รับเข้ามาคือ ค่าแอดเดรสที่ต้องการติดต่อด้วย

ในกรณีที่พอร์ตอนุกรมทำงานอยู่ในโหมด 1 ถ้าบิต SM2 เซต บิต RI จะไม่มีการเปลี่ยนแปลง จนกว่าจะได้รับข้อมูลบิตหยุดหรือบิตปิดท้าย และข้อมูลที่รับได้จะเป็นข้อมูลแอดเดรสที่ต้องการติดต่อด้วย

ในกรณีที่วงจรพอร์ตอนุกรมทำงานในโหมด 0 บิต SM2 นี้จะเป็น "0"

REN (Receive enable bit) : ใช้เอ็นเอเบิลความสามารถในการรับรู้ข้อมูลของวงจรพอร์ตอนุกรม โดยที่

"1" เอ็นเอเบิลการรับข้อมูล

"0" ดิสเอเบิลการรับข้อมูล

การเซตหรือเคลียร์บิตนี้ต้องกระทำด้วยกระบวนการทางซอฟต์แวร์

TB8 (Transmit data bit 8) : ใช้เก็บข้อมูลบิต 8 หรือบิตที่ 9 ที่ต้องการส่งออกทางพอร์ตอนุกรม เมื่อทำงานในโหมด 2 และ 3

RB8 (Receive data bit 8) : ใช้เก็บข้อมูลบิต 8 ของข้อมูลที่รับเข้ามาของพอร์ตอนุกรมเมื่อทำงานในโหมด 2 และ 3 ในโหมด 1 ถ้าบิต SM2 = 0 ข้อมูลของบิตหยุดจะเก็บไว้ที่บิตนี้ ไม่ใช้งานบิตนี้ในโหมด 0

TI (Transmit interrupt flag) : บิตแสดงการเกิดอินเตอร์รัปต์จากการส่งข้อมูลออกทางพอร์ตอนุกรมเมื่อทำงานในโหมด 0 บิตนี้จะเซตเมื่อมีการส่งข้อมูลบิต 7 หรือบิตที่ 8 ออกไป แต่ถ้าทำงาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ในโหมดอื่น บิตนี้จะเซตเมื่อมีการเริ่มต้นส่งบิตหยุดหรือบิตปิดท้าย การเคลียร์บิตนี้ต้องกระทำด้วยกระบวนการทางซอฟต์แวร์

RI (Receive interrupt flag) : บิตแสดงการเกิดอินเตอร์รัปต์เนื่องจากการรับข้อมูลเข้ามาทางพอร์ตอนุกรม เมื่อทำงานในโหมด 0 บิตนี้จะเซตเมื่อรับข้อมูลบิต 7 หรือบิตที่ 8 เสร็จสมบูรณ์ แต่ถ้าทำงานในโหมดอื่น บิตนี้จะเซตเมื่อการรับบิตปิดท้ายดำเนินไปได้ครึ่งทาง นอกจากนี้การเซตบิตนี้ยังมีเงื่อนไขที่กำหนดโดยบิต SM2 ร่วมด้วย การเคลียร์บิตนี้ต้องกระทำด้วยกระบวนการทางซอฟต์แวร์ โดยส่วนใหญ่แล้วมักจะตั้งค่าของ SCON ดังนี้

SCON = 0x40; พอร์ตอนุกรมทำงานในโหมด 1 สามารถส่งข้อมูลได้อย่างเดียว

SCON = 0x50; พอร์ตอนุกรมทำงานในโหมด 1 สามารถรับส่งข้อมูลได้

2.5 การมอดูเลตแบบฟรีเควนซีชิพคีย์อิ้ง (Frequency Shift Keying)

การมอดูเลตสัญญาณดิจิตอล (Digital Modulation) เข้ากับคลื่นพาห์ที่เป็นสัญญาณรูปไซน์ (Sinusoidal Wave) ทำเพื่อต้องการให้ได้สัญญาณดิจิตอลเหล่านั้นสามารถส่งผ่านตัวกลางที่ออกแบบมาสำหรับสัญญาณแบบอนาล็อกได้ หรือทำการแปลงจากสัญญาณอนาล็อกนั่นเอง ซึ่งวิธีที่นิยมใช้ 4 แบบ

1. แอมปลิจูดชิพคีย์อิ้ง (Amplitude Shift Keying หรือ ASK)
2. ฟรีเควนซีชิพคีย์อิ้ง (Frequency Shift Keying หรือ FSK)
3. เฟสชิพคีย์อิ้ง (Phase Shift Keying หรือ PSK)
4. ควอดราเจอร์แอมปลิจูดมอดูเลชั่น (Quadrature Amplitude Modulation หรือ QAM)

การมอดูเลตแบบเอฟเอสเคเป็นการมอดูเลตที่รูปคลื่นสัญญาณที่มีการเปลี่ยนแปลงความถี่ซึ่งถูกควบคุมโดยระดับแรงดันของสัญญาณข่าวสาร โดยเมื่อมีการเปลี่ยนแปลงระดับแรงดันสัญญาณก็จะมีการเปลี่ยนแปลงความถี่ของสัญญาณคลื่นพาห์ส่วนแอมปลิจูดและเฟสคงที่ตลอด การสร้างสัญญาณเอฟเอสเคสามารถทำได้ 2 แบบ คือ

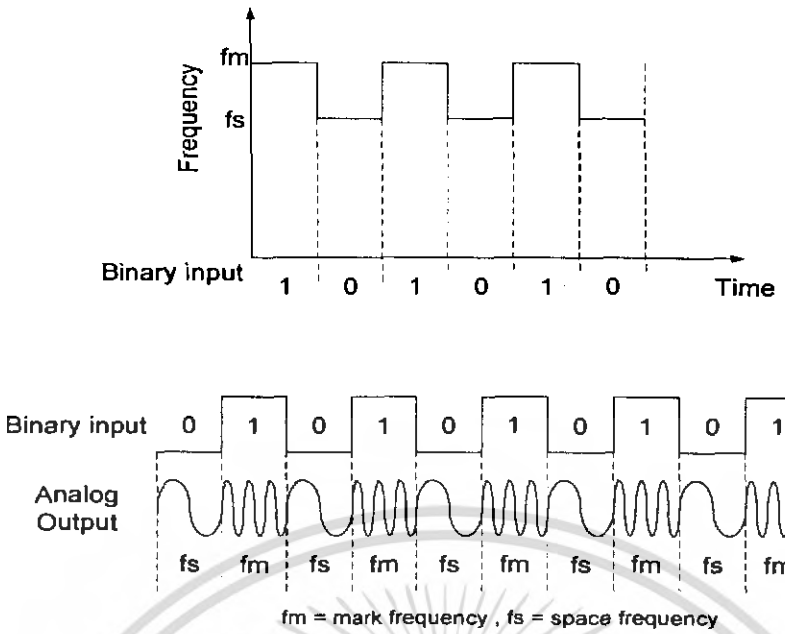
1. สร้างจากสวิตช์จากดิจิตอล สัญญาณดิจิตอลจะเลือกออสซิลเลเตอร์ที่มีความถี่ที่ต้องการเพื่อส่งสัญญาณออกไปโดยจำนวนออสซิลเลเตอร์ที่ใช้จะต้องมีจำนวนเท่ากับจำนวนระดับแรงดันของสัญญาณข่าวสาร จึงทำให้เฟสของสัญญาณแต่ละช่วงไม่มีความต่อเนื่องกัน
2. สร้างโดยการใช้ออสซิลเลเตอร์ตัวเดียวกัน โดยที่ออสซิลเลเตอร์นั้นสามารถเปลี่ยนแปลงความถี่ได้ สัญญาณที่ได้จากวิธีนี้จะมีความต่อเนื่องของสัญญาณ จึงเรียกว่า เอฟเอสเคต่อเนื่อง

เมื่อพิจารณาถึงลักษณะของสัญญาณเอฟเอสเคพบว่า เป็นการเปลี่ยน Digital Signal ให้เป็น Analog Data โดยเมื่อข้อมูล "0" เข้ามา จะถูกเปลี่ยนเป็นสัญญาณไซน์ที่มีความถี่ ω_1 และเมื่อข้อมูล "1" เข้ามา ก็จะถูกเปลี่ยนเป็นสัญญาณไซน์ที่มีความถี่ ω_2 ซึ่งสามารถเขียนเป็นสมการ

$S_1(t) = A\cos(\omega_1 t)$ สำหรับสัญญาณไบนารี 0

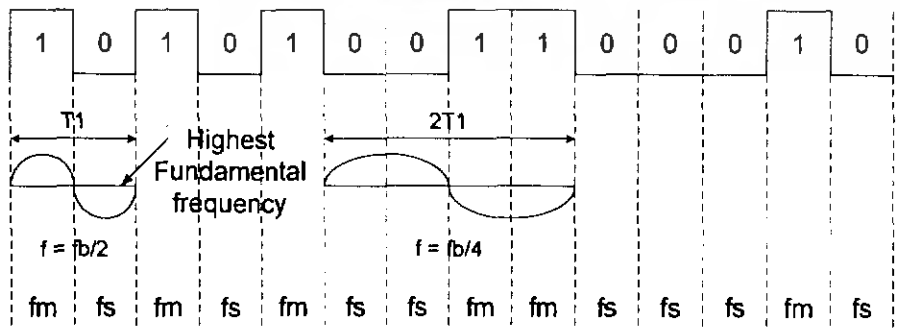
$S_2(t) = A\cos(\omega_2 t)$ สำหรับสัญญาณไบนารี 1

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.12 ลักษณะของสัญญาณเอฟเอสเค

หลักการของเอฟเอสเค คือ เมื่อสัญญาณดิจิตอลที่มีลักษณะเป็นข้อมูลไบนารีถูกส่งเข้าจะทำให้ความถี่เลื่อนหรือเบี่ยงเบนไปตามการเปลี่ยนแปลงของข้อมูลไบนารีที่เข้ามาดังนั้นสัญญาณทางเอาต์พุตของเอฟเอสเคมอดูเลเตอร์จะอยู่ในรูปของความถี่ที่มีการเปลี่ยนแปลงอย่างต่อเนื่อง (Frequency Continuous) เมื่อข้อมูลไบนารีด้านอินพุตเปลี่ยนแปลงจากสถานะลอจิก "1" เป็นลอจิก "0" (หรือในทางกลับกันคือลอจิก "0" เป็นลอจิก "1") สัญญาณเอาต์พุตก็จะเลื่อนความถี่ระหว่าง 2 ความถี่ด้วยกัน คือ ความถี่ที่ลอจิก "1" หรือ Mark Frequency (f_m) และความถี่ที่ลอจิก "0" หรือ Space Frequency (f_s) โดยการเปลี่ยนแปลงหรือการเลื่อนของความถี่แต่ละครั้งจะเกิดขึ้นเมื่อสถานะของลอจิกด้านสัญญาณเข้าเปลี่ยนแปลงนั่นคืออัตราการเปลี่ยนแปลงสัญญาณออกจะเท่ากับสัญญาณเข้า ซึ่งในดิจิตอลมอดูลเลชั่นนั้นอัตราการเปลี่ยนแปลงของสัญญาณ ด้านอินพุตของสัญญาณมอดูเลเตอร์จะเรียกว่าอัตราบิต (Bit Rate) มีหน่วยเป็นบิตต่อวินาที (bps) ส่วนอัตราการเปลี่ยนแปลงของสัญญาณด้านเอาต์พุตของสัญญาณมอดูเลเตอร์เรียกว่า อัตราบอด (Baud Rate) ดังนั้น ในการส่งข้อมูลของเอฟเอสเคมอดูเลเตอร์ อัตราบิตจะเท่ากับอัตราบอดเสมอ



รูปที่ 2.13 อัตราการเปลี่ยนแปลงของสัญญาณอินพุต

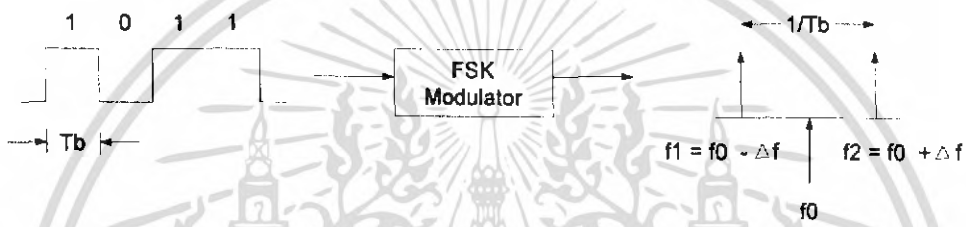
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากรูปที่ 2.13 อัตราการเปลี่ยนแปลงที่เร็วที่สุดของสัญญาณอินพุตจะเกิดขึ้นเมื่อไบนารีมีลักษณะเป็น 1 และ 0 สลับกันซึ่งก็คือ สัญญาณสี่เหลี่ยม (Square Wave) นั่นเอง ซึ่งความถี่หลัก (Fundamental Frequency) ของคลื่นสี่เหลี่ยมจะเท่ากับครึ่งหนึ่งของอัตราการส่งข้อมูล ดังนั้นถ้าพิจารณาเฉพาะความถี่หลักเพียงอย่างเดียวแล้วความถี่สูงสุดของสัญญาณดิจิตอลที่ต้องการนำมามอดูเลตแบบเอฟเอสเคจะเท่ากับครึ่งหนึ่งของอัตราการส่งข้อมูล คือ

$$f_{a\max} = Bit\ Rate/2$$

เมื่อ $f_{a\max}$ คือความถี่สูงสุดของสัญญาณดิจิตอลที่จะนำมามอดูเลต

ความถี่กลาง (Center Frequency หรือ f_o) ของวงจรออสซิลเลเตอร์ควบคุมด้วยแรงดัน (Voltage Controlled Oscillator) จะอยู่ในตำแหน่งกลางระหว่าง f_m กับ f_s ดังแสดงในรูปที่ 2.14



รูปที่ 2.14 การเบี่ยงเบนความถี่ของเอฟเอสเค

ลอจิก 1 ด้านอินพุตจะเลื่อนความถี่ของวงจรแรงดันควบคุมออสซิลเลเตอร์ จาก f_o ไปจะเห็นว่าการเปลี่ยนแปลงของข้อมูลไบนารีด้านอินพุต "1" ไป "0" หรือ "0" ไป "1" จะทำให้ความถี่เอาต์พุตของเอฟเอสเคมอดูเลเตอร์เลื่อนหรือเบี่ยงเบนกลับไปกลับมาระหว่าง f_m หรือ f_s เนื่องจากเอฟเอสเคนั่นก็คือ รูปแบบหนึ่งของการมอดูเลตทางความถี่ดังนั้นดัชนีการมอดูเลต (Modulate Index หรือ MI) .ในเอฟเอสเคก็สามารถหาได้เช่นเดียวกับการมอดูเลตทางความถี่

$$MI = \Delta f / f_a$$

เมื่อ MI คือ ดัชนีการมอดูเลต

Δf คือ การเบี่ยงเบนของความถี่ใดๆ จากความถี่กลาง (Hz)

f_a คือ ความถี่ของสัญญาณที่นำมามอดูเลต (Hz)

ค่า MI ที่ยอมให้ได้สูงสุด คือ ค่า MI ที่ทำให้แบนวิดธ์กว้างที่สุดซึ่งจะเกิดขึ้นจากการเบี่ยงเบนของความถี่ที่ถูกมอดูเลตแล้ว และความถี่ของสัญญาณที่นำมามอดูเลตมีค่าสูงสุดในเอฟเอสเค ค่า Δf เป็นการเบี่ยงเบนของความถี่สูงสุด (Peak Frequency Deviation) ของสัญญาณที่ถูกมอดูเลตซึ่งมีค่าเท่ากับความแตกต่างระหว่าง f_m กับ f_o หรือ f_o กับ f_s ซึ่งก็คือ ครึ่งหนึ่งของความแตกต่างระหว่าง f_m กับ f_s นั่นคือ

$$\Delta f = (f_m - f_s)/2$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การเบี่ยงเบนของความถี่สูงสุดจะขึ้นอยู่กับขนาด หรือแอมปลิจูดของสัญญาณที่นำมามอดูเลต (สัญญาณดิจิตอล) เมื่อสถานะทางลอจิกเป็น "1" จะให้แรงดันออกมาคงที่ค่าหนึ่ง (เช่น 5 โวลต์) หรือถ้าในลอจิก "0" แรงดันออกมาคงที่ในระดับลอจิกเช่นกัน (เช่น 0 โวลต์)

f_a เป็นความถี่ของข้อมูลไบนารี่ด้านอินพุต ซึ่งจะทำให้แบนวิดธ์กว้างที่สุดเมื่อ f_a = Bit Rate/2 เท่านั้น เพราะฉะนั้นเราสามารถหาค่า ได้จาก

$$MI = (f_s - f_m) / f_b$$

เมื่อ $(f_s - f_m)$ คือ ความถี่เบี่ยงเบนสูงสุด f_b คือ อัตราบิตของไบนารี่อินพุตโดยทั่วไป ความกว้างของแบนวิดธ์จะแปรผันตรงกับค่า MI ซึ่งเช่นเดียวกับฟรีเควนซีชิพคีย์อิ้ง ที่ค่า MI โดยทั่วๆไป จะต้องมีค่าต่ำกว่า 1.0 เพื่อให้มีค่าแบนวิดธ์ ที่แคบที่สุดเรียกว่า Minimum Nyquist Bandwidth (f_b) ส่วนทางด้านฟรีเควนซีชิพคีย์อิ้งมอดูเลชั่น จะเป็นตัวแยกสัญญาณไบนารีออกจากฟรีเควนซีชิพคีย์อิ้ง ซึ่งสามารถกระทำได้หลายแบบแต่ในที่นี้จะกล่าวถึง 3 แบบ คือ

1. ใช้วงจร Match Filter วงจรจะประกอบไปด้วยส่วนของวงจรกรองแถบความถี่ผ่าน (Bandpass Filter) ซึ่งจะทำหน้าที่กรองความถี่ f_1 และ f_2 ซึ่งเราจะแทนด้วยลอจิก 1 และลอจิก 0 ตามลำดับ ต่อมานำเอาต์พุตของวงจรกรองแถบความถี่ผ่าน มาเข้าวงจรตรวจจับกรอบสัญญาณ (Envelope Detector) ซึ่งจะทำการจับกรอบของสัญญาณที่ออกมา จากวงจรกรองแถบความถี่ผ่าน (Bandpass Filter) และนำมาเข้าวงจรเปรียบเทียบ (Comparator) ซึ่งจะได้สัญญาณออกมา

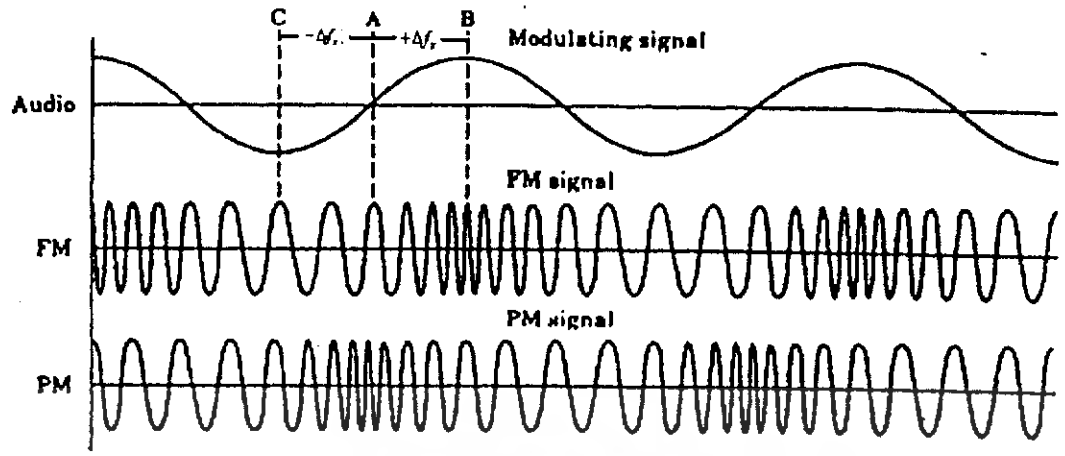
2. ใช้วงจรเฟสล็อกลูป (Phase locked loop:PLL) จะมีความถี่ฟรีรันนิ่งเท่ากับ center

3. Frequency (f_o) และในขณะที่ความถี่อินพุตของวงจรเฟสล็อกลูปเลื่อนไปเลื่อนมาระหว่าง f_s กับ f_m จะทำให้เกิดแรงดันคลาดเคลื่อนไฟตรง (DC Error Voltage) ซึ่งเป็นผลมาจากการเปรียบเทียบทางเฟส (Phase Comparator) ของสัญญาณอินพุตเนื่องจากความถี่อินพุตที่เข้ามายังวงจรเฟสล็อกลูปมีเพียง 2 ความถี่ คือ f_m กับ f_s ดังนั้นค่าแรงดันดังกล่าวจึงมีเพียง 2 ระดับเท่านั้น ซึ่งสามารถแทนค่าลอจิก "1" และลอจิก "0" เมื่อความถี่ทางอินพุตเป็น f_m กับ f_s ตามลำดับ เราจึงได้สัญญาณเอาต์พุตจากวงจรเฟสล็อกลูปกลับมาเป็นข้อมูลไบนารีเหมือนกับตอนแรกที่ส่งมาทุกประการ

2.6 การมอดูเลทของคลื่น

จากรูปที่ 2.15 เป็นรูปสัญญาณที่แสดงในรูปของ Time-domain ที่เกิดจากการมอดูเลตสัญญาณเสียงกับสัญญาณพาหะซึ่งมีการมอดูเลตใน 2 ลักษณะ คือ การมอดูเลตทางความถี่ (Frequency Modulation : FM) และการมอดูเลตทางเฟส (Phase Modulation : PM) ซึ่งรูปสัญญาณจากการมอดูเลตจะเป็นลักษณะขบวนความถี่ จากรูปที่ 2.15 จุด A คือ ความถี่ศูนย์กลางของความถี่เสียง ส่วนจุด B คือ คลื่น FM ซึ่งจะมีความถี่ที่สูงกว่าความถี่ศูนย์กลางของความถี่เสียง และที่จุด C คือ คลื่น FM ที่มีความถี่ต่ำกว่าความถี่ศูนย์กลางของความถี่เสียงและจะสังเกตเห็นว่าสัญญาณพาหะจะเปลี่ยนแปลงความถี่ตามความถี่ของสัญญาณความถี่เสียงที่เข้ามามอดูเลต

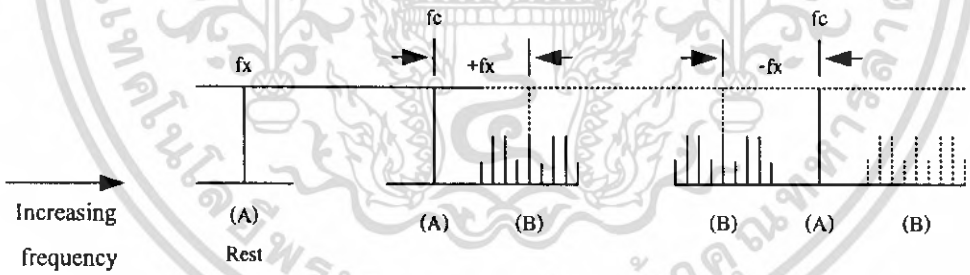
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.15 แสดงสัญญาณความถี่เสียงกับสัญญาณที่ถูกมอดูเลตแบบ FM และ PM

จากรูปที่ 2.15 เป็นรูปสัญญาณที่เกิดจากการมอดูเลตสัญญาณใน 1 ไซเคิล ซึ่งจะแสดงในรูปของ Time– domain โดยมีความถี่ศูนย์กลางอยู่ที่จุด A ส่วนที่จุด B จะเป็นจุดที่มีการเปลี่ยนแปลงความถี่ที่มีค่ามากกว่าความถี่ศูนย์กลาง ($+\Delta f_x$) และที่จุด C จะเป็นจุดที่มีการเปลี่ยนแปลงความถี่ที่มีน้อยกว่าความถี่ศูนย์กลาง ($-\Delta f_x$)

การเปลี่ยนแปลงความถี่ของสัญญาณไปพร้อม ๆ กับความถี่ของสัญญาณที่กำลังถูกมอดูเลตนั้น จะมีผลทำให้มีไซแบนด์เกิดขึ้นรอบ ๆ สัญญาณพาหะ และจำนวนไซแบนด์ที่เกิดขึ้นจะมีความถี่ที่ออกห่างจากความถี่ศูนย์กลาง โดยที่ผลรวมของแรงดันและไซแบนด์ของสัญญาณพาหะที่ออกห่างจากความถี่ศูนย์กลางจะมีค่าคงที่ และจะมีค่าที่เท่ากับแรงดันและไซแบนด์ที่ค่าของความถี่ศูนย์กลาง



รูปที่ 2.16 แสดงการมอดูเลตคลื่น FM ใน Frequency-domain

2.6.1 ดัชนีการมอดูเลต

ในระบบ FM เราวัดเปอร์เซนต์การมอดูเลตโดยดูจากการเปลี่ยนแปลงความถี่ ซึ่งเรานิยมเรียกว่า ดัชนีการมอดูเลต ถ้าลองพิจารณาความหมายของดัชนีการมอดูเลตต่อไปนี้

$$m = \frac{f_d}{f_m}$$

เมื่อ f_d คือช่วงความถี่เบี่ยงเบน

f_m คือความถี่ของสัญญาณที่เข้ามอดูเลต

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ค่าตัวเลขของดัชนีการมอดูเลตจะมีค่าสูง (แตกต่างจากเปอร์เซ็นต์การมอดูเลตซึ่งเมื่อคิดเป็นอัตราส่วนจะได้อยู่ระหว่าง 0 ถึง 1) ตัวอย่างเช่น ในระบบวิทยุกระจายเสียง FM เรากำหนดให้ความถี่เบี่ยงเบนของระบบสูงสุดไว้เท่ากับ 75 kHz สมมติว่าเราใช้สัญญาณเสียง 1 kHz มอดูเลตให้เกิดความถี่เบี่ยงเบนเต็มที่ ค่าดัชนีการมอดูเลตจะเป็น

$$m = \frac{75kHz}{1kHz} = 75$$

สังเกตว่าค่าดัชนีการมอดูเลตในระบบ FM ขึ้นอยู่กับความถี่ของสัญญาณเสียงที่เข้ามอดูเลต ในทางปฏิบัติเรานิยมวัดเป็นอัตราส่วนการเบี่ยงเบน (deviation ratio) ซึ่งเป็นอัตราส่วนระหว่างความถี่เบี่ยงเบน (ของระบบ) สูงสุด ($f_{d\max}$) ต่อความถี่สูงสุดของสัญญาณที่เข้ามอดูเลต ($f_{m\max}$) ในระบบกระจายเสียง FM ค่าอัตราการเบี่ยงเบน (Δ) จะเท่ากับ

$$\Delta = \frac{f_{d\max}}{f_{m\max}}$$

$$\Delta = \frac{75kHz}{15kHz} = 5$$

ในระบบ FM เมื่อเพิ่มแอมพลิจูดของสัญญาณที่เข้ามอดูเลตสูงขึ้น การเบี่ยงเบนความถี่ของพาหะจะเบี่ยงเบนได้มากขึ้น ในระบบวิทยุกระจายเสียง FM กำหนดให้ความถี่เบี่ยงเบนของระบบเต็มที่ไม่เกิน 75 kHz ถ้าเรามอดูเลตทำให้ความถี่ของพาหะเบี่ยงเบนไปเท่ากับ 75 kHz แสดงว่าเรามอดูเลต 100 เปอร์เซนต์ ซึ่งเราสามารถเขียนเป็นสมการได้ดังนี้

$$\text{เปอร์เซนต์การมอดูเลต} = \frac{f_d}{f_{m\max}} \times 100$$

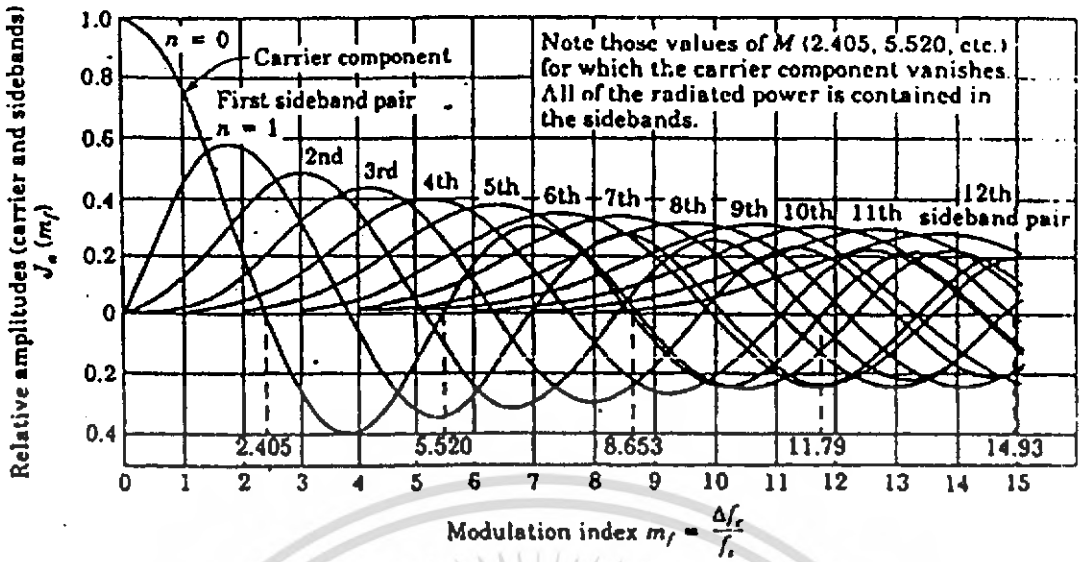
ในที่นี้ f_d คือความถี่เบี่ยงเบนเนื่องจากสัญญาณที่เข้ามอดูเลต

$f_{m\max}$ คือความถี่เบี่ยงเบนสูงสุดของระบบ

2.6.2 ไซด์แบนด์ FM

จากการมอดูเลตแบบ FM ถ้าเรามอดูเลตด้วยสัญญาณซายน์ก็จะเกิดไซแบนด์จำนวนนับอนันต์ เนื่องจากการเบี่ยงเบนความถี่ของพาหะทำให้เกิดความถี่เพิ่มขึ้นอีกมากมาย ความจริงแล้วไซด์แบนด์ที่อยู่ห่างจากความถี่กลางมาก ๆ มักมีแอมพลิจูดเล็กมากจนไม่ต้องคำนึงถึง ในระบบ FM สัญญาณ FM เรารักษาแอมพลิจูดไว้คงที่เสมอ ซึ่งหมายความว่ากำลังของคลื่นพาหะย่อมกระจายไปอยู่ในไซด์แบนด์ ความสัมพันธ์ของพาหะกับไซด์แบนด์ในระบบ FM ขึ้นอยู่กับดัชนีการมอดูเลต เนื่องจากดัชนีการมอดูเลตเป็นตัวกำหนดจำนวนของไซด์แบนด์ที่สำคัญและแอมพลิจูดของพาหะ กับไซด์แบนด์ต่าง ๆ

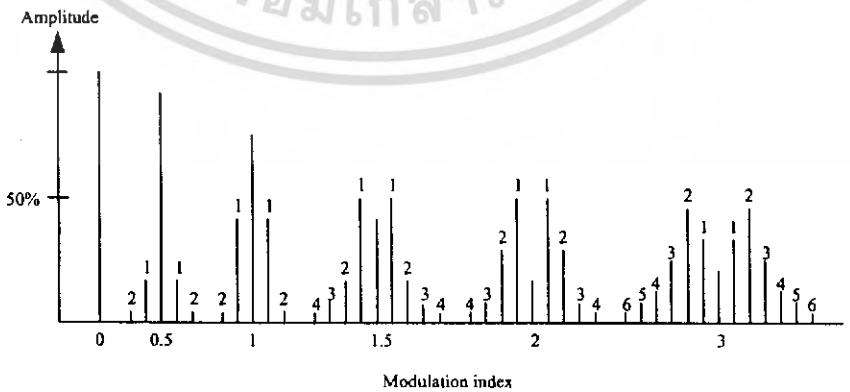
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.17 กราฟแสดงแอมพลิจูดของพาหะและไซด์แบนด์ในระบบ FM

ในรูปที่ 2.17 แสดงกราฟแอมพลิจูดของคลื่นพาหะกับไซด์แบนด์ที่ดัชนีมอดูเลตค่าต่าง ๆ จะเห็นว่าเมื่อดัชนีการมอดูเลตเป็นศูนย์จะมีแต่คลื่นพาหะอย่างเดียว (เท่ากับ 1 หน่วย) คลื่นไซด์แบนด์เป็นศูนย์เมื่อดัชนีการมอดูเลตเพิ่มขึ้นจำนวนไซด์แบนด์จะเพิ่มขึ้น แอมพลิจูดของไซด์แบนด์ก็จะใหญ่ขึ้น แต่แอมพลิจูดของพาหะกลับมีขนาดเล็กลงจนกระทั่งดัชนีการมอดูเลตเท่ากับ 2.4 คลื่นพาหะจะเป็นศูนย์ ตอนนี้กำลังของคลื่น FM จะไปอยู่ในไซด์แบนด์ทั้งสิ้น เมื่อดัชนีการมอดูเลตเพิ่มขึ้นอีก คลื่นพาหะก็จะมีค่าเพิ่มขึ้นอีก (เป็นค่าที่แสดงว่าเฟสตรงกันข้ามกับตอนแรก เช่น เมื่อดัชนีการมอดูเลตเป็น 3.1 แอมพลิจูดของพาหะจะเท่ากับ - 0.3 แอมพลิจูดของพาหะจะเท่ากับ - 0.3 หน่วย) สังเกตว่าที่จุดคลื่นพาหะเป็นศูนย์นั้นมีอยู่หลายจุด

กราฟในรูปที่ 2.17 เขียนเป็นตารางดังแสดงในตารางที่ 2.1 ในที่นี้เราตัดไซด์แบนด์ที่มี แอมพลิจูดน้อยกว่า 1 เปอร์เซนต์ของพาหะเดิม (ก่อนมอดูเลต) ออกไปโดยไม่คำนึงถึงเช่น เมื่อดัชนีการมอดูเลตเท่ากับ 0.5 แอมพลิจูดของพาหะจะเท่ากับ 0.94 หน่วย ไซด์แบนด์คู่แรกมีแอมพลิจูดเท่ากับ 0.24 หน่วย ไซด์แบนด์คู่ที่สองถัดไปมีแอมพลิจูดเท่ากับ 0.03 หน่วย ไซด์แบนด์อื่นนอกจากนี้มีแอมพลิจูดน้อยจนสามารถตัดทิ้งไปได้ เมื่อดัชนีการมอดูเลทสูงขึ้นการกระจายคลื่นไซด์แบนด์จะเป็นดังตารางที่ 2.1



รูปที่ 2.18 รูปคลื่น FM ในเชิงความถี่ที่ค่าดัชนีการมอดูเลตเท่ากับ 0,0.5,1.1.5,2,3

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้
62725

ตารางที่ 2.1 แสดงการกระจายคลื่นพาหะและไซด์แบนด์ที่ดัชนีการมอดูเลตค่าต่างๆ

ดัชนีการมอดูเลต	พาหะ	ไซด์แบนด์คู่ที่															
		1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
0	1	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
0.25	0.98	0.12	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
0.5	0.94	0.24	0.03	-	-	-	-	-	-	-	-	-	-	-	-	-	-
1	0.77	0.44	0.11	0.02	-	-	-	-	-	-	-	-	-	-	-	-	-
1.5	0.51	0.56	0.23	0.06	0.01	-	-	-	-	-	-	-	-	-	-	-	-
2	0.22	0.58	0.35	0.13	0.03	-	-	-	-	-	-	-	-	-	-	-	-
2.5	-0.05	0.5	0.45	0.22	0.07	0.02	-	-	-	-	-	-	-	-	-	-	-
3	-0.26	0.34	0.49	0.31	0.13	0.04	0.01	-	-	-	-	-	-	-	-	-	-
4	-0.4	-0.07	0.36	0.43	0.28	0.13	0.05	0.02	-	-	-	-	-	-	-	-	-
5	-0.18	-0.33	0.05	0.36	0.39	0.26	0.13	0.05	0.02	-	-	-	-	-	-	-	-
6	0.15	0.28	-0.24	0.11	0.36	0.36	0.25	0.13	0.06	0.02	-	-	-	-	-	-	-
7	0.3	0	-0.3	-0.17	0.16	0.35	0.34	0.23	0.13	0.06	0.02	-	-	-	-	-	-
8	0.17	0.23	-0.11	-0.29	-0.1	0.19	0.34	0.32	0.22	0.13	0.06	0.03	-	-	-	-	-
9	-0.09	0.24	0.14	-0.18	-0.27	-0.26	0.2	0.33	0.3	0.21	0.12	0.06	0.03	0.01	-	-	-
10	-0.25	0.04	0.25	0.06	-0.22	-0.23	-0.01	0.22	0.31	0.29	0.2	0.12	0.06	0.03	0.01	-	-
12	-0.05	-0.22	-0.08	0.2	0.18	-0.17	-0.24	-0.17	0.05	0.23	0.3	0.27	0.2	0.12	0.07	0.03	0.01
15	-0.01	0.21	0.04	0.19	-0.12	0.13	0.21	0.03	-0.17	-0.22	-0.09	0.1	0.24	0.28	0.25	0.18	0.12

2.6.3 แบนด์วิดท์ของสัญญาณ FM

ในระบบ FM จำนวนไซด์แบนด์และแอมพลิจูดของไซด์แบนด์ขึ้นอยู่กับค่าดัชนีการมอดูเลต โดยความถี่ของไซด์แบนด์มีค่าสัมพันธ์กับความถี่ของสัญญาณที่เข้ามอดูเลต กล่าวคือไซด์แบนด์คู่แรกมีความถี่เท่ากับ $f_o \pm f_m$ ไซด์แบนด์คู่ที่สองมีความถี่เท่ากับ $f_o \pm 2f_m$ แบนด์วิดท์ของคลื่น FM ต้องครอบคลุมจำนวนไซด์แบนด์ที่สำคัญทุกตัว นั้นคือแบนด์วิดท์ขึ้นกับดัชนีการมอดูเลตและความถี่ของสัญญาณที่เข้ามามอดูเลต แต่ดัชนีการมอดูเลตเท่ากับ f_d / f_m

ดังนั้นถ้าเราทราบความถี่เบี่ยงเบนและความถี่ของสัญญาณมอดูเลตเราก็สามารถคำนวณหาแบนด์วิดท์ได้ ตัวอย่างเช่นความถี่ของสัญญาณเสียงที่เข้ามามอดูเลตเท่ากับ 3 kHz ความถี่เบี่ยงเบนเท่ากับ 18 kHz เราสามารถคำนวณค่าดัชนีการมอดูเลตได้ดังนี้

$$m = \frac{f_d}{f_m}$$
$$= \frac{18kHz}{3kHz} = 6$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

นำค่า m = 6 ไปหาไซด์แบนด์สำคัญที่พิจารณาได้จากตารางในรูปที่ 2.1 จะเห็นว่าเมื่อดัชนีการมอดูเลตเท่ากับ 6 จำนวนไซด์แบนด์จะมีอยู่ 9 คู่ เราจึงคำนวณหาแบนด์วิดท์ได้ดังนี้

$$BW = f_m \times \text{จำนวนไซด์แบนด์} \times 2$$

$$= 3kHz \times 9 \times 2$$

$$= 54kHz$$

ในความจริงแล้วในทางปฏิบัตินิยมใช้สูตรคำนวณแบบประมาณค่า $f_{d\max}$ และ $f_{m\max}$ เลยไม่ต้องเสียเวลานับจำนวนไซด์แบนด์ ดังนี้

$$BW = 2(m+1)f_{m\max}$$

หรือ $$BW = 2(f_{d\max} + f_{m\max}) \quad \text{เมื่อ} \quad m = \frac{f_{d\max}}{f_{m\max}}$$

จากตัวอย่างดังกล่าวเราคำนวณได้ว่า

$$BW = 2(6+1) \times 3$$

$$= 42kHz$$

หรือ $$BW = 2(18+3)$$

$$= 42kHz$$

เสมือนกับว่าเราพิจารณาใช้จำนวนไซด์แบนด์เพียง 7 คู่ เมื่อเทียบกับการคำนวณในตอนต้น

2.7 โพล และ ซีเลคท์

ในการเชื่อมต่อแบบหลายจุด ระหว่างสถานีปฐมภูมิ และ สถานีทุติยภูมิมากกว่าหนึ่งตัวนั้น โอกาสที่จะเกิดความผิดพลาด เนื่องจากการส่งข้อมูลพร้อมกันนั้นมีสูง ดังนั้นเพื่อหลีกเลี่ยงปัญหานี้ ในโครงงานนี้จึงใช้วิธีการ โพล และ ซีเลคท์

โพล (Poll) คือการที่สถานีปฐมภูมิสั่งการร้องขอข้อมูลจากสถานีทุติยภูมิ

ซีเลคท์ (Select) คือการที่สถานีปฐมภูมิเลือกที่จะติดต่อกับสถานีทุติยภูมิ เมื่อมันมีข้อมูลที่จะส่งและเป็นการบอกแก่สถานีทุติยภูมิว่าข้อมูลกำลังมาถึง

รูปที่ 2.19 เป็นการอธิบายหลักการเหล่านี้ ในรูปทางซ้าย สถานีปฐมภูมิทำการโพลสถานีทุติยภูมิ โดยย่อให้ในกรณีนี้ สถานีทุติยภูมิไม่มีอะไรที่จะส่งก็ตอบรับด้วยข่าวสาร (nak) เวลาทั้งหมดที่แสดงลำดับในการทำงาน คือ

$$T_N = t_{prop} + t_{poll} + t_{proc} + t_{nak} + t_{prop}$$

โดยที่

T_N คือ เวลาที่ใช้ทั้งหมดในการโพลสถานีปลายทาง โดยไม่มีการส่งข้อมูล

(total time to poll terminal with nothing to send)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

t_{prop} คือ เวลาที่ข้อมูลใช้ในการเดินทาง

(propagation time = $t_1 - t_0, t_5 - t_4$)

t_{poll} คือ เวลาที่ใช้ในการส่งโพล

(time to tranmit a poll = $t_2 - t_1$)

t_{proc} คือ เวลาที่จัดการกับโพลก่อนการตอบรับ

(time to process poll before acknowledging = $t_3 - t_2$)

t_{nak} คือ เวลาที่ใช้ส่งการรับรู้เชิงปฏิเสธ

(time to transmit a negative acknowledging = $t_4 - t_5$)

รูปทางขวาแสดงกรณีที่โพลทำงานได้สำเร็จ เวลาที่ใช้ทั้งหมดก็คือ

$$T_p = t_{prop} + t_{poll} + t_{ack} + t_{data} + 2t_{proc}$$
$$= t_N + t_{prop} + t_{data} + 2t_{proc}$$

โดยเราสมมติว่าเวลาของขบวนการที่ตอบสนองต่อข่าวสารใดๆคงที่รูปแบบที่เป็นที่รู้จักกันมากที่สุดของการโพลก็คือ roll-call polling โดยที่สถานีปฐมภูมิเลือกโพลสถานีทุติยภูมิและสถานี โดยการตรวจสอบตามลำดับ ในกรณีที่ง่ายที่สุด สถานีปฐมภูมิจะเลือกทุกๆสถานีทุติยภูมิในลำดับวนรอบ $S_1, S_2, ..., S_n$ สำหรับทุกค่าของ n ของสถานีทุติยภูมิ และทำซ้ำเป็นลำดับ โดยในโครงงานนี้ได้เลือกใช้การโพลลักษณะนี้ และรูปแบบของเวลาสามารถแสดงได้ดังนี้

$$T_c = nT_N + kT_D$$

โดยที่

T_c = เวลาในการโพล 1 รอบ ที่สำเร็จสมบูรณ์

(time for one complete polling cycle)

T_N = เวลาเฉลี่ยเพื่อที่จะโพลการส่งถ่ายข้อมูลทุติยภูมิ

(average time to poll a secondary exclusive of data transfer)

T_D = เวลาที่ส่งถ่ายข้อมูล

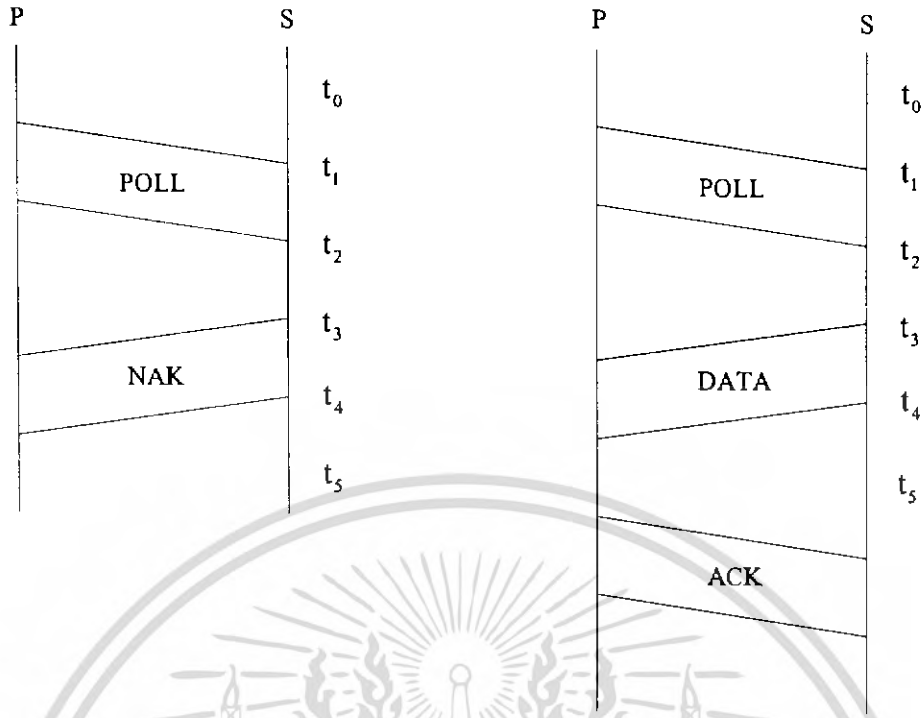
(time to transfer data = $t_{prop} + t_{data} + t_{proc}$)

n = จำนวนของสถานีทุติยภูมิ

(number of Secondary station)

k = จำนวนของสถานีทุติยภูมิที่ส่งข้อมูลระหว่างรอบปฏิบัติการนั้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



(ก) poll แล้วไม่มีการส่งผ่านข้อมูล (ข) poll แล้วมีการส่งผ่านข้อมูล

รูปที่ 2.19 แสดงการทำงานของโพลและซีเลคท์

2.8 แพ็คเกตเรดิโอ

การสื่อสารแบบไร้สาย เป็นรูปแบบที่ให้ความสะดวกสบายในการใช้งานมากกว่าแบบมีสายเป็นอย่างมากซึ่งระบบการสื่อสารแบบไร้สายที่ใช้ในปัจจุบันนี้ จะมีลักษณะการส่งข้อมูลอยู่ 2 แบบ คือ แบบอนาล็อกที่มีการส่งสัญญาณแบบต่อเนื่องตลอดคาบเวลา และแบบดิจิตอล ที่สัญญาณมีลักษณะเปิด – ปิด (0 กับ 1) แพ็คเกตเรดิโอก็เป็นการสื่อสารแบบไร้สายแบบหนึ่งที่มีการส่งสัญญาณเป็นแบบอนาล็อก

2.8.1 ประวัติความเป็นมาของแพ็คเกตเรดิโอ

แพ็คเกตเรดิโอเกิดขึ้นต้นปี 1978 ที่ประเทศแคนาดาโดยกรมการสื่อสารของแคนาดาได้ประกาศอนุญาตให้นักวิทยุสมัครเล่นแคนาดา ได้ใช้แพ็คเกตเรดิโออย่างเป็นทางการโดยที่ไม่มีการร้องขอแต่อย่างใด ซึ่งเป็นการจุดประกายความคิดให้กับนักวิทยุสมัครเล่นแคนาดา ให้ออกแบบเครื่องมืออุปกรณ์สำหรับการติดต่อสื่อสารในระบบนี้ และหลังจากนั้นก็ได้แพร่หลายเข้ามาในประเทศสหรัฐอเมริกาโดย FCC (Federal Communication Commission) ซึ่งเป็นหน่วยงานที่ควบคุมดูแลการใช้ความถี่ของประเทศสหรัฐอเมริกา ได้ประกาศให้นักวิทยุสมัครเล่นในประเทศสามารถใช้การติดต่อสื่อสารในระบบของแพ็คเกตเรดิโอได้ ทำให้เกิดการขยายระบบเครือข่ายของแพ็คเกตเรดิโอขึ้นอย่างรวดเร็วในประเทศสหรัฐอเมริกา และต่อมาก็ได้แพร่หลายไปทั่วโลกอย่างรวดเร็ว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.8.2 แพ็กเกตเรดิโอคืออะไร

แพ็คเกตเรดิโอ คือ การติดต่อสื่อสารส่งข้อมูลด้วยเครื่องคอมพิวเตอร์ซึ่งส่งผ่านข้อมูลทีละชุด หรือเรียกว่าทีละแพ็คเกต โดยใช้ความถี่วิทยุสมัครเล่นทุกย่านความถี่คือ HF,VHF,UHF,SHF ผ่านเครื่องวิทยุสมัครเล่น โดยมีอุปกรณ์ตัวกลางที่เข้ามาทำหน้าที่แปลงสัญญาณเรียกว่า TNC (Terminal Node Controller) จะทำหน้าที่แปลงสัญญาณจากคอมพิวเตอร์ที่เป็นสัญญาณดิจิตอลให้เป็นสัญญาณอนาล็อก แล้วส่งออกอากาศผ่านทางวิทยุสมัครเล่นไปยังคู่สถานี ก็จะรับสัญญาณอนาล็อกผ่านทาง TNC แล้วแปลงกลับมาเป็นสัญญาณดิจิตอลตามเดิม

2.8.3 แพ็กเกตโมเด็ม

เป็นอุปกรณ์ตัวหนึ่งที่ถูกสร้างขึ้นมาเพื่อใช้ในการติดต่อแบบแพ็คเกตเรดิโอ มีหน้าที่ในการแปลงสัญญาณจากดิจิตอลเป็นอนาล็อก และอนาล็อกไปเป็นดิจิตอล เหมือนกับโมเด็มทั่ว ๆ ไป แต่ลักษณะการทำงานของแพ็คเกตโมเด็มจะต่างออกไปเล็กน้อย

หน้าที่หลักของแพ็คเกตโมเด็มคือการเปลี่ยนข้อมูลที่เป็นสัญญาณดิจิตอลให้เป็นอนาล็อกและเปลี่ยนกลับ โดยใช้มาตรฐาน BELL 202 หรือ CCITT โดยให้ "0" เป็น 2200Hz และให้ "1" เป็น 1200Hz

2.9 หลักการของระบบบัส I^2C

2.9.1 ความรู้เบื้องต้นเกี่ยวกับ I^2C

I^2C ย่อมาจาก Inter-IC Communication หมายถึง การติดต่อสื่อสารระหว่างไอซีโดยบัส I^2C ได้รับการพัฒนาขึ้นโดยฟิลิปส์ (Philips) ด้วยจุดมุ่งหมายหลัก คือ ต้องการให้ไอซีหรือโมดูลสามารถติดต่อ สั่งงาน และควบคุมภายใต้สัญญาณเพียง 2 เส้น เส้นหนึ่ง คือ สายข้อมูล อีกเส้นหนึ่ง คือ สายสัญญาณนาฬิกา ที่ใช้ในการกำหนดจังหวะการทำงาน การต่อรวมกันของอุปกรณ์บนบัส I^2C ทำได้ง่ายมาก เพียงต่อสายข้อมูลและสายสัญญาณของอุปกรณ์แต่ละตัวขนานหรือพ่วงกันไป ส่วนการกำหนดแอดเดรสหรือตำแหน่งสำหรับติดต่ออุปกรณ์แต่ละตัว จะใช้รหัสข้อมูลและการกำหนดสภาวะลอจิกที่ขาแอดเดรสของอุปกรณ์แต่ละตัว

สายข้อมูลบนบัส I^2C มีชื่อเรียกอย่างเป็นทางการว่า สายข้อมูลอนุกรมหรือ SDA (Serial Data Line) ส่วนสายสัญญาณนาฬิกาอนุกรม หรือ SCL (Serial Clock Line)

อุปกรณ์ที่ทำการเชื่อมต่อบนบัส I^2C มีหลากหลาย ไม่ว่าจะเป็นไอซีขยายพอร์ตอินพุตเอาท์พุต (I/O Expander), ไอซีแปลงสัญญาณอนาลอกเป็นดิจิตอล (ADC) และแปลงดิจิตอลเป็นอนาล็อก (DAC), ไอซีรีลไทม์คล็อก (RTC), ไอซีขับโมดูลLCD, หน่วยความจำอีอีพรอมและไมโครคอนโทรลเลอร์

2.9.2 คุณสมบัติโดยทั่วไปของบัส I^2C

สาย SDA และ SCL เป็นสายสัญญาณ 2 ทิศทาง (bi-directional line) ต้องมีการต่อตัวต้านทานพูลอัพกับแรงดัน +5V ไว้ตลอดเวลา เพื่อให้สายมีสถานะลอจิกสูงในขณะที่ไม่มีการติดต่อใช้งาน ทั้งยัง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ช่วยในการป้องกันสัญญาณรบกวนที่อาจมีเข้ามาในสายสัญญาณทั้งสองวงจรเอาท์พุทของอุปกรณ์ที่อยู่บนบัส I^2C ต้องมีลักษณะเป็นวงจรเดรนเปิด (open-drain) หรือคอลเล็กเตอร์เปิด (open-collector) อัตราการถ่ายทอดข้อมูลบนบัส I^2C สูงถึง 100 กิโลบิตต่อวินาทีในโหมดปกติ (standard mode) และสูงถึง 400 กิโลบิตต่อวินาทีในโหมดความเร็วสูง (fast mode) อุปกรณ์ที่ต่อร่วมอยู่บนบัส I^2C จะต้องมีค่าความจุไฟฟ้ารวมที่เกิดขึ้นระหว่างสาย SDA และ SCL ไม่เกิน 400 pF การเข้าถึงอุปกรณ์บนบัส I^2C ใช้ข้อมูลการเข้าถึง 2 ค่า คือ 7 บิต(7-bit addressing) หรือ 10 บิต (10-bit addressing)
ข้อเด่นอีกประการหนึ่งของบัส I^2C คือ สามารถเชื่อมต่ออุปกรณ์ที่ใช้ไฟเลี้ยงไม่เท่ากันให้สามารถติดต่อสื่อสารกันได้ โดยอุปกรณ์บนบัส I^2C ตัวหนึ่งอาจใช้ไฟเลี้ยง +5V ในขณะที่อีกตัวหนึ่งใช้ไฟเลี้ยง +12V การต่อรวมกันบนบัส I^2C สามารถกระทำได้ในลักษณะเดียวกันกับกรณีที่อุปกรณ์ทั้งสองใช้ไฟเลี้ยงเท่ากัน กล่าวคือ ให้ต่อสาย SDA และ SCL ของอุปกรณ์แต่ละตัวเข้าด้วยกัน และต้องต่อตัวต้านทานพลูอัพ(R_p) เข้ากับแรงดัน +5V ไว้ด้วยเสมอ
ในกรณีที่อาจมีแรงดันไฟกระชากขนาดใหญ่ปะปนเข้ามาในบัส I^2C ที่ขา SDA และ SCL ของอุปกรณ์แต่ละตัว ต้องต่อตัวต้านทานอนุกรมกับขา SDA และ SCL เรียก R_p ก่อนต่อเข้าสู่บัส I^2C

2.9.3 หลักการของบัส I^2C

บัส I^2C ประกอบด้วยสายสัญญาณ 2 เส้น ดังที่ได้กล่าวมาแล้ว คือ SDA และ SCL อุปกรณ์ที่ต่อพ่วงบนบัสสามารถมีได้มากมาย ดังนั้น จึงต้องมีการกำหนดรูปแบบของการติดต่อบนบัส หรือเรียกว่า โปรโตคอล (protocal) เพื่อให้ผู้ใช้สามารถทราบได้ว่า ขณะนี้อุปกรณ์ใดติดต่อกันอยู่ และอุปกรณ์ใดเป็นตัวรับตัวส่ง ต่อไปนี้จะอธิบายลักษณะ หน้าที่ และนิยามของอุปกรณ์ที่ต่ออยู่บนบัส I^2C เพื่อเป็นข้อตกลงพื้นฐานก่อนที่จะอธิบายการทำงานของบัส I^2C ต่อไป
อุปกรณ์ที่เป็นผู้สร้างข้อมูลหรือส่งข้อมูล เรียกว่า ตัวส่ง (transmitter)
อุปกรณ์ที่เป็นผู้รับข้อมูล เรียกว่า ตัวรับ (receiver) ในอุปกรณ์บนบัส I^2C สามารถเป็นได้ทั้งตัวรับและตัวส่ง บางอุปกรณ์ทำหน้าที่เป็นตัวรับเพียงอย่างเดียว จะไม่มีอุปกรณ์ใดบนบัส I^2C ที่ทำหน้าที่เป็นตัวส่งเพียงอย่างเดียว
อุปกรณ์ที่ทำหน้าที่ควบคุมจังหวะการติดต่อบนบัส I^2C เรียกว่า มาสเตอร์ (master)
อุปกรณ์ที่ถูกควบคุมหรืออุปกรณ์ที่ต่อพ่วงเข้าไปบนบัส I^2C เรียกว่า สเลฟ (slave)

ข้อกำหนด 2 ประการสำคัญของการติดต่อบนบัส I^2C คือ

1. การถ่ายทอดข้อมูลจะเกิดขึ้นได้เมื่อบัสว่างเท่านั้น
2. ในระหว่างการถ่ายทอดข้อมูล เมื่อใดก็ตามที่สาย SCL มีสถานะเป็นลอจิกสูง สายข้อมูลต้องรักษาข้อมูลไว้ อย่าให้เกิดการเปลี่ยนแปลงขึ้นเด็ดขาด มิฉะนั้น สัญญาณที่เกิดขึ้นจะได้รับการแปลความหมายเป็นสัญญาณควบคุมแทน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.9.4 สภาวะที่เกิดขึ้นบนบัส I^2C

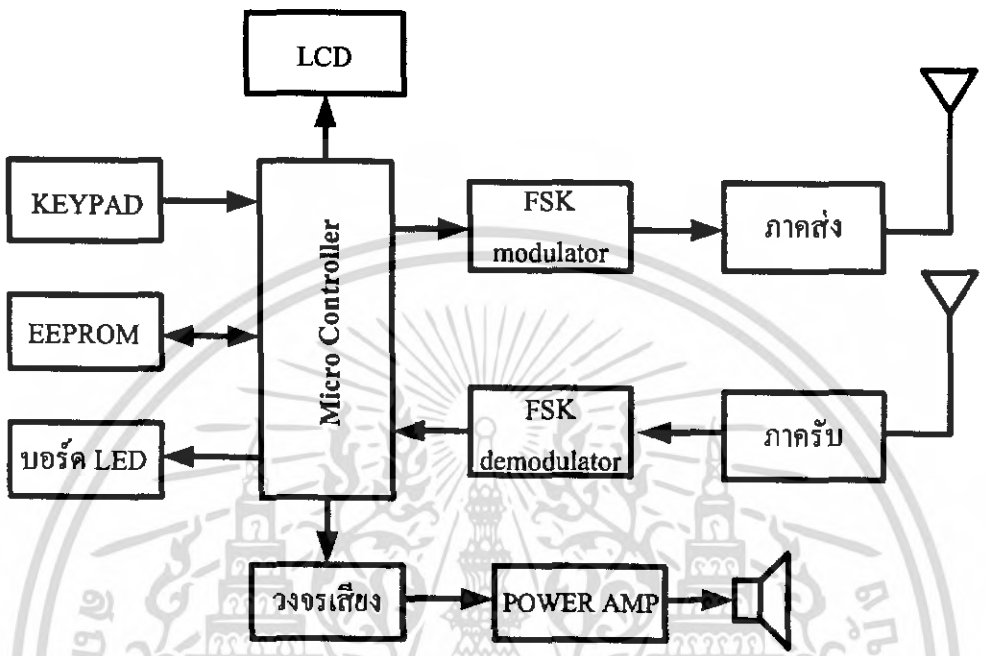
มีด้วยกัน 5 สภาวะ ดังนี้

1. บัสว่าง (Bus not busy) สภาวะนี้เกิดขึ้นเมื่อสถานะลอจิกบนสาย SDA และ SCL เป็นลอจิกสูงทั้งคู่ นั่นหมายความว่าการถ่ายทอดข้อมูลสามารถเริ่มต้นขึ้นได้
2. เริ่มต้นการถ่ายทอดข้อมูล (Start data transfer) เกิดขึ้นเมื่อสาย SDA มีการเปลี่ยนแปลงระดับลอจิกจากสูงไปต่ำ ในขณะที่สาย SCL มีสถานะลอจิกสูง เรียกสภาวะที่เกิดขึ้นนี้ว่า สภาวะเริ่มต้น (START)
3. หยุดการถ่ายทอดข้อมูล (Stop data transfer) เกิดขึ้นเมื่อสาย SDA มีการเปลี่ยนแปลงระดับลอจิกจากต่ำไปสูง ในขณะที่สาย SCL มีสถานะลอจิกสูง เรียกสภาวะที่เกิดขึ้นนี้ว่า สภาวะหยุด (STOP)
4. ข้อมูลดำรงอยู่บนบัส (Data valid) สภาวะนี้เกิดขึ้นถัดจากสภาวะเริ่มต้น โดยสถานะลอจิกที่เกิดขึ้นบนสาย SDA คือ ข้อมูลที่ทำการถ่ายทอดเมื่อสาย SCL เป็นลอจิกสูง สถานะที่สาย SDA ต้องคงที่ เพื่อให้อุปกรณ์รับรู้ข้อมูลในจังหวะนั้นว่าเป็น 0 หรือ 1 ข้อมูลอาจเกิดการเปลี่ยนแปลงได้ ในขณะที่สาย SCL เป็นลอจิกต่ำ แต่เมื่อใดก็ตามที่ต้องการให้เกิดการถ่ายทอดข้อมูลอย่างสมบูรณ์ สถานะลอจิกที่ขา SDA ต้องคงที่ช่วงเวลาที่สาย SCL มีสถานะลอจิกสูง หากมีการเปลี่ยนแปลงสถานะลอจิกในขณะที่สาย SCL มีลอจิกสูงอยู่นั้น อุปกรณ์มาสเตอร์ที่ทำการควบคุมการถ่ายทอดข้อมูลจะแปลความหมายเป็นสภาวะหยุด หรือสภาวะเริ่มต้นก็ได้ ทำให้ข้อมูลที่ทำการถ่ายทอดนั้นเกิดความผิดพลาดขึ้น
5. รับรู้ข้อมูล (Acknowledge) เกิดขึ้นหลังจากที่การถ่ายทอดข้อมูลจากตัวส่งมายังตัวรับเกิดขึ้นอย่างสมบูรณ์ โดยตัวส่งจะทำการส่งข้อมูลมา 1 บิต เรียกว่า บิตรับรู้ (Acknowledge) มีสถานะเป็นลอจิกสูง หลังจากส่งข้อมูลมาครบถ้วน ส่วนอุปกรณ์มาสเตอร์จะทำการส่งสัญญาณรับรู้พิเศษซึ่งสัมพันธ์กับสัญญาณนาฬิกา เพื่อตอบสนองบิตรับรู้ที่ส่งมาจากตัวส่ง ทางด้านตัวรับจะส่งบิตรับรู้ที่มีสถานะลอจิกต่ำลงบนบัส อุปกรณ์สเลฟที่ถูกอ้างถึงในการติดต่อ หรือกำลังติดต่ออยู่ในขณะนั้น ก็จะกำเนิดบิตรับรู้ เพื่อตอบสนองให้ทราบว่าได้รับข้อมูลในแต่ละไบต์เรียบร้อยแล้ว

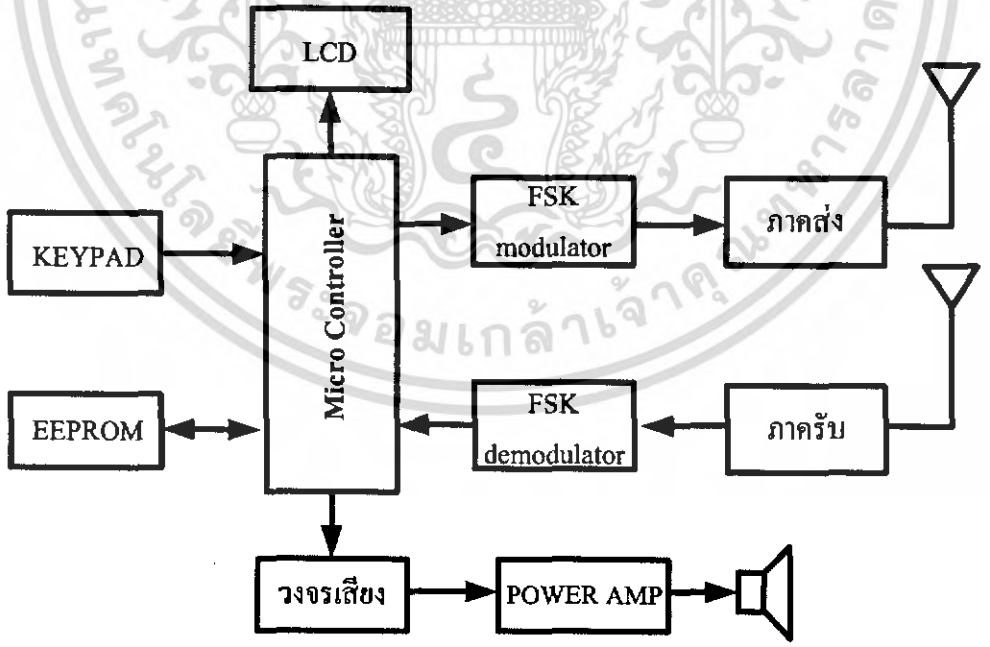
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 3
การคำนวณและการสร้าง

3.1 ส่วนประกอบของโครงงาน



รูปที่ 3.1 แสดง Block diagram ในส่วนของป้ายรถประจำทาง



รูปที่ 3.2 แสดง Block diagram ในส่วนของรถประจำทาง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากรูปที่ 3.1 จะแสดงบล็อกไดอะแกรม (Block diagram) ในส่วนของป้ายรถประจำทางซึ่งจะเห็นได้ว่ามีไมโครคอนโทรเลอร์เป็นหน่วยประมวลผลกลาง (CPU) ซึ่งจะทำหน้าที่ติดต่อกับอุปกรณ์ต่าง ๆ ที่นำมาต่อร่วมกันคือ คีย์แพด (Key pad) , จอแสดงข้อความ (LCD) , ส่วนเก็บข้อมูล (EEPROM) , บอร์ดแสดงหมายเลข , วงจรบันทึกเสียง , วงจรมอดูเลตแบบ FSK , วงจรดีมอดูเลตแบบ FSK และในรูปที่ 3.2 แสดงบล็อกไดอะแกรมในส่วนของรถประจำทาง ซึ่งจะมีลักษณะคล้ายกับบล็อกไดอะแกรมของป้ายรถประจำทาง โดยจะต่างกันเพียงแค่ของรถประจำทางจะไม่มีในส่วนของการแสดงผลทางตัวเลข ซึ่งในการออกแบบและหลักการทำงานของวงจรต่าง ๆ สามารถอธิบายได้ดังนี้

ในระบบจะมีส่วนเก็บข้อมูล (EEPROM) เอาไว้สำหรับเก็บข้อมูล โดยในส่วนของป้ายประจำทางจะใช้ในการเก็บรหัสประจำเส้นทางว่าไปหรือกลับ หมายเลขป้ายรถประจำทาง และหมายเลขรถประจำทางสายที่ผ่านป้ายนั้น ๆ และในส่วนของรถประจำทางจะใช้ในการเก็บรหัสประจำเส้นทางว่าไปหรือกลับ หมายเลขรถ ID ประจำตัวของรถแต่ละคัน เพื่อที่จะทำการตรวจสอบรถประจำทางที่ผ่านและส่งข้อมูลของป้ายไปยังรถประจำทางแล้วให้รถประจำทางส่งกลับ โดยครั้งแรกที่ทำการเปิดเครื่องเราจะต้องทำการใส่ข้อมูลก่อน ในส่วนของ คีย์แพด จะเป็นแบบ 4 ×4 จะเป็นส่วนที่ใช้ในการป้อนค่าเข้าไปตามที่ต้องการ โดยเมื่อทำการป้อนค่าเข้าไปจะไปทำการกำหนดค่าต่าง ๆ โดยดึงเอาโปรแกรมต่าง ๆ ที่ถูกเขียนอยู่ในไมโครคอนโทรเลอร์ออกมาใช้ กล่าวคือจะใช้ในการเลือกเมนูการทำงานต่าง ๆ การใส่ข้อมูลต่าง ๆ ลงในส่วนเก็บข้อมูล ในส่วนของจอแสดงผล (LCD) จะเป็นส่วนในการแสดงข้อความต่าง ๆ ให้สามารถดำเนินการในแต่ละส่วนได้อย่างถูกต้อง ในส่วนของบอร์ดแสดงหมายเลข จะมีไว้สำหรับแสดงหมายเลขของรถประจำทางที่จะมาถึง ซึ่งจะใช้เป็นหลอด LED มาเรียงต่อกันเป็นตัวเลขเพื่อให้สามารถเห็นได้อย่างชัดเจน ในส่วนของวงจรเสียง ในวงจรนี้ได้ทำการบันทึกเสียงต่าง ๆ ไว้แล้ว โดยเมื่อรถประจำทางสายใดเข้ามาก็จะแสดงเสียงที่ได้บันทึกไว้แล้วออกมาโดยนำแต่ละเสียงมาเรียงต่อกันเพื่อให้ได้เป็นหมายเลขของรถประจำทาง ส่วนของวงจรมอดูเลตแบบ FSK มีไว้เพื่อแปลงสัญญาณที่ได้รับจากไมโครคอนโทรเลอร์ให้อยู่ในรูปอนาล็อกแล้วส่งต่อไปยังภาคส่งซึ่งจะใช้วิธีการมอดดูเลตแบบ FM แล้วส่งออกไปยังอากาศ โดยภาครับและภาคส่งในโครงงานนี้จะวิทยุสื่อสารเป็นตัวรับส่งข้อมูล โดยที่สัญญาณที่มาจากไมโครคอนโทรเลอร์จะเป็นข้อมูลต่าง ๆ ที่ได้เก็บบันทึกไว้ในส่วนเก็บข้อมูลหรือ EEPROM ในส่วนของวงจรดีมอดูเลตแบบ FSK มีไว้เพื่อรับสัญญาณที่มาจากภาครับสัญญาณ FM แล้วนำมาถอดรหัสเป็นสัญญาณดิจิตอลแล้วทำการส่งไปยังไมโครคอนโทรเลอร์เพื่อทำการประมวลผลต่อไป

3.2 หลักการทำงาน

โดยหลักการทำงานของโครงงานนี้คือ โดยเริ่มต้นจากที่ป้ายรถประจำทางจะทำการส่งข้อมูลในการตรวจสอบรถประจำทางแต่ละคันออกไป โดยที่ข้อมูลที่ส่งออกไปนั้นจะประกอบด้วยข้อมูลจำนวน 4 ไบต์ โดยที่ข้อมูลในไบต์แรกจะเป็น Header ไบต์ที่ 2 จะเป็นข้อมูลของรหัสผู้ส่งและรหัสเส้นทาง ไบต์ที่ 3 เป็นข้อมูลของหมายเลขรถที่ต้องการตรวจสอบ ไบต์ที่ 4 เป็นหมายเลขของป้ายที่ทำการตรวจสอบ และถ้าหากมีรถประจำทางเข้ามาในรัศมีของเครื่องส่ง รถคันนั้นก็จะทำการรับข้อมูลที่ส่งจากป้ายเข้ามาตรวจสอบดูว่าตรงกับข้อมูลของตัวเองหรือไม่ ถ้าหากว่าตรงก็จะทำการแสดงผลเป็นเสียงของหมายเลข

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

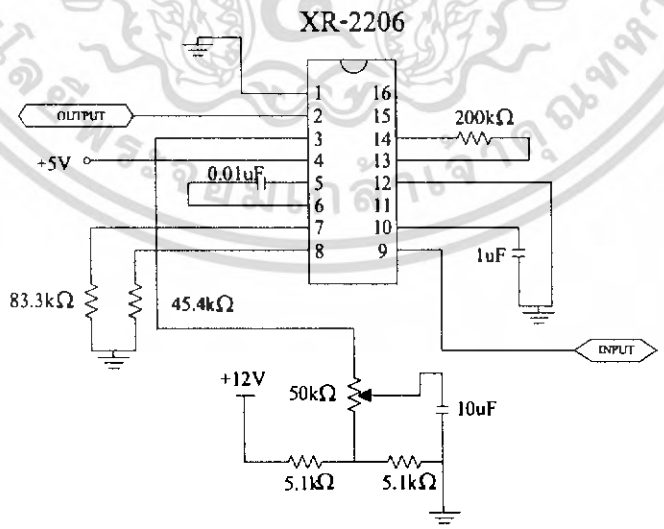
ของป้ายนั้นที่รถและทำการส่งข้อมูลตอบกลับไปยังป้ายโดยข้อมูลที่ทำการส่งกลับไปนั้นจะประกอบด้วยข้อมูลจำนวน 4 ไบต์ โดยที่ไบต์แรกจะเป็น Header ไบต์ที่ 2 จะเป็นข้อมูลของรหัสผู้ส่งและรหัสเส้นทาง ไบต์ที่ 3 จะเป็นข้อมูลของหมายเลขรถที่ทำการตอบกลับ ไบต์ที่ 4 จะเป็นข้อมูลของหมายเลข ID ของรถที่ทำการตอบกลับ และเมื่อทางป้ายรับข้อมูลที่ทำการตอบกลับไปแล้วก็จะทำการตรวจสอบดูว่าข้อมูลนั้นตรงกับข้อมูลที่ได้ทำการเรียกหรือไม่ ถ้าหากว่าตรงก็จะทำการแสดงผลออกมาในรูปของตัวเลขบนบอร์ด LED และ ข้อความเสียง

และถ้าหากที่ป้ายรถประจำทางมีผู้ต้องการที่จะขึ้นรถประจำทางสายที่แสดงผลอยู่ก็สามารถทำการติดต่อกับรถคันนั้นได้โดยให้ป้ายทำการส่งข้อมูลที่เป็นข้อมูลในการเรียกรถคันนั้นออกไป ซึ่งเมื่อทางด้านรถประจำทางรับข้อมูลนี้ได้ก็จะทำการตรวจดูว่าข้อมูลนี้เป็นของตัวเองและเป็นข้อมูลในการเรียกรถหรือไม่ ถ้าหากว่าตรงก็จะทำการแสดงผลให้คนขับรู้ว่ามีผู้โดยสารต้องการขึ้นที่ป้ายหน้า

3.3 การแปลงสัญญาณ FSK

3.3.1 การออกแบบวงจรเข้ารหัส (FSK Modulation)

วงจรการเข้ารหัส (FSK Modulation) ในโครงงานได้เลือกใช้ไอซีเบอร์ XR-2206 ไอซีเบอร์นี้ใช้สำหรับการเปลี่ยนสัญญาณจากสัญญาณดิจิตอลให้เป็นสัญญาณอนาลอก ไอซีเบอร์ XR-2206 ผลิตสัญญาณโดยใช้ไบนารี่ FSK ความถี่จะถูกชิพโดยอินพุทข้อมูล เอาท์พุทของ FSK Modulate จะอยู่ในสเต็ปฟังก์ชั่น ของโดเมนทางความถี่ อินพุทของวงจรจะถูกกำหนดโดยลอจิก “0” และ “1” ส่วนเอาท์พุท FSK จะเปลี่ยนแปลง 2 ความถี่ มาร์ค คือ ลอกจิก “1” และสเปซ คือลอจิก “0” ซึ่งไบนารี FSK จะเปลี่ยนเอาท์พุทตามอินพุทที่เข้ามา ในหลักการมอดูเลตแบบดิจิตอลการเปลี่ยนแปลงของอินพุทที่เข้ามาที่นั่น เรียกว่า “บิทเรต” ซึ่งมีหน่วยเป็นบิทต่อวินาที (Bit per Second)



รูปที่ 3.3 วงจรเข้ารหัส (FSK Modulation)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การออกแบบวงจร FSK Modulation ใช้ไอซีเบอร์ XR-2206 ดังรูปที่ 3.3 สามารถทำงานได้ 2 ความถี่ โดยการกำหนดค่า R_1 และ R_2 ที่ขา 7 และขา 8 ของไอซีเบอร์ XR-2206 ดังรูปที่ 3.3 และขา 9 เปิดวงจร หรือทำการไบอัสแรงดันมากกว่าหรือเท่ากับ $2V$ R_1 จะทำงานถ้าแรงดันที่ระดับขา 9 น้อยกว่าหรือเท่ากับ $1V$ R_2 จะทำงานดังนั้น เอาท์พุทของไอซีเบอร์ XR-2206 สามารถเปลี่ยนระดับความถี่ได้ 2 ความถี่ คือ f_1 กับ f_2

วิธีหาค่า

1. หาความถี่ $f_1 = 1,200Hz$, กำหนดให้ $C_1 = 0.01\mu F$

$$f_1 = \frac{1}{C_1 R_1}$$

$$R_1 = \frac{1}{(1200)x(0.01x10^{-6})}$$

$$R_1 = 83.3K\Omega$$

2. หาความถี่ $f_2 = 2,200Hz$, กำหนดให้ $C_2 = 0.01\mu F$

$$f_2 = \frac{1}{C_2 R_2}$$

$$R_2 = \frac{1}{(2200)x(0.01x10^{-6})}$$

$$R_2 = 45.4K\Omega$$

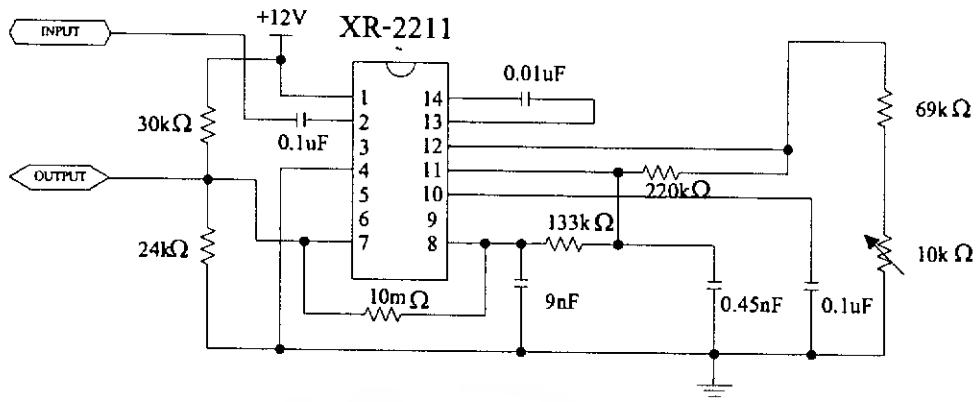
3.3.2 การออกแบบวงจรถอดรหัส (FSK Demodulation)

หลักการของการดีมอดูเลตคือการกรองหรือการแยกสัญญาณข้อมูลข่าวสารออกจากคลื่นพาห์ที่จะนำสัญญาณที่ได้ไปใช้งานต่อ

ภาครับสัญญาณ FSK นี้จะรับสัญญาณ RF โดยที่ภาคนี้จะทำการแปลงสัญญาณ FSK ที่มี 2 ความถี่ ให้กลายเป็นสัญญาณดิจิตอล "0" (แทนแรงดันประมาณ $0V$) และ "1" (แทนด้วยแรงดันประมาณ $5V$) โดยที่ไอซีเบอร์ XR-2211 นี้ถูกออกแบบมาเพื่อทำการแปลงสัญญาณเสียง (Tone Encoding) การดีเทคสัญญาณ FM การดีเทคสัญญาณคลื่นพาห์ (Carrier Detector)

การออกแบบวงจร FSK Demodulation ใช้ไอซีเบอร์ XR-2211 นำมาใช้สำหรับการเปลี่ยนสัญญาณอนาลอกกลับมาเป็นสัญญาณดิจิตอล ในการดีมอดูเลชั่น สัญญาณอนาลอกที่ส่งมาจากเครื่องส่งวิทยุให้กลับมาเป็นสัญญาณข้อมูลดิจิตอล ซึ่งสามารถดีเทคสัญญาณอนาลอกที่มีขนาด $2mV$ ถึง $3V$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.4 วงจรถอดรหัส (FSK Modulation)

วิธีการหาค่า

1. หาค่าความถี่กลางของเฟสล็อกลูป f_0 ดังสมการ โดยค่าที่ f_1 และ f_2 คือความถี่ทั้งสองของสัญญาณอินพุท FSK

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

ถ้าทางด้านส่งนั้นได้กำหนดความถี่มาตรฐานของ CCITT V.23 คือ ที่ บอดเรต 1200 บิทต่อวินาที ซึ่งมีความถี่ f_1 และ f_2 มีค่า $1,200Hz$ และ $2,200Hz$ ตามลำดับนั้นจะได้

$$f_0 = \sqrt{1200x2200}$$

$$= 1624.8Hz$$

2. เลือกค่าความต้านทาน R_0 ซึ่งค่า R_0 นี้ควรจะอยู่ในช่วง $10K\Omega$ จนถึง $100K\Omega$ สำหรับความต้านทานของ R_0 ที่ใช้ในโครงงานนี้มีค่า $70K\Omega$

3. หาค่า C_0

$$C_0 = \frac{1}{f_0 R_0}$$

$$C_0 = \frac{1}{(1624.8)x(70x10^3)}$$

$$C_0 = 9nF$$

4. หาค่าความต้านทาน R_1 จากสมการ

$$R_1 = \frac{2R_0 f_0}{\Delta f}$$

$$R_1 = \left(\frac{(70x10^3)x1624.8}{1200-2200}\right)x2$$

$$R_1 = 224K\Omega \approx 220K\Omega$$

5. หาค่าตัวเก็บประจุ C_1 เพื่อกำหนดลูปแดมปิ้ง (Damping Loop) ซึ่งควรมีค่าประมาณ 0.5 ดังนั้นจะได้ค่าตัวเก็บประจุ C_1 ดังสมการ

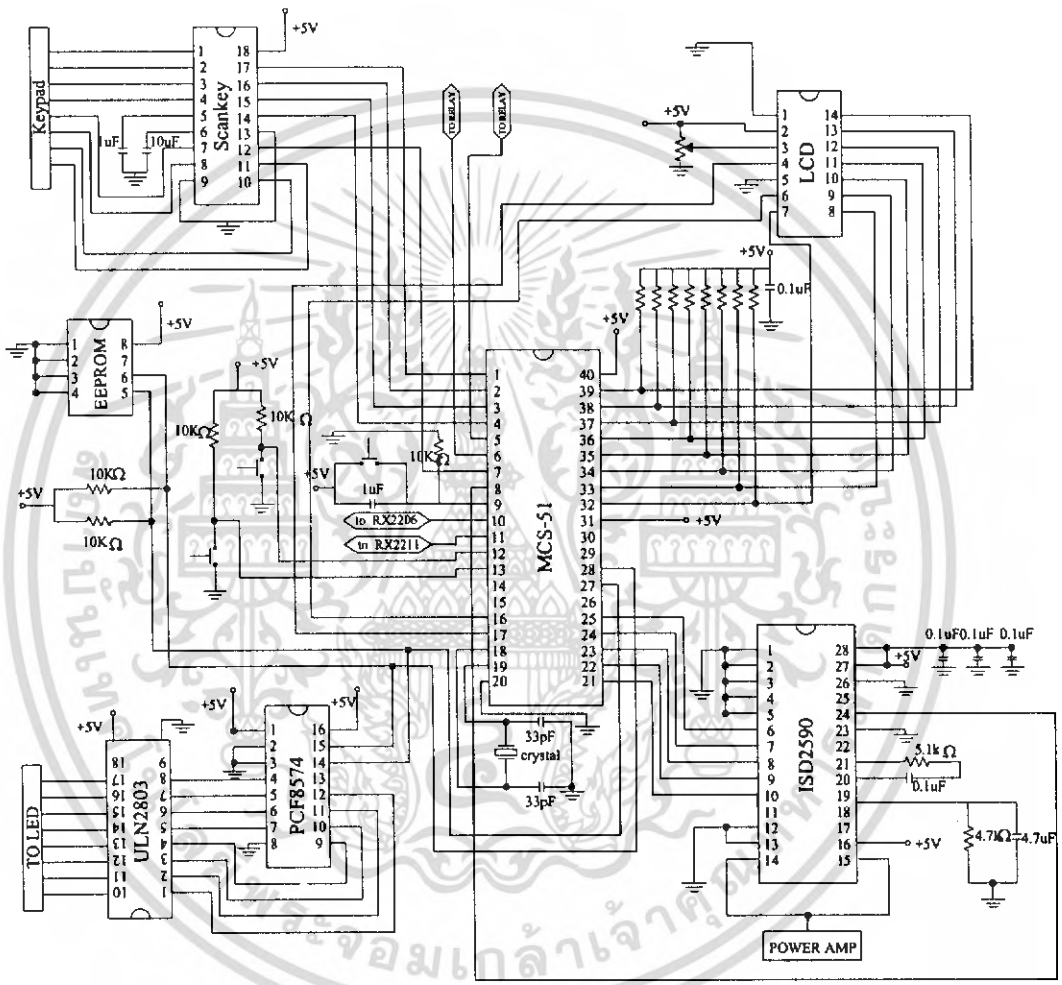
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$$C_1 = \frac{1250xC_0}{R_1x\xi^2} = \frac{1250x(9x10^{-9})}{(220x10^3)x(0.5)^2} = 0.45nF$$

$$C_1 = \frac{1250x(9x10^{-9})}{(220x10^3)x(0.5)^2}$$

$$C_1 = 0.45nF$$

3.4 การเชื่อมต่ออุปกรณ์ภายนอกต่าง ๆ กับไมโครคอนโทรเลอร์



รูปที่ 3.5 การเชื่อมต่ออุปกรณ์ภายนอกต่าง ๆ เข้ากับไมโครคอนโทรเลอร์

จากรูปที่ 3.5 แสดงการเชื่อมต่อคีย์แพด 4x4 กับไมโครคอนโทรลเลอร์ โดยจะมี IC Scankey เบอร์ 74C922 ซึ่งเป็น 16 Key encoder เป็นตัวเชื่อมต่อระหว่างคีย์แพด 4x4 กับไมโครคอนโทรลเลอร์ โดยที่เอาท์พุทของไอซีนี้จะมี 4 เส้น เพื่อต่อกับไมโครคอนโทรลเลอร์ดังรูปที่ 3.5 ข้อดีของ IC Scankey นี้คือจะช่วยลดความยุ่งยากในการเขียนโปรแกรม ทำให้โปรแกรมมีขนาดสั้นลง ไม่เปลือง Flash Memory ของไมโครคอนโทรลเลอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การเชื่อมต่อกับ LCD ในโครงงานนี้เราจะใช้ LCD แบบ16 ตัวอักษร 2 บรรทัด แบบ 8 บิตในการเลือกการต่อร่วมกับไมโครคอนโทรลเลอร์ มีให้เลือกทั้งแบบ 8 บิต และ 4 บิต ซึ่งจะมีข้อดีและข้อเสีย

1. ความเร็วในการแสดงผลแบบ 8 บิต จะเร็วกว่าแบบ 4 บิต
2. การใช้สายสัญญาณติดต่อกับไมโครคอนโทรลเลอร์แบบ 4 บิต จะใช้น้อยกว่าแบบ 8 บิต

ในส่วนของหน่วยความจำ ในโครงงานนี้เราจะใช้สำหรับเก็บข้อมูลต่าง ๆ ที่คีย์เข้าไป โดยในส่วนของป้ายจะใช้เก็บข้อมูลของ หมายเลขของป้าย และหมายเลขรถประจำทางที่ผ่านป้ายนั้น และในส่วนของตัวรถจะใช้เก็บข้อมูลของหมายเลขรถนั้น ๆ โดยในโครงงานนี้เราจะใช้ IC เบอร์ 24LC16 ซึ่งเป็นหน่วยความจำแบบนอน – โวลาไทล์ (non - volatile) คือสามารถเก็บข้อมูลอยู่ได้โดยไม่ต้องจ่ายไฟเลี้ยง สามารถเขียน – อ่าน – ลบได้ด้วยสัญญาณไฟฟ้า ซึ่งจะใช้การเชื่อมต่อแบบระบบบัส I^2C มีขนาดของหน่วยความจำ เท่ากับ 2 กิโลไบต์

ในส่วนของบอร์ดแสดงผล LED ในโครงงานนี้จะใช้หลอด LED นำมาเรียงต่อกันเป็นตัวเลข ซึ่งใน 1 segment จะใช้หลอด LED จำนวน 5 หลอด โดยตัวเลข 1 ตัวจะใช้ IC ขยายพอร์ตเบอร์ PCF 8574 หนึ่งตัวซึ่ง IC ตัวนี้เป็น IC ขยายพอร์ต โดยจะเชื่อมต่อกับไมโครคอนโทรลเลอร์ด้วยสายเพียง 2 เส้น (I^2C) แต่เนื่องจากเอาต์พุตที่ออกจาก IC ตัวนี้จะจ่ายกระแสได้น้อย ซึ่งจะต้องนำบัฟเฟอร์มาช่วยในการขับกระแส โดยในโครงงานนี้จะใช้ IC เบอร์ ULN 2803 ซึ่งเป็น IC ไดรเวอร์แบบอินเวอร์เตอร์ที่สามารถจ่ายกระแสได้สูงสุด 500 mA และมีไดโอดป้องกันแรงดันย้อนกลับ

ในส่วนของการแสดงผลทางเสียงจะใช้ IC บันทึกเสียงเบอร์ ISD25120 ซึ่งสามารถบันทึกได้นาน 120 วินาที โดยการกำหนด address จากเอาท์พุตของพอร์ต 2 โดยได้ในแต่ละ address ได้ทำการบันทึกคำพูดไว้ดังนี้

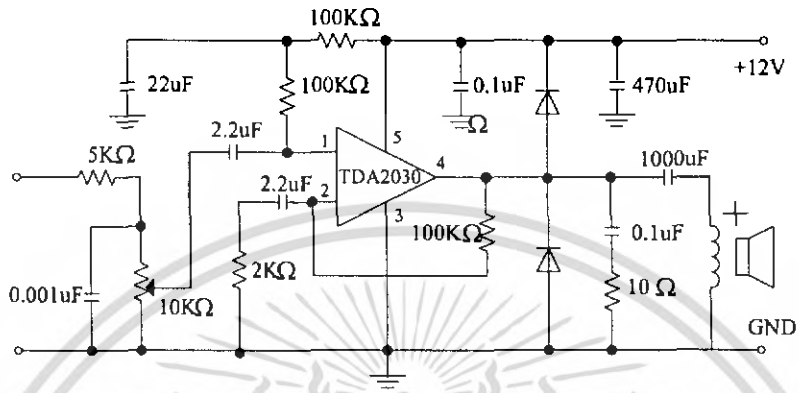
ตำแหน่ง address ที่ทำการบันทึก										คำพูดที่ทำการบันทึก
10	9	8	7	6	5	4	3	2	1	
0	0	0	0	1	0	0	0	0	0	หนึ่ง
0	0	0	1	0	0	0	0	0	0	สอง
0	0	0	1	1	0	0	0	0	0	สาม
0	0	1	0	0	0	0	0	0	0	สี่
0	0	1	0	1	0	0	0	0	0	ห้า
0	0	1	1	0	0	0	0	0	0	หก
0	0	1	1	1	0	0	0	0	0	เจ็ด
0	1	0	0	0	0	0	0	0	0	แปด
0	1	0	0	1	0	0	0	0	0	เก้า
0	1	0	1	0	0	0	0	0	0	สิบ
0	1	0	1	1	0	0	0	0	0	สายที่ / ป้ายที่
0	1	1	0	1	0	0	0	0	0	กำลังจะเข้าป้าย

ตารางที่ 3.1 ตารางแสดงคำพูดที่บันทึกในแต่ละแอดเดรส

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โดยในส่วนวงจรของรถประจำทางการเชื่อมต่ออุปกรณ์ต่าง ๆ กับไมโครคอนโทรเลอร์นั้นจะมีความคล้ายคลึงกับในส่วนของป้ายรถประจำทางต่างกันเพียงในส่วนของรถจะไม่มีในส่วนของการแสดงผลทางบอร์ด LED และการแสดงผลทางเสียง

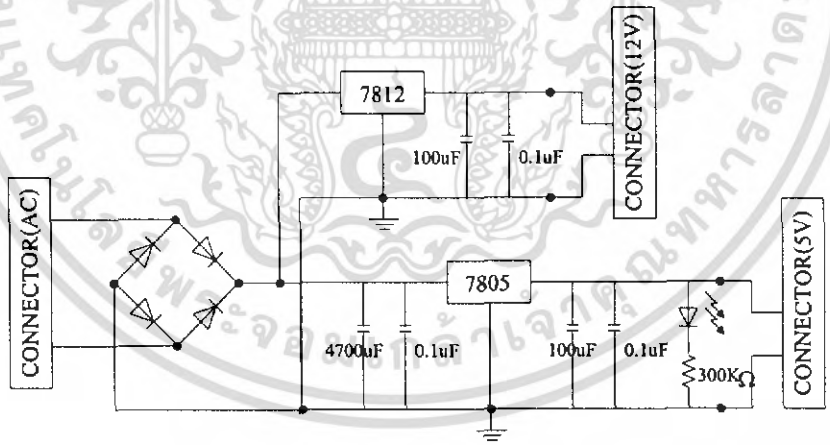
3.5 วงจรขยายเสียง



รูปที่ 3.6 วงจรขยายกำลัง

จากรูปที่ 3.6 เป็นวงจรขยายกำลัง โดยในโครงงานนี้เราจะใช้ IC เบอร์ TDA 2030 เป็นตัวขยายกำลังโดยอินพุตของวงจรได้มาจากเอาท์พุตของ IC บันทึกเสียง ISD 25120

3.6 วงจรจ่ายไฟ



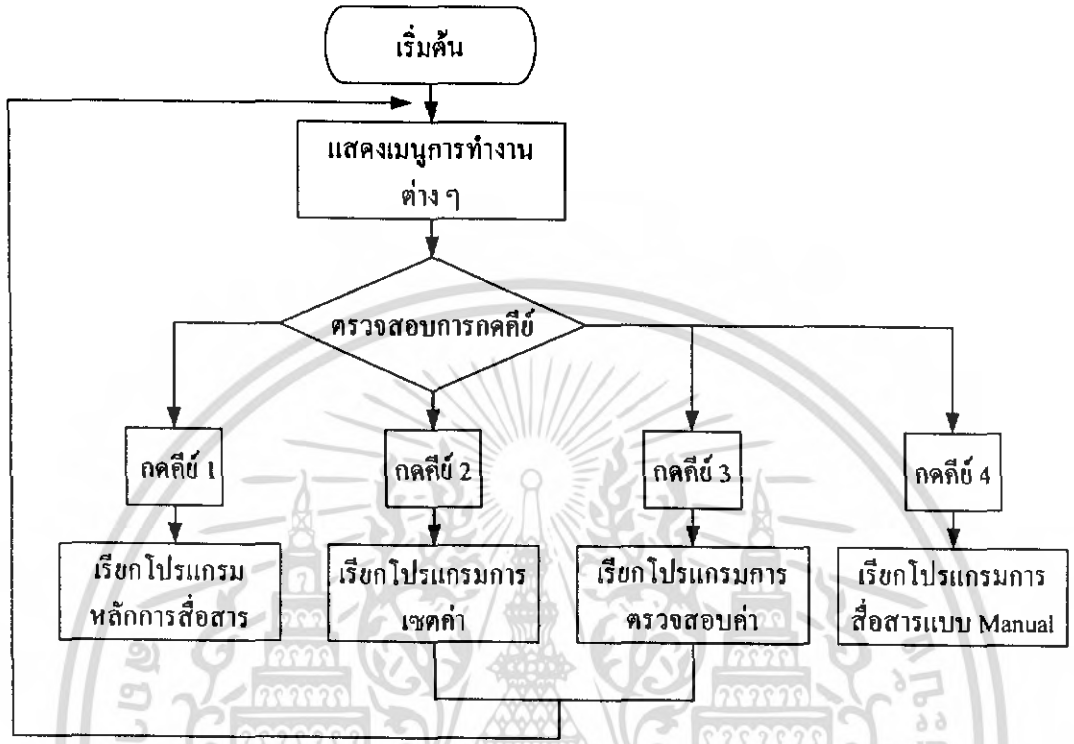
รูปที่ 3.7 วงจรจ่ายไฟ

จากรูปที่ 3.7 เป็นวงจรจ่ายไฟโดยในโครงงานนี้เราต้องการใช้ไฟ 5V และ 12V จ่ายให้แก่อุปกรณ์ต่าง ๆ โดยจะใช้บริดจ์ไดโอดในการแปลงจากไฟ 12V AC ให้เป็น 12 V DC และจากนั้นนำไปเข้า IC regulator เบอร์ 7805 ในการแปลง เป็น 5V และใช้ เบอร์ 7812 ในการแปลงเป็น 12V

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.7 การออกแบบโปรแกรม

3.7.1 การทำงานในส่วนของเมนูหลักของป้ายและรถประจำทาง

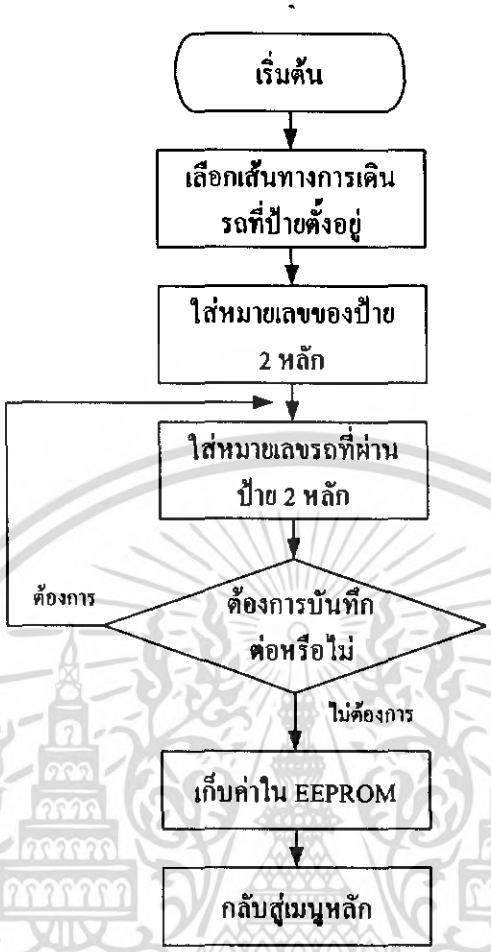


รูปที่ 3.8 Flowchart แสดงการทำงานของเมนูหลัก

จากรูปที่ 3.8 เป็นการแสดงการทำงานในส่วนของเมนูหลัก โดยจะแบ่งการทำงานออกเป็น 4 ส่วน โดยหากกดคีย์ 1 จะเป็นการเรียกโปรแกรมหลักในการสื่อสารระหว่างป้ายรถประจำทางกับรถในรูปแบบที่ทำการเริ่มต้นการสื่อสารจากรถก่อน หากทำการกดคีย์ 2 จะเป็นการเรียกโปรแกรมย่อยในการตั้งค่าข้อมูล หากกดคีย์ 3 จะเป็นการเรียกโปรแกรมย่อยในการตรวจสอบค่าที่ได้ทำการบันทึกไว้ และหากกดคีย์ 4 จะเป็นการเรียกโปรแกรมการสื่อสารแบบไม่อัตโนมัติคือให้ทางด้านรถเริ่มต้นทำการสื่อสารก่อน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.7.2 โปรแกรมย่อยในการตั้งค่าให้กับป้ายรถประจำทาง

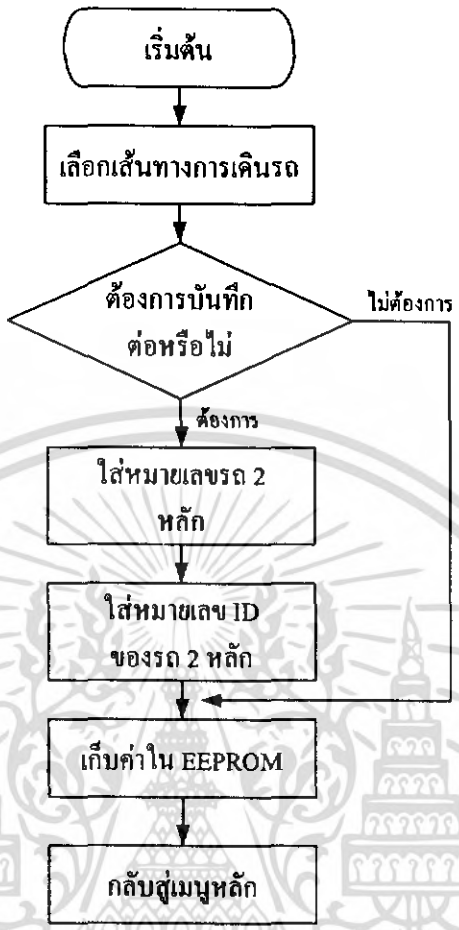


รูปที่ 3.9 Flowchart แสดงโปรแกรมย่อยในการตั้งค่าให้กับป้ายรถประจำทาง

จากรูปที่ 3.9 แสดงการทำงานในการตั้งค่าให้กับป้ายรถประจำทาง โดยเริ่มต้นจากการเลือกเส้นทางการเดินรถที่ป้ายนั้นตั้งอยู่เพื่อป้องกันการสื่อสารข้อมูลกับรถที่วิ่งอยู่ในทิศทางตรงกันข้ามกัน จากนั้นทำการป้อนค่าหมายเลขของป้ายรถประจำทางจำนวน 2 หลัก แล้วทำการใส่ค่าหมายเลขของรถประจำทางที่ผ่านโดยใส่เป็นเลข 2 หลัก โดยสามารถใส่ได้มากสุด 3 ชุด จากนั้นทำการเก็บค่าลงใน EEPROM จนครบจำนวนที่ทำการป้อน จากนั้นจึงกลับสู่เมนูหลัก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.7.3 โปรแกรมย่อยในการตั้งค่าให้กับรถประจำทาง

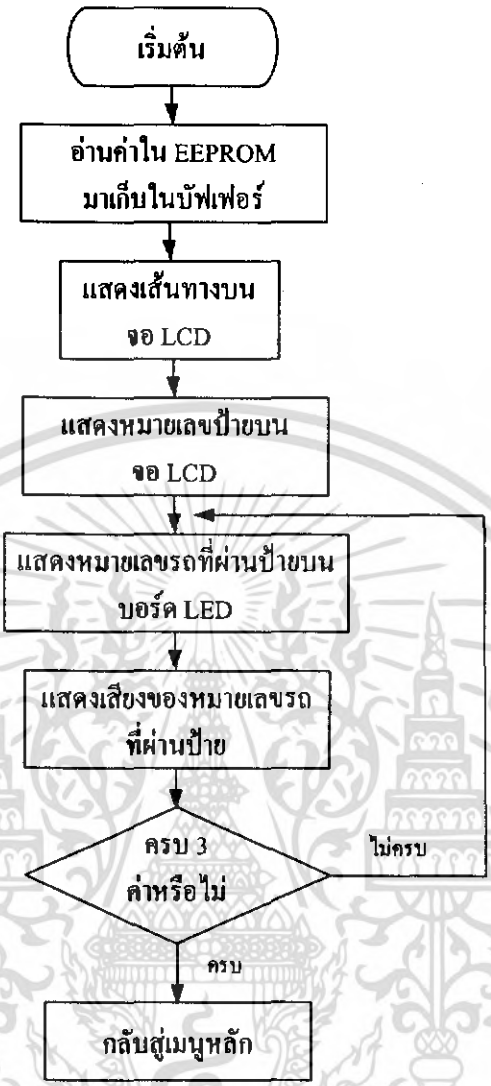


รูปที่ 3.10 Flowchart แสดงโปรแกรมย่อยในการตั้งค่าให้กับรถประจำทาง

จากรูปที่ 3.10 แสดงการทำงานในการตั้งค่าให้กับรถประจำทาง โดยเริ่มจากการเลือกเส้นทางการเดินรถเพื่อให้สามารถสื่อสารกับป้ายที่ตั้งอยู่ในทิศทางที่วิ่งได้ถูกต้อง จากนั้นใส่ค่าหมายเลขของรถประจำทางจำนวน 1 ชุด ซึ่งมี 2 หลัก แล้วทำการใส่หมายเลข ID ของรถคันนั้น (โดยที่ ID ของรถแต่ละคันในสายนั้นจะต้องไม่เหมือนกัน) จากนั้นทำการเก็บค่าลงใน EEPROM แล้วจึงกลับสู่เมนูหลัก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.7.4 โปรแกรมย่อยในการตรวจสอบหมายเลขป้ายรถประจำทางที่ได้บันทึกไว้

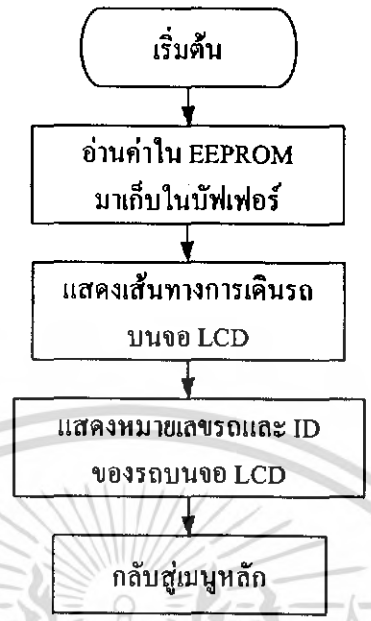


รูปที่ 3.11 Flowchart แสดงโปรแกรมย่อยในการตรวจสอบหมายเลขที่ป้ายรถประจำทาง

จากรูปที่ 3.11 แสดงการทำงานในการตรวจสอบหมายเลขที่ป้ายรถประจำทาง โดยเริ่มต้นจากการอ่านค่าใน EEPROM มาเก็บไว้ในบัฟเฟอร์ จากนั้นจึงแสดงเส้นทางที่ป้ายนั้นตั้งอยู่และค่าหมายเลขของป้ายบนจอ LCD แล้วจึงแสดงค่าหมายเลขของรถประจำทางที่ผ่านป้ายนั้น ๆ บนจอ LCD แล้วแสดงหมายเลขรถที่ผ่านบนบอร์ด LED แล้วจึงแสดงเสียงของแต่ละหมายเลข จากนั้นจึงตรวจสอบว่าแสดงครบตามที่ได้บันทึกแล้วหรือยัง ถ้าครบแล้วก็จะกลับไปสู่เมนูหลัก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.7.5 โปรแกรมย่อยในการตรวจสอบหมายเลขที่รถประจำทางที่ได้บันทึกไว้

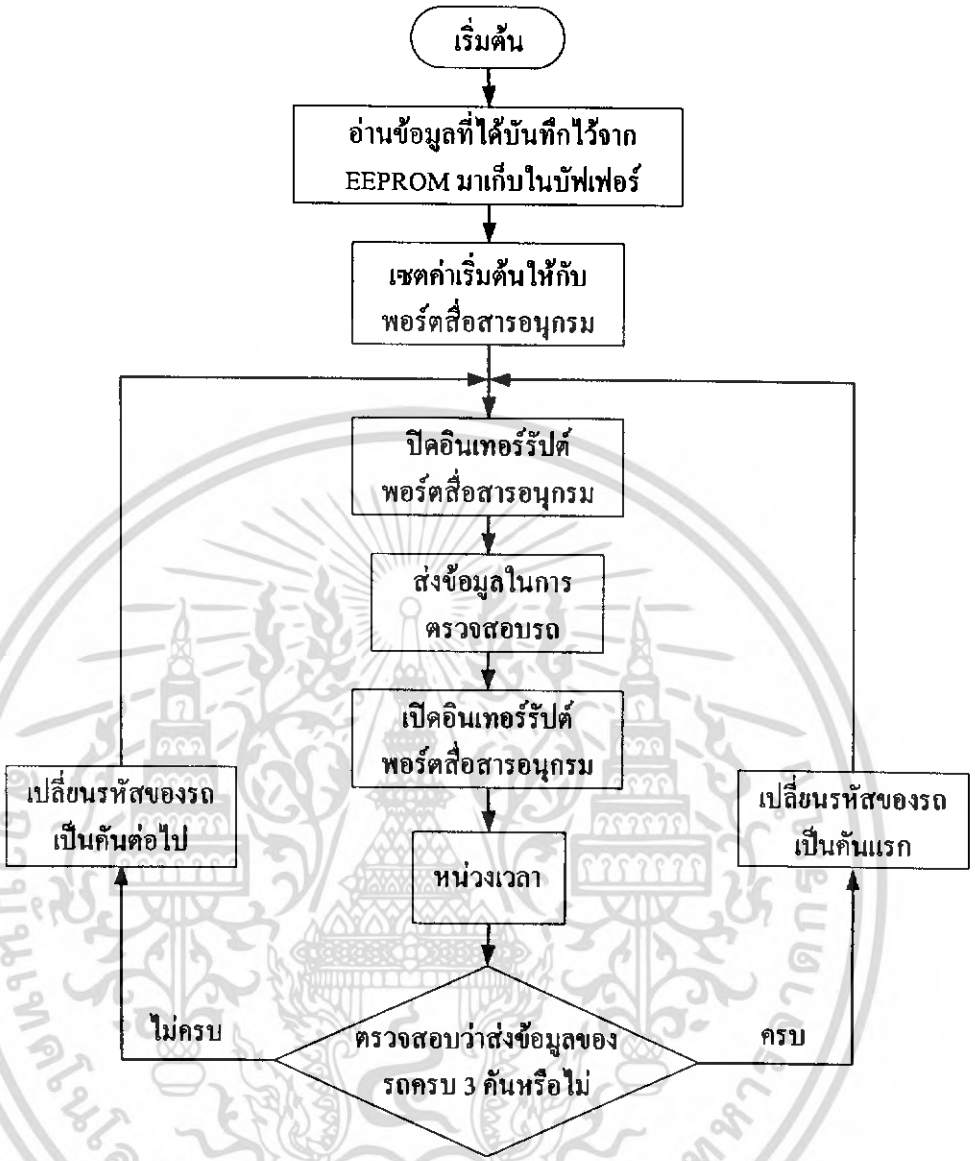


รูปที่ 3.12 Flowchart แสดงโปรแกรมย่อยในการตรวจสอบหมายเลขที่รถประจำทาง

จากรูปที่ 3.12 แสดงการทำงานในการตรวจสอบหมายเลขของรถประจำทาง โดยเริ่มต้นจากการ อ่านค่าหมายเลขรถใน EEPROM มาเก็บไว้ในบัฟเฟอร์ จากนั้นจึงทำการแสดงเส้นทางที่รถวิ่งแล้วแสดงหมายเลขของรถและ ID ของรถคันนั้นบนจอ LCD จากนั้นก็กลับสู่เมนูหลัก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.7.6 โปรแกรมหลักการสื่อสารของป้ายรถประจำทาง

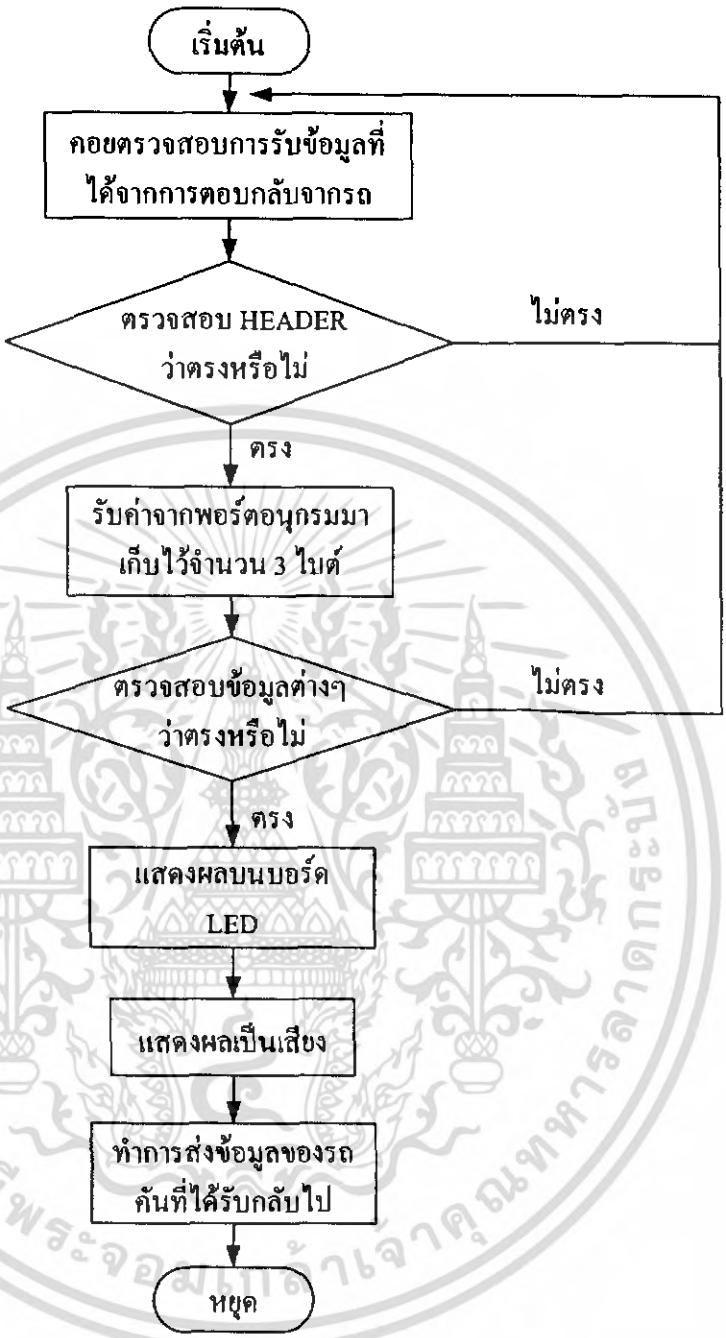


รูปที่ 3.13 Flowchart แสดงโปรแกรมหลักการสื่อสารของป้ายรถประจำทาง

จากรูปที่ 3.13 แสดง Flowchart การทำงานของโปรแกรมหลักในการสื่อสารของป้ายประจำทางกับรถประจำทาง โดยเริ่มต้นจากการอ่านค่าของหมายเลขรถที่วิ่งผ่านป้ายนั้น ๆ ทั้งหมดจาก EEPROM จากนั้นเซตค่าเริ่มต้นให้กับพอร์ตสื่อสาร แล้วทำการปิดอินเทอร์รัปป์พอร์ตสื่อสาร แล้วทำการส่งค่ารหัสของรถแต่ละคันออกไป จากนั้นทำการเปิดอินเทอร์รัปป์พอร์ตสื่อสารเพื่อรอการตอบกลับจากรถ ถ้าหากมีการตอบกลับจากรถก็จะข้ามไปทำโปรแกรมบริการอินเทอร์รัปป์ แต่ถ้าไม่มีข้อมูลส่งกลับมาก็จะหน่วงเวลาจนกว่าจะครบแล้วทำการส่งรหัสของรถคันต่อไป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.7.7 โปรแกรมย่อยบริการอินเทอร์รัปปี้เมื่อมีการตอบกลับจากรถของป้ายรถประจำทาง

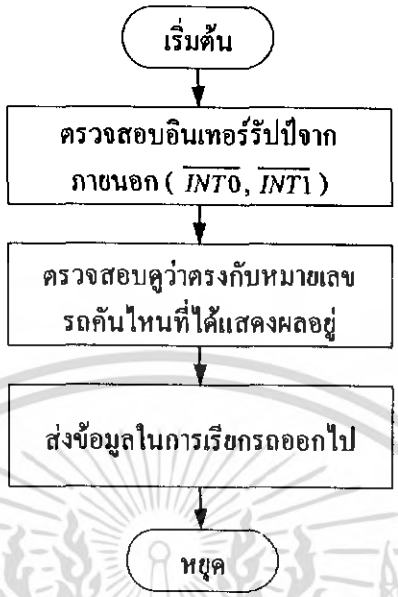


รูปที่ 3.14 Flowchart แสดงโปรแกรมย่อยบริการอินเทอร์รัปปี้ของป้ายรถประจำทาง

จากรูปที่ 3.14 แสดง Flowchart การทำงานของโปรแกรมย่อยบริการอินทอร์รัปปี้โดยการเริ่มต้นจากการรับข้อมูลไบต์แรกเข้ามาแล้วทำการตรวจสอบว่าตรงกับ HEADER หรือไม่ ถ้าหากตรงก็จะทำการรับค่าเข้ามาอีก 3 ไบต์ แล้วตรวจดูว่าข้อมูลที่รับได้นั้นตรงกับข้อมูลที่บันทึกไว้ใน EEPROM หรือไม่ ถ้าตรงก็จะทำการแสดงผลออกมาในลักษณะต่างๆ แล้วทำการส่งข้อมูลของรถคันที่รับได้กลับไปเพื่อบอกทางรถว่าได้รับข้อมูลแล้ว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.7.8 โปรแกรมการติดต่อกับรถเพื่อเรียกรถ

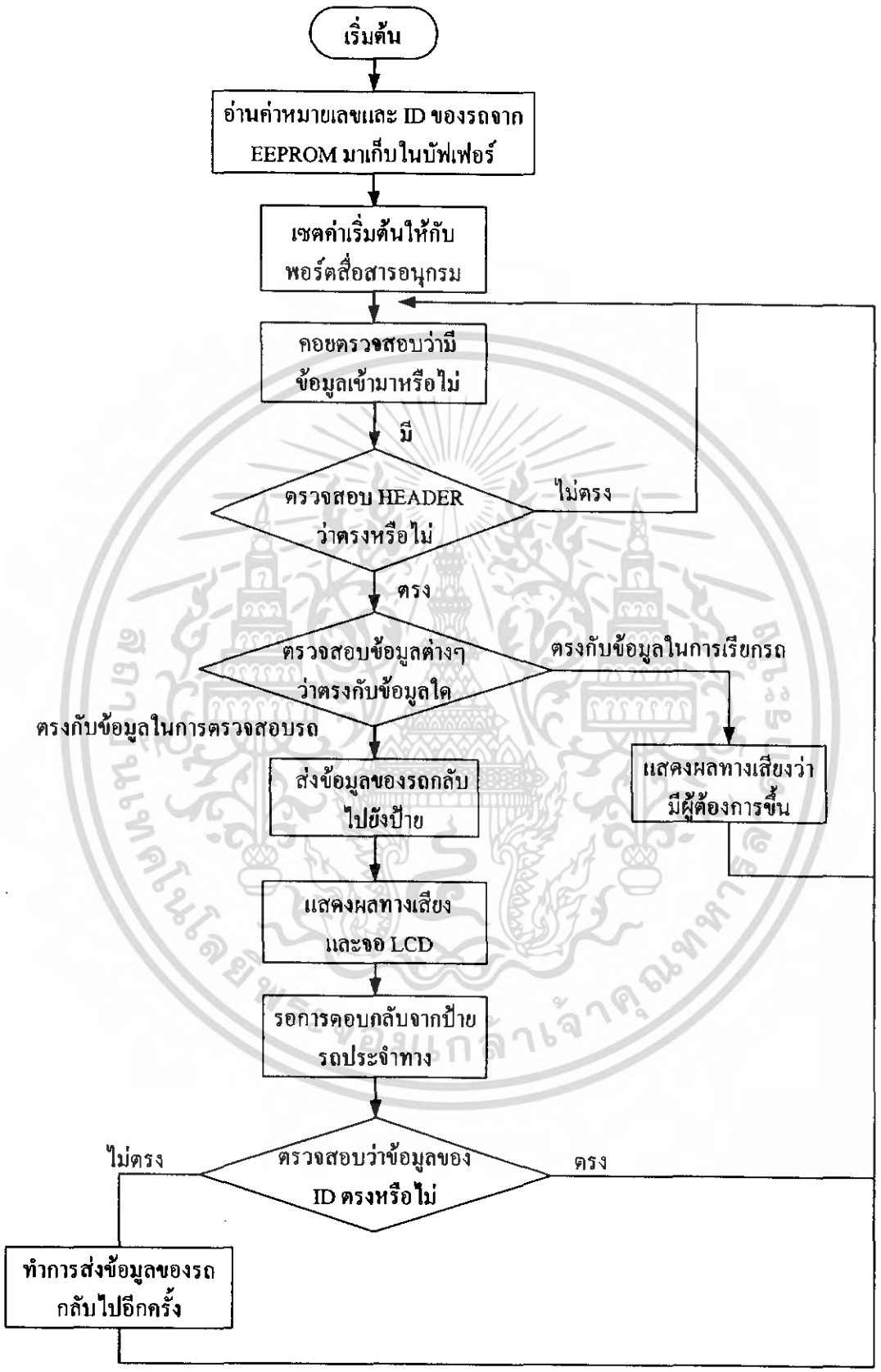


รูปที่ 3.15 Flowchart แสดงโปรแกรมการติดต่อกับรถเพื่อเรียกรถ

จากรูปที่ 3.15 แสดง Flowchart การติดต่อกับรถเพื่อเรียกรถ โดยทำการตรวจสอบอินเทอร์รัปป์จากภายนอก เมื่อมีการอินเทอร์รัปป์จากภายนอกเข้ามาก็จะทำการตรวจสอบดูว่าต้องการติดต่อกับรถคันใด แล้วทำการส่งข้อมูลในการเรียกรถคันนั้นออกไป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.7.9 โปรแกรมหลักการสื่อสารของรถประจำทาง



รูปที่ 3.16 Flowchart โปรแกรมหลักการสื่อสารของรถประจำทาง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากรูปที่ 3.16 แสดง Flowchart การทำงานของโปรแกรมหลักการสื่อสารของรถประจำทาง โดยเริ่มต้นจากการเซตค่าเริ่มต้นให้กับพอร์ตสื่อสารอนุกรมจากนั้นคอยตรวจสอบดูว่ามีข้อมูลส่งมาจากป้ายหรือไม่ ถ้ามีก็จะทำการตรวจสอบดูว่าไบต์แรกนั้นตรงกับ HEADER หรือไม่ ถ้าตรงก็จะทำการรับค่าของข้อมูลอีก 3 ไบต์ แล้วตรวจสอบดูว่าเป็นข้อมูลชนิดใดและตรงกับที่ได้บันทึกไว้ใน EEPROM หรือไม่ ถ้าตรงก็จะทำการส่งข้อมูลของตัวเองกลับไปยังป้าย แล้วจึงแสดงผล จากนั้นรอการตอบกลับจากป้ายรถประจำทางว่าได้รับข้อมูลที่ถูกต้องแล้วหรือไม่ จากนั้นเมื่อได้รับข้อมูลที่ตอบกลับมาก็จะทำการตรวจสอบดูว่าข้อมูลนั้นตรงกับของตัวเองหรือไม่ ถ้าหากว่าตรงก็จะไม่กระทำการใด ๆ แต่หากว่าไม่ตรงก็จะทำการส่งข้อมูลของตัวเองกลับไปอีกครั้ง

3.7.10 โปรแกรมการสื่อสารแบบ Manual ที่รถประจำทาง

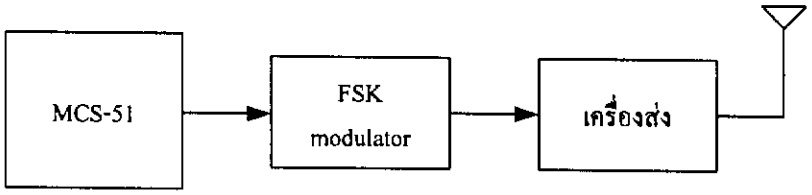
รูปที่ 3.17 Flowchart โปรแกรมการสื่อสารแบบ Manual ที่รถประจำทาง

จากรูปที่ 3.17 แสดง Flowchart โปรแกรมการสื่อสารแบบ Manual ที่รถประจำทาง โดยเริ่มต้นจากการอ่านข้อมูลจาก EEPROM แล้วทำการเซตค่าเริ่มต้นให้กับพอร์ตสื่อสารอนุกรม จากนั้นทำการตรวจสอบดูว่ามีอินเทอร์รัปป์จากภายนอกเข้ามาหรือไม่ ถ้ามีก็จะทำการส่งข้อมูลที่ได้บันทึกไว้ออกไปยังป้าย

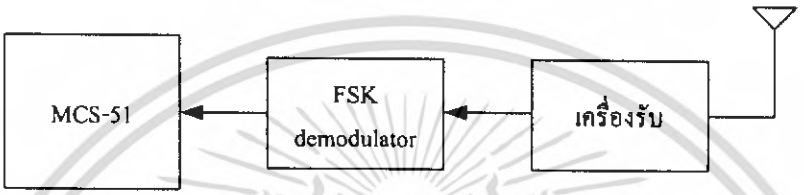
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 4

การทดลองและผลการทดลอง

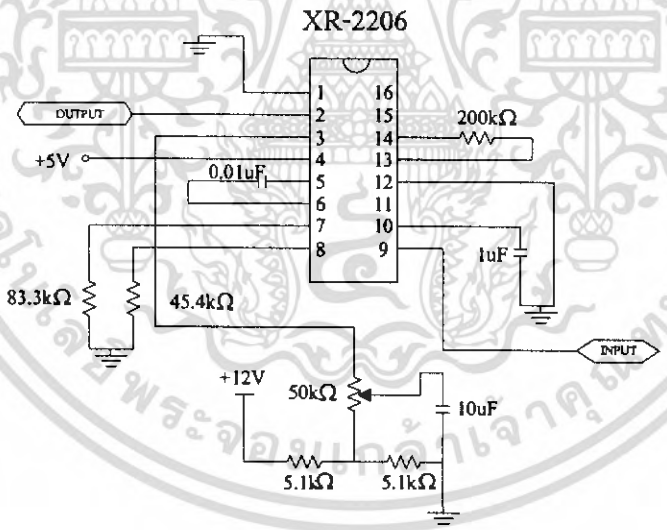


รูปที่ 4.1 Block diagram ของการส่งข้อมูล



รูปที่ 4.2 Block diagram ของการรับข้อมูล

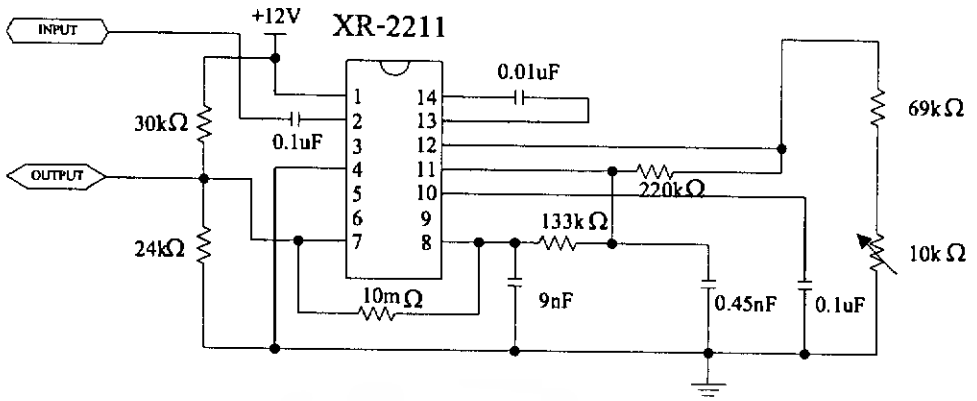
4.1 การทดลองวงจรเข้ารหัสและถอดรหัส FSK



รูปที่ 4.3 วงจรเข้ารหัส (FSK Modulation)

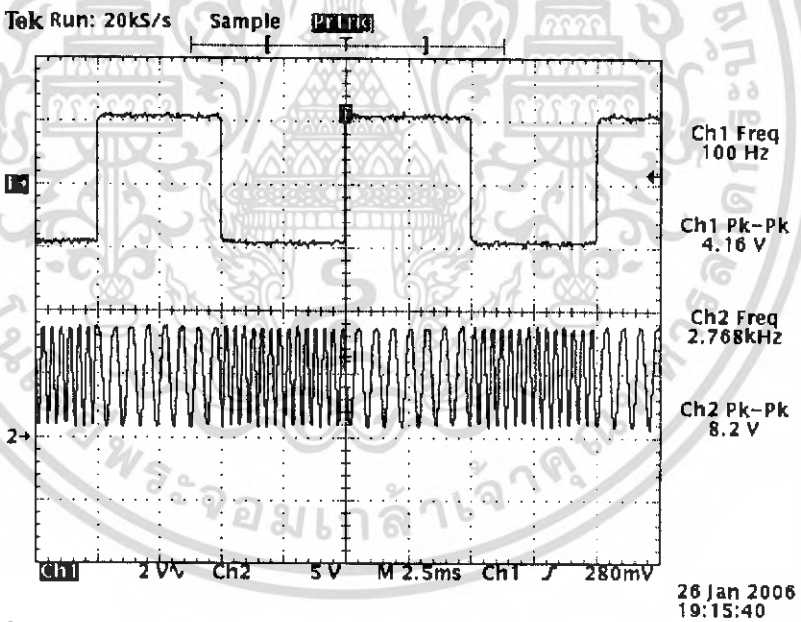
1. ต่อวงจรเข้ารหัส FSK ตามรูปที่ 4.3
2. ทำการป้อนสัญญาณ pulse แอมปลิจูด 5V ความถี่ 100Hz เข้าที่ขา 9 ของ IC เบอร์ XR-2206
3. วัดสัญญาณ output ที่ขา 2 ของ IC เบอร์ XR-2206 เทียบกับ input

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.4 วงจรถอดรหัส (FSK Modulation)

4. ต่อวงจรถอดรหัส FSK ตามรูปที่ 4.4
5. ทำการป้อนสัญญาณ FSK ที่ได้จาก output ของวงจรเข้ารหัส FSK เข้าที่ขา 2 ของ IC XR-2211
6. ทำการวัดสัญญาณ output ที่ขา 7 ของ IC XR-2211 เทียบกับสัญญาณ input

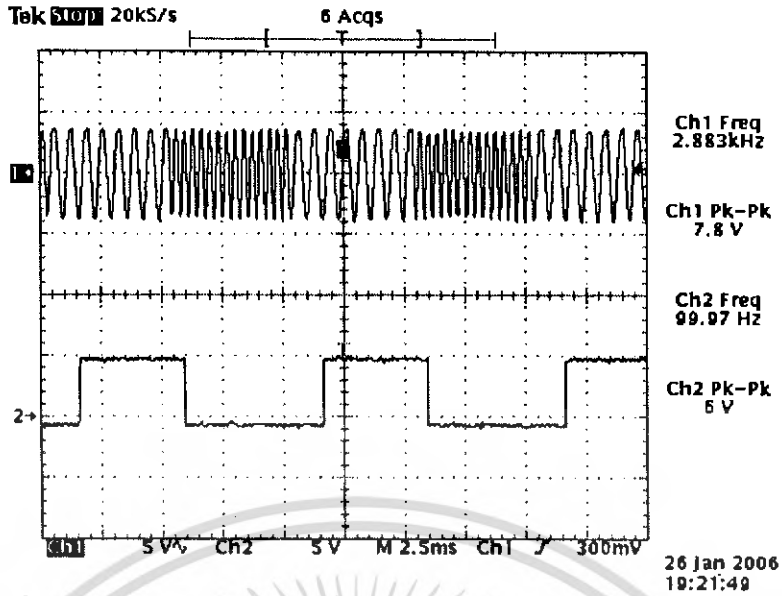


รูปที่ 4.5 แสดงสัญญาณ output ของวงจรเข้ารหัส FSK เทียบกับสัญญาณ input

Ch1 สัญญาณ input pulse ความถี่ 100Hz 5V

Ch2 สัญญาณ output FSK

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



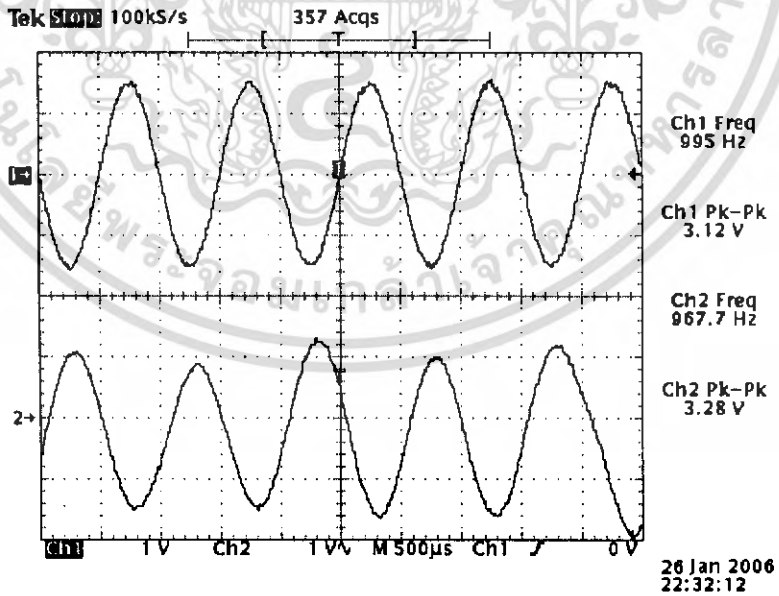
รูปที่ 4.6 แสดงสัญญาณ output ที่ได้จากวงจรถอดรหัส FSK เทียบกับ input

Ch1 สัญญาณ FSK input ที่ได้จาก output ของ IC XR-2206

Ch2 สัญญาณ output

4.2 การทดลองวัดสัญญาณที่ได้รับเมื่อใช้เครื่องรับ – ส่ง (วิทยุสื่อสาร)

1. ทำการป้อนสัญญาณ sine 1KHz ที่ MIC ของเครื่องส่ง
2. ทำการวัดสัญญาณ output ที่ SPEAKER ของเครื่องรับ
3. ทำการป้อนสัญญาณจากพอร์ตอนุกรมเข้าวงจรเข้ารหัส แล้วนำ output ต่อกับเครื่องส่ง
4. ทำการป้อนสัญญาณที่ได้จาก SPEAKER ของเครื่องรับเข้าไปยัง input ของวงจรถอดรหัส แล้วทำการวัดสัญญาณ output ที่ได้เทียบกับ input



รูปที่ 4.7 แสดงสัญญาณ output จากเครื่องรับเทียบกับ input ของเครื่องส่ง

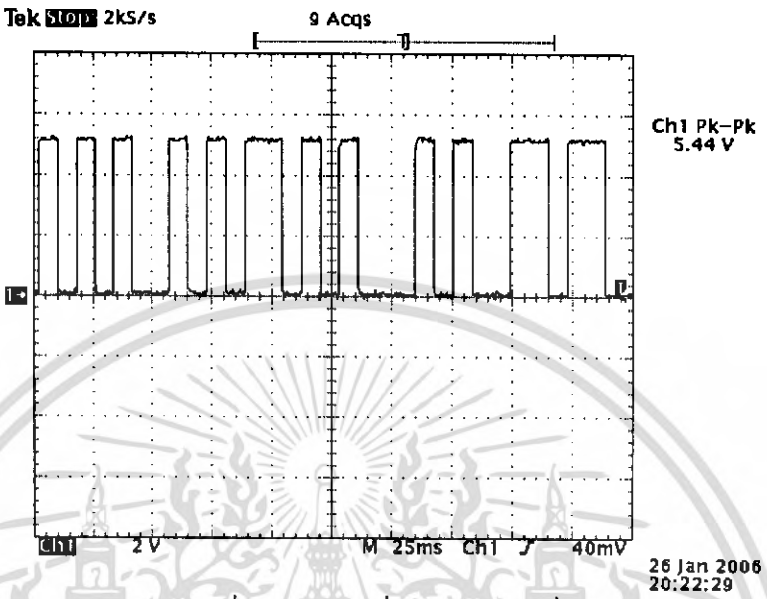
Ch1 สัญญาณ output จากเครื่องรับ

Ch2 สัญญาณ input ของเครื่องส่ง

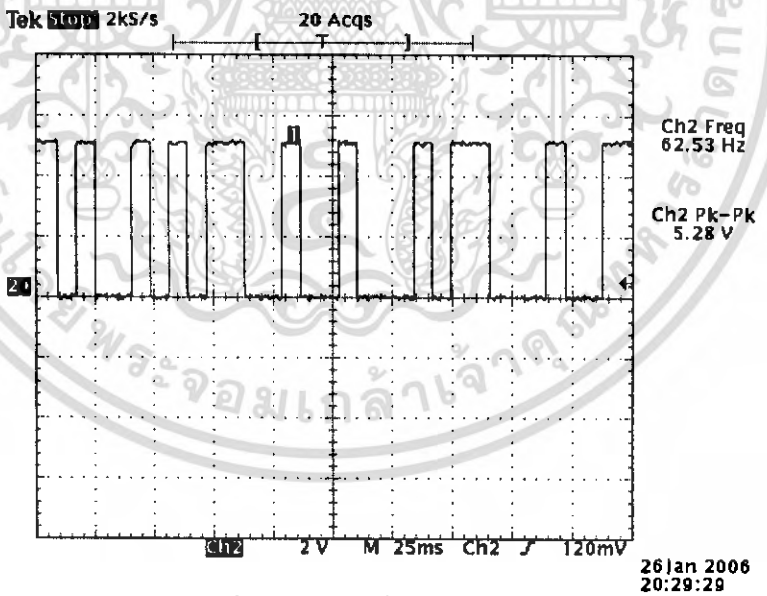
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.3 การทดลองวัดสัญญาณที่ได้จากพอร์ตสื่อสารอนุกรม

1. ทำการวัดสัญญาณที่ได้จากพอร์ตสื่อสารอนุกรมก่อนจะทำการเข้ารหัส FSK เมื่อทำการส่งข้อมูลในการตรวจสอบรถประจำทาง



รูปที่ 4.8 แสดงสัญญาณ output ที่ได้จากพอร์ตสื่อสารอนุกรม เมื่อทำการตรวจสอบรถหมายเลข 45 ที่วิ่งในทิศทางไปจากป้ายหมายเลข 36



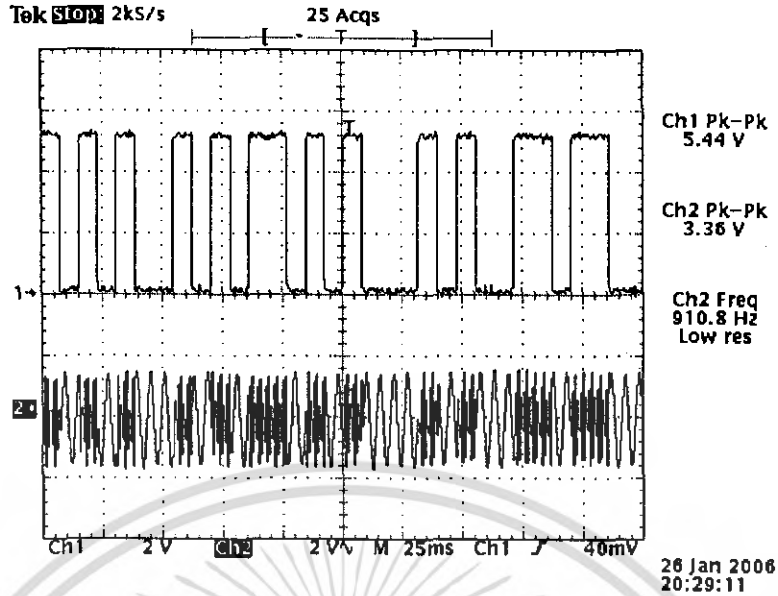
รูปที่ 4.9 แสดงสัญญาณ output ที่ได้จากพอร์ตสื่อสารอนุกรม เมื่อทำการตรวจสอบรถหมายเลข 12 ที่วิ่งในทิศทางไปจากป้ายหมายเลข 36

4.4 การทดลองเข้ารหัสสัญญาณที่ได้จากพอร์ตสื่อสารอนุกรม

1. ต่อวงจรเข้ารหัส FSK ตามรูปที่ 4.3
2. ทำการป้อนสัญญาณจากพอร์ตสื่อสารอนุกรมเข้าวงจรเข้ารหัส

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

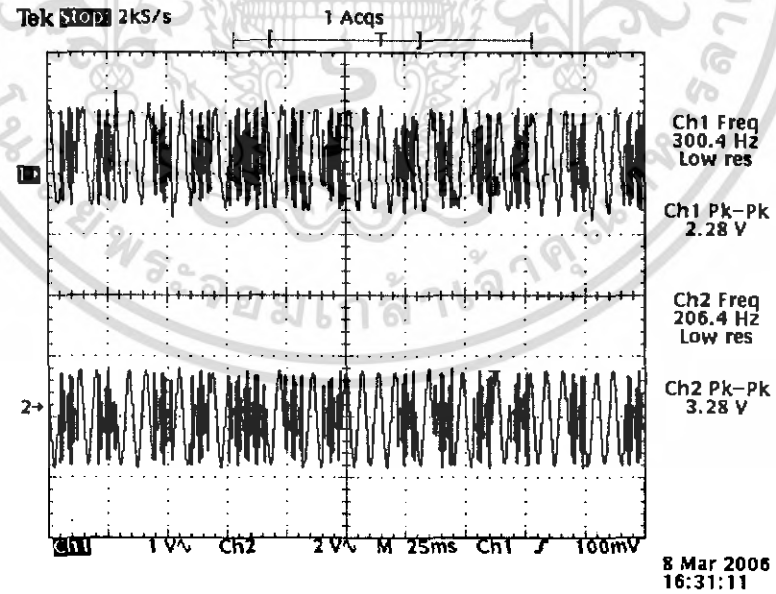
3. วัดสัญญาณ output ที่ขา 2 ของ IC เบอร์ XR 2206 เทียบกับ input



รูปที่ 4.10 แสดงสัญญาณ input เทียบกับ output ของวงจรเข้ารหัส
เมื่อป้อนสัญญาณจากพอร์ตอนุกรม
Ch1 สัญญาณ input จากพอร์ตอนุกรม
Ch2 สัญญาณ output

4.5 การทดลองวัดสัญญาณที่ได้รับจากเครื่องรับวิทยุ

1. ทำการป้อนสัญญาณที่ได้จาก output ของวงจรเข้ารหัส FSK เข้าที่ MIC ของเครื่องส่ง
2. ทำการวัดสัญญาณที่ได้รับจาก SPEAKER ของเครื่องรับเทียบกับ input ของเครื่องส่ง



รูปที่ 4.11 แสดงสัญญาณ input ของเครื่องส่งเทียบกับ output ของเครื่องรับวิทยุ

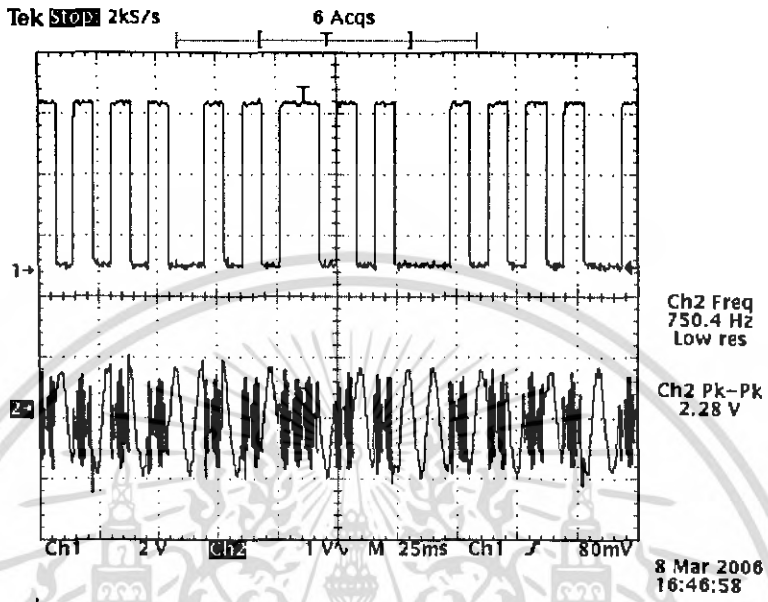
Ch1 สัญญาณ output จากเครื่องรับ

Ch2 สัญญาณ input ของเครื่องส่ง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.6 การทดลองถอดรหัสสัญญาณที่ได้รับจากเครื่องรับวิทยุ

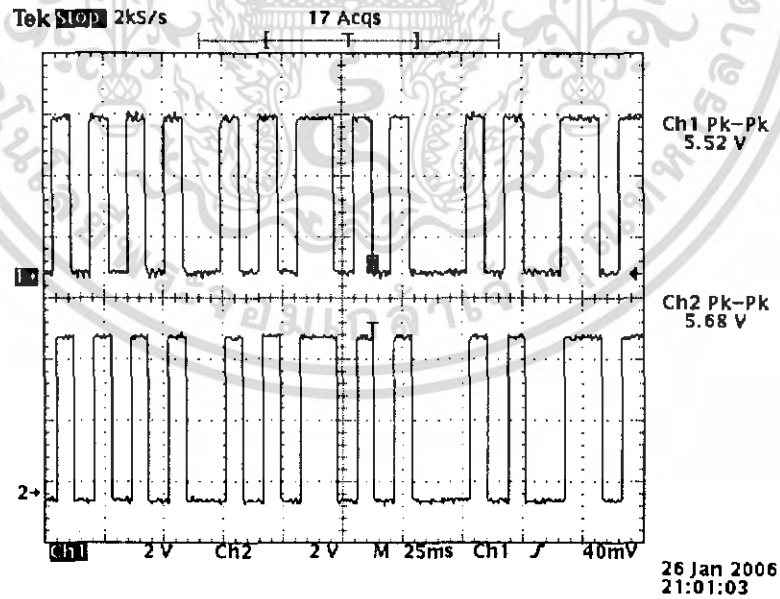
1. ทำการป้อนสัญญาณที่ได้รับจาก SPEAKER ของเครื่องรับวิทยุเข้าวงจรถอดรหัส FSK
2. ทำการวัดสัญญาณ output ของวงจรถอดรหัส เทียบกับสัญญาณอินพุต
3. ทำการวัดสัญญาณ output ของวงจรถอดรหัสเทียบกับสัญญาณที่ได้จากพอร์ตสื่อสารอนุกรม



รูปที่ 4.12 แสดงสัญญาณ input เทียบกับ output ของวงจรถอดรหัส

Ch1 สัญญาณ input จากวงจรเข้ารหัส

Ch2 สัญญาณ output

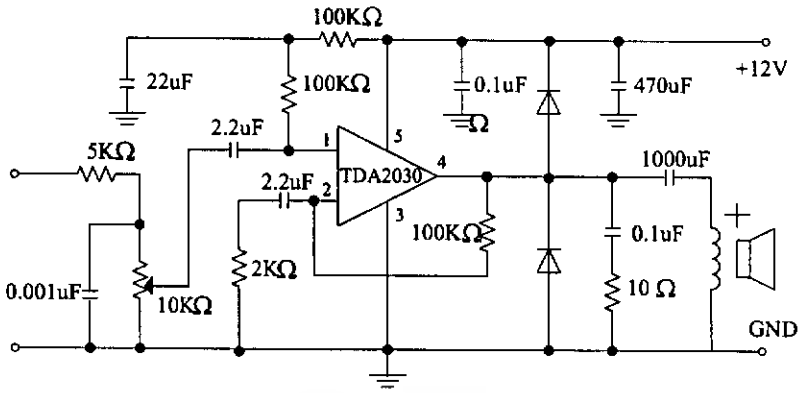


รูปที่ 4.13 แสดงสัญญาณ input จากพอร์ตอนุกรม เทียบกับสัญญาณ output ของวงจรถอดรหัส

Ch1 สัญญาณ input ที่ได้จากพอร์ตอนุกรม

Ch2 สัญญาณ output

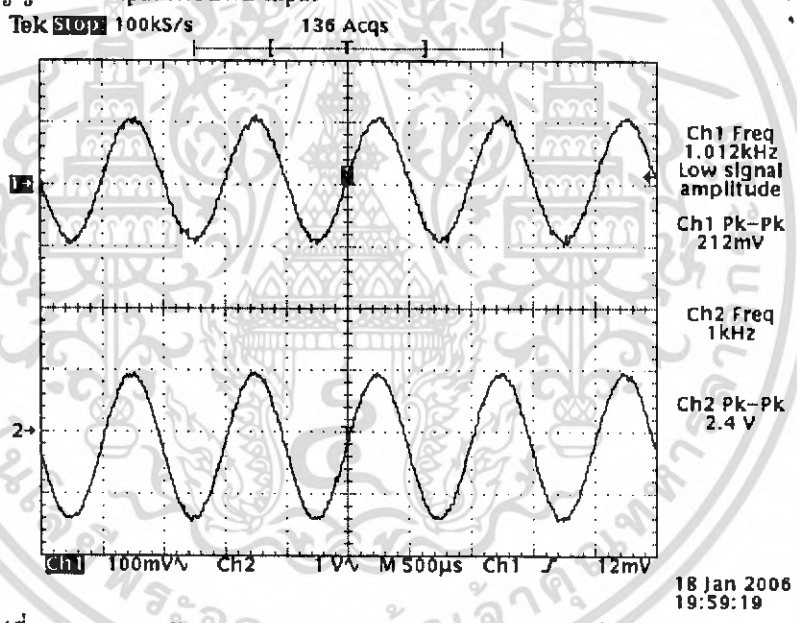
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.14 วงจรขยายกำลัง

4.7 การทดลองวงจรขยายกำลัง

1. ต่อวงจรขยายกำลังดังรูปที่ 4.14
2. ป้อนสัญญาณ sine 1KHz 200mV เข้าที่ input
3. วัดสัญญาณที่ output เทียบกับ input



รูปที่ 4.15 แสดงสัญญาณ output ของวงจรขยายสัญญาณเทียบกับ input

Ch1 สัญญาณ input

Ch2 สัญญาณ output

4.8 การทดลองวัดระยะทางที่สามารถติดต่อระหว่างป้ายกับรถ

จากการทดลองวัดระยะทางที่รถกับป้ายสามารถติดต่อกันได้พบว่าในสถานที่โล่ง ไม่มีสิ่งกีดขวางสามารถทำการติดต่อได้ไกลประมาณ 200 เมตร

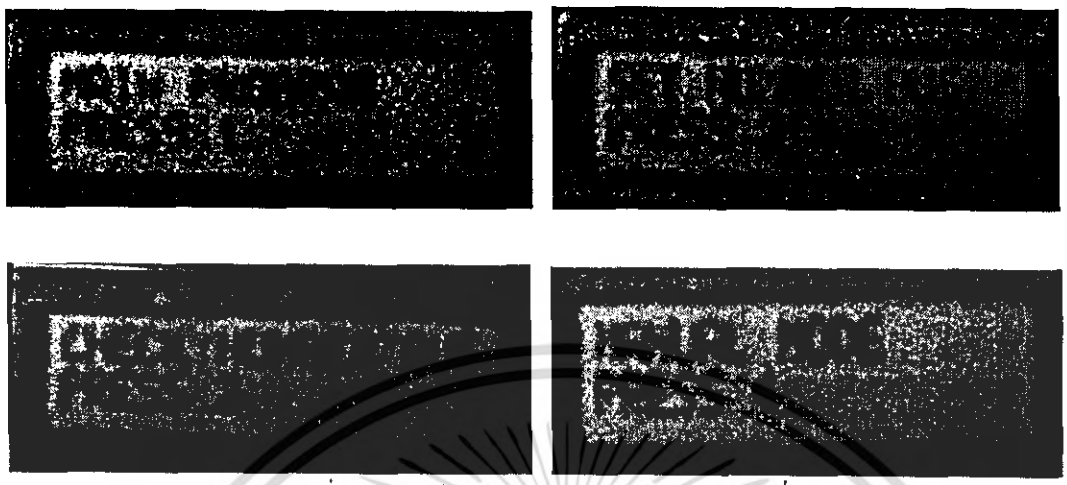
และในสถานที่ที่มีสิ่งก่อสร้างบังอยู่สามารถทำการติดต่อได้ไกลประมาณ 100 เมตร

โดยระยะทางที่สามารถติดต่อได้นั้นขึ้นอยู่กับกำลังของเครื่องรับเครื่องส่งวิทยุด้วย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

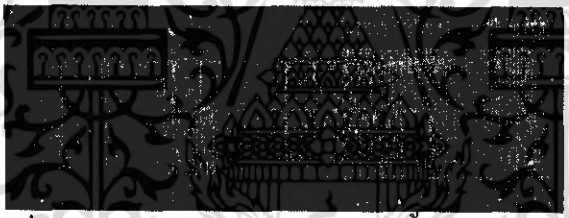
4.9 ขั้นตอนการทำงานในส่วนของป้ายรถประจำทาง

เมื่อเริ่มต้นทำการเปิดเครื่องที่จอ LCD จะทำการแสดงข้อความของเมนูต่าง ๆ ที่สามารถทำการเลือกได้ดังรูปที่ 4.16



รูปที่ 4.16 หน้าจอแสดงเมนูการทำงานต่าง ๆ ที่สามารถเลือกได้

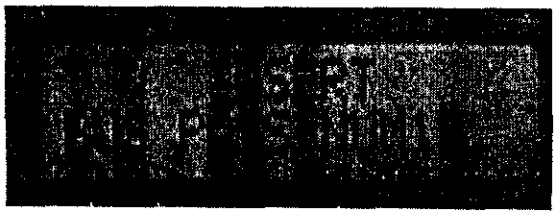
จากนั้นเมื่อแสดงข้อความต่าง ๆ ครบแล้วก็จะรอการกดคีย์ว่าต้องการเข้าสู่เมนูใด หากเราต้องการทำการเซตค่าข้อมูลต่าง ๆ ให้กับป้ายรถประจำทางก็ให้ทำการกดคีย์ 2 หน้าจอจะปรากฏดังรูปที่ 4.17 เพื่อให้เลือกว่าป้ายรถประจำทางนี้ตั้งอยู่ในทิศทางการเดินรถใด (ไปหรือกลับ)



รูปที่ 4.17 หน้าจอแสดงให้เลือกว่าป้ายนั้นอยู่ในเส้นทางใด

เมื่อเลือกเส้นทางแล้วหน้าจอก็จะแสดงข้อความให้ใส่ว่าป้ายนี้มีหมายเลขใดดังรูปที่ 4.18 โดยทำการใส่เป็นตัวเลขจำนวน 2 หลัก เมื่อใส่ครบแล้วหน้าจอก็จะแสดงข้อความดังรูปที่ 4.19 ให้ใส่หมายเลขของรถประจำทางที่วิ่งผ่านป้ายนี้ โดยทำการใส่หมายเลขจำนวน 2 หลัก

รูปที่ 4.18 หน้าจอแสดงข้อความให้บันทึกหมายเลขป้าย



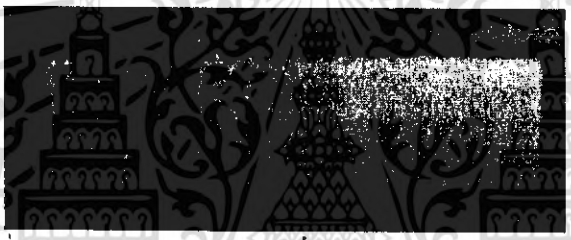
รูปที่ 4.19 หน้าจอแสดงข้อความให้บันทึกหมายเลขรถที่ผ่านป้าย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เมื่อใส่หมายเลขของรถคันแรกแล้วหน้าจอก็จะแสดงว่าต้องการใส่หมายเลขของรถคันต่อไปหรือไม่ดังรูปที่ 4.20 ถ้าหากจะบันทึกต่อก็จะกลับไปยังหน้าจอดังรูปที่ 4.19 เพื่อทำการบันทึกหมายเลขของรถคันต่อไป หากไม่ต้องการบันทึกต่อก็จะออกจากเมนูนี้โดยหน้าจอจะแสดงข้อความเหมือนตอนเริ่มต้นใหม่อีกครั้ง

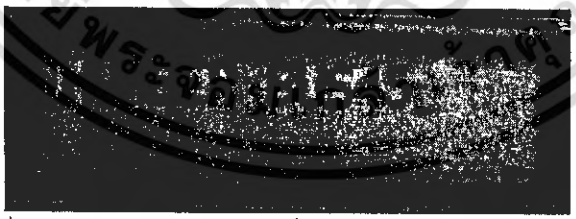
รูปที่ 4.20 หน้าจอแสดงข้อความให้เลือกว่าจะบันทึกข้อมูลต่อหรือไม่

หากเราต้องการทำการตรวจสอบดูว่าข้อมูลที่ได้บันทึกไว้นั้นมีข้อมูลใดบ้างก็ให้ทำการกดคีย์ 3 หน้าจอ LCD ก็จะแสดงเส้นทางการเดินรถที่ป้ายนั้นตั้งอยู่ดังรูปที่ 4.21 และแสดงหมายเลขของป้ายดังรูปที่ 4.22 และแสดงหมายเลขของรถที่วิ่งผ่านป้ายดังรูปที่ 4.23 ทีละหมายเลขพร้อมกับแสดงทางบอร์ด LED และเสียง



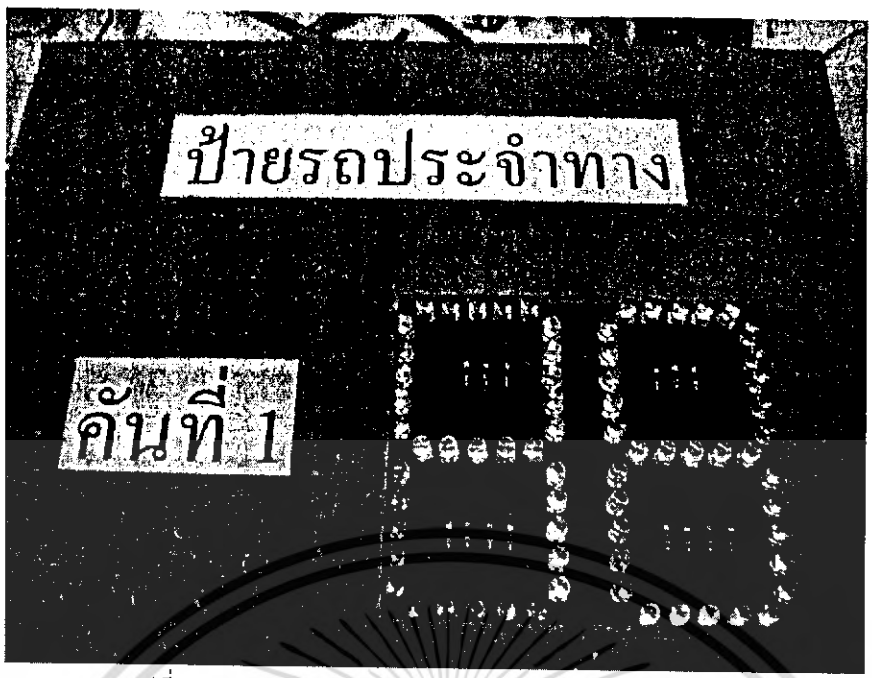
รูปที่ 4.21 หน้าจอแสดงข้อมูลที่บันทึกไว้ว่าอยู่ในเส้นทางใด

รูปที่ 4.22 หน้าจอแสดงข้อมูลที่บันทึกไว้ว่าป้ายหมายเลขใด



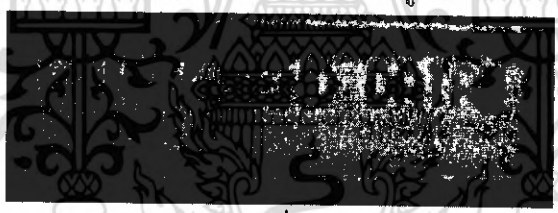
รูปที่ 4.23 หน้าจอแสดงข้อมูลที่ได้บันทึกไว้ว่ามีรถสายใดผ่าน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.24 รูปแสดงการแสดงผลหมายเลขรถบนบอร์ด LED

หากเราต้องการเข้าสู่โปรแกรมหลักการสื่อสารก็ให้ทำการกดคีย์ 1 หน้าจอก็จะแสดงข้อความดังรูปที่ 4.25 อุปกรณ์ก็จะทำการส่งข้อมูลในการตรวจสอบรถวนออกไปเรื่อย ๆ เพื่อรอการตอบกลับจากรถประจำทาง เมื่อมีการตอบกลับจากรถประจำทาง ทางป้ายก็จะตรวจสอบว่าข้อมูลถูกต้องหรือไม่ถ้าหากถูกต้องก็จะแสดงผลเป็นเสียงและหมายเลขบนบอร์ด LED ดังรูปที่ 4.24

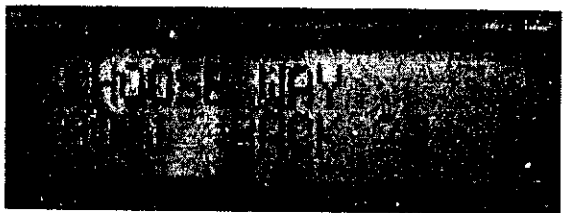


รูปที่ 4.25 หน้าจอแสดงข้อความเมื่อเข้าสู่โปรแกรมการทำงานหลัก

4.10 ขั้นตอนการทำงานในส่วนของรถประจำทาง

เมื่อเริ่มต้นทำการเปิดเครื่องที่จอ LCD จะทำการแสดงข้อความของเมนูต่าง ๆ ที่สามารถทำการเลือกได้ดังรูปที่ 4.16

จากนั้นเมื่อแสดงข้อความต่าง ๆ ครบแล้วก็จะรอการกดคีย์ว่าต้องการเข้าสู่เมนูใด หากเราต้องการทำการเซตค่าข้อมูลต่าง ๆ ให้กับรถประจำทางก็ให้ทำการกดคีย์ 2 หน้าจอจะปรากฏดังรูปที่ 4.20 เพื่อให้เลือกว่ารถประจำทางนี้วิ่งอยู่ในทิศทางใด (ไปหรือกลับ)



รูปที่ 4.26 หน้าจอแสดงให้เลือกว่ารถนั้นวิ่งอยู่ในเส้นทางใด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เมื่อเลือกเส้นทางการเดินรถแล้วหน้าจอก็จะแสดงข้อความดังรูปที่ 4.27 เพื่อให้เลือกว่าต้องการที่จะบันทึกข้อมูลต่อไปหรือไม่ หากไม่ต้องการบันทึกต่อก็จะออกจากเมนูนี้โดยหน้าจอจะแสดงข้อความเหมือนตอนเริ่มต้นใหม่อีกครั้ง แต่หากต้องการบันทึกต่อก็จะปรากฏข้อความดังรูปที่ 4.28 เพื่อให้ทำการใส่หมายเลขของรถประจำทาง โดยใส่หมายเลขจำนวน 2 หลัก

รูปที่ 4.27 หน้าจอแสดงข้อความให้เลือกว่าจะบันทึกข้อมูลต่อหรือไม่

รูปที่ 4.28 หน้าจอแสดงข้อความให้บันทึกหมายเลขรถ

เมื่อใส่หมายเลขรถประจำทางครบ 2 หลักแล้วหน้าจอก็จะปรากฏข้อความดังรูปที่ 4.29 เพื่อให้ทำการใส่หมายเลข ID ของรถคันนั้น โดยใส่เป็นตัวเลขเป็นจำนวน 2 หลัก เมื่อใส่ครบแล้วก็ออกจากเมนูนี้แล้วแสดงข้อความเหมือนตอนเริ่มต้นใหม่อีกครั้ง

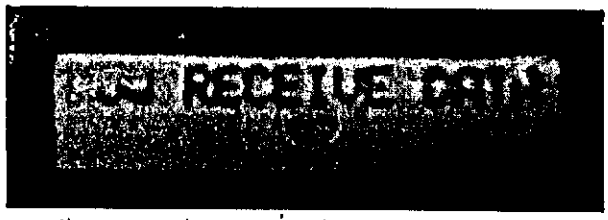
รูปที่ 4.29 หน้าจอแสดงข้อความให้บันทึกหมายเลข ID ประจำรถ

หากเราต้องการทำการตรวจสอบดูว่าข้อมูลที่ได้บันทึกไว้นั้นมีข้อมูลใดบ้างก็ให้ทำการกดคีย์ 3 หน้าจอ LCD ก็จะแสดงหมายเลขของรถประจำทางดังรูปที่ 4.30

รูปที่ 4.30 หน้าจอแสดงข้อมูลของหมายเลขและ ID ของรถที่ได้บันทึกไว้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หากเราต้องการเข้าสู่โปรแกรมหลักการสื่อสารก็ให้ทำการกดคีย์ 1 หน้าจอก็จะแสดงข้อความดัง รูปที่ 4.31 .อุปกรณ์ก็จะทำการรอรับข้อมูลจากป้ายรถประจำทางเมื่อมีข้อมูลเข้ามาก็จะตรวจสอบดูว่าตรงหรือไม่ถ้าหากว่าตรงก็จะทำการส่งข้อมูลของตัวเองกลับไปยังป้าย และแสดงผลทางเสียง



รูปที่ 4.31 หน้าจอแสดงข้อความเมื่อเข้าสู่โปรแกรมการทำงานหลักของรถ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 5
บทสรุปและวิจารณ์

จากการทดลองในส่วนของโปรแกรมย่อยต่าง ๆ เช่นการบันทึกข้อมูล การอ่านข้อมูลที่บันทึกไว้สามารถทำงานได้ดี เมื่อทำการทดลองแยกแต่ละส่วนของวงจรสามารถใช้งานได้ดีแต่เมื่อนำมาต่อร่วมกันบางครั้งยังมีความผิดพลาดอยู่ ซึ่งอาจจะมีผลมาจากในชิ้นงานทั้งหมดมีการเชื่อมต่อแต่ละจุดเป็นจำนวนมาก และบางครั้งสายที่ใช้เชื่อมต่อไม่แน่นหรือใช้ไม่ได้ทำให้ต้องเสียเวลาในการเช็ควงจรทีละส่วน และเมื่อทดลองการรับส่งระหว่างป้ายกับรถประจำทางด้วยเครื่องรับเครื่องส่งในบางครั้งข้อมูลที่รับได้ยังมีความผิดพลาดอยู่

สำหรับแนวทางการแก้ไขและพัฒนาคือ วงจรในแต่ละส่วนควรที่จะลงลายวงจรให้อยู่ในแผ่นเดียวกันเพื่อเป็นการลดการใช้สายในการเชื่อมต่อ สำหรับแนวทางในการพัฒนาคือ เนื่องจากการใช้วิทยุสื่อสารมีความไม่สะดวกในหลาย ๆ เรื่อง ดังนั้นเราอาจจะใช้โมดูลรับส่งสัญญาณที่สามารถรับส่งได้ระยะทางไกล ๆ ในการรับส่งข้อมูลระหว่างป้ายกับรถประจำทาง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หนังสืออ้างอิง

1. นคร ภักดีชาติ "ปฏิบัติการไมโครคอนโทรลเลอร์ MCS-51 ด้วยโปรแกรมภาษาซี" Innovative Experiment Press Thailand ,กรุงเทพมหานคร
2. ผศ. ธีรวัฒน์ ประกอบผล "การพัฒนาไมโครคอนโทรลเลอร์ด้วยภาษาซี"สำนักพิมพ์ส.ส.ท. , กรุงเทพมหานคร,2548
3. รศ.สมยศ จุณณะปิยะ "การประยุกต์ใช้งานไมโครคอนโทรลเลอร์"สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง,กรุงเทพมหานคร,2546
4. ศาสตราจารย์ ดร.วิวัฒน์ กิรานนท์ "วิศวกรรมการสื่อสาร Communication Engineering" สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง,กรุงเทพมหานคร,2544

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โปรแกรมในส่วนของป้ายรถประจำทาง

```
#include<AT89X51.H>
#include <string.h>
#include<INTRINS.H>

#define E P3_6
#define RS P3_7
#define SDA P2_6
#define SCL P2_7
#define key_press P1_6
#define PD P1_7
#define TRA P1_5
#define TRB P1_4

unsigned char
MEM[6],NUM[5],line_1,line_2,line_1ID,line_2ID,WAYN,WAYC,WAYB;
unsigned char RXin = 0,ID_1 = 0,ID_2 = 0,ID_3 = 0,cou = 0,cou_1 =
0,cou_2 = 0,cou_3 = 0;

void delayfifty(int fiftyms)
{
      int x;
      for(x=0;x<fiftyms;x++)
      {
        TH0 = 0x4c;
        TL0 = 0x00;
        TF0 = 0;
        TR0 = 1;
        while(TF0==0);
        TR0 = 0;
      };
}
void lcd_delay(int tick)
{
     int i,j;
     for(i=0;i<tick;i++)
     for(j=0;j<250;j++);
}

void lcd_command(unsigned char com)
{
     RS = 0;
     E =1;
     P0 = com;
     lcd_delay(5);
     E = 0;
     lcd_delay(5);
}

void lcd_text(unsigned char text)
{
     RS = 1;
     E = 1;
     P0 = text;
     lcd_delay(5);
     E = 0;
     lcd_delay(5);
}

void lcd_init()
{
```

เอกสารนี้มีไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        lcd_delay(500);
        lcd_command(0x38);
        lcd_command(0x0C);
        lcd_command(0x01);
}

void lcd_show(unsigned char text[],bit row)
{
        unsigned char a;
        if (row == 0)
              lcd_command(0x80);
        else
              lcd_command(0xC0);
        for(a=0;a<strlen(text);a++)
             lcd_text(text[a]);
}

void key2lcd(unsigned char pad)
{
        pad += 0x30;
        lcd_text(pad);
}

void lcd_clear(void)
{
        lcd_command(0x01);
}

void lcd_blink()
{
        lcd_command(0x0F);
}
void i2c_delay(void)
{
        unsigned char i;
        for(i=0;i<21;i++)
        _nop_();
}

void i2c_clk(void)
{
        i2c_delay();
        SCL = 1;
        i2c_delay();
        SCL = 0;
}

void i2c_start(void)
{
        if(SCL)
        SCL = 0;
        SDA = 1;
        SCL = 1;
        i2c_delay();
        SDA = 0;
        i2c_delay();
        SCL = 0;
}

void i2c_stop(void)
{
        if(SCL)
        SCL = 0;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
	SDA = 0;
	i2c_delay();
	SCL = 1;
	i2c_delay();
	SDA = 1;
}

void i2c_NACK(void)
{
	SDA = 1;
	i2c_delay();
	i2c_clk();
	SCL = 1;
}

bit i2c_wrdata(unsigned char dat)
{
	bit data_bit;
	unsigned char i;
	for(i=0;i<8;i++)
	{
		data_bit = dat & 0x80;
		SDA = data_bit;
		i2c_clk();
		dat = dat<<1;
	}
	SDA = 1;
	i2c_delay();
	SCL = 1;
	i2c_delay();
	data_bit = SDA;
	SCL = 0;
	i2c_delay();
	return(data_bit);
}

unsigned char i2c_rddata(void)
{
	bit rd_bit;
	unsigned char i,dat;
	dat = 0x00;
	for(i=0;i<8;i++)
	{
		i2c_delay();
		SCL = 1;
		i2c_delay();
		rd_bit = SDA;
		dat = dat<<1;
		dat = dat | rd_bit;
		SCL = 0;
	}
	SDA = 1;
	i2c_delay();
	i2c_clk();
	SCL = 1;
	return(dat);
}
void  Wr_EEPROM(unsigned int  Addr,unsigned char Data)
{
	i2c_start();
	i2c_wrdata(0xA0);
	i2c_wrdata(Addr >> 8);
	i2c_wrdata(Addr & 0x00FF);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
      i2c_wrdata(Data);
      i2c_stop();
      lcd_delay(50);
}

unsigned char   Rd_EEPROM(unsigned int  Addr)
{  unsigned char Temp;
      i2c_start();
      i2c_wrdata(0xA0);
      i2c_wrdata(Addr >> 8);
      i2c_wrdata(Addr & 0x00ff);
      i2c_start();
      i2c_wrdata(0xA1);
      Temp = i2c_rddata();
      i2c_NACK();
      i2c_stop();
      return(Temp);
}
unsigned char scankey(void)
{
    unsigned char d_key,key_data;
      while(~key_press);
        delayfifty(20);
        d_key = P1;
        d_key &= 0x0F;
          if(d_key == 0x00)
            key_data = 0x01;
        else if(d_key == 0x01)
            key_data = 0x02;
        else if(d_key == 0x02)
            key_data = 0x03;
        else if(d_key == 0x04)
            key_data = 0x04;
        else if(d_key == 0x05)
            key_data = 0x05;
        else if(d_key == 0x06)
            key_data = 0x06;
        else if(d_key == 0x08)
            key_data = 0x07;
        else if(d_key == 0x09)
            key_data = 0x08;
        else if(d_key == 0x0A)
            key_data = 0x09;
        else if(d_key == 0x0C)
            key_data = 0x18;
        else if(d_key == 0x0D)
            key_data = 0x00;
        else if(d_key == 0x0E)
            key_data = 0x19;
        else
            key_data = 0x1A;
        return(key_data);
}
unsigned char buff_1(unsigned char buff)
{
      buff &= 0x0F;
      return(buff);
}
unsigned char buff_2(unsigned char buff)
{
      buff &= 0xF0;
      buff = buff >> 4;
      return(buff);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
}
unsigned char buffer(unsigned char bufa , bufb)
{
      bufa = bufa<<4;
      bufa = bufa | bufb;
      return(bufa);
}
void two2lcd(unsigned char st)
{
      unsigned char a;
      a = buff_2(st);
      key2lcd(a);
      a = buff_1(st);
      key2lcd(a);
}

void show_1(unsigned char coun)
{
      lcd_clear();
      lcd_show("     INSERT",0);
      lcd_show("BUS PASS NUM",1);
      key2lcd(coun);
      lcd_blink();
}

void show_2(unsigned char num)
{
      lcd_clear();
      lcd_show("NUMBER BUS",0);
      two2lcd(num);
}

void show_3(void)
{
      lcd_clear();
      lcd_show("CHOOSE WAY",0);
      lcd_show("GO 1  BACK 2",1);
      lcd_blink();
}

void show_4(unsigned char way)
{
      lcd_clear();
      if(way == 0x01)
            {
                  lcd_show("WAY  GO",0);
            }
      else
            {
                  lcd_show("WAY  BACK",0);
            }
}

void show_5(void)
{
      lcd_clear();
      lcd_show("     INSERT",0);
      lcd_show("NUM BUS STOP",1);
      lcd_blink();
}

void show_6(unsigned char stop)
{
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        lcd_clear();
        lcd_show("   BUS STOP",0);
        lcd_show("   NUMBER ",1);
        two2lcd(stop);
}

void sound(unsigned char s)
{
        if (s == 0x00)
        P2 = 0x0A;
        else if (s == 0x01)
        P2 = 0x10;
        else if (s == 0x02)
        P2 = 0x08;
        else if (s == 0x03)
        P2 = 0x18;
        else if (s == 0x04)
        P2 = 0x04;
        else if (s == 0x05)
        P2 = 0x14;
        else if (s == 0x06)
        P2 = 0x0C;
        else if (s == 0x07)
        P2 = 0x1C;
        else if (s == 0x08)
        P2 = 0x02;
        else
        P2 = 0x12;
        PD = 0;
        delayfifty(170);
        PD = 1;
}

unsigned char seven(unsigned char w)
{
        unsigned char s;
        if(w == 0)
                s = 0x77;
        else if(w == 1)
                s = 0x24;
        else if(w == 2)
                s = 0x6B;
        else if(w == 3)
                s = 0x6E;
        else if(w == 4)
                s = 0x3C;
        else if(w == 5)
                s = 0x5E;
        else if(w == 6)
                s = 0x6F;
        else if(w == 7)
                s = 0x64;
        else if(w == 8)
                s = 0x7F;
        else
                s = 0x7E;
        return(s);
}

void LED(unsigned char t,PCF)
{
        i2c_start();
        i2c_wrdata(PCF);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
      i2c_wrdata(0x00);
      i2c_wrdata(t);
      i2c_stop();
}
void show_seven(unsigned char z , d)
{
      unsigned char v;
      if(cou == 0)
            {
            cou++;
            v = buff_2(z);
            v = seven(v);
            LED(v,0x70);
            v = buff_1(z);
            v = seven(v);
            LED(v,0x72);
            line_1 = z;
            line_1ID = d;
            }
      else
            {
            cou--;
            v = buff_2(z);
            v = seven(v);
            LED(v,0x74);
            v = buff_1(z);
            v = seven(v);
            LED(v,0x76);
            line_2 = z;
            line_2ID = d;
            }
}

void turn_led(unsigned char dd)
{
      if(dd == line_1)
      {
            LED(0x00,0x70);
            LED(0x00,0x72);
            line_1 = 0x00;
      }
      if(dd == line_2)
      {
            LED(0x00,0x74);
            LED(0x00,0x76);
            line_2 = 0x00;
      }
}

void show_pres(void)
{
      P2 = 0x1A;
      PD = 0;
      delayfifty(200);
      PD = 1;
}

void show_sound(unsigned char y)
{
      unsigned char x,z;
      x = buff_2(y);
      sound(x);
      delayfifty(10);
```

เอกสารนี้เป็นเอกสารสงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
	PD = 1;
	z = buff_1(y);
	sound(z);
	delayfifty(10);
	PD = 1;
}


void show_afters(void)
{
	P2 = 0x16;
	PD = 0;
	delayfifty(300);
	PD = 1;
}

void receive_num(void)
{
	if (NUM[1] == WAYB)
	{
	if (NUM[2] == MEM[1])
		{
		lcd_clear();
		two2lcd(MEM[1]);
		if (NUM[3] != ID_1)
			{
			ID_1 = NUM[3];
			cou_1 = 0;
			show_seven(MEM[1],ID_1);
			show_pres();
			delayfifty(10);
			PD = 1;
			show_sound(MEM[1]);
			show_afters();
			delayfifty(10);
			PD = 1;
			lcd_clear();
			lcd_show("  NOW SEND DATA",0);
			}
		else
			{
				delayfifty(100);
				lcd_clear();
				lcd_show("  NOW SEND DATA",0);
				cou_1++;
				if(cou_1 == 2)
				{
					turn_led(MEM[1]);
					cou_1 = 0;
				}
			}
		}
	 else if(NUM[2] == MEM[2])
		{
		lcd_clear();
		two2lcd(MEM[2]);
		if (NUM[3] != ID_2)
			{
			ID_2 = NUM[3];
			cou_2 = 0;
			show_seven(MEM[2],ID_2);
			show_pres();
			delayfifty(10);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                PD = 1;
                show_sound(MEM[2]);
                show_afters();
                delayfifty(10);
                PD = 1;
                lcd_clear();
                lcd_show("  NOW SEND DATA",0);
                }
          else
                {
                      delayfifty(100);
                      lcd_clear();
                      lcd_show("  NOW SEND DATA",0);
                      cou_2++;
                      if(cou_2 == 2)
                      {
                            turn_led(MEM[2]);
                            cou_2 = 0;
                      }
                }
          }
     else if(NUM[2] == MEM[3])
          {
          lcd_clear();
          two2lcd(MEM[3]);
          if (NUM[3] != ID_3)
                {
                ID_3 = NUM[3];
                cou_3 = 0;
                show_seven(MEM[3],ID_3);
                show_pres();
                delayfifty(10);
                PD = 1;
                show_sound(MEM[3]);
                show_afters();
                delayfifty(10);
                PD = 1;
                lcd_clear();
                lcd_show("  NOW SEND DATA",0);
                }
          else
                {
                      delayfifty(100);
                      lcd_clear();
                      lcd_show("  NOW SEND DATA",0);
                      cou_3++;
                      if(cou_3 == 2)
                      {
                            turn_led(MEM[3]);
                            cou_3 = 0;
                      }
                }
          }
      else  delayfifty(30);
      }
}

void service_serial(void) interrupt 4
{
      if(RI)
      {
            RI = 0;
            NUM[RXin] = SBUF;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
            RXin++;
            if(RXin == 4)
            {
                  RXin = 0;
                  receive_num();
            }
      }
}

void service_0(void) interrupt 0
{
      TRB = 1;
      TRA = 1;
      delayfifty(100);
      SBUF = 0x55;
      while(~TI);
      TI = 0;
      SBUF = WAYC;
      while(~TI);
      TI = 0;
      SBUF = line_1;
      while(~TI);
      TI = 0;
      SBUF = MEM[5];
      while(~TI);
      TI = 0;
      delayfifty(100);
      TRA = 0;
      TRB = 0;
}

void service_1(void) interrupt 2
{
      TRB = 1;
      TRA = 1;
      delayfifty(100);
      SBUF = 0x55;
      while(~TI);
      TI = 0;
      SBUF = WAYC;
      while(~TI);
      TI = 0;
      SBUF = line_2;
      while(~TI);
      TI = 0;
      SBUF = MEM[5];
      while(~TI);
      TI = 0;
      delayfifty(100);
      TRA = 0;
      TRB = 0;
}
void set_num(void)
{
      unsigned char keya , keyb;
      unsigned char count = 0;
            show_3();
            keya = scankey();
            MEM[4] = keya;
            delayfifty(30);
            lcd_clear();
            show_5();
            keya = scankey();
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
            delayfifty(20);
            key2lcd(keya);
            lcd_blink();
            keyb = scankey();
            key2lcd(keyb);
            delayfifty(20);
            MEM[5] = buffer(keya,keyb);
            do
            {
            count++;
            show_1(count);
            keya = scankey();
            delayfifty(20);
            key2lcd(keya);
            lcd_blink();
            keyb = scankey();
            key2lcd(keyb);
            MEM[count] = buffer(keya,keyb);
            delayfifty(100);
            lcd_clear();
            lcd_show("EXIT PRESS    #",0);
            lcd_show("MEM PRESS    *",1);
            lcd_blink();
            keyb = scankey();
            }
            while(keyb == 0x18);
                  Wr_EEPROM(0x05FF,MEM[1]);
                  Wr_EEPROM(0x04FF,MEM[2]);
                  Wr_EEPROM(0x03FF,MEM[3]);
                  Wr_EEPROM(0x02FF,MEM[4]);
                  Wr_EEPROM(0x01FF,MEM[5]);
}

void check_num(void)
{
      unsigned char a , r;
      MEM[1] = Rd_EEPROM(0x05FF);
      MEM[2] = Rd_EEPROM(0x04FF);
      MEM[3] = Rd_EEPROM(0x03FF);
      MEM[4] = Rd_EEPROM(0x02FF);
      MEM[5] = Rd_EEPROM(0x01FF);
      show_4(MEM[4]);
      lcd_delay(750);
      show_6(MEM[5]);
      lcd_delay(750);
      for (a=1;a<4;a++)
      {
      lcd_clear();
      show_2(MEM[a]);
      r = buff_2(MEM[a]);
      r = seven(r);
      LED(r,0x70);
      r = buff_1(MEM[a]);
      r = seven(r);
      LED(r,0x72);
      show_sound(MEM[a]);
      lcd_delay(1200);
      LED(0x00,0x70);
      LED(0x00,0x72);
      }
}

void send_num(void)
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
{
      unsigned char a;
      lcd_clear();
      lcd_show("  NOW SEND DATA",0);
      MEM[1] = Rd_EEPROM(0x05FF);
      MEM[2] = Rd_EEPROM(0x04FF);
      MEM[3] = Rd_EEPROM(0x03FF);
      MEM[4] = Rd_EEPROM(0x02FF);
      MEM[5] = Rd_EEPROM(0x01FF);
      WAYN = 0xA8 | MEM[4];
      WAYC = 0xA4 | MEM[4];
      WAYB = 0x50 | MEM[4];
      TMOD = 0x20;
      SCON = 0x50;
      TH1 = 0x1D;
      TL1 = 0x1D;
      RI = 0;
      TI = 0;
      EA = 1;
      ES = 0;
      IT0 = 1;
      IT1 = 1;
      EX0 = 1;
      EX1 = 1;
      TR1 = 1;
      while(1)
      {
      for(a=0;a<3;a++)
      {
      TRB = 1;
      TRA = 1;
      delayfifty(100);
      SBUF = 0x55;
      while(~TI);
      TI = 0;
      SBUF = WAYN;
      while(~TI);
      TI = 0;
      SBUF = MEM[a];
      while(~TI);
      TI = 0;
      SBUF = MEM[5];
      while(~TI);
      TI = 0;
      delayfifty(100);
      TRA = 0;
      TRB = 0;
      ES = 1;
      delayfifty(1000);
      ES = 0;
      }
      a = 0;
      }
}

void manual(void)
{
      lcd_clear();
      lcd_show("MANUAL MODE",0);
      MEM[1] = Rd_EEPROM(0x05FF);
      MEM[2] = Rd_EEPROM(0x04FF);
      MEM[3] = Rd_EEPROM(0x03FF);
      MEM[4] = Rd_EEPROM(0x02FF);
```

เอกสารนี้เป็นทรัพย์สินของจุฬาลงกรณ์มหาวิทยาลัย สงวนไว้สำหรับการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        MEM[5] = Rd_EEPROM(0x01FF);
        WAYB = 0x50 | MEM[4];
        TMOD = 0x20;
        SCON = 0x50;
        TH1 = 0x1D;
        TL1 = 0x1D;
        RI = 0;
        TI = 0;
        TR1 = 1;
        while(1)
        {
              while(~RI);
              RI = 0;
              NUM[4] = SBUF;
              if(NUM[4] == 0x55)
              {
                    while(~RI);
                    RI = 0;
                    NUM[1] = SBUF;
                    while(~RI);
                    RI = 0;
                    NUM[2] = SBUF;
                    while(~RI);
                    RI = 0;
                    NUM[3] = SBUF;
                    receive_num();
              }
        }
}

void main (void)
{
        unsigned char key;
        P1 = 0xCF;
        LED(0x00,0x70);
        LED(0x00,0x72);
        LED(0x00,0x74);
        LED(0x00,0x76);
        lcd_init();
        lcd_show("     WELCOME",0);
        delayfifty(40);
        while(1)
        {
              lcd_clear();
              lcd_show("MAIN PROGRAM",0);
              lcd_show("PRESS 1",1);
              delayfifty(100);
              lcd_clear();
              lcd_show("SET STATION",0);
              lcd_show("PRESS 2",1);
              delayfifty(100);
              lcd_clear();
              lcd_show("CHECK STATION",0);
              lcd_show("PRESS 3",1);
              delayfifty(100);
              lcd_clear();
              lcd_show("MANUAL MODE",0);
              lcd_show("PRESS 4",1);
              delayfifty(100);
              lcd_clear();
              lcd_show("CHOOSE NUMBER",0);
              lcd_blink();
              key = scankey();
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
      if(key == 0x01)    {
            key2lcd(key);
            delayfifty(25);
            send_num();
          }
          else if(key == 0x02)  {
            key2lcd(key);
            delayfifty(25);
            set_num();
          }
          else if(key == 0x03)  {
            key2lcd(key);
            delayfifty(25);
            check_num();
          }
          else if(key == 0x04)  {
            key2lcd(key);
            delayfifty(25);
            manual();
          }
       else  delayfifty(25);
       }

}
```

โปรแกรมในส่วนของรถประจำทาง

```
#include<AT89X51.H>
#include <string.h>
#include<INTRINS.H>

#define E P3_6
#define RS P3_7
#define SDA P2_6
#define SCL P2_7
#define key_press P3_2
#define CALL P3_3
#define PD P1_7
#define LAMP P1_6
#define TRA P1_5
#define TRB P1_4

unsigned char NUM[6],MEM,ID,WAY,WAYB,WAYSN,WAYSC;
unsigned char RXin = 1,NUMS = 0,cou = 0;

void delayfifty(int fiftyms)
{
       int x;
       for(x=0;x<fiftyms;x++)
       {
         TH0 = 0x4c;
         TL0 = 0x00;
         TF0 = 0;
         TR0 = 1;
         while(TF0==0);
         TR0 = 0;
       }
}
void lcd_delay(int tick)
{
      int i,j;
      for(i=0;i<tick;i++)
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    for(j=0;j<250;j++);
}

void lcd_command(unsigned char com)
{
    RS = 0;
    E =1;
    P0 = com;
    lcd_delay(5);
    E = 0;
    lcd_delay(5);
}

void lcd_text(unsigned char text)
{
    RS = 1;
    E = 1;
    P0 = text;
    lcd_delay(5);
    E = 0;
    lcd_delay(5);
}

void lcd_init()
{
      lcd_delay(500);
      lcd_command(0x38);
      lcd_command(0x0C);
      lcd_command(0x01);
}

void lcd_show(unsigned char text[],bit row)
{
      unsigned char a;
      if (row == 0)
            lcd_command(0x80);
      else
            lcd_command(0xC0);
      for(a=0;a<strlen(text);a++)
           lcd_text(text[a]);
}

void key2lcd(unsigned char pad)
{
      pad += 0x30;
      lcd_text(pad);
}

void lcd_clear(void)
{
      lcd_command(0x01);
}

void lcd_blink()
{
      lcd_command(0x0F);
}
void i2c_delay(void)
{
      unsigned char i;
      for(i=0;i<21;i++)
      _nop_();
}
```

เอกสารนี้เป็นเอกสารสงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
void i2c_clk(void)
{
      i2c_delay();
      SCL = 1;
      i2c_delay();
      SCL = 0;
}

void i2c_start(void)
{
      if(SCL)
      SCL = 0;
      SDA = 1;
      SCL = 1;
      i2c_delay();
      SDA = 0;
      i2c_delay();
      SCL = 0;
}

void i2c_stop(void)
{
      if(SCL)
      SCL = 0;
      SDA = 0;
      i2c_delay();
      SCL = 1;
      i2c_delay();
      SDA = 1;
}

void i2c_NACK(void)
{
      SDA = 1;
      i2c_delay();
      i2c_clk();
      SCL = 1;
}

bit i2c_wrdata(unsigned char dat)
{
      bit data_bit;
      unsigned char i;
      for(i=0;i<8;i++)
      {
            data_bit = dat & 0x80;
            SDA = data_bit;
            i2c_clk();
            dat = dat<<1;
      }
      SDA = 1;
      i2c_delay();
      SCL = 1;
      i2c_delay();
      data_bit = SDA;
      SCL = 0;
      i2c_delay();
      return(data_bit);
}

unsigned char i2c_rddata(void)
{
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับใช้เพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
      bit rd_bit;
      unsigned char i,dat;
      dat = 0x00;
      for(i=0;i<8;i++)
      {
            i2c_delay();
            SCL = 1;
            i2c_delay();
            rd_bit = SDA;
            dat = dat<<1;
            dat = dat | rd_bit;
            SCL = 0;
      }
      SDA = 1;
      i2c_delay();
      i2c_clk();
      SCL = 1;
      return(dat);
}
void  Wr_EEPROM(unsigned int  Addr,unsigned char Data)
{
      i2c_start();
      i2c_wrdata(0xA0);
      i2c_wrdata(Addr >> 8);
      i2c_wrdata(Addr & 0x00FF);
      i2c_wrdata(Data);
      i2c_stop();
      lcd_delay(50);
}

unsigned char   Rd_EEPROM(unsigned int  Addr)
{  unsigned char Temp;
      i2c_start();
      i2c_wrdata(0xA0);
      i2c_wrdata(Addr >> 8);
      i2c_wrdata(Addr & 0x00ff);
      i2c_start();
      i2c_wrdata(0xA1);
      Temp = i2c_rddata();
      i2c_NACK();
      i2c_stop();
      return(Temp);
}
unsigned char scankey(void)
{
    unsigned char d_key,key_data;
      while(~key_press);
        delayfifty(20);
        d_key = P1;
        d_key &= 0x0F;
          if(d_key == 0x00)
            key_data = 0x01;
        else if(d_key == 0x01)
            key_data = 0x02;
        else if(d_key == 0x02)
            key_data = 0x03;
        else if(d_key == 0x04)
            key_data = 0x04;
        else if(d_key == 0x05)
            key_data = 0x05;
        else if(d_key == 0x06)
            key_data = 0x06;
        else if(d_key == 0x08)
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
            key_data = 0x07;
        else if(d_key == 0x09)
            key_data = 0x08;
        else if(d_key == 0x0A)
            key_data = 0x09;
        else if(d_key == 0x0C)
            key_data = 0x18;
        else if(d_key == 0x0D)
            key_data = 0x00;
        else if(d_key == 0x0E)
            key_data = 0x19;
        else
            key_data = 0x1A;
        return(key_data);
}
unsigned char buff_1(unsigned char buff)
{
      buff &= 0x0F;
      return(buff);
}
unsigned char buff_2(unsigned char buff)
{
      buff &= 0xF0;
      buff = buff >> 4;
      return(buff);
}
unsigned char buffer(unsigned char bufa , bufb)
{
      bufa = bufa<<4;
      bufa = bufa | bufb;
      return(bufa);
}
void two2lcd(unsigned char st)
{
      unsigned char a;
      a = buff_2(st);
      key2lcd(a);
      a = buff_1(st);
      key2lcd(a);
}

void show_1(void)
{
      lcd_clear();
      lcd_show("     INSERT",0);
      lcd_show("NUMBER BUS",1);
      lcd_blink();
}

void show_2(void)
{
      lcd_clear();
      lcd_show("     INSERT",0);
      lcd_show("BUS ID",1);
      lcd_blink();
}

void show_3(unsigned char num,id)
{
      lcd_clear();
      lcd_show("NUMBER BUS",0);
      two2lcd(num);
      lcd_show("BUS ID    ",1);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        two2lcd(id);
}

void show_4(void)
{
        lcd_clear();
        lcd_show("CHOOSE WAY",0);
        lcd_show("GO 1  BACK 2",1);
        lcd_blink();
}

void show_5(unsigned char way)
{
        lcd_clear();
        if(way == 0x01)
                {
                        lcd_show("WAY  GO",0);
                }
        else
                {
                        lcd_show("WAY  BACK",0);
                }
}

void sound(unsigned char s)
{
        if (s == 0x00)
        P2 = 0x0A;
        else if (s == 0x01)
        P2 = 0x10;
        else if (s == 0x02)
        P2 = 0x08;
        else if (s == 0x03)
        P2 = 0x18;
        else if (s == 0x04)
        P2 = 0x04;
        else if (s == 0x05)
        P2 = 0x14;
        else if (s == 0x06)
        P2 = 0x0C;
        else if (s == 0x07)
        P2 = 0x1C;
        else if (s == 0x08)
        P2 = 0x02;
        else
        P2 = 0x12;
        PD = 0;
        delayfifty(170);
        PD = 1;
}

void show_sound(unsigned char y)
{
        unsigned char x,z;
        x = buff_2(y);
        sound(x);
        delayfifty(10);
        PD = 1;
        z = buff_1(y);
        sound(z);
        delayfifty(10);
        PD = 1;
}
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
void show_pres(void)
{
      P2 = 0x1A;
      PD = 0;
      delayfifty(220);
      PD = 1;
}

void show_call(void)
{
      P2 = 0x16;
      PD = 0;
      delayfifty(400);
      PD = 1;
}

void receive(void)
{
      if(NUM[1] == WAYSN)
      {
            if(NUM[2] == MEM)
            {
            TRA = 1;
            TRB = 1;
            delayfifty(150);
            SBUF = 0x55;
            while(~TI);
            TI = 0;
            SBUF = WAYB;
            while(~TI);
            TI = 0;
            SBUF = MEM;
            while(~TI);
            TI = 0;
            SBUF = ID;
            while(~TI);
            TI = 0;
            delayfifty(100);
            TRA = 0;
            TRB = 0;
            lcd_clear();
            two2lcd(NUM[3]);
            cou++;
                  if(NUM[3] != NUMS)
                  {
                  NUMS = NUM[3];
                  show_pres();
                  delayfifty(10);
                  PD = 1;
                  show_sound(NUM[3]);
                  lcd_clear();
                  lcd_show("NOW RECEIVE DATA",0);
                  }
                  else
                  {
                  delayfifty(300);
                  lcd_clear();
                  lcd_show("NOW RECEIVE DATA",0);
                  }
            }
      }
      else if(NUM[1] == WAYSC)
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        {
              if(NUM[2] == MEM)
              {
                    LAMP = 0;
                    show_call();
                    delayfifty(10);
                    PD = 1;
                    cou = 0;
              }
        }
}

void send(void)
{
        TRB = 1;
        TRA = 1;
        delayfifty(150);
        SBUF = 0x55;
        while(~TI);
        TI = 0;
        SBUF = WAYB;
        while(~TI);
        TI = 0;
        SBUF = MEM;
        while(~TI);
        TI = 0;
        SBUF = ID;
        while(~TI);
        TI = 0;
        delayfifty(100);
        TRA = 0;
        TRB = 0;
}

void set_num(void)
{
        unsigned char keya , keyb;
              show_4();
              keya = scankey();
              WAY = keya;
              Wr_EEPROM(0x03FF,WAY);
              delayfifty(30);
              lcd_clear();
              lcd_show("EXIT PRESS   #",0);
              lcd_show("MEM PRESS   *",1);
              lcd_blink();
              keyb = scankey();
              if(keyb == 0x18)
              {
              show_1();
              keya = scankey();
              key2lcd(keya);
              lcd_blink();
              keyb = scankey();
              key2lcd(keyb);
              MEM = buffer(keya,keyb);
              Wr_EEPROM(0x02FF,MEM);
              delayfifty(50);
              show_2();
              keya = scankey();
              key2lcd(keya);
              lcd_blink();
              keyb = scankey();
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
            key2lcd(keyb);
            ID = buffer(keya,keyb);
            Wr_EEPROM(0x01FF,ID);
            delayfifty(100);
            }
}

void check_num(void)
{
      WAY = Rd_EEPROM(0x03FF);
      MEM = Rd_EEPROM(0x02FF);
      ID = Rd_EEPROM(0x01FF);
      lcd_init();
      show_5(WAY);
      lcd_delay(1200);
      show_3(MEM,ID);
      lcd_delay(1200);
}

void receive_num(void)
{
      lcd_clear();
      lcd_show("NOW RECEIVE DATA",0);
      WAY = Rd_EEPROM(0x03FF);
      MEM = Rd_EEPROM(0x02FF);
      ID = Rd_EEPROM(0x01FF);
      WAYB = 0x50 | WAY;
      WAYSN = 0xA8 | WAY;
      WAYSC = 0xA4 | WAY;
      TMOD = 0x20;
      SCON = 0x50;
      TH1 = 0x1D;
      TL1 = 0x1D;
      RI = 0;
      TI = 0;
      TR1 = 1;
      while(1)
      {
            while(~RI);
            RI = 0;
            NUM[4] = SBUF;
            if(NUM[4] == 0x55)
            {
                  while(~RI);
                  RI = 0;
                  NUM[1] = SBUF;
                  while(~RI);
                  RI = 0;
                  NUM[2] = SBUF;
                  while(~RI);
                  RI = 0;
                  NUM[3] = SBUF;
                  delayfifty(200);
                  receive();
                  if(cou == 2)
                  {
                        LAMP = 1;
                        cou = 0;
                  }
            }
      }
}
```

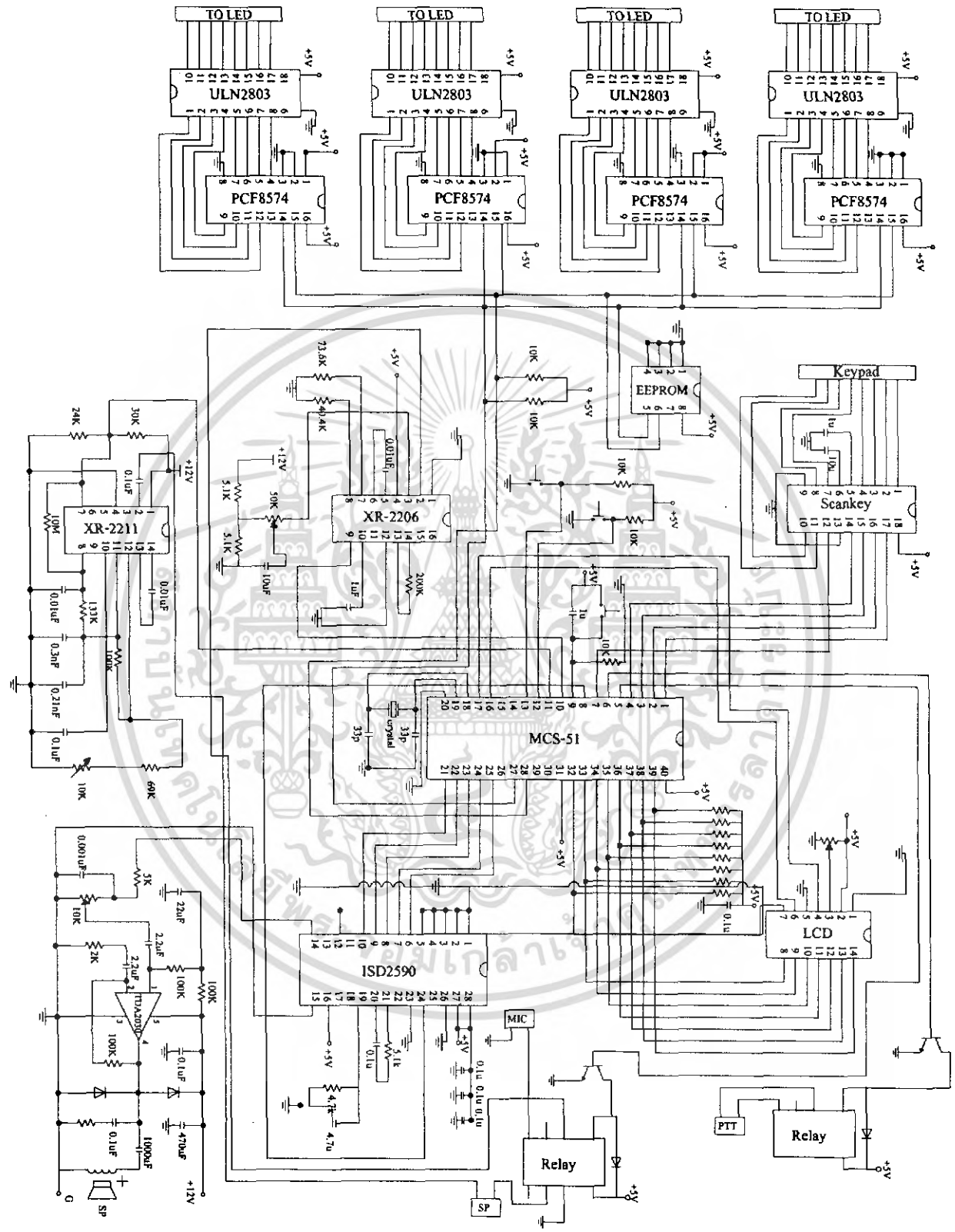
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
void manual(void)
{
	lcd_clear();
	lcd_show("MANUAL SEND",0);
	WAY = Rd_EEPROM(0x03FF);
	MEM = Rd_EEPROM(0x02FF);
	ID = Rd_EEPROM(0x01FF);
	WAYB = 0x50 | WAY;
	TMOD = 0x20;
	SCON = 0x50;
	TH1 = 0x1D;
	TL1 = 0x1D;
	RI = 0;
	TI = 0;
	EA = 1;
	ES = 0;
	IT1 = 1;
	EX1 = 1;
	TR1 = 1;
	while(1)
	{
		while(~CALL);
		send();
	}
}

void main (void)
{
	unsigned char key;
	TRA = 0;
	TRB = 0;
	PD = 1;
	LAMP = 1;
	lcd_init();
	lcd_show("     WELCOME",0);
	delayfifty(40);
	while(1)
	{
		lcd_clear();
		lcd_show("MAIN PROGRAM",0);
		lcd_show("PRESS 1",1);
		delayfifty(100);
		lcd_clear();
		lcd_show("SET NUMBER BUS",0);
		lcd_show("PRESS 2",1);
		delayfifty(100);
		lcd_clear();
		lcd_show("CHECK NUMBER BUS",0);
		lcd_show("PRESS 3",1);
		delayfifty(100);
		lcd_clear();
		lcd_show("MANUAL MODE",0);
		lcd_show("PRESS 4",1);
		delayfifty(100);
		lcd_clear();
		lcd_show("CHOOSE NUMBER",0);
		lcd_blink();
		key = scankey();
	if(key == 0x01)   {
		key2lcd(key);
		delayfifty(25);
		receive_num();
	}
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

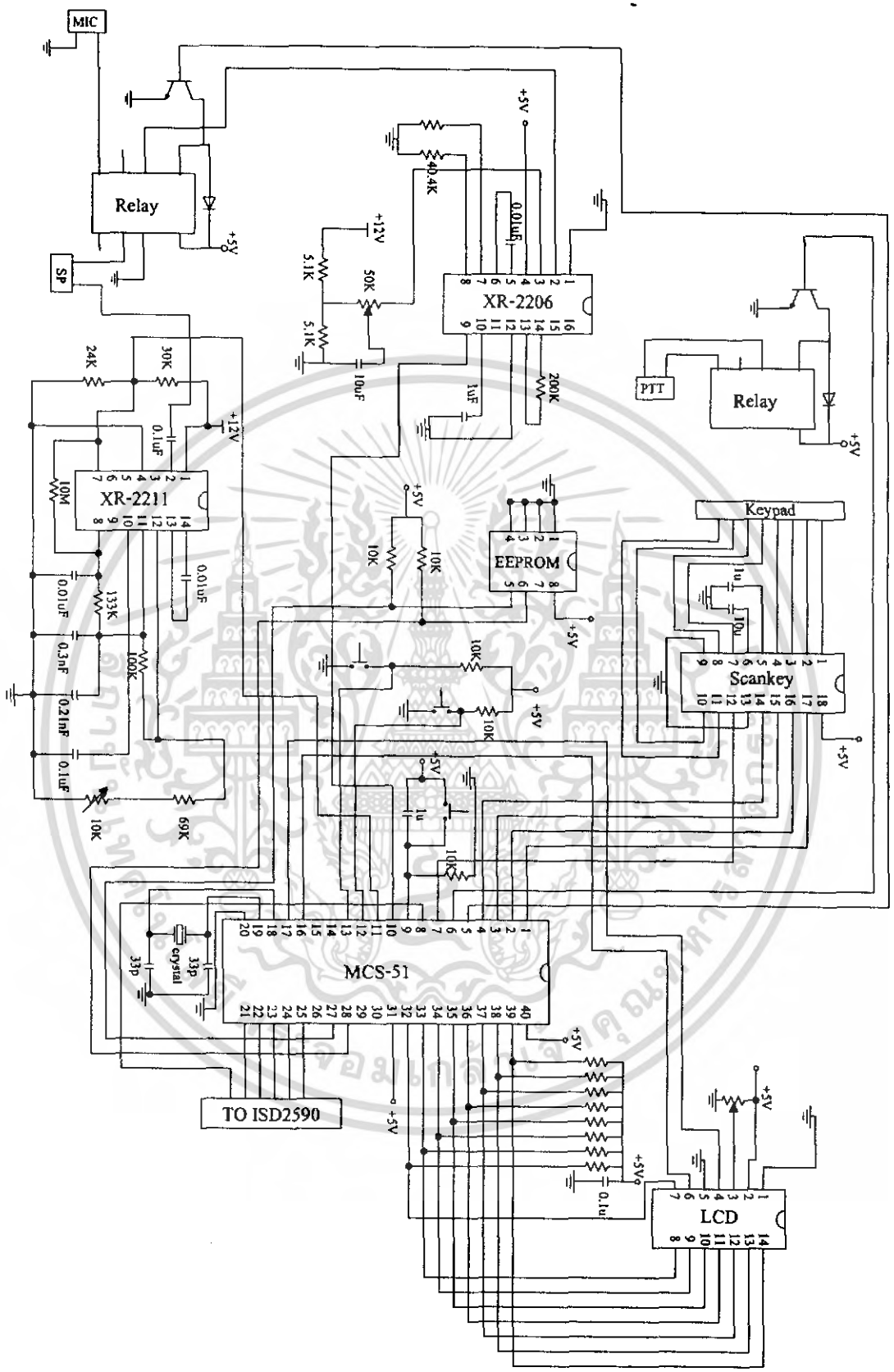
```
        else if(key == 0x02)  {
            key2lcd(key);
            delayfifty(25);
            set_num();
        }
        else if(key == 0x03)  {
            key2lcd(key);
            delayfifty(25);
            check_num();
        }
        else if(key == 0x04)  {
            key2lcd(key);
            delayfifty(25);
            manual();
        }
      else  delayfifty(25);
      }

}
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปแสดงวงจรรวมที่ป้ายรถประจำทาง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

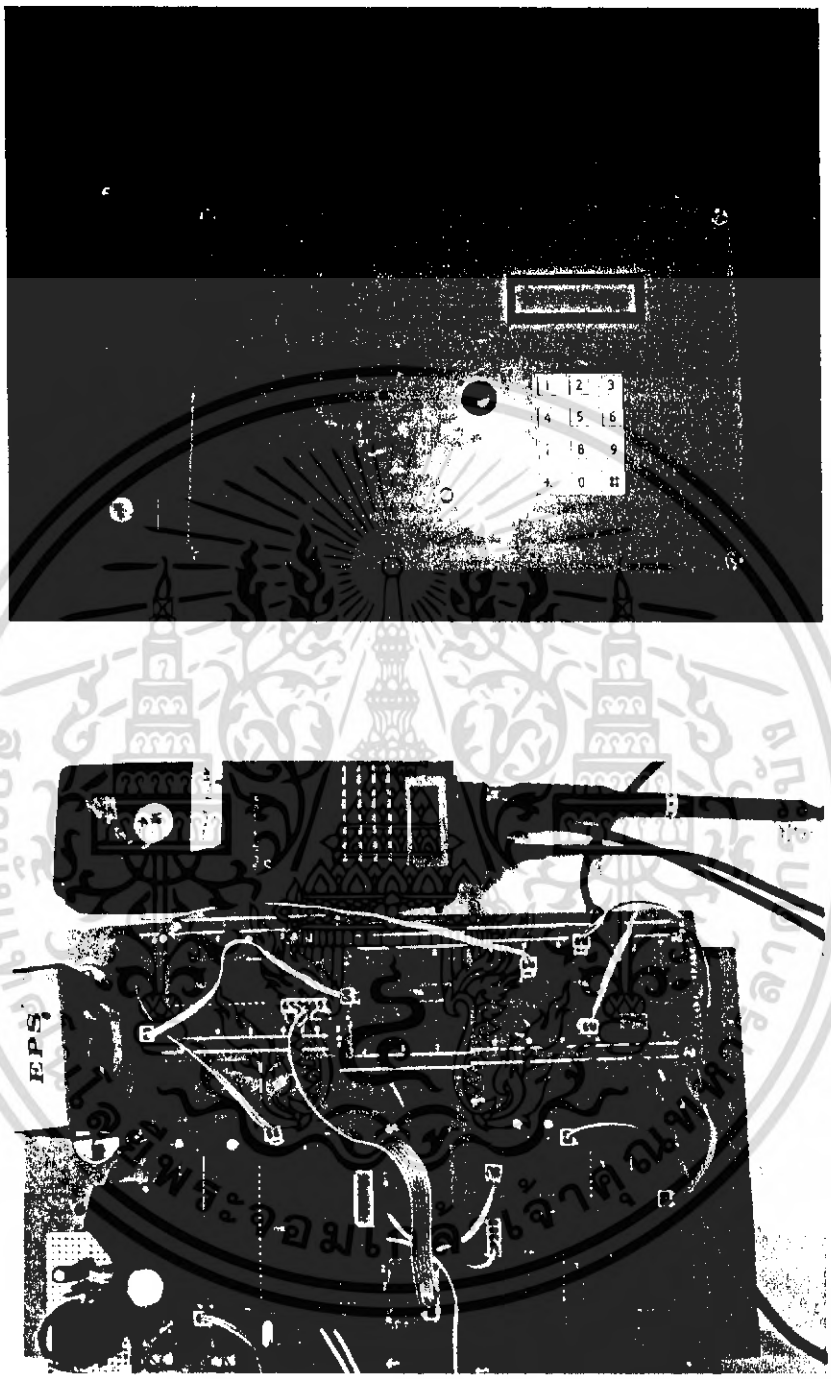


รูปแสดงวงจรรวมที่รถประจำทาง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปแสดงอุปกรณ์ที่ป้ายรถประจำทาง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปแสดงอุปกรณ์ที่รถประจำทาง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้