สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

รถสำรวจ

EXPLORER CAR

โดย

นางสาวอภิญญา ศรีสุวรรณ

นางสาวอภิญญา เอี่ยมโพธิ์

นางสาวอัจฉรา บุญประทักษ์เวช

เลขหมู่.....
เลขทะเบียน 103052
วัน,เดือน,ปี 24 ส.ค. 2552

.b.....
.i.....

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต

สาขาวิชาวิศวกรรมโทรคมนาคม

คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2551

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รถสำรวจ

EXPLORER CAR

โดย

นางสาวอภิญญา ศรีสุวรรณ รหัส 48011063

นางสาวอภิญญา เอี่ยมโพธิ์ รหัส 48011065

นางสาวอัจฉรา บุญประทักษ์เวช รหัส 48011102

อาจารย์ที่ปรึกษา

ผศ.ดร.พรชัย ทรัพย์นิธิ

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต

สาขาวิชาวิศวกรรมโทรคมนาคม

คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2551

ภาควิชา
วิศวกรรมโทรคมนาคม

ผ่านการตรวจรูปเล่มแล้ว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ปริญญานิพนธ์ปีการศึกษา 2551

ภาควิชาวิศวกรรมโทรคมนาคม

คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เรื่อง รถสำรวจ

EXPLORER CAR

ผู้จัดทำ

1. นางสาวอภิญญา ศรีสุวรรณ
2. นางสาวอภิญญา เอี่ยมโพธิ์
3. นางสาวอัจฉรา บุญประทักษ์เวช

อาจารย์ที่ปรึกษา

(ผศ.ดร.พรชัย ทรัพย์นิธิ)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รถสำรวจ

EXPLORER CAR

โดย นางสาวอภิญญา ศรีสุวรรณ 48011063
นางสาวอภิญญา เอี่ยมโพธิ์ 48011065
นางสาวอัจฉรา บุญประทักษ์เวช 48011102

อาจารย์ที่ปรึกษา ผศ.ดร.พรชัย ทรัพย์นิธิ

บทคัดย่อ

โครงงานนี้นำเสนอระบบรักษาความปลอดภัย โดยใช้รถบังคับที่มีการติดต่อกล้องวีดีโอนำมาใช้ประโยชน์ในการสำรวจบริเวณที่เข้าถึงได้ยากหรือบริเวณที่อาจก่อให้เกิดอันตรายกับมนุษย์ ซึ่งสามารถสังเกตเห็นเหตุกาณ์ต่างๆ ผ่านทางหน้าจอคอมพิวเตอร์และเมื่อมีเหตุการณ์ผิดปกติสามารถทำการบันทึกภาพเป็นวีดีโอและเรียกดูเหตุการณ์ที่บันทึกได้ โดยจะมีการบอกรายละเอียดของวันที่และเวลาที่เกิดเหตุการณ์นั้นๆด้วย

ABSTRACT

This project proposes the security system by using the radio car with monitoring camera to explore the area difficult to be accessible or the area causing damage to humans. We can observe situations through a computer screen. When there is something unusual, we can record an event in the form of video and recall that event with the details in terms of date and time that it took place.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ

หน้า

บทที่ 1 บทนำ

1.1 แนวคิดและเหตุผลของโครงงาน 1

1.2 วัตถุประสงค์ของโครงงาน 1

1.3 ขั้นตอนการดำเนินงาน 1

1.4 ประโยชน์ที่ได้รับ 1

1.5. ภาพรวมของโครงงาน 2

บทที่ 2 ทฤษฎีและหลักการ

2.1 ไมโครคอนโทรลเลอร์ MCS-51 3

2.1.1 โครงสร้างภายในของไมโครคอนโทรลเลอร์ MCS-51 4

2.1.2 คุณสมบัติทั่วไปของไมโครคอนโทรลเลอร์ MCS-51 5

2.1.3 การจัดการของไมโครคอนโทรลเลอร์ตระกูล MCS-51 6

2.2 ไทเมอร์ / เคาน์เตอร์ (Timer/Counter) 8

2.2.1 การทำงานในโหมดไทเมอร์ (Timer Mode) ของไทเมอร์/เคาน์เตอร์ 0 และ 1 9

2.2.2 การทำงานในโหมดเคาน์เตอร์ (Counter Mode) ของไทเมอร์/เคาน์เตอร์ 0 และ 1 9

2.2.3 รีจิสเตอร์ของไทเมอร์/ เคาน์เตอร์ 0 และ 1 (Timer / Counter0,1 Register) 9

2.2.4 รีจิสเตอร์ควบคุมการทำงานของไทเมอร์/เคาน์เตอร์ 0 และ1 10

2.2.5 รีจิสเตอร์เลือกโหมดการทำงานของไทเมอร์/เคาน์เตอร์ 0 และ 1 11

2.3 การคำนวณความเร็วการรับและส่งข้อมูลแบบอนุกรม (Generating Baud Rate) 12

2.4 RS-232 14

2.4.1 การเชื่อมต่อแบบ Null modem 16

2.4.2 ระดับแรงดัน 16

2.5 มอเตอร์กระแสตรง (DC Motor) 17

2.5.1 หลักการเบื้องต้นของมอเตอร์ 17

2.5.2 โครงสร้างของมอเตอร์กระแสตรง (DC Motor Structure) 17

2.5.3 วงจร H-Bridge Switching ควบคุมมอเตอร์รถ 19

2.6 สเต็ปมอเตอร์ (Stepping Motor) 21

2.6.1 ชนิดและโครงสร้างของสเต็ปมอเตอร์ 22

2.6.2 การตรวจสอบหาสายร่วม และสายกราวด์ ของสเต็ปมอเตอร์ แบบ PM (แบบแกนโรเตอร์เป็นแม่เหล็กถาวร) 24

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

	หน้า
2.6.3 วงจรขับสเต็ปปิ้งมอเตอร์	25
2.6.4 ตัวอย่างวงจรขับสเต็ปปิ้งมอเตอร์	25
2.7 โมดูล RS232 to RF-Wireless (RF2.4GHz) CONVERTER รุ่น ET-RF24G V1.0	26
2.8 วงจรรักษาแรงดันใช้ไอซีรักษาแรงดัน (IC Regulator)	27
2.9 กล้องโทรทัศน์วงจรปิด	27
2.9.1 การเปรียบเทียบเทคโนโลยีของ CMOS และ CD	29
2.9.2 ลักษณะของสัญญาณภาพ	29
2.9.3 การสแกนภาพและเรื่องที่เกี่ยวข้อง	32
บทที่ 3 การออกแบบและการสร้าง	
3.1 ส่วนของภาคส่งสัญญาณ	35
3.1.1 โปรแกรมควบคุมจากคอมพิวเตอร์	36
3.1.2 Flow Chart การทำงานของโปรแกรม	37
3.2 ส่วนของภาครับสัญญาณ	37
3.2.1 ไมโครคอนโทรลเลอร์	38
3.2.2 Flow Chart การทำงานของภาครับ	38
3.2.3 วงจรขับมอเตอร์	39
3.4 การติดตั้งและการใช้งานชุดรับส่งสัญญาณภาพ	39
3.3 การติดตั้งและ กำหนดค่าต่างๆ ของเครื่อง ET-RF24G V1.0	40
บทที่ 4 ผลการทดลอง และวิเคราะห์ผลการทดลอง	
4.1 ผลการทดลองของภาครับและภาคส่ง ของรถสำรวจ	41
4.1.1 กดปุ่มเลี้ยวซ้าย	41
4.1.2 กดปุ่ม เดินหน้า	42
4.1.3 กดปุ่ม เลี้ยวขวา	43
4.1.4 กดปุ่มถอยหลังทางซ้าย	44
4.1.5 กดปุ่ม ถอยหลัง	45
4.1.6 กดปุ่ม ถอยหลังทางขวา	46
4.2 การทดลองและผลการทดลองการใช้โมดูล RF Wireless	47
4.3 ผลการทดลองวัดระยะทางที่รถสำรวจสามารถวิ่งได้จากการรับส่งสัญญาณ จากเครื่องส่งสัญญาณRF ในบริเวณพื้นที่โล่ง	48

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ (ต่อ)

หน้า

4.4 ผลการทดลองลักษณะภาพที่รับได้จากกล้อง CCD ที่ระยะต่างๆในพื้นที่โล่ง 48

4.5 ผลการทดลองเมื่อควบคุมรถสำรวจให้ออกไปทำการสำรวจและทำการส่งภาพกลับมาจากกล้องวีดีโอที่เราสามารถรับสัญญาณภาพได้ 49

4.6 ผลการทดลองการควบคุมให้สเตปปิ้งมอเตอร์หมุนโดยสั่งการจากไมโครคอนโทรลเลอร์ 49

4.5.1 ผลการทดลองสั่งให้สเตปปิ้งมอเตอร์หมุนตามเข็มนาฬิกา 50

4.5.2 ผลการทดลองสั่งให้สเตปปิ้งมอเตอร์หมุนทวนเข็มนาฬิกา 51

บทที่ 5 สรุปผลและวิจารณ์การทดลอง

5.1 บทสรุป 53

5.2 วิจารณ์ 53

5.3 แนวทางการพัฒนา 53

กิตติกรรมประกาศ

เอกสารอ้างอิง

ภาคผนวก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูปภาพ

	หน้า
รูปที่ 1.1 ภาพรวมของโครงงาน	2
รูปที่ 2.1 บล็อกไดอะแกรมของไมโครคอนโทรลเลอร์ตระกูล MCS-51	3
รูปที่ 2.2 ขาต่างๆของ 8051	6
รูปที่ 2.3 การต่อแหล่งกำเนิดสัญญาณภายนอกให้กับ 8051	8
รูปที่ 2.4 โครงสร้างโดยรวมของไทเมอร์ / เคาน์เตอร์	9
รูปที่ 2.5 บิตต่างๆในรีจีสเตอร์ TCON	10
รูปที่ 2.6 บิตต่างๆในรีจีสเตอร์ TMOD	11
รูปที่ 2.7 พอร์ตอนุกรม DB9 ตัวผู้	15
รูปที่ 2.8 พอร์ตอนุกรม DB9 ตัวเมีย	15
รูปที่ 2.9 DB9 ตัวผู้ เมื่อมองจากด้านหลัง	15
รูปที่ 2.10 การเชื่อมต่ออุปกรณ์ภายนอกผ่าน DB9 แบบ Null modem	16
รูปที่ 2.11 ระดับแรงดันในมาตรฐาน RS – 232	17
รูปที่ 2.12 โครงสร้างพื้นฐานของมอเตอร์กระแสตรง	18
รูปที่ 2.13 การทำงาน ของ วงจร H-Bridge Switching ควบคุมมอเตอร์รถ	18
รูปที่ 2.14 อธิบายการทำงาน ของ วงจร H-Bridge Switching ควบคุมมอเตอร์รถ	19
รูปที่ 2.15 อธิบายการทำงาน ของ วงจร H-Bridge Switching ควบคุมมอเตอร์รถ	19
รูปที่ 2.16 สเต็ปมอเตอร์แบบมีสาย 5 เส้น	20
รูปที่ 2.17 สเต็ปมอเตอร์แบบมีสาย 6 เส้น	20
รูปที่ 2.18 สเต็ปมอเตอร์หลายแบบ ไบโพลาร์	21
รูปที่ 2.19 ภาพถ่ายโครงสร้างสเต็ปมอเตอร์	21
รูปที่ 2.20 (ก) โครงสร้าง (ข) วงจรเทียบเท่า (equivalent circuit) ของมอเตอร์ ชนิด 4 ขด	22
รูปที่ 2.21 วงจรขับสเต็ปปิ้งมอเตอร์โดยใช้ไอซีเบอร์ ULN2003	24
รูปที่ 2.22 วงจรรักษาแรงดันแบบแรงดันคงที่ใช้ MC 78xx	26
รูปที่ 2.23 กล้องโทรทัศน์วงจรปิดชนิด CCD	27
รูปที่ 2.24 กล้องโทรทัศน์วงจรปิดชนิด CMOS	27
รูปที่ 2.25 ลักษณะของสัญญาณทางด้านแนวนอน	28
รูปที่ 2.26 ลักษณะของสัญญาณทางด้านแนวตั้ง	28
รูปที่ 2.27 ความสัมพันธ์ระหว่างสัญญาณทางด้านแนวนอนและแนวตั้ง	29

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูปภาพ (ต่อ)

หน้า

รูปที่ 2.28 เวลาในการสุ่มสำหรับเส้นสแกนทางแนวนอนของสัญญาณคอมโพสิตวีดีโอ 30
รูปที่ 2.29 การสแกนสองครั้งสำหรับภาพนิ่งแต่ละภาพ โดยแบ่งหนึ่งเฟรมออกเป็นสองฟิลด์ 31
รูปที่ 2.30 รูปที่เกิดจากการสแกนฟิลด์คู่ (Even line trace) 32
รูปที่ 2.31 รูปที่เกิดจากการสแกนฟิลด์คี่ (Odd line trace) 32
รูปที่ 2.32 การรวมฟิลด์ 1 และฟิลด์ 2 32
รูปที่ 3.1 บล็อกไดอะแกรมการทำงานรวมของระบบ 33
รูปที่ 3.2 บล็อกไดอะแกรมของภาคส่ง 33
รูปที่ 3.3 หน้าต่างโปรแกรมที่ใช้ควบคุมจากคอมพิวเตอร์ 34
รูปที่ 3.4 โฟล์วชาร์ตการทำงานของโปรแกรมควบคุม 35
รูปที่ 3.5 บล็อกไดอะแกรมของภาครับ 35
รูปที่ 3.6 โฟล์วชาร์ตการทำงานของภาครับ 36
รูปที่ 3.7 การเลือกโหมดการทำงาน สำหรับกำหนดค่า Configuration (Setup Mode) ของเครื่อง ET-RF24G V1.0 37
รูปที่ 3.8 บล็อกไดอะแกรมของชุดรับส่งสัญญาณภาพ 38
รูปที่ 3.9 รถสำรวจ 39
รูปที่ 3.10 ลักษณะของการ์ดแคปเจอร์ 39
รูปที่ 3.11 ชุดรับส่งสัญญาณวีดีโอไร้สาย 40
รูปที่ 3.12 กล้องวีดีโอ 40
รูปที่ 4.1 สัญญาณข้อมูลไบนารีที่ทำการวัดที่สาย RX ในขณะทำการสั่งให้รถเลี้ยวซ้าย 41
รูปที่ 4.2 สัญญาณข้อมูลไบนารีในขณะทำการสั่งให้รถสำรวจ เดินเลี้ยวซ้าย 42
รูปที่ 4.3 สัญญาณข้อมูลไบนารีที่ทำการวัดที่สาย RX ในขณะทำการสั่งให้รถสำรวจเดินหน้า 42
รูปที่ 4.4 สัญญาณข้อมูลไบนารีในขณะทำการสั่งให้รถสำรวจ เดินหน้า 43
รูปที่ 4.5 สัญญาณข้อมูลไบนารีที่ทำการวัดที่สาย RX ในขณะทำการสั่งให้รถสำรวจเลี้ยวขวา 43
รูปที่ 4.6 สัญญาณข้อมูลไบนารีในขณะทำการสั่งให้รถสำรวจ เดินเลี้ยวขวา 44
รูปที่ 4.7 สัญญาณข้อมูลไบนารีที่ทำการวัดที่สาย RX ในขณะทำการสั่งให้รถสำรวจถอยหลังทางซ้าย 44
รูปที่ 4.8 สัญญาณข้อมูลไบนารีในขณะทำการสั่งให้รถสำรวจ ถอยหลังทางซ้าย 45
รูปที่ 4.9 สัญญาณข้อมูลไบนารีที่ทำการวัดที่สาย RX ในขณะทำการสั่งให้รถสำรวจ ถอยหลัง 45
รูปที่ 4.10 สัญญาณข้อมูลไบนารีในขณะทำการสั่งให้รถสำรวจ ถอยหลัง 46
รูปที่ 4.11 สัญญาณข้อมูลไบนารีที่ทำการวัดที่สาย RX ในขณะทำการสั่งให้รถสำรวจถอยหลังทางขวา 46
รูปที่ 4.12 สัญญาณข้อมูลไบนารีในขณะทำการสั่งให้รถสำรวจ ถอยหลังทางขวา 47
รูปที่ 4.13 ภาพที่ได้แสดงบนหน้าจอคอมพิวเตอร์ของรถสำรวจที่ติดกล้องวีดีโอ 49

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญตาราง

หน้า

ตารางที่ 2.1 Baud Rate ต่างๆ และค่า Reload ของ Time1 13

ตารางที่ 2.2 ค่าเริ่มต้นของรีจิสเตอร์ SFR เมื่อ 8051 ถูกรีเซ็ต 13

ตารางที่ 2.3 การจัดขา ของคอนเน็กเตอร์ อนุกรมแบบ DB9 และหน้าที่การใช้งานต่างๆ 15

ตารางที่ 2.4 มุมของโรเตอร์เทียบกับกระแสไฟฟ้าที่จ่ายแก่เฟสต่าง ๆ 8 ตำแหน่ง 22

ตารางที่ 2.5 การจ่ายกระแสให้แก่เฟสต่าง ๆ 22

ตารางที่ 4.1 ผลการทดลองในการวัดระยะทางที่รถสำรวจสามารถวิ่งได้จากการรับสัญญาณ
จากเครื่องส่งสัญญาณ RF 48

ตารางที่ 4.2 ผลการทดลองลักษณะภาพจากกล้อง CCD ที่ระยะต่างๆในพื้นที่โล่ง 48

ตารางที่ 4.3 ผลการทดลองการติดดับของหลอดไฟ LED แทนการหมุนของสเตปปิ้งมอเตอร์ใน
ทิศทางตามเข็มนาฬิกา 50

ตารางที่ 4.4 ผลการทดลองการติดดับของหลอดไฟ LED แทนการหมุนของสเตปปิ้งมอเตอร์ใน
ทิศทางทวนเข็มนาฬิกา 50

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 1
บทนำ

1.1 แนวคิดและเหตุผลของโครงงาน

ปัจจุบันหุ่นยนต์ได้เข้ามามีบทบาทในชีวิตประจำวันของมนุษย์มากขึ้น โดยเทคโนโลยีที่มนุษย์คิดค้นและผลิตขึ้นไม่มีวันหยุดยั้งเมื่อมนุษย์ต้องการความสะดวกสบายมากขึ้นจึงเป็นที่มาของการพัฒนารูปแบบหุ่นยนต์ให้ใช้งานได้หลากหลายด้าน เช่น ใช้เป็นเครื่องทุนแรง เครื่องอำนวยความสะดวกต่างๆ หรือแม้แต่ในด้านการสำรวจ

ในการสำรวจสถานที่ที่เป็นพื้นที่อันตรายซึ่งมนุษย์ไม่สามารถเข้าไปสำรวจหรือเก็บข้อมูลได้โดยตรง จึงจำเป็นอย่างยิ่งที่จะต้องมีอุปกรณ์หรือเครื่องมือที่ทำหน้าที่แทนมนุษย์ ดังนั้นรถหุ่นยนต์สำรวจจึงเข้ามามีบทบาทในด้านวิศวกรรม และทางการทหาร เช่น ในการสำรวจเมืองแร่ สำรวจพื้นที่ที่มีวัตถุระเบิดหรือพื้นที่ที่มีสารเคมีที่เป็นพิษต่อร่างกายมนุษย์ ฯลฯ โดยมีการควบคุมจากระยะไกล ซึ่งโครงงานนี้เป็นการสร้างรถสำรวจควบคุมโดยคอมพิวเตอร์แบบไร้สาย และมีกล้องวิดีโอติดอยู่ที่ตัวรถ ซึ่งสามารถส่งข้อมูลภาพกลับมาเพื่อแสดงผลยังคอมพิวเตอร์ได้

1.2 วัตถุประสงค์ของโครงงาน

1.2.1 เพื่อใช้รถหุ่นยนต์เข้าไปสำรวจและทำการบันทึกภาพในบริเวณที่มนุษย์ไม่สามารถเข้าไปทำการสำรวจได้

1.2.2 เพื่อศึกษาถึงการทำงานของไมโครคอนโทรลเลอร์ตระกูล MCS-51

1.3 ขั้นตอนการดำเนินงาน

1.3.1 ศึกษาถึงลักษณะการทำงานไมโครคอนโทรลเลอร์ตระกูล MCS-51 ตลอดจนคำสั่งต่าง ๆ ที่ใช้ในการเขียนโปรแกรม

1.3.2 ศึกษาการใช้งานโปรแกรมที่ใช้ในการเขียนโปรแกรม และกระบวนการโปรแกรมไฟล์ข้อมูลลงไมโครคอนโทรลเลอร์ตระกูล MCS-51

1.3.3 ศึกษาการทำงานและรายละเอียดของวงจรการขับเคลื่อนโดยใช้มอเตอร์แบบ DC

1.3.4 ศึกษาการเขียนโปรแกรมโดยภาษา C# เพื่อนำมาใช้ในการควบคุมรถและกล้อง

1.3.5 ทดลองนำวงจรและโปรแกรมมาติดต่อกัน เพื่อดูว่าสามารถทำงานร่วมกันได้หรือไม่

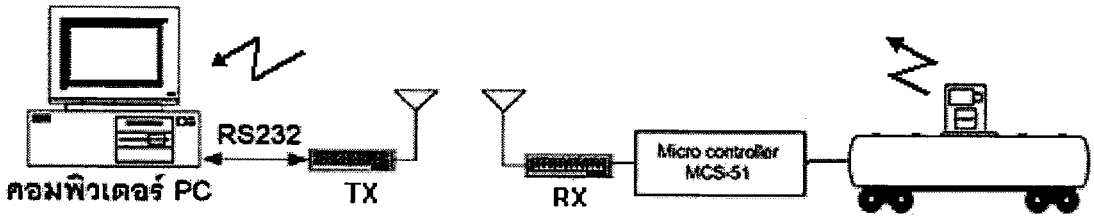
1.4 ประโยชน์ที่ได้รับ

ทำให้ได้รับความสะดวกสบายมากขึ้น โดยจากที่มนุษย์ต้องทำการเดินสำรวจเอง ก็ใช้รถหุ่นยนต์สำรวจไปแทน แล้วทำการบังคับและดูภาพได้จากจอคอมพิวเตอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1.5 ภาพรวมของโครงงาน

โดยรวมของโครงงานนี้จะสามารถแสดงได้ดังรูปที่ 1.1



รูปที่ 1.1 ภาพรวมของโครงงาน

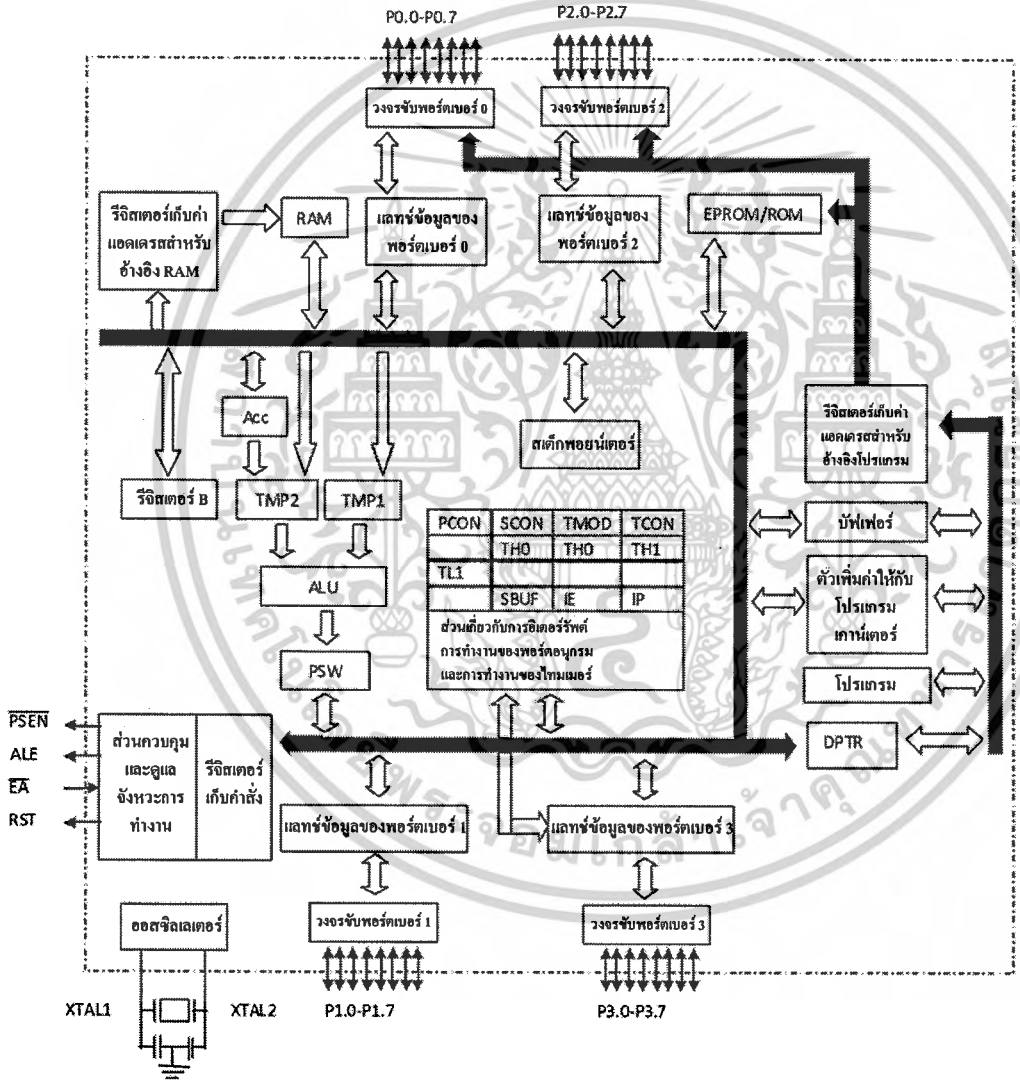
จากรูปที่ 1.1 แสดงให้เห็นถึงหลักการทำงานของรถหุ่นยนต์สำรวจซึ่งสามารถแบ่งออกได้เป็น 2 ส่วน คือ ส่วนที่ทำหน้าที่ในการควบคุมโดยใช้คอมพิวเตอร์ และส่วนของตัวรถสำรวจ โดยส่วนที่ทำหน้าที่ในการควบคุมโดยใช้คอมพิวเตอร์จะส่งสัญญาณควบคุมต่างๆ ออกไปยังรถสำรวจ ที่ตัวรถจะมีไมโครคอนโทรลเลอร์ทำหน้าที่ในการตัดสินใจว่าสัญญาณควบคุมที่ได้รับนั้นเป็นสัญญาณที่ต้องการให้รถสำรวจทำงานอย่างไร เช่น เดินหน้าหรือถอยหลัง เลี้ยวซ้ายหรือเลี้ยวขวา หรือควบคุมตำแหน่งของกล้อง และกล้องที่ติดอยู่กับตัวรถสำรวจก็จะส่งสัญญาณภาพกลับมายังส่วนที่ทำหน้าที่ในการควบคุมเพื่อแสดงผลออกทางหน้าจอคอมพิวเตอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 2
ทฤษฎีและหลักการ

2.1 ไมโครคอนโทรลเลอร์ MCS-51

ไมโครคอนโทรลเลอร์หรืออาจเรียกว่าชิงเกิลชิพไมโครคอนโทรลเลอร์ (Single Chip-Microcontroller) เป็นอุปกรณ์ที่นำเอาไมโครโพรเซสเซอร์มารวมกับหน่วยความจำและอุปกรณ์ไอโอต่างๆเอาไว้ในตัวเดียวกัน สามารถทำงานได้ทันทีเมื่อป้อนไฟเลี้ยงและสัญญาณนาฬิกาให้ตัวไมโครคอนโทรลเลอร์ จึงทำให้การออกแบบวงจรง่ายขึ้นและมีขนาดเล็กลงเป็นอย่างมาก ในที่นี้จะขอยกตัวอย่างไมโครคอนโทรลเลอร์ตระกูล MCS-51 ซึ่งมีบล็อกไดอะแกรมแสดงในรูปที่ 2.1



รูปที่ 2.1 บล็อกไดอะแกรมของไมโครคอนโทรลเลอร์ตระกูล MCS-51

การพัฒนาขีดความสามารถของไมโครคอนโทรลเลอร์ได้เป็นมาอย่างต่อเนื่อง ซึ่งมีการพัฒนาทั้งในด้านฮาร์ดแวร์(Hardware) และซอฟต์แวร์ (Software) ในการพัฒนาด้านฮาร์ดแวร์นั้นสิ่งที่เห็นได้ชัดเจนคือการพัฒนาเกี่ยวกับหน่วยความจำ ซึ่งได้เริ่มมีการพัฒนามาจาก PROM EPROM EEPROM และในปัจจุบัน ได้พัฒนามาจนถึงหน่วยความจำที่เรียกว่า Flash Memory

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ในส่วนของซอฟต์แวร์นั้นได้พัฒนาการเขียนโปรแกรมมาจากการเขียนโปรแกรมในระดับบิต (เลขฐาน2), ระดับไบต์ (เลขฐาน 16), แอสแซมบลีและภาษาในระดับสูง เช่น ภาษาเบสิกและภาษาซี เป็นต้น

2.1.1 โครงสร้างภายในของไมโครคอนโทรลเลอร์ MCS-51

ไมโครคอนโทรลเลอร์ตระกูล MCS-51 ใช้เทคโนโลยีในการผลิตแบบ NMOS และ CMOS ซึ่งภายในได้รวมวงจรต่างๆไว้อย่างครบถ้วนพร้อมที่จะทำงานได้เมื่อจ่ายไฟเลี้ยงและสัญญาณนาฬิกา ไมโครคอนโทรลเลอร์ตระกูล MSC-51 ได้ถูกผลิตออกมามากมายหลายเบอร์โดยบริษัทต่างๆ เช่น บริษัท Atmel, Philips, Dallas และบริษัทอื่นๆซึ่งไม่ว่าจะเป็นเบอร์อะไรก็ตาม ถ้าเป็นไมโครคอนโทรลเลอร์ตระกูล MSC-51 และจะมีโครงสร้างต่างๆที่คล้ายกันจะแตกต่างกันออกไปในส่วนของความสามารพิเศษของแต่ละเบอร์ ยกตัวอย่างเช่น เบอร์ AT89C51 มีไทเมอร์ 2 ตัว ในขณะที่เบอร์ AT89C52 มีไทเมอร์ 3 ตัว เป็นต้น

โครงสร้างโดยทั่วไป ของไมโครคอนโทรลเลอร์ นั้น สามารถแบ่งออกมาได้เป็น 5 ส่วนใหญ่ๆ ดังต่อไปนี้

1. หน่วยประมวลผลกลางหรือซีพียู (CPU : Central Processing Unit)
2. หน่วยความจำ (Memory) สามารถแบ่งออกเป็น 2 ส่วน คือ หน่วยความจำที่มีไว้สำหรับเก็บโปรแกรมหลัก (Program Memory) เปรียบเสมือนฮาร์ดดิสก์ของเครื่องคอมพิวเตอร์ตั้งโต๊ะ คือข้อมูลใดๆที่ถูกเก็บไว้ในนี้จะไม่สูญหายปแม้ไม่มีไฟเลี้ยง อีกส่วนหนึ่งคือหน่วยความจำข้อมูล (Data Memory) ใช้เป็นเหมือนกระดาษทดในการคำนวณของซีพียู และเป็นที่พักข้อมูลชั่วคราวขณะทำงาน แต่หากไม่มีไฟเลี้ยง ข้อมูลก็จะหายไปคล้ายกับหน่วยความจำแรม (RAM) ในเครื่องคอมพิวเตอร์ทั่วๆ ไป

แต่สำหรับไมโครคอนโทรลเลอร์สมัยใหม่ หน่วยความจำข้อมูลจะมีทั้งที่เป็นหน่วยความจำแรม ซึ่งข้อมูลจะหายไปเมื่อไม่มีไฟเลี้ยง และเป็นอีอีพรอม (EEPROM : Erasable Electrically Read-Only Memory) ซึ่งสามารถเก็บข้อมูลได้แม้ไม่มีไฟเลี้ยง

3. ส่วนติดต่อกับอุปกรณ์ภายนอก หรือพอร์ต (Port) มี 2 ลักษณะคือ พอร์ตอินพุต (Input Port) และพอร์ตส่งสัญญาณหรือพอร์ตเอาต์พุต (Output Port) ส่วนนี้จะใช้ในการเชื่อมต่อกับอุปกรณ์ภายนอก ถือว่าเป็นส่วนที่สำคัญมาก ใช้ร่วมกันระหว่างพอร์ตอินพุต เพื่อรับสัญญาณ อาจจะด้วยการกดสวิตช์ เพื่อนำไปประมวลผลและส่งไปพอร์ตเอาต์พุต เพื่อแสดงผลเช่น การติดสว่างของหลอดไฟ เป็นต้น
4. ช่องทางเดินของสัญญาณ หรือบัส (BUS) คือเส้นทางการแลกเปลี่ยนสัญญาณข้อมูลระหว่างซีพียู หน่วยความจำและพอร์ต เป็นลักษณะของสายสัญญาณ จำนวนมากอยู่ภายในตัวไมโครคอนโทรลเลอร์ โดยแบ่งเป็นบัสข้อมูล (Data Bus) , บัสแอดเดรส (Address Bus) และบัสควบคุม (Control Bus)

บัสข้อมูลเป็นสายสัญญาณที่บรรจุข้อมูล เพื่อการประมวลผลทั้งหมด ขนาดของบัสจะขึ้นอยู่กับความสามารถการประมวลผลของซีพียู สำหรับในงานทั่วๆ ไป ขนาดของบัสข้อมูลจะเป็น 8 บิต และในปัจจุบันได้มีการพัฒนาขึ้นมาจนถึง 16, 32 และ 64 บิต

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บัสแอดเดรสเป็นสายสัญญาณที่บรรจุค่าตำแหน่งของหน่วยความจำ โดยการติดต่อกับหน่วยความจำนั้น ซีพียู ต้องกำหนดตำแหน่งที่ต้องการอ่านหรือเขียนก่อน ดังนั้นจำนวนสายสัญญาณของแอดเดรสจึงต้องมีจำนวนมาก ยิ่งมากเท่าไหร่ ก็จะเป็นการแสดง ขนาดของหน่วยความจำที่ไมโครคอนโทรลเลอร์ สามารถติดต่อได้ โดยสามารถคำนวนได้จาก

จำนวนแอดเดรสของหน่วยความจำ = 2 ยกกำลัง n (n คือจำนวนของเส้นทาง)

ยกตัวอย่าง ไมโครคอนโทรลเลอร์ ตัวหนึ่งมีสายแอดเดรส 10 เส้น ดังนั้นไมโครคอนโทรลเลอร์ตัวนี้ สามารถติดต่อกับหน่วยความจำได้ 2 ยกกำลัง 10 = 1,024 ตำแหน่ง

หากต้องการทราบความจุของหน่วยความจำจริงๆ จะต้องทราบถึงขนาดของบัสข้อมูลก่อนว่าเป็นเท่าใด หากเป็น 8 บิต ความจุของหน่วยความจำที่มีสายแอดเดรส 10 เส้น จะเท่ากับ 8x1024 = 8,192 บิต และ 1 กิโลไบต์ เท่ากับ 1,024 ไบต์ ดังนั้นไมโครคอนโทรลเลอร์ดังกล่าว จึงมีความจุของหน่วยความจำเท่ากับ 8,192 บิต หรือ 1,024 ไบต์ หรือ 1 กิโลไบต์

บัสควบคุมเป็นกลุ่มของสายสัญญาณควบคุมการติดต่อทั้งหมดของซีพียูกับหน่วยความจำและพอร์ต สำหรับสายสัญญาณเลือกควบคุมหลักได้แก่ สายสัญญาณเลือก-อ่าน-เขียนหน่วยความจำ สายสัญญาณเลือกเลือก อ่าน-เขียน ข้อมูล กับพอร์ต

5. วงจรกำเนิดสัญญาณนาฬิกา นับเป็นส่วนประกอบที่สำคัญมากอีกส่วนหนึ่ง เนื่องจากการทำงานที่เกิดขึ้นในตัวไมโครคอนโทรลเลอร์ จะขึ้นอยู่กับการกำหนดจังหวะ หากสัญญาณนาฬิกามีความถี่สูง จังหวะการทำงานก็จะสามารถทำได้ถี่ขึ้นส่งผลให้ไมโครคอนโทรลเลอร์ ดังนั้น มีความเร็วในการประมวลผลสูงตามไปด้วย

2.1.2 คุณสมบัติทั่วไปของไมโครคอนโทรลเลอร์ MCS-51

คุณสมบัติของ MCS-51 คุณสมบัติที่สำคัญๆของชิปไมรครคอนโทลเลอร์ตระกูล MCS-51 มีดังนี้

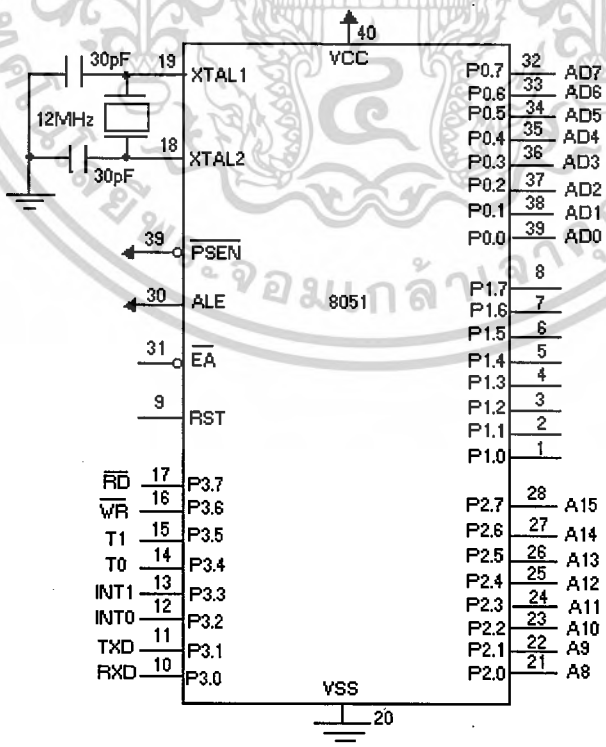
- ต้องการแหล่งจ่ายไฟ 5 โวลต์เพียงชุดเดียว
- มีหน่วยความจำสำหรับเก็บโปรแกรมควบคุมการทำงานอยู่ภายในชิป จำนวน 4 กิโลไบต์ (เบอร์ 8031,8032 ไม่มีหน่วยความจำส่วนนี้ ส่วนเบอร์ 8052 มีหน่วยความจำส่วนนี้ 8 กิโลไบต์และสำหรับเบอร์ 83C51FB จะมีหน่วยความจำส่วนนี้รวมทั้งสิ้น 16 กิโลไบต์)
- มีหน่วยความจำสำหรับโปรแกรมและข้อมูล (RAM) อยู่ภายในชิป จำนวน 128 ไบต์ (ในเบอร์ 8031,8051) หรือ 256ไบต์ (ในเบอร์ 8031,8052)
- สามารถใช้หน่วยความจำสำหรับโปรแกรมและข้อมูลที่อยู่ภายนอกชิปได้ อย่างละ 64 กิโลไบต์แยกจากกัน
- คำสั่งส่วนใหญ่ใช้การทำงานเพียง 1 ไมโครวินาที เมื่อใช่คริสตอลความถี่ 12 แมกกะเฮิรตซ์
- มีพอร์ตที่สามารถรับและส่ง ข้อมูลได้ทั้งสองทิศทางจำนวน 4 พอร์ตๆละ 8 บิตหรือสามารถใช้งานเป็นพอร์ตขนาด 1 บิตแยกจากกัน ทำให้เสมือนมีพอร์ตขนาด 1 บิต ใช้งานรวมกันทั้งสิ้น 32 พอร์ต

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- รับข้อมูลและส่งข้อมูลได้ในตัว โดยสามารถกำหนดอัตราเร็วในการรับและส่งข้อมูล (Baud Rate) ได้ตั้งแต่ 300 ถึง 375 กิโลบิตต่อวินาที
- จัดลำดับความสำคัญของสัญญาณอินเตอร์รัปต์ได้ สองระดับ
- มีรีจีสเตอร์สำหรับใช้งานเป็นไทม์เมอร์หรือเคาน์เตอร์ เพื่อนับสัญญาณนาฬิกาภายในชิป หรือนับเปลี่ยนสถานะของสัญญาณภายนอกขนาด 16 บิตจำนวน 2 ตัว เพื่อใช้งานสำหรับนับจำนวนพัลส์วัดความกว้างของพัลส์หรือใช่วัดช่วงเวลา (ในเบอร์ 8052 จะมี 3 ตัว)
- หน่วยความจำสำหรับเก็บข้อมูลภายในบางส่วน สามารถเข้าถึงข้อมูลได้ทั้งระดับบิตเพื่อให้การออกแบบโปรแกรมและการควบคุมระบบทำได้ง่ายขึ้น
- มีคำสั่งคูณและหารขนาด 8 บิตในตัวเอง
- สามารถประมวลผลแบบบูลีนเพื่อใช้งานควบคุมโดยเฉพาะ
- ใช้โปรแกรมของไมโครคอนโทลเลอร์ตะกูล MCS-48 (upwardly compatible) ได้ ไมโครคอนโทลเลอร์ตะกูล MCS-51 เบอร์ที่จัดว่าเป็นเบอร์พื้นฐานในตระกูลนี้ คือเบอร์ 8051,8751และ 8031 ซึ่งมีจำนวนขาภายนอก 40 ขาเหมือนกันใช้เวลาในการปฏิบัติคำสั่งแต่ละคำสั่งเท่ากันใช้แรงดันไฟเท่ากันสิ่งที่แตกต่างกันระหว่างสามเบอร์นี้คือขนาดของหน่วยความจำสำหรับเก็บในชิปมีไม่เท่ากัน

2.1.3 การจัดการของไมโครคอนโทรลเลอร์ตระกูล MCS-51

ไอซีไมโครคอนโทรลเลอร์ 8051 โครงสร้าง ไอซี เป็นแบบ DIP มีขาทั้งหมด 40 ขา โดยขาต่างๆ เป็นขาพอร์ตอินพุท, ขาพอร์ตเอาท์พุท, ขาสัญญาณควบคุม, ขาตำแหน่งหน่วยความจำข้อมูล ดังรูปที่ 2.2



รูปที่ 2.2 ขาต่างๆของ 8051

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

พอร์ต 0 (Port 0) พอร์ต 0 ได้แก่ ขาที่ 32 - 39 ของ MCS - 51 สามารถใช้เป็นอินพุตเอาต์พุตได้ นอกจากนี้ในการติดต่อกับหน่วยความจำภายนอกยังใช้เป็นขาบัสตำแหน่ง และบัสข้อมูล อีกด้วย

พอร์ต 1 (Port 1) พอร์ต 1 ได้แก่ ขาที่ 1 - 8 เป็นพอร์ต 8 บิต สามารถอ้างทีละบิตได้ คือ P1.0 - P1.7

พอร์ต 2 (Port 2) พอร์ต 2 ได้แก่ ขาที่ 21 - 28 จะใช้งาน 2 หน้าที่คือ ใช้เป็นพอร์ต 8 บิตกับใช้เป็นขาแอดเดรส 8 บิตในการอ้างหน่วยความจำภายนอก

พอร์ต 3 (Port 3) พอร์ต 3 ได้แก่ ขาที่ 10 - 17 จะใช้งานสองหน้าที่คือ เป็นพอร์ตอินพุตและเอาต์พุต และใช้เป็นขาควบคุมต่าง ๆ

PSEN (Program Store Enable) ขา PSEN เป็นขาที่ส่งสัญญาณออกคือขา 29 ขานี้จะแอคทีฟเมื่อ MCS-51 ต้องการอ่าน โปรแกรมจากหน่วยความจำภายนอก โดยปกติถ้าหน่วยความจำภายนอกเป็น อีพรอม ขา PSEN จะต่อกับขาสัญญาณเปิดทางด้านเอาต์พุต (Output Enable: OE) ของ อีพรอม

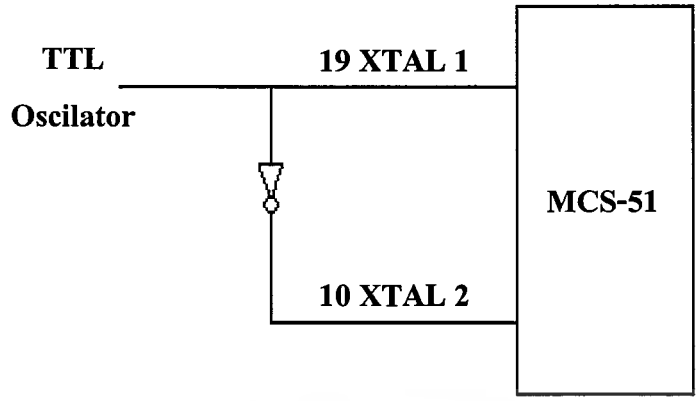
ALE (Address Latch Enable) เนื่องจากพอร์ต 0 สามารถใช้เป็นขาอ้างตำแหน่ง และขาข้อมูล MCS-51 จะมีขา ALE ได้แก่ขา 30 ขานี้จะใช้ มัลติเพล็กซ์ (Multiplex) สัญญาณ บัสตำแหน่ง ของ พอร์ต 0 ในการใช้งานระบบ MCS-51 นั้น จะต้องมีอุปกรณ์มาต่อกับ พอร์ต 0 ที่ทำหน้าที่ คงค่า (Latch) สัญญาณบัสตำแหน่ง เมื่อ MCS-51 ต้องการติดต่อกับหน่วยความจำภายนอก MCS-51 จะส่งสัญญาณบัสตำแหน่งออกมาก่อนทาง พอร์ต 0 จากนั้นจะส่งสัญญาณ ALE มาทำหน้าที่คงค่า (Latch) อุปกรณ์ภายนอก ให้เก็บค่าบัสตำแหน่งของ พอร์ต 0 ไว้เพื่อใช้ พอร์ต 0 เป็นบัสข้อมูล ต่อไป

EA (External Access) ขา EA ได้แก่ขาที่ 31 ถ้าขานี้เป็นลอจิก "1" จะใช้กับเบอร์ 8051/8052 เพื่อบอกว่าให้อ่านโปรแกรมจากหน่วยความจำภายใน แต่ถ้าเป็นลอจิก "0" จะให้ MCS - 51 ทำโปรแกรมโดยอ่านจากหน่วยความจำโปรแกรมภายนอก (ถ้าขา EA เป็น "0" ขา PSEN จะแอคทีฟ) ถ้าหากเป็นเบอร์ 8031 หรือ 8032 ขา EA จะเป็น "0" เสมอ เพราะว่าไม่มีโปรแกรมหน่วยความจำภายใน แต่ถ้าใช้เบอร์ 8051/8052 ซึ่งมีหน่วยความจำภายในและให้ขา EA เป็น "0" ซึ่งจะหยุดการทำงานของรอม ภายในและอ่านโปรแกรมจาก อีพรอม ภายนอกแทน

RST (Reset) ขา RST ได้แก่ขา 9 จะใช้ในการรีเซ็ต MCS-51 โดยจะให้ขานี้เป็นลอจิก "1" อย่างน้อย 2 คาบเวลา จึงจะรีเซ็ตระบบได้

ความถี่สัญญาณนาฬิกาบนชิพ (ON-chip Oscillator Inputs) เป็นวงจรออสซิลเลเตอร์บนชิพ ได้แก่ ขา 18 - 19 โดยต่อคริสตอลเข้ากับขานี้ โดยปกติมักจะใช้คริสตอลความถี่ 12 เมกกะเฮิรตซ์ กับตัวเก็บประจุหรืออาจใช้สัญญาณนาฬิกาจากแหล่งกำเนิดสัญญาณภายนอก ต่อกับขา 18 และขา 19 ดังรูปที่ 2.3

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.3 การต่อแหล่งกำเนิดสัญญาณภายนอกให้กับ 8051

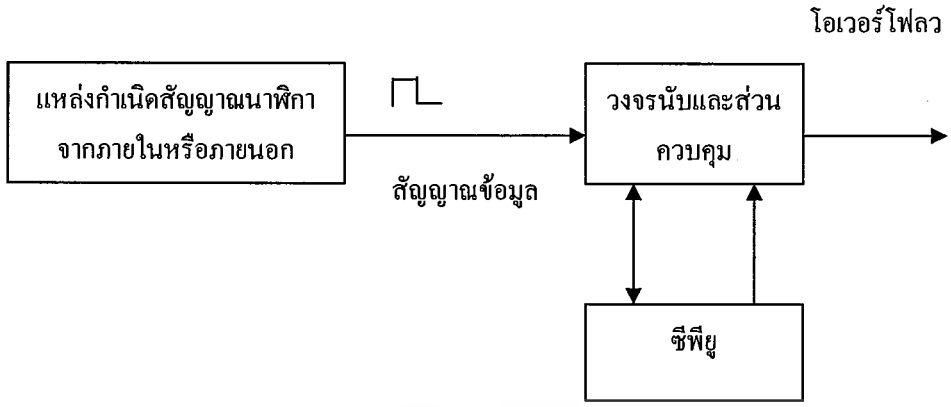
การต่อกับแหล่งจ่ายไฟ ใน MCS - 51 จะใช้แหล่งจ่ายไฟ 5 โวลต์ ต่อเข้ากับขา 40 (Vcc) ส่วนขา 20 (Vss) จะต่อลงกราวนด์ (Ground)

2.2 ไทเมอร์ / เคาน์เตอร์ (Timer/Counter)

ไทเมอร์/เคาร์เตอร์ เป็นวงจรภายในที่ช่วยเพิ่มความสามารถให้ไมโครคอนโทรเลอร์ เนื่องจากไทเมอร์/เคาน์เตอร์จะมีลักษณะการทำงานเป็นตัวนับเวลาหรือนับสัญญาณนาฬิกา ซึ่งสามารถทำงานเกี่ยวกับการสร้างฐานเวลา, การสร้างสัญญาณพัลซ์, การเปรียบเทียบค่าของเวลาหรือค่าการนับสัญญาณพัลซ์ที่รับเข้ามาจากภายนอกและรวมไปถึงการควบคุมอัตราการรับส่งข้อมูลของพอร์ตอนุกรมด้วย ไท-เมอร์/เคาน์เตอร์เป็นวงจรที่สามารถทำงานได้เองโดยเป็นอิสระจากซีพียู แต่ซีพียูสามารถควบคุมการทำงาน, การอ่านและเขียนข้อมูลต่าง ๆ ได้

ไมโครคอนโทรลเลอร์แต่ละเบอร์อาจจะมีไทเมอร์ / เคาน์เตอร์จำนวนต่างกัน หรือมีความสามารถที่แตกต่างกัน เช่น เบอร์AT89C51 มีไทเมอร์ / เคาน์เตอร์จำนวน 2 ตัว(T0 และ T1) ในขณะที่เบอร์ AT89C52 หรือเบอร์ AT89S8252 มีไทเมอร์ / เคาน์เตอร์จำนวน 3 ตัว(T1 และ T2)เป็นต้น ไทเมอร์ / เคา-เตอร์แต่ละตัวอาจจะมีความสามารถที่แตกต่างกันในรายละเอียดทางฮาร์ดแวร์และการทำงาน แต่อย่างไรก็ตามจะมีหลักการทำงานที่เหมือนกันคือ ถูกควบคุมจากซีพียูและใช้สัญญาณนาฬิกามาเป็นสัญญาณอินพุตในกรณีใช้งานเป็นไทเมอร์หรือรับสัญญาณอินพุตจากภายนอกในกรณีใช้งานเป็นเคาน์เตอร์นอกจากนี้แล้วไทเมอร์/เคาน์เตอร์ทุกตัวยังมีการทำงานที่เป็นอิสระต่อกันอีกด้วย ในที่นี้จะขอกล่าวเพียงสองตัวเท่านั้น คือไทเมอร์/เคาน์เตอร์ 0 และไทเมอร์/เคาน์เตอร์ 1

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.4 โครงสร้างโดยรวมของไทเมอร์ / เคาน์เตอร์

2.2.1 การทำงานในโหมดไทเมอร์ (Timer Mode) ของไทเมอร์/เคาน์เตอร์ 0 และ 1

เมื่อกำหนดให้มีการทำงานเป็นโหมดไทเมอร์หรือตัวตั้งเวลา วงจรนับหรือเคาน์เตอร์จะทำการนับขึ้นโดยใช้สัญญาณนาฬิกาภายในซึ่งสัญญาณนาฬิกาตัวนี้จะเกิดขึ้นทุกๆ 1 แมชชินไซเคิล (Machine cycle) นั่นหมายความว่าจะเกิดการนับขึ้น 1 ค่าในแต่ละแมชชีนไซเคิลนั่นเอง โดนในไมโครคอนโทรลเลอร์โดยทั่วๆไป 1 แมชชีนไซเคิลจะใช้เวลา 12 คาบของสัญญาณนาฬิกาที่ป้อนให้กับตัวไมโครคอนโทรลเลอร์ นั่นหมายความว่าจะเกิดการนับขึ้นทุกๆ 1/12 ของความถี่สัญญาณนาฬิกา (บางเบอร์อาจใช้แค่ 4 คาบ เช่น เบอร์ DS80C320 หรือ ใช้ 6 คาบ เช่น เบอร์ P89V51RD2 เป็นต้น)

2.2.2 การทำงานในโหมดเคาน์เตอร์ (Counter Mode) ของไทเมอร์/เคาน์เตอร์ 0 และ 1

เมื่อกำหนดให้มีการทำงานในโหมดเคาน์เตอร์หรือตัวนับ วงจรนับหรือเคาน์เตอร์ภายในจะทำการนับขึ้นโดยใช้สัญญาณขอบขาลง (เปลี่ยน 1 เป็น 0) จากภายนอกที่เข้ามาทางขา T0(P3.4),T1(P3.5) กระบวนการอ่านสัญญาณอินพุตเหล่านี้จะใช้เวลาทั้งสิ้น 2 แมชชีนไซเคิล นั่นหมายความว่าอัตราการนับจะมีค่าเท่ากับ 1/24 (กรณี 1 แมชชินไซเคิลใช้เวลา 12 คาบ) ของความถี่สัญญาณนาฬิกา ดังนั้นสามารถใช้การทำงานในโหมดเคาน์เตอร์เพื่อนับสัญญาณนาฬิกาได้สูงสุดเท่ากับ ความถี่สัญญาณนาฬิกา/24 (กรณี 1 แมชชินไซเคิลใช้เวลา 12 คาบ) เช่นใช้สัญญาณนาฬิกา 12 MHz จะสามารถรับสัญญาณอินพุตที่มีความถี่สูงได้ 12 MHz/24 = 500 kHz (กรณี 1 แมชชินไซเคิลใช้เวลา 12 คาบ)

2.2.3 รีจิสเตอร์ของไทเมอร์/ เคาน์เตอร์ 0 และ 1 (Timer / Counter0,1 Register)

การกำหนดให้ไทเมอร์ / เคาน์เตอร์ ทำงานตามความต้องการนั้นจะต้องมีการกำหนดค่าต่างๆ ให้กับรีจิสเตอร์ที่เกี่ยวข้องกับการทำงานอย่างเหมาะสม ซึ่งรีจิสเตอร์เหล่านี้เป็นรีจิสเตอร์ฟังก์ชันพิเศษ (SFR) ซึ่งมีรายละเอียดดังนี้

TL0 เป็นรีจิสเตอร์ไบต์ต่ำของไทเมอร์ 0 มีแอดเดรสอยู่ที่ 8AH

TH0 เป็นรีจิสเตอร์ไบต์สูงของไทเมอร์ 0 มีแอดเดรสอยู่ที่ 8BH

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

TL1 เป็นรีจิสเตอร์ไบต์ต่ำของไทเมอร์ 1 มีแอดเดรสอยู่ที่ 8CH

TH1 เป็นรีจิสเตอร์ไบต์สูงของไทเมอร์ 1 มีแอดเดรสอยู่ที่ 8DH

รีจิสเตอร์ของไทเมอร์ทั้ง 4 ตัวนี้เป็นรีจิสเตอร์ขนาด 8 บิต ดังนั้นเมื่อนำ TL0 มารวมกับ TH0 จะทำให้ได้รีจิสเตอร์ของไทเมอร์ 0 ขนาด 16 บิต ในทำนองเดียวกันเมื่อนำ TL1 มารวมกับ TH1 จะทำให้ได้รีจิสเตอร์ของไทเมอร์ 1 ขนาด 16 บิต ซึ่งสามารถเก็บค่าได้สูงสุดถึง 65,536 ค่า (0-65,535 หรือ 0000H ถึง FFFFH)

2.2.4 รีจิสเตอร์ควบคุมการทำงานของไทเมอร์/เคาน์เตอร์ 0 และ1

TCON เป็นรีจิสเตอร์ที่ทำงานของไทเมอร์/เคาน์เตอร์ 0 และ1 มีขนาด 8 บิตและมีแอดเดรสอยู่ที่ 88H และสามารถเข้าถึงในระดับบิตได้ ในแต่ละบิตมีหน้าที่แตกต่างกันออกไปดังนี้

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	
TF1	TR1	TF0	TR0	IE1	IT1	IE0	IT0

รูปที่ 2.5 บิตต่างๆในรีจีสเตอร์ TCON

***TF 1** (Timer 1 overflow flag) จะเกิดการเซตด้วยขบวนการฮาร์ดแวร์ คือเมื่อการนับเพิ่มขึ้นจนเกิดการ โอเวอร์โฟลว (เปลี่ยนจาก FFH ไปเป็น 00H ในกรณีที่เป็น 8บิต และเปลี่ยนจาก FFFFH ไปเป็น 0000H กรณีที่เป็น 16บิต) บิตนี้จะถูกเคลียร์โดยกระบวนการทางฮาร์ดแวร์เช่นกันโดยจะเคลียร์เมื่อซีพียูกระโดดไปทำงานในโปรแกรมบริการอินเตอร์รัปต์ของไทเมอร์ 1 โอเวอร์โฟลว์

***TR 1** (Timer 1 run control bit) ใช้ควบคุมการเปิด/ปิด (Enable/Disable) การทำงานของไทเมอร์ 1การเซตและการเคลียร์สามารถทำได้โดยกระบวนการทางซอฟแวร์

"0" ปิดการทำงานของไทเมอร์ 0 (disable)

"1" เปิดการทำงานของไทเมอร์ 0 (enable)

***TF 0** (Timer 0 overflow flag) จะเกิดการเซตด้วยกระบวนการทางฮาร์ดแวร์ คือเมื่อการนับเพิ่มขึ้นจนถึงโอเวอร์โฟลว (เปลี่ยนจาก FFH ไปเป็น 00H ในกรณีที่เป็นแบบ 8 บิต และเปลี่ยนจาก FFFFH ไปเป็น 0000H กรณีที่เป็น 16บิต) บิตนี้จะถูกเคลียร์โดยกระบวนการทางฮาร์ดแวร์เช่นกันโดยจะเคลียร์เมื่อ ซีพียูกระโดดไปทำงานในโปรแกรมบริการการอินเตอร์รัปต์ของไทเมอร์ 0โอเวอร์โฟลว

***TR 0** (Timer 0 run control bit) ใช้ควบคุมการเปิด/ปิด (Enable/Disable) การทำงานของไทเมอร์ 1การเซตและการเคลียร์สามารถทำได้โดยกระบวนการทางซอฟแวร์

"0" ปิดการทำงานของไทเมอร์ 0 (disable)

"1" เปิดการทำงานของไทเมอร์ 0 (enable)

***IE 1** (External Interrupt 1 edge flag) ใช้ในกระบวนการอินเตอร์รัปต์ คือจะเกิดการเซตโดยกระบวนการทางฮาร์ดแวร์ นั่นคือเมื่อตรวจพบว่ามีสัญญานอินเตอร์รัปต์เข้ามาที่ขาอินพุตของอินเตอร์รัปต์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1 (INT: P3.3) บิตนี้จะเซต และจะเคลียร์เมื่อ ซีพียูจะกระโดดไปทำงานในโปรแกรมบริการอินเตอร์รัปต์ของ external interrupt 1

***IT1** (interrupt 1 type control bit) ใช้ในกระบวนการอินเตอร์รัปต์ของเอ็กเทอร์นอลอินเตอร์รัปต์ 1 โดยเป็นตัวกำหนดว่าจะให้อินเตอร์รัปต์เกิดขึ้นเมื่อตรวจพบสัญญาณที่เข้ามาทางขาอินพุตของอินเตอร์รัปต์ 1 (INT: P3.3) เป็นการตรวจสอบสัญญานขอบขาลงหรือระดับลอจิก "0" การเซตและเคลียร์บิตนี้สามารถทำได้โดยกระบวนการทางซอฟแวร์

***IE 0** (External Interrupt 0 edge flag) ใช้ในกระบวนการอินเตอร์รัปต์ คือจะเกิดการเซตโดยกระบวนการทางฮาร์ดแวร์ นั่นคือเมื่อตรวจพบว่ามีสัญญานอินเตอร์รัปต์เข้ามาที่ขาอินพุตของอินเตอร์รัปต์ (INT1: P3.2) บิตนี้จะเซต และจะเคลียร์เมื่อ ซีพียูจะกระโดดไปทำงานในโปรแกรมบริการอินเตอร์รัปต์ของ external interrupt 0

***IT0** (interrupt 0 type control bit) ใช้ในกระบวนการอินเตอร์รัปต์ของเอกเทอร์นอลอินเตอร์รัปต์ 0โดยเป็นตัวกำหนดว่าจะให้อินเตอร์รัปต์เกิดขึ้นเมื่อตรวจพบสัญญานที่เข้ามาทางขาอินพุตของอินเตอร์รัปต์ 0 (INT1: P3.2 เป็นการตรวจสอบสัญญานขอบขาลงหรือระดับลอจิก "0" การเซตและเคลียร์บิตนี้สามารถทำได้โดยกระบวนการทางซอฟแวร์

"0" จะเกิดการอินเตอร์รัปต์เมื่อตรวจพบสัญญานขอบขาลง (falling edge)

"1" จะเกิดการอินเตอร์รัปต์เมื่อตรวจพบสัญญาณเป็นลอจิก "0" (low Level triggered)

2.2.5 รีจิสเตอร์เลือกโหมดการทำงานของไทเมอร์/เคาน์เตอร์ 0 และ 1

TMOD รีจิสเตอร์ที่ทำงานของไทเมอร์/เคาน์เตอร์ 0 และ1 มีขนาด 8 บิตและมีแอดเดรสอยู่ที่ 89H ไม่สามารถเข้าถึงในระดับบิตได้ รีจิสเตอร์ตัวนี้จะแยกออกเป็นสองส่วน คือ 4 บิตบนจะใช้เลือกโหมดของไทเมอร์/เคาน์เตอร์1 และ 4 บิตล่างจะใช้เลือกโหมดของไทเมอร์/เคาน์เตอร์ 0 โดยในแต่ละบิตจะมีหน้าที่แตกต่างกันออกไปดังนี้

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	
Gate	C/T'	M1	M0	GATE	C/T'	M1	M0

รูปที่ 2.6 บิตต่างๆในรีจีสเตอร์ TMOD

***GATE** ทำหน้าที่เป็นตัวเลือกรูปแบบการทำงานของไทเมอร์/เคาน์เตอร์ ว่าจะให้มีการควบคุมการทำงานแบบซอฟแวร์หรือฮาร์ดแวร์ดังนี้

"0" เลือกรูปแบบการควบคุมการทำงานเป็นแบบซอฟแวร์ คือ ไทเมอร์/เคาน์เตอร์ x จะทำงานเมื่อ TRx ในรีจิสเตอร์ TCON เป็น "1"

"1" เลือกรูปแบบการควบคุมการทำงานเป็นแบบฮาร์ดแวร์ คือ ไทเมอร์/เคาน์เตอร์ x จะทำงานเมื่อ TRx ในรีจิสเตอร์ TCON เป็น "1" และที่ขาอินพุตของอินเตอร์รัปต์ INTx เป็นสถานะลอจิก "1"

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

***M1, M0** (Mode Selector Bit) เป็นบิตที่ใช้เลือกโหมดการทำงานของไทเตอร์ซึ่งมีดังนี้ "00" เลือกโหมดการทำงานของไทเมอร์/เคาน์เตอร์เป็นแบบ 13 บิต

"01" เลือกโหมดการทำงานของไทเมอร์/เคาน์เตอร์เป็นแบบ 16 บิต

"10" เลือกโหมดการทำงานของไทเมอร์/เคาน์เตอร์เป็นแบบ 8 บิต รีโหลดค่าอัตโนมัติ

"11"เลือกโหมดการทำงานของไทเมอร์/เคาน์เตอร์0 (ไทเมอร์/เคาน์เตอร์ 1ไม่ทำงาน) ให้ทำงานแบบแยกส่วน โดยแยกออกเป็นไทเมอร์/เคาน์เตอร์สองตัว โดยตัวแรกจะใช้รีจิสเตอร์ TL0 จะถูกควบคุมการทำงานโดย TR0 ส่วนตัวที่สองจะใช้รีจิสเตอร์ TH0 จะถูกควบคุมการทำงานโดย TR1 (TR0 และTR1 อยู่ในรีจิสเตอร์ TCON)

2.3 การคำนวณความเร็วการรับและส่งข้อมูลแบบอนุกรม (Generating Baud Rate)

การกำหนดความเร็วการรับส่งข้อมูลแบบอนุกรมสามารถแบ่งออกตาม Mode การทำงานดังนี้

Mode 0

ความเร็วในการรับส่งข้อมูลแบบอนุกรมใน Mode นี้จะกำหนดอัตราการรับส่งตายตัวเท่ากับ 1/12 ของความถี่ของชุดกำเนิดความอ้างอิงของ 8051 และจะไม่ใช้ Timer/Counter ดังนั้นเพียงกำหนดที่รีจิสเตอร์ SCON ก็เพียงพอ จะได้

Baud Rate = Osc.Freq/12

Mode 1

ในการกำหนดความเร็วการรับส่งข้อมูลแบบอนุกรมใน Mode 1 นี้จะใช้ Timer 1 เป็นฐานเวลาของการทำงานของ Timer 1 ใน Mode 2 (Auto-Reload) โดยสามารถคำนวณได้ดังนี้

Baud Rate = [K*Osc.Freq] / [32*12*(256-(TH1))]

K = 1 เมื่อ SMOD ในรีจิสเตอร์ PCON = 0

K = 2 เมื่อ SMOD ในรีจิสเตอร์ PCON = 1

ส่วนมากแล้ว ผู้ใช้จะทราบค่าของ Baud Rate ที่จะส่งได้ค่านั้น จะได้ค่าของ Time 1 สำหรับ Reload ได้

TH1 = 256 – [(k*Osc.Freq) / (384 * Baud Rate)]

จากตารางต่อไปนี้ แสดง Baud Rate ต่างๆ และค่า Reload ของ Time 1

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 2.1 Baud Rate ต่างๆ และค่า Reload ของ Time1

Baud Rate	Freq.	SMOD	Timer1		
			C/T	Mode Value	Reload
Mode 0 Max : 1MHz	12 MHz	X	X	X	X
Mode 2 Max : 375 K	12 MHz	1	X	X	X
Modes 1.3 : 62.5 K	12 MHz	1	0	2	FFH
19.2 K	11.059 MHz	1	0	2	FDH
9.6 K	11.059 MHz	0	0	2	FDH
4.8 K	11.059 MHz	0	0	2	FAH
2.4 K	11.059 MHz	0	0	2	F4H
1.2 K	11.059 MHz	0	0	2	E8H
137.5	11.986 MHz	0	0	2	1DH
110	6 MHz	0	0	2	72H
110	12 MHz	0	0	1	FEEBH

Mode 2

ความเร็วการรับส่งใน Mode นี้จะเป็นค่าคงที่ซึ่งมี 2 ค่า ขึ้นกับค่า SMOD ในรีจิสเตอร์ PCON ดังนี้

เมื่อ SMOD = 1 Baud Rate = 1/32 Osc. Freq

เมื่อ SMOD = 0 Baud Rate = 1/64 Osc. Freq

Mode 3

การกำหนดความเร็วการรับส่งใน Mode 3 จะคิดเช่นเดียวกับการคิดใน Mode 1 สำหรับค่าเริ่มต้นของรีจิสเตอร์ SFR เมื่อ 8051 ถูกรีเซ็ต จะมีค่าดังแสดงในตารางต่อไปนี้

ตารางที่ 2.2 ค่าเริ่มต้นของรีจิสเตอร์ SFR เมื่อ 8051 ถูกรีเซ็ต

SFR Name	Reset Value
PC	0000H
ACC	00H
B	00H
PSW	00H
SP	07H

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

DPTR	0000H
P0 – P3	FFH
IP (8051)	XXX0000B
IP (8052)	XX000000B
IE(8051)	0XX00000B
IE (8052)	0X000000B
TMOD	00H
TCON	00H
TH0	00H
TL0	00H
TH1	00H
TL1	00H
TH2(8052)	00H
TL2(8052)	00H
RCAP2H (8052)	00H
RCAP2L(8052)	00H
SCON	00H
SBUF	Indeterminate
PCON(HMOS)	0XXXXXXXXB
PCON(CHMOS)	0XXX0000B

2.4 RS-232

การสื่อสารแบบอนุกรมนับว่ามีความสำคัญต่อการใช้งานไมโครคอนโทรลเลอร์มาก เพราะสามารถใช้แป้นพิมพ์ และจอภาพของ PC เป็น อินพุต และ เอาต์พุต ในการติดต่อ หรือ ควบคุมไมโครคอนโทรลเลอร์ด้วยสัญญาณอย่างน้อยเพียง 3 เส้นเท่านั้น คือ

- สายส่งสัญญาณ TX
- สายรับสัญญาณ RX
- และสาย GND

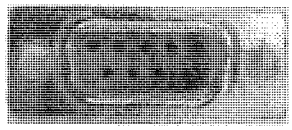
โดยปกติพอร์ตอนุกรม RS-232 จะสามารถต่อสายได้ยาว 50 ฟุตโดยประมาณ ขึ้นอยู่กับ ชนิดของสายสัญญาณ, ระยะทาง, และ ปริมาณสัญญาณรบกวน

พอร์ตอนุกรมของ PC จะเป็นคอนเน็คเตอร์แบบ DB9 ตัวผู้ (Male)

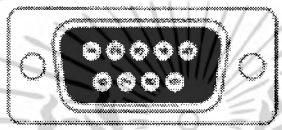
พอร์ตอนุกรมของอุปกรณ์ภายนอก จะเป็นคอนเน็คเตอร์แบบ DB9 ตัวเมีย (Female)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 2.7 พอร์ตอนุกรม DB9 ตัวผู้



รูปที่ 2.8 พอร์ตอนุกรม DB9 ตัวเมีย



รูปที่ 2.9 DB9 ตัวผู้ เมื่อมองจากด้านหลัง

ตารางที่ 2.3 การจัดขา ของคอนเน็กเตอร์ อนุกรมแบบ DB9 และหน้าที่การใช้งานต่างๆ

Pin	Description	Type
1	Data Carrier Detect (DCD)	Input
2	Received Data (RxD)	Input
3	Transmitted Data (TxD)	Output
4	Data Terminal Ready (DTR)	Output
5	Signal Ground (GND)	Input
6	Data Set Ready (DSR)	Input
7	Request To Send (RTS)	Output
8	Clear to Send (CTS)	Input
9	Ring Indicator (RI)	Input

จากตารางที่ 2.3 ความหมายของชื่อ pin ต่างๆ (แบบ 9 pin)

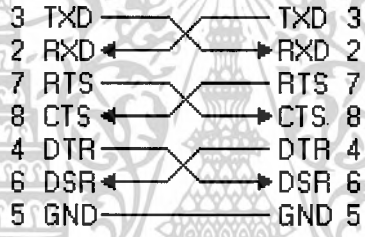
1. Signal Ground เป็น Ground ของสัญญาณ
2. Transmitted Data (TxD) เป็นขาที่ใช้ส่งข้อมูลแบบอนุกรมออกไปยังอุปกรณ์อื่นๆ
3. Received Data (RxD) เป็นขาที่ใช้รับข้อมูลอนุกรมจากอุปกรณ์ภายนอกเข้ามา
4. Request to Send (RTS) เป็นขาที่ใช้ส่งสัญญาณจากอุปกรณ์ DTE ให้อุปกรณ์ DCE เตรียมตัวรับข้อมูลได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5. Clear to Send (CTS) เป็นขาที่อุปกรณ์ DCE จะส่งสัญญาณตอบกลับอุปกรณ์ DTE ว่าการส่งข้อมูลจะเกิดขึ้นได้
6. DCE Ready (DSR) เป็นขาที่ใช้บอกว่าอุปกรณ์ DCE ต่ออยู่อย่างถูกต้อง
7. DTE Ready (DTR) เป็นขาที่ใช้บอกว่าอุปกรณ์ DTE ต้องการเปิดช่องสัญญาณในการสื่อสาร
8. Received line Signal Detector (DCD) บางทีก็เรียกว่า Data Carrier Detector เป็นขาที่ใช้ในการรติดต่อ ในกรณีที่อุปกรณ์ DCE เป็นโมเด็ม ขานี้จะรับสัญญาณจากการตรวจสอบ Answer tone ของโมเด็ม
9. Ring Indicator (RI) เป็นขาที่ใช้ในการตรวจสอบสัญญาณกระดิ่ง เมื่ออุปกรณ์ DCE เป็นโมเด็ม

2.4.1 การเชื่อมต่อแบบ Null modem

เป็นการเชื่อมต่อคอมพิวเตอร์ด้วยกันโดยใช้ Port อนุกรม โดยไม่ต้องใช้โมเด็ม การนำไปใช้งานต้องมีการต่อสายให้ถูกต้องดังนี้



รูปที่ 2.10 การเชื่อมต่ออุปกรณ์ภายนอกผ่าน DB9 แบบ Null modem

การทำงานของขาสัญญาณ DB9

TxD เป็นขาที่ใช้ส่งข้อมูล

RxD เป็นขาที่ใช้รับข้อมูล

DTR แสดงสภาวะพอร์ตว่าเปิดใช้งาน DSR ตรวจสอบว่าพอร์ต ที่ติดต่อด้วย เปิดอยู่หรือไม่

- เมื่อเปิดพอร์ตอนุกรม ขา DTR จะ ON เพื่อให้อุปกรณ์ได้รับทราบว่าต้องการติดต่อด้วย
- ในขณะเดียวกันก็จะตรวจสอบขา DSR ว่าอุปกรณ์พร้อมหรือไม่

RTS แสดงสภาวะพอร์ตว่าต้องการส่งข้อมูล CTS ตรวจสอบว่าพอร์ตที่ติดต่ออยู่ ต้องการส่งข้อมูลหรือไม่

- เมื่อต้องการส่งข้อมูลขา RTS จะ ON และจะส่งข้อมูลออกที่ขา TXD เมื่อส่งเสร็จก็จะ OFF
- ในขณะเดียวกันก็จะตรวจสอบขา CTS ว่าอุปกรณ์ต้องการที่จะส่งข้อมูลหรือไม่

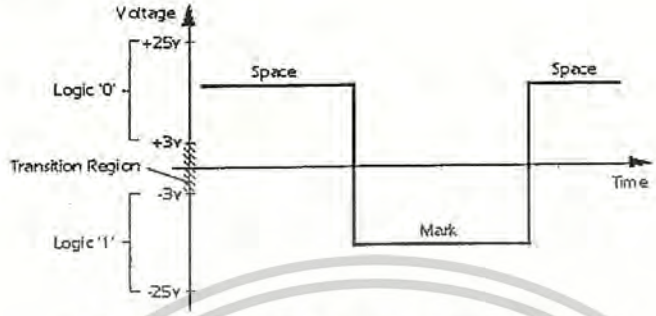
GND ขา ground

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

2.4.2 ระดับแรงดัน

ระดับแรงดันตามมาตรฐาน RS232 จะไม่เหมือนกับ logic แบบ TTL เนื่องจากในการส่งผ่านสายสัญญาณ จะมีสัญญาณรบกวนเกิดขึ้นและรบกวนข้อมูลทางของเราได้ มาตรฐานระดับแรงดัน logic 0 และ 1 แสดงในรูปที่ 2.11



รูปที่ 2.11 ระดับแรงดันในมาตรฐาน RS - 232

จากรูปที่ 2.11 จะเห็นได้ว่าช่วง logic 0 จะมีระดับแรงดันเป็น +3 ถึง +25 V ส่วนช่วง logic 1 จะมีระดับแรงดันเป็น -3 ถึง -25 แต่โดยทั่วไปช่วง logic 0 จะมีระดับแรงดันเป็น +8 ถึง +14 ส่วนช่วง logic 1 จะมีระดับแรงดันเป็น -8 ถึง -14

2.5 มอเตอร์กระแสตรง (DC Motor)

เครื่องใช้ไฟฟ้าประเภทเครื่องกล ที่ใช้งานกันอยู่โดยทั่วไปภายในอาคารบ้านเรือนและโรงงานอุตสาหกรรม ได้แก่ พัดลม เครื่องบด เครื่องปั่น เครื่องซักผ้า ปั๊มน้ำ เครื่องกลึง เครื่องไส เครื่องเจาะ เครื่องคว้าน เครื่องเจียระไน ฯลฯ ต่างก็ทำงานด้วยการหมุนขับของมอเตอร์ ดังนั้นมอเตอร์จึงเป็นเครื่องกลไฟฟ้าที่ให้กำเนิดพลังงานกลที่จำเป็นและสำคัญยิ่งประเภทหนึ่ง

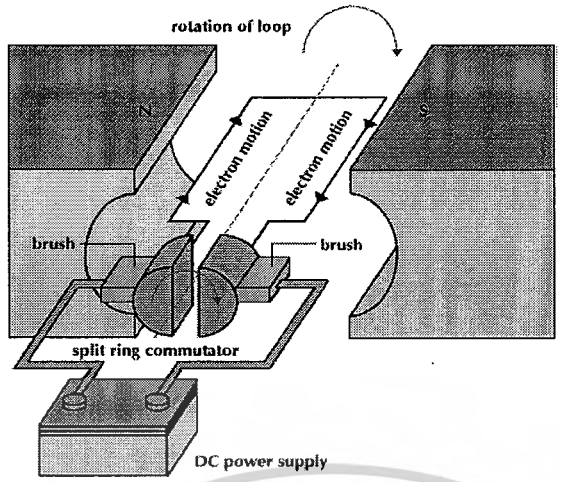
2.5.1 หลักการเบื้องต้นของมอเตอร์

มอเตอร์คือเครื่องกลไฟฟ้าที่เปลี่ยนพลังงานไฟฟ้า ให้เป็นพลังงานกล โดยทั่วไปมอเตอร์สามารถแบ่งออกเป็น 2 ชนิดใหญ่ ๆ ได้แก่ มอเตอร์ไฟฟ้ากระแสตรง (DC Motor) และมอเตอร์ ไฟฟ้ากระแสสลับ (AC Motor) สามารถที่จะควบคุมการทำงานของมอเตอร์ให้เป็นไปตามที่เราต้องการได้ การที่จะให้มอเตอร์หมุนไปในทิศทางใดนั้น ย่อมขึ้นอยู่กับความสัมพันธ์ระหว่างทิศทางของเส้นแรงแม่เหล็กระหว่างขั้วเหนือ ขั้วใต้ กับทิศของกระแสไฟฟ้าที่ไหลผ่านขดลวดอาร์เมเจอร์ ตามกฎมือซ้ายของเฟลมมิง ดังนั้นถ้าต้องการให้มอเตอร์กลับทางหมุนจะต้องกลับทิศทางการไหลของเส้นแรงแม่เหล็กหรือกลับทิศทางการไหลของกระแสไฟฟ้า อย่างใดอย่างหนึ่ง ถ้ากลับทิศทางทั้งคู่มอเตอร์จะหมุนในทิศทางเดิม ซึ่งการกลับทิศทางของมอเตอร์นั้นเป็นหัวใจสำคัญของการนำมอเตอร์ไปใช้งาน

2.5.2 โครงสร้างของมอเตอร์กระแสตรง (DC Motor Structure)

โครงสร้างของมอเตอร์กระแสตรงโดยพื้นฐานจะประกอบด้วยขดลวดสร้างสนามแม่เหล็ก (Field Winding) และขดลวด Armature โดยขดสร้างสนามแม่เหล็กที่ตัว Stator และขด Armature จะอยู่ที่ตัว Rotor ของตัวมอเตอร์ ดังรูปที่ 2.12

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้
103052

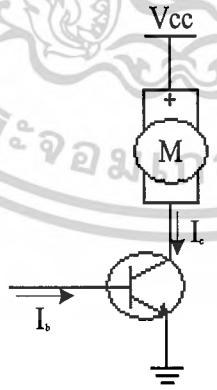


รูปที่ 2.12 โครงสร้างพื้นฐานของมอเตอร์กระแสตรง

แต่โครงสร้างของมอเตอร์ที่ใช้จริงในโครงงานนี้ ขดลวดที่สร้างสนามจะถูกเปลี่ยนเป็นแม่เหล็กถาวร โดยจะสร้างสนามแม่เหล็กที่มีค่าคงที่ตลอดเวลา ดังนั้นการควบคุมมอเตอร์ชนิดนี้จึงสามารถควบคุมค่าแรงดัน(V_t) และกระแส (I_a) ได้เท่านั้น โดยความเร็วของมอเตอร์สามารถควบคุมได้จากแรงดัน (V_t) ของมอเตอร์

2.5.3 วงจร H-Bridge Switching ควบคุมมอเตอร์รถ

จากรูปที่ 2.13 เป็นวงจรสาธิตแบบง่าย ๆ โดยการนำทรานซิสเตอร์ มาเป็นสวิทช์ควบคุมมอเตอร์ หลักการคิดง่าย ๆ ก็คือ เมื่อเราป้อนกระแส I_b ด้วยปริมาณที่มากพอ ก็จะทำให้ทรานซิสเตอร์ทำงาน (On) จะทำให้กระแส I_c ไหล แปลว่ามีกระแสไหลผ่านมอเตอร์ได้ (กระแส I_b จะต้องมากเพียงพอที่จะทำให้ทรานซิสเตอร์อยู่ในสภาวะ "อิ่มตัว" ได้)



รูปที่ 2.13 การทำงาน ของ วงจร H-Bridge Switching ควบคุมมอเตอร์รถ

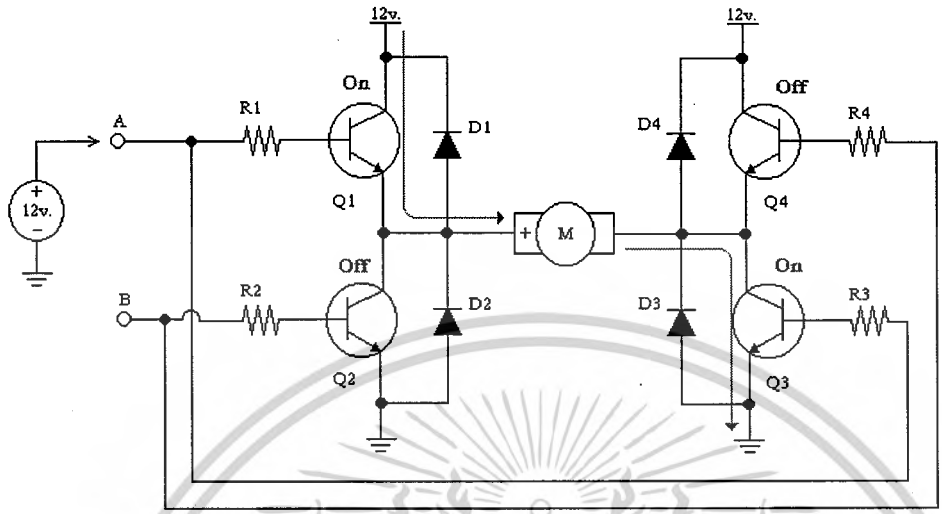
ในสภาวะ อิ่มตัว (Saturation mode) นี้ ทรานซิสเตอร์จะทำงานเหมือนกับสวิทช์ปิดวงจร ค่าความต้านทานระหว่างขา C และขา E จะมีค่าเข้าใกล้ศูนย์ กระแส I_c ที่ไหลจะมีค่าเข้าใกล้ I_c(max)

ในสภาวะ คัตออฟ (Cutoff mode) นี้ จะเกิดขึ้นเมื่อเราหยุดจ่ายกระแส Ib (Ib = 0) ทรานซิสเตอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จะทำงานเหมือนกับสวิทช์เปิดวงจร ค่าความต้านทานระหว่างขา C และขา E จะมีค่าเป็นอนันต์ กระแส Ic จะมีค่าเข้าใกล้ศูนย์

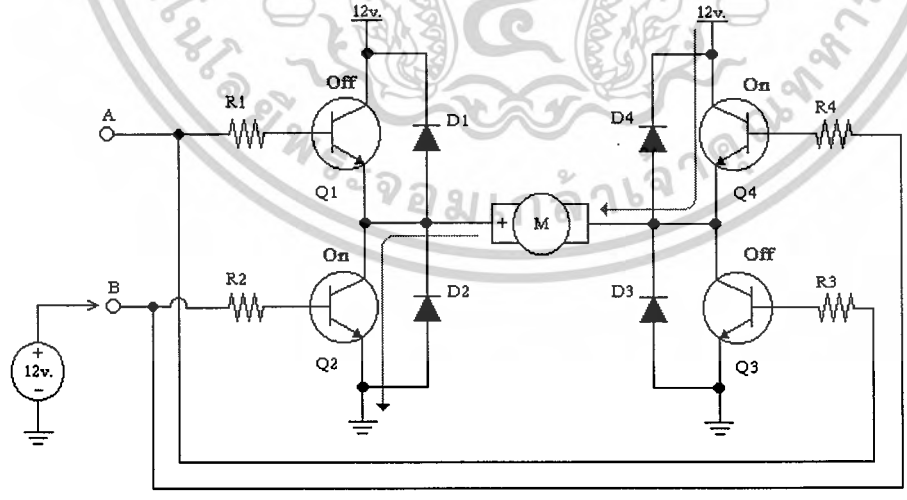
กรณีที่ Q1 และ Q3 ทำงาน



รูปที่ 2.14 อธิบายการทำงาน ของ วงจร H-Bridge Switching ควบคุมมอเตอร์รถ

เมื่อมีการจ่ายแรงดัน 12v. เข้าที่จุด A ทำให้มีกระแสไหลผ่าน R1 เข้าสู่ขา base ของ Q1 และมีกระแสไหลผ่าน R3 เข้าสู่ขา base ของ Q3 ทำให้ Q1 และ Q3 ทำงาน (On) เปรียบเสมือนสวิทช์ปิดวงจร ส่งผลให้มีกระแสไหลจากแหล่งจ่าย (12v.) ผ่านขา Collecter และEmitter ของ Q1 ผ่านเข้าสู่ขั้วบวก (+) ของมอเตอร์ ผ่านไปยังขา Collecter และEmitter ของ Q3 ทำให้มีกระแสไหลผ่านมอเตอร์ในทิศทางบวก และครบวงจร จึงทำให้มอเตอร์สามารถหมุน ในทิศทาง Forward ได้

กรณีที่ Q2 และ Q4 ทำงาน



รูปที่ 2.15 อธิบายการทำงาน ของ วงจร H-Bridge Switching ควบคุมมอเตอร์รถ

เมื่อมีการจ่ายแรงดัน 12v. เข้าที่จุด B ทำให้มีกระแสไหลผ่าน R2 เข้าสู่ขา base ของ Q2 และมีกระแสไหลผ่าน R4 เข้าสู่ขา base ของ Q4 ทำให้ Q2 และ Q4 ทำงาน (On) เปรียบเสมือนสวิทช์ปิดวงจร

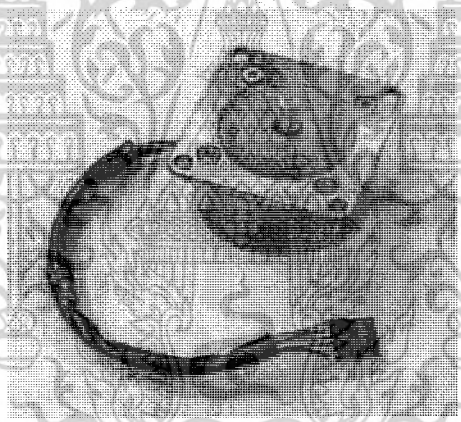
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ส่งผลให้มีกระแสไหลจากแหล่งจ่าย (12v.) ผ่านขา Collector และEmitter ของ Q4 ผ่านเข้าสู่ขั้วลบ (-) ของมอเตอร์ ผ่านไปยังขา Collector และEmitter ของ Q2 ทำให้มีกระแสไหลผ่านมอเตอร์ในทิศทางลบ และครบวงจร จึงทำให้มอเตอร์สามารถหมุน ในทิศทาง Reward ได้

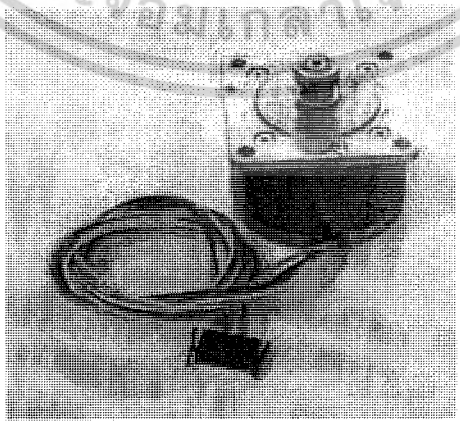
2.6 สเต็ปมอเตอร์ (Stepping Motor)

สเต็ปปิ้งมอเตอร์เป็นอุปกรณ์ทางไฟฟ้าที่มีลักษณะพิเศษเป็นตัวต้นกำลังแบบดิจิตอลที่สำคัญอย่างหนึ่ง สามารถทำงานโดยเปลี่ยนพลังงานไฟฟ้าเป็นพลังงานกลในรูปของการเคลื่อนที่ ซึ่งมีอินพุตเป็นกลุ่มของแรงดันไบนารี และเอาท์พุตเป็นลักษณะของการเคลื่อนที่เชิงมุมเป็นสเต็ปที่มีค่ามุมคงที่ โดยการจ่ายแรงดันหรือกระแสแก่ขดลวดแต่ละเฟสตามลำดับอย่างถูกต้อง การนำไปใช้งานเมื่อต้องการควบคุมการเคลื่อนที่แต่ละตำแหน่งที่ต้องการความถูกต้องแม่นยำ การเคลื่อนที่แต่ละสเต็ปเกี่ยวข้องกับสัญญาณพัลส์แต่ละอันที่จ่ายแก่ขดลวดสเตเตอร์ทั้งหมด จำนวนสเต็ปที่เคลื่อนที่จะเท่ากับสัญญาณพัลส์ที่จ่ายแก่มอเตอร์ สามารถเคลื่อนที่ทั้งตามเข็มและทวนเข็มนาฬิกาขึ้นอยู่กับการจัดลำดับการจ่ายพัลส์แก่ขดลวด

2.6.1 ชนิดและโครงสร้างของสเต็ปมอเตอร์

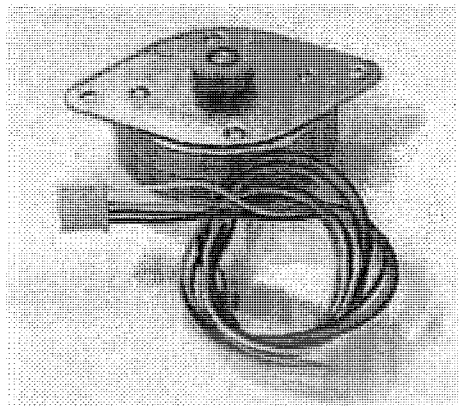


รูปที่ 2.16 สเต็ปมอเตอร์แบบมีสาย 5 เส้น

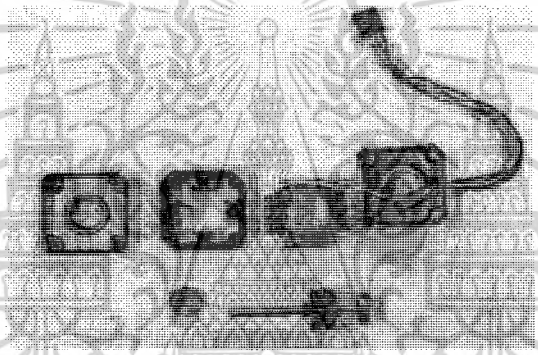


รูปที่ 2.17 สเต็ปมอเตอร์แบบมีสาย 6 เส้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.18 สเต็ปมอเตอร์หลายแบบ ไบโพลาร์



รูปที่ 2.19 ภาพถ่ายโครงสร้างสเต็ปมอเตอร์

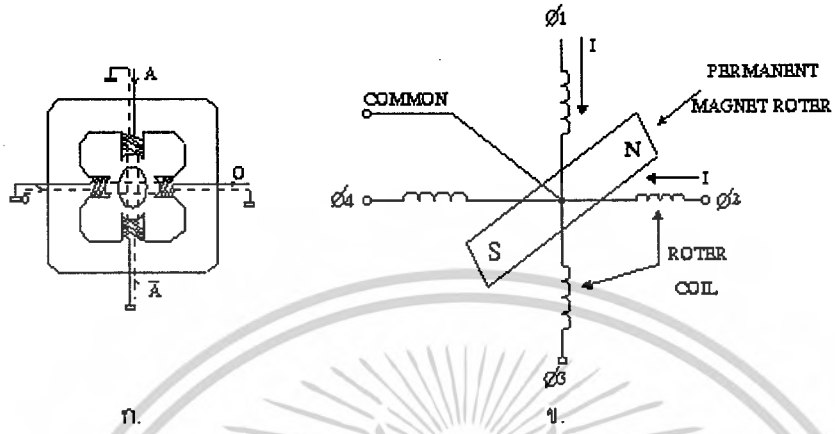
สเต็ปมอเตอร์ที่พบในปัจจุบันมี 3 ลักษณะดังนี้

1. แบบแม่เหล็กถาวร (Permanent Magnet - PM) สเต็ปมอเตอร์แบบ PM จะมีสเตเตอร์ (Stator) ที่พันขดลวดไว้หลาย ๆ โพล โดยมี โรเตอร์ (Rotor) เป็นรูปทรงกระบอกฟันเลื่อย และโรเตอร์ทำด้วยแม่เหล็กถาวร เพื่อป้อนไฟกระแสตรง ให้กับขดสเตเตอร์ จะทำให้เกิดแรงแม่เหล็กไฟฟ้าผลักต่อโรเตอร์ ทำให้มอเตอร์หมุนมอเตอร์แบบ PM จะเกิดแรงฉุดยึดให้โรเตอร์หยุดอยู่กับที่ แม้จะไม่ได้ป้อนไฟเข้าขดลวด

2. แบบแปรค่ารีลักแตนซ์ (Variable Reluctance - VR) สเต็ปมอเตอร์แบบ VR จะมีการหมุนโรเตอร์ได้อย่างอิสระ แม้จะไม่ได้จ่ายไฟให้โรเตอร์ทำจากสารเฟอร์โรแมกเนติก กำลังอ่อน มีลักษณะเป็นฟันเลื่อย รูปทรงกระบอกโดยจะมีความสัมพันธ์ โดยตรงกับจำนวนโพลในสเตเตอร์ แรงบิดที่เกิดขึ้นจะไปหมุนโรเตอร์ ไปในเส้นทางของอำนาจแม่เหล็กที่มีค่ารีลักแตนท์ต่ำที่สุด ตำแหน่งที่จะเกิดแน่นอนและมีเสถียรภาพแต่จะเกิดขึ้นได้หลาย ๆ จุดดังนั้นเมื่อป้อนไฟเข้าขดลวดต่าง ๆ ในมอเตอร์แตกต่างขดกันไป ก็ทำให้มอเตอร์หมุนไปตำแหน่งต่าง ๆ กันโรเตอร์ของ VR จะมีความเฉื่อยของโรเตอร์น้อยจึงมีความเร็วรอบสูงกว่ามอเตอร์แบบ PM

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3. แบบผสม (HYBRID-H) สเต็ปมอเตอร์แบบ H จะเป็นลูกผสมของ VR กับ PM โดยจะมีสเตเตอร์คล้ายกับที่ใช้ใน VR โรเตอร์มีหมวกหุ้ม ปลายซึ่งมีลักษณะของสารแม่เหล็กที่มีกำลังสูง โดยการควบคุมขนาดรูปร่างของหมวกแม่เหล็กอย่างดีทำให้ได้มุม การหมุนและครั้งน้อยและแม่นยำ ข้อดีก็คือ ให้แรงบิดสูงและมีขนาดกระทัดรัด และให้แรงฉุดยึดโรเตอร์นิ่งกับที่ตอนไม่จ่ายไฟ



รูปที่ 2.20 (ก) โครงสร้าง (ข) วงจรเทียบเท่า (equivalent circuit) ของมอเตอร์ ชนิด 4 ขด

ตารางที่ 2.4 มุมของโรเตอร์เทียบกับกระแสไฟฟ้าที่จ่ายแก่เฟสต่าง ๆ 8 ตำแหน่ง

เฟสที่จ่ายกระแสไฟฟ้า	ϕ_1	$\phi_1\phi_2$	ϕ_2	$\phi_2\phi_3$	ϕ_3	$\phi_3\phi_4$	ϕ_4	$\phi_4\phi_1$
ตำแหน่งโรเตอร์	⇧	⇗	⇨	⇘	⇩	⇙	⇦	⇖

จากลักษณะของมุมโรเตอร์หมุนกับกระแสไฟที่ป้อนแก่เฟสต่าง ๆ จะสามารถสั่งงานให้ สเต็ปมอเตอร์หมุนได้ 3 อย่าง คือ

1. แบบจ่ายกระแสไฟให้เฟสเดียววนเวียนกันไป เรียก ONE-EXCITATION หรือ HALF DRIVE คือ f1 , f2 f3, f4 การ OUT EXCITATION แบบนี้แรงบิดจะน้อย

2. แบบจ่ายกระแสไฟให้พร้อมกันทีเดียว 2 f เรียก TWO-EXCTATION หรือ FULL STEP คือ f1f2 , f2f3 , f3f4 , f4f1 หมุนเวียนกันไปแบบนี้แรงบิดจะมาก

3. แบบจ่ายกระแสไฟให้ทีละ 1 เฟส สลับกับ 2 เฟส เรียก ONE-TWO EXCITATION หรือ ครึ่งสเต็ป เหมือนรูปแสดงของมุมโรเตอร์ตารางที่ 2.6 แต่แบบนี้จำนวนสเต็ปทวนเข็มจะเป็นตรงกันข้าม

ตารางที่ 2.5 การจ่ายกระแสให้แก่เฟสต่าง ๆ

เฟส	ϕ_4	ϕ_3	ϕ_2	ϕ_1
ϕ_1	1	0	0	1
ϕ_2	0	0	1	0
ϕ_3	0	1	0	0
ϕ_4	1	0	0	0

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.6.2 การตรวจสอบหาสายร่วม และสายกราวนด์ ของสเต็ปมอเตอร์ แบบ PM (แบบแกนโรเตอร์เป็นแม่เหล็กถาวร)

โดยทั่วไป สเต็ปมอเตอร์ แบบ PM จะมีอยู่ 2 ชนิด

1. ชนิดที่เป็นสายร่วมภายนอก (สเต็ปมอเตอร์) แบบนี้มีสายอยู่ 6 เส้น คือ สายที่เป็นสายร่วม 2 เส้น สายที่เป็นกราวนด์ 4 เส้น สายร่วม 1 เส้น จะต่อกราวนด์ 2 เส้น ในการเช็คให้ใช้มิเตอร์วัดหาสายที่เป็นสายร่วมก่อนโดยการตั้งย่านการวัดของมิเตอร์ที่ R*1 จับที่สายทีละคู่ ถ้าหากวัดสายร่วมเทียบกับสายกราวนด์ ได้ถูกต้องค่าความต้านทานที่อ่านได้จะน้อย แต่ถ้าวัดผิดสาย คือวัดสายกราวนด์ เทียบกับกราวนด์ ค่าความต้านทานที่อ่านได้จะสูงกว่าแต่ถ้าวัดสายร่วมเทียบกับสายกราวนด์ ที่ไม่ใช่คู่กันแล้ว เข็มมิเตอร์ก็จะไม่กระดิก ให้ทดลองวัดเปรียบเทียบกันทีละคู่ ก็จะทราบว่าสายใดเป็นสายร่วม สายใดเป็นสาย กราวนด์

2. ชนิดที่เป็นสายร่วมภายใน SP มอเตอร์ แบบนี้มีสายอยู่ 5 เส้น คือ สายที่เป็นสายร่วม 1 เส้น สายที่เป็นกราวนด์ 4 เส้น ในการวัดให้ทำแบบเดียวกับการวัดสเต็ปมอเตอร์ชนิดสายร่วม ภายนอกแตกต่างกัน เพียงแบบสายร่วม ภายในสายร่วม 1 เส้น สายกราวนด์ 4 เส้น ดังนั้นหากสายเส้นใดเมื่อวัดเทียบกับสายเส้นอื่น แล้วมีค่าความต้านทานน้อยที่สุดสายเส้นนั้นเป็นสายร่วม และที่เหลืออีก 4 เส้นจะเป็นสายกราวนด์ การเรียงเฟสของสเต็ปมอเตอร์ แบบ PM เมื่อเราทราบว่าสายเส้นใดเป็นสายร่วมแล้วแต่เรายังไม่ทราบว่าสายกราวนด์ เส้นใดเป็นเฟสที่ 1 เฟสที่ 2 เฟสที่ 3 และเฟสที่ 4 ในการเรียงเฟสนั้นให้ใช้มิเตอร์วัดโดยนำไฟบวกเข้าที่สาย COMMON วัดเทียบกับสายกราวนด์เส้นใดก็ได้ 1 เส้น จะทำให้แกนโรเตอร์เคลื่อนไปข้างหน้า 1 สเต็ป เมื่อเปลี่ยนสายกราวนด์ เส้นแรกเป็นเส้นที่ 2 ลากมอเตอร์ไม่เคลื่อนที่ไปข้างหน้าแสดงว่าการเรียงเฟสไม่ถูกต้องก็ให้วัดเทียบกับสายกราวนด์เส้นใหม่ต่อไป หากมอเตอร์เคลื่อนที่ไปข้างหน้าตามก้น วัดที่สายกราวนด์เส้นต่อไปเรื่อย ๆ ก็จะทำให้ทราบว่าสายเส้นใดเป็นเฟสแรก สายเส้นใดเป็นเฟสที่ 2 เฟสที่ 3 และเฟสที่ 4 การเรียงเฟสของสเต็ปมอเตอร์ แบบ PM ทั้งชนิดที่เป็นสายร่วมภายนอกและชนิดที่เป็นสายร่วมภายใน ใช้หลักการเดียวกัน

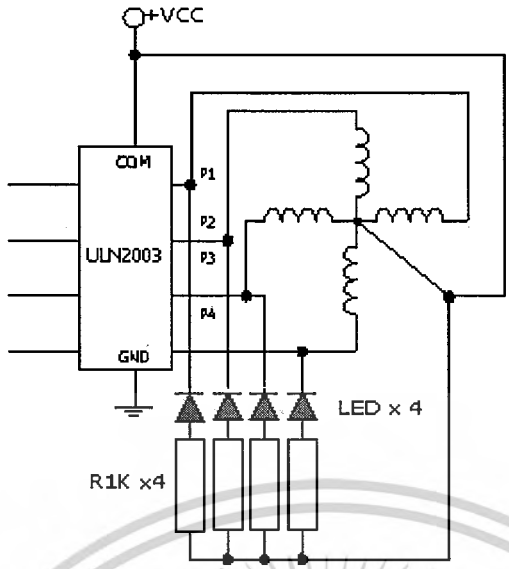
2.6.3 วงจรขับสเต็ปปิ้งมอเตอร์

สัญญาณกระตุ้นสำหรับสเต็ปปิ้งมอเตอร์ ปัจจุบันจะได้จากไอซีดิจิตอลหรือไมโครโปรเซสเซอร์ จะเป็นสัญญาณขนาดกำลังต่ำประเภทตระกูล ทีทีแอล 5 โวลต์ 18 มิลลิแอมป์ แต่สเต็ปปิ้งมอเตอร์ที่ใช้งานจะต้องการกระแสมาก ดังนั้นจึงจำเป็นจะต้องใช้วงจรสร้างสัญญาณกระตุ้นกับสเต็ปปิ้งมอเตอร์

2.6.4 ตัวอย่างวงจรขับสเต็ปปิ้งมอเตอร์

จากที่กล่าวมาข้างต้นในการควบคุมสเต็ปปิ้งมอเตอร์จะต้องมีการป้อนสัญญาณที่เป็นซีเควนเชียลออกไป โดยที่หลังจากเราป้อนสัญญาณจะต้องคอยเวลาดีเลย์จนกระทั่งสเต็ปปิ้งมอเตอร์หมุนไปเสร็จสิ้นแล้ว จึงจะสามารถส่งสัญญาณต่อไปได้ ซึ่งจะต้องมีภาคขับ (Driver Circuit) ที่สามารถจ่ายกระแสได้เพียงพอที่จะทำให้สเต็ปปิ้งมอเตอร์หมุนไปได้ ซึ่งในวงจรภาคขับนี้เราอาจจะใช้ไอซีแบบโอเพนคอลเลคเตอร์ เช่น ULN2003, ULN2803 เป็นต้น หรืออาจใช้เพาเวอร์ทรานซิสเตอร์หรือเพาเวอร์มอสเฟท เช่น BD139, IRF510 เป็นต้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.21 วงจรขับสเต็ปปิ้งมอเตอร์โดยใช้ไอซีเบอร์ ULN2003

2.7 โมดูล RS232 to RF-Wireless (RF2.4GHz) CONVERTER รุ่น ET-RF24G V1.0

ET-RF24G V1.0 เป็นชุด Signal Converter สำ หรับใช้แปลงสัญญาณระหว่าง RS232 และ RF-Wireless โดยในโหมดการทำ งานของการส่งข้อมูล (Transmitter) จะทำ หน้าที่รอรับข้อมูลจากพอร์ตสื่อสารอนุกรม RS232จากขา RX แล้วแปลงเป็นสัญญาณความถี่ (GFSK) ส่งออกไปในอากาศ และในทางกลับกันในโหมดการทำ งานแบบรับ (Receiver) ชุด ET-RF24G V1.0 ก็จะทำ หน้าที่คอยตรวจจับข้อมูลที่อยู่ในรูปของสัญญาณความถี่ (GFSK)จากด้าน RF เพื่อแปลงกลับเป็นข้อมูลแบบ RS232 ส่งออกไปทางขา TX ได้ด้วย ซึ่งชุดแปลงสัญญาณนี้สามารถนำไปต่อใช้งานร่วมกับพอร์ตสื่อสารอนุกรม แบบ RS232 เพื่อใช้งานในลักษณะของการสื่อสารอนุกรมแบบไร้สาย (Wireless Transceiver) ได้โดยตรงโดยเครื่อง ET-RF24G V1.0 นั้น ต้องการไฟเลี้ยงวงจร ซึ่งเป็นแหล่งจ่ายกระแสตรง ขนาดประมาณ +5VDC ถึง +9VDC

การทำงานแบบเป็นแบบ RF Auto Direction เป็นการทำ งานชนิด 2 ทิศทาง แบบ Half Duplex หรือ ผลัดกันรับผลัดกันส่ง ซึ่งสามารถใช้รับส่งข้อมูลระหว่างต้นทาง และ ปลายทาง ได้ โดยใช้เครื่อง ET-RF24G V1.0 ด้านละ 1 ชุด เท่านั้น เพียงแต่การรับส่งข้อมูลแบบนี้จะไม่สามารถส่งข้อมูลสวนทางกันได้เหมือนกับแบบ Full Duplex แต่จะต้องใช้วิธีการผลัดกันรับข้อมูลและส่งข้อมูลแทน โดยเมื่อฝ่ายรับทำการรับข้อมูลได้จนครบแล้วจึงจะสลับหน้าที่เป็นฝ่ายส่งเพื่อส่งข้อมูลย้อนกลับไป โดยในโหมดนี้ เครื่อง ET-RF24G V1.0 จะทำ หน้าที่เป็นทั้ง ฝ่ายรับ และ ฝ่ายส่ง ข้อมูล แบบอัตโนมัติ โดยในสภาวะปรกติจะอยู่ในสภาวะของการรอรับข้อมูล ทั้งด้าน RF และ RS232 ซึ่งถ้าพบว่ามีข้อมูลส่งเข้ามาทางด้านของ RF ก็จะนำ ข้อมูลนั้นส่งออกไปทางด้านขา TX ของ RS232 ทันที และในทำ นองเดียวกัน ถ้าพบว่ามีข้อมูลส่งเข้ามาทางด้าน RX ของ RS232 มันก็จะทำ การรับข้อมูลนั้นจาก RS232 พร้อมกับเปลี่ยนทิศทางของอุปกรณ์ RF จากการรอรับข้อมูลให้ทำ หน้าที่เป็นตัวส่งข้อมูลแทน เพื่อทำ การส่งข้อมูลที่รับได้จาก RS232 ออกไปทาง RF ในทันที ซึ่งหลังจากที่เครื่อง ET-RF24G V1.0 ทำ การสลับโหมดการทำ งานของอุปกรณ์ด้าน RF จากการรอรับเป็นการส่งและทำการเริ่มต้นส่งข้อมูลออกไปทางด้าน RF เรียบร้อยแล้ว มันจะวนกลับไป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตรวจสอบการรับข้อมูลจากด้าน RS232อีกว่ายังมีข้อมูลส่งเข้ามาอีกหรือไม่ ถ้าพบว่ายังมีข้อมูลส่งเข้ามีอีกก็จะทำ การแปลงข้อมูลนั้นเพื่อส่งออกไปยังด้านRF ต่อไปอีกจนกว่าการส่งข้อมูลด้าน RS232 จะสิ้นสุดลง ซึ่งข้อมูลด้าน RS232 ที่ส่งเข้ามานั้น ควรส่งอย่างต่อเนื่อง

โดยเมื่อเครื่อง ET-RF24G V1.0 ทำ การส่งข้อมูลแต่ละ Byte ออกไปทางด้าน RF เรียบร้อยแล้ว มันจะวนรอบรอรับข้อมูล Byte ถัดไปจาก RS232 ภายในเวลา 2.5 mS ถ้าไม่พบข้อมูลส่งเข้ามาอีกภายในระยะเวลาดังกล่าวมันจึงจะทำ การเปลี่ยนหน้าที่ของอุปกรณ์ด้าน RF ให้กลับมาทำ หน้าที่เป็นการรอรับข้อมูลตามเดิม โดยในขณะที่อุปกรณ์ด้าน RF ถูกกำ หนดให้เป็นฝ่ายส่งข้อมูลอยู่นั้น จะไม่สามารถทำ การรับข้อมูลจาก RF ได้ ซึ่งถ้ามีการส่งข้อมูลเข้ามาในขณะนั้นก็จะไม่สามารถรับได้ โดยค่าเวลาที่จะใช้ในการสลับโหมดการทำ งานของ RF จากฝ่ายส่งข้อมูลให้เป็นฝ่ายรับข้อมูลนั้น จะมีค่าเป็น 2.5mS ดังนั้นเมื่อฝ่ายรับสามารถรับข้อมูลได้ครบหมดแล้วก่อนที่จะทำ การส่งข้อมูลเพื่อตอบกลับไปยังฝ่ายตรงข้ามนั้น ควรทำการหน่วงเวลาไว้ไม่น้อยกว่า 3mS นับจากรับข้อมูล Byte สุดท้ายได้เรียบร้อยแล้วจึงเริ่มต้นส่งข้อมูล Byte แรกย้อนกลับไป ซึ่งถ้าฝ่ายรับทำ การส่งข้อมูลตอบกลับไปยังฝ่ายตรงข้ามเร็วกว่านี้อาจทำ ให้ฝ่ายตรงข้ามไม่สามารถรับข้อมูล Byte แรกได้ทันสามารถรับข้อมูลได้อย่างต่อเนื่องสูงสุด ไม่เกิน 64 Byte

ข้อดี

1. สามารถรับส่งข้อมูลกันได้ในระยะทางที่ไกลกว่า RS232 หลายเท่าตัว
2. ไม่จำเป็นต้องใช้สายสัญญาณที่เป็นตัวนำ สัญญาณทางไฟฟ้าในการสื่อสารข้อมูลกันทำให้สามารถเปลี่ยนแปลงหรือเคลื่อนย้ายจุดรับส่งข้อมูลได้ตลอดเวลา

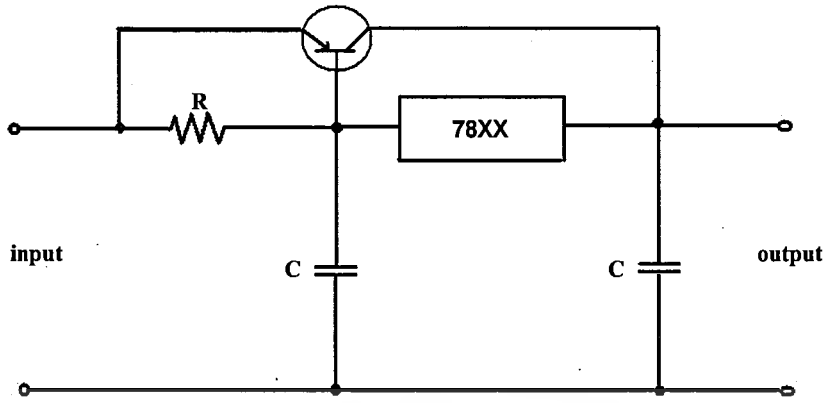
ข้อเสีย

1. การรับส่งข้อมูลโดยใช้อากาศเป็นตัวกลางในการสื่อสารนั้น มีข้อจำกัดบางประการโดยเฉพาะอย่างยิ่ง เรื่องความน่าเชื่อถือของข้อมูลที่รับส่งกัน ซึ่งมีโอกาสผิดพลาดหรือสูญหายได้เหมือนกัน เนื่องจากในการลำเลียงข้อมูลนั้นใช้อากาศเป็นตัวกลางในการรับส่งข้อมูลแทนซึ่งมีโอกาศที่ข้อมูลจะเกิดการรบกวนจากสัญญาณอื่นๆที่มีย่านความถี่ใกล้เคียงกันแล้วทำให้ข้อมูลผิดเพี้ยนไปได้บ้างเหมือนกัน

2.8 วงจรรักษาแรงดันใช้ไอซีรักษาแรงดัน (IC Regulator)

ข้อดีของวงจรแบบนี้คือ ราคาถูก มีขนาดเล็ก และรูปแบบวงจรที่ง่าย สามารถจ่ายกระแสเอาท์พุทได้ตั้งแต่ 3 mA ถึง 100 mA ตามเบอร์ที่เราเลือกใช้ ยังมีวงจรป้องกันกระแสเกินภายในและวงจรป้องกันอุณหภูมิเกินภายในด้วย โดย IC เบอร์ต่างๆ จะมีคุณสมบัติด้านกระแสเอาท์พุทสูงสุดแรงดันอินพุทไลน์เรคกูเลชั่น โหลดเรคชัน และช่วงอุณหภูมิที่ทำงานให้เราเลือกตามต้องการวงจรรักษาแรงดับแรงดันแบบ IC

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.22 วงจรรักษาแรงดันแบบแรงดันคงที่ใช้ MC 78xx

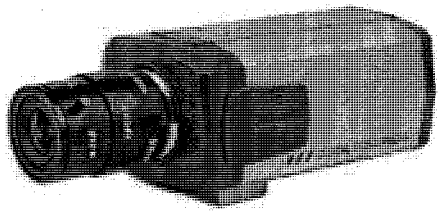
2.9 กล้องโทรทัศน์วงจรปิด

กล้องโทรทัศน์วงจรปิดเป็นอุปกรณ์สำคัญสำหรับงานระบบโทรทัศน์วงจรปิด เพราะจะเป็นอุปกรณ์ที่ทำให้เราสามารถมองเห็นภาพที่เกิดขึ้นในบริเวณที่เราต้องการได้ ซึ่งการมองเห็นของภาพนี้จะชัดเจนมากหรือน้อยเพียงใดก็ขึ้นอยู่กับชนิดของกล้องโทรทัศน์วงจรปิดที่จะนำมาใช้งานในระบบ ดังนั้นการเลือกใช้งานกล้องโทรทัญศน์วงจรปิดที่เหมาะสมตามรูปแบบของการใช้งานจะสามารถทำให้เกิดประสิทธิภาพในการทำงานของระบบที่ดีขึ้นได้ ซึ่งสามารถแยกชนิดของกล้องโทรทัศน์วงจรปิดได้หลายชนิดตามลักษณะของอุปกรณ์ที่เป็นส่วนประกอบของกล้องโทรทัศน์วงจรปิด ซึ่งจะขออธิบายถึงกล้องชนิดที่มีใช้งานอยู่ทั่วไปในปัจจุบัน 2 ชนิด คือ

1) **กล้องโทรทัศน์วงจรปิดชนิด CCD (Charge Coupled Device)**

กล้องโทรทัศน์วงจรปิดแบบนี้ใช้อุปกรณ์การแปรเปลี่ยนสัญญาณภาพเป็นลักษณะของชิพไอซีอิเล็กทรอนิกส์ มีให้เลือกตั้งแต่ขนาด 1/6 1/3 1/2 และ 2/3 นิ้ว ลักษณะของกล้อง CCD จะมีลักษณะเป็นดังรูปที่ 2.26 กล้องโทรทัศน์วงจรปิดชนิดนี้เป็นกล้องวงจรปิดชนิดกลาง มีขนาดกว้าง 4.6 ซม. ยาว 9 ซม. โดยประมาณ ตัวกล้องใช้เลนส์ CCD มีความคมชัดสูงสามารถเปลี่ยนเลนส์ซูมภาพได้ ความคมชัดของภาพจะอยู่ที่ประมาณ 380 – 420 เส้น ที่แสงสว่างตั้งแต่ 1.5 – 0.45 ลักซ์ จะเหมาะสำหรับติดตั้งในออฟฟิตสำนักงาน ห้างร้านมินิมาร์ท หรือภายในตึกอาคารต่างๆ ตัวกล้องมีความคงทนสูงติดตั้งง่าย มีอายุการใช้งานยาวนานถึง 10 – 20 ปี ภาพที่ออกมาเป็นภาพสีมีความคมชัดสูงสามารถต่อสายสัญญาณแบบหัวต่อ BNC และ RCA ได้ยาวถึง 300 เมตร สามารถใช้เลนส์ซูมแบบออโต้ไอริสต่อเข้ากล้องได้ ใช้กำลังไฟในการทำงาน 12 โวลต์

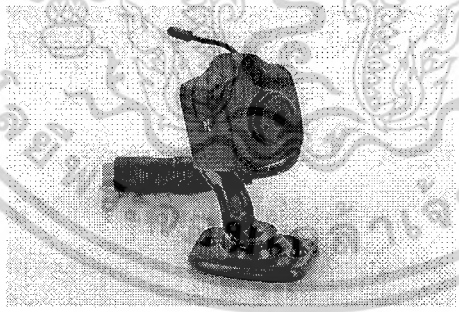
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.23 กล้องโทรทัศน์วงจรปิดชนิด CCD

2) กล้องโทรทัศน์วงจรปิดชนิด CMOS (Complementary Metal Oxide Semiconductor)

กล้องโทรทัศน์วงจรปิดแบบนี้ใช้อุปกรณ์การแปรเปลี่ยนสัญญาณภาพเป็นลักษณะของชิพไอซีอิเล็กทรอนิกส์ขนาดเล็กมากเรียกว่าเลนส์แบบ CMOS มีขนาดที่เล็กกว่าแบบ CCD มาก ขนาดกล้องยาวประมาณ 3.5 ซม. สามารถจะแสดงภาพสีหรือภาพแบบขาวดำก็ได้ แล้วแต่การเลือกการใช้งาน สามารถต่อดูภาพและเสียงผ่านทางช่อง AV ของเครื่องรับโทรทัศน์ได้ทันทีผ่านทางสายสัญญาณ AV ที่มีความยาวได้ถึง 100 เมตร ใช้กำลังไฟฟ้าในการทำงาน 9 โวลต์ และบางรุ่นสามารถใช้แบตเตอรี่แบบ 9 โวลต์ต่อเข้ากับตัวกล้องได้ กล้องชนิดนี้เหมาะสำหรับติดตั้งในออฟฟิต สำนักงาน ห้างร้านมินิมาร์ทที่ต้องการไม่ให้บุคคลภายนอกทราบได้ว่ามีการติดตั้งกล้องโทรทัศน์วงจรปิดไว้ เนื่องจากขนาดของกล้องที่มีขนาดเล็กมากนี้จะทำให้สามารถติดตั้งไว้ในจุดที่บุคคลภายนอกไม่สามารถพบเห็นได้ ลักษณะของกล้อง CMOS จะมีลักษณะเป็นดังรูปที่ 2.26



รูปที่ 2.24 กล้องโทรทัศน์วงจรปิดชนิด CMOS

2.9.1 การเปรียบเทียบเทคโนโลยีของ CMOS และ CD

จะเห็นว่าเทคโนโลยี CMOS จะมีจุดเด่นที่มีราคาถูก ใช้พลังงานต่ำ และทำให้มีขนาดเล็กได้ง่ายกว่า ยิ่งหากใช้เทคโนโลยีล่าสุดมาใช้ในการผลิตก็สามารถลดสัญญาณรบกวนได้มากกว่าในอดีต (ตัวอย่างเช่นกล้องดิจิตอลแคนนอนหลายรุ่นใช้ CMOS เซนเซอร์ในการผลิตตัวรับแสง) แต่สำหรับกล้อง CMOS ที่อยู่ในตลาดกล้องวงจรปิด โดยเฉพาะหากผลิตมากจากประเทศจีนคงคาดหวัง “แลตเตอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

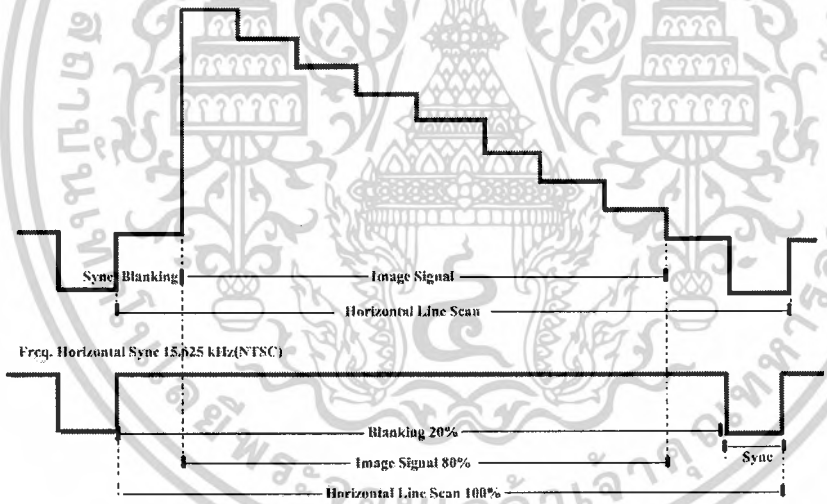
เทคโนโลยี" ได้ยาก การใช้งานจึงต้องเน้นรองรับการใช้งานแบบ พอดูได้ว่ามีคนหรือวัตถุอยู่ตรงหน้ากล้องเท่านั้น หากต้องลงรายละเอียดมากก็คงไม่ได้ รวมทั้งการเดินสายยาวๆ ก็จะไม่เหมาะสมทั้งกำลังของสัญญาณที่ต่ำกว่าและการถูกรบกวนของสัญญาณที่ถูกรบกวนได้ง่ายกว่า

ส่วนกรณีของ CCD พบว่ามีจุดเด่นที่มีความไวแสงสูงกว่า สัญญาณรบกวนต่ำกว่า และมีช่องว่างในการรับแสงที่น้อยกว่า คือตั้งแต่ UV จนถึง IR ได้เลยทีเดียว ซึ่งเป็นคุณสมบัติหลักที่อุปกรณ์รับแสงในกล้องพึงจะมี การใช้งานจึงให้ภาพที่สว่างใสกว่า ให้รายละเอียดมิติมากกว่าและสามารถเดินสายไกลได้ยาวกว่ากล้อง CMOS แต่ราคาของกล้อง CCD ก็สูงกว่าด้วยเช่นกัน

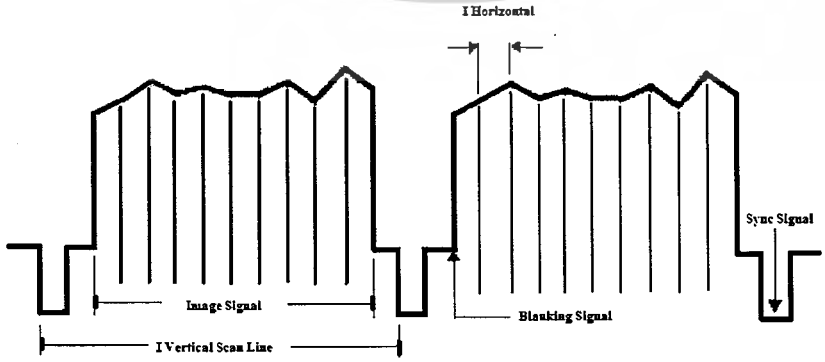
ส่วนของเรื่องอายุการใช้งานแม้ว่ากล้อง CMOS จะมีการสะสมความร้อนที่ต่ำกว่าแต่ด้วยมาตรฐานการผลิตที่สูงกว่าของ CCD ทำให้อายุการใช้งานของกล้อง CCD ที่ผลิตจากโรงงานที่น่าเชื่อถือสามารถใช้งานได้ยาวนานกว่าเช่นกัน

2.9.2 ลักษณะของสัญญาณภาพ

สัญญาณภาพโดยทั่วไปส่วนใหญ่จะมีลักษณะเป็นสัญญาณคอมโพสิตวีดีโอคือ จะประกอบไปด้วยข้อมูลสัญญาณภาพ (Image Signal), สัญญาณซิงค์ (Sync Signal) และสัญญาณแบลงกิ้ง (Blanking Signal) โดยลักษณะของสัญญาณภาพดังกล่าวแสดงดังรูปที่ 2.25

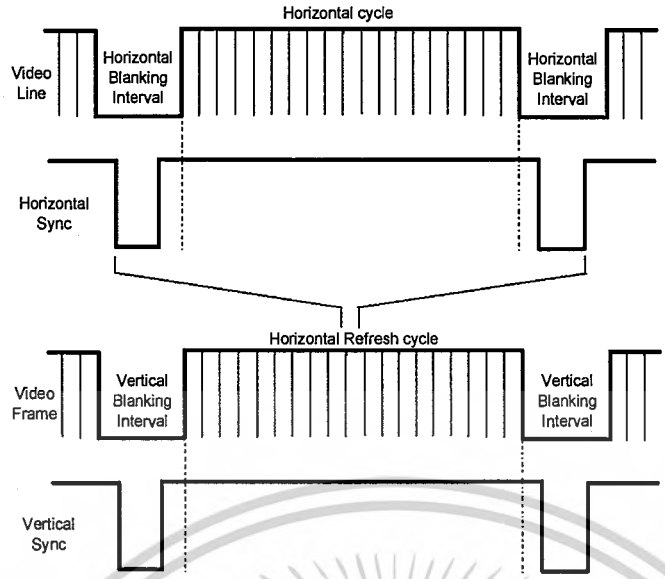


รูปที่ 2.25 ลักษณะของสัญญาณทางด้านแนวนอน



รูปที่ 2.26 ลักษณะของสัญญาณทางด้านแนวตั้ง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.27 ความสัมพันธ์ระหว่างสัญญาณทางด้านแนวนอนและแนวตั้ง

สัญญาณภาพที่แสดงในรูปที่ 2.27 จะเป็นสัญญาณภาพรวมที่ประกอบด้วย เส้นสแกนทางแนวนอนและเส้นสแกนทางแนวตั้ง โดยเส้นสแกนทั้งสองจะประกอบด้วยสัญญาณซิงค์ สัญญาณแบลงกิ้ง และสัญญาณภาพ สัญญาณดังกล่าวจะถูกส่งไปยังมอนิเตอร์ทำให้เกิดการสแกนที่หน้าจอมอนิเตอร์ ซึ่งจะทำให้ปรากฏเป็นภาพขึ้นมา

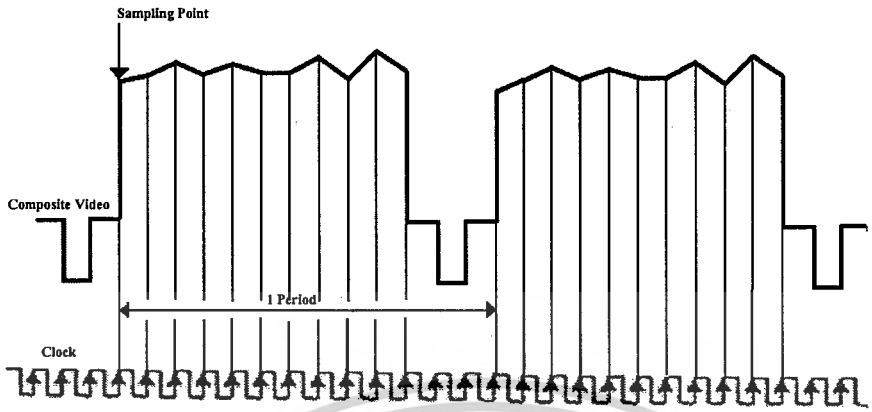
การจัดเก็บสัญญาณาภพจากคอมโพสิตวีดีโอ ซึ่งจะมีลักษณะเป็นสัญญาณอนาล็อกเข้าไปเก็บไว้ยังหน่วยความจำ ซึ่งจำเป็นอย่างยิ่งที่จะต้องทำการแปลงสัญญาณภาพดังกล่าวจากสัญญาณอนาล็อกไปเป็นสัญญาณดิจิตอล เพื่อให้สามารถที่จะเก็บเข้าไปยังหน่วยความจำได้ ในการแปลงสัญญาณอนาล็อกนั้นก็ต้องประกอบด้วย ส่วนของการสุ่มตัวอย่างสัญญาณและการควอนไตซ์สัญญาณ (Quantization) หรือการจัดระดับสัญญาณ ซึ่งการจัดระดับสัญญาณนั้นกระทำได้โดยการนำสัญญาณที่ผ่านสุ่มสัญญาณแล้วนำมาทำการแปลงให้เป็นสัญญาณทางดิจิตอล โดยสัญญาณภาพที่ถูกเปลี่ยนเป็นสัญญาณดิจิตอลแล้วสามารถที่จะถูกจัดเก็บลงในหน่วยความจำได้ แต่ในการสแกนทางด้านแนวนอนของสัญญาณภาพนั้นจะใช้เวลาในการสแกนที่สั้นมาก ดังนั้นจึงทำให้เกิดปัญหาในขั้นตอนของการเปลี่ยนสัญญาณภาพจากสัญญาณอนาล็อกให้เป็นสัญญาณดิจิตอล (Analog to Digital Converter) รวมทั้งการเขียนข้อมูลเข้าไปเก็บยังหน่วยความจำ

ปัญหาแรกคือ ปัญหาในเรื่องของการแปลงสัญญาณอนาล็อกเป็นดิจิตอล การที่สัญญาณภาพมีเวลาที่ใช้ในการสแกนทางด้านแนวนอนที่สั้นมาก ในการจัดเก็บเพื่อให้ได้ความละเอียดของภาพคงเดิมดังเช่นที่แสดงทางหน้าจอมอนิเตอร์ จำเป็นจะต้องใช้วงจรแปลงสัญญาณอนาล็อกเป็นสัญญาณดิจิตอลที่มีอัตราในการสุ่มตัวอย่างที่สูงพียงพอกับสัญญาณดังกล่าว

ปัญหาที่สองนั้นเกิดจากข้อมูลที่ถูกแปลงแล้วจะนำไปเขียนยังหน่วยความจำ ซึ่งในการเขียนข้อมูลดังกล่าวนี้ จะต้องมีการจัดเวลาอย่างเหมาะสมเพื่อไม่ให้เกิดปัญหาขึ้นในตอนของการเขียนข้อมูลลง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หน่วยความจำ ในการจัดเก็บข้อมูลภาพลงสู่หน่วยความจำนั้นจะต้องอาศัยหลักการจัดเก็บภาพลงหน่วยความจำแบบแอดเดรสเป็นแบบต่อเนื่อง ดังแสดงดังรูปที่ 2.28



รูปที่ 2.28 เวลาในการสุ่มสำหรับเส้นสแกนทางแนวนอนของสัญญาณคอมโพสิตวีดีโอ

2.9.3 การสแกนภาพและเรื่องที่เกี่ยวข้อง

ภาพบนจอหลอดภาพของเครื่องรับโทรทัศน์สีโดยทั่วไปจะประกอบไปด้วยเส้นขวางเล็กๆ ในแนวนอนจำนวนมาก ซึ่งแต่ละเส้นเหล่านี้มีทั้งส่วนที่ดำสนิทหรือมีสีเข้ม ส่วนที่ดำจางหรือมีสีจาง และส่วนที่สว่างมากปะปนกันอยู่ เส้นขวางเล็กๆ ในแนวนอนเหล่านี้มีชื่อเรียกว่าเส้นสแกน เส้นเหล่านี้ประกอไปด้วยจุดเล็กๆ ซึ่งมีทั้งมืดและสว่างปะปนกันอยู่ ภาพที่ปรากฏบนจอหลอดภาพจึงประกอบด้วยจุดเล็กๆ ที่มีระดับของความสว่างแตกต่างกันเป็นจำนวนมาก จุดเล็กๆ เหล่านี้เรียกว่าส่วนประกอบของภาพหรือพิคเจอร์ อีลิเมนต์ ซึ่งมีส่วนสำคัญกับความละเอียดของภาพเช่นเดียวกับจุดดำหรือจุดเล็กๆในรูปภาพของสิ่งตีพิมพ์ ภาพที่เห็นบนจอหลอดภาพจะเห็นรายละเอียดน่าดูหากจำนวนจุดเล็กๆ หรือจำนวนเส้นสแกนในแนวนอนมากเพียงพอ อย่างไรก็ตามภาพที่เห็นบนหลอดจอภาพและระยะทางที่มองดูภาพ เป็นต้น สำหรับโทรทัศน์ระบบอเมริกันซึ่งมีจำนวนเส้นสแกนน้อยกว่า แต่ถ้ามองดูภาพในระยะห่างประมาณ 4 – 8 เท่า ของความสูงของภาพแล้วจะรู้สึกว่าภาพพอใช้ได้เหมือนกัน จุดที่เห็นสว่างในหลอดภาพของจอภาพเกิดขึ้นเพราะอิเล็กตรอนที่หลุดออกจากแคโทดถูกดึงให้วิ่งเป็นลำไปกระทบแอโนดหรือจอหลอดภาพ ซึ่งฉาบวัสดุเรืองแสงบางชนิดเอาไว้ การสแกนหลอดภาพมี 2 วิธี คือ

1) การสแกนแบบก้าวหน้า (Progressive scanning)

2) การสแกนแบบสลับเส้น (Interlaced scanning)

การที่จะให้การสแกนมีความต่อเนื่องขององค์ประกอบภาพดังที่กล่าวมาแล้วจะต้องคำนึงถึงหลัก 3 ประการ คือ

1) ลำอิเล็กตรอนที่กวาดในแนวนอน (Horizontal scanning) ในแต่ละครั้งจะต้องครอบคลุมองค์ประกอบภาพทั้งหมดของเส้นนั้นๆ

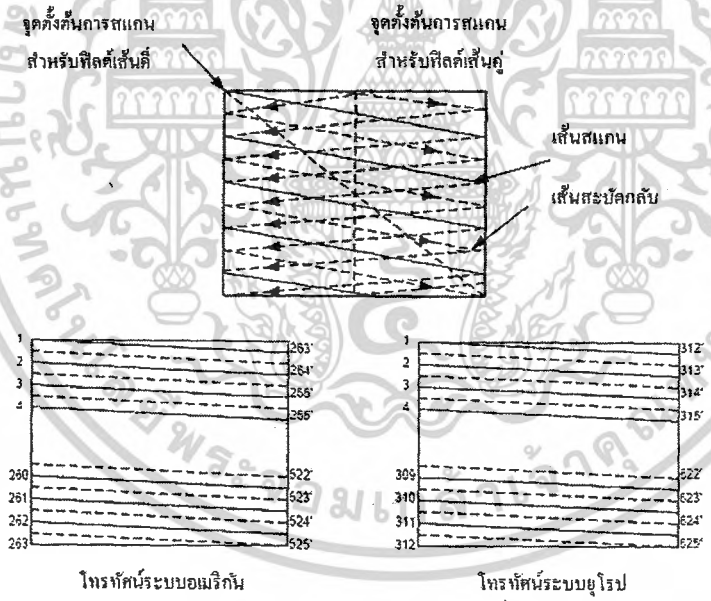
2) ในแต่ละเส้นของการสแกนลำแสงอิเล็กตรอน ลำแสงจะกวาดกลับไปยังด้านซ้ายเพื่อเริ่มต้นภาพทางแนวนอนลำดับต่อไป เวลาของการสะบัดกลับเราเรียกว่า “รีเทรซ” หรือ ฟลายแบ็ค ในกรณี

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ดังกล่าวจะต้องไม่มีข้อมูลภาพใดๆ เพราะว่าทั้งกล้องถ่ายและหลอดภาพจะเกิดอาการแบลงค์เอาท์ในขณะนั้น

3) ในขณะที่เส้นสแกนสะบัดกลับมาเพื่อเริ่มต้นทางซ้ายใหม่ ตำแหน่งทางแนวตั้งต้องต่ำกว่าตำแหน่งเดิมเพื่อทำให้การสแกนเส้นต่อไปไม่ทับกัน ทั้งนี้โดยการควบคุมของสัญญาณในแนวตั้ง (Vertical scanning)

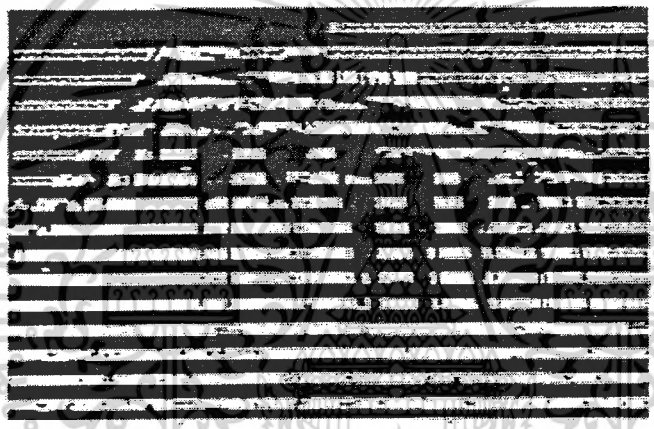
การสแกนที่ใช้ในเครื่องรับโทรทัศน์ ถึงแม้เราจะพบว่าหากให้มีการเรียงภาพเกินกว่า 16 ภาพต่อวินาที แล้วสายตาจะเห็นเป็นภาพต่อเนื่อง จากการทดลองเส้นภาพเราพบว่าแม้ภาพที่เกิดขึ้นจะเกิดขึ้น 24 ภาพต่อวินาทีแล้วก็ตาม ก็ยังมีการกระพริบ (Flicker) เกิดขึ้นเนื่องจากว่าในขณะที่การสแกนเริ่มจากขอบบนลงมาด้านล่าง (ซึ่งคล้ายกับการเขียนหนังสือเริ่มจากซ้ายไปขวา บนลงล่าง) เมื่อเส้นสแกนลงมาถึงขอบด้านล่างแสงทางด้านบนในความรู้สึกของมนุษย์จะเริ่มมืดกว่าด้านล่าง เวลาที่ลำแสงการสแกนวกกลับไปด้านบน ด้านล่างก็เกิดปัญหาเช่นเดียวกัน ความรู้สึกต่อกรณีนี้คือเกิดแสงกระพริบหรือวูบวาบขึ้น จึงต้องใช้การสแกนสลับเส้นหรือบางคนเรียกว่าการสแกนแบบสอดแทรก โดยครั้งแรกจะสแกนฟิลด์คู่ (Even line trace) เป็นการสแกนแบบเส้นเว้นเส้นนั่นหมายถึงว่า การได้ภาพหนึ่งภาพหรือที่เรียกว่าภาพหนึ่งเฟรมต้องใช้การสแกนถึง 2 ครั้งหรือ 2 ฟิลด์



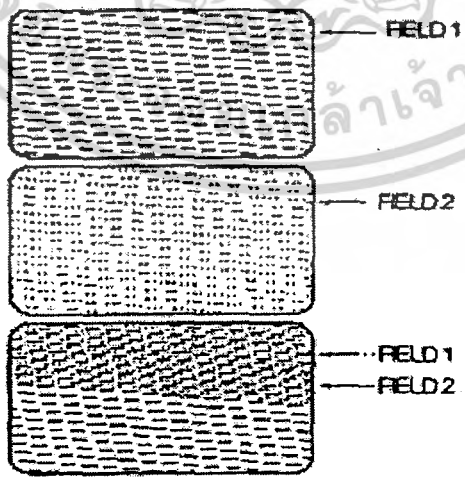
รูปที่ 2.29 การสแกนสองครั้งสำหรับภาพนิ่งแต่ละภาพ โดยแบ่งหนึ่งเฟรมออกเป็นสองฟิลด์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 2.30 รูปที่เกิดจากการสแกนฟิลด์คู่ (Even line trace)



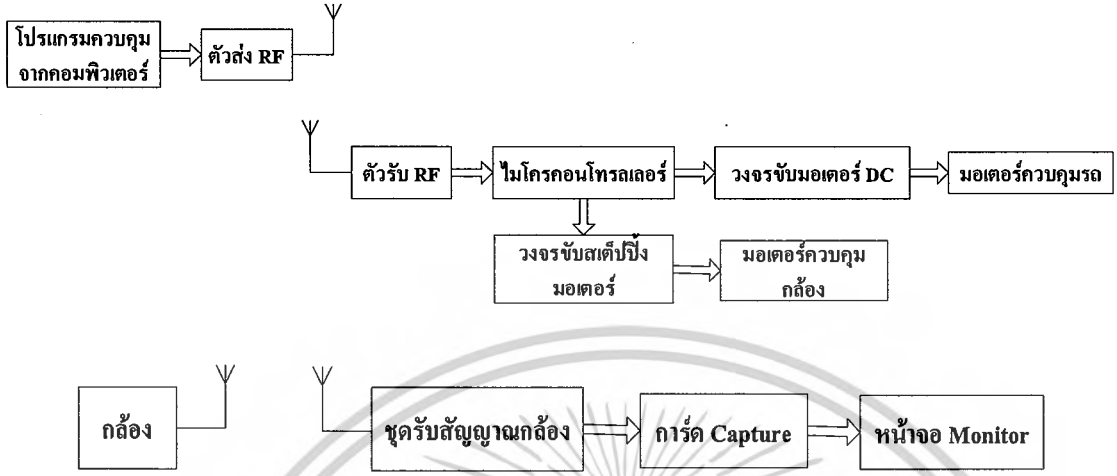
รูปที่ 2.31 รูปที่เกิดจากการสแกนฟิลด์คี่ (Odd line trace)



รูปที่ 2.32 การรวมฟิลด์ 1 และฟิลด์ 2

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 3
การออกแบบและการสร้าง



รูปที่ 3.1 บล็อคไดอะแกรมการทำงานรวมของระบบ

จากบล็อคไดอะแกรมสามารถอธิบายการทำงานของระบบได้ดังนี้ เมื่อเราเปิดโปรแกรมขึ้นมาใช้งาน คำสั่งที่เราเลือกใช้ซึ่งเราได้กำหนดไว้ที่ปุ่มต่างๆ จะถูกส่งไปยังเครื่องส่ง RF Wireless ผ่านทางพอร์ทอนุกรมและที่ตัวรถสำรวจจะมีเครื่องรับ RF นำสัญญาณที่ได้ส่งไปที่ไมโครคอนโทรลเลอร์ทำการประมวลผลทำตามคำสั่งที่เราต้องการ โดยแบ่งเป็นสองส่วนหลักๆ คือ ควบคุมทิศทางของรถสำรวจและควบคุมทิศทางการหมุนของกล้อง

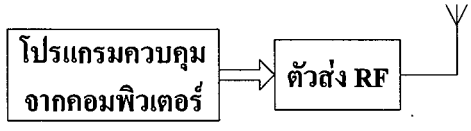
สำหรับการรับส่งสัญญาณภาพของกล้องวีดีโอจะเป็นแบบไวเลส โดยส่งสัญญาณภาพจากกล้องมายังตัวรับ Radio AV Receiver ผ่านการ์ดแคปเจอร์เพื่อแปลงสัญญาณภาพอะนาล็อกเป็นสัญญาณดิจิตอลแล้วแสดงผลออกทางหน้าต่างโปรแกรม

โดยเราสามารถแบ่งการทำงานออกเป็นส่วนต่างๆ ได้ดังนี้

1. ส่วนของภาคส่งสัญญาณ
2. ส่วนของภาครับสัญญาณ

3.1 ส่วนของภาคส่งสัญญาณ

ในส่วนนี้จะทำหน้าที่สั่งงานรถสำรวจและกล้อง การเข้ารหัสสัญญาณๆไปกับคลื่นวิทยุ โดยมีบล็อคไดอะแกรมเป็นดังนี้



รูปที่ 3.2 บล็อคไดอะแกรมของภาคส่ง

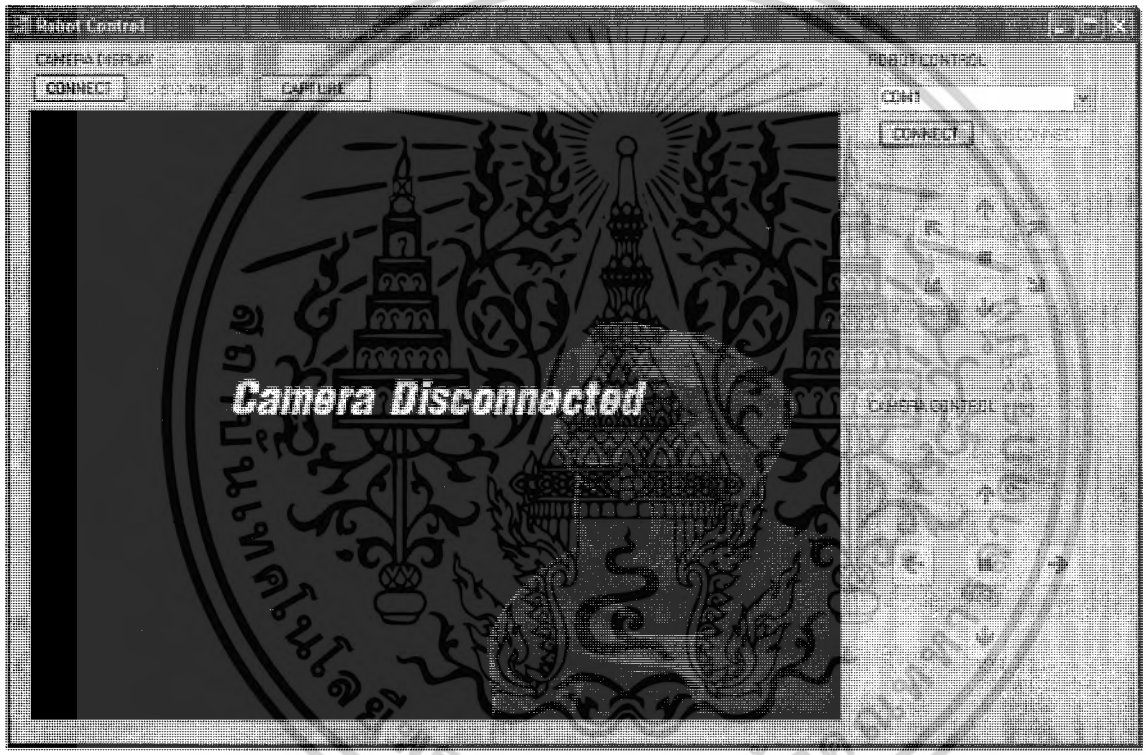
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ในส่วนของภาคส่งนี้จะมีส่วนที่เกี่ยวข้องคือ

- โปรแกรมควบคุมจากคอมพิวเตอร์
- ตัวรับส่ง RF

3.1.1 โปรแกรมควบคุมจากคอมพิวเตอร์

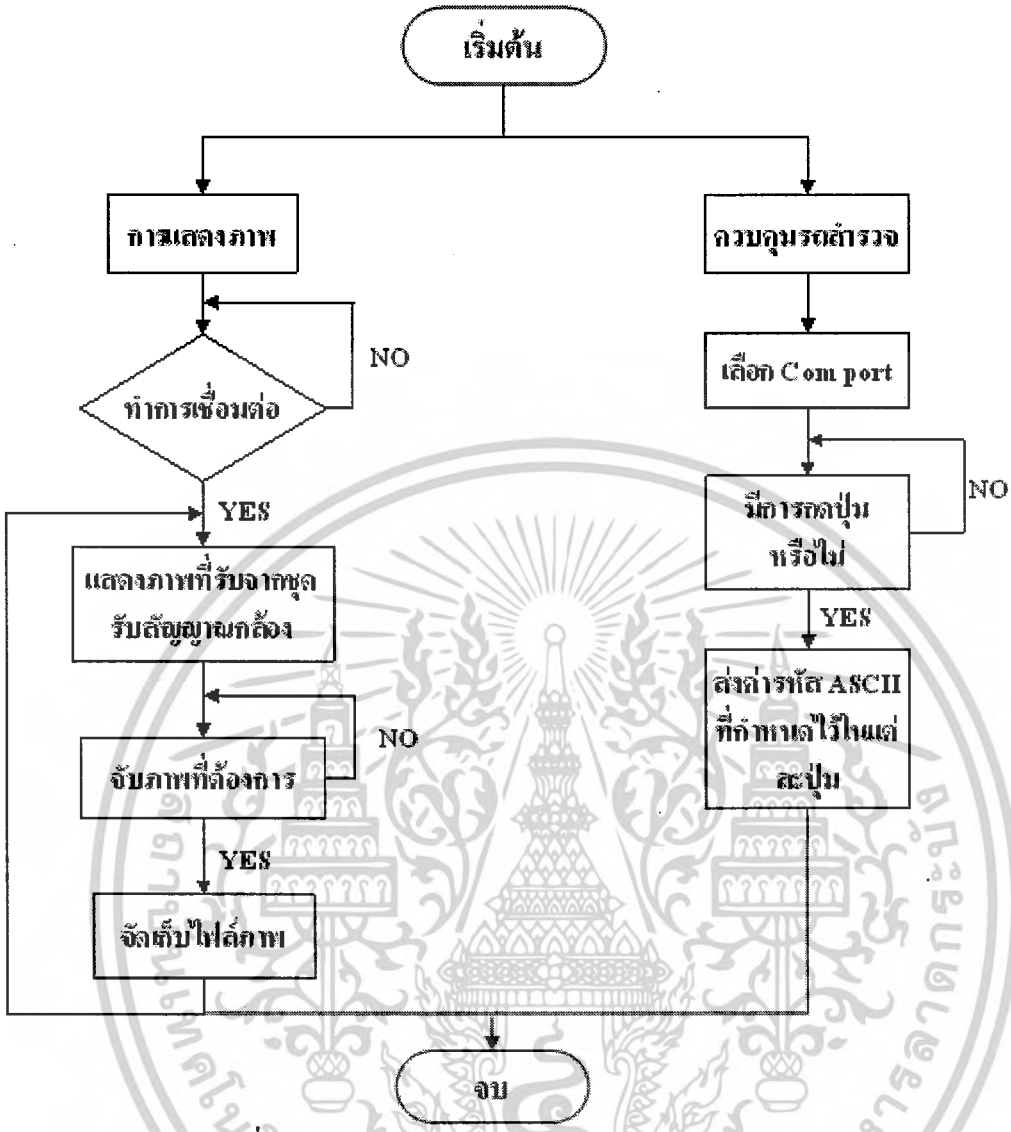
ใช้การเขียนโปรแกรมด้วยภาษา C# ซึ่งเขียนขึ้นจากโปรแกรม Visual C# เป็นโปรแกรมที่ใช้ส่งคำสั่งที่กำหนดไว้ตาม Object ต่างๆ บนโปรแกรม โดยทิศทางของรถจะกำหนดให้มี 6 ทิศทาง ได้แก่ เดินหน้า ถอยหลัง เลี้ยวซ้าย เลี้ยวขวา ถอยหลังไปทางซ้าย และถอยหลังไปทางขวา ส่วนสำหรับกล้องมี 4 ทิศทาง ได้แก่ หมุนซ้าย หมุนขวา หมุนขึ้น และหมุนลง นอกจากนั้นยังมีปุ่มเพื่อเปิดปิดไฟเมื่อรถสำรวจอยู่ในที่มืด เพื่อให้เห็นภาพได้ชัดเจน



รูปที่ 3.3 หน้าต่างโปรแกรมที่ใช้ควบคุมจากคอมพิวเตอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

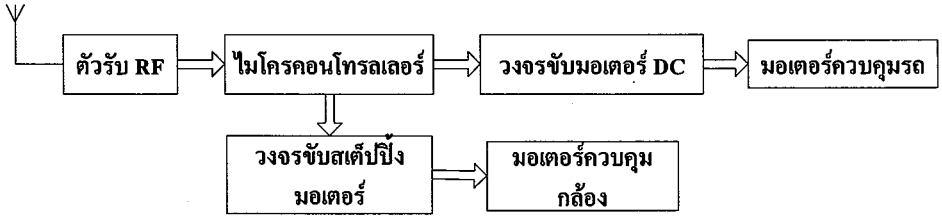
3.1.2 Flow Chart การทำงานของโปรแกรม



รูปที่ 3.4 โฟล์วชาร์ตการทำงานของโปรแกรมควบคุม

3.2 ส่วนของภาครับสัญญาณ

ในส่วนนี้จะทำหน้าที่ถอดรหัสสัญญาณที่รับมา แล้วนำไปควบคุมให้มอเตอร์และกล้องหมุนตามที่เราต้องการ ในส่วนของภาครับจะมีบล็อกไดอะแกรมเป็นดังนี้



รูปที่ 3.5 บล็อกไดอะแกรมของภาครับ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

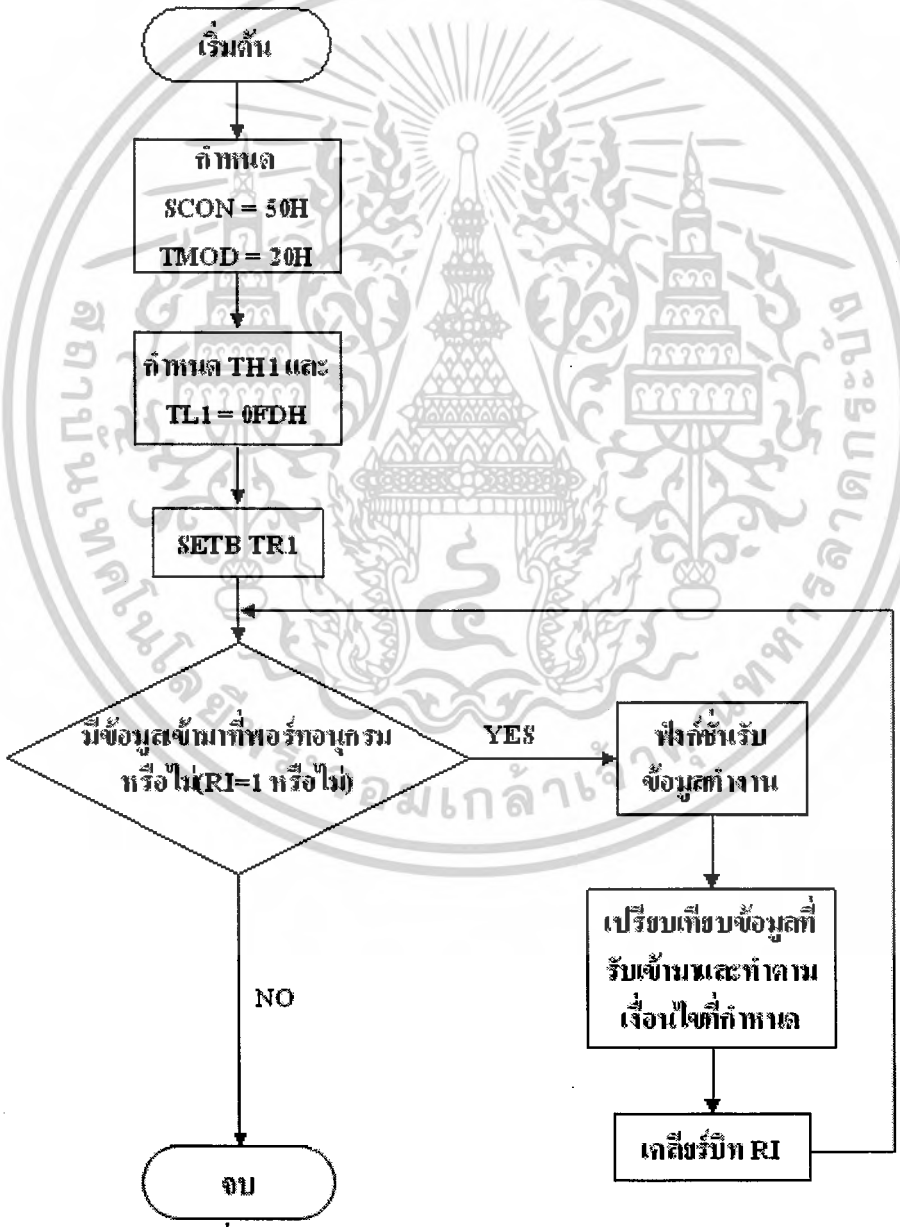
จากบล็อกไดอะแกรมในส่วนของภาครับจะแบ่งเป็นส่วนต่างๆดังนี้

- ตัวรับ RF
- ไมโครคอนโทรลเลอร์
- วงจรขับมอเตอร์

3.2.1 ไมโครคอนโทรลเลอร์

ไมโครคอนโทรลเลอร์จะทำหน้าที่รับสัญญาณที่ได้จากเครื่องรับ RF Wireless มาประมวลผลเพื่อทำตามคำสั่งที่เราต้องการผ่านทางพอร์ทอนุกรม โดยสัญญาณที่เข้ามาจะเป็นตัวกำหนดการจ่ายแรงดันในรูปลักษณะ 0, 5 โวลต์ต่างกัน ที่จะทำให้มอเตอร์ทำงานเพื่อควบคุมบังคับทิศทางการเคลื่อนที่ของรถสำรวจและกล้อง จากนั้นก็จะทำการส่งแรงดันให้กับวงจรขับมอเตอร์ต่อไป

3.2.2 Flow Chart การทำงานของภาครับ



รูปที่ 3.6 โฟล์วชาร์ตการทำงานของภาครับ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.2.3 วงจรขับมอเตอร์

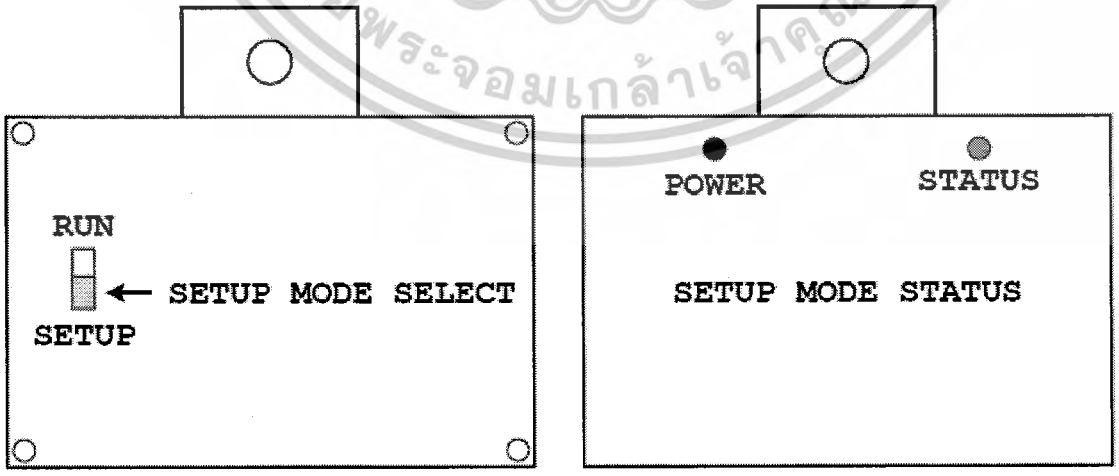
ในที่นี้เราใช้วงจร H-bridge เป็นวงจรขับสำหรับ DC motor โดยเป็นมอเตอร์กระแสตรงขนาด 12 โวลต์ จำนวน 2 ตัวซึ่งใช้ในการควบคุมการเคลื่อนที่ของรถและใช้ไอซี ULN2003 เป็นวงจรขับสำหรับสเต็ปปิ้งมอเตอร์ที่ใช้ในการควบคุมการหมุนของกล้อง เนื่องจากกล้องจำเป็นต้องมีการเคลื่อนที่ด้วยระยะที่ละเอียดกว่าการเคลื่อนที่ของรถ จึงเลือกที่จะใช้สเต็ปปิ้งมอเตอร์

3.3 การติดตั้งและ กำหนดค่าต่างๆ ของเครื่อง ET-RF24G V1.0

การติดตั้งและกำหนดค่าต่างๆของเครื่อง ET-RF24G V1.0 นั้นเราต้องทำการกำหนดใน Setup Mode ซึ่งเป็นโหมดสำหรับใช้กำหนดค่า Configurationต่างๆ สำหรับควบคุมการทำงานของเครื่อง ET-RF24G V1.0 ที่จะใช้ในขณะที่เครื่องทำงานอยู่ใน Run Mode โดยในการ Setup ค่า Configuration ต่างๆ นั้นจะกระทำร่วมกับโปรแกรม **"ET_RF24G_V1.EXE"** ของ อีทีที

ขั้นตอนการกำหนดค่าต่างๆของเครื่อง ET-RF24G V1.0

1. เลื่อนสวิตซ์ด้านหลังตัวเครื่องไปที่ SET UP
2. นำเครื่อง ET-RF24G V1.0 9 ต่อเข้ากับสาย RS232 ที่ต่อกับเครื่องคอมพิวเตอร์อยู่
3. สังเกตที่ตัวเครื่อง ET-RF24G V1.0 จะเห็นหลอดไฟแสดงสถานะการทำงาน หรือ LED STATUS ติดสว่างค้างอยู่ตลอดเวลา (แต่เมื่อมีการสั่งอ่านหรือเขียนข้อมูลกับบอร์ด สถานะการทำงานของ LED STATUS กระพริบตามจังหวะการส่งข้อมูล)
4. เปิดโปรแกรม **ET_RF24G_V1.EXE** เพื่อกำหนดค่าต่างๆให้กับเครื่อง ET_RF24G_V1เพื่อสำหรับการทำงานใน RUN MODE โดยก่อนการใช้งานควรจะกำหนดค่าให้ถูกต้องก่อน
5. หลังจากกำหนดค่าต่างๆเสร็จแล้วให้ทำการเปลี่ยนโหมดการทำงาน เป็น RUN MODE พร้อมกับการปิดไฟที่จ่ายให้กับตัวเครื่องชั่วขณะหนึ่ง จากนั้นค่อยจ่ายไฟให้กับตัวเครื่องใหม่ ก็จะสามารถทำงานได้ตามค่าที่เรากำหนดไว้
6. ค่าที่เราทำการกำหนดให้กับตัวเครื่องนั้นจาถูกเก็บไว้กับตัวเครื่องอย่างถาวร ถึงแม้ว่าจะไม่มีการจ่ายไฟให้กับตัวเครื่อง ดังนั้นหากต้องการเปลี่ยนแปลงค่าใหม่ให้ทำตามขั้นตอนข้างต้น



รูปที่ 3.7 การเลือกโหมดการทำ งาน สำ หรับกำ หนดค่า Configuration (Setup Mode) ของเครื่อง ET-RF24G V1.0

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รายละเอียดค่าต่างๆของเครื่อง ET_RF24G_V1

• **User RS232 Baud rate** ใช้สำหรับกำหนดค่าความเร็วในการรับส่งข้อมูลทางด้าน RS232 ของตัวเครื่อง ในขณะที่ทำงานอยู่ใน Run Mode ซึ่ง กำหนดให้ใช้งานที่ 9600 BPS

• **RF Data Rate** ใช้สำหรับกำหนดความเร็วในการรับส่งข้อมูลทางด้าน RF ของ ET-RF24G V1.0 โดยค่า RF Data Rate ได้กำหนดค่าไว้ คือ 250 Kbps

• **RF Operation Mode** ใช้สำหรับกำหนดโหมดการทำงานของ ET-RF24G V1.0 ซึ่งสามารถกำหนดหน้าที่การทำงานได้ 3 แบบ ด้วยกันแต่ที่ติดตั้งใช้งานสำหรับโหมดนี้นั้น ใช้งานอยู่ 2 แบบ คือ

- RF Receive Only คือ การกำหนดให้ ET-RF24G V1.0 ทำหน้าที่เป็นฝ่ายรอรับข้อมูลทางด้านRF เพื่อเปลี่ยนเป็นข้อมูลแบบ RS232 และส่งออกไปทางด้านขา TX ของ RS232 ตลอดเวลา

- RF Transmit Only คือ การกำหนดให้ ET-RF24G V1.0 ทำหน้าที่เป็นฝ่ายรอรับข้อมูลทางด้าน RS232 จากขา RX เพื่อเปลี่ยนเป็นข้อมูลแบบ GFSK และส่งออกไปทางด้าน RF ตลอดเวลา

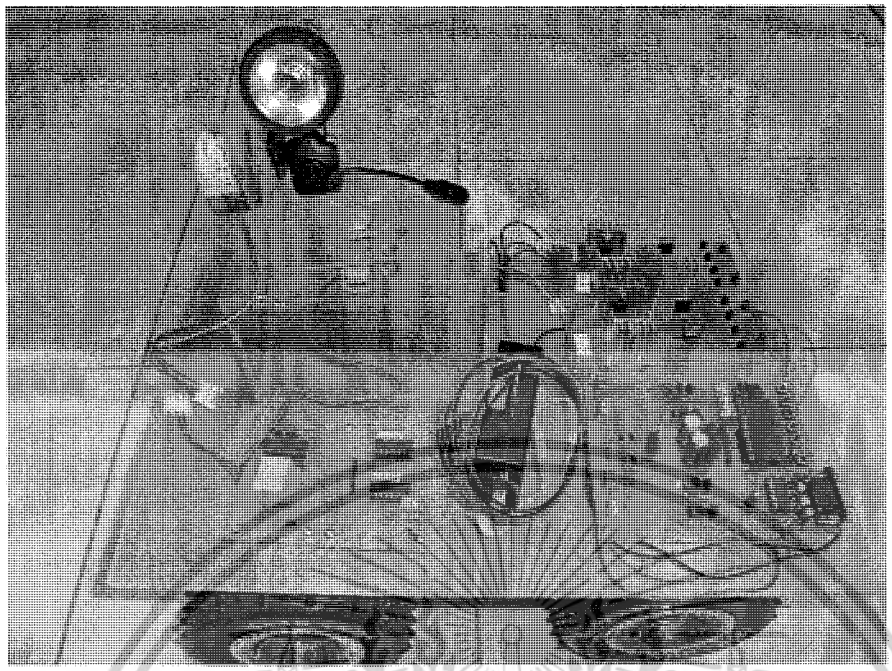
• **RF Power Gain** คือการกำหนดกำลังส่งของวงจร RF Power ที่ใช้ในการส่งข้อมูล โดยตั้งค่าไว้ที่ +0dBm ซึ่ง เป็นค่ากำลังส่งสูงสุด

3.4 การติดตั้งและการใช้งานชุดรับส่งสัญญาณภาพ

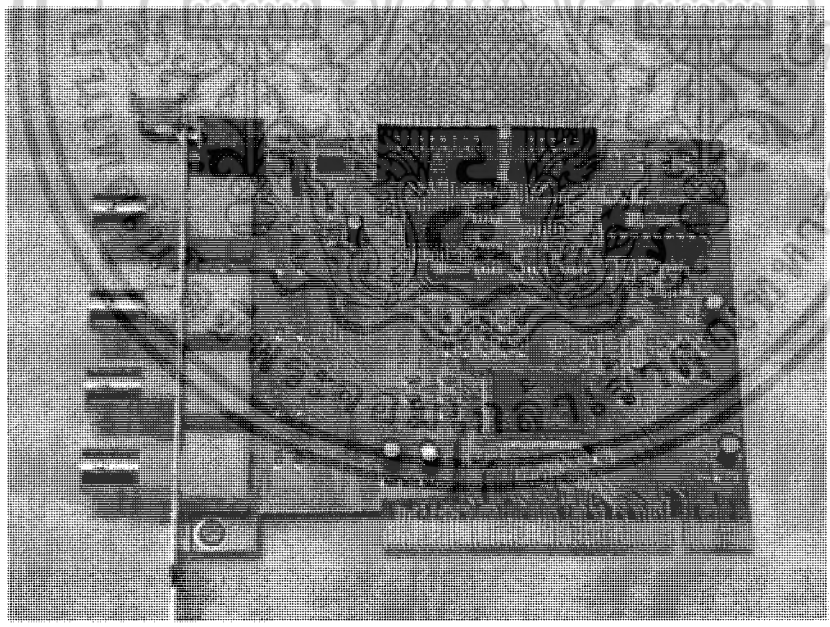
รูปที่ 3.8 บล็อกไดอะแกรมของชุดรับส่งสัญญาณภาพ

กล้องวีดีโอที่ใช้ในโครงงานนี้เป็นชุดกล้องวีดีโอไวเลสขนาดเล็ก พร้อมตัวรับสัญญาณ โดยตัวกล้องใช้พลังงานจากแบตเตอรี่ขนาด 9 โวลต์ ทำการส่งสัญญาณอัตโนมัติ ความถี่ 2.4 GHz รับสัญญาณได้ในระยะ 60-100 เมตร การแสดงผลถ้าหากต้องการให้แสดงผลออกทางหน้าจอโทรทัศน์สามารถที่จะต่อเข้ากับโทรทัศน์ได้เลยทันที แต่ในที่นี้ต้องการแสดงผลบนหน้าจอคอมพิวเตอร์จึงจะต้องใช้การ์ดแคปเจอร์แปลงสัญญาณเพื่อเข้าสู่คอมพิวเตอร์ก่อน ซึ่งต้องมีการติดตั้งไดรเวอร์ของการ์ดแคปเจอร์ด้วยเพื่อที่เราจะได้เรียกเอาภาพขึ้นมาดูได้สะดวกขึ้น ดังรูปข้างล่างจะแสดงลักษณะของการ์ดแคปเจอร์และชุดรับส่งสัญญาณภาพกล้องวีดีโอไร้สาย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

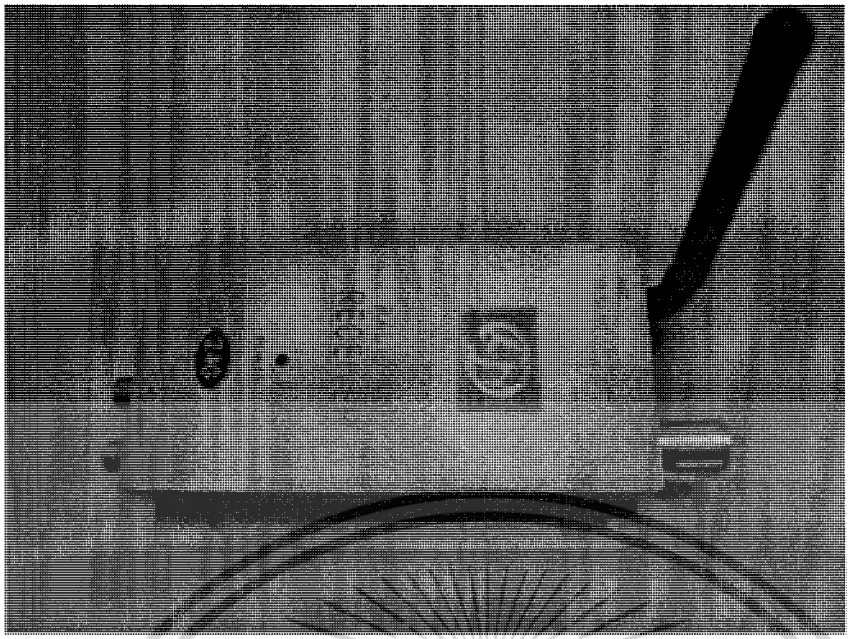


รูปที่ 3.9 รถสำรวจ

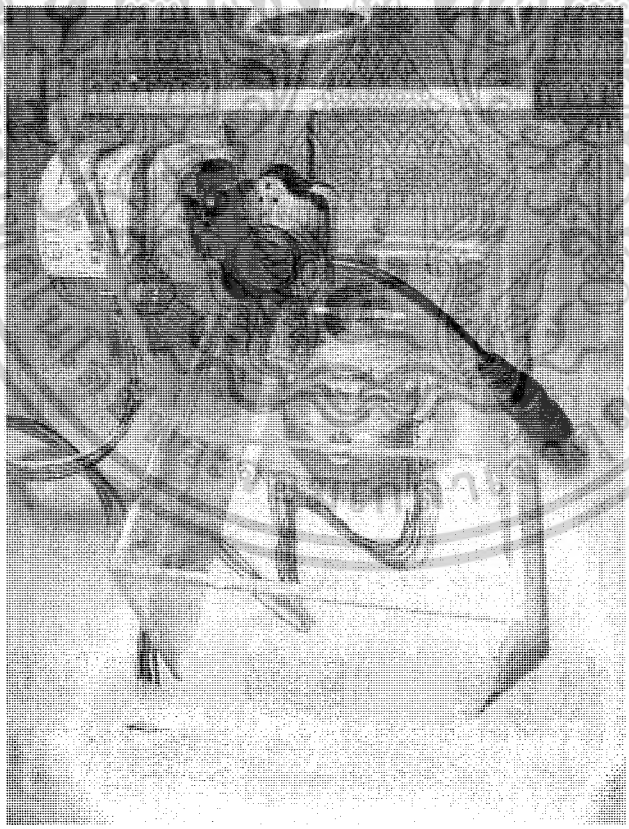


รูปที่ 3.10 ลักษณะของการ์ดแคปเจอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.11 ชุดรับส่งสัญญาณวีดีโอไร้สาย



รูปที่ 3.12 กล้องวีดีโอ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 4
ผลการทดลอง และวิเคราะห์ผลการทดลอง

4.1 การทดลองและผลการทดลองของข้อมูลที่ส่งออกมาจากคอมพิวเตอร์ ในการส่งชุดควบคุมจากภาคส่งไปยังภาครับ

4.1.1 วิธีการทดลอง

เราทำการทดลองโดยการกดปุ่มควบคุมแบบต่างๆเพื่อส่งข้อมูลไปยังไมโครคอนโทรลเลอร์ ซึ่งเป็นข้อมูล 8 บิตส่งออกมาทางพอร์ตอนุกรมที่ขา ซึ่งสัญญาณที่ได้ส่งไปยังโมดูล RF-Wireless เพื่อจะส่งต่อไปยังวงจรภาครับให้ไมโครคอนโทรลเลอร์ทางภาครับทำการประมวลผลต่อไป

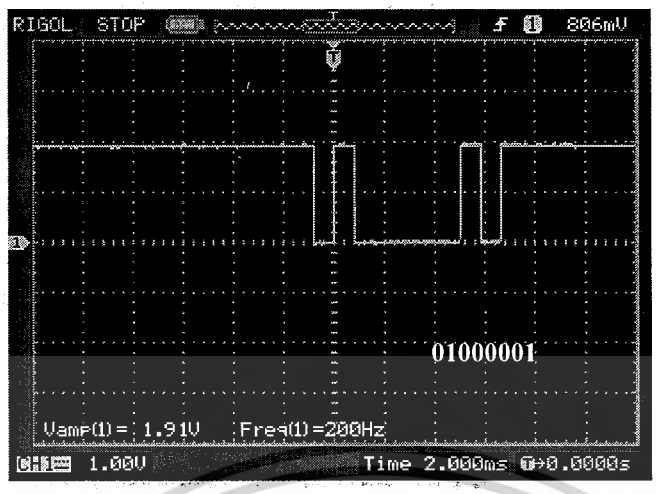
4.1.2 ผลการทดลองวัดสัญญาณข้อมูลที่ส่งออกมาจากคอมพิวเตอร์ในการส่งข้อมูลควบคุมจากภาคส่งไปยังภาครับ

สัญญาณนี้เป็นสัญญาณที่ได้จากการกดปุ่มควบคุมให้เลี้ยวซ้าย โดยสัญญาณที่ออกมามี Start และ Stop Bit อย่างละ 1 บิต และมีข้อมูลอีก 8 บิต

รูปที่ 4.1 สัญญาณข้อมูล จาก คอมพิวเตอร์ภาคส่งเมื่อสั่งให้รถสำรวจเลี้ยวซ้าย

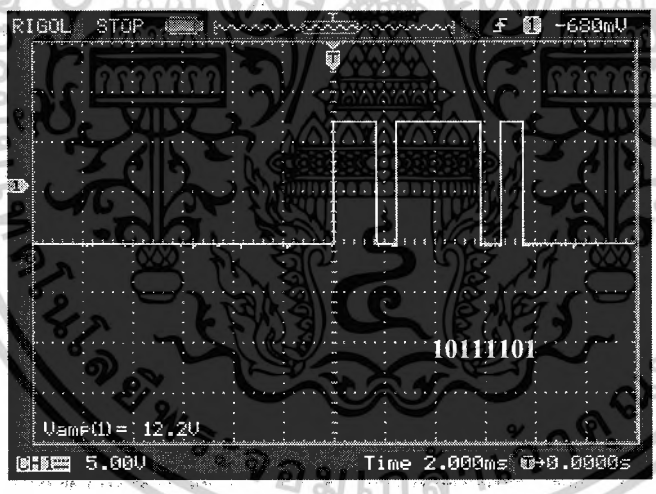
ในฝั่ง ภาครับ รับสัญญาณมาประมวลผล และได้ทำการวัดสัญญาณที่ขา 9 ของ RS-232 ซึ่งเป็นขาที่ส่งข้อมูลไปยัง MCS-51 ซึ่งมีรูปสัญญาณที่ส่งมาดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.2 สัญญาณข้อมูลไบนารีในขณะทำการสั่งให้รถสำรวจ เลี้ยวซ้าย

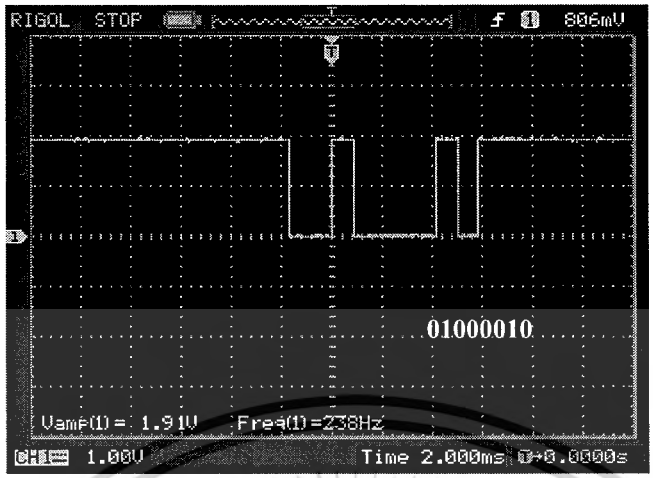
สัญญาณนี้เป็นสัญญาณที่ได้จากการกดปุ่มควบคุมให้เดินหน้า โดยสัญญาณที่ออกมามี Start และ Stop Bit อย่างละ 1 บิต และมีข้อมูลอีก 8 บิต



รูปที่ 4.3 สัญญาณข้อมูล จาก คอมพิวเตอร์ภาคส่งเมื่อสั่งให้รถสำรวจเดินหน้า

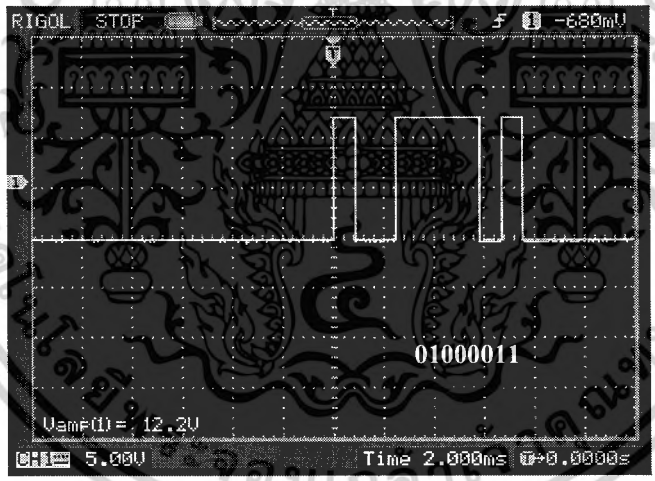
ในฝั่ง ภาครับ รับสัญญาณมาประมวลผล และได้ทำการวัดสัญญาณที่ขา 9 ของ RS-232 ซึ่งเป็นขาที่ส่งข้อมูลไปยัง MCS-51 ซึ่งมีรูปสัญญาณที่ส่งมาดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.4 สัญญาณข้อมูลไบนารีในขณะทำการสั่งให้รถสำรวจ เดินหน้า

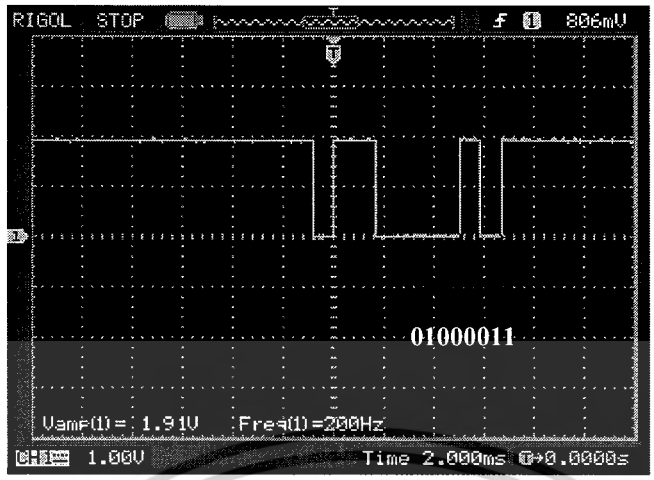
สัญญาณนี้เป็นสัญญาณที่ได้จากการกดปุ่มควบคุมให้เลี้ยวขวา โดยสัญญาณที่ออกมามี Start และ Stop Bit อย่างละ 1 บิต และมีข้อมูลอีก 8 บิต



รูปที่ 4.5 สัญญาณข้อมูล จากคอมพิวเตอร์ภาคส่งเมื่อสั่งให้รถสำรวจเลี้ยวขวา

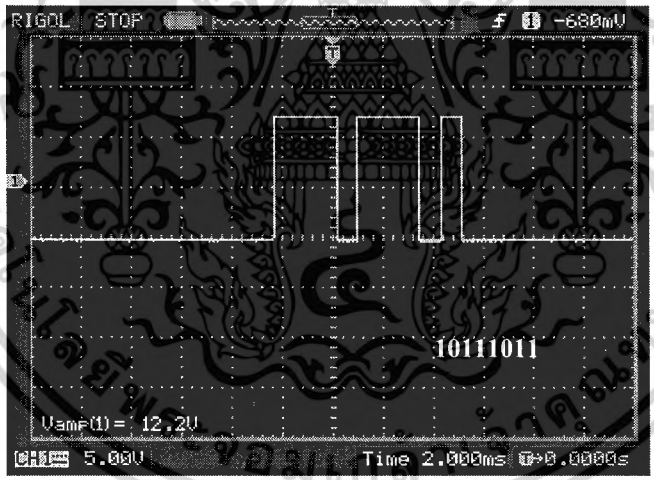
ในฝั่ง ภาครับ รับสัญญาณมาประมวลผล และได้ทำการวัดสัญญาณที่ขา 9 ของ RS-232 ซึ่งเป็นขาที่ส่งข้อมูลไปยัง MCS-51 ซึ่งมีรูปสัญญาณที่ส่งมาดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.6 สัญญาณข้อมูลไบนารีในขณะทำการสั่งให้รถสำรวจ เลี้ยวขวา

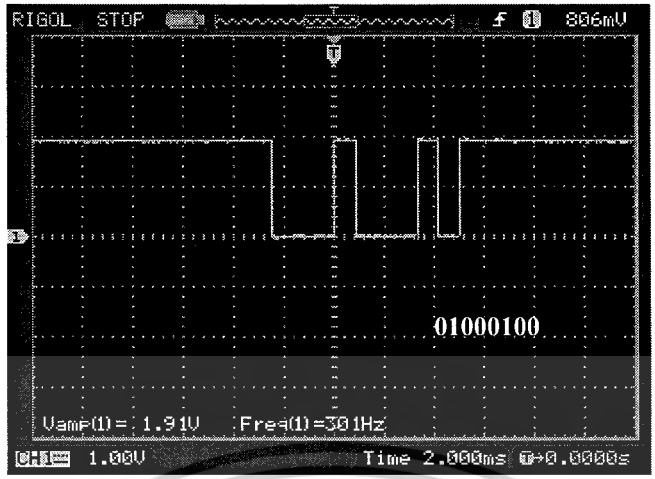
สัญญาณนี้เป็นสัญญาณที่ได้จากการกดปุ่มควบคุมให้ถอยหลัง โดยสัญญาณที่ออกมามี Start และ Stop Bit อย่างละ 1 บิต และมีข้อมูลอีก 8 บิต



รูปที่ 4.7 สัญญาณข้อมูล จากคอมพิวเตอร์ภาคส่งเมื่อสั่งให้รถสำรวจถอยหลังทางซ้าย

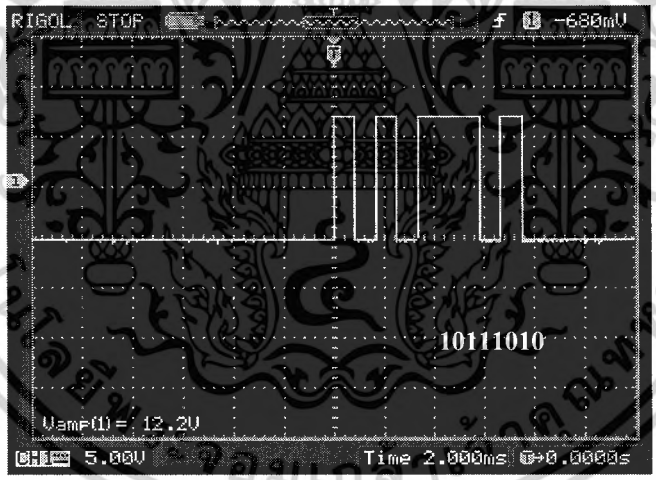
ในฝั่ง ภาครับ รับสัญญาณมาประมวลผล และได้ทำการวัดสัญญาณที่ขา 9 ของ RS-232 ซึ่งเป็นขาที่ส่งข้อมูลไปยัง MCS-51 ซึ่งมีรูปสัญญาณที่ส่งมาดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.8 สัญญาณข้อมูลไบนารีในขณะทำการสั่งให้รถสำรวจ ถอยหลังทางซ้าย

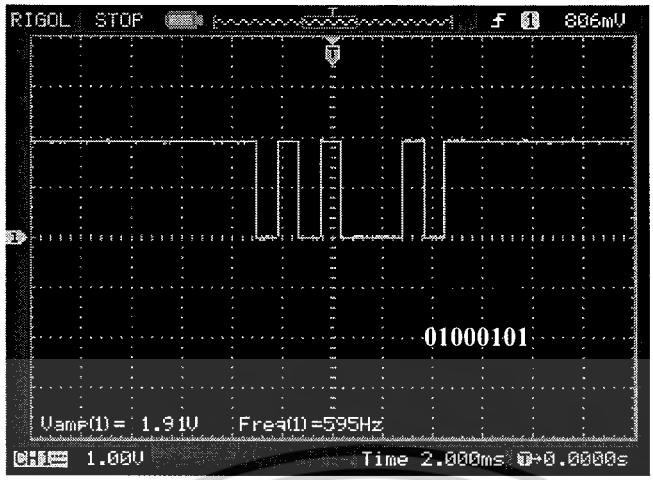
สัญญาณนี้เป็นสัญญาณที่ได้จากการกดปุ่มควบคุมให้เลี้ยวซ้าย โดยสัญญาณที่ออกมามี Start และ Stop Bit อย่างละ 1 บิต และมีข้อมูลอีก 8 บิต



รูปที่ 4.9 สัญญาณข้อมูล จาก คอมพิวเตอร์ภาคส่งเมื่อสั่งให้รถสำรวจถอยหลัง

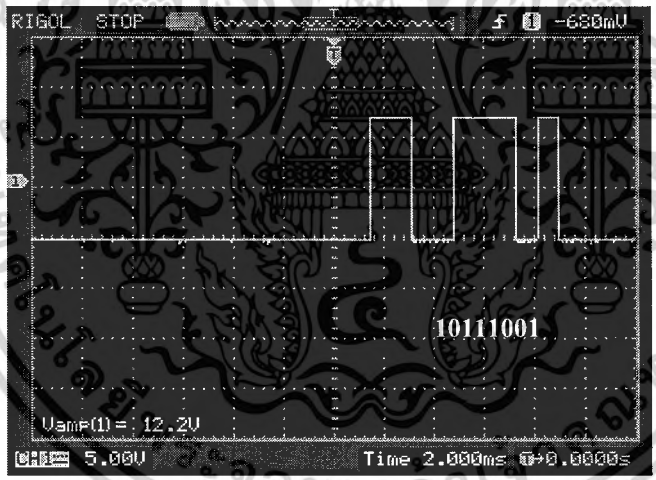
ในฝั่ง ภาครับ รับสัญญาณมาประมวลผล และได้ทำการวัดสัญญาณที่ขา 9 ของ RS-232 ซึ่งเป็นขาที่ส่งข้อมูลไปยัง MCS-51 ซึ่งมีรูปสัญญาณที่ส่งมาดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.10 สัญญาณข้อมูลไบนารีในขณะทำการสั่งให้รถสำรวจ ถอยหลัง

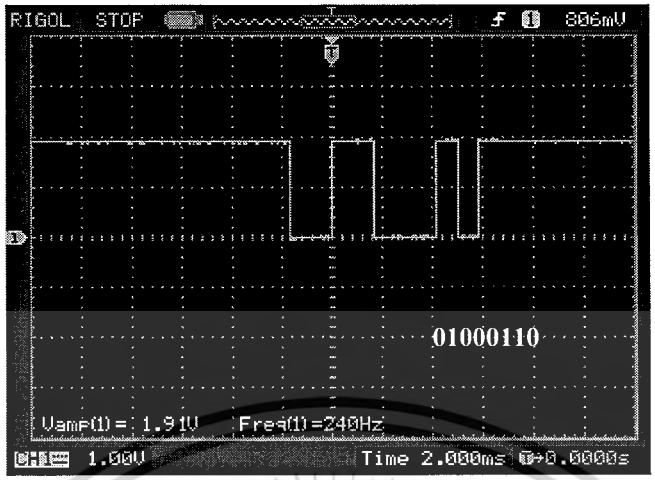
สัญญาณนี้เป็นสัญญาณที่ได้จากการกดปุ่มควบคุมให้ถอยหลังขวา โดยสัญญาณที่ออกมามี Start และ Stop Bit อย่างละ 1 บิต และมีข้อมูลอีก 8 บิต



รูปที่ 4.11 สัญญาณข้อมูล จาก คอมพิวเตอร์ภาคส่งเมื่อสั่งให้รถสำรวจถอยหลังทางขวา

ในฝั่ง ภาครับ รับสัญญาณมาประมวลผล และได้ทำการวัดสัญญาณที่ขา 9 ของ RS-232 ซึ่งเป็นขาที่ส่งข้อมูลไปยัง MCS-51 ซึ่งมีรูปสัญญาณที่ส่งมาดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.12 สัญญาณข้อมูลไบนารีในขณะทำการสั่งให้รถสำรวจ ถอยหลังทางขวา

4.2 การทดลองและผลการทดลองการใช้โมดูล RF Wireless

4.2.1 วิธีการทดลอง

ทำการทดลองการตังค่าและทดลองการใช้งานโมดูล RF Wireless โดยตั้งค่า baud rate เท่ากับ 9600 Bps , RF Data Rate เท่ากับ 250 Kbps , การส่งเป็นแบบ Auto Direction , RF Power Gain เป็น +0 dB ช่องสัญญาณ RF Frequency ที่ช่อง 2 , โมดูลตัวที่ 1 RX ID Code 01 และ TX ID Code 02 , โมดูล RX ID Code 02 และ TX ID Code 01 หลังจากการตั้งค่าให้กับโมดูลแล้วทำการทดลองสื่อสารระหว่างโมดูลโดยใช้คอมพิวเตอร์ 2 ตัวผ่านทาง Hyper Terminal แล้วทำการส่งข้อมูลดูว่าโมดูล RF Wireless สามารถส่งข้อมูลได้จริง

4.2.2 ผลการทดลอง

โมดูล RF Wireless ทั้งสองสามารถส่งข้อมูลติดต่อสื่อสารกันได้ โดยห้องระยะทางที่ทำการติดต่อกันได้มีระยะทางเท่ากับ 35 เมตร ซึ่งเป็นระยะทางระหว่างห้องที่ทำการทดลอง (ทดลองที่ระยะมากกว่านี้ไม่สามารถส่งข้อมูลได้)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.3 ผลการทดลองวัดระยะทางที่รถสำรวจสามารถวิ่งได้จากการรับส่งสัญญาณ จากเครื่องส่งสัญญาณ RF ในบริเวณพื้นที่โล่ง

ทำการวัดระยะทางที่รถสามารถวิ่งได้จากการสั่งงานจากหน้าต่างโปรแกรมโดยส่งสัญญาณผ่านทางเครื่องส่งสัญญาณ RF ให้รถเดินหน้าให้ได้ไกลที่สุดจนกว่ารถจะหยุดวิ่ง แล้วทำการวัดระยะทางนั้นมาบันทึกเป็นผลการทดลอง

ตารางที่ 4.1 แสดงผลการทดลองในการวัดระยะทางที่รถสำรวจสามารถวิ่งได้จากการรับสัญญาณจากเครื่องส่งสัญญาณ RF

ครั้งที่	ระยะทางที่รถสามารถวิ่งได้ (เมตร)
1	78
2	84
3	88.5
4	92
ค่าเฉลี่ย	85.625

4.4 ผลการทดลองลักษณะภาพที่รับได้จากกล้อง CCD ที่ระยะต่างๆในพื้นที่โล่ง

ทำการต่อเครื่องรับสัญญาณ RF ของตัวกล้องเข้ากับคอมพิวเตอร์ จากนั้นนำกล้องไปที่ระยะต่างๆ ดังตารางที่ 4.2 โดยดูภาพที่รับได้จากการส่งสัญญาณภาพจากกล้อง CCD ที่ระยะต่างๆกัน สังเกตความคมชัดของภาพที่ส่งกลับมาแสดงที่จอคอมพิวเตอร์ โดยทำการทดลองที่ตึกภาคโทรคมนาคม

ตารางที่ 4.2 แสดงการผลการทดลองลักษณะภาพจากกล้อง CCD ที่ระยะต่างๆในพื้นที่โล่ง

ระยะทาง (เมตร)	ลักษณะภาพที่รับได้
25	ภาพชัดเจนมาก
50	ภาพชัดเจน
75	ภาพเริ่มขาดหาย
100	ไม่มีภาพ
125	ไม่มีภาพ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.5 ผลการทดลองเมื่อควบคุมรถสำรวจให้ออกไปทำการสำรวจและทำการส่งภาพกลับมาจากกล้องวีดีโอที่เราสามารถรับสัญญาณภาพได้

เมื่อควบคุมให้รถสำรวจออกไปจับภาพซึ่งเราเป็นผู้ควบคุมจากคอมพิวเตอร์จะส่งภาพกลับมาแสดงผลที่หน้าจอในทันที โดยภาพที่ส่งกลับมามีลักษณะดังรูปที่ 4.13

รูปที่ 4.13 ภาพที่ได้แสดงบนหน้าจอคอมพิวเตอร์ของรถสำรวจที่ติดกล้องวีดีโอ

4.6 ผลการทดลองการควบคุมให้สเตปปิ้งมอเตอร์หมุนโดยสั่งการจากไมโครคอนโทรลเลอร์

จากการทดลองต้องการนำสเตปปิ้งมอเตอร์มาติดกับกล้องเพื่อให้กล้องสามารถหมุนได้จากการสั่งการจากไมโครคอนโทรลเลอร์ ดังนั้นจึงทำการทดลองหมุนตัวสเตปปิ้งมอเตอร์ในทิศทางทวนเข็มนาฬิกาและตามเข็มนาฬิกา โดยเราดูว่าสเตปปิ้งมอเตอร์หมุนไปในทิศทางไหนจากการติดดับของหลอด LED ที่จะติดไล่กันไปตามบิท 1 LED หลอดที่ 1 ใช้แสดงแทนการจ่ายไฟไปที่เฟสที่ 1 ของสเตปปิ้งมอเตอร์และตามลำดับ โดยเรากำหนดไว้ในไมโครคอนโทรลเลอร์ จากการกำหนดค่าที่จ่ายออกแปลงเป็นเลขฐานสิบได้คือ 1,2,4,8 เรียงกันไป

4.6.1 ผลการทดลองสั่งให้สเตปปิ้งมอเตอร์หมุนตามเข็มนาฬิกา

หลอด LED จะติดไล่จากซ้ายไปขวาแสดงถึงการหมุนตามเข็มนาฬิกาของสเตปปิ้งมอเตอร์โดยการติดของหลอด LED แสดงถึงแต่ละบิทของค่าที่เรากำนดไว้ในไมโครคอนโทรลเลอร์ และการหมุน 1 สเต็ปของสเตปปิ้งมอเตอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 4.3 ผลการทดลองการติดดับของหลอดไฟ LED แทนการหมุนของสเตปปิ้งมอเตอร์ในทิศทางตามเข็มนาฬิกา

หลอดไฟ / เลขฐาน2	หลอดที่ 1	หลอดที่ 2	หลอดที่ 3	หลอดที่ 4
1000				
0100				
0010				
0001				

4.6.2 ผลการทดลองสั่งให้สเตปปิ้งมอเตอร์หมุนทวนเข็มนาฬิกา

หลอด LED จะติดไล่จากขวาไปซ้ายแสดงถึงการหมุนทวนเข็มนาฬิกาของสเตปปิ้งมอเตอร์ โดยการติดของหลอด LED แสดงถึงบิท 1 และการหมุน 1 ครั้งของสเตปปิ้งมอเตอร์

ตารางที่ 4.4 ผลการทดลองการติดดับของหลอดไฟ LED แทนการหมุนของสเตปปิ้งมอเตอร์ในทิศทางทวนเข็มนาฬิกา

หลอดไฟ / เลขฐาน2	หลอดที่ 1	หลอดที่ 2	หลอดที่ 3	หลอดที่ 4
0001				
0010				
0100				
1000				

ทดลองสั่งการให้สเตปปิ้งมอเตอร์หมุนจนครบ 360 องศาเพื่อดูว่าสเตปปิ้งมอเตอร์หมุนไปกี่ครั้งจนครบรอบ ได้ผลการทดลองคือ สเตปปิ้งมอเตอร์หมุนหนึ่งครั้ง 7.5 องศา

ดังนั้นเมื่อเราทำการติดกล้อง CCD เข้ากับสเตปปิ้งมอเตอร์ กล้อง CCD ก็จะหมุนเป็นสเตปที่เราส่งประมาณ 7.5 องศา

4.7 การทดลองควบคุมการหมุนของสเตปปิ้งมอเตอร์และกล้องวีดีโอ

การทดลองการหมุนของสเตปปิ้งมอเตอร์ซึ่งเรานำมาควบคุมทิศทางการหมุนของกล้องวีดีโอนั้น เมื่อเราทำการสั่งให้สเตปปิ้งมอเตอร์หมุนไปหนึ่งครั้งกล้องวีดีโอที่ติดอยู่กับตัวสเตปปิ้งมอเตอร์ก็จะหมุน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ไปพร้อมๆกัน โดยเราใช้สเตปปิ้งมอเตอร์สองตัวในแนวแกน X และ แกน Y ในทั้งสองแกนนั้นสเตปปิ้งมอเตอร์จะหมุนครั้งละ 7.5 องศา กล้องวีดีโอที่ติดอยู่กับสเตปปิ้งมอเตอร์ก็จะหมุนไป 7.5 องศาเช่นกัน สำหรับตำแหน่งของกล้องนั้นได้ทำการทดลองโดยผลการทดลองแสดงในตารางที่ 4.5 และ 4.6

ตารางที่ 4.5 ผลการทดลองวัดตำแหน่งสเตปปิ้งมอเตอร์ที่ใช้ควบคุมกล้องวีดีโอในแนวแกน X

ครั้งที่	ตำแหน่งของกล้องในแนวแกน X (องศา)	ตำแหน่งของกล้องที่หมุนจริง (องศา)	ค่าความคลาดเคลื่อน (เปอร์เซ็นต์)
1	7.5	7	6.67
2	15	14.5	3.33
3	22.5	22.5	0
4	30	29.7	1
5	37.5	38	1.33
6	45	45	0
7	52.5	52	0.95
8	60	60.5	0.83
9	67.5	67	0.74
10	75	75.5	0.67

ตารางที่ 4.6 ผลการทดลองวัดตำแหน่งสเตปปิ้งมอเตอร์ที่ใช้ควบคุมกล้องวีดีโอในแนวแกน Y

ครั้งที่	ตำแหน่งของกล้องในแนวแกน Y (องศา)	ตำแหน่งของกล้องที่หมุนจริง (องศา)	ค่าความคลาดเคลื่อน (เปอร์เซ็นต์)
1	7.5	7.5	0
2	15	14.5	3.33
3	22.5	22	2.22
4	30	30.5	1.67
5	37.5	37.5	0
6	45	45.5	1.11
7	52.5	52.5	0
8	60	60.	0
9	67.5	67	0.74
10	75	75.5	0.67

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากตารางผลการทดลองจะเห็นว่าในบางครั้งการหมุนของสเตปปิ้งที่ควบคุมกล้องวีดีโอมีความคลาดเคลื่อนไปบ้าง แต่ความคลาดเคลื่อนนั้นก็เป็นองศาที่น้อยมาก ไม่มีผลกระทบกับการมองดูภาพมากนัก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 5
สรุปและวิจารณ์

5.1 บทสรุป

จากการทดลองและศึกษาวงจรขับมอเตอร์และวงจรควบคุมทิศทางของรถโดยใช้ไมโครคอนโทรลเลอร์เพื่อควบคุมให้รถสามารถขับเคลื่อนไปในทิศทางที่ต้องการจากการรับคำสั่งจากคอมพิวเตอร์ที่ผู้ใช้งานเป็นผู้สั่งการ ทำให้ได้เรียนรู้ในเรื่องดังต่อไปนี้

1. สามารถเรียนรู้การเขียนโปรแกรมไมโครคอนโทรลเลอร์โดยใช้ภาษา C# ในการควบคุมการสั่งการทิศทางของรถ
2. สามารถใช้อุปกรณ์อิเล็กทรอนิกส์ในการนำมาสร้างเป็นวงจรขับมอเตอร์และวงจรรับค่าจากไมโครคอนโทรลเลอร์ตระกูล MCS-51
3. ใช้การรับส่งคำสั่งในการควบคุมรถโดยผ่านอุปกรณ์ ET-RF24G V1.0 ซึ่งเป็นการรับส่งข้อมูลแบบไร้สาย

5.2 วิจารณ์

1. เมื่อเริ่มต้นทำการทดลองได้ทำการศึกษาวงจรขับเคลื่อนรถก่อน ซึ่งการหาวงจรที่เหมาะสมกับการใช้งานนั้นค่อนข้างยากเนื่องจากต้องการใช้ประโยชน์จากรถในการสำรวจ ดังนั้นความเร็วของรถจึงมีผลอย่างมาก หากเร็วเกินไปก็จะทำให้ไม่สามารถจับภาพที่ต้องการได้ และหากว่าช้าเกินไปก็จะเป็นอุปสรรคในการทำงานเช่นกัน

2. น้ำหนักของอุปกรณ์ที่นำมาวางบนรถนั้นก็มีผลต่อการขับเคลื่อนของรถเช่นกัน เนื่องจากเมื่อน้ำหนักของอุปกรณ์มากเกินไปจะทำให้แรงส่งของรถที่มาจากแบตเตอรี่เพียงอย่างเดียวไม่เพียงพอต่อการขับเคลื่อนรถ ทำให้รถขับเคลื่อนได้ช้าลง

3. เนื่องจากผู้ทำการทดลองนั้นขาดประสบการณ์เกี่ยวกับรถบังคับทำให้ในตอนเริ่มต้นของการปฏิบัติงานค่อนข้างลำบากในการหาจุดเริ่มต้น

4. เมื่อทำการทดลองวงจรแล้วในตอนแรกรถสำรวจสามารถเลี้ยวทางขวาเพียงด้านเดียว เมื่อตรวจสอบโดยละเอียด จึงพบว่าอุปกรณ์ในแผงวงจรบางชิ้นเสีย แสดงว่าอุปกรณ์อิเลกทรอนิกส์บางชิ้นนเสียได้ง่ายถ้ามีความร้อนสูงๆ จากการทำงานของรถสำรวจเป็นเวลานาน

5. การส่งสัญญาณของเครื่อง ET-RF24G V1.0 นั้นสามารถทำได้ในระยะทางที่ค่อนข้างจำกัดทำให้การส่งรถสำรวจไปในระยะทางไกลๆและมีสิ่งกีดขวางหนาๆไม่สามารถทำได้

5.3 แนวทางการพัฒนา

1. การส่งสัญญาณคำสั่งจากคอมพิวเตอร์ไปยังรถนั้นควรให้การส่งสัญญาณทำได้ไกลขึ้น
2. แบตเตอรี่ที่ใช้ควรมีน้ำหนักที่เบาลงเพื่อให้การเคลื่อนที่ของรถนั้นทำได้เร็วขึ้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3. เมื่อรถสำรวจวิ่งออกไปเรื่อยๆ การรับภาพสัญญาณจะเริ่มไม่ค่อยชัดเจน และจะไม่สามารถจับสัญญาณภาพได้ในที่สุด จึงอาจจะพัฒนาโดยการปรับปรุงด้านการรับสัญญาณภาพที่ส่งมาจากระยะทางให้ได้ไกลขึ้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กิตติกรรมประกาศ

ปริญญานิพนธ์ฉบับนี้สำเร็จลุล่วงไปได้ด้วยความเมตตากรุณาของท่านอาจารย์ที่ปรึกษา ผศ.ดร. พรชัย ทรัพย์นิธิ ที่กรุณาให้คำปรึกษาในการจัดทำโครงงานนี้ด้วยความดียิ่ง ศิษย์ระลึกถึงเสมอสำหรับความเมตตากรุณาของอาจารย์

ขอขอบพระคุณท่านอาจารย์ภาควิชาวิศวกรรมโทรคมนาคมทุกท่าน ที่ประสิทธิ์ประสาทวิชาความรู้ให้ศิษย์ด้วยความทุ่มเท ความรักและคามปรารถนาดีเสมอมา

ขอขอบคุณเพื่อนๆ ที่ให้ความช่วยเหลือทั้งอุปกรณ์ หนังสือ ตลอดจนคำแนะนำต่างๆ และที่ขาดไม่ได้คือกำลังใจและมิตรภาพที่เรามีให้กันเสมอมาเพื่อเป็นแรงผลักดันซึ่งกันและกัน

ขอขอบพระคุณกำลังใจอันสำคัญที่สุดคือ ครอบครัวของพวกเรา ที่คอยสนับสนุนพวกเราเสมอมา จนพวกเราสามารถทำปริญญานิพนธ์ฉบับนี้สำเร็จลุล่วงได้

สุดท้ายนี้พวกเราขอขอบคุณซึ่งกันและกันสำหรับความร่วมมือที่ทำงานร่วมกันจนเอาชนะอุปสรรคต่างๆ

ความดีงามและคุณประโยชน์ที่ปริญญานิพนธ์ฉบับนี้พึงมี พวกเราขอมอบให้แด่ท่านผู้มีพระคุณทุกท่าน

คณะผู้จัดทำ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เอกสารอ้างอิง

[1] สุธี พงศาสกุลชัย, หทัยชนก งามอินทร, "คัมภีร์ Visual C# 2005", กรุงเทพ: เคทีพี คอมพ์ แอนด์ คอนซัลท์, 2549.

[2] อุดม รานอก, "ภาษา C สำหรับงานควบคุมไมโครคอนโทรลเลอร์ MCS-51", นนทบุรี: ไอดีซี อินโฟ ดิสทริบิวเตอร์ เซ็นเตอร์, 2548.

[3] วรลักษณ์ อินทร์มี, วิทยา กลีบเมฆ, " รถสำรวจเคลื่อนที่ควบคุมโดยคอมพิวเตอร์ ", ปริญญานิพนธ์ คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง, 2546.

[4] ปิยรัตน์ ม่วงพานิช, "รถสำรวจเคลื่อนที่ควบคุมโดยคอมพิวเตอร์", ปริญญานิพนธ์ คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง, 2542.

[5] ฐานันดร์ ยันตะกนก, อนรรถพงษ์ ทองทัต, "รถสำรวจควบคุมแบบไร้สายโดยคอมพิวเตอร์", ปริญญานิพนธ์ คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง, 2546.

[6] ธีระศักดิ์ อัครปทุม, บัณฑิตย์ ฉิมรัมย์, "ระบบรักษาความปลอดภัยโดยใช้กล้องวงจรปิด", ปริญญานิพนธ์ คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง, 2550.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ส่วนของโปรแกรม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ซอร์สโค้ด โปรแกรมหลัก

```
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Data.SqlClient;
using System.Drawing;
using System.Text;
using System.Windows.Forms;
using Microsoft.VisualBasic;
using System.Diagnostics;
using TheGeniuz.Capture;

namespace TEST
{
   public partial class Form1 : Form
   {
      string date1;
      string starttime1;
      string endtime1;
      string name1;

      int device_index = -1;
      string txt_com = "";
      private Capture capture = null;
      private Filters filters = new Filters();
      public Form1()
      {
         InitializeComponent();
         #if DEBUG
         capture = new Capture(filters.VideoInputDevices[0],
filters.AudioInputDevices[0]);
         #endif
         try { updateMenu(); }
         catch { }
      }
      private void updateMenu()
      {
         Filter f;
         Control oldPreviewWindow = null;

         if (capture != null)
         {
            oldPreviewWindow = capture.PreviewWindow;
            capture.PreviewWindow = null;
         }


         Filter videoDevice = null;
         if (capture != null)
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
            videoDevice = capture.VideoDevice;
        }

        Form2 main = new Form2();

        SqlConnection Conn;
        SqlTransaction tr;
        StringBuilder sb = new StringBuilder();

        private void button1_Click(object sender, EventArgs e)
        {
            string pname = comboBox1.Text
            port.PortName = pname;
            port.BaudRate = 9600;
            try
            {
                port.Open();
                MessageBox.Show(String.Format("{0} Activated!",pname),
"Success",MessageBoxButtons.OK,MessageBoxIcon.Information);
                button1.Enabled = false;
                button4.Enabled = true;
                button5.Enabled = true;
                button6.Enabled = true;
                button7.Enabled = true;
                button8.Enabled = true;
                button9.Enabled = true;
                button10.Enabled = true;
                button11.Enabled = true;
                button12.Enabled = true;
                button13.Enabled = true;
                button14.Enabled = true;
                button15.Enabled = true;
                button16.Enabled = true;
            }
            catch
            {
                MessageBox.Show(String.Format("Error opening {0}",pname),
"Error",MessageBoxButtons.OK,MessageBoxIcon.Error);
            }
        }

        private void SendData(string txtsend)
        {
            port.Write(txtsend);
        }

        private void timer1_Tick(object sender, EventArgs e)
        {
            if (port.IsOpen)
            {
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
            SendData(txt_com);
        }
        else
            timer1.Enabled = false;
    }

    private void button2_Click(object sender, EventArgs e)
    {

        try
        {

            if (capture.PreviewWindow == null)
            {
                capture.PreviewWindow = panelVideo;
                button3.Enabled = true;
            }
            else
            {
                capture.PreviewWindow = null;

            }

        }
        catch (Exception ex)
        {
            MessageBox.Show("ไม่สามารถ Preview ได้เนื่องจาก.\n\n" + ex.Message + "\n\n" +
ex.ToString());
        }

    }

    private void button3_Click(object sender, EventArgs e)
    {
        try
        {
            timer1.Enabled = false;
            button1.Enabled = true;
            capture.PreviewWindow = null;
        }
        catch
        { }
    }

    private void button4_Click(object sender, EventArgs e)
    {
        button1.Enabled = true;
        button4.Enabled = false;
        button5.Enabled = false;
        button6.Enabled = false;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    button7.Enabled = false;
    button8.Enabled = false;
    button9.Enabled = false;
    button10.Enabled = false;
    button11.Enabled = false;
    button12.Enabled = false;
    button13.Enabled = false;
    button14.Enabled = false;
    button15.Enabled = false;
    button16.Enabled = false;
    try
    {
       port.Close();
    }
    catch { }
}

private void Form1_Load(object sender, EventArgs e)
{
    button3.Enabled = false;
    button4.Enabled = false;
    button5.Enabled = false;
    button6.Enabled = false;
    button7.Enabled = false;
    button8.Enabled = false;
    button9.Enabled = false;
    button10.Enabled = false;
    button11.Enabled = false;
    button12.Enabled = false;
    button13.Enabled = false;
    button14.Enabled = false;
    button15.Enabled = false;
    button16.Enabled = false;
    getConnnect();

}

private void button8_Click(object sender, EventArgs e)
{
    txt_com = "";
    timer1.Enabled = false;
}

private void button6_Click(object sender, EventArgs e)
{
    txt_com = "A";
    timer1.Enabled = true;
}

private void button5_Click(object sender, EventArgs e)
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
{
   txt_com = "B";
   timer1.Enabled = true;
}

private void button7_Click(object sender, EventArgs e)
{
   txt_com = "C";
   timer1.Enabled = true;
}

private void button9_Click(object sender, EventArgs e)
{
   txt_com = "D";
   timer1.Enabled = true;
}

private void button10_Click(object sender, EventArgs e)
{
   txt_com = "E";
   timer1.Enabled = true;
}

private void button11_Click(object sender, EventArgs e)
{
   txt_com = "F";
   timer1.Enabled = true;
}

private void button12_Click(object sender, EventArgs e)
{
   txt_com = "";
   timer1.Enabled = false;
}

private void button13_Click(object sender, EventArgs e)
{
   txt_com = "G";
   timer1.Enabled = true;
}

private void button14_Click(object sender, EventArgs e)
{
   txt_com = "H";
   timer1.Enabled = true;
}

private void button16_Click(object sender, EventArgs e)
{
   txt_com = "I";
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        timer1.Enabled = true;
    }

    private void button15_Click(object sender, EventArgs e)
    {
        txt_com = "J";
        timer1.Enabled = true;
    }

    private void cap_but_Click(object sender, EventArgs e)
    {

        try
        {

            if (cap_but.Text == "CAPTURE")
            {
                string datex = System.DateTime.Now.ToShortDateString().Replace("/",
"");
                date1 = System.DateTime.Now.ToLongDateString();
                string timex = System.DateTime.Now.ToShortTimeString().Replace(":",
"");
                starttime1 = System.DateTime.Now.ToLongTimeString();
                if (capture == null)
                    throw new ApplicationException("ไม่มีกล้องอยู่ในระบบ");
                if (!capture.Cued)
                {
                    capture.Filename = "c:\\d" + datex + timex + ".wmv";
                    name1 = capture.Filename;
                }
                capture.Start();
                cap_but.Text = "STOP";
            }
            else
            {
                if (capture == null)
                    throw new ApplicationException("ไม่มีกล้องอยู่ในระบบ");
                capture.Stop();
                endtime1 = System.DateTime.Now.ToLongTimeString();
                cap_but.Text = "CAPTURE";

                string a = name1;
                string b = starttime1;
                string c = endtime1;
                string d = date1;

                functionsave(name1,starttime1,endtime1,date1);

            }
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        }
        catch (Exception ex)
        {
            MessageBox.Show(ex.Message + "\n\n" + ex.ToString());
        }
    }

    private void saveFileDialog1_FileOk(object sender, CancelEventArgs e)
    {

    }

    private void getConnnect()
    {
        string strConn;
        strConn = "Data Source=.\\SQLExpress;Initial Catalog=noon;Integrated
Security=True";
        Conn = new SqlConnection();
        if (Conn.State == ConnectionState.Open)
        {
            Conn.Close();
        }

        Conn.ConnectionString = strConn;
        Conn.Open();
    }

    private void functionsave(string name,string starttime,string endtime,string date)
    {
        SqlCommand com = new SqlCommand();

            try
            {
                sb.Remove(0, sb.Length);
                sb.Append("INSERT INTO lognoon ( name, starttime, endtime, date)");
                sb.Append(" VALUES (@name, @starttime, @endtime, @date)");
                string sqlAdd;
                sqlAdd = sb.ToString();

                com.CommandText = sqlAdd;
                com.CommandType = CommandType.Text;
                com.Connection = Conn;
                com.Transaction = tr;
                com.Parameters.Clear();
                com.Parameters.Add("@name", SqlDbType.NChar).Value = name;
                com.Parameters.Add("@starttime", SqlDbType.NChar).Value =
starttime;
                com.Parameters.Add("@endtime", SqlDbType.NChar).Value = endtime;
                com.Parameters.Add("@date", SqlDbType.NChar).Value = date;
                com.ExecuteNonQuery();
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
            }
            catch
            {
                MessageBox.Show("เกิดปัญหาในการบันทึกข้อมูลลงในฐานข้อมูล");
            }
        }

    public string GetTime()
    {
        string TimeInString = "";
        int hour = DateTime.Now.Hour;
        int min = DateTime.Now.Minute;
        int sec = DateTime.Now.Second;

        TimeInString = (hour < 10) ? "0" + hour.ToString() : hour.ToString();
        TimeInString += ":" + ((min < 10) ? "0" + min.ToString() : min.ToString());
        TimeInString += ":" + ((sec < 10) ? "0" + sec.ToString() : sec.ToString());
        return TimeInString;

    }

    private void BTlog_Click(object sender, EventArgs e)
    {
        main.ShowDialog();
    }

    private void dateTimePicker2_ValueChanged(object sender, EventArgs e)
    {
    }
     private void label1_Click(object sender, EventArgs e)
    {
     }
    private void groupBox1_Enter(object sender, EventArgs e)
    {
     }

    private void button17_Click(object sender, EventArgs e)
    {
        txt_com = "L";
        timer1.Enabled = true;
    }

    private void button18_Click(object sender, EventArgs e)
    {
        txt_com = "K";
        timer1.Enabled = true;
    }
  }
}
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ซอร์สโค้ด หน้าต่างย่อย

```
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Data.SqlClient;
using System.Drawing;
using System.Text;
using System.Windows.Forms;

namespace TEST
{
   public partial class Form2 : Form
   {
      public Form2()
      {
         InitializeComponent();
      }

      SqlConnection Conn;
      SqlCommand com;
      SqlTransaction tr;
      private void Form2_Load(object sender, EventArgs e)
      {
         getConnnect();
         ShowData();
         FormatDgv();

      }

      private void ShowData()
      {
         StringBuilder sb = new StringBuilder();
         sb.Remove(0, sb.Length);
         sb.Append("SELECT name,starttime,endtime,date");
         sb.Append(" FROM lognoon");
         sb.Append(" ORDER BY date");
         string sqlshow = sb.ToString();

         SqlDataReader dr;
         DataTable dt;

         com = new SqlCommand();
         com.CommandType = CommandType.Text;
         com.CommandText = sqlshow;
         com.Connection = Conn;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        dr = com.ExecuteReader();
        if (dr.HasRows)
        {
            dt = new DataTable();
            dt.Load(dr);
            DGVshow.DataSource = dt;
        }
        else
        {
            DGVshow.DataSource = null;
        }
        dr.Close();
    }

    private void FormatDgv()
    {
        if (DGVshow.RowCount > 0)
        {
            DGVshow.Columns[0].HeaderText = "NAME";
            DGVshow.Columns[1].HeaderText = "START";
            DGVshow.Columns[2].HeaderText = "END";
            DGVshow.Columns[3].HeaderText = "DATE";

            DGVshow.Columns[0].Width = 150;
            DGVshow.Columns[1].Width = 75;
            DGVshow.Columns[2].Width = 75;
            DGVshow.Columns[3].Width = 150;

        }
    }

    private void getConnnect()
    {
        string strConn;
        strConn = "Data Source=.\\SQLExpress;Initial Catalog=noon;Integrated
Security=True";
        Conn = new SqlConnection();
        if (Conn.State == ConnectionState.Open)
        {
            Conn.Close();
        }
        Conn.ConnectionString = strConn;
        Conn.Open();
    }
  }
}
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ซอร์สโค้ด ส่วนของ คอนโทรลเลอร์

```
#pragma SMALL
#include <reg51.h>
void delay1();

void init_sio_poll(void);
char getkey(void);
 sbit  P0_3 = P0^3;
sbit P0_4 = P0^4;
sbit P0_5 = P0^5;
sbit P0_6 = P0^6;
sbit P0_7 = P0^7;

 void delay(int time)
{
        int i,j;                                    // For keep counter loop
        for(i=0;i<time;i++)     // Loop delay
        for(j=0;j<100;j++);
}

void delay1(){
      unsigned char i,j,k;
      for(i=0;i<6;i++)
            for(j=0;j<100;j++)
                  for(k=0;k<100;k++);
}


void main()
{

 unsigned char c;
 P1 = 0x0;
 P0_3 = 0;

 init_sio_poll();

 while(1){
    c = getkey();
        if(c == 'A'){
            P0_4 = 1;
            P0_5 = 0;
            P0_6 = 0;
            P0_7 = 0;
           delay(70);
            P0_4 = 1;
            P0_5 = 1;
            P0_6 = 1;
            P0_7 = 1;
       }
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
else if(c == 'B'){

      P0_4 = 1;
      P0_5 = 0;
      P0_6 = 1;
      P0_7 = 0;
     delay(80);
      P0_4 = 1;
      P0_5 = 1;
      P0_6 = 1;
      P0_7 = 1;
  }
else if(c == 'C'){

      P0_4 = 0;
      P0_5 = 0;
      P0_6 = 1;
      P0_7 = 0;
    delay(80);
      P0_4 = 1;
      P0_5 = 1;
      P0_6 = 1;
      P0_7 = 1;
 }
else if(c == 'D'){

    P0_4 = 0;
    P0_5 = 1;
    P0_6 = 0;
    P0_7 = 0;
    delay(80);
    P0_4 = 1;
    P0_5 = 1;
    P0_6 = 1;
    P0_7 = 1;
  }
 else if(c == 'E'){
     P0_4 = 0;
     P0_5 = 1;
     P0_6 = 0;
     P0_7 = 1;
      delay(70);
     P0_4 = 1;
     P0_5 = 1;
     P0_6 = 1;
     P0_7 = 1;
   }
  else if(c == 'F'){
     P0_4 = 0;
     P0_5 = 0;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    P0_6 = 0;
    P0_7 = 1;
   delay(80);
    P0_4 = 1;
    P0_5 = 1;
    P0_6 = 1;
    P0_7 = 1;
  }
 else if(c=='G'){

   P1 = 0x10;
   delay1();
   P1 = 0x80;
   delay1();
   P1 = 0x40;
   delay1();
   P1 = 0x20;
   delay1();
   P1 = 0x00;
  delay(1000);
   }
 else if (c=='H'){

   P1 = 0x80;
   delay1();
   P1 = 0x40;
   delay1();
   P1 = 0x20;
   delay1();
   P1 = 0x10;
   delay1();
   P1 = 0x00;
  delay(1000);
 }
  else if (c=='I'){

   P1 = 0x01;
   delay1();
   P1 = 0x08;
   delay1();
   P1 = 0x04;
   delay1();
   P1 = 0x02;
   delay1();
   P1 = 0x00;
   delay(1000);
   }
  else if (c=='J'){
    P1 = 0x01;
   delay1();
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
            P1 = 0x02;
            delay1();
            P1 = 0x04;
            delay1();
            P1 = 0x08;
            delay1();
            P1 = 0x00;
            delay(1000);
           }

         else if (c=='K'){
             P0_3 = 0;
           }
         else {
             P0_3 = 1;
            }
      }
 }

void init_sio_poll(void)
{

    TR1 =0;
    ET1 =0;
    TMOD = 0x22;
    TH1 =0xFD;
    TR1 =1;
    SCON =0x50;
    TI =1;
}

char getkey(void)
{

  char inChar;
  while(!RI);
  inChar = SBUF;
  RI = 0;
  return inChar;
}
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้