สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

รถบังคับผ่านคอมพิวเตอร์

ROBOT CONTROLLED BY COMPUTER

T055498

โดย

นายกิดากร บัวประหยด

นายมณฑล เศรษฐบุตร

นายเอนก โตสงค์

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต

สาขาวิชาอิเล็กทรอนิกส์

คณะวิศวกรรมศาสตร์

วศ.
ก 669 ร
2546

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2546

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เลขหมู่
เลขทะเบียน 55498
วัน,เดือน,ปี 10 พ.ค. 2548

b 11574212
i

รถบังคับผ่านคอมพิวเตอร์

ROBOT CONTROLLED BY COMPUTER

โดย

นายกิดากร บัวประหรอด 44015229

นายมณฑล เศรษฐบุตร 44015250

นายเอนก โตสงค์ 44015271

อาจารย์ที่ปรึกษา

ผศ.ดร.ยุทธนา คิดใจเดียว

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต

สาขาวิชาอิเล็กทรอนิกส์

คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2546

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รายงานปีการศึกษา 2546

ภาควิชาอิเล็กทรอนิกส์

คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เรื่อง รถบังคับผ่านคอมพิวเตอร์

ผู้จัดทำ

1.นายกิดากร บัวประหรอด

2.นายมณฑล เศรษฐบุตร

3. นายเอนก โตสงค์

..... อาจารย์ที่ปรึกษา

(ผศ.ดร.ยุทธนา คิดใจเดียว)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รถบังคับผ่านคอมพิวเตอร์

ROBOT CONTROLLED BY COMPUTER

1. นายกิดากร บัวประหรอด 3R/2 44015229

2. นายมณฑล เศรษฐบุตร 3R/2 44015250

3. นายเอนก โตสงค์ 3R/2 44015271

โครงงานนี้ได้รับการตรวจสอบแล้ว พร้อมที่จะทำการสอบได้

.....อาจารย์ที่ปรึกษา

(ผศดร.ยุทธนา คิดใจเดียว)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รถบังคับผ่านคอมพิวเตอร์

นายกิดากร บัวประหรอด
นายมณฑล เศรษฐบุตร
นายเอนก โตสงค์
ผศ ดร.ยุทธนา คิดใจเดียว (อาจารย์ที่ปรึกษา)
ภาคเรียนที่ 2 ปีการศึกษา 2546

บทคัดย่อ

ปริญญานิพนธ์ฉบับนี้แสดงการออกแบบ และ สร้างหุ่นยนต์ที่ควบคุมผ่านคอมพิวเตอร์ การทำงานโดยรวมแล้วจะเป็นการศึกษาเกี่ยวกับ การประมวลผลโดยใช้ภาพ, ไมโครคอนโทรลเลอร์, การใช้งานเครื่องรับส่งสัญญาณแบบไร้สาย(wireless) , การควบคุมมอเตอร์ไฟฟ้ากระแสตรง การติดต่อสื่อสารระหว่างอุปกรณ์ภายนอกกับคอมพิวเตอร์ผ่าน พอร์ทขนาน รวมถึงการใช้งานเซนเซอร์ เพื่อตรวจสอบวัตถุ

การเคลื่อนที่ของตัวรถจะใช้ มอเตอร์ไฟฟ้ากระแสตรงเป็นตัวขับเคลื่อน บนรถจะติดกล้องวีดีโอ ไร้สาย เพื่อส่งสัญญาณภาพกลับมายังคอมพิวเตอร์ สำหรับประมวลผล และส่งสัญญาณกลับไปควบคุมการการเคลื่อนที่ของรถ ส่วนตัวรถจะติดเซนเซอร์อินฟราเรดตรวจจับวัตถุ เพื่อป้องกันการชนกับวัตถุไม่ให้เกิดความเสียหายกับตัวรถ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ROBOT CONTROLLED BY COMPUTER

KidaKon Buaprarod
Monthol Satthabut
Anek Tosong
Yutthana KIdjaideaw (Advisor)

2 nd Semester Academic year 2003

Abstract

This thesis presents the design and construct a robot .which is controlled by computer. The project focuses on image processing , microcontroller , wireless transmission , dc motor and communication between I/O module and computer through parallel port and sensor for detection an object.

For moving the robot , it use dc motor .The video camera is fixed with the robot to capture images send to computer .The images are processed by computer and send signal back to control the robot . Lastly , the robot is equipped with sensor to detect object for crash protection .

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กิตติกรรมประกาศ

รายงานนี้สามารถสำเร็จลุล่วงได้ด้วยดีนั้น ทางผู้จัดทำขอขอบคุณอาจารย์ที่ปรึกษา ผศดร.ยุทธนา คิดใจเดียว ที่ให้คำปรึกษาเกี่ยวกับข้อมูลรายละเอียดต่าง ๆ ขั้นตอนในการทำงาน รวมทั้งอาจารย์ในภาควิชา ทุกท่านที่ได้ให้คำแนะนำ ขอขอบคุณรุ่นพี่และเพื่อน ๆ ทุกคนที่ให้คำปรึกษาตลอดจนช่วยเหลือเรื่องข้อมูลและอุปกรณ์รวมทั้งเครื่องมือต่างๆ ขอขอบคุณบิดา มารดา และผู้มีพระคุณทุกท่าน

ผู้จัดทำ
กิดากร บัวประหรอด
มณฑล เศรษฐบุตร
เอนก โตสงค์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ

	หน้า
บทคัดย่อ	I
ABSTRACT	II
กิตติกรรมประกาศ	III
สารบัญ	IV
สารบัญรูป	VI
สารบัญตาราง	VIII
บทที่ 1 บทนำ	1
1.1 ความเป็นมาและโครงสร้างของโครงงาน	1
1.2 ขอบเขตของโครงงาน	1
1.3 ประโยชน์ที่คาดว่าจะได้รับ	1
1.4 ข้อกำหนดของโครงงาน	2
บทที่ 2 ทฤษฎีและหลักการ	3
2.1 การใช้งานไมโครคอนโทรลเลอร์ MCS-51	3
2.1.1 คุณสมบัติทางเทคนิคของไมโครคอนโทรลเลอร์ MCS-51	3
2.1.2 การจัดขาของไมโครคอนโทรลเลอร์ MCS-51	6
2.1.3 ไทเมอร์/เคาน์เตอร์ในไมโครคอนโทรลเลอร์ MCS-51	8
2.1.4 อินเตอร์รัปต์ในไมโครคอนโทรลเลอร์ MCS-51	14
2.2 การสื่อสารไร้สายด้วยคลื่นวิทยุความถี่สูงระยะใกล้	18
2.2.1 หลักการและทฤษฎีของระบบควบคุมระยะไกลแบบไร้สาย	18
2.2.2 ความรู้พื้นฐานเกี่ยวกับอุปกรณ์ส่งคลื่นวิทยุ (Radio Controller)	19
2.2.3 อุปกรณ์ส่งสัญญาณ	20
2.2.4 สัญญาณจากสัญญาณรหัสกว้าง (Pulse Width)	22
2.2.5 อุปกรณ์รับส่งสัญญาณ (Receiver)	22
2.2.6 การสื่อสารด้วยพัลส์	23
2.2.7 วงจร Pulse Width Modulation (PWM)	25
2.2.8 วงจรกำเนิดความถี่วิทยุ (Radio Frequency Oscillator)	25
2.2.9 การทำงานของภาคส่งสัญญาณสื่อสาร	25
2.2.10 การทำงานของภาครับสัญญาณ	26

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หน้า

2.2.11 ปัญหาการทำงานของการสื่อสารไร้สายด้วยคลื่นวิทยุ 26
2.3 การสื่อสารผ่านพอร์ทคอมพิวเตอร์ 26
2.3.1 การสื่อสารแบบขนาน ผ่านทางพรินเตอร์พอร์ต 26
2.3.2 การสื่อสารข้อมูลแบบอนุกรม 29
2.3.2.1 มาตรฐาน EIA RS-232 29
2.3.2.2 รูปแบบของข้อมูลอนุกรมและอัตราบอดในการสื่อสารข้อมูลอนุกรม 32
2.3.2.3 การเชื่อมต่อแบบอนุกรมและ UART 33
2.4 การประมวลผลภาพเชิงตัวเลข (Digital image Processing) 36
2.4.1การทำเทรชโชลด์(Thresholding technique) 38
2.4.1.1 รูปแบบในการทำเทรชโชลด์ 39
2.4.1.2 วิธีการหาค่าเทรชโชลด์ 41
2.4.2 โมเดลสีแบบ RGB 44
บทที่ 3 การออกแบบ
3.1 การออกแบบ วงจรควบคุมรถใช้ MCS-51 46
3.2 วงจรตรวจสอบการชน 47
3.3 ส่วนการขับเคลื่อน 49
3.3 ส่วนหน้าตาโปรแกรมควบคุมจากคอมพิวเตอร์ 50
3.4 ส่วนของโปรแกรมภายในตัวไมโครคอนโทรลเลอร์ 53
3.5 การทำงาน 54
บทที่ 4 การทดลองและผลการทดลอง 55
4.1สัญญาณที่ส่งไปขับดีซีมอเตอร์ 55
4.2 ระยะไกลของการบังคับแบบไร้สาย 57
4.3 เซนเซอร์ 57
4.4 การเคลื่อนที่ตามวัตถุ 58
บทที่ 5 สรุปและวิจารณ์ 62
เอกสารอ้างอิง 63

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป

	หน้า
รูปที่ 2.1 รูปแบบของการเชื่อมต่อของ SPI อย่างง่าย	5
รูปที่ 2.2 การจัดขามาตราฐานของไมโครคอนโทรลเลอร์ MCS-51	8
รูปที่ 2.3 บล็อกไดอะแกรมแสดงหลักการทำงานภาคส่งของระบบคลื่นวิทยุ	18
รูปที่ 2.4 บล็อกไดอะแกรมแสดงหลักการทำงานภาครับของระบบคลื่นวิทยุ	19
รูปที่ 2.5 บล็อกไดอะแกรมแสดงรูปแบบการส่งสัญญาณแบบพัลส์	19
รูปที่ 2.6 แสดงหลักการทำงานของระบบคลื่นวิทยุผ่านทางตัวกลางที่เป็นอากาศ	20
รูปที่ 2.7 การใช้สัญญาณพัลส์กระตุ้นให้อุปกรณ์ควบคุมสวิตซ์ทำงาน	20
รูปที่ 2.8 วงจรกำหนดความกว้างของพัลส์	21
รูปที่ 2.9 วงจรผลิตความถี่	21
รูปที่ 2.10 วงจรส่งคลื่นวิทยุที่มีทั้งวงจรผลิตความถี่และวงจรกำหนดความกว้างสัญญาณ	22
รูปที่ 2.11 วงจรแปลงสัญญาณคลื่นวิทยุ	23
รูปที่ 2.12 สัญญาณพัลส์	23
รูปที่ 2.13 วงจรปรับสัญญาณพัลส์	24
รูปที่ 2.14 วงจรกรองสัญญาณความถี่ต่ำ	25
รูปที่ 2.15 วงจรกำเนิดความถี่วิทยุ	26
รูปที่ 2.16 ขาของพอร์ตพาราแรลแบบ 25 ขา	27
รูปที่ 2.17 แสดงแอดเดรสของพาราแรลพอร์ต และลำดับการส่งข้อมูลของพรินเตอร์	27
รูปที่ 2.18 หน้าที่และทิศทางของขาทั้งหมดของพาราแรลพอร์ต (ทิศทางออกจาก PC)	28
รูปที่ 2.19 อินเตอร์เฟสระหว่างข้อต่อที่พรินเตอร์และข้อต่อที่ PC	28
รูปที่ 2.20 รูปแบบการต่อสัญญาณป้อนกลับ	29
รูปที่ 2.21 การเชื่อมต่อกันระหว่างอุปกรณ์ DTE กับ DCE	31
รูปที่ 2.22 การเชื่อมต่อกันระหว่างอุปกรณ์ DTE กับ DTE	32
รูปที่ 2.239 การส่งข้อมูลแบบอนุกรม	32
รูปที่ 2.24 การแปลงข้อมูลอนุมกรมไปเป็นข้อมูลแบบขนาน	33

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หน้า

รูปที่ 2.25 ข้อมูลอนุกรมขนาด 8 บิต 34
รูปที่ 2.26 การแทนสัญญาณอนาลอกด้วยสัญญาณเชิงตัวเลข 37
รูปที่ 2.27 ฮีสโตแกรมระดับความเข้มของภาพที่มีความเหมาะสมสำหรับ
การทำเทรชโชลด์แบบครอบคลุม 39
รูปที่ 2.28 ฮีสโตแกรมระดับความเข้มของภาพที่เหมาะสมสำหรับการทำ
เทรชโชลด์แบบปรับค่า 40
รูปที่ 2.29 การแบ่งภาพออกเป็นภาพย่อยๆ และหาค่าเทรชโชลด์ในแต่ละภาพย่อย 41
รูปที่ 2.30 แนวความคิดในการคำนวณค่าเทรชโชลด์โดยวิธีพิจารณาจากฮีสโตแกรม 43
รูปที่ 2.31 แสดงรูปลูกบาศก์สีของโมเดลแบบ RGB 44
รูปที่ 3.1 ไมโครคอนโทรลเลอร์(MCS-51) 46
รูปที่ 3.2 การต่อเพื่อใช้งานกับพอร์ทอนุกรม 47
รูปที่ 3.3 วงจรส่ง อินฟราเรด 48
รูปที่ 3.4 วงจรรับอินฟราเรด 48
รูปที่ 3.5 รูปวงจรขับมอเตอร์ 49
รูปที่ 3.6 แสดงรูปหน้าต่างโปรแกรม 50
รูปที่ 3.7 แสดงรูปโปรแกรมสีของ windows 51
รูปที่ 3.8 แสดงรูปหน้าต่างโปรแกรมการปรับค่า 52
รูปที่ 3.9 Flow chart แสดงการทำงานหลักของโปรแกรม 53
รูปที่ 3.10 บล็อกไดอะแกรมการทำงานแบบไร้สาย 54
รูปที่ 4.1 กราฟแสดงลักษณะสัญญาณพัลส์ สั่งเดินหน้า 55
รูปที่ 4.2 กราฟแสดงลักษณะสัญญาณพัลส์ สั่งถอยหลัง 56
รูปที่ 4.3 กราฟแสดงลักษณะสัญญาณพัลส์ เอาท์พุทจากวงจรขับมอเตอร์ สั่งเดินหน้า 56
รูปที่ 4.4 กราฟแสดงสัญญาณพัลส์ เอาท์พุทจากวงจรขับมอเตอร์ สั่งถอยหลัง 57
รูปที่ 4.5 แสดงระยะการตรวจจับสิ่งกีดขวาง 58
รูปที่ 4.6 แสดงการจับภาพวัตถุและส่งสัญญาณให้รถเคลื่อนที่เข้าหา 59
รูปที่ 4.7 แสดงการจับภาพวัตถุและส่งสัญญาณให้รถเลี้ยวซ้าย 59
รูปที่ 4.8 แสดงการจับภาพวัตถุและส่งสัญญาณให้รถเลี้ยวขวา 60
รูปที่ 4.9 แสดงความ เพี้ยนของ white balance ของกล้องเมื่อระยะห่าง
ระหว่างจากรถและวัตถุเปลี่ยนแปลง 60

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญตาราง

	หน้า
ตารางที่ 2.1 แสดงข้อมูลการเชื่อมต่ออนุกรม 9 ขา	30
ตารางที่ 2.2 อัตราบอดเรท	33
ตารางที่ 2.3 อัตราบอดเรททั่วไปที่ใช้ในการโอนย้ายข้อมูลแบบอนุกรม	35
ตารางที่ 3.1 ค่าจากพอร์ท P2 ของไมโครคอนโทรเลอร์	50
ตารางที่ 4.1 แสดงความสามารถในการตรวจสอบสิ่งกีดขวางของวงจรเซนเซอร์	57

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 1
บทนำ

1.1 ความเป็นมาและโครงสร้างของโครงงาน

ในปัจจุบันได้มีการนำเทคโนโลยีด้านหุ่นยนต์มาใช้งานกันอย่างกว้างขวางขึ้น โดยการนำหุ่นยนต์มาทำงานเป็นกิจวัตร หรืองานรูทีน (routine job) ข้อดีประการแรกของการนำหุ่นยนต์มาใช้งานคือหุ่นยนต์มีความแม่นยำกว่ามนุษย์ ประการที่สองคือหุ่นยนต์สามารถทำงานได้นานกว่ามนุษย์หากมีการดูแลรักษาที่ดี

พื้นฐานที่เบื้องต้นของหุ่นยนต์ คือรถบังคับที่ ที่สามารถโปรแกรมการทำงานได้ดังนั้นในโครงงานนี้จึงเลือกศึกษาการบังคับรถผ่านตัวบังคับคือคอมพิวเตอร์และควบคุมการทำงานของตัวรถด้วย ไมโครคอนโทรลเลอร์ MCS 51

1.2 ขอบเขตของโครงงาน

1. เป็นการสื่อสาร หรือเป็นการส่งข้อมูลแบบใช้สาย และไร้สาย ระหว่างด้านส่งคอมพิวเตอร์และด้านรับ (ตัวรถ)
2. ทางด้านการ Control รถใช้ โปรแกรม visual basic เป็นตัว Control จากคอมพิวเตอร์โดยบังคับ การเคลื่อนที่ ของรถ
3. การใช้การประมวลผลโดยใช้ภาพ
4. ขั้นตอนการทำงานการเคลื่อนที่ของรถทั้งหมดควบคุมด้วยไมโครคอนโทรลเลอร์
5. การตรวจสอบการชนโดยเซนเซอร์ อินฟาเรด (IR)

1.3 ประโยชน์ที่คาดว่าจะได้รับ

1. สามารถประยุกต์ใช้วงจรสื่อสาร โดยไร้สาย (Wireless)
2. สามารถใช้งาน ไมโครคอนโทรลเลอร์ เพื่อควบคุมและประมวลผล
3. สามารถเข้าใจหลัก เชื่อมต่อคอมพิวเตอร์ กับอุปกรณ์ภายนอกได้
4. รู้จักการนำสัญญาณภาพเข้ามาร่วมในการประมวลผล เพื่อควบคุมการเคลื่อนที่ของตัวรถ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1.4 ข้อกำหนดของโครงงาน

1. สามารถควบคุมรถได้จากคอมพิวเตอร์ โดยไร้สาย
2. ใช้ไมโครคอนโทรลเลอร์ เพื่อความสามารถในการควบคุม การพัฒนา ตลอดจนการทำงานของรถที่ดีขึ้น
3. มีความละเอียดและแม่นยำในการใช่ภาพประมวลผลพอควร
4. มีการตรวจสอบป้องกันการการวิ่งชนวัตถุ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 2
ทฤษฎีและหลักการ

2.1 การใช้งานไมโครคอนโทรลเลอร์ MCS-51

ไมโครคอนโทรลเลอร์ตระกูล MCS-51 ที่ใช้ในการทำโครงงานนี้จะอ้างอิงไมโครคอนโทรลเลอร์ตระกูล MCS-51 ซึ่งมีหน่วยความจำภายในแบบแฟลช (flash memory) ของ Atmel Corporation มีเบอร์ขึ้นต้นด้วย AT89 เหตุผลที่ใช้ไมโครคอนโทรลเลอร์แบบนี้ในการทำโครงงานเพื่อใช้งานไมโครคอนโทรลเลอร์ตระกูล MCS-51 มีด้วยกันหลายประการดังนี้

1. หน่วยความจำโปรแกรมภายในตัวไมโครคอนโทรลเลอร์เป็นแบบแฟลช ทำให้สามารถลบและเขียนใหม่ได้นับพันครั้ง จึงสามารถใช้งานในรูปแบบของไมโครคอนโทรลเลอร์ชิปเดี่ยวไม่ต้องใช้หน่วยความจำภายนอกส่งผลให้สามารถใช้งานพอร์ตอินพุทเอาท์พุทของไมโครคอนโทรลเลอร์ได้อย่างเต็มประสิทธิภาย

2. ต้นทุนและเวลาในการพัฒนาระบบไมโครคอนโทรลเลอร์ลดลงอย่างมากเนื่องจากไม่ต้องใช้เครื่องมือพัฒนาจำพวกอีมูเลเตอร์และเครื่องโปรแกรมอีพรอม (EPROM)

3. บริษัทผู้ผลิตได้ทำการผลิตไมโครคอนโทรลเลอร์ตระกูลนี้ออกมาหลายเบอร์ และมีความสามารถแตกต่างกันไป ทำให้มีทางเลือกในการใช้งานสูง

4. ด้วยการใช้หน่วยความจำภายในตัวไมโครคอนโทรลเลอร์ทำให้สามารถป้องกันการคัดลอกข้อมูลของหน่วยความจำโปรแกรมได้เป็นอย่างดี

5. ในบางเบอร์ของไมโครคอนโทรลเลอร์ที่ผลิตโดย Atmel สามารถทำการโปรแกรมข้อมูลในหน่วยความจำได้โดยที่ไม่ต้องถอดตัวไมโครคอนโทรลเลอร์ออกมาทำการโปรแกรมใหม่หรือเรียกว่า การโปรแกรมในวงจร หรือ ในระนาบ (In-system programming) โดยใช้ลักษณะการติดต่อแบบ SPI (Serial Peripheral Interface) ทำให้การพัฒนาหรือการซ่อมบำรุง ตลอดจนการปรับปรุงหรืออัปเกรดข้อมูลในหน่วยความจำโปรแกรมทำได้อย่างสะดวก ภายใต้งบประมาณที่ไม่สูงมากนัก

6. ชุดคำสั่งและสถาปัตยกรรมพื้นฐานเหมือนกับไมโครคอนโทรลเลอร์ MCS-51 ของผู้ผลิตอื่นไม่ว่าจะเป็นอินเทล, ซีเมนส์ หรือดัลลัส

2.1.1 คุณสมบัติทางเทคนิคของไมโครคอนโทรลเลอร์ MCS-51

คุณสมบัติทางเทคนิคของไมโครคอนโทรลเลอร์ตระกูล MCS-51 อนุกรม AT89xx มีดังนี้

- เป็นไมโครคอนโทรลเลอร์ที่ใช้ซีพียูขนาด 8 บิต
- ภายในมีหน่วยความจำโปรแกรมเป็นแบบแฟลชสามารถลบและเขียนใหม่ได้พันครั้ง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

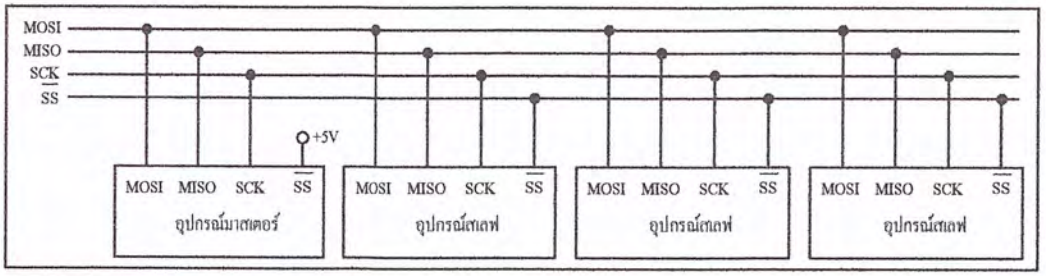
- หน่วยความจำข้อมูลพื้นฐานเป็นหน่วยความจำแบบแรม ในบางเบอร์จะมีหน่วยความจำแบบอีอีพรอมเพิ่มเติม
- ขาพอร์ตเป็นแบบสองทิศทาง สามารถใช้งานเป็นได้ทั้งอินพุทและเอาท์พุท
- มีวงจรสื่อสารอนุกรมแบบฟูลดูเพล็กซ์
- ไมเมอร์/เคาร์เตอร์ขนาด 16 บิตอย่างน้อย 2 ตัว
- สามารถรองรับแหล่งกำเนิดอินเตอร์รัปต์ได้ 6 ประเภท
- สามารถขยายหน่วยความจำภายนอกเพิ่มเติมได้สูงสุด 64 กิโลไบต์
- มีวงจรกำเนิดสัญญาณนาฬิกาอยู่ในภายชิป
- มีวงจรสื่อสารอนุกรมแบบ SPI สำหรับในอนุกรม AT89Sxx
- มีวอตช์ด็อกไทเมอร์ในตัว สำหรับในอนุกรม AT89Sxx

ในโครงงานนี้ใช้ไมโครคอนโทรลเลอร์ MCS-51 แบบแฟลช เบอร์ AT89S8252 ซึ่งมีหน่วยความจำโปรแกรมขนาด 8 กิโลไบต์ หน่วยความจำข้อมูล แรม 256 ไบต์ อีอีพรอม 2 กิโลไบต์และจำนวนไทเมอร์/เคาร์เตอร์ 16 บิต 3 ตัว

คุณสมบัติเด่นอีกประการหนึ่งที่มีอยู่ในไมโครคอนโทรลเลอร์ MCS-51 แบบแฟลช ในอนุกรม AT89Sxx ของ Atmel คือ ส่วนเชื่อมต่ออุปกรณ์อนุกรม หรือ SPI (Serial Peripheral Interface) ซึ่งเป็นอีกหนึ่งรูปแบบของการขยายขีดความสามารถของไมโครคอนโทรลเลอร์สมัยใหม่ ซึ่งในอดีตมีเพียงไมโครคอนโทรลเลอร์ตระกูล 68HC11 ของโมโตโรล่าเท่านั้นที่มีความสามารถในส่วนนี้ แต่ในปัจจุบันและอนาคต SPI จะกลายเป็นคุณสมบัติมาตรฐานที่ต้องมีในไมโครคอนโทรลเลอร์ ดังจะเห็นได้จากในไมโครคอนโทรลเลอร์ PIC รุ่นใหม่ รวมถึงไมโครคอนโทรลเลอร์ AVR ซึ่งเป็นไมโครคอนโทรลเลอร์อีกตระกูลหนึ่งของ Atmel ต่างก็บรรจุส่วน SPI ไว้ด้วย

ส่วนเชื่อมต่ออุปกรณ์อนุกรม หรือ SPI นี้เป็นการเชื่อมต่อไมโครคอนโทรลเลอร์เข้ากับอุปกรณ์ภายนอกที่มีลักษณะการถ่ายทอดข้อมูลในลักษณะอนุกรมแบบซิงโครนัส (Synchronous) ความเร็วสูง ในการสื่อสารข้อมูลแบบซิงโครนัสจะต้องใช้สัญญาณนาฬิกาเพื่อกำหนดจังหวะในการถ่ายทอดข้อมูลร่วมกัน และต้องกำหนดสถานภาพของอุปกรณ์ในระบบอย่างชัดเจน เนื่องจากในการเชื่อมต่อแบบ SPI สามารถต่อพ่วงอุปกรณ์ได้เป็นจำนวนมาก โดยต้องมีการกำหนดตัวควบคุมหลักเป็นอุปกรณ์ตัวแม่หรือมาสเตอร์ (master) และอุปกรณ์ที่นำมาต่อพ่วงในระบบเป็นอุปกรณ์ลูกหรือสเลฟ (slave) ดังแสดงรูปแบบ SPI อย่างง่ายในรูปที่ 2.2

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.1 รูปแบบของการเชื่อมต่อของ SPI อย่างง่าย

อุปกรณ์ตัวแม่หรือมาสเตอร์โดยส่วนใหญ่จะเป็นไมโครคอนโทรลเลอร์ และอุปกรณ์ตัวลูกหรือสเลฟ มักเป็นไอซีที่ทำหน้าที่เฉพาะพิเศษ เช่น หน่วยความจำ ไอซีขยายพอร์ต หรือ UART เป็นต้น แต่ด้วยการใช้ความสามารถของ SPI อาจสร้างโครงข่ายระบบควบคุมแบบมัลติโปรเซสเซอร์ได้ โดยทำการเชื่อมต่อไมโครคอนโทรลเลอร์หลายตัวเข้าด้วยกัน โดยผ่านทางขาเชื่อมต่อของระบบ SPI นั่นหมายความว่า อุปกรณ์ตัวลูกหรือสเลฟนั้นก็คือ ไมโครคอนโทรลเลอร์หรือไมโครโปรเซสเซอร์อีกชุดหนึ่งหรืออีกหลาย ๆ ชุดในกรณีต่อพ่วงกันมากกว่า 2 ตัว

นอกเหนือไปจากการใช้ SPI ในการติดต่อสื่อสารข้อมูลแบบซิงโครนัสแล้ว ในไมโครคอนโทรลเลอร์อนุกรม AT89Sxx ยังใช้ความสามารถของส่วน SPI ในการเขียนข้อมูลลงในหน่วยความจำโปรแกรมภายในตัวชิปด้วย จึงทำให้กระบวนการที่เรียกว่า การโปรแกรมหน่วยความจำโปรแกรมในระบบหรือในวงจร (In-System Programming: ISP) ทำให้สามารถแก้ไขหน่วยความจำโปรแกรมได้โดยไม่ต้องถอดตัวไมโครคอนโทรลเลอร์ออกจากระบบ เพียงต่อสายเข้าที่พอร์ต SPI คือขา SS, MOSI, MISO และ SCK (ซึ่งก็คือขา P1.4-P1.7) แล้วทำการรันซอฟท์แวร์ที่เขียนขึ้นเพื่อการโปรแกรมในระบบ ก็จะสามารถแก้ไขหน่วยความจำโปรแกรมได้แล้ว ซอฟท์แวร์ที่ใช้ในการเขียนข้อมูลลงหน่วยความจำโปรแกรมภายในตัวชิปนี้ ได้แก่

- SXA51 เป็นตัว Compile ไฟล์นามสกุล ASM ที่เขียนด้วยโปรแกรม Editor ต่าง ๆ ให้เป็นไฟล์นามสกุล HEX
- PonyProg - Serial Device Programmer Version 1.17h เป็นโปรแกรมที่ใช้ในการทำให้ไฟล์นามสกุล HEX ที่ได้จากการ Compile จากโปรแกรม SXA51 เป็นไฟล์นามสกุล HEX ในรูปแบบบน Windows สามารถดาวน์โหลดได้ที่ http://www.LancOS.com
- Atmel AVR ISP Version 2.65 เป็นโปรแกรมที่ใช้ในการเขียนข้อมูลลงในหน่วยความจำโปรแกรมภายในตัวชิปไมโครคอนโทรลเลอร์ โดยใช้ไฟล์นามสกุล HEX จากโปรแกรม PonyProg สามารถดาวน์โหลดโปรแกรมได้ที่ http://www.atmel.com/

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

วิธีการโปรแกรมแบบนี้จะช่วยลดต้นทุนในเรื่องของค่าเครื่องโปรแกรมเวลาในการปรับปรุงและแก้ไขหน่วยความจำโปรแกรมภายในไมโครคอนโทรลเลอร์ เพราะเพียงแต่ต่อสายเข้าไปเท่านั้น ไม่ต้องถอดฝาเครื่องหรือส่วนห่อหุ้มตัวชิปออก เพื่อนำตัวชิปออกมาโปรแกรมภายนอก ทำให้ลดค่าใช้จ่ายในการนำเครื่องหรือชิ้นงานกลับมายังแผนกซ่อมบำรุง เพียงเพื่อแก้ไขข้อมูลภายในหน่วยความจำโปรแกรมของไมโครคอนโทรลเลอร์

2.1.2 การจัดขาของไมโครคอนโทรลเลอร์ MCS-51

ไมโครคอนโทรลเลอร์ MCS-51 ทุกเบอร์จะมีขาใช้งานพื้นฐานเหมือนกันดังแสดงในรูปที่ 5.1 โดยมีรายละเอียดขั้นต้น ดังนี้

ขา Vcc ใช้สำหรับต่อไฟเลี้ยง +5 V

ขา GND เป็นขากราวด์ สำหรับต่อกับกราวด์ของระบบ

ขาพอร์ต 0 (P0.0-P0.7) มี 8 ขา แต่ละขาสามารถกำหนดให้เป็นได้ทั้งอินพุทและเอาท์พุท สำหรับใช้งานทั่วไป ถ้าหากต้องการกำหนดให้ขาพอร์ต 0 ขาใดขาหนึ่งเป็นอินพุทสามารถทำได้โดยการเขียนข้อมูล “1” ไปยังแต่ละบิตของพอร์ตที่ต้องการติดต่อด้วย ส่งผลให้ขาพอร์ตนั้นมีสถานะปล่อยลอย (Float) จึงมีอินพุทอิมพีแดนซ์สูง สามารถใช้งานเป็นขาพอร์ตอินพุทได้ นอกจากนั้นขาพอร์ตนี้ยังถูกใช้งานในการติดต่อกับขาแอดเดรสไบต์ต่ำของหน่วยความภายนอก (A0-A7) และขาข้อมูล (D0-D7) โดยใช้กระบวนการมัลติเพล็กซ์เข้าช่วย เพื่อสลับการทำงานให้เป็นได้ทั้งขาติดต่อแอดเดรสและขาข้อมูล

ขาพอร์ต 1 (P1.0-P1.7) มี 8 ขา แต่ละขาสามารถกำหนดให้เป็นได้ทั้งอินพุทและเอาท์พุท สำหรับใช้งานทั่วไป ถ้าหากต้องการกำหนดให้ขาพอร์ต 1 ขาใดขาหนึ่งเป็นอินพุทสามารถทำได้โดยการเขียนข้อมูล “1” ไปยังแต่ละบิตของพอร์ตที่ต้องการติดต่อด้วย นอกจากนั้นในอนุกรม AT89Sxx จะใช้ขา P1.0 เป็นขาอินพุทสำหรับนับค่าของไทเมอร์ 2 และ P1.1 เป็นขาอินพุททริกเกอร์ของไทเมอร์ 2 ในขณะที่ขา P1.4 ถึง P1.7 เป็นขาสำหรับเชื่อมต่อแบบ SPI เพื่อทำการโปรแกรมข้อมูลในระบบ

P1.4 ขา $\overline{SS}$ (Slave Select) เป็นขาเลือกการติดต่อในกรณีที่ไมโครคอนโทรลเลอร์เป็นอุปกรณ์สเลฟในระบบการติดต่อแบบ SPI

P1.5 ขา MOSI (Master data Output, Slave data Input) ในการติดต่อกับพอร์ต SPI

P1.6 ขา MISO (Master data Input, Slave data Output) ในการติดต่อกับพอร์ต SPI

P1.7 ขา SCK (Master clock output) เป็นขาสัญญาณนาฬิกาของการติดต่อกับพอร์ต SPI

ขาพอร์ต 2 (P2.0-P2.7) มี 8 ขา แต่ละขาสามารถกำหนดให้เป็นได้ทั้งอินพุทและเอาท์พุท

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สำหรับใช้งานทั่วไป ถ้าหากต้องการกำหนดให้ขาพอร์ต 2 ขาใดขาหนึ่งเป็นอินพุทสามารถทำได้โดยการเขียนข้อมูล "1" ไปยังแต่ละบิตของพอร์ตที่ต้องการติดต่อด้วย ส่งผลให้ขาพอร์ตนั้นมีสถานะปล่อยลอย (Float) จึงมีอินพุทอิมพีแดนซ์สูง สามารถใช้งานเป็นขาพอร์ตอินพุทได้ นอกจากนั้นขาพอร์ตนี้ยังถูกใช้งานในการติดต่อกับขาแอดเดรสไบต์สูงของหน่วยความภายนอก (A8-A15)

ขาพอร์ต 3 (P3.0-P3.7) มี 8 ขา แต่ละขาสามารถกำหนดให้เป็นได้ทั้งอินพุทและเอาท์พุท สำหรับใช้งานทั่วไป ถ้าหากต้องการกำหนดให้ขาพอร์ต 3 ขาใดขาหนึ่งเป็นอินพุทสามารถทำได้โดยการเขียนข้อมูล "1" ไปยังแต่ละบิตของพอร์ตที่ต้องการติดต่อด้วย ส่งผลให้ขาพอร์ตนั้นมีสถานะปล่อยลอย (Float) จึงมีอินพุทอิมพีแดนซ์สูง สามารถใช้งานเป็นขาพอร์ตอินพุทได้ นอกจากนั้นขาพอร์ต 3 ยังเป็นขาที่มีหน้าที่การใช้งานพิเศษ ดังมีรายละเอียดขั้นต้นต่อไปนี้

P3.0 ใช้เป็นขาอินพุทสำหรับรับข้อมูลจากการสื่อสารแบบอนุกรม หรือขา RxD

P3.1 ใช้เป็นขาอินพุทสำหรับรับข้อมูลจากการสื่อสารแบบอนุกรม หรือขา TxD

P3.2 ใช้เป็นขาอินพุทสำหรับรับสัญญาณอินเตอร์รัปต์จากภายนอกช่องที่ 0 หรือขา $\overline{\text{INT0}}$

P3.3 ใช้เป็นขาอินพุทสำหรับรับสัญญาณอินเตอร์รัปต์จากภายนอกช่องที่ 1 หรือขา $\overline{\text{INT1}}$

P3.4 ใช้เป็นขาอินพุทสำหรับรับสัญญาณไทเมอร์จากภายนอกช่องที่ 0 หรือขา T0

P3.5 ใช้เป็นขาอินพุทสำหรับรับสัญญาณไทเมอร์จากภายนอกช่องที่ 1 หรือขา T1

P3.6 ใช้เป็นขาสัญญาณ $\overline{\text{WR}}$ ในกรณีที่ใช้เชื่อมต่อกับหน่วยความจำภายนอก

P3.7 ใช้เป็นขาสัญญาณ $\overline{\text{RD}}$ ในกรณีที่ใช้เชื่อมต่อกับหน่วยความจำภายนอก

ขารีเซต (Reset) ใช้ในการรีเซตการทำงานของไมโครคอนโทรลเลอร์ โดยในการป้อนสัญญาณเพื่อรีเซต สถานะที่ขานี้ต้องอยู่ในระดับรีเซตอย่างน้อย 2 แมชีนไซเกิล โดยที่วงจรกำเนิดสัญญาณนาฬิกายังคงทำงานต่อเนื่องไปอย่างเป็นปกติ

ขา ALE / $\overline{\text{PROG}}$ (Address Latch Enable/Program pulse input) เป็นขาที่ใช้ในการควบคุมการแลตช์ของขาพอร์ต 0 เมื่อมีการใช้งานหน่วยความจำภายนอก นอกจากนั้นขานี้ยังใช้เป็นขาสำหรับรับพัลส์ของการโปรแกรมสำหรับโปรแกรมข้อมูลลงในไมโครคอนโทรลเลอร์ MCS-51 ในรุ่นที่มีหน่วยความจำโปรแกรมแบบอีอีพรอม

ขา $\overline{\text{PSEN}}$ (Program Store Enable) ขานี้ใช้ในการส่งสัญญาณเพื่อร้องขอติดต่อกับหน่วยความจำโปรแกรมภายนอก เมื่อไมโครคอนโทรลเลอร์ต้องการอ่านข้อมูลจากหน่วยความจำโปรแกรมภายนอก ตัวไมโครคอนโทรลเลอร์จะส่งสัญญาณออกมาที่ขานี้ 2 ครั้งในแต่ละแมชีนไซเกิล แต่ถ้าหากติดต่อกับหน่วยความจำข้อมูลภายนอก ขานี้จะไม่มีการส่งสัญญาณใด ๆ ออกมา

ขา $\overline{\text{EA}}$ / Vpp (External Access enable/Programming voltage input) ใช้สำหรับเลือกการติดต่อหน่วยความจำโปรแกรมจากภายนอกหรือภายในตัวไมโครคอนโทรลเลอร์ ถ้าหากขานี้เป็น "0" เป็นการเลือกให้ไมโครคอนโทรลเลอร์ติดต่อกับหน่วยความจำโปรแกรมภายนอกแต่ถ้าหากขานี้เป็น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

"1" เป็นการเลือกให้ไมโครคอนโทรลเลอร์ติดต่อกับหน่วยความจำภายในตัวไมโครคอนโทรลเลอร์ นอกจากนี้ ที่ขานี้ยังใช้เป็นขาอินพุทสำหรับรับแรงดันไฟสูงสำหรับการโปรแกรมหน่วยความจำภายในไมโครคอนโทรลเลอร์ สำหรับไมโครคอนโทรลเลอร์ MCS-51 แบบแฟลชต้องการแรงดันสำหรับการโปรแกรม +12 V

ขา XTAL1 และ XTAL2 เป็นขาสำหรับต่อคริสตอลเพื่อสร้างสัญญาณนาฬิกาในการกำหนดจังหวะการทำงานของไมโครคอนโทรลเลอร์

รูปที่ 2.2 การจัดขามาตราฐานของไมโครคอนโทรลเลอร์ MCS-51

2.1.3 ไทเมอร์/เคาน์เตอร์ในไมโครคอนโทรลเลอร์ MCS-51

ไทเมอร์/เคาน์เตอร์ (Timer/Counter) เป็นอีกหนึ่งส่วนประกอบที่สำคัญของไมโครคอนโทรลเลอร์ เนื่องจากในการทำงานของไมโครคอนโทรลเลอร์จะต้องมีการเก็บและตรวจสอบค่าของเวลาและจำนวนของสัญญาณนาฬิกาอยู่ตลอดเวลา ทั้งนี้เพื่อประโยชน์ในการสร้างฐานเวลา สร้างสัญญาณพัลส์ เปรียบเทียบค่าเวลา หรือเปรียบเทียบค่าของการนับ รวมไปถึงการกำหนดอัตราเร็วในการสื่อสารข้อมูลของพอร์ตอนุกรมด้วย

ในไมโครคอนโทรลเลอร์ MCS-51 แบบแฟลชเบอร์ AT89C51 มีวงจรไทเมอร์/เคาน์เตอร์ขนาด 16 บิต 2 ตัว โดยค่าของวงจรไทเมอร์/เคาน์เตอร์นี้จะเก็บไว้ในรีจิสเตอร์ขนาด 16 บิตที่ชื่อ ไทเมอร์ 0 (Timer0) และไทเมอร์ 1 (Timer1) เรียกสั้น ๆ ว่า T0 และ T1 ในไมโครคอนโทรลเลอร์ MCS-51 แบบแฟลชเบอร์ AT89C52 และอนุกรม AT89Sxx จะมีไทเมอร์/เคาน์เตอร์ถึง 3 ตัว คือ ไทเมอร์ 2 (Timer2: T2) เพิ่มเติมมา โดยรีจิสเตอร์ไทเมอร์ /เคาน์เตอร์ทั้งสามตัวสามารถกำหนดให้ทำงานเป็นตัวตั้งเวลาหรือ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ไทเมอร์และตัวนับหรือเคาน์เตอร์ได้อย่างอิสระต่อกัน การนับค่าของไทเมอร์/เคาน์เตอร์ในไมโครคอนโทรลเลอร์ MCS-51 จะเป็นการนับขึ้นหรือเพิ่มค่าเพียงทางเดียว

- การทำงานเป็นไทเมอร์

เมื่อกำหนดให้ทำงานเป็นตัวตั้งเวลาหรือไทเมอร์ ค่าของรีจิสเตอร์จะเพิ่มขึ้นในทุก ๆ แมชชีนไซเกิล ดังนั้นเมื่อทำงานเป็นไทเมอร์ รีจิสเตอร์จะทำการนับค่าของแมชชีนไซเกิลนั่นเอง และเนื่องจากแมชชีนไซเกิลประกอบด้วยคาบเวลาของวงจรกำเนิดสัญญาณนาฬิกา 12 คาบเวลา ดังนั้นอัตราในการนับของรีจิสเตอร์จึงเท่ากับ 1/12 ของความถี่สัญญาณนาฬิกา

- การทำงานเป็นเคาน์เตอร์

เมื่อทำงานเป็นตัวนับหรือเคาน์เตอร์ ค่าของรีจิสเตอร์จะเพิ่มขึ้นก็ต่อเมื่อมีการเปลี่ยนแปลงของระดับลอจิกจาก "1" เป็น "0" เกิดขึ้นที่ขาอินพุททางฮาร์ดแวร์ของวงจรไทเมอร์/เคาน์เตอร์ ซึ่งก็คือ ขาT0 (P3.4), ขาT1 (P3.5) และขาT2 (P1.0) ของไมโครคอนโทรลเลอร์ MCS-51 แบบแฟลชเบอร์ AT89C52 และอนุกรม AT89Sxx โดยจะมีการสุ่มรับสัญญาณจากขาอินพุทในทุก ๆ คาบเวลาที่ 2 ของสเตตที่ 5 (S5P2) ในแต่ละแมชชีนไซเกิล

เมื่อสัญญาณอินพุทเปลี่ยนแปลงจาก "1" เป็น "0" เป็นเวลาหนึ่งไซเกิล ในไซเกิลต่อมาค่าของการนับจะเพิ่มขึ้นหนึ่งค่า และจะไปปรากฏในรีจิสเตอร์ภายในคาบเวลาที่ 1 ของสเตตที่ 3 (S3P1) ของแมชชีนไซเกิลต่อไปหลังจากที่ตรวจจับพบการเปลี่ยนแปลงที่ขาไทเมอร์อินพุทแล้ว เมื่อเป็นเช่นนี้ในกระบวนการตรวจจับการเปลี่ยนแปลงของสัญญาณอินพุทที่ขาไทเมอร์จะต้องใช้ 2 แมชชีนไซเกิล อัตราการนับของเคาน์เตอร์จึงเท่ากับ 1/24 ของความถี่สัญญาณนาฬิกา ดังนั้น ความถี่สูงสุดของสัญญาณอินพุทที่ไทเมอร์/เคาน์เตอร์ภายในไมโครคอนโทรลเลอร์ MCS-51 แบบแฟลชสามารถตรวจจับได้จึงเท่ากับความถี่ของสัญญาณนาฬิกาหารด้วย 24 ยกตัวอย่าง ในไมโครคอนโทรลเลอร์ MCS-51 แบบแฟลชเบอร์ AT89S8252 สามารถใช้สัญญาณนาฬิกาได้สูงสุด 24 MHz ดังนั้นความถี่สูงสุดของสัญญาณอินพุทที่ไทเมอร์/เคาน์เตอร์สามารถตรวจจับได้คือ 1 MHz

- รีจิสเตอร์ที่เกี่ยวข้องกับการทำงานของไทเมอร์/เคาน์เตอร์ 0 และ 1

ในการทำงานของไทเมอร์/เคาน์เตอร์ 0 และ 1 ในไมโครคอนโทรลเลอร์ MCS-51 แบบแฟลชมีรีจิสเตอร์ที่ต้องเกี่ยวข้องเป็นพื้นฐานอยู่ 6 ตัว ดังมีรายละเอียดต่อไปนี้

รีจิสเตอร์ไทเมอร์

มีด้วยกัน 4 ตัวคือ TL0 มีแอดเดรสอยู่ที่ 8AH, TH0 มีแอดเดรสอยู่ที่ 8CH, TL1 มีแอดเดรสอยู่ที่ 8BH และ TH1 มีแอดเดรสอยู่ที่ 8DH รีจิสเตอร์ทั้ง 4 ตัวจะอยู่ในพื้นที่ของรีจิสเตอร์ฟังก์ชั่นพิเศษ หรือ SFR รีจิสเตอร์แต่ละตัวมีขนาด 8 บิต แต่ในการใช้งานโดยทั่วไปมักใช้รวมกันโดยจัดเป็นคู่คือ TL0 กับ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

TH0 รวมกันเป็นรีจิสเตอร์ Timer 0 ขนาด 16 บิต และTL1 กับ TH1 รวมกันเป็นริจิสเตอร์ Timer 1 ขนาด 16 บิต โดยใน TL0 และ TL1 จะเก็บข้อมูล 8 บิตล่าง ส่วน TH0 และ TH1 เก็บข้อมูล 8 บิตบน รีจิสเตอร์ไทเมอร์ทั้ง 2 คู่เมื่อนำมาใช้งานร่วมกันจะสามารถเก็บค่าของการนับได้สูงสุด 65,536 ค่าหรือ FFFFH เมื่อนับถึงค่านี้แล้วก็จะวนไปเริ่มนับ 0000H ใหม่ และเมื่อเกิดการนับรอบใหม่ จะมีการเซตบิต TF0 หรือ TF1 ในรีจิสเตอร์ TCON ที่ใช้ควบคุมการทำงานของไทเมอร์ เพื่อแจ้งให้ทราบว่าเกิดการนับเกินค่าสูงสุดแล้ว การเซตบิต TF0 หรือ TF1 ขึ้นอยู่กับว่าเลือกใช้งานรีจิสเตอร์ไทเมอร์ตัวใดอยู่

รีจิสเตอร์ควบคุมการทำงานของไทเมอร์/เคาน์เตอร์ หรือ TCON (Timer/Counter Control Register)

เป็นรีจิสเตอร์ขนาด 8 บิต มีแอดเดรสอยู่ที่ 88H ในพื้นที่ของรีจิสเตอร์ SFR สามารถเข้าถึงได้ในระดับบิต มีรายละเอียดการทำงานดังนี้

บิต 7	บิต 6	บิต 5	บิต 4	บิต 3	บิต 2	บิต 1	บิต 0
TF1	TR1	TF0	TR0	IE1	IT1	IE0	IT0

TF1 (Timer 1 overflow flag): เซตด้วยกระบวนการทางฮาร์ดแวร์ เมื่อค่าของรีจิสเตอร์ Timer 1 เกิดการนับเกินหรือเกิดโอเวอร์โฟลว การเคลียร์บิตนี้ทำได้ด้วยกระบวนการทางฮาร์ดแวร์เช่นกันโดยบิตนี้จะเคลียร์เมื่อมีการอินเตอร์รัปต์เกิดขึ้น

TR1 (Timer 1 run control bit): ใช้ในการเปิดปิดการทำงานของไทเมอร์ 1 (อีนาเบิลหรือดิสเอเบิล) ทำการเซตและเคลียร์ด้วยกระบวนการทางซอฟท์แวร์ ถ้าต้องการให้ไทเมอร์ 1 ทำงานต้องเซตบิตนี้ให้เป็น "1"

TF0 (Timer 0 overflow flag): เซตด้วยกระบวนการทางฮาร์ดแวร์ เมื่อค่าของรีจิสเตอร์ Timer 0 เกิดการนับเกินหรือเกิดโอเวอร์โฟลว การเคลียร์บิตนี้ทำได้ด้วยกระบวนการทางฮาร์ดแวร์เช่นกันโดยบิตนี้จะเคลียร์เมื่อมีการอินเตอร์รัปต์เกิดขึ้น

TR0 (Timer 0 run control bit): ใช้ในการเปิดปิดการทำงานของไทเมอร์ 0 (อีนาเบิลหรือดิสเอเบิล) ทำการเซตและเคลียร์ด้วยกระบวนการทางซอฟท์แวร์ ถ้าต้องการให้ไทเมอร์ 0 ทำงานต้องเซตบิตนี้ให้เป็น "1"

IE1 (External Interrupt 1 edge flag): บิตนี้จะใช้ในกระบวนการอินเตอร์รัปต์ สามารถเซตได้ด้วยกระบวนการทางฮาร์ดแวร์ เมื่อสามารถตรวจจับขอบขาของสัญญาณอินเตอร์รัปต์จากภายนอกที่ขาอินพุทอินเตอร์รัปต์ 1 (INT1) ได้ และจะทำการเคลียร์เมื่อมีการบริการอินเตอร์รัปต์เกิดขึ้น

IT1 (Interrupt 1 type control bit): บิตนี้จะใช้ในกระบวนการอินเตอร์รัปต์ โดยใช้ในการเลือกลักษณะของสัญญาณอินเตอร์รัปต์จากภายนอกที่ต้องการให้ทำการตอบสนองสำหรับขาอินพุทอินเตอร์รัปต์ 1 (INT1) การเซตและเคลียร์ทำได้ด้วยกระบวนการซอฟท์แวร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

"0" เลือกขอบขาลงของสัญญาณ (Falling edge)

"1" เลือกระดับลอจิกต่ำ (low level triggered)

IE0 (External Interrupt 0 edge flag): บิตนี้จะใช้ในกระบวนการอินเตอร์รัปต์ สามารถเซตได้ด้วยกระบวนการทางฮาร์ดแวร์ เมื่อสามารถตรวจจับขอบขาของสัญญาณอินเตอร์รัปต์จากภายนอกที่ขาอินพุทอินเตอร์รัปต์ 0 (INT0) ได้ และจะทำการเคลียร์เมื่อมีการบริการอินเตอร์รัปต์เกิดขึ้น

IT0 (Interrupt 0 type control bit): บิตนี้จะใช้ในกระบวนการอินเตอร์รัปต์ โดยใช้ในการเลือกลักษณะของสัญญาณอินเตอร์รัปต์จากภายนอกที่ต้องการให้ทำการตอบสนองสำหรับขาอินพุทอินเตอร์รัปต์ 0 (INT0) การเซตและเคลียร์ทำได้ด้วยกระบวนการซอฟท์แวร์

"0" เลือกขอบขาลงของสัญญาณ (Falling edge)

"1" เลือกระดับลอจิกต่ำ (low level triggered)

รีจิสเตอร์เลือกโหมดการทำงานของไทเมอร์/เคาน์เตอร์หรือTMOD (Timer/Counter Mode Control Register)

เป็นรีจิสเตอร์ขนาด 8 บิต มีแอดเดรสอยู่ที่ 89H ในพื้นที่ของรีจิสเตอร์ SFR ไม่สามารถเข้าถึงได้ในระดับบิต แบ่งการทำงานออกเป็น 2 ส่วนคือ 4 บิตล่างใช้ในการเลือกโหมดการทำงานของไทเมอร์ 0 และ 4 บิตบนใช้ในการเลือกโหมดการทำงานของไทเมอร์ 1 ดังนั้นในการอธิบายการทำงานจะขออธิบายเพียงส่วนเดียวดังนี้

บิต 7	บิต 6	บิต 5	บิต 4	บิต 3	บิต 2	บิต 1	บิต 0
GATE	C/$\overline{T}$	M1	M0	GATE	C/$\overline{T}$	M1	M0
ไทเมอร์ 1				ไทเมอร์ 0			

GATE: ใช้เลือกลักษณะการควบคุมการทำงานของไทเมอร์/เคาน์เตอร์

"0" ไทเมอร์/เคาน์เตอร์จะทำงานเมื่อบิต TRx ในรีจิสเตอร์ TCON เป็น "1" เรียกการควบคุมแบบนี้ว่า การควบคุมทางซอฟท์แวร์

"1" ไทเมอร์/เคาน์เตอร์จะทำงานเมื่อบิต TRx ในรีจิสเตอร์ TCON เป็น "1" และสถานะลอจิกที่ขาอินพุทอินเตอร์รัปท์ INT0 และ INT1 เป็น "1" เรียกการควบคุมแบบนี้ว่า การควบคุมทางฮาร์ดแวร์

C/$\overline{T}$ (Timer or Counter selector) : ใช้เลือกลักษณะการทำงานของไทเมอร์/เคาน์เตอร์

"0" เลือกให้ทำงานเป็นไทเมอร์ โดยใช้สัญญาณอินพุทจากสัญญาณนาฬิกาภายในไมโครคอนโทรลเลอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

"1" เลือกให้ทำงานเป็นเคาน์เตอร์ โดยรับสัญญาณอินพุทจากภายนอกที่เข้ามาทางขา T0 และ T1

M1, M0 (Mode selector bit): ใช้เลือกโหมดการทำงานของไทเมอร์/เคาน์เตอร์

"00" เลือกให้ทำงานในโหมดไทเมอร์/เคาน์เตอร์ 13 บิต

"01" เลือกให้ทำงานในโหมดไทเมอร์/เคาน์เตอร์ 16 บิต

"10" เลือกให้ทำงานในโหมดไทเมอร์/เคาน์เตอร์ขนาด 8 บิตแบบตั้งค่าอัตโนมัติ

"11" สำหรับไทเมอร์ 0 เลือกให้ทำงานในโหมดไทเมอร์/เคาน์เตอร์แยกส่วน โดยแยกออกเป็นไทเมอร์/เคาน์เตอร์ 8 บิต 2 ตัว รีจิสเตอร์ TL0 จะได้รับการควบคุมการเปิดปิดจากบิต TR0 ในรีจิสเตอร์ TCON และรีจิสเตอร์ TH0 ซึ่งเป็นไทเมอร์/เคาน์เตอร์ 8 บิตอีกตัวหนึ่ง จะได้รับการควบคุมจากบิต TR1 ในรีจิสเตอร์ TCON

สำหรับไทเมอร์ 1 เป็นการสั่งให้ไทเมอร์/เคาน์เตอร์ 1 หยุดทำงาน(ดิสเอเบิล)

โหมดการทำงานของไทเมอร์/เคาน์เตอร์ 0 และ 1

ไทเมอร์ 0 และ ไทเมอร์ 1 สามารถเลือกโหมดการทำงานได้ 4 โหมดคือ โหมด 0 : ไทเมอร์/เคาน์เตอร์ 13 บิต (13 bit timer/counter), โหมด 1 : ไทเมอร์/เคาน์เตอร์ 16 บิต (16 bit timer/counter), โหมด 2: ตั้งค่าอัตโนมัติขนาด 8 บิต (8 bit auto-reload timer/counter) และ โหมด 3: ไทเมอร์/เคาน์เตอร์แยกส่วน (split timer/counter) หรืออาจเรียกว่าโหมดไทเมอร์/เคาน์เตอร์ 8 บิตก็ได้ ในขณะที่ไทเมอร์ 2 มีโหมดการทำงาน 3 โหมดคือ โหมดแคปเจอร์หรือตรวจจับสัญญาณ(capture), โหมดตั้งค่าอัตโนมัติ (auto-reload) และโหมดกำเนิดอัตราเร็วในการสื่อสารข้อมูลอนุกรมหรืออัตราบอด (baud rate generator)

เนื่องจากในภาคการศึกษานี้ จะใช้เพียง ไทเมอร์ 0 เป็นเคาน์เตอร์ และไทเมอร์ 1 เป็นไทเมอร์ จะไม่ได้ใช้ไทเมอร์ 2 ดังนั้นจะอธิบายการทำงานในแต่ละโหมดของไทเมอร์ 0 และไทเมอร์ 1 เท่านั้น จะไม่อธิบายโหมดการทำงานของไทเมอร์ 2

การเลือกโหมดการทำงานของไทเมอร์/เคาน์เตอร์ 0 และ 1 สามารถกระทำได้ที่รีจิสเตอร์ TCON และ TMOD ร่วมกัน โดยรีจิสเตอร์ TCON ใช้ในการอีนาเบิลหรือดิสเอเบิลไทเมอร์/เคาน์เตอร์ ส่วน TMOD ใช้ในการเลือกโหมดและลักษณะการทำงาน และในการอธิบายโหมดการทำงานนี้จะใช้ไทเมอร์/เคาน์เตอร์ 1 ในการอธิบาย ซึ่งโหมดการทำงานจะมีลักษณะเหมือนกับไทเมอร์/เคาน์เตอร์ 0 ทุกประการ เพียงแต่เปลี่ยนรีจิสเตอร์และขาสัญญาณที่เกี่ยวข้องให้เป็นของไทเมอร์/เคาน์เตอร์ 0

1) การทำงานในโหมด 0 : ไทเมอร์/เคาน์เตอร์ 13 บิต

การทำงานในโหมดนี้จะเป็นการกำหนดให้ใช้งานรีจิสเตอร์ TL1 เพียง 5 บิต และ TH1 8 บิต โดย TL1 จะทำหน้าที่คล้ายเป็นปรีสเกลเลอร์หาร 32 สัญญาณอินพุทสำหรับการนับจะเลือกจาก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สัญญาณนาฬิกาภายในหรือภายนอกผ่านทางขา T1 ขึ้นอยู่กับการควบคุมของบิต C/T และ GATE ในรีจิสเตอร์ TMOD, บิต TR1 ในรีจิสเตอร์ TCON และสถานะของลอจิกที่ขาอินพุท INT1

เมื่อ TL1 นับครบ 32 คือจาก 0-31 ก็จะส่งสัญญาณไปยัง TH1 เพื่อทำการเพิ่มค่า ดังนั้นในโหมดนี้ค่าของการนับจะมีขนาด 13 บิต เมื่อทำการนับครบรอบ ก็จะทำการเซตบิต TF1 ในรีจิสเตอร์ TCON

2) การทำงานในโหมด 1: ไทเมอร์/เคาน์เตอร์ 16 บิต

การทำงานในโหมดนี้จะคล้ายกับโหมด 0 แต่จะใช้งานรีจิสเตอร์ TL1 ครบ 8 บิต และ TH1 ครบ 8 บิต ดังนั้นในโหมดนี้ค่าของการนับจะมีขนาด 16 บิต คือ 0000H-FFFFH เมื่อทำการนับครบรอบค่าของการนับจะเปลี่ยนจาก FFFFH เป็น 0000H ก็จะทำการเซตบิต TF1 ในรีจิสเตอร์ TCON

3) การทำงานในโหมด 2: ไทเมอร์/เคาน์เตอร์ 8 บิตแบบตั้งค่าอัตโนมัติ

การทำงานในโหมดนี้จะแยกรีจิสเตอร์ไทเมอร์ออกเป็น 2 ตัว ตัวละ 8 บิต โดยรีจิสเตอร์ TL1ทำหน้าที่เป็นตัวนับค่า ส่วน TH1 ใช้ในการเก็บค่าเริ่มต้นของการนับ เมื่อเริ่มต้นการทำงาน ค่าของรีจิสเตอร์ TH1 จะถูกส่งไปยังรีจิสเตอร์ TL1 ทำให้เมื่อเริ่มต้นการทำงานค่าของรีจิสเตอร์ TL1 และ TH1 จะเหมือนกัน เมื่อ TL1 นับถึง FFH และจะเริ่มต้นการนับรอบใหม่ จะทำการเซตบิต TF1 พร้อม ๆ กับทำการรับค่าการนับเริ่มต้นจาก TH1 ใหม่โดยอัตโนมัติ หรือเรียกกระบวนการนี้ว่า รีโหลด (reload) แม้ว่าจะมีการส่งค่าเริ่มต้นไปยัง TL1 แล้วก็ตาม ค่าของข้อมูลในรีจิสเตอร์ TH1 ก็ยังคงเป็นค่าเดิม ไม่มีการเปลี่ยนแปลง จนกว่าจะมีการกำหนดค่าใหม่ด้วยกระบวนการทางซอฟท์แวร์

4) การทำงานในโหมด 3: ไทเมอร์/เคาน์เตอร์แยกส่วนหรือไทเมอร์/เคาน์เตอร์ 8 บิต

ในโหมดนี้เป็นโหมดเดียวที่การทำงานของไทเมอร์ 0 และไทเมอร์ 1 ไม่เหมือนกัน ขออธิบายในส่วนของไทเมอร์ 1 ก่อน เมื่อเข้าสู่โหมดนี้ จะเป็นการสั่งให้ไทเมอร์/เคาน์เตอร์หยุดนับ ค่าของการนับก่อนหน้านี้จะถูกเก็บไว้ในรีจิสเตอร์ไทเมอร์ 1 มีลักษณะการทำงานเหมือนกับการดิสเอเบิลไทเมอร์/เคาน์เตอร์ 1 ด้วยการเคลียร์บิต TF1 ในรีจิสเตอร์ TCON

ส่วนการทำงานของไทเมอร์ 0 ในโหมดนี้จะแยกรีจิสเตอร์ไทเมอร์ 0 ออกเป็น 2 ตัว ตัวละ 8บิต คือรีจิสเตอร์ TL0 และ TH0 โดยแยกการทำงานออกจากกัน รีจิสเตอร์ TL0 สามารถเลือกการทำงานได้เหมือนกับไทเมอร์/เคาน์เตอร์ตามปกติ ส่วนรีจิสเตอร์ TH0 สามารถทำงานในโหมดไทเมอร์ได้เพียงอย่างเดียว กล่าวคือ สามารถรับสัญญาณอินพุทจากสัญญาณนาฬิกาภายในเพียงทางเดียวเท่านั้น แต่การแจ้งการนับเกินยังคงเหมือนเดิม หากแต่ TL0 แจ้งผ่านบิต TF0 ในขณะที่ TH0 จะแจ้งผ่านทางบิต TF1

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.1.4 อินเตอร์รัปต์ในไมโครคอนโทรลเลอร์ MCS-51

การอินเตอร์รัปต์ (Interrupt) เป็นชื่อเรียกกระบวนการที่เข้ามาขัดจังหวะการทำงานโดยปกติของไมโครคอนโทรลเลอร์ ซึ่งในไมโครคอนโทรลเลอร์ MCS-51 เบอร์ AT89S8252 สามารถตอบสนองการอินเตอร์รัปต์ได้จาก 6 แหล่งกำเนิด ประกอบด้วย การรับสัญญาณอินเตอร์รัปต์จากภายนอกผ่านทางขา $\overline{\text{INT0}}$ และ $\overline{\text{INT1}}$, สัญญาณอินเตอร์รัปต์จากไทเมอร์/เคาน์เตอร์ T0, T1 และ T2 และสัญญาณอินเตอร์รัปต์จากพอร์ตอนุกรมภายในไมโครคอนโทรลเลอร์ในขณะที่ไมโครคอนโทรลเลอร์

การจัดการอินเตอร์รัปต์

เมื่อมีการอินเตอร์รัปต์ในไมโครคอนโทรลเลอร์ MCS-51 เกิดขึ้น และมีการอีนาเบิลการตอบสนองการอินเตอร์รัปต์ไว้ กระบวนการหลังจากนั้นซีพียูจะทำการกระโดดไปยังแอดเดรสในหน่วยความจำที่กำหนดไว้ เรียกตำแหน่งแอดเดรสนี้ว่า แอดเดรสอินเตอร์รัปต์เวกเตอร์ (Interrupt vector address) ดังนั้นจะต้องมีการเขียนโปรแกรมย่อยการบริการอินเตอร์รัปต์ไว้ที่แอดเดรสอินเตอร์รัปต์เวกเตอร์นี้ โดยค่าของแอดเดรสอินเตอร์รัปต์เวกเตอร์จะแตกต่างกันไปในการอินเตอร์รัปต์แบบต่าง ๆ ดังมีรายละเอียดดังนี้

การอินเตอร์รัปต์ภายนอกที่ขา $\overline{\text{INT0}}$ มีค่าแอดเดรสอินเตอร์รัปต์เวกเตอร์อยู่ที่ 0003H
การอินเตอร์รัปต์จากไทเมอร์ 0 มีค่าแอดเดรสอินเตอร์รัปต์เวกเตอร์อยู่ที่ 000BH
การอินเตอร์รัปต์ภายนอกที่ขา $\overline{\text{INT1}}$ มีค่าแอดเดรสอินเตอร์รัปต์เวกเตอร์อยู่ที่ 0013H
การอินเตอร์รัปต์จากไทเมอร์ 1 มีค่าแอดเดรสอินเตอร์รัปต์เวกเตอร์อยู่ที่ 001BH
การอินเตอร์รัปต์จากพอร์ตอนุกรม มีค่าแอดเดรสอินเตอร์รัปต์เวกเตอร์อยู่ที่ 0023H
การอินเตอร์รัปต์จากไทเมอร์ 2 มีค่าแอดเดรสอินเตอร์รัปต์เวกเตอร์อยู่ที่ 002BH

การเขียนโปรแกรมย่อยบริการอินเตอร์รัปต์

มีหลักการโดยทั่ว ๆ ไป ดังนี้

1. ต้องเริ่มต้นด้วยแอดเดรสอินเตอร์รัปต์เวกเตอร์เสมอ เพื่อให้การตรวจสอบการทำงานทำได้ง่าย และแยกส่วนของโปรแกรมย่อยนี้ออกจากโปรแกรมหลักหรือโปรแกรมย่อยอื่น ๆ อย่างชัดเจนด้วยคำสั่ง ORG xxxxH (ค่าแอดเดรสอินเตอร์รัปต์เวกเตอร์)

2. เมื่อเข้าสู่โปรแกรมย่อย ควรเก็บค่าของรีจิสเตอร์หรือแฟลกที่ใช้แสดงสถานะต่าง ๆ ซึ่งต้องมีการใช้งานในโปรแกรมย่อยบริการอินเตอร์รัปต์นี้ไว้ในสแต็กเสียก่อน เพื่อป้องกันความผิดพลาดที่อาจเกิดขึ้นต่อการทำงานของทั้งโปรแกรมบริการอินเตอร์รัปต์นี้และโปรแกรมหลัก ด้วยคำสั่ง PUSH

3. เมื่อเขียนโปรแกรมบริการอินเตอร์รัปต์เรียบร้อยแล้ว ให้ทำการคืนค่าของรีจิสเตอร์ที่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

นำมาใช้ในโปรแกรมบริการอินเตอร์รัปต์ด้วยคำสั่ง POP ยกเว้นรีจิสเตอร์ที่ต้องการนำผลการกระทำในโปรแกรมบริการอินเตอร์รัปต์นี้ไปใช้งาน ซึ่งในทางปฏิบัติจริง ไม่พบมากนัก และไม่แนะนำให้เขียนโปรแกรมในลักษณะนี้

4. ปิดท้ายโปรแกรมย่อยบริการอินเตอร์รัปต์ด้วยคำสั่ง RETI เสมอ

รีจิสเตอร์ที่เกี่ยวข้องกับการบริการอินเตอร์รัปต์

การอินเตอร์รัปต์ในไมโครคอนโทรลเลอร์ MCS-51 แบบแฟลช มีรีจิสเตอร์ที่ต้องเกี่ยวข้องอยู่ 2 ตัว ดังมีรายละเอียดดังต่อไปนี้

1) รีจิสเตอร์อีนาเบิลการอินเตอร์รัปต์ หรือ IE (Interrupt Enable register)

มีแอดเดรสอยู่ที่ A8H ในพื้นที่ของรีจิสเตอร์ฟังก์ชั่นพิเศษ หรือ SFR มีขนาด 8 บิต สามารถเข้าถึงได้ในระดับบิต โดยสามารถเซตและเคลียร์ด้วยกระบวนการทางซอฟท์แวร์ ใช้ในการอีนาเบิลการตอบสนองการอินเตอร์รัปต์ในแบบต่าง ๆ มีรายละเอียดการทำงานดังนี้

บิต 7	บิต 6	บิต 5	บิต 4	บิต 3	บิต 2	บิต 1	บิต 0
EA	-	ET2	ES	ET1	EX1	ET0	EX0

EA (Global enable/disable interrupt bit) : ใช้ในการอีนาเบิลและดิสเอเบิลการตอบสนองการอินเตอร์รัปต์ทั้งหมด

"0" ดิสเอเบิลการอินเตอร์รัปต์ นั่นคือ กำหนดให้ไมโครคอนโทรลเลอร์ไม่ตอบสนองการอินเตอร์รัปต์ ไม่ว่าจะมาจากแหล่งกำเนิดใดก็ตาม

"1" อีนาเบิลการอินเตอร์รัปต์ นั่นคือ กำหนดให้ไมโครคอนโทรลเลอร์สามารถตอบสนองการอินเตอร์รัปต์จากแหล่งกำเนิดต่าง ๆ

นั่นคือ ถ้าต้องการให้ไมโครคอนโทรลเลอร์ตอบสนองการอินเตอร์รัปต์ไม่ว่าจะแหล่งกำเนิดใดจะต้องเซตบิตนี้ก่อนเสมอ

ET2 (Timer 2 interrupt enable bit): ใช้ในการอีนาเบิลการอินเตอร์รัปต์อันเนื่องมาจากการโอเวอร์โฟลวหรือการแคปเจอร์ ในไทเมอร์/เคาน์เตอร์ 2 จะมีเฉพาะในเบอร์ AT89C52 และอนุกรม AT89Sxx เท่านั้น

ES (Serial port interrupt enable bit): ใช้ในการอีนาเบิลการอินเตอร์รัปต์อันเนื่องมาจากการรับหรือส่งข้อมูลทางพอร์ตอนุกรมภายในไมโครคอนโทรลเลอร์ MCS-51

ET1 (Timer 1 interrupt enable bit): ใช้ในการอีนาเบิลการอินเตอร์รัปต์อันเนื่องมาจากการโอเวอร์โฟลวในไทเมอร์/เคาน์เตอร์ 1

EX1 (External interrupt 1 enable bit): ใช้ในการอีนาเบิลการอินเตอร์รัปต์อันเนื่องมาจากสัญญาณภายนอกที่ป้อนเข้ามายังขา $\overline{\text{INT1}}$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ET0 (Timer 0 interrupt enable bit): ใช้ในการอีนาเบิลการอินเตอร์รัปต์อันเนื่องมาจากการโอเวอร์โฟลวในไทเมอร์/เคาน์เตอร์ 0

EX0 (External interrupt 0 enable bit): ใช้ในการอีนาเบิลการอินเตอร์รัปต์อันเนื่องมาจากสัญญาณภายนอกที่ป้อนเข้ามายังขา $\overline{\text{INT0}}$

สำหรับบิต 6 ของรีจิสเตอร์ IE ไม่มีการใช้งาน ต้องกำหนดให้เป็น "0" เสมอ

2) รีจิสเตอร์จัดลำดับความสำคัญการตอบสนองการอินเตอร์รัปต์ หรือ IP (Interrupt Priority register)

มีแอดเดรสอยู่ที่ B8H ในพื้นที่ของรีจิสเตอร์ฟังก์ชั่นพิเศษหรือ SFR มีขนาด 8 บิต สามารถเข้าถึงได้ในระดับบิต โดยสามารถเซตและเคลียร์ด้วยกระบวนการทางซอฟท์แวร์ ใช้ในการเลือกลำดับความสำคัญของการตอบสนองการอินเตอร์รัปต์ว่า ต้องการให้ตอบสนองสัญญาณอินเตอร์รัปต์จากแหล่งกำเนิดใดเป็นลำดับก่อนหลัง ถ้าต้องการให้การอินเตอร์รัปต์จากแหล่งกำเนิดใดมีความสำคัญสูงสุด ให้กำหนดที่บิตนั้นเป็น "1" มีรายละเอียดของรีจิสเตอร์ IP ดังนี้

บิต 7	บิต 6	บิต 5	บิต 4	บิต 3	บิต 2	บิต 1	บิต 0
-	-	PT2	PS	PT1	PX1	PT0	PX0

PT2 (Timer 2 interrupt priority bit): ใช้ในการกำหนดความสำคัญของการอินเตอร์รัปต์อันเนื่องมาจากการโอเวอร์โฟลวหรือการแคปเจอร์ในไทเมอร์/เคาน์เตอร์ 2 จะมีเฉพาะในเบอร์ AT89C52 และอนุกรม AT89Sxx เท่านั้น

PS (Serial port interrupt priority bit): ใช้ในการกำหนดความสำคัญของการอินเตอร์รัปต์อันเนื่องมาจากการรับหรือส่งข้อมูลทางพอร์ตอนุกรมภายในไมโครคอนโทรลเลอร์ MCS-51

PT1 (Timer 1 interrupt priority bit): ใช้ในการกำหนดความสำคัญของการอินเตอร์รัปต์อันเนื่องมาจากการโอเวอร์โฟลวในไทเมอร์/เคาน์เตอร์ 1

PX1 (External interrupt 1 priority bit): ใช้ในการกำหนดความสำคัญของการอินเตอร์รัปต์อันเนื่องมาจากสัญญาณภายนอกที่ป้อนเข้ามายังขา $\overline{\text{INT1}}$

PT0 (Timer 0 interrupt priority bit): ใช้ในการกำหนดความสำคัญของการอินเตอร์รัปต์อันเนื่องมาจากการโอเวอร์โฟลวในไทเมอร์/เคาน์เตอร์ 0

PX0 (External interrupt 0 priority bit): ใช้ในการกำหนดความสำคัญของการอินเตอร์รัปต์อันเนื่องมาจากสัญญาณภายนอกที่ป้อนเข้ามายังขา $\overline{\text{INT0}}$

สำหรับบิต 6 และ 7 ของรีจิสเตอร์ IP ไม่มีการใช้งาน ต้องกำหนดให้เป็น "0" เสมอ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

แหล่งกำเนิดสัญญาณอินเตอร์รัปต์

ในที่นี้จะกล่าวถึงสัญญาณอินเตอร์รัปต์จากไทเมอร์/เคาน์เตอร์ T0 และ T1 ตามที่ได้ใช้งานเท่านั้น

สัญญาณอินเตอร์รัปต์จกาไทเมอร์/เคาน์เตอร์ T0 และ T1

แหล่งกำเนิดสัญญาณอินเตอร์รัปต์นี้จัดเป็นแหล่งกำเนิดสัญญาณอินเตอร์รัปต์ภายในแบบหนึ่ง โดยใช้การเกิดโอเวอร์โฟลวจากการนับค่าในไทเมอร์/เคาน์เตอร์ภายในไมโครคอนโทรลเลอร์ MCS-51 เมื่อไทเมอร์ 0 เกิดการโอเวอร์โฟลวก็จะทำการเซตบิต TF0 ในรีจิสเตอร์ TCON และถ้าไทเมอร์ 1 เกิดโอเวอร์โฟลวบิต TF1 ในรีจิสเตอร์ TCON จะได้รับการเซตเช่นเดียวกัน

อย่างไรก็ตามการอินเตอร์รัปต์แบบนี้ จะเกิดขึ้นหรือมีการตอบสนองก็ต่อเมื่อมีการอีนา เบิลการอินเตอร์รัปต์ โดยการเซตบิต EA, ET0 และ ET1 ในรีจิสเตอร์ IE

ลำดับความสำคัญของการอินเตอร์รัปต์

การกำเนิดสัญญาณอินเตอร์รัปต์ในในไมโครคอนโทรลเลอร์ MCS-51 เบอร์ AT89S8252 มีได้จาก 6 แหล่ง ดังนั้นจึงต้องมีการจัดลำดับความสำคัญ ในกรณีที่เกิดการอินเตอร์รัปต์ขึ้นพร้อม ๆ กันจากหลายแหล่งกำเนิด โดยสามารถกำหนดลำดับความสำคัญได้ที่รีจิสเตอร์ IP

อย่างไรก็ตาม ลำดับความสำคัญของการอินเตอร์รัปต์โดยปกติหรือในกรณีกำหนดข้อมูลในรีจิสเตอร์ IP ให้เป็น "1" ทุกบิต (ยกเว้นบิต 6 และ 7) จะเรียงลำดับจากความสำคัญสูงสุดไปจนถึงต่ำสุดดังนี้

1. อินเตอร์รัปต์ภายนอกที่ขา $\overline{\text{INT0}}$ หรือการเซตของบิต IE0
2. อินเตอร์รัปต์จากไทเมอร์ 0 หรือการเซตของบิต TF0
3. อินเตอร์รัปต์ภายนอกที่ขา $\overline{\text{INT1}}$ หรือการเซตของบิต IE1
4. อินเตอร์รัปต์จากไทเมอร์ 1 หรือการเซตของบิต TF1
5. อินเตอร์รัปต์จากพอร์ตอนุกรม หรือการเซตของบิต RI หรือ TI
6. อินเตอร์รัปต์จากไทเมอร์ 2 หรือการเซตของบิต TF2 หรือ EXF2

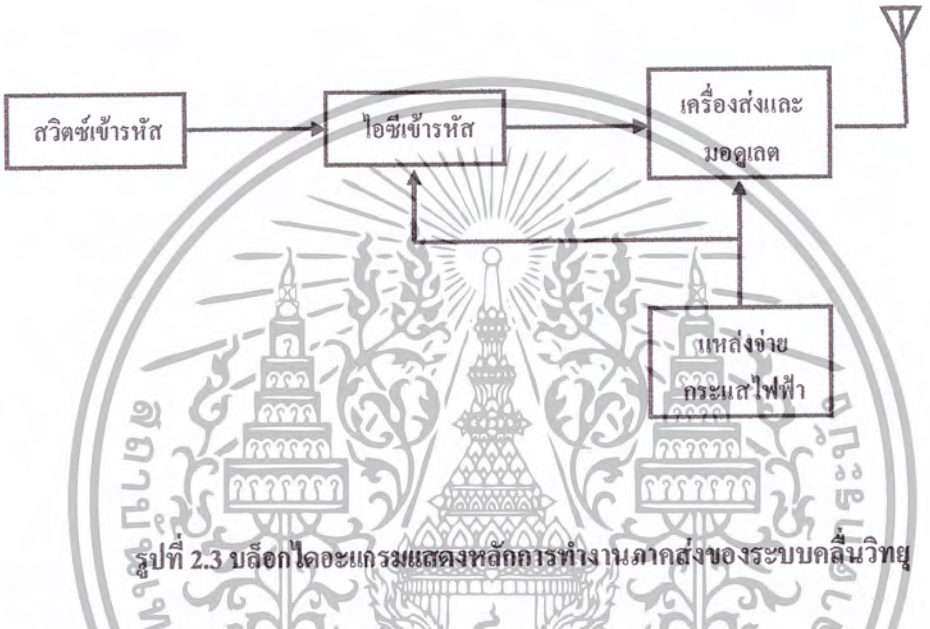
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

55498

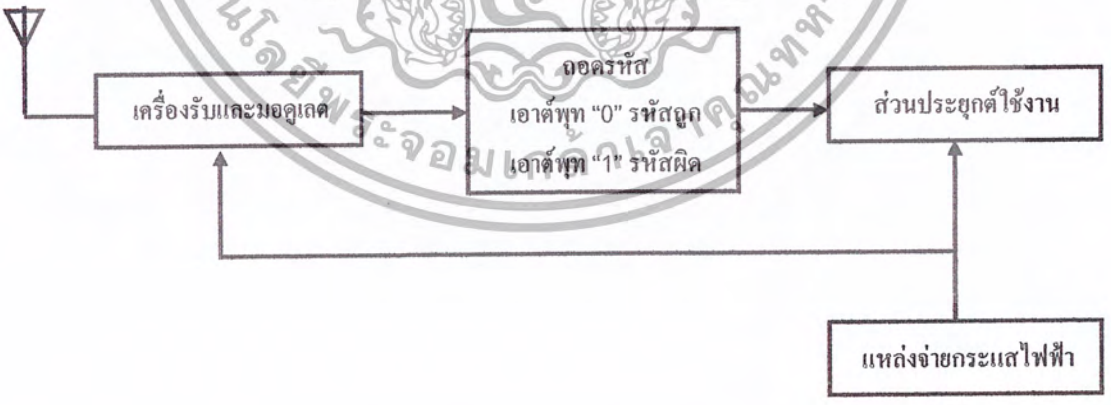
2.2 การสื่อสารไร้สายด้วยคลื่นวิทยุความถี่สูงระยะไกล

ส่วนของการสื่อสารไร้สายนี้เราสามารถแบ่งได้ด้วยกันเป็นสามส่วนใหญ่ เนื่องจากว่าในสองส่วนนั้น ได้มาจากอุปกรณ์จากตัวรถวิทยุบังคับ และตัวกล้องถ่ายวีดีโอไร้สายด้วย เพราะฉะนั้นในส่วนนี้ เราจะอธิบายถึงแต่เรื่องการส่งสัญญาณไร้สายระยะที่สามารถทำขึ้นมาเองได้ในเรื่องนี้

2.2.1 หลักการและทฤษฎีของระบบควบคุมระยะไกลแบบไร้สาย



รูปที่ 2.3 บล็อกไดอะแกรมแสดงหลักการทำงานภาคส่งของระบบคลื่นวิทยุ



รูปที่ 2.4 บล็อกไดอะแกรมแสดงหลักการทำงานภาครับของระบบคลื่นวิทยุ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.2.2 ความรู้พื้นฐานเกี่ยวกับอุปกรณ์ส่งคลื่นวิทยุ (Radio Controller)

เป็นตัวควบคุมทางไฟฟ้าหรืออิเล็กทรอนิกส์ ซึ่งมีอุปกรณ์ที่ควบคุมอยู่ระยะไกลจากตัวควบคุม ระหว่างตัวควบคุมและตัวถูกควบคุมจะต้องมีตัวกลางเป็นตัวส่งผ่านสัญญาณในการควบคุม เช่น สัญญาณจากคลื่นวิทยุ คลื่นเสียง ไมโครเวฟ หรืออื่นๆ ซึ่งตัวกลางนี้อาจเป็นอากาศ หรือ สายส่งสัญญาณก็ได้

จากรูปที่ 2.3 เป็นการแสดงถึงระบบของอุปกรณ์ส่งสัญญาณวิทยุ ซึ่งเป็นตัวผลิตสัญญาณที่ใช้ในการบังคับตัวถูกควบคุม โดยมีตัวกลาง เป็นตัวรับสัญญาณจากอุปกรณ์ส่งสัญญาณวิทยุ ไปยังวงจรที่ถูกควบคุม ในภาพ ตัวสัญญาณที่ถูกส่งมาออกมาจากอุปกรณ์ควบคุม จะเป็นในรูปของสัญญาณพัลส์ (Pulse) เมื่อสัญญาณพัลส์เข้าไปในตัวอุปกรณ์ที่ถูกควบคุมแล้วสัญญาณพัลส์จะเป็นตัวกระตุ้นระบบสวิตซ์ (Switch) ของอุปกรณ์ควบคุม

รูปที่ 2.5 บล็อกไดอะแกรมแสดงรูปแบบการส่งสัญญาณแบบพัลส์

ในทางปฏิบัติตัวอุปกรณ์ควบคุมนี้อาจเป็นตัวผลิตสัญญาณพัลส์ ที่ตัวเครื่องส่ง (Transmitter) ตัวกลางอาจจะเป็นอากาศ และอุปกรณ์ที่ถูกควบคุมอาจเป็นเครื่องรับ (Receiver) และระบบสวิตซ์ ดังรูป

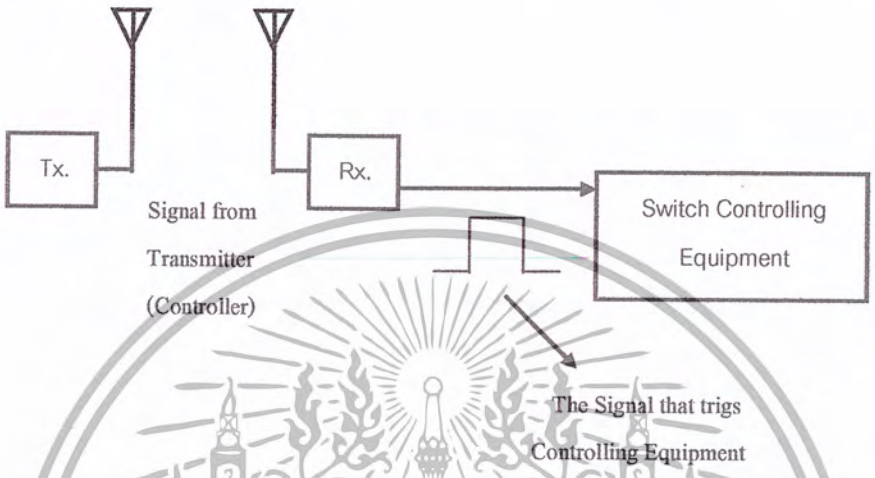
รูปที่ 2.6 แสดงหลักการทำงานของระบบคลื่นวิทยุผ่านทางตัวกลางที่เป็นอากาศ

2.2.3 อุปกรณ์ส่งสัญญาณ

โดยอาศัยหลักการของเครื่องส่งวิทยุ เราได้ใช้ความถี่บางความถี่ที่สามารถเดินทางบนอากาศได้ เช่นความถี่ย่าน MF, RF,ไมโครเวฟ (Microwave),อินฟราเรด (Infrared) เป็นต้น เมื่อเราผลิตความถี่นี้

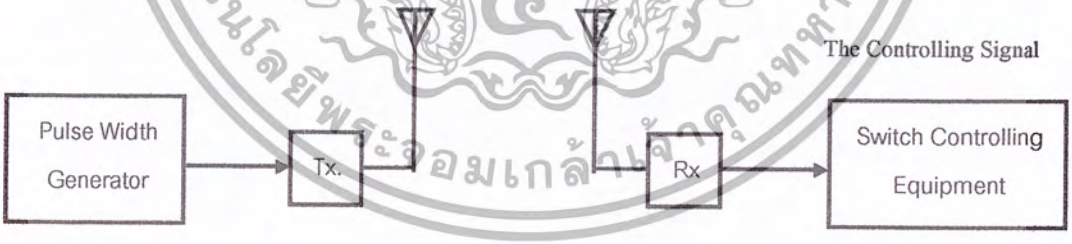
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ออกมาแล้วส่งให้มันเดินทางไปในอากาศ ดังนั้นสำหรับตัวเครื่องส่งแล้วจะต้องมีตัวที่ผลิตความถี่ที่จะส่งออกอากาศ หรือตัวกำหนดสัญญาณวิทยุ และที่ตัวเครื่องรับก็เช่นเดียวกันกับเครื่องรับวิทยุคือ จะรับเอาคลื่นที่อยู่ในอากาศนั้นมาแล้วแปลความหมาย ซึ่งตัวผลิตความถี่ของเครื่องส่งจะผลิตความถี่เท่าไรนั้น ขึ้นอยู่กับจุดประสงค์ในการใช้งานของเรา เพราะในย่านความถี่ต่างๆก็มีข้อดีและข้อเสียต่างกันออกไปในการส่งสัญญาณทางอากาศของเครื่องส่ง สำหรับอุปกรณ์ส่งสัญญาณวิทยุนั้น



รูปที่ 2.7 การใช้สัญญาณพัลส์กระตุ้นให้อุปกรณ์ควบคุมสวิตซ์ทำงาน

อาศัยความถี่ที่เราผลิตได้จากเครื่องส่ง เป็นเพียงคลื่นพาหะ (Carrier) เพื่อนำเอาสัญญาณพัลส์ไปเป็นตัวกระตุ้นให้อุปกรณ์ควบคุมสวิตซ์ทำงาน ดังรูป

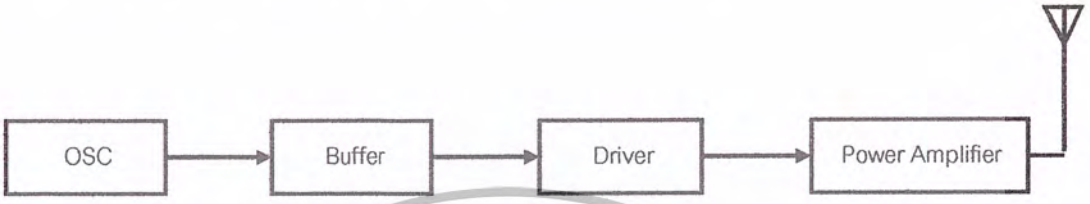


รูปที่ 2.8 วงจรกำหนดความกว้างของพัลส์

สรุปแล้วในภาคเครื่องส่งก็สามารถแบ่งลักษณะของเครื่องส่งออกได้เป็นสองชนิดคือ ชนิดที่มีเครื่องส่งอย่างเดียว และชนิดที่มีสัญญาณพัลส์กับเครื่องส่ง ประกอบกันดังรูปที่ 2.3 และ 2.4 ซึ่ง Tx คือ เครื่องส่ง และ Rx คือ เครื่องรับ และ อุปกรณ์กำหนดความกว้างของสัญญาณพัลส์ (Pulse Width Generator) คือสัญญาณพัลส์นั่นเอง สำหรับบล็อกไดอะแกรมของเครื่องส่งจะประกอบด้วยภาคความถี่

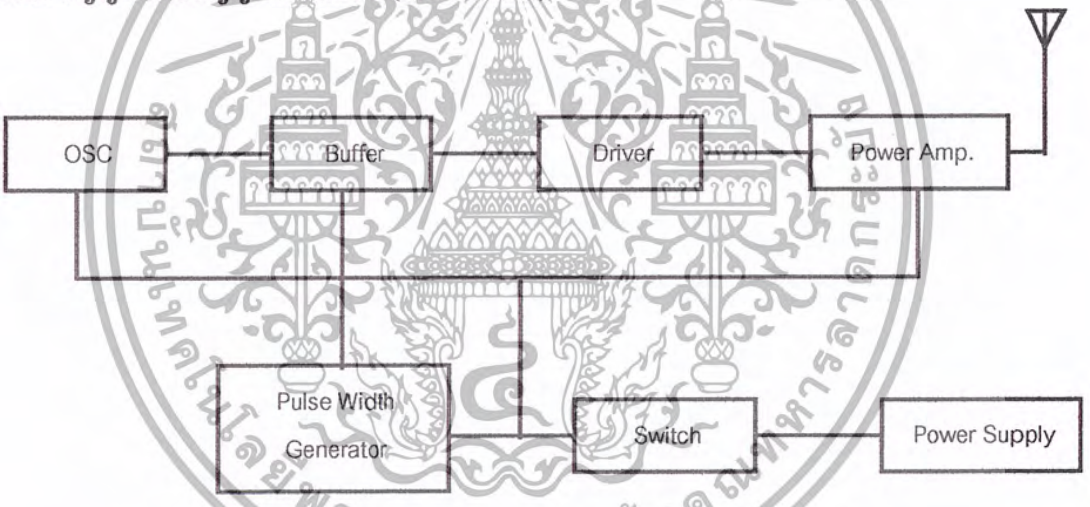
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

(Oscillator) ภาคที่กั้น OSC กับ บัฟ-เฟอร์ (Buffer), ภาคขับสัญญาณ (Driver), และภาคขยายกำลัง (Power Amplifier) ดังรูปที่ 2.7 ซึ่งเป็นตัวอย่างของเครื่องส่งอย่างง่ายๆ สำหรับวงจรที่มีการกำหนดความกว้างของสัญญาณก็เพิ่มอุปกรณ์กำหนดความกว้างของสัญญาณพัลส์เข้ามาอีก วงจรผลิตความถี่ (OSC) ผลิตความถี่ออกมา เพื่อที่จะผสมกับความกว้างสัญญาณ เราเรียกการผสมสัญญาณนั้นว่า ตัวปรับสัญญาณ (Modulator) แล้วสัญญาณจะผ่านทางบัฟเฟอร์ ภาคขับสัญญาณจนกระทั่งถึงภาคขยายกำลังเพื่อขยายสัญญาณให้มีขนาดความสูงขึ้น จากนั้นก็ส่งออกอากาศ ดังรูป



รูปที่ 2.9 วงจรผลิตความถี่

2.2.4 สัญญาณจากสัญญาณรหัสกว้าง (Pulse Width)



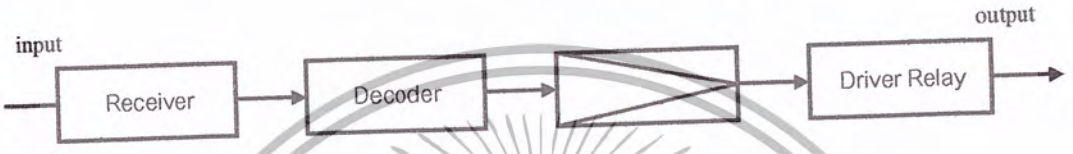
รูปที่ 2.10 วงจรส่งคลื่นวิทยุที่มีทั้งวงจรผลิตความถี่และวงจรกำหนดความกว้างสัญญาณ

สัญญาณจากวงจรผลิตความถี่ (OSC) จะเป็นตัวกระตุ้นตัวถูกควบคุม (Controlling Equipment) ทำงาน สัญญาณจากสัญญาณรหัสกว้าง จะเป็นตัวถูกกระตุ้นให้ตัวถูกควบคุม (Controlling Equipment) ทำงาน ในทางปฏิบัติสัญญาณจากวงจรผลิตความถี่ และสัญญาณรหัสก็คือ ความถี่ทั้งคู่ แต่อาจจะมีความแตก-ต่างกันบ้าง โดยทั่วไปแล้วความถี่ของ วงจรผลิตความถี่ (OSC) จะสูงกว่า สัญญาณกว้าง และความถี่ของภาคกำหนดความกว้างของสัญญาณ จะออกมาในรูปของสัญญาณกว้าง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.2.5 อุปกรณ์รับส่งสัญญาณ (Receiver)

อุปกรณ์รับส่งสัญญาณของวงจร Radio Controller จะเป็นตัวที่รับสัญญาณจากอากาศที่ถูกส่งมาโดย Transmitter ดังนั้นสำหรับภาครับจึงต้องมีความสามารถในการรับความถี่ที่ถูกส่งโดยอุปกรณ์ส่งสัญญาณเท่านั้น ทั้งนี้เพื่อป้องกันการรบกวน สัญญาณจากที่อื่นสำหรับอุปกรณ์รับสัญญาณส่วนประกอบใหญ่ของวงจรจะประกอบด้วย ภาครับและวงจรถอดรหัสสัญญาณ ภาครับจะรับเอาสัญญาณที่ส่งออกมาจากอุปกรณ์ส่งสัญญาณเข้าไปในเครื่องและตัวถอดรหัสสัญญาณจะทำการถอดเอาสัญญาณที่ได้รับออกมาเป็นพัลส์สัญญาณ เพื่อที่จะใช้งานต่อไปก่อนที่จะเอาไปใช้งานสัญญาณที่ได้ควรจะได้รับการขยายก่อนเพื่อให้สัญญาณมีขนาดใหญ่ขึ้น (แรงมากขึ้น)

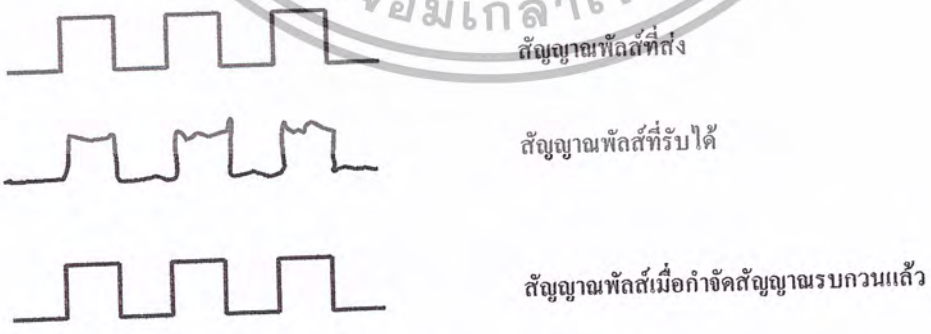


รูปที่ 2.11 วงจรแปลงสัญญาณคลื่นวิทยุ

2.2.6 การสื่อสารด้วยพัลส์

การสื่อสารด้วยพัลส์ได้เข้ามามีบทบาทอย่างมากในกิจการสื่อสารและโทรคมนาคม ตลอดจนการสื่อข้อมูลในรูปโทรเลข โทรพิมพ์ โทรศัพท์ การสื่อสารที่ใช้ในระบบราชการ การทหาร การพาณิชย์ การสื่อสารนี้มักจะส่งหรือสื่อสารกันด้วยสัญญาณพัลส์ ในบทนี้เราจะได้กล่าวถึงการสื่อสารที่ใช้พัลส์เป็นหลัก

ข้อดีของการสื่อสารแบบนี้เห็นได้ชัดในเรื่องความถูกต้องของข้อมูล ทั้งนี้เพราะเราส่งสัญญาณเป็น พัลส์ เมื่อมีสัญญาณอื่นรบกวนเข้ามาในระบบ การถอดรหัสออกมาเป็นพัลส์ก็ยังเหมือนเดิม ดังตัวอย่างในรูป



รูปที่ 2.12 สัญญาณพัลส์

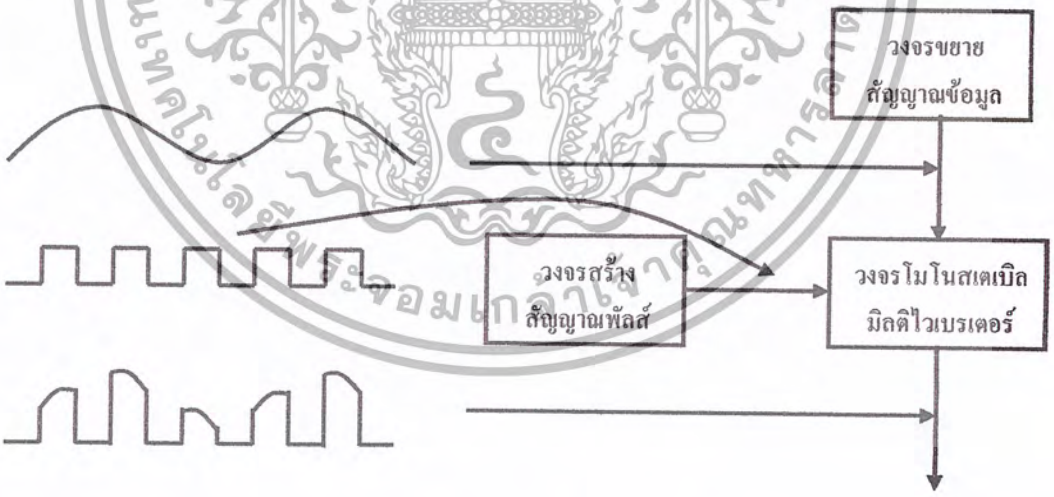
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

นอกจากนี้การสื่อสารด้วยพัลส์ยังมีข้อดีในเรื่องประสิทธิภาพของการส่งด้วย เพราะถ้าเป็นการส่งแบบวิทยุ การทำงานของเอาต์พุตทรานซิสเตอร์จะอยู่ในลักษณะของสวิตช์คัตออฟและอิ่มตัว ซึ่งทำให้กำลังงานสูญเสียในตัวเอาต์พุตทรานซิสเตอร์มีค่าน้อยกว่า

วงจรปรับสัญญาณพัลส์ (Pulse Amplitude Modulation)

การปรับสัญญาณพัลส์ คือ การสื่อสารอย่างหนึ่งที่นำคลื่นสัญญาณข้อมูลมามอดูเลตกับสัญญาณพัลส์ ซึ่งผลที่ได้คือ เอาต์พุตของสัญญาณจะประกอบด้วยสัญญาณพัลส์ที่มีแอมปลิจูดขึ้นอยู่กับคลื่นสัญญาณข้อมูลข่าวสาร โดยที่ความกว้างของสัญญาณพัลส์และความถี่ยังมีค่าคงเดิม การปรับสัญญาณพัลส์มีชื่อเรียกย่อๆว่า PAM (Pulse Amplitude Modulation)

บล็อกไดอะแกรมวงจรปรับสัญญาณพัลส์ ที่แสดงในรูปที่ 2.11 ประกอบด้วยวงจรแหล่งสัญญาณที่มีความถี่ไม่สูงมากนัก เพื่อนำไปมอดูเลตกับสัญญาณพัลส์ที่ได้มาจากวงจรสัญญาณพัลส์ วงจรมอดูเลตที่ใช้ก็เป็นวงจรโมโนสเตเบิลที่ได้รับการทริกจากวงจรสัญญาณพัลส์นั้นเอง เอาต์พุตที่ได้รับจากวงจรโมโนสเตเบิ้ลจะมีลักษณะของแอมปลิจูดของพัลส์ไม่เท่ากันขึ้นอยู่กับสัญญาณข้อมูล

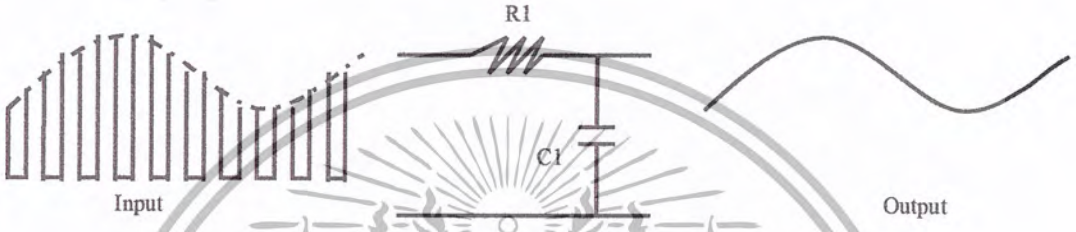


รูปที่ 2.13 วงจรปรับสัญญาณพัลส์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปนี้ เป็นการแสดงวงจรที่ใช้แทนบล็อกไดอะแกรมของรูปที่ 2.12 ถ้าพิจารณาให้ดีจะเห็นว่าส่วนของ Q1 และ Q2 ต่อเป็นวงจรโมโนสเตเบิลมัลติไวเบรเตอร์ เอาต์พุตของโมโนสเตเบิลจะขับทรานซิสเตอร์ Q3 โดยที่คอลเลคเตอร์ของ Q3 จะป้อนสัญญาณข้อมูลที่มันยกระดับมาทางไปตรงด้านบวก เมื่อ Q3 นำกระแสแรงดันตรงเอาต์พุต จะมีค่าเท่ากับแรงดันตกคร่อมตัวต้านทาน R6 ครั้นเมื่อ Q3 คัตออฟเอาต์พุตจะได้เท่ากับแรงดันข้อ-มูลที่ป้อนเข้ามาทางคอลเลคเตอร์

เมื่อมีการส่งข้อมูลออกไปยังผู้รับปลายทาง ผู้รับก็ต้องทำการดีมอดูเลต การดีมอดูเลตสัญญาณ (PAM) นี้ทำได้โดยการผ่านวงจรสัญญาณความถี่ต่ำที่ประกอบด้วย RC วงจรก็จะกรองแต่สัญญาณส่วนขอบเช่นเดียวกับการตรวจจับสัญญาณ (Detect) ในการรับสัญญาณ AM



รูปที่ 2.14 วงจรกรองสัญญาณความถี่ต่ำ

2.2.7 วงจร Pulse Width Modulation (PWM)

เป็นวงจรปรับคลื่นสัญญาณทางความกว้างของพัลส์สัญญาณ ซึ่งจะแบ่งตามขนาดของสัญญาณข้อ-มูล นั่นคือ ถ้าข้อมูลมีแอมปลิจูด (Amplitude) ความกว้างของพัลส์ ก็จะมากขึ้นตามไปด้วย วงจรปรับความกว้างสัญญาณ (Pulse Width Modulation) แบบง่าย

ปัจจุบันนี้เราสามารถเลือกใช้ IC ได้หลายเบอร์ด้วยกันมากมาย ซึ่งสามารถดัดแปลงความกว้างของสัญญาณพัลส์ได้โดยตรงและมีประสิทธิภาพที่ดี การดัดแปลงนี้กระทำได้โดยการแปลงสัญญาณความกว้างให้เป็นการแปรค่าทางแอมปลิจูด แล้วใช้วงจรกรองความถี่ต่ำผ่านเป็นตัวตรวจจับสัญญาณ

ปกติแล้วจะมีวงจรรวมสัญญาณ ที่ทำหน้าที่แปรสัญญาณพัลส์ให้เป็นสัญญาณลาดถ้าความกว้างของพัลส์มีมากแอมปลิจูดของสัญญาณลาด ก็จะมีค่ามากตามไปด้วย เมื่อนำเอามาสัญญาณปรับความกว้างผ่านวงจรรวมสัญญาณ เราก็จะได้สัญญาณที่มีแอมปลิจูดเปลี่ยนแปรตามสัญญาณข้อมูลทันที

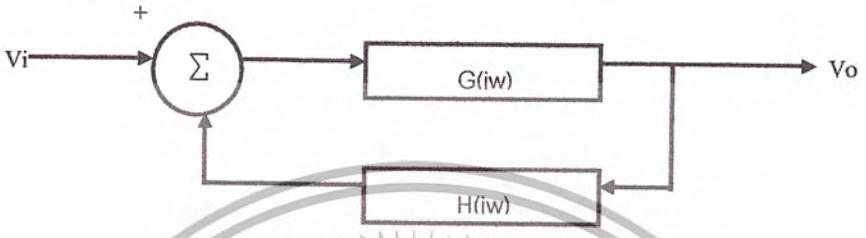
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.2.8 วงจรกำเนิดคามถี่วิทยุ (Radio Frequency Oscillator)

หลักการสำคัญในการกำเนิดความถี่มีอยู่ สองประการ

1.เฟสชิฟต์ (Phase Shift) รอบวงของระบบที่มีค่ารวมกันเป็น 0 องศา

2.อัตราการขยายครบรอบวง (loop gain) มีค่าเป็นหนึ่งพอดี ถ้าต้องการให้ขนาดของสัญญาณที่ได้จากการกำเนิดมีค่าคงที่



รูปที่ 2.15 วงจรกำเนิดความถี่วิทยุ

จากแผนผังวงจรในรูปจะเห็นได้ว่าส่วนของวงจรขยาย มีอัตราการขยายเปลี่ยนแปลงไปตามความถี่ G(jw) และมีอัตราการป้อนกลับเปลี่ยนแปรความถี่เป็น H(jw) เราสามารถกล่าวได้อีกนัยหนึ่งว่า ในระบบที่มีการป้อนกลับแบบติดลบ ถ้าเฟสชิฟต์ของวงจรป้อนกลับมีค่าเป็น 180 องศา ที่บางความถี่วงจรจะกำเนิดที่ความถี่นั้น และสำหรับวงจรที่มีการป้อนกลับแบบบวกเฟสชิฟต์ของการป้อนกลับต้องเป็น 0 องศา

อิมิตเตอร์ร่วม (Common Emitter) จะเห็นได้ว่าเฟสชิฟต์ของวงจรป้อนกลับต้องมีค่า 180 องศา แต่ถ้าในกรณีที่เราใช้วงจรแบบ เบสร่วม (Common Base Emitter) แล้วต้องไม่มีเฟสชิฟต์ระหว่างอินพุตและเอาต์พุต เพื่อให้กำเนิดสัญญาณอย่างต่อเนื่องได้ แต่ถ้ามีเฟสชิฟต์วงจรก็จะป้อนกลับเป็น 0 องศา

2.2.9 การทำงานของภาคส่งสัญญาณสื่อสาร

สามารถแยกได้เป็นสองส่วนคือ ภาคเข้ารหัส โดยใช้ IC และภาคส่งซึ่งใช้ความถี่วิทยุโดยการวงจร (RF) Oscillator

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.2.10 การทำงานของภาครับสัญญาณ

วงจรภาครับจะรับสัญญาณแบบ Regenerative และผ่านไปยังวงจรขยายแบบกลับ ซึ่งมีอัตราการขยายสูง แล้วต่อไปผ่านไปยังชมิตต์ทริกเกอร์ ซึ่งทำหน้าที่รับสัญญาณลูกคลื่นให้เป็นวงจรพัลส์ที่ถูกต้อง โดยผ่านตัวกลับสัญญาณเพื่อกลับเฟสของสัญญาณ

2.2.11 ปัญหาการทำงานของการสื่อสารไร้สายด้วยคลื่นวิทยุ

ปัญหาในส่วนนี้นั้นเราได้พบว่าปัญหาส่วนมากจะเกิดขึ้นคล้ายๆกับตัวรถ คือหาซื้ออุปกรณ์ได้ยากและเป็นปัญหาทางด้านเทคนิคเฉพาะด้านแต่น้อยกว่ารถที่ประสบมา และปัญหาที่ไม่สามารถตอบได้เช่น ต่อแล้วสัญญาณไม่ออกมา

2.3 การสื่อสารผ่านพอร์ทคอมพิวเตอร์

2.3.1 การสื่อสารแบบพาราเรล ผ่านทางพรินเตอร์พอร์ต

พอร์ตพาราเรลมาตรฐานของ IBM 80x86 PC จะมีอยู่ทั้งหมด 25 ขาด้วยกัน แบ่งหน้าที่ของแต่ละพิน
ออกได้ดังนี้

- **ขาส่งข้อมูลและกราวนด์ (Data lines and grounds)**

ขาส่งข้อมูลพาราเรลจะทั้งหมด 8 ขา สามารถส่งข้อมูลได้ทีละหนึ่งไบต์ แบบซิมเพลกหรือฮาล์ฟดูเพลกก็ได้ (Simplex and Half-duplex transfer)

- **ขาแสดงสถานะพรินเตอร์ (Printer status signals)**

ขาแสดงสถานะของพรินเตอร์จะมีทิศทางของสัญญาณจากพรินเตอร์ไปยัง PC มีอยู่ทั้งหมด 5 ขาด้วยกัน คือ

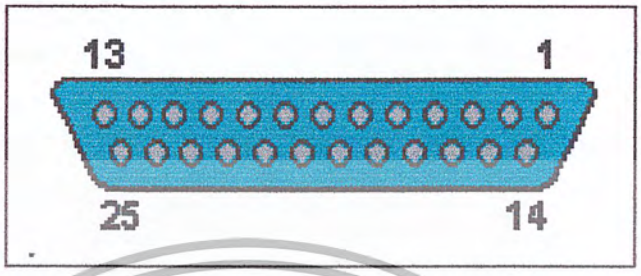
  - **PE** (ขา 12) ใช้เมื่อกระดาษหมด
  - **BUSY** (ขา 11) ใช้เมื่อพรินเตอร์ยังไม่สามารถรับข้อมูลใหม่ๆมาได้
  - **/ERROR** (ขา 15) ใช้เมื่อพรินเตอร์เกิดขัดข้องขึ้น ไม่สามารถใช้งานได้
  - **SLCT** (ขา 13) แสดงสัญญาณว่าพรินเตอร์เปิดอยู่และพร้อมใช้งาน
  - **/ACKNLG** (ขา 10) สัญญาณตอบรับ ACK เพื่อขอรับไบต์ใหม่

- **ขาควบคุมพรินเตอร์ (Printer control signals)**

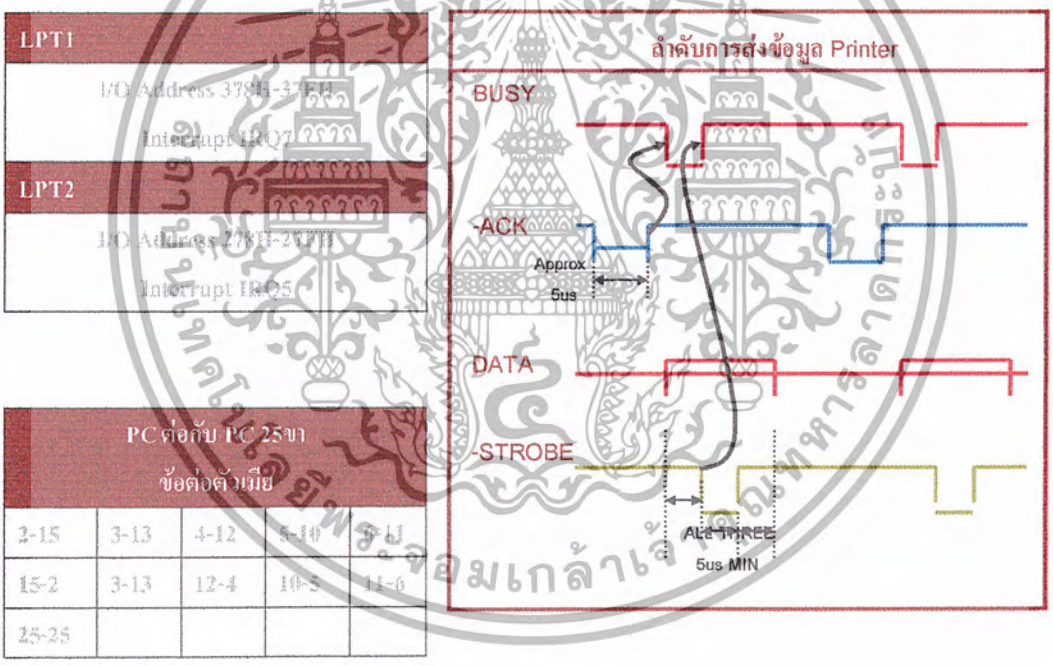
ขาควบคุมจะมีทิศทางการส่งข้อมูลจาก PC ไปยังพรินเตอร์ มีทั้งหมด 4 ขาด้วยกัน คือ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- **/STROBE** (ขา 1) ใช้ส่งสัญญาณถามสถานะพรินเตอร์
- **/AUTOFD** (ขา 14) ใช้สั่งว่าให้ใส่กระดาษเองทีละแผ่นหรือไม่
- **/INIT** (ขา 16) สัญญาณรีเซ็ต
- **/SELIN** (ขา 17) พรินเตอร์จะรับข้อมูลเมื่อสัญญาณเป็น low เท่านั้น



รูปที่ 2.16 ของพอร์ตพาราเรลแบบ 25 ขา



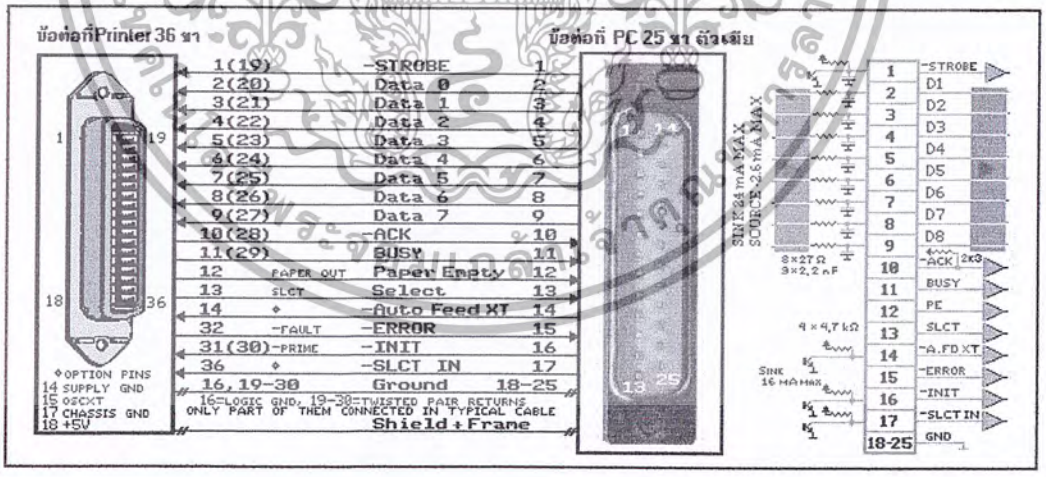
รูปที่ 2.17 แสดงแอดเดรสของพาราเรลพอร์ต และลำดับการส่งข้อมูลของพรินเตอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Pin	Name	Dir	Description
1	/STROBE	→	Strobe
2	D0	→	Data Bit 0
3	D1	→	Data Bit 1
4	D2	→	Data Bit 2
5	D3	→	Data Bit 3
6	D4	→	Data Bit 4
7	D5	→	Data Bit 5
8	D6	→	Data Bit 6
9	D7	→	Data Bit 7
10	/ACK	←	Acknowledge
11	BUSY	←	Busy
12	PE	←	Paper End
13	SEL	←	Select

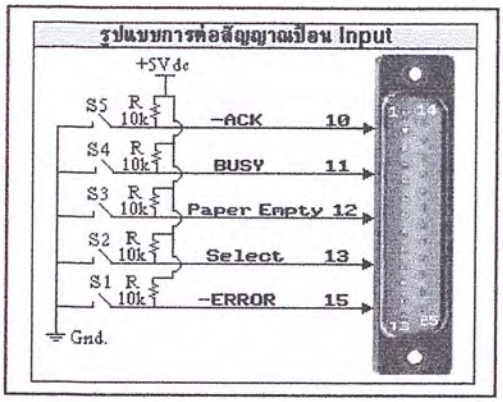
Pin	Name	Dir	Description
14	/AUTOFD	→	Autofeed
15	/ERROR	←	Error
16	/INIT	→	Initialize
17	/SELIN	→	Select In
18	GND	—	Select Ground
19	GND	—	Select Ground
20	GND	—	Select Ground
21	GND	—	Select Ground
22	GND	—	Select Ground
23	GND	—	Select Ground
24	GND	—	Select Ground
25	GND	—	Select Ground

รูปที่ 2.18 หน้าที่และทิศทางของขาทั้งหมดของพาราเรลพอร์ต (ทิศทางออกจาก PC)



รูปที่ 2.19 อินเตอร์เฟสระหว่างข้อต่อที่พรินเตอร์และข้อต่อที่ PC

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.20 รูปแบบการต่อสัญญาณป้อนกลับ

2.3.2 การสื่อสารข้อมูลแบบอนุกรม

การที่อุปกรณ์สองตัวจะสื่อสารกันได้นั้น จะต้องมีการเชื่อมต่อด้วยวิธีใดวิธีหนึ่ง เพื่อให้สัญญาณไฟฟ้าที่ถูกส่งโดยฝ่ายหนึ่งสามารถถูกรับโดยอีกฝ่ายหนึ่งได้ การสื่อสารอาจเกิดขึ้นโดยตรงด้วยการเชื่อมต่ออุปกรณ์สองตัวด้วยสายสัญญาณหรือโดยอ้อมด้วยสื่อกลางที่สอดแทรกเข้ามา สื่อกลางนี้มักจะเป็นระบบโทรศัพท์สาธารณะซึ่งในกรณีนี้ โมเด็ม (Modem) ทำหน้าที่แปลงสัญญาณที่ปลายด้านหนึ่งให้เป็นสัญญาณที่เหมาะสมกับการส่งผ่านสายโทรศัพท์ และเพื่อแปลงสัญญาณกลับที่ ปลายอีกด้านหนึ่งสื่อชนิดอื่น เช่น เส้นใยนำแสงและการส่งผ่านคลื่นวิทยุ สามารถนำมาใช้ได้เช่นกัน อุปกรณ์การสื่อสารจะทำให้คอมพิวเตอร์สามารถสื่อสารโดยใช้สื่อเหล่านี้ เสมือนถูกเชื่อมต่อด้วยวิธีเดียวกับอุปกรณ์อนุกรมปกติ ดังนั้นหลักการที่กล่าวไว้สำหรับการสื่อสารโดยตรงจำสามารถประยุกต์ใช้กับการสื่อสารโดยอ้อมด้วยเช่นกัน โดยต่อไปจะกล่าวถึงการเชื่อมต่ออุปกรณ์สองตัวโดยตรง สายสัญญาณและหัวต่อ (Connector) ที่ต้องใช้และมาตรฐานที่ใช้กันทั่วไปในการเชื่อมต่อสาย

การเชื่อมต่อระหว่างพอร์ตอนุกรมแต่ละพอร์ตมีมาตรฐานอยู่สามแบบ คือ มาตรฐาน EIA RS-442, ระบบวงรอบกระแส และมาตรฐาน EIA RS- 232 แต่เราส่งสัญญาณอนุกรมโดยใช้มาตรฐาน EIA RS- 232 มากที่สุด โดยที่จะมีความยาวของสายได้ไม่เกิน 50 เมตร

2.3.2.1 มาตรฐาน EIA RS-232

มาตรฐาน EIA RS-232 หรือเรียกอีกอย่างว่า RS- 232 กำหนดให้ระดับศักดาไฟฟ้าที่มีค่าเท่ากับ 3 โวลต์หรือสูงกว่า มีค่างตรรกะเป็น 1 กำหนดให้ระดับศักดาไฟฟ้าที่มีค่าเท่ากับ –3 โวลต์หรือต่ำกว่า มีค่าทางตรรกะเป็น 0 วงจรไอซีที่ใช้สร้างสัญญาณเหล่านี้ต้องการใช้แหล่งจ่ายไฟขนาด +12 โวลต์ RS-232 จะใช้สาย 1 เส้นสำหรับส่งข้อมูลและอีก 1 เส้นสำหรับรับข้อมูล ตารางต่อไปนี้แสดงตัวเชื่อมต่อ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

แบบ D ชนิด 25 ขา (25 pin D-connector) และสัญญาณต่างๆของ RS- 232 บ่อยครั้งที่สัญญาณต่างของ RS- 232 จะใช้สาย 1เส้น สำหรับส่งข้อมูลและสายอีกหนึ่งเส้นจะใช้สำหรับรับข้อมูล โดยสัญญาณในแต่ละสายนี้จะถูกอ้างอิงเทียบกับกราวนด์ (ขาเบอร์ 7) มาตรฐาน RS- 232 นี้ยังได้กำหนดสัญญาณ RS- 232 ถึงแม้ว่าบ่อยครั้งที่สายสัญญาณส่ง RS- 232
นี้ได้กำหนดไว้สัญญาณตอบรับเพื่อใช้ในการควบคุมการรับส่งข้อมูลด้วย

ในการเชื่อมต่อแบบ D ชนิด 25 ขา (25 pin D-connector) และสัญญาณต่างๆของ RS- 232 ถึงแม้ว่าบ่อยครั้งที่สายส่งสัญญาณ RS- 232 จะใช้ควบคู่กับตัวเชื่อมต่อแบบ 25 ขานี้ แต่ก็ยังไม่เป็นรูปแบบที่มาตรฐาน เนื่องจากตัวเชื่อมต่อแบบใหม่ที่มีขาเพียง 9 ขา (ดังตารางด้านล่างนี้) มาใช้แทน โดยเราจะพบได้ในเครื่องคอมพิวเตอร์ส่วนบุคคลรุ่นใหม่

มาตรฐาน RS- 232 จะสามารถใช้ส่งข้อมูลได้ไกลที่สุด 50 เมตร ด้วยอัตรา 9600 บิทต่อวินาทีถ้าหากว่าจะส่งข้อมูลมากกว่า 50 เมตรขึ้นไปจะต้องทำการส่งที่เร็วกว่า 9600 บิทต่อวินาที

9 Pin	EIA RS-232 Circuit	CCIT V.24 Circuit	RS-232 Description	Signal type & Direction
5	AB	102	Signal ground/common return	Ground/common
2	BB	104	Received data	Data from DCE
3	AA	103	Transmitted data	Data to DCE
1	CF	109	Received line signal detector	Control from DCE
4	CD	108,2	Data terminal ready	Control to DCE
6	CC	107	Data set ready	Control from DCE
7	CA	105	Request to send	Control to DCE
8	CB	106	Clear to send	Control from DCE
9	CE	125	Ring indicator	Control from DCE

ตารางที่ 2.1 แสดงข้อมูลการเชื่อมต่ออนุกรม 9 ขา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

RS-232C

คุณสมบัติของ RS-232C

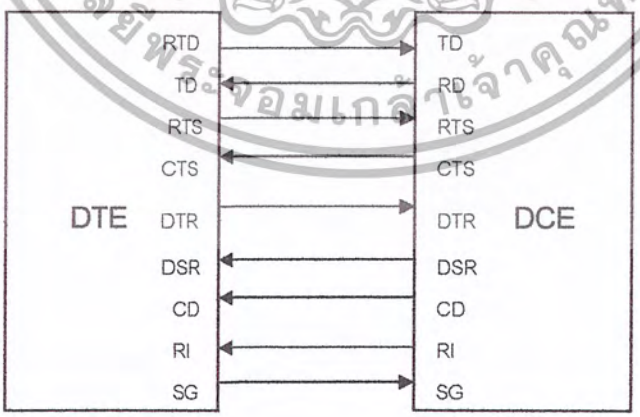
อัตราการรับส่งข้อมูล	: 0-20000 บิต/วินาที
ระดับแรงดันไฟฟ้าเอาต์พุตสูงสุดในภาวะไม่มีโหลด	: -25 โวลต์ (ลอจิก 1)
	+25 โวลต์ (ลอจิก 0)
ระดับแรงดันไฟฟ้าเอาต์พุตสำหรับโหลด 3-7 กิโลโอห์ม	: ลอจิก"1" -15 โวลต์ (7 กิโลโอห์ม)
	-5 โวลต์ (3 กิโลโอห์ม)
	ลอจิก"0" +15 โวลต์ (7 กิโลโอห์ม)
	+5 โวลต์(3 กิโลโอห์ม)
กระแสเอาต์พุตเมื่อลัดวงจร	: สูงสุด 500 มิลลิแอมป์เอาต์พุตอิพีแคนซ์
	เมื่อไม่จ่ายไฟเลี้ยงต่ำสุด 300 โอห์ม
สลูว์เรตทางเอาต์พุตสูงสุด	: 30 โวลต์/ไมโครวินาที
ความต้านทานอินพุตของภาครับ	: สูงสุด 7 กิโลโอห์ม
	ต่ำสุด 3 กิโลโอห์ม
ค่าความจุอินพุตของภาครับ	: สูงสุด 2500 กิโลฟารัด
ย่านแรงดันอินพุตของภาครับ	: -25 โวลต์ ถึง +25 โวลต์

การจัดขาสัญญาณของ RS-232C

มีลักษณะเชื่อมต่อ 2 แบบด้วยกัน

1. การเชื่อมต่อกันระหว่างอุปกรณ์ ดีทีอี (DTE : Data Terminal Equipment) เช่นคอมพิวเตอร์กับอุปกรณ์

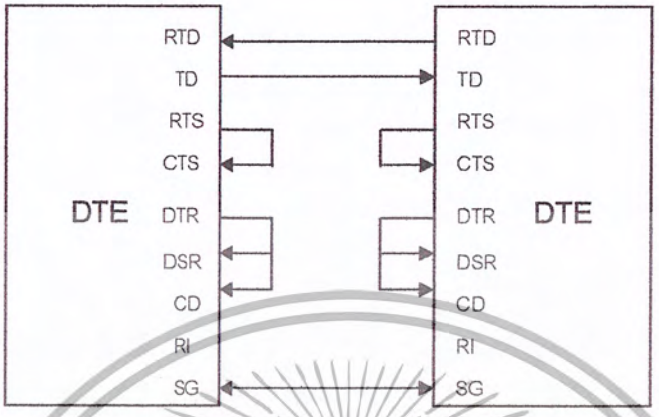
ดีซีอี (DCE : Data Circuit Terminal) เช่น โมเด็ม แสดงดังรูปต่อไปนี้



รูปที่ 2.21 การเชื่อมต่อกันระหว่างอุปกรณ์ DTE กับ DCE

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.การเชื่อมต่อกันระหว่างอุปกรณ์ ดีทีอี (Data terminal Equipment) กับอุปกรณ์ ดีทีอี แสดงดังรูปต่อไปนี้



รูปที่ 2.22 การเชื่อมต่อกันระหว่างอุปกรณ์ DTE กับ DTE

2.3.2.2 รูปแบบของข้อมูลอนุกรมและอัตราบอดในการสื่อสารข้อมูลอนุกรม

อัตราบอด (baud rate) คือความเร็วในการส่งข้อมูลอนุกรม มีหน่วยเป็นบิตต่อวินาที

รูปที่ 2.23 การส่งข้อมูลแบบอนุกรม

เมื่อเราไม่ต้องการให้อุปกรณ์เชื่อมต่อภายนอกเชื่อมต่อโดยตรงกับระบบที่มีไปโครโปรเซสเซอร์เป็นพื้นฐาน เราจะต้องนำออปโตซิเลเตอร์(optoisolator) มาใช้เป็นตัวป้องกันไมโครโปรเซสเซอร์ UART และวงจรอื่นๆในระบบที่มีศักดาไฟฟ้าสูงๆ เพื่อให้กระแสไหลเข้าที่สายกราวนด์

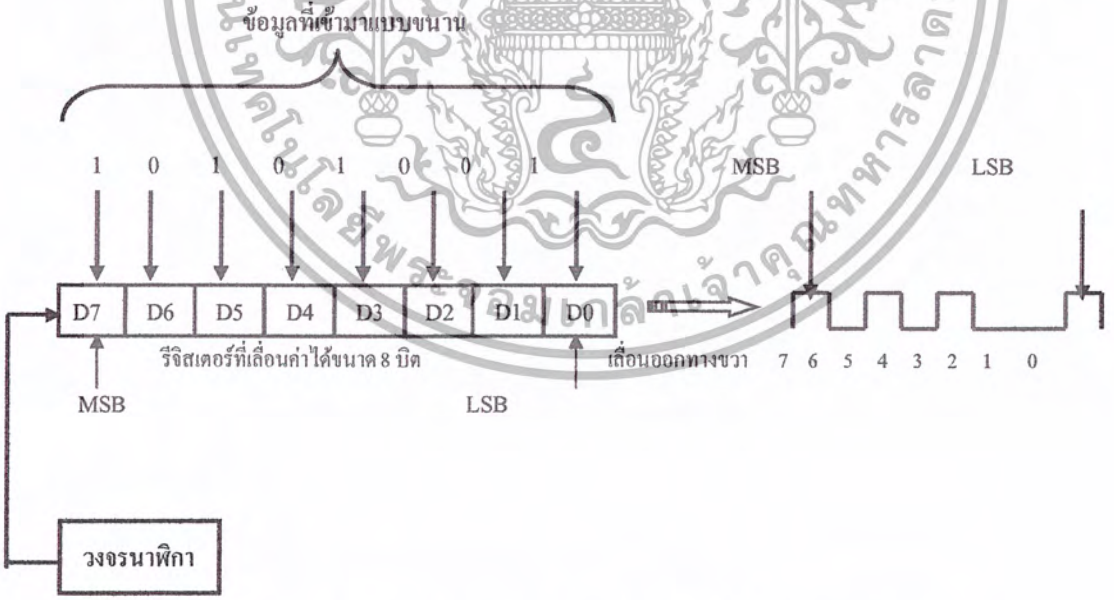
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

อัตราบอดเรท	ช่วงเวลาของแต่ละบิต
110	9.91 ms
150	6.67 ms
300	3.33 ms
600	1.67 ms
1200	.833 ms
2400	.417 ms
4800	.208 ms
9600	.104 ms
19200	.052 ms

ตารางที่ 2.2 อัตราบอดเรท

2.3.2.3 การเชื่อมต่อแบบอนุกรมและ UART

ในรูปด้านล่างนี้ แสดงการแปลงข้อมูลแบบขนานเป็นข้อมูลแบบอนุกรม โดยเริ่มแรกข้อมูลแบบขนานจะถูกนำไปเก็บไว้ในชิฟต์รีจิสเตอร์ (Shift Register) หลังจากนั้นจะใช้สัญญาณนาฬิกาในการเลื่อนค่ารีจิสเตอร์ออกมาทีละบิต โดยเลื่อนค่าไปทางขวามือ บิตแรกที่ถูกเลื่อนออกมาคือบิต LSB ของข้อมูล และบิตที่สองที่ถูกเลื่อนออกมาคือบิตที่อยู่ถัดจาก LSB และบิตต่อๆไป สำหรับบิตสุดท้ายที่ถูกเลื่อนออกมาคือบิต MSB ของข้อมูล



รูปที่ 2.24 การแปลงข้อมูลอนุมกรมไปเป็นข้อมูลแบบขนาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การเปลี่ยนแปลงข้อมูลแบบอนุกรมไปเป็นข้อมูลแบบขนานนั้นจะมีขั้นตอนตรงกันข้ามกับที่กล่าวมา นั่นคือข้อมูลแบบอนุกรมจะถูกเลื่อนเข้าเก็บใน ชิฟต์รีจิสเตอร์โดยใช้สัญญาณนาฬิกาเป็นตัวควบคุม และหลังจากที่ได้มีการเลื่อนข้อมูลทุกบิตเข้าไปใน ชิฟต์รีจิสเตอร์ได้หมดแล้ว ข้อมูลในรีจิสเตอร์นี้จะถูกนำออกเป็นแบบขนานเพื่อนำไปใช้งานต่อไป

อุปกรณ์ที่ทำหน้าที่แปลงข้อมูลแบบอนุกรมเป็นแบบขนานและแปลงข้อมูลแบบขนานเป็นแบบอนุกรม เรียกว่า UART (Universal Asynchronous Receiver-Transmitter) ซึ่งเป็นวงจร LSI ซึ่งนอกจากจะมีหน้าที่ในการเปลี่ยนแปลงข้อมูลแล้ว UART ยังมีหน่วยควบคุมและตรวจสอบการทำงานอีกด้วย

ในการขนส่งข้อมูลขนาด 8 บิตแบบอนุกรมนั้นจะต้องมีบิตสตาร์ต (Start Bit) และบิตสตอป (Stop Bit) เพิ่มขึ้นมาซึ่งจะทำให้ข้อมูลที่ถูกส่งไปจริงๆมีขนาด 10 บิต ในรูปต่อไปนี้ แสดงข้อมูลที่มีขนาด 8 บิต บิตสตาร์ต 1 บิต และบิตสตอป 1 บิต โดยที่บิตสตาร์ตมีค่าเป็น 0 บอก UART ที่ทำหน้าที่รับข้อมูลให้รู้ว่ามีข้อมูลกำลังเข้ามาและบิตสตอปมีค่า 1 บอกให้รู้ว่าการส่งข้อมูลได้เสร็จสิ้นลงแล้ว

รูปที่ 2.25 ข้อมูลอนุกรมขนาด 8 บิต

เราเรียกความเร็วของการส่งข้อมูลแบบ UART ว่าอัตราบอด (Baud Rate) มีหน่วยเป็นจำนวนบิตต่อวินาที (Bit per Second) ซึ่งจะบอกจำนวนบิตที่รับส่งในเวลา 1 วินาที เช่นการส่งข้อมูลด้วยอัตราบอด 1200 บอด ก็คือการส่งข้อมูลด้วยอักขระขนาด 10 บิต (บิตสตาร์ต 1 บิต บิตข้อมูลที่มีขนาด 8 บิต และบิตสตอป 1 บิต) ได้ 120 ตัวอักขระใน 1 วินาที ซึ่งในตารางต่อไปนี้ แสดงอัตราบอดของ UART ที่ใช้กันโดยทั่วไป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

อัตราบอด	ไบต์/วินาที
110	10
150	15
300	30
600	60
1200	120
2400	240
4800	480
9600	960
19200	1920
38400	3840

ตารางที่ 2.3 อัตราบอดเรททั่วไปที่ใช้ในการโอนย้ายข้อมูลแบบอนุกรม

การส่งข้อมูลด้วยอัตรา 110 บอดนั้นจะมีรูปแบบแตกต่างกับอัตราบอดอื่นๆซึ่งในอัตรานี้จะต้องใช้บิต
สตาร์ต 1 บิต และบิตสตอป 2 บิต ดังนั้นจึงทำให้ต้องส่งข้อมูลที่มีขนาด 11 บิต

เมื่อนำบิตที่ 8 ของข้อมูลมาใช้ตรวจสอบความผิดพลาดในการสื่อสารข้อมูล ซึ่งเรียกว่าบิตพาริตี (Parity Bit) UART ส่วนใหญ่สามารถสร้างและทำการตรวจสอบข้อมูลนั้นว่าเป็นพาริตีคู่หรือคี่ได้ ในการสร้างพาริตีคู่ UART จะทำการเซตหรือเคลียร์ค่าในพาริตีเพื่อให้ข้อมูลทั้ง 8 บิต มีตัวเลข 1 เป็นจำนวนคู่และในการสร้างพาริตีคี่ UART จะทำการเซตหรือเคลียร์ค่าในบิตพาริตีเพื่อให้ข้อมูลทั้ง 8 บิต มีตัวเลข 1 เป็นจำนวนคี่

เราสามารถใช้พาริตีคู่หรือคี่ในการตรวจสอบความผิดพลาดในการรับส่งข้อมูล โดยเมื่อ UART ได้รับข้อมูลก็จำทำการทดสอบว่าข้อมูลที่ได้รับมานั้นเป็นพาริตีคู่หรือคี่ ถ้า UART ตรวจสอบพบว่าข้อมูลที่ได้รับมานั้นมีพาริตีไม่ตรงตามที่กำหนด บิตพาริตีในริจิสเตอร์ของ UART จะถูกเซตเพื่อขอให้มีการส่งข้อมูลชุดเดิมนั้นกลับมาใหม่อีกครั้ง

เราสามารถแบ่ง UART ออกเป็นส่วนๆได้ 4 ส่วน ได้แก่ ส่วนที่ทำหน้าที่ส่งข้อมูล ส่วนรับข้อมูล ส่วนกำหนดสถานะ และส่วนที่เป็นวงจรควบคุม

ส่วนที่ทำหน้าที่ส่งข้อมูลของ UART แยกออกได้เป็น 2 ส่วน ซึ่งได้แก่บัฟเฟอร์ส่งข้อมูล (Transmitter data output buffer) กับรีจิสเตอร์ส่งข้อมูล (Transmitter register) โดยบิตข้อมูล 8 บิตจะถูกนำไปเก็บที่บัฟเฟอร์ส่งข้อมูล เมื่อสัญญาณควบคุมดาตาอินพุตสโรตบ (Data input strobe) เปลี่ยนค่าจาก 1 เป็น 0 และการส่งข้อมูลแบบอนุกรมจะเริ่มขึ้นเมื่อสัญญาณนี้เปลี่ยนค่าจาก 0 เป็น 1 หลังจากนั้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รีจิสเตอร์ส่งข้อมูลจะทำหน้าที่เลื่อนข้อมูลส่งออกไปยังเส้นส่งข้อมูลอนุกรม โดยเริ่มจากบิตสตาร์ตถัดมาเป็นบิตข้อมูล D0 ถึง D7 และบิตสตอป

ส่วนที่ทำหน้าที่รับข้อมูลของ UART จะมีการทำงานที่ตรงกันข้ามกับส่วนที่ทำหน้าที่ส่งข้อมูล ข้อมูลที่เข้ามาทางเส้นรับข้อมูลอนุกรม (เข้า) จะถูกเลื่อนเข้าไปเก็บในรีจิสเตอร์รับข้อมูล โดยการเลื่อนค่า 10 หรือ 11 ครั้ง การทำงานจะเริ่มเมื่อบิตสตาร์ตเข้ามา และเมื่อข้อมูลทั้งหมดถูกเลื่อนเข้าไปเก็บในรีจิสเตอร์รับข้อมูล (Receiver register) แล้ว ข้อมูลในรีจิสเตอร์รับข้อมูลจะถูกนำไปเก็บบัฟเฟอร์รับข้อมูล (Receiver-data output buffer) เมื่อมีสัญญาณควบคุมดาตาเอาต์พุตสโตรบ ทั้งรีจิสเตอร์รับข้อมูลและรีจิสเตอร์ส่งข้อมูลจะได้รับสัญญาณนาฬิกาขนาด 16 หรือ 64 เท่าของอัตราบอดที่ใช้ในการเลื่อนค่ารีจิสเตอร์ทั้งรีจิสเตอร์รับข้อมูลและส่งข้อมูลจะได้รับสัญญาณนาฬิกาขนาด 16 หรือ 64 เท่าของอัตราบอดที่ใช้ในการเลื่อนค่ารีจิสเตอร์

รีจิสเตอร์ควบคุมการทำงานของ UART ทำให้เราสามารถกำหนดโหมดการทำงานของ UART ได้ โดยบิตควบคุมจำนวนข้อมูลทั้ง 2 บิตจะช่วยให้เราสามารถเลือกจำนวนของข้อมูลจริงที่รับมาว่าเป็น 5,6,7 หรือ 8 บิต ก็ได้

ในปัจจุบัน UART ถูกนำมารวมกับไอซีพิเศษที่ทำหน้าที่สร้างสัญญาณอัตราบอด โดยไอซีนี้จะสามารถสร้างสัญญาณที่มีอัตราบอดเป็น 16 หรือ 64 เท่าของอัตราบอดมาตรฐาน โดยสัญญาณนี้จะถูกสร้างจากวงจรกำเนิดสัญญาณแบบคริสตอล ทำให้อัตราบอดที่มีอัตราคงที่และเที่ยงตรง

2.4 การประมวลผลภาพเชิงตัวเลข (Digital image Processing)

การเกิดภาพโดยใช้เลนส์เป็นวิธีการพื้นฐานและพบทั่วไปในการถ่ายภาพ แสงที่กระทบวัตถุและสะท้อนกลับมาที่เลนส์จะถูกรวมและนำมาแสดงยังจุดที่สอดคล้องกันกับวัตถุ ดังนั้นกระบวนการถ่ายภาพจึงเป็นการเปลี่ยนข้อมูลของวัตถุอย่าง 3 มิติ มาเป็นข้อมูลภาพ 2 มิติ โดยที่การบันทึกหรือประมวลผลรูปแบบของแสงที่สะท้อนมาจากวัตถุทำได้โดยการใส่เซ็นเซอร์ที่ให้สัญญาณทางไฟฟ้าออกมาในรูปแบบที่ต่อเนื่อง ในขั้นตอนนี้เซ็นเซอร์จะทำการสแกนหรือทำการวัดผลรวมความเข้มของแสงที่จุดเล็กๆ ทีละจุดไปเรื่อยๆ ตามแนวทางที่กำหนดไว้ หรือตามแนวราสเตอร์ (Raster scan) ซึ่งปกติจะไล่จากซ้ายไปขวาและจากบนลงล่าง ค่าที่เซ็นเซอร์วัดได้นี้ถือว่าเป็นค่าความเข้มภาพ f(x,y) ที่พิกัด (x,y) และก็มีความต่อเนื่องด้วย ดังนั้นในการที่จะนำภาพนี้มาประมวลผลโดยคอมพิวเตอร์จำเป็นต้องทำให้ภาพที่มีลักษณะต่อเนื่องนี้กลายเป็นภาพดิจิตอลหรือภาพเชิงตัวเลขเสียก่อน โดยทำการดิจิไตซ์ (digitization) ค่าความเข้ม f(x,y) ซึ่งเป็นฟังก์ชันต่อเนื่องไปเป็นฟังก์ชันไม่ต่อเนื่อง g(x,y) โดยการแบ่ง f(x,y) ออกเป็นช่วงๆ นี้สามารถแทนด้วยค่าตัวเลขค่าใดค่าหนึ่งจาก L ระดับ ซึ่งโดยทั่วไปจุดภาพแต่ละจุด (pixel) จะเป็นสมาชิกของเมตริกซ์ของจุดภาพที่มีขนาด M แถว N หลัก เพราะฉะนั้น x,y จะมีค่าอยู่ในช่วง ($1 \leq y \leq M$, $1 \leq x \leq N$) และจำนวนช่วงระดับความเข้มของจุดภาพ L ระดับนี้จะบ่งบอกถึงความละเอียด (resolution) ของภาพเชิงตัวเลขด้วย ปกติแล้วนิยมกำหนดให้ L มีค่าเท่ากับ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

256 ระดับ ซึ่งจะทำให้ระดับความเข้มของภาพอยู่ในช่วง (0-255) โดยใช้ 8 บิตของเลขฐาน สองในการเก็บข้อมูลภาพแต่ละจุดนั่นเอง แต่ก็ยังมีการเก็บข้อมูลที่มากกว่านี้ได้ คือ อาจจะเป็น 16 หรือ 24 บิต ซึ่งจะทำให้ระดับความเข้มของจุดภาพเป็นไปได้ถึง 2^{16} หรือ 2^{24} ตามลำดับ

รูปที่ 2.26 การแทนสัญญาณอนาลอกด้วยสัญญาณเชิงตัวเลข

จากการที่นำภาพเชิงตัวเลขไปใช้ในการประมวลผลในรูปแบบต่างๆ มากมายนั้น สามารถที่จะแบ่งรูปแบบของการประมวลภาพเหล่านั้นออกเป็น 2 ระดับด้วยกัน คือ การประมวลผลภาพในระดับต่ำ (Low-level Image Processing) และการประมวลผลภาพในระดับสูง (High-level Image Processing)

การประมวลผลภาพในระดับต่ำนั้นเป็นการประมวลผลเชิงตัวเลขเกือบทั้งหมดเพื่อหาตัวแปรต่างๆ มาอธิบายข้อมูลภาพ และมีจุดประสงค์ที่จะนำตัวแปรเหล่านี้ไปใช้ในการประมวลผลภาพในระดับสูงต่อไป แต่การประมวลผลภาพในระดับสูง คือการทำให้คอมพิวเตอร์รู้จักและเข้าใจภาพได้ เช่น การจดจำรูปแบบของตัวเลข ตัวอักษร เป็นต้น ในขณะที่การประมวลผลภาพในระดับต่ำโดยทั่วไปจะประกอบด้วย การกำจัดสัญญาณรบกวน , การทำให้ภาพคมชัดขึ้น, การหาขอบเขตของภาพ, การทำเซกเมนท์ภาพหรือการแบ่งแยกวัตถุภายในภาพ, การสร้างภาพไบนารี เป็นต้น

ความแตกต่างที่สำคัญอีกข้อหนึ่งของการประมวลผลภาพใน 2 มิติ คือ ข้อมูลที่จะนำมาใช้ประมวลภาพ ซึ่งการประมวลผลในระดับต่ำจะใช้ค่าความสว่างหรือระดับความเข้มของจุดภาพโดยตรง ส่วนการประมวลผลภาพในระดับสูงข้อมูลที่จะถูกนำมาประมวลผลจะถูกแสดงในรูปของสัญลักษณ์ โดยสัญลักษณ์เหล่านี้จะแสดงถึงสิ่งต่างๆ ที่อยู่มนภาพ และใช้ตัวแปรที่ได้จากการประมวลผลในระดับต่ำมาอธิบายถึงสัญลักษณ์เหล่านั้น ซึ่งจะเห็นได้ว่าการประมวลผลภาพในระดับต่ำนี้มีความสำคัญมากสำหรับที่จะทำให้คอมพิวเตอร์สามารถเข้าใจและรู้จักภาพได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.4.1 การทำเทรชโชลด์ (Thresholding technique)

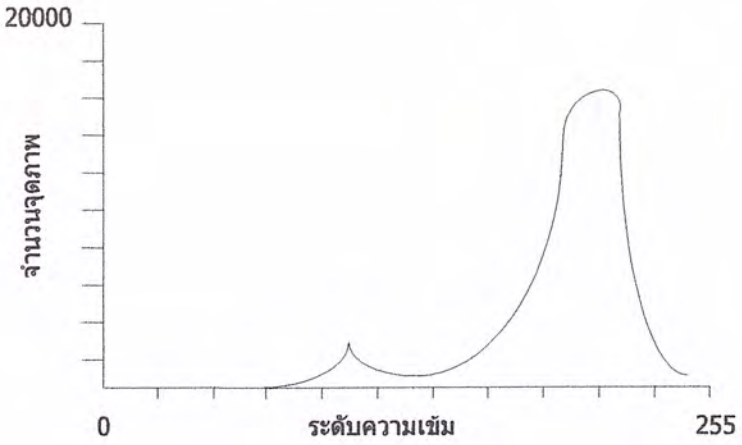
การทำเทรชโชลด์ถือว่าเป็นเทคนิคที่สำคัญในการประมวลผลภาพในส่วนของการทำเซกเมนต์ภาพ ซึ่งจุดประสงค์ของการทำเซกเมนต์ภาพ คือ การแยกองค์ประกอบของภาพไปเป็นส่วนประกอบย่อย ๆ ที่มีความสัมพันธ์กันทางกายภาพของภาพนั้น และส่วนประกอบที่ถูกแยกออกมานั้นอาจถูกนำไปประมวลผลภาพในส่วนอื่นได้ต่อไป ซึ่งการทำเซกเมนต์ภาพจะมีหลักการทำงานในแนวเดียวกันกับสายตาของคน คือ สามารถแยกลักษณะเด่นออกมาจากภาพที่มองเห็นได้ และเทคนิคการทำเทรชโชลด์ซึ่งถือว่าเป็นเทคนิคในการแยกองค์ประกอบของภาพที่ง่ายเทคนิคหนึ่ง มีหลักการว่าจุดภาพที่มีคุณสมบัติอยู่ในบางช่วงใด ๆ จะถูกจัดเป็นกลุ่มได้โดยที่ระดับความเข้มหนึ่งนั้นสามารถที่จะแบ่งแยกกลุ่มของจุดภาพออกเป็น 2 กลุ่มได้อย่างชัดเจน คือ กลุ่มของวัตถุ (object) ซึ่งจะมีระดับความเข้มของภาพ g(x,y) ค่อนข้างต่ำ (มืด) กับกลุ่มของส่วนที่เป็นพื้นหลัง (background) ที่จะมีระดับความเข้มของภาพ g(x,y) ค่อนข้างสูง (สว่าง) ดังเช่นภาพที่ 2.27 ซึ่งแสดง
ฮิสโตแกรมของระดับความเข้มของภาพที่ถูกแบ่งออกเป็น 256 ระดับ จะเห็นได้ว่าการที่จะแยกกลุ่มข้อมูลออกเป็น 2 กลุ่มอย่างชัดเจนย่อมสามารถทำได้โดยการเลือกค่าเทรชโชลด์ที่มีค่าความเข้มอยู่ระหว่างกลุ่มทั้งสองบนฮิสโตแกรมระดับความเข้มของภาพ แล้วทำการตรวจสอบแต่ละจุดภาพว่าถ้ามีค่า g(x,y) น้อยกว่าค่าเทรชโชลด์ถือว่าเป็นจุดภาพของวัตถุที่แสดงได้ด้วยจุดดำ แต่หากว่าจุด g(x,y) นั้นมีค่ามากกว่าหรือเท่ากับค่าเทรชโชลด์ก็ถือว่าเป็นจุดภาพในส่วนพื้นหลังที่แสดงได้ด้วยจุดขาว ดังนั้นข้อมูลภาพ gthr(x,y) ที่ผ่านการทำเทรชโชลด์สามารถนิยามได้ดังนี้

$$g_{thr}(x,y) = \begin{cases} 0 \text{ if } g(x,y) < T \\ 1 \text{ if } g(x,y) \geq T \end{cases} \quad \text{....(2.1)}$$

โดยที่

$g_{thr}(x,y)$	คือ	ข้อมูลภาพผลลัพธ์เป็นไบนารี
g(x,y)	คือ	ข้อมูลภาพอินพุทที่มีระดับความเข้ม 0 ถึง L ระดับ
T	คือ	ค่าเทรชโชลด์ เป็นค่าคงที่ที่มีค่าระหว่าง 0 ถึง L
0	คือ	จุดดำ (ส่วนที่เป็นวัตถุ)
1	คือ	จุดขาว (ส่วนที่เป็นพื้นหลัง)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.27 ฮีสโตแกรมระดับความเข้มของภาพที่มีความเหมาะสม สำหรับการทำเทรชโฮลด์แบบครอบคลุม

จะเห็นได้ว่า การทำเซกเมนต์ภาพโดยใช้เทคนิคการทำเทรชโฮลด์เพื่อให้ได้ภาพผลลัพธ์ที่มีความถูกต้องเหมาะสมนั้นสิ่งที่สำคัญที่สุด คือ ค่าเทรชโฮลด์ที่ใช้นั่นเอง เนื่องจากถ้าเลือกค่าเทรชโฮลด์ที่ไม่เหมาะสมแล้วภาพผลลัพธ์ที่ได้อาจไม่ถูกต้อง ดังนั้นปัญหาของการทำเซกเมนต์ภาพโดยวิธีการทำเทรชโฮลด์นี้ก็คือทำอย่างไรจึงจะสามารถคำนวณหาค่าเทรชโฮลด์ที่เหมาะสมสำหรับภาพแต่ละภาพที่นำมาทำการเซกเมนต์ได้ ซึ่งได้มีผู้เสนอวิธีการในการคำนวณหาค่าเทรชโฮลด์ไว้หลายวิธี ซึ่งแต่ละวิธีย่อมมีความเหมาะสมกับภาพที่แตกต่างกันไป

2.4.1.1 รูปแบบในการทำเทรชโฮลด์

ภาพที่มีระดับความเข้มของจุดภาพของส่วนที่เป็นวัตถุและจุดภาพของส่วนที่เป็นพื้นหลังแตกต่างกันอย่างชัดเจนและมีความสม่ำเสมอตลอดทั้งภาพ สามารถใช้ค่าเทรชโฮลด์เพียงค่าเดียวในการทำเซกเมนต์กับแต่ละจุดภาพทั่วทั้งภาพได้ เรียกการทำเทรชโฮลด์แบบนี้ว่า การทำเทรชโฮลด์แบบครอบคลุม (Globol Thresholding) แต่ถ้าภาพนั้นมีระดับความเข้มไม่สม่ำเสมอเกิดขึ้นในส่วนของวัตถุหรือพื้นหลัง หรือในทั้งสองส่วน การใช้ค่าเทรชโฮลด์เพียงค่าเดียวตลอดทั้งภาพย่อมไม่เหมาะสมกับภาพนั้น ในกรณีนี้ค่าเทรชโฮลด์ที่ดีควรมีการปรับเปลี่ยนค่าไปตามตำแหน่งของจุดภาพนั้นได้ คือ การใช้ค่าเทรชโฮลด์ที่ต่างกันสำหรับจุดภาพที่ตำแหน่งต่างกัน และเรียกการทำเทรชโฮลด์ในลักษณะดังกล่าวนี้ว่า การทำเทรชโฮลด์แบบปรับค่า (Adaptive Thresholding)

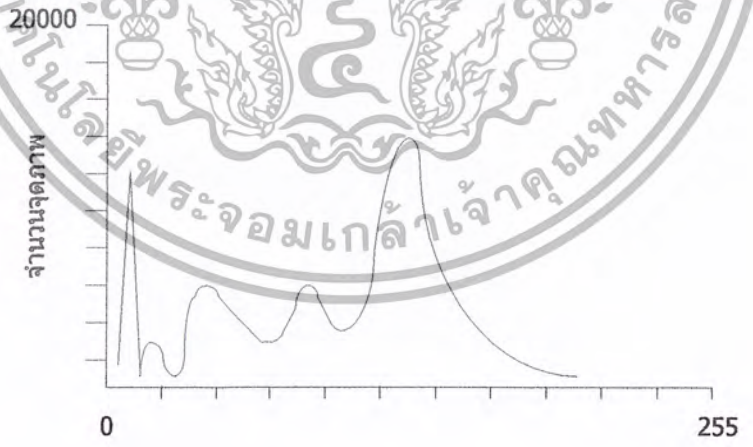
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การทำเทรชโชลด์แบบครอบคลุม (Global Thresholding)

สำหรับขั้นตอนการหาค่าเทรชโชลด์ที่ครอบคลุมตลอดทั้งภาพโดยอัตโนมัติ ปกติจะมีพื้นฐานของการดำเนินการอยู่บนฮีสโตแกรมของระดับความเข้มของจุดภาพ ซึ่งฮีสโตแกรมระดับความเข้มนี้สามารถสร้างได้จากการนับจำนวนของจุดภาพที่มีระดับความเข้มเท่ากับค่าความเข้มที่จุดนั้นทั้งหมดทั่วภาพนั่นเอง จากนั้นจึงทำการหาค่าเทรชโชลด์ในรูปแบบต่าง ๆ ที่สามารถแบ่งฮีสโตแกรมนี้ออกเป็น 2 ส่วน (ส่วนที่เป็นระดับความเข้มของวัตถุ กับส่วนที่เป็นระดับความเข้มของพื้นหลัง) ได้อย่างถูกต้องตรงตามความต้องการ โดยเฉพาะอย่างยิ่ง ในกรณีของภาพที่มีอัตราความแตกต่างของระดับความเข้มระหว่างส่วนที่เป็นวัตถุกับส่วนที่เป็นพื้นหลังมีค่าสูง (แตกต่างกันมาก) และยังมีระดับความเข้มที่เกิดขึ้นในแต่ละส่วนมีความสม่ำเสมอ ย่อมเหมาะสมที่จะใช้ระดับความเข้มที่มีจำนวนของจุดภาพที่ต่ำสุดซึ่งอยู่ระหว่างกลุ่มระดับความเข้มทีมีค่าสูงสุด (peak) ทั้งสองกลุ่มบนฮีสโตแกรมเป็น "ค่าเทรชโชลด์" ดังเช่นตัวอย่างในรูปที่ 2.28 หรือในกรณีทั่ว ๆ ไป ค่าเทรชโชลด์อาจจะพิจารณาจากค่าระดับความเข้มที่สามารถแบ่งฮีสโตแกรมออกเป็น 2 กลุ่ม แล้วทำให้ความแปรปรวนที่เกิดขึ้นระหว่างกลุ่มมีค่ามากที่สุด แต่ความแปรปรวนที่เกิดขึ้นระหว่างกลุ่มมีค่าต่ำสุด หลังจากนั้นนำค่าเทรชโชลด์ที่คำนวณได้ไปทำเทรชโชลด์กับแต่ละจุดภาพทั่วทั้งภาพเพื่อให้ได้ภาพผลลัพธ์ที่เป็นไบนารีในที่สุด

การทำเทรชโชลด์แบบปรับค่า (Adaptive Thresholding)

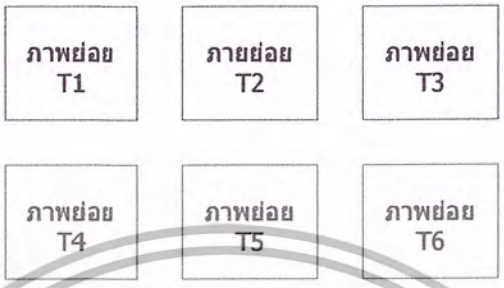
ในกรณีที่ข้อมูลภาพมีความไม่สม่ำเสมอเกิดขึ้นในส่วนของวัตถุ หรือส่วนของพื้นหลัง หรือในทั้งสองส่วน ซึ่งภาพในลักษณะเช่นนี้ ฮีสโตแกรมระดับความเข้มของภาพที่เกิดขึ้นอาจมีลักษณะดังรูปต่อไปนี้



รูปที่ 2.28 ฮีสโตแกรมระดับความเข้มของภาพที่เหมาะสมสำหรับการทำเทรชโชลด์แบบปรับค่า

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การใช้ค่าเทรชโชลด์แบบครอบคลุมเพียงค่าเดียวกับภาพทั้งภาพนี้ อาจทำให้ผลลัพธ์ที่ได้ไม่ถูกต้อง จากปัญหาที่เกิดขึ้นนี้สามารถแก้ไขได้โดยการแบ่งข้อมูลของภาพทั้งภาพออกเป็นภาพย่อย ๆ ที่แสดงได้ดังรูปที่ 2.29 ซึ่งแต่ละภาพย่อยก็จะมีการหาค่าเทรชโชลด์ในรูปแบบที่กำหนดเพื่อหาค่าเทรชโชลด์ที่เหมาะสมสำหรับภาพย่อยนั้น และใช้ค่าเทรชโชลด์ที่ได้ทำการเซกเมนต์กับแต่ละภาพย่อยนั้น ขั้นตอนสุดท้ายคือนำแต่ละภาพย่อยที่ผ่านการทำเซกเมนต์แล้วมารวมกันตามพิกัดเดิม



รูปที่ 2.29 การแบ่งภาพออกเป็นภาพย่อยๆ และหาค่าเทรชโชลด์ในแต่ละภาพย่อย

2.4.1.2 วิธีการหาค่าเทรชโชลด์

ขั้นตอนในการทำเซกเมนต์ภาพโดยใช้เทคนิคเทรชโชลด์เพื่อให้ได้ผลลัพธ์ที่เหมาะสมและคมชัดนั้น สิ่งสำคัญที่สุด คือการหาค่าเทรชโชลด์ เนื่องจากถ้าเลือกค่าเทรชโชลด์ที่ไม่เหมาะสม (ค่าเทรชโชลด์มีค่ามากหรือน้อยเกินไป) หรือภาพที่ได้มีสิ่งรบกวน (noise) เกิดขึ้น แล้วย่อมเป็นผลทำให้ภาพผลลัพธ์ที่ได้ไม่ตรงตามความต้องการ ดังนั้นปัญหาของการสร้างภาพไบนารีโดยวิธีเทรชโชลด์นี้ก็คือ ทำอย่างไรจึงจะสามารถคำนวณหาค่าเทรชโชลด์ที่เหมาะสมสำหรับแต่ละภาพที่จะนำมาทำการเซกเมนต์ ซึ่งได้มีผู้เสนอวิธีการในการคำนวณหาค่าเทรชโชลด์นี้ไว้หลายวิธี โดยแต่ละวิธีก็เหมาะกับลักษณะการทำงานที่แตกต่างกันไป เช่น การหาค่าเทรชโชลด์โดยการกำหนดค่าล่วงหน้า (Preassigned threshold value), การหาค่าเทรชโชลด์จากค่ากลาง (Mid-range threshold value), และการหาค่าเทรชโชลด์โดยพิจารณาจากฮีสโตแกรม (Histogram threshold value) โดยที่แต่ละวิธีสามารถอธิบายได้ดังนี้

การหาค่าเทรชโชลด์โดยการกำหนดค่าล่วงหน้า (Preassigned threshold value)

การหาค่าเทรชโชลด์ด้วยวิธีนี้เป็นวิธีที่ง่ายที่สุด เพราะสามารถหาค่าเทรชโชลด์ได้จากการกำหนดค่าเองจากผู้ใช้ (User) ซึ่งการกำหนดนี้จะขึ้นอยู่กับประสบการณ์ของผู้ใช้นั้น ๆ โดยการเลือกค่าคงที่ค่าหนึ่งที่อยู่ระหว่างค่าต่ำสุดและค่าสูงสุดของระดับความเข้มของข้อมูลภาพอินพุท เช่น ข้อมูลภาพอินพุทมีระดับเทา 256 ระดับ (0-255) ค่าเทรชโชลด์ที่สามารถเลือกได้ก็คือค่าที่อยู่ระหว่างค่า

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

0 ถึง 255 เมื่อเลือกค่าเทรชโชลด์ได้แล้วก็สามารถทำเซกเมนต์ได้โดยใช้สมการที่ 2.1 ทั้งนี้การกำหนดค่าเทรชโชลด์ขึ้นมานี้ เช่น 50, 100, 200 อาจนำมาทดลองทำการเซกเมนต์ภาพก่อน แล้วดูผลลัพธ์ที่ได้จากนั้นจึงเลือกใช้ค่าที่เหมาะสมที่สุดมาใช้งาน

การหาค่าเทรชโชลด์จากค่ากลาง (Mid-range threshold value)

การหาค่าเทรชโชลด์โดยพิจารณาจากค่ากลาง เป็นการหาค่าเทรชโชลด์ที่แตกต่างจากวิธีแรก เนื่องจากเป็นการหาค่าเทรชโชลด์โดยอัตโนมัติโดยไม่ต้องให้ผู้ใช้เป็นผู้ใช้เป็นผู้กำหนด โดยการหาค่าเทรชโชลด์วิธีนี้ได้อาศัยการคำนวณพื้นฐานทางสถิติในเรื่องของการหาค่ากลางแบบที่เป็นค่าเฉลี่ย (Mean) มาประยุกต์ใช้ ค่าเทรชโชลด์ที่คำนวณได้จะเป็นค่าที่ได้จากค่ากลางที่อยู่ระหว่างค่าระดับความเข้มสูงสุด (Maximum level) และหาค่าระดับความเข้มต่ำสุด (Minimum level) ของข้อมูลภาพอินพุท สำหรับการคำนวณค่ากึ่งกลางนี้สามารถคำนวณได้จากสมการ

$$T = \frac{Max[g(x,y)] + Min[g(x,y)]}{2} \quad \text{....(2.2)}$$

โดยที่

T คือ ค่าเทรชโชลด์

G(x,y) คือ ข้อมูลภาพอินพุท ที่มีระดับความเข้ม 0 ถึง L ระดับ

Max [g(x,y)] คือ ค่าสูงสุดของระดับเทาของข้อมูลอินพุท

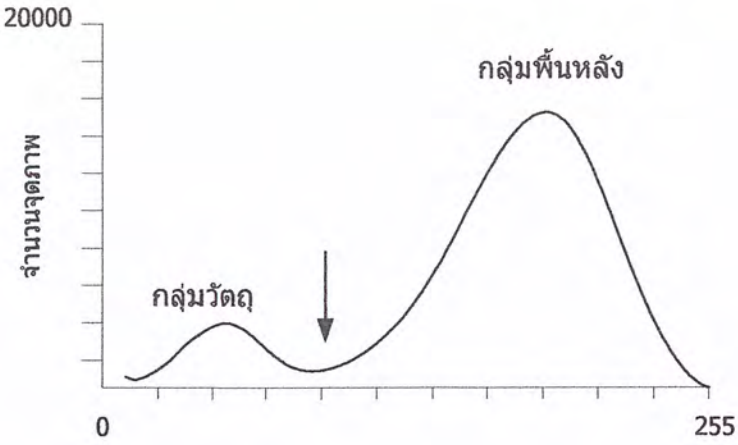
Min [g(x,y)] คือ ค่าต่ำสุดของระดับเทาของข้อมูลอินพุท

เมื่อคำนวณค่าเทรชโชลด์ได้แล้ว ก็สามารถทำการเซกเมนต์ภาพได้โดยนำค่าเทรชโชลด์ที่ได้มาแทนค่าในสมการที่ 2.1

การหาค่าเทรชโชลด์โดยพิจารณาจากฮีสโตแกรม (Histogram threshold value)

การหาค่าเทรชโชลด์โดยวิธีพิจารณาจากฮีสโตแกรมระดับเทาของข้อมูลอินพุท โดยที่การหาค่าเทรชโชลด์วิธีนี้ ข้อมูลภาพอินพุทที่เหมาะสมต้องมีลักษณะที่สามารถแบ่งแยกเป็นสองกลุ่มได้อย่างชัดเจน คือ กลุ่มหนึ่งจะเป็นกลุ่มของวัตถุ และอีกกลุ่มหนึ่งเป็นพื้นหลัง ซึ่งแนวคิดในการคำนวณค่าเทรชโชลด์โดยวิธีนี้สามารถแสดงได้ดังรูป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.30 แนวความคิดในการคำนวณค่าเทรชโชลด์โดยวิธีพิจารณาจากฮีสโตแกรม

สำหรับการคำนวณหาค่าเทรชโชลด์โดยวิธีพิจารณาจากฮีสโตแกรม สามารถสรุปเป็นขั้นตอนได้ดังนี้

1. ข้อมูลภาพอินพุทมีระดับความเข้ม 0 ถึง L ระดับ
2. คำนวณหาฮีสโตแกรมระดับความเข้มของภาพหน้าเอกสาร โดยการนับจำนวนจุดภาพที่ระดับความเข้มแต่ละระดับ
3. จากฮีสโตแกรมจะพบว่าจะเกิดกลุ่มของระดับความเข้มสูงสุด (Peak) 2 กลุ่ม กลุ่มหนึ่งคือกลุ่มของวัตถุ และอีกกลุ่มหนึ่งคือกลุ่มของพื้นหลัง
4. เลือกค่าที่ต่ำที่สุด (Valley) ที่อยู่ระหว่างสองกลุ่มนั้น กำหนดค่านี้เป็นค่าเทรชโชลด์
5. ทำการเซกเมนต์ภาพ โดยนำค่าเทรชโชลด์ที่ได้มาแทนค่าในสมการที่ 2.1

จากขั้นตอนการทำงานข้างต้น ถ้าได้ทดลองทำกับภาพ 256 ระดับเทาที่เป็นภาพตัวอักษรบนพื้นกระดาษที่สีอ่อนกว่า ซึ่งเมื่อทำการสร้างฮีสโตแกรมระดับเทาของภาพแล้วจะปรากฏกลุ่มของข้อมูลสูงสุด (Peak) 2 กลุ่ม คือกลุ่มของข้อความ และกลุ่มของพื้นหลัง จากนั้นทำการเลือกค่าต่ำสุด (Valley) ระหว่าง 2 กลุ่มนั้นเป็นค่าเทรชโชลด์ ซึ่งค่าที่ได้จากวิธีนี้จะมีค่าที่เที่ยงตรงที่สุด แต่อย่างไรก็ดีวิธีนี้ไม่เหมาะสมกับภาพที่ไม่สามารถแยกกลุ่มของสิ่งที่อยู่ในภาพได้อย่างชัดเจนระหว่างกลุ่มของวัตถุและกลุ่มของพื้นหลัง เนื่องจากว่าถ้าหากภาพอินพุทไม่สามารถแยกแยะได้แล้วจะทำให้ค่าเทรชโชลด์ที่คำนวณได้ผิดไปจากความเป็นจริง คืออาจจะมากหรือน้อยเกินไป อันเป็นผลทำให้ได้ภาพที่ไม่เหมาะสม รายละเอียดบางส่วนขาดหายไปได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.4.2 โมเดลสีแบบ RGB

โมเดลสีชนิดนี้ประกอบด้วยการรวมกันของแม่สีหลักซึ่งได้แก่ แดง (Red:R), เขียว (Green:G) และน้ำเงิน (Blue:B) ซึ่งค่าสีต่าง ๆ ในแถบสเปคตรัมของสีจะได้มาจากการผสมกันในอัตราส่วนที่แตกต่างกันของแม่สีทั้งสาม โมเดลสี RGB นี้จะแสดงด้วยแกนของลูกบาศก์สามแกนในระนาบ 3 มิติ ซึ่งค่าสีแดง เขียว และน้ำเงินจะอยู่ที่มุมทั้งสามของแต่ละแกนดังแสดงด้วยรูปที่ 2.31 ซึ่งจะเห็นว่าค่าสีดำจะอยู่ที่จุดกำเนิด (origin) สีขาวจะอยู่ที่มุมตรงข้ามกับสีดำ ค่าของสีในช่วงระดับเทาจะอยู่ตามเส้นที่เชื่อมระหว่างค่าสีดำและค่าสีขาว จากรูปถ้าเป็นในระบบการแสดงสีแบบ 24 บิต (แบ่งออกเป็น 8 บิตต่อแม่สีหนึ่งสี) นั้น ค่าสีแดงจะถูกแทนด้วยค่า (255, 0, 0) เป็นต้น

รูปที่ 2.31 แสดงรูปลูกบาศก์สีของโมเดลแบบ RGB

โมเดลสี RGB ง่ายต่อการออกแบบและใช้งานในระบบคอมพิวเตอร์กราฟฟิค แต่ไม่เหมาะสำหรับการประยุกต์ใช้งานอื่น ๆ ค่าของสีแดง สีเขียว และสีน้ำเงิน จะมีความสัมพันธ์กันอย่างมากซึ่งจะเป็นการยากที่จะนำไปประมวลผลเกี่ยวกับภาพ ดังนั้นหลายครั้งที่มีความจำเป็นที่จะต้องทำการแปลงภาพจากโมเดลสี RGB ให้อยู่ในรูปแบบของภาพแบบระดับเทา (Gray scale image) เพื่อความสะดวกดังกล่าว

ในการแปลงภาพจากโมเดลสี RGB ให้อยู่ในรูปแบบภาพระดับเทานั้น สามารถทำได้โดยการใช้สมการการแปลงดังนี้

$$\text{Gray scale intensity} = 0.299R + 0.587G + 0.114B \quad \ldots(2.4)$$

ซึ่งเป็นสมการที่ใช้สำหรับการแปลงภาพจากมาตรฐาน NTSC

$$\text{Gray scale intensity} = 0.333R + 0.333G + 0.333B \quad \ldots(2.5)$$

ซึ่งเป็นสมการที่ใช้ในการแปลงจาก RGB ไปเป็น HIS

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

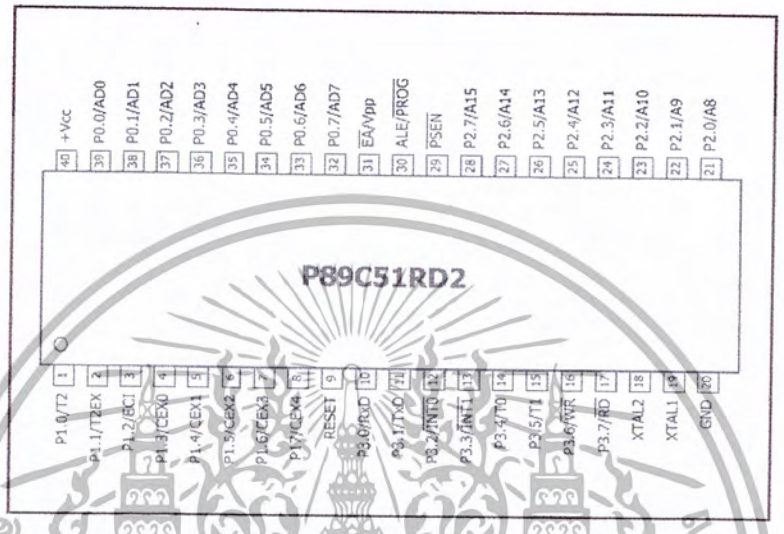
ในระบบดิจิตอลวีดีโอนั้น เรากล่าวถึงจำนวนบิตที่ใช้ในการสุ่มตัวอย่างสัญญาณแต่ละครั้ง เป็นตัววัดความละเอียดของภาพที่ได้เป็นจำนวนบิตต่อพิคเซล (bit per pixel : bpp)

เนื่องจากในการสุ่มตัวอย่างสัญญาณแต่ละครั้งนั้นจะได้เป็นหนึ่งพิคเซล โดยในระบบ monochrome ที่มีคุณภาพสูงนั้นจะใช้ 8 bpp ซึ่งหมายถึงจำนวนระดับเทา (gray level) ของภาพเท่ากับ 256 ระดับ แต่ในระบบการแสดงผลแบบสีนั้นเราต้องการหนึ่งช่องสัญญาณแบบ monochrome ต่อแม่สีแต่ละสี (แดง, เขียว และน้ำเงิน) ซึ่งจะต้องใช้จำนวนบิตทั้งหมดเป็น 24 บิต ซึ่งจะได้ค่าระดับสีที่เป็นไปได้คือ 16,777,216 สี ในระบบดิจิตอลวีดีโอบางระบบซึ่งจะใช้ 24 บิตต่อพิคเซล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 3
การออกแบบ

3.1 การออกแบบ วงจรควบคุมรถใช้ MCS-51



รูปที่ 3.1 ไมโครคอนโทรลเลอร์(MCS-51)

นำไมโครคอนโทรลเลอร์เบอร์ P89C51RD2 มาใช้เป็นตัวหลักในการส่งข้อมูลเพื่อควบคุมการทำงานของวงจรส่วนอื่นๆ โดยแต่ละพอร์ทมีหน้าที่ดังนี้

พอร์ท 1 เป็นพอร์ท INPUT รับข้อมูลจากเครื่องรับที่ส่งผ่านมาจากคอมพิวเตอร์ และรับสัญญาณจากวงจรรับเซนเซอร์ เข้ามาประมวลผลเพื่อควบคุมการทำงานของมอเตอร์

P1.0 - P1.3 รับข้อมูลจากคอมพิวเตอร์ มี เดินหน้า ถอยหลัง เลี้ยวซ้าย และเลี้ยวขวา ซึ่งจะเชื่อมต่อกับ โมดูลของเครื่องรับวิทยุไร้สาย

P1.4 – P1.7 รับสัญญาณจากวงจรเซนเซอร์ ซึ่งจะเชื่อมต่อกับ วงจรรับอินฟราเรด ซึ่งที่ตัวรถได้ติดตั้งไว้มีด้วยกัน 3 ชุด ข้างหน้า 2 ชุด ข้างหลัง 1 ชุด โดยแต่ละชุดจะมีการต่อตัวรับอินฟราเรด ไว้ 2 ตัว

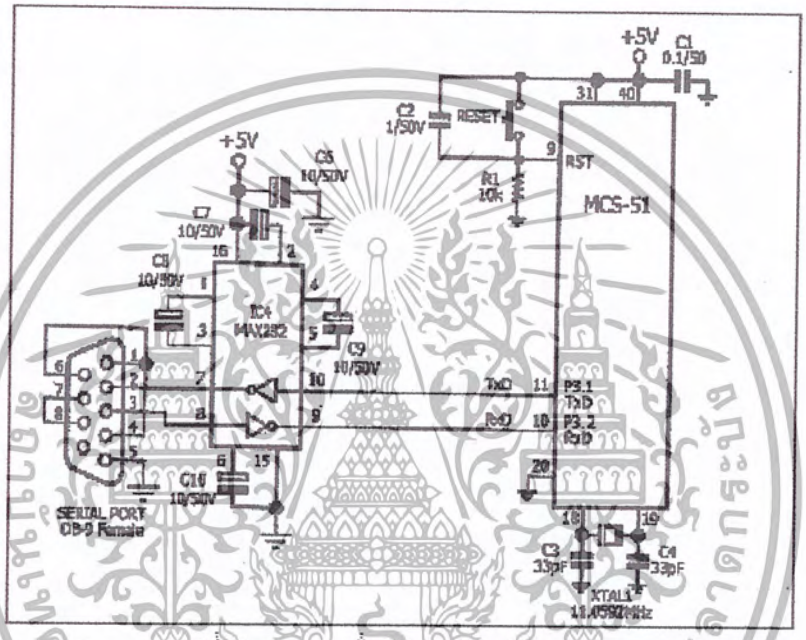
พอร์ท 2 เป็นพอร์ท OUTPUT ส่งข้อมูลไปยัง วงจรขับดีซีมอเตอร์ เพื่อควบคุมการหมุนของดีซีมอเตอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

พอร์ท 3 เป็นพอร์ทที่ใช้รับข้อมูลจากพอร์ทอนุกรมของคอมพิวเตอร์ เมื่อมีการโปรแกรมการทำงานลงบนตัวไมโครคอนโทรลเลอร์ และ ทำหน้าที่รับข้อมูลเพื่อมาควบคุมการทำงานของตัวรถเมื่อต้องการควบคุมรถแบบใช้สาย

P3.1 เป็นขา TxD ต่อกับขา Tx จากพอร์ทอนุกรมของคอมพิวเตอร์

P3.2 เป็นขา RxD ต่อกับขา Rx จากพอร์ทอนุกรมของคอมพิวเตอร์ ดังรูปที่ 3.2

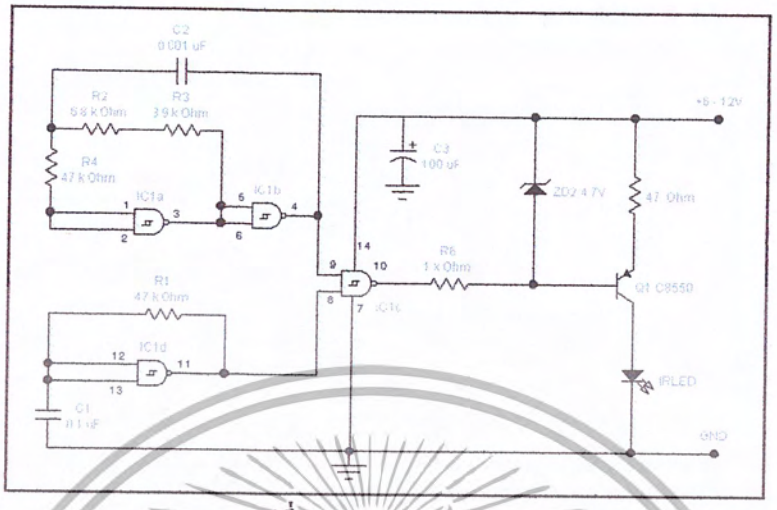


รูปที่ 3.2 การต่อเพื่อใช้งานกับพอร์ทอนุกรม

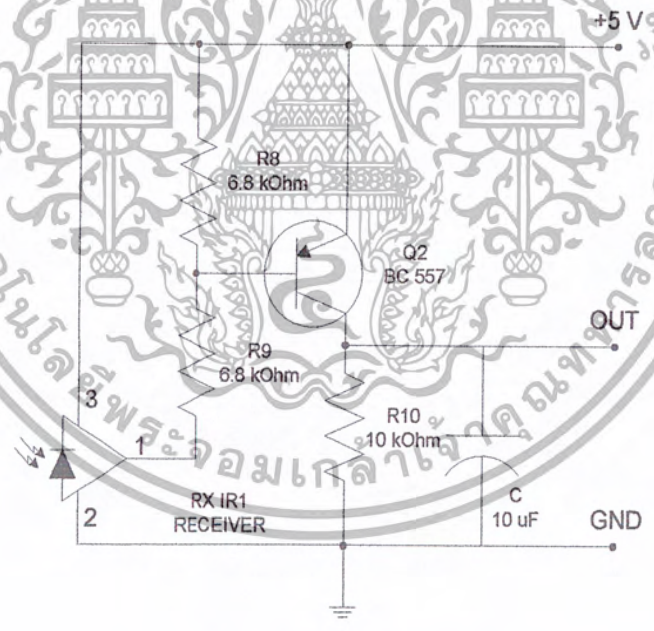
3.2 วงจรตรวจสอบการชน

วงจรตรวจสอบการชนโดยใช้ ระบบแสงอินฟราเรด จากรูปที่ 3.3 เป็นวงจรในภาคส่งใช้ไดโอดเปล่งแสงย่านความถี่อินฟราเรด IRLED(Infrared Light Emitting Diode) โดยมันโดยมันจะเปล่งแสงออกมาในย่านความถี่ที่คนเรามองไม่เห็น วงจรในภาคส่งจะทำงานเมื่อ IRLED ได้รับไบอัสตรง เมื่อมีวัตถุผ่านเข้ามาสะท้อนคลื่นกลับมา ยังตัวรับอินฟราเรดของวงจรภาครับ(รูปที่ 3.4) วงจรก็จะส่งสัญญาณไปตัวไมโครคอนโทรลเลอร์ทันที

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.3 วงจรส่ง อินฟราเรด



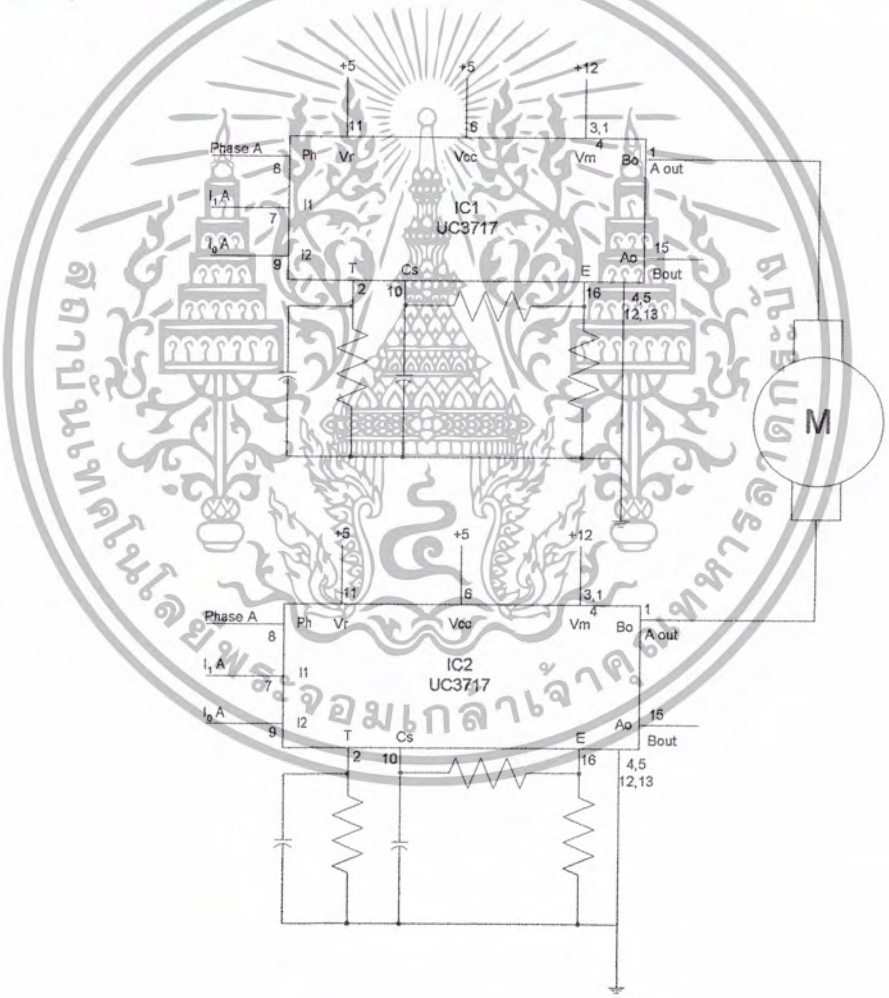
รูปที่ 3.4 วงจรรับอินฟราเรด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.3ส่วนการขับเคลื่อน

ในส่วนของการขับเคลื่อนตัวรถ จากเทอมที่แล้วเนื่องจากประสบปัญหาเรื่องกำลังของมอเตอร์ที่ใช้ ทำให้ตัวรถไม่สามารถ เคลื่อนที่ได้อย่างมีเสถียรภาพ ในเทอมนี้ได้ทำการเปลี่ยนจาก STEP MOTOR มาเป็น DC MOTOR ซึ่งจะช่วยให้ตัวรถมีกำลังขับเคลื่อนที่ดีกว่าแบบ STEP โดยในส่วนของวงจรที่ใช้ในการ คอนโทรลความเร็วนั้นได้ทำการประยุกต์ใช้วงจรที่ใช้ในการขับ STEP MOTOR ซึ่งใช้ IC เบอร์ UC3717 มาทำการคอนโทรล DC MOTOR ได้ โดยการใช้ PORT OUTPUT ที่ขา 1 ของ IC แต่ละตัว และทำการ ต่อทำการ pull down ขาใดขาหนึ่งของอินพุท

การควบคุมความเร็วจะควบคุมที่ความกว้างของพัลซ์ เพื่อให้เกิดแรงดันไฟเฉลี่ยใน 1 คาบเวลา หรือ วิธีการพัลซ์วิดมอด โดยจะทำการเขียนโปรแกรมหน่วงเวลาในไมโครคอนโทรลเลอร์ เพื่อเป็น อินพุทให้กับ IC EA3717



รูปที่ 3.5 รูปวงจรขับมอเตอร์

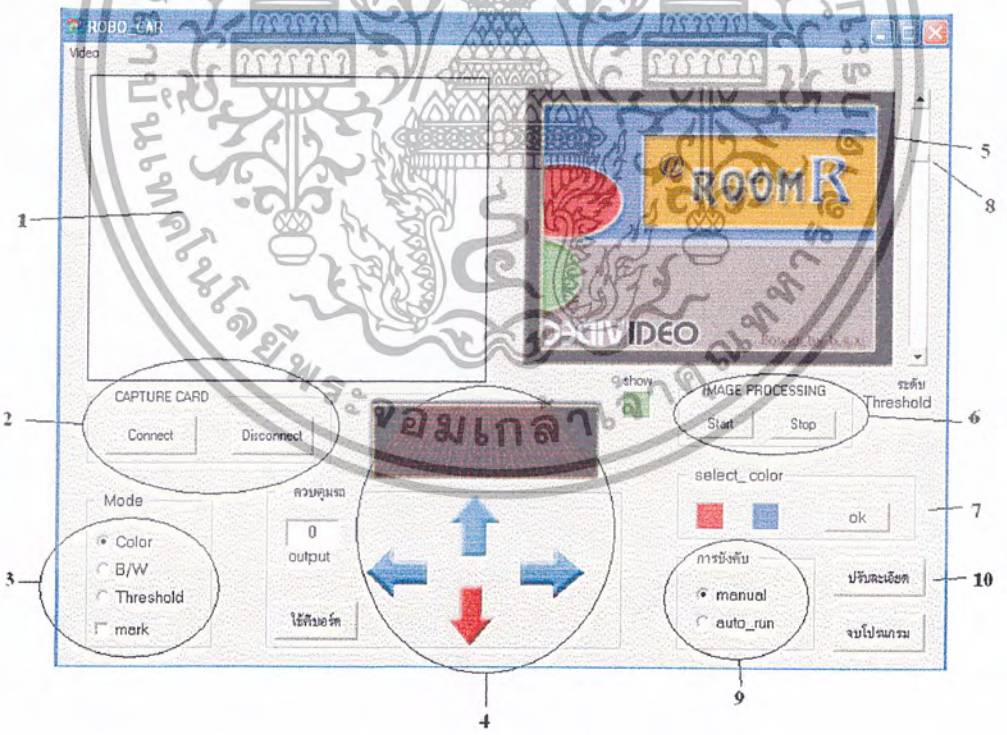
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ข้อมูลจากพอร์ต P2 ไมโครคอนโทรเลอร์	ทิศทาง
00	เดินหน้า
0F	ถอยหลัง
0E	เดินหน้า+ซ้าย
07	เดินหน้า+ขวา

ตารางที่ 3.1 ค่าจากพอร์ท P2 ของไมโครคอนโทรเลอร์

3.4 ส่วนหน้าตาโปรแกรมควบคุมจากคอมพิวเตอร์

การทำงานของรถจะถูกบังคับด้วยคำสั่งที่สั่งมาจากโปรแกรม โดยคอมพิวเตอร์ซึ่งโปรแกรมที่ใช้ จะใช้ vitual basic เป็นโปรแกรมดังรูป



รูปที่ 3.6 แสดงรูปหน้าต่างโปรแกรม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

<u>ส่วนประกอบต่างๆของโปรแกรมมีดังนี้</u>

ส่วนที่ 1 เป็นส่วนแสดงผลของ videoocx ที่จะทำการติดต่อดึงภาพจาก capture card

ส่วนที่ 2 เป็นปุ่ม ที่ใช้เมื่อต้องการติดต่อกับ capture card เพื่อแสดงผลภาพจากกล้องวีดีโอ

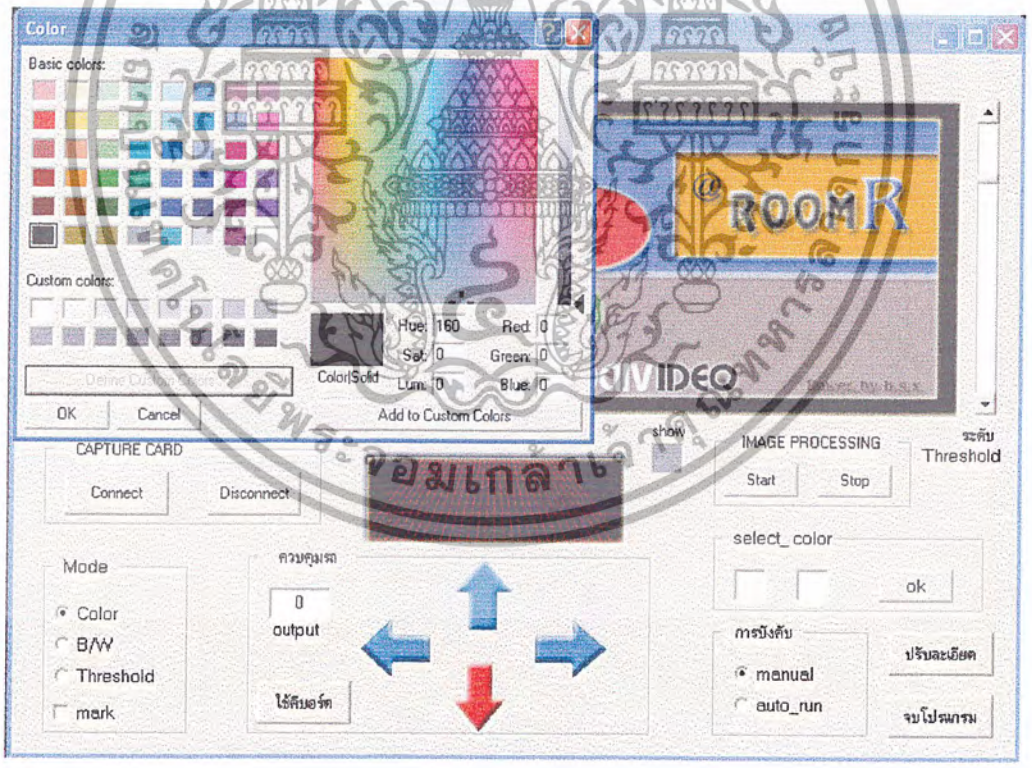
ส่วนที่ 3 เป็นส่วนที่ใช้กำหนดรูปแบบของภาพที่จะนำมาใช้ในการประมวลผลมีด้วยกัน 3 แบบคือ สี , ขาวดำ และ เทรชโชลด์

ส่วนที่ 4 ใช้สำหรับการบังคับการเคลื่อนที่ของรถด้วยมือ และแสดง การส่งสัญญาณออกทางพอร์ทขนาน

ส่วนที่ 5 เป็นส่วนของภาพที่จะทำการประมวลผล

ส่วนที่ 6 ปุ่มเริ่มการประมวลผลด้วยภาพ

ส่วนที่ 7 จะมีช่องสี่เหลี่ยม 2 ช่องเป็นส่วนที่ใช้กำหนดช่วงของสีที่ต้องการ การเลือกสีทำได้โดยการคลิ๊กเลือกจากภาพที่ใช้ประมวลผล(ส่วนที่ 5) ได้โดยตรงหรือจาก หน้าต่างสีของ Windows



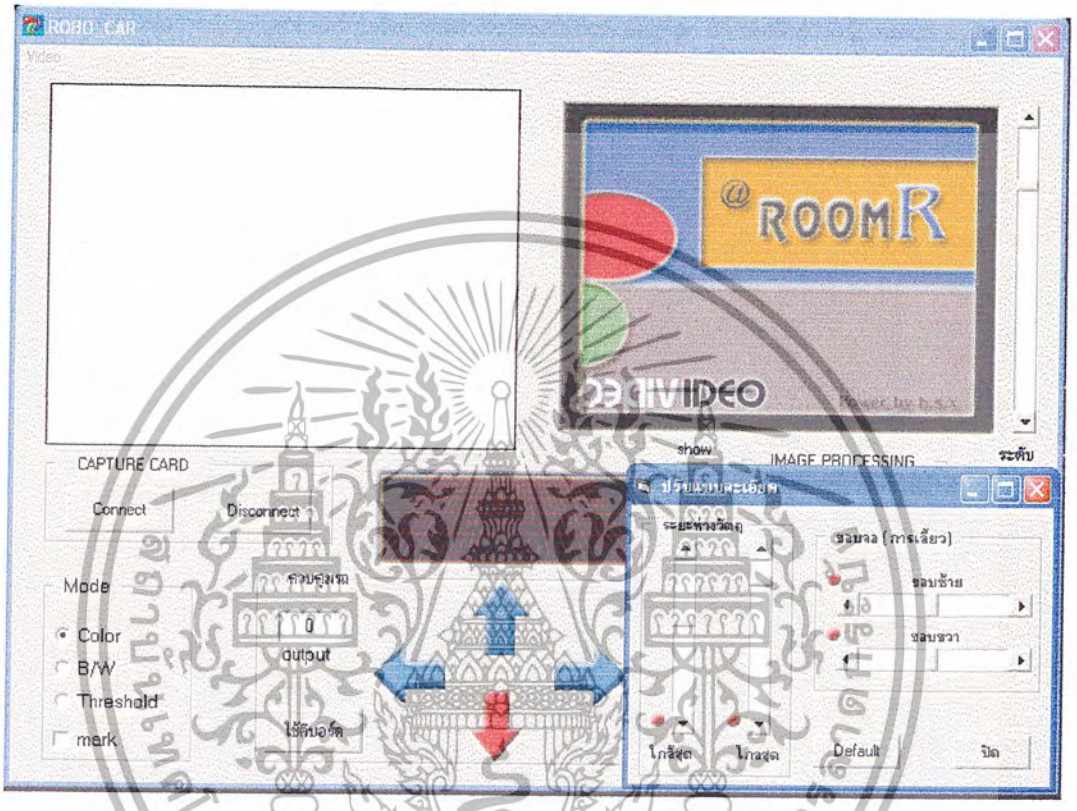
รูปที่ 3.7 แสดงรูปโปรแกรมสีของ windows

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ส่วนที่ 8 ใช้สำหรับปรับค่าเทรชโชลด์ เมื่อทำงานอยู่ใน เทรชโชลด์โหมด

ส่วนที่ 9 ใช้เลือกการบังคับรถว่าจะใช้แบบคนบังคับหรือ แบบใช้ การภาพประมวลผลให้รถวิ่งเอง

ส่วนที่ 10 ปุ่มเลือกใช้หน้าต่างปรับละเอียด เกี่ยวกับการประมวลผล

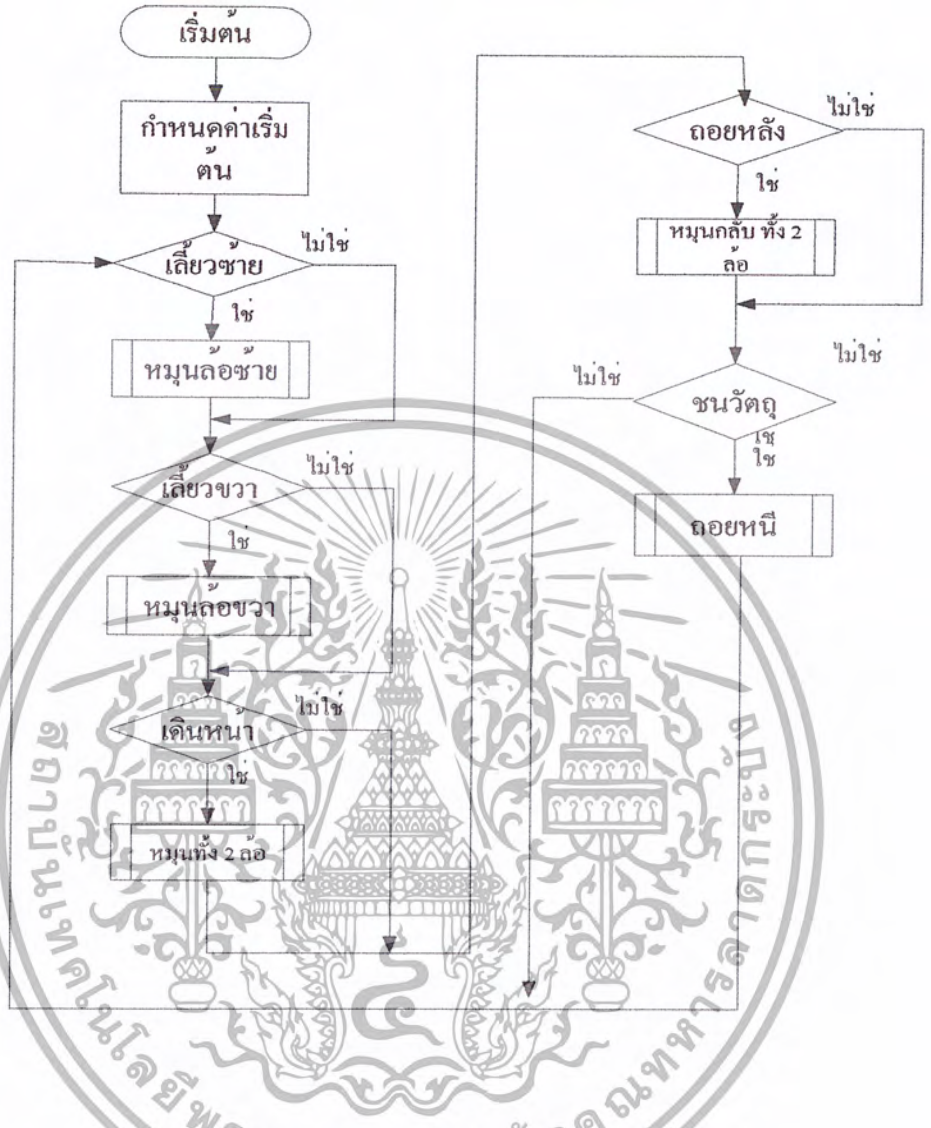


รูปที่ 3.8 แสดงรูปหน้าต่างโปรแกรมการปรับค่า

3.5 ส่วนของโปรแกรมภายในตัวไมโครคอนโทรลเลอร์

ในส่วนโปรแกรมหลัก จะทำงานโดยการตรวจสอบสถานะ ของพอร์ท ต่างๆ ว่ามีการเปลี่ยนแปลงอย่างไร โดยจะมีด้วยกันหลักๆ คือ การตรวจสอบ การเลี้ยวซ้าย เลี้ยวขวา เดินหน้า ถอยหลัง และการเลี้ยวแบบใช้ตำแหน่งการเลี้ยว ร่วมกับ สัญญาณจากวงจรรับเซนเซอร์ มาร่วมในการประมวลผล และ ทำการควบคุมการทำงานของมอเตอร์ ทั้ง 2 ข้าง ดังรูปที่ 3.8

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

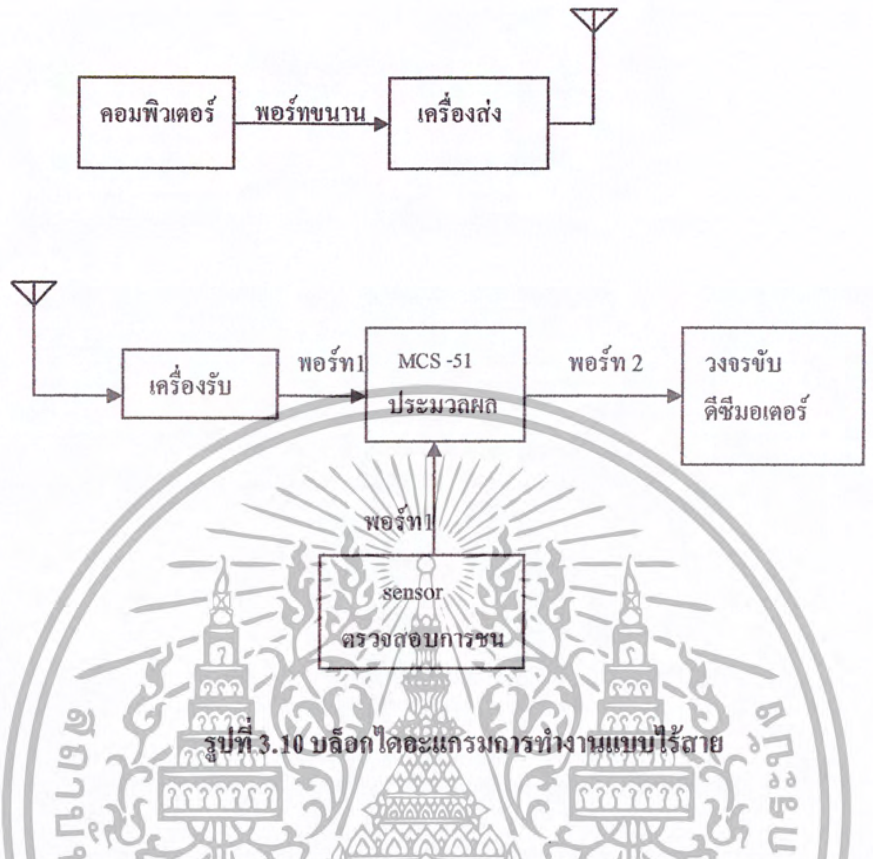


รูปที่ 3.9 Flow chart แสดงการทำงานหลักของโปรแกรม

3.6 การทำงาน

การทำงานคอมพิวเตอร์จะส่งสัญญาณผ่านพอร์ทขนานเข้าไปใน เครื่องส่งสัญญาณวิทยุมอดดูเลท แล้วส่งออกเสาส่ง ทางด้านตัวรถจะมีวงจรรับอยู่เมื่อรับสัญญาณได้ จะทำการดีมอดดูเลทจะได้ข้อมูลออกมา ส่งเข้าไปยังไมโครคอนโทรลเลอร์ประมวลผลต่อไป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.10 บล็อกไดอะแกรมการทำงานแบบไร้สาย

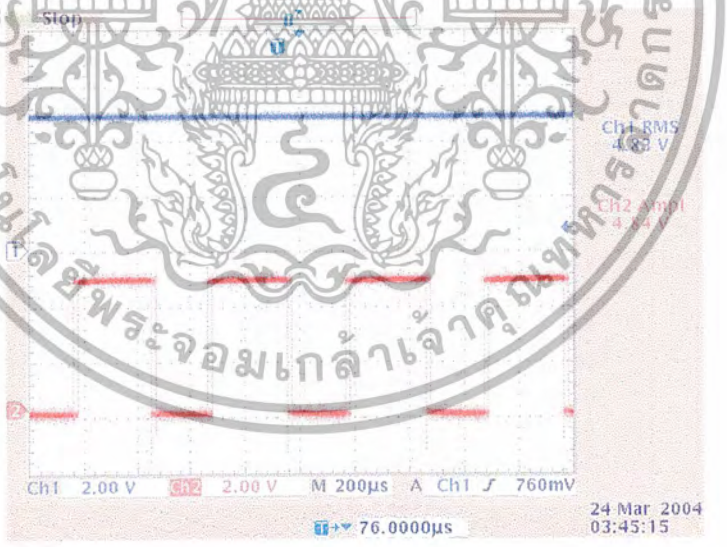
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 4
การทดลองและผลการทดลอง

4.1 สัญญาณที่ส่งไปขับดีซีมอเตอร์

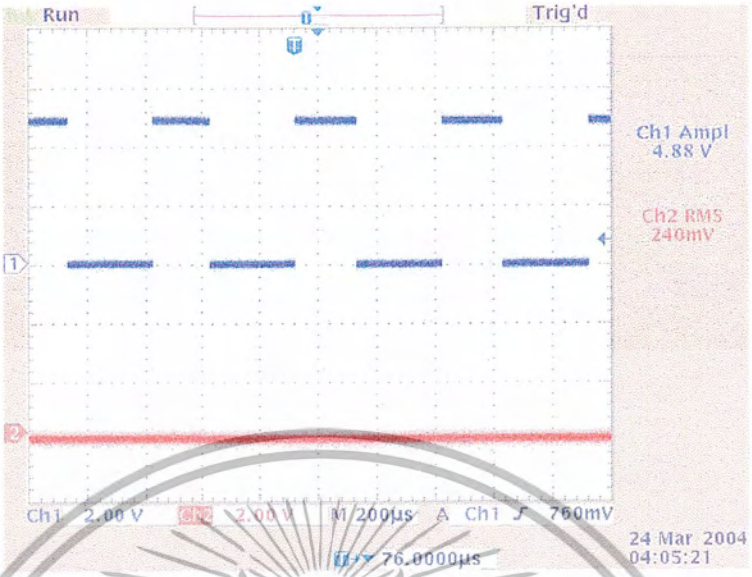
ในการทดลองโครงงานนี้จะใช้วิธีพัลส์วิดมอด โดยการเขียนโปรแกรมควบคุมความเร็ วโดยการกำหนดความกว้างของพัลส์ แล้วทำให้เกิดการเฉลี่ยแรงดัน ซึ่งค่าความเร็วของมอเตอร์ จะถูกกำหนดด้วยพัลส์ ส่วนทิศทางจะถูกกำหนดด้วย แรงดันอ้างอิง โดยที่สัญญาณ ไมโครคอนโทรเลอร์จะส่งไปยังวงจรขับมอเตอร์ ด้วย IC UC3717 โดยที่ IC เบอร์นี้สามารถจ่ายกระแสให้กับมอเตอร์ได้สูงสุด 1.5 A ซึ่งก็เพียงพอกับขนาดของมอเตอร์ และขนาดของตัวรถ ซึ่งมีขนาดไม่ใหญ่มากนัก

ผลของความเร็วของมอเตอร์จะมีผลในการขับสัญญาณภาพ IMAGE PROCESSING ซึ่งถ้ามอเตอร์มีความเร็วมากจะทำให้โปรแกรมประมวลผลภาพไม่ทัน ทำให้ไม่สามารถล็อกเป้าหมายได้ โดยในโครงงานนี้ สามารถปรับความเร็วของมอเตอร์ได้ด้วยการ สั่งจากส่วนควบคุมที่หน้าจอคอมพิวเตอร์ โดยลักษณะสัญญาณที่ใช้ในการควบคุมมอเตอร์มีลักษณะดังรูป

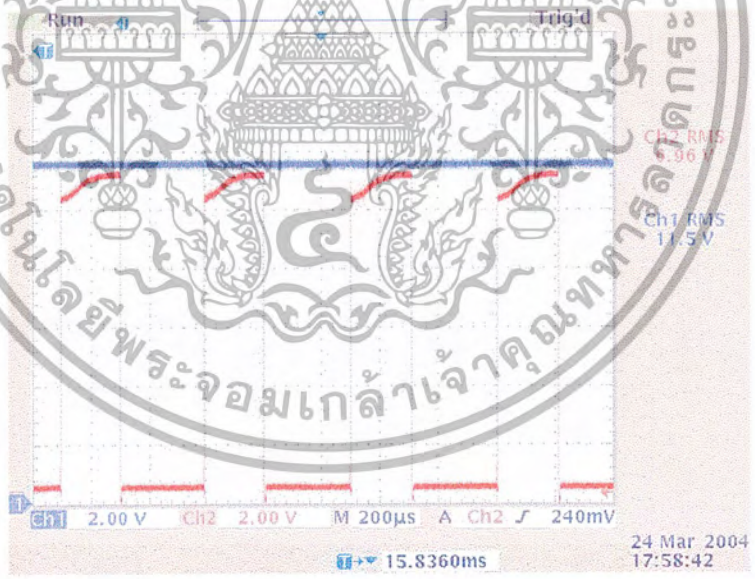


รูปที่ 4.1 กราฟแสดงลักษณะสัญญาณพัลส์ สั่งเดินหน้า

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

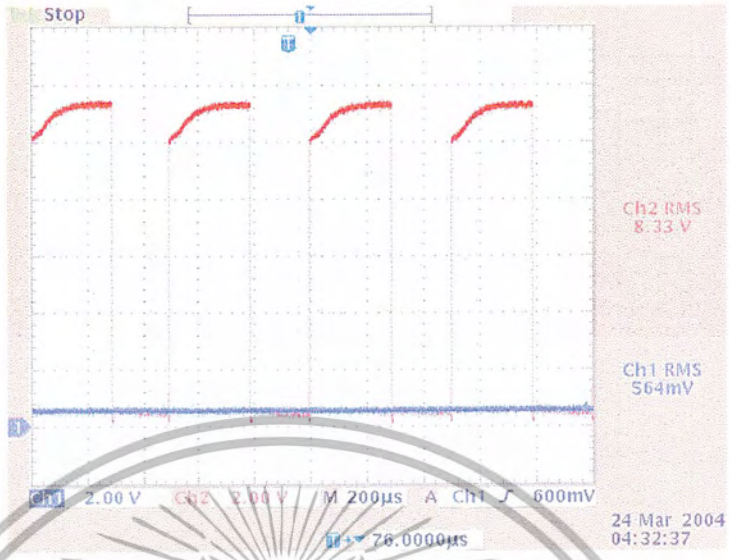


รูปที่ 4.2 กราฟแสดงลักษณะสัญญาณพัลส์ สั่งถอยหลัง



รูปที่ 4.3 กราฟแสดงลักษณะสัญญาณพัลส์ เอาท์พุทจากวงจรขับมอเตอร์ สั่งเดินหน้า

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.4 กราฟแสดงสัญญาณพัลส์ เอาท์พุทจากวงจรขับมอเตอร์ สั่งถอยหลัง

4.2 ระยะความไกลของการบังคับแบบไร้สาย

ตัวรับส่งวิทยุและสัญญาณภาพที่ใช้ มีความสามารถในการรับส่ง ได้ในระยะ 10 เมตร

4.3 เซนเซอร์

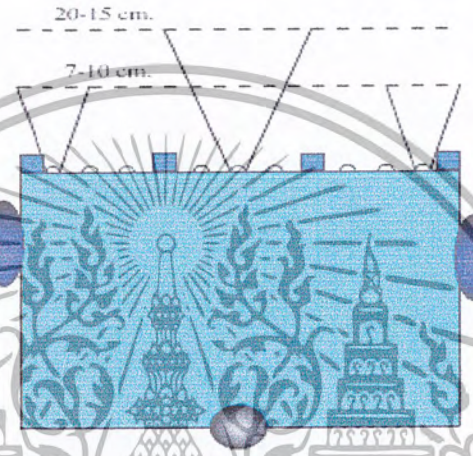
1.1 ระยะห่างการชน หรับการตรวจสอบระยะห่างของการชน ได้ทำการนำวัสดุต่างๆ มาใช้เพื่อทดสอบความสามารถ ในการตรวจจับวัตถุชนิดต่างๆ ดังนี้

วัตถุที่ตรวจสอบ	ระยะ (ซ.ม.)
กระดาษสีขาว	16
แผ่นสแตนเลสตัน	20
ปกวิทยานิพนธ์	7
กล่องพลาสติกผิวเรียบใส	15

ตารางที่ 4.1 แสดงความสามารถในการตรวจสอบสิ่งกีดขวางของวงจรเซนเซอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1.2 องศาที่ (ความกว้าง) ครอบคลุมของการตรวจจับความกว้างที่ครอบคลุมการตรวจจับวัตถุที่จะชน จะอยู่ในระยะแนวตรงแนวเดียวกับ ตัวรับอินฟาเรดซึ่งความสามารถในการสะท้อนกลับของอินฟาเรดจะมีไม่นักดังนั้น มุมองศาในการวัดจึงมีน้อย ซึ่งจากการทดลองนำตัวรับส่งอินฟาเรดมาส่งหากันในลักษณะตั้งฉากตรงกัน จะสามารถส่งได้ไกลถึง 5 เมตร ซึ่งมีค่ามากกว่าการสะท้อนอย่างมาก แต่ในการทำโครงงานนี้ ความสามารถในการตรวจจับสิ่งกีดขวางต้องการเพียงไม่ให้ชนซึ่งค่าที่ได้เฉลี่ยคือ 10 ซ.ม. ก็เพียงพอ



รูปที่ 4.5 แสดงระยะการตรวจจับสิ่งกีดขวาง

1.3 ความสูงของระยะตรวจจับ

ความสูงของวัตถุที่สามารถตรวจจับได้ คือ 5 ซม. ขึ้นไป

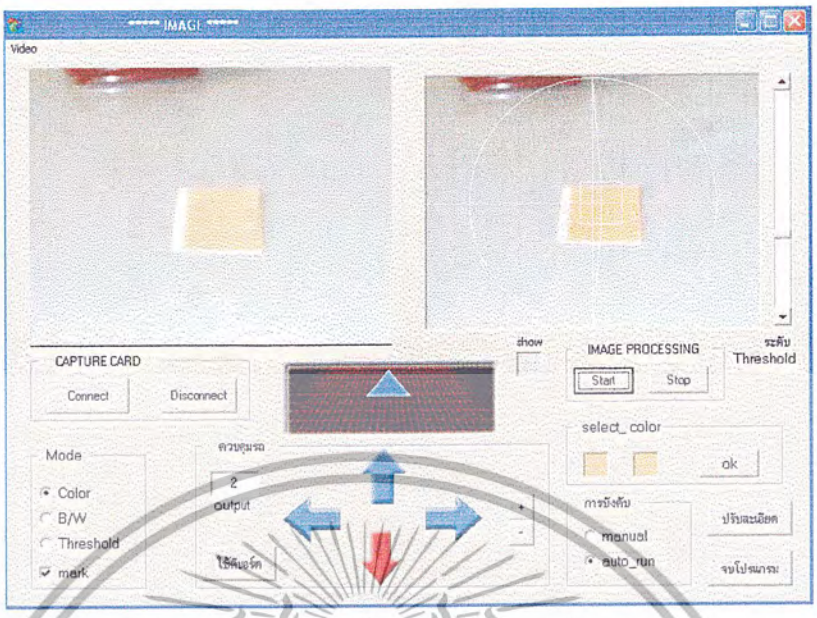
4.4 การเคลื่อนที่ตามวัตถุ

4.4.1 นำวัตถุที่ต้องการใช้ปืนเป้าหมายมาวางใช่ตำแหน่งที่กล้องสามารภจับภาพได้

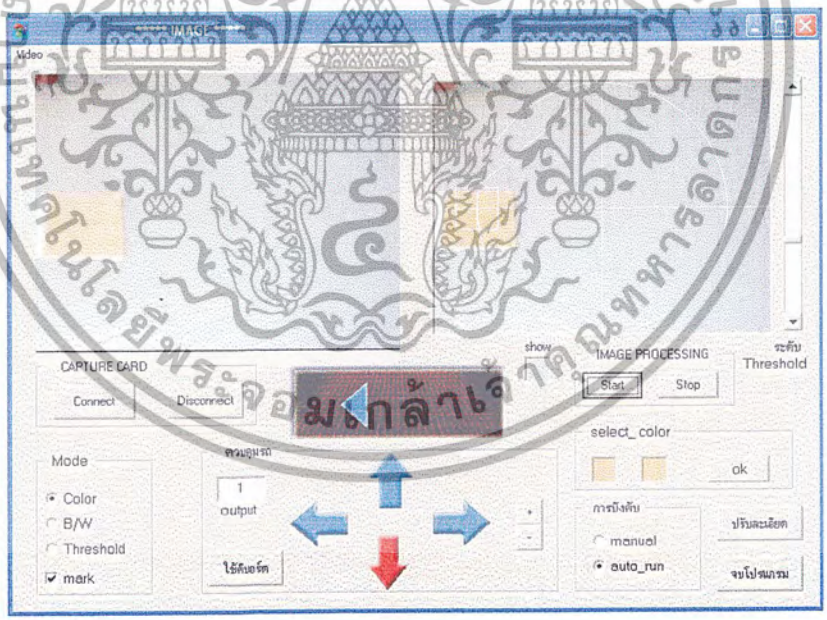
4.4.2 ทำการเลือกช่วงของสีให้ตรงกับวัตถุ

4.4.3 เลือก auto_run ที่ตัวโปรแกรม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



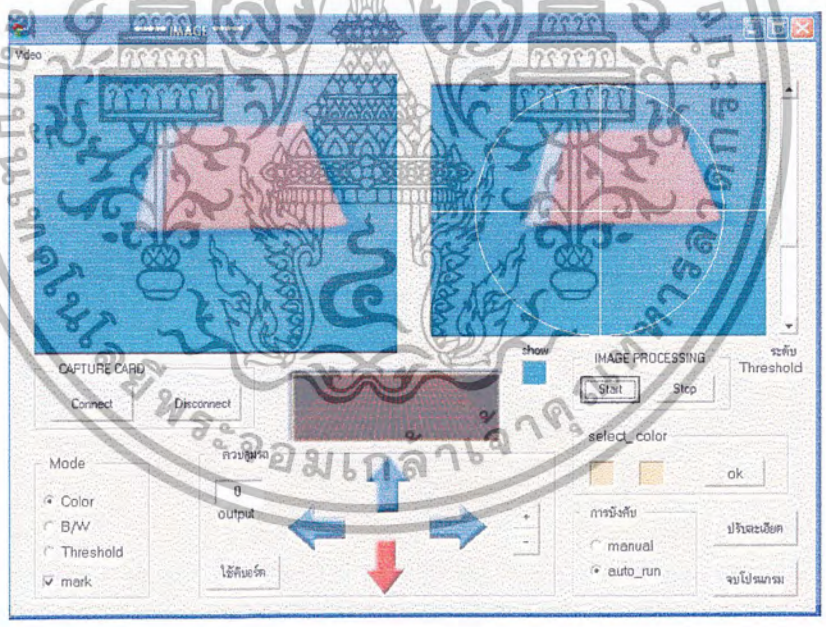
รูปที่ 4.6 แสดงการจับภาพวัตถุและส่งสัญญาณให้รถเคลื่อนที่เข้าหา



รูปที่ 4.7 แสดงการจับภาพวัตถุและส่งสัญญาณให้รถเลี้ยวซ้าย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 4.8 แสดงการจับภาพวัตถุและส่งสัญญาณให้รถเลี้ยวขวา



รูปที่ 4.9 แสดงความ เพี้ยนของ white balance ของกล้องเมื่อระยะห่างระหว่างจากรถและวัตถุเปลี่ยนแปลง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.4.4 จากการทดลองผลที่เกิดจากความเพี้ยนของ white balance ของตัวกล้องทำให้ค่าสีของวัตถุเป้าหมายผิดไปจากช่วงสีที่กำหนดไว้ ทำให้รถไม่สามารถติดตามวัตถุได้ถูกต้องซึ่งเกิดจาก

1. การเปลี่ยนแปลงของแสง
2. ระยะห่างระหว่างรถกับวัตถุเปลี่ยนแปลงไป เช่น รถเข้าใกล้วัตถุเกิน 30- 60 ซ.ม.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 5
สรุปและวิจารณ์

5.1 ปัญหาที่เกิดขึ้นกับโครงงาน

5.1.1 ด้านฮาร์ดแวร์ (Hardware)

1. วงจรป้องกันการชนวัตถุใช้ อินฟราเรด เป็นตัวตรวจจับวัตถุทำให้ระยะการตรวจจับขึ้นอยู่กับพื้นผิววัตถุและสีของวัตถุ ทำให้เกิดความไม่แน่นนอนและมีปัญหากับวัตถุที่มีสีดำ

2 .ในการออกแบบส่วนล้อนั้น ล้อหลังเป็นล้อที่หมุนได้อย่างอิสระทำให้การเคลื่อนที่ของรถเกิดการส่ายขึ้นเวลาปรับการเลี้ยวเพื่อหาตำแหน่งวัตถุ

3. คุณภาพของกล้องที่ใช้ไม่ดีเท่าที่ควรทำให้มีปัญหาเรื่องกล้องปรับ white balance ของตัวกล้องเองช้ามากซึ่งทำให้สีเกิดการผิดพลาดจากช่วงที่กำหนด

5.1.2 ด้านซอฟแวร์ (Software)

1. โปรแกรมที่ใช้เขียนใช้โปรแกม visual basic และใช้ส่วนของคอมโปรแนน videoocx เป็นตัวดึงภาพจากการ์ด capture ซึ่งดาวโหลดมาเพิ่มและ code ของมันใช้ได้เพียง 14 วันดังนั้นการเขียนโปรแกรมต้องเขียนและแก้ให้เสร็จแล้วทำเป็นนามสกุล .exe ภายใน 14 วัน

5.2 แนวทางการแก้ไข

5.2.1 ด้านฮาร์ดแวร์ (Hardware)

1. เปลี่ยนวงจรตรวจจับวัตถุใหม่ เช่นเปลี่ยนไปใช้ อัลตราโซนิก

2. เปลี่ยนไปใช้กล้องที่รับได้แสงน้อยๆ และปรับ white balance ได้ดีเพื่อความถูกต้องในการประมวลผล

3. ออกแบบตัวรถและล้อใหม่ให้วิ่งได้ถูกต้องขึ้นหรือเพิ่มส่วนการตรวจสอบการหมุนของล้อ

5.1.2 ด้านซอฟแวร์ (Software)

พัฒนาส่วนของโปรแกรมการประมวลผลให้สามารถจับสีของวัตถุได้แม่นยำขึ้นและทำงานได้เร็วขึ้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เอกสารอ้างอิง

1. ฉัททวุฒิ พีชผล, พิชิต สันติกุลานนท์ , “Visual Basic 6” , Provision , 2542
2. ผศ.ธีรวัฒน์ ประกอบผล, “การพัฒนาไมโครคอนโทรลเลอร์ด้วยภาษาซี” ,สมาคมส่งเสริมเทคโนโลยี(ไทย-ญี่ปุ่น) ,2545

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้