รถเอนกประสงค์

(Multipurpose Vehicle)

โดย

นางสาวกรรณิการ์ พรหมหิตาธร

นายเพชร พงษ์พานิช

นายวิเชษฐ์ สัตนา

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

-1 ตค 2541

วัน เดือน ปี.....

เลขทะเบียน.....038375.....

เลขเรียกหนังสือ.....T 39942 ก1731

รายงานนี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตร์บัณฑิต

สาขาอิเล็กทรอนิกส์

คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2539

รถเอนกประสงค์

(Multipurpose Vehicle)

โดย

นางสาวกรรณิการ์ พรหมหิตาธร

นายเพชร พงษ์พานิช

นายวิเชษฐ์ สัตนา

อาจารย์ที่ปรึกษา

อาจารย์พลศาสตร์ เลิศประเสริฐ

รายงานนี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตร์บัณฑิต

สาขาอิเล็กทรอนิกส์

คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2539

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รายงานปีการศึกษา 2539

ภาควิชาอิเล็กทรอนิกส์

คณะวิศวกรรมศาสตร์สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เรื่อง รถเอนกประสงค์ (Multipurpose Vehicle)

ผู้จัดทำ

1. นางสาวกรรณิการ์ พรหมหิตาธร รหัส 36014005
2. นายเพชร พงษ์พานิช รหัส 36014301
3. นายวิเชษฐ์ สัตนา รหัส 36014405

.....

(อาจารย์พลศาสตร์ เลิศประเสริฐ)

อาจารย์ที่ปรึกษา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เรื่อง รถเอนกประสงค์

(Multipurpose Vehicle)

ผู้จัดทำ

1. นางสาวกรรณิการ์	พรหมหิตาธร	รหัส 36014005
2. นายเพชร	พงษ์พานิช	รหัส 36014301
3. นายวิเชษฐ์	สัตนา	รหัส 36014405

โครงงานได้รับการตรวจสอบแล้ว พร้อมที่จะทำการสอบได้

.....

(อาจารย์พลศาสตร์ เลิศประเสริฐ)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รถเอนกประสงค์(Multipurpose Vehicle)

กรรณิการ์ พรหมหิตาธร
เพชร พงษ์พานิช
วิเชษฐ์ สัตนา
อาจารย์พลศาสตร์ เลิศประเสริฐ อาจารย์ที่ปรึกษา
ปีการศึกษา 2539

บทคัดย่อ

โครงงานนี้นำเสนอรถเอนกประสงค์(Multipurpose Vehicle) ซึ่งสามารถขับเคลื่อนเข้าสู่เป้าหมาย และสามารถหลบหลีกสิ่งกีดขวางได้อัตโนมัติ โดยใช้การควบคุมด้วยไมโครคอนโทรลเลอร์ ตัวรถบังคับมีขนาด 57*63 เซนติเมตร โดยส่วนของฮาร์ดแวร์ประกอบด้วย วงจรไดรเวอร์ใช้ในการขับเคลื่อน และวงจรเซนเซอร์ใช้ในการตรวจสอบเป้าหมายและสิ่งกีดขวาง นอกจากนี้ได้มีการประยุกต์นำเข็มทิศอิเล็กทรอนิกส์มาใช้ในการควบคุมทิศทางของรถด้วย โดยในการทำงานจะมีซอฟแวร์เป็นตัวควบคุม จากโครงงานนี้สามารถนำรถบังคับไปใช้ประโยชน์ได้ เช่น การติดตั้งกล้องโทรทัศน์กับตัวรถเพื่อเคลื่อนที่ไปสำรวจบริเวณที่มนุษย์ไม่สามารถเข้าไปได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Multipurpose Vehicle

Kunniga Promchitaton

Petch Pongpanit

Vichate Satana

Pollasat Leartprasheart Advisor

1996

Abstract

This project is Multipurpose Vehicle(MPV) which can drive to the target and go far away from the barriers automatically and microcontroller is used. The dimension of the car is about 57*63 centimeters. The parts of the hardware are driver circuit is used to drive and a sensor circuit is used to check the target and the barriers which can control by software and the jyroelectronics can use to control the way of the car. This project can apply by installation of television camera on it for discovery somewhere that human beings can't.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ

	หน้า
บทที่ 1	**1**
บทที่ 2	**2**
2.1 ไมโครคอนโทรลเลอร์(Microcontroller) 8031	2
2.1.1 คุณลักษณะพื้นฐานของ 8031	2
2.1.2 การจัดขาของไมโครคอนโทรลเลอร์ 8031	2
2.1.3 หน่วยความจำของ 8031	5
2.1.3.1 หน่วยความจำโปรแกรม	5
2.1.3.2 หน่วยความจำข้อมูล	5
2.1.4 การจัดการทางสถาปัตยกรรม	5
2.1.5 หน่วยประมวลผลกลางหรือซีพียู	7
2.1.5.1 แอกคิวมูเลเตอร์ (Accumulator : ACC)	7
2.1.5.2 รีจีสเตอร์ B	7
2.1.5.3 รีจีสเตอร์คำแสดงสถานะโปรแกรม (PWF)	8
2.1.5.4 ตัวชี้ข้อมูล (Data Pointor : DPTR)	9
2.1.5.5 ตัวชี้สแตก (Stack Pointor : SP)	9
2.1.5.6 พอร์ต 0 ถึง 3	9
2.1.5.7 บัฟเฟอร์ข้อมูลอนุกรม (Serial Data Buffer : SBUF)	9
2.1.5.8 รีจีสเตอร์ควบคุม (Control Register)	10
2.1.6 ชุดคำสั่งของ 8031	11
2.2 สเตปปิ้งมอเตอร์ (DC Motor)	12
2.3 วงจรขับดีซีมอเตอร์	14
2.3.1 หลักการทำงาน	14
2.4 อุลตร้าโซนิคทรานสดิวเซอร์	16
2.4.1 ตัวรับตัวส่ง	17
2.4.2 วงจรภาคส่ง	17
2.4.3 วงจรภาครับ	18
2.5 สเตปปิ้งมอเตอร์ (Stepping Motor)	20

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.5.1 คุณประโยชน์ของสเตปปิ้งมอเตอร์ 20
2.5.2 ชนิดของสเตปปิ้งมอเตอร์ 20
2.5.3 หลักการทำงานของการขับสเตปปิ้งมอเตอร์ 21
2.6 วงจรตรวจจับแสง (Photo Detector Circuit) 23
2.7 เข็มทิศอิเล็กทรอนิกส์ 24
2.7.1 การประยุกต์ใช้งาน 24
2.8 วงจรเช็คความเร็วล้อ (Encoder Circuits) 24
2.9 ลักษณะการต่อวงจรต่างๆ เข้ากับพอร์ตของไมโครคอลโทรลเลอร์ 25
บทที่ 3 การทดลองและผลการทดลอง 27
3.1 การออกแบบและสร้างตัวรถส่วน Mechanic 27
3.2 การทดสอบวงจรขับเคลื่อน 28
3.3 การทดสอบวงจรตรวจจับสิ่งกีดขวางโดยใช้อุลตร้าโซนิก 29
3.4 การทดสอบวงจรขับเคลื่อนสเตปปิ้งมอเตอร์ 30
3.5 การทดสอบวงจรตรวจจับเป้าหมาย(Photo Detector) 31
บทที่ 4 รูปแบบและโปรแกรมการทำงานของรถ 32
4.1 บล็อกไดอะแกรมการทำงานของรถ 32
4.2 โปรแกรมการทำงานของรถ 38
บทที่ 5 สรุปผล 50
บรรณานุกรม 51
ภาคผนวก 52
ภาคผนวก ก. ชุดคำสั่งของไมโครคอนโทรลเลอร์ตระกูล MCS 51 ก. - 1
ภาคผนวก ข. ข้อมูลรายละเอียดของไมโครคอนโทรลเลอร์ 8031 ข. - 1
ภาคผนวก ค. ข้อมูลรายละเอียดของ IC เบอร์ต่างๆ ค. - 1
ภาคผนวก ง. ข้อมูลรายละเอียดของเข็มทิศอิเล็กทรอนิกส์ ง. - 1

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป

	หน้า
รูปที่ 2.1 โครงสร้างทางสถาปัตยกรรมภายในของ MCS 51	6
รูปที่ 2.2 โครงสร้างเบื้องต้นของสเตปปิ้งมอเตอร์	12
รูปที่ 2.3 แสดงหลักการทำงานอย่างง่ายของสเตปปิ้งมอเตอร์	13
รูปที่ 2.4 แสดงวงจรขับสเตปปิ้งมอเตอร์	14
รูปที่ 2.5 โครงสร้างภายในของอุลตร้าโซนิค	16
รูปที่ 2.6 วงจรภาคส่ง 40 KHz โดยใช้วงจรอะสเตเบิ้ลมัลติไวเบรเตอร์	17
รูปที่ 2.7 แสดงวงจรภาครับอุลตร้าโซนิค	19
รูปที่ 2.8 แสดงวงจรขับสเตปปิ้งมอเตอร์	22
รูปที่ 2.9 แสดงวงจรตรวจจับแสง (Photo Detector Circuit)	23
รูปที่ 2.10 แสดงวงจร Encoder	25
รูปที่ 2.11 แสดงการต่อวงจรต่างๆ เข้ากับไมโครคอลโทรลเลอร์	26
รูปที่ 3.1 แสดงโครงสร้างของรถบังคับ MPV	27
รูปที่ 4.1 บล็อกไดอะแกรม (Block diagram) การทำงานของรถเอ็มพีวี	32

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญตาราง

	หน้า
ตารางที่ 2.1 แสดงขาต่างๆ ของ Port 3	4
ตารางที่ 2.2 รีจีสเตอร์แสดงสถานะโปรแกรม PSW	8
ตารางที่ 2.3 การจัดแบ่งตำแหน่งสำหรับ SFR ให้ทำหน้าที่เป็นรีจีสเตอร์ต่างๆ	10

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 1
บทนำ

โครงงานนี้เป็นโครงงานซึ่งประดิษฐ์รถขึ้นมา โดยมีวัตถุประสงค์ให้รถมีความสามารถในการวิ่งเข้าสู่เป้าหมายที่ต้องการ และหลบหลีกสิ่งกีดขวางได้โดยอัตโนมัติ โดยขั้นตอนการทำงานจะแบ่งออกเป็น 3 ส่วน ได้แก่

ส่วนที่ 1 เป็นส่วนของการออกแบบโครงสร้างและสร้างตัวรถ ถือเป็นส่วนทาง Mechanic

ส่วนที่ 2 เป็นส่วนของวงจรควบคุมการทำงานของรถ และใช้ไมโครคอนโทรลเลอร์เบอร์ 8031 เข้ามาควบคุมการทำงาน ในส่วนนี้ประกอบด้วย วงจรขับเคลื่อนดีซีมอเตอร์ (DC Motor Driver Circuits) , วงจรขับเคลื่อนสเตปปิ้งมอเตอร์ (Stepping Motor Driver Circuit) , วงจรตรวจจับสิ่งกีดขวาง (Ultrasonic Sensor Circuits) , วงจรตรวจจับเป้าหมายโดยใช้ Photo Detector และ วงจรเช็คความเร็วล้อ (Encoder Circuits) นอกจากนี้ได้มีการประยุกต์นำเข็มทิศอิเล็กทรอนิกส์มาใช้ เพื่อการเคลื่อนที่ในทิศทางที่แน่นอนของรถ ทั้งหมดนี้ถือเป็นส่วนทาง Hardware

ส่วนที่ 3 เป็นส่วนการเขียนโปรแกรมควบคุมให้รถทำงานได้ตามความต้องการ ถือเป็นส่วน Software

ภายในรายงานฉบับนี้ประกอบด้วย บทที่ 2 กล่าวถึงทฤษฎีต่างๆ ที่เกี่ยวข้องกับโครงงาน ได้แก่ ทฤษฎีของไมโครคอนโทรลเลอร์ตระกูล MCS 51 , มอเตอร์กระแสตรง (DC Motor) , วงจรขับเคลื่อนมอเตอร์กระแสตรง (DC Motor Driver Circuits) , สเตปปิ้งมอเตอร์ (Stepping Motor), วงจรขับเคลื่อนสเตปปิ้งมอเตอร์ (Stepping Motor Driver Circuits) , วงจรตรวจจับสิ่งกีดขวาง (Ultrasonic Sensor Circuits) และวงจรตรวจจับเป้าหมาย (Photo Detector) , วงจรเช็คความเร็วล้อ (Encoder Circuits) และการทำงานของเข็มทิศอิเล็กทรอนิกส์ บทที่ 3 เป็นการทดสอบและผลการทดสอบวงจรต่างๆ บทที่ 4 รูปแบบและโปรแกรมควบคุมการทำงานของรถ และบทที่ 5 เป็นการสรุปและวิจารณ์ผล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 2
ทฤษฎี

2.1 ไมโครคอนโทรลเลอร์ (Microcontroller) 8031

2.1.1 คุณลักษณะพื้นฐานของ 8031

หน่วยการทำงานพื้นฐานของไมโครคอนโทรลเลอร์เบอร์ 8031 ซึ่งจัดอยู่ในตระกูล MCS 51 ประกอบด้วย

1. ใช้ HMOS และ CHMOS เทคโนโลยีในการสร้างและทำงานด้วยแหล่งจ่ายไฟขนาด 5 โวลต์ เพียงแหล่งเดียว
2. หน่วยประมาลผลกลางขนาด 8 Bit
3. ความสามารถในการอ้างตำแหน่งของหน่วยความจำโปรแกรม 64 กิโลไบต์
4. ชุดแบงค์ (Bank) รีจิสเตอร์มี 4 ชุด แต่ละชุดมีรีจิสเตอร์ 8 ตัว
5. มีพอร์ตไอโอแบบขนาน 2 ทิศทางจำนวน 4 พอร์ต พอร์ตละ 8 Bit
6. พอร์ตแบบอนุกรมสามารถที่จะโปรแกรมการรับส่งแบบฟูลดูเพล็กซ์(fullduplex) ที่ความเร็วสูง
7. แอดเดรสข้อมูลภายนอกได้ 64 กิโลไบต์
8. แอดเดรสโปรแกรมภายนอกได้ 64 กิโลไบต์
9. สามารถกำหนดเลขที่อยู่ข้อมูลขนาด Byte หรือ Bit ได้โดยตรง
10. มีซอฟต์แวร์บิทแฟลก (Bit Flag) สำหรับผู้ใช้ที่จะกำหนดได้ถึง 128 ตำแหน่งBit
11. ตัวเลขทางคณิตศาสตร์ใช้ได้ทั้งระบบไบนารี (Binary) และ เดซิมอล (Decimal)
12. มีคำสั่งคูณ และหารทางฮาร์ดแวร์ที่ทำได้ภายใน 4 ไมโครวินาที
13. โครงสร้างอินเตอร์รัพต์จะติดตั้งได้ถึง 5 แหล่ง พร้อมด้วยการจัดไพโอริตี้ (Priority) ได้ 2 ระดับ
14. ตัวโพรเซสเซอร์สามารถใช้งานแบบบูลีน (Boolean) ได้ สำหรับการใช้กับกระบวนการงานควบคุม

2.1.2 การจัดขาของไมโครคอนโทรลเลอร์ 8031

ตามลักษณะภายนอกของชิพ ขาตามสถาปัตยกรรมของ 8031 จัดเป็น 4 กลุ่ม คือ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1. กลุ่มขารับแหล่งจ่ายไฟ และระบบสัญญาณนาฬิกา
2. กลุ่มขาแอดเดรสและข้อมูล
3. กลุ่มขาควบคุม
4. กลุ่มพอร์ตแบบขนาน และอนุกรม

รายละเอียดหน้าที่ของขาแต่ละขาดังนี้

- ขา Vss (ขา 20) เป็นขาสำหรับต่อลงดิน
- ขา Vcc (ขา 40) เป็นขาที่ต่อแรงดันไฟฟ้ากระแสตรงขนาด 5 โวลต์
- ขา SRT (ขา 9) ต้องคงสถานะค่าสูงเป็นเวลาอย่างน้อย 2 วัฏจักร ระหว่างที่ออสซิเลเตอร์ทำงานขณะที่ต้องการรีเซตทั้งระบบการทำงาน
- ขา ALE/PROG (ขา 30) เป็นขาแอดเดรสแลทช์อีนาเบิ้ลด้วยการส่งพัลส์ออกไปใช้สำแลทช์ค่าแอดเดรสไบต์ต่ำจาก Port 0 ในระหว่างการเข้าถึงข้อมูลในหน่วยความจำภายใน ALE จะถูกส่งสัญญาณนาฬิกาออกมาในอัตราความเร็วคงที่ ที่ 1/8 ของความถี่ ออสซิเลเตอร์ตลอดเวลา แม้ว่าจะไม่มีการเข้าถึงข้อมูลจากภายใน ดังนั้นจึงสามารถที่จะใช้สัญญาณจากขานี้เป็นตัวตั้งเวลาภายนอกหรือเป็นความถี่สัญญาณนาฬิกา แต่อย่างไรก็ตามความถี่สัญญาณนี้จะลดความถี่ช้าลงไปเท่าหนึ่งระหว่างการทำงานแบบการเข้าถึงของหน่วยความจำข้อมูลภายนอก ขานี้ยังใช้เป็นสัญญาณพัลส์เข้าสำหรับการควบคุมการโปรแกรม EPROM ภายในชิพ
- ขา PSEN (ขา 29) Program Storage Enable เป็นสโตรบอ่านข้อมูลจากโปรแกรมหน่วยความจำภายนอก เมื่อชิพทำงานด้วยโปรแกรมจากหน่วยความจำภายนอก ขา PSEN จะสร้างสโตรบ 2 ครั้งภายใน 1 วัฏจักรแมชชีน
- ขา EA/Vpp (ขา 31) การทำให้ขา EA อยู่ในสถานะต่ำ จะเป็นการควบคุมให้ซีพียูทำงานตามโปรแกรมหน่วยความจำภายนอก
- ขา XTAL 1 (ขา 19) ใช้เป็นตัวอินพุทเข้าสู่ตัวออสซิเลเตอร์ขยายแบบ Invert
- ขา XTAL 1 (ขา 18) ใช้เป็นตัวเอาท์พุทเข้าสู่ตัวออสซิเลเตอร์ขยายแบบ Invert
- ขา Port 0 (P0.0 - P0.7 ขา 32 - 39) เป็นPort I/O 8 Bit แบบ OPEN DRAIN BIDIRECTIONAL สามรถรับโหลดทีทีแอลได้ 8 ตัว การเขียนค่า '1' ลงไปที่พอร์ตนี้จะเป็นการปล่อยลอยขา (Float) ขาของพอร์ตนี้ ทำให้มันทำงานเป็นอินพุทมีอิมพีแดนซ์สูงในการให้พอร์ตนี้บริการแบบ I/O แลอีกหน้าที่หนึ่งของ Port 0 จะทำงานเป็นมัลติเพลกซ์ ด้วยสัญญาณแอดเดรสไบต์ต่ำกับบัสข้อมูล สำหรับการใช้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

งานด้ายหน่วยความจำภายนอก ในการใช้งานแบบนี้จะใช้ลักษณะภายในเป็นตัวพูลอัพ นอกจากหน้าที่หลัก 2 หน้าที่นี้แล้ว

- ขา Port 1 (P1.0 - P1.7 ขา 1 - 8) เป็น Port I/O แบบ OPEN DRAIN BIDIRECTIONAL พร้อมด้วยการพูลอัพภายใน ถ้าเป็ตพอร์ทเอาท์พุท บัพเฟอร์สามารถขับโหลดทีทีแอลตระกูล SI ได้ 4 ตัว Port 1 เมื่อถูกเขียนค่า '1' ด้วยโปรแกรมมันจะมีสถานะสูงด้วยการพูลอัพภายใน การให้สถานะเช่นนี้จะเป็นการ Initial ใช้งานพอร์ตนี้ให้เป็นอินพุท ขณะที่ Port 1 เป็นอินพุท การให้สัญญาณลงต่ำจะเป็นการจ่ายออกเนื่องจากเป็นการพูลอัพภายใน
- ขา Port 2 (P2.0 - P2.7 ขา 21 -28) เป็น Port I/O แบบ OPEN DRAIN BIDIRECTIONAL ด้วยการพูลอัพภายใน Port 2 ที่ทำหน้าที่เป็นบัฟเฟอร์เอาท์พุทสามารถจ่ายโหลดทีทีแอลตระกูล SI ได้ 4 ตัว อีกหน้าที่หนึ่งของพอร์ตจะถูกใช้งานเป็นตัวส่งแอดเดรสไบต์สูงด้วย เมื่อใช้งานร่วมกับหน่วยความจำภายนอกเพื่อให้แอดเดรสได้ถึง 16 บิต ด้วยการใช้งานแบบนี้มันจะมีพูลอัพภายในที่ช่วยให้การส่งค่า '1' ได้ระดับที่แน่นอน
- ขา Port 3 (P3.0 - P3.7 ขา 10 - 17) เป็น Port I/O แบบพูลอัพภายใน Port 2 นอกจากทำเป็นพอร์ต I/O ที่สามารถจ่ายโหลดทีทีแอลตระกูล SI ได้ 4 ตัวแล้ว ยังมีหน้าที่พิเศษอีกหน้าที่หนึ่งตามตารางที่ 2.1

การที่จะให้ทำงานตามฟังก์ชันตามตารางจะต้องเริ่มโปรแกรมด้วยการส่งค่า '1' ไป แลทช์ไว้ก่อนที่ให้ทำงานตามฟังก์ชันในตาราง

ขา Port	ขา	การทำงานตามฟังก์ชันพิเศษ
P3.0	10	RXD พอร์ตอนุกรมอินพุท
P3.1	11	TXD พอร์ตอนุกรมเอาร์พุท
P3.2	12	INYO อินเตอร์รัพภายนอกตัวที่ 0
P3.3	13	INTI อินเตอร์รัพภายนอกตัวที่ 1
P3.4	14	TO สัญญาณกระตุ้นเข้าที่ตัวนับเวลาและตัวนับ 0
P3.5	15	TI สัญญาณกระตุ้นเข้าที่ตัวตั้งเวลาและตัวนับ 1
P3.6	16	WR สัญญาณควบคุมการเขียน
P3.7	17	RD สัญญาณควบคุมการอ่าน

ตารางที่ 2.1 แสดงขาต่างๆ ของ Port 3

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ไมโครคอนโทรลเลอร์เบอร์ 8031 นี้ จะไม่มีหน่วยความจำของโปรแกรมบนชิพ จึงต้องต่อหน่วยความจำโปรแกรมจากภายนอกด้วย ROM , EPROM หรือ PROM ได้ถึง 64 กิโลไบต์

2.1.3 หน่วยความจำของ 8031

ไมโครคอนโทรลเลอร์เบอร์ 8031 จะแยกหน่วยความจำออกเป็น 2 หน่วยอย่างชัดเจน คือ หน่วยความจำข้อมูลภายใน (Data Memory) มีขนาด 128 Byte และหน่วยความจำโปรแกรม (Promgram Memory) หน่วยความจำทั้ง 2 นี้จะมีหน้าที่แตกต่างไปจากกันและใช้วิธีการอ้างอิงแอดเดรสสัญญาณการติดต่อออกจากกัน

2.1.3.1 หน่วยความจำโปรแกรม

หน่วยความจำโปรแกรมเป็นบริเวณสำหรับเก็บข้อมูลและคำสั่งใช้งานต่างๆ ซึ่งแม้ว่าจะไม่มีการจ่ายกระแสไฟฟ้าให้ระบบ ข้อมูลเหล่านี้ก็จะไม่สูญหาย สำหรับหน่วยความจำโปรแกรมของ 8031 จะใช้ EPROM ต่ออยู่ภายนอก มีความสามารถในการอ้างอิงแอดเดรสได้ไม่เกิน 64 กิโลไบต์

2.1.3.2 หน่วยความจำข้อมูล

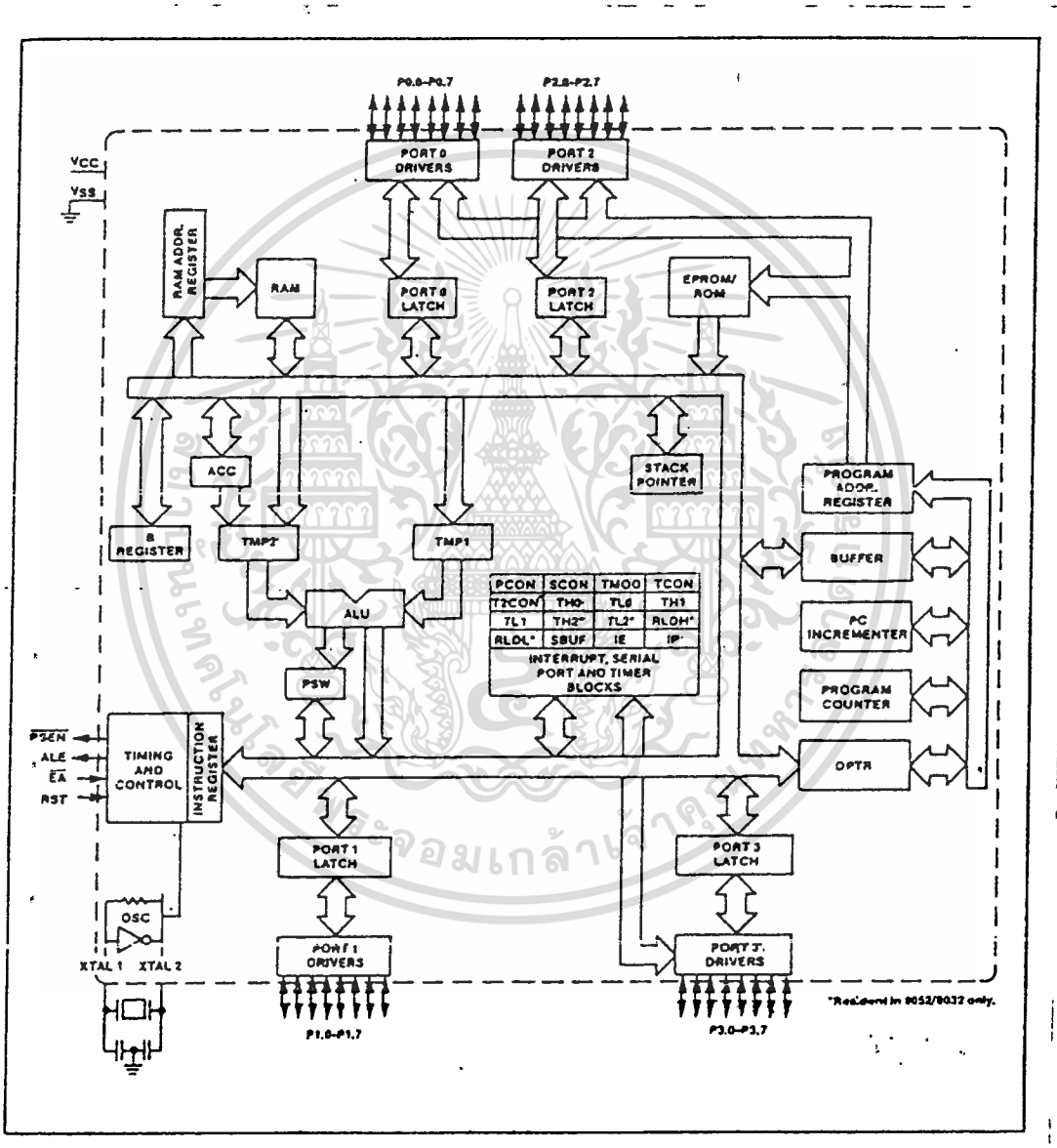
หน่วยความจำข้อมูลจะสามารถเขียนหรืออ่านข้อมูลได้ (Read or Write Memory) ใช้สำหรับเก็บข้อมูลหรือตัวแปรที่เกิดขึ้นในขณะประมวณผลโปรแกรมไว้เป็นการชั่วคราว ซึ่งโดยพื้นฐานแล้วจะเป็นหน่วยความจำ RAM แบบ Static ดังนั้นเมื่อไม่มีการจ่ายไฟฟ้าให้กับระบบก็จะมีผลทำให้ข้อมูลที่จัดเก็บไว้สูญหายได้ พื้นที่ของหน่วยความจำข้อมูลสามารถมีได้สูงสุดไม่เกิน 64 กิโลไบต์ ซึ่งจะแยกออกมาเป็น 2 ลักษณะ คือ หน่วยความจำข้อมูลภายใน และหน่วยความจำข้อมูลภายนอก

2.1.4 การจัดการทางสถาปัตยกรรม

สถาปัตยกรรมภายในของไมโครคอนโทรลเลอร์ตระกูล MCS 51 จะประกอบด้วยหน่วยประมวลผลกลาง , หน่วยความจำ , พอร์ตเอาท์พุท - อินพุท , โหมดรีจีสเตอร์สถานะและข้อมูล และส่วนวงจรตรรก RANDOM ที่จำเป็นสำหรับตัวแปรของฟังก์ชันซึ่งแสดงตามบล็อกไดอะแกรม (Block Diagram) ดังรูปที่ 2.1 การต่อพ่วงส่วนต่างๆ จะติดต่อกันด้วยบัสข้อมูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

(Data Bus) ขนาด 8 Bit และจะมีบัฟเฟอร์สำหรับการติดต่อข้อมูลกับภายนอกผ่าน I/O Port เมื่อต้องการขยายหน่วยความจำหรือ I/O Port



รูปที่ 2.1 โครงสร้างสถาปัตยกรรมภายในของ MCS 51

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.1.5 หน่วยประมวลผลกลางหรือซีพียู

ซีพียูเป็นสมองของระบบไมโครคอมพิวเตอร์ การอ่านโปรแกรมและการกระทำตามคำสั่งโปรแกรมจะกระทำที่ส่วนนี้ โดยใช้ส่วนคณิตศาสตร์และตรรกศาสตร์ร่วมกับ รีจิสเตอร์ต่างๆ ส่วนการคำนวณทางคณิตศาสตร์ 8031 จะดำเนินการโดยใช้หน่วยประมวลผลทางคณิตศาสตร์และลอจิก (Arithmetic and Logic Unit : ALU) ซึ่งจะมีรีจิสเตอร์ที่เข้ามาเกี่ยวข้องด้วย คือ รีจิสเตอร์เก็บข้อมูลชั่วคราว (Temporary Register) เพื่อเก็บข้อมูลตัวเลขที่จะมาดำเนินการนั้น ดังนั้นเมื่อเราได้สั่งคำสั่งที่ต้องดำเนินการโดย ALU ก็มักจะต้องส่งค่าของข้อมูลมาทางตัวดำเนินการเพื่อย้ายข้อมูลเข้าไปยังรีจิสเตอร์เก็บข้อมูลชั่วคราวนี้โดยอัตโนมัติ และเมื่อเสร็จสิ้นการดำเนินการเรียบร้อยแล้ว จึงจะนำผลลัพธ์ที่ได้ออกมาเก็บไว้ในรีจิสเตอร์ A เพื่อให้ผู้ใช้นำไปประมวลผลในโปรแกรมต่อไป

นอกจากนี้ 8031 ยังมีคำสั่งที่เกี่ยวข้องกับการคูณหรือการหารด้วย ซึ่งก็มีการดำเนินการที่เหมือนกัน เพียงแต่จะต้องนำข้อมูลมาไว้ในรีจิสเตอร์ A และ รีจิสเตอร์ B เท่านั้น นอกจากนี้รีจิสเตอร์อีกตัวหนึ่งที่ข้อมูลของมันเป็นผลมาจากการทำงานของหน่วย ALU คือ รีจิสเตอร์ PSW ซึ่งทำหน้าที่เก็บสภาวะที่เกี่ยวข้องกับผลลัพธ์จากการดำเนินการทางคณิตศาสตร์ หรือลอจิก ซึ่งเราสามารถพิจารณาได้ว่าเป็นกลุ่มของแฟล็ก (Flags) เพื่อบอกผลของการทำคำสั่ง ตัวอย่างเช่น การบวกค่าตัวเลขและมีการทดค่าเกิดขึ้น แฟล็กทด (Carry Flag) ก็จะมีสภาวะตามการทดตัวเลขนั้น เป็นต้น

2.1.5.1 แอกคิวมูเลเตอร์ (Accumulator : ACC)

MCS 51 จะใช้ ACC ที่มีขนาด 8 บิต เป็นแอกคิวมูเลเตอร์ คำสั่งส่วนใหญ่จะอ้างถึงตัวรีจิสเตอร์ตัวนี้ โดยถือค่าภายในเป็นค่าตั้งต้น และรับค่าผลลัพธ์ที่ได้จากคำสั่งทางคณิตศาสตร์ เช่น การบวก , ลบ , คูณ และหารเข้ามาเก็บไว้ ตัว ACC ยังสามารถใช้เป็นตัวกระทำหรือถูกกระทำในการทำงานทางตรรก และใช้เป็นตัวกลางในการถ่ายเทข้อมูลในการติดต่อกับอุปกรณ์ภายนอก (I/O) และหน่วยความจำภายนอก รวมถึงการตรวจสอบตารางข้อมูล

2.1.5.2 รีจิสเตอร์ B

เป็นรีจิสเตอร์พิเศษที่ใช้งานสำหรับคำสั่งของการคูณ และหาร โดยใช้เป็นที่เก็บตัวคูณหรือตัวหาร และเป็นที่เก็บผลลัพธ์ตัวที่สอง หลังการคูณ และเศษหลังการหาร

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.1.5.3 รีจิสเตอร์คำแสดงสถานะโปรแกรม (Program Status Word : PSW)

รีจิสเตอร์ PSW เป็นรีจิสเตอร์ที่แสดงผลที่ได้หลังจากการใช้คำสั่งต่างๆ และใช้เป็นตัวเลือกกลุ่มการทำงานของรีจิสเตอร์กลุ่มต่างๆ ซึ่งมีรายละเอียดดังตารางที่ 2.2

(MSB)	CY	AC	F0	RS1	RS0	OV	-	P	(LSB)

สัญลักษณ์	ตำแหน่ง	ข้อกำหนดการทำงาน
CY	PSW7	แฟลกตัวทด จะเซต/เคลียร์ด้วยฮาร์ดแวร์หรือซอฟต์แวร์ ระหว่างผลลัพธ์หลังการใช้คำสั่งทางคณิตศาสตร์ หรือ ตรรกศาสตร์ที่แน่นอน
AC	PSW6	แฟลกตัวทดของ Auxiliary จะเซต/เคลียร์ด้วยฮาร์ดแวร์ ระหว่างการบวกและลบที่แสดงผลจากการทดหรือยืมจากบิตที่ 3 ของ ACC
F0	PSW5	แฟลก 0 จะเซต/เคลียร์ด้วยซอฟต์แวร์ที่ผู้ใช้กำหนดสถานะแฟลกนี้เอง
RS1	PSW4	เรจิสเตอร์ตัวควบคุมการเลือกแบงค์ด้วยค่า RS1 และ RS0
RS0	PSW3	จะเซต/เคลียร์ด้วยซอฟต์แวร์ เพื่อเลือกกลุ่มเรจิสเตอร์ทำงานในแต่ละแบงค์ โดยปรับค่าใน RS1 และ RS0 ให้อ้างเบิ้ลคลุมลักษณะการเลือกแบงค์ต่อไปนี้
	RS1 RS0	เลือกแบงค์ ค่าแอดเดรส
	0 0	แบงค์ 0 00H - 07H
	0 1	แบงค์ 1 08H - 0FH
	1 0	แบงค์ 2 10H - 17H
	1 1	แบงค์ 3 18H - 1FH
OV	PSW2	แฟลก Overflow จะเซต/เคลียร์ด้วยฮาร์ดแวร์ระหว่างการใช้คำสั่ง ที่แสดงผลถึงการเกิดลักษณะ Overflow ทางคณิตศาสตร์
-	PSW1	บิตสำรอง จะไม่สามารถเซต/เคลียร์ด้วยผู้ใช้ เพราะสำรองไว้สำหรับโรงงานผู้สร้าง
P	PSW0	แฟลกพาริตี้ จะเซต/เคลียร์ด้วยฮาร์ดแวร์ในแต่ละวัฏจักรคำสั่ง แสดงถึงตัวเลขค่า '1' ในแต่ละบิตของแอกคูมิวเลเตอร์ เช่น '1' มี 6 ตัวจะเป็นพาริตี้คู่ P บิตจะเท่ากับ 0

หมายเหตุ ความหมายของฮาร์ดแวร์ และซอฟต์แวร์ในตารางต่าง ๆ ที่จะกล่าวต่อไปนี้ในแต่ละบิตของตัวเรจิสเตอร์ การที่บิตจะเซตหรือเคลียร์นั้น ถ้าเกิดขึ้นจากฮาร์ดแวร์จะหมายถึงว่า คำบิตในเรจิสเตอร์จะเกิดเซตตัวเองเนื่องจากผลของความหมายของการทำงานตามคำสั่งของบิตนั้น เช่น IE จะเซตตัวเองด้วยฮาร์ดแวร์ เมื่อการส่งข้อมูลได้สิ้นสุดถึง STOP บิตแล้ว ช่วยให้เราสามารถตรวจสอบได้ว่าการส่งข้อมูลครั้งละไบต์นั้นสิ้นสุดหรือยัง ถ้ายังจะได้รอต่อไปก่อน หรือมีการคำนวณแล้วผลลัพธ์เกิด Overflow ใน PSW ก็จะเซตตัวเองที่บิต OV ส่วนทางซอฟต์แวร์ หมายถึงว่าเราสามารถที่จะเซต หรือเคลียร์ได้ด้วยการใช้คำสั่งโปรแกรมต่าง ๆ ในการเซตหรือเคลียร์ในบิตแต่ละบิตของเรจิสเตอร์เป็นลักษณะทางซอฟต์แวร์

ตารางที่ 2.2 รีจิสเตอร์คำแสดงสถานะโปรแกรม PSW

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.1.5.4 ตัวชี้ข้อมูล (Data Pointer : DPTR)

DPTR เป็นรีจิสเตอร์ขนาด 16 บิต ที่ประกอบด้วยไบต์สูง (DPH) และไบต์ต่ำ (DPL) ที่สามารถเลือกแบ่งออกเป็น รีจิสเตอร์ 8 บิต สองตัวที่ใช้ได้อย่างอิสระ หรือจะใช้รวมกันทั้ง 16 บิตก็ได้ ในการ Increment หรือ Decrement เพื่อใช้ประโยชน์ในการเป็นฐานของตัวเลขที่อยู่ในรีจิสเตอร์ในการกระโดดโดยทางอ้อมในการใช้คำสั่งเกี่ยวกับตารางข้อมูล และชี้ตำแหน่งของหน่วยความจำภายนอก

2.1.5.5 ตัวชี้สแตก (Stack Pointer : SP)

MCS 51 จะใช้ RAM ภายในเป็นบริเวณสแตกทางฮาร์ดแวร์ สำหรับการเชื่อมต่อระหว่างโปรแกรมหลัก สแตกการผ่านพารามิเตอร์ระหว่างงานในแต่ละส่วนโปรแกรม และสแตกเก็บตัวแปรข้อมูลชั่วคราว หรือสแตกการเก็บสถานะระหว่างการบริการงานอินเตอร์รัพต์ไว้ภายในชิพ โดยที่ SP จะมีขนาด 8 บิต และจะเพิ่มค่าขึ้นโดยอัตโนมัติก่อนที่ข้อมูลจะนำมาเก็บในหน่วยความจำระหว่างการใช้คำสั่ง PUSH และ CALL และจะลดค่าของ SP ลงหลังจากที่ได้ถ่ายเทข้อมูลออกไปแล้วในคำสั่ง POP หรือ RETURN โดยทฤษฎีทางสถาปัตยกรรม MCS 51 สามารถใช้สแตกให้มีเนื้อที่ถึง 128 Byte แต่ในทางปฏิบัติสำหรับโปรแกรมทั่วไปจะใช้น้อยกว่านี้ SP จะเริ่มที่ตำแหน่ง 07H ดังนั้นสแตกจะเริ่มบรรจุข้อมูลที่ตำแหน่ง 08H MCS 51 สามารถเปลี่ยนแปลงค่าใน SP ได้ ซึ่งจะเป็นการเปลี่ยนตำแหน่งสแตกไปยังที่ใดๆ ของ RAM ภายในชิพ

2.1.5.6 พอร์ต 0 ถึง 3

รีจิสเตอร์ P0 , P1 , P2 และ P3 ของกลุ่มรีจิสเตอร์ฟังก์ชันพิเศษ (Special Function Register : SFR) จะเป็นตัวรีจิสเตอร์ที่แลทช์ค่าของพอร์ต 0 , 1 , 2 และ 3 ตามลำดับในขณะใช้งาน

2.1.5.7 บัฟเฟอร์ข้อมูลอนุกรม (Serial Data Buffer : SBUF)

บัฟเฟอร์ข้อมูลอนุกรมแบ่งออกเป็นรีจิสเตอร์สองตัว ตัวหนึ่งเป็นบัฟเฟอร์การส่งและอีกตัวเป็นบัฟเฟอร์การรับ เมื่อข้อมูลถ่ายเทเข้า SBUF มันจะถ่ายเข้าบัฟเฟอร์ส่งซึ่งเป็นตัวจัดการส่งข้อมูลอนุกรม วิธีการเคลื่อนย้ายเข้า SBUF ขึ้นอยู่กับการติดตั้งโปรแกรม (Initial) การส่งเมื่อข้อมูลย้ายออกจาก SBUF จะเป็นการรับข้อมูลจากบัฟเฟอร์ตัวรับ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.1.5.8 รีจิสเตอร์ควบคุม (Control Register)

กลุ่ม SFR ที่เป็น IP , IE , TMOD , TCON , T2CON , SCON และ PCON จะประกอบด้วยบิตที่ใช้ในการควบคุม และแสดงสถานะของการใช้งานในระบบอินเตอร์รัพต์ ตัวจับเวลา/ตัวนับ และพอร์ตอนุกรม

SPECIAL FUNTION REGISTERS	ตำแหน่ง
* ACC Accumulator	0E0H
* B เรจิสเตอร์	0F0H
* PSW Program Status Word	0D0H
SP Stack Pointer	081H
DPTR ตัวชี้ข้อมูล	
ประกอบด้วย DPH	083H
และ DPL	082H
* P0 พอร์ต 0	080H
* P1 พอร์ต 1	090H
* P2 พอร์ต 2	0A0H
* P3 พอร์ต 3	0B0H
* IP ตัวควบคุมการอินเตอร์รัพต์ตามลำดับ	0B8H
* IE ตัวควบคุมการอินเตอร์รัพต์อีนาเบิ้ล	0A8H
TMOD ตัวควบคุมการเลือกโหมดตัวจับเวลา/ตัวนับ	089H
* T2CON ตัวควบคุมตัวจับเวลา/ตัวนับ 2	088H
TCON ตัวควบคุมตัวจับเวลา/ตัวนับ	0C8H
TH0 เรจิสเตอร์ตัวจับเวลา/ตัวนับ0 (ไบต์สูง)	08CH
TL0 เรจิสเตอร์ตัวจับเวลา/ตัวนับ0 (ไบต์ต่ำ)	08AH
TH1 เรจิสเตอร์ตัวจับเวลา/ตัวนับ1 (ไบต์สูง)	08DH
TL1 เรจิสเตอร์ตัวจับเวลา/ตัวนับ1 (ไบต์ต่ำ)	08BH
+ TH2 เรจิสเตอร์ตัวจับเวลา/ตัวนับ2 (ไบต์สูง)	0CDH
+ TL2 เรจิสเตอร์ตัวจับเวลา/ตัวนับ2 (ไบต์ต่ำ)	0CCH
+ RLDH เรจิสเตอร์ตัวจับเวลา/ตัวนับ2 ประจุใหม่อัตโนมัติ(ไบต์สูง)	0CBH
+ RLDL เรจิสเตอร์ตัวจับเวลา/ตัวนับ2 ประจุใหม่อัตโนมัติ(ไบต์ต่ำ)	0CAH
* SCON ควบคุมการส่งข้อมูลอนุกรม	098H
SBUF บัฟเฟอร์ข้อมูลการส่งอนุกรม	099H
PCON ควบคุมการใช้พลังงาน (Power)	097H

เครื่องหมาย* หน้าตัวเรจิสเตอร์แสดงว่า เรจิสเตอร์นั้นสามารถที่จะแอดเดรสข้อมูลได้ทั้งข้อมูลขนาดไบต์และบิต

เครื่องหมาย + นั้นแสดงว่าจะมีเฉพาะในเบอร์ 8032/8052 เท่านั้น

ตารางที่ 2.3 การจัดแบ่งตำแหน่งสำหรับ SFR ให้ทำหน้าที่เป็นรีจีสเตอร์ต่างๆ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.1.6 ชุดคำสั่งของ 8031

ไมโครคอนโทรลเลอร์ 8031 ประกอบด้วยคำสั่งทั้งหมดจำนวนมากซึ่งแสดงไว้ในตารางของภาคผนวก ซึ่งสามารถจะจัดกลุ่มคำสั่งเหล่านี้ตามลักษณะหน้าที่การทำงาน ที่คล้ายคลึงกัน เพื่อความสะดวกต่อการศึกษาทำความเข้าใจและใช้งานดังนี้

1. กลุ่มคำสั่งในการโอนย้ายข้อมูล ทำหน้าที่ในการโอนย้ายข้อมูลระหว่างรีจิสเตอร์หรือหน่วยความจำ

2. กลุ่มคำสั่งดำเนินการแบบลอจิก ทำหน้าที่เกี่ยวกับการประมวลผลแบบลอจิกต่างๆ เช่น การ AND , OR หรือ XOR ระหว่างข้อมูลในรีจิสเตอร์ A กับรีจิสเตอร์ทั่วไป และการทำคำสั่งเกี่ยวกับการเลื่อน หรือหมุนบิตข้อมูลภายในรีจิสเตอร์ A เป็นต้น

3. กลุ่มคำสั่งในการกระโดดไปยังตำแหน่งต่างๆ ภายในโปรแกรม ซึ่งจะเปลี่ยนลำดับของการประมวลผลภายในโปรแกรมไปยังส่วนต่างๆ แทนที่จะดำเนินการเป็นลำดับต่อเนื่องเช่นปกติ ซึ่งสามารถจะกระทำการกระโดดได้ทั้งแบบมี หรือไม่มีเงื่อนไข นอกจากนั้นกลุ่มคำสั่งนี้ยังรวมถึงการเรียกใช้งานโปรแกรมย่อย และการกลับมาทำโปรแกรมหลักด้วย

4. กลุ่มคำสั่งการคำนวณทางคณิตศาสตร์ เช่น การบวก , ลบ , คูณ และหารข้อมูลภายในรีจิสเตอร์ต่างๆ

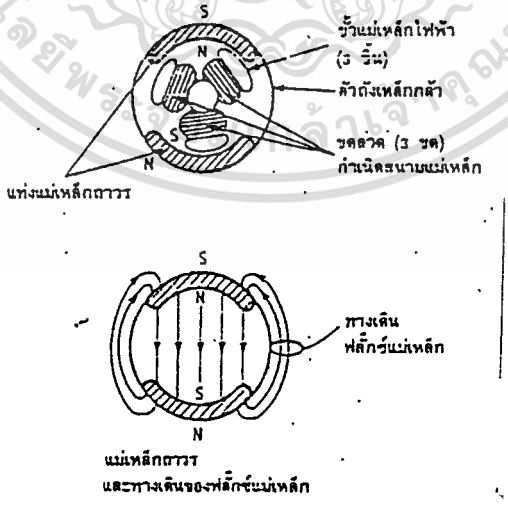
5. กลุ่มคำสั่งการประมวลผลแบบบิต ซึ่งเป็นความสามารถของ 8031 ที่จะดำเนินการประมวลผลแบบบิต แทนที่จะเป็นข้อมูลทั้งไบต์เช่นปกติโดยมีชุดคำสั่งที่จัดการโดยตรง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.2 มอเตอร์กระแสตรง (DC Motor)

หลักการทำงานของมอเตอร์กระแสตรง โดยการผ่านกระแสไฟฟ้าให้ขดลวดในสนามแม่เหล็กซึ่งจะทำให้เกิดแรงแม่เหล็ก สัดส่วนของแรงนี้ขึ้นอยู่กับกระแสและกำลังของสนามแม่เหล็ก และเกิดขึ้นเป็นมุมฉากกับกระแสและสนามแม่เหล็ก ขณะที่ทิศทางของแรงจะตรงกันข้าม ถ้ากระแสของสนามแม่เหล็กไหลย้อนกลับ การเปลี่ยนแปลงของสนามแม่เหล็กและกระแสจะเป็นผลให้ทิศทางและแรงเปลี่ยนแปลงเช่นกัน ด้วยคุณสมบัติเช่นนี้ทำให้มอเตอร์กระแสตรงกลับทิศทางการหมุนได้

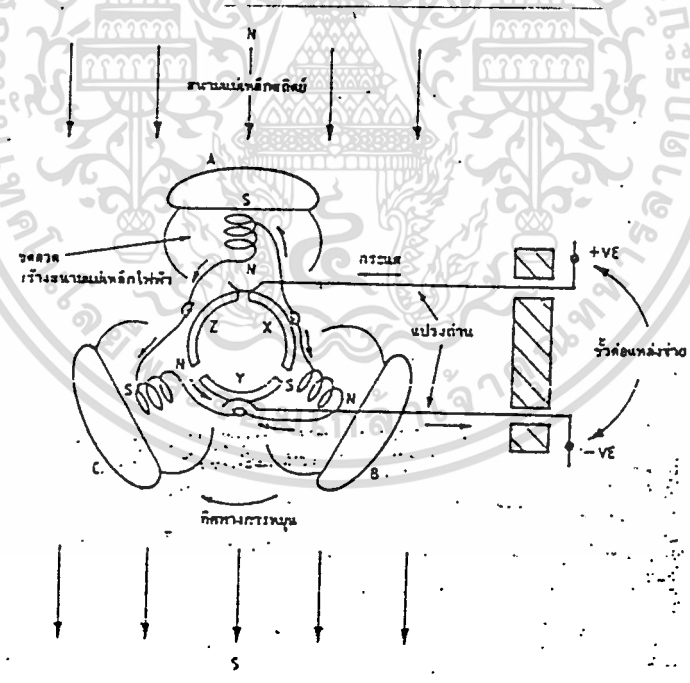
โครงสร้างโดยทั่วๆ ไปของมอเตอร์กระแสตรงแยกได้เป็น 3 ส่วน ดังรูปที่ 2.2 เป็นโครงสร้างของมอเตอร์รุ่นเล็ก ทางเดินของฟลักซ์แม่เหล็กและสนามแม่เหล็กซึ่งเกิดจากแท่งแม่เหล็กเฟอร์ไรต์ 2 ชิ้น ขึ้นรูปเป็นแบบโค้งยึดกับตัวถังได้พอดี เพื่อให้เส้นแรงแม่เหล็กวิ่งเข้าสู่ใจกลางมอเตอร์ ความเข้มของสนามแม่เหล็กจะขึ้นอยู่กับความหนาของแม่เหล็กด้วย ฟลักซ์แม่เหล็กจะวิ่งไปบนตัวถัง กระแสไฟฟ้าในขดลวดที่พันกับทุ่นโรเตอร์ จะทำให้เกิดสนามแม่เหล็กไฟฟ้าด้านกับสนามแม่เหล็กถาวร เกิดเป็นแรงบิดหมุนทุ่นโรเตอร์ไปตามทิศทางเดียวกับทิศทางของสนามแม่เหล็กที่มีแรงมากกว่า กระแสจะผ่านไปยังทุ่นโรเตอร์โดยผ่านแปรงถ่านซึ่งสัมผัสกับแหวนตัวนำบนทุ่นโรเตอร์แหวน(คอมมิวเตเตอร์) ถูกแบ่งเป็น 3 เซกเมนต์ ทำหน้าที่นำกระแสเข้าขดลวด



รูปที่ 2.2 โครงสร้างเบื้องต้นของมอเตอร์กระแสตรง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 2.3 แสดงการทำงานของมอเตอร์ กระแสในขดลวด B จะมีทิศทางตรงข้ามกับในขดลวด A และ C โดยที่ขดลวดทั้ง 3 จะต่อกันอยู่ในลักษณะอนุกรม ทิศทางการไหลของกระแสในขดลวด B ทำให้ขั้วแม่เหล็ก B มีสภาพเป็นขั้วเหนือถูกดูดไปทางขั้วใต้ของแม่เหล็กสถิตย์ ส่วนขดลวด A และ C มีสภาพเป็นขดลวดใต้จึงถูกดูดไปทางขั้วเหนือ แรงดูดแบบนี้ทำให้ทุ่นโรเตอร์เกิดแรงบิดมีทิศตามเข็มนาฬิกา เมื่อโรเตอร์หมุนไปได้เล็กน้อยแปรงถ่านจะสัมผัสกับเซกเมนต์ Z ของคอมมิวเตเตอร์ เป็นผลให้กระแสในขดลวด A มีทิศทางตรงข้ามกับขดลวดแรก ส่วนทิศทางของกระแสในขดลวดที่เหลืออีก 2 ขด ไหลทิศทางเดิม ขั้ว A ก็จะกลายเป็นขั้วเหนือและถูกผลักออกจากขั้วเหนือของแม่เหล็กสถิตย์ไปยังขั้วใต้แทน ในลักษณะที่กล่าวมาจึงทำให้มอเตอร์หมุนต่อไปได้ เมื่อขั้ว B อยู่ตรงกับขั้วใต้ของแม่เหล็กสถิตย์ แปรงถ่านขั้วลบจะเปลี่ยนจากอาร์เมเจอร์เซกเมนต์ Y เป็นเซกเมนต์ X กระแสในขดลวด B มีทิศทางตรงกันข้ามกับขดลวดเดิม และขั้ว B ซึ่งเป็นขั้วใต้จะถูกผลักออกจากขั้วใต้ของแม่เหล็กสถิตย์ การหมุนจะเป็นวัฏจักรในทิศทางนี้ไปเรื่อยๆ จนกว่ากระแสภายนอกจะกลับทิศทาง

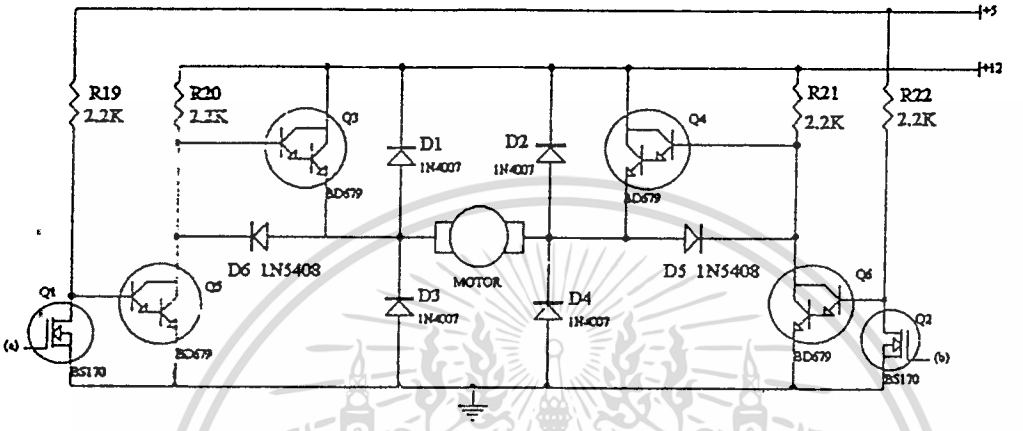


รูปที่ 2.3 แสดงหลักการทำงานอย่างง่ายของมอเตอร์กระแสตรง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.3 วงจรขับมอเตอร์กระแสตรง

เป็นส่วนของวงจรที่ใช้ขับให้มอเตอร์หมุน โดยจะเป็นวงจรที่จ่ายกระแสและแรงดันให้กับมอเตอร์ วงจรขับมอเตอร์แสดงดังรูป 2.4



รูปที่ 2.4 วงจรขับมอเตอร์กระแสตรง

โดยกระแสจะไหลผ่านมอเตอร์ได้สองทางคือ ผ่านทาง Q_4 และ Q_5 และผ่านทาง Q_3 และ Q_6 เมื่อทิศทางการไหลของกระแสต่างกันทำให้ทิศทางการหมุนของมอเตอร์จะกลับข้างกัน

2.3.1 หลักการทำงาน

เมื่อให้สัญญาณเข้ามาทางจุด (a) ส่วนจุด (b) มีแรงดัน 5 โวลต์จ่ายเข้ามา สัญญาณที่จุด (a) จะเปลี่ยนแปลงไปมาระหว่าง 5 โวลต์ และ 0 โวลต์

เมื่อจุด (b) มีแรงดัน 5 โวลต์จะทำให้ Q_2 นำกระแส เมื่อ Q_2 นำกระแสจะเป็นการดึงกระแสที่จะเข้าขาเบสของ Q_6 ทำให้ Q_6 ไม่นำกระแสและถ้าสัญญาณที่จุด (a) มีแรงดันเป็น 5 โวลต์เหมือนจุด (b) จะทำให้ Q_1 นำกระแส เมื่อ Q_1 นำกระแสจะเป็นการดึงกระแสที่จะเข้าขาเบสของ Q_5 ทำให้ Q_5 ไม่นำกระแส เมื่อทั้ง Q_5 และ Q_6 ไม่นำกระแสจะมีผลทำให้ไม่มีกระแสไหลผ่านมอเตอร์ทำให้ไม่มีกระแสจ่ายมอเตอร์

แต่เมื่อสัญญาณที่จุด (a) มีค่าแรงดันเป็น 0 โวลต์ จะทำให้ Q_1 ไม่นำกระแสซึ่งทำให้มีกระแสไบอัสไหลผ่าน R20 เข้าขาเบสของ Q_5 ทำให้ Q_5 นำกระแส และเป็นการดึงกระแสเบสของ Q_3 ทำให้ Q_3 ไม่นำกระแส แต่ในขณะนี้สัญญาณที่จุด (b) มีแรงดันเป็น 5 โวลต์ ซึ่งจะเป็น

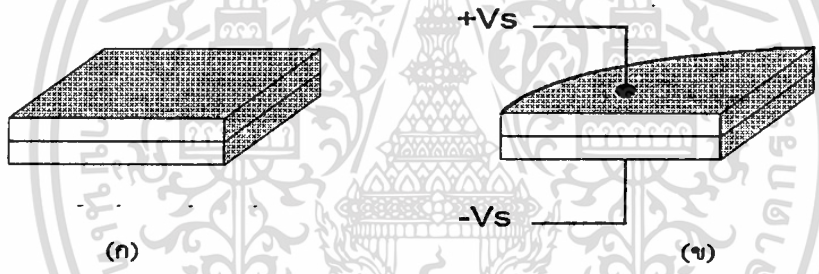
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การทำให้ Q_6 ไม่นำกระแส ซึ่งจะเป็นการทำให้มีกระแสไบอัสไหลผ่าน R22 ไหลเข้าขาเบสของ Q_4 ทำให้ Q_4 นำกระแส ทำให้มีกระแสไหลผ่านจาก Q_4 ผ่าไปยังคอยล์ , D_5 และ Q_5 ทำให้มีกระแสจ่ายให้แก่มอเตอร์และทำให้มอเตอร์หมุน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.4 อุลตร้าโซนิคทรานสดิวเซอร์

อุลตร้าโซนิค หมายถึงคลื่นเสียงที่มีความถี่สูงเกินกว่าที่หูมนุษย์จะได้ยิน ซึ่งจะสูงกว่า 20 kHz ขึ้นไป และเป็นคลื่นที่มีทิศทาง คุณสมบัติการมีทิศทางของคลื่นนี้ทำให้เรานำไปใช้งานได้หลายอย่าง เช่น ใช้ในเครื่องควบคุมระยะไกล , เครื่องล้างอุปกรณ์ , เครื่องวัดความลึกและทำแผนที่ใต้ท้องทะเล และเครื่องหาตำแหน่งอวัยวะบางส่วนของร่างกาย เป็นต้น โดยความถี่ที่ใช้ขึ้นกับการใช้งาน แต่ถ้าคลื่นเสียงต้องเดินทางผ่านอากาศแล้ว ความถี่ที่ใช้มักจะจำกัดอยู่เพียงไม่เกิน 50 kHz อุปกรณ์ที่สามารถแปลงไปมาระหว่างพลังงานไฟฟ้ากับพลังงานทางกลในย่าอุลตร้าโซนิคเรียกว่า เปียโซทรานสดิวเซอร์(Piezo-electric transducer) ภายในเปียโซทรานสดิวเซอร์จะประกอบด้วยชิ้นสารเซรามิคสี่เหลี่ยมซึ่งมีผิวโลหะเงินฉาบอยู่ทั้ง 2 หน้าเพื่อให้ต่อสายไฟออกมาเป็นขา 2 ขา ชิ้นสารเซรามิคนี้ประกอบขึ้นจากเซรามิค 2 ชิ้นประกบกันอยู่โดยวางให้ขั้วไดโพลทางไฟฟ้าภายในอะตอมของมันมีทิศทางตรงข้ามกันดังรูปที่ 2.5



รูปที่ 2.5 (ก) โครงสร้างภายในตัวอุลตร้าโซนิค เปียโซทรานสดิวเซอร์ ที่ใช้สารเซรามิค (ข) เมื่อป้อนแรงดันให้เปียโซทรานสดิวเซอร์ จะทำให้ชิ้นสารโก่งงอไปมาทำให้เกิดคลื่นเสียงอุลตร้าโซนิคกระจายไปในอากาศ

เมื่อป้อนแรงดันมาตกคร่อมขาทั้ง 2 ของสารเซรามิค ดังรูป 2.5 (ข) จะทำให้ชิ้นสารโก่งงอมากหรือน้อยหรือในทิศทางใด ตามขนาดและทิศทางของการเปลี่ยนแปลงขนาดของสัญญาณนั้นๆ ทำให้เกิดการกดอัดอากาศโดยรอบเกิดเป็นคลื่นเสียงที่มีความถี่เดียวกับสัญญาณนั้นๆ ออกไป ในทำนองกลับกันเมื่อมีคลื่นเสียงที่มีความถี่ตรงกับความถี่เรโซแนนซ์ของชิ้นสารเซรามิคเข้ามาก็จะทำให้ชิ้นสารโก่งงอไปมาและเกิดสัญญาณแรงดันซึ่งมีขนาดเล็กขึ้นมาคร่อมขั้วทั้งสองของตัวมันเอง

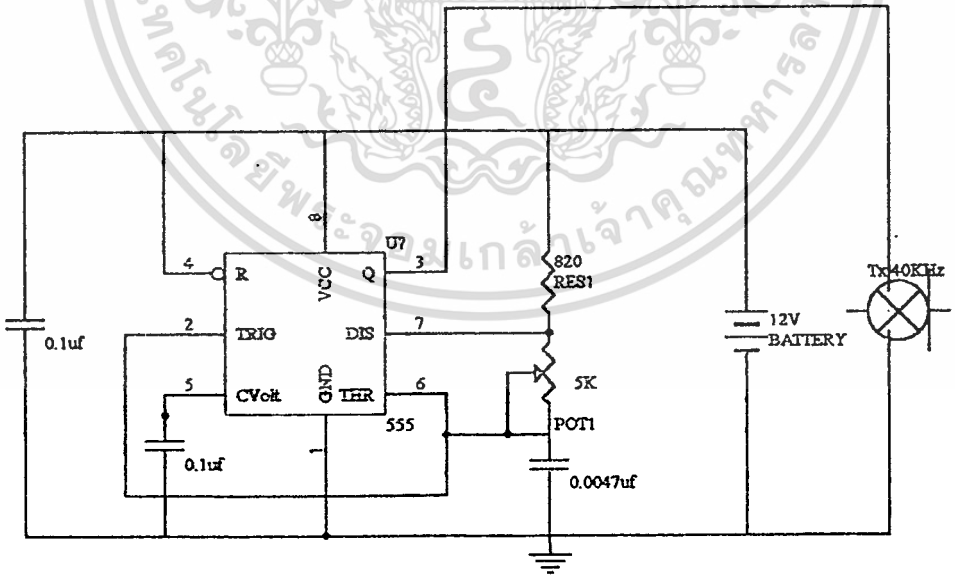
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.4.1 ตัวรับและตัวส่ง

ทรานสดิวเซอร์จะมีอยู่ 2 อย่างคือ ตัวส่ง(Transmitter) และ ตัวรับ(Receiver) ซึ่งตัวรับ คือ อุลตร้าโซนิคทรานสดิวเซอร์ที่ถูกออกแบบเจาะจงมาให้แปลงสัญญาณไฟฟ้าที่ให้แก่ตัวมันให้ออกมาเป็นคลื่นเสียงย่านอุลตร้าโซนิค ส่วนตัวรับ คือ อุลตร้าโซนิคทรานสดิวเซอร์ที่ถูกออกแบบเจาะจงมาให้แปลงคลื่นเสียงย่านอุลตร้าโซนิคที่มาตกกระทบตัวมันให้ออกมาเป็นสัญญาณไฟฟ้า

2.4.2 วงจรภาคส่ง

วิธีการสร้างเครื่องส่งเสียงอุลตร้าโซนิคทำได้โดยการสร้างวงจรกำเนิดสัญญาณรูปสี่เหลี่ยม ที่สามารถสร้างความถี่ที่ตรงกับความถี่เรโซแนนซ์ของอุลตร้าโซนิคทรานสดิวเซอร์ตัวส่งที่จะใช้ แล้วต่อคร่อมขั้วเอาท์พุททำหน้าที่เป็นโหลด วงจรภาคส่งในรูปที่ 2.6 เป็นวงจรที่ใช้ไอซี 555 มาต่อเป็นวงจรอะสเตเบิ้ลมัลติไวเบรเตอร์ ซึ่งให้สัญญาณสี่เหลี่ยมออกมาป้อนแก่ทรานสดิวเซอร์ตัวส่งความถี่ 40 kHz วงจรจะส่งเสียงอุลตร้าโซนิคทุกครั้งที่กดสวิทช์ S_1 ให้ไฟเลี้ยงเข้าวงจร R_2 มีไว้เพื่อใช้ปรับแต่งความถี่ของการออสซิเลทของวงจรให้เท่ากับความถี่เรโซแนนซ์ของตัวส่งที่ใช้



รูปที่ 2.6 วงจรภาคส่ง 40 KHz โดยใช้วงจรอะสเตเบิลมัลติไวเบรเตอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.4.3 วงจรภาครับ

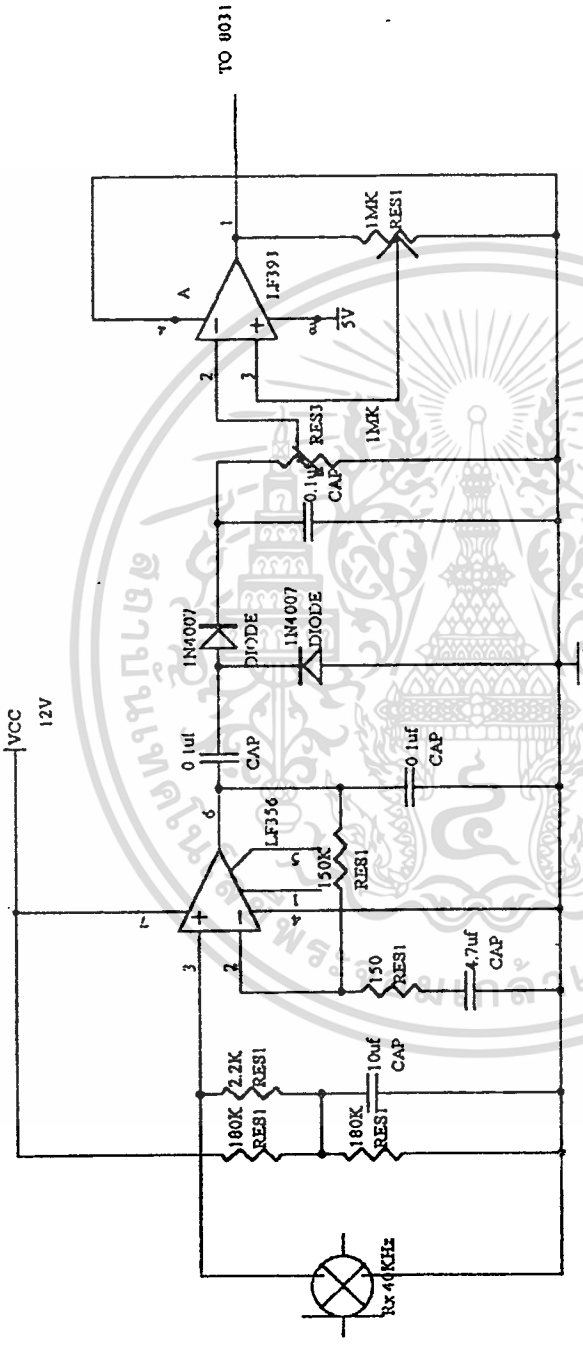
เมื่อมีคลื่นอุลตร้าโซนิคมาตกกระทบทรานสดิวเซอร์ตัวรับ จะเกิดแรงดันขนาดเล็กเป็นคลื่นรูปซายน์ขึ้นมาคร่อมตัวทรานสดิวเซอร์ซึ่งมีความถี่เดียวกับคลื่นที่มาตกกระทบ และจะมีขนาดสูงสุดเมื่อความถี่ของคลื่นเสียงที่เข้ามาตรงกับความถี่เรโซแนนซ์ของตัวรับ ส่วนที่ความถี่อื่นๆ สัญญาณจะออกมาน้อยมาก เนื่องจากระยะทางในการเดินทางของคลื่นมีผลต่อขนาดของคลื่นดังนั้นในวงจรภาครับจึงต้องมีวงจรขยายสัญญาณ เพื่อทำหน้าที่ขยายสัญญาณที่ออกมาจากตัวรับให้มีขนาดสูงพอที่จะให้ภาคเอาท์พุทแปลงสัญญาณที่ได้นี้ไปควบคุมหรือสั่งงานอุปกรณ์อื่นๆ ให้ทำงานตามต้องการต่อไป วงจรขยายในภาครับมักจะมีอัตราขยายสูงมาก ไฟเลี้ยงที่จ่ายเข้ามาจึงควรใช้วงจรเรกูเลเตอร์มาช่วยเพื่อทำให้แรงดันไฟเลี้ยงเรียบและคงที่ ทำให้วงจรเกิดเสถียรภาพในการทำงาน

วงจรในรูปที่ 2.7 เป็นวงจรภาครับซึ่งมีปรีแอมป์ภาคขยายที่มีอัตราขยายสูงและมีสัญญาณรบกวนต่ำ เนื่องจากวงจรนี้ใช้ไฟเลี้ยงชุดเดียว จึงต้องมีตัวต้านทาน R_1 และ R_3 ทำหน้าที่แปลงแรงดันไฟเลี้ยงลงมาเหลือครึ่งหนึ่งเพื่อมาป้อนไบอัสให้แก่อินพุทขา 5 ของวงจรขยาย แต่เมื่อมีสัญญาณเข้ามา C_1 จะทำหน้าที่เสมือนหนึ่งลัดวงจรจุดต่อระหว่าง R_1 และ R_3 ลงกราวด์ ทำให้ตัวรับมองเห็น R_2 เป็นโหลดที่ต่อคร่อมตัวมันเท่านั้น เป็นการช่วยเพิ่มแถบความถี่ในการทำงานของตัวรับให้กว้างขึ้น

อัตราขยายของภาคแรกจะถูกกำหนดโดยอัตราส่วนระหว่าง R_5 และ R_4 โดยมี C_2 คอยทำหน้าที่เพิ่มอัตราขยายที่ความถี่สูงๆ

เอาท์พุทจากขา 6 จะป้อนเข้า C_4 เข้ามายังภาคไดโอดปั๊ม (Diode pump circuit) ซึ่งทำหน้าที่แปลงรูปคลื่นอุลตร้าโซนิคให้ออกมาเป็นแรงดันไฟตรงที่ราบเรียบคร่อม C_5 C_5 จะคายประจุผ่าน VR_1 เข้าไปยังภาคเอาท์พุทซึ่งเป็นตัวคอมพาราเตอร์ (Comparator) จะเกิดการเปรียบเทียบค่าแรงดันที่ขาอินพุทของตัวคอมพาราเตอร์และให้เอาท์พุทออกมา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.7 แสดงวงจรภาครับคลื่นอุลตร้าโซนิค

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

038375

2.5 สเตปปิ้งมอเตอร์ (Stepping Motor)

สเตปปิ้งมอเตอร์เป็นอุปกรณ์จำพวกเชิงกลทางไฟฟ้า ที่มีอินพุทเป็นกลุ่มของไบนารีโลเจคและเป็นลักษณะของการเคลื่อนที่แบบเชิงมุม หรือหมุนไปตามสเตป (แต่ละสเตปอยู่ในช่วง 0.1 ถึง 90 องศา ขึ้นอยู่กับโครงสร้างของมอเตอร์) ตามสัญญาณพัลซ์ป้อนให้กับขดสเตเตอร์ จะเกิดแรงผลักต่อโรเตอร์ทำให้มอเตอร์หมุนไป แต่ลักษณะของสเตปปิ้งมอเตอร์จะมีขดของสเตเตอร์อยู่หลายขดซึ่งเรียกว่า "เฟส" ดังนั้นเมื่อป้อนสัญญาณซึ่งเป็นพัลซ์ในลักษณะซีเควนของเลขไบนารีโดยผ่านวงจรไดรเวอร์ (Driver) จะทำให้โรเตอร์หมุนได้อย่างต่อเนื่อง

2.5.1 คุณประโยชน์ของสเตปปิ้งมอเตอร์

ข้อดีของระบบควบคุมตำแหน่งโดยใช้สเตปปิ้งมอเตอร์

1. เป็นลักษณะการควบคุมแบบไม่ต้องการการป้อนกลับ ไม่ว่าจะเป็นการควบคุมตำแหน่งหรือความเร็ว
2. ความผิดพลาดเกี่ยวกับตำแหน่งแทบไม่มีเลย เนื่องจากการเคลื่อนที่ของสเตปปิ้งมอเตอร์ จะเคลื่อนที่เป็นสเตปด้วยจำนวนองศาที่แน่นอน
3. สเตปปิ้งมอเตอร์จะถูกนำมาใช้กับเครื่องมือที่ต้องการความละเอียดแม่นยำ
4. ไม่จำเป็นต้องใช้วงจรแปลงดิจิตอลเป็นอนาลอกเมื่ออินเตอร์เฟสกับไมโครอมพิวเตอร์
5. สเตปปิ้งมอเตอร์เป็นอุปกรณ์ที่เปลี่ยนสัญญาณดิจิตอลเป็นการเคลื่อนที่ทางกล ดังนั้นการติดต่อกับอุปกรณ์ดิจิตอลเป็นไปได้โดยง่าย
6. การออกแบบวงจรควบคุมสามารถทำได้ง่าย และให้ทำงานหรือหยุดได้แบบทันทีทันใด

2.5.2 ชนิดของสเตปปิ้งมอเตอร์

ได้แก่

1. VR Motor
2. PM Stepping Motor
3. Hybrid Stepping Motor
4. Linear Stepping Motor
5. Monofilar and Bifiler Winding

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สำหรับมอเตอร์ที่ใช้ในโครงงานนี้เป็นชนิด VR Motor แบบ Smaller Step Angles ซึ่งเป็นแบบที่มี step ในการทำงานที่ละเอียดจะพบว่ามีฟันของ stator มากเพื่อทำให้ step แต่ละ step มีมุมเล็กลงซึ่งมีสูตรในการคำนวณดังนี้

$$S = 360/Q_S$$

$$S = mN_R$$

โดย

S = จำนวนของ Step

m = จำนวนของเฟส

N_R = จำนวนของโรเตอร์

Q_S = มุมระหว่าง Step

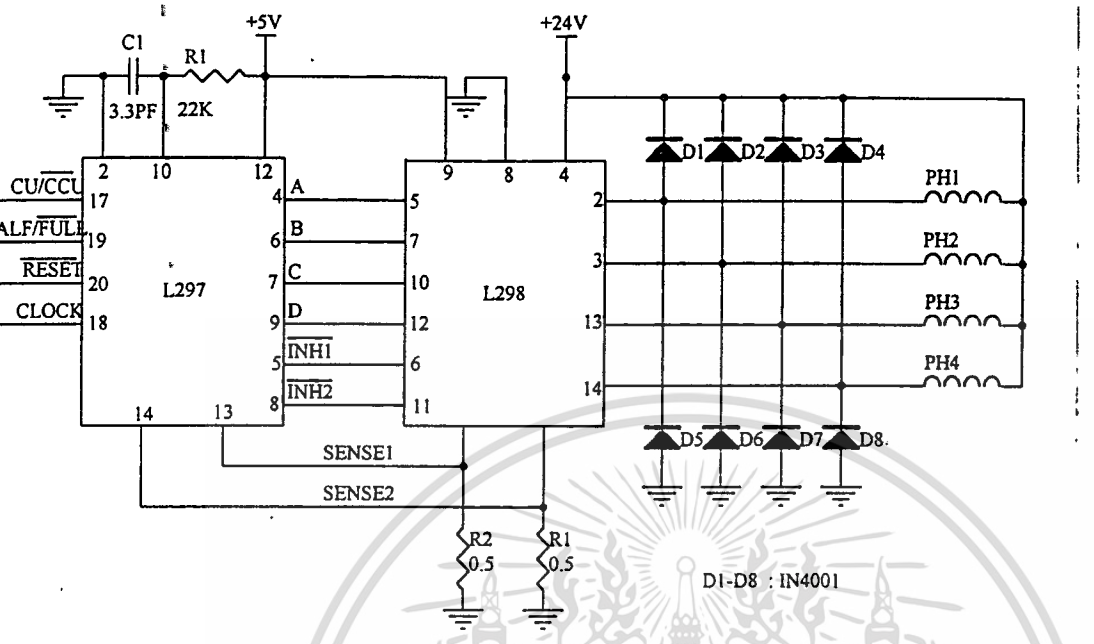
2.5.3 หลักการทำงานของวงจรขับสเตปปิ้งมอเตอร์

โดยใช้ IC เบอร์ L297 และ L298 ดังรูปที่ 2.8 ซึ่ง IC เบอร์ L297 เป็น IC ควบคุมการขับมอเตอร์ โดยเป็นตัวรับสัญญาณนาฬิกา (Clock) , สัญญาณควบคุมความเร็วของมอเตอร์ , สัญญาณรีเซต และสัญญาณควบคุมให้มอเตอร์หมุนไปตามทิศทางต่างๆ ด้วยขา 18 , ขา 19 , ขา 20 และ ขา 17 ตามลำดับ การควบขาอินพุทต่างๆ ใช้ระดับสัญญาณลอจิก "0" และ "1" (ใช้แรงดัน 0 โวลต์ และ 5 โวลต์) เป็นตัวควบคุม คือ

- ที่ ขา 18 ถูกป้อนด้วยสัญญาณนาฬิกาต่อเนื่อง
- ที่ ขา 19 ถูกป้อนด้วยลอจิก 1 ตลอดเวลา ซึ่งจะทำให้มีเพียงความเร็วเดียว
- ที่ ขา 20 ถูกป้อนด้วยลอจิก 1 ตลอดเวลา เมื่อต้องการให้มอเตอร์หยุดหมุนก็ป้อนด้วยลอจิก 0
- ที่ ขา 17 เมื่อป้อนด้วยลอจิก 1 มอเตอร์จะหมุนตามเข็มนาฬิกา และถ้าป้อนลอจิก 0 มอเตอร์จะหมุนทวนเข็มนาฬิกา

จากนั้นเอาพุทของ IC เบอร์ L297 จะไปเข้า IC เบอร์ L298 ซึ่งเป็น IC ขับมอเตอร์ IC เบอร์ L298 เป็น IC ที่ทำหน้าที่ขยบกระแสสัญญาณที่ได้จากขา A , B , C และ C ของขา IC เบอร์ L297 เพื่อให้กระแสเพียงพอที่มอเตอร์หมุนได้ ส่วน Diode D1 ถึง D8 ทำหน้าที่ป้องกันการยุบตัวของสนามแม่เหล็ก ซึ่งจะทำให้เกิดกระแสไหลย้อนกลับและมีผลให้ IC เสียหายได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

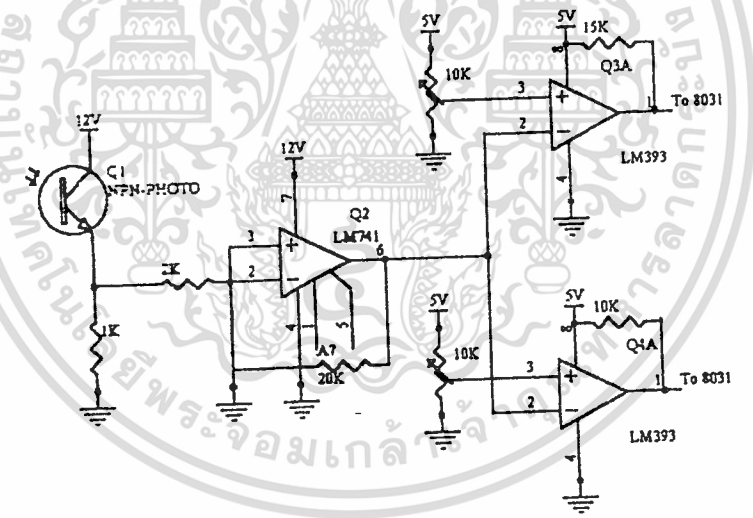


รูปที่ 2.8 แสดงวงจรขับสเตปปิ้งมอเตอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.6 วงจรตรวจจับแสง (Photo Detector Circuit)

คือวงจรตรวจจับสัญญาณคลื่นอินฟาเรด จากรูปที่ 2.9 เมื่อมีคลื่นอินฟาเรดมาตกกระทบตัว Photo Detector จะมีกระแสไหลซึ่งแปรตามความเข้มของคลื่นที่ตกกระทบ จะทำให้ทรานซิสเตอร์แบบ เอ็นพีเอ็นทำงาน และเกิดสัญญาณขึ้นที่ขาอีมิตเตอร์ของทรานซิสเตอร์ นำไปผ่านวงจรขยาย และทำการเปรียบเทียบสัญญาณโดยผ่านคอมแพราเตอร์ (Comparator) 2 ส่วน โดยส่วนแรก เปรียบเทียบกับสัญญาณอ้างอิงเพื่อกำหนดระยะตรวจจับระยะใกล้ และส่วนที่ 2 เปรียบเทียบกับสัญญาณอ้างอิงเพื่อกำหนดระยะตรวจจับระยะไกล สัญญาณที่ได้จากคอมแพราเตอร์ คือเอาท์พุทที่จะป้อนปยังไมโครคอลโทรลเลอร์



รูปที่ 2.9 แสดงวงจรตรวจจับแสง (Photo Detector Circuit)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.7 เข็มทิศอิเล็กทรอนิกส์

เข็มทิศอิเล็กทรอนิกส์ เป็นอุปกรณ์อิเล็กทรอนิกส์ที่สามารถบอกทิศทางที่อุปกรณ์ชี้อยู่เหมือนกับเข็มทิศธรรมดาทั่วไป โดยให้สัญญาณเอาพุทเป็นสัญญาณดิจิตอลขนาด 16 บิต สามารถวัดได้ตั้งแต่ 0 องศา ถึง 360 องศา โดยอ้างอิงทิศตะวันออกเป็น 0 องศา

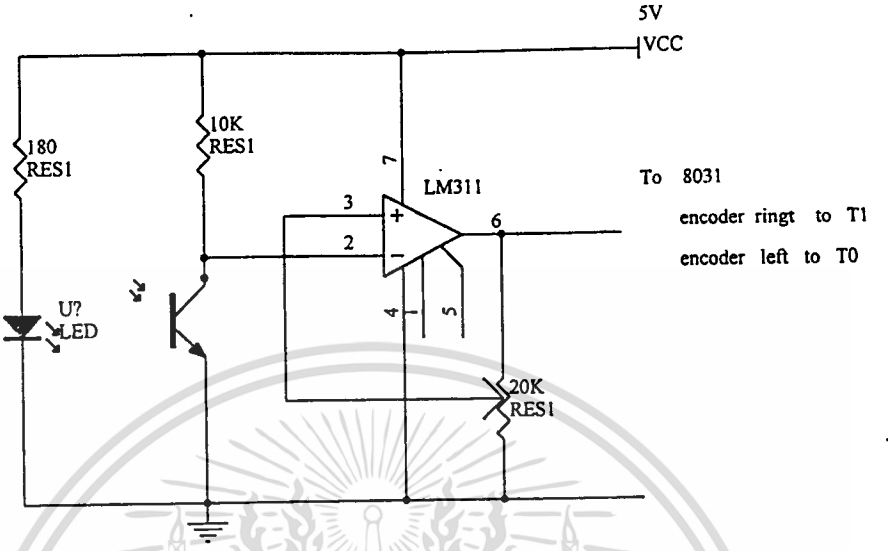
2.7.1 การประยุกต์ใช้งาน

สำหรับโครงงานนี้จะนำเข็มทิศอิเล็กทรอนิกส์มาช่วยในการอ้างอิงตำแหน่งตัวรถ เพื่อให้รู้ว่าขณะนั้นรถหันหน้าไปทางทิศใด และเพื่อช่วยให้การหมุนของรถอย่างแน่นอน เช่น ขณะหนึ่งรถอยู่ที่ตำแหน่ง 100 องศา ถ้าเราต้องการให้รถหมุนไปทางขวาอีก 90 องศา นั้นคือหน้ารถต้องอยู่ที่ตำแหน่ง 190 องศา เราจึงเขียนโปรแกรมให้รถหมุนไปทางขวา แล้วรับสัญญาณจากเข็มทิศอิเล็กทรอนิกส์เพื่อตรวจสอบว่าเท่ากับ 190 องศา หรือยัง ถ้ายังก็ให้รถหมุนไปเรื่อยๆ และเมื่อถึงแล้วก็ให้รถหยุดหมุน ซึ่งจะเห็นว่าทำให้เราสามารถควบคุมทิศทางของรถได้แน่นอน

2.8 วงจรเช็คความเร็วล้อ (Encoder Circuits)

เป็นวงจรที่ใช้วัดความเร็วของล้อ เพื่อประโยชน์ในการควบคุมให้ล้อทั้งด้านซ้ายและขวาเคลื่อนที่ไปพร้อมกันหรือสัมพันธ์กันซึ่งวงจรนี้ถูกติดไว้ที่ล้อด้านหลังทั้งสองข้างของรถ ลักษณะวงจรเป็นดังรูปที่ 2.10 มีอุปกรณ์สำคัญคือ Slotted Optical Switches ทำหน้าที่เป็นตัวเซนเซอร์แสงที่ผ่านสล็อตของจานวงกลมที่ได้เจาะรู 24 รูเพื่อแบ่งองศาระหว่างรูออกเป็น 15 องศา การทำงานของวงจรคือ เมื่อรถมีการเคลื่อนที่จะทำให้จานวงกลมหมุนตาม แสงจะตัดผ่านจานวงกลมเป็นช่วงๆ ในช่วงที่แสงผ่านรูของจานวงกลมจะตกกระทบตัว Slotted Optical Switches จะเกิดกระแสไหลผ่านทรานซิสเตอร์ เกิดสัญญาณที่ขาคอลเลคเตอร์ สัญญาณนี้ถูกนำไปผ่านออปแอมป์ขยายสัญญาณไปให้กับไมโครคอลโทรลเลอร์ การเคลื่อนที่ของล้อจะสัมพันธ์กันโดยจำนวนสัญญาณที่เกิดขึ้นที่ด้านซ้ายและขวาของล้อต้องเท่ากัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

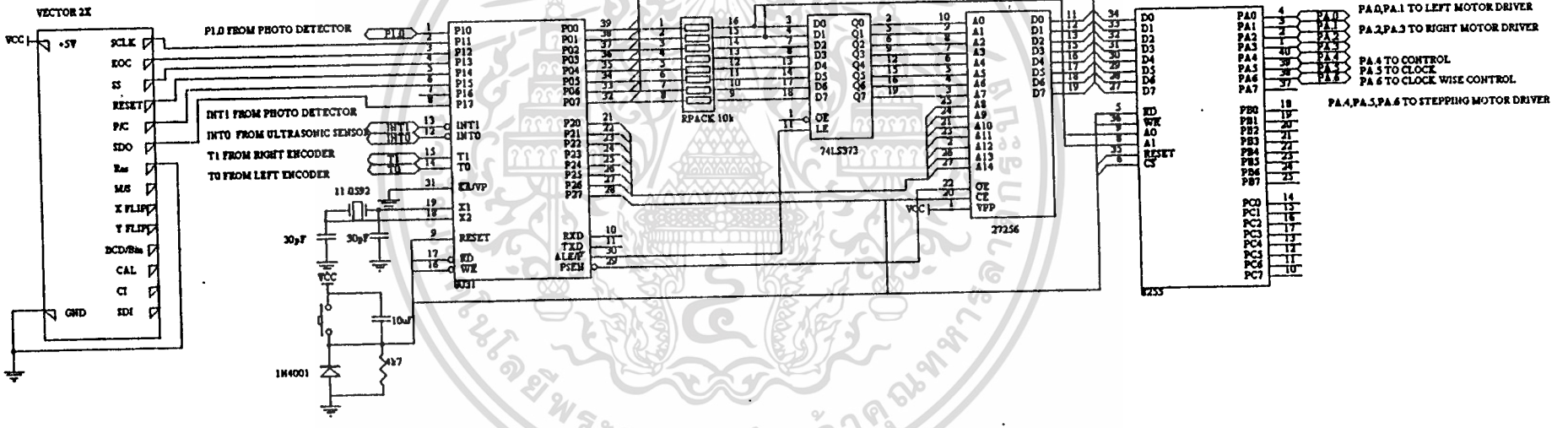


รูปที่ 2.10 แสดงวงจร Encoder

2.9 ลักษณะการต่อวงจรต่างๆ เข้ากับพอร์ตของไมโครคอลโทรลเลอร์

เป็นการนำวงจรต่างๆที่ใช้ในการควบคุมการทำงานของรถเข้ามาต่อกับพอร์ตของไมโครคอลโทรลเลอร์และพอร์ตของ 8255 ซึ่งเป็นดังรูปที่ 2.11

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.11 แสดงการต่อวงจรต่างๆ เข้ากับไมโครคอลโทรลเลอร์

บทที่ 3

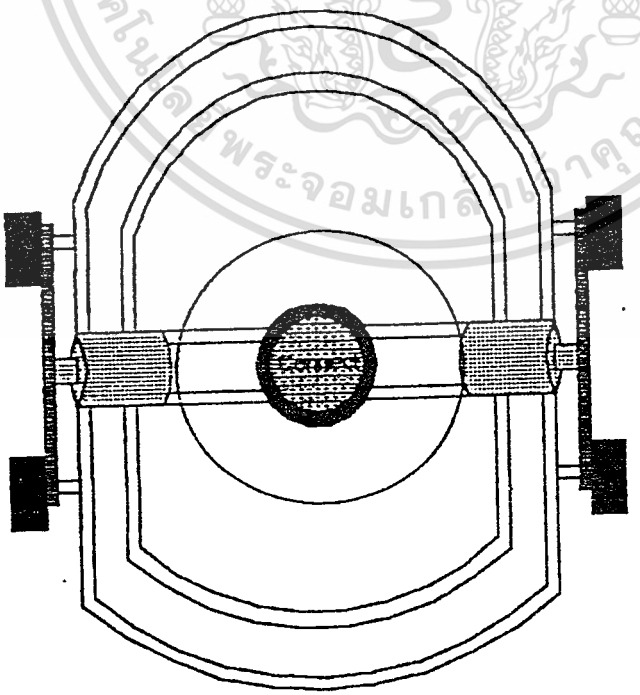
การทดลองและผลการทดลอง

3.1 การออกแบบ และสร้างตัวรถส่วน Mechanic

ลักษณะของรถประกอบด้วย

1. มีขนาด 57 * 63 เซนติเมตร ส่วนหน้าจะโค้งเป็นครึ่งวงกลม ส่วนหลังตัดตรง
2. โครงสร้างจะใช้แท่งอะลูมิเนียมกลวง
3. ประกอบด้วยล้อ 4 ล้อ ใช้โซคล้องระหว่างล้อหน้า และล้อหลัง โดยผ่านเฟืองที่ต่ออยู่กับมอเตอร์กระแสตรง ทั้งด้านซ้าย และด้านขวา
4. มีวงจรควบคุมการทำงานอยู่ด้านบนของตัวรถ
5. ใช้การขับเคลื่อนด้วยแบตเตอร์รี่ขนาด 12 โวลต์

จากการออกแบบจะได้รูปลักษณ์ของรถ ดังรูปที่ 3.1



รูปที่ 3.1 แสดงโครงสร้างของรถบังคับ MPV

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.2 การทดสอบวงจรขับเคลื่อน (Driver Circuit)

จากวงจรขับเคลื่อนที่ได้สร้างขึ้นดังรูปที่ 2.4 นำมาทดสอบการขับเคลื่อนโดย

1. ทดสอบหมุนของมอเตอร์ ด้วยการป้อนอินพุท 5 โวลต์ ที่ขา (a) และ ขา (b) แล้วสังเกตทิศทางการหมุนของมอเตอร์

ผลการทดลอง

อินพุท		เอาท์พุท
ขา (a)	ขา (b)	
0	0	มอเตอร์ไม่หมุน
0	1	มอเตอร์หมุนไปทางซ้าย
1	0	มอเตอร์หมุนไปทางขวา
1	1	มอเตอร์ไม่หมุน

2. ทดสอบลักษณะการขับเคลื่อนรถ โดยการป้อนอินพุท 5 โวลต์ ให้กับวงจรขับเคลื่อนทั้ง 2 ด้าน (ทางซ้าย และขวา)

ผลการทดลอง

อินพุทวงจรด้านซ้าย		อินพุทวงจรด้านขวา		เอาท์พุท
(a)	(b)	(a)	(b)	
1	0	1	0	เดินหน้า
0	1	0	1	ถอยหลัง
1	0	0	1	เลี้ยวซ้าย
0	1	1	0	เลี้ยวขวา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.3 การทดสอบวงจรตรวจจับสิ่งกีดขวางโดยใช้อุลตร้าโซนิก (Ultrasonic Sensor Circuits)

จากวงจรรูปที่ 2.6 และ 2.7 คือวงจรภาคส่ง และภาครับของวงจรตรวจจับโดยใช้คลื่นอุลตร้าโซนิค ในการทดสอบวงจรนี้ จะกระทำโดย

1. ให้วงจรภาคส่ง ส่งคลื่นอุลตร้าโซนิคที่ความถี่ 40 KHz ออกไป โดยตอนแรกจะไม่มีสิ่งกีดขวางใดๆ ทำการวัดแรงดันตกเอาท์พุทที่ภาครับ
2. นำสิ่งกีดขวางมาวาง แล้วทำการวัดแรงดันเอาท์พุทที่ภาครับอีกครั้ง
3. วัดระยะทางของสิ่งกีดขวางกับวงจรภาครับ

ผลการทดลอง

เมื่อไม่มีสิ่งกีดขวาง	เอาท์พุทเป็น 1
เมื่อนำสิ่งกีดขวางมาวาง	เอาท์พุทเป็น 0

ระยะทางที่ภาครับจะรับคลื่นสะท้อนของภาคส่ง ห่างจากสิ่งกีดขวางประมาณ 30 เซนติเมตร

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.4 การทดสอบวงจรขับเคลื่อนสเตปปิ้งมอเตอร์ (Stepping Motor Driver Circuits)

จากวงจรรูปที่ 2.8 คือ วงจรขับเคลื่อนสเตปปิ้งมอเตอร์ ทำการทดสอบการหมุนของมอเตอร์โดย

1. ให้สัญญาณนาฬิกา (Clock) เข้าที่ขา 18 ตลอดเวลา
2. ให้แรงดันค่า 5 โวลต์ ที่ขา 20
3. ให้ระดับสัญญาณลอจิกเป็น "1" (แรงดันค่า 5 โวลต์) ที่ขา 17
4. สังเกตุการหมุนของมอเตอร์ บันทึกผลการทดลอง
5. เปลี่ยนระดับสัญญาณลอจิกเป็น "0" (ค่าแรงดัน เป็น 0 โวลต์) ที่ขา 17
6. สังเกตุการหมุนของมอเตอร์ บันทึกผลการทดลอง
7. เปลี่ยนแรงดันที่ขา 20 เป็น 0 โวลต์
8. ทำการทดลองเช่นเดียวกับข้อ 3 - 6 แล้วบันทึกผลการทดลอง

ผลการทดลอง

เมื่อขา 20 เป็น 5 โวลต์

สัญญาณที่ขา 17	ลักษณะการหมุนของมอเตอร์
มีระดับลอจิก 1	หมุนตามเข็มนาฬิกา
มีระดับลอจิก 0	หมุนทวนเข็มนาฬิกา

เมื่อขา 20 เป็น 0 โวลต์

สัญญาณที่ขา 17	ลักษณะการหมุนของมอเตอร์
มีระดับลอจิก 1	ไม่หมุน
มีระดับลอจิก 0	ไม่หมุน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.5 การทดสอบวงจรตรวจจับเป้าหมาย (Photo Detector)

จากวงจรรูปที่ 2.9 คือ วงจร Photo Detector ซึ่งใช้เป็นตัวตรวจจับเป้าหมาย สามารถทำการทดสอบวงจร ดังนี้

จากวงจรที่ 2.9 ทำการทดสอบการรับแสงอินฟาเรท แล้วแสดงผลโดย นำไฟฉาย (แหล่งกำเนิดแสงอินฟาเรท) มาส่องที่ตัวตรวจจับแสง สังเกตว่าสัญญาณที่เอาท์พุทเป็นอย่างไร

ผลการทดลอง

เมื่อมีแสงจากไฟฉาย	เมื่อไม่มีแสงจากไฟฉาย
มีสัญญาณลอจิก 0	มีสัญญาณลอจิก 1

ทดลองระยะการจรวจจับโดยการปรับ ค่า R ปรับค่าได้ทั้ง 2 ตัว แล้วทำการส่องแสงจากไฟฉายไปยังตัวตรวจจับ เปลี่ยนระยะการตรวจจับไปจนการทั้ง เอาท์พุทที่ขา 1 และ ขา 7 ของ IC เบอร์ 393 เป็น 0

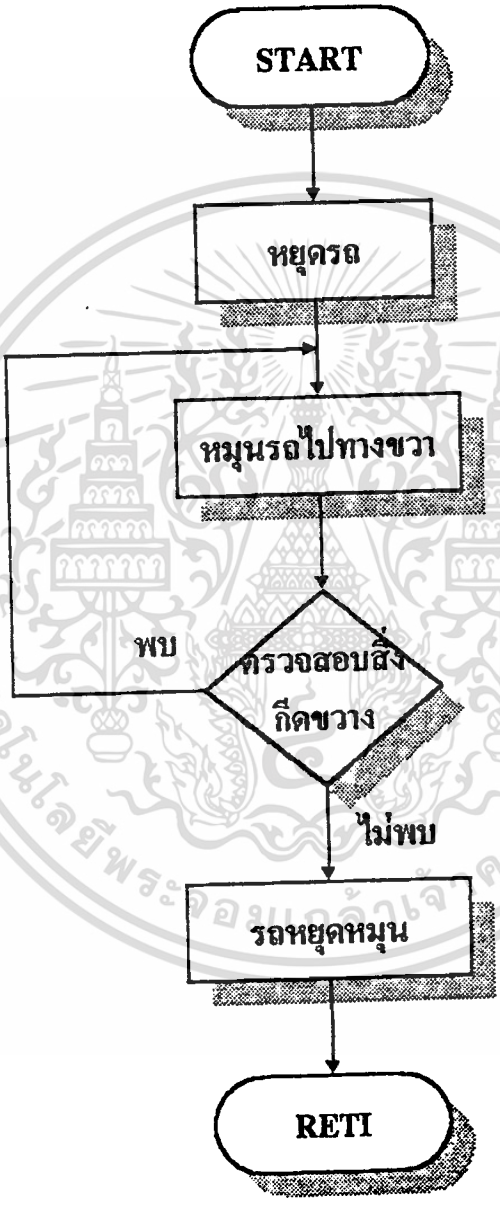
ผลการทดลอง

ระยะทางที่จับได้ ที่ เอาท์พุทขา 1	ระยะทางที่จับได้ ที่ เอาท์พุทขา 7
50 เซนติเมตร	120 เซนติเมตร

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

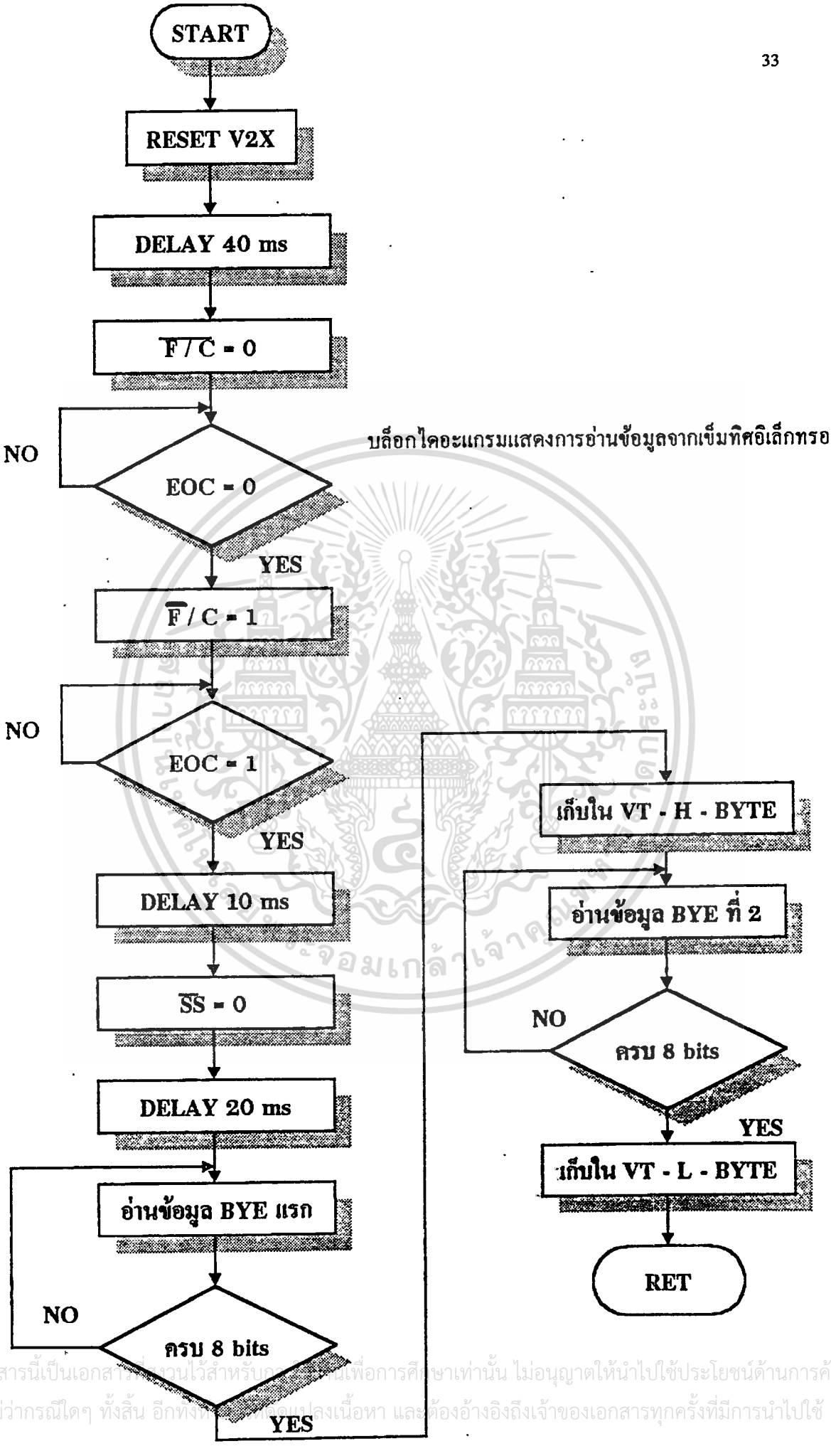
บทที่ 4
รูปแบบและโปรแกรมการทำงานของรถ

4.1 บล็อกไดอะแกรมการทำงานของรถ



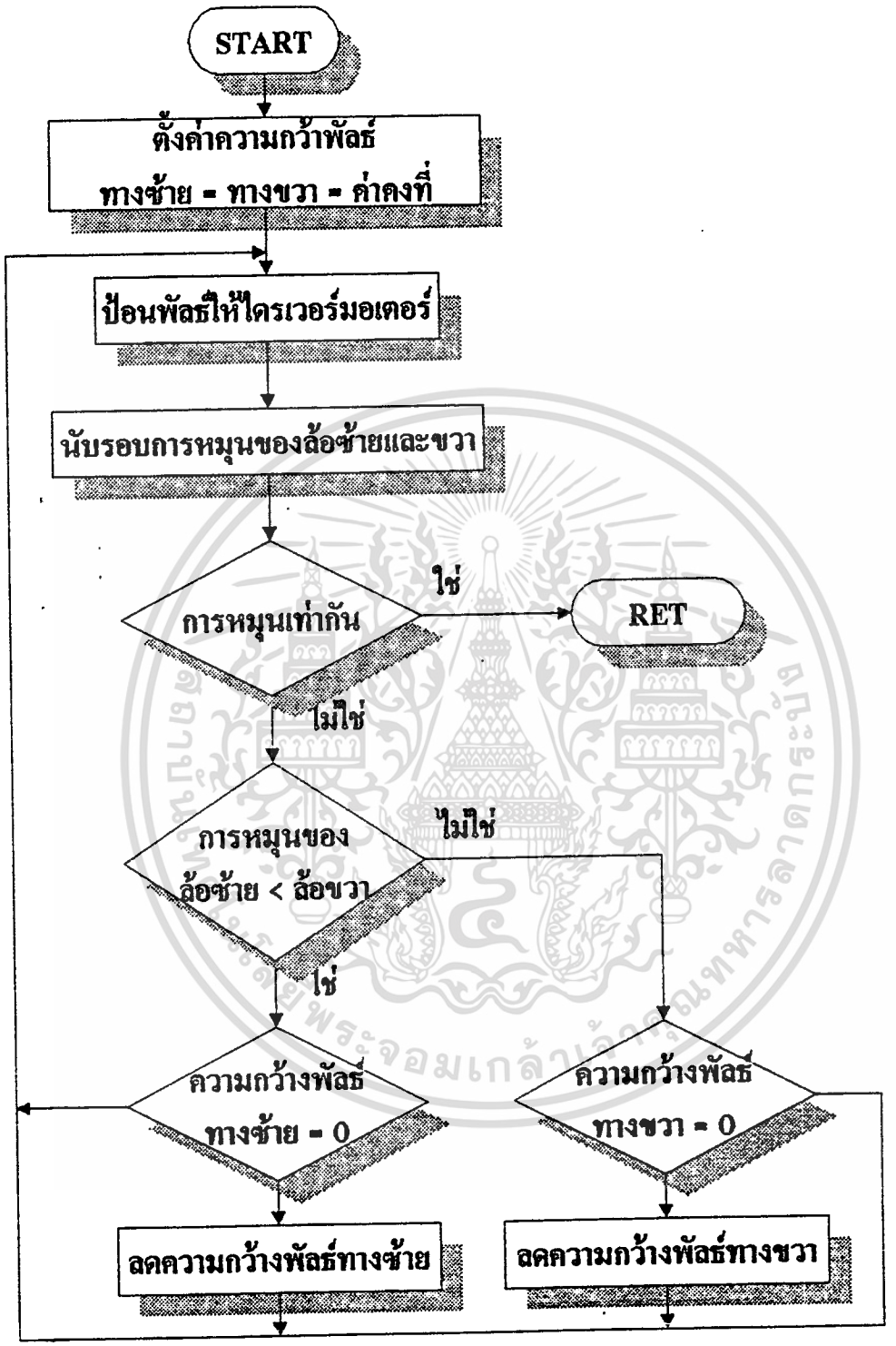
บล็อกไดอะแกรมแสดงการหลบสิ่งกีดขวางเมื่อมีสัญญาณพบสิ่งกีดขวาง อินเตอร์รัพท์เข้ามา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



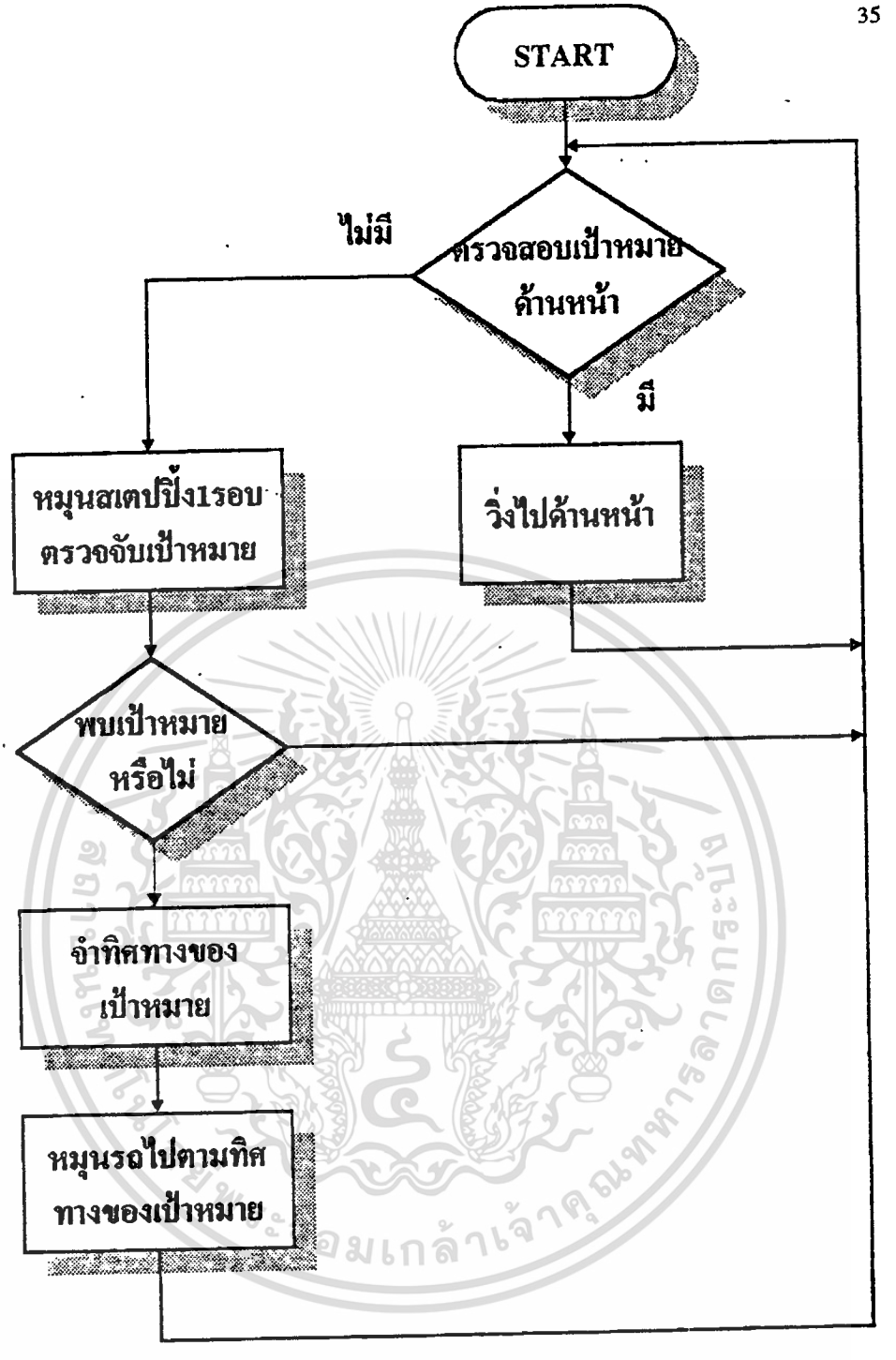
บล็อกไดอะแกรมแสดงการอ่านข้อมูลจากเข็มทิศอิเล็กทรอนิก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งไม่อนุญาตให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



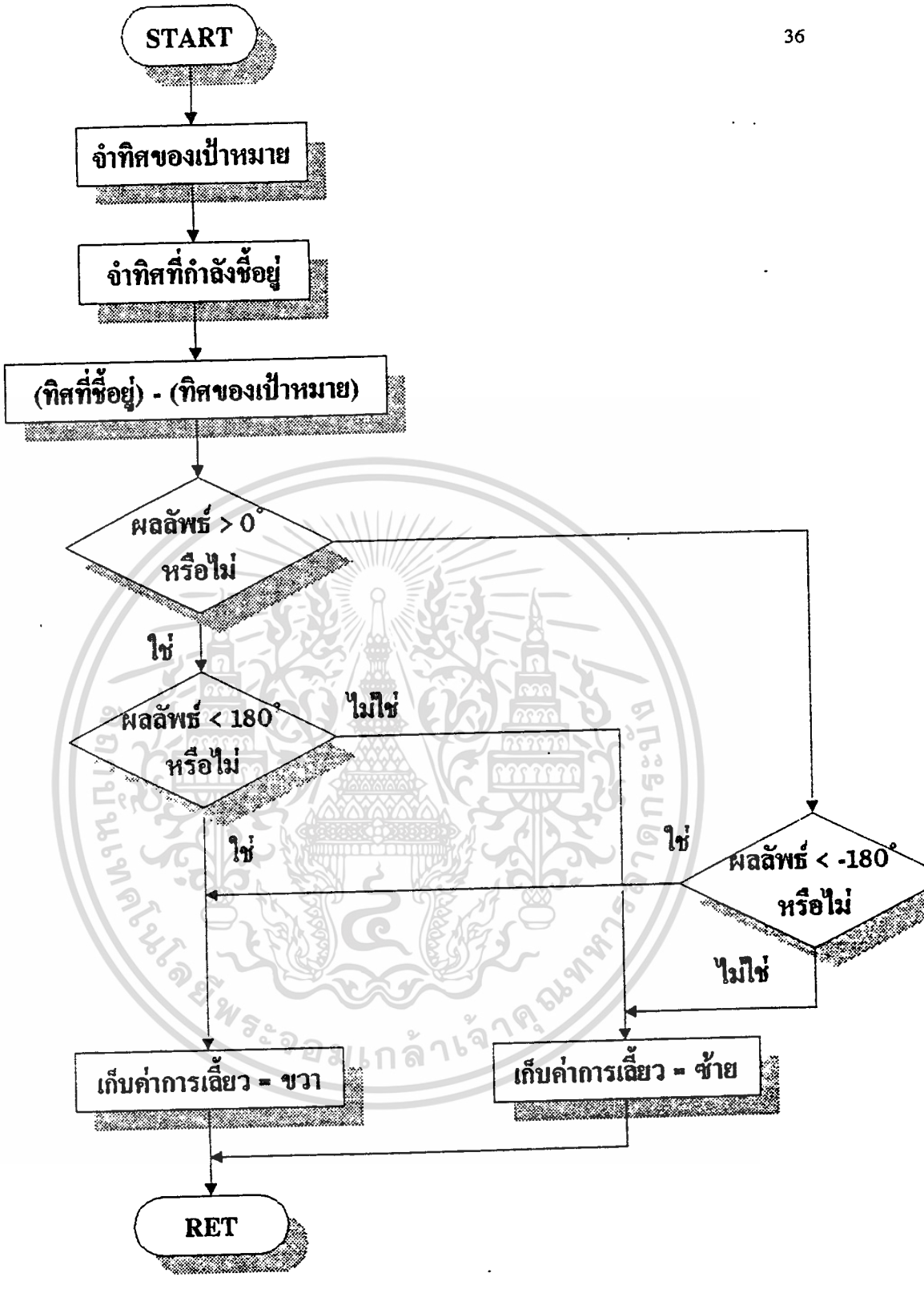
บล็อกไดอะแกรมแสดงการเลือกทิศทางการหมุนของรถ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



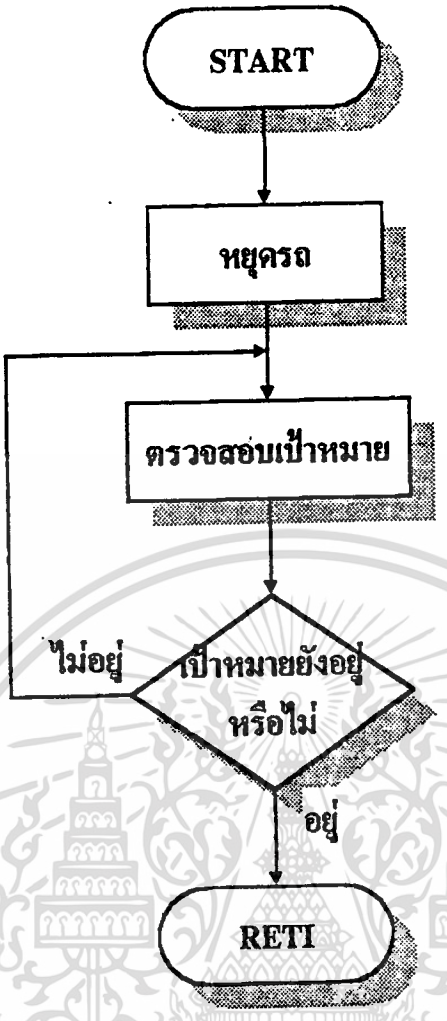
บล็อกไดอะแกรมแสดงการทำงานในการตรวจสอบเป้าหมายและวิ่งตามเป้าหมาย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



บล็อกไดอะแกรมแสดงการควบคุมการหมุนของรถ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



บล็อกไดอะแกรมแสดงการจบการทำงานเมื่อเป้าหมายอยู่ใกลรถในระยะที่ตั้งไว้
โดยรับสัญญาณอินเตอร์รัพท์เข้ามา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.2 โปรแกรมการทำงานของรถ

```
PORT_A      EQU     00H
  CWD       EQU     03H

REM_L       EQU     20H
REM_H       EQU     21H
SUB_L       EQU     22H
SUB_H       EQU     23H
ANS_L       EQU     24H
ANS_H       EQU     25H

NOW_D_H     EQU     26H
NOW_D_L     EQU     27H
TAR_D_H     EQU     28H
TAR_D_L     EQU     29H

VT_H_BYTE   EQU     2AH
VT_L_BYTE   EQU     2BH
SHF_BUFF    EQU     2CH

CHK_RND     EQU     2DH
N_CLOCK     EQU     2EH
STEP_BUF    EQU     2FH

FWD_TMP     EQU     30H
TR_TMP      EQU     31H
TL_TMP      EQU     32H

FWD_BUF     EQU     33H
TR_BUF      EQU     34H
TL_BUF      EQU     35H
TURN_BUF    EQU     36H

PA_BUFFER   EQU     37H

R_ST_BUF    EQU     01000000B
L_ST_BUF    EQU     01010000B

;*************************************************************************
;                       START PROGRAM
;*************************************************************************

        ORG  0000H

        LJMP MAIN

        ORG  0003H

        LJMP OBSTRUCT

        ORG  0013H

        LJMP MEET
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
;*************************************************************************
;                          MAIN PROGRAM
;*************************************************************************

MAIN:     MOV SP,#0007H
          MOV A,#80H                ;
          MOV DPTR,#CWD             ; SET 8255 PORT A IS OUTPUT
          MOVX @DPTR,A              ;

          MOV DPTR,#PORT_A
          MOV STEP_BUF,#00H         ; CLEAR STEP_BUF
          MOV PA_BUFFER,#00H
          MOV TMOD,#01010101B       ; SET MODE COUNTER1,2
          MOV TCON,#01010000B       ; SET COUNTER CONTROL

          SETB IE.0                 ; ENABLE INTERRUPT 0
          SETB IE.2                 ; ENABLE INTERRUPT 1
          SETB IE.7                 ; ENABLE INTERRUPT
          SETB IP.2                 ; INT1 PRIORITY
          CLR  P1.6

START:    MOV R2,#15D               ;
LP_CHK1:  MOV R1,#100D              ;
LP_CHK2:  MOV R0,#250D              ;

CHK_F:    SETB P1.0                 ;
          MOV A,P1                  ; DETECT TARGET AT FRONT OF VEHICLE
          JNB ACC.0,GO_AHADE        ;

          DJNZ  R0,CHK_F            ;
          DJNZ  R1,LP_CHK2          ;
          DJNZ  R2,LP_CHK1          ;

          LJMP ROTATE

GO_AHADE: LCALL RUN_FWD             ; RUN TO FORWARD
          MOV A,P1                  ;
          JNB ACC.0,GO_AHADE        ; CHECK TARGET IS STILL STAY
          LJMP BRAKE

ROTATE:   MOV  R2,#200D             ; NUMBER OF PULSES WIIL FEED TO STEPPIG MORTOR
          MOV N_CLOCK,#00H          ; NUMBER OF PULSES HAVE FED TO STEPPIG MORTOR

          MOV A,STEP_BUF            ; STEP_BUF KEEP THE ROTATE DIRECTION OF STEPPING MORTOR
          JNZ L

R:        LCALL R_STEP              ;
          INC N_CLOCK               ;
          MOV A,P1                  ; TURN STEPPING MORTOR RIGHT AND DETECT TARGET
          JNB ACC.0,DETECT          ; ULTIL FOUND OR COMPLETE 1 ROUND
          DJNZ R2,R                 ;
          LJMP REVERSE              ;

L:        LCALL L_STEP              ;
          INC N_CLOCK               ;
          MOV A,P1                  ; TURN STEPPING MORTOR LEFT AND DETECT TARGET
          JNB ACC.0,DETECT          ; ULTIL FOUND OR COMPLETE 1 ROUND
          DJNZ R2,L                 ;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
REVERSE:    MOV A,STEP_BUF          ;
            CPL ACC.0               ; REVERSE ROTATION DIRECTION OF STEPPING MORTOR
            MOV STEP_BUF,A          ;
            LJMP START              ;

DETECT:     LCALL VECTOR            ;
            MOV  TAR_D_H,VT_H_BYTE  ; READ VECTOR DATA AND KEEP IN BUFFER
            MOV  TAR_D_L,VT_L_BYTE  ;

            MOV R2,N_CLOCK          ;
            MOV A,STEP_BUF          ;
            JNZ REV_L               ;

REV_R:      LCALL L_STEP
            DJNZ R2,REV_R           ; ROTATE STEPPING MORTOR LEFT
            LJMP LOCK

REV_L:      LCALL R_STEP            ; ROTATE STEPPING MORTOR RIGHT
            DJNZ R2,REV_L

LOCK:       LCALL VECTOR            ;
            MOV NOW_D_H,VT_H_BYTE   ; READ VECTOR DATA OF THE FRONT OF VEHICLE
            MOV NOW_D_L,VT_L_BYTE   ; AND KEEP IN BUFFER

            MOV REM_H,NOW_D_H       ;
            MOV REM_L,NOW_D_L       ; COMPARE DIRECTION OF FRONT OF VEHICLE
            MOV SUB_H,TAR_D_H       ; WITH DIRECTION OF TARGET
            MOV SUB_L,TAR_D_L       ;
            LCALL SUB_2_B           ;

            MOV A,ANS_H             ;
            JNZ FOLLOW              ; CHECK DIFFERENCE OF ANSWERS IF IT IS EQUAL
            MOV A,ANS_L             ;
            JZ BRAKE                ;

FOLLOW:     LCALL T_LOCK            ;
            MOV A,TURN_BUF          ; SELECT THE DIRECTION TO TURN
            JZ T_RIGHT              ;

T_LEFT:     LCALL TURN_L            ; TURN LEFT THE VECHICLE AND JUMP TO CHECK
            LJMP LOCK               ; THE EQUALATION OF TARGET AND FRONT OF VEHICLE DIRECTION

T_RIGHT:    LCALL TURN_R            ; TURN RIGHT THE VECHICLE AND JUMP TO CHECK
            LJMP LOCK               ; THE EQUALATION OF TARGET AND FRONT OF VEHICLE DIRECTION

BRAKE:      LCALL STOP              ; STOP THE VECHICLE.
            LJMP START              ;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
;*************************************************************************
;  PROGRAM:      AVOILD OBSTRUCT
;                { INTTRUPT (INT0) PROGRAM }
;
; REGISTOR:      A,R1,TH0,TL0
;  ROUTINE:      TURN_L,STOP
;*************************************************************************

 OBSTRUCT:      PUSH ACC
                PUSH 01H                ; PUSH R1
                PUSH 8AH                ; PUSH TL0
                PUSH 8BH                ; PUSH TL1

                MOV  R1,#36D

 CLEAR_C2:      MOV TL0,#00H
                MOV TL1,#00H

    AVOID:      MOV A,TURN_BUF

                JZ AL_R
                LCALL TURN_L
                LJMP CHK_P2
     AL_R:      LCALL TURN_R

   CHK_P2:      CLR CY                  ;
                MOV A,TL0               ; COMPARE ROTATE OF RIGHT AND LEFT DC MOTOR
                SUBB A,TL1              ;
                JNZ NOT_EQU             ;

                MOV A,R1                ;
                SUBB A,TL0              ; CHECK ROTATION OF DC MOTOR
                JNZ AVOID               ;
                LJMP NO_DET             ;

  NOT_EQU:      JC STOP_R_M2            ; PULSE FROM R MOTOR > L MOTOR

 STOP_L_M2:     MOV A,PA_BUFFER         ;
                ANL A,#11111100B        ; STOP LEFT MOTOR
                MOVX @DPTR,A            ;
                LJMP CHK_P2             ;

 STOP_R_M2:     MOV A,PA_BUFFER         ;
                ANL A,#11110011B        ;
                MOVX @DPTR,A            ; STOP RIGHT MOTOR
                LJMP CHK_P2             ;

    NO_DET:     LCALL STOP

                POP 8BH                 ; POP TL1
                POP 8AH                 ; POP TL0
                POP 01H                 ; POP R1
                POP ACC
                RETI
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
;*************************************************************************
;  PROGRAM:       CHECK TARGET TO BE STILL NEAR
;                 { INTTERUPT (INT1) PROGRAM }
;
; REGISTOR:       A,R0
;  ROUTINE:       DELAY_10,STOP
;*************************************************************************

     MEET:       PUSH ACC
                 PUSH 00H

                 SETB P1.6
                 SETB P3.3                ; PUSH R0

     STAY:       LCALL STOP               ;
                 LCALL DELAY_10           ;
                 LCALL DELAY_10           ; CHECK THE TARGET IS STILL STAY OR NOT
                 MOV A,P3.3               ; IF IT IS DISAPPEAR,PROGRAM WILL RETURN INTERRUPT
                 JNB ACC.3,STAY           ;
                 CLR P1.6                 ;

                 POP 00H
                 POP ACC
                 RETI

;*************************************************************************
;  PROGRAM:       CONTROL STEPPING MORTOR
;
;   OUTPUT:       PORT A
; REGISTOR:       A,R0,DPTR
;  ROUTINE:       DELAY_10
;*************************************************************************

     R_STEP:     ANL PA_BUFFER,#00001111B   ;
                 ORL PA_BUFFER,#R_ST_BUF    ;
                 MOV A,PA_BUFFER            ;
                 MOVX @DPTR,A               ;
                 LCALL DELAY_10             ;
                                            ; DRIVE STEPPING MORTOR RIGHT BY USING 1 CLOCK
                 MOV A,PA_BUFFER            ;
                 CPL ACC.5                  ;
                 MOVX @DPTR,A               ;
                 LCALL DELAY_10             ;
                 RET

     L_STEP:     ANL PA_BUFFER,#00001111B   ;
                 ORL PA_BUFFER,#L_ST_BUF    ;
                 MOV A,PA_BUFFER            ;
                 MOVX @DPTR,A               ;
                 LCALL DELAY_10             ;
                                            ; DRIVE STEPPING MORTOR LEFT BY USING 1 CLOCK
                 MOV A,PA_BUFFER            ;
                 CPL ACC.5                  ;
                 MOVX @DPTR,A               ;
                 LCALL DELAY_10             ;
                 RET
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
;*************************************************************************
;  PROGRAM:      CONTROL MOVMENT OF VEHICLE
;
;   OUTPUT:      PORT A
; REGISTOR:      A,DPTR
;*************************************************************************

  FORWARD:      ANL PA_BUFFER,#11110000B    ;
                ORL PA_BUFFER,#00001010B    ; DRIVE DC MOTORS FOR VEHICLE RUN TO FORWARD
                MOV A,PA_BUFFER             ;
                MOVX @DPTR,A                ;

                RET

   TURN_L:      ANL PA_BUFFER,#11110000B    ;
                ORL PA_BUFFER,#00001001B    ; DRIVE DC MOTORS FOR VEHICLE TURN LEFT
                MOV A,PA_BUFFER             ;
                MOVX @DPTR,A                ;

                RET

   TURN_R:      ANL PA_BUFFER,#11110000B    ;
                ORL PA_BUFFER,#00000110B    ; DRIVE DC MOTORS FOR VEHICLE TURN RIGHT
                MOV A,PA_BUFFER             ;
                MOVX @DPTR,A                ;

                RET

     STOP:      ANL PA_BUFFER,#11110000B    ;
                MOV A,PA_BUFFER             ; STOP VEHICLE
                MOVX @DPTR,A                ;

                RET

;*************************************************************************
;  PROGRAM:      SUB 2 BYTE HEXADESIMAL
;
;    INPUT:      REM_H,REM_L,SUB_H,SUB_L
;   OUTPUT:      ANS_H,_ANS_L
; REGISTOR:      A
;*************************************************************************

   SUB_2_B:     CLR F0                      ; USE F0 AS SIGN FLEG
                CLR CY

                MOV A,REM_H                 ; COMPARE HIGH BYTE OF REMAINDER WITH HIGH BYTE OF SUBBER
                SUBB A,SUB_H                ;
                JC XCH_H_BY

                JZ CHK_L_BY

                LJMP SUB_L_B
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
CHK_L_BY:       MOV A,REM_L             ;
                SUBB A,SUB_L            ; COMPARE LOW BYTE OF REMAINDER WITH LOW BYTE OF SUBBER
                JC XCH_L_BY             ;

                LJMP SUB_L_B

XCH_H_BY:       MOV A,REM_H             ;
                XCH A,SUB_H             ;
                MOV REM_H,A             ;
                                        ; EXCHANGE REMAINDER TO SUBBER
XCH_L_BY:       MOV A,REM_L             ;
                XCH A,SUB_L             ;
                MOV REM_L,A             ;

                SETB F0                 ; SET SIGN FLAG TO BE NEGATIVE

SUB_L_B:        CLR CY                  ;
                MOV A,REM_L             ;
                SUBB A,SUB_L            ;
                MOV ANS_L,A             ;
                                        ; SUB 2 BYTES HEXADESIMAL
SUB_H_B:        MOV A,REM_H             ;
                SUBB A,SUB_H            ;
                MOV ANS_H,A             ;

SUB_2_OK:       RET

;*************************************************************************
;  PROGRAM:     SELECT THE WAY FOR THE VEHICLE
;
;   OUTPUT:     TURN_BUF
; REGISTOR:     A
;  ROUTINE:     R_OR_L
;*************************************************************************

T_LOCK:         MOV REM_H,NOW_D_H       ;
                MOV REM_L,NOW_D_L       ; COMPARE FRONT OF VEHICLE DIRECTION WITH TARGET DIRECTION
                MOV SUB_H,TAR_D_H       ; AND DECIDE TO SELECT DIRECTION TO TURN
                MOV SUB_L,TAR_D_L       ;
                LCALL R_OR_L            ;

                RET

;*************************************************************************
;  PROGRAM:     SELECT RIGHT OR LEFT WAY
;
;    INPUT:     REM_H,REM_L,SUB_H,SUB_L
;   OUTPUT:     TURN_BUF
; REGISTOR:     A
;  ROUTINE:     SUB_2_B
;*************************************************************************
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
R_OR_L:     LCALL SUB_2_B

            JB F0,CHK_MIN

            MOV REM_H,ANS_H          ;
            MOV REM_L,ANS_L          ;
            MOV SUB_H,#00H           ;
            MOV SUB_L,#180D          ;
                                     ; COMPARE ANSWER WITH 180 AND SELECT THE DIRECT TO TURN
            LCALL SUB_2_B            ;
                                     ;
            JB F0,SEL_TR             ;
            LJMP SEL_TL              ;

CHK_MIN:    MOV REM_H,ANS_H          ;
            MOV REM_L,ANS_L          ;
            MOV SUB_H,#00H           ;
            MOV SUB_L,#180D          ;
                                     ; COMPARE ANSWER WITH 180 AND SELECT THE DIRECT TO TURN
            LCALL SUB_2_B            ;
                                     ;
            JNB F0,SEL_TR            ;

SEL_TL:     MOV TURN_BUF,#01H        ; SET TURN_BUF TO BE 1 FOR SELECTING TURN LEFT
            LJMP END_RL

SEL_TR:     MOV TURN_BUF,#00H        ; SET TURN_BUF TO BE 1 FOR SELECTING TURN RIGHT

END_RL:     RET

;*************************************************************************
;  PROGRAM:     READ VECTOR DATA
;
;   OUTPUT:     VT_H_BYTE,VT_H_BYTE
; REGISTOR:     A,R0
;  ROUTINE:     GETDATA
;*************************************************************************

VECTOR:     ORL P1,#10111110B

RES_V2X:    CLR P1.3                 ;
 DELAY1:    LCALL DELAY_10           ; RESET V2X BY TAKE 10ms PULSE
            SETB P1.3                ;

            LCALL DELAY_10           ;
            LCALL DELAY_10           ; DELAY 40ms AFTER RESETING V2X
            LCALL DELAY_10           ; FOR WAITING STEADY STATE OF V2X
            LCALL DELAY_10           ;

            CLR P1.4                 ;
            LCALL DELAY_10           ; TAKE 10ms PULSE TO P/C PIN
            SETB P1.4                ;

CHK_EOC_L:  SETB P1.2                ;
            MOV A,P1                 ; CHECK EOC PIN TO BE LOW
            JNB P1.2,CHK_EOC_H       ;
            SJMP CHK_EOC_L           ;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
CHK_EOC_H:      SETB P1.2                ;
                MOV A,P1                 ; CHECK EOC PIN TO BE HIGH
                JB P1.2,DELAY2           ;
                SJMP CHK_EOC_H           ;

                SETB P1.4                ;*

   DELAY2:      LCALL DELAY_10
                CLR P1.5                 ; CLEAR  SS PIN
                LCALL DELAY_10
                LCALL DELAY_10

                MOV VT_L_BYTE,#00H
                MOV CHK_RND,#00H
                MOV SHF_BUFF,#00H

     READ:      LCALL GETDATA            ; READ DATA  FROM  ACC.7

                MOV R0,SHF_BUFF          ;
                XCH A,R0                 ; CHECK SEQUENCE OF DATA
                JNZ SHIFTR               ;
                XCH A,R0                 ; D0 -  D7
                LJMP STORE               ;

   SHIFTR:      XCH A,R0                 ;
   RSHIFT:      RR A                     ; IF DATA BIT > 1TH,IT WILL BE SHIFTED RIGHT
                DJNZ R0,RSHIFT           ;

    STORE:      ORL VT_L_BYTE,A          ; ASSEMBLE NEW DATA BIT WITH FORMER DATA

                INC SHF_BUFF             ;
                MOV A,#08H               ;
                CLR CY                   ; CHECK 8th BIT DATA HAVE BEEN READ COMPLETELY
                SUBB A,SHF_BUFF          ;
                JNZ READ                 ;

                MOV A,#01H               ;
                CLR CY                   ;
                SUBB A,CHK_RND           ; CHECK 2nd BYTE DATA HAVE BEEN READ COMPLETELY
                JZ END_V2X               ;

                INC CHK_RND              ;
                MOV SHF_BUFF,#00H        ;
                MOV VT_H_BYTE,VT_L_BYTE  ; KEEP HIGH BYTE OF V2X DATA
                MOV VT_L_BYTE,#00H       ;
                LJMP READ                ;

END_V2X:
RET
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
;*************************************************************************
;  PROGRAM:      READ SERIAL DATA
;   OUTPUT:      ACC.7
; REGISTOR:      A
;*************************************************************************

   GETDATA:      CLR  P1.1                ;
                 SETB P1.7                ;
                 CALL DELAY20U            ; TAKE 1 PULSE AND GET 1 BIT DATA FROM P1.7
                 MOV A,P1                 ;
                 ANL A,#80H               ;
                 SETB P1.1                ;

                 RET

;*************************************************************************
;  PROGRAM:      FORWARD RUNNING WITH CONTROL SPEED
;
;   OUTPUT:      PORT A
; REGISTOR:      A,R0,TL0,TL1
;  ROUTINE:      GO_FWD
;*************************************************************************

   RUN_FWD:      MOV TL1,#00H
                 MOV TL0,#00H

                 MOV FWD_BUF,#80D         ; SET PULSE WIDTH FOR CONTROL VEHICLE RUN TO FORWARD
                 MOV TR_BUF,#00H          ; SET PULSE WIDTH FOR CONTROL VEHICLE TURN RIGHT
                 MOV TL_BUF,#00H          ; SET PULSE WIDTH FOR CONTROL VEHICLE TURN LEFT

        GO:      MOV FWD_TMP,FWD_BUF
                 MOV TR_TMP,TR_BUF
                 MOV TL_TMP,TL_BUF

                 LCALL SPEED              ; TAKE PULSES TO RIGHT AND LEFT DC MOTOR

                 CLR  CY                  ;
                 MOV A,TL0                ; COMPARE ROTATION OF RIGHT AND LEFT DC MOTOR
                 SUBB A,TL1               ;

                 JZ END_FWD

                 JC DOWN_R

    DOWN_L:      MOV A,FWD_BUF            ;
                 JZ  GO                   ;
                 INC TL_BUF               ; INCRESE TL_BUFF FOR DECRESE PULSE WIDTH OF LEFT DC MOTOR
                 DEC FWD_BUF              ;
                 LJMP GO                  ;

    DOWN_R:      MOV A,FWD_BUF            ;
                 JZ GO                    ;
                 INC TR_BUF               ; INCRESE TR_BUFF FOR DECRESE PULSE WIDTH OF RIGHT DC MOTOR
                 DEC FWD_BUF              ;
                 LJMP GO                  ;

   END_FWD:      RET
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
;*****************************************************************************
;  PROGRAM:      CONTROL VEHICLE SPEED
;
;    INPUT:      FWD_TMP,TL_TMP,TR_TMP
;   OUTPUT:      PORT A
; REGISTOR:      A,R0
;  ROUTINE:      DELAY200U
;*****************************************************************************

    SPEED:       MOV A,FWD_TMP              ;
                 JZ SLOW_R                  ;
                 LCALL FORWARD              ; TAKE PULSE THAT HAVE PULSE WIDTH EQUAL TO DATA IN FWD_TMP
   LOOP_F:       LCALL DEL_200U             ; MULTIPLY BY DELAY TIME FOR VEHICLE RUN TO FORWARD
                 DJNZ FWD_TMP,LOOP_F        ;

   SLOW_R:       MOV A,TR_TMP               ;
                 JZ SLOW_L                  ; TAKE PULSE THAT HAVE PULSE WIDTH EQUAL TO DATA IN TR_TMP
                 ANL PA_BUFFER,#11110011B   ; MULTIPLY BY DELAY TIME FOR STOPPING RIGHT DC MOTOR
                 MOV A,PA_BUFFER            ;
                 MOVX @DPTR,A               ;

  LOOP_SR:       LCALL DEL_200U
                 DJNZ TR_TMP,LOOP_SR

   SLOW_L:       MOV A,TL_TMP               ;
                 JZ STOP_FWD                ; TAKE PULSE THAT HAVE PULSE WIDTH EQUAL TO DATA IN TL_TMP
                 ANL PA_BUFFER,#11111100B   ; MULTIPLY BY DELAY TIME FOR STOPPING LEFT DC MOTOR
                 MOV A,PA_BUFFER            ;
                 MOVX @DPTR,A               ;

  LOOP_SL:       LCALL DEL_200U
                 DJNZ TL_TMP,LOOP_SL

 STOP_FWD:       LCALL STOP
                 LCALL DEL_200U

                 RET

;*****************************************************************************
;  PROGRAM:      DELAY TIME
;
; REGISTOR:      A,R0
;*****************************************************************************

  DELAY20U:      MOV A,#05H                 ;
   LOOP20U:      DEC A                      ; DELAY 20us
                 ANL A,#0FFH                ;
                 JNZ LOOP20U                ;

                 RET
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
DEL_200U:       MOV R0,#100D            ;
LOOP200U:       ANL A,#0FFH             ; DELAY 200us
                DJNZ R0,LOOP200U        ;

                RET

DELAY_10:       MOV R0,#10D             ;
  D1LOOP:       MOV A,#230D             ;
  D2LOOP:       DEC A                   ; DELAY 10ms
                ANL A,#0FFH             ;
                JNZ D2LOOP              ;
                DJNZ R0,D1LOOP          ;

                RET
;*******************************************************************************
;                               THE END
;*******************************************************************************
                END
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 5
สรุปผล

ในการประดิษฐ์รถเอนกประสงค์ (MPV) รถที่สร้างขึ้นใช้อลูมิเนียมเป็นโครงสร้าง มีลักษณะเป็นรูปคล้ายวงรี ขนาด 57*63 เซนติเมตร ประกอบด้วยล้อ 4 ล้อ โดยล้อหน้าและหลังของแต่ละด้านจะคล้องถึงกันด้วยโซ่ผ่านเฟืองที่ติดอยู่กับดีซีมอเตอร์ ที่ล้อหลังของล้อทั้งสองจะติดวงจรเช็คความเร็วล้อ ด้านบนมีจานรูปวงกลมขนาดเส้นผ่านศูนย์กลาง 30 เซนติเมตร ซึ่งมีฐานติดกับสเตปปิ้งมอเตอร์ จานนี้ใช้วางวงจรตรวจจับเป้าหมายและเข็มทิศอิเล็กทรอนิกส์ ด้านหน้าสุดของรถจะติดตั้งวงจรตรวจจับสิ่งกีดขวางและถัดมาเป็นวงจรขับดีซีมอเตอร์ สำหรับการทำงานเริ่มต้นด้วย การสแกนหาเป้าหมายซึ่งเป็นแสงโดยให้จานรูปวงกลมด้านบนหมุนรอบตัวเองทางด้านซ้ายและขวาสลับกัน ขณะเดียวกันวงจรตรวจจับเป้าหมายจะเริ่มตรวจจับเป้าหมายและเข็มทิศอิเล็กทรอนิกส์จับค่าองศาของทิศที่จานหมุนไป จนกระทั่งพบเป้าหมายเข็มทิศจะจดจำค่าองศาของเป้าหมายเอาไว้ หลังจากนั้นรถจะหมุนไปตามทิศทางที่ถูกจำไว้แล้วจะเคลื่อนตรงไปยังทิศนั้น เป็นการเคลื่อนไปยังเป้าหมาย ในขณะที่รถเคลื่อนไปนั้นถ้ามีการตรวจพบสิ่งกีดขวางที่อยู่ด้านหน้า รถจะทำการหลบหลีกสิ่งกีดขวางโดยการบังคับให้หมุนไปทางซ้ายแล้วหยุด หลังจากนั้นจะสแกนหาเป้าหมายในลักษณะเดิมอีกครั้ง แต่หากไม่พบสิ่งกีดขวางใดๆ รถก็จะเคลื่อนที่ไปตามทิศที่มีเป้าหมายอยู่ เมื่อเข้าไปใกล้ระยะหนึ่งรถจะหยุดเป็นการสิ้นสุดการทำงาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บรรณานุกรม

1. พิพัฒน์ เลาหสงคราม "ไมโครคอนโทรลเลอร์ MCS-48 MCS-51" ภาควิชาเทคโนโลยีการวัดคุมอุตสาหกรรม คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง ,พฤษภาคม 2537
2. สุนทร วิทูสุรพจน์ "การใช้งานไมโครคอนโทรลเลอร์ 8051"
3. "มอเตอร์กระแสตรง" วารสารเซมิคอนดัคเตอร์เล่มที่ 104
4. "อุลตร้าโซนิคทรานสดิวเซอร์" วารสารรวมเล่มเซมิคอนดัคเตอร์เล่มที่ 4 ประจำเดือนเมษายน 2534
5. "Vector 2X Compass Module Application Note" Precision Navigation, Inc. , Version 3 January 1996
6. "National Semiconductor Corporation"

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ก.

ชุดคำสั่งของไมโครคอนโทรลเลอร์ตระกูล MCS 51

คำสั่งประมวลผลทางคณิตศาสตร์ (Arithematic Operations)

รูปแบบคำสั่ง		ความหมาย	จำนวนไบต์	จำนวนแมชชีนไซเคิล
ADD	A,Rn	บวกค่า Rn กับ A	1	1
ADD	A,direct	บวกค่าในหน่วยความจำ direct กับ A	2	1
ADD	A,@Ri	บวกค่าในหน่วยความจำที่เก็บอยู่ใน Ri กับ A	1	1
ADD	A,#data	บวกค่าคงที่ 8 บิตกับ A	2	1
ADDC	A,Rn	บวกค่า Rn กับ A พร้อมแฟล็ก Carry	1	1
ADDC	A,direct	บวกค่าในหน่วยความจำ direct กับ A พร้อมแฟล็ก Carry	2	1
ADDC	A,@Ri	บวกค่าในหน่วยความจำที่เก็บอยู่ใน Ri กับ A พร้อมแฟล็ก Carry	1	1
ADDC	A,#data	บวกค่าคงที่ 8 บิตกับ A พร้อมแฟล็ก Carry	2	1
SUBB	A,Rn	ลบค่า Rn กับ A พร้อมแฟลก Borrow	1	1
SUBB	A,direct	ลบค่าในหน่วยความจำ direct กับ A พร้อมแฟล็ก Borrow	2	1
SUBB	A,@Ri	ลบค่าในหน่วยความจำที่เก็บอยู่ใน Ri กับ A พร้อมแฟล็ก Borrow	1	1
SUBB	A,#data	ลบค่าคงที่ 8 บิตกับ A พร้อมแฟล็ก Borrow	2	1
INC	A	เพิ่มค่าใน A	1	1
INC	Rn	เพิ่มค่าใน Rn	1	1
INC	direct	เพิ่มค่าในหน่วยความจำ direct	2	1
INC	@Ri	เพิ่มค่าในหน่วยความจำที่เก็บอยู่ใน Ri	1	1
DEC	A	ลดค่าใน A	1	1
DEC	Rn	ลดค่าใน Rn	1	1
DEC	direct	ลดค่าในหน่วยความจำ direct	2	1
DEC	@Ri	ลดค่าในหน่วยความจำที่เก็บอยู่ใน Ri	1	1
INC	DPTR	เพิ่มค่าใน DPTR	1	2
MUL	AB	คูณ A กับ B แล้วเก็บค่าใน A	1	4
DIV	AB	หาร A ด้วย B แล้วเก็บค่าใน A	1	4
DA	A	ทำ decimal adjust ค่าใน A	1	1

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ก.

ชุดคำสั่งของไมโครคอนโทรลเลอร์ตระกูล MCS 51

คำสั่งทางตรรก (Logical Operations)

รูปแบบคำสั่ง		ความหมาย	จำนวนไบต์	จำนวนแมชชีนไซเคิล
ANL	A,Rn	AND ค่าใน Rn กับ A	1	1
ANL	A,direct	AND ค่าในหน่วยความจำ direct กับ A	2	1
ANL	A,@Ri	AND ค่าในหน่วยความจำที่เก็บใน Rn กับ A	1	1
ANL	A,#data	AND ค่าคงที่ 8 บิตกับ A	2	
ANL	direct,A	AND ค่า A กับหน่วยความจำ direct	2	1
ANL	direct,#data	AND ค่าคงที่ 8 บิตกับหน่วยความจำ direct	3	2
ORL	A,Rn	OR ค่าใน Rn กับ A	1	1
ORL	A,direct	OR ค่าในหน่วยความจำ direct กับ A	2	1
ORL	A,@Ri	OR ค่าในหน่วยความจำที่เก็บอยู่ใน Rn กับ A	1	1
ORL	A,#data	OR ค่าคงที่ 8 บิตกับ A	2	1
ORL	direct,A	OR ค่า A กับหน่วยความจำ direct	2	1
ORL	direct,#data	OR ค่าคงที่ 8 บิตกับหน่วยความจำ direct	3	2
XRL	A,Rn	EX-OR ค่าใน Rn กับ A	1	
XRL	A,direct	EX-OR ค่าในหน่วยความจำ direct กับ A	2	1
XRL	A,@Ri	EX-OR ค่าในหน่วยความจำที่เก็บอยู่ใน Ri กับ A	1	1
XRL	A,#data	EX-OR ค่าคงที่ 8 บิตกับ A	2	1
XRL	direct,A	EX-OR ค่า A กับหน่วยความจำ direct	2	1
XRL	direct,#data	EX-OR ค่าคงที่ 8 บิตกับหน่วยความจำ direct	3	2
CLR	A	ทำค่าใน A ให้เป็นศูนย์	1	1
CPL	A	กลับค่าบิตใน A เป็นตรงข้ามทุกบิต	1	1
RL	A	หมุนบิตใน A ไปทางซ้าย 1 บิตและบิต 0 มีค่า 0	1	1
RLC	A	หมุนบิตใน A ไปทางซ้าย 1 บิต และบิต 0 เป็นค่าบิตที่อยู่ในแฟล็ก Carry	1	1
RR	A	หมุนบิตใน A ไปทางขวา 1 บิต และบิต 0 เป็นค่าจากบิต 7	1	1
RRC	A	หมุนบิตใน A ไปทางขวา 1 บิต และค่าจาก บิต 0 นำไปเก็บในแฟล็ก Carry และบิตที่อยู่ในแฟล็ก Carry เดิมจะย้ายมาเก็บในบิต 7	1	1
SWAP	A	สลับค่าสี่บิตบนกับสี่บิตล่างภายใน A	1	1

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ก.

ชุดคำสั่งของไมโครคอนโทรลเลอร์ตระกูล MCS 51

คำสั่งเคลื่อนย้ายข้อมูล (Data Transfer)

รูปแบบคำสั่ง		ความหมาย	จำนวนไบต์	จำนวนแมชชีนไซเคิล
MOV	A,Rn	ย้ายข้อมูลจาก Rn ไป A	1	1
MOV	A,direct	ย้ายข้อมูลจากหน่วยความจำ direct ไป A	2	1
MOV	A,@Ri	ย้ายข้อมูลจากหน่วยความจำที่เก็บอยู่ใน Ri ไป A	1	1
MOV	A,#data	ย้ายค่าคงที่ 8 บิตไปเก็บที่ A	2	1
MOV	Rn,A	ย้ายข้อมูลจาก A ไป Rn	1	1
MOV	Rn,direct	ย้ายข้อมูลจากหน่วยความจำ direct ไป Rn	2	2
MOV	direct,A	ย้ายข้อมูลจาก A ไปยังหน่วยความจำ direct	2	1
MOV	direct,Rn	ย้ายข้อมูลจาก Rn ไปยังหน่วยความจำ direct	2	2
MOV	direct,direct	ย้ายข้อมูลระหว่างหน่วยความจำภายใน	3	2
MOV	direct,@Ri	ย้ายข้อมูลจากหน่วยความจำที่เก็บอยู่ใน Ri ไปยังหน่วยความจำ direct	2	2
MOV	direct,#data	ย้ายค่าคงที่ 8 บิตไปยังหน่วยความจำ direct	3	2
MOV	@Ri,A	ย้ายข้อมูลใน A ไปยังหน่วยความจำที่เก็บอยู่ใน Ri	1	1
MOV	@Ri,direct	ย้ายข้อมูลจากหน่วยความจำ direct ไปยังหน่วยความจำที่เก็บอยู่ใน Ri	2	2
MOV	@Ri,#data	ย้ายค่าคงที่ 8 บิตไปยังหน่วยความจำที่เก็บอยู่ใน Ri	2	1
MOV	DPTR,#data16	ย้ายค่าคงที่ 16 บิตไปยัง DPTR	3	2
MOVC	A,@A+DPTR	ย้ายข้อมูลจากหน่วยความจำข้อมูลที่สัมพัทธ์กับ DPTR ไปยัง A	1	2
MOVC	A,@A+PC	ย้ายข้อมูลจากหน่วยความจำข้อมูลที่สัมพัทธ์กับ PC ไปยัง A	1	2
MOVX	A,@Ri	ย้ายข้อมูลจากหน่วยความจำที่เก็บอยู่ใน Ri ไปยัง A	1	2
MOVX	A,@DPTR	ย้ายข้อมูลจากหน่วยความจำที่เก็บอยู่ใน DPTR ไปยัง A	1	2
MOVX	@Ri,A	ย้ายข้อมูลที่เก็บอยู่ใน A ไปยังหน่วยความจำที่เก็บอยู่ใน Ri	1	2
MOVX	@DPTR,A	ย้ายข้อมูลที่เก็บอยู่ใน A ไปยังหน่วยความจำที่เก็บอยู่ใน DPTR	1	2

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ก.

ชุดคำสั่งของไมโครคอนโทรลเลอร์ตระกูล MCS 51

คำสั่งเคลื่อนย้ายข้อมูล (Data Transfer) (ต่อ)

PUSH	direct	ย้ายข้อมูลหน่วยความจำ direct ไปเก็บยัง stack	2	2
POP	direct	ย้ายข้อมูลจาก stack ไปยังหน่วยความจำ direct	2	2
XCH	A,Rn	แลกเปลี่ยนข้อมูลระหว่าง A กับ Rn	1	1
XCH	A,direct	แลกเปลี่ยนข้อมูลระหว่างหน่วยความจำ direct กับ A	2	1
XCH	A,@Ri	แลกเปลี่ยนข้อมูลระหว่างหน่วยความจำที่เก็บอยู่ใน Ri กับ A	1	1
XCHD	A,@Ri	แลกเปลี่ยนข้อมูลสี่บิตล่างจากหน่วยความจำที่เก็บอยู่ใน Ri กับ A	1	1

คำสั่งจัดการข้อมูลแบบบิต (Boolean Variable Manipulation)

รูปแบบคำสั่ง		ความหมาย	จำนวนไบต์	จำนวนแมชชีนไซเคิล
CLR	C	ทำค่าแฟล็ก Carry ให้เป็น 0	1	1
CLR	bit	ทำค่า bit ให้เป็น 0	2	1
SETB	C	ทำค่าแฟล็กCarry ให้เป็น 1	1	1
SETB	bit	ทำค่า bit ให้เป็น 1	2	1
CPL	C	กลับค่าแฟล็ก Carry ให้เป็นตรงข้าม	1	1
CPL	bit	กลับค่า bit ให้เป็นตรงข้าม	2	1
ANL	C,bit	AND ค่า bit กับแฟล็ก Carry	2	2
ANL	C,/bit	AND ค่าตรงข้ามของ bit กับแฟล็ก Carry	2	2
ORL	C,bit	ORL ค่า bit กับแฟล็ก Carry	2	2
ORL	C,/bit	ORL ค่าตรงข้ามของ bit กับแฟล็ก Carry	2	2
MOV	C,bit	ย้ายค่า bit มายังแฟล็ก Carry	2	1
MOV	bit,C	ย้ายค่าแฟล็ก Carry มายัง bit	2	2
JC	rel	กระโดด ถ้าค่าแฟล็กCarry เป็น 1	2	2
JNC	rel	กระโดด ถ้าค่าแฟล็ก Carry เป็น 0	2	2
JB	bit,rel	กระโดด ถ้าค่า bit เป็น 1	3	2
JNB	bit,rel	กระโดด ถ้าค่า bit เป็น 0	3	2
JBC	bit,rel	กระโดด ถ้าค่า bit เป็น 1 และเปลี่ยนค่า bit เป็น 0	3	2

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ก.

ชุดคำสั่งของไมโครคอนโทรลเลอร์ตระกูล MCS 51

คำสั่งควบคุมการทำงานโปรแกรม (Program and Machine Control)

รูปแบบคำสั่ง		ความหมาย	จำนวนไบต์	จำนวนสัญญาณนาฬิกาที่ใช้ในการทำงาน
CLR	C	ทำค่าแฟล็ก Carry ให้เป็น 0	1	1
ACALL	addr 11	ไปทำโปรแกรมย่อยจากค่าแอดเดรส 11 บิต	2	2
LCALL	addr 16	ไปทำโปรแกรมย่อยจากค่าแอดเดรส 16 บิต	3	2
RET		คำสั่งสิ้นสุดการทำงานโปรแกรมย่อย	1	2
RETI		คำสั่งสิ้นสุดการทำงานโปรแกรมย่อย อินเตอร์รัปต์	1	2
AJMP	addr 11	กระโดดไปยังตำแหน่งจากค่าแอดเดรส 11 บิต	2	2
LJMP	addr 16	กระโดดไปยังตำแหน่งจากค่าแอดเดรส 16 บิต	3	2
SJMP	rel	กระโดดไปยังตำแหน่งที่สัมพัทธ์กับตำแหน่งปัจจุบัน	2	2
JMP	@A+DPTR	กระโดดไปยังตำแหน่งที่สัมพัทธ์กับ DPTR	1	2
JZ	rel	กระโดดไปยังตำแหน่งที่สัมพัทธ์กับตำแหน่งปัจจุบัน ถ้าหากค่า A เป็นค่า 0	2	2
JNZ	rel	กระโดดไปยังตำแหน่งที่สัมพัทธ์กับตำแหน่งปัจจุบัน ถ้าหากค่า A เป็นค่า 1	2	2
CJNE	A,direct,rel	เปรียบเทียบ A กับหน่วยความจำ direct และกระโดดไปยังตำแหน่งที่สัมพัทธ์กับตำแหน่งปัจจุบัน ถ้าค่าไม่เท่ากัน	3	2
CJNE	A,#data,rel	เปรียบเทียบ A กับค่าคงที่ และกระโดดไปยังตำแหน่งที่สัมพัทธ์กับตำแหน่งปัจจุบัน ถ้าค่าไม่เท่ากัน	3	2
CJNE	Rn,#data,rel	เปรียบเทียบ Rn กับค่าคงที่ และกระโดดไปยังตำแหน่งที่สัมพัทธ์กับตำแหน่งปัจจุบัน ถ้าค่าไม่เท่ากัน	3	2
CJNE	@Ri,#data,rel	เปรียบเทียบค่าในหน่วยความจำที่เก็บใน Ri กับค่าคงที่และกระโดดไปยังตำแหน่งที่สัมพัทธ์กับตำแหน่งปัจจุบัน ถ้าค่าไม่เท่ากัน	3	2
DJNZ	Rn,rel	ลดค่าใน Rn และกระโดดไปยังตำแหน่งที่สัมพัทธ์กับตำแหน่งปัจจุบัน ถ้าค่าไม่เป็น 0	2	2
DJNZ	direct,rel	ลดค่าในหน่วยความจำ direct และกระโดดไปยังตำแหน่งที่สัมพัทธ์กับตำแหน่งปัจจุบัน ถ้าค่าไม่เป็น 0	3	2
NOP		ไม่มีการทำงานใดๆ เกิดขึ้น	1	1

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ข.

ข้อมูลรายละเอียดของไมโครคอนโทรลเลอร์ 8031

Signetics

SCN8031AH/SCN8051AH
Single-Chip 8-Bit Microcontroller

Product Specification

Microprocessor Division

DESCRIPTION

The Signetics SCN8031AH/SCN8051AH is a high-performance microcontroller fabricated using the Signetics highly reliable +5V, depletion-load, N-channel, silicon-gate, N500 MOS process technology. It provides the hardware features, architectural enhancements and instructions that are necessary to make it a powerful and cost-effective controller for applications requiring up to 64K bytes of program memory and/or up to 64K bytes of data storage.

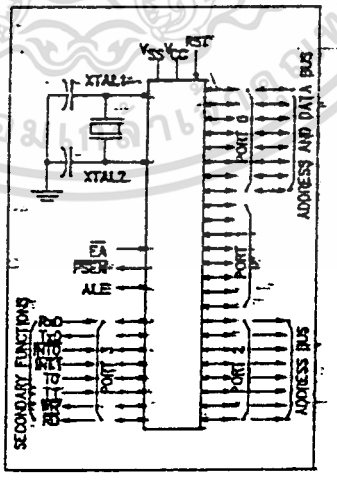
The SCN8051AH contains a 4K X 8 read-only program memory, a 128 X 8 read/write data memory, 32 I/O lines, two 16-bit timer/counters, a five-source two-priority-level nested interrupt structure, a serial I/O port for either multiprocessor communications, I/O expansion, or full-duplex UART, and on-chip oscillator and clock circuits. The SCN-8031AH is identical, except that it lacks the program memory. For systems that require extra capability, the SCN8051AH can be expanded using standard TTL compatible memories and byte oriented peripheral controllers.

The SCN8051AH microcontroller, like its SCN8048 predecessor, is efficient both as a controller and as an arithmetic processor. It has extensive facilities for binary and BCD arithmetic and excels in bit-handling capabilities. Efficient use of program memory results from an instruction set consisting of 44% one-byte, 41% two-byte, and 15% three-byte instructions. With a 12MHz crystal, 58% of the instructions execute in 1µs, 40% in 2µs and multiply and divide require only 4µs.

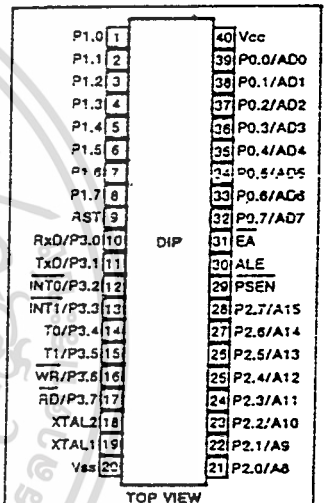
FEATURES

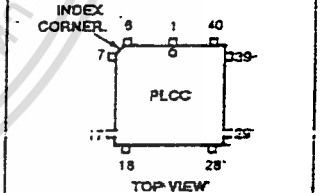
- Reduced supply current
- 4K X 8 ROM (SCN8051AH)
- 128 X 8 RAM
- Four 8-bit ports, 32 I/O lines
- Two 16-bit timer/event counters
- High-performance full-duplex serial channel
- External memory expandable to 128K
- Boolean processor
- Industry standard 8051 architecture:
  - Non-paged jumps
  - Direct addressing
  - Four 8-register banks
  - Stack depth up to 128-bytes
  - Multiply, divide, subtract, compare
- Most instructions execute in 1µs
- 4µs multiply and divide

LOGIC SYMBOL



PIN CONFIGURATION





Pin	Function	Pin	Function
1	NC	23	NC
2	P1.0	24	P2.0/A8
3	P1.1	25	P2.1/A9
4	P1.2	26	P2.2/A10
5	P1.3	27	P2.3/A11
6	P1.4	28	P2.4/A12
7	P1.5	29	P2.5/A13
8	P1.6	30	P2.6/A14
9	P1.7	31	P2.7/A15
10	RST	32	PSEN
11	RxD/P3.0	33	ALE
12	NC	34	NC
13	TxD/P3.1	35	EA
14	INT0/P3.2	36	P0.7/AD7
15	INT1/P3.3	37	P0.6/AD6
16	T0/P3.4	38	P0.5/AD5
17	T1/P3.5	39	P0.4/AD4
18	WR/P3.6	40	P0.3/AD3
19	RD/P3.7	41	P0.2/AD2
20	XTAL2	42	P0.1/AD1
21	XTAL1	43	P0.0/AD0
22	Vss	44	Vcc

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ข.

ข้อมูลรายละเอียดของไมโครคอนโทรลเลอร์ 8031

Single-Chip 8-Bit Microcontroller

SCN8031AH/SCN8051AH

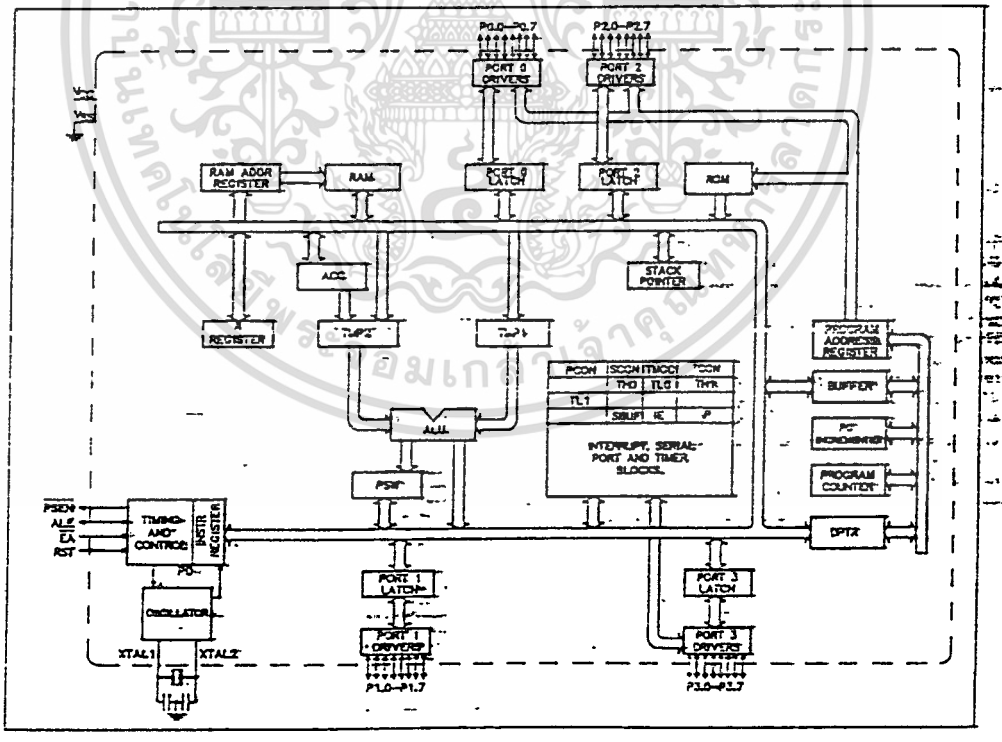
ORDERING INFORMATION

SCN80 □□H□□□□□ (CPxxxx)

- ROM/RAM (bytes): 31 = Ext/128; 51 = 4K/128
- Power Consumption: H = Reduced Power AH
- Custom ROM Pattern No. Applies to masked ROM versions only. Number will be assigned by Signetics. Contact Signetics sales office for ROM pattern submission requirements.
- Pins: 40 = 40-pin; 44 = 44-pin
- Package: A = Plastic LCC; I = Ceramic DIP; N = Plastic DIP
- Speed: C = 3.5 to 12MHz; F = 3.5 to 15MHz
- Operating Temperature Range: A = −40°C to +85°C; C = 0°C to +70°C

PART NUMBER SELECTION			
ROMless	ROM	Temperature and Package	Frequency
SCN8031HACN40	SCN8051HACN40	−40 to +85°C plastic DIP	3.5 to 12MHz
SCN8031HCCN40	SCN8051HCCN40	0 to +70°C plastic DIP	3.5 to 12MHz
SCN8031HCFN40	SCN8051HCFN40	0 to +70°C plastic DIP	3.5 to 15MHz
SCN8031HAFN40	SCN8051HAFN40	−40 to +85°C plastic DIP	3.5 to 15MHz
SCN8031HCCA44	SCN8051HCCA44	0 to +70°C plastic PLCC	3.5 to 12MHz
SCN8031HACA44	SCN8051HACA44	−40 to +85°C plastic PLCC	3.5 to 12MHz
SCN8031HCFA44	SCN8051HCFA44	0 to +70°C plastic PLCC	3.5 to 15MHz
SCN8031HAFA44	SCN8051HAFA44	−40 to +85°C plastic PLCC	3.5 to 15MHz

BLOCK DIAGRAM



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ข.

ข้อมูลรายละเอียดของไมโครคอนโทรลเลอร์ 8031

Signetics Microprocessor Products — Product Specification

Single-Chip 8-Bit Microcontroller — SCN8031AH/SCN8051AH

PIN DESCRIPTION

MNEMONIC	PIN NO. DIP	PIN NO. LCC	TYPE	NAME AND FUNCTION
V_{SS}	20	22	I	Ground: 0V reference.
V_{CC}	40	44	I	Power Supply: This is the power supply voltage for normal, idle, and power-down operation.
P0.0–P0.7	39–32	43–36	I/O	Port 0: Port 0 is an open-drain, bidirectional I/O port. Port 0 pins that have 1s written to them float and can be used as high-impedance inputs. Port 0 is also the multiplexed low-order address and data bus during accesses to external program and data memory. In this application, it uses strong internal pullups when emitting 1s.
P1.0–P1.7	1–8	2–9	I/O	Port 1: Port 1 is an 8-bit bidirectional I/O port with internal pull-ups. Port 1 pins that have 1s written to them are pulled high by the internal pullups and can be used as inputs. As inputs, port 1 pins that are externally pulled low will source current because of the internal pull-ups. (See DC Electrical Characteristics: I_{IL}).
P2.0–P2.7	21–28	24–31	I/O	Port 2: Port 2 is an 8-bit bidirectional I/O port with internal pull-ups. Port 2 pins that have 1s written to them are pulled high by the internal pullups and can be used as inputs. As inputs, port 2 pins that are externally being pulled low will source current because of the internal pullups. (See DC Electrical Characteristics: I_{IL}). Port 2 emits the high-order address byte during fetches from external program memory and during accesses to external data memory that use 16-bit addresses (MOVX @DPTR). In this application, it uses strong internal pull-ups when emitting 1s. During accesses to external data memory that use 8-bit addresses (MOVX @Ri), port 2 emits the contents of the P2 special function register.
P3.0–P3.7	10–17	11, 13–19	I/O	Port 3: Port 3 is an 8-bit bidirectional I/O port with internal pullups. Port 3 pins that have 1s written to them are pulled high by the internal pullups and can be used as inputs. As inputs, port 3 pins that are externally being pulled low will source current because of the pull-ups. (See DC Electrical Characteristics: I_{IL}). Port 3 is also used for the special features listed below:
	10	11	I	RxD (P3.0): Serial input port
	11	13	O	TxD (P3.1): Serial output port
	12	14	I	INT0 (P3.2): External interrupt
	13	15	I	INT1 (P3.3): External interrupt
	14	16	I	T0 (P3.4): Timer 0 external input
	15	17	I	T1 (P3.5): Timer 1 external input
	16	18	O	WR (P3.6): External data memory write strobe
	17	19	O	RD (P3.7): External data memory read strobe
RST	9	10	I	Reset: A high on this pin for two machine cycles while the oscillator is running, resets the device. An internal diffused resistor to V_{SS} permits a power-on reset using only an external capacitor to V_{CC}.
ALE	30	33	I/O	Address Latch Enable: Output pulse for latching the low byte of the address during an access to external memory. In normal operation, ALE is emitted at a constant rate of 1/6 the oscillator frequency, and can be used for external timing or clocking. Note that one ALE pulse is skipped during each access to external data memory.
$\overline{PSEN}$	29	32	O	Program Store Enable: The read strobe to external program memory. When the device is executing code from external program memory, PSEN is activated twice each machine cycle except that two PSEN activations are skipped during each access to external data memory. PSEN is not activated during fetches from internal program memory.
$\overline{EA}$	31	35	I	External Access Enable: EA must be externally held low to enable the device to fetch code from external program memory locations 0000H and 0FFFH. If EA is held high, the device executes from internal program memory unless the program counter contains an address greater than 0FFFH.
XTAL1	19	21	I	Crystal 1: Input to the inverting oscillator amplifier.
XTAL2	18	20	O	Crystal 2: Output from the inverting oscillator amplifier and input to the internal clock generator circuits.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ข.

ข้อมูลรายละเอียดของไมโครคอนโทรลเลอร์ 8031

Single-Chip 8-Bit Microcontroller — SCN8031AH/SCN8051AH

AC ELECTRICAL CHARACTERISTICS T_A = 0°C to +70°C or −40°C to +85°C, V_{CC} = 5V ±20%, V_{SS} = 0V[1, 2]

SYMBOL	FIGURE	PARAMETER	12MHz CLOCK Min	12MHz CLOCK Max	VARIABLE CLOCK Min	VARIABLE CLOCK Max	UNIT
Program Memory							
$1/t_{CLCL}$	1	Oscillator frequency: Speed Versions SCN8051 C SCN8051 F			3.5 3.5	12 15	MHz MHz
t_{LHLL}	1	ALE pulse width	127		$2t_{CLCL}-40$		ns
t_{AVLL}	1	Address valid to ALE low	43		$t_{CLCL}-40$		ns
t_{LLAX}	1	Address hold after ALE low	48		$t_{CLCL}-35$		ns
t_{LLIV}	1	ALE low to valid instruction in		233		$4t_{CLCL}-100$	ns
t_{LLPL}	1	ALE low to $\overline{PSEN}$ low	58		$t_{CLCL}-25$		ns
t_{PLPH}	1	$\overline{PSEN}$ pulse width	215		$3t_{CLCL}-35$		ns
t_{PLIV}	1	$\overline{PSEN}$ low to valid instruction in		125		$3t_{CLCL}-125$	ns
t_{PXIX}	1	Input instruction hold after $\overline{PSEN}$	0		0		ns
t_{PXIZ}	1	Input instruction float after $\overline{PSEN}$		63		$t_{CLCL}-20$	ns
t_{AVIV}	1	Address to valid instruction in		302		$5t_{CLCL}-115$	ns
t_{PLAZ}	1	$\overline{PSEN}$ low to address float		30		30	ns
t_{PXAV}	1	$\overline{PSEN}$ to address valid	75		$t_{CLCL}-8$		ns
Data Memory							
t_{RLRH}	2, 3	$\overline{RD}$ pulse width	400		$6t_{CLCL}-100$		ns
t_{WLWH}	2, 3	$\overline{WR}$ pulse width	400		$6t_{CLCL}-100$		ns
t_{RLDV}	2, 3	$\overline{RD}$ low to valid data in		252		$5t_{CLCL}-165$	ns
t_{RHDX}	2, 3	Data hold after $\overline{RD}$	0		0		ns
t_{RHDZ}	2, 3	Data float after $\overline{RD}$		97		$2t_{CLCL}-70$	ns
t_{LLDV}	2, 3	ALE low to valid data in		517		$8t_{CLCL}-150$	ns
t_{AVDV}	2, 3	Address to valid data in		585		$9t_{CLCL}-165$	ns
t_{LLWL}	2, 3	ALE low to $\overline{RD}$ or $\overline{WR}$ low	200	300	$3t_{CLCL}-50$	$3t_{CLCL}+50$	ns
t_{AVWL}	2, 3	Address valid to $\overline{WR}$ low or $\overline{RD}$ low	203		$4t_{CLCL}-130$		ns
t_{QVWX}	2, 3	Data valid to $\overline{WR}$ transition	23		$t_{CLCL}-60$		ns
t_{QVWH}	2, 3	Data valid to $\overline{WR}$ high	433		$7t_{CLCL}-150$		ns
t_{WHQX}	2, 3	Data hold after $\overline{WR}$	33		$t_{CLCL}-8$		ns
t_{RLAZ}	2, 3	$\overline{RD}$ low to address float		20		20	ns
t_{WHLH}	2, 3	$\overline{RD}$ or $\overline{WR}$ high to ALE high	43	123	$t_{CLCL}-40$	$t_{CLCL}+40$	ns
External Clock							
t_{CHCX}	5	High time	20		20		ns
t_{CLCX}	5	Low time	20		20		ns
t_{CLCH}	5	Rise time		20		20	ns
t_{CHCL}	5	Fall time		20		20	ns
Shift Register							
t_{XLXL}	4	Serial port clock cycle time	1.0		$12t_{CLCL}$		µs
t_{QVXH}	4	Output data setup to clock rising edge	700		$10t_{CLCL}-133$		ns
t_{XHQX}	4	Output data hold after clock rising edge	50		$2t_{CLCL}-117$		ns
t_{XHDX}	4	Input data hold after clock rising edge	0		0		ns
t_{XHDV}	4	Clock rising edge to input data valid		700		$10t_{CLCL}-133$	ns

NOTES:
1. Parameters are valid over operating temperature range unless otherwise specified.
2. Load capacitance for port 0, ALE, and $\overline{PSEN}$ = 100pF, load capacitance for all other outputs = 80pF.

EXPLANATION OF THE AC SYMBOLS

Each timing symbol has five characters. The first character is always 't' (= time). The other characters, depending on their positions, indicate the name of a signal or the logical status of that signal. The designations are:

A – Address
C – Clock
D – Input data
H – Logic level high
I – Instruction (program memory contents)
L – Logic level low, or ALE
P – $\overline{PSEN}$
Q – Output data
R – $\overline{RD}$ signal
t – Time
V – Valid
W – $\overline{WR}$ signal
X – No longer a valid logic level
Z – Float

Examples: t_{AVLL} = Time for address valid to ALE low.
t_{LLPL} = Time for ALE low to $\overline{PSEN}$ low.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ข.

ข้อมูลรายละเอียดของไมโครคอนโทรลเลอร์ 8031

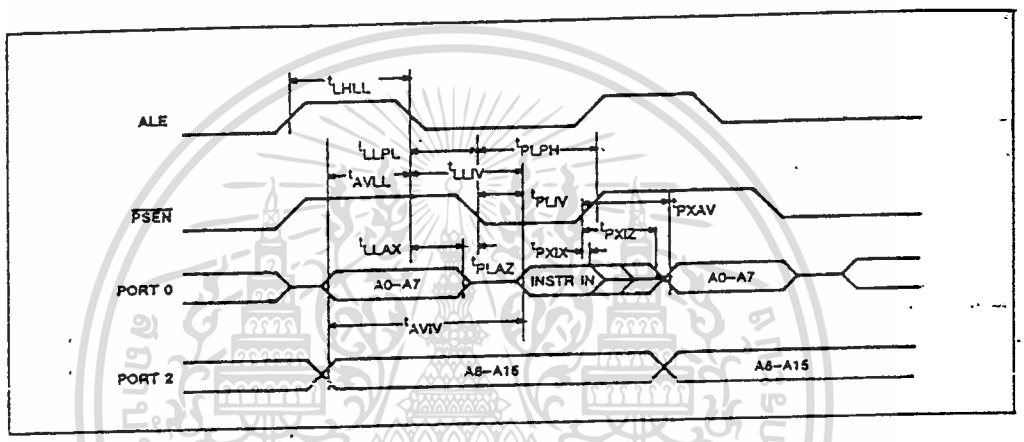


Figure 1. External Program Memory Read Cycle

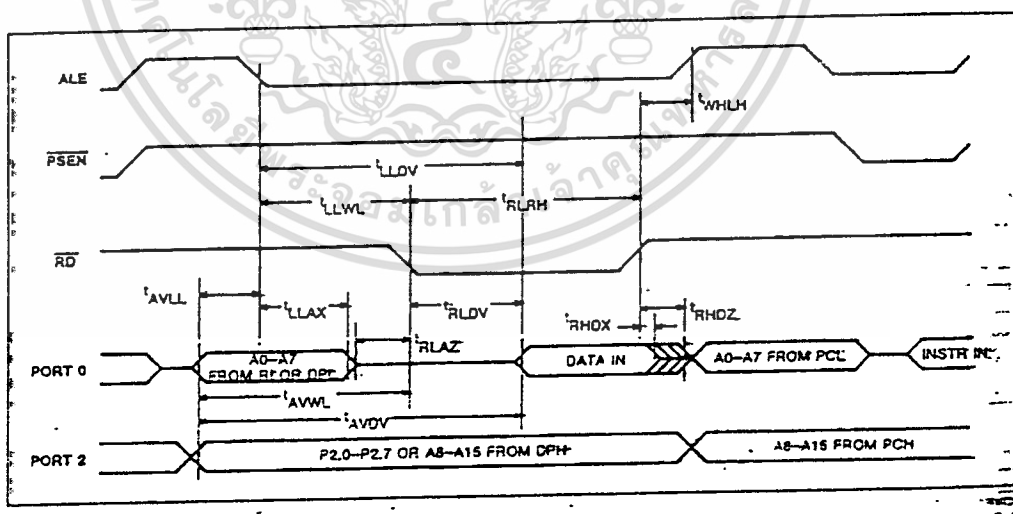


Figure 2. External Data Memory Read Cycle

ภาคผนวก ข.

ข้อมูลรายละเอียดของไมโครคอนโทรลเลอร์ 8031

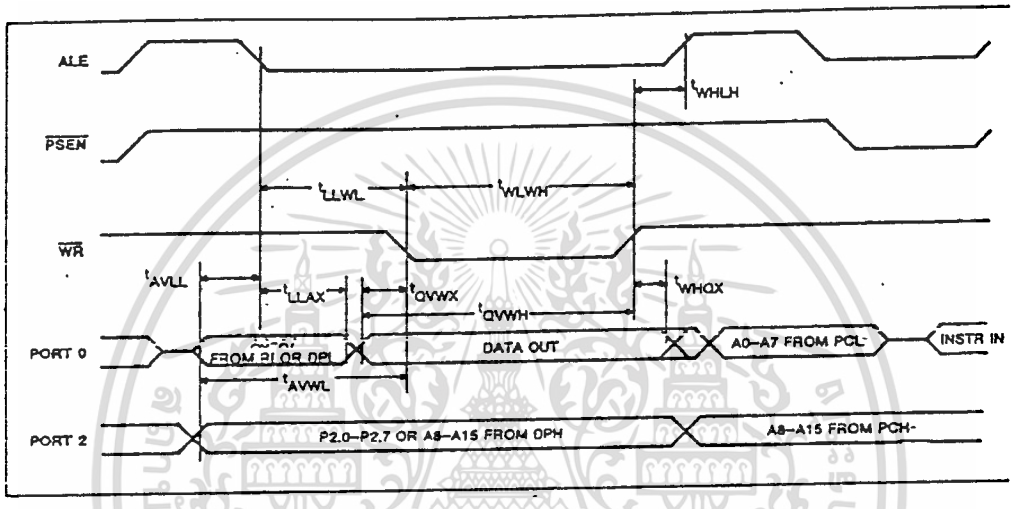


Figure 3. External Data Memory Write Cycle

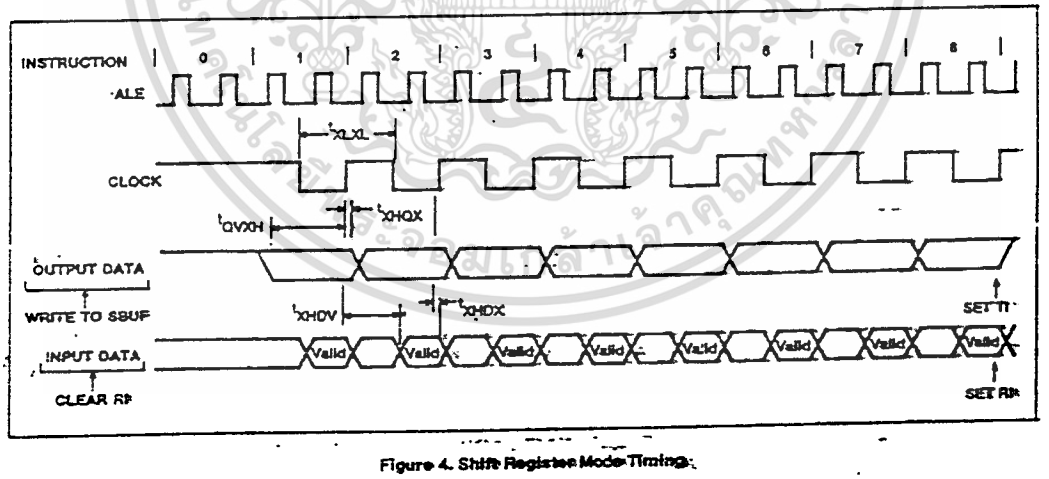


Figure 4. Shift Register Mode Timing

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ข.

ข้อมูลรายละเอียดของไมโครคอนโทรลเลอร์ 8031

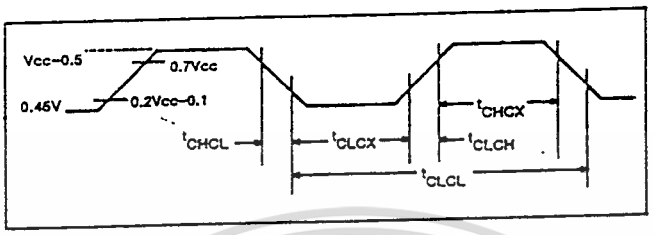


Figure 5. External Clock Drive

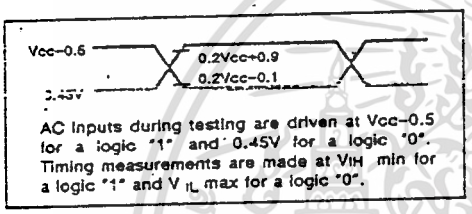


Figure 6. AC Testing Input/Output

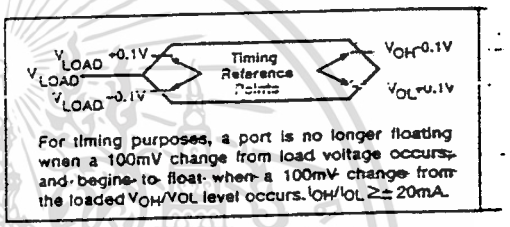


Figure 7. Float Waveform

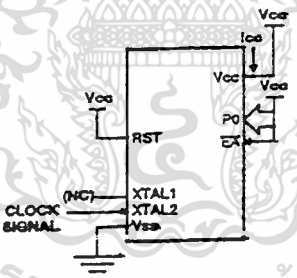


Figure 8. I_{CC} Test Condition, Active Mode
All other pins are disconnected

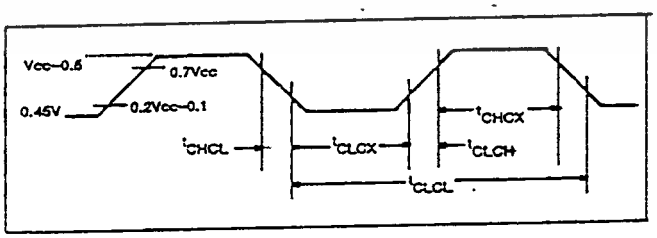


Figure 9. Clock Signal Waveform for I_{CC} Tests in Active and Idle Modes
$t_{CLCH} = t_{CHCL} = 5ns$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ข.

ข้อมูลรายละเอียดของไมโครคอนโทรลเลอร์ 8031

OSCILLATOR CHARACTERISTICS

XTAL1 and XTAL2 are the input and output, respectively, of an inverting amplifier. The pins can be configured for use as an on-chip oscillator, as shown in the logic symbol, page 1.

To drive the device from an external clock source, XTAL2 should be driven while XTAL1 is left unconnected. There are no requirements on the duty cycle of the external clock signal, because the input to the internal clock circuitry is through a divide-by-two flip-flop. However, minimum and maximum high and low times specified in the data sheet must be observed.

DESIGN CONSIDERATIONS

At power-on, the voltage on V_{CC} and RST should come up at the same time for a proper start-up.

ABSOLUTE MAXIMUM RATINGS[1, 2, 3]

PARAMETER	RATING	UNIT
Storage temperature range	−65 to +150	°C
All voltages with respect to ground	−0.5 to +7.0	V
Power dissipation	1.0	W

DC ELECTRICAL CHARACTERISTICS T_A = 0°C to +70°C, V_{CC} = 4.5 to 5.5V, V_{SS} = 0V[4, 5]

Symbol	Parameter	Test Conditions	Limits Min	Limits Max	Unit
V_{IL}	Input low voltage		−0.5	0.8	V
V_{IH}	Input high voltage, except RST and XTAL2		2	V_{CC}+0.5	V
V_{IH1}	Input high voltage to RST for reset, XTAL2	XTAL1 to V_{SS}	2.5	V_{CC}+0.5	V
V_{OL}	Output low voltage, ports 1, 2, 3[6]	I_{OL} = 1.6mA		0.45	V
V_{OL1}	Output low voltage, port 0, ALE, $\overline{PSEN}$[6]	I_{OL} = 3.2mA		0.45	V
V_{OH}	Output high voltage, ports 1, 2, 3	I_{OH} = −80µA	2.4		V
V_{OH1}	Output high voltage port 0, ALE, $\overline{PSEN}$)[3]	I_{OH} = −400µA	2.4		V
I_{IL}	Logical 0 input current, ports 1, 2, 3	V_{IN} = 0.45V		−500	µA
I_{IH1}	Input high curent to RST for reset	V_{IN} < V_{CC} − 1.5V		500	µA
I_{LI}	Input leakage current, port 0, EA	0.45 < V_{IN} < V_{CC}		±10	µA
I_{IL2}	Logical 0 input current for XTAL2	XTAL1 = V_{SS}, V_{IN} = 0.45V		−3.2	mA
I_{CC}	Power supply current:	All outputs disconnected and EA = V_{CC}		125	mA
C_{IO}	Pin capacitance			10	pF
T_A = −40°C to +85°C — Extended temperature range — SCN8051HAC only:					
V_{IH}	Input high voltage, except RST and XTAL2		2.2	V_{CC}+0.5	V
V_{IH1}	Input high voltage to RST for reset, XTAL2	XTAL1 to V_{SS}	2.7		V
I_{CC}	Power supply current:	All outputs disconnected and EA = V_{CC}		175	mA

NOTES:

1. Stresses above those listed under Absolute Maximum Ratings may cause permanent damage to the device. This is a stress rating only and functional operation of the device at these or any conditions other than those described in the AC and DC Electrical Characteristics section of this specification is not implied.
2. For operating at elevated temperatures, the device must be derated based on +150°C maximum junction temperature.
3. This product includes circuitry specifically designed for the protection of its internal devices from the damaging effects of excessive static charges. Nonetheless, it is suggested that conventional precautions be taken to avoid applying voltages greater than the rated maximum.
4. Parameters are valid over operating temperature range unless otherwise specified.
5. All voltage measurements are referenced to ground. For testing, all inputs signals swing between 0.45V and 2.4V with a transition time of 20ns maximum. All time measurements are referenced at input voltages of 0.8V and 2.0V and at output voltages of 0.8V and 2.0V as appropriate.
6. V_{OL} is degraded when the device rapidly discharges external capacitance. This AC noise is most pronounced during emission of address data. When using external memory, locate the latch or buffer as close to the device as possible.

Datum	Emitting Ports	Degraded I/O Lines	V_{OL} (Peak Max)
Address	P2, P0	P1, P3	0.8V
Write Data	P0	P1, P3, ALE	0.8V

7. CL = 100pF for port 0, ALE and $\overline{PSEN}$ outputs; CL = 80pF for all other ports.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ค.

ข้อมูลรายละเอียดของ IC เบอร์ต่างๆ

MOTOROLA
SEMICONDUCTOR
TECHNICAL DATA

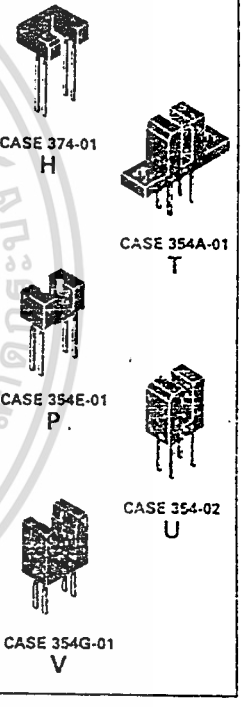
Slotted Optical Switches
Darlington Output

MOC71 Series

SLOTTED OPTICAL SWITCHES DARLINGTON OUTPUT

Each device consists of a gallium arsenide infrared emitting diode facing a silicon NPN photodarlington in a molded plastic housing. A slot in the housing between the emitter and the detector provides the means for mechanically interrupting the infrared beam. These devices are widely used as position sensors in a variety of applications.

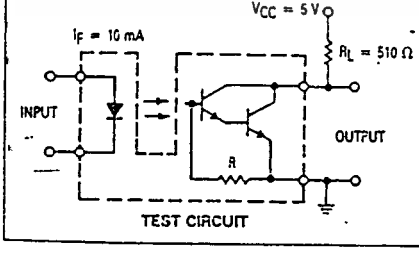
- Single Unit for Easy PCB Mounting
- Non-Contact Electrical Switching
- Long-Life Liquid Phase Epi Emitter
- Several Convenient Package Styles



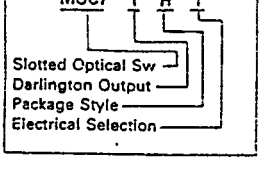
MAXIMUM RATINGS

Rating	Symbol	Value	Unit
INPUT LED			
Reverse Voltage	V_R	6	Volts
Forward Current — Continuous	I_F	60	mA
Input LED Power Dissipation @ T_A = 25°C Derate above 25°C	P_D	150 2	mW mW/°C
OUTPUT DARLINGTON			
Collector-Emitter Voltage	V_{CEO}	30	Volts
Output Current — Continuous	I_C	100	mA
Output Darlington Power Dissipation @ T_A = 25°C Derate above 25°C	P_D	150 2	mW mW/°C
TOTAL DEVICE			
Ambient Operating Temperature Range	T_A	−40 to +100	°C
Storage Temperature	T_{stg}	−40 to +100	°C
Lead Soldering Temperature (5 seconds max)	—	260	°C
Total Device Power Dissipation @ T_A = 25°C Derate above 25°C	P_D	300 4	mW mW/°C

SWITCHING TIMES



PART NUMBER DERIVATION



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ค.

ข้อมูลรายละเอียดของ IC เบอร์ต่างๆ

MOC71 Series

ELECTRICAL CHARACTERISTICS (T_A = 25°C unless otherwise noted. Note 1.)

Characteristic		Symbol	Min	Typ	Max	Unit
INPUT LED						
Forward Voltage (I_F = 50 mA)		V_F	0.9	1.3	1.8	Volts
Reverse Leakage (V_R = 6 V)		I_R	—	0.05	100	μA
Capacitance (V = 0 V, f = 1 MHz)		C_J	—	18	—	pF
OUTPUT DARLINGTON						
Dark Current (V_{CE} = 10 V)		I_{CEO}	—	10	100	nA
Collector-Emitter Breakdown Voltage (I_C = 1 mA)		$V_{(BR)CEO}$	30	90	—	Volts
Capacitance (V = 0 V, f = 1 MHz)		C_{CE}	—	5.5	—	pF
DC Current Gain (V_{CE} = 10 V, I_C = 2 mA)		h_{FE}	—	10,000	—	—
COUPLED (Note 2)						
Output Collector Current (I_F = 5 mA, V_{CE} = 5 V)	MOC71_1	I_C	2.5	5	—	mA
	MOC71_3		8	14	—	
Output Collector Current (I_F = 10 mA, V_{CE} = 5 V)	MOC71_1	I_C	7.5	15	—	mA
	MOC71_3		20	35	—	
Collector-Emitter Saturation Voltage (I_C = 1.8 mA, I_F = 10 mA)		$V_{CE(sat)}$	—	—	1	Volts
Turn-On Time (I_F = 10 mA, V_{CC} = 5 V, R_L = 510 Ω)		t_{on}	—	120	—	μs
Turn-Off Time (I_F = 10 mA, V_{CC} = 5 V, R_L = 510 Ω)		t_{off}	—	500	—	μs

Notes: 1. Stray radiation can alter values of characteristics. Adequate light shielding should be provided.
2. No actuator in sensing gap.

TYPICAL CHARACTERISTICS

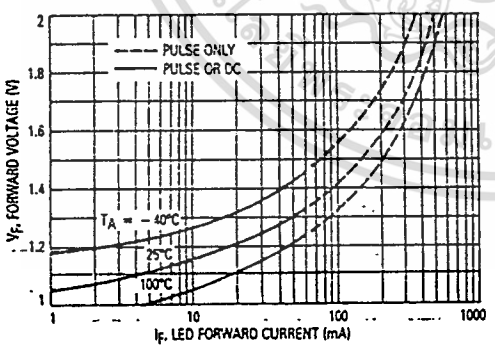


Figure 1. LED Forward Voltage versus Forward Current

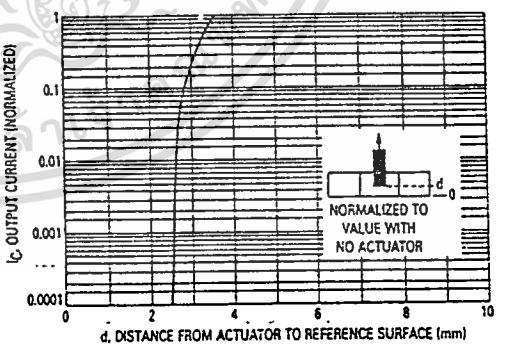


Figure 2. Output Current versus Actuator Position

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ค.

ข้อมูลรายละเอียดของ IC เบอร์ต่างๆ

MOC71 Series

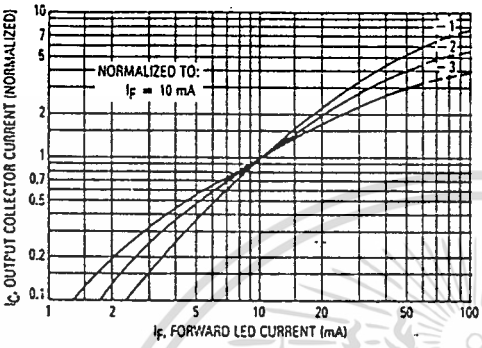


Figure 3. Output Current versus Input Current

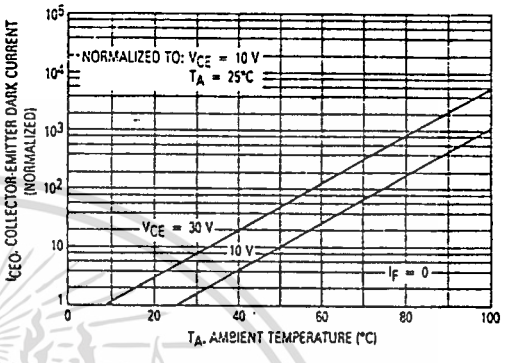


Figure 4. Collector-Emitter Dark Current versus Ambient Temperature

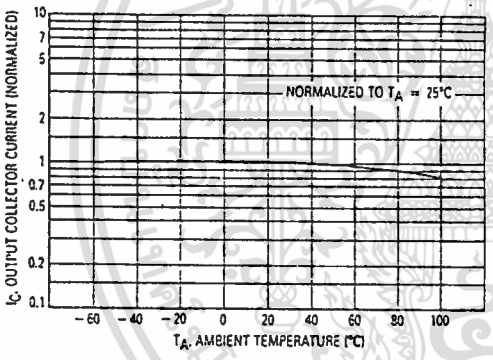


Figure 5. Output Current versus Ambient Temperature

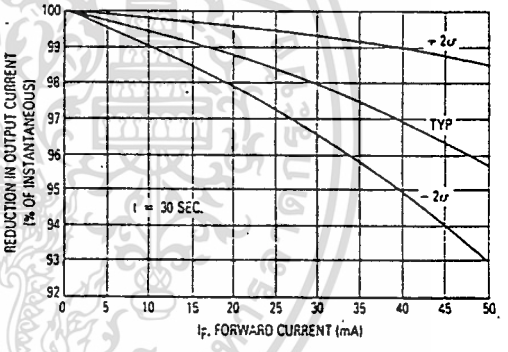


Figure 6. Reduction in Output Current Due to LED Heating versus Forward Current

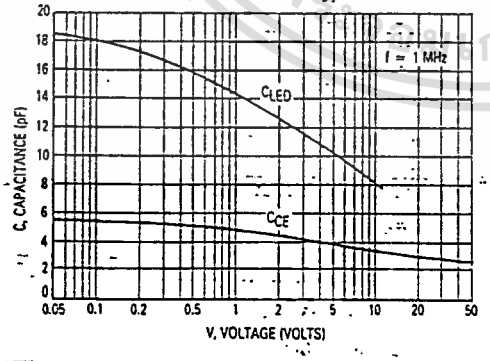


Figure 7. Capacitances versus Voltage

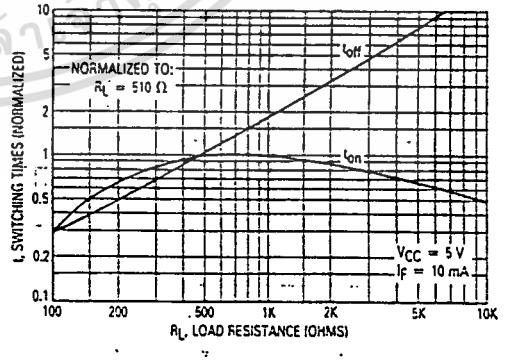


Figure 8. Switching Times versus Load Resistance

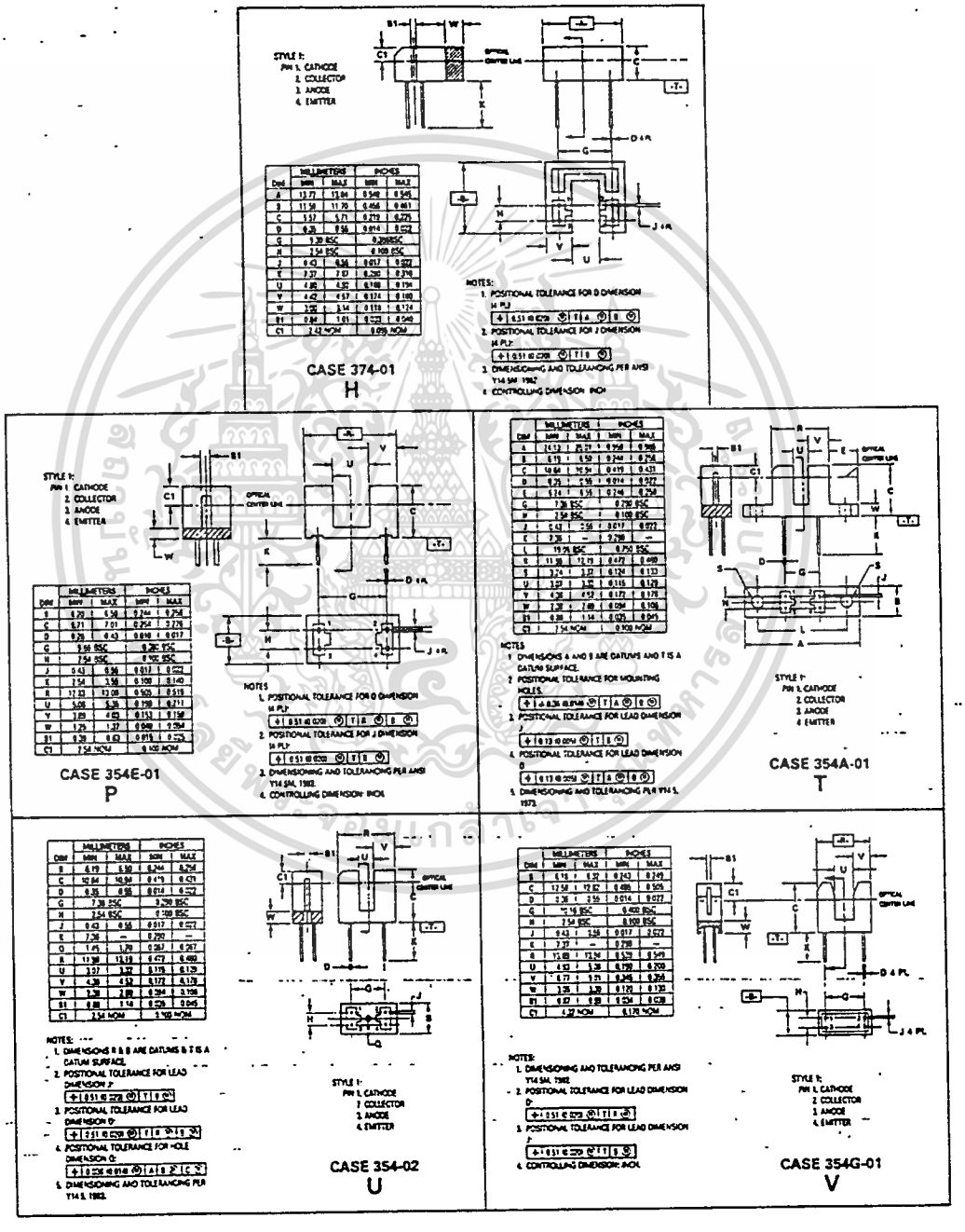
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ค.

ข้อมูลรายละเอียดของ IC เบอร์ต่างๆ

MOC71 Series

OUTLINE DIMENSIONS



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ค.

ข้อมูลรายละเอียดของ IC เบอร์ต่างๆ

MOTOROLA
SEMICONDUCTOR
TECHNICAL DATA

Slotted Optical Switch
Darlington Output

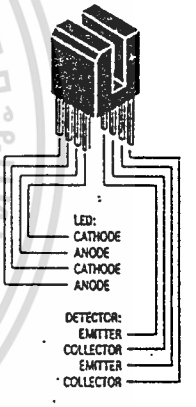
MOC71W1

DUAL CHANNEL
SLOTTED
OPTICAL SWITCH
DARLINGTON OUTPUT

These devices consist of two gallium arsenide infrared emitting diodes facing two NPN silicon photodarlingtons across a 0.100" wide slot in the housing. Switching takes place when an opaque object in the slot interrupts the infrared beam.

Dual channel interrupters (switches) can sense *direction* of motion as well as position and speed.

- High Gain Darlington Output
- 0.020" Detector Aperture Width
- Easy PCB Mounting
- Cost Effective
- Uses Long-Lived LPE IRED



CASE 792-01

ABSOLUTE MAXIMUM RATINGS (25°C)

Rating	Symbol	Value	Unit
INPUT LED			
Power Dissipation	P_D	150*	mW
Forward Current (Continuous)	I_F	60	mA
Reverse Voltage	V_R	6	V
OUTPUT DARLINGTON			
Power Dissipation	P_D	150*	mW
Collector-Emitter Voltage	V_{CEO}	30	V
TOTAL DEVICE			
Storage Temperature	T_{stg}	−40 to +85	°C
Operating Temperature	T_J	−40 to +85	°C
Lead Soldering Temperature (5 seconds maximum)	T_L	260	°C

*Derate 2 mW/°C above 25°C ambient.

INDIVIDUAL ELECTRICAL CHARACTERISTICS (25°C) (See Note 1)

Characteristic	Symbol	Min	Typ	Max	Unit
INPUT LED					
Reverse Breakdown Voltage (I_R = 100 μA)	$V_{(BR)R}$	6	—	—	V
Forward Voltage (I_F = 60 mA)	V_F	—	—	1.8	V
Reverse Current (V_R = 6 V, R_L = 1 MΩ)	I_R	—	50	—	nA
Capacitance (V = 0 V, f = 1 MHz)	C	—	25	—	pF
OUTPUT DARLINGTON					
Breakdown Voltage (I_C = 1 mA, I_F = 0)	$V_{(BR)CEO}$	30	—	—	V
Collector Dark Current (V_{CE} = 10 V, I_F = 0)	I_{CEO}	—	—	100	nA

NOTE 1: Stray irradiation can alter values of characteristics. Adequate shielding should be provided.

7

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ค.

ข้อมูลรายละเอียดของ IC เบอร์ต่างๆ

MOC71W1

COUPLED ELECTRICAL CHARACTERISTICS (25°C, See Note 1)

Characteristics	Symbol	Min	Typ	Max	Unit
I_F = 5 mA, V_{CE} = 5 V	$I_{CE(on)}$	2.5	—	—	mA
I_F = 10 mA, V_{CE} = 5 V	$I_{CE(on)}$	7.5	—	—	mA
I_F = 10 mA, I_C = 1.8 mA	$V_{CE(sat)}$	—	—	1	V
I_F (opposite LED) = 10 mA, V_{CE} = 5 V	I_{CX}	—	20	—	μA

NOTE 1: Stray irradiation can alter values of characteristics. Adequate shielding should be provided.

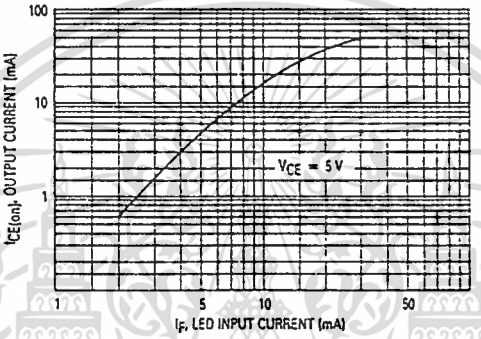


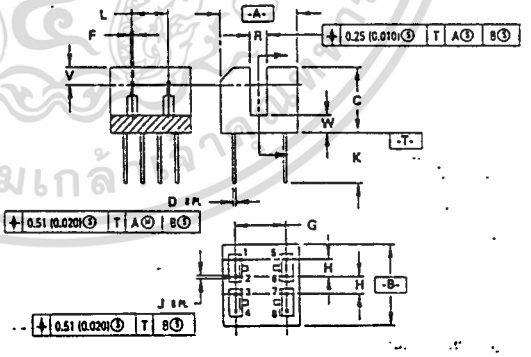
Figure 1. Typical Output Current versus Input Current

OUTLINE DIMENSIONS

NOTES:
1. DIMENSIONING AND TOLERANCING PER ANSI Y14.5M, 1982.
2. CONTROLLING DIMENSION: INCH.

DIM	MILLIMETERS MIN	MAX	INCHES MIN	MAX
A	11.21	11.50	0.445	0.457
B	11.77	12.11	0.463	0.477
C	9.35	9.70	0.368	0.382
D	0.36	0.55	0.014	0.022
F	0.49	0.53	0.019	0.021
G	7.62 BSC		0.300 BSC	
H	2.54 BSC		0.100 BSC	
J	0.44	0.55	0.017	0.022
K	7.37	—	0.290	—
L	5.38 BSC		0.212 BSC	
R	2.37	2.71	0.093	0.107
V	2.54 BSC		0.100 BSC	
W	2.37	2.71	0.093	0.107

STYLE 2:
PIN 1. CATHODE
2. ANODE
3. CATHODE
4. ANODE
5. COLLECTOR
6. EMITTER
7. COLLECTOR
8. EMITTER



CASE 792-01

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ค.

ข้อมูลรายละเอียดของ IC เบอร์ต่างๆ

MOTOROLA
SEMICONDUCTOR
TECHNICAL DATA

Photo Detectors
Transistor Output

MRD300
MRD310

PHOTO DETECTORS
TRANSISTOR OUTPUT
NPN SILICON

... designed for application in industrial inspection, processing and control, counters, sorters, switching and logic circuits or any design requiring radiation sensitivity, and stable characteristics.

- Popular TO-18 Type Package for Easy Handling and Mounting
- Sensitive Throughout Visible and Near Infrared Spectral Range for Wider Application
- Minimum Light Current 4 mA at H = 5 mW/cm$_2$ (MRD300)
- External Base for Added Control
- Annular Passivated Structure for Stability and Reliability

CASE 82-05
METAL

4

MAXIMUM RATINGS (T_A = 25°C unless otherwise noted)

Rating	Symbol	Value	Unit
Collector-Emitter Voltage	V_{CEO}	50	Volts
Emitter-Collector Voltage	V_{ECO}	7	Volts
Collector-Base Voltage	V_{CBO}	80	Volts
Total Device Dissipation @ T_A = 25°C Derate above 25°C	P_D	250 2.27	mW mW/°C
Operating Temperature Range	T_A	−55 to +125	°C
Storage Temperature Range	T_{stg}	−65 to +150	°C

STATIC ELECTRICAL CHARACTERISTICS (T_A = 25°C unless otherwise noted)

Characteristic	Symbol	Min	Typ	Max	Unit
Collector Dark Current (V_{CE} = 20 V, H = 0) T_A = 25°C T_A = 100°C	I_{CEO}	— —	5 4	25 —	nA μA
Collector-Base Breakdown Voltage (I_C = 100 μA)	$V_{(BR)CBO}$	80	120	—	Volts
Collector-Emitter Breakdown Voltage (I_C = 100 μA)	$V_{(BR)CEO}$	50	85	—	Volts
Emitter-Collector Breakdown Voltage (I_E = 100 μA)	$V_{(BR)ECO}$	7	8.5	—	Volts

OPTICAL CHARACTERISTICS (T_A = 25°C unless otherwise noted)

Light Current (V_{CC} = 20 V, R_L = 10 Ohms) Note 1	MRD300 MRD310	I_L	4 1	7 3.5	— —	mA
Light Current (V_{CC} = 20 V, R_L = 100 Ohms) Note 2	MRD300 MRD310	I_L	— —	2.5 0.8	— —	mA
Photo Current Rise Time (Note 3) (R_L = 100 Ohms, I_L = 1 mA peak)		t_r	—	2	2.5	μs
Photo Current Fall Time (Note 3) (R_L = 100 Ohms, I_L = 1 mA peak)		t_f	—	2.5	4	μs

NOTES: 1. Radiation flux density (H) equal to 5 mW/cm^2 emitted from a tungsten source at a color temperature of 2870 K.
2. Radiation flux density (H) equal to 0.5 mW/cm^2 (pulsed) from a GaAs (gallium-arsenide) source at λ = 940 nm.
3. For unsaturated response time measurements, radiation is provided by pulsed GaAs (gallium-arsenide) light-emitting diode (λ = 940 nm) with a pulse width equal to or greater than 10 microseconds (see Figure 2) I_L = 1 mA peak.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ค.

ข้อมูลรายละเอียดของ IC เบอร์ต่างๆ

MRD300, MRD310

TYPICAL CHARACTERISTICS

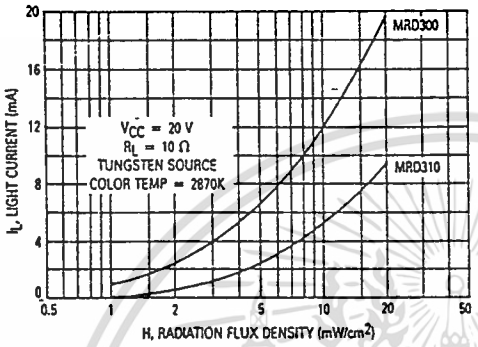


Figure 1. Light Current versus Irradiance

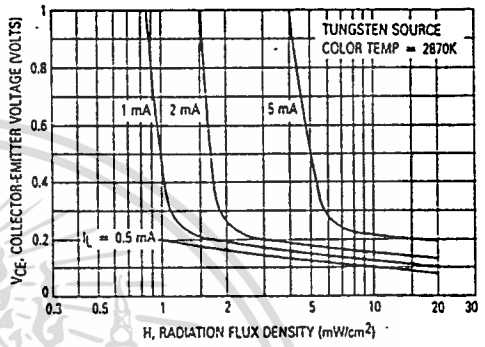


Figure 2. Collector-Emitter Saturation Characteristic

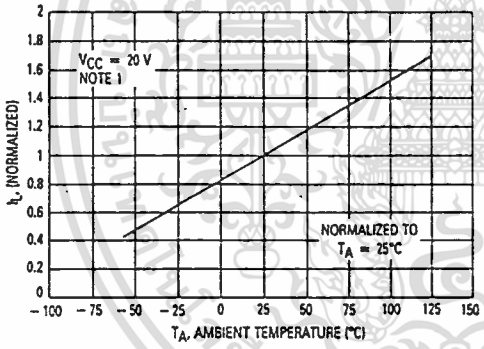


Figure 3. Normalized Light Current versus Temperature

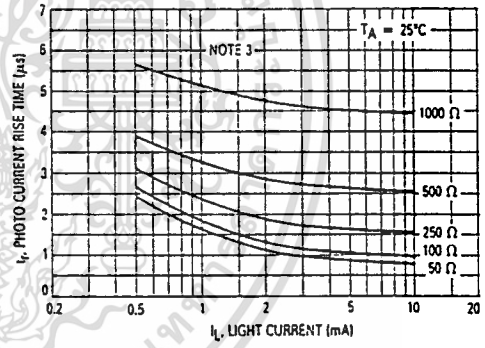


Figure 4. Rise Time versus Light Current

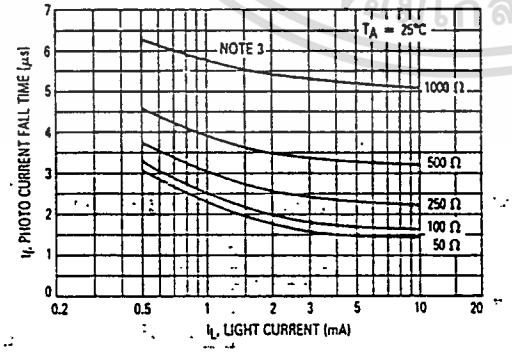


Figure 5. Fall Time versus Light Current

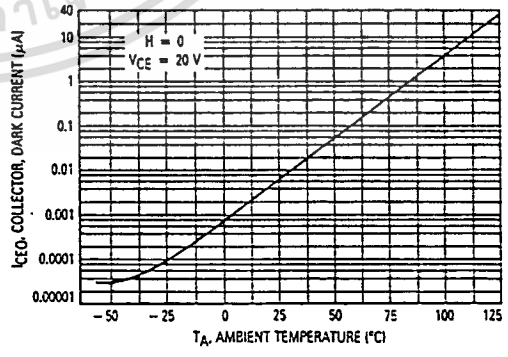


Figure 6. Dark Current versus Temperature

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ค.

ข้อมูลรายละเอียดของ IC เบอร์ต่างๆ

MRD300, MRD310

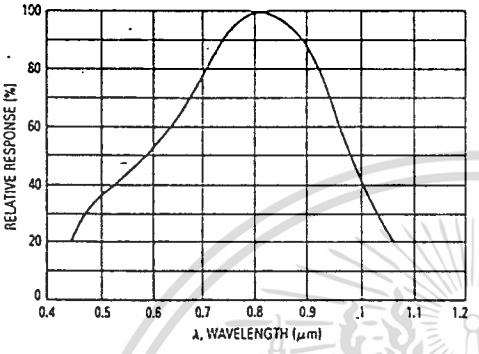


Figure 7. Constant Energy Spectral Response

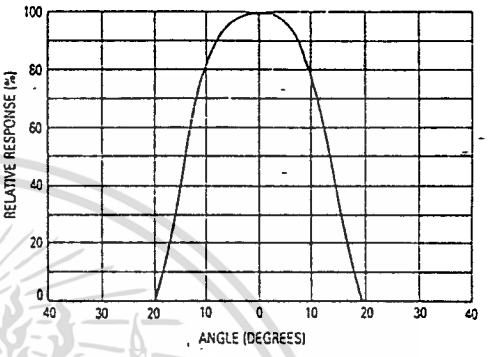


Figure 8. Angular Response

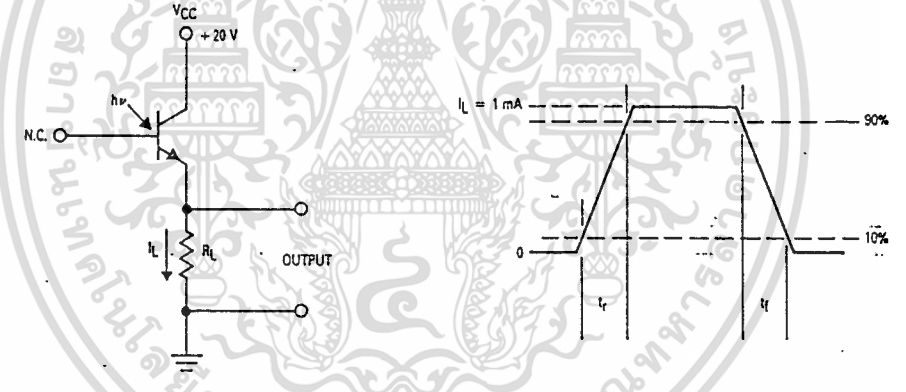
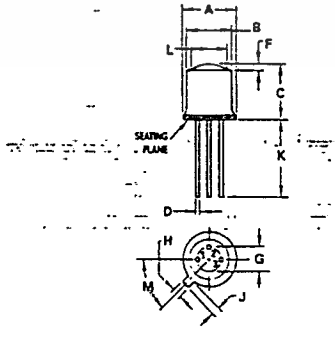


Figure 9. Pulse Response Test Circuit and Waveform

OUTLINE DIMENSIONS



NOTES:
1. LEADS WITHIN .13 mm (.005) RADIUS OF TRUE POSITION AT SEATING PLANE, AT MAXIMUM MATERIAL CONDITION.
2. PIN 3 INTERNALLY CONNECTED TO CASE.

STYLE 1:
PIN 1. EMITTER
2. BASE
3. COLLECTOR

DIM	MILLIMETERS MIN	MILLIMETERS MAX	INCHES MIN	INCHES MAX
A	5.31	5.84	0.209	0.230
B	4.52	4.95	0.178	0.195
C	4.57	6.48	0.180	0.255
D	0.41	0.48	0.016	0.019
F	—	1.14	—	0.045
G	2.54 BSC		0.100 BSC	
H	0.99	1.17	0.039	0.046
J	0.84	1.22	0.033	0.048
K	12.70	—	0.500	—
L	3.35	4.01	0.132	0.158
M	45° BSC		45° BSC	

CASE 82-05
METAL

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

This page intentionally left blank

1.0 DESCRIPTION

Precision Navigation, Inc.'s Vector 2X is a low-cost, 2-axis compass module suitable for OEM applications. Using PNI's patented magneto-inductive magnetometer technology (United States Patent #4,851,775), the Vector 2X delivers high accuracy, low power, and low cost in the same small size package. The module delivers 2-degree accuracy with 1-degree resolution, outperforming all compasses in its price range. The Vector 2X is simple to calibrate and incorporate into systems designed by first-time users or experienced experts.

The Vector 2X Compass Module offers:

- 2° accuracy, 1° resolution
- Small size and weight: 1.50" x 1.30" x 0.39"; 0.3 oz.
- Low power requirement: single-sided 5 volt (regulated) supply, less than 10 milliamp (mA) draw (<1 mA in powerdown mode)
- Low resolution output is 5 hertz (Hz); high resolution output is 2.5 Hz
- Input voltage on all I/O ports: -0.3 to V_{DD} +0.3 volts
- Output voltage on all I/O ports: -0.3 to V_{DD} +0.3 volts
- Output ports can source or sink 5 mA
- 3-wire serial output format (Motorola SPI- and National Micro-wire-compatible) makes for easy installation into OEM systems
- Hard iron calibration
- Raw uncalibrated output option
- Pin-selectable Binary-Coded Decimal (BCD) or Binary output format
- Polled or continuous mode options
- Master or slave capability for data clock generation
- Operating temperature range: -20°C to 70°C

Possible applications for the Vector 2X include:

- GPS system backup azimuth data
- Remote terrestrial antenna direction indicators
- Auto and industrial navigation systems
- Consumer and hobbyist markets
- Survey instruments
- Laboratory equipment
- Robotics systems
- Azimuth indicators

2.0 OVERVIEW

2.1 Features and Specifications

A general board layout of the Vector 2X is shown in Figure 1; Table 1 presents the pin configuration. The pin programmability allows users to set up the compass for their specific needs. The Vector 2X uses a synchronous serial port for communication to the host device (compatible with Motorola SPI, QSPI, and National Microwire). The output data format is pin-selectable between BCD and binary mode, and the serial data output (SDO) is tri-stated when the unit is not selected in Slave mode.

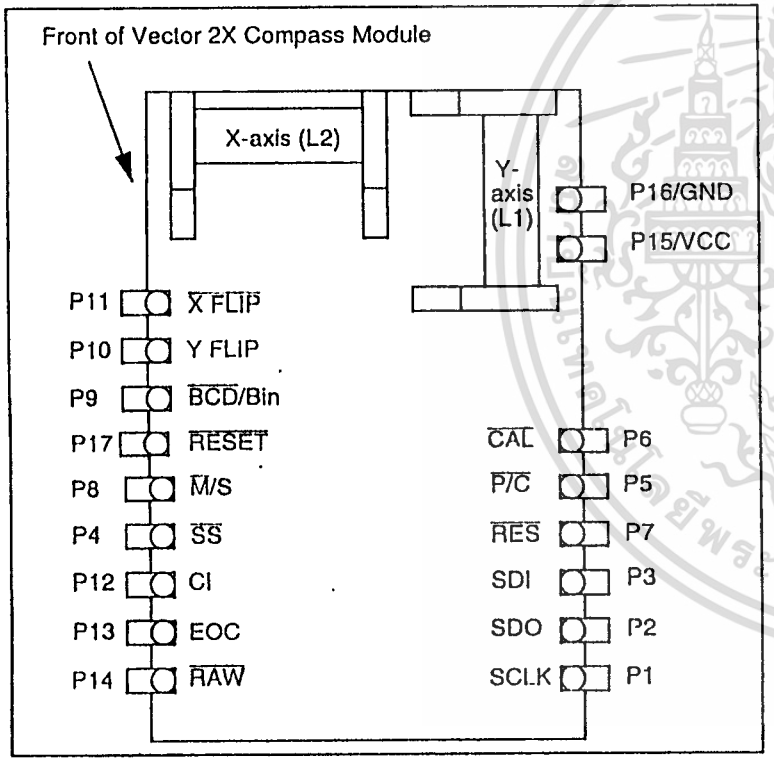


Figure 1: Vector 2X Board Layout

Table 1: Vector 2X FUNCTION DESCRIPTION

Pin	Name	Description	Input or Output
P1	SCLK	Serial Clock Input; Data valid on rising clock edge*	Input/Output
P2	SDO	Serial Data Output	Output
P3	SDI	Serial Data Input	No Connect
P4	$\overline{SS}$	Slave Select; Active Low	Input
P5	$\overline{P/C}$	Poll / Continuous; Active when $\overline{P/C}$ = Low	Input
P6	$\overline{CAL}$	Calibrate Select; Active when $\overline{CAL}$ = Low	Input
P7	$\overline{RES}$	Resolution; Low Resolution when $\overline{RES}$ = Low	Input
P8	$\overline{M/S}$	Master/Slave select; Master Mode when $\overline{M/S}$ = Low	Input
P9	$\overline{BCD}$/Bin	BCD/Binary select; BCD when $\overline{BCD}$/Bin = Low	Input
P10	Y FLIP	Flip Y-axis direction. Low = Normal Direction	Input
P11	$\overline{X\ FLIP}$	Flip X-axis direction. High = Normal Direction	Input
P12	CI	Calibrate indicator signal; Active High	Output
P13	EOC	End of Conversion signal; Conversion complete on rising edge of EOC	Output
P14	$\overline{RAW}$	Raw data mode select pin; Active Low	Input
P15	VCC	5 volt power input	
P16	GND	Power supply return	
P17	$\overline{RESET}$	Reset pin; Active Low	Input

* Maximum clock rate is 1 MHz in Slave mode

The $\overline{\text{X FLIP}}$ and Y FLIP pins provide a means to re-orient the Vector 2X so that it can be mounted in an inverted position. The $\overline{\text{X FLIP}}$ pin inverts the north and south directions, and the Y FLIP inverts the east and west directions. The normal mode is $\overline{\text{X FLIP}}$ = 1 and Y FLIP = 0.

The $\overline{\text{RES}}$ pin is normally grounded unless higher resolution output is required for more counts in the RAW mode or for areas with weak magnetic field strength (with the Vector 2X in either Master or Slave mode). High resolution output is 2.5 Hz; low resolution output is 5 Hz.

The $\overline{\text{P/C}}$, $\overline{\text{CAL}}$, $\overline{\text{RESET}}$ and $\overline{\text{SS}}$ pins must be logic high during powerup; these four ports have internal 100 kohm pullup resistors and must be connected to bidirectional I/Os set to inputs. Otherwise, the Vector 2X will hang upon powerup and can be reset by toggling the $\overline{\text{RESET}}$ pin after startup.

The Vector 2X is in sleep mode under normal operation and draws very low current, only becoming active when acquiring or outputting data. The power required in sleep mode is less than 1 mA; during operation the power required rises to less than 10 mA.

2.2 Operating modes of the Vector 2X

The Vector 2X operates in three modes: Master mode, Slave mode, and RAW mode. The configurations for running the Vector 2X in each of these modes are described below:

2.2.1 Master mode

The Vector 2X can run in Master mode and generate the serial data clock. In Master mode the Vector 2X clocks out the data using an internally-generated clock. This feature allows the Vector 2X to load data into devices such as serial-to-parallel shift registers without the intervention of a host processor. When in Master mode the EOC provides a positive edge to indicate the complete 16-bit word has been sent. SDO is never tri-stated in Master mode.

Figure 2 shows the timing diagram for the Vector 2X in Master mode. The $\overline{\text{M/S}}$ pin must be tied low to enable Master mode, and $\overline{\text{SS}}$ should not be connected. The $\overline{\text{P/C}}$ pin is used to request data from the Vector 2X, and EOC is used to frame data. The Vector 2X, in Master mode, can collect and output heading information in either polled or continuous mode. Pulsing the $\overline{\text{P/C}}$ pin low for a minimum of 10 millisecond (msec) polls the compass for a single reading, and holding the $\overline{\text{P/C}}$ pin low causes the Vector 2X to output continuously (both $\overline{\text{P/C}}$ modes are represented in Figure 2 - and subsequent figures - by a solid line showing the pulsed polled mode, and a dotted line illustrating the continuous mode). EOC frames the continuous readings; the readings are updated at a rate of approximately 5 Hz at low resolution and 2.5 Hz at high resolution. The SCLK pin is an output in Master mode; the SCLK pin outputs 16 clock pulses at a rate of 4 KHz for each conversion. If $\overline{\text{P/C}}$ is held low, the Vector 2X starts another cycle of readings automatically. The first six bits output from the SDO pin will always be zero in either Binary or BCD mode, and the first significant bit (FSB) after the first six bits will always be a zero in Binary mode.

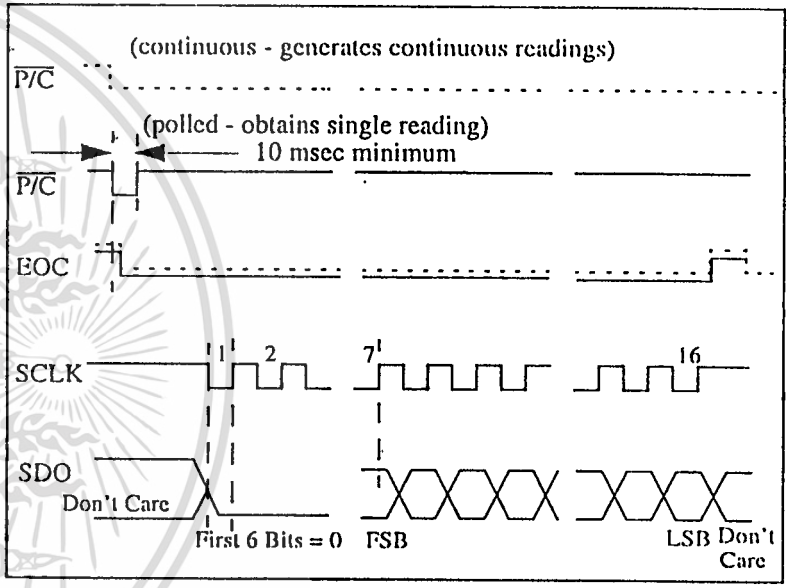


Figure 2: Data Clock Timing Diagram (Master Mode 16-bit Data Format)

2.2.2 Slave Mode

When the Vector 2X operates in Slave mode, the host generates the data clock; the $\overline{\text{M/S}}$ pin must be open or tied high to enable this mode. The Vector 2X, in Slave mode, can be connected to a synchronous serial port along with other devices. SDO remains in a tri-state mode until the host requests data from the device by toggling and holding $\overline{\text{SS}}$ low for the duration of the read, at which time it becomes an output. The EOC pin supplies a rising edge to indicate that the Vector 2X has completed the calculations. This feature frees the serial port for other devices to communicate with the host.

Figure 3 illustrates the data clock timing diagram with the Vector 2X in Slave mode, 16-bit data format. In Slave mode, the SCLK is an input. External SCLK must be supplied by the host processor to clock out data on SDO; frequencies of up to a maximum of 1 MHz are acceptable. Sixteen complete clocks (i.e., 16 falling edges and 16 rising edges) must be supplied to the Vector 2X in order to obtain data; otherwise, data may remain in the Vector 2X registers and the compass will hang. When the $\overline{P}/\overline{C}$ pin is held low in Slave mode, the Vector 2X outputs a single heading until polled again; minimum pulse width on $\overline{P/C}$ is 10 msec. The timing between the rise of EOC and the falling edge of $\overline{SS}$ should be greater than 10 msec; data from the Vector 2X is output on the falling edge of SCLK and valid on the rising edge of each SCLK pulse. The first six leading 0s in the data format are constant in both Binary and BCD formats. The FSB will always be a zero in binary mode. The host must clock out all 16 bits from the Vector 2X to complete data transfer before attempting to acquire new data.

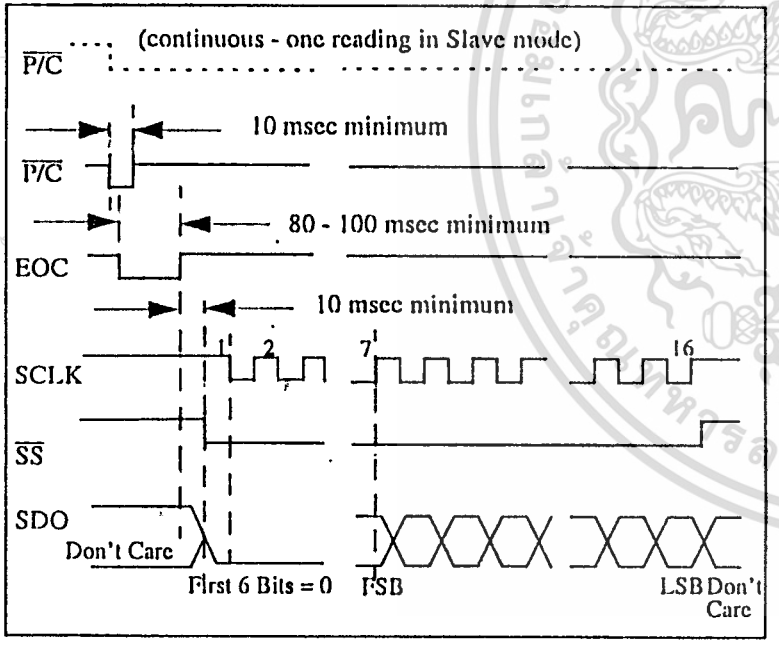


Figure 3: Data Clock Timing Diagram (Slave Mode 16-bit Data Format)

2.2.3 RAW Mode

RAW data output is only available in Slave mode and not in Master mode. The SCLK pin is an input in RAW mode. Figure 4 displays the RAW mode data format. The $\overline{M}/S$ pin must be open or tied high to enable this mode, and the $\overline{RAW}$ pin must be tied low. The data format is available only in Binary output format, as four bytes (total 32 bits). The outputs are signed 16-bit integers; the first two bytes make up the binary number of the X-axis field intensity, and the last two bytes are the binary number for the Y-axis field intensity. The 16-bit values for the X-axis and Y-axis are not calibrated for magnetic scale.

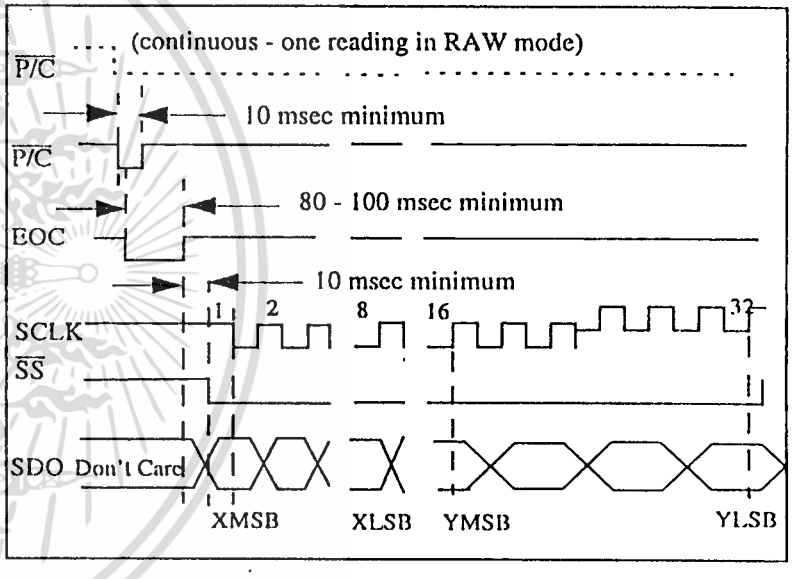


Figure 4: Raw Data Mode Format (Slave Mode only, 16-bit Data Format)

3.0 CALIBRATION AND RESET

3.1 Calibrating the Vector 2X

Hard iron distortions from magnetized materials can cause the compass to display improper heading and RAW mode information. The $\overline{CAL}$ pin is used to calibrate offsets caused by these hard iron distortions. The Vector 2X measures the magnetic field strength in both the X-axis and Y-axis; calibration removes the distortion, and proper heading is obtained.

To calibrate the unit, pulse the $\overline{CAL}$ pin low for at least 10 msec, then rotate the Vector 2X in a half-circle (180 degrees). Pulse the $\overline{CAL}$ pin again for at least 10 msec and calibration is complete. When $\overline{CAL}$ is pulsed the first time, the CI-pin goes high and stays high until the $\overline{CAL}$ pin is pulsed a second time. To abort the calibration or clear a previous calibration, pulse the $\overline{CAL}$ pin twice without moving the Vector 2X. **Note: $\overline{P/C}$ cannot be low when the Vector 2X is in calibration mode.**

3.2 Resetting the Vector 2X

In the event the Vector 2X enters an unknown state or hangs, it may be necessary to reset the processor. Reset is performed by applying a low-going pulse to the $\overline{RESET}$ pin for a duration of not less than 10 msec.

4.0 APPLICATIONS FOR THE VECTOR 2X

Typical applications for the Vector 2X are described below; they demonstrate how to connect the Vector 2X to various types of circuits and are **not intended to be finished designs.**

Note: For the schematics shown in Figures A1 to A3 of Appendix A, the battery is used to maintain RAM data, so hard iron calibration is not lost during powerdown of circuit. Regulated +5 volt is still required for operation of the Vector 2X.

4.1 Application 1: Serial-to-Parallel Conversion

The circuit described here is a simple Vector 2X with serial-to-parallel shift registers and an LED display. The schematic for this application is shown as Figure A1 in Appendix A.

The Vector 2X is used in the following configuration:

Inputs:	$\overline{M/S} = 0$	Master mode operation
	$\overline{BCD}/Bin = 0$	BCD output format
	$\overline{RES} = 0$	Low resolution
	$\overline{X\ FLIP}$ = open	X normal
	Y FLIP = 0	Y normal
	$\overline{RAW}$ = open	Not RAW magnetometer output mode; inaccessible in Master mode

Figure 5 illustrates the timing diagram for this arrangement.

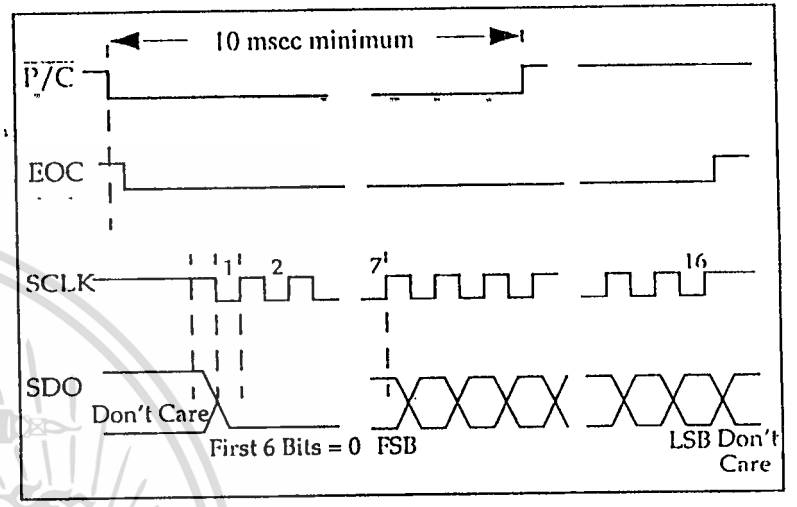


Figure 5: Timing Diagram for Connecting Vector 2X with Serial-to-Parallel Shift Registers

In this configuration, the Vector 2X is ready to output data to the shift registers when the $\overline{P/C}$ pin is grounded. If $\overline{P/C}$ is held low continuously, the heading data should be read out entirely before the Vector 2X outputs the next reading. SDO of the Vector 2X is connected to the serial input of the shift registers, which converts the serial data stream to a parallel format. The data from the Vector 2X is clocked with the SCLK signal, and the format is most significant bit (MSB) first. Ten bits are required to indicate 0 to 359 degrees (BCD), and the first six bits are always sent as zeroes. The parallel outputs of the shift registers are connected to two 8-bit latches. The data is latched on the rising edge of EOC; this EOC output frames the data and drops low at the beginning of the transfer and returns high after the last bit is transferred. When $\overline{P/C} = 0$, the Vector 2X acquires heading information and outputs the data when the calculations are complete. The data at the output of the latches is BCD-formatted data which can be connected to an LCD or LED through the appropriate display driver ICs. The Vector 2X acquires and outputs heading data for as long as the $\overline{P/C}$ pin is pulled low; when the $\overline{P/C}$ pin is pulled high, the Vector 2X finishes the last heading, outputs the data and then returns to sleep mode.

4.2 Application 2: Vector 2X Connected to a Motorola Processor

The Vector 2X can be connected to a Motorola 6811C705C8 processor, or similar processor, with a synchronous serial bus. A good discussion on SPI can be found in any Motorola Microprocessor, Microcontroller and Peripheral Data handbook. Sample SPI read code can be found in Appendix B; its corresponding schematic is shown in Figure B1.

Figure 6 shows the timing diagram for this connection; the schematic is presented in Figure A2 of Appendix A.

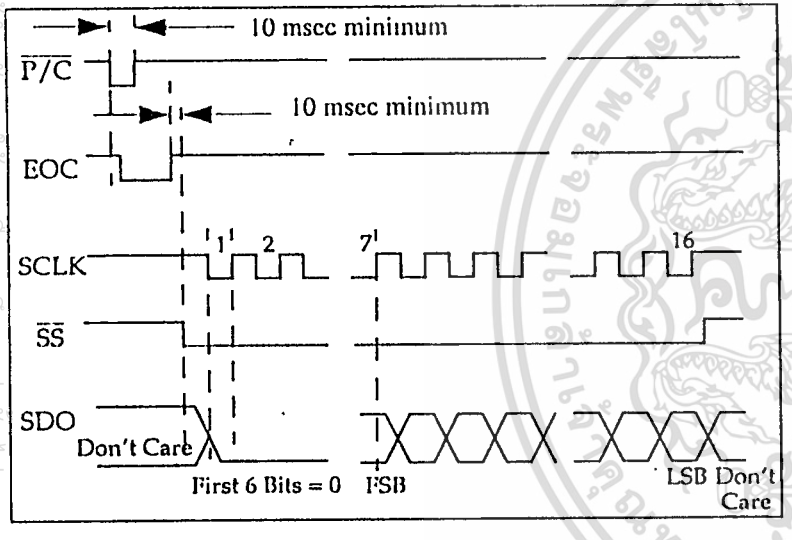


Figure 6: Timing Diagram for Connecting Vector 2X to a Motorola or Intel-Type Processor

As shown in Figure A2, the Vector 2X shares the serial bus with a MC14489P display driver. The SCLK line from the processor is connected to both the display driver and the Vector 2X; SDO is connected to PD2/MISO (master in, slave out) and SDI is not connected. The processor requests heading data from the Vector 2X, retrieves the data, reformats the data into BCD format suitable for the display driver, then outputs the data to the display driver. (The output from the Vector 2X in this case is set to Binary output mode, but it can also be set to BCD format if so desired.)

The Vector 2X is used in the following configuration:

Inputs:	M/S = open	Slave mode operation
	BCD/Bin = open or closed	Either format
	RES = 0 or 1	Depending on application
	X FLIP = open	X normal
	Y FLIP = 0	Y normal
	RAW = open	Not RAW magnetometer output mode; inaccessible in Master mode

These settings set the Vector 2X to Slave mode. X-axis and Y-axis are in the normal positions, and the input to the RES pin determines resolution.

The Motorola processor was set to acquire data from the Vector 2X at a rate of 0.5 second per data request. The processor updates the display and goes into a wait mode until 0.5 second elapses, then it repeats the cycle. The P/C pin is used to tell the Vector 2X to acquire new heading data. The SS pin and the P/C pin are connected to different ports of the processor, which allows the processor to talk to other devices on the bus while the Vector 2X calculates heading.

The Vector 2X wakes up from sleep mode on the falling edge of P/C. The EOC pin drops when the Vector 2X starts to acquire data; the rising edge on the EOC pin indicates heading data has been acquired, and the Vector 2X returns to sleep mode. At this point the data is available at the serial port of the Vector 2X; the processor must now drop SS low to retrieve the data and provide 16 complete clocks (16 falling and 16 rising edges) to retrieve the data. In Slave mode, SCLK must be supplied by the processor in order to output the data.

The data is valid on each rising edge of the 16 clock pulses of SCLK. The processor must supply the SCLK pin with the serial data clock; 16 complete clocks (16 falling and 16 rising edges) are required to transfer all of the data from the Vector 2X. After the 16th clock elapses, the processor must raise SS, which allows the Vector 2X to return to sleep mode and prepare for the next cycle of P/C readings.

Note: The P/C, CAL, RESET and SS pins must be logic high during powerup; these four ports have internal 100 kohm pullup resistors and must be connected to bidirectional I/Os set to inputs. Otherwise, the Vector 2X will hang and can be reset by toggling the RESET pin after startup.

4.3 Application 3: Vector 2X Connected to an Intel-type Processor

Some applications may employ the Vector 2X to control a stepper motor to keep the orientation of the directional component of the system constant with the direction of the compass. In this configuration, the Vector 2X is connected to a non-SPI processor (see Figure A3 of Appendix A). To start the Vector 2X, the host drops $\overline{P/C}$ for 10 msec and raises it again, which initiates the compass data acquisition process. The EOC line drops soon after the falling edge of $\overline{P/C}$ and returns to a high state when the heading calculation is complete; at this time, the data is ready and can be retrieved from the Vector 2X by dropping $\overline{SS}$ and providing 16 complete clocks to SCLK.

The data is valid on each rising edge of the 16 clock pulses of SCLK. The processor must supply the SCLK pin with the serial data clock; 16 complete clocks are required to transfer all of the data from the Vector 2X (see Figure 6 for timing diagram). The data from the Vector 2X appears at SDO and must be logically ORed with the accumulator, then shifted right eight times for each byte.

An alternate method for connecting the Vector 2X to a host processor is illustrated in Figure A4 of Appendix A; this configuration can be used if there are no other serial devices to be connected.

The $\overline{P/C}$ pin and the $\overline{SS}$ pin are connected together to reduce the number of I/O pins required. The EOC pin can be connected to the processor to signal the processor that the data is ready to be clocked out using SCLK. This application ties up the serial port for the entire time the Vector 2X is calculating the heading. The $\overline{P/C}$ pin is kept low until the data is retrieved only because it is tied together to the $\overline{SS}$ pin; $\overline{P/C}$ does not have to be held low. Toggling $\overline{P/C}$ before the data is completely output causes the Vector 2X to overwrite the old data, even if there is untransmitted data from the previous conversion.

Note: The $\overline{P/C}$, $\overline{CAL}$, $\overline{RESET}$ and $\overline{SS}$ pins must be logic high during powerup; these ports have internal 100 kohm pullup resistors and must be connected to bidirectional I/Os set to inputs. Otherwise, the Vector 2X will hang and can be reset by toggling the $\overline{RESET}$ pin after startup.

Appendix A

Schematics for Section 4.0

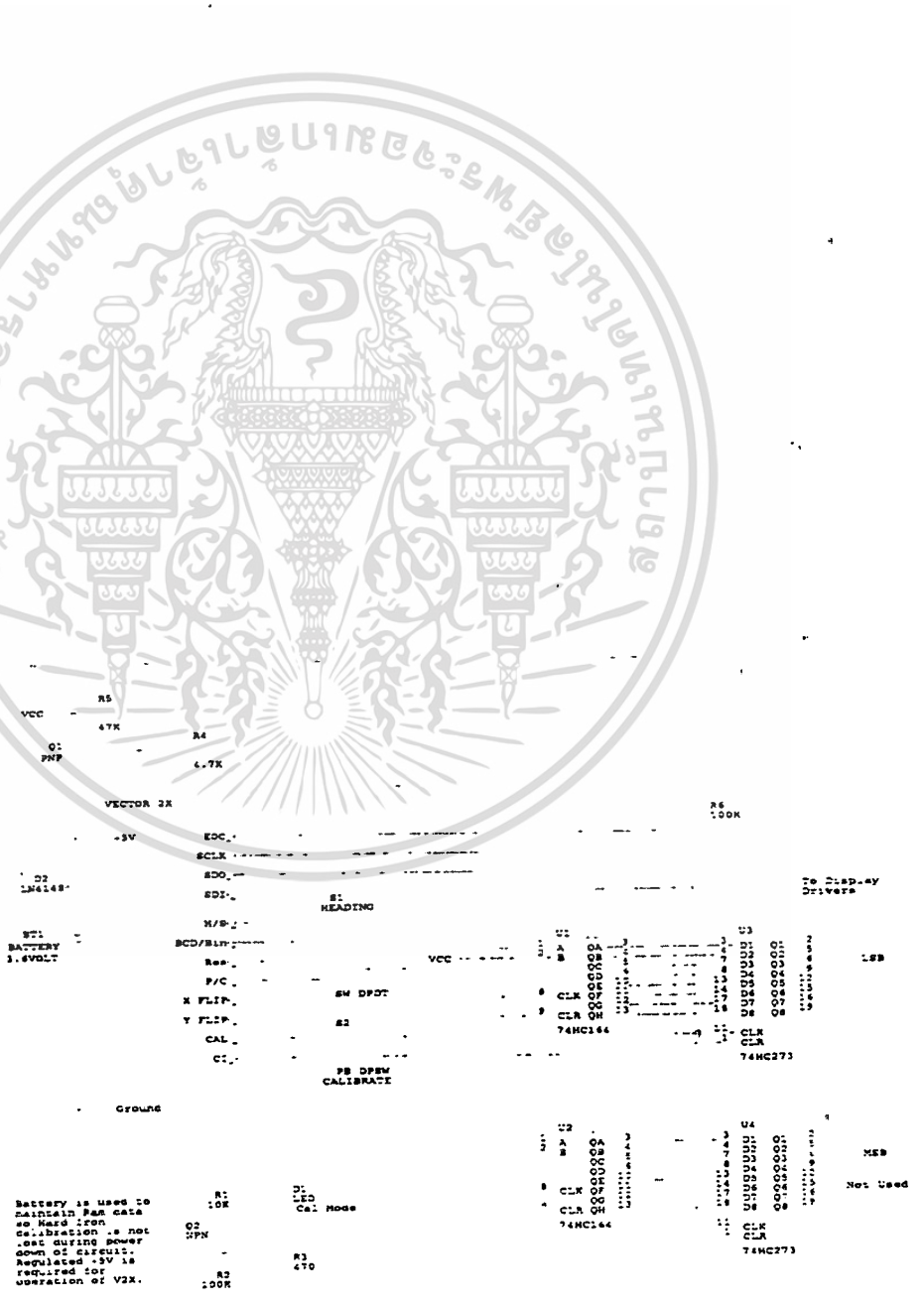


Figure A1: Schematic for Connecting Vector 2X with Serial-to-Parallel Shift Registers

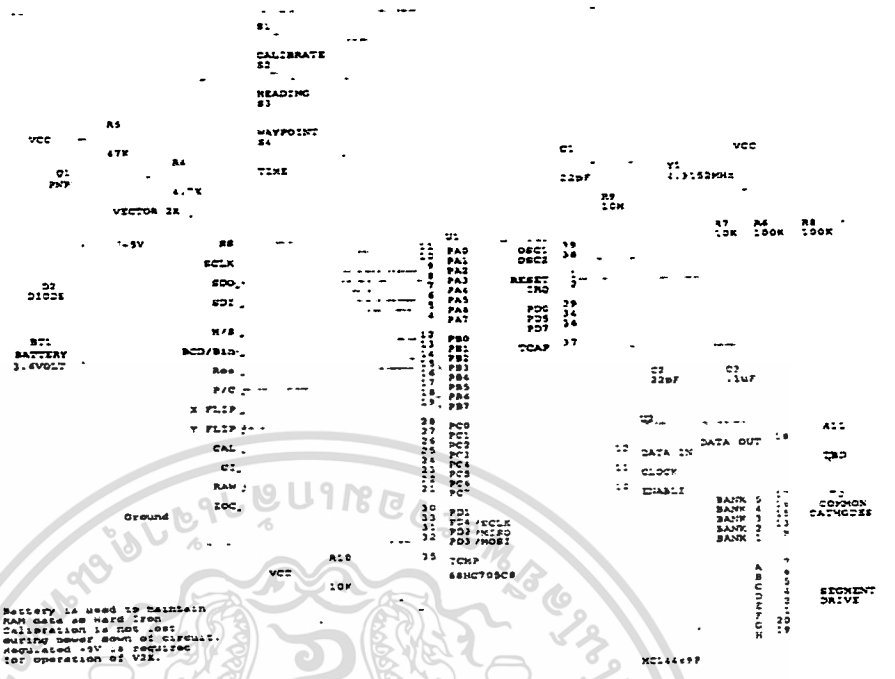


Figure A2: Schematic for Connecting Vector 2X to a Motorola Processor

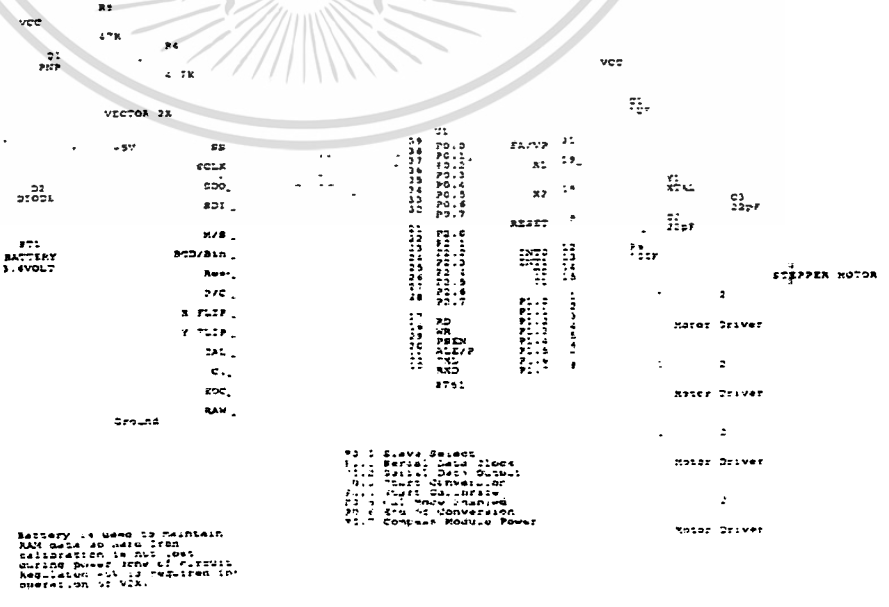


Figure A3: Schematic for Connecting Vector 2X to an Intel-Type Processor

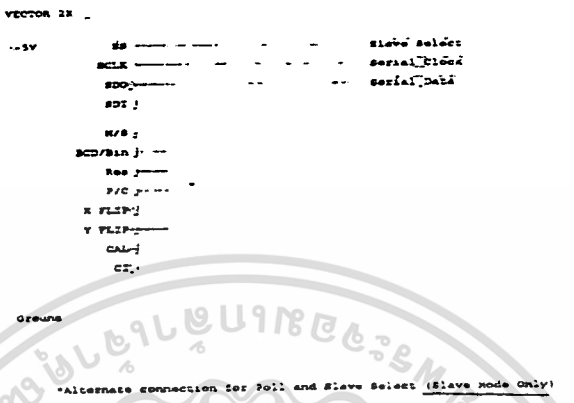


Figure A4: Schematic for Alternate Metod of Connecting Vector 2X to an Intel-Type Processor

Appendix B

Sample SPI read Code

```
*Sample SPI read code for Motorola 68HC705 family
*                    processor
*  Note: Not executable with schematics shown in
*       Appendix A of V2X Application Notes!
*************************************************************
*   SPI - Master mode (68HC705)
*   Port A: Input
*     A0:  EOC from V2X(End of Conversion)
*     A7:  RES select (Resolution Select)
*
*   Port C1: Slave select bar to V2X
*     C0:  SS  (Slave Select [bar]) to V2X
*     C7:  P/C (Poll/Continuous [bar])
*************************************************************
*   Equates
          EOC     equ     0     ;EOC input to processor
          RES     equ     7     ;Resolution select
          SS      equ     0     ;Slave select from processor
          POLL    equ     1     ;Poll/Continuous control
*************************************************************
*RAM Location
          org  $050             ;ORG RAM

BYTE1             rmb     1     ;high order byte temp storage
BYTE2             rmb     1     ;low order byte temp storage
*************************************************************
*ROM Location
          org  $200             ;ORG ROM

*   slave_read - read the 2 bytes of data from the V2X over
    the SPI port
*   data is returned in byte1, byte2
*************************************************************
slave_read:
*Start V2X to acquire heading
    bclr poll,portc    ;drop POLL line to compass
    jsr delay10msec    ;wait a while ~10 msec
    bset poll,portc    ;raise POLL line after 10 msec
```

```
*Wait for V2X to complete conversion for heading data
    jsr   waiteoc      ;now wait for EOC to go high

*Retrieve data from V2X
    bclr 1,portc       ;lower SS line to enable xfer
    tst SPSR           ;see if SPI is clear
    sta SPDR           ;write to SPDR to start SPI xfer

*Wait for data transfer to complete
rspiflp1:
    brclr 7,spsr,rspiflp1     ;keep looping till SPI xfer
                              ;done

*Get the first data byte
    lda SPDR           ;
    sta BYTE1          ;put in BYTE1 reg
    jsr   delayinterb ;delay between bytes
    clra               ;set a = 0

*Start transfer of second byte
    sta SPDR           ;write to SPI

*Wait for data transfer to complete
rspiflp2:
    brclr 7,spsr,rspiflp2     ;keep looping till done

*Get the second data byte
    lda SPDR           ;get second byte
    sta byte2          ;

*****************************************************
*   Waiteoc: after the poll has been started wait for the
    eoc to go high to mark end of measurements and computa
    tions
*****************************************************
waiteoc:
      brclr  EOC,Porta,waiteEOC    ;branch if EOC bit is
                                   ;clear; keep looping
                                   ;till EOC returns high

      rts
```

```
*****************************************************
* Delay 10msec - delay 10 milliseconds
* uses a & x registers for delay countdown
*****************************************************

delay 10msec:
      ldx    #!20          ;20 passes of olp: delay loop
      bra    olp
olp:
      lda    #!156         ;156 x 3 µsec delay
ilp:
      deca
      bne    ilp
      decx
      bne    olp
      rts

*****************************************************
* delayinterb - delay about 160 µsec between bytes read in
*order to give time to SPI to elad second byte.
*****************************************************

delayinterb:
      lda    #!50          ;50 x 3 µsec delay
      bra    olp
dilp:
      deca
      bne    dilp
      rts
```

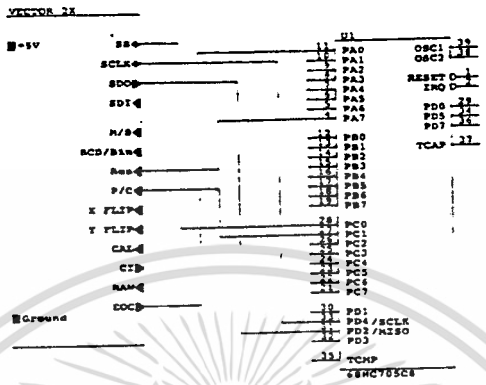


Figure B1: Schematic for SPI read Code Example

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รายงานความก้าวหน้าครั้งที่ 1

รายชื่อผู้เข้าร่วมโครงงาน

1. นางสาวกรรณิการ์	พรหมหิตาธร	รหัส 36014005
2. นายเพชร	พงษ์พานิช	รหัส 36014301
3. นายวิเชษฐ์	สัตนา	รหัส 36014405

ชื่อโครงงาน

(ไทย) รถเอนกประสงค์

(อังกฤษ) Multipurpose Vehicle (MPV)

อาจารย์ที่ปรึกษา

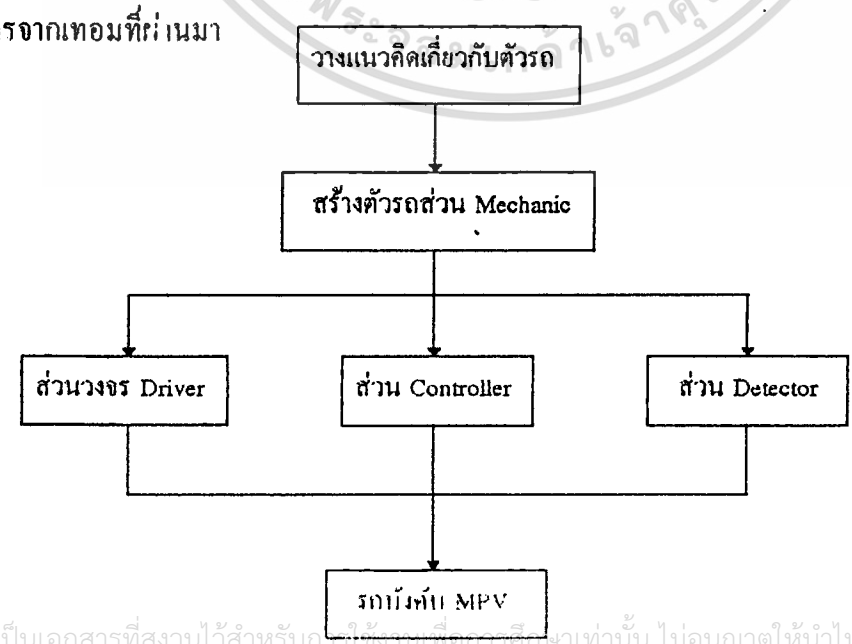
อาจารย์พลศาสตร์ เลิศประเสริฐ

วัตถุประสงค์

เพื่อประดิษฐ์รถที่สามารถควบคุมให้เดินทางเข้าสู่เป้าหมายและหลบหลีกสิ่งกีดขวางได้ โดยใช้ Microcontroller

แผนการดำเนินงาน (Block Diagram)

แผนการจากเทอมที่ผ่านมา



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

แผนการดำเนินงานในเทอมนี้

เป็นการเพิ่มเติมประสิทธิภาพให้กับรถ โดย การเพิ่มวงจร และเขียนโปรแกรมที่สมบูรณ์

ส่วนที่เพิ่มเข้ามาในเทอมนี้

1. ส่วนวงจรขับเคลื่อน (Driver Circuit)
   - เพิ่มประสิทธิภาพโดยการควบคุมความเร็วของรถด้วยวิธี พัลส์วิธมอดดูเลชั่น (Pulse Width Modulation)
2. ส่วนวงจรตรวจจับ (Detector Circuits)
   - เพิ่มวงจร Pyroelectric Detector เป็นวงจรตรวจจับเป้าหมาย
   - เพิ่มวงจรขับเคลื่อน Stepping Motor เพื่อเพิ่มประสิทธิภาพให้กับวงจรตรวจจับสิ่งกีดขวางในรัศมี 180 องศา
3. ส่วนการควบคุม
   - เขียนโปรแกรมควบคุมการทำงานของรถโดยสมบูรณ์
   - เพิ่มประสิทธิภาพในการควบคุมทิศทางที่แน่นอนของรถโดยใช้ เข็มทิศอิเล็กทรอนิกส์

สิ่งที่ได้ดำเนินงานไปแล้ว

- สร้างวงจรขับเคลื่อน Stepping Motor
- สร้างวงจรพัลส์วิธมอดดูเลชั่น (Pulse Width Modulation)
- ศึกษาการทำงานของ เข็มทิศอิเล็กทรอนิกส์

สิ่งที่จะดำเนินงานต่อไป

- สร้างวงจร Pyroelectric Detector
- เขียนโปรแกรมควบคุมการทำงานของรถ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ปัญหาและอุปสรรค

-

ความเห็นอาจารย์ที่ปรึกษา

1. ติดตั้งชุด STEPPING MOTOR
2. ทดลองโปรแกรมสั่งการ SCAN
3. นัดวัน DEMO.

ลายเซ็นอาจารย์ที่ปรึกษา

(อาจารย์พลศาสตร์ เลิศประเสริฐ)

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

รายงานความก้าวหน้าครั้งที่ 2

รายชื่อผู้เข้าร่วมโครงงาน

1. นางสาวกรรณิการ์ พรหมหิตาธร รหัส 36014005
2. นายเพชร พงษ์พานิช รหัส 36014301
3. นายวิเชษฐ์ สัตนา รหัส 36014405

ชื่อโครงงาน

(ไทย) รถเอนกประสงค์

(อังกฤษ) Multipurpose Vehicle (MPV)

อาจารย์ที่ปรึกษา

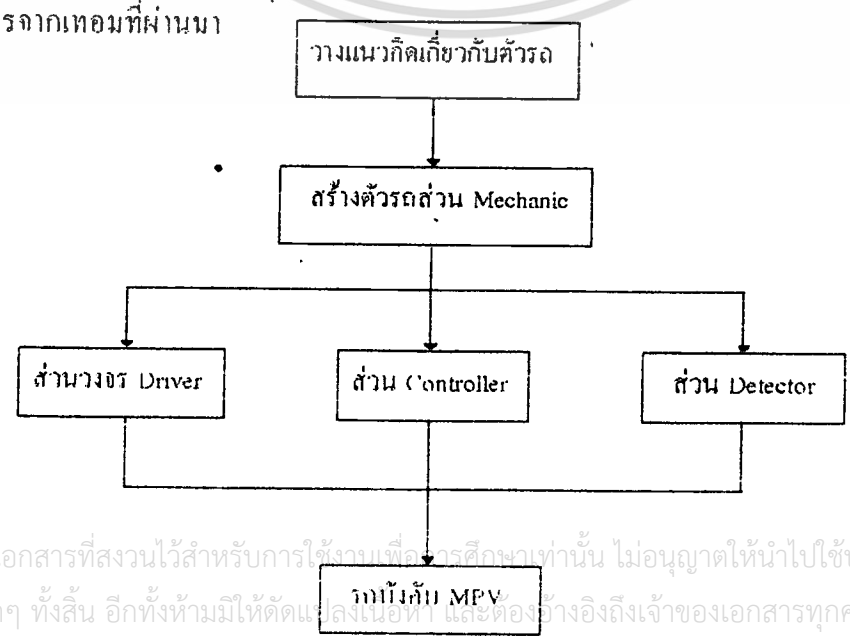
อาจารย์พลศาสตร์ เลิศประเสริฐ

วัตถุประสงค์

เพื่อประดิษฐ์รถที่สามารถควบคุมให้เดินทางเข้าสู่เป้าหมายและหลบหลีกสิ่งกีดขวางได้ โดยใช้ Microcontroller

แผนการดำเนินงาน (Block Diagram)

แผนการจากเทอมที่ผ่านมา



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

แผนการดำเนินงานในเทอมนี้

เป็นการเพิ่มเติมประสิทธิภาพให้กับรถ โดย การเพิ่มวงจร และเขียนโปรแกรมที่สมบูรณ์ ส่วนที่เพิ่มเข้ามาในเทอมนี้

1. ส่วนวงจรตรวจจับ (Detector Circuits)

- เพิ่มวงจร Pyroelectric Detector เป็นวงจรตรวจจับเป้าหมาย
- เพิ่มวงจรขับเคลื่อน Stepping Motor เพื่อเพิ่มประสิทธิภาพให้กับวงจรตรวจจับสิ่งกีดขวางในรัศมี 180 องศา

2. ส่วนการควบคุม

- เขียนโปรแกรมควบคุมการทำงานของรถโดยสมบูรณ์
- เพิ่มประสิทธิภาพในการควบคุมทิศทางที่แน่นอนของรถโดยใช้ เข็มทิศอิเล็กทรอนิกส์

สิ่งที่ได้ดำเนินงานไปแล้ว

- สร้างวงจร Pyroelectric Detector
- รวบรวมทฤษฎีต่างๆ ที่เกี่ยวข้องกับการทำงานในส่วนต่างๆ

สิ่งที่จะดำเนินงานต่อไป

- เขียนโปรแกรมควบคุมการทำงานของรถ
- จัดทำรายงานฉบับสมบูณร์

ปัญหาและอุปสรรค

วงจร Pyroelectric Detector ให้ผลที่ไม่สมบูณร์เท่าที่ต้องการ

ความเห็นอาจารย์ที่ปรึกษา

_____

_____

ลายเซ็นอาจารย์ที่ปรึกษา

_____

(อาจารย์พงศ์สันต์ เกิดประเสริฐ)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่อนุญาตให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รายงานความก้าวหน้าครั้งที่ 3

รายชื่อผู้เข้าร่วมโครงงาน

1. นางสาวกรรณิการ์ พรหมหิตาธร รหัส 36014005
2. นายเพชร พงษ์พานิช รหัส 36014301
3. นายวิเชษฐ์ สัตนา รหัส 36014405

ชื่อโครงงาน

(ไทย) รถเอนกประสงค์

(อังกฤษ) Multipurpose Vehicle (MPV)

อาจารย์ที่ปรึกษา

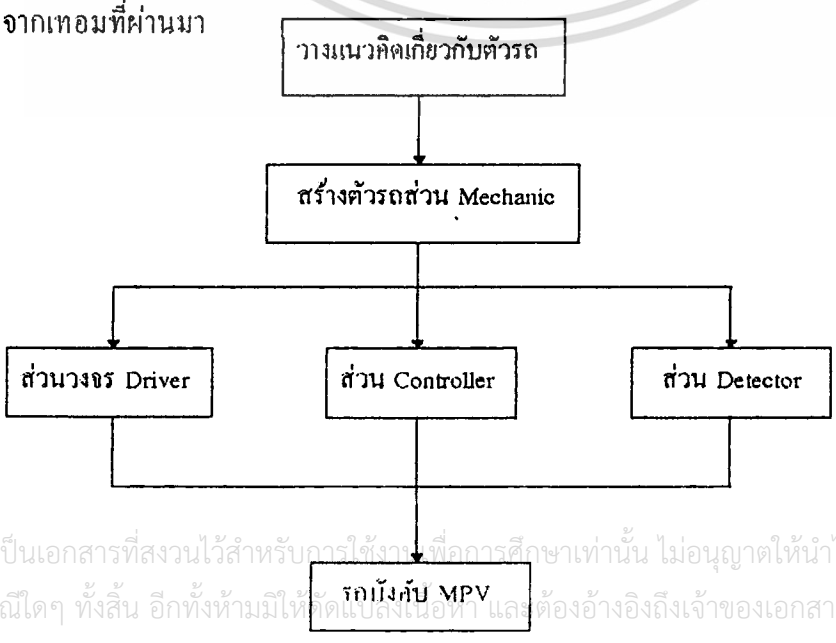
อาจารย์พลศาสตร์ เลิศประเสริฐ

วัตถุประสงค์

เพื่อประดิษฐ์รถที่สามารถควบคุมให้เดินทางเข้าสู่เป้าหมายและหลบหลีกสิ่งกีดขวางได้ โดยใช้ Microcontroller

แผนการดำเนินงาน (Block Diagram)

แผนการจากเทอมที่ผ่านมา



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

แผนการดำเนินงานในเทอมนี้

เป็นการเพิ่มเติมประสิทธิภาพให้กับรถ โดย การเพิ่มวงจร และเขียนโปรแกรมที่สมบูรณ์ ส่วนที่เพิ่มเข้ามาในเทอมนี้

1. ส่วนวงจรตรวจจับ (Detector Circuits)

- เพิ่มวงจร Pyroelectric Detector เป็นวงจรตรวจจับเป้าหมาย
- เพิ่มวงจรขับเคลื่อน Stepping Motor เพื่อเพิ่มประสิทธิภาพให้กับวงจรตรวจจับสิ่งกีดขวางในรัศมี 180 องศา

2. ส่วนการควบคุม

- เขียนโปรแกรมควบคุมการทำงานของรถโดยสมบูรณ์
- เพิ่มประสิทธิภาพในการควบคุมทิศทางที่แน่นอนของรถโดยใช้ เข็มทิศอิเล็กทรอนิกส์

สิ่งที่ได้ดำเนินงานไปแล้ว

- รวบรวมทฤษฎีและรายละเอียดส่วนต่างๆ ของโครงงาน
- จัดพิมพ์รายงานไปแล้วบางส่วน
- เขียนโปรแกรมการทำงานไปแล้วบางส่วน

สิ่งที่จะดำเนินงานต่อไป

- เขียนโปรแกรมควบคุมการทำงานของรถ
- จัดทำรายงานฉบับสมบูณร์

ปัญหาและอุปสรรค

วงจร Pyroelectric Detector ให้ผลที่ไม่สมบูณร์เท่าที่ต้องการ

ความเห็นอาจารย์ที่ปรึกษา

_____

_____

ลายเซ็นอาจารย์ที่ปรึกษา

_____

(อาจารย์พลศาสตร์ เลิศประเสริฐ)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่อนุญาตให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้