สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

รถหลบหลีกสิ่งกีดขวางอัตโนมัติ

AUTOMATIC CAR

โดย

นายกิตติ โรจนภิญญู

นายเกรียงไกร ลักษณะบัณฑิต

เลขหมู่.....
เลขทะเบียน 50404
วัน,เดือน,ปี 13 พ.ค. 2547

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

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต

สาขาวิชาอิเล็กทรอนิกส์

คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา2545

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รถหลบหลีกสิ่งกีดขวางอัตโนมัติ

Automatic car

โดย

นายกิตติ โรจนภิญญู 42010018

นายเกรียงไกร ลักษณะบัณฑิต 42010028

อาจารย์ที่ปรึกษา

อาจารย์ชินภัทร นันทจิวากรชัย

ปริญญานิพนธ์สำหรับปริญญาวิศวกรรมศาสตร์บัณฑิต

สาขาวิชา Eletronics

คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2545

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กิตติกรรมประกาศ

ขอขอบคุณทุกคนที่มีส่วนเกี่ยวข้องที่ทำให้รายงานนี้ประสบผลสำเร็จลงได้ ทั้งนี้เนื่องจากบุคคลหลายฝ่ายไม่ว่าจะเป็นคณาจารย์ ซึ่งได้แก่ อาจารย์ชินภัทร นันทจิวากรชัย อาจารย์ที่ปรึกษาโครงงาน ที่ให้คำแนะนำต่างๆ ที่สำคัญอยู่เสมอ อาจารย์โกศล ชวนขยัน ที่เอื้อเฟื้อห้องโปรเจคให้ได้ทดลอง อาจารย์วัฒนา โพธิ์เจริญ ที่ช่วยไขปัญหาที่ยากจะขบคิดเองได้ อีกทั้งอาจารย์ทุกท่านในภาคที่ให้คำแนะนำที่เป็นประโยชน์ และเพื่อนๆที่ช่วยเหลือเรื่องเล็กๆน้อยๆแต่แสนสำคัญ รวมทั้งทุกคนที่ไม่ได้กล่าวถึงในที่นี้ด้วย

ผู้จัดทำรู้สึกซาบซึ้งในความช่วยเหลือทั้งหมดที่กล่าวนี้อย่างยิ่ง

ผู้จัดทำ

นายกิตติ โรจนภิญญู 42010018

นายเกรียงไกร ลักษณะบัณฑิต 42010028

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รายงานเรื่อง	รถหลบหลีกสิ่งกีดขวางอัตโนมัติ	
	AUTOMATIC CAR	
จัดทำโดย	นาย กิตติ	โรจนภิญญู
	นายเกรียงไกร	ลักษณะบัณฑิต
อาจารย์ที่ปรึกษา	อ.ชินภัทร	นันทจิวากรชัย

รายงานฉบับนี้ได้ผ่านการตรวจสอบโดยอาจารย์ที่ปรึกษาแล้ว

ลงชื่อ.....อาจารย์ที่ปรึกษา

(อ.ชินภัทร นันทจิวากรชัย)

วันที่ 3 / ๒๙ / 46

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รถหลบหลีกสิ่งกีดขวางอัตโนมัติ

นายกิตติ โรจนภิญญู 42010018
นายเกรียงไกร ลักษณะบัณฑิต 42010028
อาจารย์ชินภัทร นันทจิวากรชัย (อาจารย์ที่ปรึกษา)
ภาคการศึกษาที่ 2 ปีการศึกษา 2545

บทคัดย่อ

รถหลบหลีกสิ่งกีดขวางอัตโนมัติสามารถเคลื่อนที่ไปยังเป้าหมายและหลบหลีกสิ่งกีดขวาง โดยรับข้อมูลเป้าหมายในพิกัด (x,y) ผ่านทางคีย์โทรศัพท์และแสดงผลทางโมดูลLCD ไมโครคอนโทรลเลอร์จะควบคุมการหมุนของล้อขวาและล้อซ้ายผ่านทางวงจรขับมอเตอร์และมีการนับพัลซ์จากการหมุนของล้อป้อนกลับเข้าสู่ไมโครคอนโทรลเลอร์ ระหว่างทางไปยังเป้าหมาย รถหลบหลีกสิ่งกีดขวางอัตโนมัติจะตรวจจับวัตถุที่กีดขวางการเคลื่อนที่โดยใช้เซ็นเซอร์อินฟราเรด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Automatic Car

Mr. Kitti Rojjanapinyoo 42010018

Mr. KriangKrai Laksanabandit 42010028

Mr. Chinnapat Nantajiwakornchai (Advisor)

2 nd Semester , Education Year 2002

Abtract

Automatic car can move to the destination point and avoid hitting the obstacle. By enter the destination point in Cartesian space via key pad and LCD display. The microcontroller will control the rotating of left and right wheel via the motor driver circuit which feedback the rotation count . By counting the rotation , microcontroller can recognize the current position . On the way to destination , an automatic car will detect the obstacle using infrared sensor.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ

	หน้า
กิตติกรรมประกาศ	I
บทคัดย่อ	II
Abstract	III
สารบัญ	IV
บทที่ 1 บทนำ	1
บทที่ 2 ระบบขับเคลื่อน	2
2.1 หลักการทำงานของ DC MOTOR	2
2.2 การใช้งาน D/A Converter	3
2.3 Pulse Width Modulation(PWM)	4
2.3.1 การทำงานของ PWM	5
2.3.2 การเลือกความถี่การสวิตซ์	6
2.4 หลักการออกแบบและการสร้างระบบขับเคลื่อน	7
บทที่ 3 ส่วนควบคุม	9
3.1 โครงสร้างและการทำงานของพอร์ต	11
3.2 การใช้งานเป็นพอร์ตอินพุต	14
3.3 การใช้งานเป็นพอร์ตเอาท์พุต	14
3.4 การอ่านค่าลอจิกจากพอร์ต	14
3.5 การใช้งาน 8255 กับ 8051	15
3.5.1 รายละเอียดการจัดเรียงขาของ8255	16
3.6 รายละเอียดเกี่ยวกับโมดูล LCD	18
3.6.1 โครงสร้างภายในโมดูล LCD	18
3.6.2 โมดูล LCD ขนาด 16 ตัวอักษร 1 บรรทัด(LCD 16*1)	20
3.6.3 คำสั่งควบคุม LCD โมดูล	20
3.6.4 การเขียนคำสั่งและข้อมูลให้แก่โมดูล LCD	23
3.6.5 จังหวะการทำงานของโมดูล LCD	24
3.7 การเชื่อมต่อกับคีร์แพดหรือสวิตซ์เมตริก 4*3จุด	25
บทที่ 4 ส่วนเซ็นเซอร์	29
4.1 ภาคส่งอินฟราเรด	29

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ

	หน้า
4.2 ภาครับอินฟราเรด	29
4.3 เฟสล็อคลูป	29
4.4 แหล่งจ่ายพลังงาน	31
บทที่ 5 โปรแกรมควบคุมการทำงาน	34
5.1 โปรแกรมหลักควบคุมการทำงานของรถหลบหลีกสิ่งกีดขวางอัตโนมัติ	34
5.2 โปรแกรมควบคุมให้รถวิ่งไปยังเป้าหมาย	34
5.3 โปรแกรมวัดระยะทาง	37
5.4 โปรแกรมควบคุมการเลี้ยว	37
บทที่ 6 ผลการทดลอง	39
ภาคผนวก	46
เอกสารอ้างอิง	73

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่1
บทนำ

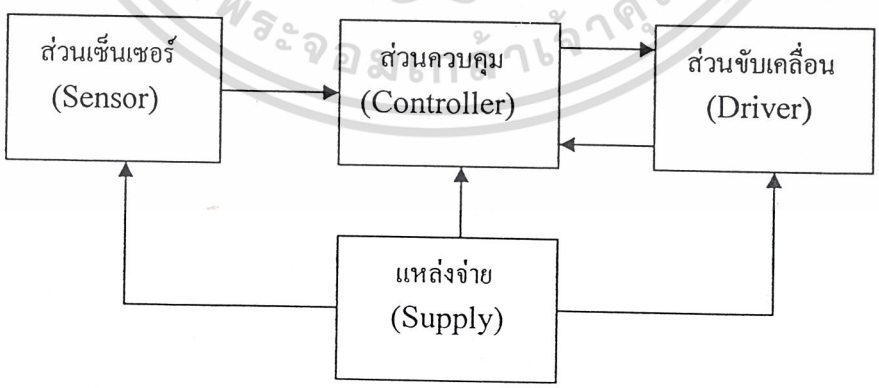
หุ่นยนต์เป็นสิ่งประดิษฐ์ที่มนุษย์สร้างขึ้นซึ่งมีด้วยกันหลายรูปแบบแล้วแต่ลักษณะการใช้งานในปัจจุบันมนุษย์ได้นำหุ่นยนต์มาใช้งานในด้านต่างๆมากมาย เช่น ในโรงงานอุตสาหกรรมมีการใช้หุ่นยนต์มาช่วยในการประกอบชิ้นส่วนผลิตภัณฑ์ ช่วยในการบรรจุภัณฑ์ ในด้านการสำรวจมีการใช้หุ่นยนต์เข้ามาช่วยสำรวจในพื้นที่ที่มนุษย์ไม่สามารถเข้าถึงได้หรือพื้นที่ที่มีการเสี่ยงภัยสูง ในด้านการแพทย์ มีการใช้หุ่นยนต์ช่วยในการผ่าตัด เป็นต้น

รถหลบหลีกสิ่งกีดขวางอัตโนมัตินี้ถือว่าเป็นหุ่นยนต์แบบหนึ่งที่สามารถหลบหลีกวัตถุที่กีดขวางการเคลื่อนที่โดยไม่มีการชนเกิดขึ้น โครงงานนี้เป็นเพียงแบบจำลองยังไม่สามารถนำไปใช้งานได้จริง แต่สามารถนำแนวคิดไปประยุกต์ใช้กับหุ่นยนต์ที่มีขนาดใหญ่กว่านี้เพื่อใช้งานได้จริง เช่น ทำเป็นหุ่นยนต์ส่งของ เป็นต้น

รถหลบหลีกสิ่งกีดขวางอัตโนมัติประกอบด้วย 4 ส่วนใหญ่ๆได้แก่

1. ส่วนขับเคลื่อน(Driver)
2. ส่วนเซ็นเซอร์(Sensor)
3. ส่วนควบคุม (Controller)
4. ส่วนแหล่งจ่ายพลังงาน(Supply)

ทั้ง 4 ส่วนจะประกอบกันดังบล็อกไดอะแกรมรูป 1.1 โดยใช้โปรแกรมภาษาแอสเซมบลี้ควบคุมการทำงาน รายละเอียดของแต่ละส่วนจะกล่าวถึงในบทต่อไป



รูปที่ 1.1 แสดงบล็อกไดอะแกรมแสดงส่วนประกอบของรถหลบหลีกสิ่งกีดขวางอัตโนมัติ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 2

ระบบขับเคลื่อน(Driving system)

ระบบขับเคลื่อนเป็นระบบที่สำคัญของโครงงานนี้โดยจะใช้ DC Motor ต้องมีการควบคุมการทำงาน ความเร็วและทิศทางการหมุน ทั้งนี้โดยได้รับข้อมูลจากเซ็นเซอร์

2.1 หลักการทำงานของ DC Motor

DC Motor เป็นทรานดิวเซอร์แรงบิด โดยมีคุณสมบัติคือ แรงบิดของเพลาจะเป็นสัดส่วนโดยตรงของกระแสอาร์เมเจอร์ และเส้นแรงแม่เหล็กดังรูป 1

สามารถเขียนเป็นสูตรง่ายๆได้ดังนี้

$$T = K\emptyset I$$

เมื่อ T คือ แรงบิดของเพลา (N-m)

$\emptyset$ คือ เส้นแรงแม่เหล็ก (Weber)

I คือ กระแสอาร์เมเจอร์ (A)

เมื่อขดลวดเคลื่อนที่ในสนามแม่เหล็กทำให้เกิดแรงดันตกคร่อมตัวมันเอง ซึ่งแรงดันนี้เป็นสัดส่วนโดยตรงกันกับความเร็วมอเตอร์

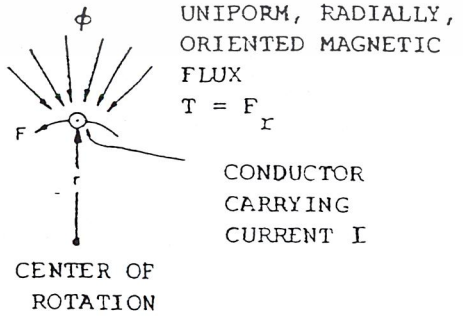
สามารถเขียนเป็นสูตรง่ายๆได้ดังนี้

$$E = K\emptyset w$$

เมื่อ E คือ แรงดันย้อนกลับ (V)

$\emptyset$ คือ เส้นแรงแม่เหล็ก (Weber)

w คือ ความเร็วมอเตอร์ (Radian/s)

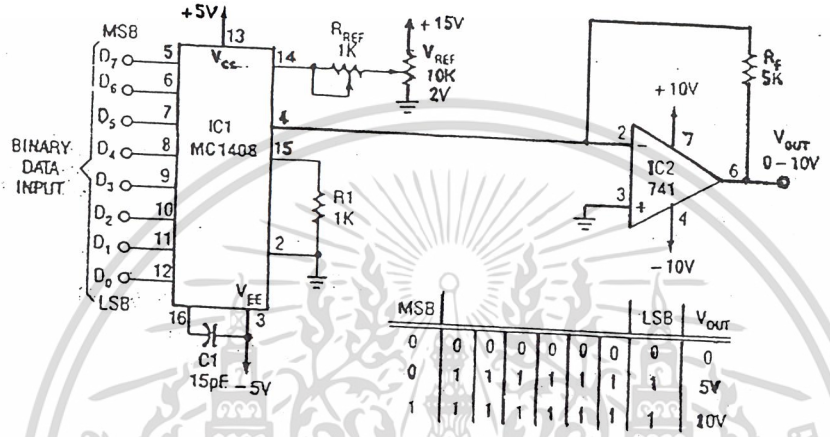


รูปที่ 2.1แสดงการเกิดแรงบิดของ DC Motor

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.2 การใช้งาน D/A Converter (DAC)

2.2.1 การประยุกต์เบื้องต้น



รูปที่ 2.2 วงจร Unipolar D/A

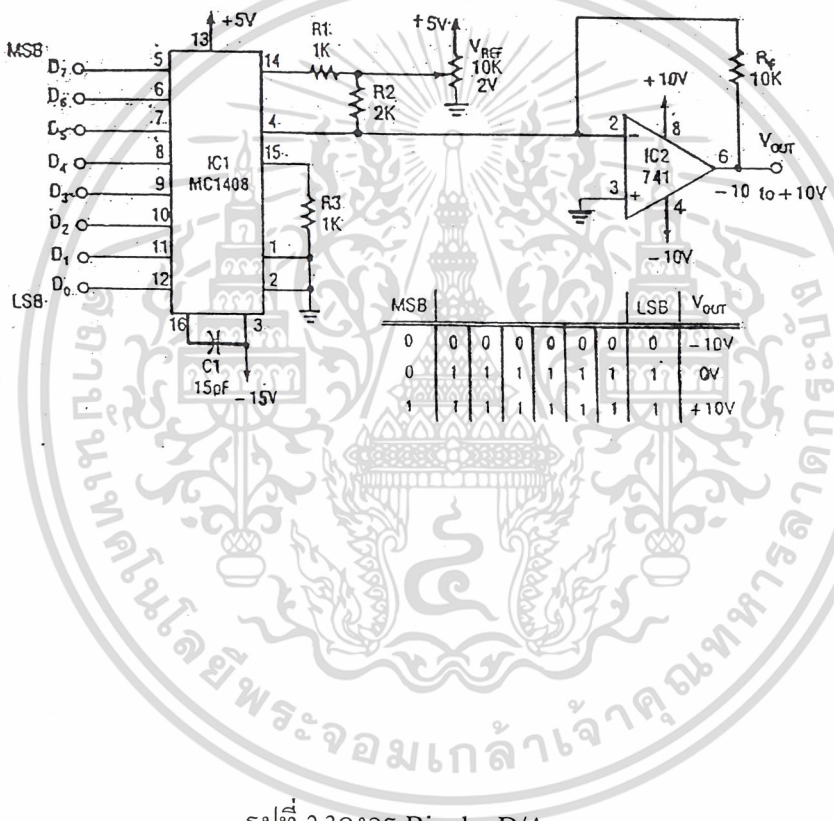
รูป 2 เป็นการประยุกต์เบื้องต้น โดยใช้ Unipolar D/A ของบริษัทโมโตโรล่า เบอร์ MC1408 โดยระดับสัญญาณเป็น TTL แบบ 8 บิต เป็นรหัสเลขฐาน 2 ถูกป้อนให้อินพุทขา D0 – D7 ภายใน MC1408 ใช้เทคนิคไบนารีแลดเดอร์ในการสวิชต์หรือแปลงสัญญาณให้กลายเป็นกระแสไฟฟ้า ออกไปทางเอาท์พุทของ LM741 ซึ่งเป็นไอซีออปแอมป์แบบ transresistance (แปลงจากกระแสเป็นแรงดัน) ให้ได้แรงดันตามต้องการ

2.2.2 การประยุกต์เบื้องต้น โดยใช้ Bipolar D/A

จากวงจรที่ผ่านมาเป็นวงจรแบบ Unipolar D/A ซึ่งหมายถึง เอาท์พุทวงจรเป็นได้แค่บวกหรือลบอย่างใดอย่างหนึ่งเท่านั้น ดังนั้นถ้าต้องการให้แปลงได้ตั้งแต่บวกถึงลบ ต้องใช้วงจร Bipolar D/A แทนตามรูป 3

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ซึ่งการต่ออินพุทของออปแอมป์เข้ากับความต้านทานอ้างอิง R2 เป็นการทำให้เอาท์พุทภาคขยายสวิงได้ระหว่าง -10 V ถึง +10V เมื่อเลขฐาน 2 ทางอินพุทเพิ่มขึ้นถึง 7FH สัญญาณอนาลอกทางเอาท์พุทจะเพิ่มขึ้นเป็น 0 V และถ้าเมื่อเลขฐาน 2 ทางอินพุทเพิ่มขึ้นถึง FFH สัญญาณอนาลอกทางเอาท์พุทจะเพิ่มขึ้นถึง +10V



รูปที่ 2.3วงจร Bipolar D/A

2.3Pulse Width Modulation Amplifier (PWM)

เป็นระบบที่ใช้ควบคุมความเร็วมอเตอร์โดยทั่วไปได้แก่ ระบบDC Linear Servo Amplifier สร้างขึ้นเพื่อเป็นอุปกรณ์ขยายแบบ Linear มีหน้าที่ควบคุมกระแสและแรงดันที่ใช้ในการขับมอเตอร์เพื่อให้ได้ความเร็วตามต้องการ และเนื่องจากการควบคุมนี้ทำโดยการบังคับขนาดของแรงดันที่จ่ายให้กับมอเตอร์ ดังนั้น Amplifier จะเป็นตัวลดแรงดันเอาไว้ ซึ่งเท่ากับผลต่างของแหล่งจ่ายแรงดันและแรงดันที่มอเตอร์ใช้จริง

เอกสารนี้สงวนไว้เพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ในระบบที่มีกระแสเหมาะสม กำลังงานจะสูญเสียเป็นจำนวนมาก โดยเฉพาะที่ความเร็วต่ำแรงบิดสูง

วิธีที่จะแก้ปัญหาการสูญเสียกำลังงาน คือ ใช้ Amplifier ควบคุมมอเตอร์โดยการเปลี่ยนแปลง Duty Cycle ของแรงดันที่จ่ายให้มอเตอร์ เรียกว่า Switching Amplifier lสามารถควบคุมความเร็วต่ำแรงบิดสูง โดยไม่เปลืองพลังงานมากนัก

2.3.1 การทำงานของ Pulse Width Modulation Amplifier (PWM)

Amplifier แบบ PWM สามารถแบ่งได้เป็น 3 ชนิดตามลักษณะการทำงาน คือ ไบโพล่าร์ ยูนิโพล่าร์ ลิมิตยูนิโพล่าร์ พิจารณาง่ายๆตามรูป 4 โดยเราจะกำหนดให้ความถี่สวิตช์เป็น fs โดย

Ton เมื่อ $0 < t < tr$

Toff เมื่อ $0 < t < tf$

ไบโพล่าร์จะมี T1 ,T4 นำกระแสระหว่างเฟส on ส่วน T2 , T3 นำกระแสระหว่างเฟส off จะได้แรงดันตกคร่อมมอเตอร์ดังนี้

$$Vm = Vab = \begin{cases} Vs & \text{เมื่อ } 0 < t < tr \\ -Vs & \text{เมื่อ } 0 < t < tf \end{cases}$$

แบบยูนิโพล่าร์ จะลดจำนวนทรานซิสเตอร์ลง การสวิตช์ขึ้นกับ Vin เป็นบวกหรือลบ เมื่อ Vin เป็นบวก T4 จะนำกระแสตลอดคาบ ขณะที่ T1 นำกระแสช่วงเฟส on T2 นำกระแสช่วงเฟส off ส่วนเมื่อVin เป็นลบ T2 จะนำกระแสตลอดคาบ ขณะที่ T3 T4 จะสลับกันทำงาน

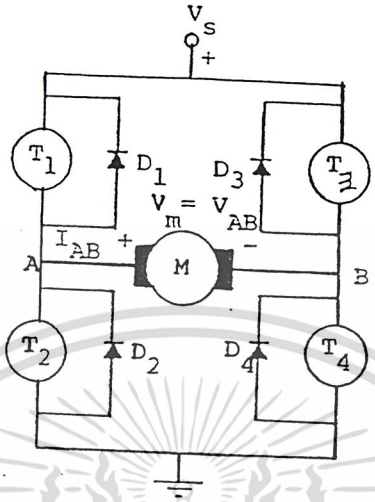
ถ้า Vin เป็นบวกจะได้

$$Vin = \begin{cases} Vs & \text{เมื่อ } 0 < t < tr \\ 0 & \text{เมื่อ } 0 < t < tf \end{cases}$$

ถ้า Vin เป็นลบจะได้

$$-Vin = \begin{cases} Vs & \text{เมื่อ } 0 < t < tr \\ 0 & \text{เมื่อ } 0 < t < tf \end{cases}$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.4 Amplifier แบบ PWM และ DC Motor

จากลักษณะที่กล่าว เห็นได้ว่าทรานซิสเตอร์คู่หนึ่งจะทำงาน แต่ทรานซิสเตอร์อีกคู่หนึ่งจะไม่ทำงานเพราะฉะนั้นถ้าทรานซิสเตอร์ทั้ง 4 ตัวทำงานพร้อมกันหมดจะเกิดการลัดวงจรขึ้น ต้องหลีกเลี่ยงภาวะดังกล่าว ทำได้โดยการสร้างช่วงดีเลย์ (delay time) ระหว่างการหยุดและการนำกระแสของทรานซิสเตอร์ ด้วยเหตุผลดังกล่าวความถี่สวิตช์จะถูกจำกัดในวงที่แคบลง

ลิมิตยูนิโพล่าร์ จำเป็นต้องสร้างช่วงดีเลย์ (delay time) ลง การสวิตช์ขึ้นกับ Vin เป็นบวกหรือลบ เมื่อVin เป็นบวก T4 จะนำกระแสตลอดคาบ ขณะที่ T1 นำกระแสช่วงเฟส on เพราะฉะนั้นในช่วงเฟส on T1 T4 จะ on

$$Vm = Vab \quad \text{เมื่อ} \quad 0 < t < tr$$

2.3.2 การเลือกความถี่การสวิตช์

การเลือกความถี่การสวิตช์ สามารถกระทำได้อย่างอิสระ คือ

- ความถี่การสวิตช์ต้องมีค่าสูงพอ ที่ทำให้ค่า L ของมอเตอร์มีค่าอิมพีแดนซ์สูงกว่าความถี่ที่ถูกเลือก ดังนั้น

$$2TfsL \gg R$$

- อัตราการสวิตช์ต้องมีค่าสูงพอ ต่อการที่ระบบเซอร์โวไม่มีผลตอบสนองต่ออัตราดังกล่าว ดังนั้นต้องกำหนด fBW คือ

$$fs \geq 10\, fBW$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- fs ต้องมากกว่าความถี่เรโซแนนซ์

$$fs > (f\ res)\ max$$

- การเพิ่มอัตราการสวิตซ์ เป็นผลทำให้ค่ากระแสสูญเสียกำลังงานมากขึ้นในทรานซิสเตอร์ด้วยเวลาของการสวิตซ์ค่าหนึ่ง เพราะฉะนั้น fs จึงไม่อาจที่จะสูงมากได้

2.4หลักการออกแบบและการสร้างระบบขับเคลื่อน

ในระบบขับเคลื่อนเลือกใช้ DC Motor แบบเส้นแรงแม่เหล็กมีค่าคงที่ และใช้การควบคุมแบบ PWM การออกแบบสามารถแบ่งได้ คือ ส่วนการควบคุม DC Motor

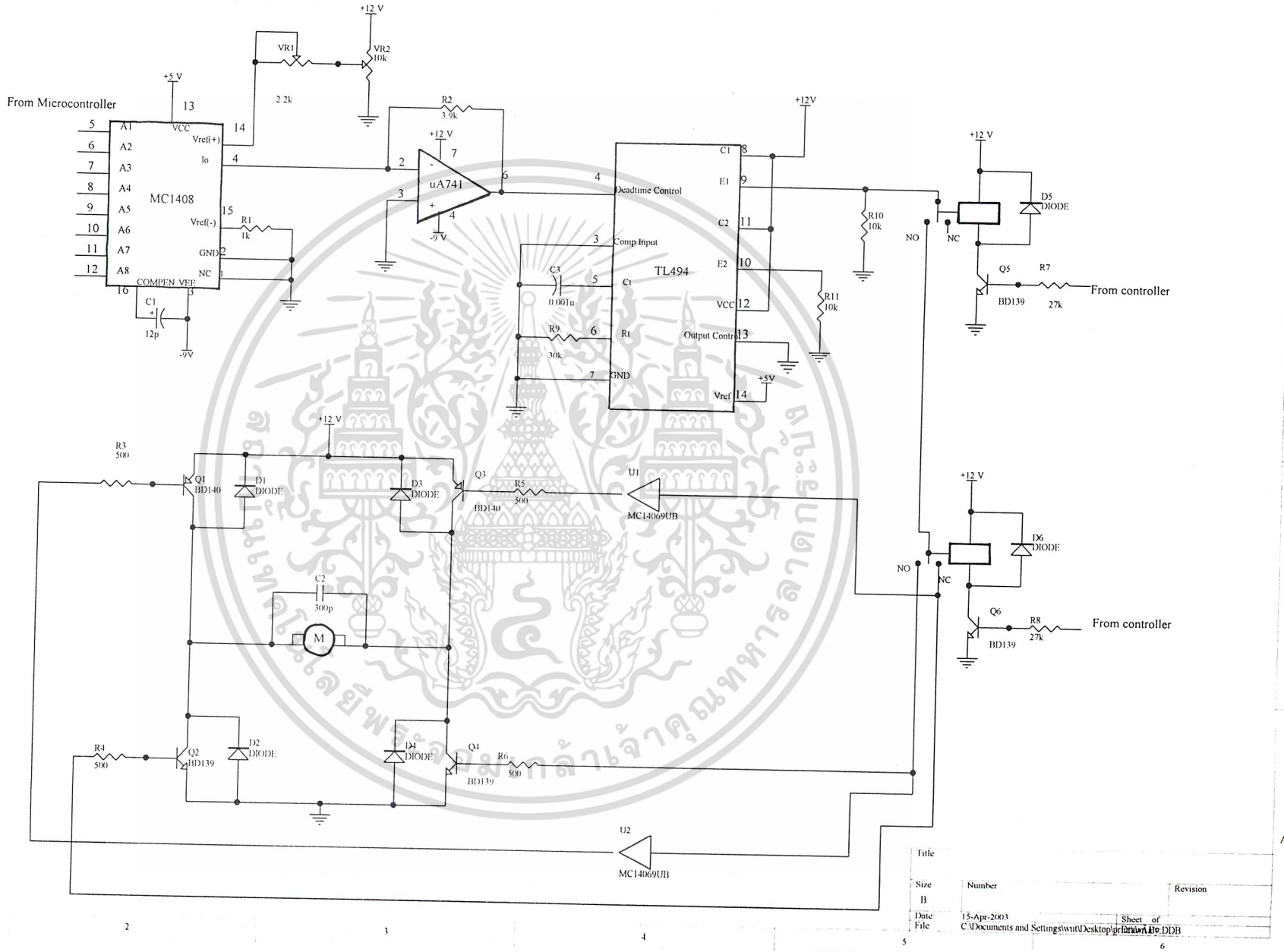
มีหลักการทำงานดังนี้ส่วน controller จะส่งข้อมูลความเร็วออกมาจากไอโอพอร์ต ซึ่เป็นข้อมูลทางดิจิตอลขนาด 8 บิต เข้าสู่ D/A เบอร์ MC1408 ซึ่งจะแปลงข้อมูลความเร็วทางดิจิตอลไปเป็นข้อมูลความเร็วทางอนาลอกออกทางขา 6 ของLM741 ซึ่งเป็นไอซีออปแอมป์

ข้อมูลความเร็วทางอนาลอกจะถูกนำไปต่อที่ขา 4 ของ TL494 โดยถ้ารับแรงดันค่าต่างๆที่ขานี้จะทำให้เอาท์พุทออกมาเป็นพัลซ์ที่ขา 11 ซึ่งความกว้างของพัลซ์จะขึ้นกับระดับสัญญาณแรงดันนั้นๆ

เอาท์พุทออกมาจาก TL494 จะต่อเข้ากับวงจรบริดจ์ ขยายสัญญาณควบคุมมอเตอร์โดยผ่านขาคอมมอนของรีเลย์เพื่อควบคุมการเปิดปิดของ DC Motor ซึ่งใช้ข้อมูลออกมาจากไอโอพอร์ตของคอนโทรลเลอร์

สำหรับทิศทางการหมุนนั้นใช้รีเลย์เป็นตัวสวิตช์ให้หมุนด้านหน้าหรือด้านหลังซึ่งใช้ข้อมูลออกมาจากไอโอพอร์ตของคอนโทรลเลอร์เช่นเดียวกัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

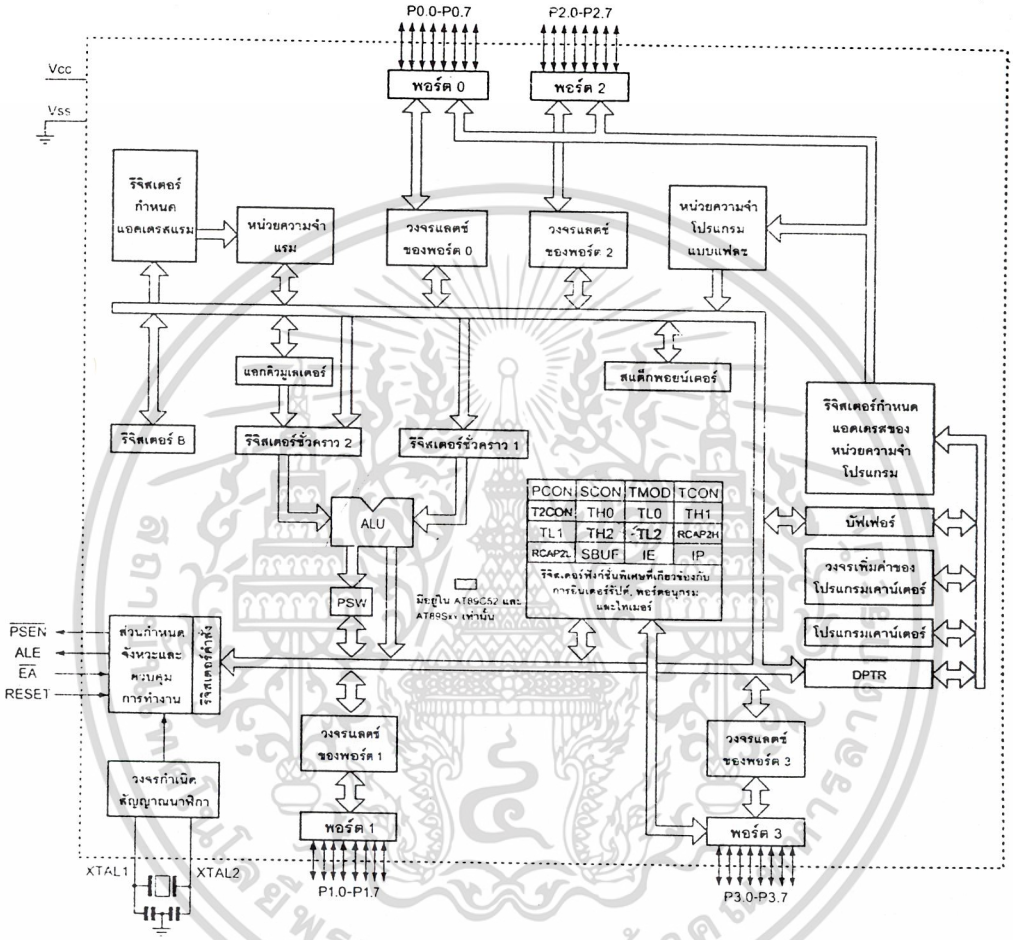


รูปที่ 2.5 รูปวงจรส่วนขับมอเตอร์

บทที่ 3

ส่วนควบคุม

ส่วนควบคุมของรถหลบหลีกสิ่งกีดขวางอัตโนมัติจะใช้ไมโครคอนโทรลเลอร์ MCS 51 ซึ่งมีข้อมูลดังนี้



รูปที่ 3.1 รายละเอียดโครงสร้างหลักของไมโครคอนโทรลเลอร์ MCS 51

ขา Vcc ใช้สำหรับต่อไฟเลี้ยง

ขาGND เป็นสายดิน

ขาพอร์ต 0 (P0.0-P0.7) แต่ละขาสามารถเป็นได้ทั้งอินพุทและเอาท์พุทสำหรับใช้งานทั่วไป ถ้าต้องการให้ขาพอร์ต 0 ขาใดขาหนึ่งเป็นอินพุททำได้โดยเขียนข้อมูล 1 ไปยังแต่ละพอร์ตที่ต้องการติดต่อด้วย ทำให้ขามีสถานะปล่อยลอย(float) นอกจากนั้นยังใช้งานในการติดต่อกับขาแอดเดรสไบต์ต่ำของหน่วยความจำภายนอก (A0-A7) และขาข้อมูล (D0-D7) โดยใช้กระบวนการ multiplex เข้าช่วย เพื่อสลับการทำงานให้เป็นได้ทั้งขาติดต่อกับขาแอดเดรสไบต์ และขาข้อมูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ขาพอร์ต 1 (P1.0-P1.7) แต่ละขาสามารถเป็นได้ทั้งอินพุทและเอาท์พุทสำหรับใช้งานทั่วไป ถ้าต้องการให้ขาพอร์ต 1 ขาใดขาหนึ่งเป็นอินพุททำได้โดยเขียนข้อมูล 1 ไปยังแต่ละพอร์ตที่ต้องการติดต่อด้วย นอกจากนั้นในอนุกรม AT89SXX ยังใช้ขา P1.0 เป็นขาอินพุทสำหรับนับค่าของไทเมอร์ 2 (T2) และ P1.1 เป็นขาอินพุท ทริกเกอร์ของไทเมอร์ 2 (T2) ส่วนขา P1.4-P1.7 เป็นขาสำหรับเชื่อมต่อแบบ SPI เพื่อทำงานโปรแกรมข้อมูลในระบบ

ขาพอร์ต 2 (P2.0-P2.7) แต่ละขาสามารถเป็นได้ทั้งอินพุทและเอาท์พุทสำหรับใช้งานทั่วไป ถ้าต้องการให้ขาพอร์ต 2 ขาใดขาหนึ่งเป็นอินพุททำได้โดยเขียนข้อมูล 1 ไปยังแต่ละพอร์ตที่ต้องการติดต่อด้วยทำให้ขามีสถานะปล่อยลอย(float) นอกจากนั้นยังใช้งานในการติดต่อกับขาแอดเดรสไบต์สูงของหน่วยความจำภายนอก (A8-A15)

ขาพอร์ต 3 (P3.0-P3.7) แต่ละขาสามารถเป็นได้ทั้งอินพุทและเอาท์พุทสำหรับใช้งานทั่วไป ถ้าต้องการให้ขาพอร์ต 3 ขาใดขาหนึ่งเป็นอินพุททำได้โดยเขียนข้อมูล 1 ไปยังแต่ละพอร์ตที่ต้องการติดต่อด้วยทำให้ขามีสถานะปล่อยลอย(float)) นอกจากนั้นยังใช้งานในหน้าที่พิเศษดังนี้

P3.0 สามารถเป็นขาอินพุทสำหรับรับข้อมูลจากการสื่อสารแบบอนุกรม หรือขา R X D

P3.1 สามารถเป็นขาอินพุทสำหรับส่งข้อมูลจากการสื่อสารแบบอนุกรม หรือขา T X D

P3.2 เป็นขาอินพุทสำหรับรับสัญญาณอินเตอร์รัปต์จากภายนอกช่องที่ 0หรือขา INT0

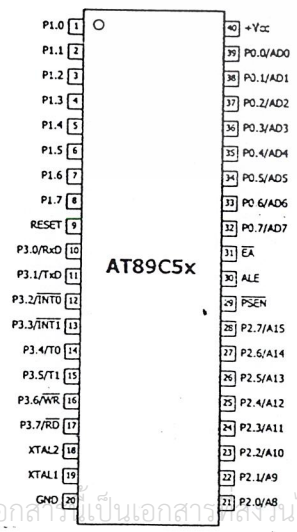
P3.2เป็นขาอินพุทสำหรับรับสัญญาณอินเตอร์รัปต์จากภายนอกช่องที่ 1หรือขา INT1

P3.3 สามารถเป็นขาอินพุทสำหรับรับสัญญาณไทเมอร์จากภายนอกช่องที่ 0หรือขา TO

P3.4 สามารถเป็นขาอินพุทสำหรับรับสัญญาณไทเมอร์จากภายนอกช่องที่ 1หรือขา T1

P3.5 สามารถเป็นขาสัญญาณ WR ในกรณีที่ใช้เชื่อมต่อกับหน่วยความจำภายนอก

P3.6 สามารถเป็นขาสัญญาณ RD ในกรณีที่ใช้เชื่อมต่อกับหน่วยความจำภายนอก



รูปที่ 3.2 การจัดขามาตรฐานของMCS51

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ขารีเซต ใช้ในการรีเซตการทำงาน โดยการป้อนสัญญาณ 1 อย่างน้อย 2 แมชชีนไซเกิล โดยที่ clk ยังทำงานเป็นปกติ

ขา ALE/PROG (Address Latch Enable / Program Pulse Input) เป็นขาที่ใช้ควบคุมการแลตซ์ของขาพอร์ต 0 ในกรณีที่ใช้งานกับหน่วยความจำภายนอก นอกจากนั้นยังเป็นขารับ Pulse ของการโปรแกรมข้อมูลลงใน MCS-51 รุ่นที่มีหน่วยความจำแบบ EPROM

ขา PSEN (Program Store Ensble) ใช้ในการส่งสัญญาณเพื่อร้องขอติดต่อกับหน่วยความจำภายนอก เมื่อ MCS-51 ต้องการอ่านข้อมูลจากหน่วยความจำภายนอก โดย MCS-51 จะต้องส่งสัญญาณออกที่ขานี้ 2 ครั้งในแต่ละแมชชินไซเกิล หากต้องการอ่านข้อมูลจากหน่วยความจำภายในจะไม่มีการส่งสัญญาณใดๆออกมาเลย

ขาEA/Vpp (External Access Enable /Programming Voltage Input) ใช้ในการเลือกติดต่อหน่วยความจำภายนอกหรือหน่วยความจำภายในของ MCS-51 หากขานี้เป็น 1 เป็นการเลือกติดต่อหน่วยความจำภายนอก แต่หากขานี้เป็น 0 เป็นการเลือกติดต่อหน่วยความจำภายใน นอกจากนี้ยังเป็นขารับแรงดันไฟสูงสำหรับการโปรแกรมหน่วยความจำภายในของ MCS-51

ขา XTAL1 , XTAL2 เป็นขาสำหรับต่อขาคริสตรอลเพื่อสร้าง clk ในการกำหนดจังหวะการทำงานของ MCS-51

3.1 โครงสร้างและการทำงานของพอร์ต

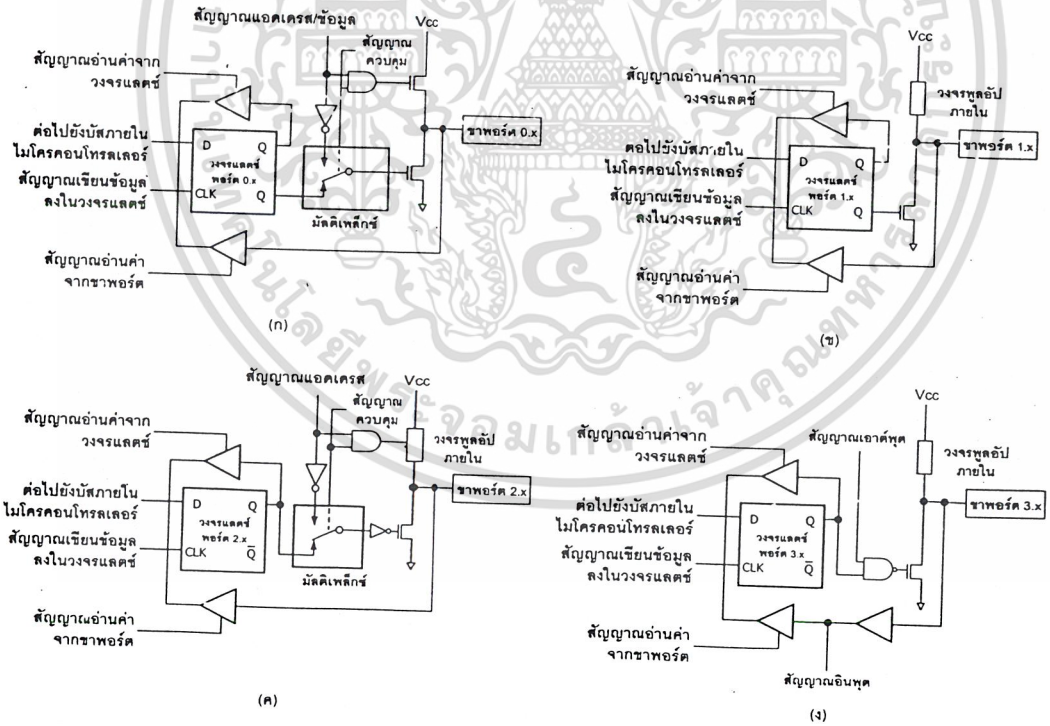
พอร์ตใช้งานมีทั้งหมด 4 พอร์ต คือ พอร์ต0 พอร์ต1 พอร์ต2 พอร์ต3 โดยแต่ละพอร์ตจะมีขนาด 8 บิตและเป็นพอร์ต 2ทิศทาง คือสามารถเป็นทั้งอินพุทและเอาท์พุท ทุกพอร์ตจะมีวงจรแลตซ์และวงจรขับ บัฟเฟอร์อินพุท

ที่พอร์ต 0 , 2 สามารถเป็นทั้งพอร์ตอินพุทและพอร์ตเอาท์พุทสำหรับใช้งานทั่วไป และใช้ติดต่อกับหน่วยความจำภายนอก สำหรับพอร์ต 3 และพอร์ต 1 บางขานอกจากจะเป็นทั้งพอร์ตอินพุทและพอร์ตเอาท์พุทแล้วยังทำงานในหน้าที่พิเศษดังตาราง

ขา	เบอร์ของไมโครคอนโทรลเลอร์	หน้าที่พิเศษ
P1.0	AT89C52/AT89SXX	ขา T2 เป็นขาอินพุทนับค่าไทเมอร์/เคาน์เตอร์ 2 และเป็นขาเอาท์พุทของแหล่งกำเนิดสัญญาณนาฬิกาโดยไทเมอร์ 2
P1.1	AT89C52/AT89SXX	ขา T2EX เป็นขาอินพุททริกเกอร์ สำหรับการแคปเจอร์/รีโหลด/ควบคุมทิศทางของ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

		สัญญาณ
P1.2	AT89SXX	ขา SS เป็นขาเลือกการติดต่อในกรณีที่ไมโครคอนโทรลเลอร์เป็นอุปกรณ์สเลฟในระบบการติดต่อแบบ SPI
P1.3	AT89SXX	ขาMOSI (Master data output) ใช้ในการติดต่อกับพอร์ต SPI
P1.6	AT89SXX	ขาMOSO (Master data input) ใช้ในการติดต่อกับพอร์ต SPI
P1.7	AT89SXX	ขา SCK (Master clock output) เป็นขาสัญญาณนาฬิกาของการติดต่อกับพอร์ต SPI



รูปที่ 3.3วงจรภายในของแต่ละพอร์ตในไมโครคอนโทรลเลอร์

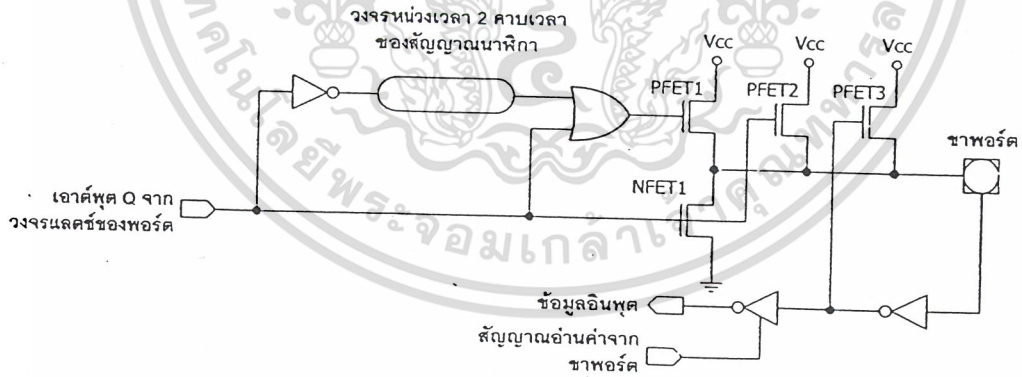
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โดยรูป ก เป็นวงจรพอร์ต 0 วงจรแลตช์แต่ละบิต คือ วงจรดีฟลิบฟลอบ การอ่านสถานะของพอร์ตและสถานะของแลตช์กระทำได้อย่างอิสระต่อกันด้วยสัญญาณที่แยกจากกัน ส่วนการเขียนข้อมูลยังพอร์ตที่ต้องการส่งสัญญาณมายังขา clk ของวงจรดีฟลิบฟลอบ โดยขณะที่ข้อมูลจะผ่านทางขาบัสของข้อมูลภายในเข้าสู่ขา D ของวงจรดีฟลิบฟลอบ

ที่พอร์ตนี้มีวงจรมัลติเพล็กซ์สำหรับกำหนดลักษณะการทำงานของพอร์ตว่าต้องการใช้งานเป็นพอร์ตอินพุทปกติหรือใช้ติดต่อกับหน่วยความจำภายนอกของไมโครคอนโทรลเลอร์

เนื่องจากขาพอร์ต 0 ไม่มีวงจรพูลอัพภายใน หากมีการนำพอร์ต 0 ไปใช้งานเป็นพอร์ตอินพุทจะต้องต่อตัวต้านทานพูลอัพภายนอกเข้าที่พอร์ต 0 ทุกขาด้วย

รูป ข เป็นวงจรของพอร์ต 1 มีลักษณะคล้ายพอร์ต 0 หากแต่ไม่มีวงจรมัลติเพล็กซ์เนื่องจากพอร์ตนี้จะไม่ใช้ติดต่อกับหน่วยความจำภายนอกแต่จะมีวงจรพูลอัพภายในที่แต่ละบิตของพอร์ตนี้แทน ส่วนรูปของวงจรพูลอัพภายในแสดงดังรูปข้างล่าง



รูปที่ 3.4 วงจรพูลอัพภายในพอร์ตของ MCS51

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูป ค เป็นวงจรภายในของพอร์ต 2 จะมีลักษณะคล้ายพอร์ต 0 มากๆ ต่างเพียงมีวงจรพูลอัพเพิ่มเติมมา

รูป ง เป็นวงจรภายในของพอร์ต 3 มีลักษณะคล้ายพอร์ต 1 มีการเพิ่มเติมวงจรบัฟเฟอร์และวงจรอินพุทเอาท์พุทเมื่อทำงานในฟังก์ชั่นพิเศษ

3.2 การใช้งานเป็นพอร์ตอินพุท

การกำหนดเป็นพอร์ตอินพุทต้องเริ่มต้นด้วยการเขียนข้อมูล 1 มาที่แต่ละบิตของพอร์ตที่ต้องการใช้งานเป็นอินพุท เพื่อหยุดการทำงานของเฟตที่ใช้ในการขับสัญญาณเอาท์พุทของบิตนั้นๆ ทำให้ขาสัญญาณของพอร์ตที่บิตนั้นๆเชื่อมต่อกับวงจรพูลอัพภายในโดยตรงส่งผลให้ขาพอร์ตนั้นๆมีลอจิก 1 สามารถรับสัญญาณลอจิก 0 จากอุปกรณ์ภายนอกได้ง่าย

สัญญาณจากข้อมูลจากอุปกรณ์ภายนอกจะถูกส่งเข้ามาเก็บไว้ในวงจรบัฟเฟอร์ภายในพอร์ต แล้วรอให้ cpu มาอ่านค่าเข้าไป เมื่อเป็นเช่นนี้อุปกรณ์ภายนอกที่เชื่อมต่อกับพอร์ตอินพุทของไมโครคอลทรอลเลอร์ ควรกำหนดให้สภาวะการทำงานที่ลอจิก 0 จะสะดวกที่สุด

3.3 การใช้งานเป็นพอร์ตเอาท์พุท

โดยปกติขาพอร์ตถูกกำหนดเป็นพอร์ตเอาท์พุทอยู่แล้วจึงส่งข้อมูลออกไปได้อย่างง่ายดาย คือ เมื่อต้องการส่งข้อมูล 0 ก็เขียนข้อมูล 0 ไปยังวงจรแลตช์ซึ่งจะส่งค่าไปขับเฟตให้ทำงาน

หากส่งข้อมูล 1 ไปยังวงจรแลตช์ซึ่งจะส่งค่าไปขับเฟตให้หยุดทำงาน ทำให้ขาพอร์ตเชื่อมต่อกับวงจรพูลอัพภายในโดยตรงส่งผลให้ขาพอร์ตนั้นๆมีลอจิก 1 ซึ่งคล้ายกับการกำหนดให้เป็นขาอินพุท แต่ต่างกันที่กระบวนการย้ายข้อมูล โดยถ้าเป็นขาอินพุทจะมีสัญญาณมาอ่านข้อมูลที่วงจรบัฟเฟอร์ ถ้าเป็นเอาท์พุทจะไม่มีสัญญาณมาอ่านข้อมูลที่วงจรบัฟเฟอร์แต่อย่างใด

3.4 การอ่านค่าลอจิกจากพอร์ต

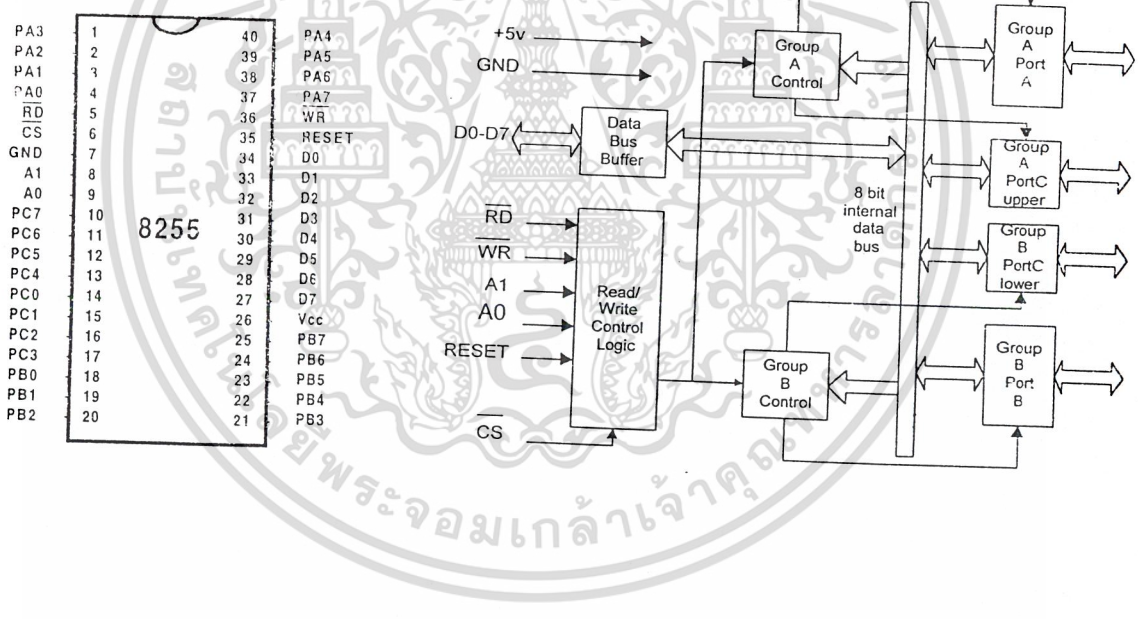
สามารถอ่านค่าลอจิกจากพอร์ตได้ 2 ลักษณะ คือ อ่านโดยจากพอร์ตโดยตรง และอ่านจากวงจรแลตช์ของแต่ละพอร์ต

ที่สำคัญการอ่านลอจิกจากพอร์ตควรเลือกวิธีการให้เหมาะสมกับอุปกรณ์ที่นำมาต่อด้วย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.5 การใช้งาน 8255 กับ 8051

ไอซี 8255 ได้รับการออกแบบมาเพื่อทำหน้าที่เป็นพอร์ต สำหรับการรับส่งข้อมูลแบบขนานระหว่างอุปกรณ์ภายนอกกับไมโครคอนโทรลเลอร์ ความอ่อนตัวในการใช้งานของ 8255 ได้แก่ การที่สามารถเปลี่ยนแปลงลักษณะการทำงานของพอร์ต ให้เป็นการเอาท์พุตหรืออินพุตได้สะดวก โดยการส่งข้อมูลควบคุมจากไมโครคอนโทรลเลอร์ ก่อนที่จะเริ่มต้นใช้งาน ความสามารถเช่นนี้เรียกว่า programable คือสามารถโปรแกรมการทำงานได้ ทำให้ได้รับความนิยมนำไปใช้งานอย่างแพร่หลาย



รูปที่ 3.5 แสดงแผนผังภายในและการจัดขาของไอซี 8255

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 3.5 แสดงบล็อกไดอะแกรมของ 8255 ซึ่งหน้าที่ของแต่ละบล็อกมีดังนี้

บล็อกกลุ่มแรกมีจำนวน 4 บล็อก อยู่ทางด้านขวาของรูป เป็นส่วนที่เชื่อมต่อกับอุปกรณ์ภายนอกอื่นๆ โดยมีสาย P0-P7,PB0-PB7 และ PC0-PC7 พอร์ต เป็นทางผ่านของข้อมูลระหว่างอุปกรณ์ภายนอกกับ 8255 สายสัญญาณเหล่านี้ถูกแบ่งเป็นสามพอร์ตได้แก่ พอร์ตA(PA) พอร์ต B(PB) และพอร์ตC(PC) พอร์ตเหล่านี้แต่ละพอร์ตสามารถเป็นได้ทั้งพอร์ตอินพุตและเอาท์พุตแต่ละบล็อกจะมีสายสัญญาณเชื่อมเข้ากับบัสข้อมูลภายในของ 8255

บล็อกกลุ่มถัดมาได้แก่ กลุ่มควบคุม A และกลุ่มควบคุม B ซึ่งจะเป็นตัวกำหนดลักษณะการทำงานของทั้งสามไอโอพอร์ต (8255 มีลักษณะการทำงานที่แตกต่างกันอยู่ 3 โหมด สามารถกำหนดได้โดยการโปรแกรมส่งคอนโทรลเวิร์ดให้กับ 8255) จากรูปจะเห็นว่าพอร์ต C นี้จะประกอบด้วยพอร์ตขนาด 4 บิต 2 พอร์ต กลุ่มหนึ่งจะถูกควบคุมโดยกลุ่มควบคุม A และอีก กลุ่มหนึ่งจะถูกควบคุมโดยกลุ่มควบคุม B

บล็อกสุดท้ายที่จะกล่าวถึงได้แก่ ดาต้าบัสบัฟเฟอร์ และลอจิกควบคุมการอ่านและการเขียน ซึ่งบล็อกเหล่านี้จะเป็นส่วนที่ติดต่อกับซีพียู ดาต้าบัสบัฟเฟอร์นี้จะเป็นบัฟเฟอร์ให้กับบัสข้อมูลของซีพียู ส่วนลอจิกควบคุมการอ่านและการเขียน จะเป็นส่วนที่ควบคุมให้ข้อมูลเข้าหรือออกจากรีจิสเตอร์ภายในตัวที่ถูกต้องและในเวลาที่เหมาะสม

3.5.1 รายละเอียดการจัดเรียงขาของ8255

ใน ส่วนนี้จะพิจารณาหน้าที่ของขาแต่ละขาของ 8255 ซึ่งข้อมูลเหล่านี้จะมีประโยชน์ในการเชื่อมต่อเข้ากับระบบบัสของซีพียู รายละเอียดของแต่ละขามีดังนี้คือ

D0-D7 : เป็นขาข้อมูลอินพุตเอาท์พุตแบบ 2 ทิศทาง (Bidirectional Bus) จะเป็นทางผ่านของข้อมูลพอร์ตต่างๆของ 8255 กับบัสข้อมูลของ 8051

CS (Chip Select Input) : เมื่อขานี้มีสถานะลอจิกเป็น 0 ซีพียูจะสามารถอ่านหรือเขียนข้อมูลกับ 8255 ได้

RD (Read Input) : เมื่อขานี้มีสถานะลอจิกเป็น 0 และขาสัญญาณ CS มีลอจิกเป็น 0 ข้อมูลจาก 8255 จะปรากฏสู่ระบบบัสข้อมูล ซีพียูก็จะสามารถอ่านข้อมูลออกไปได้

WR (Write Input) : เมื่อขานี้มีสถานะลอจิกเป็น 0 และขาสัญญาณ CS มีลอจิกเป็น 0ข้อมูลจากระบบบัสข้อมูลจะถูกเขียนเข้าไปยัง 8255 ได้

A0-A1 (Address Input) : จะเป็นตัวกำหนดการเลือกใช้รีจิสเตอร์ภายในของ 8255

RESET : เมื่อขานี้มีสถานะเป็น 1 8255 จะอยู่ในสภาวะรีเซ็ตทุกๆพอร์ตของ 8255 จะถูกเซ็ตให้อยู่ในโหมดอินพุต

PA0-PA7 , PB0-PB7 : ขาสัญญาณเหล่านี้จะถูกใช้เป็นไอโอพอร์ตขนาด 8 บิตใช้ต่อเข้ากับอุปกรณ์ภายนอกอื่นๆ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

PC0-PC7 : ขาสัญญาณเหล่านี้จะถูกใช้เป็นไอโอพอร์ตขนาด 8 บิตใช้ต่อเข้ากับอุปกรณ์ภายนอกอื่นๆเช่นเดียวกับ PA0-PA7 , PB0-PB7 แต่กลุ่มของขาสัญญาณเหล่านี้สามารถแบ่งออกเป็น 2 กลุ่มโดยแต่ละกลุ่มมีขนาด 4 บิต กลุ่มแรกจะใช้ควบคุม PB0-PB7 และกลุ่มที่สองใช้ควบคุม PA0-PA7

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

50404

3.6 รายละเอียดเกี่ยวกับโมดูล LCD

จะประกอบด้วยส่วนหลักๆ 3 ส่วน ดังนี้

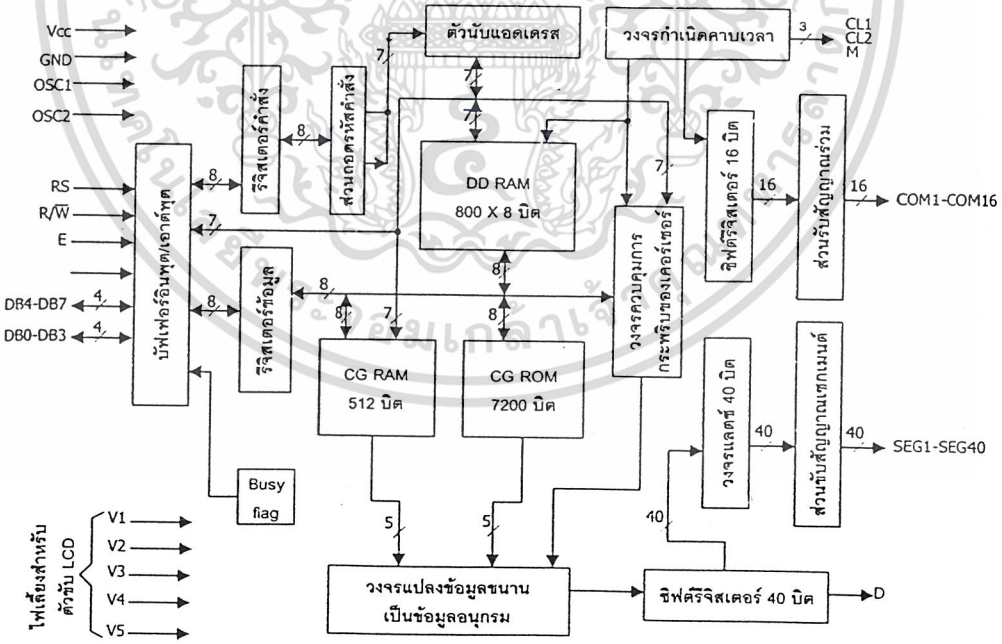
1.ตัวแสดงผล(display) ภายในเป็นผลึกเหลวแสดงผลโดยอาศัยแสงจากภายนอก ดังนั้นจึงต้องมีมุมในการมองข้อมูลที่แสดงบนจอ LCD

2.ตัวควบคุม(controller) เป็นตัวรับข้อมูลจากอุปกรณ์ภายนอกมาควบคุมการทำงานของ LCD เช่น ลบจอภาพ แสดงอักษร เป็นต้น ใช้ชิปควบคุมเฉพาะ คือ เบอร์ 44780 ควบคุม LCD แบบอักขระ ส่วน HD61830 ควบคุม LCD แบบกราฟฟิก

3.ตัวขับ(driver) รับสัญญาณจากตัวควบคุมมาขับตัวแสดงผลให้แสดงข้อมูลตามกำหนด ใช้ชิปควบคุม คือ เบอร์ HD 44100H MSM5259

3.6.1 โครงสร้างภายในโมดูล LCD

การใช้งานโมดูล LCD ต้องเข้าใจโครงสร้างและคำสั่งที่ใช้ควบคุมที่นี้ยกตัวอย่างของโมดูล LCD แบบอักขระ โดยในรูป เป็นเบอร์ 44780 ประกอบด้วย



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 3.6 ไดอะแกรมการทำงานของโมดูล LCD แบบอักขระ

บัฟเฟอร์อินพุทเอาทพุท ใช้ติดต่อรับส่งข้อมูลกับอุปกรณ์ภายนอก เพื่อจะถ่ายทอดข้อมูลเข้าออกภายในตัวควบคุม

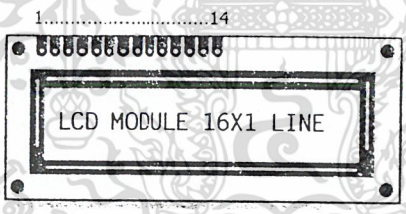
รีจิสเตอร์คำสั่ง(Instruction Register : IR) เป็นรีจิสเตอร์ใช้รับข้อมูลคำสั่งจากอุปกรณ์ภายนอก เพื่อไปควบคุมตัวแสดงผล

รีจิสเตอร์ข้อมูล(Data Register : DR) เป็นรีจิสเตอร์ใช้รับข้อมูลจากอุปกรณ์ภายนอกเพื่อถ่ายทอดไปยังหน่วยความจำที่ทำหน้าที่เก็บข้อมูลแสดงผล หรือนำข้อมูลไปสร้างตัวอักษรเพิ่มเติมในแรมเก็บตัวอักษร

แรมเก็บข้อมูลการแสดงผล(Display Data Ram :DDRAM) ทำหน้าที่เก็บข้อมูลจากรีจิสเตอร์ DR ตัวควบคุมจะนำข้อมูลใน DDRAM นี้ไปเปิดตาราง (look up-lable) ของตัวอักษรที่เก็บในหน่วยความจำรอมและแรมเก็บตัวอักษร เพื่อนำไปแสดงที่ตัวแสดงผล

รอมเก็บตัวอักษร (Character Generator ROM : CGROM) ใช้เก็บข้อมูลตัวอักษร หรือสัญลักษณ์ที่อ่านออกไปแสดงที่ตัวแสดงผลได้ ถูกอ่านด้วยค่า DDRAM

แรมเก็บตัวอักษร (Character Generator RAM : CGRAM) ใช้เก็บข้อมูลตัวอักษรที่มีการสร้างเพิ่มเติมในกรณีตัวอักษรในDDRAM ไม่เพียงพอ การอ่านทำเช่นเดียวกับ DDRAM



ขา 1 : GND
ขา 2 : +V
ขา 3 : Brightness ปรับความสว่าง
ขา 4 : RS
ขา 5 : R/W
ขา 6 : E
ขา 7-14 : D0-D7

รูปที่ 3.7 รูปร่างและการจัดขาของโมดูล LCD แบบอักขระ

แฟลก busy แจ้งสถานะการทำงานของตัวควบคุมให้อุปกรณ์ภายนอกทราบว่า ตัวรับข้อมูลพร้อมรับคำสั่งหรือข้อมูลหรือไม่ ดังนั้นก่อนส่งข้อมูลต้องตรวจสอบสถานะแฟลก busy ว่าว่างหรือไม่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.6.2 โมดูล LCD ขนาด 16 ตัวอักษร 1 บรรทัด(LCD 16*1)

ที่ยกโมดูลตัวนี้เพราะถูก หาง่าย มีโครงสร้างที่มีมาตราฐาน มีผู้ผลิตหลายบริษัท ซึ่งมีระบุเบอร์ที่แตกต่างกัน ที่นี้ใช้เบอร์ HD44750 ของ HITASHI

โมดูล LCD ขนาด 16 ตัวอักษร 1 บรรทัด มีขาทั้งหมด 14ขา มีการจัดขาตามรูป สำหรับรายละเอียดของแต่ละขามีดังนี้

Vss (ขา1): ต่อกราวน์

Vdd(ขา2): ต่อไฟเลี้ยง +5 โวลต์

Vo(ขา3): เป็นขาอินพุท รับแรงดันเพื่อปรับความเข้มการแสดงผล

RS(ขา4): เป็นขาอินพุทใช้แยกข้อมูลที่ใช้ในการประมวลผลในขณะนั้นว่าเป็นคำสั่งของริจิสเตอร์ IR หรือ DR ถ้าขานี้เป็น 0 ข้อมูลที่ส่งมาจะเป็นคำสั่ง แต่ถ้าขานี้เป็น 1 จะเป็นข้อมูลสำหรับการแสดงผล

R/W(ขา5): เป็นขาที่ใช้เลือกการอ่านหรือเขียนข้อมูลกับ LCD ถ้าเป็น 0 เป็นการกำหนดให้เขียนข้อมูล แต่ถ้าเป็น 1 จะเป็นการอ่านข้อมูล

E(ขา6): เป็นขา enable LCD ให้ทำงาน

D0-D7(ขา7-14): ใช้เป็นทางผ่านระหว่าง LCD กับอุปกรณ์ภายนอกขนาด 8 บิต

3.6.3 คำสั่งควบคุม LCD โมดูล

การเขียนคำสั่งลงในตัวควบคุมต้องกำหนดขา RS และ R/W เป็น 0 แล้วเขียนคำสั่งตามไป คำสั่งที่สำคัญมี 10 คำสั่ง ดังนี้

1.คำสั่งเคลียร์การแสดงผล (Clear Display)

มีข้อมูลคำสั่งเป็น 01H เป็นคำสั่งที่ใช้เขียนข้อมูลช่องว่าง(Space) เข้าไปในDDRAM ทั้งหมด เมื่อตัวควบคุมเอ็กซิคิวส์คำสั่งนี้ จะกำหนดแอสแดรสของ DDRAM เป็น 0 เคอร์เซอร์เลื่อนมาที่ซ้ายสุดของจอภาพ แล้วเซตบิต I/D ให้เป็น 1

2.คำสั่ง Return Home

ต้องกำหนดให้บิต 1 เป็น "1" เป็นคำสั่งเลื่อนให้เคอร์เซอร์ไปด้านซ้ายสุดของจอแสดงผล แต่ข้อมูลบนจอแสดงผลไม่เปลี่ยนแปลง นั่นคือ ข้อมูลของคำสั่งนี้จะเป็น 02H หรือ 03H ก็ได้

3.คำสั่งเลือกโหมดการป้อนข้อมูล(enter mode set)

มีรายละเอียดดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บิต7	บิต6	บิต5	บิต4	บิต3	บิต2	บิต1	บิต0
0	0	0	0	0	1	I/D	S

บิต S เป็นบิตกำหนดลักษณะการแสดงผล เมื่อมีการป้อนข้อมูล หากบิต S เป็น 1 เมื่อเกิดข้อมูลใหม่บนจอแสดงผล ตัวเคอร์เซอร์ จะอยู่กับที่ แต่ตัวอักษรข้อมูลเดิมจะถูกดันไปทางซ้ายมือ หากบิต S เป็น 0 เมื่อเกิดข้อมูลใหม่บนจอแสดงผล ตัวเคอร์เซอร์จะถูกดันไปทางขวามือ

บิตI\D ใช้ในการกำหนดว่า เมื่อเขียนหรืออ่านข้อมูลแล้ว ทำให้แอดเดรสของDDRAM เพิ่มขึ้นหรือลดลงไป1แอดเดรส โดยถ้าบิตนี้เป็น 1 แอดเดรสของDDRAMจะเพิ่มขึ้น แต่ถ้าเป็น 0 แอดเดรสของDDRAMจะลดลง

4.คำสั่งควบคุมการแสดงผล

มีรายละเอียดดังนี้

บิต7	บิต6	บิต5	บิต4	บิต3	บิต2	บิต1	บิต0
0	0	0	0	1	D	C	B

บิต Dใช้ควบคุมการเปิดปิดจอแสดงผล ถ้าบิตนี้เป็น 1 เป็นการเปิดจอแสดงผล ถ้าบิตนี้เป็น 0 เป็นการปิดจอแสดงผล

บิต C ใช้ควบคุมตัวเคอร์เซอร์บนจอแสดงผล ถ้าบิตนี้เป็น 1 จะมีเคอร์เซอร์แสดงผลบนจอแสดงผล ถ้าบิตนี้เป็น 0 จะไม่มีเคอร์เซอร์แสดงผลบนจอแสดงผล หรือไม่มีการแสดงเคอร์เซอร์

บิต B ควบคุมการกระพริบของตัวเคอร์เซอร์ ถ้าบิตนี้เป็น 1 เคอร์เซอร์จะกระพริบ ถ้าบิตนี้เป็น 0 เคอร์เซอร์จะไม่กระพริบ

5. คำสั่งควบคุมการเลื่อนเคอร์เซอร์และข้อมูลตัวอักษร

มีรายละเอียดดังนี้

บิต7	บิต6	บิต5	บิต4	บิต3	บิต2	บิต1	บิต0
0	0	0	1	S/R	R/L	*	*

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การควบคุมการเลื่อนเคอร์เซอร์และตัวอักษรบนจอแสดงผลขึ้นอยู่กับการกำหนดบิต S/C และR/L สามารถสรุปได้ดังนี้

S/C	R/L	ลักษณะการเลื่อน	ข้อมูลคำสั่ง
0	0	เลื่อนเคอร์เซอร์ไปทางซ้าย	10H-13H
0	1	เลื่อนเคอร์เซอร์ไปทางขวา	14H-17H
1	0	เลื่อนตัวอักษรใหม่ไปทางซ้าย	18H-1BH
1	1	เลื่อนตัวอักษรใหม่ไปทางขวา	1CH-1FH

6.คำสั่งกำหนดฟังชั่นการทำงาน

มีรายละเอียดดังนี้

บิต7	บิต6	บิต5	บิต4	บิต3	บิต2	บิต1	บิต0
0	0	1	DL	N	F	*	*

บิต DL ใช้กำหนดจำนวนบิตที่ใช้ติดต่อส่งผ่านข้อมูล ถ้าบิตนี้เป็น 0 จะเป็นการติดต่อแบบ 4 บิต แต่ถ้าบิตนี้เป็น 1 จะเป็นการติดต่อแบบ 8 บิต

บิต N ใช้กำหนดจำนวนบรรทัดของการแสดงผล ถ้าบิตนี้เป็น 0 จะแสดงผล 1 บรรทัด ถ้าบิตนี้เป็น 1 จะแสดงผล 2 บรรทัด กรณีต้องการให้แสดงผลมากกว่า 2 บรรทัด และสามารถที่จะแสดงผลมากกว่า 2 บรรทัด ต้องกำหนดบิต N ให้เป็น 1

บิต F เลือกความละเอียดของตัวอักษร ถ้าบิตนี้เป็น 0 จะแสดงผลแบบ 5*7 จุด ถ้าบิตนี้เป็น 1 จะแสดงผลแบบ 5*10 จุด

จุดน่าสังเกต คือ โมดูล LCD ขนาด 16 ตัวอักษร 1 บรรทัด แม้จะมีบรรทัดการแสดงผลแค่ 1 บรรทัด แต่จะกำหนด N ให้เป็น 1 เนื่องจากแอดเดรสของ DDRAM แบ่งเป็น 2 ช่อง คือ 00H และ40H

7.คำสั่งเลือกแอดเดรสของ CGRAM

เมื่อต้องการกำหนดแอดเดรสของ CGRAM ต้องกำหนดให้บิต 7 เป็น 0 บิต 6 เป็น 1 ส่วนอีก 6 บิตที่เหลือจะแทนด้วยค่าแอดเดรสของ CGRAM จะต้องกำหนดแอดเดรสด้วยคำสั่งนี้ก่อนที่จะอ่านหรือเขียนข้อมูลให้ CGRAM ซึ่งแอดเดรสของ CGRAMจะอยู่ระหว่าง 00H-3FH

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

8. คำสั่งเลือกแอดเดรสของDDRAM

ใช้ในการเลือกแอดเดรสของDDRAMก่อนที่จะอ่านหรือเขียนข้อมูล ต้องกำหนดให้บิต 7 เป็น 1 ส่วนอีก 7 บิตที่เหลือจะเป็นค่าแอดเดรสของ DDRAM ซึ่งแอดเดรสของ CGRAMจะอยู่ระหว่าง 8CH-FFH ทั้งนี้จำนวนแอดเดรสยังขึ้นอยู่กับการกำหนดสถานะที่บิต N หากแทนบิต N เป็น 0 ซึ่งแอดเดรสของ CGRAMจะอยู่ระหว่าง 80H-0CFH บิต N หากแทนบิต N เป็น 1 ซึ่งแอดเดรสของ CGRAMจะมีอยู่ 2 ช่วง คือ อยู่ระหว่าง 80H-0CFH และ 0C0H-0C7H

9.คำสั่งการแฟลก Busy และแอดเดรส

มีรายละเอียดดังนี้

บิต7	บิต6	บิต5	บิต4	บิต3	บิต2	บิต1	บิต0
BF	A	A	A	A	A	A	A

เป็นคำสั่งที่ใช้อ่านแฟลก Busy (BF) โดยแฟลกนี้จะเป็นตัวบอกสถานะของตัวควบคุม LED ว่าพร้อมจะรับข้อมูลหรือไม่ ถ้าบิต BF เป็น 0 แสดงว่าตัวควบคุม LED พร้อมจะรับข้อมูลหรือคำสั่ง แต่ถ้าบิต BF เป็น 1 แสดงว่าขณะนี้ตัวควบคุม LED ยังอยู่ในกระบวนการทำงานภายในหรือกำลังประมวลผลข้อมูลอยู่ ยังไม่พร้อมจะรับข้อมูลหรือคำสั่ง

เมื่อต้องการอ่านแฟลกต้องกำหนดให้ขา R/W เป็น 1 ด้วย แต่สัญญาณที่ RS ยังคงเป็น 0 อยู่ เพราะข้อมูลนี้เป็นข้อมูลคำสั่ง

นอกจากนี้ยังใช้เป็นข้อมูลคำสั่งอ่านข้อมูลแอดเดรสของ CGRAM และ DDRAM โดยบิต 0-6 เป็นค่าของข้อมูลของแอดเดรสที่ต้องการอ่าน

3.6.4 การเขียนคำสั่งและข้อมูลให้แก่โมดูล LCD

ในการเขียนข้อมูลเพื่อควบคุมให้โมดูล LCDแสดงผลตามที่ผู้ใช้งานต้องการต้องส่งคำสั่ง (instruction) แล้วกำหนดโหมดของการทำงานให้แก่โมดูล LCD ก่อน จากนั้นค่อยส่งข้อมูล (data) ที่ต้องการแสดงผล เนื่องจากข้อมูลของโมดูล LCD มี 8 เส้น คือ D0-D7 และใช้เป็นเส้นทางผ่านของทั้งคำสั่งและข้อมูล ดังนั้นในการส่งคำสั่งและข้อมูลจึงต้องอาศัยการกำหนดสัญญาณลอจิกที่ขา RS ถ้าหากที่ขา RS ได้รับลอจิก 0 หมายความว่า ข้อมูลที่ป้อนให้แก่โมดูล LCD ขณะนั้นเป็นคำสั่ง แต่ถ้าหากที่ขา ได้รับลอจิก 1 หมายความว่า ข้อมูลที่ป้อนให้แก่โมดูล LCD ขณะนั้นเป็นข้อมูลที่ใช้ในการแสดงผล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เมื่อต้องการเขียนหรืออ่านข้อมูลใน CGRAM และ DDRAM เริ่มต้นต้องกำหนดแอดเดรสที่ต้องการอ่านหรือเขียนก่อน โดยใช้คำสั่งเลือกแอดเดรส จากนั้นกำหนดให้ขา RS เป็น 1 เพื่อแจ้งให้ตัวควบคุมภายใน
โมดูล LCD ทราบว่าข้อมูลที่ปรากฏต่อไปนี้เป็นข้อมูลปกติไม่ใช่คำสั่ง

ในกรณีต้องการอ่านข้อมูลต้องกำหนดให้ขา R/W เป็น 1 ข้อมูลขนาด 8 บิต (หรือ 4 บิต) ก็จะปรากฏบนบัสข้อมูล โดยข้อมูลที่อ่านมาได้จะเป็นข้อมูลจากแอดเดรส CGRAM หรือ DDRAM ตามที่ต้องการ

ในกรณีต้องการเขียนข้อมูล เมื่อกำหนดแอดเดรสและป้อนลอจิก 1 ให้ขา RS แล้ว แล้วาต้องกำหนดให้ขา R/W เป็น 0 ข้อมูลที่อยู่บนบัสข้อมูลจะถูกเขียนลงในรีจิสเตอร์ DR จากนั้นค่อยถ่ายทอดไปยัง DDRAM ต่อไป

3.6.5 จังหวะการทำงานของโมดูล LCD

ในการติดต่อกับโมดูล LCD จะต้องมีการหน่วงเวลาหลังจากที่ทำการส่งรหัสคำสั่งหรือข้อมูลเนื่องจากต้องรอให้ไมโครคอนโทรลเลอร์ภายในโมดูล LCD แปลความหมายของรหัสคำสั่งและทำงานตามคำสั่งให้เรียบร้อยก่อน จากนั้นค่อยรับข้อมูลหรือดำเนินการต่อไป

ดังนั้นการใช้งานโมดูล LCD ผู้เขียนโปรแกรมต้องมีโปรแกรมเพื่อหน่วงเวลาทำให้โมดูล LCD พร้อมทำงานด้วย โดยเริ่มจ่ายไฟให้แก่โมดูล LCD ต้องรอประมาณ 10 ms เพื่อให้โมดูล LCD ทำการเตรียมความพร้อมหรืออินทิเชียล (initial) หลังจากนั้นก็กำหนดลอจิกให้แก่ขา RS ของโมดูล LCD แล้วต้องหน่วงเวลาอีกประมาณ 2 ms เพื่อให้ไมโครคอนโทรลเลอร์ในโมดูล LCD แปลความหมายของลอจิกที่ขา RS ว่าข้อมูลต่อไปนี้จะได้รับเป็นรหัสคำสั่งหรือข้อมูลที่ต้องแสดงผล จากนั้นจะเป็นการส่งข้อมูลมารอที่บัสข้อมูล D0-D7 (ในกรณีข้อมูล 8 บิต) ขั้นตอนต่อไปนี้จะเป็นการส่งสัญญาณพัลซ์ ไปที่ขา E เพื่อ enable โมดูล LCDให้รับข้อมูลจากบัสข้อมูลเข้าไป โดยสัญญาณพัลซ์ ที่ป้อนเข้าขา E ของโมดูล LCD ต้องเป็นพัลซ์ขอบขาขึ้น จากนั้นหน่วงเวลาอีกประมาณ 2 ms

ทั้งหมดที่กล่าวมาคือขั้นตอนและจังหวะในการทำงาน 1 รอบของโมดูล LCD จะเห็นได้ว่าจะมีโปรแกรมย่อยที่สำคัญอยู่ 3 โปรแกรมย่อย คือ โปรแกรมอินทีเชียล LCD โปรแกรมหน่วงเวลา โปรแกรมการส่งพัสซ์เพื่อ enable โมดูล LCD

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.7 การเชื่อมต่อกับคีร์แพดหรือสวิตซ์เมตริก 4*3จุด

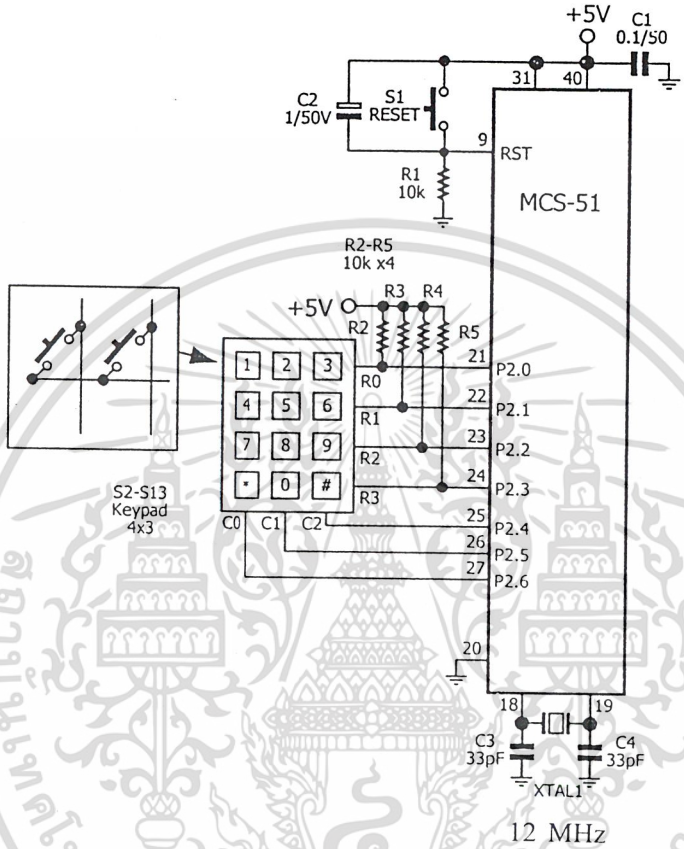
การอ่านค่าหรือรับค่าการกดสวิตซ์เป็นอีกงานหนึ่งที่ไมโครคอนโทรลเลอร์ต้องสามารถรองรับและเชื่อมต่อใช้งานร่วมด้วยวงจรของสวิตซ์มีด้วยกัน 2 ลักษณะใหญ่ๆคือต่อเข้ากับไฟเลี้ยงหรือกราวด์โดยตรงเมื่อสวิตซ์ตัวใดต่อวงจรสามารถอ่านค่าได้โดยตรง วงจรในลักษณะนี้ไม่มีความซับซ้อน สามารถอ่านค่าของสวิตซ์ได้ง่ายและรวดเร็ว แต่มีข้อเสียคือถ้าหากจำนวนของสวิตซ์มีมากๆจำนวนของสายข้อมูลจะมีมากตาม ทำให้ระบบหรือวงจรโดยรวมมีขนาดใหญ่และสิ้นเปลือง

รูปที่ 3.8 วงจรของสวิตซ์แบบเมตริกซ์หรือคีย์แพด

วงจรของสวิตซ์อีกลักษณะหนึ่งคือ การต่อวงจรแบบเมตริกซ์ (metrix switch) ดังในรูปที่3.7 สวิตซ์จะถูกต่อในแนวแกนตั้งและแกนนอน จะเรียกแนวตั้งว่า หลักหรือคอลัมน์ (column) ในขณะที่แนวนอนจะเรียกว่า แถวหรือโรว์ (row) ดังนั้นค่าของสวิตซ์จะต้องประกอบด้วย ตำแหน่งในแนวหลักและแถว กระบวนการที่จะทำให้ได้มาซึ่งค่าของสวิตซ์มีขั้นตอนซับซ้อนพอสมควร แต่วงจรของสวิตซ์แบบนี้มีข้อดีคือสามารถรองรับการเพิ่มของสวิตซ์ได้อย่างสะดวก เพียงเพิ่มเติมจำนวนสวิตซ์และแก้ไขซอฟต์แวร์อีกเล็กน้อยเท่านั้น ทำให้วงจรสวิตซ์แบบเมตริกเป็นที่นิยมใช้มากในระบบควบคุมอัตโนมัติหรือกึ่งอัตโนมัติที่มีจำนวนสวิตซ์มากกว่า 8 ตัว ในการใช้งานทั่วไปจะเรียกสวิตซ์แบบเมตริกซ์นี้ว่า คีร์แพด(keypad)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การเชื่อมต่อคีร์แพดเข้ากับไมโครคอนโทรลเลอร์ MCS-51มีตัวอย่างวงจรดังรูปที่ 3.9 จะใช้พอร์ต 2 ของไมโครคอนโทรลเลอร์เชื่อมต่อเข้ากับคีร์แพดทั้ง 7 เส้นคือสายของคอลัมน์ 3 สาย C0-C2 และสายทางโรว์อีก 4 สายคือ R0-R3



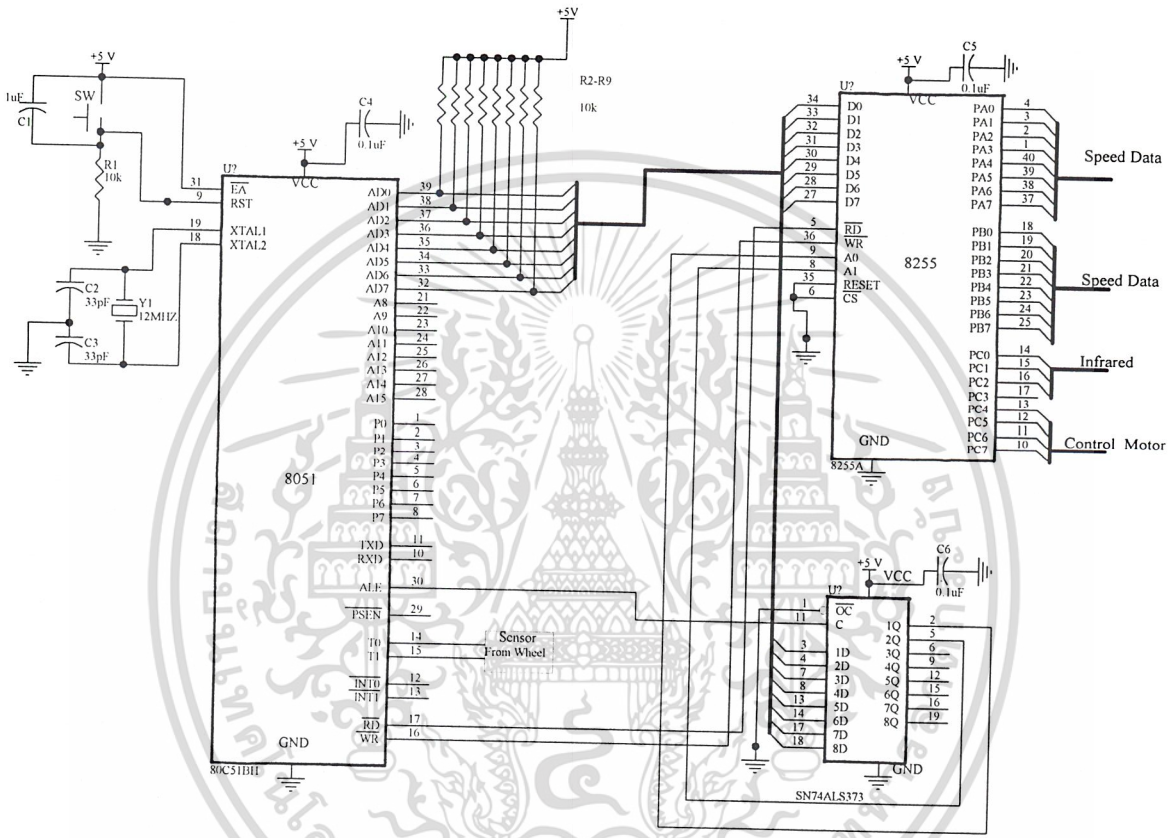
รูปที่ 3.9 วงจรเชื่อมต่อคีย์แพดเข้ากับไมโครคอนโทรลเลอร์

โดยเฉพาะที่ขาพอร์ต P2.0-P2.3 จะต้องต่อตัวต้านทานพูลอัปไว้เพื่อกำหนดสภาวะเริ่มต้นที่ไม่มีการกดคีย์ โดยไมโครคอนโทรลเลอร์จะทำการส่งข้อมูล 0 ไปยัง P2.6,P2.5 และ P 2.4 ตามลำดับ ในทุกครั้งที่มีการส่งข้อมูลไปยังสายคอลัมน์ของคีย์แพด ไมโครคอนโทรลเลอร์จะทำการอ่านค่าที่ P2.0-P2.3 เข้ามาด้วย หากไม่มีการกดค่าของ P2.0-P2.3 ก็จะเป็น 1 ทั้งหมด ถ้าหากมีการกดคีย์ ค่าของ P2.0-P2.3 ก็จะไม่เป็น 1111อีกต่อไป เป็นการแจ้งให้ทราบว่า มีการกดคีย์แพดขึ้นแล้ว จากนั้นไมโครคอนโทรลเลอร์ก็จะทำการค้นหาตำแหน่งต่อไป โดยการค้นหาตำแหน่งสิ่งที่จะได้มาอย่างแรกคือ ค่าตำแหน่งของคีย์นั้น จากนั้นก็จะนำตำแหน่งนั้นไปเปิดตารางข้อมูล เพื่อจะได้หมายเลขของคีย์ที่กดอย่างแท้จริง ยกตัวอย่างจากวงจรในรูปที่ หากคีย์ 0 ถูกกด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ไมโครคอนโทรลเลอร์จะได้ค่าตำแหน่งของคีย์ 0 เป็น 0BH จากนั้นนำค่า 0BH นี้ไปเปิดตารางก็จะได้ข้อมูลเป็น 0 หมายถึงคีย์ 0 ซึ่งกระบวนการหลังนี้จะเป็นหน้าที่ของซอฟแวร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.6 รูปวงจรส่วนไมโครคอนโทรลเลอร์

บทที่ 4
ส่วนเซ็นเซอร์

แสงอินฟราเรด (infrared light) คือคลื่นแม่เหล็กไฟฟ้าชนิดหนึ่งที่มีความยาวคลื่นอยู่ในช่วงประมาณ 0.75 μm ถึงประมาณ 1,000μm ถูกค้นพบโดย Sir William Herchel เมื่อ ค.ศ. 1800 แสงอินฟราเรด แบ่งออกเป็นช่วงย่อยๆได้ 3 ช่วงคือ 1)ช่วงอินฟราเรดใกล้ (near infrared : 0.75-3 m) 2)ช่วงอินฟราเรดกลาง (middle infrared : 3-25 m) และ 3)ช่วงอินฟราเรดไกล (far infrared : 25-1,000 m) เนื่องจากตาของมนุษย์มีความไวต่อแสงเฉพาะในช่วงแสงมองเห็นซึ่งมีความยาวคลื่นประมาณ 0.4-0.7 μm ดังนั้นตาของมนุษย์จึงไม่สามารถมองเห็นแสงอินฟาเรด แต่เนื่องจากวัสดุทุกชนิดที่มีอุณหภูมิสูงกว่าศูนย์องศาสมบูรณ์จะเปล่งแสงอินฟราเรดออกมาเสมอ อินฟราเรดมีประโยชน์ต่อมนุษย์มากมายเช่น ใช้ตรวจวัสดุ ใช้วัดอุณหภูมิของวัสดุโดยไม่มีการสัมผัส ใช้ในการสื่อสาร วัดคุมอุตสาหกรรม อุปกรณ์ทางการแพทย์

สำหรับรถหลบหลีกสิ่งกีดขวางอัตโนมัติก็ได้มีการนำแสงอินฟราเรดมาประยุกต์ใช้งานในส่วนเซนเซอร์โดยการให้ภาคส่ง ส่งแสงอินฟราเรดออกไปถ้ามีวัตถุมากีดขวางการเคลื่อนที่ของแสงอินฟราเรดก็จะเกิดการสะท้อนขึ้น โดยจะให้ภาครับทำหน้าที่รับแสงอินฟราเรดที่สะท้อนเข้ามา ดังนั้นรถ หลบหลีกสิ่งกีดขวางอัตโนมัติจะสามารถรับรู้ได้ว่ามีวัตถุมากีดขวางการเคลื่อนที่โดยดูจากแสงอินฟราเรดที่สะท้อนเข้ามานั่นเองสำหรับรายละเอียดของวงจรภาคส่งและภาครับอินฟราเรดแสดงดังรูปที่

4.1 ภาคส่งอินฟราเรด

สำหรับภาคส่งอินฟราเรดจะใช้ไอซี 555 เป็นตัวกำเนิดความถี่ที่จะใช้ส่ง เอาท์พุทจากไอซี 555 จะต่อเข้ามาที่ขาเบสของทรานซิสเตอร์เพื่อทำการขยายให้มีค่ากระแสมากขึ้นกำลังส่งก็จะมีค่ามากขึ้น

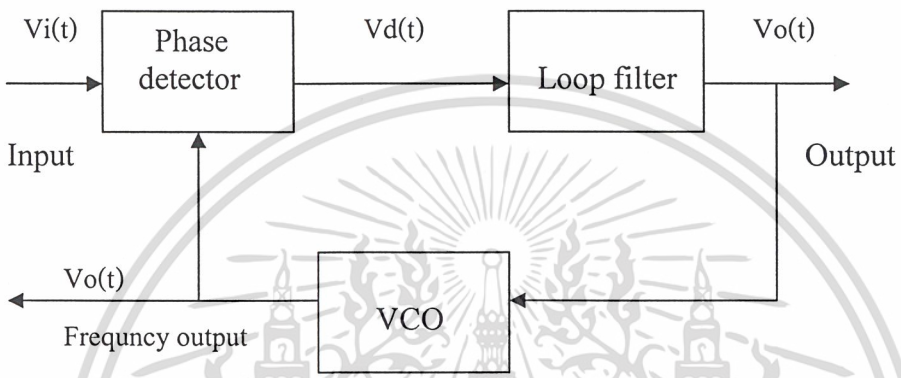
4.2 ภาครับอินฟราเรด

ภาครับอินฟราเรดจะใช้ไอซีเฟสล็อกลูปเป็นตัวช่วยในการตรวจจับความถี่โดยมีโฟโต้ไดโอดรับสัญญาณที่สะท้อนเข้ามา ผ่านวงจรจูนเพื่อให้ได้ความถี่ตรงกับความถี่ที่ส่งมาจากภาคส่งจากนั้นเข้าสู่วงจรปรีแอมป์เพื่อทำการขยายสัญญาณให้แรงขึ้น ต่อไปจะเข้าสู่วงจรเฟสล็อกลูปเพื่อทำการเปรียบเทียบความถี่ที่เข้ามาว่าตรงกับความถี่ที่ไอซีเฟสล็อกลูปล็อกไว้หรือไม่ [ความถี่ที่ไอซีเฟสล็อกลูปทำการล็อกถูกกำหนดโดยสมการ fo = 1.1/[(R7+VR1)C7]] ถ้าความถี่ตรงกันก็จะทำให้เอาท์พุทของไอซีเฟสล็อกลูปมีค่าลอจิก 0 แต่ถ้าความถี่ที่เข้ามาไม่ตรงกับความถี่ที่ไอซีเฟสล็อกลูปล็อกไว้หรือไม่มีสัญญาณเข้ามาที่ภาครับ เอาท์พุทของไอซีเฟสล็อกลูปจะมีค่าลอจิก 1 ซึ่งค่าลอจิกเหล่านี้ จะถูกส่งไปยังไมโครคอนโทรลเลอร์เพื่อทำการประมวลผลต่อไป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

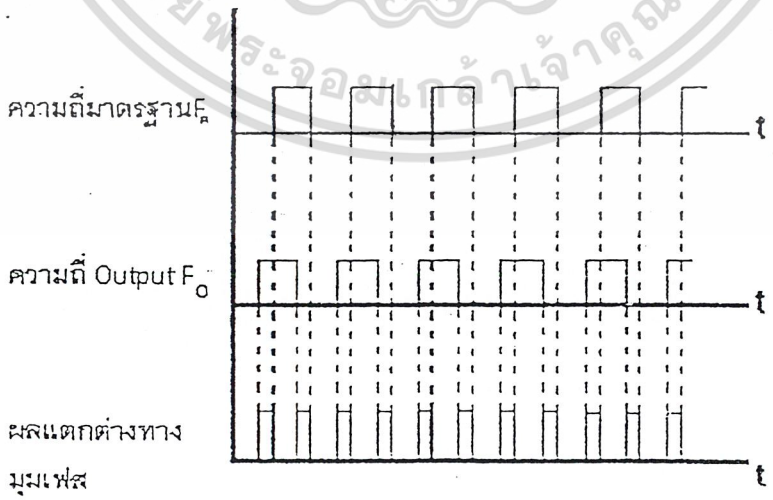
4.3 เฟสล็อคลูป

เฟสล็อกลูป (Phase-locked loop) เป็นหลักการทางการป้อนกลับแบบหนึ่งและนำมาประยุกต์ใช้ในวงจรต่างๆมากมายเช่น Frequency synthesis , Frequency multiplication , Pulse synchronization และ FM/Am demodulation วงจรเฟสล็อกลูปมีองค์ประกอบพื้นฐานดังรูปซึ่งแต่ละส่วนมีผลต่อคุณสมบัติและการทำงานของระบบ หน้าที่ของแต่ละส่วนมีดังนี้



รูปที่ 4.1 แสดงบล็อคไดอะแกรมของวงจรเฟสล็อกลูป

1.เฟสดีเทคเตอร์ (Phase detector) จะทำหน้าที่เปรียบเทียบเฟสของอินพุทกับเอาท์พุทที่มาจากVCO (Voltage-controlled oscillator) เอาท์พุทที่ได้จะอยู่ในรูปของความต่างเฟสที่จะมีค่าน้อยที่สุดเป็นศูนย์ และมีค่ามากที่สุดเป็น $\pi/2$



รูปที่ 4.2 ผลการเปรียบเทียบเฟส

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.ลูปฟิลเตอร์ (Loopfilter) ทำหน้าที่กรองสัญญาณความถี่สูงที่ออกมาจากเฟสดีเทคเตอร์ ให้เป็นสัญญาณ DC เพื่อนำไปทำการควบคุมการผลิตความถี่ของ VCO จนกระทั่ง VCO ทำการล็อคกับสัญญาณอินพุท นั่นคือ fvco เท่ากับ fin ด้วยค่าความต่างเฟสคงที่ค่าหนึ่ง

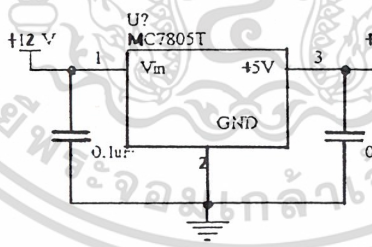
3. VCO (Voltage-controlled oscillator)เป็นส่วนที่จำเป็นสำหรับเฟสล็อกลูปคือเป็น Oscillator ซึ่งจะผลิตความถี่ที่สามารถควบคุมได้จากเอาท์พุทของ Phase detecter

สามารถสรุปการทำงานของเฟสล็อกลูปอย่างคร่าวๆได้ดังนี้ เฟสดีเทคเตอร์จะทำการเปรียบเทียบเฟสของสัญญาณอินพุท Vi(t) กับ เอาท์พุทของ VCO ทำให้ได้เออเรอร์โวลเตจ Vd(t) สัญญาณ

เออเรอร์โวลเตจนี้จะถูกกรองด้วยลูปฟิลเตอร์และถูกป้อนไปยังคอนโทรลอินพุทของ VCO ในรูปของแรงดันควบคุมเพื่อควบคุมการผลิตความถี่ของ VCO จนกระทั่งความถี่ของเอาท์พุทล็อคกับความถี่ของสัญญาณอินพุท

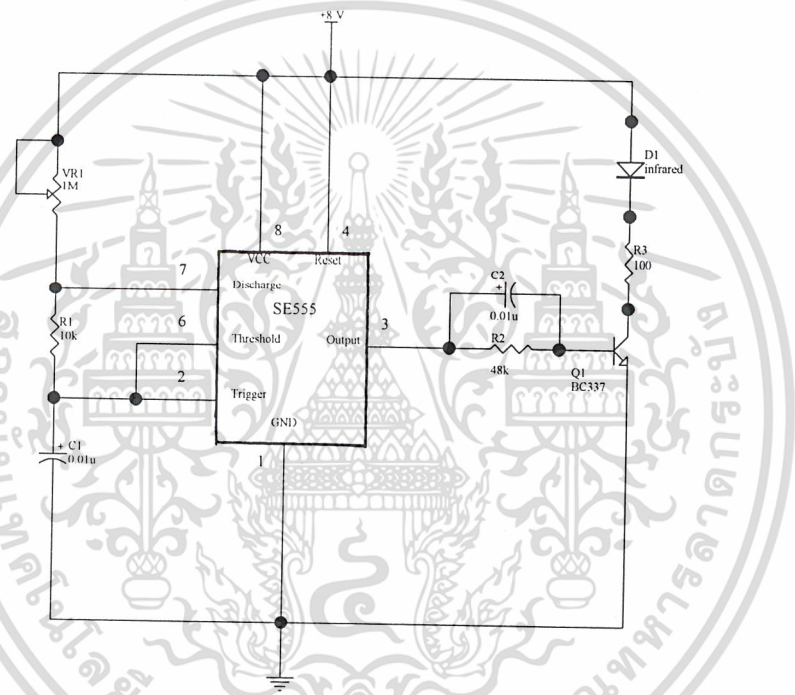
4.4 แหล่งจ่ายพลังงาน

สำหรับแหล่งจ่ายพลังงานที่ใช้กับรถ หลบหลีกสิ่งกีดขวางอัตโนมัติจะใช้แบตเตอร์รี่ 12Vเป็นแหล่งจ่ายพลังงาน ส่วนระดับแรงดันอื่นๆที่ใช้ในวงจรจะใช้ไอซีเบอร์ 78xx เป็นตัวปรับระดับแรงดัน ซึ่งวงจรสามารถแสดงได้ดังรูป



รูปที่ 4.3 วงจรเรกกูเรเตอร์แรงดัน 5V

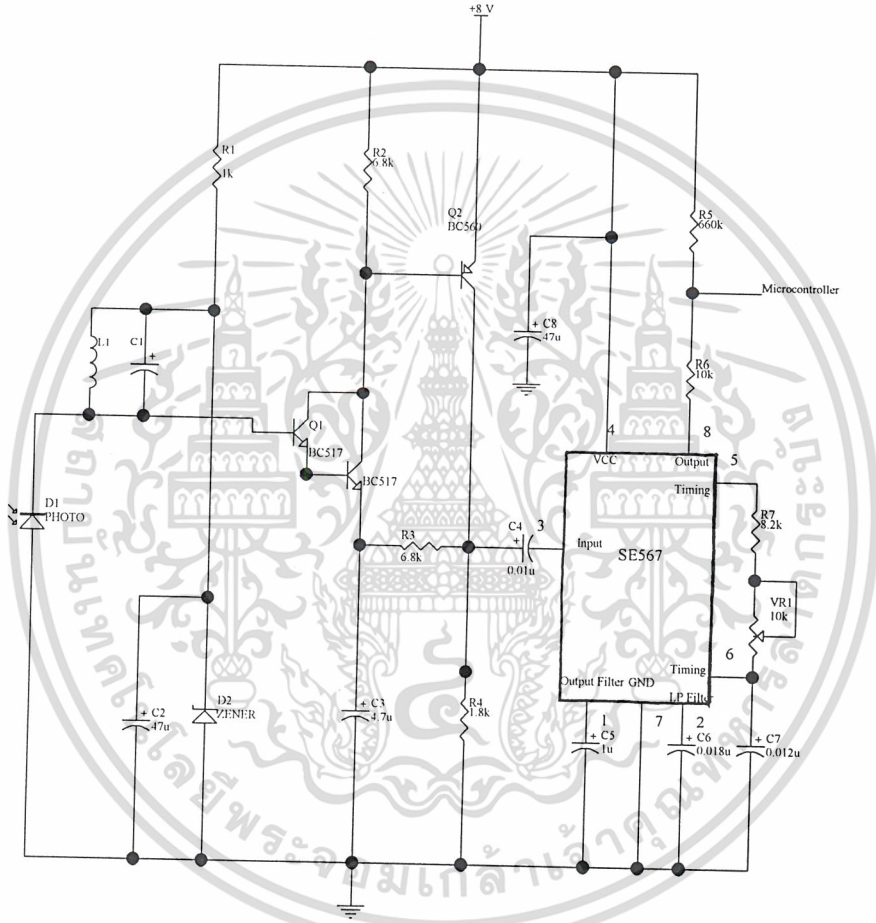
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

Title		
Size: B	Number	Revision
Date: 15-Apr-2003		Sheet of
File: C:\Documents and Settings\wut\Desktop\printed.DDB		

รูปที่ 4.4 รูปวงจรภาคส่งอินฟราเรด



รูปที่ 4.5 รูปวงจรภาครับอินฟราเรด

บทที่ 5
โปรแกรมควบคุมการทำงาน

5.1 โปรแกรมหลักควบคุมการทำงานของรถหลบหลีกสิ่งกีดขวางอัตโนมัติ

การทำงานของรถหลบหลีกสิ่งกีดขวางอัตโนมัติ จะถูกควบคุมโดยการเขียนโปรแกรมภาษาแอสเซมบลี้ของไมโครคอนโทรลเลอร์ MCS51 โดยในตอนแรกจะมีการกำหนดค่าเริ่มต้นต่างๆ จากนั้นจะทำการรับค่าระยะทางเป้าหมายในพิกัด (X,Y) จากคีย์บอร์ด และสุดท้ายรถจะวิ่งไปยังเป้าหมายโดยสามารถหลบหลีกสิ่งกีดขวางได้อย่างอัตโนมัติ ซึ่งสามารถแสดงได้โดยแผนภาพการไหลดังรูป

5.2 โปรแกรมควบคุมให้รถวิ่งไปยังเป้าหมาย

ขั้นตอนของโปรแกรมแสดงได้ดังรูป ซึ่งสามารถอธิบายได้ดังนี้คือ
ขั้นตอนแรกจะเป็นการกำหนดค่าในรีจิสเตอร์ต่างๆ เช่น กำหนดโหมดไทเมอร์เคาน์เตอร์ จากนั้นจะทำการตรวจสอบว่าถึงเป้าหมายหรือยังถ้ายังไม่ถึงเป้าหมายก็จะไปตรวจสอบเซ็นเซอร์ว่าตรวจพบวัตถุหรือไม่ถ้าเซ็นเซอร์ตรวจไม่พบวัตถุก็จะไปตรวจสอบระยะเป้าหมายอีกครั้งหนึ่ง แต่ถ้าเซ็นเซอร์ตรวจพบวัตถุก็จะทำการเลี้ยวหลบตามลักษณะของบิทข้อมูลตำแหน่งวัตถุที่เซ็นเซอร์ตรวจพบซึ่งแบ่งเป็น 2 กรณีดังนี้

1. ถ้าบิทข้อมูลตำแหน่งวัตถุที่เซ็นเซอร์ตรวจพบมีค่าเป็น 000 (เซ็นเซอร์ทั้ง3ตัวตรวจพบวัตถุ) , 011(เซ็นเซอร์ตัวทางซ้ายตรวจพบวัตถุ) , 001 (เซ็นเซอร์ตัวทางซ้ายและเซ็นเซอร์ตัวกลางตรวจพบวัตถุ) , 101 (เซ็นเซอร์ตัวกลางตรวจพบวัตถุ) , 010 (เซ็นเซอร์ตัวทางซ้ายและเซ็นเซอร์ตัวทางขวาตรวจพบวัตถุ) โปรแกรมจะให้รถจะเลี้ยวหลบทางขวา

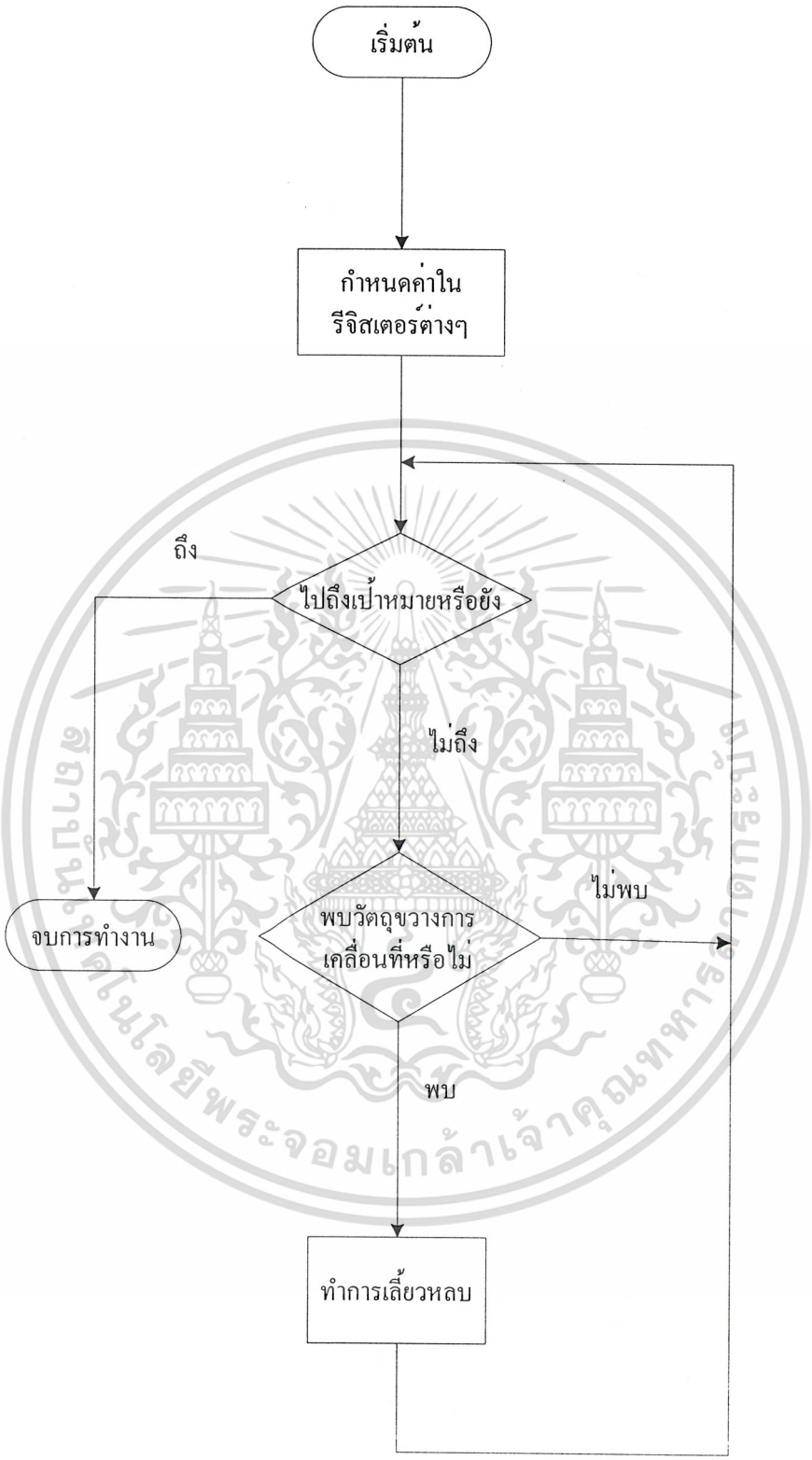
2. ถ้าบิทข้อมูลตำแหน่งวัตถุที่เซ็นเซอร์ตรวจพบมีค่าเป็น 100 (เซ็นเซอร์ตัวทางขวาและเซ็นเซอร์ตัวกลางตรวจพบวัตถุ , 110 (เซ็นเซอร์ตัวทางขวาตรวจพบวัตถุ) โปรแกรมจะให้รถจะเลี้ยวหลบทางซ้าย

เมื่อทำการเลี้ยวหลบแล้วโปรแกรมก็จะกลับไปตรวจสอบระยะเป้าหมายอีกครั้งหนึ่งจนกระทั่งถึงระยะพิกัดเป้าหมายโปรแกรมก็จะสิ้นสุดการทำงาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 5.1 แผนภาพการไหลของโปรแกรมหลักควบคุมการทำงานของรถ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 5.2 แผนภาพการไหลของโปรแกรมย่อยควบคุมให้รถวิ่งไปยังเป้าหมาย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5.3 โปรแกรมวัดระยะทาง

การวัดระยะทางในการเคลื่อนที่ของรถจะอาศัยการนับพัลซ์จากการหมุนของมอเตอร์ สำหรับการคำนวณระยะทางนั้นสามารถทำได้ดังนี้

- เส้นรอบวงของล้อยาว 39 เซ็นติเมตร
- รถเคลื่อนที่ครบ 1 รอบสามารถนับพัลซ์ได้ 22 พัลซ์
- ถ้ารถเคลื่อนที่ได้ระยะทาง 30 เซนติเมตร จะคิดเป็นจำนวณพัลซ์เท่ากับ (22/39)* 30 = 17 พัลซ์

จากข้อมูลข้างต้นทำให้โปรแกรมสามารถรู้ระยะทางที่รถเคลื่อนที่ได้โดยการนับจำนวณพัลซ์นั่นเอง สำหรับโปรแกรมวัดระยะทางจะทำโดยการเก็บค่าระยะทางเป้าหมายไว้ในรีจิสเตอร์ (ระยะทางเป้าหมายจะอยู่ในรูปของจำนวณพัลซ์) แล้วควบคุมให้เคาน์เตอร์นับจำนวณพัลซ์ที่เกิดจากการเคลื่อนที่ของรถ จากนั้นจะทำการเปรียบเทียบค่าระยะทางเป้าหมายกับค่าระยะทางที่นับได้ ถ้าระยะทางทั้ง 2 เท่ากันแสดงว่ารถได้เคลื่อนที่มาถึงเป้าหมายแล้วโปรแกรมก็จะจบการทำงาน

5.4 โปรแกรมควบคุมการเลี้ยว

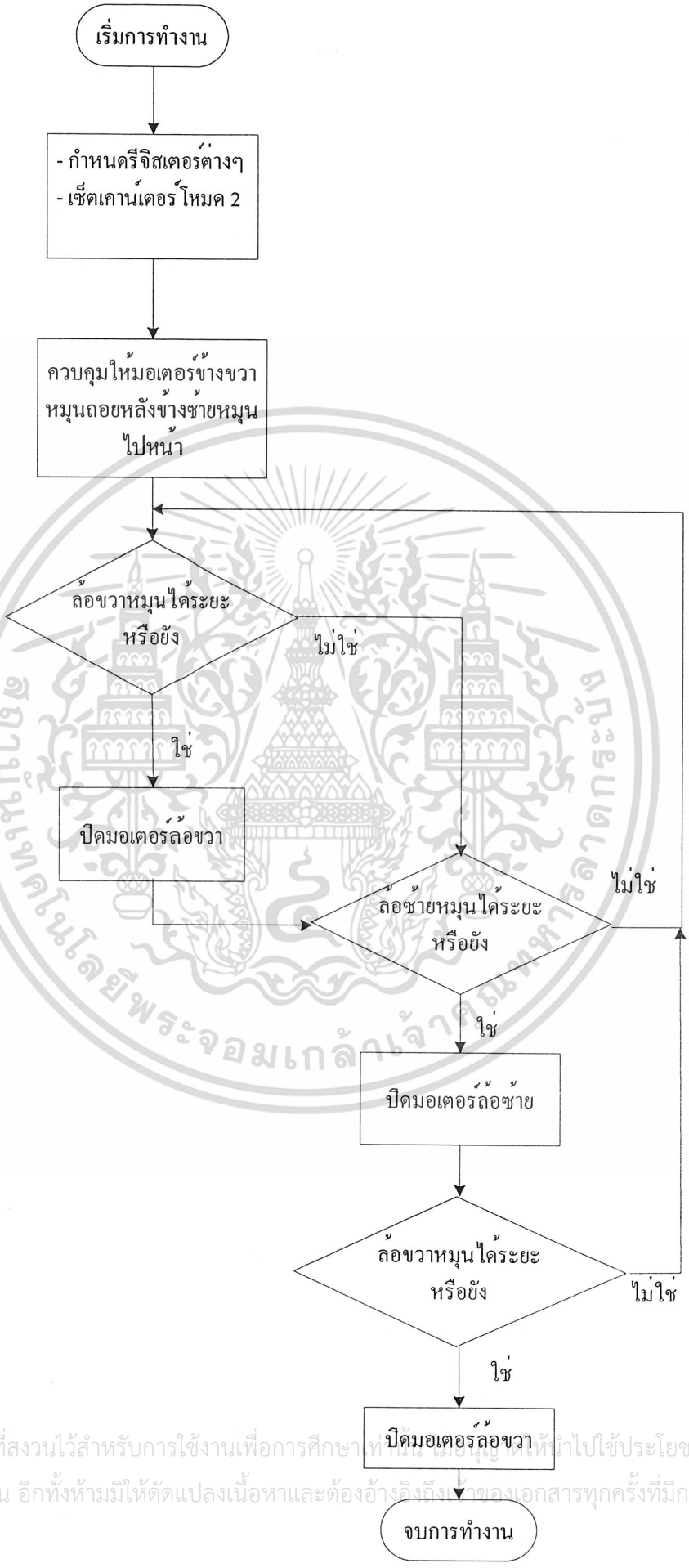
ในการเลี้ยวซ้ายและเลี้ยวขวา เพื่อไม่ให้เสียระยะทางที่เกิดจากการเลี้ยวจึงต้องให้ล้อข้างหนึ่งหมุนไปข้างหน้าอีกข้างหมุนกลับหลังเป็นระยะทางที่เท่ากันเพื่อให้รถหมุนรอบจุดศูนย์กลาง

การเลี้ยวจะควบคุมให้รถเลี้ยวเป็นมุมฉากเพื่อให้รถอยู่บนแกนของพิกัดเป้าหมาย ในการเลี้ยวจะมีการคำนวณดังนี้

- ระยะห่างระหว่างล้อมีค่าเท่ากับ 29.5 เซนติเมตร
- ล้อทั้ง 2 ต้องหมุนรอบจุดศูนย์กลางที่อยู่ระหว่างล้อเป็นแนวโค้งของวงกลม
- ระยะจากจุดศูนย์กลางระหว่างล้อไปยังล้อแต่ละด้านคิดเป็น (29.5/2) เซนติเมตร
- การที่จะให้รถเลี้ยวเป็นมุมฉากจะต้องให้ล้อของรถแต่ละด้านหมุนไปเป็นระยะทางเท่ากับหนึ่งในสี่ของเส้นรอบวง ซึ่งมีค่าเท่ากับ (1/4)*2π*(29.5/2) = 23.169 เซ็นติ-เมตร คิดเป็นจำนวณพัลซ์เท่ากับ (22/39)*23.169 = 13 พัลซ์

สำหรับโปรแกรมควบคุมการเลี้ยวแสดงได้ดังแผนภาพการไหลรูป ซึ่งเป็นโปรแกรมการเลี้ยวขวาสามารถอธิบายได้ดังนี้คือ เริ่มต้นจะมีการกำหนดเคาน์เตอร์ทั้ง 2 ข้างเป็นโหมด 2 ซึ่งเป็นไทเมอร์เคาน์เตอร์ 8 บิทแบบตั้งค่าอัตโนมัติ มีการกำหนดค่าเริ่มของการนับ ในส่วนโปรแกรมจะทำการตรวจสอบบิท TF0 และ TF1 ว่ามีการเซ็ตหรือไม่ (เกิดโอเวอร์โฟลวหรือไม่) ถ้าพบว่าบิทข้างใดเกิดการเซ็ตก็จะทำการปิดมอเตอร์ข้างนั้นจนกระทั่งปิดมอเตอร์ครบทั้ง 2 ข้าง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 5.3 แผนภาพการไหลของโปรย่อยควบคุมการเลี้ยวขวา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่6

สรุปผลการทดลอง

การทำงานของรถหลบหลีกสิ่งกีดขวางจะมีวงจรต่างๆหลายวงจรที่ทำงานสัมพันธ์กัน โดยจะแบ่งเป็นภาคๆดังนี้

6.1Drive Motor Circiut

จากตารางด้านล่างอินพุทคือ ลอจิกที่ใส่ให้ D/A 1408 ซึ่งจะได้เอาท์พุทเป็นแรงดันรูป Square wave ที่ขาเอาท์พุทของ TL494 โดยมี Duty cycle ที่แตกต่างกัน

A1	A2	A3	A4	A5	A6	A7	A8	แรงดันที่ออกจาก 741 (V)	Duty cycle ของแรงดันที่ TL494 (%)
0	0	0	0	0	0	0	0	0	86.94
0	0	0	0	0	0	0	1	0.05	82.61
0	0	0	0	0	0	1	1	0.1	81.52
0	0	0	0	0	1	1	1	0.22	78.26
0	0	0	0	1	1	1	1	0.45	71.74
0	0	0	1	1	1	1	1	0.8	58.69
0	0	1	1	1	1	1	1	1.65	33.33
0	1	1	1	1	1	1	1	3.3	-
1	1	1	1	1	1	1	1	6.6	-

ค่า Duty cycle คือ ค่าเปอร์เซ็นต์ของแรงดันไฟบวกต่อคาบของแรงดันทั้งหมด โดยที่ค่า Duty cycle ยิ่งมากจะทำให้มอเตอร์ยิ่งหมุนด้วยความเร็วที่มากขึ้น (ทั้งนี้เพราะมีไฟเลี้ยงจ่ายให้กับมอเตอร์นานขึ้นในหนึ่งคาบของแรงดัน) ซึ่งจากตารางด้านบนค่า Duty cycle ที่มอเตอร์จะหมุนเร็วที่สุดคือ ค่า 0 0 0 0 0 0 0 0

จากรูป ..... ที่รีเลย์ตัวบนจะเป็นคล้ายกับสวิตซ์ที่เป็นตัวเปิด-ปิดมอเตอร์ว่าจะเริ่มทำงาน (รถวิ่ง)หรือยัง โดยถ้าให้แรงดันลอจิก 1 ที่ขาเบสของทรานซิสเตอร์ควบคุมรีเลย์ตัวบน และที่ขาเบสของทรานซิสเตอร์ควบคุมรีเลย์ตัวล่าง ถ้าให้มีค่าลอจิก 0 จะเป็นการให้แรงดัน Square wave ผ่านเข้ามาที่ขาเบสของ Q1 และQ4 เป็นผลให้แรงดัน +12 โวลต์ไหลผ่านมอเตอร์ ตามแรงดัน Square wave ที่มี Duty cycle มากน้อยแตกต่างกัน (ขณะ Square wave มีแรงดันจะเป็นการ turn on ตัว Q1 และQ4) ในกรณีนี้เป็นการให้มอเตอร์หมุนไปข้างหน้า

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ถ้าแรงดันที่ขาเบสของทรานซิสเตอร์ควบคุมรีเลย์ตัวล่างมีค่าเป็นลอจิก 1 จะเป็นการให้แรงดัน Square wave ผ่านเข้ามาที่ขาเบสของ Q2 และQ3 เป็นผลให้แรงดัน +12 โวลต์ไหลผ่านมอเตอร์ ตามแรงดัน Square wave แต่การไหลของกระแสจะกลับทิศทางกับกรณีแรก เป็นผลให้มอเตอร์หมุนไปข้างหลัง

กล่าวโดยสรุป ที่ขาเบสของทรานซิสเตอร์ควบคุมรีเลย์ตัวบนถ้าให้มีค่าลอจิก 1 และที่ขาเบสของทรานซิสเตอร์ควบคุมรีเลย์ตัวล่าง ถ้าให้มีค่าลอจิก 0 จะเป็นการให้มอเตอร์หมุนไปข้างหน้า และถ้าขาเบสของทรานซิสเตอร์ควบคุมรีเลย์ตัวล่าง ถ้าให้มีค่าลอจิก 1 จะเป็นการให้มอเตอร์หมุนไปข้างหลัง (การที่จะทำให้มอเตอร์ทำงานในกรณีไหนก็ตามต้องให้ค่าลอจิก 1ที่ขาเบสของทรานซิสเตอร์ควบคุมรีเลย์ตัวบนเสมอ)

สถานการณ์	เอาท์พุทของ Infared Sensor
มีสิ่งกีดขวาง Infrared ถูกสะท้อนกลับ	ลอจิก 0
ไม่มีสิ่งกีดขวาง Infrared ไม่ถูกสะท้อนกลับ	ลอจิก 1

6.2Infared Sensor Circuit

ซึ่งการทำงานได้ใช้วงจร Phase lock loop โดยจะมีตัวส่งและตัวรับที่ทำงานสัมพันธ์กัน ได้แก่ ตัวส่งจะส่ง Infrared ที่มีความถี่ที่กำหนดไว้ค่าๆหนึ่ง ออกไป ส่วนตัวรับจะคอย detect ความถี่ค่าที่ถูกส่งออกไป ถ้าInfrared ความถี่เดิมถูกสะท้อนกลับมา เอาท์พุทของ Infrared Sensor จะมีค่าเป็นลอจิก 0 โดยในกรณีที่ไม่มีสิ่งกีดขวาง จะมีค่าเป็นลอจิก 1 ค่าเอาท์พุทนี้จะถูกส่งไปยัง Microcontroller circuit เพื่อประมวลผลสำหรับหักเลี้ยวหลบต่อไป

ค่า R3 ที่ภาคส่ง (โอมห์)	ระยะทางที่เซ็นเซอร์สามารถที่จะ detect สิ่งกีดขวางได้ (cm)
83.33	19.2
100	17.7
500	9.5
1K	4.8

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

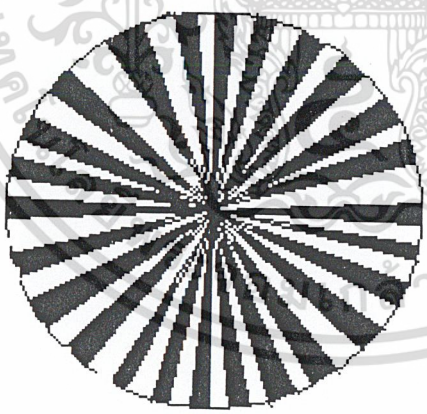
สรุปง่ายๆคือ เป็นการ lock ความถี่ที่ต้องการไว้ และทำการตรวจสอบว่าความถี่ดังกล่าวได้ถูกสะท้อนกลับมาหรือเปล่า

6.3Counter นับรอบล้อ

เป็นวงจรที่ตรวจสอบว่ามอเตอร์หมุนได้จำนวนรอบเท่าไรแล้ว ทั้งนี้เพื่อการประมวลผลที่ถูกต้องแม่นยำ การทำงานจะใช้แถบมืด-สว่าง ตัดผ่าน Opto device จะเกิดแรงดัน ลอจิก 0 หรือ 1 ขึ้น โดยให้วงจรนี้นับว่าแถบมืด-สว่างจำนวน 22 แถบถือเป็น 1 รอบการหมุนของมอเตอร์

การนับดังกล่าวนี้มีจุดประสงค์หลายประการซึ่งเป็นเหตุผลที่สำคัญทั้งนั้น อาทิ

- เป็นการนับรอบถอยหลังเพื่อให้สามารถรู้ได้ว่ารถถึงจุดหมายหรือยัง
- เพื่อเป็นการควบคุมให้มอเตอร์ทั้ง 2 ข้าง หมุนได้ระยะทางที่เท่ากัน โดยการเปรียบเทียบจำนวนแถบมืด-สว่างให้มีค่าเท่ากัน
- เพื่อเป็นการควบคุมการเลี้ยวซ้าย-ขวาให้เลี้ยวได้เป็นมุมฉาก โดยการให้มอเตอร์ข้างหนึ่งหมุนไปข้างหน้า และอีกข้างหมุนไปข้างหลัง เพื่อให้พิกัดขณะเลี้ยวไม่เปลี่ยนแปลง
- เพื่อให้ปรับความเร็วได้เท่ากันทั้ง 2 ข้างของมอเตอร์
-



รูปที่ 6.1 รูปแถบวงกลมที่ใช้คำนวณระยะทาง

ด้านบนคือ รูปตัวที่ใช้ในการตรวจเช็คระยะทางในการเคลื่อนที่ของรถ จะถูกติดไว้ที่แกนของมอเตอร์ ซึ่งจะหมุนตามแกนของมอเตอร์และตัดอุปกรณ์ Opto device H21A3 เพื่อให้ได้เป็น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Square wave และถูกนำไปประมวลผลต่อที่ส่วน Microcontroller circuit เพื่อจุดประสงค์ดังกล่าวข้างต้น

สถานการณ์	เอาท์พุทของ Counter นับรอบล้อ
แถบดำที่แผ่นนับรอบล้อมาขวางการรับ-ส่งอินฟาเรดของ H21A3	ลอจิก 0
ไม่มีแถบดำที่แผ่นนับรอบล้อมาขวางการรับ-ส่งอินฟาเรดของ H21A3	ลอจิก 1

6.4Microcontroller circuit

เป็นวงจรที่ควบคุมการทำงานของทั้ง 3 วงจรข้างต้นให้ทำงานสัมพันธ์กันอย่างมีประสิทธิภาพ การทำงานจะมีขั้นตอนคร่าวๆดังต่อไปนี้

- รับพิกัดจุดหมายจากคีย์โทรศัพท์
- ไมโครคอนโทรลเลอร์จะสั่งมอเตอร์ให้หมุนเพื่อไปยังจุดหมาย
- ขณะเดินทางถ้าพบสิ่งกีดขวาง เซ็นเซอร์จะส่งข้อมูลให้ไมโครคอนโทรลเลอร์และไมโครคอนโทรลเลอร์จะสั่งให้มอเตอร์หลักหลบ
- ขณะเดินทางไมโครคอนโทรลเลอร์จะนับระยะทางลงซึ่งได้ข้อมูลจาก Counter นับรอบล้อ เพื่อตรวจสอบว่ารถถึงจุดหมายหรือยัง

จากรูป 3.6 พบว่า 8051 จะต้องต่อ 8255 เพิ่มเพื่อเพิ่มพอร์ตที่จะนำมาใช้งาน โดย พอร์ต A , พอร์ต B จะเป็นเอาท์พุทจ่ายข้อมูลให้กับ Drive Motor Circiut และ PC0-PC3 เป็นอินพุทรับข้อมูลจาก Infared Sensor Circuit และ PC4-PC7 เป็นเอาท์พุทจ่ายข้อมูลให้กับ Drive Motor Circiut ในการควบคุมมอเตอร์

และที่ T0,T1 ของ 8051 จะเป็นที่รับสัญญาณ Square wave จาก H21A3 เพื่อการควบคุมต่างๆข้างต้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

6.5 Supply Circuit

โดยพลังงานจะถูกส่งจากแบตเตอรี่ 12 โวลต์ 1.3 แอมป์อาวส์ (12 V 1.3 AH) จำนวน 2 ก้อน ซึ่งจะถูกแบ่งจ่ายไปให้กับ 4 วงจรดังนี้

ก้อนแรกจะจ่ายให้กับ Drive Motor Circiut เท่านั้น

ก้อนที่ 2 จะจ่ายให้กับ Infared Sensor , Circuit Counter นับรอบล้อ , Microcontroller circuit

โดย Drive Motor Circiut จะต้องเรคกูเลตแรงดันให้เหลือ 5,-9 โวลต์ด้วย 7805 กับ 7809

Infared Sensor จะต้องเรคกูเลตแรงดันให้เหลือ 8 โวลต์ด้วย 7808

Circuit Counter นับรอบล้อ จะต้องเรคกูเลตแรงดันให้เหลือ 5 โวลต์ด้วย 7805

Microcontroller circuit จะต้องเรคกูเลตแรงดันให้เหลือ 5 โวลต์ด้วย 7805

ทั้งนี้เราไม่สามารถใช้แบตเตอรี่เพียงตัวเดียวได้เพราะ Drive Motor Circiut จะดึงกระแสมากจนทำให้วงจรอื่นๆไม่สามารถทำงานได้อย่างปกติ จึงต้องแยกแบตเตอรี่ให้กับ Drive Motor Circiut ต่างหาก

6.6 ส่วนคีย์โทรศัพท์

รับข้อมูลพิกัดเป็นระนาบ (x,y) จากผู้ใช้บริการเพื่อเป็นจุดหมายให้รถวิ่งไปยังจุดดังกล่าว โดยข้อมูลนี้จะถูกส่งให้กับ Microcontroller circuit นำไปประมวลผลต่อไป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สรุปรวม

ลักษณะทางกายภาพของตัวรถมีข้อมูลต่างๆดังนี้

- ความยาวตัวรถ 30 cm
- ความกว้างตัวรถ 29.5 cm
- ความสูงเมื่อวัด ณ จุดที่สูงสุด 19 cm
- เส้นรอบวงของล้อ 39 cm

พิกัดระยะทางที่กำหนด คือ 1 หน่วยจะเท่ากับความยาวเส้นรอบวงของล้อมีค่า 39 cm ซึ่งการวัดจุดต่างๆจะใช้พิกัดในระนาบ (x,y) เป็นตัวบอก

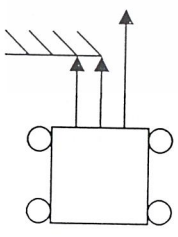
1. *เมื่อให้รถวิ่งโดยไม่มีสิ่งกีดขวาง*

กำหนดพิกัดจุดหมาย(x,y) และพิกัดที่รถวิ่งได้(x,y) โดยไม่มีสิ่งกีดขวาง

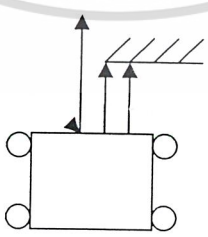
พิกัดที่กำหนด (x,y) หน่วย	ระยะทางที่กำหนด (x,y) cm	พิกัดที่รถวิ่งได้(x,y) หน่วย	ระยะทางที่วิ่งได้ (x,y) cm	ความผิดพลาด (x,y) หน่วย	ความผิดพลาด (x,y) cm
(2,3)	(78,117)	(1.33,2.51)	(52,98)	(0.67.0.49)	(26,19)
(5,5)	(195,195)	(3.92,4.07)	(153,159)	(1.07,0.92)	(42,36)
(7,1)	(273,39)	(5.54,0.85)	(216,33)	(1.46,0.15)	(57,6)
(2,8)	(78,312)	(1.28,6.77)	(50,264)	(0.74,1.23)	(28,48)

2. *พิจารณาการตอบสนองต่อสิ่งกีดขวางในรูปแบบต่างๆ*

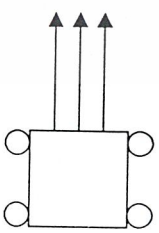
พิจารณาการตอบสนองตามรูปแบบต่างๆดังรูปข้างล่าง เพื่อดูผลก่อนที่จะทดลองในสถานการณ์จริง



เอาท์พุทที่เซ็นเซอร์ คือ 0 0 1
การตอบสนองของรถ คือ เลี้ยวขวา

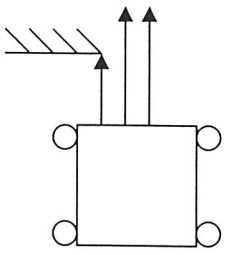


เอาท์พุทที่เซ็นเซอร์ คือ 1 0 0
การตอบสนองของรถ คือ เลี้ยวซ้าย



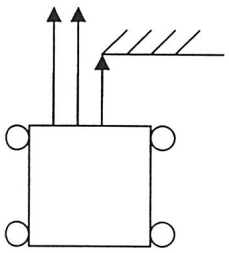
เอาท์พุทที่เซ็นเซอร์ คือ 1 1 1
การตอบสนองของรถ คือ ตามเส้นทางเดิน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



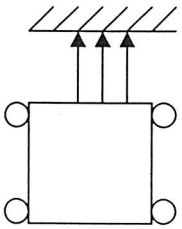
เอาท์พุทที่เซ็นเซอร์ คือ 0 1 1

การตอบสนองของรถ คือ เลี้ยวขวา



เอาท์พุทที่เซ็นเซอร์ คือ 1 1 0

การตอบสนองของรถ คือ เลี้ยวซ้าย



เอาท์พุทที่เซ็นเซอร์ คือ 0 0 0

การตอบสนองของรถ คือ เลี้ยวขวา

จากการทดลองข้างต้นจะพบว่าถ้าเซ็นเซอร์ตัวซ้ายจับสิ่งกีดขวางได้ตัวรถจะเลี้ยวขวา แต่ถ้าเซ็นเซอร์ตัวขวาจับสิ่งกีดขวางได้ตัวรถจะเลี้ยวซ้าย เพื่อหลบสิ่งกีดขวางนั้นๆ

วิจารณ์

จากการทดลองรถหลบหลีกสิ่งกีดขวางอัตโนมัติพบว่าการเคลื่อนที่ของรถในพิกัด x,y ยังมีความผิดพลาดอยู่มาก เนื่องจากการเคลื่อนที่ของรถไม่อยู่ในแนวตรงแม้จะมีการเขียนโปรแกรมควบคุมให้รถเคลื่อนที่ตรงแล้วก็ตาม ที่เป็นเช่นนั้นก็เพราะข้อจำกัดทางฮาร์ดแวร์ เช่น ความละเอียดของจำนวนพัลซ์ที่นับได้ใน 1 วงรอบล้อไม่มากพอรวมทั้งความเร็วของรถที่ไม่สามารถปรับเปลี่ยนได้มากนักเนื่องจากต้องรับน้ำหนักจากแบตเตอรี่ถึง 2 ก้อนการแก้ไขในส่วนนี้อาจทำได้โดยการเพิ่มจำนวนพัลซ์ในแต่ละวงรอบล้อให้มากขึ้นและเปลี่ยนแปลงวงจรส่วนขับมอเตอร์ให้สามารถจ่ายกระแสเข้ามอเตอร์ได้มากขึ้น ปัญหาที่พบอีกอย่างหนึ่งก็คือ เซ็นเซอร์อินฟราเรดไม่สามารถตรวจจับวัตถุสีดำได้ ซึ่งปัญหานี้อาจแก้ไขโดยการใช้เซ็นเซอร์อินฟราเรดร่วมกับเซ็นเซอร์อัลตร้าโซนิคซึ่งจะช่วยแก้จุดอ่อนตรงส่วนนี้ได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
1:
2:   ;-------------------------------------------------------------------------
     --
3:   ; Define Port&Pin Name
4:   ;-------------------------------------------------------------------------
     --
5:   KPAD_ROW0       BIT     P2.0    ; Keypad Input Row 0
6:   KPAD_ROW1       BIT     P2.1    ; Keypad Input Row 1
7:   KPAD_ROW2       BIT     P2.2    ; Keypad Input Row 2
8:   KPAD_ROW3       BIT     P2.3    ; Keypad Input Row 3
9:   KPAD_COL2       BIT     P2.4    ; Keypad Output Column 2
10:  KPAD_COL1       BIT     P2.5    ; Keypad Output Column 1
11:  KPAD_COL0       BIT     P2.6    ; Keypad Output Column 0
12:  LCD_EN          BIT     P3.0
13:  LCD_RS          BIT     P3.1
14:  ;-------------------------------------------------------------------------
     --
15:  ; Define User Register
16:  ;-------------------------------------------------------------------------
     --
17:  LCD_ADDR        EQU     050H
18:  LCD_DATA        EQU     051H
19:  KPAD_DATA       EQU     052H
20:  LCD_DATA_X      EQU     053H
21:  LCD_DATA_Y      EQU     054H
22:
23:  SPD1F           EQU     61H
24:  SPD             EQU     60H
25:  DATA_LX         EQU     40H
26:  DATA_LY         EQU     42H
27:  ERROR_R         EQU     44H
28:  ERROR_L         EQU     45H
29:  ER_L            EQU     46H
30:  ER_R            EQU     47H
31:  ER_RR           EQU     48H
32:  ER_RL           EQU     49H
33:  CASE_Y          EQU     4AH
34:  DATA_X          EQU     4BH
35:  DATA_Y          EQU     4CH
36:  ;-------------------------------------------------------------------------
     --
37:
38:                  ORG     0000H
39:                  ACALL   MODE8255
40:                  ACALL   STOP
41:                  MOV     P1,#00000000B
42:                  CLR     LCD_EN          ; Clear LCD Enable
43:                  CLR     LCD_RS          ; Clear LCD RS
44:
45:  MAIN:           ACALL   INIT_LCD        ; Call LCD Initial subroutine
46:
47:                  MOV     LCD_ADDR,#000H  ; Set Address 00H
48:                  ACALL   SET_ADDR_LCD
49:                  MOV     DPTR,#TITLE_1
50:                  ACALL   WRLINE_LCD
51:
52:                  MOV     LCD_ADDR,#040H  ; Set Address 40H
53:                  ACALL   SET_ADDR_LCD
54:                  MOV     DPTR,#TITLE_2
55:                  ACALL   WRLINE_LCD
56:
57:                  ACALL   DELAY_1s
58:                  ACALL   DELAY_1s
59:
60:  GETKEY:         ACALL   GET_KEY_DATA
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
61:                     ACALL   SHOW_POS_X
62:                     ACALL   GET_KEY_DATA
63:                     ACALL   SHOW_POS_Y      ;Main Of Display
64:                     ACALL   DELAY_1s
65:                     ACALL   SHOW_ALL
66:   WAIT:             LJMP    RUN
67:
68:   SHOW_POS_X:       ACALL   LCD_CLR
69:                     MOV     LCD_ADDR,#00H
70:                     ACALL   SET_ADDR_LCD
71:                     ACALL   LCD_BLINK
72:
73:                     ACALL   DELAY_1s        ;Show Position X
74:                     MOV     A,KPAD_DATA
75:
76:                     CJNE    A,#11,THEN_X
77:                     MOV     DATA_X,#0
78:
79:                     MOV     LCD_DATA,#30H
80:                     AJMP    SH_X
81:   THEN_X:           MOV     DATA_X,A
82:                     ADD     A,#30H
83:                     MOV     LCD_DATA,A
84:   SH_X:             MOV     LCD_DATA_X,LCD_DATA
85:                     ACALL   WRCHAR_LCD
86:
87:                     ACALL   DELAY_1s
88:                     RET
89:
90:   SHOW_POS_Y:       ACALL   LCD_CLR
91:                     MOV     LCD_ADDR,#00H
92:                     ACALL   SET_ADDR_LCD
93:                     ACALL   LCD_BLINK
94:                     ACALL   DELAY_1s
95:                     MOV     A,KPAD_DATA     ;Show Position Y
96:
97:
98:                     CJNE    A,#11,THEN_Y
99:                     MOV     DATA_Y,#0
100:                    MOV     LCD_DATA,#30H
101:                    AJMP    SH_Y
102:
103:  THEN_Y:           MOV     DATA_Y,A
104:                    ADD     A,#30H
105:                    MOV     LCD_DATA,A
106:  SH_Y:             MOV     LCD_DATA_Y,LCD_DATA
107:                    ACALL   WRCHAR_LCD
108:                    ACALL   DELAY_1s
109:                    RET
110:
111:  SHOW_ALL:         ACALL   LCD_CLR
112:                    MOV     LCD_ADDR,#00H
113:                    ACALL   SET_ADDR_LCD
114:                    MOV     DPTR,#TITLE_3
115:                    ACALL   WRLINE_LCD
116:                    MOV     LCD_ADDR,#40H
117:                    ACALL   SET_ADDR_LCD
118:                    MOV     LCD_DATA,#'('
119:                    ACALL   WRCHAR_LCD
120:                    MOV     LCD_DATA,#' '
121:                    ACALL   WRCHAR_LCD
122:                    MOV     LCD_DATA,LCD_DATA_X
123:                    ACALL   WRCHAR_LCD
124:                    MOV     LCD_DATA,#','
125:                    ACALL   WRCHAR_LCD
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
126:                    MOV     LCD_DATA,LCD_DATA_Y
127:                    ACALL   WRCHAR_LCD
128:                    MOV     LCD_DATA,#' '
129:                    ACALL   WRCHAR_LCD
130:                    MOV     LCD_DATA,#')'
131:                    ACALL   WRCHAR_LCD
132:
133:                    RET
134:
135:
136:
137:
138: ;-------------------------------------------------------------------------
     --
139: ; LCD Initialize
140: ;-------------------------------------------------------------------------
     --
141: INIT_LCD:          ACALL   DELAY_100ms
142:                    CLR     LCD_RS
143:
144:                    MOV     P1,#00111000B
145:                    ACALL   LCD_CLK
146:                    ACALL   DELAY_10ms
147:
148:                    MOV     P1,#00111000B
149:                    ACALL   LCD_CLK
150:
151:                    ACALL   LCD_OFF
152:
153:                    ACALL   LCD_CLR
154:
155:                    MOV     P1,#00000110B
156:                    ACALL   LCD_CLK
157:
158:                    ACALL   LCD_HOME
159:
160: ;-------------------------------------------------------------------------
     --
161: ; LCD Clear Display
162: ;-------------------------------------------------------------------------
     --
163: LCD_CLR:           CLR     LCD_RS
164:                    MOV     P1,#00000001B
165:                    ACALL   LCD_CLK
166:                    RET
167:
168: ;-------------------------------------------------------------------------
     --
169: ; LCD Return Home
170: ;-------------------------------------------------------------------------
     --
171: LCD_HOME:          CLR     LCD_RS
172:                    MOV     P1,#00000010B
173:                    ACALL   LCD_CLK
174:                    RET
175:
176: ;-------------------------------------------------------------------------
     --
177: ; LCD Display Off
178: ;-------------------------------------------------------------------------
     --
179: LCD_OFF:           CLR     LCD_RS
180:                    MOV     P1,#00001000B
181:                    ACALL   LCD_CLK
182:                    RET
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
183:
184: ;-------------------------------------------------------------------------
     --
185: ; LCD Clk
186: ;-------------------------------------------------------------------------
     --
187: LCD_CLK:          SETB    LCD_EN
188:                   ACALL   LCD_DELAY
189:                   CLR     LCD_EN
190:                   ACALL   LCD_DELAY
191:                   RET
192:
193: ;-------------------------------------------------------------------------
     --
194: ; LCD Display On
195: ;-------------------------------------------------------------------------
     --
196: LCD_ON:           CLR     LCD_RS
197:                   MOV     P1,#00001100B
198:                   ACALL   LCD_CLK
199:                   RET
200:
201: ;-------------------------------------------------------------------------
     --
202: ; LCD Cursor On
203: ;-------------------------------------------------------------------------
     --
204: LCD_BLINK:        CLR     LCD_RS
205:                   MOV     P1,#00001111B
206:                   ACALL   LCD_CLK
207:                   RET
208:
209: ;-------------------------------------------------------------------------
     --
210: ; LCD Left Shift Display
211: ;-------------------------------------------------------------------------
     --
212: LCD_LSHF:         CLR     LCD_RS
213:                   MOV     P1,#00011000B
214:                   ACALL   LCD_CLK
215:                   RET
216:
217: ;-------------------------------------------------------------------------
     --
218: ; LCD Right Shift Display
219: ;-------------------------------------------------------------------------
     --
220: LCD_RSHF:         CLR     LCD_RS
221:                   MOV     P1,#00011100B
222:                   ACALL   LCD_CLK
223:                   RET
224:
225: ;-------------------------------------------------------------------------
     --
226: ; Set LCD Address
227: ; I/P:      LCD_ADDR
228: ;-------------------------------------------------------------------------
     --
229: SET_ADDR_LCD:     CLR     LCD_RS
230:                   MOV     A,LCD_ADDR
231:                   SETB    ACC.7
232:                   MOV     P1,A
233:                   ACALL   LCD_CLK
234:                   RET
235:
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
236: ;-------------------------------------------------------------------------
     --
237: ; Write Character to show LCD
238: ; I/P:      LCD_DATA
239: ;-------------------------------------------------------------------------
     --
240: WRCHAR_LCD:      SETB    LCD_RS
241:                  MOV     P1,LCD_DATA
242:                  ACALL   LCD_CLK
243:                  ACALL   LCD_ON
244:                  RET
245:
246: ;-------------------------------------------------------------------------
     --
247: ; Write Line of 8 Character from ROM
248: ; I/P:      DPTR : Locate ROM Address
249: ;-------------------------------------------------------------------------
     --
250: WRLINE_LCD:      MOV     R0,#0
251: WRLINE_LCD_1:    SETB    LCD_RS
252:                  CLR     A
253:                  MOVC    A,@A+DPTR
254:                  MOV     P1,A
255:                  ACALL   LCD_CLK
256:                  INC     DPTR
257:                  INC     R0
258:                  CJNE    R0,#8,WRLINE_LCD_1
259:                  ACALL   LCD_ON
260:                  RET
261:
262:  ;-------------------------------------------------------------------------
     ---
263:                          ;KEY PAD SECTION
264:  ;-------------------------------------------------------------------------
     ---
265:
266:
267:
268:
269: GET_KEY_DATA:    MOV     P2,#11111111B
270:
271:
272: LOOP:           ACALL   GET_KPAD        ; Get Keypad Data
273:                  MOV     A,KPAD_DATA     ;
274:                  CJNE    A,#00H,BACK
275:                  AJMP    LOOP
276: BACK:           RET
277: ;-------------------------------------------------------------------------
     --
278: ; Keypad Scan key Subroutine
279: ;-------------------------------------------------------------------------
     --
280: GET_KPAD:       MOV     P2,#0FFH
281:                  MOV     KPAD_DATA,#0
282:
283: CHK_COL0:       CLR     KPAD_COL0       ; Begin Scan Column 0
284:                  MOV     A,P2
285:                  ANL     A,#00FH
286:                  CJNE    A,#00FH,COL0_DETECT
287:                  AJMP    CHK_COL1
288:
289: COL0_DETECT:    MOV     KPAD_DATA,#01
290:                  AJMP    GET_ROW
291:
292: CHK_COL1:       SETB    KPAD_COL0
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
293:                CLR     KPAD_COL1       ; Begin Scan Column 1
294:                MOV     A,P2
295:                ANL     A,#00FH
296:                CJNE    A,#00FH,COL1_DETECT
297:                AJMP    CHK_COL2
298:
299: COL1_DETECT:    MOV     KPAD_DATA,#02
300:                AJMP    GET_ROW         ; Jump to get row value
301:
302: CHK_COL2:       SETB    KPAD_COL1
303:                CLR     KPAD_COL2       ; Begin Scan Column 2
304:                MOV     A,P2
305:                ANL     A,#00FH
306:                CJNE    A,#00FH,COL2_DETECT
307:                RET
308:
309: COL2_DETECT:    MOV     KPAD_DATA,#03
310:
311: GET_ROW:        CLR     KPAD_COL0
312:                CLR     KPAD_COL1
313:                CLR     KPAD_COL2
314:
315:                JB      KPAD_ROW0,CHK_ROW1
316:                RET
317:
318: CHK_ROW1:       JB      KPAD_ROW1,CHK_ROW2
319:                MOV     A,KPAD_DATA
320:                ADD     A,#3
321:                MOV     KPAD_DATA,A
322:                RET
323:
324: CHK_ROW2:       JB      KPAD_ROW2,CHK_ROW3
325:                MOV     A,KPAD_DATA
326:                ADD     A,#6
327:                MOV     KPAD_DATA,A
328:                RET
329:
330: CHK_ROW3:       MOV     A,KPAD_DATA
331:                ADD     A,#9
332:                MOV     KPAD_DATA,A
333:                RET
334:
335: ;-------------------------------------------------------------------------
    --
336:
337: ;-------------------------------------------------------------------------
    --
338: ;  Delay time LCD_DELAY, 10m, 100m, 1s
339: ;-------------------------------------------------------------------------
    --
340: LCD_DELAY:      MOV     R7,#002
341: LCD_DELAY_1:    MOV     R6,#0E6H
342: LCD_DELAY_2:    NOP
343:                NOP
344:                DJNZ    R6,LCD_DELAY_2
345:                DJNZ    R7,LCD_DELAY_1
346:                RET
347:
348: DELAY_10ms:     MOV     R7,#010
349: DELAY_10ms_1:   MOV     R6,#0E6H
350: DELAY_10ms_2:   NOP
351:                NOP
352:                DJNZ    R6,DELAY_10ms_2
353:                DJNZ    R7,DELAY_10ms_1
354:                RET
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
355:
356: DELAY_100ms:     MOV     R7,#100
357: DELAY_100ms_1:   MOV     R6,#0E6H
358: DELAY_100ms_2:   NOP
359:                  NOP
360:                  DJNZ    R6,DELAY_100ms_2
361:                  DJNZ    R7,DELAY_100ms_1
362:                  RET
363:
364: DELAY_1s:        MOV     R5,#100
365: DELAY_1s_1:      ACALL   DELAY_10ms
366:                  DJNZ    R5,DELAY_1s_1
367:                  RET
368:
369: ;-------------------------------------------------------------------------
     --
370: ;Define Constant
371: ;-------------------------------------------------------------------------
     --
372: ;                       01234567
373: TITLE_1:         DB      'GET__POS'
374: TITLE_2:         DB      'ION     '
375:
376: TITLE_3:         DB      'POSITION'
377: TITLE_4:         DB      '(   ,   ) '
378:
379:
380:
381: ;*****************************************************************
382:                  ;RUNNING
383: ;*****************************************************************

384: RUN:             MOV     ERROR_R,#00H
385:                  MOV     ERROR_L,#00H
386:                  MOV     ER_R,#00H
387:                  MOV     ER_L,#00H
388:                  MOV     ER_RR,#00H
389:                  MOV     ER_RL,#00H
390:                  MOV     CASE_Y,#00H
391:                  MOV     DATA_LX,#00H
392:                  MOV     DATA_LY,#00H
393:                  MOV     SPD,#7FH
394:                  MOV     SPD1F,#7FH
395:                  MOV     30H,#00H
396:                  MOV     31H,#00H
397:                  MOV     32H,#00H
398:
399:
400:                  MOV     A,DATA_X
401:                  MOV     B,#16H
402:                  MUL     AB              ;Multiply With 22 Pulse
403:                  MOV     DATA_LX,A
404:
405:                  MOV     A,DATA_Y
406:                  MOV     B,#16H
407:                  MUL     AB
408:                  MOV     DATA_LY,A
409:
410:                  ACALL   STOP
411:                  ACALL   DELAY_1S
412:                  ACALL   DELAY_1S
413:                  ACALL   DELAY_1S
414:                  MOV     30H,DATA_LY
415:
416:                  MOV     TMOD,#01010101B         ;Set Counter Mode 1
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
417:                    MOV     TH0,#00H
418:                    MOV     TL0,#00H
419:                    MOV     TH1,#00H
420:                    MOV     TL1,#00H
421:                    SETB    TR0
422:                    SETB    TR1
423:                    ACALL   MODE8255                ;Set Mode 8255
424:
425:
426: CK:
427:                    MOV     TMOD,#01010101B
428:                    SETB    TR0
429:                    SETB    TR1
430:                    ACALL   GO                      ;Run
431:                    MOV     A,30H
432:                    CJNE    A,TL0,THEN
433: ST:                ACALL   STOP
434:                    ACALL   DELAY_1S
435:                    ACALL   DELAY_1S
436:                    AJMP    CHANGE_DI
437: THEN:               CLR     C
438:                    SUBB    A,TL0
439:                    JC      ST
440:                    ACALL   CHK_OB                  ;Check Object
441:                    AJMP    CK
442:
443: ;*****************************************************
444:
445:
446: CHK_OB:            MOV     R0,#0F2H
447:                    MOVX    A,@R0
448:                    ANL     A,#00000111B
449:                    CJNE    A,#07H,FIND
450:                    AJMP    REE
451:
452: FIND:              CJNE    A,#06H,FIND_R
453:
454: FIND_L:            ACALL   STOP                    ;Turn Left Procedure
455:                    ACALL   DELAY_1S
456:                    ACALL   DELAY_1S
457:
458:                    ACALL   TURN_L
459:                    ACALL   DELAY_1S
460:                    ACALL   DELAY_1S
461:
462:                    ACALL   GO1FEET
463:                    ACALL   STOP
464:                    ACALL   DELAY_1S
465:
466:
467:                    MOV     A,#11H
468:                    ADD     A,ERROR_L
469:                    MOV     ERROR_L,A
470:                    ACALL   DELAY_1S
471:                    ACALL   TURN_R
472:                    ACALL   STOP
473:                    ACALL   DELAY_1S
474:                    ACALL   DELAY_1S
475:
476:                    RET
477:
478: FIND_R:            ACALL   STOP                    ;Turn Right Procedure
479:                    ACALL   DELAY_1S
480:                    ACALL   DELAY_1S
481:                    ACALL   TURN_R
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งไม่อนุญาตให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
482:                    ACALL   DELAY_1S
483:                    ACALL   DELAY_1S
484:                    ACALL   GO1FEET
485:                    ACALL   STOP
486:                    ACALL   DELAY_1S
487:                    MOV     A,#11H
488:                    ADD     A,ERROR_R
489:                    MOV     ERROR_R,A
490:                    ACALL   DELAY_1S
491:                    ACALL   TURN_L
492:                    ACALL   STOP
493:                    ACALL   DELAY_1S
494:                    ACALL   DELAY_1S
495:
496:
497: REE:               RET
498:
499: ;*****************************************************
500:                                ;X  DIRECTION
501: ;*****************************************************
502:
503: CHANGE_DI:         ACALL   CHK_CASE_Y                  ;Check Case
504:
505:
506:                    MOV     A,CASE_Y
507:                    CJNE    A,#01H,C_2345
508:                    ACALL   CASE_1
509:                    AJMP    DI_X
510:
511: C_2345:            CJNE    A,#02H,C_345
512:                    ACALL   CASE_2
513:                    AJMP    DI_X
514:
515: C_345:             CJNE    A,#03H,C_45
516:                    ACALL   CASE_3
517:                    AJMP    DI_X
518:
519: C_45:              CJNE    A,#04H,C_5
520:                    AJMP    STX
521: C_5:               ACALL   CASE_5
522:
523: DI_X:              MOV     TL0,#00H
524:                    MOV     TL1,#00H
525: E:                 MOV     TMOD,#01010101B
526:                    SETB    TR0
527:                    SETB    TR1
528:                    ACALL   GO
529:                    MOV     A,31H
530:                    CJNE    A,TL0,THENX
531: STX:               ACALL   STOP
532:                    AJMP    EN
533: THENX:             CLR     C
534:                    SUBB    A,TL0
535:                    JC      STX
536:                    ACALL   CHK_OBX
537:                    AJMP    E
538:
539: EN:                AJMP    EN
540:
541: ;*****************************************************
542:
543: CHK_CASE_Y:        MOV     A,ERROR_R
544:                    CLR     C
545:                    SUBB    A,ERROR_L
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น รวมทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
546:                    JC      CA_Y1
547:                    MOV     ER_R,A
548:                    CJNE    A,#00H,CA_Y345
549:
550: CA_Y2:             MOV     CASE_Y,#02H
551:                    RET
552: CA_Y1:             MOV     A,ERROR_L
553:                    CLR     C
554:                    SUBB    A,ERROR_R
555:                    MOV     ER_L,A
556:                    MOV     CASE_Y,#01H
557:                    RET
558:
559: CA_Y345:           MOV     A,ER_R
560:                    CLR     C
561:                    SUBB    A,DATA_LX
562:                    JC      CA_Y3
563:                    MOV     ER_RR,A
564:                    CJNE    A,#00H,CA_Y5
565:
566: CA_Y4:             MOV     CASE_Y,#04H
567:                    RET
568:
569: CA_Y3:             MOV     A,DATA_LX
570:                    CLR     C
571:                    SUBB    A,ER_R
572:                    MOV     ER_RL,A
573:                    MOV     CASE_Y,#03H
574:                    RET
575:
576: CA_Y5:             MOV     CASE_Y,#05H
577:                    RET
578:
579: ;*************************************************************
580: ;                   Run By Case Procedure
581: ;*************************************************************
582: CASE_1:            MOV     A,ER_L
583:                    ADD     A,DATA_LX
584:                    MOV     31H,A
585:                    ACALL   TURN_R
586:                    RET
587:
588: CASE_2:            MOV     31H,DATA_LX
589:                    ACALL   TURN_R
590:                    RET
591:
592: CASE_3:            MOV     31H,ER_RL
593:                    ACALL   TURN_R
594:                    RET
595:
596: CASE_5:            MOV     31H,ER_RR
597:                    ACALL   TURN_L
598:                    RET
599:
600: ;*****************************************************************
601: ;                     Delay
602: ;*****************************************************************
603:
604: D_10MS:            MOV     R7,#010
605: D_1:               MOV     R6,#0E6H
606: D_2:               NOP
607:                    NOP
608:                    DJNZ    R6,D_2
609:                    DJNZ    R7,D_1
610:                    RET
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
611: DELAY_1S:      MOV     R5,#100
612: S_1:           ACALL   D_10MS
613:                DJNZ    R5,S_1
614:                RET
615:
616: ;*****************************************************************
617: ;                       Go Streightly
618: ;*****************************************************************

619: GO:
620:                MOV     R0,#0F3H
621:
622:
623:                MOV     A,#00001111B
624:                MOVX    @R0,A
625:
626:                MOV     A,#00001100B
627:                MOVX    @R0,A
628:
629:                MOV     A,#00001011B
630:                MOVX    @R0,A
631:
632:                MOV     A,#00001000B
633:                MOVX    @R0,A
634:
635:
636:                MOV     A,#04H
637:                MOV     R0,#0F0H
638:                MOVX    @R0,A
639:
640:                MOV     A,SPD
641:                MOV     R0,#0F1H
642:                MOVX    @R0,A
643: COMPARE:       MOV     A,TL0
644:                CJNE    A,TL1,DEC_INC
645:                RET
646:
647: DEC_INC:       MOV     A,TL1
648:                CLR     C
649:                SUBB    A,TL0
650:                JC      DECSPD
651:                RET
652: INCSPD:        MOV     A,SPD
653:                INC     A
654:                MOV     SPD,A
655:                MOV     R0,#0F1H
656:                MOVX    @R0,A
657:                AJMP    COMPARE
658: DECSPD:        MOV     A,SPD
659:                CJNE    A,#07H,REY
660:                RET
661: REY:           DEC     SPD
662:                MOV     A,SPD
663:                MOV     R0,#0F1H
664:                MOVX    @R0,A
665:                AJMP    COMPARE
666: ;*****************************************************************

667: ;                       Turn Right
668: ;*****************************************************************
```

669: เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
670: TURN_R:
671: ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
672:                MOV     32H,TL0
                    ACALL   STOP
```

```
673:                    CLR     TR0
674:                    CLR     TR1
675:                    CLR     TF0
676:                    MOV     TL0,#0F9H
677:                    MOV     TL1,#0F9H
678:                    MOV     TMOD,#66H
679:                    SETB    TR0
680:                    SETB    TR1
681:                    MOV     R0,#0F3H
682:
683:                    MOV     A,#00001111B
684:                    MOVX    @R0,A
685:
686:                    MOV     A,#00001101B
687:                    MOVX    @R0,A
688:
689:                    MOV     A,#00001011B
690:                    MOVX    @R0,A
691:
692:                    MOV     A,#00001000B
693:                    MOVX    @R0,A
694:
695:
696:                    MOV     A,#05H
697:                    MOV     R0,#0F0H
698:                    MOVX    @R0,A
699:
700:                    MOV     A,#07H
701:                    MOV     R0,#0F1H
702:                    MOVX    @R0,A
703: CK_TF0:
704:                    JNB     TF0,CK_TF0
705:                    ACALL   STOP
706:                    CLR     TR0
707:                    CLR     TR1
708:                    MOV     TL0,32H
709:                    RET
710: STOP:              MOV     A,#81H
711:                    MOV     R0,#0F3H
712:                    MOVX    @R0,A
713:
714:                    MOV     A,#00001110B
715:                    MOVX    @R0,A
716:
717:                    MOV     A,#00001010B
718:                    MOVX    @R0,A
719:
720:                    RET
721: ;*************************************************************
722: ;                       Mode 8255
723: ;*************************************************************

724: MODE8255:          MOV     A,#81H
725:                    MOV     R0,#0F3H
726:                    MOVX    @R0,A
727:                    RET
728: ;*********************************************************
729: ;                       Go 1 Feet
730: ;*********************************************************
731: GO1FEET:           MOV     32H,TL0
732:                    ACALL   STOP
733:                    CLR     TR0
734:                    CLR     TR1
735:                    CLR     TF0
736:                    MOV     TL0,#0EFH
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
737:                    MOV     TL1,#0EFH
738:                    MOV     TMOD,#66H
739:                    SETB    TR0
740:                    SETB    TR1
741:                    MOV     R0,#0F3H
742:
743:                    MOV     A,#00001111B
744:                    MOVX    @R0,A
745:
746:                    MOV     A,#00001100B
747:                    MOVX    @R0,A
748:
749:                    MOV     A,#00001011B
750:                    MOVX    @R0,A
751:
752:                    MOV     A,#00001000B
753:                    MOVX    @R0,A
754:
755:
756:                    MOV     A,#04H
757:                    MOV     R0,#0F0H
758:                    MOVX    @R0,A
759:
760:                    MOV     A,SPD1F
761:                    MOV     R0,#0F1H
762:                    MOVX    @R0,A
763: CK_TF1:
764: COMPARE1F:         MOV     A,TL0
765:                    CJNE    A,TL1,DEC_INC1F
766:                    AJMP    DU
767:
768: DEC_INC1F:         MOV     A,TL1
769:                    CLR     C
770:                    SUBB    A,TL0
771:                    JC      DECSPD1F
772:                    AJMP    DU
773: INCSPD1F:          MOV     A,SPD1F
774:                    INC     A
775:                    MOV     SPD1F,A
776:                    MOV     R0,#0F1H
777:                    MOVX    @R0,A
778:                    AJMP    COMPARE1F
779: DECSPD1F:         MOV     A,SPD1F
780:                    CJNE    A,#07H,REY1F
781:                    AJMP    DU
782: REY1F:             DEC     SPD1F
783:                    MOV     A,SPD1F
784:                    MOV     R0,#0F1H
785:                    MOVX    @R0,A
786:                    AJMP    COMPARE1F
787:
788: DU:                JNB     TF0,CK_TF1
789:                    ACALL   STOP
790:                    CLR     TR0
791:                    CLR     TR1
792:                    MOV     TL0,32H
793:                    RET
794: ;*************************************************************
795: ;                       Turn Left
796: ;*************************************************************
797: TURN_L:            MOV     32H,TL0
798:                    ACALL   STOP
799:                    CLR     TR0
800:                    CLR     TR1
801:                    CLR     TF0
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
802:                    MOV     TL0,#0F9H
803:                    MOV     TL1,#0F9H
804:                    MOV     TMOD,#66H
805:                    SETB    TR0
806:                    SETB    TR1
807:                    MOV     R0,#0F3H
808:
809:                    MOV     A,#00001111B
810:                    MOVX    @R0,A
811:
812:                    MOV     A,#00001100B
813:                    MOVX    @R0,A
814:
815:                    MOV     A,#00001011B
816:                    MOVX    @R0,A
817:
818:                    MOV     A,#00001001B
819:                    MOVX    @R0,A
820:
821:
822:                    MOV     A,#05H
823:                    MOV     R0,#0F0H
824:                    MOVX    @R0,A
825:
826:                    MOV     A,#07H
827:                    MOV     R0,#0F1H
828:                    MOVX    @R0,A
829: CK_TF2:            JNB     TF0,CK_TF2
830:                    ACALL   STOP
831:                    CLR     TR0
832:                    CLR     TR1
833:                    MOV     TL0,32H
834:                    RET
835: ;*****************************************************
836: ;                  Check Object In X Position
837: ;*****************************************************
838: CHK_OBX:           MOV     R0,#0F2H
839:                    MOVX    A,@R0
840:                    ANL     A,#00000111B
841:                    CJNE    A,#07H,FIND_X
842:                    AJMP    REEX
843:
844: FIND_X:            CJNE    A,#06H,FIND_RX
845:
846: FIND_LX:           ACALL   STOP
847:                    ACALL   DELAY_1S
848:                    ACALL   DELAY_1S
849:
850:                    ACALL   TURN_L
851:                    ACALL   DELAY_1S
852:                    ACALL   DELAY_1S
853:
854:                    ACALL   GO1FEET
855:                    ACALL   STOP
856:                    ACALL   DELAY_1S
857:
858:
859:                    ACALL   TURN_R
860:                    ACALL   STOP
861:                    ACALL   DELAY_1S
862:                    ACALL   DELAY_1S
863:
864:                    RET
865:
866: FIND_RX:           ACALL   STOP
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
867:                    ACALL   DELAY_1S
868:                    ACALL   DELAY_1S
869:                    ACALL   TURN_R
870:                    ACALL   DELAY_1S
871:                    ACALL   DELAY_1S
872:                    ACALL   GO1FEET
873:                    ACALL   STOP
874:                    ACALL   DELAY_1S
875:
876:                    ACALL   DELAY_1S
877:                    ACALL   TURN_L
878:                    ACALL   STOP
879:                    ACALL   DELAY_1S
880:                    ACALL   DELAY_1S
881:
882:
883: REEX:               RET
884:
885: ; END*************************************************************************
     *************
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

TL494, NCV494

SWITCHMODE™ Pulse Width Modulation Control Circuit

The TL494 is a fixed frequency, pulse width modulation control circuit designed primarily for SWITCHMODE power supply control.

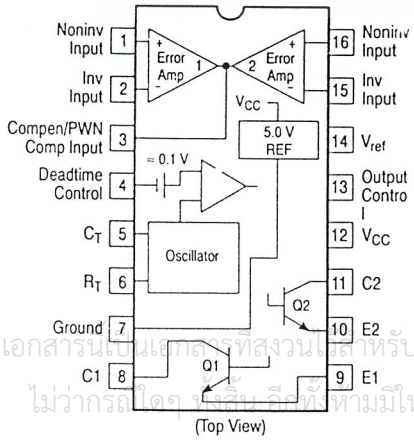
- Complete Pulse Width Modulation Control Circuitry
- On–Chip Oscillator with Master or Slave Operation
- On–Chip Error Amplifiers
- On–Chip 5.0 V Reference
- Adjustable Deadtime Control
- Uncommitted Output Transistors Rated to 500 mA Source or Sink
- Output Control for Push–Pull or Single–Ended Operation
- Undervoltage Lockout

MAXIMUM RATINGS (Full operating ambient temperature range applies, unless otherwise noted.)

Rating	Symbol	Value	Unit
Power Supply Voltage	V_{CC}	42	V
Collector Output Voltage	V_{C1}, V_{C2}	42	V
Collector Output Current (Each transistor) (Note 1)	I_{C1}, I_{C2}	500	mA
Amplifier Input Voltage Range	V_{IR}	–0.3 to +42	V
Power Dissipation @ $T_A \le 45°C$	P_D	1000	mW
Thermal Resistance, Junction–to–Ambient	$R_{\theta JA}$	80	°C/W
Operating Junction Temperature	T_J	125	°C
Storage Temperature Range	T_{stg}	–55 to +125	°C
Operating Ambient Temperature Range TL494B TL494C TL494I NCV494B	T_A	 –40 to +125 0 to +70 –40 to +85 –40 to +125	°C
Derating Ambient Temperature	T_A	45	°C

1. Maximum thermal limits must be observed.

PIN CONNECTIONS



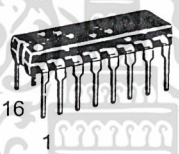
(Top View)

ON Semiconductor®

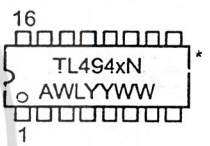
http://onsemi.com

MARKING DIAGRAMS

SO–16
D SUFFIX
CASE 751B



PDIP–16
N SUFFIX
CASE 648



x = B, C or I
A = Assembly Location
WL, L = Wafer Lot
YY, Y = Year
WW, W = Work Week

*This marking diagram also applies to NCV494.

ORDERING INFORMATION

Device	Package	Shipping
TL494BD	SO–16	48 Units/Rail
TL494BDR2	SO–16	2500 Tape & Reel
TL494CD	SO–16	48 Units/Rail
TL494CDR2	SO–16	2500 Tape & Reel
TL494CN	PDIP–16	25 Units/Rail
TL494IN	PDIP–16	25 Units/Rail
NCV494BDR2*	SO–16	2500 Tape & Reel

*NCV494: T_{low} = –40°C, T_{high} = +125°C. Guaranteed by design. NCV prefix is for automotive and other applications requiring site and change control.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

RECOMMENDED OPERATING CONDITIONS

Characteristics	Symbol	Min	Typ	Max	Unit
Power Supply Voltage	V_{CC}	7.0	15	40	V
Collector Output Voltage	V_{C1}, V_{C2}	–	30	40	V
Collector Output Current (Each transistor)	I_{C1}, I_{C2}	–	–	200	mA
Amplified Input Voltage	V_{in}	–0.3	–	V_{CC} – 2.0	V
Current Into Feedback Terminal	I_{fb}	–	–	0.3	mA
Reference Output Current	I_{ref}	–	–	10	mA
Timing Resistor	R_T	1.8	30	500	kΩ
Timing Capacitor	C_T	0.0047	0.001	10	μF
Oscillator Frequency	f_{osc}	1.0	40	200	kHz

ELECTRICAL CHARACTERISTICS (V_{CC} = 15 V, C_T = 0.01 μF, R_T = 12 kΩ, unless otherwise noted.)

For typical values T_A = 25°C, for min/max values T_A is the operating ambient temperature range that applies, unless otherwise noted.

Characteristics	Symbol	Min	Typ	Max	Unit
REFERENCE SECTION					
Reference Voltage (I_O = 1.0 mA)	V_{ref}	4.75	5.0	5.25	V
Line Regulation (V_{CC} = 7.0 V to 40 V)	Reg_{line}	–	2.0	25	mV
Load Regulation (I_O = 1.0 mA to 10 mA)	Reg_{load}	–	3.0	15	mV
Short Circuit Output Current (V_{ref} = 0 V)	I_{SC}	15	35	75	mA
OUTPUT SECTION					
Collector Off–State Current (V_{CC} = 40 V, V_{CE} = 40 V)	$I_{C(off)}$	–	2.0	100	μA
Emitter Off–State Current V_{CC} = 40 V, V_C = 40 V, V_E = 0 V)	$I_{E(off)}$	–	–	–100	μA
Collector–Emitter Saturation Voltage (Note 2)					V
Common–Emitter (V_E = 0 V, I_C = 200 mA)	$V_{sat(C)}$	–	1.1	1.3	
Emitter–Follower (V_C = 15 V, I_E = –200 mA)	$V_{sat(E)}$	–	1.5	2.5	
Output Control Pin Current					
Low State ($V_{OC} \le$ 0.4 V)	I_{OCL}	–	10	–	μA
High State (V_{OC} = V_{ref})	I_{OCH}	–	0.2	3.5	mA
Output Voltage Rise Time	t_r				ns
Common–Emitter (See Figure 12)		–	100	200	
Emitter–Follower (See Figure 13)		–	100	200	
Output Voltage Fall Time	t_f				ns
Common–Emitter (See Figure 12)		–	25	100	
Emitter–Follower (See Figure 13)		–	40	100	

2. Low duty cycle pulse techniques are used during test to maintain junction temperature as close to ambient temperature as possible.

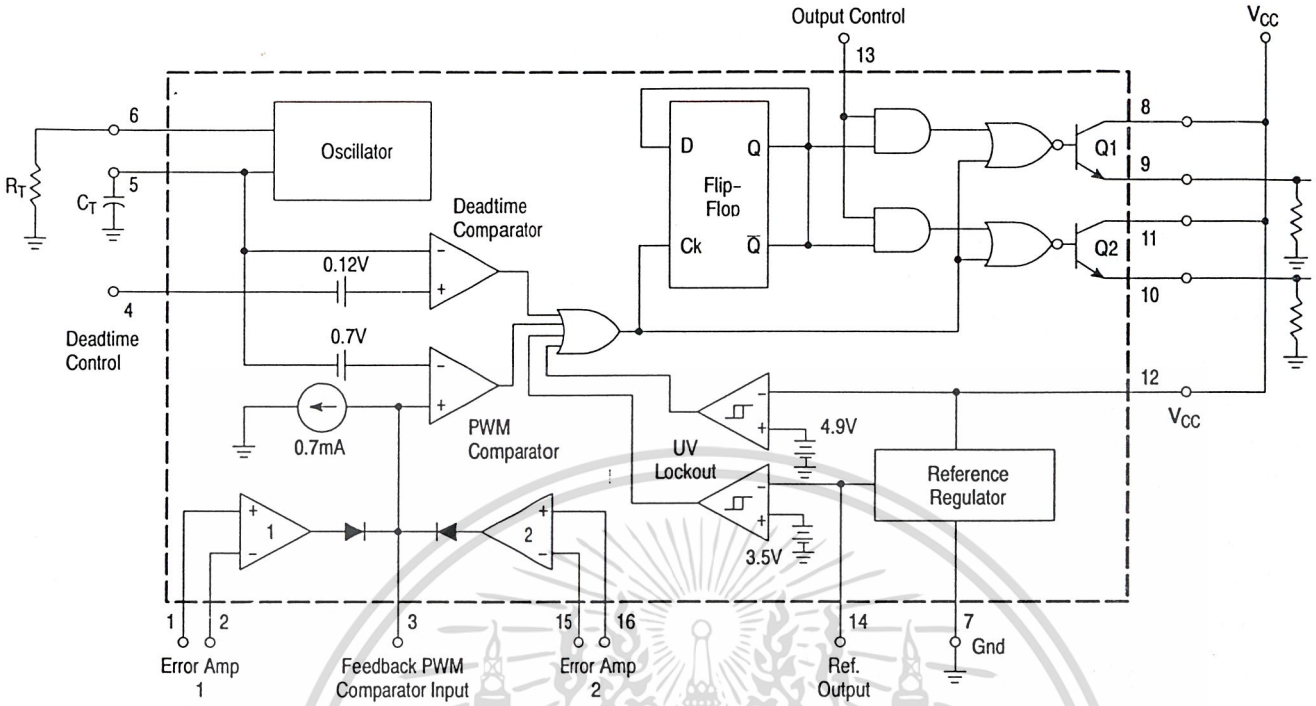
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ELECTRICAL CHARACTERISTICS (V_{CC} = 15 V, C_T = 0.01 μF, R_T = 12 kΩ, unless otherwise noted.)
For typical values T_A = 25°C, for min/max values T_A is the operating ambient temperature range that applies, unless otherwise noted.

Characteristics	Symbol	Min	Typ	Max	Unit
ERROR AMPLIFIER SECTION					
Input Offset Voltage ($V_{O\ (Pin\ 3)}$ = 2.5 V)	V_{IO}	–	2.0	10	mV
Input Offset Current ($V_{O\ (Pin\ 3)}$ = 2.5 V)	I_{IO}	–	5.0	250	nA
Input Bias Current ($V_{O\ (Pin\ 3)}$ = 2.5 V)	I_{IB}	–	–0.1	–1.0	μA
Input Common Mode Voltage Range (V_{CC} = 40 V, T_A = 25°C)	V_{ICR}	–0.3 to V_{CC}–2.0			V
Open Loop Voltage Gain (ΔV_O = 3.0 V, V_O = 0.5 V to 3.5 V, R_L = 2.0 kΩ)	A_{VOL}	70	95	–	dB
Unity–Gain Crossover Frequency (V_O = 0.5 V to 3.5 V, R_L = 2.0 kΩ)	f_{C-}	–	350	–	kHz
Phase Margin at Unity–Gain (V_O = 0.5 V to 3.5 V, R_L = 2.0 kΩ)	ϕ_m	–	65	–	deg.
Common Mode Rejection Ratio (V_{CC} = 40 V)	CMRR	65	90	–	dB
Power Supply Rejection Ratio (ΔV_{CC} = 33 V, V_O = 2.5 V, R_L = 2.0 kΩ)	PSRR	–	100	–	dB
Output Sink Current ($V_{O\ (Pin\ 3)}$ = 0.7 V)	I_{O-}	0.3	0.7	–	mA
Output Source Current ($V_{O\ (Pin\ 3)}$ = 3.5 V)	I_O^+	2.0	–4.0	–	mA
PWM COMPARATOR SECTION (Test Circuit Figure 11)					
Input Threshold Voltage (Zero Duty Cycle)	V_{TH}	–	2.5	4.5	V
Input Sink Current ($V_{(Pin\ 3)}$ = 0.7 V)	I_{I-}	0.3	0.7	–	mA
DEADTIME CONTROL SECTION (Test Circuit Figure 11)					
Input Bias Current (Pin 4) ($V_{Pin\ 4}$ = 0 V to 5.25 V)	$I_{IB\ (DT)}$	–	–2.0	–10	μA
Maximum Duty Cycle, Each Output, Push–Pull Mode ($V_{Pin\ 4}$ = 0 V, C_T = 0.01 μF, R_T = 12 kΩ) ($V_{Pin\ 4}$ = 0 V, C_T = 0.001 μF, R_T = 30 kΩ)	DC_{max}	 45 –	 48 45	 50 50	%
Input Threshold Voltage (Pin 4) (Zero Duty Cycle) (Maximum Duty Cycle)	V_{th}	 – 0	 2.8 –	 3.3 –	V
OSCILLATOR SECTION					
Frequency (C_T = 0.001 μF, R_T = 30 kΩ)	f_{osc}	–	40	–	kHz
Standard Deviation of Frequency* (C_T = 0.001 μF, R_T = 30 kΩ)	σf_{osc}	–	3.0	–	%
Frequency Change with Voltage (V_{CC} = 7.0 V to 40 V, T_A = 25°C)	Δf_{osc} (ΔV)	–	0.1	–	%
Frequency Change with Temperature ($\Delta T_A = T_{low}$ to T_{high}) (C_T = 0.01 μF, R_T = 12 kΩ)	Δf_{osc} (ΔT)	–	–	12	%
UNDERVOLTAGE LOCKOUT SECTION					
Turn–On Threshold (V_{CC} increasing, I_{ref} = 1.0 mA)	V_{th}	5.5	6.43	7.0	V
TOTAL DEVICE					
Standby Supply Current (Pin 6 at V_{ref}, All other inputs and outputs open) (V_{CC} = 15 V) (V_{CC} = 40 V)	I_{CC}	 – –	 5.5 7.0	 10 15	mA
Average Supply Current (C_T = 0.01 μF, R_T = 12 kΩ, $V_{(Pin\ 4)}$ = 2.0 V) (V_{CC} = 15 V) (See Figure 12)		–	7.0	–	mA

* Standard deviation is a measure of the statistical distribution about the mean as derived from the formula, $\sigma \sqrt{\frac{\sum_{n=1}^{N} (X_n - \bar{X})^2}{N-1}}$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



This device contains 46 active transistors.

Figure 1. Representative Block Diagram

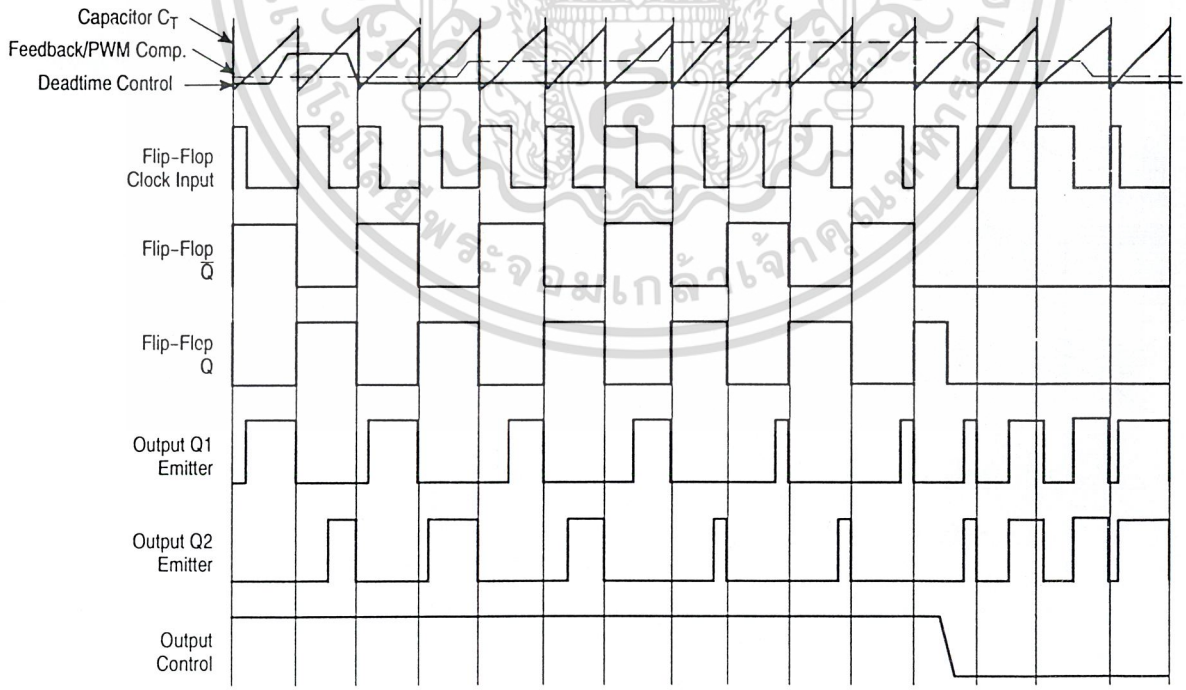


Figure 2. Timing Diagram

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

APPLICATIONS INFORMATION

Description

The TL494 is a fixed–frequency pulse width modulation control circuit, incorporating the primary building blocks required for the control of a switching power supply. (See Figure 1.) An internal–linear sawtooth oscillator is frequency– programmable by two external components, R_T and C_T. The approximate oscillator frequency is determined by:

$$f_{osc} \approx \frac{1.1}{R_T \bullet C_T}$$

For more information refer to Figure 3.

Output pulse width modulation is accomplished by comparison of the positive sawtooth waveform across capacitor C_T to either of two control signals. The NOR gates, which drive output transistors Q1 and Q2, are enabled only when the flip–flop clock–input line is in its low state. This happens only during that portion of time when the sawtooth voltage is greater than the control signals. Therefore, an increase in control–signal amplitude causes a corresponding linear decrease of output pulse width. (Refer to the Timing Diagram shown in Figure 2.)

The control signals are external inputs that can be fed into the deadtime control, the error amplifier inputs, or the feedback input. The deadtime control comparator has an effective 120 mV input offset which limits the minimum output deadtime to approximately the first 4% of the sawtooth–cycle time. This would result in a maximum duty cycle on a given output of 96% with the output control grounded, and 48% with it connected to the reference line. Additional deadtime may be imposed on the output by setting the deadtime–control input to a fixed voltage, ranging between 0 V to 3.3 V.

Functional Table

Input/Output Controls	Output Function	$\frac{f_{out}}{f_{osc}}$ =
Grounded	Single–ended PWM @ Q1 and Q2	1.0
@ V_{ref}	Push–pull Operation	0.5

The pulse width modulator comparator provides a means for the error amplifiers to adjust the output pulse width from the maximum percent on–time, established by the deadtime control input, down to zero, as the voltage at the feedback pin varies from 0.5 V to 3.5 V. Both error amplifiers have a common mode input range from –0.3 V to (V_{CC} – 2V), and may be used to sense power–supply output voltage and current. The error–amplifier outputs are active high and are ORed together at the noninverting input of the pulse–width modulator comparator. With this configuration, the amplifier that demands minimum output on time, dominates control of the loop.

When capacitor C_T is discharged, a positive pulse is generated on the output of the deadtime comparator, which clocks the pulse–steering flip–flop and inhibits the output transistors, Q1 and Q2. With the output–control connected to the reference line, the pulse–steering flip–flop directs the modulated pulses to each of the two output transistors alternately for push–pull operation. The output frequency is equal to half that of the oscillator. Output drive can also be taken from Q1 or Q2, when single–ended operation with a maximum on–time of less than 50% is required. This is desirable when the output transformer has a ringback winding with a catch diode used for snubbing. When higher output–drive currents are required for single–ended operation, Q1 and Q2 may be connected in parallel, and the output–mode pin must be tied to ground to disable the flip–flop. The output frequency will now be equal to that of the oscillator.

The TL494 has an internal 5.0 V reference capable of sourcing up to 10 mA of load current for external bias circuits. The reference has an internal accuracy of ± 5.0% with a typical thermal drift of less than 50 mV over an operating temperature range of 0° to 70°C.

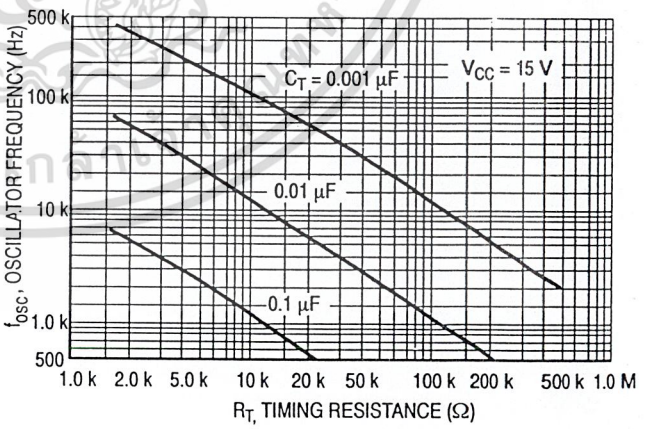


Figure 3. Oscillator Frequency versus Timing Resistance

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

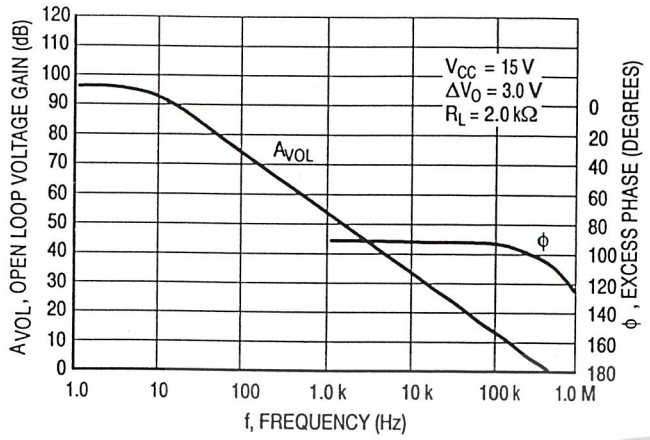


Figure 4. Open Loop Voltage Gain and Phase versus Frequency

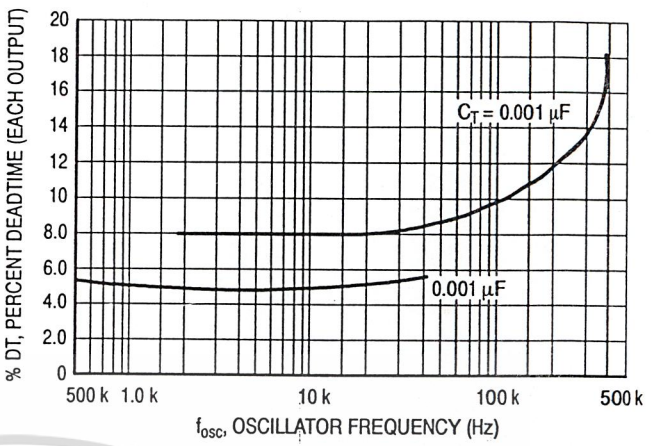


Figure 5. Percent Deadtime versus Oscillator Frequency

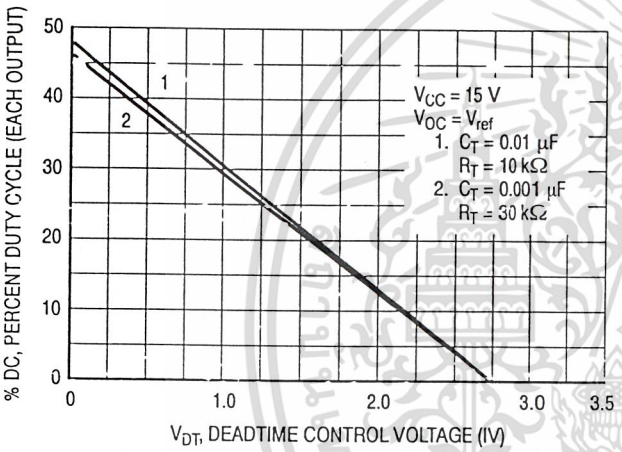


Figure 6. Percent Duty Cycle versus Deadtime Control Voltage

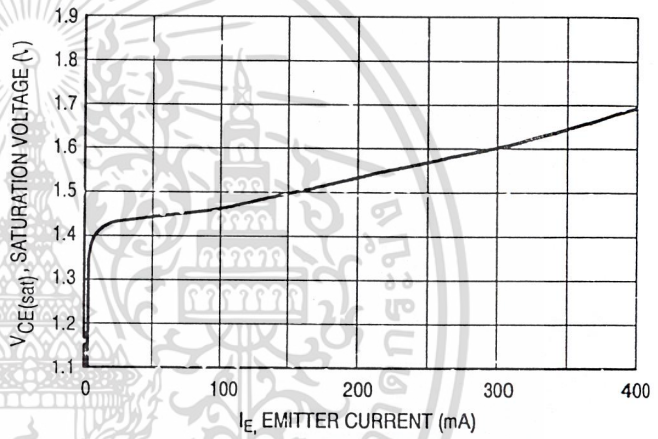


Figure 7. Emitter–Follower Configuration Output Saturation Voltage versus Emitter Current

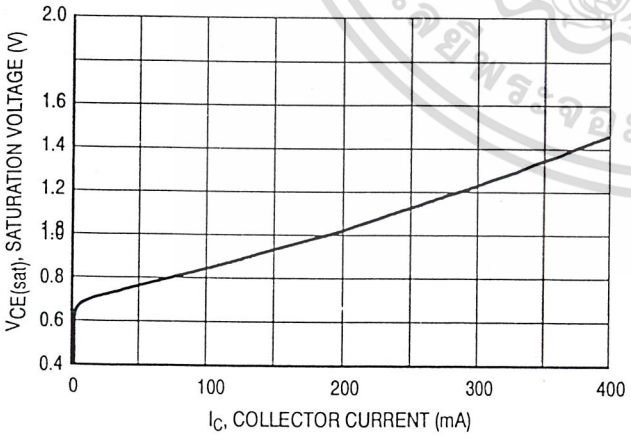


Figure 8. Common–Emitter Configuration Output Saturation Voltage versus Collector Current

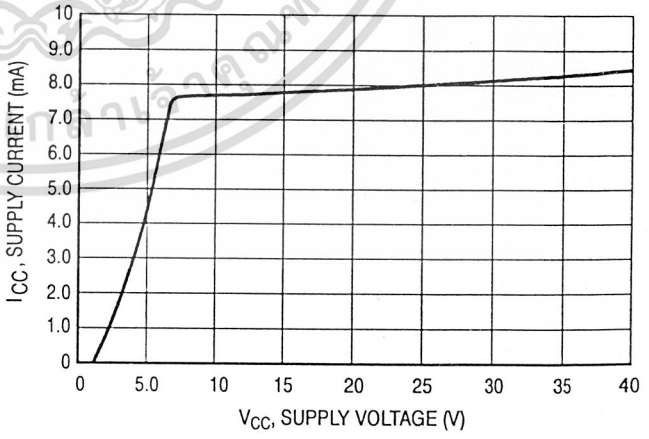


Figure 9. Standby Supply Current versus Supply Voltage

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

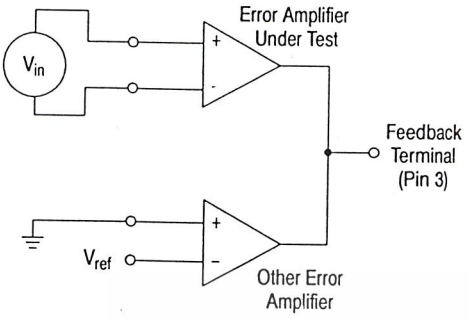


Figure 10. Error–Amplifier Characteristics

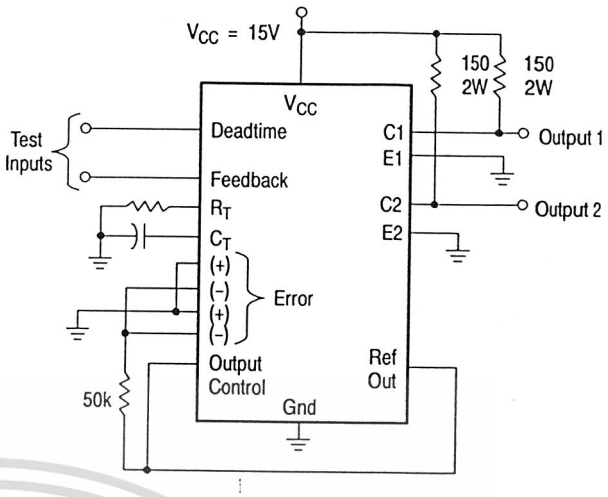


Figure 11. Deadtime and Feedback Control Circuit

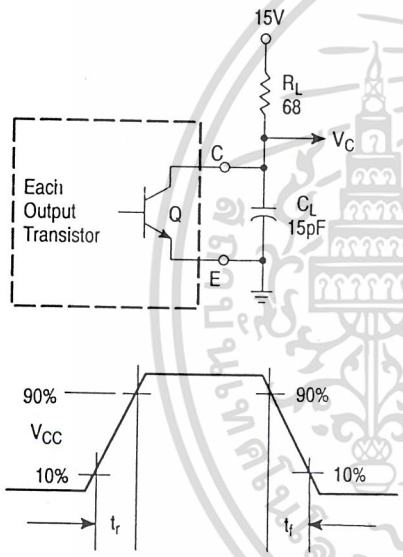


Figure 12. Common–Emitter Configuration Test Circuit and Waveform

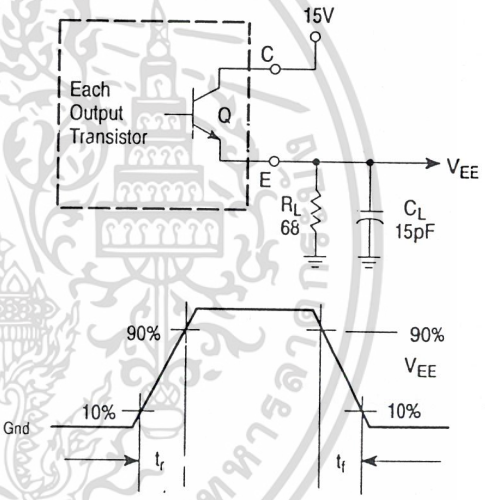


Figure 13. Emitter–Follower Configuration Test Circuit and Waveform

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

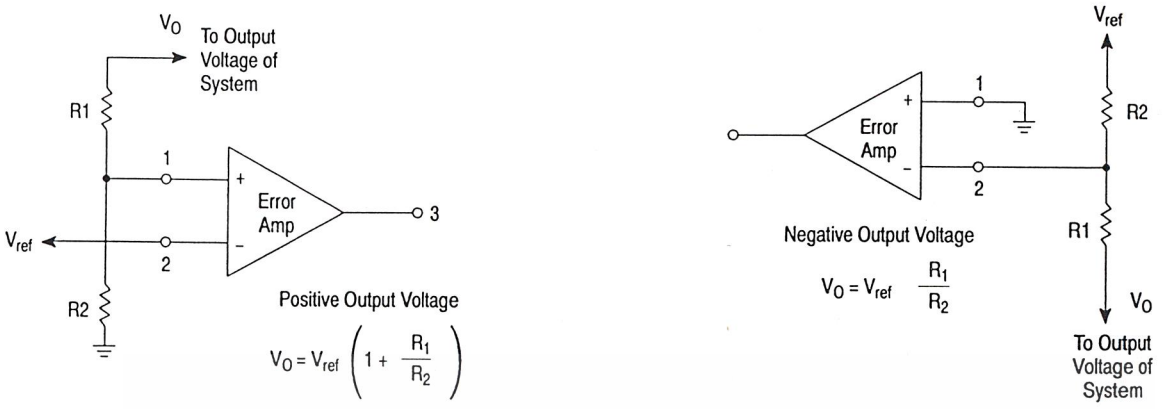


Figure 14. Error–Amplifier Sensing Techniques

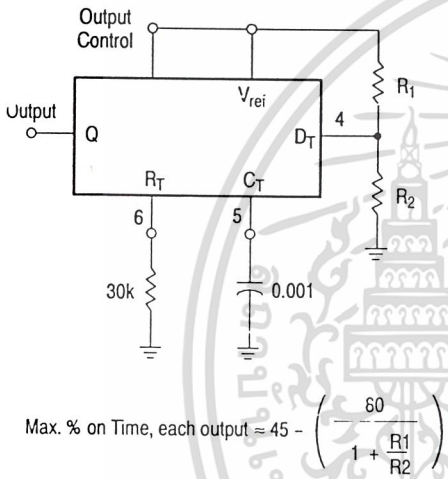


Figure 15. Deadtime Control Circuit

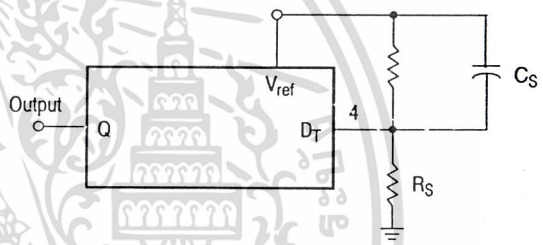


Figure 16. Soft–Start Circuit

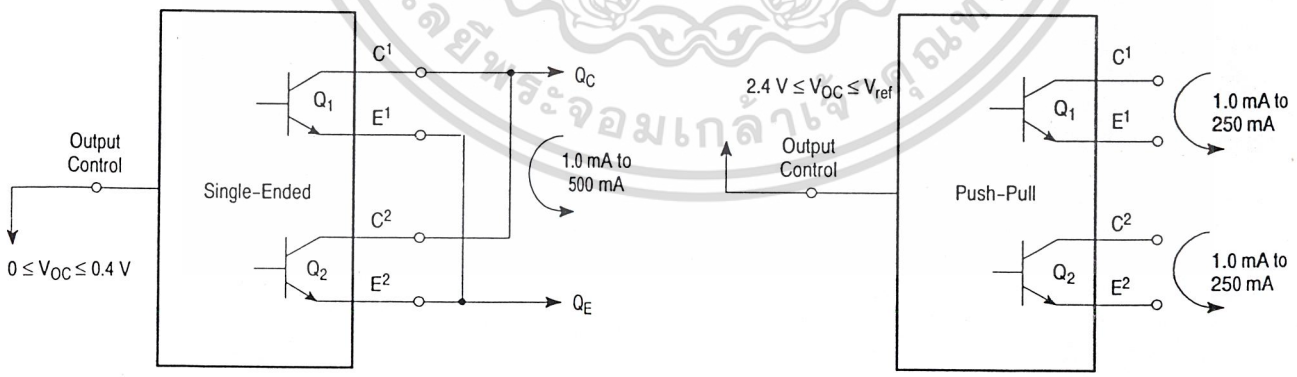


Figure 17. Output Connections for Single–Ended and Push–Pull Configurations

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

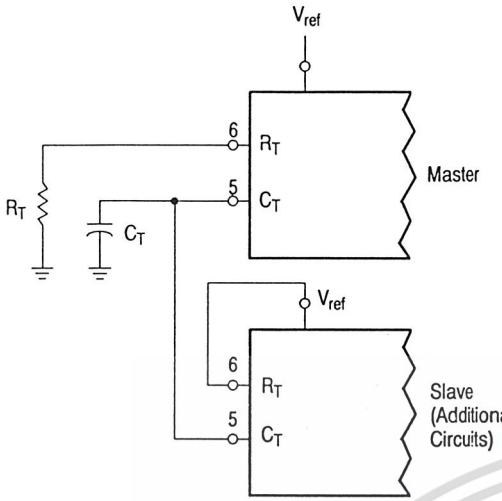


Figure 18. Slaving Two or More Control Circuits

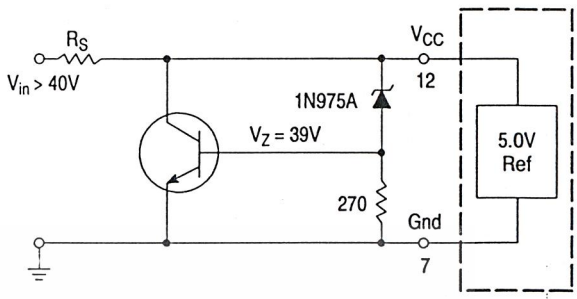


Figure 19. Operation with V_{in} > 40 V Using External Zener

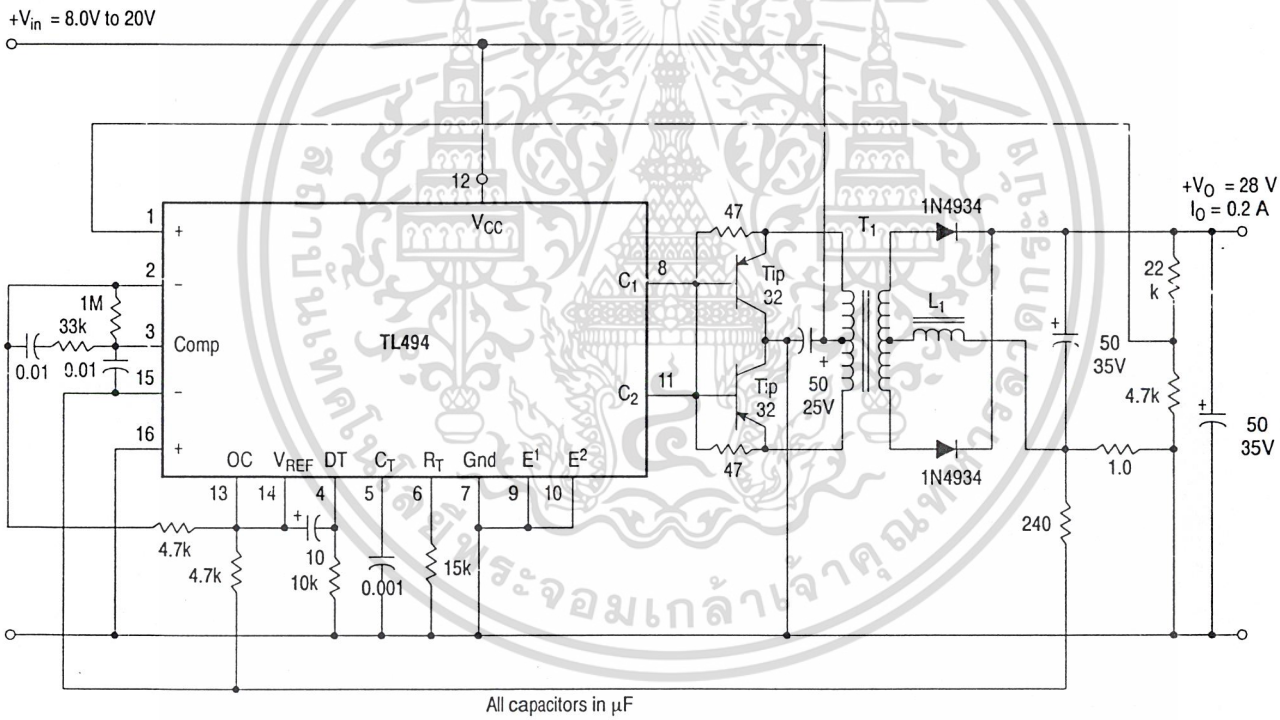


Figure 20. Pulse Width Modulated Push–Pull Converter

Test	Conditions	Results
Line Regulation	V_{in} = 10 V to 40 V	14 mV 0.28%
Load Regulation	V_{in} = 28 V, I_O = 1.0 mA to 1.0 A	3.0 mV 0.06%
Output Ripple	V_{in} 28 V, I_O = 1.0 A	65 mV pp P.A.R.D.
Short Circuit Current	V_{in} = 28 V, R_L = 0.1 Ω	1.6 A
Efficiency	V_{in} = 28 V, I_O = 1.0 A	71%

L1 - 3.5 mH @ 0.3 A
T1 - Primary: 20T C.T. #28 AWG
Secondary: 12OT C.T. #36 AWG
Core: Ferroxcube 1408P-L00-3CB

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

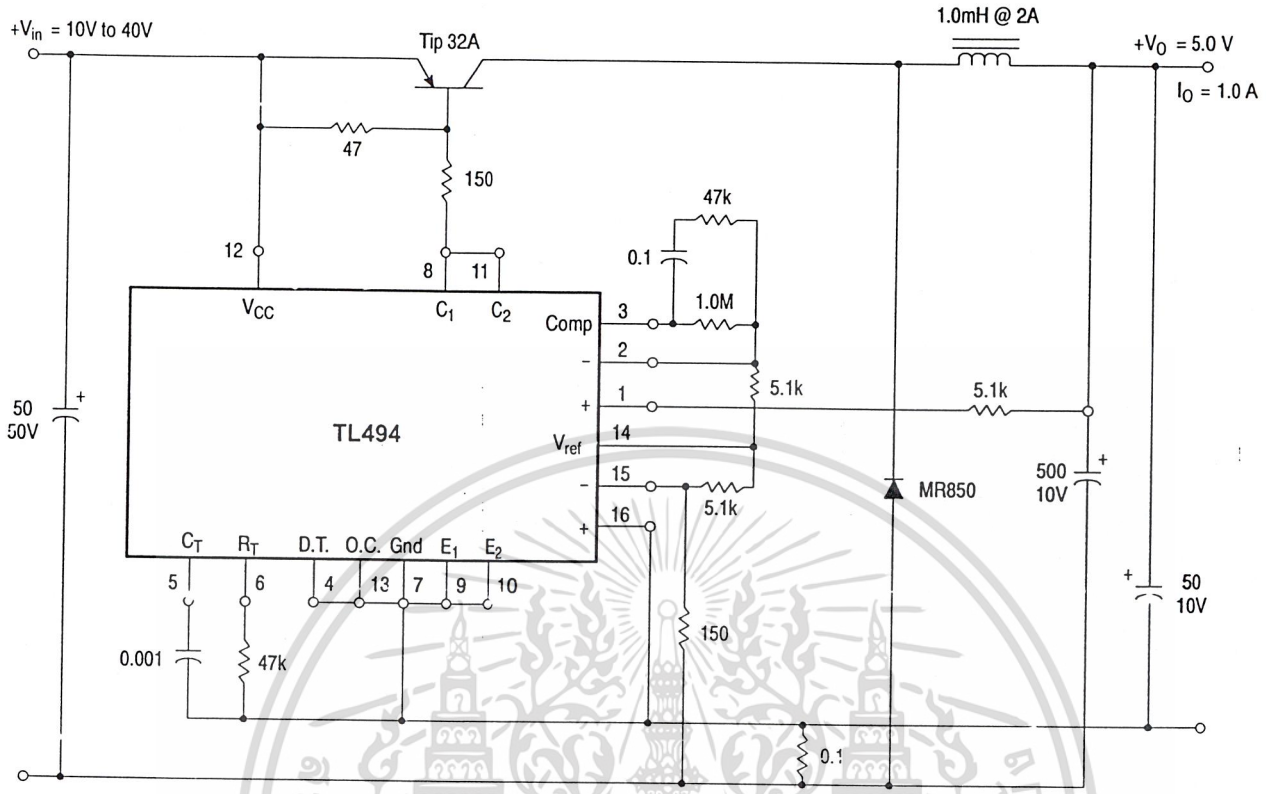


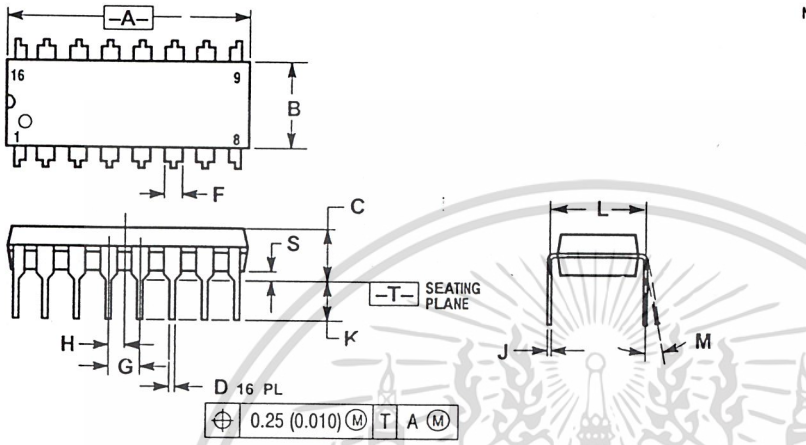
Figure 21. Pulse Width Modulated Step−Down Converter

Test	Conditions	Results
Line Regulation	V_{in} = 8.0 V to 40 V	3.0 mV 0.01%
Load Regulation	V_{in} = 12.6 V, I_O = 0.2 mA to 200 mA	5.0 mV 0.02%
Output Ripple	V_{in} = 12.6 V, I_O = 200 mA	40 mV pp P.A.R.D.
Short Circuit Current	V_{in} = 12.6 V, R_L = 0.1 Ω	250 mA
Efficiency	V_{in} = 12.6 V, I_O = 200 mA	72%

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

PACKAGE DIMENSIONS

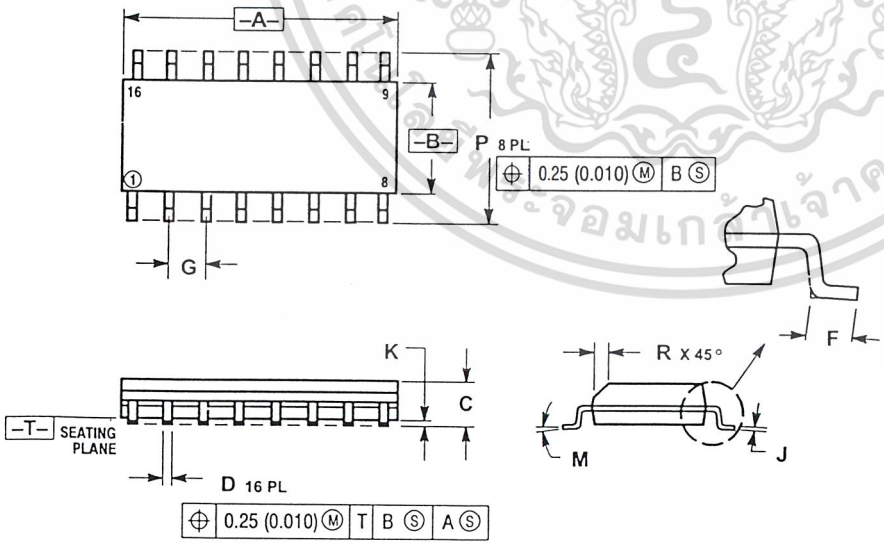
PDIP–16
N SUFFIX
CASE 648–08
ISSUE R



NOTES:
1. DIMENSIONING AND TOLERANCING PER ANSI Y14.5M, 1982.
2. CONTROLLING DIMENSION: INCH.
3. DIMENSION L TO CENTER OF LEADS WHEN FORMED PARALLEL.
4. DIMENSION B DOES NOT INCLUDE MOLD FLASH.
5. ROUNDED CORNERS OPTIONAL.

	INCHES		MILLIMETERS	
DIM	MIN	MAX	MIN	MAX
A	0.740	0.770	18.80	19.55
B	0.250	0.270	6.35	6.85
C	0.145	0.175	3.69	4.44
D	0.015	0.021	0.39	0.53
F	0.040	0.70	1.02	1.77
G	0.100 BSC		2.54 BSC	
H	0.050 BSC		1.27 BSC	
J	0.008	0.015	0.21	0.38
K	0.110	0.130	2.80	3.30
L	0.295	0.305	7.50	7.74
M	0°	10°	0°	10°
S	0.020	0.040	0.51	1.01

SO–16
D SUFFIX
CASE 751B–05
ISSUE J



NOTES:
1. DIMENSIONING AND TOLERANCING PER ANSI Y14.5M, 1982.
2. CONTROLLING DIMENSION: MILLIMETER.
3. DIMENSIONS A AND B DO NOT INCLUDE MOLD PROTRUSION.
4. MAXIMUM MOLD PROTRUSION 0.15 (0.006) PER SIDE.
5. DIMENSION D DOES NOT INCLUDE DAMBAR PROTRUSION. ALLOWABLE DAMBAR PROTRUSION SHALL BE 0.127 (0.005) TOTAL IN EXCESS OF THE D DIMENSION AT MAXIMUM MATERIAL CONDITION.

	MILLIMETERS		INCHES	
DIM	MIN	MAX	MIN	MAX
A	9.80	10.00	0.386	0.393
B	3.80	4.00	0.150	0.157
C	1.35	1.75	0.054	0.068
D	0.35	0.49	0.014	0.019
F	0.40	1.25	0.016	0.049
G	1.27 BSC		0.050 BSC	
J	0.19	0.25	0.008	0.009
K	0.10	0.25	0.004	0.009
M	0°	7°	0°	7°
P	5.80	6.20	0.229	0.244
R	0.25	0.50	0.010	0.019

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

SWITCHMODE is a trademark of Semiconductor Components Industries, LLC.

ON Semiconductor and ON are registered trademarks of Semiconductor Components Industries, LLC (SCILLC). SCILLC reserves the right to make changes without further notice to any products herein. SCILLC makes no warranty, representation or guarantee regarding the suitability of its products for any particular purpose, nor does SCILLC assume any liability arising out of the application or use of any product or circuit, and specifically disclaims any and all liability, including without limitation special, consequential or incidental damages. "Typical" parameters which may be provided in SCILLC data sheets and/or specifications can and do vary in different applications and actual performance may vary over time. All operating parameters, including "Typicals" must be validated for each customer application by customer's technical experts. SCILLC does not convey any license under its patent rights nor the rights of others. SCILLC products are not designed, intended, or authorized for use as components in systems intended for surgical implant into the body, or other applications intended to support or sustain life, or for any other application in which the failure of the SCILLC product could create a situation where personal injury or death may occur. Should Buyer purchase or use SCILLC products for any such unintended or unauthorized application, Buyer shall indemnify and hold SCILLC and its officers, employees, subsidiaries, affiliates, and distributors harmless against all claims, costs, damages, and expenses, and reasonable attorney fees arising out of, directly or indirectly, any claim of personal injury or death associated with such unintended or unauthorized use, even if such claim alleges that SCILLC was negligent regarding the design or manufacture of the part. SCILLC is an Equal Opportunity/Affirmative Action Employer.

PUBLICATION ORDERING INFORMATION

Literature Fulfillment:
Literature Distribution Center for ON Semiconductor
P.O. Box 5163, Denver, Colorado 80217 USA
Phone: 303–675–2175 or 800–344–3860 Toll Free USA/Canada
Fax: 303–675–2176 or 800–344–3867 Toll Free USA/Canada
Email: ONlit@hibbertco.com

N. American Technical Support: 800–282–9855 Toll Free USA/Canada

JAPAN: ON Semiconductor, Japan Customer Focus Center
4–32–1 Nishi–Gotanda, Shinagawa–ku, Tokyo, Japan 141–0031
Phone: 81–3–5740–2700
Email: r14525@onsemi.com

ON Semiconductor Website: http://onsemi.com

For additional information, please contact your local Sales Representative.

สำเนาเอกสารนี้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเอกสารทุกครั้งที่มีการนำไปใช้

เอกสารอ้างอิง

1. ชัยวัฒน์ ลิ้มพรจิตรวิไล , การใช้งานไมโครคอนโทรลเลอร์ MCS 51 , สำนักพิมพ์ INEX, กรุงเทพ , 2540
2. ยืน ภู่วรวรรณ , เทคนิคการประยุกต์ใช้งานลิเนียร์ไอซี , บริษัท ซีเอ็ดยูเคชั่น จำกัด , กรุงเทพ , 2523

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้