

หุ่นยนต์ส่งเอกสาร  
ROBOT MESSENGER



นางสาวมิ่งมานัส ทิวรัักษ์  
นายศิริชัย วิชชาวุธ

เลขหมู่.....  
เลขทะเบียน.....49956  
วัน,เดือน,ปี 16 เม.ย. 2547

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

ปฏิญานិพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต  
ภาควิชาวิศวกรรมคอมพิวเตอร์  
คณะวิศวกรรมศาสตร์  
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง  
ปีการศึกษา 2545

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หุ่นยนต์ส่งเอกสาร  
ROBOT MESSENGER



ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต

ภาควิชาวิศวกรรมคอมพิวเตอร์

คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2545

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ปริญญาบัตรปีการศึกษา 2545

ภาควิชา วิศวกรรมคอมพิวเตอร์

คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เรื่อง ทุนยนต์ส่งเอกสาร

Robot Messenger

ผู้จัดทำ

1. นางสาวมิ่งมานัส ศิวรักษ์ รหัสประจำตัว 42010274
2. นายศิริชัย วิชชาวูธ รหัสประจำตัว 42010344



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## หุ่นยนต์ส่งเอกสาร

นางสาวมิ่งมานัส ศิวรักษ์	42010274
นายศิริชัย วิชชาวุธ	42010344
ดร.อรฉัตร จิตตโสภักดิ์	อาจารย์ที่ปรึกษา
อ.เจริญ วงษ์ชุ่มเย็น	อาจารย์ที่ปรึกษา
ปีการศึกษา 2545	

### บทคัดย่อ

ในปัจจุบันได้มีการนำเทคโนโลยีทางด้านหุ่นยนต์มาใช้งานกันกว้างขวางขึ้น โดยการนำหุ่นยนต์มาทำงานอันเป็นกิจวัตร หรืองานรูทีน (Routine job) ข้อดีประการแรกของการนำหุ่นยนต์มาใช้งานคือหุ่นยนต์มีความแม่นยำมากกว่ามนุษย์ ประการที่สองคือหุ่นยนต์สามารถทำงานได้นานกว่ามนุษย์หากมีการดูแลรักษาที่ดี

การส่งเอกสารก็เป็นตัวอย่างหนึ่งของงานรูทีน ซึ่งสามารถนำหุ่นยนต์มาใช้งานแทนมนุษย์ได้ โดยการนำหลักการทำงานของการประมวลผลรูปภาพ (Image Processing) มาใช้งานเป็นตาของหุ่นยนต์ ซึ่งทฤษฎีการประมวลผลรูปภาพเป็นเทคโนโลยีใหม่ที่มีความซับซ้อนมาก และมีพัฒนาการมาโดยตลอด

ปัญญานิพนธ์ฉบับนี้เป็นอีกแนวทางหนึ่งในการนำหุ่นยนต์มาทำงานร่วมกับทฤษฎีการประมวลผลรูปภาพ และนำไปทดลองใช้งานจริงโดยหุ่นยนต์จะเดินไปตามทางและใช้กล้อง (web camera) ส่องตรวจทางเดินบนตึก ECC และตำแหน่งของห้องเป้าหมาย เมื่อหุ่นยนต์เดินถึงประตูหน้าห้องใดห้องหนึ่งแล้ว หุ่นยนต์จะหยุดและทำการถ่ายภาพป้ายห้องด้วยกล้องและตรวจสอบว่าใช่ห้องเป้าหมายหรือไม่ หากใช่จะทำการส่งเอกสาร หากไม่ใช่จะนำหมายเลขห้องนั้นไปประมวลผลหาทิศทางของห้องเป้าหมายต่อไป ซึ่งในการตรวจจับทางเดินและหมายเลขห้องเป้าหมายจะใช้เทคนิคทางด้านการประมวลผลรูปภาพร่วมกับการรู้จำตัวอักษร (Character Recognition) หลายวิธีด้วยกัน ซึ่งแต่ละวิธีจะให้ผลลัพธ์และข้อเสียที่แตกต่างกันอย่างสิ้นเชิง

## Robot Messenger

Mingmanas Sivaraksa	42010274
Sirichai Wichawut	42010344
Dr.Orachat chitsopak	Advisor
Charoen Vongchumyen	Advisor

### Abstract

In recent years, robots have been used widely. Most robots are used mainly in routine jobs. One of the advantages of using robots is that they have more accuracy than human being. Another advantage is that robot can work in longer hours than human.

This thesis is another way of incorporating image processing into a robot by using web camera as a robot's eyes to guide the way on the ECC building and the position of the target room. When robot arrives in front of any room, it would stop and capture an image of the room number by using a camera and check whether the room is the target room. If that room is the target room the robot will deliver documents. If it is not the target room, robot will determine the direction of the targeted room and continue moving to that room. Several Image processing and Character Recognition techniques are used in order to detect room numbers.

### กิตติกรรมประกาศ

วิทยานิพนธ์ฉบับนี้คงไม่อาจสำเร็จได้ด้วยดี หากไม่ได้รับความช่วยเหลือและความร่วมมือจากทุกฝ่ายอย่างพร้อมเพรียง บุคคลสองท่านแรกที่ต้องกล่าวถึงคือ ดร.อรฉัตร จิตตโสภักดิ์ และ อ.เจริญ วงษ์ชุ่มเย็น อาจารย์ที่ปรึกษาวิทยานิพนธ์ ที่ให้ความเอาใจใส่ แนะนำ เต็มใจ เชื่อ ว่ากล่าว ตักเตือน และช่วยเหลือในทุกด้านอย่างต่อเนื่องทุกๆวันเป็นเวลาหนึ่งปีเต็ม และอ.อวัจนพงษ์ที่ให้ความรู้ทางด้านการรู้จำตัวอักษรอย่างลึกซึ้ง รวมถึงอาจารย์ภาควิชาวิศวกรรมคอมพิวเตอร์ทุกท่านที่อบรมสั่งสอนให้ข้าพเจ้าสามารถนำความรู้ที่ร่ำเรียนมาใช้ในการจัดทำวิทยานิพนธ์ฉบับนี้ ต้องขอขอบพระคุณอย่างสูง

และต้องขอขอบพระคุณบุคคลที่ทำให้ข้าพเจ้ามีวันนี้ ก็คือ บิดา มารดา อันเป็นที่เคารพรักยิ่ง ซึ่งได้เลี้ยงดูให้ผู้เขียนเติบโตใหญ่มีประสบการณ์สูง พร้อมทั้งให้โอกาสในการศึกษาอย่างเต็มที่ และยังให้กำลังใจ เอาใจใส่เสมอมาในทุกๆด้านอันหาที่เปรียบไม่ได้ ข้าพเจ้าขอระลึกถึงพระคุณอันสุดประมาณ และขอกราบขอบคุณมา ณ ที่นี้

และที่ขาดไม่ได้คือ เพื่อนๆร่วมภาควิชาคอมพิวเตอร์ทุกคนที่ให้ความสนใจแวะเวียนกันมาคุยข่าวคราวความคืบหน้าของหุ่นยนต์ของข้าพเจ้ากันอย่างเป็นนิจ ขอขอบคุณมากที่ให้ความสนใจ

มิ่งมานัส ศิวรักษ์  
ศิริชัย วิชชาวุธ

# สารบัญ

	หน้าที่
รายชื่อผู้จัดทำและอาจารย์ที่ปรึกษา	I
บทคัดย่อภาษาไทย	II
บทคัดย่อภาษาอังกฤษ	III
กิตติกรรมประกาศ	IV
สารบัญ	V
สารบัญตาราง	IX
สารบัญรูปภาพ	X
<b>บทที่ 1 บทนำ</b>	<b>1</b>
1.1 ความสำคัญและที่มา	1
1.2 ขอบเขตของงานวิจัย	1
1.3 ประโยชน์ที่คาดว่าจะได้รับ	1
1.4 วิธีการดำเนินงาน	2
<b>บทที่ 2 ทฤษฎีทางด้านฮาร์ดแวร์</b>	<b>3</b>
2.1 สเตปเปอร์มอเตอร์	3
2.1.1 ชนิดของสเตปเปอร์มอเตอร์	4
2.1.1.1 สเตปเปอร์มอเตอร์แบบไบโพลาร์	4
2.1.1.2 สเตปเปอร์มอเตอร์แบบยูนิโพลาร์	4
2.1.2 การใช้สเตปเปอร์มอเตอร์แบบไบโพลาร์	5
2.1.2.1 การควบคุมสเตปเปอร์มอเตอร์แบบเวฟ	5
2.1.2.2 การควบคุมสเตปเปอร์มอเตอร์แบบฟูลสเต็ป	7
2.2 การสื่อสารแบบพาราเรลผ่านทางพริ้นเตอร์พอร์ต	8
2.2.1 ตัวอย่างโปรแกรมการส่งข้อมูลออกจากพาราเรลพอร์ตด้วยภาษา C++	10
2.3 การสื่อสารแบบ USB	10
<b>บทที่ 3 ทฤษฎีทางการประมวลผลรูปภาพ</b>	<b>13</b>
3.1 การแทนภาพด้วยข้อมูลแบบดิจิทัล	14
3.2 การประมวลผลแบบสปีเชียล	14
3.3 ทฤษฎีสี RGB, YCbCr และการเปลี่ยนรูป	14
3.3.1 สี RGB	14

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.3.2	ระดับสีเทา (Grey Scale)	15
3.3.3	สี YcbCr	16
3.3.4	การเปลี่ยนรูประหว่าง RGB และ YcbCr	16
3.4	ฮิสโตแกรม (Histogram)	17
3.4.1	การปรับฮิสโตแกรมของรูปภาพ	18
3.5	การนอร์มัลไลเซชันของสี RGB	19
3.6	การทำเทรซโฮล	20
3.7	ตัวรบกวนภาพชนิดต่างๆและการกำจัด	21
3.7.1	การทำฟิลเตอร์เพื่อกำจัดตัวรบกวนภาพ	22
3.7.1.1	ฟิลเตอร์ค่ามัทยะฐาน	23
3.8	การขยายขอบเขตของภาพ	23
3.9	การแปลงภาพทางเรขาคณิต	23
3.9.1	การปรับขนาดของภาพ	23
3.9.2	การหมุนภาพ	23
3.9.3	ปัญหาของ Forward Mapping และ Backward Mapping	23
3.10	การทำแมตซ์จิงโดยวิธีคอร์เรลชัน	24
3.11	หลักการจดจำรูปแบบโดยใช้วิธีสโตรป	26
<b>บทที่ 4</b>	<b>การออกแบบฮาร์ดแวร์</b>	<b>28</b>
4.1	ส่วนประมวลผลกลาง	29
4.2	ส่วนขับเคลื่อน	29
4.2.1	การออกแบบวงจรขับเคลื่อน	30
4.2.2	การอินเตอร์เฟสวงจรขับเคลื่อนเข้ากับพีซีบอร์ด	32
4.2.3	การเขียนโปรแกรมควบคุมส่วนขับเคลื่อน	32
4.3	ส่วนรับข้อมูลหมายเลขห้อง	33
4.4	กล้อง	35
4.5	ส่วนควบคุมมุมกล้อง	35
4.5.1	แนวคิดในการออกแบบ	35
4.5.2	การประกอบส่วนควบคุมมุมกล้องทางด้านกลศาสตร์	36
4.5.3	การออกแบบวงจรขับเคลื่อนมอเตอร์	37
4.5.4	การเขียนโปรแกรมควบคุมส่วนควบคุมมุมกล้อง	40
<b>บทที่ 5</b>	<b>การออกแบบซอฟต์แวร์</b>	<b>41</b>
5.1	การตรวจหาประตูโดยใช้กล้อง	41

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

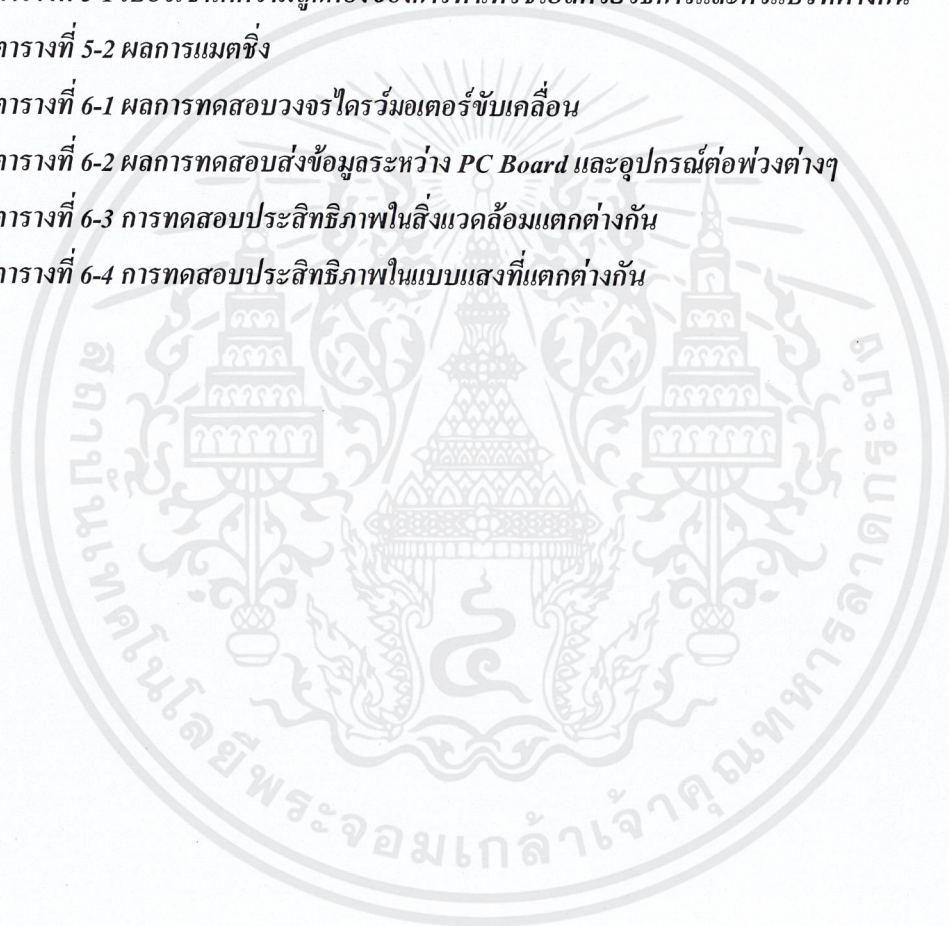
5.2 การตรวจจับประตู่	43
5.3 การตรวจสอบหมายเลขห้องจากภาพ	43
5.3.1 การแยกตัวเลขออกจากส่วนอื่นๆของภาพ	44
5.3.1.1 ความพยายามในการแยกสีโดยใช้ความแตกต่างของค่า Y,Cb,Cr	45
5.3.1.2 ความพยายามแยกสีโดยใช้ความแตกต่างของค่าออร์มัลไลซ์ของค่า RGB	46
5.3.2 การทำเทรซโฮลให้ภาพ	48
5.3.3 การกำจัดสิ่งรบกวน	49
5.3.3.1 สแกนตามแนวแกน X และแกน Y	49
5.3.3.2 แยกสิ่งรบกวนโดยใช้ผลรวมของพิกเซลตามแนวแกน X และ Y	50
5.3.3.3 การนำคอรีเลทเมตซึ่งมาใช้	51
5.3.3.4 การทำคอรีเลทเมตซึ่งแบบเลือกทำ	53
5.3.4 การแบ่งหลักของตัวเลขห้อง	53
5.3.5 การจดจำหมายเลขห้อง	54
5.3.5.1 การจดจำรูปแบบโดยการคอรีเลทเมตซึ่ง	55
5.3.5.2 การแบ่งตัวเลขระหว่าง 1 กับ 7	56
5.3.5.3 การแยกความแตกต่างของตัวเลข 5,6, และ 8	56
<b>บทที่ 6 การทดสอบ</b>	<b>57</b>
6.1 การทดสอบฮาร์ดแวร์	57
6.1.1 การทดสอบวงจรไครว์มอเตอร์ขับเคลื่อน	57
6.1.1.1 การทดสอบวงจรไครว์แบบไม่มีโหลด	57
6.1.1.2 การทดสอบวงจรไครว์ด้วยโหลด	57
6.1.1.2.1 การใช้วงจรไครว์ที่ 1: วงจร JK-FF	
กับ Darlington Transistors	57
6.1.1.2.2 การใช้วงจรไครว์ที่ 2: วงจร L297	
กับ Darlington Transistors	58
6.1.1.2.3 การใช้วงจรไครว์ที่ 3: วงจร L297	
กับไอซีขยายสัญญาณ L298	59
6.1.1.2.4 บทสรุปของวงจรไครว์มอเตอร์ขับเคลื่อน	59
6.1.1.3 การทดสอบด้วยการนำไปวิ่งจริง	60
6.1.2 การทดสอบวงจรไครว์มอเตอร์ของส่วนควบคุมมูมกลิ้ง	61
6.1.2.1 การทดสอบวงจรโดยไม่มีโหลด	61
6.1.2.2 การทดสอบวงจรที่มีโหลด	61
6.1.3 การทดสอบ PC Board	62

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

6.1.3.1 การทดสอบวัดกระแสที่บอร์ดใช้	62
6.1.4 การทดสอบโครงหุ่นยนต์	63
6.1.5 การทดสอบการอินเตอร์เฟส PC Board เข้ากับอุปกรณ์อื่น	63
6.2 การทดสอบซอฟต์แวร์	64
6.2.1 การทดสอบประสิทธิภาพการตัดสิ่งรบกวนภายใต้สิ่งแวดล้อมต่างๆ	64
6.2.2 การทดสอบประสิทธิภาพการจดจำภาพตัวเลขภายใต้แสงแบบต่างๆ	65
6.3 การทดสอบนำหุ่นยนต์ไปใช้งานจริง	66
<b>บทที่ 7 สรุปผลและวิจารณ์</b>	<b>67</b>
ภาคผนวก ก: คำดัชนีของไอซี L297	
ภาคผนวก ข: คำดัชนีของไอซี L298	
ภาคผนวก ค: คำดัชนีของไอซี ULN2803	
ภาคผนวก ง: คำดัชนีของไอซี 74LS139	
ภาคผนวก จ: คำดัชนีของไอซี 74LS76	
ภาคผนวก ฉ: คำดัชนีของไอซี 74LS86	
ภาคผนวก ช: คำดัชนีของทรานซิสเตอร์ TIP120	
ภาคผนวก ญ: ฟังก์ชัน API ของ Logitech SDK 1.0	

## สารบัญตาราง

	หน้าที่
ตารางที่ 2-1 ลอจิกการหมุนสเตปเปอร์มอเตอร์แบบ <i>Half Step</i>	7
ตารางที่ 2-2 ลอจิกการหมุนสเตปเปอร์มอเตอร์แบบ <i>Full Step</i>	7
ตารางที่ 3-1 ตัวอย่างค่าของสีต่างๆใน <i>RGB</i>	15
ตารางที่ 4-1 การออกแบบส่วนขับเคลื่อน	30
ตารางที่ 5-1 เปอร์เซ็นต์ความถูกต้องของการทำทรานซิสต์ด้วยวิธีการและตัวแปรที่ต่างกัน	52
ตารางที่ 5-2 ผลการแมตซิง	55
ตารางที่ 6-1 ผลการทดสอบวงจรไดรฟ์มอเตอร์ขับเคลื่อน	60
ตารางที่ 6-2 ผลการทดสอบส่งข้อมูลระหว่าง <i>PC Board</i> และอุปกรณ์ต่อพ่วงต่างๆ	64
ตารางที่ 6-3 การทดสอบประสิทธิภาพในสิ่งแวดล้อมแตกต่างกัน	64
ตารางที่ 6-4 การทดสอบประสิทธิภาพในระบบแสงที่แตกต่างกัน	65



## สารบัญรูปภาพ

	หน้าที่
รูปที่ 2-1 สเตปเปอร์มอเตอร์แบบไบโพลาร์	4
รูปที่ 2-2 สเตปเปอร์มอเตอร์แบบยูนิโพลาร์	5
รูปที่ 2-3 การทำงานของสเตปเปอร์มอเตอร์ ชั้นที่1	5
รูปที่ 2-4 การทำงานของสเตปเปอร์มอเตอร์ ชั้นที่2	6
รูปที่ 2-5 การทำงานของสเตปเปอร์มอเตอร์ ชั้นที่3	6
รูปที่ 2-6 ลักษณะสัญญาณที่ส่งไปควบคุมสเตปเปอร์มอเตอร์ แบบ <i>Half step</i>	7
รูปที่ 2-7 ขาของพอร์ตพาราลแบบ 25 ขา	8
รูปที่ 2-8 แสดงแอดเดรสของพาราลพอร์ต และลำดับการส่งข้อมูลของพรินเตอร์	8
รูปที่ 2-9 หน้าทีและทิศทางของขาทั้งหมดของพาราลพอร์ต(ทิศทางออกจาก PC)	9
รูปที่ 2-10 อินเตอร์เฟสระหว่างข้อต่อที่พรินเตอร์และข้อต่อที่ PC	9
รูปที่ 2-11 รูปแบบการต่อสัญญาณป้อนกลับ	9
รูปที่ 2-12 การอินเตอร์เฟส USB กับอุปกรณ์ต่อพ่วง	11
รูปที่ 2-13 พอร์ต USB แสดงถึงหน้าที่ของขาสัญญาณ	11
รูปที่ 3-1 โคออร์ดิเนตคาร์ทีเซียนของสี RGB	14
รูปที่ 3-2 เปรียบเทียบเลเยอร์ระหว่างรูปเกรย์สเกลและรูปภาพ RGB	15
รูปที่ 3-3 รูปภาพที่มีช่วงของฮิสโตแกรมต่างกัน	17
รูปที่ 3-4 ความสัมพันธ์ระหว่างฮิสโตแกรมและคอนทราสต์	18
รูปที่ 3-5 ความสัมพันธ์ระหว่าง S และ $T(r)$	18
รูปที่ 3-6 รูปต้นฉบับที่มีฮิสโตแกรมต่างกันที่ความเข้มแสงขณะถ่าย เมื่อนำมาปรับฮิสโตแกรมแล้ว	19
รูปที่ 3-7 ตัวรบกวนภาพชนิดต่างๆ	21
รูปที่ 3-8 การทำฟิลเตอร์ภาพขนาด $(x,y)$ ด้วยวินโดวขนาด $(j,k)$	22
รูปที่ 3-9 การแปลงภาพไปด้านหน้า และกลับหลัง	24
รูปที่ 3-10 แสดงตำแหน่งของวินโดวในการคอร์เรเลตแมตซิง	25
รูปที่ 3-11 ผลของการคอร์เรเลตแมตซิง	25
รูปที่ 3-12 การแบ่งหลักของตัวเลข	26
รูปที่ 3-13 สโตรปขนาด $2 \times 2$	26
รูปที่ 3-14 การแบ่งช่องของสโตรป	27
รูปที่ 3-15 ค่าสโตรปทั้งแกนตั้งและแกนนอนของทุกช่อง	27
รูปที่ 4-1 คอมโพเนนต์ของหุ่นยนต์	28

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 4-2 การออกแบบโครงสร้างของหุ่นยนต์	29
รูปที่ 4-3 Control Word ของมอเตอร์ขับเคลื่อน	31
รูปที่ 4-4 วงจรขับเคลื่อนมอเตอร์สำหรับมอเตอร์ขนาด 2 แอมแปร์ต่อนิ้ว จำนวนหนึ่งตัว	31
รูปที่ 4-5 คาตาพาซของวงจรขับเคลื่อน	32
รูปที่ 4-6 ส่วนรับข้อมูล โหมด 1: รับข้อมูลป้ายห้อง	33
รูปที่ 4-7 ส่วนรับข้อมูลป้ายห้อง โหมด 2: บังคับหุ่นยนต์โดยตรง	34
รูปที่ 4-8 ส่วนรับข้อมูลหมายเลขห้องที่ติดอยู่บนตัวหุ่นยนต์	34
รูปที่ 4-9 กล้องที่วางไว้บนตำแหน่งที่สามารถมองเห็นได้กว้างที่สุด	35
รูปที่ 4-10 การออกแบบส่วนควบคุมมุกกล้องขั้นต้น	36
รูปที่ 4-11 ส่วนควบคุมมุกกล้องที่ออกแบบโดยใช้ฐานกล้องเข้ามาช่วย	36
รูปที่ 4-12 ส่วนควบคุมมุกกล้องที่ประกอบเสร็จเรียบร้อยแล้ว	37
รูปที่ 4-13 คาตาพาซของวงจรไครว์สเตปเปอร์มอเตอร์	37
รูปที่ 4-14 Control Word ของส่วนควบคุมมุกกล้อง	38
รูปที่ 4-15 วงจรภายในของไอซี 74LS139	38
รูปที่ 4-16 วงจรภายในของไอซี ULN2803	39
รูปที่ 4-17 วงจรขับเคลื่อนมอเตอร์หนึ่งเฟส จำนวนสองตัว	39
รูปที่ 5-1 ส่วนประกอบต่างๆของซอฟต์แวร์	41
รูปที่ 5-2 โพลวัชารต์อัลกอริทึมการเดิน	42
รูปที่ 5-3 ภาพถ่ายประตู ในโหมดสี RGB	43
รูปที่ 5-4 ระดับความเข้มสีของ R G และ B เรียงตามลำดับ	43
รูปที่ 5-5 ภาพที่ตัดเฉพาะส่วนหมายเลขห้องของภาพที่ได้จากการใช้ และไม่ใช้แว่นขยาย	43
รูปที่ 5-6 ขั้นตอนของการตรวจหาเลขห้องโดยใช้กล้อง	44
รูปที่ 5-7 ภาพถ่ายป้ายห้องในโหมด RGB	45
รูปที่ 5-8 ภาพถ่ายป้ายห้องในรูปแบบ YCbCr แยกเฉพาะ Y	45
รูปที่ 5-9 ภาพถ่ายป้ายห้องในรูปแบบ YCbCr แยกเฉพาะ Cb	46
รูปที่ 5-10 ภาพถ่ายป้ายห้องในรูปแบบ YCbCr แยกเฉพาะ Cr	46
รูปที่ 5-11 ภาพค่า r ที่ปรับให้อยู่ในรูปภาพเป็นสีเทา	47
รูปที่ 5-12 ภาพค่า g ที่ปรับให้อยู่ในรูปภาพเป็นสีเทา	47
รูปที่ 5-13 ภาพค่า b ที่ปรับให้อยู่ในรูปภาพเป็นสีเทา	47
รูปที่ 5-14 ภาพที่ทำทรชโฮลแล้วมีส่วนที่ไม่ต้องการติดตามด้วย	48
รูปที่ 5-15 ภาพที่ทำทรชโฮลแล้วมีเฉพาะส่วนตัวเลขติดตามเท่านั้น	48
รูปที่ 5-16 ภาพที่ทำทรชโฮลแล้วมีเฉพาะส่วนตัวเลขติดตามเท่านั้น	49

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 5-17 ภาพที่ตัดทรซโฮลแล้วมีเฉพาะส่วนตัวเลขติดมาเท่านั้น	49
รูปที่ 5-18 ตัวอย่างภาพที่ได้หลังจากตัดทรซโฮล	50
รูปที่ 5-19 ภาพหลังจากสแกนตามแกน X และ Y	50
รูปที่ 5-20 การหาผลรวมตามแนวแกน X และ Y	50
รูปที่ 5-21 ภาพ ต้นแบบที่นำไปเปรียบเทียบ	51
รูปที่ 5-22 ภาพ ต้นแบบที่นำไปเปรียบเทียบ	51
รูปที่ 5-23 ผลจากการแมทซิงแบบเทียบพิกเซลของ ภาพที่แปลงเป็นระดับสีเทา	52
รูปที่ 5-24 ภาพที่ได้ตำแหน่งเลขอื่น ใกล้เคียงกว่า เลขชั้น	52
รูปที่ 5-25 ภาพการตัดตำแหน่งโดยใช้วิธีการประมาณ	53
รูปที่ 5-26 ขั้นตอนการหาขนาดโดยประมาณ	54
รูปที่ 5-27 โพลีชาร์ตขั้นตอนการจดจำตัวเลข	55
รูปที่ 5-28 ตัวอย่างของเลข 1 และเลข 7 ที่มีความคล้ายคลึงกันมาก	56
รูปที่ 5-29 ตัวเลข 5 6 และ 8 แบบต่างๆ	56
รูปที่ 6-1 วงจรไมโครสเตปเปอร์มอเตอร์วงจรที่ 1	58
รูปที่ 6-2 การทดสอบหมุนฐานวางกลิ้งด้วยสเตปเปอร์มอเตอร์	61
รูปที่ 6-3 การทดสอบหมุนส่วนควบคุมมุกกลิ้งที่ติดตั้งกลิ้งเรียบร้อยแล้ว	62
รูปที่ 6-4 มีแสงไฟแนวยาว	64
รูปที่ 6-5 มีแสงไฟกลม	65
รูปที่ 6-6 ตัวอย่างภาพที่ตัดทรซโฮลจากภาพที่มีแสงเข้ามาบกรวน	65
รูปที่ 6-7 ตัวอย่างภาพที่ตัดทรซโฮลจากภาพที่มีแสงสะท้อนที่แว่นขยาย	66

# บทที่ 1

## บทนำ

### 1.1 ความสำคัญและที่มา

ในปัจจุบันได้มีการนำเทคโนโลยีทางด้านหุ่นยนต์มาใช้งานกันกว้างขวางขึ้น โดยการนำหุ่นยนต์มาทำงานอันเป็นกิจวัตร หรืองานรูทีน (routine job) ข้อดีประการแรกของการนำหุ่นยนต์มาใช้งานคือหุ่นยนต์มีความแม่นยำมากกว่ามนุษย์ ประการที่สองคือหุ่นยนต์สามารถทำงานได้นานกว่ามนุษย์หากมีการดูแลรักษาที่ดี

การส่งเอกสารก็เป็นตัวอย่างหนึ่งของงานรูทีน ซึ่งสามารถนำหุ่นยนต์มาใช้งานแทนมนุษย์ได้ โดยการนำหลักการทำงานของการประมวลผลรูปภาพ (Image Processing) มาใช้งานเป็นตาของหุ่นยนต์ ซึ่งทฤษฎีการประมวลผลรูปภาพเป็นเทคโนโลยีใหม่ที่มีความซับซ้อนมาก และมีพัฒนาการมาโดยตลอด

ปริญญาพิพนธ์ฉบับนี้เป็นอีกแนวทางหนึ่งในการนำหุ่นยนต์มาทำงานร่วมกับทฤษฎีการประมวลผลรูปภาพ และนำไปทดลองใช้งานจริง

### 1.2 ขอบเขตของโครงการ

- หุ่นยนต์จะรับอินพุตหมายเลขห้องปลายทางจากผู้ส่งเอกสาร
- หุ่นยนต์ค้นหาห้องเป้าหมาย ด้วยกล้องที่ติดอยู่บนตัวหุ่นยนต์
- เมื่อถึงหน้าห้องแล้วหุ่นยนต์จะสามารถตรวจจับประตูได้และจะหันหน้าเข้าหาประตูแล้วยกมุกกล้องขึ้นถ่ายรูปป้ายห้องเพื่อนำรูปนั้นมาทำการจดจำรูปแบบ
- ถ้าหากว่าห้องนั้นไม่ใช่ห้องเป้าหมายหุ่นยนต์จะนำห้องนั้นเป็นต้นทางแล้วคำนวณทิศทางของห้องเป้าหมายถัดไป

โครงการนี้มีข้อกำหนดคือ

- หุ่นยนต์จะสามารถเดินได้บนตึก ECC เพียงชั้นเดียวเท่านั้นไม่สามารถข้ามชั้นได้
- หุ่นยนต์สามารถทำงานได้เฉพาะในที่ที่มีแสงสว่างจากดวงอาทิตย์เพียงพอเท่านั้น
- หุ่นยนต์ไม่สามารถตรวจจับสิ่งกีดขวางแปลกปลอมได้
- ต้องไม่มีอะไรกระทบกับตัวหุ่นยนต์ขณะกำลังเดิน
- หุ่นยนต์สามารถส่งเอกสารที่มีน้ำหนักไม่มากกว่า 2 กิโลกรัมเท่านั้น
- การส่งเอกสารแต่ละครั้งสามารถส่งได้เพียงห้องเดียวเท่านั้น

### 1.3 ประโยชน์ที่คาดว่าจะได้รับ

นอกจากความรู้ที่จะได้รับจากการวิจัยแล้วยังจะได้หุ่นยนต์ที่สามารถส่งเอกสารได้ ทำให้การส่งเอกสารไม่จำเป็นต้องให้มนุษย์เป็นผู้เดินส่งอีก นอกจากนี้ยังสามารถนำไปพัฒนาต่อเพื่อทำแมชชีนวิชั่น เรื่องที่เกี่ยวข้องได้อีกด้วย

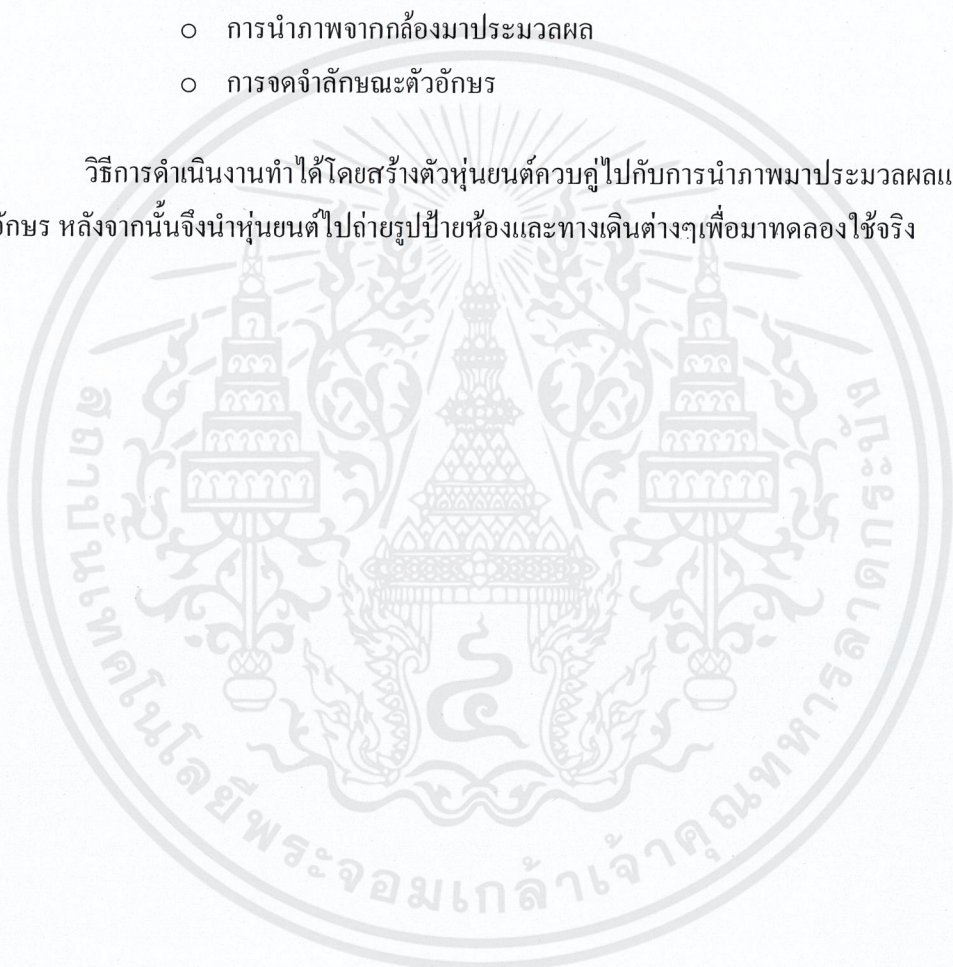
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

#### 1.4 วิธีการดำเนินงาน

โครงสร้างการทำงานของหุ่นยนต์ทั้งหมดนั้นแบ่งออกได้ดังนี้

- ส่วนของฮาร์ดแวร์
  - ตัวหุ่นยนต์
  - เซนเซอร์ต่างๆ
  - กล้อง
- ส่วนของซอฟต์แวร์
  - การนำภาพจากกล้องมาประมวลผล
  - การจดจำลักษณะตัวอักษร

วิธีการดำเนินงานทำได้โดยสร้างตัวหุ่นยนต์ควบคู่ไปกับการนำภาพมาประมวลผลและจดจำตัวอักษร หลังจากนั้นจึงนำหุ่นยนต์ไปถ่ายรูปป้ายห้องและทางเดินต่างๆเพื่อมาทดลองใช้จริง



## บทที่ 2

### ทฤษฎีทางด้านฮาร์ดแวร์

#### 2.1 สเตปเปอร์มอเตอร์ (Stepper motor)

สเตปเปอร์มอเตอร์คือมอเตอร์ชนิดหนึ่งที่ทำกรหมุนเป็นขั้นๆทีละน้อยตามเส้นรอบวงและหยุดแตกต่างกับมอเตอร์ทั่วไปซึ่งจะหมุนไปเรื่อยๆตลอดเวลา เราสามารถควบคุมการหมุนของสเตปเปอร์มอเตอร์ได้อย่างละเอียดโดยใช้คอมพิวเตอร์ ไมโครคอนโทรลเลอร์ หรือแม้กระทั่งวงจรถลอจิก สเตปเปอร์มอเตอร์สามารถใช้งานที่ต้องการความละเอียดต่างๆเช่น ควบคุมการหมุนกระดาษในพริ้นเตอร์ หรือใช้ในการเคลื่อนที่หัวอ่านเพื่ออ่านและเขียนของฟลอปปีดิสก์ เป็นต้น สเตปเปอร์มอเตอร์ใช้งานในลักษณะระบบเปิด (Open Loop System) คือสเตปเปอร์มอเตอร์สามารถทำงานได้โดยไม่ต้องมีการป้อนค่า พารามิเตอร์กลับมาที่วงจรควบคุม (Feedback) แต่ในงานเฉพาะทางที่ต้องการความแน่นอนของตำแหน่งของมอเตอร์ ก็สามารถใช้เซนเซอร์ตรวจจับตำแหน่งร่วมกับสเตปเปอร์มอเตอร์ได้โดยง่าย

สเตปเปอร์มอเตอร์เป็นอุปกรณ์ที่ทำหน้าที่แปลงสัญญาณข้อมูลดิจิทัล ไปเป็นการเคลื่อนที่เชิงกลอย่างเป็นสัดส่วน โดยแกนหรือโรเตอร์ของมอเตอร์ชนิดนี้จะถูกควบคุมให้หมุนไปเป็นลำดับขั้น

ประโยชน์จากการใช้งานสเตปเปอร์มอเตอร์ที่มีการควบคุมโดยใช้สัญญาณดิจิทัล มีความถูกต้องและแม่นยำ และสามารถเปลี่ยนตำแหน่งของโหนดได้อย่างรวดเร็ว เนื่องจากแต่ละ สัญญาณเข้ามาหนึ่งพัลส์ จะทำให้สเตปเปอร์มอเตอร์เคลื่อนที่ไปหนึ่งสเตปอย่างเที่ยงตรง

เมื่อพิจารณาในแง่ทางกลจะพบว่าสเตปเปอร์มอเตอร์ที่มีความเที่ยงตรงที่ยอมรับได้ซึ่งในการเปลี่ยนตำแหน่งอย่างง่ายอาจใช้สวิตช์ ในการควบคุมมอเตอร์ นั่นคือสร้างวงจรถควบคุมได้ง่าย ถ้าต้องการเพิ่มประสิทธิภาพของส่วนควบคุม ให้ดีขึ้น อาจใช้ไอซีที่มีความเร็วสูง เนื่องจากประกอบด้วยมอเฟตอยู่ภายใน จึงได้กำลังงานสูง และต้นทุนต่ำ ความสะดวกในการใช้งานนี้เองจึงทำให้นำไปสู่การใช้งานอย่างแพร่หลาย

การได้รับประโยชน์จากสเตปเปอร์มอเตอร์อย่างเต็มที่นั้นขึ้นอยู่กับการขับอย่างถูกต้องเหมาะสม ซึ่งภาคขับนี้จะต้องประกอบไปด้วย แหล่งจ่ายไฟฟ้ากระแสตรง อิเล็กทรอนิกส์สวิตช์ โดยอาจจะเป็นทรานซิสเตอร์หรือมอสเฟต และแหล่งจ่ายสัญญาณข้อมูลแบบดิจิทัล ซึ่งมีลักษณะเป็นพัลส์ เพื่อใช้ควบคุมทิศทางการหมุนของมอเตอร์

ในปัจจุบันสเตปเปอร์มอเตอร์ที่นิยมใช้มีอยู่ 2 ชนิดด้วยกันคือแบบแม่เหล็กถาวร และแบบที่สามารถเปลี่ยนความเข้มสนามแม่เหล็กได้ ซึ่งแบบแม่เหล็กถาวรจะสะดวกในการใช้และควบคุมมากกว่า แบ่งออกเป็น 2 แบบด้วยกันคือ

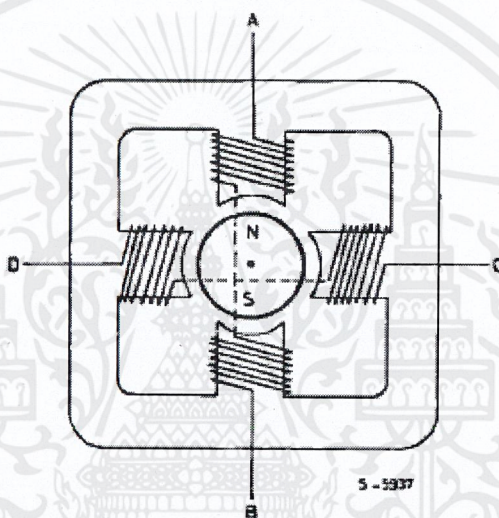
- ไบโพลาร์ (Bipolar)
- ยูนิโพลาร์ (Unipolar)

ซึ่งทั้งสองชนิดนี้มีความแตกต่างกันที่ขั้วของมอเตอร์ แบบยูนิโพลาร์จะทำให้เกิดแรงบิดน้อยกว่าแบบไบโพลาร์ ความแตกต่างทางกายภาพของสเตปเปอร์มอเตอร์ทั้งสองแบบคือสายไฟที่ต่อมาจากตัวมอเตอร์ แบบไบโพลาร์จะมี 4 สาย ส่วนแบบยูนิโพลาร์จะมี 5-6 สาย

### 2.1.1 ชนิดของสเตปเปอร์มอเตอร์

#### 2.1.1.1 สเตปเปอร์มอเตอร์แบบไบโพลาร์

เป็นสเตปเปอร์มอเตอร์ที่มีแกนเป็นแม่เหล็กถาวรและมีคอล์ยอยู่ 4 ชุดด้วยกัน ทำงานโดยใช้กระแสไบไดเรกชันแนล (Bidirectional) ส่งกระแสผ่านจาก AB/CD/BA/DC เพื่อให้มอเตอร์หมุนไปหนึ่งเฟสแบบฟูลสเตป (Full step / Wave drive) สัญลักษณ์ AB คือส่งกระแสจากขั้ว A ไปขั้ว B และ BA เป็นการส่งกระแสในทิศตรงกันข้าม



รูปที่ 2-1 สเตปเปอร์มอเตอร์แบบไบโพลาร์

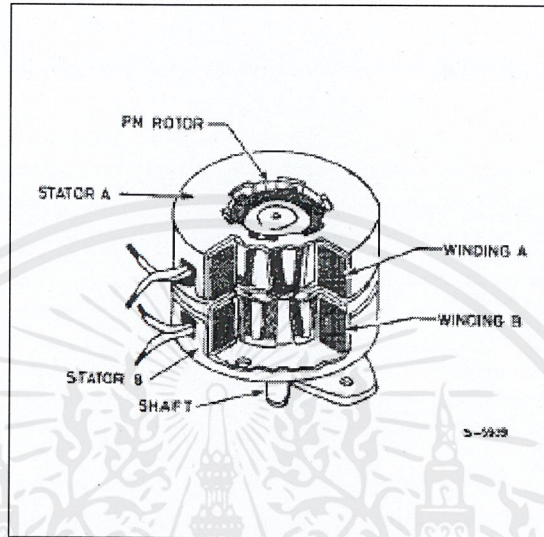
ทิศทางของกระแสไฟฟ้าที่ป้อนเข้าสู่สเตปเปอร์มอเตอร์ ถูกควบคุมโดยการทำงานของ อิเล็กทรอนิกส์สวิตช์ ดังนั้นสเตปเปอร์มอเตอร์จะหมุนไปหนึ่งช่วงในแต่ละอินพุตพัลส์ที่ป้อนเข้าสู่วงจรอิเล็กทรอนิกส์สวิตช์ ซึ่งขนาดที่หมุนไปจะขึ้นอยู่กับการออกแบบสเตปเปอร์มอเตอร์ โดยมีค่าตั้งแต่ 1.8 องศา ถึง 15 องศา ถ้าส่งสัญญาณไปยังวงจรสวิตช์ซึ่ง 24 พัลส์ โดยมีค่าหนึ่งของช่วงการหมุนเท่ากับ 15 องศา สเตปเปอร์มอเตอร์ก็จะหมุนไปครบหนึ่งรอบพอดี ขึ้นอยู่กับอัตราการจ่ายสัญญาณพัลส์ควบคุม โดยสัญญาณนี้อาจถูกผลิตโดยวงจรกำเนิดสัญญาณที่ปรับความถี่ได้ หรือจากไอซีควบคุม

#### 2.1.1.2 สเตปเปอร์มอเตอร์แบบยูนิโพลาร์

เป็นสเตปเปอร์มอเตอร์ที่มีแกนเป็นแม่เหล็กถาวรเช่นเดียวกับไบโพลาร์ที่ได้กล่าวไปข้างต้น เพียงแต่ใช้การหมุนกลับของสเตเตอร์ ฟลักซ์ แทนที่จะใช้การขับแบบไบไดเรกชัน

การขับมอเตอร์ชนิดนี้ใช้วิธีการเดียวกับไบโพลาร์เพียงแต่นำเอาขั้นตอนแบบยูนิโพลาร์เข้ามาแทน บริดจ์ ไดรเวอร์ โดยขั้นตอนแบบยูนิโพลาร์นี้คือการนำทรานซิสเตอร์แบบคาบิลิตัน สี่ตัวมาใช้ และสเตปเปอร์มอเตอร์แบบยูนิโพลาร์ยังมีราคาแพงกว่าเนื่องมียขดลวดมากกว่าเป็นเท่าตัว และยังให้ทอร์กต่ำกว่าเอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่นอนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เนื่องจากขดลวดทำมาจากเส้นลวดที่มีขนาดบางกว่า ในอดีตสเตปเปอร์มอเตอร์แบบยูนิโพลาร์เป็นที่นิยม เนื่องจากใช้บังคับได้ง่าย แต่ในปัจจุบันมีการคิดค้นไอซี L298N ที่ใช้ในการควบคุม สเตปเปอร์มอเตอร์แบบไบโพลาร์โดยเฉพาะ ทำให้การควบคุมสเตปเปอร์มอเตอร์แบบไบโพลาร์ง่ายกว่า ในปัจจุบันสเตปเปอร์มอเตอร์แบบไบโพลาร์ จึงเป็นที่นิยมใช้มากกว่า



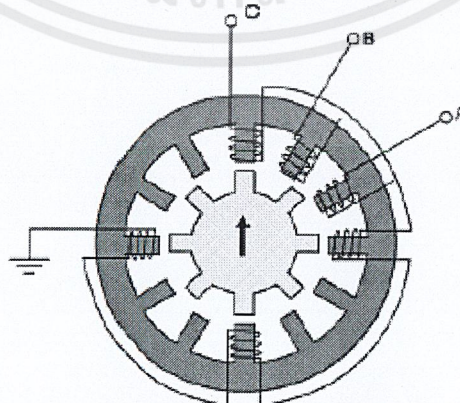
รูปที่ 2-2 สเตปเปอร์มอเตอร์แบบยูนิโพลาร์

### 2.1.2 วิธีใช้สเตปเปอร์มอเตอร์แบบไบโพลาร์

การทำงานของสเตปเปอร์มอเตอร์แบบไบโพลาร์ ตัวสเตเตอร์และโรเตอร์ที่สร้างด้วยเหล็กผสมซิลิคอน โดยที่ตัวโรเตอร์จะไม่ถูกพันด้วยคอยล์ จะเป็นส่วนที่ถูกพันด้วยคอยล์จะเป็นส่วนของสเตเตอร์เท่านั้น ดังนั้นเมื่อเราจ่ายกระแสไฟฟ้าผ่านคอยล์ จึงทำให้สเตเตอร์มีคุณสมบัติเป็นแม่เหล็ก

#### 2.1.2.1 การควบคุมสเตปเปอร์มอเตอร์แบบเวฟ (Wave or Half Step Drive)

การควบคุมมอเตอร์แบบเวฟง่ายต่อการใช้งาน ใช้การส่งกระแสเข้าไปที่แต่ละขาของ สเตเตอร์แต่ละตัวเรียงลำดับกันไปเท่านั้น วิธีนี้นอกจากจะเร็วแล้วยังกินไฟต่ำอีกด้วย แต่ก็จะมีแรงบิดไม่มากนัก

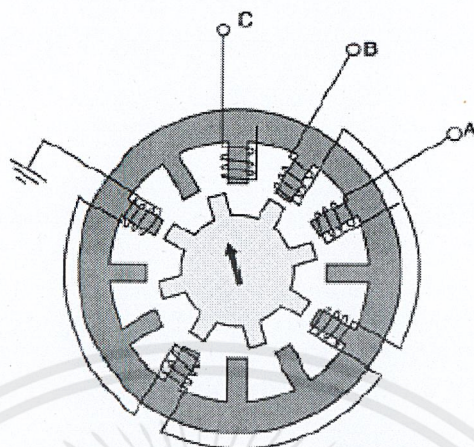


STEP 1

รูปที่ 2-3 การทำงานของสเตปเปอร์มอเตอร์ ขั้นที่ 1

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น. เมื่อนุญาตให้เข้าไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

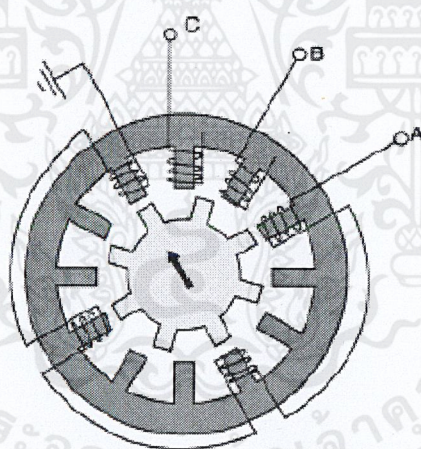
ในขั้นแรกจะจ่ายกระแสให้แก่เฟส C ดังนั้นซี่ของโรเตอร์ก็จะถูกดูดเข้าหาเฟส C



### STEP 2

รูปที่ 2-4 การทำงานของสเตปเปอร์มอเตอร์ ขั้นที่ 2

สเตปต่อไปหยุดจ่ายกระแสให้กับเฟส C และจ่ายให้กับเฟส B แทน ซึ่งของโรเตอร์อันถัดไปที่อยู่ใกล้เฟส B มากที่สุดก็จะถูกดูดเข้ามาแทน



### STEP 3

รูปที่ 2-5 การทำงานของสเตปเปอร์มอเตอร์ ขั้นที่ 3

และเมื่อหยุดจ่ายกระแสให้เฟส B และจ่าย กระแสที่เฟส A แทน โรเตอร์ที่อยู่ใกล้เฟส A มากที่สุดจะถูกดูดเข้ามาแทน

จากหลักการนี้จะมีรูปแบบการจ่ายกระแสให้กับโรเตอร์ต่างๆเรียงตามลำดับ A B C ซึ่งจะอาศัยวงจรไอซีเข้ามาช่วย โดยส่งข้อมูลขนานในการส่งสัญญาณไปควบคุมสเตปเปอร์มอเตอร์ โดยจะส่งค่า 1,2, และ 4 ไปที่พอร์ต์ ตามลำดับวนไปเรื่อยๆ ซึ่งเป็นค่าในเลขฐานสอง

Step No.	Phase 1	Phase 2	Phase 3	Phase 4
1	ON	-	-	-
2	-	ON	-	-
3	-	-	ON	-
4	-	-	-	ON
5	ON	-	-	-
6	-	ON	-	-

ตารางที่ 2-1 ลอจิกการหมุนสเตปเปอร์มอเตอร์แบบ Half Step

00000001
00000010
00000100

รูปที่ 2-6 ลักษณะสัญญาณที่ส่งไปควบคุมสเตปเปอร์มอเตอร์ แบบ Half step

### 2.1.2.2 การควบคุมสเตปเปอร์มอเตอร์แบบสองเฟส หรือแบบฟูลสเตป (Full Step Drive)

การควบคุมสเตปเปอร์มอเตอร์แบบสองเฟสจะเป็นการส่งกระแสไฟให้ขั้วของสเตเตอร์ทีละสองขั้วพร้อมๆกัน และเรียงลำดับกันไป เช่น ส่งกระแสไฟให้ขั้ว AB / BC / CD / DA / AB สลับกันไปเรื่อยๆ ซึ่งการควบคุมแบบฟูลสเตปนี้มีข้อดีตรงที่จะให้แรงบิดได้มากเป็นสองเท่าของแบบเวฟ แต่ก็จะใช้กระแสไฟมากกว่าเป็นสองเท่าเช่นเดียวกัน การควบคุมสเตปเปอร์มอเตอร์แบบฟูลสเตปมีลอจิกของการส่งกระแสไฟดังตารางข้างล่าง

Step No.	Phase 1	Phase 2	Phase 3	Phase 4
1	ON	ON	-	-
2	-	ON	ON	-
3	-	-	ON	ON
4	ON	-	-	ON
5	ON	ON	-	-
6	-	ON	ON	-

ตารางที่ 2-2 ลอจิกการหมุนสเตปเปอร์มอเตอร์แบบ Full Step

## 2.2 การสื่อสารแบบพาราลลелผ่านทางพริ้นเตอร์พอร์ต

พอร์ตพาราลลลมาตรฐานของ IBM 80x86 PC จะมีอยู่ทั้งหมด 25 ขาคู่ด้วยกัน แบ่งหน้าที่ของแต่ละพินออกได้ดังนี้

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

ขาส่งข้อมูลพาราลลจะมีทั้งหมด 8 ขา สามารถส่งข้อมูลได้ที่ละหนึ่งไบต์ แบบซิมเพลกหรือฮาล์ฟดูเพลกก็ได้ (Simplex and Half-duplex transfer)

- **ขาแสดงสถานะพริ้นเตอร์ (Printer status signals)**

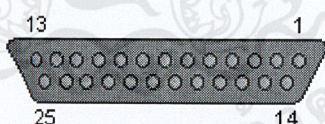
ขาแสดงสถานะของพริ้นเตอร์จะมีทิศทางของสัญญาณจากพริ้นเตอร์ไปยัง PC มีอยู่ทั้งหมด 5 ขาคู่ด้วยกันคือ

- PE (ขา 12) ใช้เมื่อกระดาษหมด
- BUSY (ขา 11) ใช้เมื่อพริ้นเตอร์ยังไม่สามารถรับข้อมูลใหม่ๆ ได้
- /ERROR (ขา 15) ใช้เมื่อพริ้นเตอร์เกิดขัดข้องขึ้น ไม่สามารถใช้งานได้
- SLCT (ขา 13) แสดงสัญญาณว่าพริ้นเตอร์เปิดอยู่และพร้อมใช้งาน
- /ACKNLG (ขา 10) สัญญาณตอบรับ ACK เพื่อขอรับไบต์ใหม่

- **ขาควบคุมพริ้นเตอร์ (Printer control signals)**

ขาควบคุมจะมีทิศทางการส่งข้อมูลจาก PC ไปยังพริ้นเตอร์ มีทั้งหมด 4 ขาคู่ด้วยกันคือ

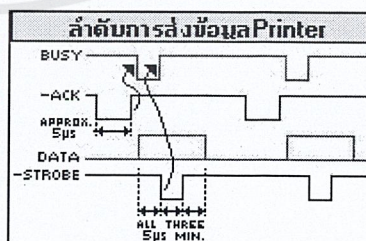
- /STROBE (ขา 1) ใช้ส่งสัญญาณถามสถานะพริ้นเตอร์
- /AUTOFD (ขา 14) ใช้สั่งว่าให้ใส่กระดาษเองที่ละแผ่นหรือไม่
- /INIT (ขา 16) สัญญาณรีเซ็ต
- /SELIN (ขา 17) พริ้นเตอร์จะรับข้อมูลเมื่อสัญญาณเป็น low เท่านั้น



รูปที่ 2-7 ขาของพอร์ตพาราลลแบบ 25 ขา

LPT1	
I/O Address 378H-37FH	
Interrupt IRQ7	
LPT2	
I/O Address 278H-27FH	
Interrupt IRQ5	

PC ต่อกับ PC 25 ขา				
ชื่อต่อตัวเมีย				
2-15	3-13	4-12	5-10	6-11
15-2	3-13	12-4	10-5	11-6
25-25				



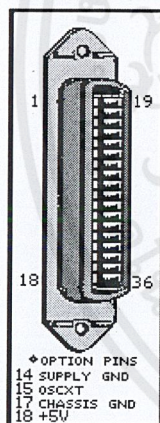
รูปที่ 2-8 แสดงแอดเดรสของพาราลลพอร์ต และลำดับการส่งข้อมูลของพริ้นเตอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Pin	Name	Dir	Description	Pin	Name	Dir	Description
1	/STROBE	→	Strobe	14	/AUTOFD	→	Autofeed
2	D0	→	Data Bit 0	15	/ERROR	←	Error
3	D1	→	Data Bit 1	16	/INIT	→	Initialize
4	D2	→	Data Bit 2	17	/SELIN	→	Select In
5	D3	→	Data Bit 3	18	GND	—	Signal Ground
6	D4	→	Data Bit 4	19	GND	—	Signal Ground
7	D5	→	Data Bit 5	20	GND	—	Signal Ground
8	D6	→	Data Bit 6	21	GND	—	Signal Ground
9	D7	→	Data Bit 7	22	GND	—	Signal Ground
10	/ACK	←	Acknowledge	23	GND	—	Signal Ground
11	BUSY	←	Busy	24	GND	—	Signal Ground
12	PE	←	Paper End	25	GND	—	Signal Ground
13	SEL	←	Select				

รูปที่ 2-9 หน้าที่และทิศทางของขาทั้งหมดของพาราลพอร์ต (ทิศทางออกจาก PC)

ข้อต่อที่ Printer 36 ขา

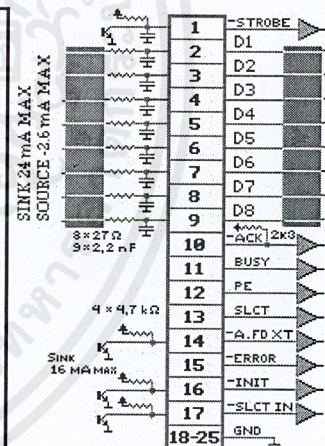
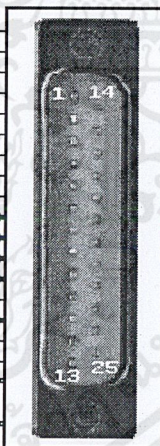


1 (19)	-STROBE	1
2 (20)	Data 0	2
3 (21)	Data 1	3
4 (22)	Data 2	4
5 (23)	Data 3	5
6 (24)	Data 4	6
7 (25)	Data 5	7
8 (26)	Data 6	8
9 (27)	Data 7	9
10 (28)	-ACK	10
11 (29)	BUSY	11
12	PAPER OUT Paper Empty	12
13	SLCT Select	13
14	+ -Auto Feed XT	14
32	-FAULT -ERROR	15
31 (30)	-PRIME -INIT	16
36	+ -SLCT IN	17
16, 19-30	Ground	18-25

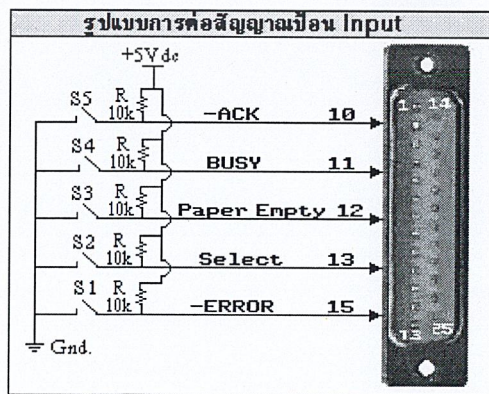
OPTION PINS  
14 SUPPLY GND  
15 OSCXT  
17 CHASSIS GND  
18 +5V

16-LOGIC GND, 19-30-TWISTED PAIR RETURNS ONLY PART OF THEM CONNECTED IN TYPICAL CABLE  
Shield + Frame

ข้อต่อที่ PC 25 ขา ตัวเมีย



รูปที่ 2-10 อินเตอร์เฟสระหว่างข้อต่อที่พริ้นเตอร์และข้อต่อที่ PC



รูปที่ 2-11 รูปแบบการต่อสัญญาณป้อนกลับ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่นับญาติให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

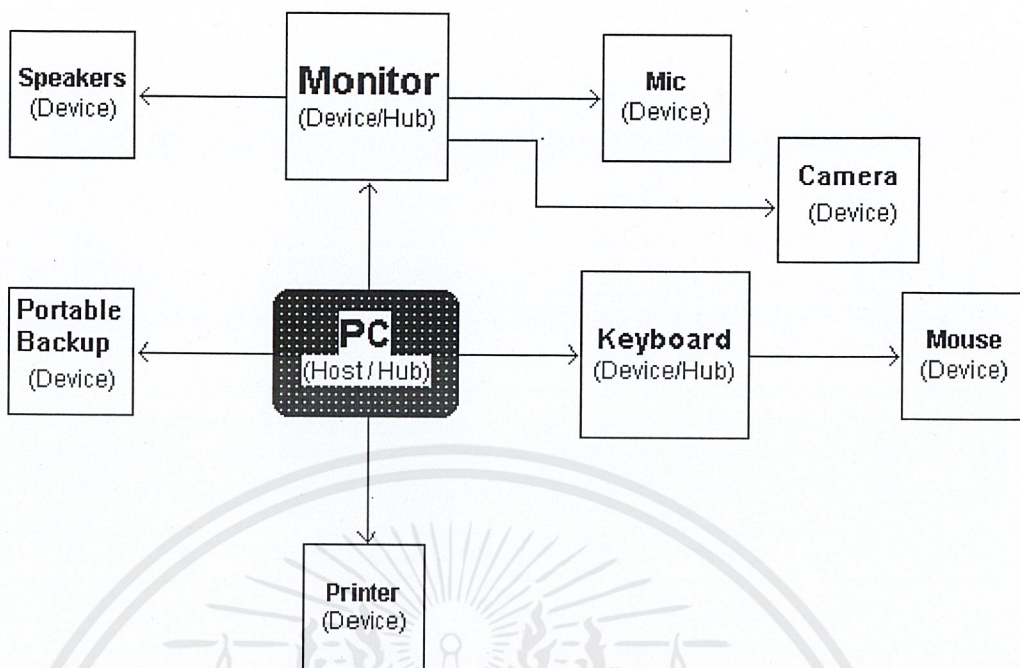
### 2.2.1 ตัวอย่างโปรแกรมการส่งข้อมูลออกพาราลเลลพอร์ตด้วยภาษา C++

```
#include <dos.h>;
#include <iostream.h>;

int main(){
    int data, port;
    port=0x378;           // LPT1
    data=0xAA;           //10101010b
    outp(port,data);     // out AAh to LPT1
    cout<<"Out data "<< data<<" to port number "<<port<<endl;
    return 1;
}
```

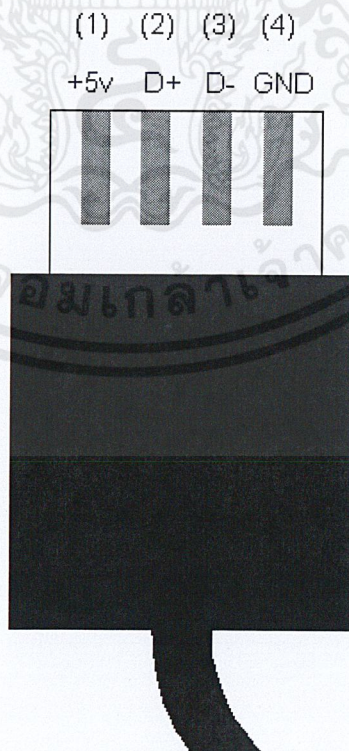
### 2.3 การสื่อสารแบบ USB

พอร์ตการสื่อสารแบบเก่าที่ใช้กันอย่างแพร่หลายคือ พาราลเลลและซีเรียลพอร์ต (Parallel and Serial Port) มีความเร็วในการส่งข้อมูลที่ไม่มากนัก ความเร็วของพอร์ตพาราลเลลอยู่ที่ 3200 กิโลบิตต่อวินาที ในขณะที่พอร์ตซีเรียลอยู่ที่ 115 กิโลบิตต่อวินาที และพอร์ตทั้งสองต้องการอินเทอร์พท์ของตัวเอง (IRQ:Interrupt) ซึ่งมีจำนวนจำกัดไม่สามารถเพิ่มจำนวนออกไปได้ และพอร์ตทั้งสองนี้ยังไม่รองรับพ्लักแอนด์เพลย์ (Plug and Play) ซึ่งค่อนข้างยุ่งยากต่อผู้ใช้ที่ไม่ค่อยมีความรู้ทางด้านเทคนิค จึงได้มีการคิดค้นการสื่อสารด้วย USB (Universal Serial Bus) ขึ้นมา ซึ่งสามารถสื่อสารด้วยความเร็ว 12 เมกะบิตต่อวินาที ขึ้นมาโดยเน้นไปที่การสื่อสารทางด้านมัลติมีเดีย เช่นการสื่อสารกับฮาร์ดแวร์ที่แสดงรูปภาพและเสียง



รูปที่ 2-12 การอินเตอร์เฟส USB กับอุปกรณ์ต่อพ่วง

พอร์ต USB มีอยู่ 4 ขาดด้วยกัน ขาแรกคือขา +5v(VCC), ขาที่สี่คือกราวด์ ซึ่งอุปกรณ์ที่ใช้การสื่อสารแบบ USB นี้จะกินไฟไม่มาก อยู่ในระหว่าง 100mA ถึง 500mA ขาสองและสาม (D+,D-) เป็นขาที่ใช้ส่งสัญญาณแบบซีเรียล และเป็นอินเวอร์ตซึ่งกันและกัน ซึ่งสามารถลดตัวรับกวนลงได้มากเมื่อต้องส่งสัญญาณไปในระยะทางไกล



รูปที่ 2-13 พอร์ต USB แสดงถึงหน้าที่ของขาสัญญาณ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ข้อมูลจะถูกส่งไปตามขา D+ และ D- โดยใช้การบีบอัดข้อมูลแบบ NRZI ด้วยความเร็ว 12 เมกะบิตต่อวินาทีด้วยสายที่มีการหุ้มตัวรบกวนได้ในระยะทางสูงสุด 5 เมตร ส่วนสายที่ไม่มีตัวหุ้มการรบกวนจะต้องมีความเร็วในการส่งน้อยลงเพื่อลดตัวรบกวน



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## บทที่ 3

### ทฤษฎีทางการประมวลผลรูปภาพ

กลไกระบบการมองเห็นภาพนั้น จะหมายความรวมถึงทุกสิ่งที่จะต้องการได้มา ซึ่งรหัส ทางดิจิทัลที่ใช้แทนภาพนั้นๆ การปรับปรุงเปลี่ยนแปลงข้อมูล และรวมทั้งการนำเสนอภาพที่ได้ หลังการปรับปรุงแล้วด้วย ซึ่งระบบการมองเห็นภาพ (Vision System) นี้อาจมีความยุ่งยากซับซ้อน ต่างกันตามลักษณะการใช้งาน แต่ก็สามารถแบ่งเป็นขั้นตอนกว้างๆ ได้ 3 ขั้นตอน ดังนี้

- **การได้มาซึ่งภาพ (Image Acquisition)**

การได้มาซึ่งภาพนี้ จะหมายถึงการแปลงภาพที่เราเห็นในลักษณะทางกายภาพนั้นให้เป็น เซตของข้อมูลทางดิจิทัล โดยในโรงงานนี้จะรับรูปภาพผ่านทางสแกนเนอร์ ทำการแปลงให้เป็น ภาพระดับสีเทา และเก็บค่าความสว่างของแต่ละพิกเซลอยู่ในรูปแบบเลขฐานสองจำนวน 8 บิต ซึ่งเซตของพิกเซลเหล่านี้จะถูกส่งไปยังส่วนประมวลผลภาพต่อไป

- **กระบวนการประมวลผลภาพ (Image Processing)**

หน้าที่พื้นฐานของกระบวนการประมวลผลภาพ ได้แก่ การสร้างภาพใหม่โดยแยกแยะส่วน ที่เราต้องการหรือสนใจออกจากสิ่งรบกวนออกจากกัน (Noise Elimination), การหาขอบภาพ (Edge Enhancement), การกรอง (Filtering), การปรับปรุงเปลี่ยนแปลงค่าระดับสีเทา (Grey scale Modification), การประมวลผลภาพเชิงตัวเลข (Digital Image Processing) หมายถึง การนำภาพมาประมวลผลด้วยเครื่องคอมพิวเตอร์ โดยภาพที่นำมาประมวลผลนี้จะถูกแทนที่ด้วยตัวเลขให้อยู่ในรูปแบบของเมตริกซ์ แต่ภาพที่ได้โดยส่วนมากแล้วจะเป็นภาพที่ได้จากตัวรับสัญญาณ ซึ่งอยู่ในรูปของฟังก์ชันต่อเนื่อง  $f(x, y)$  ในระนาบสองมิติ (คือแกน  $x$  และแกน  $y$ ) โดยจะเป็นสัดส่วนกับความ สว่างหรือความเข้มของภาพ ที่ตำแหน่ง  $(x, y)$  ซึ่งเรียกว่า ระดับสีเทา (Grey Level)

- **ผลที่ได้และการแสดงผล (Output and Display)**

รูปแบบของการแสดงผลที่ได้นั้นขึ้นอยู่กับว่าเราจะนำข้อมูลที่ได้ออกไปใช้ต่อในลักษณะใด โดยอาจแสดงผลทางเครื่องพิมพ์ หรือแสดงผลผ่านทางหน้าจอ เป็นต้น

การประมวลผลรูปภาพสามารถทำได้ในสองลักษณะใหญ่ๆคือ การปรับปรุงคุณภาพของภาพให้ดีขึ้นเพื่อให้มนุษย์ดู และกระบวนการ (process) การนำข้อมูลรูปภาพเพื่อนำไปใช้ต่อในการเก็บ, การส่งข้อมูล และการนำข้อมูลรูปภาพมาแสดงให้มนุษย์เห็นบนเครื่องกลใด ๆ

### 3.1 การแทนภาพด้วยข้อมูลแบบดิจิทัล

ภาพข้อมูลแบบดิจิทัล (Digital Image) เป็นภาพที่ถูกแปลงมาจากภาพอนาล็อกให้อยู่ในรูปของตัวเลข โดยการนำภาพอนาล็อกมาแบ่งเป็นพื้นที่ที่สี่เหลี่ยมเล็ก ๆ ที่เรียกว่า พิกเซล (pixel) ในแต่ละพิกเซลจะถูกระบุตำแหน่งโดย  $(x, y)$  และค่าระดับสีเทาของพิกเซล โดยเราสามารถแปลงภาพเป็นข้อมูลแบบดิจิทัลได้ โดยมีขั้นตอนและวิธีการดังนี้

เมื่อเรานำสัญญาณอนาล็อกที่ต้องการประมวลผลมาผ่านส่วนที่เรียกว่า ดิจิไทเซอร์ (Digitizer) ซึ่งจะมีหน้าที่ในการเปลี่ยนสัญญาณอนาล็อกให้เป็นสัญญาณดิจิทัล จากนั้นทำการควอนไทซ์ (Quantizing) เพื่อที่จะประมวลสัญญาณด้วยระบบคอมพิวเตอร์ ฟังก์ชันของภาพ  $f(x,y)$  จะถูกทำให้เป็นสัญญาณไม่ต่อเนื่องทั้งระนาบของภาพ ซึ่งเราเรียกว่าการสุ่มภาพ (Image Sampling) ของฟังก์ชันที่ได้เรียกว่า การควอนไทซ์ระดับสีเทา (Grey level quantization) ก็จะได้ข้อมูลที่เป็นดิจิทัล

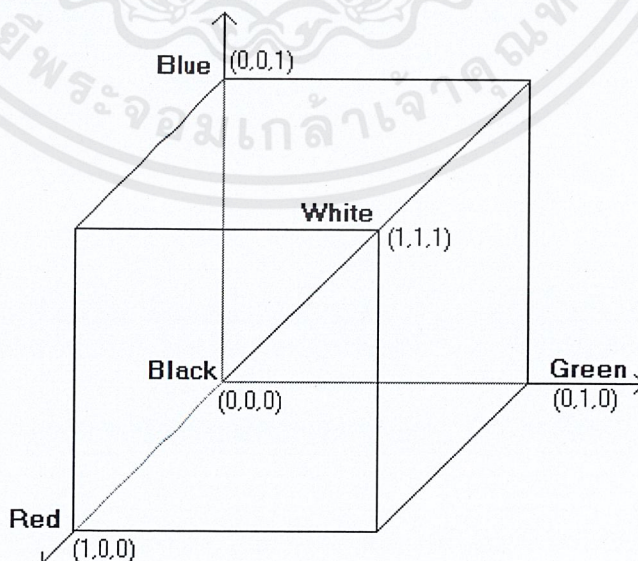
### 3.2 การประมวลผลแบบสเปเชียล (Spatial Domain Processing)

ในการประมวลผลแบบสเปเชียลเราจะแบ่งรูปภาพออกเป็นฟังก์ชันสองตัวแปร  $f(x,y)$  ซึ่ง  $x$  และ  $y$  คือโคออร์ดิเนต (Coordinate) ของภาพนั้นๆ และขนาดของ  $f$  ณ ทุกๆ โคออร์ดิเนต  $(x,y)$  คือความเข้มสี (Intensity) ของรูปภาพ ณ จุดนั้นๆ ทุกๆ  $x,y$  และ  $f$  มีขนาดที่จำกัดและเป็นฟังก์ชันต่อเนื่อง (Discrete Function) เราจะเรียกรูปภาพนั้นๆว่ารูปภาพดิจิทัล ขอบเขตของการประมวลผลรูปภาพทางดิจิทัลคือการประมวลผลโดยใช้ดิจิทัลคอมพิวเตอร์

### 3.3 ทฤษฎีสี RGB, YCbCr และการเปลี่ยนรูป

#### 3.3.1 สี RGB

เป็นสีที่อยู่บนโคออร์ดิเนตของคาร์ทีเซียน (Cartesian Coordinate System) มีแม่สีหลักอยู่ด้วยกันสามสีคือ แดง เขียว และน้ำเงิน

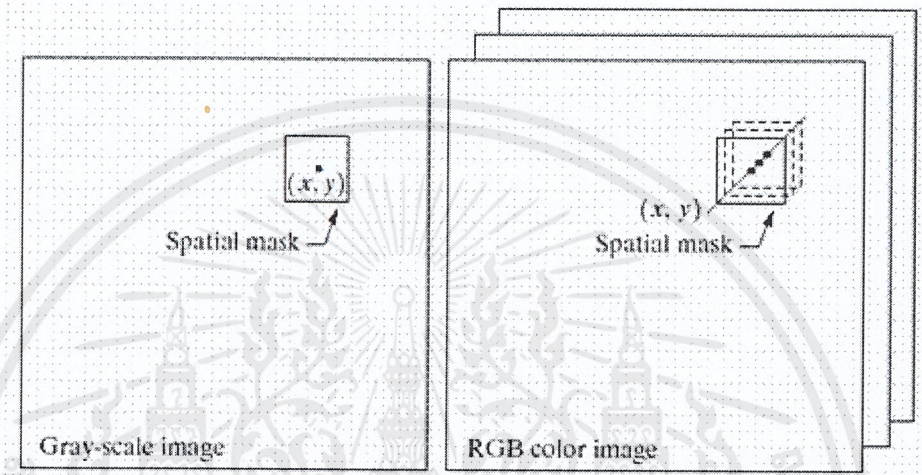


รูปที่ 3-1 โคออร์ดิเนตคาร์ทีเซียนของสี RGB

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปภาพที่แสดงด้วย RGB จะประกอบด้วยรูปเกรย์สเกล (Grey scale) ทั้งหมดสามเลเยอร์ (Layer) เข้าด้วยกัน คือเลเยอร์ของสีแดง เลเยอร์ของสีเขียว และเลเยอร์ของสีน้ำเงินเข้ามารวมกันเป็นหนึ่งเลเยอร์ และสามารถแสดงทั้งสีแดง เขียว และน้ำเงินได้

$$f(x,y) \text{ ในโหมด RGB ณ จุดใดๆ } = \{R,G,B\}$$



รูปที่ 3-2 เปรียบเทียบเลเยอร์ระหว่างรูปเกรย์สเกลและรูปภาพ RGB

จำนวนของสีที่สามารถแสดงได้ จะแปรตามจำนวนของบิตที่ใช้แสดงค่าความเข้มของแต่ละสีซึ่งเรียกว่า Pixel Dept รูปภาพที่มีจำนวน pixel dept 8 บิตหมายถึงรูปภาพที่มีจำนวนบิตของแต่ละสีเท่ากับ 8 บิต รวมเข้าด้วยกันสามสี ตามรูปที่ 3-2 จะกลายเป็นภาพขนาด 24บิต สามารถแสดงจำนวนสีได้ =  $(2^8)^3 = 16,777,216$  สี จะมีค่าของความเข้มของแต่ละสีของทั้ง 3 สี เป็น 256 สี แสดงได้ระหว่าง 0-255 ดังตาราง

Number System	Colour Equivalent					
Decimal	0	51	102	153	204	255
Hex	00	33	66	99	CC	FF

ตารางที่ 3-1 ตัวอย่างค่าของสีต่างๆใน RGB

### 3.3.2 ระดับสีเทา (Grey Scale)

หากเราต้องการค่าข้อมูลที่ละเอียดมากขึ้น ก็จำเป็นที่จะต้องเพิ่มจำนวนบิตการแสดงค่าของแต่ละพิกเซล ยกตัวอย่างเช่น หากแบ่งความเข้มของการส่องสว่างให้มี 4 ระดับ ก็ต้องใช้เลขฐานสองจำนวน 2 บิต และจำนวน 4 บิตสำหรับ 16 ระดับ, 8 บิตสำหรับ 256 ระดับ ซึ่งจำนวนระดับที่ใช้ในระดับสีเทาหรือระดับเกรย์นั้นมักเป็นเลขยกกำลังของ 2, ค่าที่ต่ำที่สุดคือ 0 กำหนดให้เป็นสีดำ และ 1 หรือตัวเลขที่น้อยกว่า

ค่าสูงสุดของระดับเกรย์อยู่ 1 (เช่น 15 สำหรับระดับเกรย์ 16 ระดับ) แทนสีขาว ค่าที่กำหนดไว้ในแต่ละเอกสารนี้เป็นเอกสารทศวรรษวิสาห์สำหรับการเรียนเพื่อการศึกษาเท่านั้น ไม่นับญาติเห็นาไปเซประเขยขนดานการค้ำ

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

พิกเซลนี้นิยมใช้เป็นจำนวนเต็ม ในยุคแรก ๆ ของระบบการมองเห็นภาพจะใช้ระบบเลขฐานสอง แต่ในปัจจุบันเทคโนโลยีไมโครโปรเซสเซอร์เข้ามามีบทบาทมากขึ้น การแบ่งระดับเป็น 16, 64 หรือ 256 เป็นเรื่องธรรมดา แต่ทั้งนี้ในการมองเห็นของมนุษย์จะสามารถแยกแยะความแตกต่างได้เพียง 10-15 ระดับเท่านั้น การแบ่งโดยละเอียดเป็น 64 หรือ 256 ระดับอาจนำไปประยุกต์ใช้กับงานการประมวลผลภาพหรือกระบวนการทางอุตสาหกรรมอื่นๆ ซึ่งจะเห็นว่าจำนวนระดับเกรย์จะเป็นตัวจำกัดรายละเอียดของภาพโดยทั่วไปแล้วยังแบ่งระดับเกรย์เป็นหลายระดับก็จะเป็นการเพิ่มคุณภาพของภาพด้วย และการเพิ่มจำนวนพิกเซล เช่น จาก  $35 * 35$  เป็น  $250 * 250$  ก็จะเป็นการเพิ่มความละเอียด (Resolution) และรายละเอียด (Detail) ของภาพเช่นกัน จะเห็นว่าจะแตกต่างกับการขยายภาพ (Zoom) โดยการขยายภาพนี้เป็นการขยายแต่ละพิกเซลให้ใหญ่ขึ้น ไม่ได้เป็นการเพิ่มจำนวนพิกเซล

### 3.3.3 สี YCbCr

เป็นแม่สีที่นิยมใช้กันอย่างแพร่หลายในกล้องถ่ายรูป กล้องถ่ายวิดีโอ และกล้องวงจรปิดต่างๆ ที่บันทึกในรูปแบบดิจิทัล โดยจะแบ่งออกเป็นแม่สีสามขั้วเหมือนกับ RGB ขั้วสีแต่ละขั้วจะแสดงข้อมูลของในพิกเซลนั้นๆ คือ

- ขั้วสีที่กำหนดความเข้มของแสง
- ขั้วสีที่กำหนดความเข้มของสี

แม่สีที่เป็นตัวกำหนดความเข้มของแสงในพิกเซลนั้นๆ โดยเฉพาะ ซึ่ง

Y จะเป็นค่าความเข้มแสง (Luminance)

Cb แสดงถึงส่วนของโทนสีน้ำเงิน โดยไม่มีความเข้มแสงมาเกี่ยวข้อง

Cr แสดงถึงส่วนของโทนสีแดง โดยไม่มีความเข้มแสงมาเกี่ยวข้อง

### 3.3.4 การเปลี่ยนรูประหว่าง RGB และ YCbCr

การแปลงรูประหว่าง RGB และ YCbCr สามารถแปลงโดยใช้หลักการของเมทริกซ์และจะได้สมการดังต่อไปนี้

$$f(x,y) \text{ ใน โหมด RGB ณ จุดใดๆ } = \{R,G,B\}$$

$$f(x,y) \text{ ใน โหมด YCbCr ณ จุดใดๆ } = \{Y,Cb,Cr\}$$

$$Y = 0.299 R + 0.587 G + 0.114 B$$

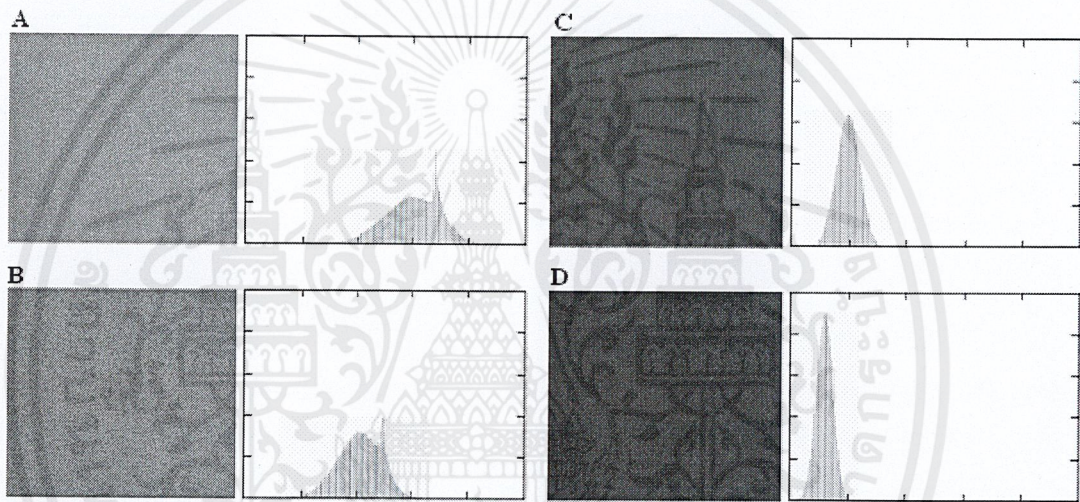
$$Cb = -0.1687 R - 0.3313 G + 0.5 B + 128$$

$$Cr = 0.5 R - 0.4187 G - 0.0813 B + 128$$

### 3.4 ฮิสโตแกรม (Histogram)

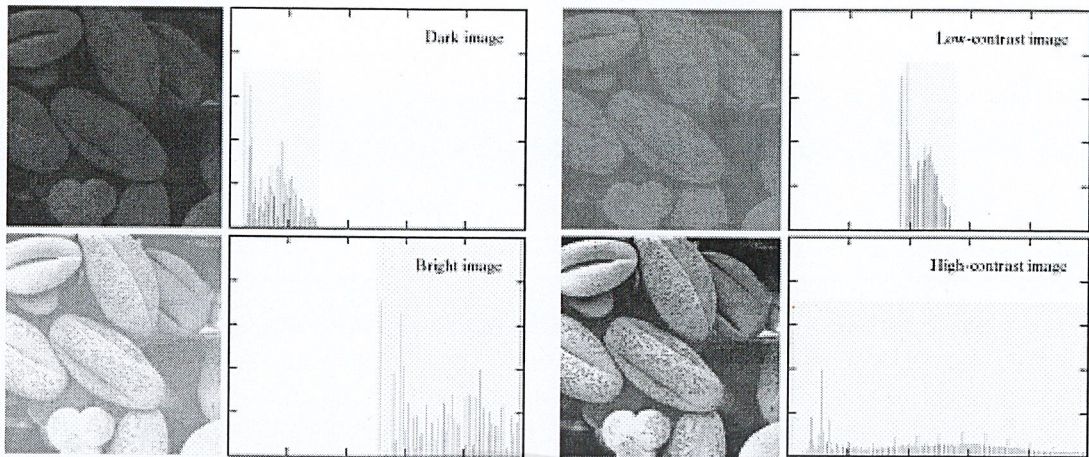
ฮิสโตแกรมของรูปภาพคือกลุ่มของผลรวมของความเข้มสีของพิกเซลทั้งหมดในรูป มีค่าตั้งแต่  $[0, \text{ขนาดภาพ}-1]$  ในฟังก์ชันต่อเนื่อง  $h(r_k) = n_k$  ซึ่ง  $r_k$  เท่ากับความเข้มแสง  $k$  หน่วยในภาพ  $n_k$  เท่ากับจำนวนพิกเซลทั้งหมดในภาพที่มีความเข้มแสงเท่ากับ  $r_k$  รูปภาพที่มีการกระจายของฮิสโตแกรมที่สม่ำเสมอจะแสดงรายละเอียดได้ชัดกว่ารูปถ่ายที่ฮิสโตแกรมแคบและกระจุกอยู่ใกล้ๆกัน

ฮิสโตแกรมเป็นเทคนิคพื้นฐานที่สุดของการประมวลผลรูปภาพในสเปซเกรย์โดเมน ฮิสโตแกรมมีประโยชน์มากทางการปรับรูปภาพที่มีแสงมากหรือน้อยเกินไปและดูด้วยตาเปล่าไม่ชัดให้อยู่ในระดับที่สามารถดูด้วยตาเปล่าได้ชัดเจน อีกทั้งยังสามารถใช้เป็นเทคนิคร่วมกับการบีบอัดรูป และการตัดแบ่งส่วนรูป (Image Compression and Image Segmentation)



รูปที่ 3-3 รูปภาพที่มีช่วงของฮิสโตแกรมต่างกัน

รูปภาพที่มีฮิสโตแกรมกระจายอย่างสม่ำเสมอจะส่งผลให้คอนทราสต์ (Contrast) หรือความชัดเจนสูง ในขณะที่เดียวกันรูปภาพที่ฮิสโตแกรมไม่กระจายอย่างสม่ำเสมอ แต่อยู่ที่ด้านใดด้านหนึ่งมากเป็นพิเศษจะเรียกว่ารูปภาพที่มีคอนทราสต์ต่ำ ซึ่งมองด้วยตาเปล่าแล้วจะแยกรายละเอียดของภาพได้ยากกว่าภาพที่มีคอนทราสต์สูง



รูปที่ 3-4 ความสัมพันธ์ระหว่างฮิสโตแกรมและคอนทราสต์

### 3.4.1 การปรับฮิสโตแกรมของรูปภาพ

เมื่อเราทราบความสัมพันธ์ระหว่างฮิสโตแกรมและคอนทราสต์แล้ว การจะทำให้รูปภาพมีความคมชัดขึ้นจึงเป็นเรื่องที่สามารถทำได้ด้วยฮิสโตแกรม ทำได้ด้วยการบีบอัตราส่วนของกลุ่มสีให้กระจายออกอย่างสม่ำเสมอให้เต็มรูปภาพ

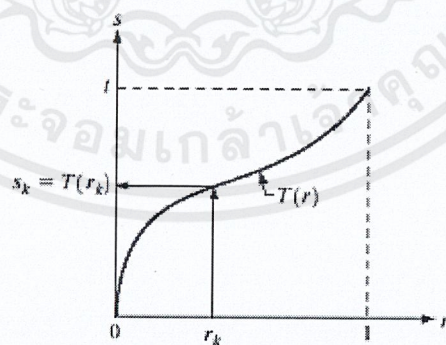
$$s = T(r) \quad ; \quad 0 \leq r \leq 1$$

เมื่อ  $r$  = จำนวนความเข้มสีทั้งหมดที่แสดงได้ในรูปต้นฉบับ อยู่ในเรนจ์  $[0, 1]$

○ พิกเซลนั้นจะเป็นสีดำเมื่อ  $r=0$

○ พิกเซลนั้นจะเป็นสีขาวเมื่อ  $r=1$

$T(r)$  = จำนวนพิกเซลทั้งหมดของรูปภาพที่มีค่าความเข้มแสงเท่ากับ  $r$



รูปที่ 3-5 ความสัมพันธ์ระหว่าง  $S$  และ  $T(r)$



รูปที่ 3-6 รูปต้นฉบับที่มีฮิสโตแกรมต่างกันที่ความเข้มแสงจะถ่าย เมื่อนำมาปรับฮิสโตแกรมแล้ว

จากรูป 3-6 จะเห็นได้ว่ารูปต้นฉบับที่ถูกถ่ายด้วยสภาพแวดล้อมและความเข้มแสงที่ต่างกัน เมื่อนำมาทำการปรับฮิสโตแกรมให้กระจายอย่างสม่ำเสมอแล้ว จะมีคอนทราสต์ หรือความคมชัดที่ดูได้ด้วยตาเปล่า มากขึ้น เทคนิคการปรับฮิสโตแกรมนี้จึงเป็นเทคนิคที่นิยมใช้กันอย่างแพร่หลาย

### 3.5 การนอร์มัลไลเซชันของสี RGB (Normalization)

ค่านอร์มัลไลซ์ของสี RGB เป็นการแสดงอัตราส่วนและความคมชัดของสีแต่ละสี

$$f(x,y) \text{ ณ จุดใดๆ} = \{R,G,B\}$$

$$f(x,y) \text{ นอร์มัลไลซ์} = \{r,g,b\} \text{ ซึ่ง}$$

$$r = \frac{R}{R+G+B}$$

$$g = \frac{G}{R+G+B}$$

$$b = \frac{B}{R+G+B}$$

### 3.6 การทำเทรชโฮล (Threshold)

อุปกรณ์ที่มีความสามารถในการแสดงผลได้แค่ 2 ระดับ หรือ 2 สี คือ สีขาวกับสีดำยังมีการใช้กันอย่างแพร่หลาย เช่น เครื่องพิมพ์ (Printer), เครื่องโทรสาร (Fax), จอภาพแสดงผลแบบโมโนโครม (Monochrome Monitor) เป็นต้น เนื่องจากอุปกรณ์เหล่านี้เป็นอุปกรณ์ที่มีราคาถูก ดังนั้นจึงเกิดปัญหาในการแสดงผลขึ้น การที่จะแสดงผลหรือพิมพ์รูปภาพที่มีระดับความเข้มของภาพหลายระดับซึ่งมีมากกว่าความสามารถในการแสดงผลของอุปกรณ์เหล่านี้ที่มีเพียงแค่ 2 ระดับ เราจะต้องทำการแปลงข้อมูลภาพให้เป็นภาพไบนารี (Binary Image) เสียก่อน ซึ่งการสร้างภาพไบนารีนั้นก็หมายถึงการแปลงข้อมูลภาพที่มีระดับความเข้มหลายระดับ (Multilevel Image) ให้เป็นภาพที่มีระดับความเข้มเพียง 2 ระดับ นั่นคือ 1 จุดภาพ มีได้ 2 ค่าเท่านั้น คือ 0 กับ 1 โดยจุดภาพที่แทนด้วย 0 หมายถึงจุดภาพที่มีสีดำ ส่วนจุดที่แทนด้วยค่า 1 จะหมายถึงจุดภาพที่มีสีขาว เมื่อทำการแปลงเป็นภาพไบนารีแล้ว จึงสามารถนำภาพนั้นไปแสดงผลบนอุปกรณ์เหล่านั้น จะเห็นได้ว่าการแปลงข้อมูลภาพหลายระดับเป็นภาพไบนารีจึงมีความจำเป็นและมีประโยชน์มากในการแสดงผลภาพที่มีความเข้มของภาพหลายระดับบนอุปกรณ์ที่มีความสามารถในการแสดงผลได้ 2 ระดับ สำหรับประโยชน์อีกประการหนึ่งในการแปลงข้อมูลภาพนั้นเป็นภาพไบนารี คือการลดเนื้อที่การเก็บข้อมูลภาพ ภาพไบนารีนี้จะสามารถลดเนื้อที่ลงได้ถึง 8 เท่า นั่นคือถ้าเดิม 1 จุดภาพใช้เนื้อที่ในการเก็บ 8 บิต เมื่อทำการแปลงเป็นภาพไบนารีแล้วจะใช้เนื้อที่ในการเก็บ 1 จุดภาพนั้นเพียง 1 บิตเท่านั้น อีกทั้งยังสามารถนำไปประยุกต์ใช้งานได้อย่างแพร่หลาย เช่น นำไปประยุกต์ใช้ในการวิเคราะห์เอกสารในขั้นตอนที่เรียกว่าการประมวลผลขั้นต้น (Preprocessing) เป็นต้น

ในการสร้างภาพไบนารี สามารถทำได้โดยใช้เทคนิคการทำเทรชโฮล (Thresholding Technique) โดยพิจารณาว่าจุดภาพใดควรจะเป็นจุดขาวหรือจุดดำ จะกระทำโดยการเปรียบเทียบระหว่างจุดภาพเริ่มต้นกับค่าคงที่ค่าหนึ่งซึ่งเรียกว่า ค่าเทรชโฮล (Threshold Value) เทคนิคนี้ใช้กันมากในกรณีที่มีข้อมูลภาพมีลักษณะแตกต่างกันระหว่างวัตถุ (Object) และพื้นหลัง (Background) โดยค่าของจุดภาพใด ๆ ที่มีค่าน้อยกว่าค่าเทรชโฮลจะถูกกำหนดให้เป็น 0 (จุดดำ) และถ้าค่าของจุดภาพใด ๆ ที่มีค่ามากกว่าหรือเท่ากับค่าเทรชโฮล จะถูกเปลี่ยนให้เป็น 1 (จุดขาว)

ในการสร้างภาพไบนารีโดยใช้เทคนิคเทรชโฮลเพื่อให้ได้ผลลัพธ์ที่เหมาะสมและคมชัด สิ่งที่สำคัญที่สุดคือ ค่าเทรชโฮล เนื่องจากถ้าเลือกค่าเทรชโฮลที่ไม่เหมาะสม (ค่าเทรชโฮลที่มีค่าน้อยเกินไปหรือมากเกินไป) ภาพที่ได้อาจจะไม่เหมาะสม ขาดความคมชัด และอาจทำให้รายละเอียดบางส่วนขาดหายไป กล่าวคือภาพที่ได้อาจจะมืดเกินไปหรือสว่างเกินไป หรือภาพที่ได้มีสิ่งรบกวน (Noise) เกิดขึ้น อันเป็นผลทำให้ภาพผลลัพธ์ที่ได้ไม่สวยงามเท่าที่ควร ดังนั้นปัญหาของการสร้างภาพไบนารีโดยวิธีเทรชโฮลนี้คือ ทำอย่างไรจึงจะสามารถคำนวณหาค่าเทรชโฮลที่เหมาะสมสำหรับแต่ละภาพที่จะนำมาทำการสร้างภาพไบนารี ซึ่งมีวิธีคำนวณหาค่าเทรชโฮลหลายวิธี โดยแต่ละวิธีเหมาะสมกับลักษณะการทำงานที่แตกต่างกันไป เช่น

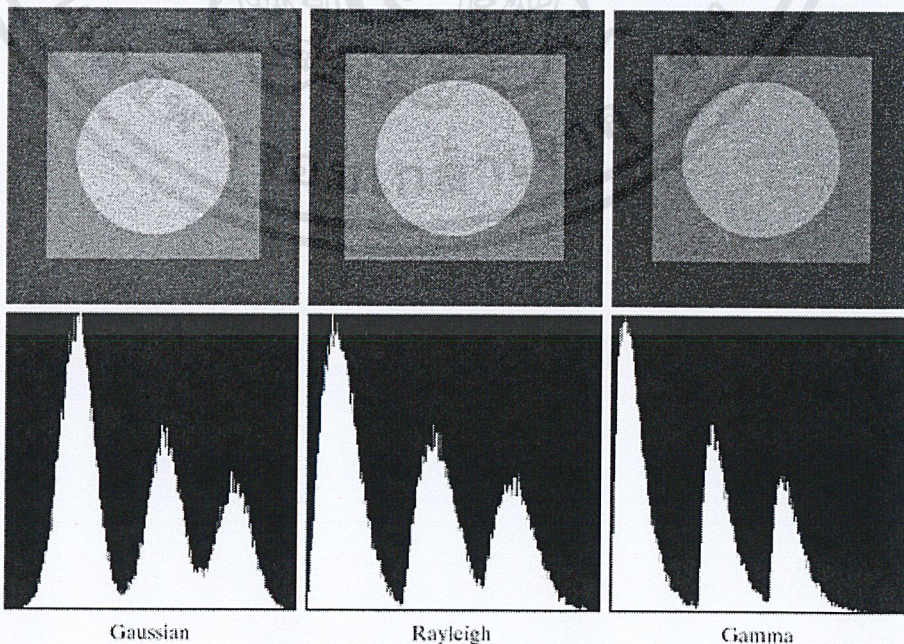
- การหาค่าเทรชโฮลโดยการกำหนดค่าล่วงหน้า (Pre-assigned Threshold Value)
- การหาค่าเทรชโฮลจากค่ากลาง (Mid-range Threshold Value)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 3.7 ตัวรบกวนภาพชนิดต่างๆและการกำจัด (Noise Model and Noise Reduction)

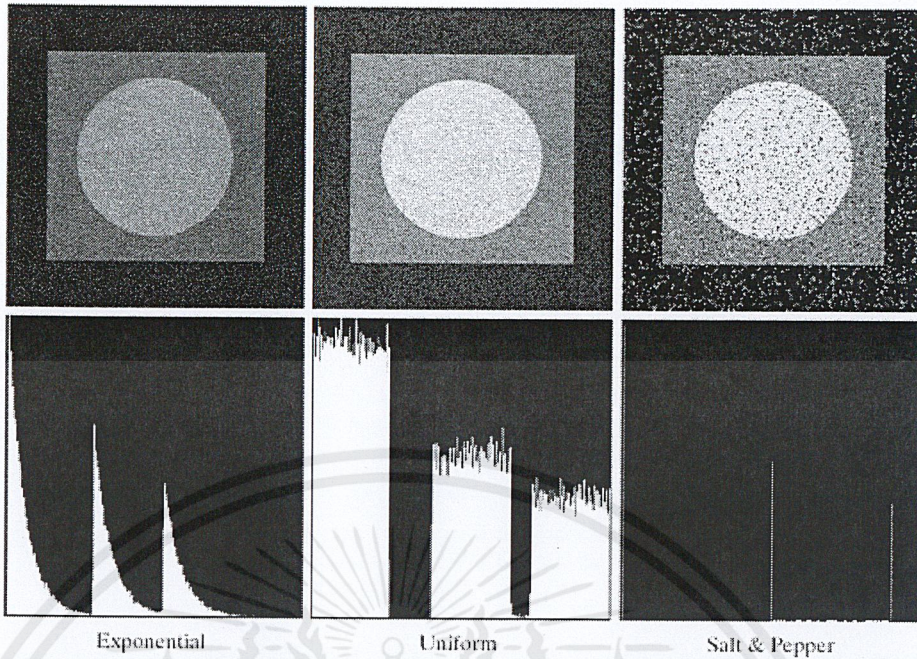
เมื่อภาพที่ถ่ายถูกรบกวนด้วยสิ่งแวดล้อมไม่ว่าจะในขณะที่ถ่ายภาพ หรือขณะที่ส่งข้อมูลภาพไปที่ต่างๆ เช่นภาพถ่ายจากดาวเทียมที่ส่งกลับมายังโลก จะรบกวนให้ภาพผิดเพี้ยนไปจากภาพต้นฉบับ มีการจำแนกประเภทตัวรบกวนต่างๆออกเป็น 6 ชนิดใหญ่ๆดังนี้

- **เกาเซียน (Gaussian Noise)**  
เกิดจากการรบกวนของวงจรอิเล็กทรอนิกส์ ทำให้ภาพที่ถ่ายได้เกิดการผิดเพี้ยนไป
- **รีเลย์ (Rayleigh Noise)**  
พบมากจากภาพถ่ายที่ต้องถ่ายระยะทางไกล เช่นภาพถ่ายทางอากาศ
- **แกมมา (Gamma Noise)**  
พบมากจากภาพที่ถ่ายด้วยเลเซอร์
- **เอกซ์โปเนนเชียล (Exponential Noise)**  
พบมากจากภาพที่ถ่ายด้วยเลเซอร์
- **ยูนิฟอร์ม (Uniform Noise)**  
เป็นตัวรบกวนที่มีการกระจายตัวอย่างเท่าเทียมกัน ไม่ค่อยเป็นที่พบเห็นนัก
- **ซอลท์แอนด์เปปเปอร์ (Salt and Pepper Noise)**  
เกิดจากการสปีคของเซนเซอร์ทำให้เกิดจุดสีขาวและสีดำขึ้นทั่วทั้งรูป



รูปที่ 3-7 ตัวรบกวนภาพชนิดต่างๆ

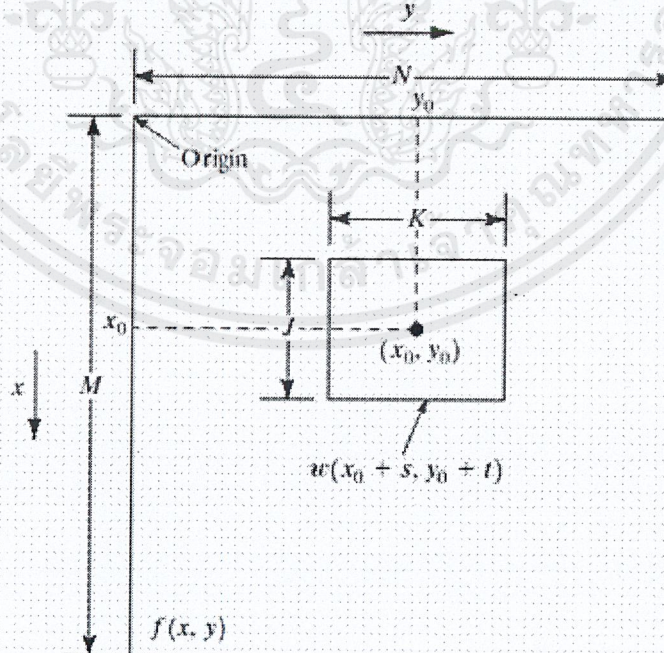
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3-7 ตัวรบกวนภาพชนิดต่างๆ (ต่อ)

### 3.7.1 การทำฟิลเตอร์เพื่อกำจัดตัวรบกวนภาพ

การทำฟิลเตอร์ทำได้โดยจะใช้วินโดว์ขนาดเล็กๆ ที่สร้างขึ้นมา โดยที่ค่าในแต่ละพิกเซลของวินโดว์จะเป็นตัวกำหนดพิกเซลของรูปที่ผ่านการฟิลเตอร์แล้ว โดยพิกเซลรอบข้างของภาพต้นแบบจะมีผลต่อพิกเซลใหม่ในภาพที่ผ่านการฟิลเตอร์แล้ว



รูปที่ 3-8 การทำฟิลเตอร์ภาพขนาด  $(x,y)$  ด้วยวินโดว์ขนาด  $(j,k)$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 3.7.1.1 ฟิลเตอร์ค่ามัธยฐาน (Median Filtering)

การคำนวณค่ามัธยฐานทำได้โดยนำค่าทั้งหมดในวินโดว์มาเรียงลำดับจากมากไปน้อย และนำค่ากลาง ซึ่งก็คือค่ามัธยฐานมาเป็นพิกเซลที่ผ่านการกรองแล้ว ดังสมการ

$$g(x,y) = \text{med} (f(x+i,y+j))$$

$$i = -a \text{ to } a$$

$$j = -b \text{ to } b$$

$a$  และ  $b$  คือจำนวนพิกเซลที่นำมาหาค่าเฉลี่ย ปกติแล้วจะเท่ากับความกว้างและความสูงของวินโดว์

### 3.8 การขยายขอบเขตของภาพ (Region Growing)

การขยายขอบเขตของภาพเป็นการขยายส่วนย่อยๆ ในภาพให้เป็นกลุ่มใหญ่ๆ ขึ้น โดยนำกลุ่มของพิกเซลย่อยๆ ที่มีคุณลักษณะคล้ายกันนำมารวม ตามข้อกำหนดที่ได้กำหนดไว้ล่วงหน้า

### 3.9 การแปลงภาพทางเรขาคณิต (Geometric Transformation)

ในการเปลี่ยนขนาดหรือมุมมองของภาพที่มีอยู่ให้มีขนาดใหญ่ขึ้น หรือหมุนให้มีมุมมองที่พอเหมาะต่อการนำไปประมวลผล สามารถใช้ความรู้ทางเรขาคณิตโดยใช้สมการในรูปแบบต่อไปนี้

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} a_0 & a_1 \\ b_0 & b_1 \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} + \begin{bmatrix} a_2 \\ b_2 \end{bmatrix}$$

#### 3.9.1 การปรับขนาดของภาพ (Image Resizing)

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} s & 0 \\ 0 & s \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix}$$

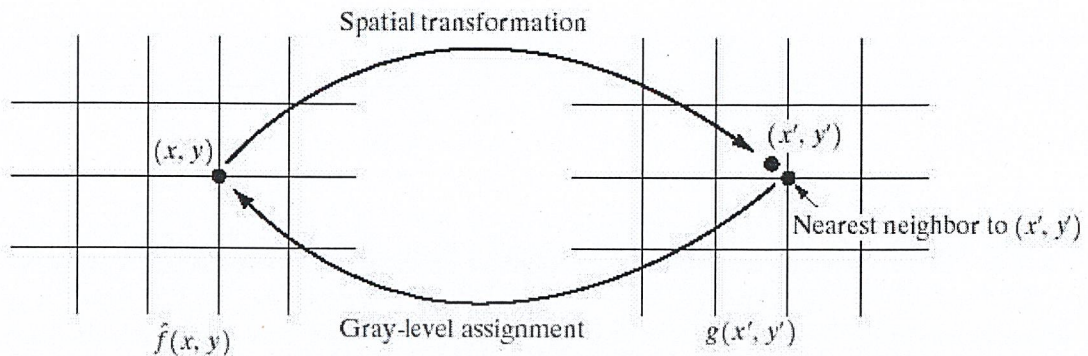
#### 3.9.2 การหมุนภาพ (Image Rotation)

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} \cos \theta & -\sin \theta \\ \sin \theta & \cos \theta \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix}$$

#### 3.9.3 ปัญหาของ Forward Mapping และ Backward Mapping

ในการแปลงตำแหน่งจากจุดเดิม  $(x, y)$  ไปเป็นจุดใหม่  $(x', y')$  โดยวิธีทางเรขาคณิตนั้นถ้าเราแปลงโดย ใช้วิธีคำนวณ โดยเริ่มต้นจากจุดเดิมแปลงไปเป็นจุดใหม่หรือการแปลงภาพไปด้านหลัง

(forward mapping)



รูปที่ 3-9 การแปลงภาพไปด้านหน้า และกลับหลัง

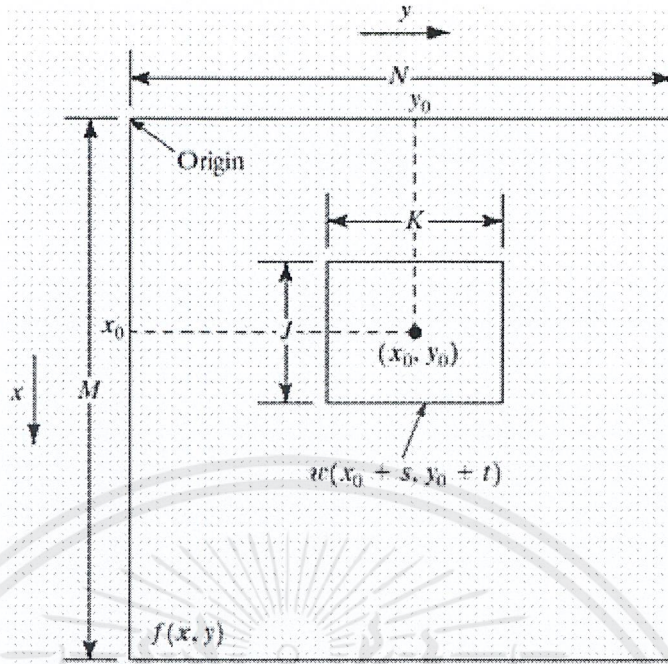
ดังนั้นผลที่ได้ในจุดใหม่  $(x', y')$  อาจทำให้มีบางจุด  $x', y'$  ที่ไม่ได้รับการแปลงมา ทำให้ภาพขาดหายไป รวมทั้งอาจมีบางจุดที่ แปลงไปซ้ำสองครั้ง ทำให้ภาพที่ได้ออกมาผิดเพี้ยนจากที่ควรเป็นมีแนวทางแก้ไข คือ ใช้ จุดใหม่แล้วแปลงกลับไปหาจุดของภาพต้นแบบ หรือการแปลงภาพกลับหลัง(backward mapping) โดยใช้สมการกลับกับสมการเดิม

### 3.10 การทำแมตชิ่งโดยวิธีการคอร์เรเลชัน (Template matching using Correlation)

วิธีการนี้จะใช้ภาพต้นแบบขนาดเล็ก  $w(x,y)$  ขนาด  $J \times K$  ภายใต้ภาพใหญ่  $f(x,y)$  ซึ่งมีขนาด  $M \times N$  โดยที่  $J \leq M$  และ  $K \leq N$  โดยสามารถอธิบายโดยใช้สมการคอร์เรเลชัน การหาค่าคอร์เรเลชันแบบที่ง่ายที่สุด ระหว่างค่า  $f(x,y)$  กับ  $w(x,y)$  คือ

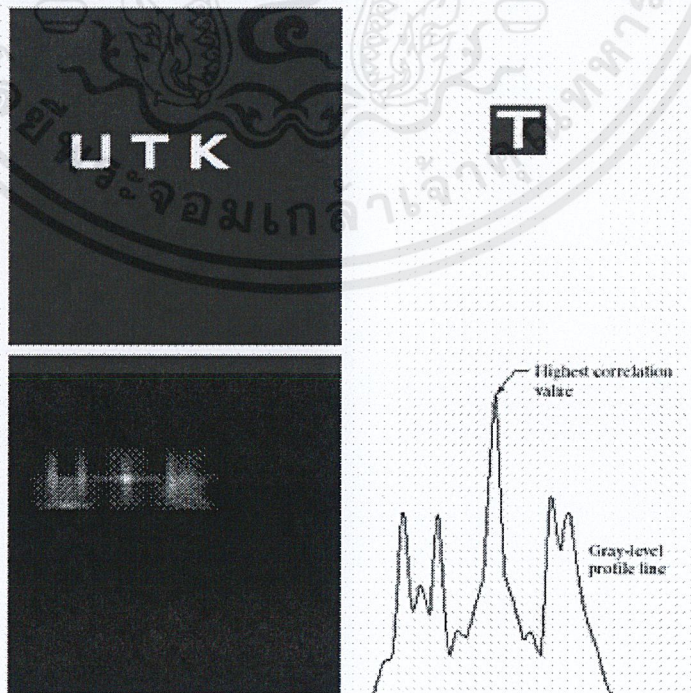
$$c(x,y) = \sum \sum f(s,t) \cdot w(x+s,y+t)$$

โดย  $x = \{0,1,2,\dots,M-1\}$  และ  $y = \{0,1,2,\dots,N-1\}$  โดยค่าผลรวมในสมการเป็นค่าผลรวม ณ ตำแหน่งที่ภาพ  $w$  กับ  $f$  ซ้อนทับกัน



รูปที่ 3-10 แสดงตำแหน่งของวินโดวในการคอร์เรเลตแมตซ์

จากรูป สมมติให้ตำแหน่งจุดออริจิน(origin) ของภาพ  $f(x)$  อยู่ที่บนซ้ายและ ของภาพ  $w(x)$  อยู่ที่ตรงกลางภาพ ค่า ณ ตำแหน่งใกล้เคียงที่สุดจะได้ค่าสูงที่สุด โดย จากภาพ ณ ตำแหน่งด้านบนของตัว T จะได้ค่าคอร์เรลชัน สูงที่สุด แต่วิธีการนี้จะมีปัญหาคือใช้เวลามากเนื่องจากจะต้องนำภาพเล็กแทนกับภาพใหญ่ทั้งภาพ และมีปัญหาในเรื่องที่ลักษณะของรูปแบบที่ต้องการจะจดจำไม่ควรจะเปลี่ยนแปลงไปมากนักอาจจะทำให้จดจำรูปแบบผิดได้



รูปที่ 3-11 ผลของการคอร์เรเลตแมตซ์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากผลที่ได้จะเห็นว่าตำแหน่งสูงที่สุดเมื่อนำวินโดว์ T ไปวิ่งในรูปจะเป็นตำแหน่งที่ตัวอักษร T อยู่ และจะเห็นได้อย่างค่อนข้างชัดเจน แต่การทำคอริเลตนี้หากตัวอักษรเปลี่ยนแปลงไปเช่นมีขนาดที่ไม่เท่าเดิม มุมองศาที่เปลี่ยนไป ก็จะทำให้ผลการทำคอริเลตนี้ผิดเพี้ยนไปได้อย่างง่ายดาย

### 3.11 หลักการจดจำรูปแบบโดยใช้วิธีสโตรป (Pattern Recognition by Strobes)

หลักการสโตรปคือเป็นการจดจำรูปแบบโดยใช้การนับค่าเฉลี่ยของระดับการเปลี่ยนแปลงของสีในแต่ละแถว โดยถ้าต้องการให้ละเอียดมากขึ้นก็สามารถทำการแบ่งส่วนออกมาเพื่อจุดสนใจในแต่ละส่วนได้ ในขั้นแรกจะต้องมีโดเมนของส่วนที่นำมาทำสโตรป



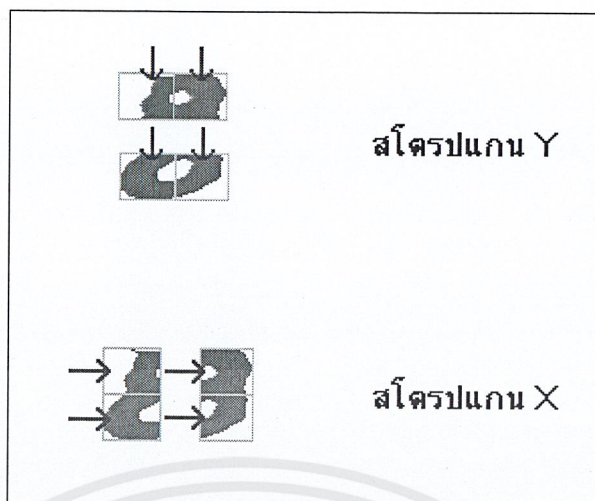
รูปที่ 3-12 การแบ่งหลักของตัวเลข

และนำตัวอักษรที่จะนำมาตรวจสอบสโตรปมาตัดแบ่งออกเป็นช่องในแนวตั้งและแนวนอน เท่าๆกัน ทั้งนี้การทำสโตรปไม่มีจำนวนของช่องที่ตายตัว แต่การแบ่งเป็นช่องจำนวนมากเท่าใดก็就会有ความถูกต้องมากขึ้นเท่านั้น แต่ก็จะมีจำนวนช่องของสโตรปที่ต้องวิเคราะห์มากขึ้นด้วย จำนวนช่องของสโตรปมีได้ตั้งแต่  $1 \times 1$ ,  $2 \times 2$ ,  $3 \times 3$ ,  $4 \times 4$ , ... เป็นต้นไป ในตัวอย่างนี้จะเป็นการทำสโตรปด้วยช่องขนาด  $2 \times 2$



รูปที่ 3-13 สโตรปขนาด  $2 \times 2$

หลังจากนั้นจึงคำนวณจำนวนครั้งของการทริกเกอร์ทั้งสองแกนของแต่ละช่อง เมื่อค่าสีเปลี่ยนจาก “0” ไปเป็น “1” หรือจากสีขาวเป็นสีดำ ค่าสโตรปทั้งสองแกนของแต่ละช่องจะเป็นค่าเฉลี่ยของแต่ละตัวอักษรที่นำมาทำสโตรป



รูปที่ 3-14 การแบ่งช่องของสโตรป

หลังจากคำนวณได้จำนวนสโตรปทั้งสองแกนของทุกช่องแล้วเมื่อนำไปเทียบกับตัวอักษรต้นแบบ จะมีรูปแบบที่คล้ายคลึงกัน

		62	150
		85	148
65	115	อ	
105	65		

รูปที่ 3-15 ค่าสโตรปทั้งแกนตั้งและแกนนอนของทุกช่อง

## บทที่ 4

### การออกแบบฮาร์ดแวร์

แนวคิดในการออกแบบตัวหุ่นยนต์

เมื่อพิจารณาถึงหน้าที่ของตัวหุ่นยนต์คือ

- ต้องเดินไปส่งเอกสาร
- ต้องใช้กล้องถ่ายรูปทางเดินต่างๆและป้ายห้อง
- ต้องรับอินพุตหมายเลขห้องปลายทางได้

จึงออกแบบคอมโพเนนต์ของหุ่นยนต์ออกเป็น 5 ส่วนตามหน้าที่ได้ดังนี้



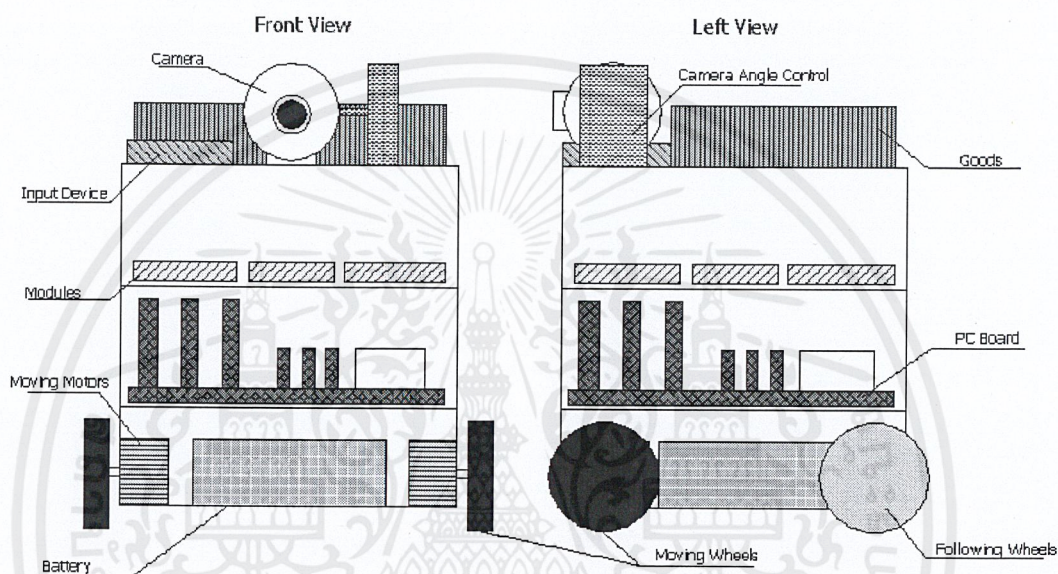
รูปที่ 4-1 คอมโพเนนต์ของหุ่นยนต์

มีคอมโพเนนต์หลักอยู่หนึ่งคอมโพเนนต์และมีคอมโพเนนต์ย่อยอยู่อีก 4 คอมโพเนนต์ แต่ละส่วนจะถูกควบคุมด้วยคอมโพเนนต์หลัก ซึ่งก็คือส่วนประมวลผลกลาง ซึ่งหน้าที่ของแต่ละคอมโพเนนต์จะแจกแจงตามหัวข้อต่อไป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## แนวคิดในการออกแบบโครงสร้างของตัวหุ่น

การออกแบบโครงสร้างของหุ่นยนต์จะเน้นที่ความแข็งแรงและความง่ายในการประกอบ จึงออกแบบให้เป็นโครงรูปสี่เหลี่ยมด้านเท่า คล้ายลูกเต๋า และออกแบบให้ภายในโครงแบ่งออกเป็นชั้นสองชั้น เพื่อที่จะวางอุปกรณ์และวงจรต่างๆ ได้อย่างเพียงพอและเป็นระเบียบ ได้เลือกใช้โซลูชันนิยม ซึ่งมีคุณสมบัติเด่นคือความแข็งแรงสามารถรับน้ำหนักได้มากและตัววัสดุมีน้ำหนักเบา มาทำเป็นโครง ประกอบกับแผ่นไม้อัดซึ่งตัดเป็นรูปทรงใด้งานพร้อมทั้งมีน้ำหนักไม่มาก



รูปที่ 4-2 การออกแบบโครงสร้างของหุ่นยนต์

### 4.1 ส่วนประมวลผลกลาง

ทำหน้าที่ประมวลผลและควบคุมคอมพิวเตอร์อื่นทั้งหมด ได้เลือกใช้พีซีบอร์ดมาทำหน้าที่ในส่วนนี้เนื่องจากสามารถรองรับการประมวลผลรูปภาพได้ดีกว่าไมโครคอนโทรลเลอร์รุ่นอื่นๆ

### 4.2 ส่วนขับเคลื่อน

ในส่วนขับเคลื่อนได้เลือกใช้ให้เป็นล้อแบบสองข้าง ซ้ายขวา เพื่อให้มีความง่ายต่อการออกแบบและใช้งาน ใช้การหมุนที่แตกต่างกันของล้อทั้งสองข้างเป็นตัวควบคุมทิศทางของหุ่นยนต์ตามตาราง

การเคลื่อนที่	ล้อข้างซ้าย	ล้อข้างขวา	
เดินหน้า	CCW	CW	CW=ตามเข็มนาฬิกา
ถอยหลัง	CW	CCW	Clockwise
เลี้ยวซ้าย	CW	CW	
เลี้ยวขวา	CCW	CCW	CCW=ทวนเข็มนาฬิกา
หยุด	ไม่หมุน	ไม่หมุน	CounterClockwise

ตารางที่ 4-1 การออกแบบส่วนขับเคลื่อน

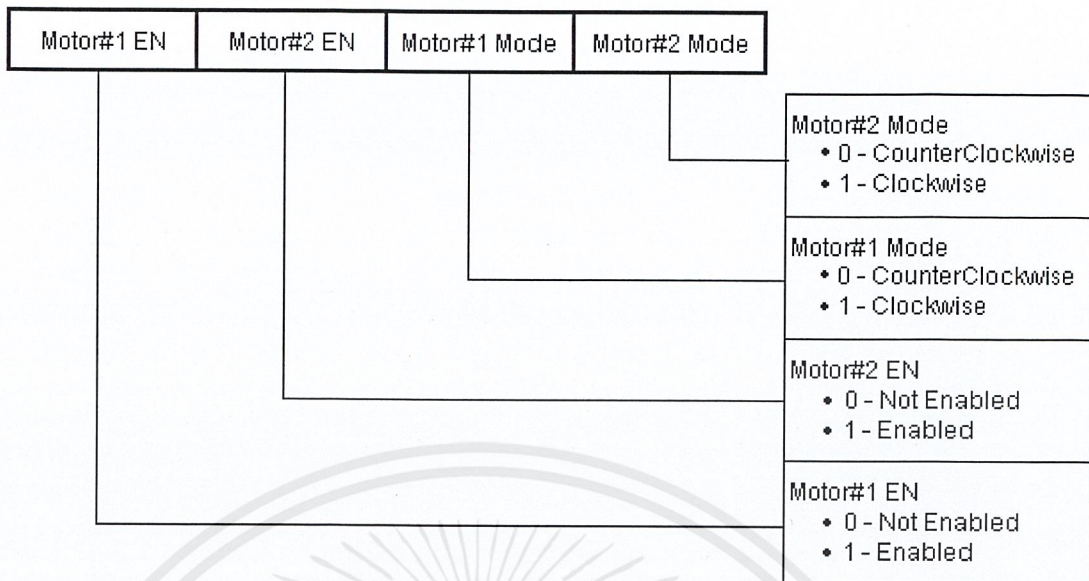
ได้เลือกใช้สเตปเปอร์มอเตอร์มาทำหน้าที่เป็นส่วนขับเคลื่อนนี้ ด้วยเหตุผลที่ว่าสเตปเปอร์มอเตอร์สองตัวที่ควบคุมด้วยวงจรเดียวกันจะหมุนด้วยแรงและความเร็วเท่ากัน ซึ่งต่างจากมอเตอร์กระแสตรงซึ่งจะหมุนด้วยความเร็วและความแรงที่ไม่เท่ากันถึงแม้ว่าจะใช้วงจรควบคุมตัวเดียวกัน

#### 4.2.1 การออกแบบวงจรขับเคลื่อน

วงจรขับเคลื่อนต้องการอินพุตจำนวน 4บิตแบบซิมเพลก (Simplex) และไม่มีสัญญาณป้อนกลับ ซึ่งอินพุตที่ต้องใช้ประกอบด้วย

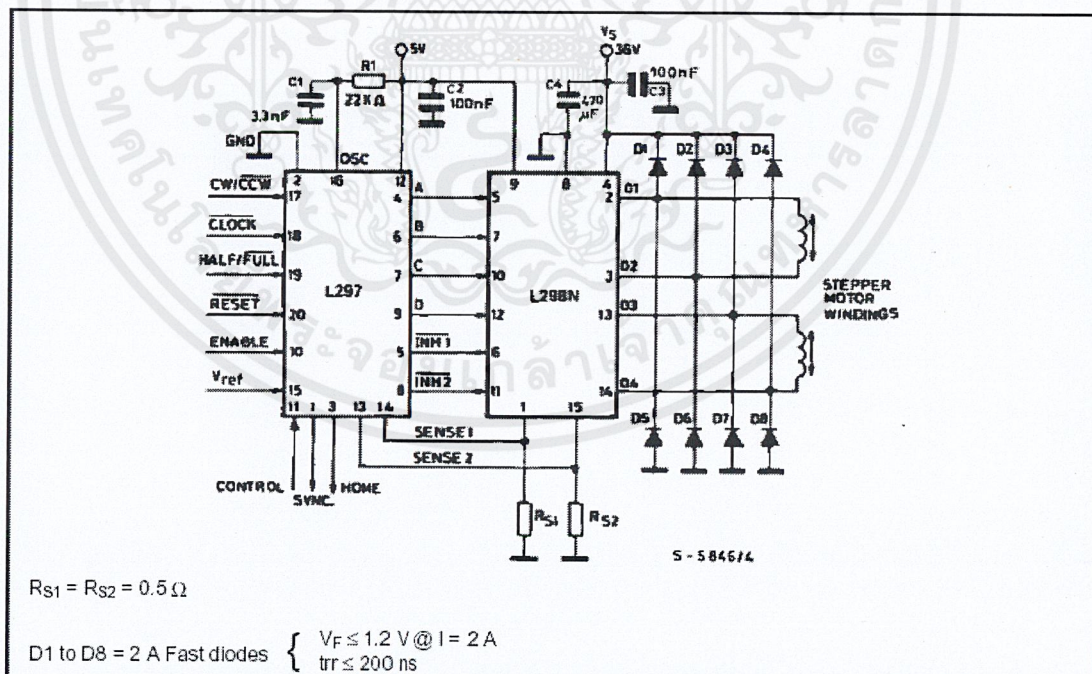
- Motor#1 Enable
- Motor#2 Enable
- Motor#1 Mode (CW/CCW)
- Motor#2 Mode (CW/CCW)

โดยที่มอเตอร์ทั้งสองตัวจะมี Enable และ Mode เพื่อกำหนดว่าจะให้มอเตอร์ตัวนั้นหมุนหรือไม่ และหมุนไปในทิศทางใด ทั้งนี้การที่มอเตอร์แต่ละตัวมี Enable เป็นของตัวเองจะสามารถสั่งให้มอเตอร์หมุนทีละตัวได้เมื่อต้องการเลี้ยวด้วยมุมมองที่ไม่มากนัก มี Control Word ดังนี้



รูปที่ 4-3 Control Word ของมอเตอร์ขับเคลื่อน

ส่วนขับเคลื่อนหรือมอเตอร์ใดเวอร์ที่จะใช้ขับเคลื่อนหุ่นยนต์ขนาดใหญ่ที่มีน้ำหนักมาก จะต้องใช้มอเตอร์ตัวใหญ่ที่ต้องการกระแสมาก ซึ่งวงจรธรรมดาไม่สามารถสร้างกระแสให้มากพอที่จะขับมอเตอร์ได้ จึงต้องใช้ส่วนของการขยายกระแสเข้ามาช่วยให้สามารถสร้างกระแสได้เพียงพอต่อความต้องการของมอเตอร์

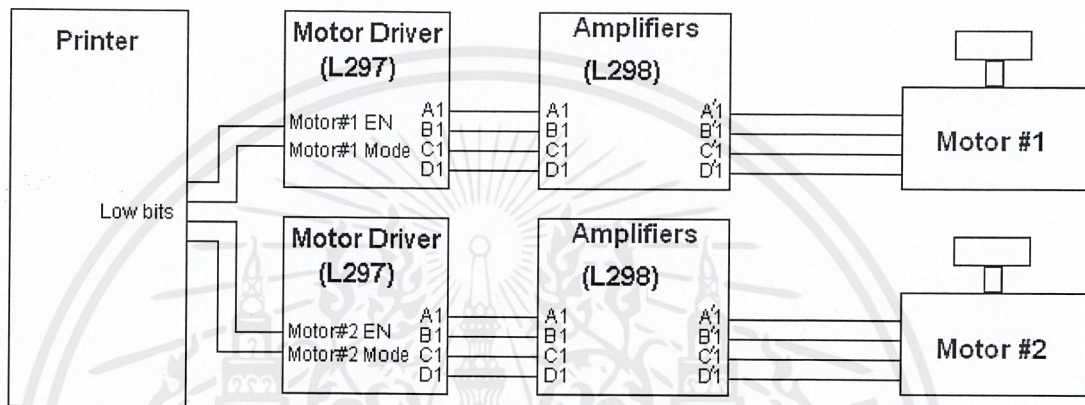


รูปที่ 4-4 วงจรขับมอเตอร์สำหรับมอเตอร์ขนาด 2 แอมแปร์ต่อขั้ว จำนวนหนึ่งตัว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

#### 4.2.2 การอินเทอร์เฟสวงจรขับเคลื่อนเข้ากับพีซีบอร์ด

เมื่อวงจรขับเคลื่อนต้องการเพียง 4 บิต การอินเทอร์เฟสกับพีซีบอร์ดที่ง่ายและสามารถทำได้รวดเร็วกว่าที่สุดจึงเป็นการสื่อสารกับพริ้นเตอร์พอร์ต ซึ่งพริ้นเตอร์พอร์ตมีจำนวนบิตให้ใช้งานได้ถึง 8 บิต ดังนั้นจึงนำบิตต่ำ (Low Bits) ของพริ้นเตอร์พอร์ตมาจำนวน 4 บิตเพื่อใช้อินเทอร์เฟสกับวงจรขับเคลื่อน ทั้งนี้จะใช้แต่ละบิตเลยโดยที่ไม่ต้องเข้ารหัส (Encode) ซึ่งนอกจากจะทำให้วงจรซ้าลงแล้วยังจะยุ่งยากในการเขียนโปรแกรมควบคุมและต่อวงจรถอดรหัสอีกด้วย



รูปที่ 4-5 สถาปัตยกรรมของวงจรขับเคลื่อน

#### 4.2.2 การเขียนโปรแกรมควบคุมส่วนขับเคลื่อน

การควบคุมส่วนขับเคลื่อนสามารถเขียนเป็นฟังก์ชันภาษา C++ ได้ดังนี้

```
#include <iostream.h>
#include <dos.h>
int moving(char[] direction){
    int data;
    switch(direction){
        case "forward": data=11; break;
        case "backward": data=14; break;
        case "left": data=12; break;
        case "right": data=15; break;
        default: data=0;
    }
    outp(0x378,data);
    return 1;
}
```

### 4.3 ส่วนรับข้อมูลหมายเลขห้อง

เมื่อผู้ส่งต้องการส่งเอกสารจะอินพุตหมายเลขห้องปลายทางและห้องต้นทางในส่วนรับข้อมูลนี้ อีกทั้งยังสามารถทำงานเป็น ปุ่มหยุดฉุกเฉินได้อีกด้วย ส่วนรับข้อมูลนี้มี 2 โหมดดังนี้

#### โหมด 1: รับข้อมูลป้ายห้อง

- หมายเลข 0-9
- ปุ่ม Enter
- ปุ่ม Cancel
- ปุ่ม Backspace
- ปุ่ม Mode
- ปุ่ม Emergency Stop

			BACKSPACE
7	8	9	STOP
4	5	6	CANCEL
1	2	3	ENTER
0			

รูปที่ 4-6 ส่วนรับข้อมูล โหมด 1: รับข้อมูลป้ายห้อง

#### โหมด 2: บังคับหุ่นยนต์โดยตรง

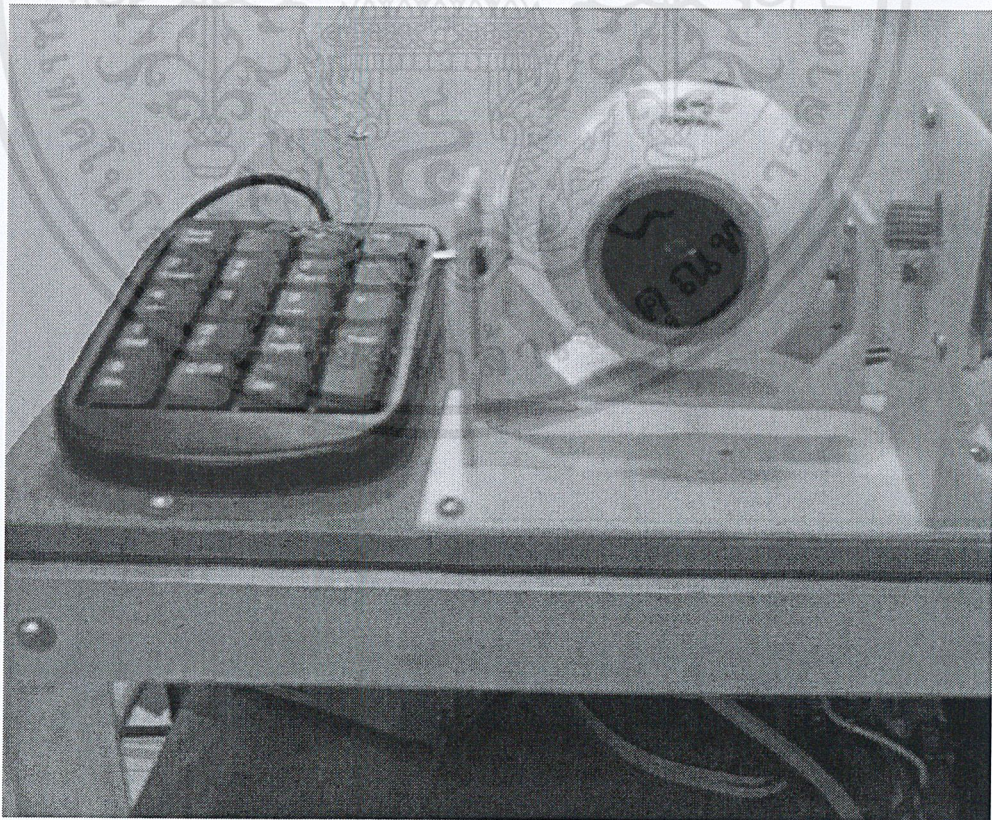
- ปุ่มบังคับล้อ
  - เดินตรง
  - เลี้ยวซ้าย
  - เลี้ยวขวา
  - ถอยหลัง
  - หยุด
- ปรับกล้องเงย
- ปรับกล้องก้ม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

<b>Mode</b>	CAMERA UP	CAMERA DOWN	BACKSPACE
	FORWARD		STOP
LEFT TURN	STOP	RIGHT TURN	CANCEL
	BACKWARD		ENTER

รูปที่ 4-7 ส่วนรับข้อมูลป้ายห้อง โหมด 2: บังคับหุ่นยนต์โดยตรง

เมื่อนำส่วนรับข้อมูลหมายเลขห้องมาติดเข้ากับตัวหุ่นยนต์บนส่วนบนซึ่งเป็นส่วนที่ผู้รับและผู้ส่งสามารถติดต่อได้ง่ายที่สุดในตัวหุ่นยนต์ ดังรูป

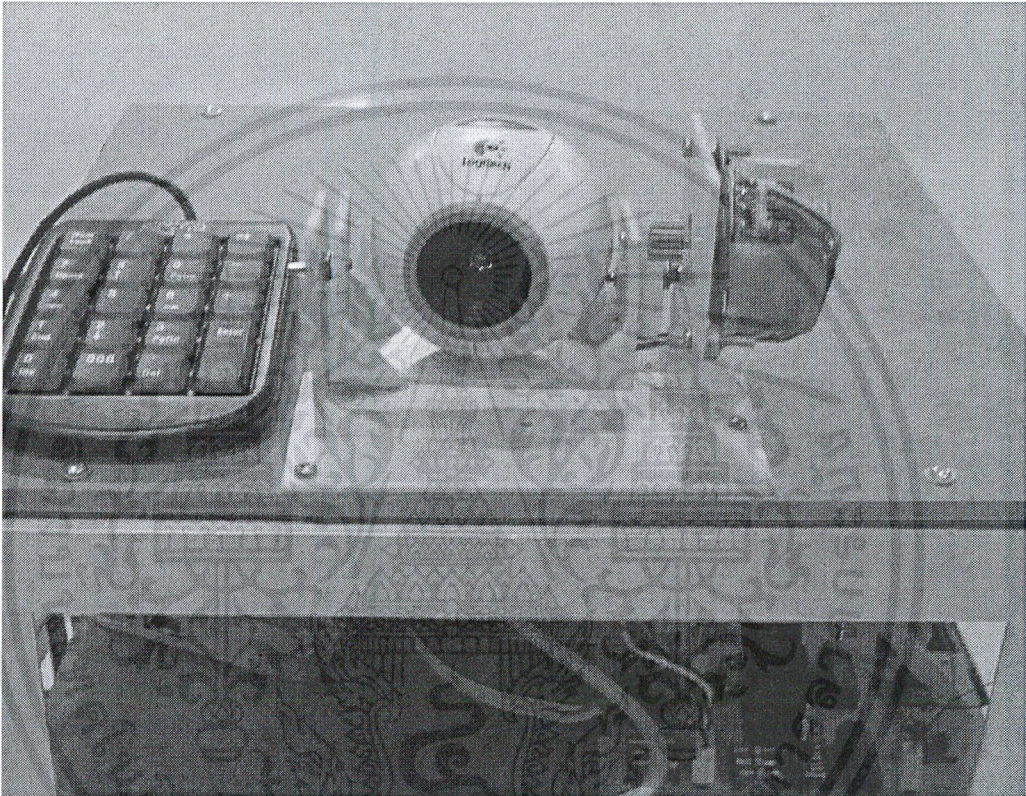


รูปที่ 4-8 ส่วนรับข้อมูลหมายเลขห้องที่ติดอยู่บนตัวหุ่นยนต์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

#### 4.4 กล้อง

กล้องจะทำหน้าที่ถ่ายรูปทางเดินต่างๆ สิ่งกีดขวาง และป้ายห้อง ได้เลือกใช้กล้อง ccd sensor ของ Logitech ซึ่งมีข้อได้เปรียบกว่ากล้องของยี่ห้ออื่นตรงที่มีฟังก์ชัน API สำหรับนักพัฒนาเพื่อควบคุมกล้องได้ด้วย ตัวกล้องนี้จะถ่ายรูปทางเดินและป้ายห้องซึ่งมีความสูงที่ต่างกันมาก จึงต้องวางไว้ส่วนบนสุดของตัวหุ่นยนต์ทำให้สามารถนำไปใช้กับส่วนควบคุมมุมกล้องเพื่อให้ ก้มและเงยได้โดยที่ไม่มีอะไรมาบัง



รูปที่ 4-9 กล้องที่วางไว้บนตำแหน่งที่สามารถมองเห็นได้กว้างที่สุด

#### 4.5 ส่วนควบคุมมุมกล้อง (Camera Angle Control)

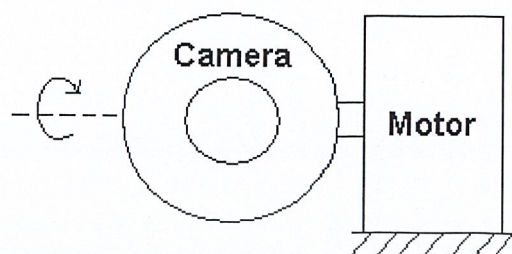
กล้องจะอยู่ในมุมขนานกับพื้นที่มีความสูง 30 เซนติเมตร ซึ่งก็คือความสูงของตัวหุ่นยนต์และจะต้องถ่ายรูปป้ายห้องที่มีความสูงประมาณสองเมตรด้วย ดังนั้นจึงต้องมีการปรับมุมกล้องได้อย่างน้อยหนึ่งทิศทาง คือขึ้นและลง

##### 4.5.1 แนวคิดในการออกแบบ

ส่วนควบคุมมุมกล้อง (Camera Angle Control) นั้นเป็นส่วนสำคัญมากที่สุดส่วนหนึ่ง เมื่อหุ่นยนต์เดินมาจนถึงประตูหน้าห้องแล้วหุ่นยนต์จะหมุนหันหน้าเข้าหาประตู และจะต้องปรับมุมกล้องให้เงยขึ้นเพื่อถ่ายภาพหมายเลขห้องซึ่งอยู่เหนือประตู ส่วนควบคุมมุมกล้องนี้จะทำหน้าที่ปรับมุมของกล้องขึ้นและลงได้อย่างสะดวก และส่วนนี้ต้องการความเที่ยงตรงมากเนื่องจากการเดินส่งเอกสารแต่ละครั้งจะมีการก้มและเงยมุมกล้องหลายครั้ง ซึ่งจะต้องกระทำที่มุมเท่ากันทุกครั้ง จึงได้เลือกใช้สเตปเปอร์มอเตอร์มา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ใช้งานในครั้งนี้ ซึ่งการควบคุมสเตปเปอร์มอเตอร์แบบเฟสเดียว (Half Step) จะเพียงพอแล้วเพราะว่าตัวกล้องมีน้ำหนักไม่มาก

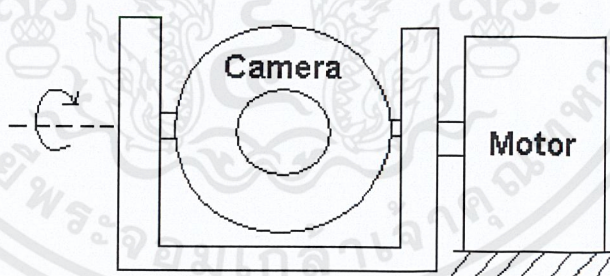


รูปที่ 4-10 การออกแบบส่วนควบคุมมอเตอร์กล้องขั้นต้น

การนำสเตปเปอร์มอเตอร์มาหมุนกล้องที่ต้องการองศาที่แน่นอนในการหมุนแต่ละครั้งนั้น ถือว่าเหมาะสมที่สุดแล้ว

#### 4.5.2 การประกอบส่วนควบคุมมอเตอร์กล้องทางด้านกลศาสตร์

เมื่อพิจารณาถึงความง่ายในการสร้างแล้ว การสร้างส่วนควบคุมมอเตอร์กล้องตามทีออกแบบไว้นั้นจะมีความยุ่งยากคือจำเป็นที่จะต้องเจาะตัวกล้อง เพื่อหลีกเลี่ยงการกระทำใดๆที่จะทำให้ตัวกล้องเสียหายจึงจำเป็นที่จะต้องมีส่วนฐานของกล้องเข้ามาเพิ่ม และให้สเตปเปอร์มอเตอร์หมุนฐานของกล้องแทนที่จะหมุนตัวกล้อง ซึ่งกล้องจะยึดติดอยู่กับฐานและจะหมุนไปพร้อมกับฐานวางกล้อง



รูปที่ 4-11 ส่วนควบคุมมอเตอร์กล้องที่ออกแบบโดยใช้ฐานกล้องเข้ามาช่วย

เมื่อพิจารณาถึงสายของตัวกล้องแล้ว ส่วนควบคุมมอเตอร์นี้สามารถรับองศาให้เงยได้ถึง 80 องศา และปรับก้มได้ถึง 60 องศา ในขณะที่หุ่นยนต์ต้องการปรับมุมกล้องเงยขึ้นมากที่สุดประมาณ 45 องศาเท่านั้น และไม่จำเป็นต้องใช้การปรับมุมก้มเลย การออกแบบส่วนควบคุมมอเตอร์แบบนี้จึงเพียงพอต่อการใช้งานแล้ว

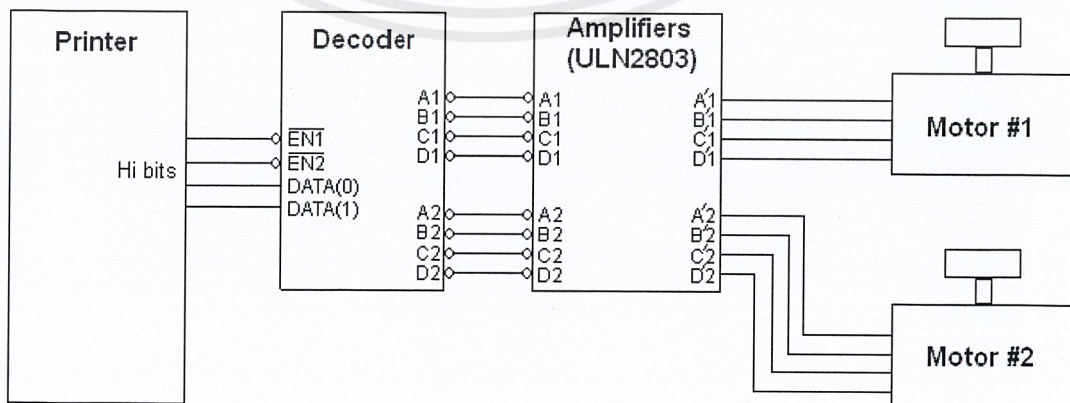


รูปที่ 4-12 ส่วนควบคุมมอเตอร์ที่ประกอบเสร็จเรียบร้อยแล้ว

#### 4.5.3 การออกแบบวงจรขับสเตปเปอร์มอเตอร์

เนื่องจากโหนดของมอเตอร์ (น้ำหนักของกลิ้ง) มีไม่สูงจึงสามารถไต่รหัสสเตปเปอร์มอเตอร์แบบเวฟได้ และจะสามารถออกแบบได้ง่ายกว่าการไต่รหัสแบบพัลสเต็มมาก และเมื่อคำนวณถึงการอินเทอร์เฟสกับตัวหุ่นยนต์แล้วจะใช้บิตสูง 4 บิตของพริ้นเตอร์พอร์ตที่ยังไม่ได้นำไปใช้งานใดๆ

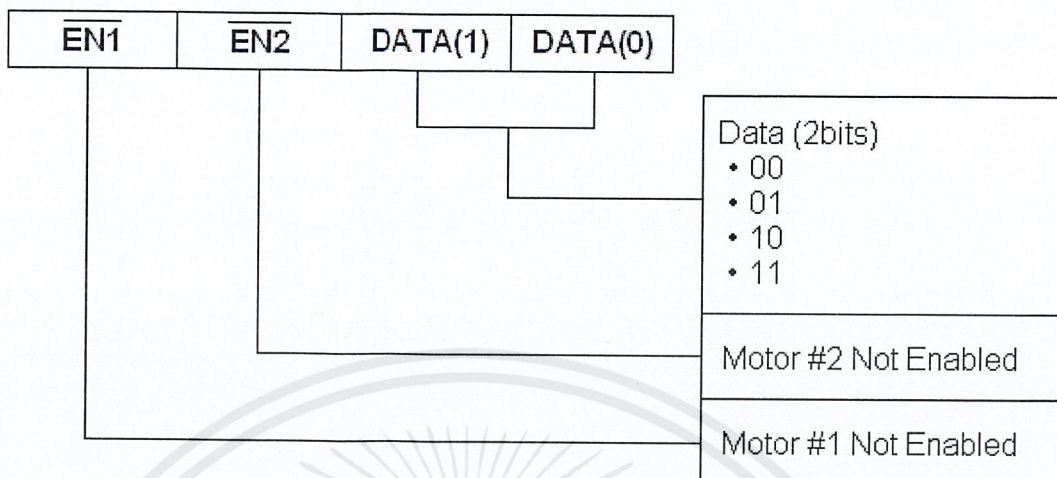
เมื่อพิจารณาถึงจำนวนบิตที่มีแล้ว สามารถออกแบบวงจรโดยมีมอเตอร์มากที่สุดได้ถึง 2 ตัวเพื่อรองรับการหมุนของมอเตอร์ได้ถึงสองมิติ จึงต้องนำวงจรถอดรหัส (Decoder) เข้ามาช่วย และจำเป็นจะต้องมีส่วนขยายกระแสด้วย เพราะกระแสที่ออกจากพริ้นเตอร์พอร์ตจะมีค่าต่ำมาก ไม่เพียงพอที่จะนำไปใช้ไต่รหัสสเตปเปอร์มอเตอร์ นอกจากนี้แล้วส่วนขยายกระแสมักทำหน้าที่ป้องกันไม่ให้ไฟไหลย้อนกลับไปที่สายพริ้นเตอร์พอร์ตอีกด้วย



รูปที่ 4-13 คาดำพารของวงจรไต่รหัสสเตปเปอร์มอเตอร์

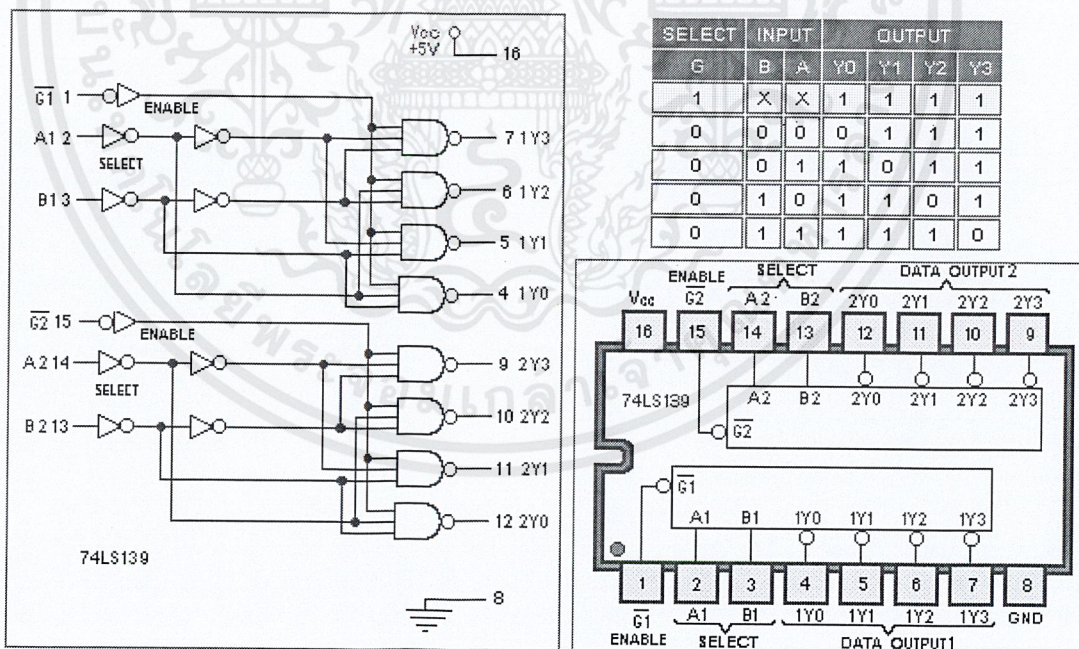
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โดยที่ไอซีถอดรหัสจะมี Control word ดังนี้



รูปที่ 4-14 Control Word ของส่วนควบคุมมอเตอร์

ซึ่งการควบคุมสเตปเปอร์แบบไบโพลาร์จำเป็นต้องใช้ขาสัญญาณถึง 4 เส้น จึงได้นำเอาไอซีถอดรหัสสัญญาณ (2to4 Decoder) เบอร์ 74LS139 มาใช้งานร่วมกันในวงจร ซึ่งการทำงานของวงจรถอดรหัสสัญญาณแสดงดังตารางข้างล่าง

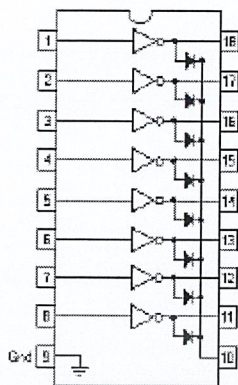


รูปที่ 4-15 วงจรภายในของไอซี 74LS139

ในส่วนของวงจรขยายสัญญาณจะนำไอซี ULN2803 เข้ามาใช้งาน ซึ่งไอซีตัวนี้สามารถสร้างกระแสได้ถึง 1 Ampere ต่อขาเอาท์พุทหนึ่งขา ซึ่งเพียงพอต่อการใช้ขับสเตปเปอร์มอเตอร์แบบเวฟ

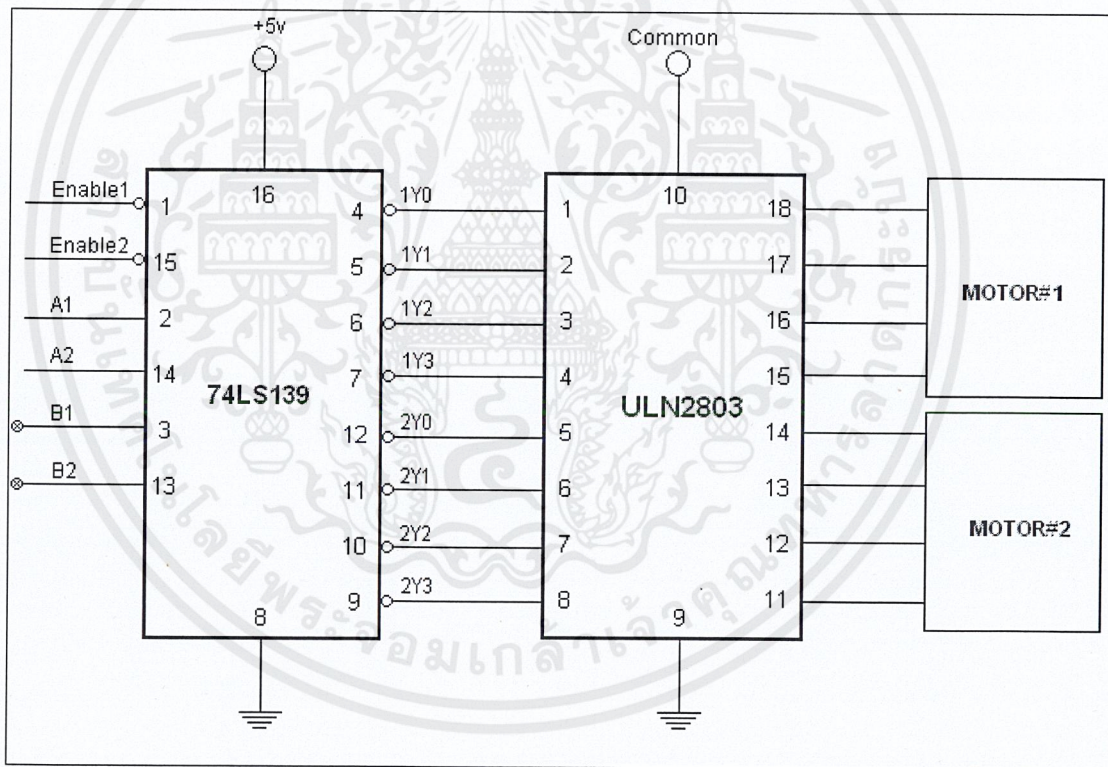
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ULN 2803 (AP)



รูปที่ 4-16 วงจรภายในของไอซี ULN2803

จากไอซีสองตัวที่กล่าวมา เมื่อนำมาต่อกันจะได้วงจรขับสเตปเปอร์มอเตอร์สองตัวดังนี้



รูปที่ 4-17 วงจรขับมอเตอร์หนึ่งเฟส จำนวนสองตัว

#### 4.5.4 การเขียนโปรแกรมควบคุมส่วนควบคุมมุกกล้อง

ในการเขียนโปรแกรมควบคุมสเตปเปอร์มอเตอร์ไครเวอร์จำนวน 1 ตัวที่ต้องผ่านวงจรถอดรหัสด้วยนั้นสามารถเขียนเป็นฟังก์ชันได้ด้วยภาษา C++ ดังนี้

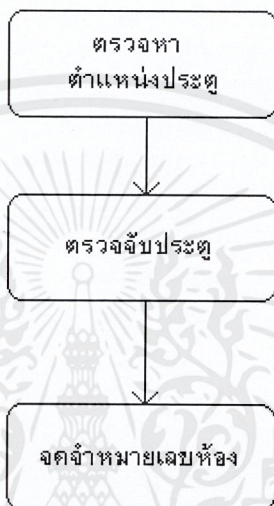
```
#include <dos.h>
#include <iostream.h>
#define data={"00","01","10","11"};

int camera_angle(int direction, int step){
    int outdata;
    for(int i=0; i<step; i++){
        if( direction=="up"){
            data++;           //step up
        }else{
            data--;           //step down
        }
        outdata="1".data;
        outp(0x378,outdata); //move
    }
    outp(0x378,0x00);       //stop
    return 1;
}
```

## บทที่ 5

### การออกแบบซอฟต์แวร์

ส่วนประกอบซอฟต์แวร์เป็นส่วนที่นำไปใช้ร่วมกับฮาร์ดแวร์เพื่อให้หุ่นยนต์ทำงานได้อย่างถูกต้อง โดยส่วนซอฟต์แวร์นี้จะประกอบไปด้วยการประมวลผลรูปภาพและการรู้จำตัวอักษร



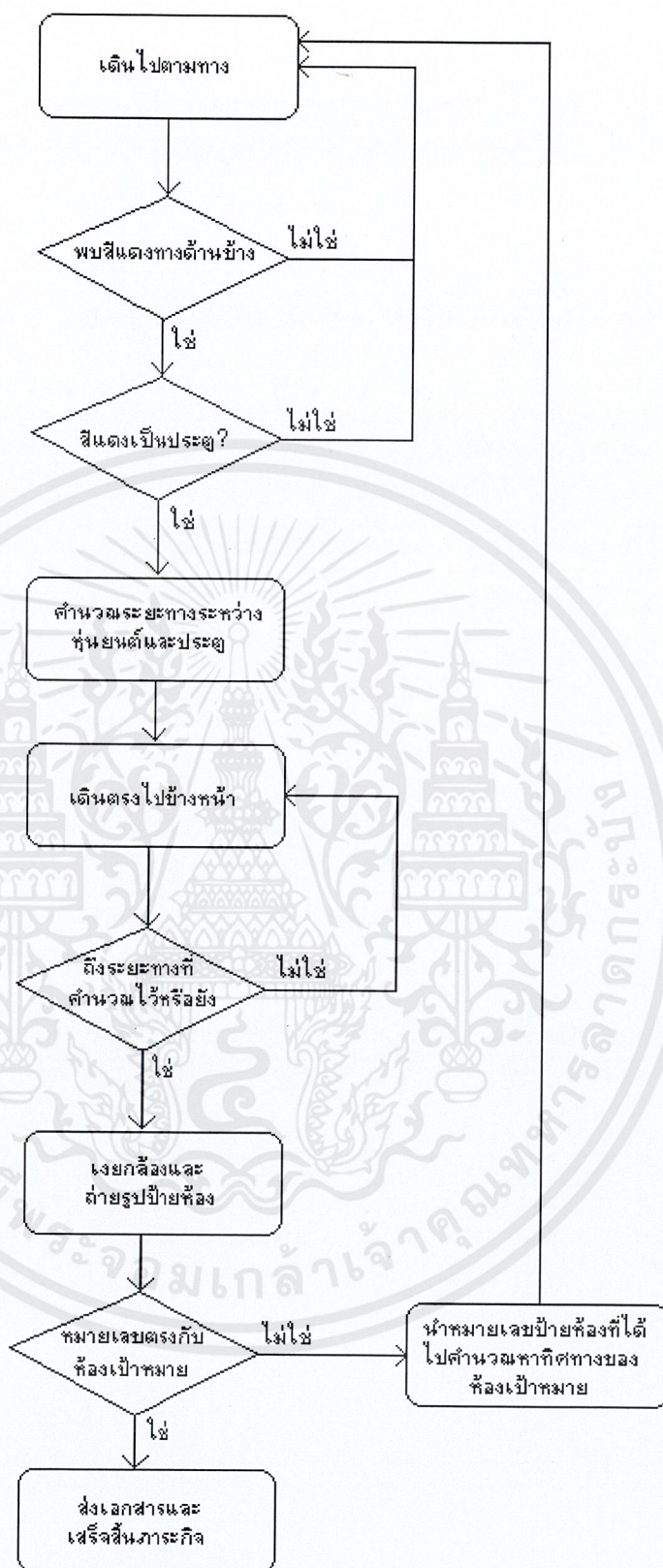
รูปที่ 5-1 ส่วนประกอบต่างๆของซอฟต์แวร์

รูปที่ 5-1 เป็นรูปของส่วนประกอบต่างๆของซอฟต์แวร์ ซึ่งประกอบไปด้วย 3 ส่วนด้วยกัน คือ

- การตรวจหาตำแหน่งของประตู
- การตรวจสอบว่าเป็นประตู
- การจดจำหมายเลขห้องจากภาพถ่าย

#### 5.1 การตรวจหาประตูโดยใช้กล้อง

การตรวจจับประตูซึ่งมีสีที่แตกต่างไปจากสิ่งแวดล้อมอื่นๆคือ ประตูมีลักษณะเป็นแถบใหญ่สองแถบสีแดงส้ม ซึ่งจะปรากฏขึ้นในมุมซ้ายหรือขวามือในขณะที่เดินไปตามทาง เมื่อเจอประตูแล้วหุ่นยนต์จะคำนวณระยะทางที่จะเดินไปจนถึงประตูและจะหยุดที่หน้าประตูแล้วคอยหันไปตรวจว่าอยู่หน้าประตูแล้วตามที่ได้คำนวณมา โดยอัลกอริทึมการเดินหาประตูโดยใช้กล้องจับมีดังนี้

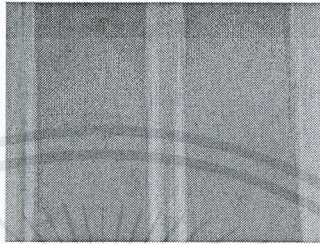


รูปที่ 5-2 โฟลว์ชาร์ตอัลกอริทึมการเดิน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## 5.2 การตรวจจับประตู

การตรวจสอบประตูซึ่งมีสีออกส้มแดง โดยสีแดงดังกล่าวจะเป็นสีแดงที่เด่นชัดดังนั้นสามารถใช้ค่า R ใน RGB ตรวจสอบได้โดยตรงเลย ใช้เทรซโซลตรวจสอบค่า R ที่เด่นขึ้นมามากกว่าค่าสีอื่นมากๆ ซึ่งเทรซโซลที่ใช้มีค่าเท่ากับ  $R > 0.4$  สามารถตรวจสอบได้ว่าตำแหน่งที่ยืนอยู่เป็นประตูรีเปลา เพื่อที่จะได้ตัดสินใจในการยกกล้องขึ้นไปตรวจจับหมายเลขห้องได้



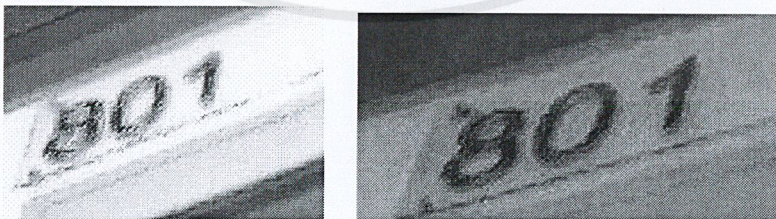
รูปที่ 5-3 ภาพถ่ายประตู ในโหมดสี RGB



รูปที่ 5-4 ระดับความเข้มสีของ R G และ B เรียงตามลำดับ

## 5.3 การตรวจสอบหมายเลขห้องจากภาพ

การตรวจหาหมายเลขห้องนี้จะใช้กล้องเป็นตัวจับ ผนวกกับการใช้แว่นขยาย เพื่อให้เห็นหมายเลขห้องอย่างชัดเจนขึ้น จากรูปที่ 5-5 จะพบว่า เมื่อใช้แว่นขยายมาช่วยในการถ่ายภาพจะได้สีของหมายเลขห้องอย่างชัดเจนกว่าดังนั้นในการถ่ายภาพหมายเลขห้องจึงได้นำแว่นขยายมาช่วย



รูปที่ 5-5 ภาพที่ตัดเฉพาะส่วนหมายเลขห้องของภาพถ่ายที่ได้จากการใช้ และไม่ใช้แว่นขยายตามลำดับ

ขั้นตอนในการตรวจหมายเลขห้องหลังจากได้ภาพมาแล้วมีดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 5-6 ขั้นตอนของการตรวจหาเลขห้องโดยใช้กล้อง

### 5.3.1 การแยกตัวเลขออกจากส่วนอื่นๆของภาพ

ความแตกต่างระหว่างตัวเลขและบริเวณส่วนอื่นคือ โทนสี ซึ่งตัวเลขจะมีโทนสีออกสีน้ำเงิน ในขณะที่ส่วนอื่นจะไม่มีส่วนที่เป็นสีน้ำเงิน จึงนำลักษณะแตกต่างที่เห็นเด่นชัดตรงนี้มาใช้เป็นกฎเกณฑ์ในการแยกตัวเลขออกจากส่วนอื่นๆ โดยเนื่องจากภาพถ่ายเริ่มแรกจะอยู่ในรูปแบบดังแสดงในรูปที่ 5-7 แต่ค่า R G และ B ในแต่ละพิกเซลไม่ได้บอกความเป็นสีใดสีหนึ่งออกมา แต่จะเป็นการรวมเอาความเข้มแสงด้วย ดังนั้นจะเห็นว่าที่จุดหนึ่งมีค่า  $RGB=\{10,20,30\}$  จะมีความเป็นสีน้ำเงินมากกว่าจุดที่มีค่า  $RGB=\{100,100,100\}$  ทั่วๆไปที่จุดที่สองมีค่า B มากกว่าในจุดแรก เราจึงต้องพยายามที่จะแยกเอาส่วนที่เป็นความเข้มแสงออกไป ดังนั้นเพื่อที่จะแยกตัวเลขจากพื้นหลัง เราได้ทำการทดลอง 2 วิธีด้วยกัน



รูปที่ 5-7 ภาพถ่ายป้ายห้องในโหมด RGB

#### 5.3.1.1 ความพยายามในการแยกสีโดยใช้ความแตกต่างของค่า Y,Cb,Cr

เนื่องจากค่า YCbCr สามารถแยกองค์ประกอบสีออกมาได้โดยไม่เกี่ยวกับความเข้มแสง เนื่องจากค่าความเข้มแสงได้นำไปรวมไว้ที่ค่า Y แล้วนำส่วนประกอบสีน้ำเงินกับแดงไปใส่ไว้ที่Cb กับ Cr โดยไม่รวมความเข้มแสง รูปที่ 5-8 ถึง 5-10 เป็นภาพที่แยกส่วนของ Y Cb และ Cr ออกจากกันตามลำดับ จะเห็นว่าในรูปแบบของ Cb จะมองเห็นความแตกต่างได้ชัดเจนที่สุด จะพบว่าในส่วนของตัวเลขจะมีค่าที่ออกมาจะสูงกว่าในจุดอื่นๆ แต่ก็ยังมีจุดอื่นๆอีกหลายจุดที่มีค่าใกล้เคียงกับค่า Cb ใน ส่วนของตัวเลข เช่น บริเวณที่เป็นกระดาษติดตั้งในภาพ ดังนั้นถ้านำค่า Cb ที่ได้ มาทำเทรชโฮลจะได้สิ่งที่ไม่ต้องการติดมาด้วย เป็นจำนวนมาก



รูปที่ 5-8 ภาพถ่ายป้ายห้องในรูปแบบ YCbCr แยกเฉพาะ Y



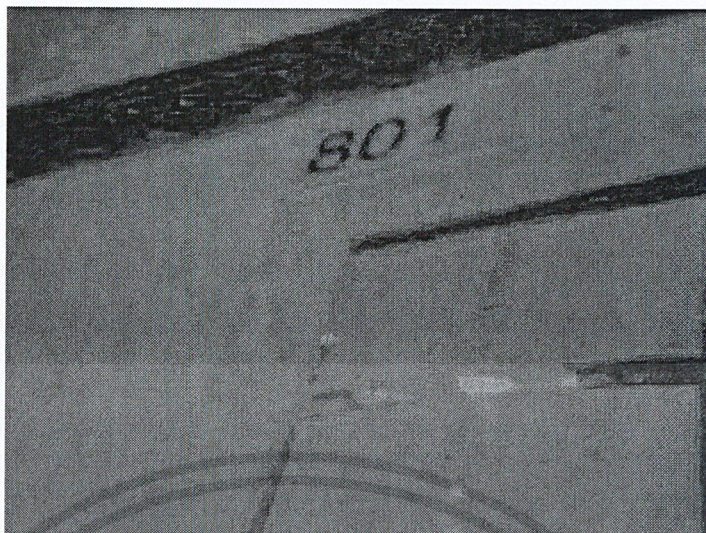
รูปที่ 5-9 ภาพถ่ายป้ายห้องในรูปแบบ YCbCr แยกเฉพาะ Cb



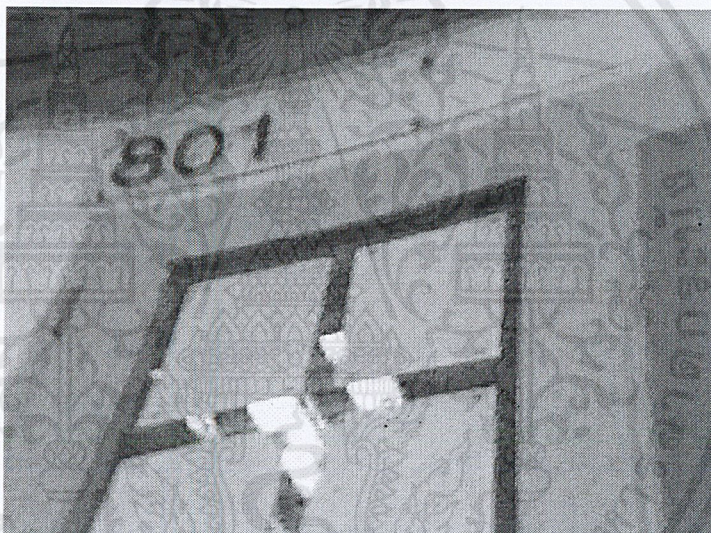
รูปที่ 5-10 ภาพถ่ายป้ายห้องในรูปแบบ YCbCr แยกเฉพาะ Cr

### 5.3.1.2 ความพยายามแยกสี โดยใช้ ความแตกต่างของ ค่าอนุกรมไลซ์ของค่า RGB

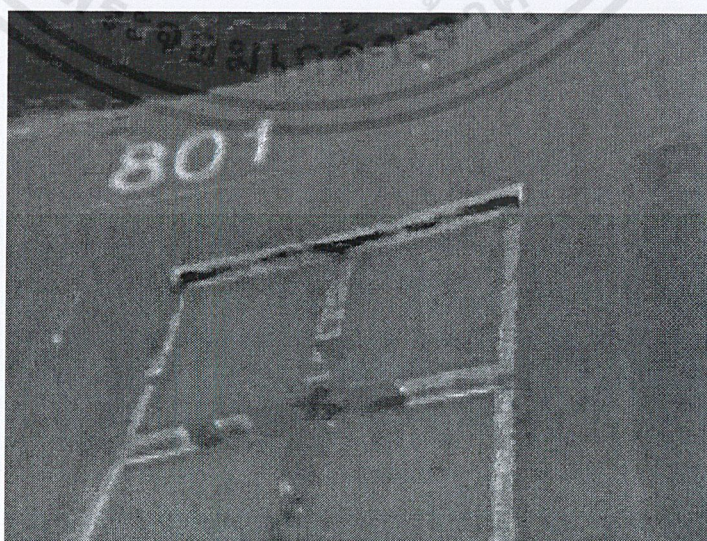
จากรูปที่ 5-11 ถึง 5-13 เป็นภาพที่ได้จากการหาค่าอนุกรมไลซ์ของ RGB ตามทฤษฎีข้อที่ 3.5 จะให้ผลดีกว่าการใช้ค่า Cb มีหลายตำแหน่งที่ให้ค่าใกล้เคียงกับตัวเลขน้อยลง และสามารถนำมาใช้ได้จริงได้ ดังนั้นจึงเลือกใช้วิธีการนี้ในการแยกหมายเลขห้อง



รูปที่ 5-11 ภาพค่า  $r$  ที่ปรับให้อยู่ในรูปภาพเป็นสี่เทา



รูปที่ 5-12 ภาพค่า  $g$  ที่ปรับให้อยู่ในรูปภาพเป็นสี่เทา



รูปที่ 5-13 ภาพค่า  $b$  ที่ปรับให้อยู่ในรูปภาพเป็นสี่เทา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

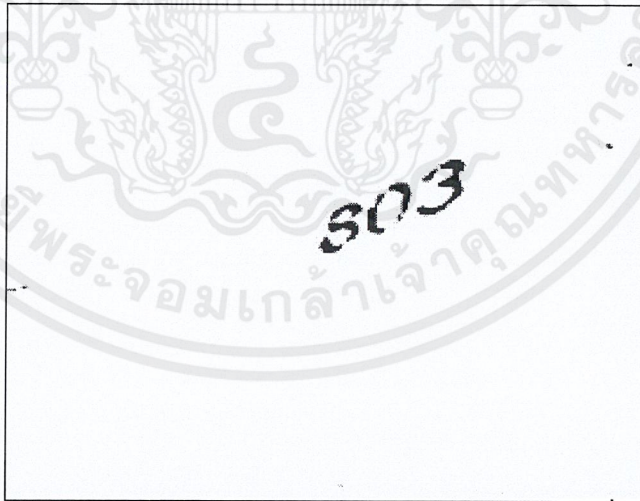
### 5.3.2 การทำเทรชโฮลให้ภาพ (Image Thresholding)

การทำเทรชโฮลเป็นการทำเพื่อแยกภาพที่เป็นสีเทาให้อยู่ในรูปแบบสี ขาว-ดำ เพื่อให้ง่ายแก่การนำไปใช้ต่อ รวมทั้งยังใช้เนื้อที่ในการเก็บข้อมูลน้อยลงด้วย

- เมื่อลองทำเทรชโฮลให้ภาพโดยให้  $b \geq 0.4$  และ  $b$  มากกว่า  $r$  และ  $g$  ให้ผ่านเป็นสีดำ ได้ผลดังรูปที่ 5-14 และ 5-15



รูปที่ 5-14 ภาพที่ทำเทรชโฮลแล้วมีส่วนที่ไม่ต้องการติดมาด้วย

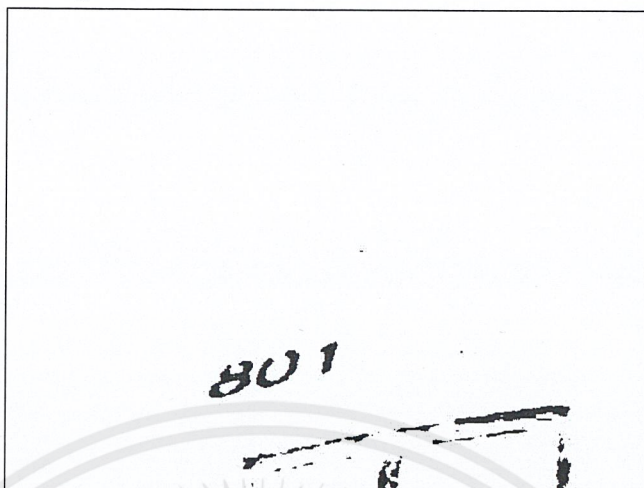


รูปที่ 5-15 ภาพที่ทำเทรชโฮลแล้วมีเฉพาะส่วนตัวเลขติดมาเท่านั้น

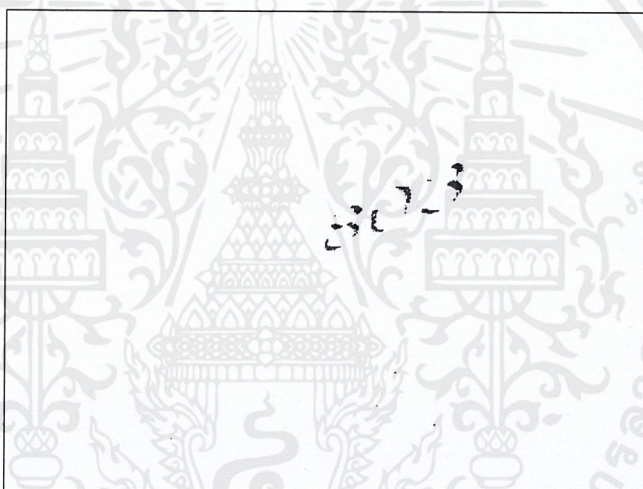
จากรูปที่ 5-16 จะพบว่าในภาพจะได้จุดที่ไม่เกี่ยวข้องเข้ามาด้วยน้อย แต่ในขณะเดียวกันรูปที่ 5-16 มีส่วนอื่นๆที่ไม่ต้องการติดมาด้วยมาก และนำไปทำการกำจัดสิ่งรบกวนต่างๆ เพื่อหาตำแหน่งทำไดยาก จึงได้ลองใช้ ค่าเทรชโฮลอื่นๆ มาเป็นตัวช่วยตัด

- ใช้ค่าเทรชโฮลที่มากจำกัดมากขึ้น โดยให้ในตำแหน่ง ที่มีค่า  $b$  มากกว่าค่าอื่นมากๆ เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่นอนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ได้ผลดังรูป



รูปที่ 5-16 ภาพที่ทำเทรซโฮลแล้วมีเฉพาะส่วนตัวเลขติดมาเท่านั้น



รูปที่ 5-17 ภาพที่ทำเทรซโฮลแล้วมีเฉพาะส่วนตัวเลขติดมาเท่านั้น

จะเห็นได้จากรูปที่ 5-16 สิ่งรบกวนน้อยลงจากรูปที่ 5-14 ถึง 5-15 ตามลำดับ ซึ่งเป็นภาพเดียวกันแต่ตัวเลขก็จะติดเข้ามาน้อยลงด้วย แต่ในขณะที่เดียวกันก็มีผลทำให้บางภาพมีตัวเลขติดเข้ามาน้อยเกินไปอาจทำให้การ นำไปตรวจสอบหมายเลขห้องในขั้นตอนต่อไปทำได้ยาก แต่ข้อดีของการใช้ค่าเทรซโฮลวิธีนี้คือ สามารถนำไปกำจัดสิ่งรบกวน และหาตำแหน่งของตัวเลขได้ง่ายขึ้น

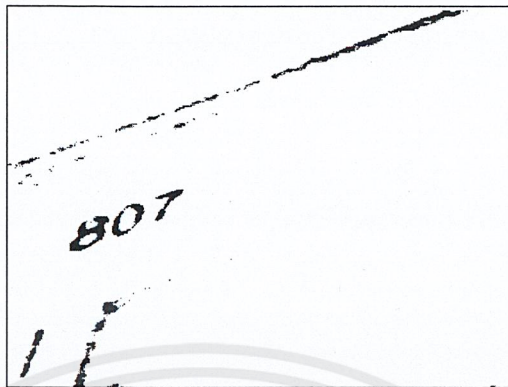
### 5.3.3 การกำจัดสิ่งรบกวน

วิธีต่างๆที่ได้นำมาทดลองใช้ มีอยู่ 4 วิธีคือ

#### 5.3.3.1 สแกนตามแนวแกน X และแกน Y

ได้ทดลองโดยการใช้การไล่ตรวจตามแนวแกน X และแกน Y เพื่อเลือกไว้เฉพาะจุดที่มีสีดำติดกันหลายๆจุด แต่ก็จะมีปัญหาในกรณีที่ตัวเลขขาดตอนทำให้ตัวเลขบางส่วนขาดหายไป รวมทั้งภาพที่มีสิ่งรบกวนหนักก็ไม่อาจตัดออกได้ และมีโอกาสทำให้ตัวเลขขาดหายไปมีสูงกว่า

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



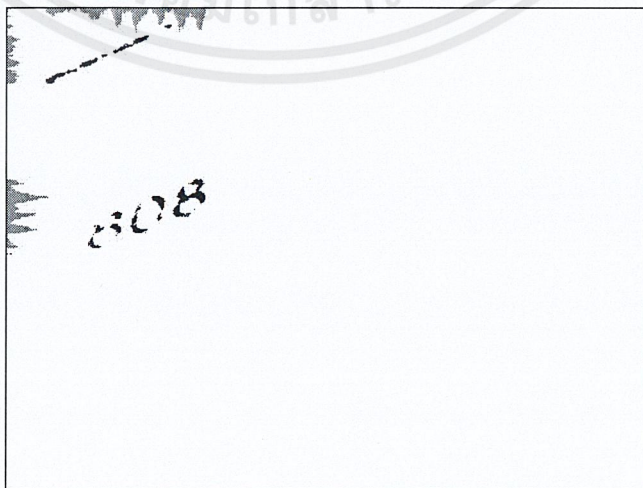
รูปที่ 5-18 ตัวอย่างภาพที่ได้หลังจากตัดเทรซโฮล



รูปที่ 5-19 ภาพหลังจากสแกนตามแกน X และ Y ตามลำดับ

5.3.3.2 แยกสิ่งรบกวนโดยใช้ผลรวมของพิกเซลตามแนวแกน X และ Y

โดยได้ลองทำการรวบรวมค่าผลรวมในแต่ละแถวตามแนวแกน X และแกน Y เพื่อหาจุดตัด จากการหาค่าเทรซโฮลของรูปที่ 5-19 พบว่า ยังไม่แยกตัวเลขออกจาก สิ่งรบกวนได้หมด โดยเฉพาะในกรณีที่มีสิ่งรบกวนเป็นจำนวนมาก



รูปที่ 5-20 การหาผลรวมตามแนวแกน X และ Y โดยผลรวมได้แสดงเป็นกราฟไว้ที่ขอบของแต่ละแถว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 5.3.3.3 การนำคอร์ไลท์เมทซึ่งมาใช้

โดยใช้สมการตามทฤษฎี 3.10 โดยใช้ภาพต้นแบบโดยให้ภาพต้นใช้



รูปที่ 5-21 ภาพ ต้นแบบที่นำไปเปรียบเทียบ

เป็นภาพของเลขชั้น ซึ่งจะเป็ตัวเลขเดียวกันในทุกๆห้องภายในชั้นเดียวกันมาใช้ โดยได้ลองกับภาพที่ผ่านการทำเทรซโฮลมาทั้งสองแบบคือ

- แบบเทรซโฮลต่ำ

จะมีสิ่งรบกวนติดมาด้วยมาก แต่ตัวเลขจะเข้มและหนา

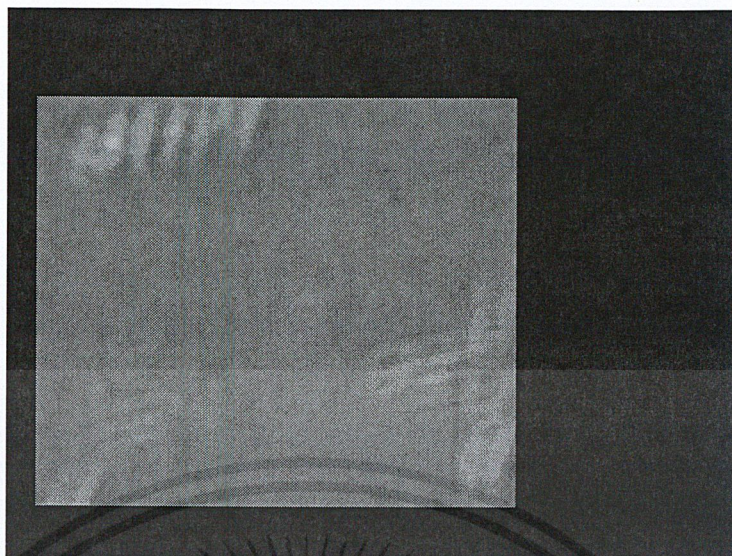
- แบบเทรซโฮลสูง

จะมีสิ่งรบกวนติดมาด้วยน้อย แต่ตัวเลขจะบางและอาจจะขาดได้แบบตามที่ได้กล่าว

ไปในหัวข้อที่ 5.3.2 ได้ผลดังนี้

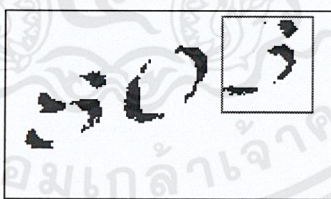
**8**

รูปที่ 5-22 ภาพ ต้นแบบที่นำไปเปรียบเทียบ



รูปที่ 5-23 ผลจากการแมทซิงแบบเทียบพิกเซลของ ภาพที่แปลงเป็นระดับสีเทา

รูปที่ 5-23 เป็นภาพที่ได้ลองนำภาพต้นแบบดังในรูปที่ 5-22 ไปวิ่งทั้งภาพได้ผลพบว่าได้ที่ตำแหน่งคานบนเลข 8 ให้ค่าสูงสุด แต่เนื่องจากมุมของตัวเลขต่างไปไม่เหมือนเดิมรวมทั้งขนาดที่ไม่ทำกัน ทำให้ในบางครั้ง จะพบว่าตำแหน่งที่ได้จากการหาตำแหน่งโดยวิธีนี้จะได้ตำแหน่งในบริเวณตัวเลขแต่อาจไม่ใช่ตำแหน่งของเลขขึ้นซึ่งเป็นเลขตัวแรก โดยผลอาจจะได้ค่าเป็นเลขอื่นๆ เช่น เลข 0 และที่เห็นอย่างเด่นชัดคือในช่องที่มี ตัวเลขในตำแหน่งอื่น เป็นเลขตัวเดียวกับหมายเลขขึ้นเช่น ช่อง 808 ทำให้ในการเทียบกับภาพต้นแบบแล้วทำให้ตัวเลขในตำแหน่งอื่นๆ ให้ค่าสูงกว่า ดังนั้นการนำภาพต้นแบบไปวิ่งนี้ทำให้จะได้เพียงตำแหน่งโดยคร่าวๆเท่านั้น จึงใช้วิธีลากกรอบครอบอย่างกว้างๆ กินบริเวณไว้ก่อน รูปที่ 5-11 เป็นภาพที่ตำแหน่งอื่นของตัวเลขได้ค่าอริเลทสูงสุด



รูปที่ 5-24 ภาพที่ได้ตำแหน่งเลขอื่น ใกล้เคียงกว่า เลขขึ้น

การทดลองทำแมทซิงได้ลองกับทั้งภาพที่ได้จากการทำเทรซโฮล 2 แบบ และได้ลองใช้วิธีแมทซิงในภาพทั้งสองแบบด้วยกัน ได้ผลดังนี้

เทรซโฮล	ความถูกต้อง
เทรซโฮลต่ำ	75%
เทรซโฮลสูง	99%

ตารางที่ 5-1 เปรอ์เซนต์ความถูกต้องของการทำเทรซโฮลด้วยวิธีการและตัวแปรที่ต่างกัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ในการนำเทรซโซลมาใช้ตัดภาพจึงต้องใช้ทั้งสองแบบผสมผสานกันไป โดยหาตำแหน่งของตัวเลขอย่างคร่าวๆจากภาพเทรซโซลสูง และนำตำแหน่งที่ได้ไปตัดตัวเลขออกมาจากภาพที่มีเทรซโซลต่ำ

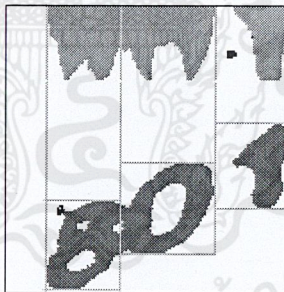
#### 5.3.3.4 การทำคอร์ริเลตซึ่งแบบเลือกทำ

การคอร์ริเลตซึ่งธรรมดาจะตรวจรูปแบบในทุกพิกเซลของรูป ซึ่งจะทำให้เสียเวลามากตรงที่บางส่วนของรูปไม่มีตัวข้อมูลตัวเลขหรือรบกวนเลย จึงต้องมีการตรวจสอบว่าตำแหน่งนั้นมีข้อมูลตัวเลขหรือตัวรบกวนก่อนหรือเปล่าโดย จะทำคอร์ริเลตซึ่งเฉพาะที่ตำแหน่งของภาพที่ซ้อนทับกับตรงกลางของหน้าต่างของภาพต้นแบบมีสี่ตำแหน่ง

#### 5.3.3 การแบ่งหลักของตัวเลขห้อง

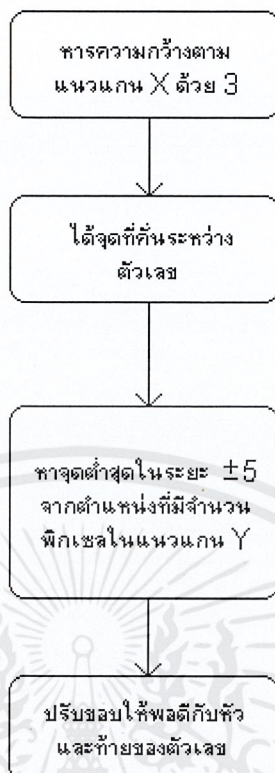
หลังจากแยกตัวเลขจากสิ่งอื่นๆ ได้หมดแล้ว ขั้นตอนต่อมาคือการแยกเลขห้องซึ่งเป็นเลขสามหลัก ออกการจากกัน อัลกอริทึมการแบ่งหลักของตัวเลขห้องมีดังนี้

- ใช้ช่องว่าง เนื่องจากในระหว่างตัวเลขแต่ละตัวจะแยกออกจากกันได้โดยใช้ระยะห่างของตัวเลข แต่จะเกิดปัญหาในกรณี ที่เป็นในลักษณะอย่างรูปที่ 5-11 จะพบว่า ตรงไม่มีระหว่าง 8 และ 0 ทำให้อาจตัดเลขผิดตำแหน่งได้
- ใช้ขนาดโดยประมาณ



รูปที่ 5-25 ภาพการตัดตำแหน่งโดยใช้วิธีการประมาณ

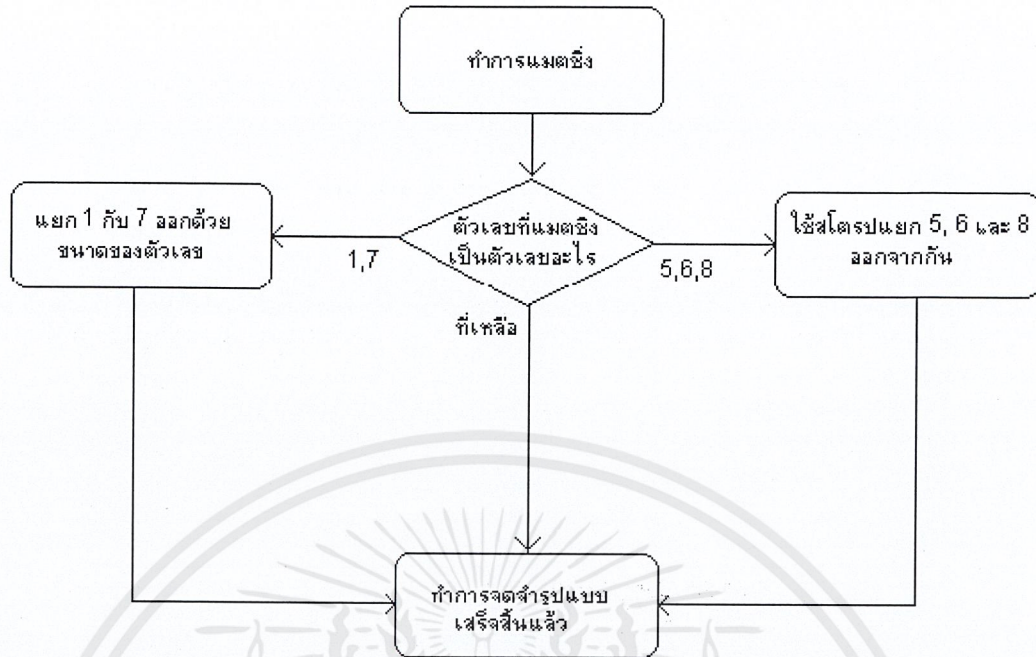
การหาขนาดโดยประมาณเป็นวิธีการใช้คุณลักษณะ ที่ขนาดแต่ละตัวของตัวเลขจะมีขนาดใกล้เคียงกัน แต่เนื่องจากตัวเลขมีมุมเอียงแตกต่างกัน ทำให้อาจผิดพลาดได้ ดังนั้นจึงใช้จำนวนพิกเซลในแนวแกน Y มาช่วยในการพิจารณาดำแหน่งที่ควรจะเป็น ดังนั้นในการแบ่งหมายเลขห้องจึงใช้วิธีการหาขนาดโดยประมาณมาใช้โดยมีขั้นตอนตามรูปที่ 5-26



รูปที่ 5-26 ขั้นตอนการหาขนาดโดยประมาณ

#### 5.3.4 การจดจำหมายเลขห้อง

หลังจากที่ได้ตำแหน่งแล้วและแบ่งตัวเลขแต่ละตัวออกจากกันแล้วก็นำตัวเลขแต่ละตัวมาทำการแยกออกจากกัน โดยมีขั้นตอนต่อไปนี้



รูปที่ 5-27 โพลีชาร์ตขั้นตอนการจดจำตัวเลข

### 5.3.3.1 การจดจำรูปแบบโดยการแมตซิ่ง

ขั้นแรกก็นำตัวเลขเหล่านั้นมาทำการแมตซิ่ง แต่การแมตซิ่งคราวนี้จะแตกต่างจากการแมตซิ่งเพื่อกำจัดสิ่งรบกวนออก เนื่องจากครั้งนี้เราทราบตำแหน่งตัวเลขเรียบร้อยแล้ว ดังนั้นเพื่อความถูกต้องมากขึ้นก็โดยทำการ ปรับขนาดตัวเลขให้มีขนาดเท่ากับภาพต้นแบบ โดยจะนำภาพต้นแบบของตัวเลข ทุกตัวตั้งแต่ 0-9 ซึ่งภาพต้นแบบทั้ง 10 ตัวจะมีขนาดเท่ากันหมด ผลของการนำไปแมตซิ่งได้ผลดังต่อไปนี้ ผลที่ได้มีดังนี้

เลขที่นำเข้า	ได้ผลของการแมตซิ่งเป็น
1	1 50%
	7 50%
5	6 10%
	5 90%
6	6 100%
8	5 10%
	6 50%
	8 40%
ตัวเลขอื่นๆ	ได้ผลตรง 100%

ตารางที่ 5-2 ผลการแมตซิ่ง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การใช้ภาพต้นแบบสามารถแยกหมายเลขห้องออกจากกันได้หลายหมายเลข แต่ยังมีตัวเลขเพียงบางตัวที่ยังไม่สามารถแยกออกจากกันได้ชัดเจน ประกอบด้วย 2 กลุ่มด้วยกัน คือ

- เลข 1 และเลข 7
- เลข 5, 6 และเลข 8

ดังนั้นในการแยกกลุ่มของตัวเลขเหล่านี้ออกมาจะต้องใช้ลักษณะอื่นๆเข้ามาช่วย

### 5.3.3.2 การแบ่งตัวเลข ระหว่างเลข 1 กับ 7

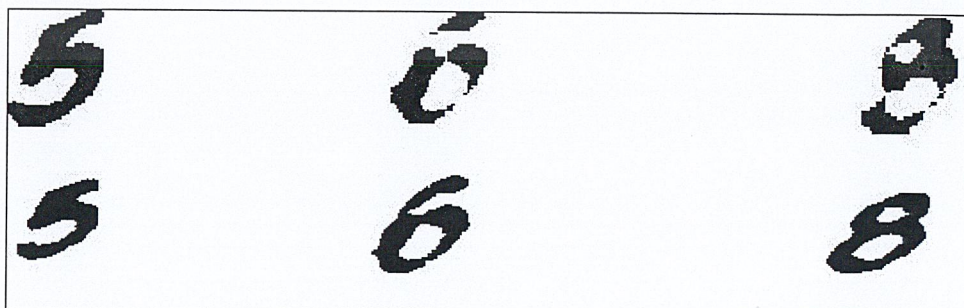
ในกลุ่มแรกซึ่งเป็นกลุ่มตัวเลข 1 กับ 7 ลักษณะที่ต่างกันของตัวเลขทั้งสองนี้ก็คือขนาด ซึ่งเลข 1 จะมีขนาดเล็กกว่าและจะมีช่องว่างก่อนจะถึงตัวเลขจากตำแหน่งก่อนหน้าสูงกว่าเลข 7 และเลข 7 จะมีมุมที่แตกต่างออกไปทำให้กรอบแคบลงซึ่งหมายถึง เลข 7 ก็จะมีกรอบสูงขึ้นด้วย ดังนั้นจึงใช้ข้อแตกต่างข้อนี้เป็นตัวแบ่งแยกเลขสองตัวนี้ออกจากกัน



รูปที่ 5-28 ตัวอย่างของเลข 1 และเลข 7 ที่มีความคล้ายคลึงกันมาก

### 5.3.3.3 การแยกความแตกต่างของตัวเลข 5, 6, และ 8

ความแตกต่างของตัวเลขเหล่านี้ จะอยู่ที่ในตัวเลข 5 จะมีช่องว่างช่วงมุมซ้ายล่าง และ ในเลข 6 จะมีช่องว่างด้านขวาบน ในขณะที่ตัวเลข 8 จะไม่มีช่องว่างที่มุมใดเลย จึงใช้วิธีสโตรปตามทฤษฎีที่ 3.11 เข้ามาช่วยในการจำแนกตัวเลขเหล่านี้



รูปที่ 5-29 ตัวเลข 5 6 และ 8 แบบต่างๆ

## บทที่ 6

### การทดสอบ

#### 6.1 การทดสอบฮาร์ดแวร์

##### 6.1.1 การทดสอบวงจรไครว์มอเตอร์ขับเคลื่อน

ในการทดสอบวงจรไครว์ได้แยกเป็นสามการทดสอบย่อยๆคือ

##### 6.1.1.1 การทดสอบตัววงจรไครว์แบบไม่มีโหลด

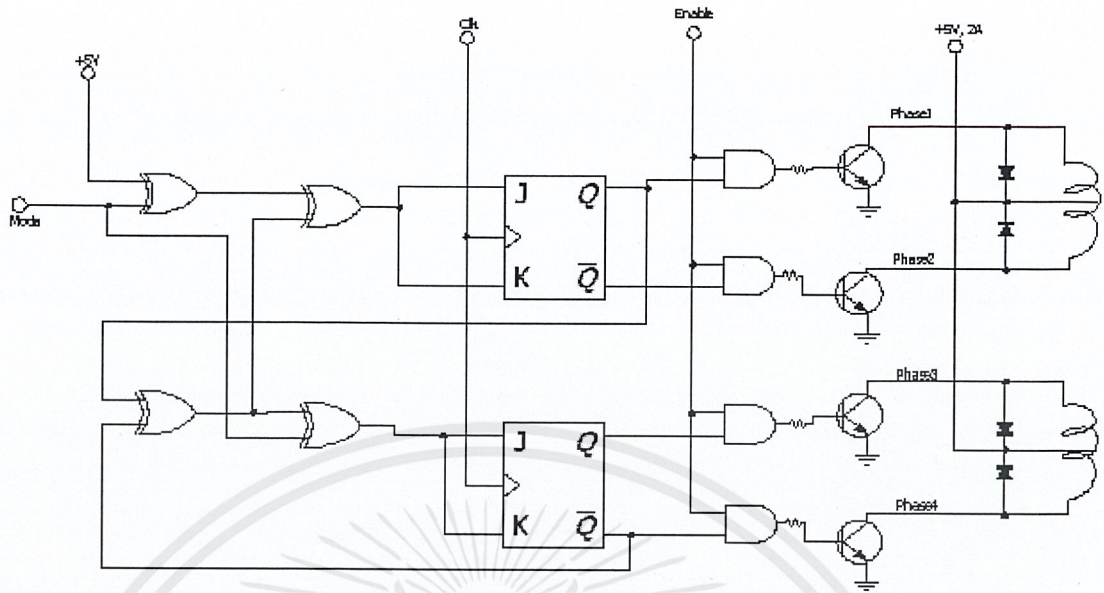
การทดสอบวงจรแบบไม่มีโหลด ทดสอบด้วยการใส่อินพุตเสมือนการใช้จริง ให้กับวงจร ผลที่ได้คือวงจรสามารถทำงานได้ถูกต้องทุกกรณีตาม Logic

##### 6.1.1.2 การทดสอบวงจรด้วยโหลด

การทดสอบวงจรด้วยโหลดคือการนำวงจรมาไครว์มอเตอร์จริงๆ โดยที่มอเตอร์ยึดติดกับล้อ แต่ล้อยังไม่แตะพื้น เป็นการทดสอบวิ่งอยู่กับที่ การทดสอบนี้ได้พบปัญหาของวงจรมาก ถึงกับต้องเปลี่ยนวงจรไครว์มอเตอร์ถึง 3 วงจรด้วยกัน การทดสอบนี้แบ่งออกเป็น 3 ช่วงด้วยกัน

##### 6.1.1.2.1 การใช้วงจรไครว์ที่ 1: วงจร JK-FF กับ Darlington Transistors

เป็นการทดสอบในช่วงแรกๆของการออกแบบวงจรขับเคลื่อนของหุ่นยนต์ โดยการใช้วงจรร่วมระหว่าง JK-FF (74LS76), ไอซีเอกซ์คลูซีฟออร์เกต (74LS86), และ ไอซีแอนด์เกต (74LS08) เป็นตัวสร้างสัญญาณไครว์แบบพลัสตีปและทำการขยายสัญญาณโดย Darlington Transistors เบอร์ TIP120 ซึ่งสามารถขยายกระแสได้ 1000 เท่า จากการวัดค่ากระแสที่ไครว์ออกจากไอซี JK-FF แล้วมีกระแสถึงข้างละ 5mA ซึ่งหลังจากคำนวณแล้วทรานซิสเตอร์หนึ่งตัวสามารถสร้างกระแสได้ถึง 5 แอมแปร์ ซึ่งมากพอที่จะใช้ขับมอเตอร์ซึ่งต้องการกระแสเพียงเส้นละ 2.1 แอมแปร์



รูปที่ 6-1 วงจรไควร์สเตปเปอร์มอเตอร์วงจรที่ 1

**ผลการทดสอบ**

วงจรสามารถไควร์สัญญาณให้กับมอเตอร์ ซึ่งมอเตอร์หมุนได้เพียงระยะเวลานั้น หลังจากนั้นจะหมุนผิดสเต็ป หลังจากการตรวจสอบแล้วพบว่า เป็นเพราะตัววงจรมีสายไฟมากเพราะประกอบด้วยไอซีถึง 3 ตัวและทรานซิสเตอร์ถึง 4 ตัวต่อวงจรมอเตอร์หนึ่งตัว จึงทำให้เกิดการหลุด หรือเกิดการช็อตกันของสายไฟซึ่งดูด้วยตาเปล่าไม่เห็น ทำให้ต้องเสียเวลามากในการตามตรวจเช็คสายไฟทุกเส้น นอกจากนี้ วงจรนี้ยังขับมอเตอร์ได้ไม่แรง เมื่อลองวัดกระแสที่เข้ามอเตอร์ดูแล้วพบว่า มีกระแสเพียงเส้นละ 1 แอมแปร์เท่านั้น ไม่ได้ตามที่คำนวณไว้

**6.1.1.2.2 การใช้วงจรไควร์ที่ 2: วงจร L297 กับ Darlington Transistors**

เมื่อปัญหาของวงจรแรกคือการที่มีสายไฟมากเกินไป จึงแก้ปัญหาด้วยการออกแบบวงจรไควร์ใหม่โดยพยายามใช้ไอซีให้จำนวนน้อยที่สุด หลังจากการหาข้อมูลจึงพบว่า ไอซี L297 สามารถไควร์มอเตอร์ได้โดยไม่ต้องการไอซีใดๆมาต่อพ่วงทั้งสิ้น จึงได้ออกแบบวงจรไควร์ที่ใช้ไอซี L297 เป็นตัวสร้างสัญญาณไควร์แบบพลัสเต็ปและทำการขยายสัญญาณโดยใช้ Darlington Transistors เหมือนเดิม ซึ่ง ไอซี L297 สามารถสร้างกระแสที่ขาออกได้ถึงข้างละ 20mA เมื่อใช้ร่วมกับทรานซิสเตอร์ซึ่งมีแรงขยายถึง 1000 เท่าแล้วทำให้สามารถขยายกระแสได้สูงถึง 20 แอมแปร์ต่อเส้น ซึ่งมากกว่าวงจรที่ 1 ถึง 4 เท่าตัวด้วยกัน

**ผลการทดสอบ**

วงจรมีสามารถขับสเตปเปอร์มอเตอร์สองตัวได้โดยที่มอเตอร์มีแรงมากพอ จึงได้ทำการทดสอบด้วยการไควร์มอเตอร์ติดต่อกันเป็นเวลาครึ่งชั่วโมง ได้ผลออกมาดีเพราะสามารถไควร์มอเตอร์ได้ในทุกกรณีและอยู่ได้นาน แต่หลังจากการนำวงจรนี้ไปใช้จริงในขั้นตอนการออกแบบการอินเตอร์เฟสและเขียน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โปรแกรมควบคุม ครั้งละประมาณครึ่งชั่วโมง ทุกวัน เป็นระยะเวลาประมาณหนึ่งเดือน วงจรนี้เริ่มไครว์มอเตอร์ผิดสแต็ป และไม่สามารถไครว์มอเตอร์ให้ถูกต้องได้อีกเลย

เมื่ วงจรนี้ไม่สามารถทำงานได้อย่างถูกต้องจึงทำการทดสอบด้วยการไล่สายไฟทุกเส้น แต่ก็ไม่พบสิ่งผิดปกติใดๆทั้งสิ้น แต่เมื่อได้ลองใช้มือสัมผัสกับไอซี L297 ขณะที่ทดสอบด้วยการไครว์มอเตอร์ซึ่งผิดสแต็ปนั้น พบว่าไอซีร้อนมาก หลังจากการปรึกษากับอาจารย์ที่ปรึกษาจึงได้ทราบว่า ไอซีร้อนเพราะสร้างกระแสมากเกินไปเป็นเวลานานๆ และเสียดงในที่สุด ปัญหาของวงจรนี้จึงตกอยู่ที่ Darlington Transistors ซึ่งดึงกระแสออกมาจากไอซี L297 มากเกินไป ซึ่งเมื่อเปิดค่าชิทของทรานซิสเตอร์เบอร์ TIP120 จึงพบว่าทรานซิสเตอร์ตัวนี้ต้องการ IB หรือกระแสอินพุต อย่างน้อยที่สุด 120mA ในขณะที่ไอซี L297 สามารถสร้างกระแสได้เพียง 20mA ซึ่งไม่พอต่อความต้องการของทรานซิสเตอร์ ทำให้วงจรนี้ไม่สามารถใช้ได้อีกต่อไปถึงแม้ว่าจะนำไอซีใหม่มาเปลี่ยน ก็ยังทำให้ไอซีเสียดงตลอดไม่มีที่สิ้นสุด

### 6.1.1.2.3 การใช้วงจรไครว์ที่ 3: วงจร L297 กับไอซีขยายสัญญาณ L298

เมื่อปัญหาของวงจรที่สองคือไอซีไครว์กระแสมากเกินไป การออกแบบวงจรไครว์ครั้งที่สามนี้จึงพิจารณาถึง Darlington Transistor เบอร์อื่นๆ แต่ก็ไม่พบเบอร์ไหนที่ตรงตามความต้องการ จึงได้เลือกใช้ไอซีเบอร์ L298 ซึ่งทางผู้ผลิตไอซีเบอร์ L297 แนะนำให้ใช้คู่กัน ไอซีเบอร์ L298 นี้ทำหน้าที่รับกระแสจากไอซี L297 แล้วขยายกระแสให้สูงถึงเส้นละ 2 แอมแปร์ซึ่งพอดีกับขนาดของมอเตอร์

#### ผลการทดสอบ

วงจรสามารถไครว์มอเตอร์ได้ถูกต้องติดต่อกันเป็นเวลานานโดยที่อุณหภูมิของไอซี L297 ไม่สูง แต่ความร้อนของไอซี L298 จะค่อนข้างสูง แต่ก็ไม่มากนักเมื่อติดฮีทซิงค์แล้ว วงจรสามารถขับมอเตอร์ได้โดยมีแรงพอสมควร ถึงแม้ว่าจะแรงน้อยกว่า Darlington Transistors แต่ก็ยังอยู่ในระดับที่รับได้ และวงจรนี้จะมีข้อดีกว่าการใช้ทรานซิสเตอร์เพิ่มขึ้นอีกหนึ่งอย่างคือ เมื่อหุ่นยนต์ไม่ได้ขยับตัว หรือไม่ได้หมุนมอเตอร์ ไอซี L298 จะไม่จ่ายไฟ ซึ่งจะทำให้ไม่สิ้นเปลืองพลังงานแล้วยังสามารถลดอุณหภูมิของไอซีลงได้มากพอสมควรข้อเสียที่พบมีอยู่อย่างเดียวคือขาของไอซีเบอร์ L298 ค่อนข้างอ่อนทำให้หลุดจากซอกเกตได้ค่อนข้างง่าย การบัดกรีไอซี L298 ลงบนบอร์ดเลยจะเป็นหนทางที่ดีกว่าการใช้ซอกเกต

### 6.1.1.2.4 บทสรุปของวงจรไครว์มอเตอร์ขับเคลื่อน

เมื่อทดลองใช้วงจรไครว์มาทั้งสามวงจรแล้วจึงได้บทสรุปว่าวงจรไครว์สเตปเปอร์มอเตอร์แบบไบโพลาร์ กระแสเส้นละ 2.1 แอมแปร์ ที่ใช้ไอซี L297 และ L298 เป็นวงจรที่ดีที่สุดซึ่งนอกจากจะใช้ไอซีจำนวนน้อยแล้วยังสามารถไครว์มอเตอร์ได้ไม่มีปัญหาใดๆทั้งสิ้น

วงจร	จำนวนไอซีที่ใช้	จำนวนทรานซิสเตอร์	กระแสที่สร้างได้	แรงบิดของมอเตอร์
วงจรที่ 1: JK-FF กับ Darlington	4 ตัวต่อมอเตอร์ 1 ตัว	4 ตัวต่อมอเตอร์ 1 ตัว	สูง	น้อย
วงจรที่ 2: L297 กับ Darlington	3 ตัวต่อมอเตอร์ 1 ตัว	4 ตัวต่อมอเตอร์ 1 ตัว	ค่อนข้างสูง	ค่อนข้างสูง
วงจรที่ 3: L297 กับ L298	2 ตัวต่อมอเตอร์ 1 ตัว	-	ค่อนข้างสูง	กลาง

ตารางที่ 6-1 ผลการทดสอบวงจรไครว์มอเตอร์ขับเคลื่อน

### 6.1.1.3 การทดสอบด้วยการนำไปจริง

ทดลองให้หุ่นยนต์วิ่งจริงโดยการที่ประกอบคอมโพเนนต์ทุกส่วนในตัวหุ่นยนต์ นำโหลขนาด 2 กิโลกรัมขึ้นไปวางไว้บนตัวหุ่น แล้วลองเดินดู ปรากฏว่าหุ่นยนต์สามารถเดินได้ในความเร็วที่ต่ำ และค่อนข้างมีปัญหาตอนเลี้ยว มอเตอร์ขับเคลื่อนมีแรงไม่พอที่จะทำให้หุ่นยนต์เคลื่อนที่ไปได้อย่างราบรื่น

#### การแก้ไข

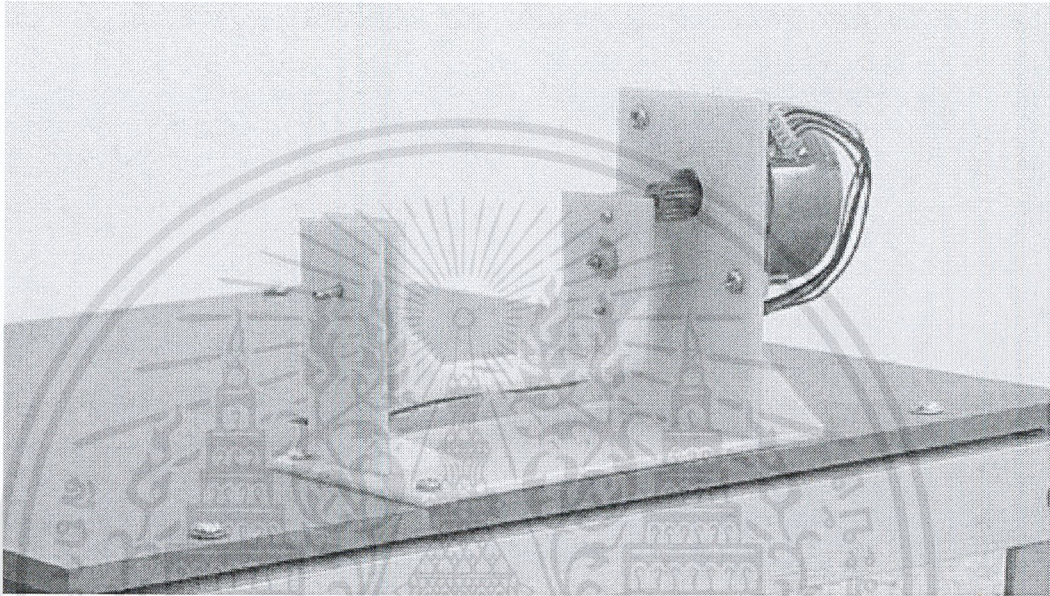
เมื่อลองตรวจกระแสทั้งหมดแล้วพบว่าใช้กระแสไปทั้งสิ้น 4 แอมแปร์ในการไครว์มอเตอร์แบบฟูลสเต็ปสองตัว หมายถึงกระแส 4 แอมแปร์ได้หารเฉลี่ยให้กับขาของมอเตอร์ข้างละ 1 แอมแปร์ ซึ่งเท่ากับครึ่งหนึ่งของขนาดที่มอเตอร์สามารถรับได้ จึงได้ลองตรวจทานดูและพบว่า เป็นเพราะวงจรสร้างสัญญาณค็อกสร้างสัญญาณไฮและโลว์ด้วยขนาดที่ต่างกันมากเกินไป จึงทำการปรับวงจรสร้างสัญญาณค็อกให้พอดี หุ่นยนต์จึงเดินได้ราบรื่นมากขึ้น

## 6.1.2 การทดสอบวงจรไดรฟ์มอเตอร์ของส่วนควบคุมมวกถ็อง

การทดสอบวงจรไดรฟ์นี้แบ่งเป็นสองส่วนด้วยกันคือ

### 6.1.2.1 การทดสอบวงจรโดยไม่มีโหลด

วงจรไดรฟ์สามารถสร้างสัญญาณแบบฮาล์ฟสเต็มไปให้กับสเตปเปอร์มอเตอร์ได้เป็นระยะเวลาานอย่างถูกต้อง ไม่พบปัญหาใดๆ มอเตอร์สามารถหมุนฐานวางถ็องได้รอบทิศ 360 องศา



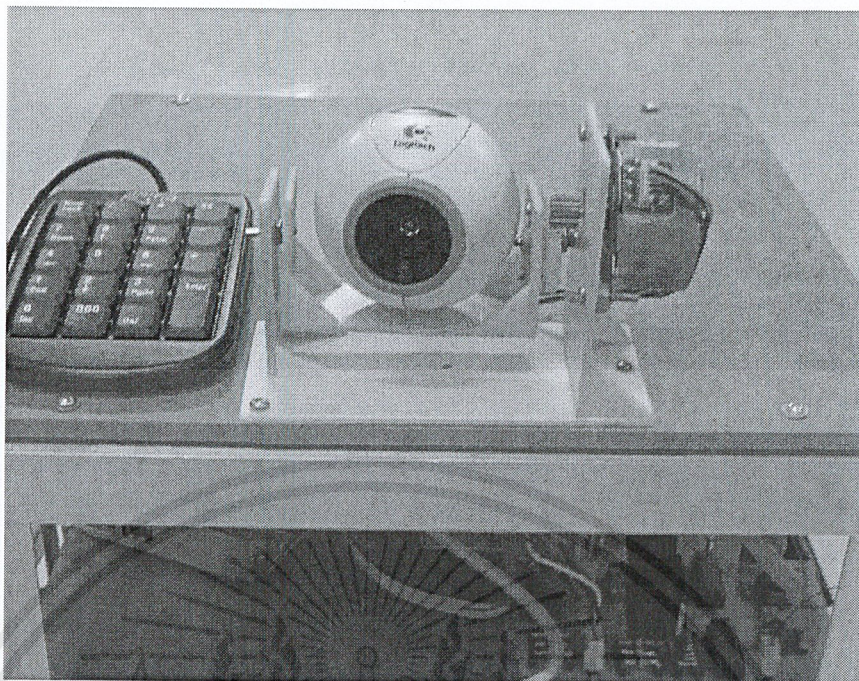
รูปที่ 6-2 การทดสอบหมุนฐานวางถ็องด้วยสเตปเปอร์มอเตอร์

### 6.1.2.2 การทดสอบวงจรที่มีโหลด

เมื่อนำถ็องมาติดตั้งลงบนฐานวางถ็องแล้วให้มอเตอร์ลองหมุนดู พบว่าในบางสเต็ม มอเตอร์ไม่มีแรงมากพอที่จะหมุนถ็องได้ ซึ่งปัญหาไม่ได้เกิดจากน้ำหนักของถ็อง แต่เกิดจากสายของถ็องที่มีความแข็งค่อนข้างมาก ซึ่งสายนี้เป็นสาย USB ที่มีฉนวนป้องกันสิ่งรบกวนหุ้มอยู่จึงหนาและค่อนข้างแข็ง การที่จะเลาะฉนวนออกจะทำให้เกิดสิ่งรบกวนขึ้นและภาพจะไม่ชัด

#### การแก้ไข

การแก้ไขทำได้โดยเจาะรูที่ส่วนบนของหุ่นยนต์และร้อยสายถ็องลงไปพร้อมทั้งทาสารหล่อลื่นที่สายส่วนที่มีการเสียดสี ผลออกมาได้เป็นที่น่าพอใจ มอเตอร์สามารถหมุนปรับมวกถ็องได้



รูปที่ 6-3 การทดสอบหมุนส่วนควบคุมมอเตอร์ที่ติดตั้งกล่องเรียบร้อยแล้ว

### 6.1.3 การทดสอบ PC Board

เพื่อให้หุ่นยนต์ตัวนี้เป็น โมบายเต็มรูปแบบ (Mobile Robot) จึงต้องการที่จะใช้แบตเตอรี่เป็นตัวจ่ายไฟให้กับ PC Board จึงได้ทดลองเป็นขั้นตอนดังนี้

#### 6.1.3.1 การทดสอบวัดกระแสที่บอร์ดใช้

ทดสอบด้วยการใส่แอมป์มิเตอร์เข้าไปที่สายไฟ DC ที่ออกจาก Power Supply แล้วเข้าสู่บอร์ดโดยตรง พบว่า

- สายไฟสีแดง (+5v) ใช้กระแส 2.1 แอมแปร์
- สายไฟสีเหลือง (+9v) ใช้กระแส 0.6 แอมแปร์
- สายไฟสีดำ (GND) ไม่ต้องการกระแส
- สายไฟสีขาว (-5v) ไม่ต้องการกระแส
- สายไฟสีน้ำเงิน (-9v) ไม่ต้องการกระแส
- สายสีส้ม (สายดิน) ไม่ต้องการกระแส

นอกจากนี้ยังวัดกระแสที่อุปกรณ์ต่อพ่วง IDE ใช้อีกด้วย ได้ผลดังนี้

- ฮาร์ดดิสก์
  - สายไฟสีแดง (+5v) ใช้กระแส 0.6 แอมแปร์
  - สายไฟสีเหลือง (+9v) ใช้กระแส 0.05 แอมแปร์
- พัดลมต่างๆ
  - สายไฟสีเหลือง (+9v) ใช้กระแสรวมทั้งหมด 0.6 แอมแปร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

และนำกระแสทั้งหมดที่วัดได้มารวมกันได้เท่ากับ 3.95 แอมแปร์ ซึ่งได้เลือกใช้เบตเตอร์ขนาด 12 โวลต์ 7 แอมแปร์มาเป็นตัวจ่ายไฟพร้อมกับไอซีเรกกูเลต 7805 และ 7809 จำนวนมากพอที่จะจ่ายกระแสได้

#### ผลการทดสอบ

การใช้เบตเตอร์เป็นตัวจ่ายไฟให้กับ PC Board ประสบผลสำเร็จ บอร์ดบูตขึ้น ไม่มีปัญหา แต่มีปัญหาเกี่ยวกับฮาร์ดดิสก์ซึ่งเสียหายหลังจากจ่ายไฟให้ เมื่อทำการวัดกระแสและแรงดันทั้งหมดที่จ่ายให้กับฮาร์ดดิสก์แล้วก็พบว่าเท่ากับที่ใช้ Supply จ่ายให้ทุกประการ จึงได้ลองนำฮาร์ดดิสก์เก่าๆมาอีกจำนวน 4 ตัวและลองวัดกระแสแล้วจ่ายไฟให้ ผลที่ได้คือฮาร์ดดิสก์เสียทุกตัว

#### การแก้ไข

เมื่อการจ่ายไฟจากเบตเตอร์มีปัญหาเกี่ยวกับฮาร์ดดิสก์แล้วจึงจำเป็นต้องใช้แหล่งไฟจาก Power Supply โดยตรง จึงทำการนำ Supply ไปติดไว้ในตัวหุ่นยนต์เพื่อให้จ่ายไฟให้กับอุปกรณ์ทุกตัว แต่ก็มีปัญหาเกี่ยวกับมอเตอร์ขับเคลื่อนที่ไม่สามารถจ่ายกระแสให้ได้มากพอ จึงจำเป็นต้องมีเบตเตอร์เฉพาะให้กับมอเตอร์ขับเคลื่อน หุ่นยนต์ตัวนี้จึงมีทั้งเบตเตอร์, Power Supply, และลากสายไฟไปด้วย จึงไม่ใช่โมบายโรบ็อตอย่างเต็มตัว

#### 6.1.4 การทดสอบโครงหุ่นยนต์

ได้ทดสอบความแข็งแรงของโครงหุ่นยนต์ด้วยการนำโหลตขึ้นไปวางไว้เป็นจำนวนมาก โครงสามารถรับน้ำหนักได้โดยไม่มีปัญหา จึงได้ทดสอบโดยการขึ้นไปยืนบนตัวหุ่นยนต์เลย พบว่าโครงสามารถรับน้ำหนักตัวได้อย่างดี ไม่น่าจะมีปัญหาใดๆในการแบกรับน้ำหนักของคอม โปเนนต์ทั้งหมดและเอกสารที่ทำการส่งได้ เพราะน้ำหนักของอุปกรณ์ทั้งหมดย่อมเบากว่าน้ำหนักของมนุษย์อยู่แล้ว

#### 6.1.5 การทดสอบการอินเตอร์เฟส PC Board เข้ากับอุปกรณ์อื่น

การทดลองอินเตอร์เฟส PC Board เข้ากับอุปกรณ์อื่นคือ

- วงจรขับเคลื่อน (บิตต่ำของพอร์ตพาราเรล)
- วงจรปรับมุมกล้อง (บิตสูงของพอร์ตพาราเรล)
- กล้อง (USB)
- ส่วนรับข้อมูลหมายเลขห้อง (USB)

พบว่า PC Board สามารถอินเตอร์เฟสกับอุปกรณ์ทุกอย่างได้ดีไม่มีปัญหาใดๆเกิดขึ้น ผลการทดสอบเป็นดังตารางต่อไปนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จำนวนข้อมูลที่ส่ง (บิต)	ความผิดพลาดที่พบ	เปอร์เซ็นต์สำเร็จ	ระยะเวลาที่เฉลี่ย
10	0	100%	มองไม่เห็นด้วยตาเปล่า
100	0	100%	มองไม่เห็นด้วยตาเปล่า
500	0	100%	มองไม่เห็นด้วยตาเปล่า
1000	0	100%	มองไม่เห็นด้วยตาเปล่า

ตารางที่ 6-2 ผลการทดสอบส่งข้อมูลระหว่าง PC Board และอุปกรณ์ต่อพ่วงต่างๆ

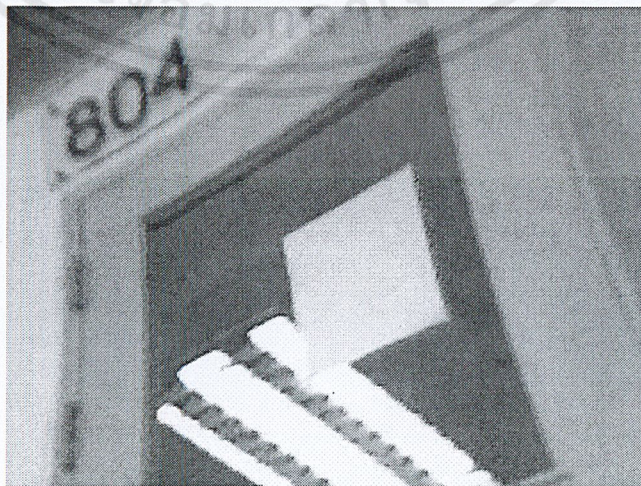
## 6.2 การทดสอบซอฟต์แวร์

### 6.2.1 การทดสอบประสิทธิภาพการตัดลิ่งรบกวนด้วยภายใต้สิ่งแวดล้อมต่างๆ

รูปแบบลิ่งรบกวน	ความถูกต้อง
ไม่มี	100%
มีแสงจากหลอดไฟแนวยาว	100%
มีแสงจากหลอดไฟวงกลม	60%

ตารางที่ 6-3 การทดสอบประสิทธิภาพในลิ่งแวดล้อมแตกต่างกัน

การทดสอบภายใต้สิ่งแวดล้อมที่มีเงาจากแสงไฟในรูปแบบต่างๆดังในตัวอย่างในรูปที่ 6-4 และ 6-5 จะพบว่าแสงที่มาจากหลอดไฟแบบกลมทำให้ ในการทำแมตซึ่งอาจให้ค่าสูงสุดที่ดวงไฟได้



รูปที่ 6-4 มีแสงไฟแนวยาว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



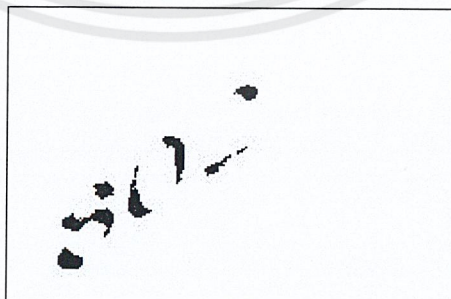
รูปที่ 6-5 มีแสงไฟกลม

## 6.2.2 การทดสอบประสิทธิภาพการจดจำภาพตัวเลขภายใต้แสงแบบต่างๆ

รูปแบบแสง	ความถูกต้องในการจดจำ
แสงปกติ	100%
มีแสงเข้ามาบกรวน	99%

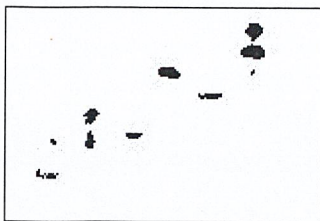
ตารางที่ 6-4 การทดสอบประสิทธิภาพในแบบแสงที่แตกต่างกัน

อย่างไรก็ตามการมีแสงเข้ามาบกรวน ถึงแม้ว่าจะให้ผลการจดจำที่ถูกต้อง ค่าที่ได้ในการทำแมตซึ่งจะมีค่าที่ได้ต่ำกว่าแสงปกติ ตัวอย่างในรูปที่ 6-6 ภาพมีแสงเข้ามาบกรวน โดยเฉพาะบริเวณ คอของเลข 2 ในภาพนี้สามารถจดจำตัวเลขได้แต่ ค่าคอร์เรลจะอยู่ที่ประมาณ 20% เมื่อเทียบกับในการจดจำปกติจะได้อยู่ที่ 60% ขึ้นไป



รูปที่ 6-6 ตัวอย่างภาพที่ตัดเทรซโฮลจากภาพที่มีแสงเข้ามาบกรวน

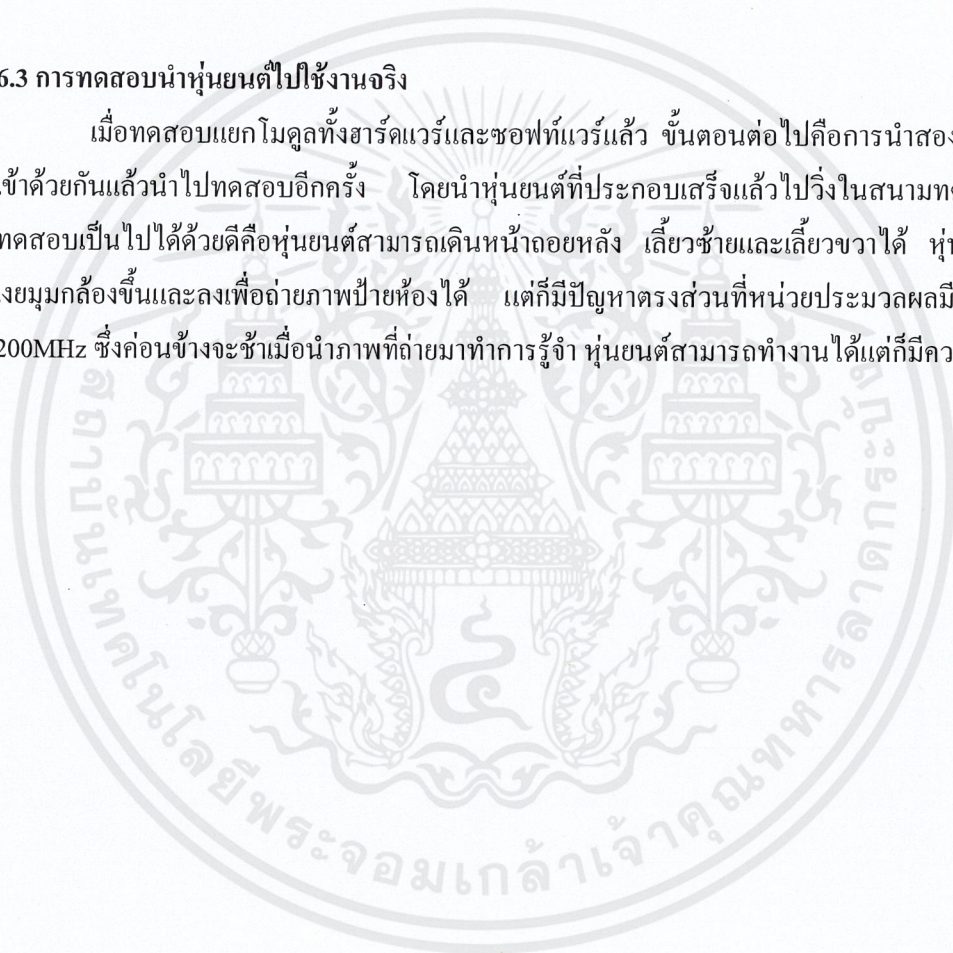
รูปที่ 6-7 เป็นรูปที่เกิดแสงเข้ามากระทบกับแว่นขยาย เข้ามารบกวนภาพอาจทำให้ผลที่ได้มาไม่ถูกต้อง โดยในรูปแบบนี้ไม่สามารถที่จะจดจำรูปแบบได้



รูปที่ 6-7 ตัวอย่างภาพที่ตัดทรานซิสเตอร์จากภาพที่มีแสงสะท้อนที่แว่นขยาย

### 6.3 การทดสอบนำหุ่นยนต์ไปใช้งานจริง

เมื่อทดสอบแยกโมดูลทั้งฮาร์ดแวร์และซอฟต์แวร์แล้ว ขั้นตอนต่อไปคือการนำสองส่วนนี้มารวมเข้าด้วยกันแล้วนำไปทดสอบอีกครั้ง โดยนำหุ่นยนต์ที่ประกอบเสร็จแล้วไปวิ่งในสนามทดสอบ การทดสอบเป็นไปได้ด้วยดีคือหุ่นยนต์สามารถเดินหน้าถอยหลัง เลี้ยวซ้ายและเลี้ยวขวาได้ หุ่นยนต์สามารถยกมุมกล้องขึ้นและลงเพื่อถ่ายภาพป้ายห้องได้ แต่ก็มีปัญหาตรงส่วนที่หน่วยประมวลผลมีความเร็วเพียง 200MHz ซึ่งค่อนข้างจะช้าเมื่อนำภาพที่ถ่ายมาทำการรู้จำ หุ่นยนต์สามารถทำงานได้แต่ก็มีความเร็วที่ต่ำ



## บทที่ 7

### สรุปผลและวิจารณ์

หุ่นยนต์ตัวนี้สามารถตอบสนองกับผู้ใช้งาน ได้อย่างไม่ค่อนแค้นที่ สามารถส่งข้อมูลควบคุมในการทำงานของหุ่นยนต์ได้ สามารถเดินได้ด้วยความเร็วต่ำ สามารถรับสัญญาณภาพจากกล้องแล้วนำไปประมวลผลต่อได้ แต่ยังมีปัญหาในการใช้งานอยู่บ้าง คือ ใช้งานได้ยาก และไม่ค่อยมีแรงเดินนัก ส่วนวงจรควบคุมการเคลื่อนที่ของหุ่นยนต์ หุ่นยนต์สามารถเคลื่อนที่ เดินหน้า ถอยหลัง เลี้ยวซ้าย เลี้ยวขวา ได้อย่างถูกต้องตามที่โปรแกรมไว้ ส่วนควบคุมมุมกล้องก็สามารถปรับมุมของกล้องให้ก้มและเงยได้ สามารถตรวจจับตำแหน่งของหมายเลขห้องได้ สามารถจดจำได้ว่าหมายเลขห้องที่ถ่ายรูปนั้นเป็นห้องอะไร

#### แนวทางในการแก้ไขปัญหา

ส่วนสำคัญที่สุดคือการเปลี่ยนมอเตอร์ขับเคลื่อนใหม่ให้มีความเร็วมากขึ้น แก้ไขโปรแกรมในส่วนติดต่อกับผู้ใช้ใหม่ ให้ง่ายต่อการใช้งานมากขึ้น และถ้าโปรแกรมส่วนติดต่อกับผู้ใช้สามารถใช้งานได้ง่ายมากขึ้น ก็จะทำให้สามารถควบคุมการทำงานของหุ่นยนต์ ได้อย่างถูกต้องมากขึ้น ไปด้วย

ในส่วนของซอฟต์แวร์จะเหลือปัญหาอยู่ที่การกำจัดตัวรบกวนเพียงอย่างเดียวและอัลกอริทึมการเดิน หุ่นยนต์ไม่สามารถจดจำหมายเลขห้องได้หากมีการเปิดไฟในห้องให้แสงผ่านออกมาเข้ากล้อง หรือสิ่งรบกวนที่อยู่ใกล้ขณะเป็นเงา

#### สิ่งที่ได้ทำมาทั้งหมด

- ส่วนของฮาร์ดแวร์
  - ศึกษาข้อมูล
  - ออกแบบตัวหุ่นยนต์
  - ประกอบโครงของหุ่นยนต์
  - สร้างส่วนขับเคลื่อน
  - ทดลองสร้างวงจรอุตราโซนิก
  - ทดลองนำหุ่นยนต์ไปวิ่งจริง
- ส่วนของซอฟต์แวร์
  - การทดลองถ่ายภาพป้ายห้องต่างๆมาทำการรู้จำ
  - การตรวจหาประตูโดยใช้กล้อง
  - การทดลองตรวจจับประตู
  - การใช้เทคนิคในการลดตัวรบกวนต่างๆ
  - การตรวจสอบตำแหน่งของหมายเลขห้องจากภาพ
  - นำหมายเลขที่ตรวจสอบตำแหน่งได้มาทำการรู้จำ

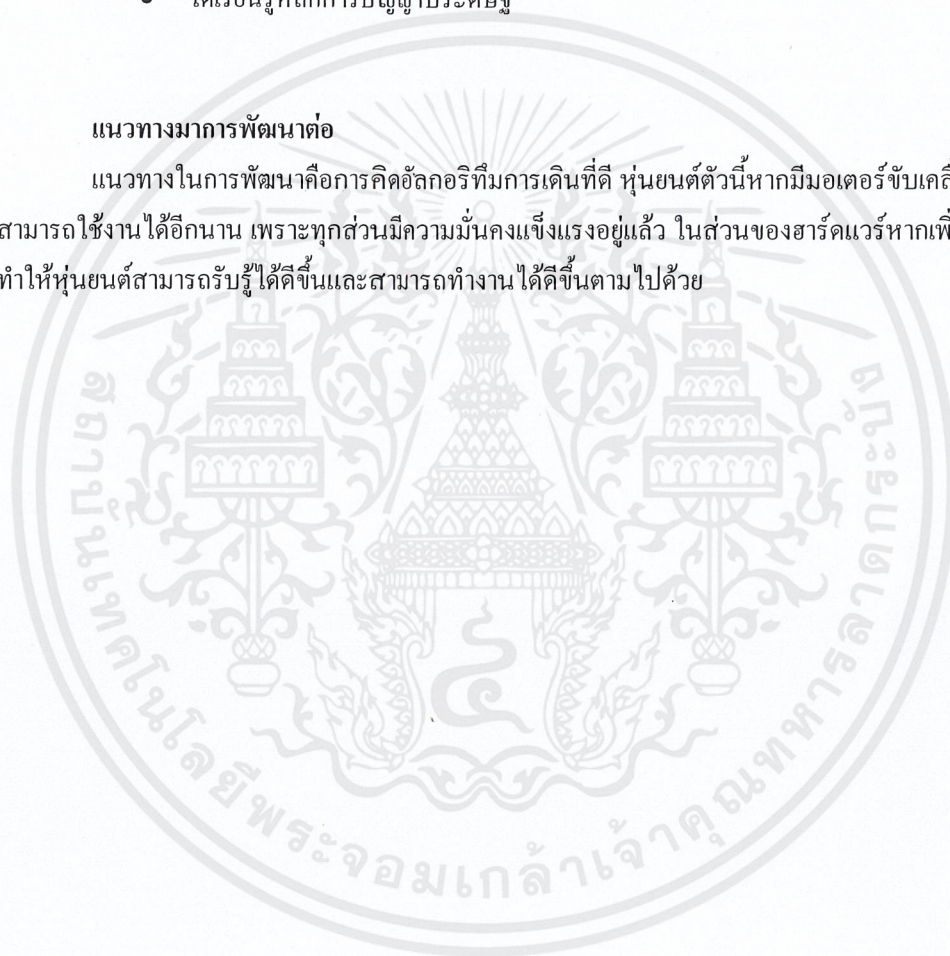
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### สิ่งที่ได้เรียนรู้

- ได้ศึกษาถึงการทำงานของหุ่นยนต์และอุปกรณ์ต่างๆซึ่งประกอบด้วย
  - สเตปเปอร์มอเตอร์
  - กล้อง
  - อุลตราโซนิก
  - การอินเตอร์เฟซพีซีบอร์ด
- การแก้ปัญหาจากความผิดพลาดที่เกิดขึ้นในโครงการ
- ได้เรียนรู้หลักการประมวลผลรูปภาพ
- ได้เรียนรู้หลักการปัญญาประดิษฐ์

### แนวทางมาตรการพัฒนาต่อ

แนวทางในการพัฒนาคือการคิดอัลกอริทึมการเดินที่ดี หุ่นยนต์ตัวนี้หากมีมอเตอร์ขับเคลื่อนที่ดีกว่านี้จะสามารถใช้งานได้อีกนาน เพราะทุกส่วนมีความมั่นคงแข็งแรงอยู่แล้ว ในส่วนของฮาร์ดแวร์หากเพิ่มฟีดแบ็คได้จะทำให้หุ่นยนต์สามารถรับรู้ได้ดีขึ้นและสามารถทำงานได้ดีขึ้นตามไปด้วย





**ภาคผนวก ก:**  
**คำตัดชี้ทของไอซี L297**

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

THE L297 STEPPER MOTOR CONTROLLER

The L297 integrates all the control circuitry required to control bipolar and unipolar stepper motors. Used with a dual bridge driver such as the L298N forms a complete microprocessor-to-bipolar stepper motor interface. Unipolar stepper motor can be driven with an L297 plus a quad darlington array. This note describes the operation of the circuit and shows how it is used.

The L297 Stepper Motor Controller is primarily intended for use with an L298N or L293E bridge driver in stepper motor driving applications.

It receives control signals from the system's controller, usually a microcomputer chip, and provides all the necessary drive signals for the power stage. Additionally, it includes two PWM chopper circuits to regulate the current in the motor windings.

With a suitable power actuator the L297 drives two phase bipolar permanent magnet motors, four phase unipolar permanent magnet motors and four phase variable reluctance motors. Moreover, it handles normal, wave drive and half step drive modes. (This is all explained in the section "Stepper Motor Basics").

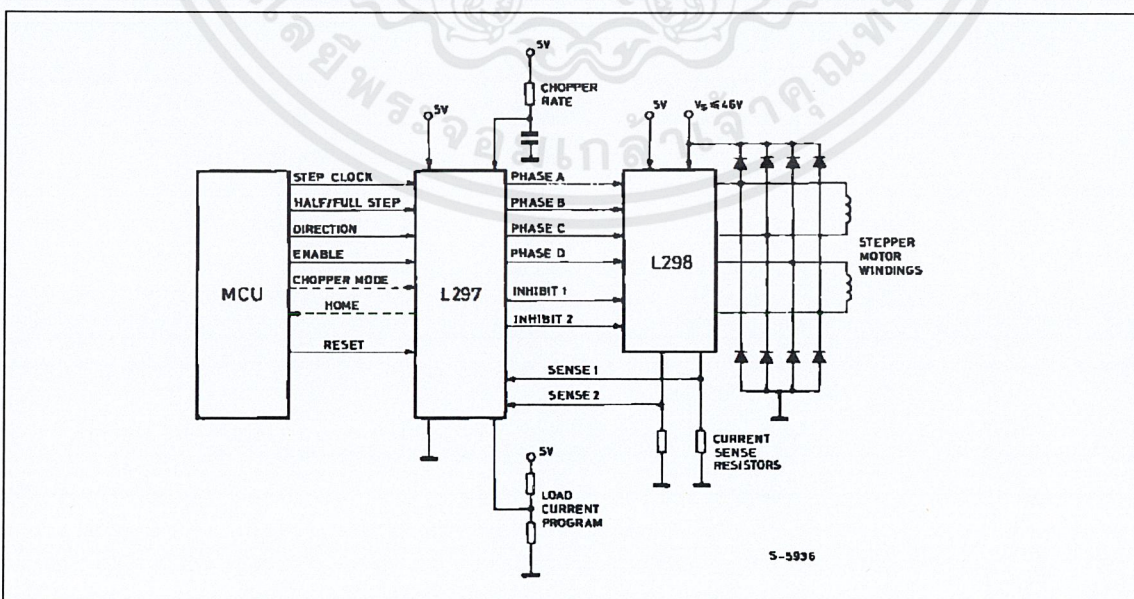
Two versions of the device are available : the regular

L297 and a special version called L297A. The L297A incorporates a step pulse doubler and is designed specifically for floppy-disk head positioning applications.

**ADVANTAGES**

The L297 + driver combination has many advantages : very few components are required (so assembly costs are low, reliability high and little space required), software development is simplified and the burden on the micro is reduced. Further, the choice of a two-chip approach gives a high degree of flexibility-the L298N can be used on its own for DC motors and the L297 can be used with any power stage, including discrete power devices (it provides 20mA drive for this purpose).

**Figure 1 :** In this typical configuration an L297 stepper motor controller and L298 dual bridge driver combine to form a complete microprocessor to bipolar stepper motor interface.



## APPLICATION NOTE

For bipolar motors with winding currents up to 2A the L297 should be used with the L298N; for winding currents up to 1A the L293E is recommended (the L293 will also be useful if the chopper isn't needed). Higher currents are obtained with power transistors or darlington's and for unipolar motors a darlington array such as the ULN2075B is suggested. The block diagram, figure 1, shows a typical system.

Applications of the L297 can be found almost everywhere ... printers (carriage position, daisy position, paper feed, ribbon feed), typewriters, plotters, numerically controlled machines, robots, floppy disk drives, electronic sewing machines, cash registers, photocopiers, telex machines, electronic carburetors, telecopiers, photographic equipment, paper tape readers, optical character recognisers, electric valves and so on.

The L297 is made with SGS' analog/digital compatible  $I^2L$  technology (like Zodiac) and is assembled in a 20-pin plastic DIP. A 5V supply is used and all signal lines are TTL/CMOS compatible or open collector transistors. High density is one of the key features of the technology so the L297 die is very compact.

### THE L298N AND L293E

Since the L297 is normally used with an L298N or L293E bridge driver a brief review of these devices will make the rest of this note easier to follow.

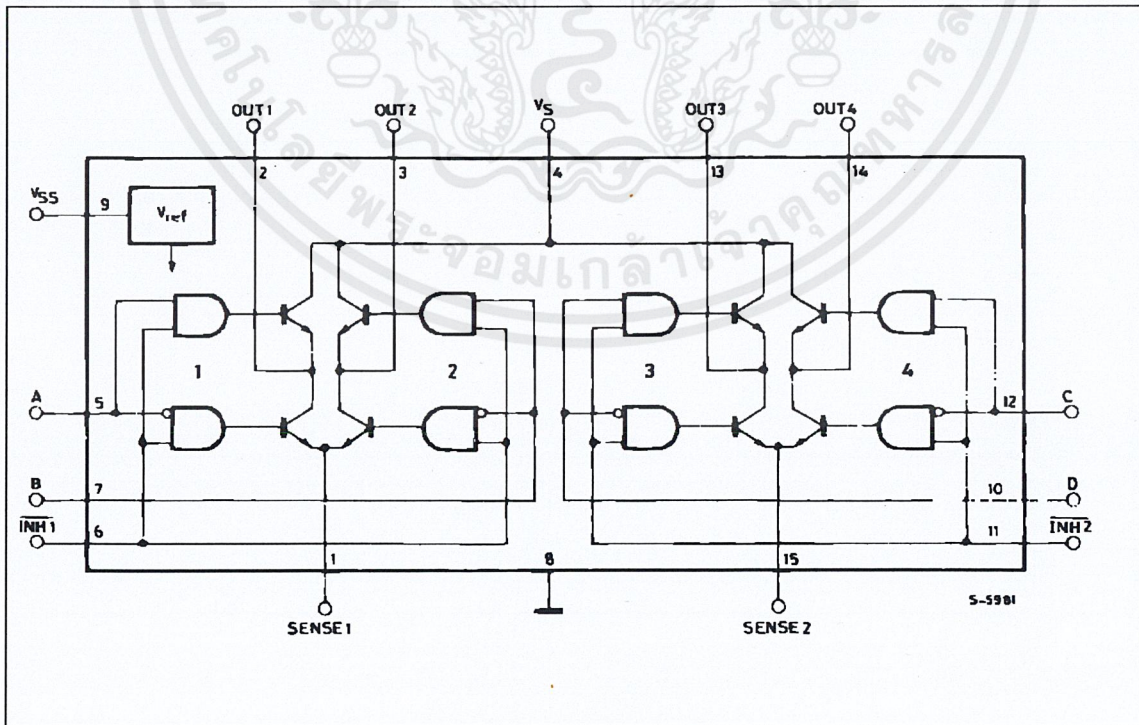
The L298N and L293E contain two bridge driver stages, each controlled by two TTL-level logic inputs and a TTL-level enable input. In addition, the emitter connections of the lower transistors are brought out to external terminals to allow the connection of current sensing resistors (figure 2).

For the L298N SGS' innovative ion-implanted high voltage/high current technology is used, allowing it to handle effective powers up to 160W (46V supply, 2A per bridge). A separate 5V logic supply input is provided to reduce dissipation and to allow direct connection to the L297 or other control logic.

In this note the pins of the L298N are labelled with the pin names of the corresponding L297 terminals to avoid unnecessary confusion.

The L298N is supplied in a 15-lead Multiwatt plastic power package. Its smaller brother, the functionally identical L293E, is packaged in a Powerdip – a copper frame DIP that uses the four center pins to conduct heat to the circuit board copper.

**Figure 2 :** The L298N contains two bridge drivers (four push pull stages) each controlled by two logic inputs and an enable input. External emitter connections are provided for current sense resistors. The L293E has external connections for all four emitters.



## STEPPER MOTOR BASICS

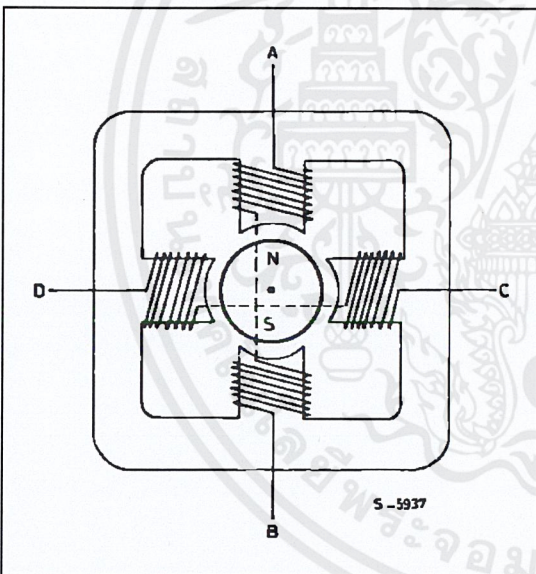
There are two basic types of stepper motor in common use: permanent magnet and variable reluctance. Permanent magnet motors are divided into bipolar and unipolar types.

### BIPOLAR MOTORS

Simplified to the bare essentials, a bipolar permanent magnet motor consists of a rotating permanent magnet surrounded by stator poles carrying the windings (figure 3). Bidirectional drive current is used and the motor is stepped by switching the windings in sequence.

For a motor of this type there are three possible drive sequences.

**Figure 3 :** Greatly simplified, a bipolar permanent magnet stepper motor consist of a rotating magnet surrounded by stator poles as shown.



The first is to energize the windings in the sequence AB/CD/BA/DC (BA means that the winding AB is energized but in the opposite sense). This sequence is known as "one phase on" full step or wave drive

mode. Only one phase is energized at any given moment (figure 4a).

The second possibility is to energize both phases together, so that the rotor always aligns itself between two pole positions. Called "two-phase-on" full step, this mode is the normal drive sequence for a bipolar motor and gives the highest torque (figure 4b).

The third option is to energize one phase, then two, then one, etc., so that the motor moves in half step increments. This sequence, known as half step mode, halves the effective step angle of the motor but gives a less regular torque (figure 4c).

For rotation in the opposite direction (counter-clockwise) the same three sequences are used, except of course that the order is reserved.

As shown in these diagrams the motor would have a step angle of 90°. Real motors have multiple poles to reduce the step angle to a few degrees but the number of windings and the drive sequences are unchanged. A typical bipolar stepper motor is shown in figure 5.

### UNIPOLAR MOTORS

A unipolar permanent magnet motor is identical to the bipolar machine described above except that bifilar windings are used to reverse the stator flux, rather than bidirectional drive (figure 6).

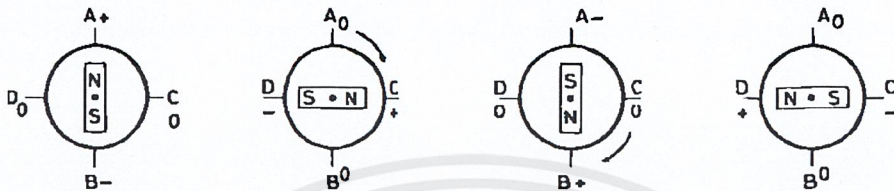
This motor is driven in exactly the same way as a bipolar motor except that the bridge drivers are replaced by simple unipolar stages - four darlington or a quad darlington array. Clearly, unipolar motors are more expensive because they have twice as many windings. Moreover, unipolar motors give less torque for a given motor size because the windings are made with thinner wire. In the past unipolar motors were attractive to designers because they simplify the driver stage. Now that monolithic push pull drivers like the L298N are available bipolar motors are becoming more popular.

All permanent magnet motors suffer from the counter EMF generated by the rotor, which limits the rotation speed. When very high slewing speeds are necessary a variable reluctance motor is used.

## APPLICATION NOTE

Figure 4 : The three drive sequences for a two phase bipolar stepper motor. Clockwise rotation is shown.

Figure 4a : Wave drive (one phase on).



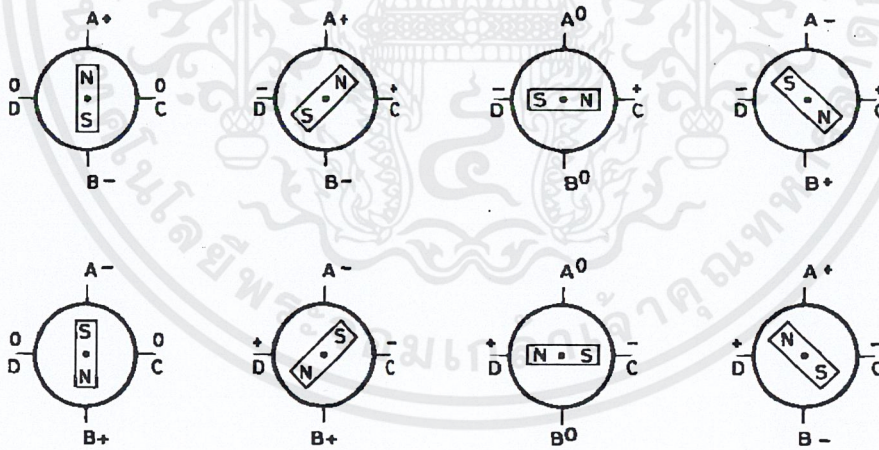
S-5952

Figure 4b : Two phase on drive.

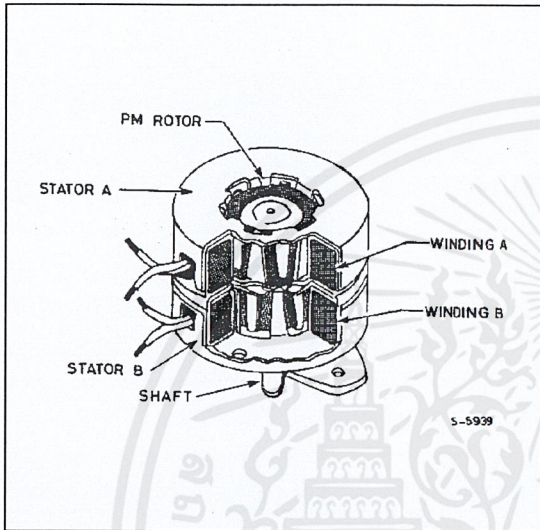


S-5953

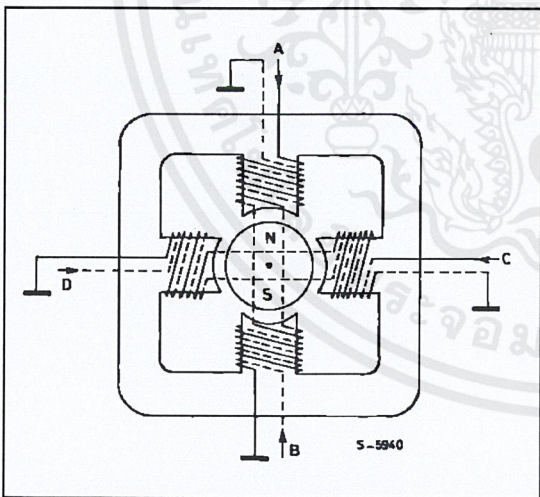
Figure 4c : Half step drive.



**Figure 5 :** A real motor. Multiple poles are normally employed to reduce the step angle to a practical value. The principle of operation and drive sequences remain the same.



**Figure 6 :** A unipolar PM motor uses bifilar windings to reverse the flux in each phase.



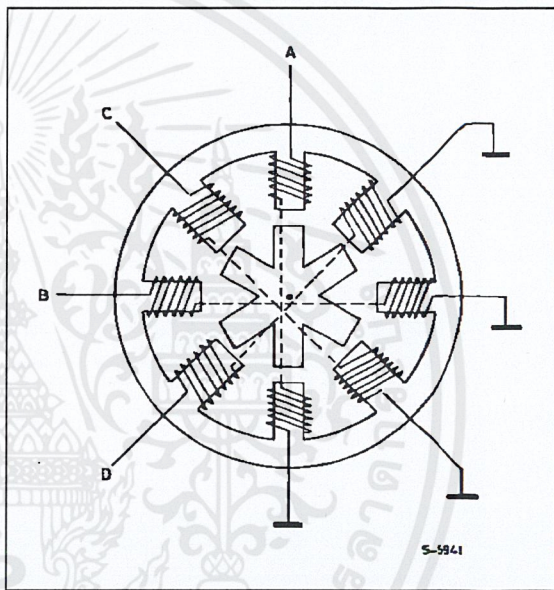
**VARIABLE RELUCTANCE MOTORS**

A variable reluctance motor has a non-magnetized soft iron rotor with fewer poles than the stator (figure 7). Unipolar drive is used and the motor is stepped by energizing stator pole pairs to align the rotor with the pole pieces of the energized winding.

Once again three different phase sequences can be used. The wave drive sequence is A/C/B/D ; two-

phase-on is AC/CB/BD/DA and the half step sequence is A/AC/C/BC/B/BD/D/DA. Note that the step angle for the motor shown above is 15°, not 45°. As before, practical motors normally employ multiple poles to give a much smaller step angle. This does not, however, affect the principle of operation of the drive sequences.

**Figure 7 :** A variable reluctance motor has a soft iron rotor with fewer poles than the stator. The step angle is 15° for this motor.



**GENERATING THE PHASE SEQUENCES**

The heart of the L297 block diagram, figure 8, is a block called the translator which generates suitable phase sequences for half step, one-phase-on full step and two-phase-on full step operation. This block is controlled by two mode inputs – direction (CW/ CCW) and HALF/ FULL – and a step clock which advances the translator from one step to the next.

Four outputs are provided by the translator for subsequent processing by the output logic block which implements the inhibit and chopper functions.

Internally the translator consists of a 3-bit counter plus some combinational logic which generates a basic eight-step gray code sequence as shown in figure 9. All three drive sequences can be generated easily from this master sequence. This state sequence corresponds directly to half step mode, selected by a high level on the HALF/ FULL input.

## APPLICATION NOTE

The output waveforms for this sequence are shown in figure 10.

Note that two other signals,  $\overline{\text{INH1}}$  and  $\overline{\text{INH2}}$  are generated in this sequence. The purpose of these signals is explained a little further on.

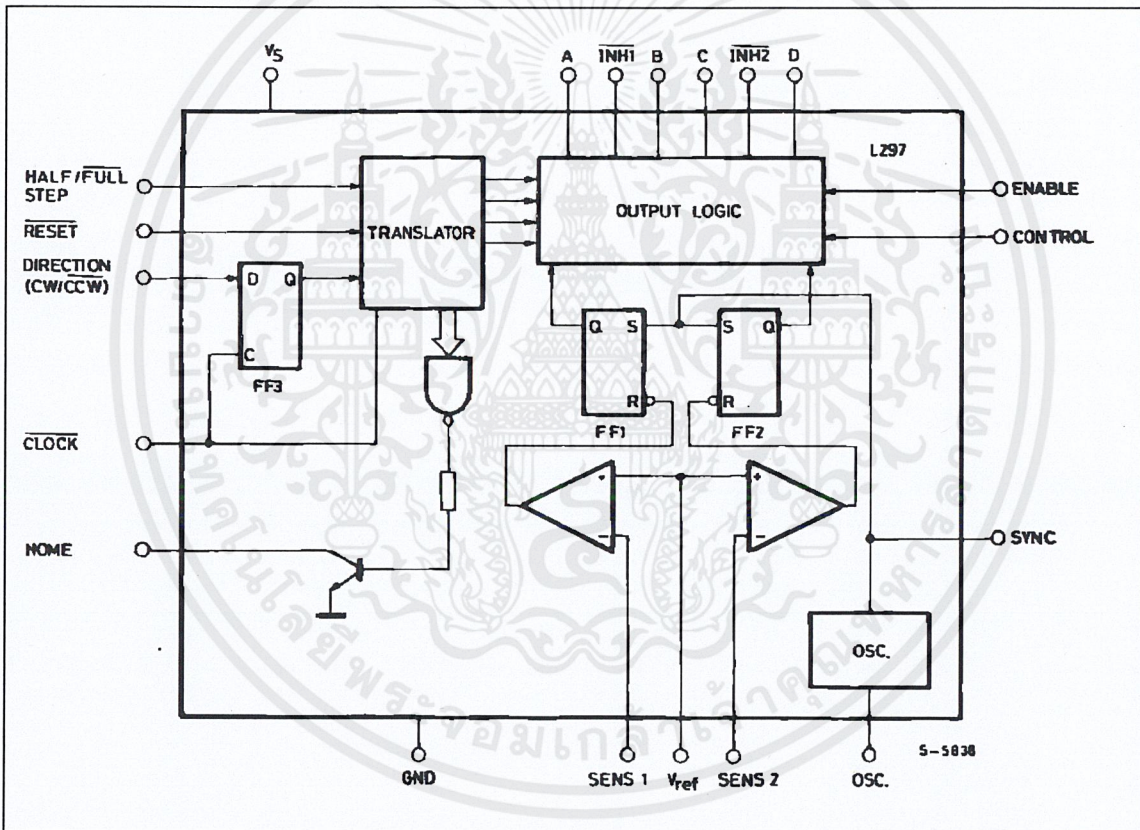
The full step modes are both obtained by skipping alternate states in the eight-step sequence. What happens is that the step clock bypasses the first stage of the 3-bit counter in the translator. The least significant bit of this counter is not affected therefore

the sequence generated depends on the state of the translator when full step mode is selected (the HALF/ FULL input brought low).

If full step mode is selected when the translator is at any odd-numbered state we get the two-phase-on full step sequence shown in figure 11.

By contrast, one-phase-on full step mode is obtained by selecting full step mode when the translator is at an even-numbered state (figure 12).

**Figure 8 :** The L297 contains translator (phase sequence generator), a dual PWM chopper and output control logic.



**Figure 9 :** The eight step master sequence of the translator. This corresponds to half step mode. Clockwise rotation is indicated.

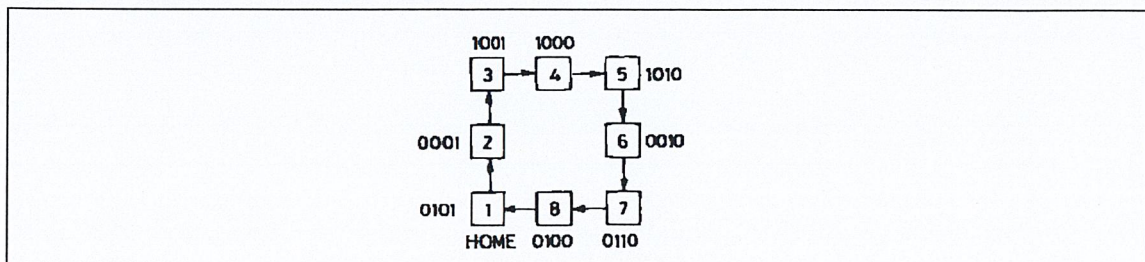


Figure 10 : The output waveforms corresponding to the half step sequence. The chopper action in not shown.

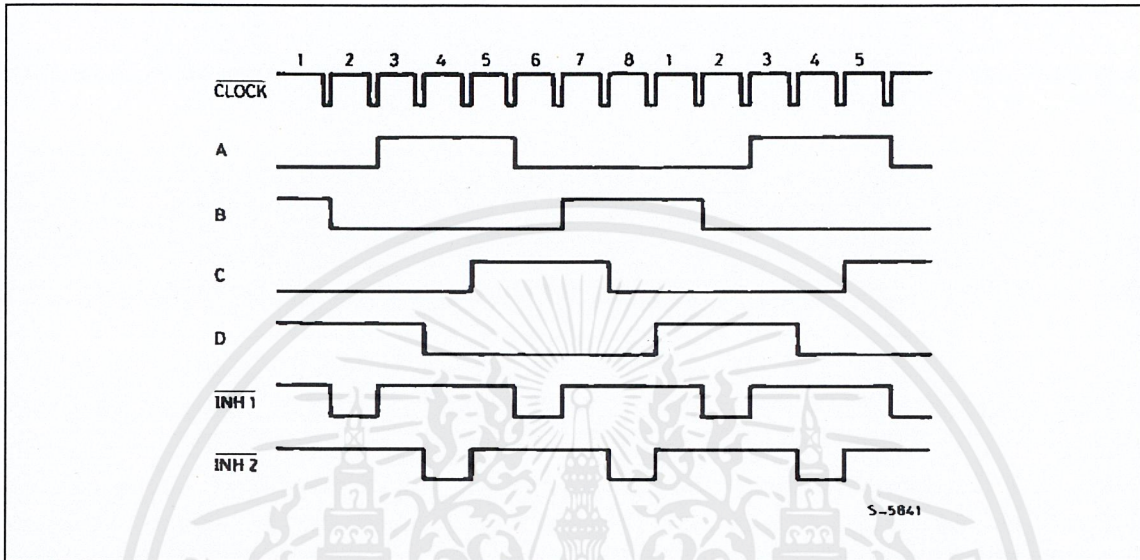
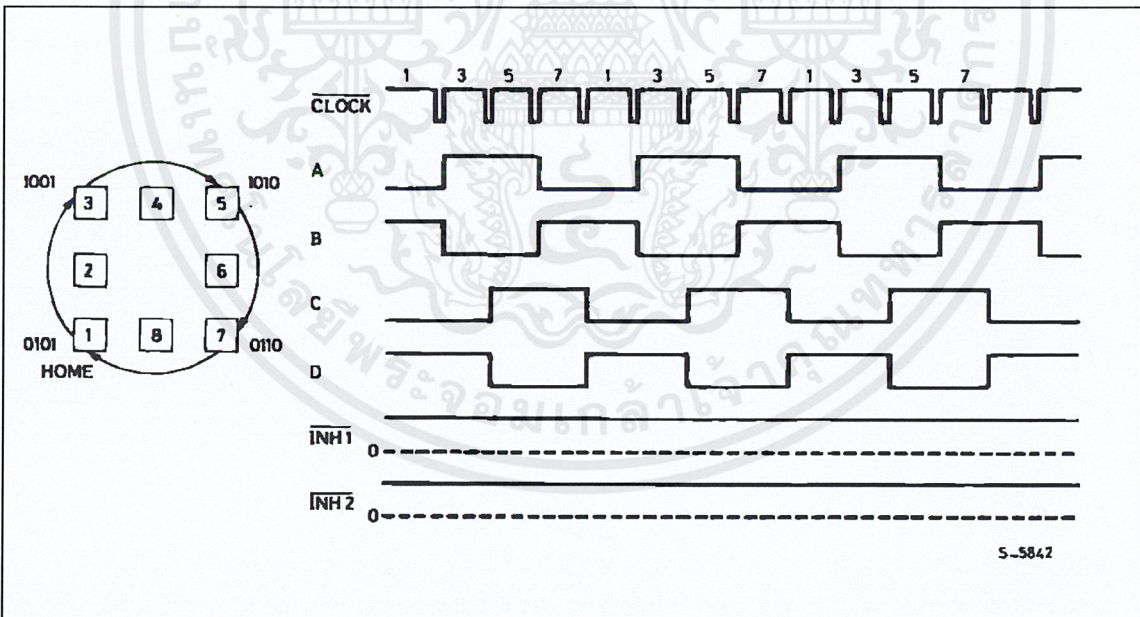
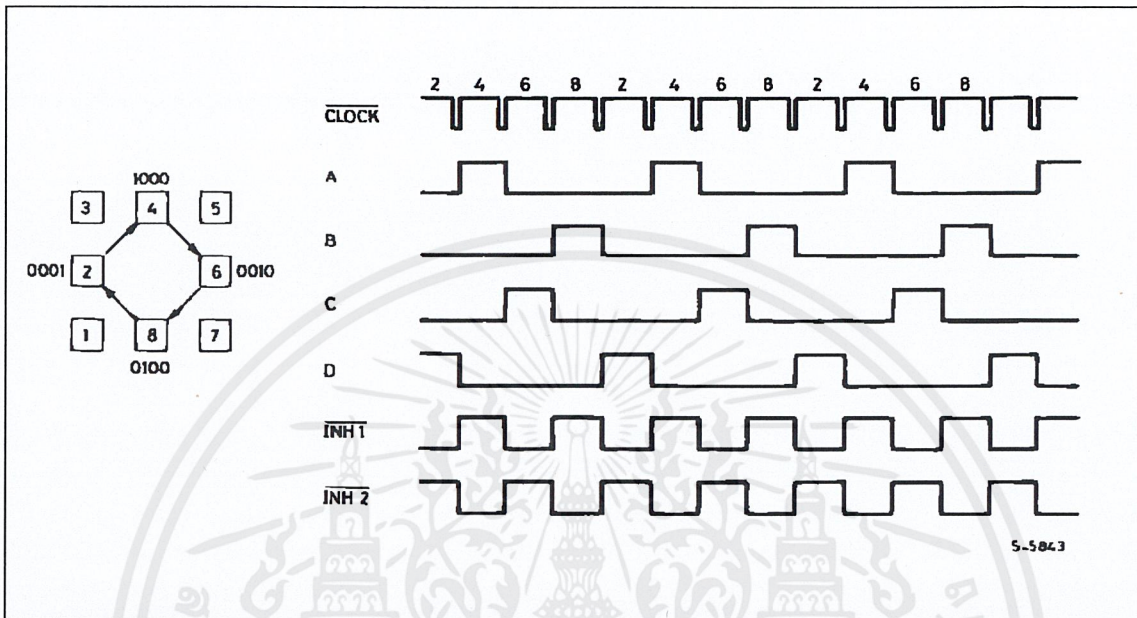


Figure 11 : State sequence and output waveforms for the two phase on sequence.  $\overline{INH1}$  and  $\overline{INH2}$  remain high throughout.



## APPLICATION NOTE

Figure 12 : State Sequence and Output Waveforms for Wave Drive (one phase on).



### $\overline{\text{INH1}}$ AND $\overline{\text{INH2}}$

In half step and one-phase-on full step modes two other signals are generated:  $\overline{\text{INH1}}$  and  $\overline{\text{INH2}}$ . These are inhibit signals which are coupled to the L298N's enable inputs and serve to speed the current decay when a winding is switched off.

Since both windings are energized continuously in two-phase-on full step mode no winding is ever switched off and these signals are not generated.

To see what these signals do let's look at one half of the L298N connected to the first phase of a two-phase bipolar motor (figure 13). Remember that the L298N's A and B inputs determine which transistor in each push pull pair will be on.  $\overline{\text{INH1}}$ , on the other hand, turns off all four transistors.

Assume that A is high, B low and current flowing through Q1, Q4 and the motor winding. If A is now brought low the current would recirculate through D2, Q4 and  $R_s$ , giving a slow decay and increased dissipation in  $R_s$ . If, on a other hand, A is brought low and  $\overline{\text{INH1}}$  is activated, all four transistors are turned off. The current recirculates in this case from ground to  $V_s$  via D2 and D3, giving a faster decay thus allowing faster operation of the motor. Also, since the recirculation current does not flow through  $R_s$ , a less expensive resistor can be used.

Exactly the same thing happens with the second winding, the other half of the L298 and the signals C, D and  $\overline{\text{INH2}}$ .

The  $\overline{\text{INH1}}$  and  $\overline{\text{INH2}}$  signals are generated by OR functions:

$$A + B = \overline{\text{INH1}} \quad C + D = \overline{\text{INH2}}$$

However, the output logic is more complex because inhibit lines are also used by the chopper, as we will see further on.

### OTHER SIGNALS

Two other signals are connected to the translator block: the RESET input and the HOME output

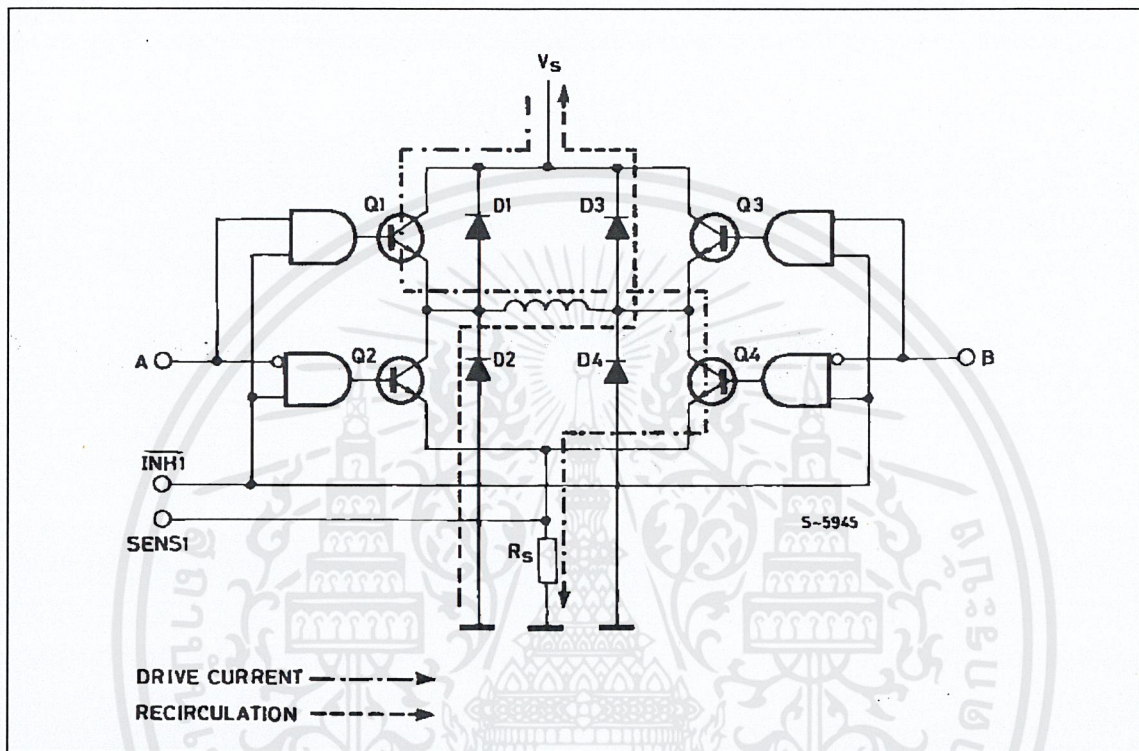
RESET is an asynchronous reset input which restores the translator block to the home position (state 1, ABCD = 0101). The HOME output (open collector) signals this condition and is intended to the AN-Ded with the output of a mechanical home position sensor.

Finally, there is an ENABLE input connected to the output logic. A low level on this input brings  $\overline{\text{INH1}}$ ,  $\overline{\text{INH2}}$ , A, B, C and D low. This input is useful to disable the motor driver when the system is initialized.

### LOAD CURRENT REGULATION

Some form of load current control is essential to obtain good speed and torque characteristics. There are several ways in which this can be done – switching the supply between two voltages, pulse rate modulation chopping or pulse width modulation chopping.

**Figure 13 :** When a winding is switched off the inhibit input is activated to speed current decay. If this were not done the current would recirculate through D2 and Q4 in this example. Dissipation in  $R_s$  is also reduced.

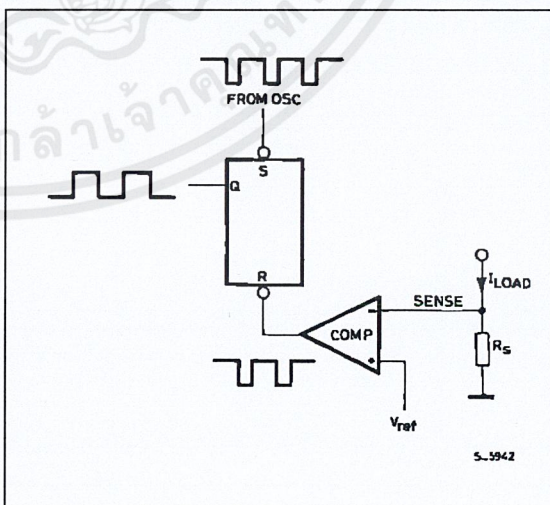


The L297 provides load current control in the form of two PWM choppers, one for each phase of a bipolar motor or one for each pair of windings for a unipolar motor. (In a unipolar motor the A and B windings are never energized together so they can share a chopper ; the same applies to C and D).

Each chopper consists of a comparator, a flip flop and an external sensing resistor. A common on chip oscillator supplies pulses at the chopper rate to both choppers.

In each chopper (figure 14) the flip flop is set by each pulse from the oscillator, enabling the output and allowing the load current to increase. As it increases the voltage across the sensing resistor increases, and when this voltage reaches  $V_{ref}$  the flip flop is reset, disabling the output until the next oscillator pulse arrives. The output of this circuit (the flip flop's Q output) is therefore a constant rate PWM signal. Note that  $V_{ref}$  determines the peak load current.

**Figure 14 :** Each chopper circuit consists of a comparator, flip flop and external sense resistor. A common oscillator clocks both circuits.



## APPLICATION NOTE

### PHASE CHOPPING AND INHIBIT CHOPPING

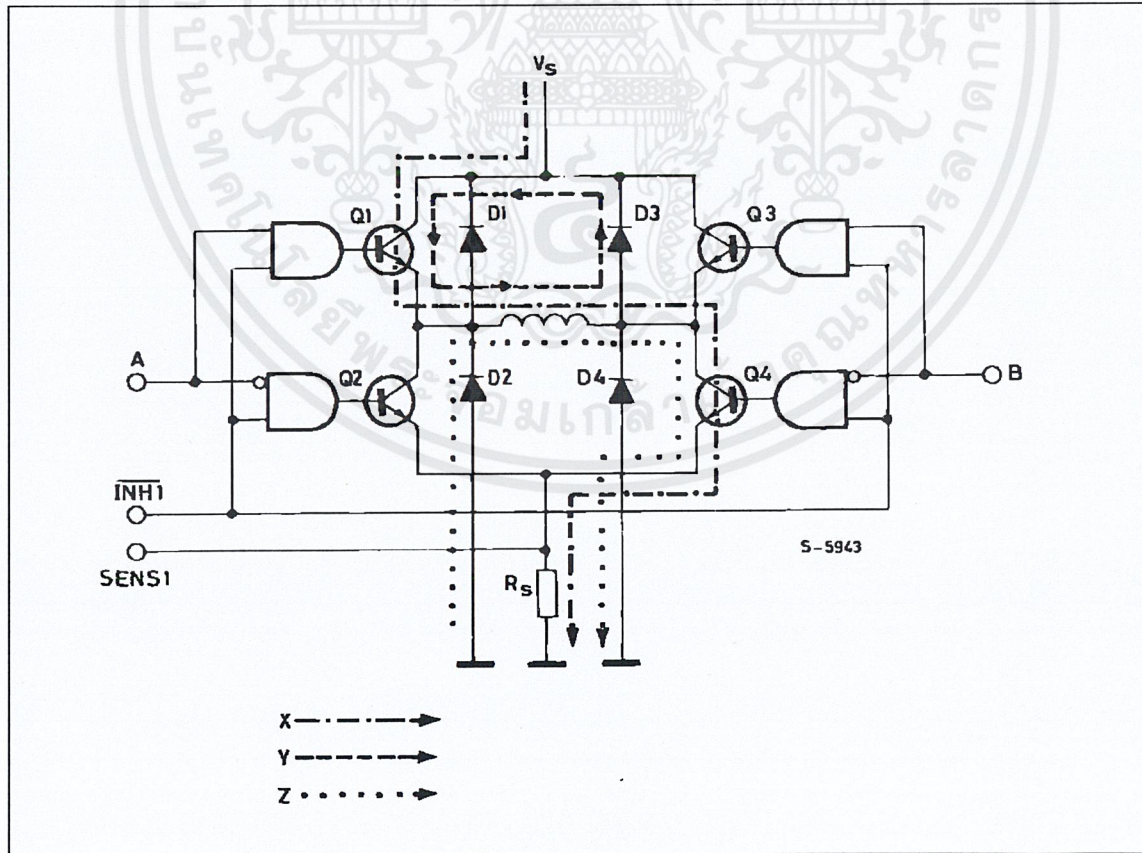
The chopper can act on either the phase lines (ABCD) or on the inhibit lines INH1 and INH2. An input named CONTROL decides which. Inhibit chopping is used for unipolar motors but you can choose between phase chopping and inhibit chopping for bipolar motors. The reasons for this choice are best explained with another example.

First let's examine the situation when the phase lines are chopped.

As before, we are driving a two phase bipolar motor and A is high, B low (figure 15). Current therefore flows through Q1, winding, Q4 and  $R_s$ . When the voltage across  $R_s$  reaches  $V_{ref}$  the chopper brings B high to switch off the winding.

The energy stored in the winding is dissipated by current recirculating through Q1 and D3. Current decay through this path is rather slow because the volt-

**Figure 15** : Phase Chopping. In this example the current X is interrupted by activating B, giving the recirculation path Y. The alternative, de-activating A, would give the recirculation path Z, increasing dissipation in  $R_s$ .



age on the winding is low ( $V_{CEsat Q1} + V_{D3}$ ) (figure 16).

Why is B pulled high, why push A low? The reason is to avoid the current decaying through  $R_s$ . Since the current recirculates in the upper half of the bridge, current only flows in the sensing resistor when the winding is driven. Less power is therefore dissipated in  $R_s$  and we can get away with a cheaper resistor.

This explains why phase chopping is not suitable for unipolar motors: when the A winding is driven the chopper acts on the B winding. Clearly, this is no use at all for a variable reluctance motor and would be slow and inefficient for a bifilar wound permanent magnet motor.

The alternative is to tie the CONTROL input to ground so that the chopper acts on INH1 and INH2. Looking at the same example, A is high and B low. Q1 and Q4 are therefore conducting and current flows through Q1, the winding, Q4 and  $R_s$ , (figure 17).

Figure 16 : Phase Chopping Waveforms. The example shows AB winding energized with A positive with respect to B. Control is high.

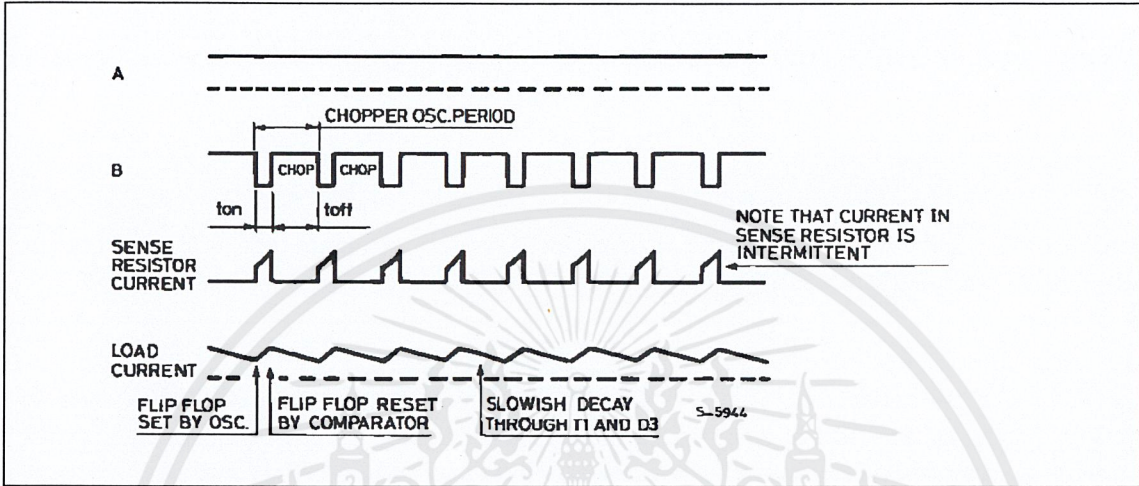
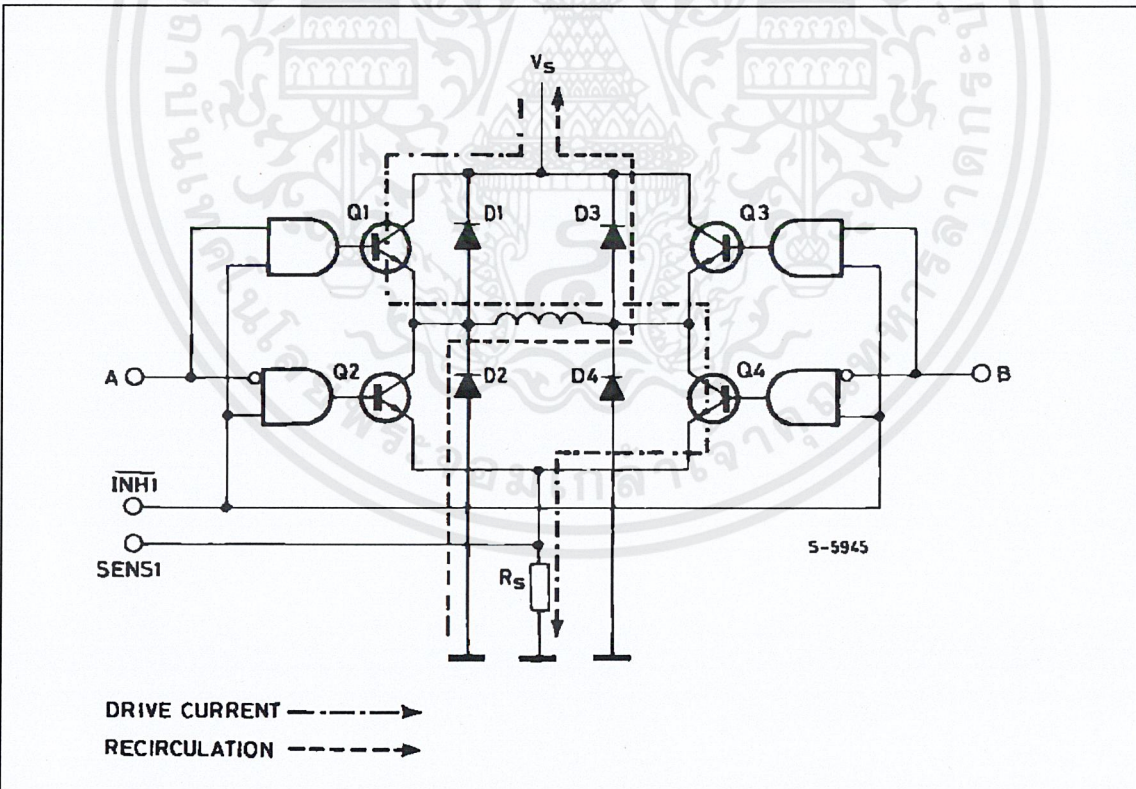


Figure 17 : Inhibit Chopping. The drive current (Q1, winding, Q4) in this case is interrupted by activating INH1. The decay path through D2 and D3 is faster than the path Y of Figure 15.



## APPLICATION NOTE

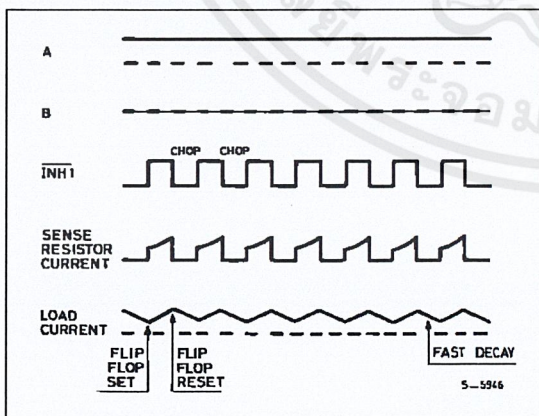
In this case when the voltage across  $R_S$  reaches  $V_{REF}$  the chopper flip flop is reset and  $INH1$  activated (brought low).  $INH1$ , remember, turns off all four transistors therefore the current recirculates from ground, through D2, the winding and D3 to  $V_S$ . Discharged across the supply, which can be up to 46V, the current decays very rapidly (figure 18).

The usefulness of this second faster decay option is fairly obvious ; it allows fast operation with bipolar motors and it is the only choice for unipolar motors. But why do we offer the slower alternative, phase chopping ?

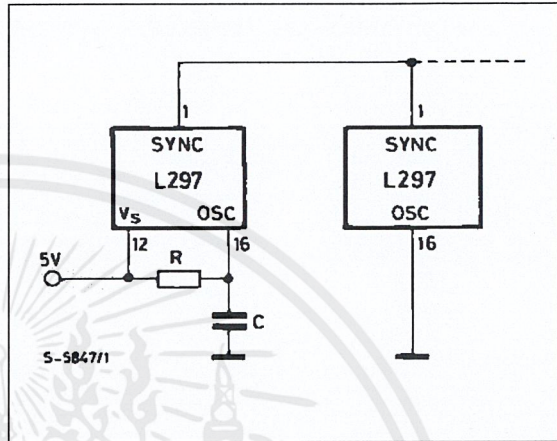
The answer is that we might be obliged to use a low chopper rate with a motor that does not store much energy in the windings. If the decay is very fast the average motor current may be too low to give an useful torque. Low chopper rates may, for example, be imposed if there is a larger motor in the same system. To avoid switching noise on the ground plane all drivers should be synchronized and the chopper rate is therefore determined by the largest motor in the system.

Multiple L297s are synchronized easily using the SYNC pin. This pin is the squarewave output of the on-chip oscillator and the clock input for the choppers. The first L297 is fitted with the oscillator components and outputs a squarewave signal on this pin (figure 19). Subsequent L297s do not need the oscillator components and use SYNC as a clock input. An external clock may also be injected at this terminal if an L297 must be synchronized to other system components.

**Figure 18 :** Inhibit Chopper Waveforms. Winding AB is energized and CONTROL is low.



**Figure 19 :** The Chopper oscillator of multiple L297s are synchronized by connecting the SYNC Inputs together.



### THE L297A

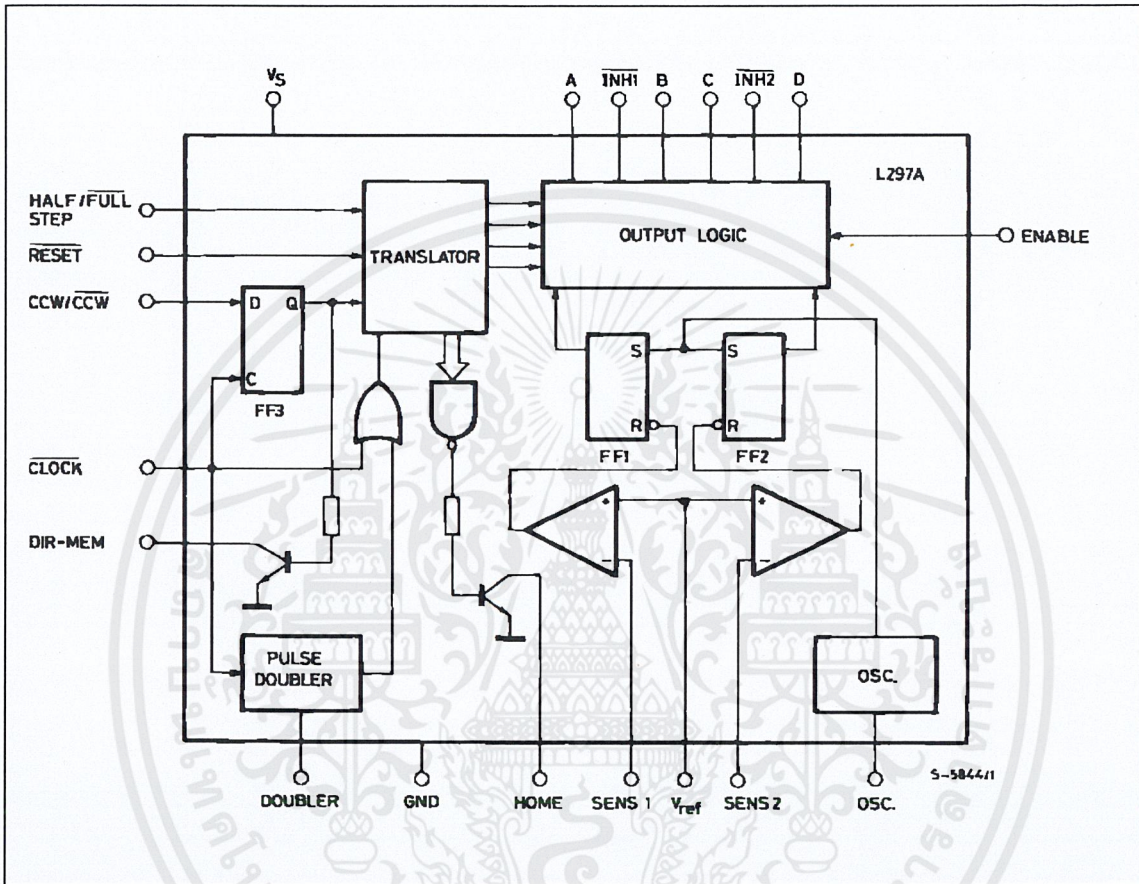
The L297A is a special version of the L297 developed originally for head positioning in floppy disk drives. It can, however, be used in other applications.

Compared to the standard L297 the difference are the addition of a pulse doubler on the step clock input and the availability of the output of the direction flip flop (block diagram, figure 20). To add these functions while keeping the low-cost 20-pin package the CONTROL and SYNC pins are not available on this version (they are not needed anyway). The chopper acts on the ABCD phase lines.

The pulse doubler generates a ghost pulse internally for each input clock pulse. Consequently the translator moves two steps for each input pulse. An external RC network sets the delay time between the input pulse and ghost pulse and should be chosen so that the ghost pulses fall roughly halfway between input pulses, allowing time for the motor to step.

This feature is used to improve positioning accuracy. Since the angular position error of a stepper motor is noncumulative (it cancels out to zero every four steps in a four step sequence motor) accuracy is improved by stepping two of four steps at a time.

Figure 20 : The L297A, includes a clock pulse doubler and provides an output from the direction flip flop (DIR - MEM).



**APPLICATION HINTS**

Bipolar motors can be driven with an L297, an L298N or L293E bridge driver and very few external components (figure 21). Together these two chips form a complete microprocessor-to-stepper motor interface. With an L298N this configuration drives motors with winding currents up to 2A ; for motors up to 1A per winding and L293E is used. If the PWM choppers are not required an L293 could also be used (it doesn't have the external emitter connections for sensing resistors) but the L297 is underutilized. If very high powers are required the bridge driver is replaced by an equivalent circuit made with discrete transistors. For currents to 3.5A two L298N's with paralleled outputs may be used.

For unipolar motors the best choice is a quad darlington array. The L702B can be used if the choppers are not required but an ULN2075B is preferred.

This quad darlington has external emitter connections which are connected to sensing resistors (figure 22). Since the chopper acts on the inhibit lines, four AND gates must be added in this application.

Also shown in the schematic are the protection diodes.

In all applications where the choppers are not used it is important to remember that the sense inputs must be grounded and VREF connected either to VS or any potential between VS and ground.

The chopper oscillator frequency is determined by the RC network on pin 16. The frequency is roughly  $1/0.7 RC$  and R must be more than 10 kΩ. When the L297A's pulse doubler is used, the delay time is determined by the network  $R_d C_d$  and is approximately  $0.75 R_d C_d$ .  $R_d$  should be in the range 3 kΩ to 100 kΩ (figure 23).

## APPLICATION NOTE

**Figure 21 :** This typical application shows an L297 and L298N driving a Bipolar Stepper Motor with phase currents up to 2A.

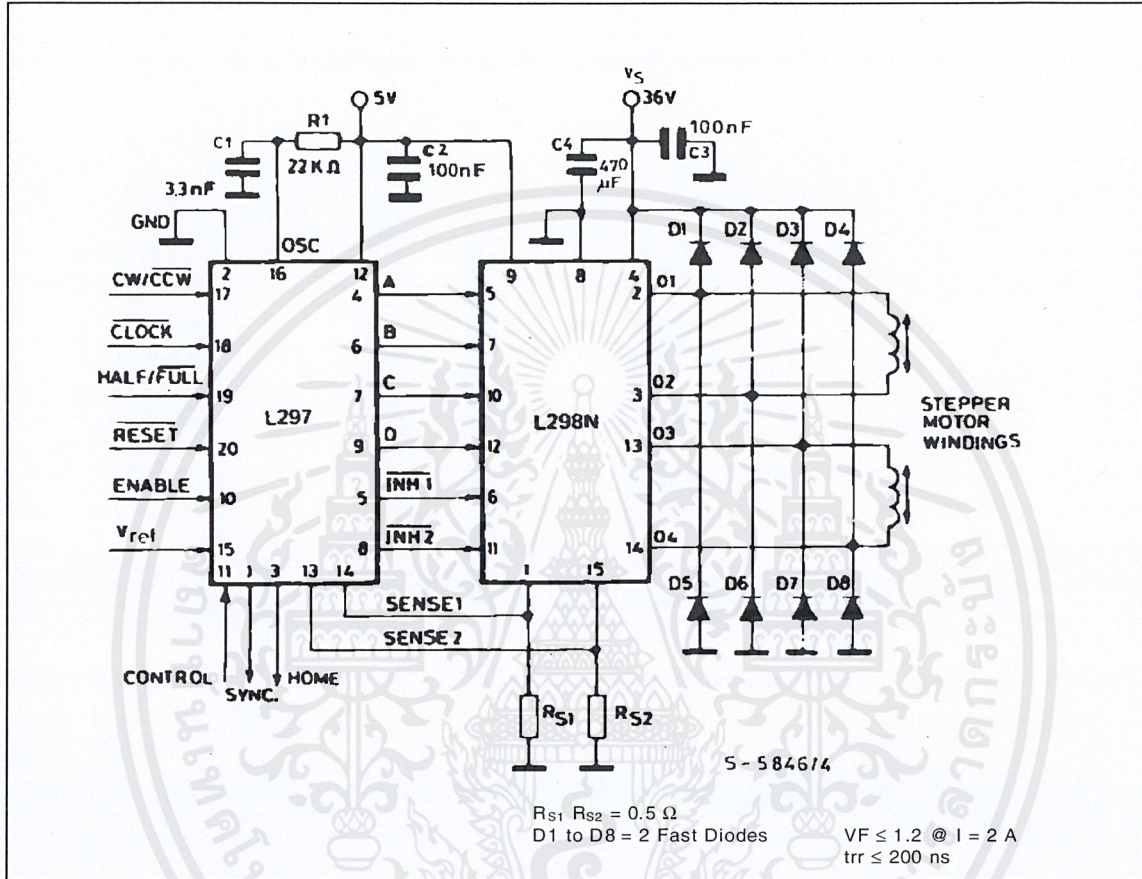


Figure 22 : For Unipolar Motors a Quad Darlington Array is coupled to the L297. Inhibit chopping is used so the four AND gates must be added.

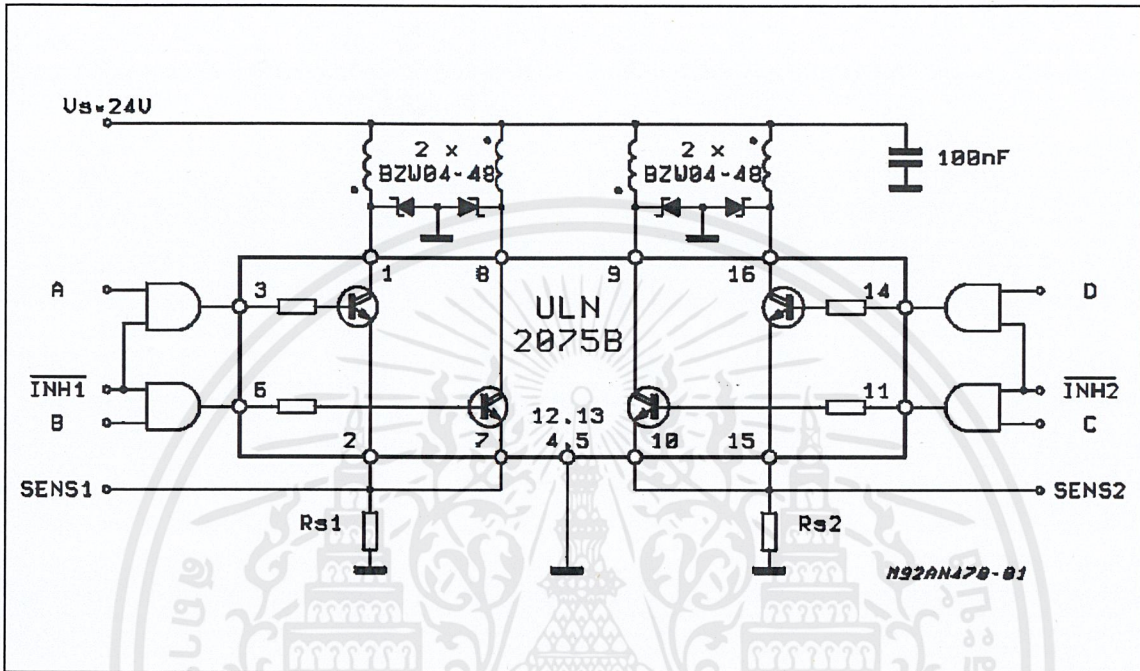
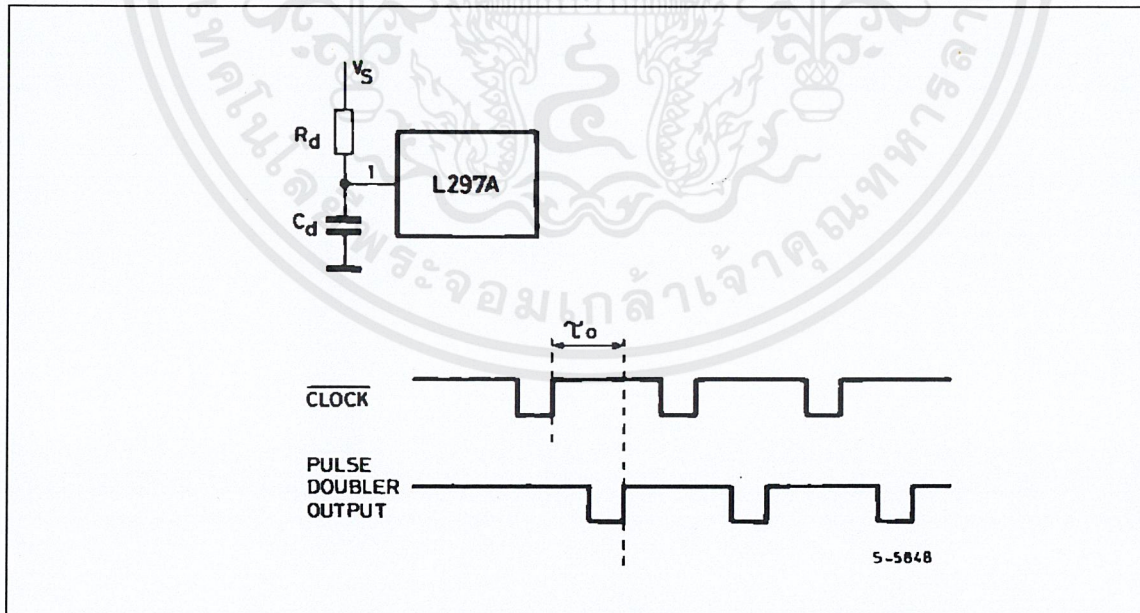


Figure 23 : The Clock pulse doubler inserts a ghost pulse  $\tau_0$  seconds after the Input clock pulse.  $R_d C_d$  is chosen to give a delay of approximately half the Input clock period.



## APPLICATION NOTE

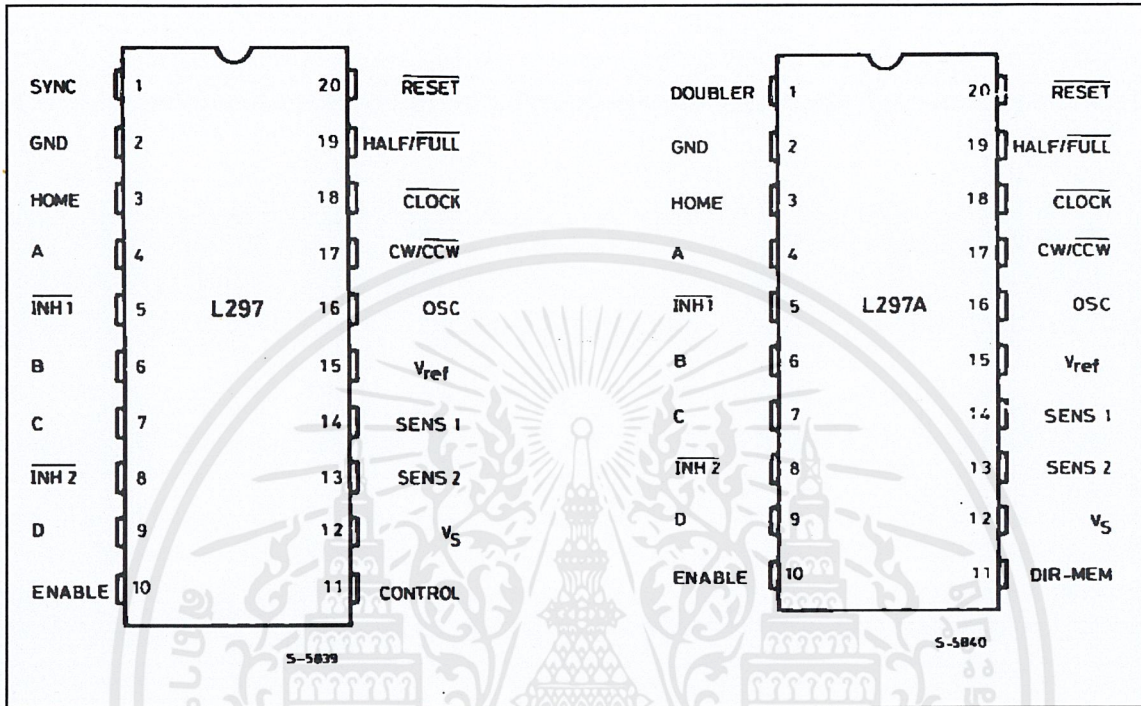
### PIN FUNCTIONS - L297

N°	NAME	FUNCTION
1	SYNC	Output of the on-chip chopper oscillator. The SYNC connections of all L297s to be synchronized are connected together and the oscillator components are omitted on all but one. If an external clock source is used it is injected at this terminal.
2	GND	Ground connection.
3	HOME	Open collector output that indicates when the L297 is in its initial state (ABCD = 0101). The transistor is open when this signal is active.
4	A	Motor phase A drive signal for power stage.
5	INH1	Active low inhibit control for driver stage of A and B phases. When a bipolar bridge is used this signal can be used to ensure fast decay of load current when a winding is de-energized. Also used by chopper to regulate load current if CONTROL input is low.
6	B	Motor phase B drive signal for power stage.
7	C	Motor phase C drive signal for power stage.
8	INH2	Active low inhibit control for drive stages of C and D phases. Same functions as INH1.
9	D	Motor phase D drive signal for power stage.
10	ENABLE	Chip enable input. When low (inactive) INH1, INH2, A, B, C and D are brought low.
11	CONTROL	Control input that defines action of chopper. When low chopper acts on INH1 and INH2; when high chopper acts on phase lines ABCD.
12	V <sub>s</sub>	5V supply input.
13	SENS <sub>2</sub>	Input for load current sense voltage from power stages of phases C and D.
14	SENS <sub>1</sub>	Input for load current sense voltage from power stages of phases A and B.
15	V <sub>ref</sub>	Reference voltage for chopper circuit. A voltage applied to this pin determines the peak load current.
16	OSC	An RC network (R to V <sub>cc</sub> , C to ground) connected to this terminal determines the chopper rate. This terminal is connected to ground on all but one device in synchronized multi - L297 configurations. $f \approx 1/0.69 RC$
17	CW/CCW	Clockwise/counterclockwise direction control input. Physical direction of motor rotation also depends on connection of windings. Synchronized internally therefore direction can be changed at any time.
18	CLOCK	Step clock. An active low pulse on this input advances the motor one increment. The step occurs on the rising edge of this signal.
19	HALF/FULL	Half/full step select input. When high selects half step operation, when low selects full step operation. One-phase-on full step mode is obtained by selecting FULL when the L297's translator is at an even-numbered state. Two-phase-on full step mode is set by selecting FULL when the translator is at an odd numbered position. (The home position is designate state 1).
20	RESET	Reset input. An active low pulse on this input restores the translator to the home position (state 1, ABCD = 0101).

### PIN FUNCTIONS - L297A (Pin function of the L297A are identical to those of the L297 except for pins 1 and 11)

1	DOUBLER	An RC network connected to this pin determines the delay between an input clock pulse and the corresponding ghost pulse.
11	DIR-MEM	Direction Memory. Inverted output of the direction flip flop. Open collector output.

Figure 24 : Pin connections.



S-5039

S-5040

## APPLICATION NOTE

---



Information furnished is believed to be accurate and reliable. However, SGS-THOMSON Microelectronics assumes no responsibility for the consequences of use of such information nor for any infringement of patents or other rights of third parties which may result from its use. No license is granted by implication or otherwise under any patent or patent rights of SGS-THOMSON Microelectronics. Specifications mentioned in this publication are subject to change without notice. This publication supersedes and replaces all information previously supplied. SGS-THOMSON Microelectronics products are not authorized for use as critical components in life support devices or systems without express written approval of SGS-THOMSON Microelectronics.

© 1995 SGS-THOMSON Microelectronics - All Rights Reserved

SGS-THOMSON Microelectronics GROUP OF COMPANIES

Australia - Brazil - France - Germany - Hong Kong - Italy - Japan - Korea - Malaysia - Malta - Morocco - The Netherlands - Singapore - Spain - Sweden - Switzerland - Taiwan - Thailand - United Kingdom - U.S.A.



**ภาคผนวก ข:**  
**คำตัดชี้ทของไอซี L298**

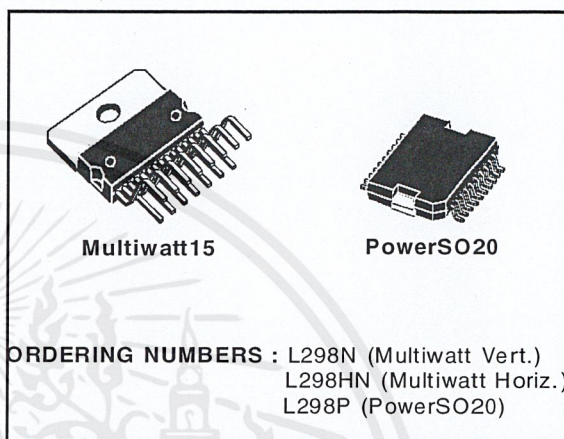
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## DUAL FULL-BRIDGE DRIVER

- OPERATING SUPPLY VOLTAGE UP TO 46 V
- TOTAL DC CURRENT UP TO 4 A
- LOW SATURATION VOLTAGE
- OVERTEMPERATURE PROTECTION
- LOGICAL "0" INPUT VOLTAGE UP TO 1.5 V (HIGH NOISE IMMUNITY)

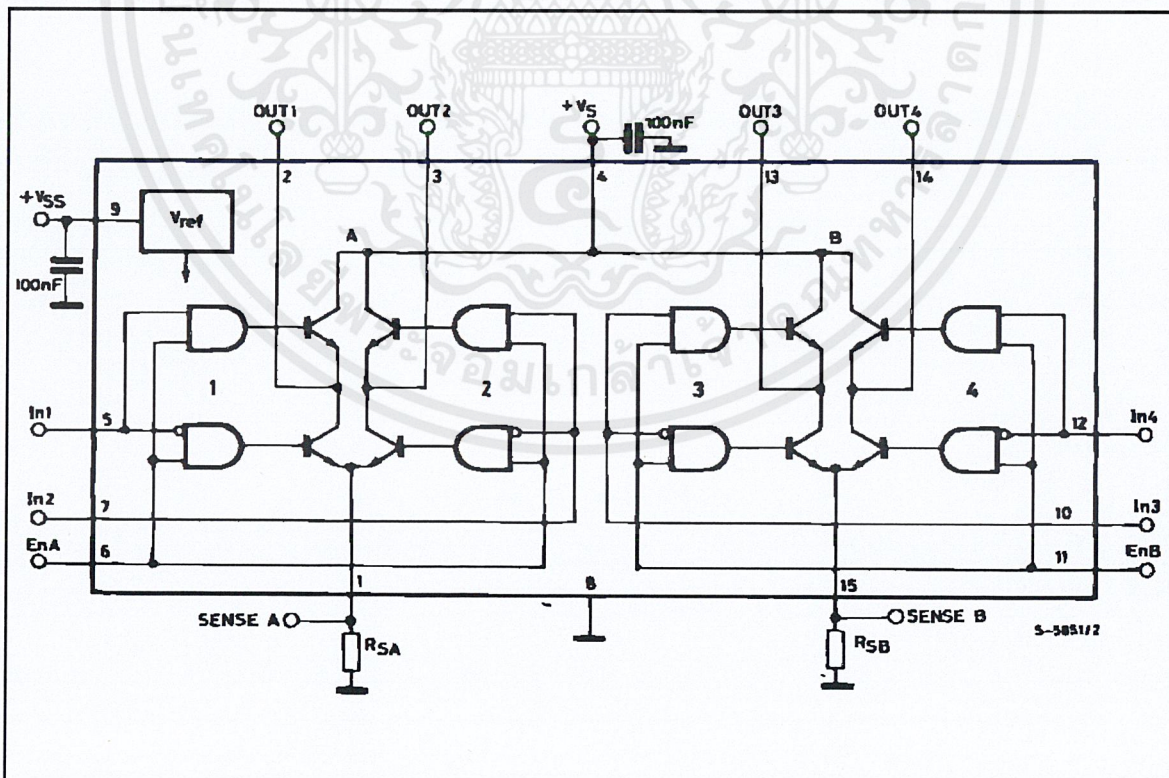
### DESCRIPTION

The L298 is an integrated monolithic circuit in a 15-lead Multiwatt and PowerSO20 packages. It is a high voltage, high current dual full-bridge driver designed to accept standard TTL logic levels and drive inductive loads such as relays, solenoids, DC and stepping motors. Two enable inputs are provided to enable or disable the device independently of the input signals. The emitters of the lower transistors of each bridge are connected together and the corresponding external terminal can be used for the con-



nection of an external sensing resistor. An additional supply input is provided so that the logic works at a lower voltage.

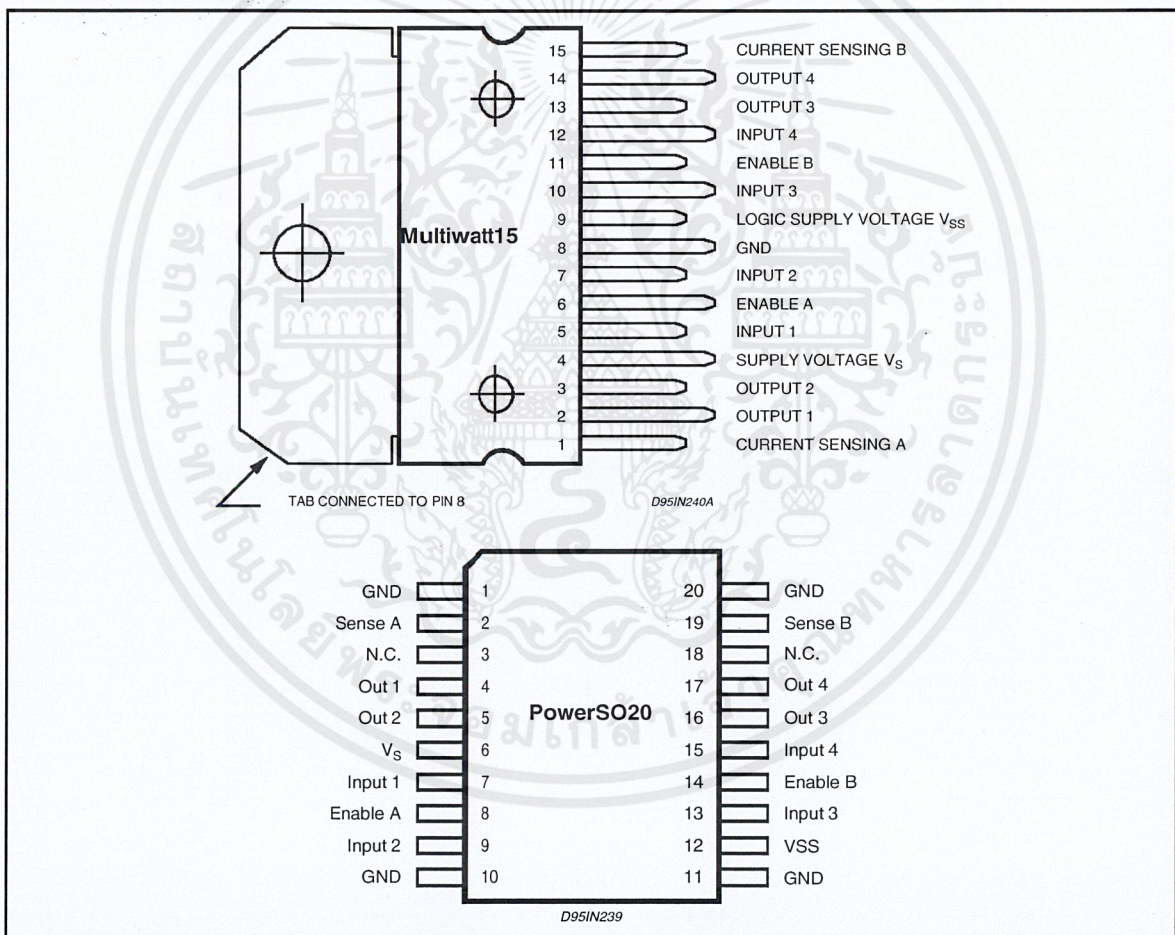
### BLOCK DIAGRAM



## ABSOLUTE MAXIMUM RATINGS

Symbol	Parameter	Value	Unit
$V_S$	Power Supply	50	V
$V_{SS}$	Logic Supply Voltage	7	V
$V_I, V_{en}$	Input and Enable Voltage	-0.3 to 7	V
$I_O$	Peak Output Current (each Channel)		
	– Non Repetitive ( $t = 100\mu s$ )	3	A
	– Repetitive (80% on -20% off; $t_{on} = 10ms$ )	2.5	A
	– DC Operation	2	A
$V_{sens}$	Sensing Voltage	-1 to 2.3	V
$P_{tot}$	Total Power Dissipation ( $T_{case} = 75^\circ C$ )	25	W
$T_{op}$	Junction Operating Temperature	-25 to 130	$^\circ C$
$T_{stg}, T_j$	Storage and Junction Temperature	-40 to 150	$^\circ C$

## PIN CONNECTIONS (top view)



## THERMAL DATA

Symbol	Parameter		PowerSO20	Multiwatt15	Unit
$R_{th\ j-case}$	Thermal Resistance Junction-case	Max.	–	3	$^\circ C/W$
$R_{th\ j-amb}$	Thermal Resistance Junction-ambient	Max.	13 (*)	35	$^\circ C/W$

(\*) Mounted on aluminum substrate

## PIN FUNCTIONS (refer to the block diagram)

MW.15	PowerSO	Name	Function
1;15	2;19	Sense A; Sense B	Between this pin and ground is connected the sense resistor to control the current of the load.
2;3	4;5	Out 1; Out 2	Outputs of the Bridge A; the current that flows through the load connected between these two pins is monitored at pin 1.
4	6	V <sub>S</sub>	Supply Voltage for the Power Output Stages. A non-inductive 100nF capacitor must be connected between this pin and ground.
5;7	7;9	Input 1; Input 2	TTL Compatible Inputs of the Bridge A.
6;11	8;14	Enable A; Enable B	TTL Compatible Enable Input: the L state disables the bridge A (enable A) and/or the bridge B (enable B).
8	1,10,11,20	GND	Ground.
9	12	V <sub>SS</sub>	Supply Voltage for the Logic Blocks. A100nF capacitor must be connected between this pin and ground.
10; 12	13;15	Input 3; Input 4	TTL Compatible Inputs of the Bridge B.
13; 14	16;17	Out 3; Out 4	Outputs of the Bridge B. The current that flows through the load connected between these two pins is monitored at pin 15.
–	3;18	N.C.	Not Connected

ELECTRICAL CHARACTERISTICS (V<sub>S</sub> = 42V; V<sub>SS</sub> = 5V, T<sub>J</sub> = 25°C; unless otherwise specified)

Symbol	Parameter	Test Conditions	Min.	Typ.	Max.	Unit
V <sub>S</sub>	Supply Voltage (pin 4)	Operative Condition	V <sub>IH</sub> +2.5		46	V
V <sub>SS</sub>	Logic Supply Voltage (pin 9)		4.5	5	7	V
I <sub>S</sub>	Quiescent Supply Current (pin 4)	V <sub>en</sub> = H; I <sub>L</sub> = 0 V <sub>i</sub> = L V <sub>i</sub> = H		13 50	22 70	mA mA
		V <sub>en</sub> = L V <sub>i</sub> = X			4	mA
I <sub>SS</sub>	Quiescent Current from V <sub>SS</sub> (pin 9)	V <sub>en</sub> = H; I <sub>L</sub> = 0 V <sub>i</sub> = L V <sub>i</sub> = H		24 7	36 12	mA mA
		V <sub>en</sub> = L V <sub>i</sub> = X			6	mA
V <sub>iL</sub>	Input Low Voltage (pins 5, 7, 10, 12)		–0.3		1.5	V
V <sub>iH</sub>	Input High Voltage (pins 5, 7, 10, 12)		2.3		V <sub>SS</sub>	V
I <sub>iL</sub>	Low Voltage Input Current (pins 5, 7, 10, 12)	V <sub>i</sub> = L			–10	μA
I <sub>iH</sub>	High Voltage Input Current (pins 5, 7, 10, 12)	V <sub>i</sub> = H ≤ V <sub>SS</sub> –0.6V		30	100	μA
V <sub>en</sub> = L	Enable Low Voltage (pins 6, 11)		–0.3		1.5	V
V <sub>en</sub> = H	Enable High Voltage (pins 6, 11)		2.3		V <sub>SS</sub>	V
I <sub>en</sub> = L	Low Voltage Enable Current (pins 6, 11)	V <sub>en</sub> = L			–10	μA
I <sub>en</sub> = H	High Voltage Enable Current (pins 6, 11)	V <sub>en</sub> = H ≤ V <sub>SS</sub> –0.6V		30	100	μA
V <sub>CEsat</sub> (H)	Source Saturation Voltage	I <sub>L</sub> = 1A I <sub>L</sub> = 2A	0.95	1.35 2	1.7 2.7	V V
V <sub>CEsat</sub> (L)	Sink Saturation Voltage	I <sub>L</sub> = 1A (5) I <sub>L</sub> = 2A (5)	0.85	1.2 1.7	1.6 2.3	V V
V <sub>CEsat</sub>	Total Drop	I <sub>L</sub> = 1A (5) I <sub>L</sub> = 2A (5)	1.80		3.2 4.9	V V
V <sub>sens</sub>	Sensing Voltage (pins 1, 15)		–1 (1)		2	V

ELECTRICAL CHARACTERISTICS (continued)

Symbol	Parameter	Test Conditions	Min.	Typ.	Max.	Unit
T <sub>1</sub> (V <sub>i</sub> )	Source Current Turn-off Delay	0.5 V <sub>i</sub> to 0.9 I <sub>L</sub> (2); (4)		1.5		μs
T <sub>2</sub> (V <sub>i</sub> )	Source Current Fall Time	0.9 I <sub>L</sub> to 0.1 I <sub>L</sub> (2); (4)		0.2		μs
T <sub>3</sub> (V <sub>i</sub> )	Source Current Turn-on Delay	0.5 V <sub>i</sub> to 0.1 I <sub>L</sub> (2); (4)		2		μs
T <sub>4</sub> (V <sub>i</sub> )	Source Current Rise Time	0.1 I <sub>L</sub> to 0.9 I <sub>L</sub> (2); (4)		0.7		μs
T <sub>5</sub> (V <sub>i</sub> )	Sink Current Turn-off Delay	0.5 V <sub>i</sub> to 0.9 I <sub>L</sub> (3); (4)		0.7		μs
T <sub>6</sub> (V <sub>i</sub> )	Sink Current Fall Time	0.9 I <sub>L</sub> to 0.1 I <sub>L</sub> (3); (4)		0.25		μs
T <sub>7</sub> (V <sub>i</sub> )	Sink Current Turn-on Delay	0.5 V <sub>i</sub> to 0.9 I <sub>L</sub> (3); (4)		1.6		μs
T <sub>8</sub> (V <sub>i</sub> )	Sink Current Rise Time	0.1 I <sub>L</sub> to 0.9 I <sub>L</sub> (3); (4)		0.2		μs
f <sub>c</sub> (V <sub>i</sub> )	Commutation Frequency	I <sub>L</sub> = 2A		25	40	KHz
T <sub>1</sub> (V <sub>en</sub> )	Source Current Turn-off Delay	0.5 V <sub>en</sub> to 0.9 I <sub>L</sub> (2); (4)		3		μs
T <sub>2</sub> (V <sub>en</sub> )	Source Current Fall Time	0.9 I <sub>L</sub> to 0.1 I <sub>L</sub> (2); (4)		1		μs
T <sub>3</sub> (V <sub>en</sub> )	Source Current Turn-on Delay	0.5 V <sub>en</sub> to 0.1 I <sub>L</sub> (2); (4)		0.3		μs
T <sub>4</sub> (V <sub>en</sub> )	Source Current Rise Time	0.1 I <sub>L</sub> to 0.9 I <sub>L</sub> (2); (4)		0.4		μs
T <sub>5</sub> (V <sub>en</sub> )	Sink Current Turn-off Delay	0.5 V <sub>en</sub> to 0.9 I <sub>L</sub> (3); (4)		2.2		μs
T <sub>6</sub> (V <sub>en</sub> )	Sink Current Fall Time	0.9 I <sub>L</sub> to 0.1 I <sub>L</sub> (3); (4)		0.35		μs
T <sub>7</sub> (V <sub>en</sub> )	Sink Current Turn-on Delay	0.5 V <sub>en</sub> to 0.9 I <sub>L</sub> (3); (4)		0.25		μs
T <sub>8</sub> (V <sub>en</sub> )	Sink Current Rise Time	0.1 I <sub>L</sub> to 0.9 I <sub>L</sub> (3); (4)		0.1		μs

- 1) Sensing voltage can be -1 V for t ≤ 50 μsec; in steady state V<sub>sens</sub> min ≥ -0.5 V.
- 2) See fig. 2.
- 3) See fig. 4.
- 4) The load must be a pure resistor.

Figure 1 : Typical Saturation Voltage vs. Output Current.

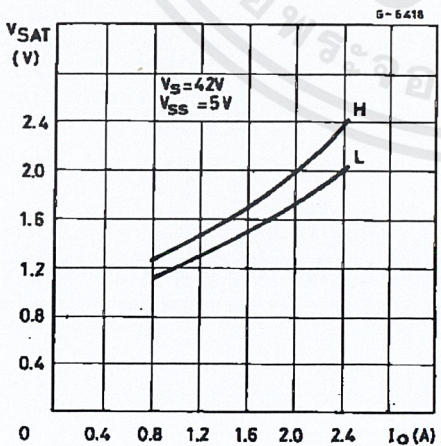
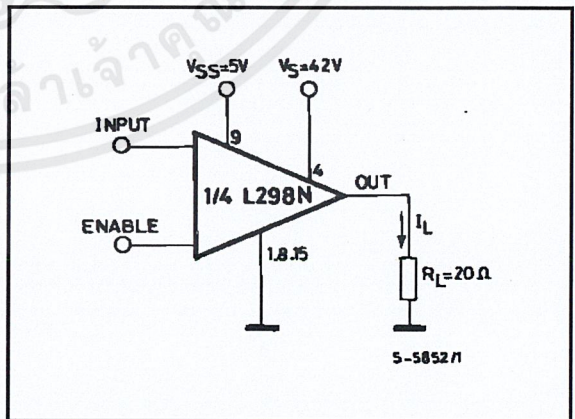


Figure 2 : Switching Times Test Circuits.



Note: For INPUT Switching, set EN = H  
For ENABLE Switching, set IN = H



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Figure 3 : Source Current Delay Times vs. Input or Enable Switching.

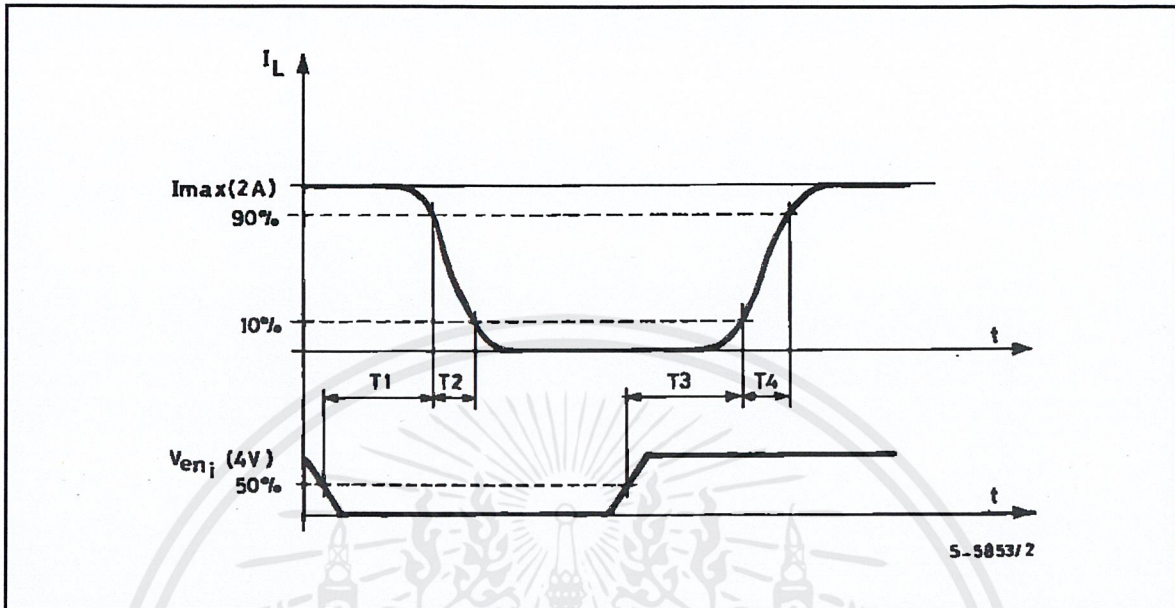
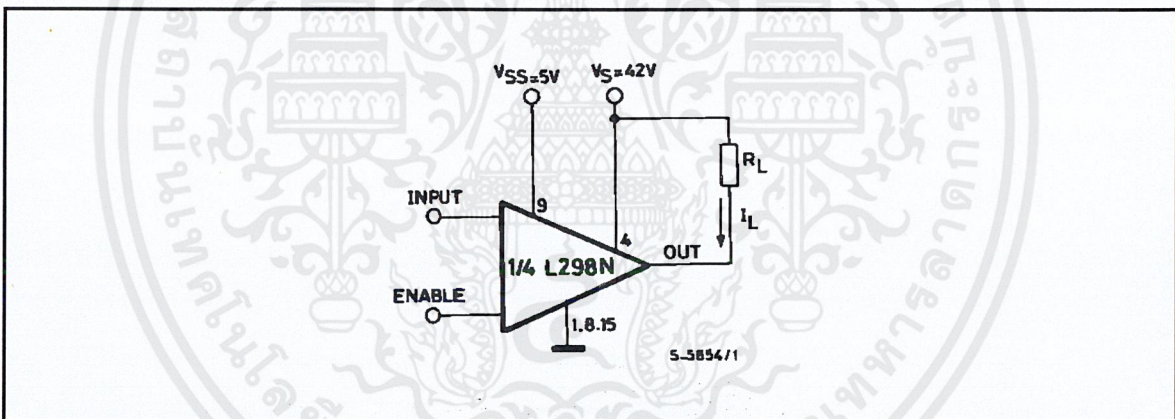


Figure 4 : Switching Times Test Circuits.



Note : For INPUT Switching, set EN = H  
 For ENABLE Switching, set IN = L

Figure 5 : Sink Current Delay Times vs. Input 0 V Enable Switching.

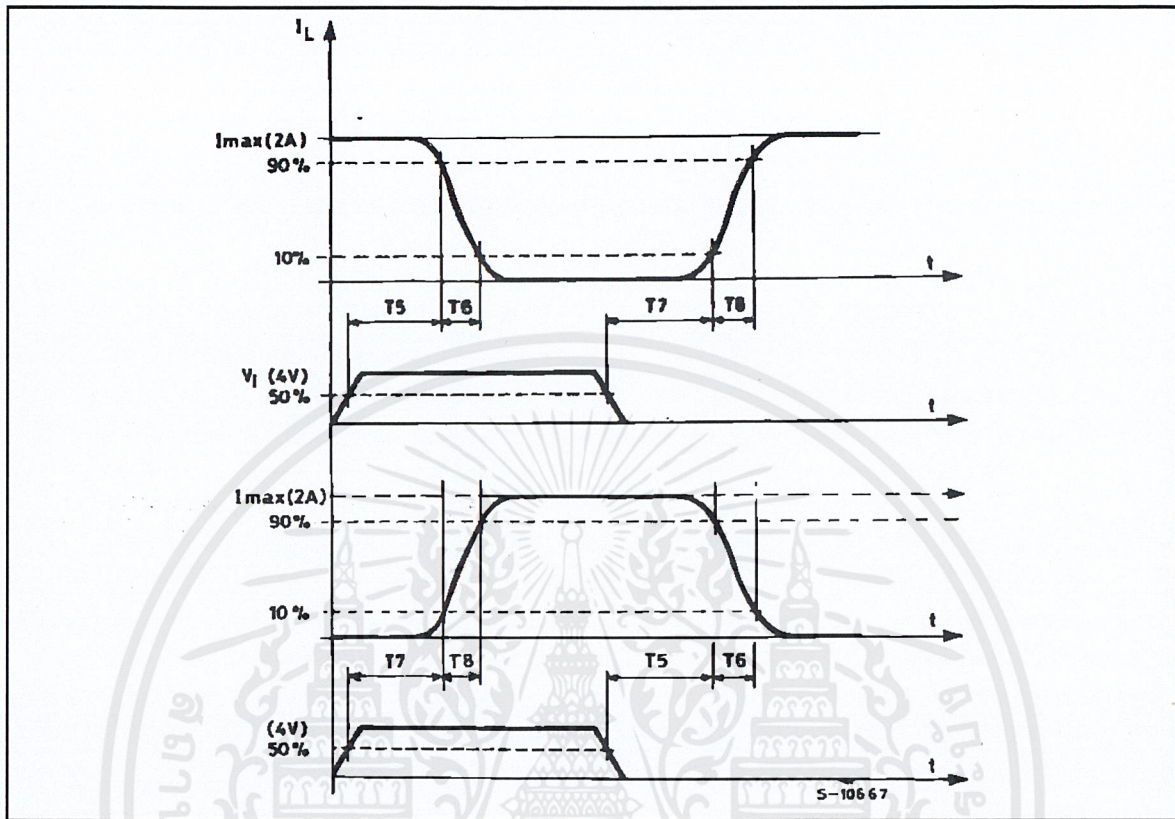
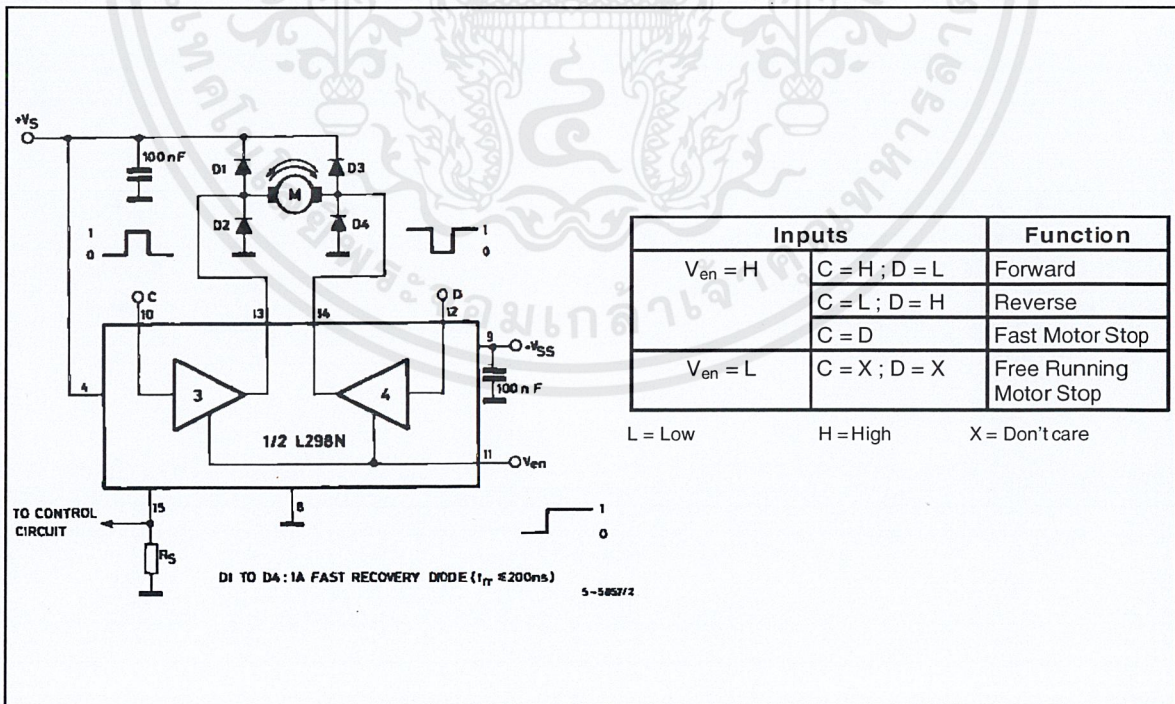
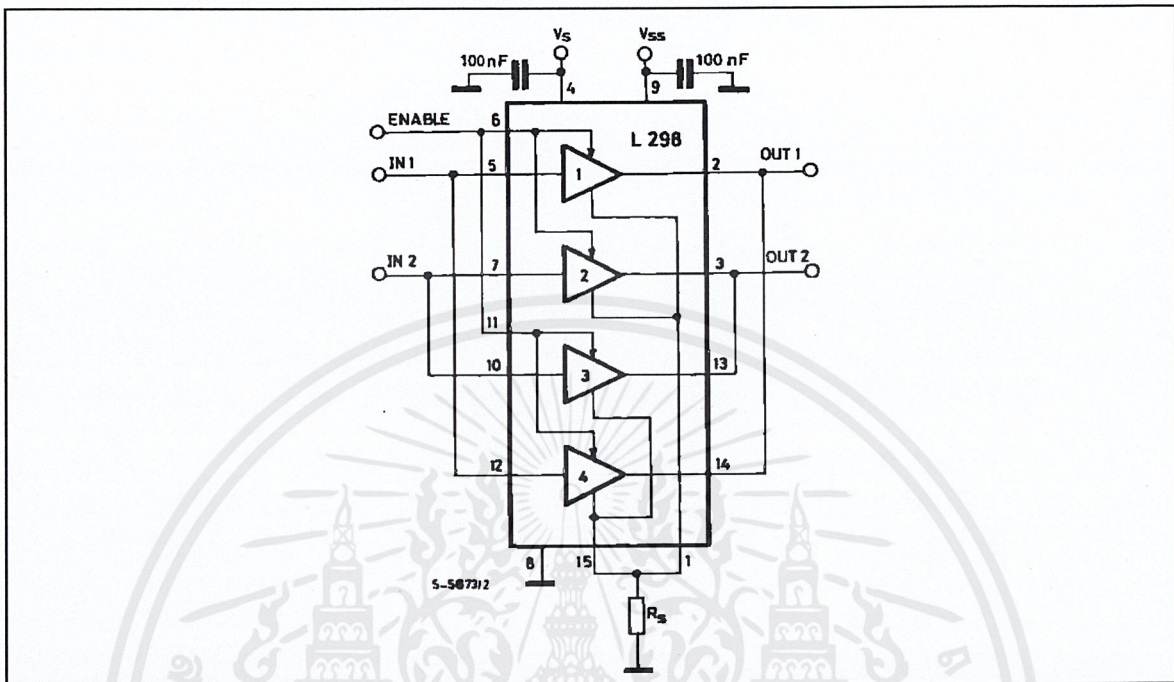


Figure 6 : Bidirectional DC Motor Control.



**Figure 7 :** For higher currents, outputs can be paralleled. Take care to parallel channel 1 with channel 4 and channel 2 with channel 3.



## APPLICATION INFORMATION (Refer to the block diagram)

### 1.1. POWER OUTPUT STAGE

The L298 integrates two power output stages (A; B). The power output stage is a bridge configuration and its outputs can drive an inductive load in common or differential mode, depending on the state of the inputs. The current that flows through the load comes out from the bridge at the sense output: an external resistor ( $R_{SA}$ ;  $R_{SB}$ .) allows to detect the intensity of this current.

### 1.2. INPUT STAGE

Each bridge is driven by means of four gates the input of which are In1; In2; EnA and In3; In4; EnB. The In inputs set the bridge state when The En input is high; a low state of the En input inhibits the bridge. All the inputs are TTL compatible.

## 2. SUGGESTIONS

A non inductive capacitor, usually of 100 nF, must be foreseen between both Vs and Vss, to ground, as near as possible to GND pin. When the large capacitor of the power supply is too far from the IC, a second smaller one must be foreseen near the L298.

The sense resistor, not of a wire wound type, must be grounded near the negative pole of Vs that must be near the GND pin of the I.C.

Each input must be connected to the source of the driving signals by means of a very short path.

Turn-On and Turn-Off : Before to Turn-ON the Supply Voltage and before to Turn it OFF, the Enable input must be driven to the Low state.

## 3. APPLICATIONS

Fig 6 shows a bidirectional DC motor control Schematic Diagram for which only one bridge is needed. The external bridge of diodes D1 to D4 is made by four fast recovery elements ( $t_{rr} \leq 200$  nsec) that must be chosen of a VF as low as possible at the worst case of the load current.

The sense output voltage can be used to control the current amplitude by chopping the inputs, or to provide overcurrent protection by switching low the enable input.

The brake function (Fast motor stop) requires that the Absolute Maximum Rating of 2 Amps must never be overcome.

When the repetitive peak current needed from the load is higher than 2 Amps, a paralleled configuration can be chosen (See Fig.7).

An external bridge of diodes are required when inductive loads are driven and when the inputs of the IC are chopped; Schottky diodes would be preferred.

**L298**

This solution can drive until 3 Amps In DC operation and until 3.5 Amps of a repetitive peak current.

On Fig 8 it is shown the driving of a two phase bipolar stepper motor ; the needed signals to drive the inputs of the L298 are generated, in this example, from the IC L297.

Fig 9 shows an example of P.C.B. designed for the application of Fig 8.

**Figure 8 :** Two Phase Bipolar Stepper Motor Circuit.

This circuit drives bipolar stepper motors with winding currents up to 2 A. The diodes are fast 2 A types.

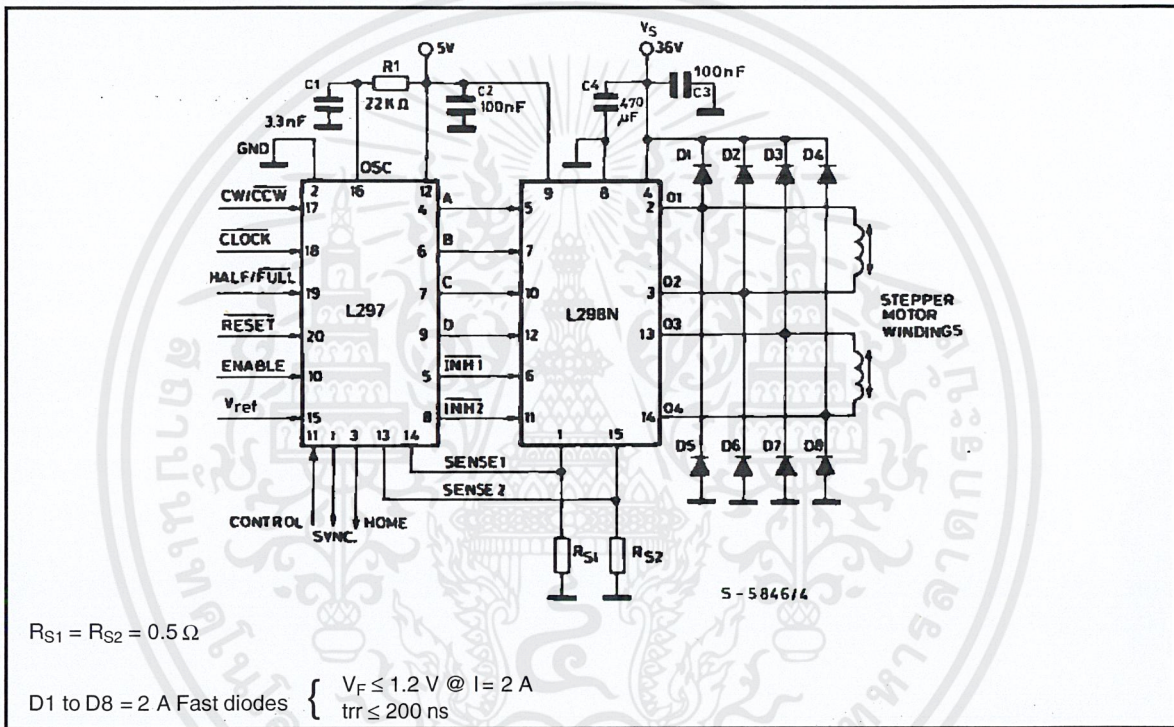


Fig 10 shows a second two phase bipolar stepper motor control circuit where the current is controlled by the I.C. L6506.

Figure 9 : Suggested Printed Circuit Board Layout for the Circuit of fig. 8 (1:1 scale).

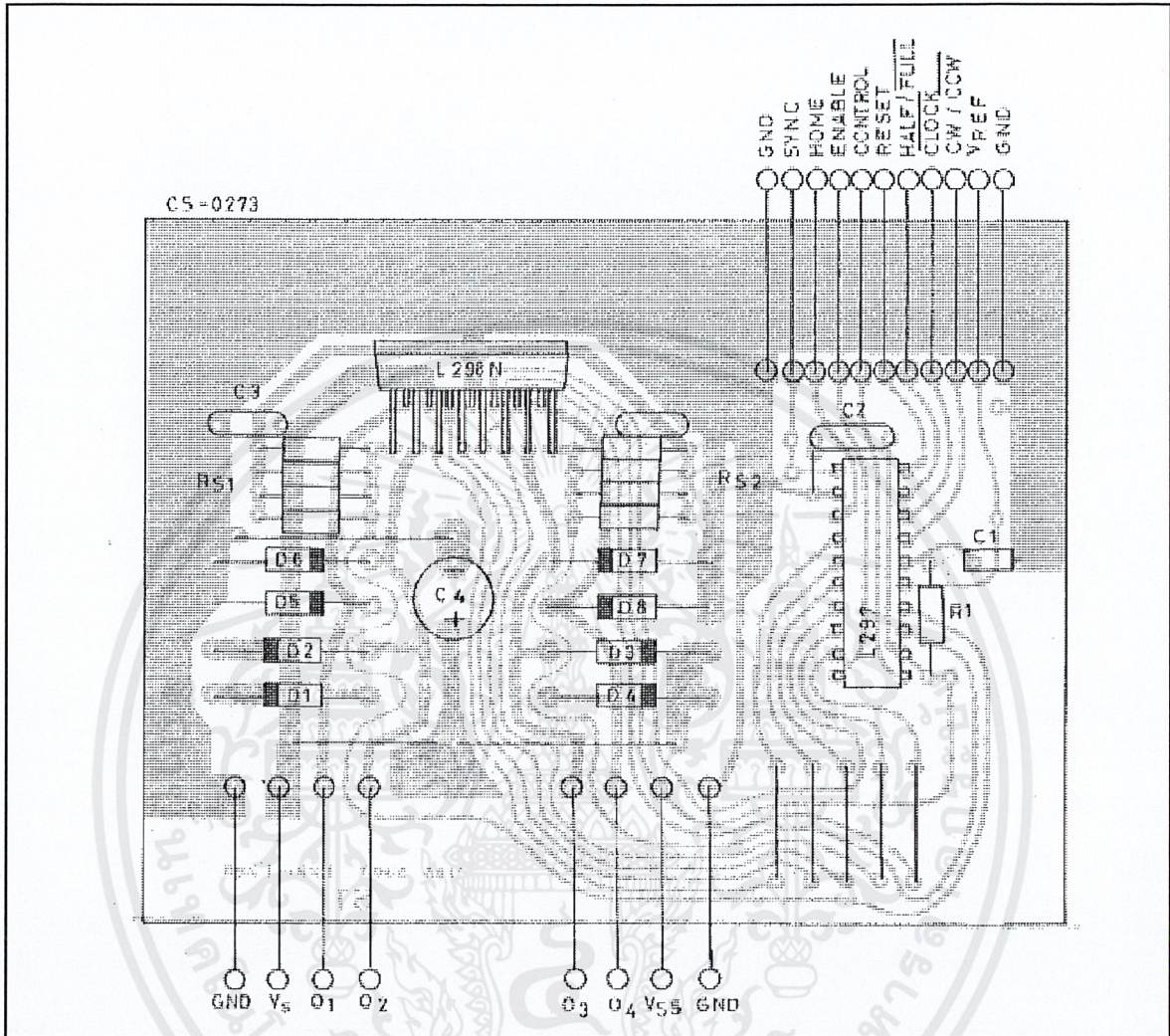
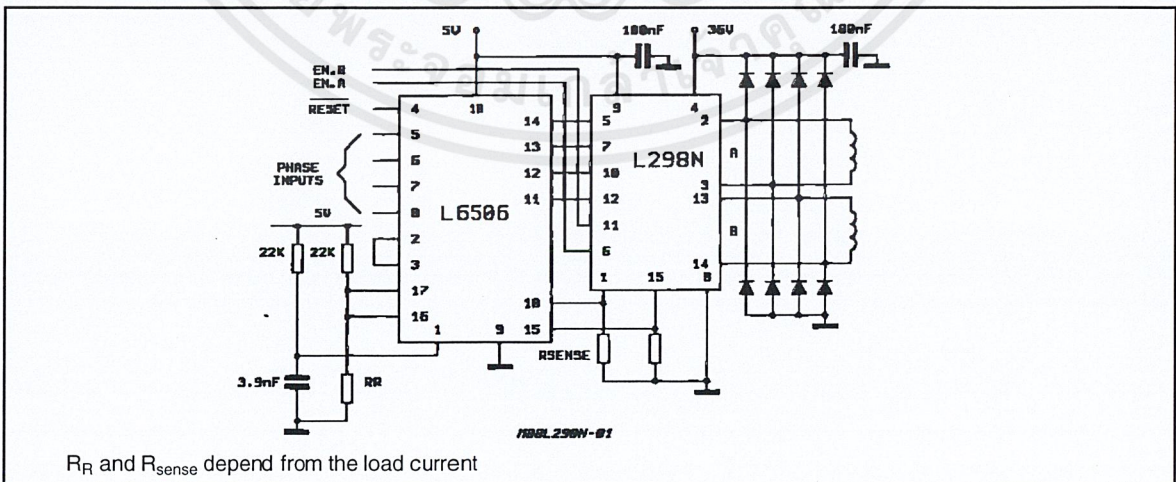
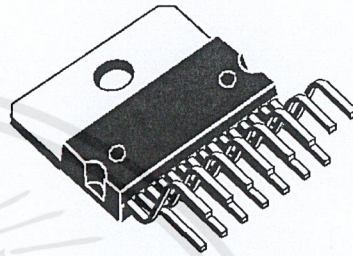


Figure 10 : Two Phase Bipolar Stepper Motor Control Circuit by Using the Current Controller L6506.

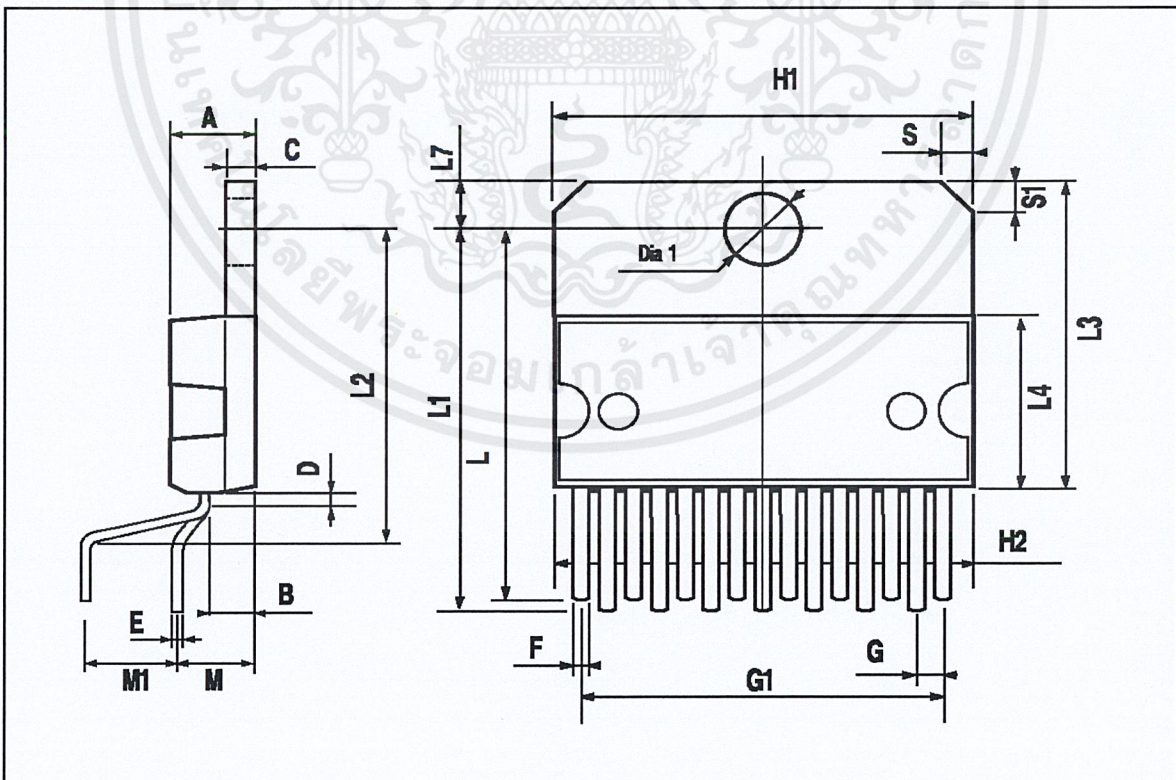


DIM.	mm			inch		
	MIN.	TYP.	MAX.	MIN.	TYP.	MAX.
A			5			0.197
B			2.65			0.104
C			1.6			0.063
D		1			0.039	
E	0.49		0.55	0.019		0.022
F	0.66		0.75	0.026		0.030
G	1.02	1.27	1.52	0.040	0.050	0.060
G1	17.53	17.78	18.03	0.690	0.700	0.710
H1	19.6			0.772		
H2			20.2			0.795
L	21.9	22.2	22.5	0.862	0.874	0.886
L1	21.7	22.1	22.5	0.854	0.870	0.886
L2	17.65		18.1	0.695		0.713
L3	17.25	17.5	17.75	0.679	0.689	0.699
L4	10.3	10.7	10.9	0.406	0.421	0.429
L7	2.65		2.9	0.104		0.114
M	4.25	4.55	4.85	0.167	0.179	0.191
M1	4.63	5.08	5.53	0.182	0.200	0.218
S	1.9		2.6	0.075		0.102
S1	1.9		2.6	0.075		0.102
Dia1	3.65		3.85	0.144		0.152

## OUTLINE AND MECHANICAL DATA

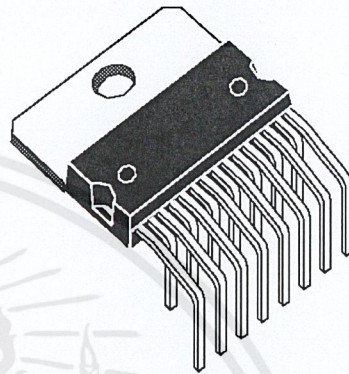


**Multiwatt15 V**

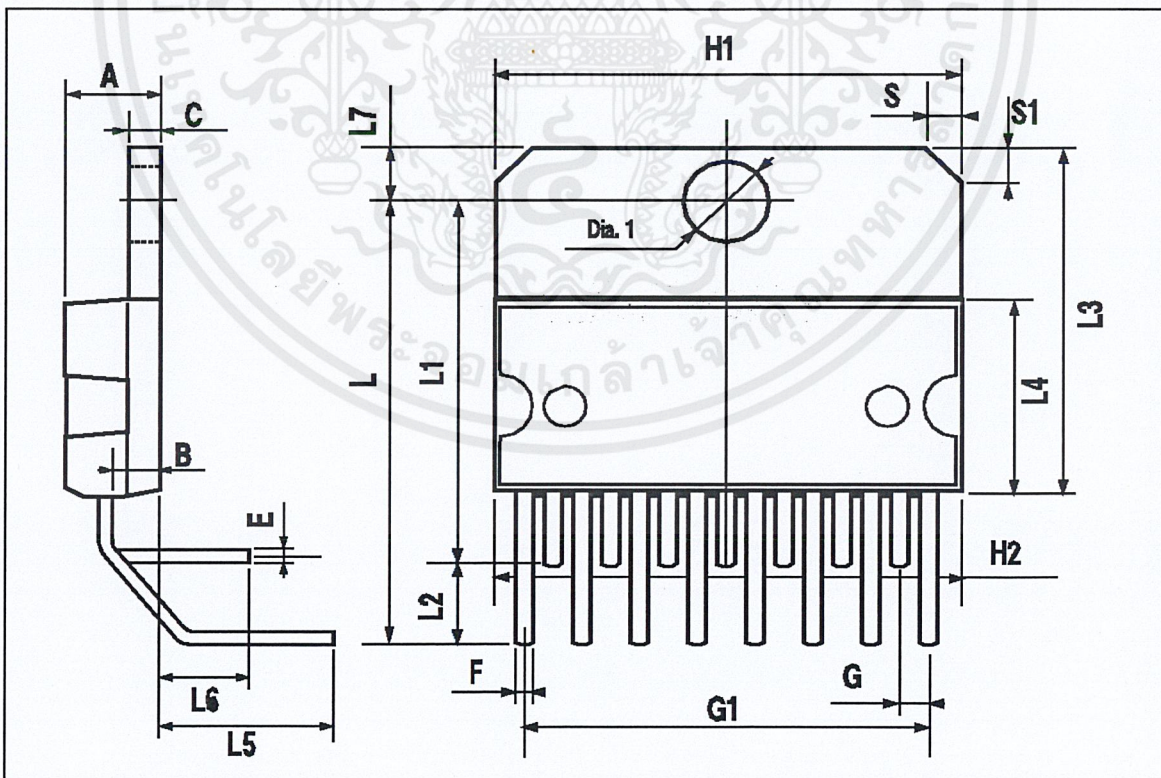


DIM.	mm			inch		
	MIN.	TYP.	MAX.	MIN.	TYP.	MAX.
A			5			0.197
B			2.65			0.104
C			1.6			0.063
E	0.49		0.55	0.019		0.022
F	0.66		0.75	0.026		0.030
G	1.14	1.27	1.4	0.045	0.050	0.055
G1	17.57	17.78	17.91	0.692	0.700	0.705
H1	19.6			0.772		
H2			20.2			0.795
L		20.57			0.810	
L1		18.03			0.710	
L2		2.54			0.100	
L3	17.25	17.5	17.75	0.679	0.689	0.699
L4	10.3	10.7	10.9	0.406	0.421	0.429
L5		5.28			0.208	
L6		2.38			0.094	
L7	2.65		2.9	0.104		0.114
S	1.9		2.6	0.075		0.102
S1	1.9		2.6	0.075		0.102
Dia1	3.65		3.85	0.144		0.152

## OUTLINE AND MECHANICAL DATA



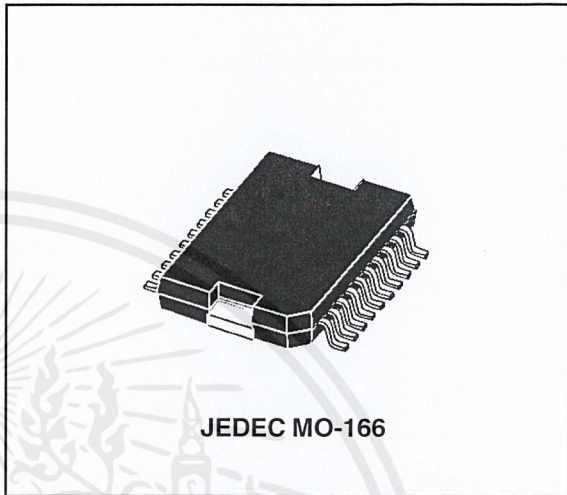
**Multiwatt15 H**



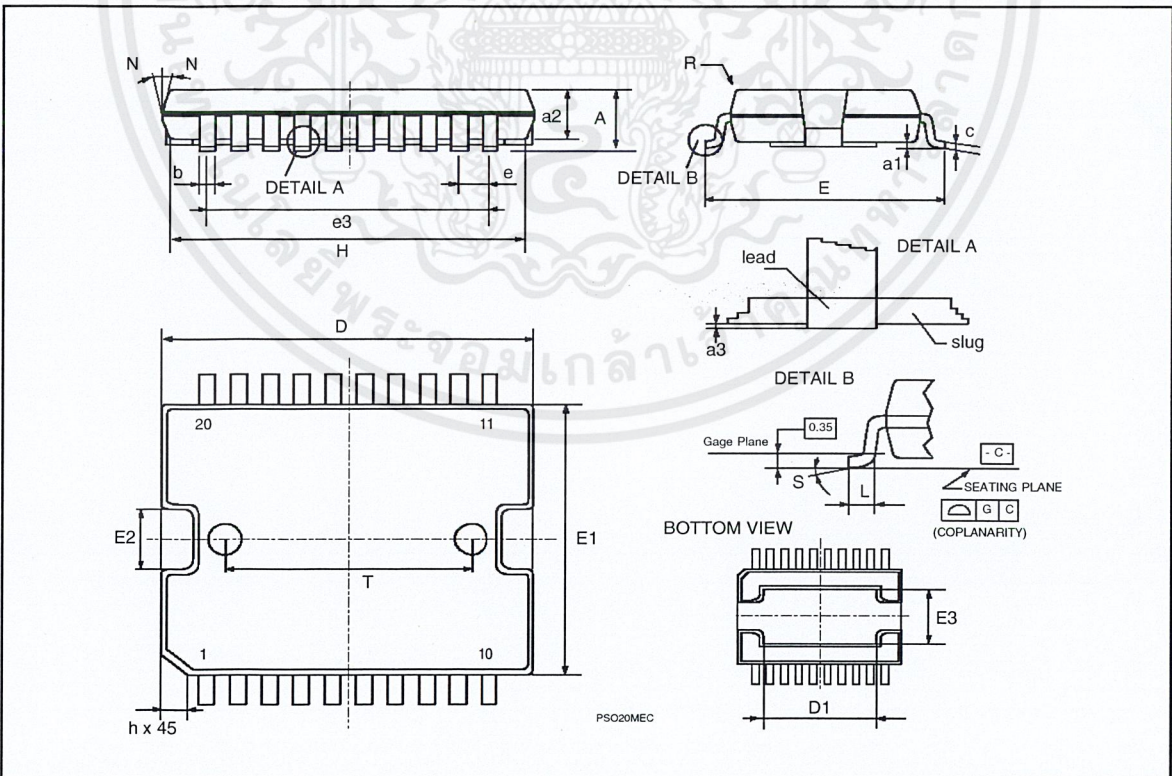
DIM.	mm			inch		
	MIN.	TYP.	MAX.	MIN.	TYP.	MAX.
A			3.6			0.142
a1	0.1		0.3	0.004		0.012
a2			3.3			0.130
a3	0		0.1	0.000		0.004
b	0.4		0.53	0.016		0.021
c	0.23		0.32	0.009		0.013
D (1)	15.8		16	0.622		0.630
D1	9.4		9.8	0.370		0.386
E	13.9		14.5	0.547		0.570
e		1.27			0.050	
e3		11.43			0.450	
E1 (1)	10.9		11.1	0.429		0.437
E2			2.9			0.114
E3	5.8		6.2	0.228		0.244
G	0		0.1	0.000		0.004
H	15.5		15.9	0.610		0.626
h			1.1			0.043
L	0.8		1.1	0.031		0.043
N	10° (max.)					
S	8° (max.)					
T		10			0.394	

(1) "D and F" do not include mold flash or protrusions.  
 - Mold flash or protrusions shall not exceed 0.15 mm (0.006").  
 - Critical dimensions "E", "G" and "a3"

**OUTLINE AND MECHANICAL DATA**



**PowerSO20**



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
 ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



Information furnished is believed to be accurate and reliable. However, STMicroelectronics assumes no responsibility for the consequences of use of such information nor for any infringement of patents or other rights of third parties which may result from its use. No license is granted by implication or otherwise under any patent or patent rights of STMicroelectronics. Specification mentioned in this publication are subject to change without notice. This publication supersedes and replaces all information previously supplied. STMicroelectronics products are not authorized for use as critical components in life support devices or systems without express written approval of STMicroelectronics.

The ST logo is a registered trademark of STMicroelectronics  
 © 2000 STMicroelectronics – Printed in Italy – All Rights Reserved  
 STMicroelectronics GROUP OF COMPANIES

Australia - Brazil - China - Finland - France - Germany - Hong Kong - India - Italy - Japan - Malaysia - Malta - Morocco -  
 Singapore - Spain - Sweden - Switzerland - United Kingdom - U.S.A.

<http://www.st.com>



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
 ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



**ภาคผนวก ค:**  
**คำนำชี้ทของไอซี ULN2803**

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



# Octal High Voltage, High Current Darlington Transistor Arrays

The eight NPN Darlington connected transistors in this family of arrays are ideally suited for interfacing between low logic level digital circuitry (such as TTL, CMOS or PMOS/NMOS) and the higher current/voltage requirements of lamps, relays, printer hammers or other similar loads for a broad range of computer, industrial, and consumer applications. All devices feature open-collector outputs and free wheeling clamp diodes for transient suppression.

The ULN2803 is designed to be compatible with standard TTL families while the ULN2804 is optimized for 6 to 15 volt high level CMOS or PMOS.

**MAXIMUM RATINGS** ( $T_A = 25^\circ\text{C}$  and rating apply to any one device in the package, unless otherwise noted.)

Rating	Symbol	Value	Unit
Output Voltage	$V_O$	50	V
Input Voltage (Except ULN2801)	$V_I$	30	V
Collector Current – Continuous	$I_C$	500	mA
Base Current – Continuous	$I_B$	25	mA
Operating Ambient Temperature Range	$T_A$	0 to +70	$^\circ\text{C}$
Storage Temperature Range	$T_{\text{stg}}$	-55 to +150	$^\circ\text{C}$
Junction Temperature	$T_J$	125	$^\circ\text{C}$

$R_{\theta JA} = 55^\circ\text{C/W}$   
Do not exceed maximum current limit per driver.

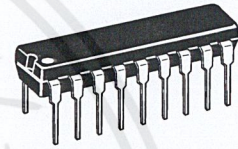
## ORDERING INFORMATION

Device	Characteristics		
	Input Compatibility	$V_{CE(\text{Max})}/I_{C(\text{Max})}$	Operating Temperature Range
ULN2803A	TTL, 5.0 V CMOS	50 V/500 mA	$T_A = 0$ to $+70^\circ\text{C}$
ULN2804A	6 to 15 V CMOS, PMOS		

# ULN2803 ULN2804

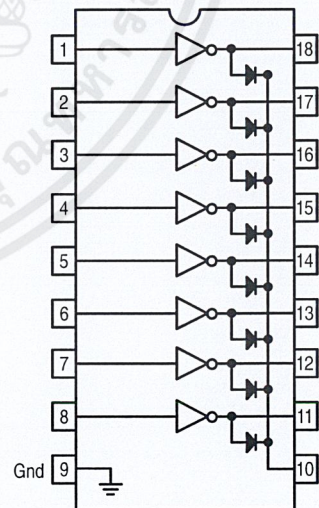
## OCTAL PERIPHERAL DRIVER ARRAYS

### SEMICONDUCTOR TECHNICAL DATA



A SUFFIX  
PLASTIC PACKAGE  
CASE 707

## PIN CONNECTIONS



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## ULN2803 ULN2804

### ELECTRICAL CHARACTERISTICS ( $T_A = 25^\circ\text{C}$ , unless otherwise noted)

Characteristic		Symbol	Min	Typ	Max	Unit
Output Leakage Current (Figure 1) ( $V_O = 50\text{ V}$ , $T_A = +70^\circ\text{C}$ ) ( $V_O = 50\text{ V}$ , $T_A = +25^\circ\text{C}$ ) ( $V_O = 50\text{ V}$ , $T_A = +70^\circ\text{C}$ , $V_I = 6.0\text{ V}$ ) ( $V_O = 50\text{ V}$ , $T_A = +70^\circ\text{C}$ , $V_I = 1.0\text{ V}$ )	All Types All Types ULN2802 ULN2804	$I_{CEX}$	–	–	100 50 500 500	$\mu\text{A}$
Collector–Emitter Saturation Voltage (Figure 2) ( $I_C = 350\text{ mA}$ , $I_B = 500\text{ }\mu\text{A}$ ) ( $I_C = 200\text{ mA}$ , $I_B = 350\text{ }\mu\text{A}$ ) ( $I_C = 100\text{ mA}$ , $I_B = 250\text{ }\mu\text{A}$ )	All Types All Types All Types	$V_{CE(sat)}$	–	1.1 0.95 0.85	1.6 1.3 1.1	V
Input Current – On Condition (Figure 4) ( $V_I = 17\text{ V}$ ) ( $V_I = 3.85\text{ V}$ ) ( $V_I = 5.0\text{ V}$ ) ( $V_I = 12\text{ V}$ )	ULN2802 ULN2803 ULN2804 ULN2804	$I_{I(on)}$	–	0.82 0.93 0.35 1.0	1.25 1.35 0.5 1.45	mA
Input Voltage – On Condition (Figure 5) ( $V_{CE} = 2.0\text{ V}$ , $I_C = 300\text{ mA}$ ) ( $V_{CE} = 2.0\text{ V}$ , $I_C = 200\text{ mA}$ ) ( $V_{CE} = 2.0\text{ V}$ , $I_C = 250\text{ mA}$ ) ( $V_{CE} = 2.0\text{ V}$ , $I_C = 300\text{ mA}$ ) ( $V_{CE} = 2.0\text{ V}$ , $I_C = 125\text{ mA}$ ) ( $V_{CE} = 2.0\text{ V}$ , $I_C = 200\text{ mA}$ ) ( $V_{CE} = 2.0\text{ V}$ , $I_C = 275\text{ mA}$ ) ( $V_{CE} = 2.0\text{ V}$ , $I_C = 350\text{ mA}$ )	ULN2802 ULN2803 ULN2803 ULN2803 ULN2804 ULN2804 ULN2804 ULN2804	$V_{I(on)}$	–	–	13 2.4 2.7 3.0 5.0 6.0 7.0 8.0	V
Input Current – Off Condition (Figure 3) ( $I_C = 500\text{ }\mu\text{A}$ , $T_A = +70^\circ\text{C}$ )	All Types	$I_{I(off)}$	50	100	–	$\mu\text{A}$
DC Current Gain (Figure 2) ( $V_{CE} = 2.0\text{ V}$ , $I_C = 350\text{ mA}$ )	ULN2801	$h_{FE}$	1000	–	–	–
Input Capacitance		$C_I$	–	15	25	pF
Turn–On Delay Time (50% $E_I$ to 50% $E_O$ )		$t_{on}$	–	0.25	1.0	$\mu\text{s}$
Turn–Off Delay Time (50% $E_I$ to 50% $E_O$ )		$t_{off}$	–	0.25	1.0	$\mu\text{s}$
Clamp Diode Leakage Current (Figure 6) ( $V_R = 50\text{ V}$ )	$T_A = +25^\circ\text{C}$ $T_A = +70^\circ\text{C}$	$I_R$	–	–	50 100	$\mu\text{A}$
Clamp Diode Forward Voltage (Figure 7) ( $I_F = 350\text{ mA}$ )		$V_F$	–	1.5	2.0	V

# ULN2803 ULN2804

## TEST FIGURES

(See Figure Numbers in Electrical Characteristics Table)

Figure 1.

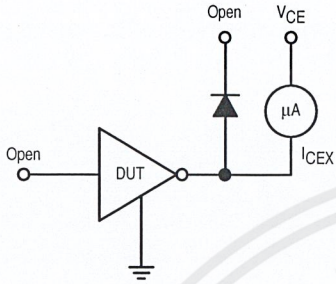


Figure 2.

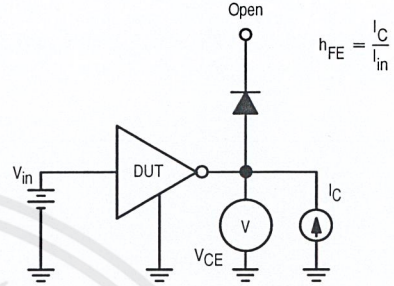


Figure 3.

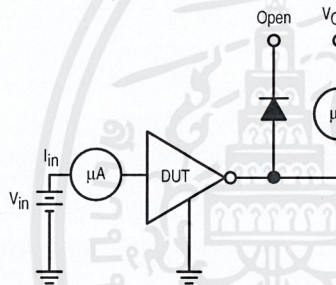


Figure 4.

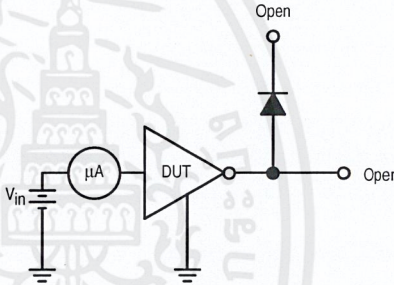


Figure 5.

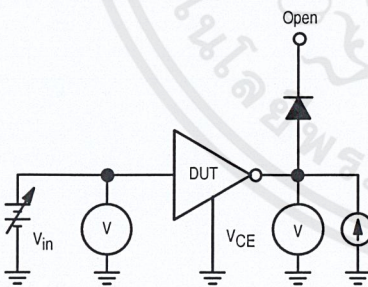


Figure 6.

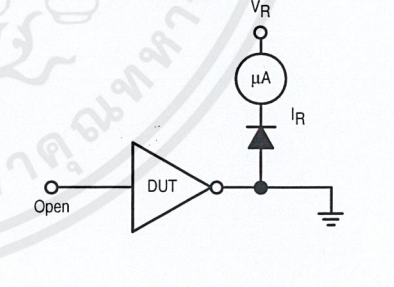
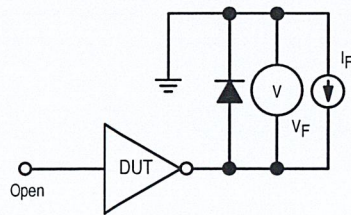


Figure 7.



# ULN2803 ULN2804

TYPICAL CHARACTERISTIC CURVES –  $T_A = 25^\circ\text{C}$ , unless otherwise noted  
Output Characteristics

Figure 8. Output Current versus Saturation Voltage

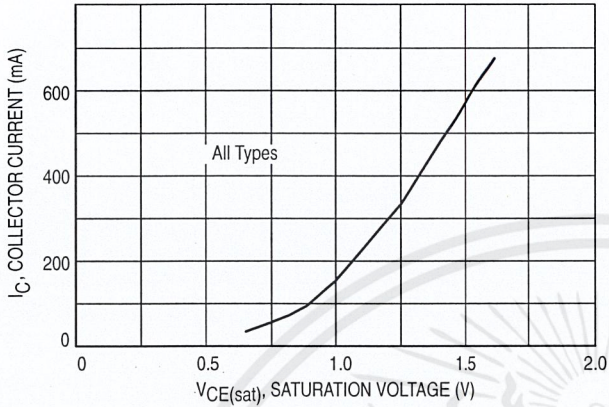
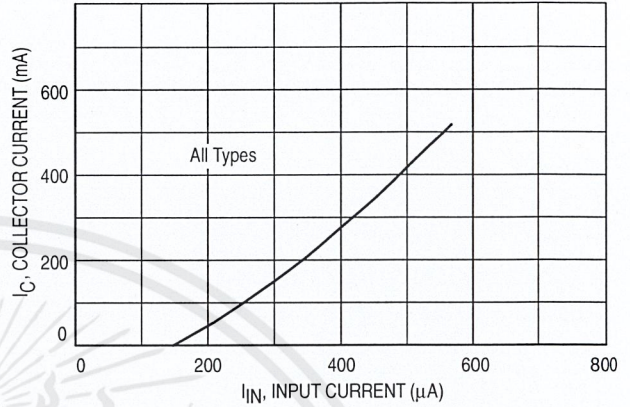


Figure 9. Output Current versus Input Current



## Input Characteristics

Figure 10. ULN2803 Input Current versus Input Voltage

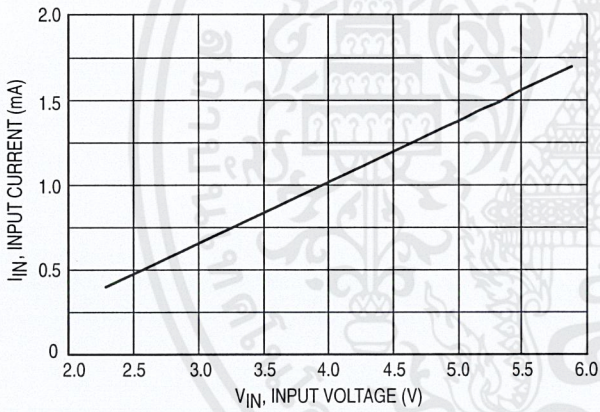


Figure 11. ULN2804 Input Current versus Input Voltage

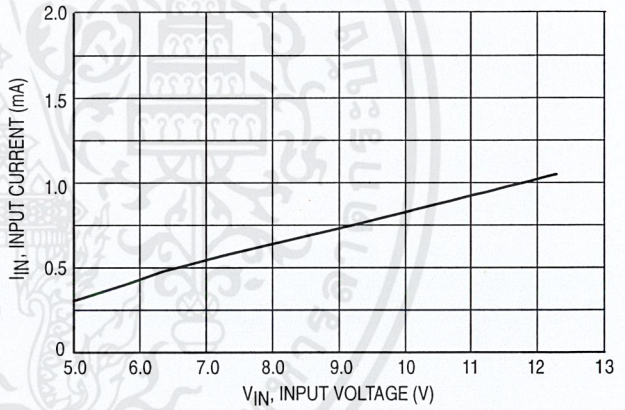
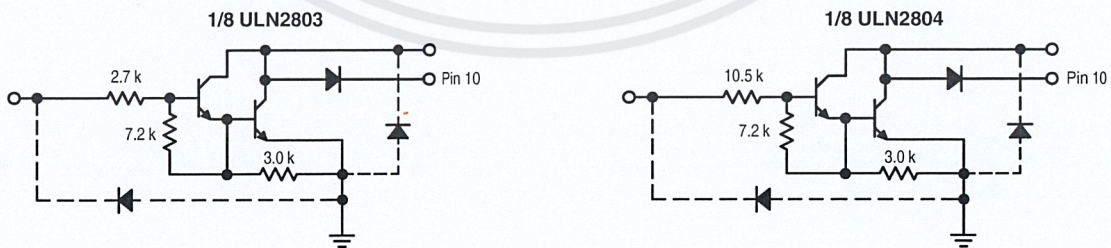


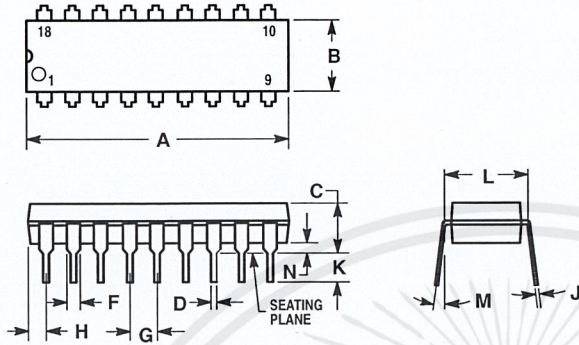
Figure 12. Representative Schematic Diagrams



# ULN2803 ULN2804

## OUTLINE DIMENSIONS

A SUFFIX  
PLASTIC PACKAGE  
CASE 707-02  
ISSUE C




NOTES:

1. POSITIONAL TOLERANCE OF LEADS (D), SHALL BE WITHIN 0.25 (0.010) AT MAXIMUM MATERIAL CONDITION, IN RELATION TO SEATING PLANE AND EACH OTHER.
2. DIMENSION L TO CENTER OF LEADS WHEN FORMED PARALLEL.
3. DIMENSION B DOES NOT INCLUDE MOLD FLASH.

DIM	MILLIMETERS		INCHES	
	MIN	MAX	MIN	MAX
A	22.22	23.24	0.875	0.915
B	6.10	6.60	0.240	0.260
C	3.56	4.57	0.140	0.180
D	0.36	0.56	0.014	0.022
F	1.27	1.78	0.050	0.070
G	2.54 BSC		0.100 BSC	
H	1.02	1.52	0.040	0.060
J	0.20	0.30	0.008	0.012
K	2.92	3.43	0.115	0.135
L	7.62 BSC		0.300 BSC	
M	0°	15°	0°	15°
N	0.51	1.02	0.020	0.040

# ULN2803 ULN2804



Motorola reserves the right to make changes without further notice to any products herein. Motorola makes no warranty, representation or guarantee regarding the suitability of its products for any particular purpose, nor does Motorola 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 consequential or incidental damages. "Typical" parameters which may be provided in Motorola 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. Motorola does not convey any license under its patent rights nor the rights of others. Motorola 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 Motorola product could create a situation where personal injury or death may occur. Should Buyer purchase or use Motorola products for any such unintended or unauthorized application, Buyer shall indemnify and hold Motorola 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 Motorola was negligent regarding the design or manufacture of the part. Motorola and  are registered trademarks of Motorola, Inc. Motorola, Inc. is an Equal Opportunity/Affirmative Action Employer.

#### How to reach us:

**USA/EUROPE/Locations Not Listed:** Motorola Literature Distribution;  
P.O. Box 20912; Phoenix, Arizona 85036. 1-800-441-2447 or 602-303-5454

**JAPAN:** Nippon Motorola Ltd.; Tatsumi-SPD-JLDC, 6F Seibu-Butsuryu-Center,  
3-14-2 Tatsumi Koto-Ku, Tokyo 135, Japan. 03-81-3521-8315

**MFAX:** RMFAX0@email.sps.mot.com – TOUCHTONE 602-244-6609  
**INTERNET:** <http://Design-NET.com>

**ASIA/PACIFIC:** Motorola Semiconductors H.K. Ltd.; 8B Tai Ping Industrial Park,  
51 Ting Kok Road, Tai Po, N.T., Hong Kong. 852-26629298



**MOTOROLA**

ULN2803/D

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุย



ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



**ภาคผนวก ง:**  
**คำตัด้าชี้ทของไอซี 74LS139**

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

# SN74LS139

## Dual 1-of-4 Decoder/ Demultiplexer

The LSTTL/MSI SN74LS139 is a high speed Dual 1-of-4 Decoder/Demultiplexer. The device has two independent decoders, each accepting two inputs and providing four mutually exclusive active LOW Outputs. Each decoder has an active LOW Enable input which can be used as a data input for a 4-output demultiplexer. Each half of the LS139 can be used as a function generator providing all four minterms of two variables. The LS139 is fabricated with the Schottky barrier diode process for high speed and is completely compatible with all ON Semiconductor TTL families.

- Schottky Process for High Speed
- Multifunction Capability
- Two Completely Independent 1-of-4 Decoders
- Active Low Mutually Exclusive Outputs
- Input Clamp Diodes Limit High Speed Termination Effects
- ESD > 3500 Volts

### GUARANTEED OPERATING RANGES

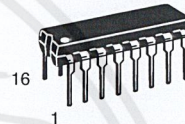
Symbol	Parameter	Min	Typ	Max	Unit
V <sub>CC</sub>	Supply Voltage	4.75	5.0	5.25	V
T <sub>A</sub>	Operating Ambient Temperature Range	0	25	70	°C
I <sub>OH</sub>	Output Current – High			-0.4	mA
I <sub>OL</sub>	Output Current – Low			8.0	mA



ON Semiconductor™

<http://onsemi.com>

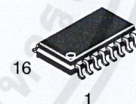
LOW  
POWER  
SCHOTTKY



PLASTIC  
N SUFFIX  
CASE 648



SOIC  
D SUFFIX  
CASE 751B



SOEIAJ  
M SUFFIX  
CASE 966

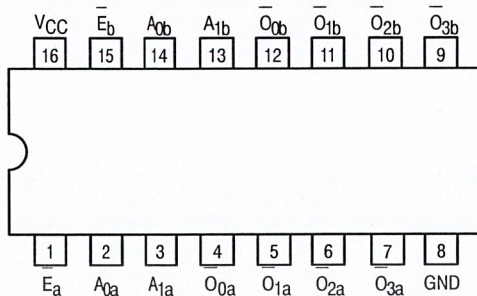
### ORDERING INFORMATION

Device	Package	Shipping
SN74LS139N	16 Pin DIP	2000 Units/Box
SN74LS139D	SOIC-16	38 Units/Rail
SN74LS139DR2	SOIC-16	2500/Tape & Reel
SN74LS139M	SOEIAJ-16	See Note 1
SN74LS139MEL	SOEIAJ-16	See Note 1

1. For ordering information on the EIAJ version of the SOIC package, please contact your local ON Semiconductor representative.

# SN74LS139

## CONNECTION DIAGRAM DIP (TOP VIEW)



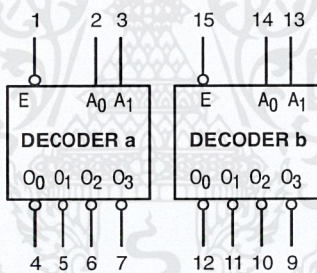
NOTE:  
The Flatpak version has the same pinouts (Connection Diagram) as the Dual In-Line Package.

PIN NAMES		LOADING (Note a)	
		HIGH	LOW
A <sub>0</sub> , A <sub>1</sub>	Address Inputs	0.5 U.L.	0.25 U.L.
E <sub>-</sub>	Enable (Active LOW) Input	0.5 U.L.	0.25 U.L.
O <sub>0</sub> - O <sub>3</sub>	Active LOW Outputs	10 U.L.	5 U.L.

### NOTES:

a) 1 TTL Unit Load (U.L.) = 40  $\mu$ A HIGH/1.6 mA LOW.

## LOGIC SYMBOL



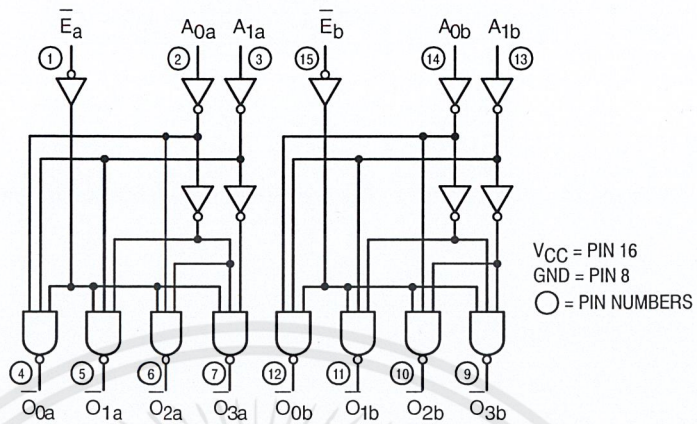
V<sub>CC</sub> = PIN 16  
GND = PIN 8

<http://onsemi.com>

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

# SN74LS139

## LOGIC DIAGRAM



### FUNCTIONAL DESCRIPTION

The LS139 is a high speed dual 1-of-4 decoder/demultiplexer fabricated with the Schottky barrier diode process. The device has two independent decoders, each of which accept two binary weighted inputs ( $A_0$ ,  $A_1$ ) and provide four mutually exclusive active LOW outputs ( $O_0$ – $O_3$ ). Each decoder has an active LOW Enable ( $E$ ). When  $E$  is HIGH all outputs are forced HIGH. The enable

can be used as the data input for a 4-output demultiplexer application.

Each half of the LS139 generates all four minterms of two variables. These four minterms are useful in some applications, replacing multiple gate functions as shown in Fig. a, and thereby reducing the number of packages required in a logic network.

### TRUTH TABLE

INPUTS			OUTPUTS			
$\bar{E}$	$A_0$	$A_1$	$O_0$	$O_1$	$O_2$	$O_3$
H	X	X	H	H	H	H
L	L	L	L	H	H	H
L	H	L	H	L	H	H
L	L	H	H	H	L	H
L	H	H	H	H	H	L

H = HIGH Voltage Level  
L = LOW Voltage Level  
X = Don't Care

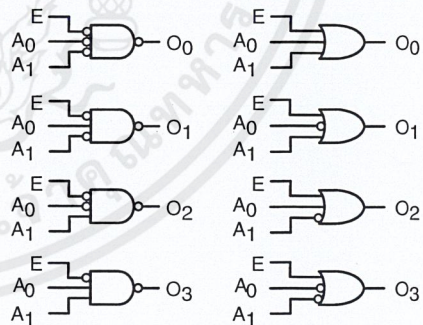


Figure a

# SN74LS139

## DC CHARACTERISTICS OVER OPERATING TEMPERATURE RANGE (unless otherwise specified)

Symbol	Parameter	Limits			Unit	Test Conditions
		Min	Typ	Max		
V <sub>IH</sub>	Input HIGH Voltage	2.0			V	Guaranteed Input HIGH Voltage for All Inputs
V <sub>IL</sub>	Input LOW Voltage			0.8	V	Guaranteed Input LOW Voltage for All Inputs
V <sub>IK</sub>	Input Clamp Diode Voltage		-0.65	-1.5	V	V <sub>CC</sub> = MIN, I <sub>IN</sub> = -18 mA
V <sub>OH</sub>	Output HIGH Voltage	2.7	3.5		V	V <sub>CC</sub> = MIN, I <sub>OH</sub> = MAX, V <sub>IN</sub> = V <sub>IH</sub> or V <sub>IL</sub> per Truth Table
V <sub>OL</sub>	Output LOW Voltage		0.25	0.4	V	I <sub>OL</sub> = 4.0 mA V <sub>CC</sub> = V <sub>CC</sub> MIN, V <sub>IN</sub> = V <sub>IL</sub> or V <sub>IH</sub> per Truth Table
			0.35	0.5	V	
I <sub>IH</sub>	Input HIGH Current			20	μA	V <sub>CC</sub> = MAX, V <sub>IN</sub> = 2.7 V
				0.1	mA	V <sub>CC</sub> = MAX, V <sub>IN</sub> = 7.0 V
I <sub>IL</sub>	Input LOW Current			-0.4	mA	V <sub>CC</sub> = MAX, V <sub>IN</sub> = 0.4 V
I <sub>OS</sub>	Short Circuit Current (Note 2)	-20		-100	mA	V <sub>CC</sub> = MAX
I <sub>CC</sub>	Power Supply Current			11	mA	V <sub>CC</sub> = MAX

2. Not more than one output should be shorted at a time, nor for more than 1 second.

## AC CHARACTERISTICS (T<sub>A</sub> = 25°C)

Symbol	Parameter	Levels of Delay	Limits			Unit	Test Conditions
			Min	Typ	Max		
t <sub>PLH</sub> t <sub>PHL</sub>	Propagation Delay Address to Output	2 2		13 22	20 33	ns	V <sub>CC</sub> = 5.0 V C <sub>L</sub> = 15 pF
t <sub>PLH</sub> t <sub>PHL</sub>	Propagation Delay Address to Output	3 3		18 25	29 38	ns	
t <sub>PLH</sub> t <sub>PHL</sub>	Propagation Delay Enable to Output	2 2		16 21	24 32	ns	

## AC WAVEFORMS

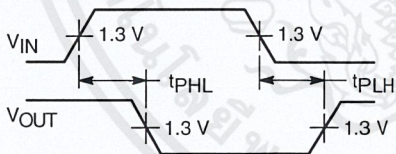


Figure 1.

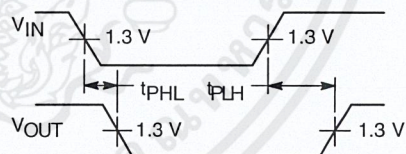
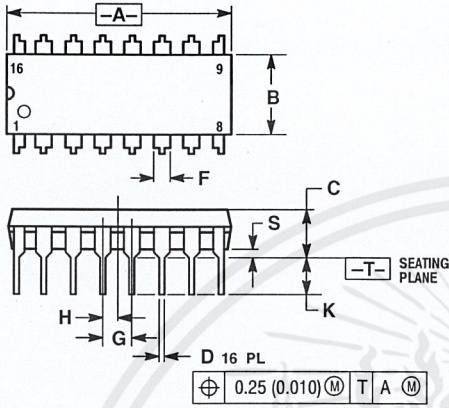


Figure 2.

# SN74LS139

## PACKAGE DIMENSIONS

N SUFFIX  
PLASTIC PACKAGE  
CASE 648-08  
ISSUE R



NOTES:

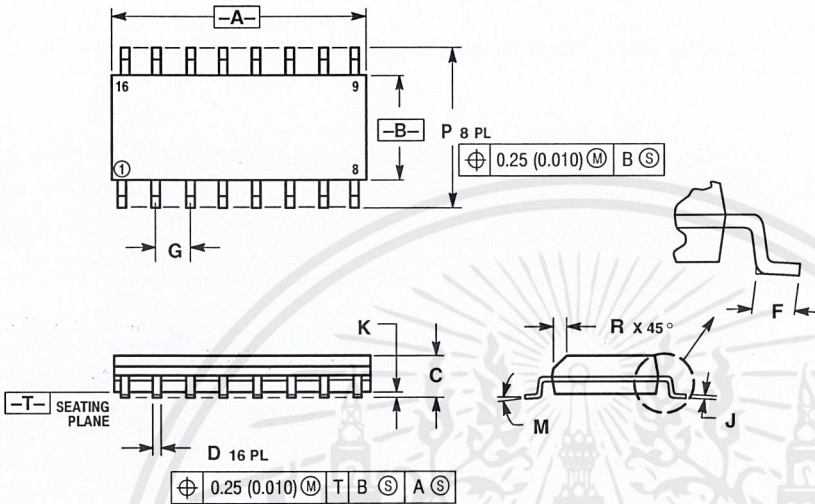
1. DIMENSIONING AND TOLERANCING PER ANSI Y14.5M, 1982.
2. CONTROLLING DIMENSION: INCH.
3. DIMENSION L TO CENTER OF LEADS WHEN FORMED PARALLEL.
4. DIMENSION B DOES NOT INCLUDE MOLD FLASH.
5. ROUNDED CORNERS OPTIONAL.

DIM	INCHES		MILLIMETERS	
	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

# SN74LS139

## PACKAGE DIMENSIONS

**D SUFFIX**  
 PLASTIC SOIC PACKAGE  
 CASE 751B-05  
 ISSUE J



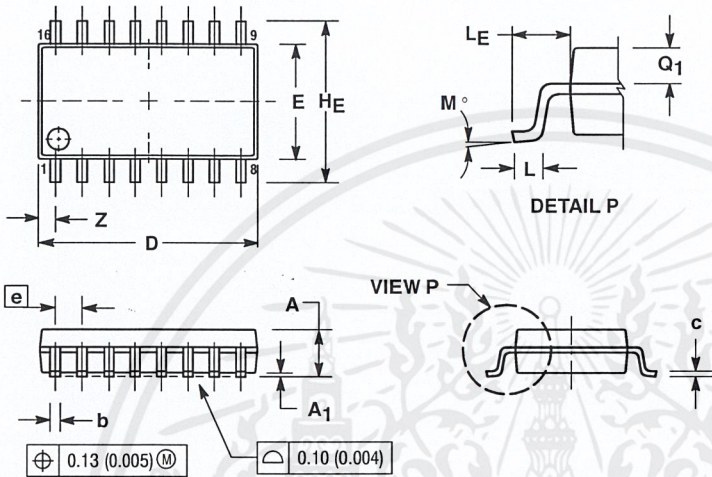
- NOTES:
1. DIMENSIONING AND TOLERANCING PER ANSI Y14.5M, 1982.
  2. CONTROLLING DIMENSION: MILLIMETER.
  3. DIMENSIONS A AND B DO NOT INCLUDE MOLD PROTRUSION.
  4. MAXIMUM MOLD PROTRUSION 0.15 (0.006) PER SIDE.
  5. DIMENSION D DOES NOT INCLUDE DAMBAR PROTRUSION. ALLOWABLE DAMBAR PROTRUSION SHALL BE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

DIM	MILLIMETERS		INCHES	
	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°	
P	5.80	6.20	0.229	0.244
R	0.25	0.50	0.010	0.019

# SN74LS139

## PACKAGE DIMENSIONS

**M SUFFIX**  
**SOEIAJ PACKAGE**  
**CASE 966-01**  
**ISSUE O**




**NOTES:**

1. DIMENSIONING AND TOLERANCING PER ANSI Y14.5M, 1982.
2. CONTROLLING DIMENSION: MILLIMETER.
3. DIMENSIONS D AND E DO NOT INCLUDE MOLD FLASH OR PROTRUSIONS AND ARE MEASURED AT THE PARTING LINE. MOLD FLASH OR PROTRUSIONS SHALL NOT EXCEED 0.15 (0.006) PER SIDE.
4. TERMINAL NUMBERS ARE SHOWN FOR REFERENCE ONLY.
5. THE LEAD WIDTH DIMENSION (b) DOES NOT INCLUDE DAMBAR PROTRUSION. ALLOWABLE DAMBAR PROTRUSION SHALL BE 0.08 (0.003) TOTAL IN EXCESS OF THE LEAD WIDTH DIMENSION AT MAXIMUM MATERIAL CONDITION. DAMBAR CANNOT BE LOCATED ON THE LOWER RADIUS OR THE FOOT. MINIMUM SPACE BETWEEN PROTRUSIONS AND ADJACENT LEAD TO BE 0.46 (0.018).

DIM	MILLIMETERS		INCHES	
	MIN	MAX	MIN	MAX
A	---	2.05	---	0.081
A1	0.05	0.20	0.002	0.008
b	0.35	0.50	0.014	0.020
c	0.18	0.27	0.007	0.011
D	9.90	10.50	0.390	0.413
E	5.10	5.45	0.201	0.215
e	1.27 BSC		0.050 BSC	
HE	7.40	8.20	0.291	0.323
L	0.50	0.85	0.020	0.033
LE	1.10	1.50	0.043	0.059
M	0°	10°	0°	10°
Q1	0.70	0.90	0.028	0.035
Z	---	0.78	---	0.031

SN74LS139



**ON Semiconductor** and  are 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.

SN74LS139/D

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



**ภาคผนวก จ:**  
**คำตัดชี้ทของไอซี 74LS76**

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



# DUAL JK FLIP-FLOP WITH SET AND CLEAR

The SN54/74LS76A offers individual J, K, Clock Pulse, Direct Set and Direct Clear inputs. These dual flip-flops are designed so that when the clock goes HIGH, the inputs are enabled and data will be accepted. The Logic Level of the J and K inputs will perform according to the Truth Table as long as minimum set-up times are observed. Input data is transferred to the outputs on the HIGH-to-LOW clock transitions.

MODE SELECT — TRUTH TABLE

OPERATING MODE	INPUTS				OUTPUTS	
	S <sub>D</sub>	C <sub>D</sub>	J	K	Q	$\bar{Q}$
Set	L	H	X	X	H	L
Reset (Clear)	H	L	X	X	L	H
*Undetermined	L	L	X	X	H	H
Toggle	H	H	h	h	q	q
Load "0" (Reset)	H	H	l	h	L	H
Load "1" (Set)	H	H	h	l	H	L
Hold	H	H	l	l	q	q

\*Both outputs will be HIGH while both S<sub>D</sub> and C<sub>D</sub> are LOW, but the output states are unpredictable if S<sub>D</sub> and C<sub>D</sub> go HIGH simultaneously.

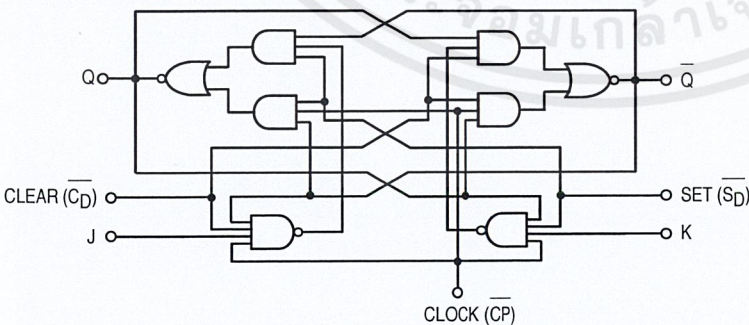
H, h = HIGH Voltage Level

L, l = LOW Voltage Level

X = Immaterial

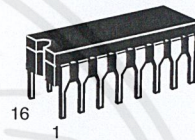
l, h (q) = Lower case letters indicate the state of the referenced input (or output) one setup time prior to the HIGH-to-LOW clock transition

LOGIC DIAGRAM

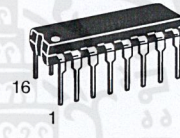


## SN54/74LS76A

DUAL JK FLIP-FLOP WITH SET AND CLEAR  
LOW POWER SCHOTTKY



J SUFFIX  
CERAMIC  
CASE 620-09



N SUFFIX  
PLASTIC  
CASE 648-08

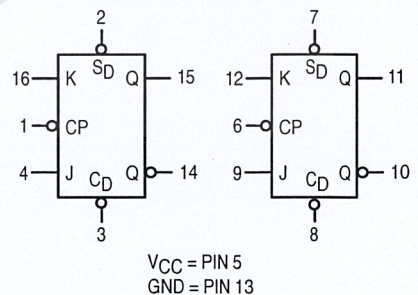


D SUFFIX  
SOIC  
CASE 751B-03

ORDERING INFORMATION

- SN54LSXXJ Ceramic
- SN74LSXXN Plastic
- SN74LSXXD SOIC

LOGIC SYMBOL



FAST AND LS TTL DATA

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษานั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

# SN54/74LS76A

## GUARANTEED OPERATING RANGES

Symbol	Parameter		Min	Typ	Max	Unit
V <sub>CC</sub>	Supply Voltage	54	4.5	5.0	5.5	V
		74	4.75	5.0	5.25	
T <sub>A</sub>	Operating Ambient Temperature Range	54	-55	25	125	°C
		74	0	25	70	
I <sub>OH</sub>	Output Current — High	54, 74			-0.4	mA
I <sub>OL</sub>	Output Current — Low	54			4.0	mA
		74			8.0	

## DC CHARACTERISTICS OVER OPERATING TEMPERATURE RANGE (unless otherwise specified)

Symbol	Parameter	Limits			Unit	Test Conditions
		Min	Typ	Max		
V <sub>IH</sub>	Input HIGH Voltage	2.0			V	Guaranteed Input HIGH Voltage for All Inputs
V <sub>IL</sub>	Input LOW Voltage	54		0.7	V	Guaranteed Input LOW Voltage for All Inputs
		74		0.8		
V <sub>IK</sub>	Input Clamp Diode Voltage		-0.65	-1.5	V	V <sub>CC</sub> = MIN, I <sub>IIN</sub> = -18 mA
V <sub>OH</sub>	Output HIGH Voltage	54	2.5	3.5	V	V <sub>CC</sub> = MIN, I <sub>OH</sub> = MAX, V <sub>IN</sub> = V <sub>IH</sub> or V <sub>IL</sub> per Truth Table
		74	2.7	3.5	V	
V <sub>OL</sub>	Output LOW Voltage	54, 74	0.25	0.4	V	I <sub>OL</sub> = 4.0 mA
		74	0.35	0.5	V	I <sub>OL</sub> = 8.0 mA
I <sub>IH</sub>	Input HIGH Current	J, K Clear Clock		20 60 80	μA	V <sub>CC</sub> = MAX, V <sub>IN</sub> = 2.7 V
				0.1 0.3 0.4	mA	V <sub>CC</sub> = MAX, V <sub>IN</sub> = 7.0 V
I <sub>IL</sub>	Input LOW Current	J, K Clear, Clock		-0.4 -0.8	mA	V <sub>CC</sub> = MAX, V <sub>IN</sub> = 0.4 V
I <sub>OS</sub>	Short Circuit Current (Note 1)		-20	-100	mA	V <sub>CC</sub> = MAX
I <sub>CC</sub>	Power Supply Current			6.0	mA	V <sub>CC</sub> = MAX

Note 1: Not more than one output should be shorted at a time, nor for more than 1 second.

## AC CHARACTERISTICS (T<sub>A</sub> = 25°C, V<sub>CC</sub> = 5.0 V)

Symbol	Parameter	Limits			Unit	Test Conditions
		Min	Typ	Max		
f <sub>MAX</sub>	Maximum Clock Frequency	30	45		MHz	V <sub>CC</sub> = 5.0 V C <sub>L</sub> = 15 pF
t <sub>PLH</sub> t <sub>PHL</sub>	Clock, Clear, Set to Output		15	20	ns	
			15	20	ns	

## AC SETUP REQUIREMENTS (T<sub>A</sub> = 25°C)

Symbol	Parameter	Limits			Unit	Test Conditions
		Min	Typ	Max		
t <sub>W</sub>	Clock Pulse Width High	20			ns	V <sub>CC</sub> = 5.0 V
t <sub>W</sub>	Clear Set Pulse Width	25			ns	
t <sub>S</sub>	Setup Time	20			ns	
t <sub>H</sub>	Hold Time	0			ns	

## FAST AND LS TTL DATA

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



**ภาคผนวก ฉ:**

**คำตัด้าชี้ทของไอซี 74LS86**

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## DM74LS86

### Quad 2-Input Exclusive-OR Gate

#### General Description

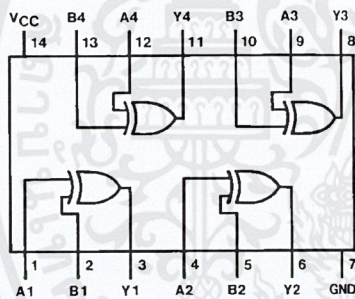
This device contains four independent gates each of which performs the logic exclusive-OR function.

#### Ordering Code:

Order Number	Package Number	Package Description
DM74LS86M	M14A	14-Lead Small Outline Integrated Circuit (SOIC), JEDEC MS-120, 0.150 Narrow
DM74LS86SJ	M14D	14-Lead Small Outline Package (SOP), EIAJ TYPE II, 5.3mm Wide
DM74LS86N	N14A	14-Lead Plastic Dual-In-Line Package (PDIP), JEDEC MS-001, 0.300 Wide

Devices also available in Tape and Reel. Specify by appending the suffix letter "X" to the ordering code.

#### Connection Diagram



#### Function Table

$$Y = A \oplus B = \bar{A}B + A\bar{B}$$

Inputs		Output
A	B	Y
L	L	L
L	H	H
H	L	H
H	H	L

H = HIGH Logic Level  
L = LOW Logic Level

**Absolute Maximum Ratings**(Note 1)

Supply Voltage	7V
Input Voltage	7V
Operating Free Air Temperature Range	0°C to +70°C
Storage Temperature Range	-65°C to +150°C

**Note 1:** The "Absolute Maximum Ratings" are those values beyond which the safety of the device cannot be guaranteed. The device should not be operated at these limits. The parametric values defined in the Electrical Characteristics tables are not guaranteed at the absolute maximum ratings. The "Recommended Operating Conditions" table will define the conditions for actual device operation.

**Recommended Operating Conditions**

Symbol	Parameter	Min	Nom	Max	Units
V <sub>CC</sub>	Supply Voltage	4.75	5	5.25	V
V <sub>IH</sub>	HIGH Level Input Voltage	2			V
V <sub>IL</sub>	LOW Level Input Voltage			0.8	V
I <sub>OH</sub>	HIGH Level Output Current			-0.4	mA
I <sub>OL</sub>	LOW Level Output Current			8	mA
T <sub>A</sub>	Free Air Operating Temperature	0		70	°C

**Electrical Characteristics**

over recommended operating free air temperature range (unless otherwise noted)

Symbol	Parameter	Conditions	Min	Typ (Note 2)	Max	Units
V <sub>I</sub>	Input Clamp Voltage	V <sub>CC</sub> = Min, I <sub>I</sub> = -18 mA			-1.5	V
V <sub>OH</sub>	HIGH Level Output Voltage	V <sub>CC</sub> = Min, I <sub>OH</sub> = Max, V <sub>IL</sub> = Max, V <sub>IH</sub> = Min	2.7	3.4		V
V <sub>OL</sub>	LOW Level Output Voltage	V <sub>CC</sub> = Min, I <sub>OL</sub> = Max, V <sub>IL</sub> = Max, V <sub>IH</sub> = Min		0.35	0.5	V
I <sub>I</sub>	Input Current @ Max Input Voltage	I <sub>OL</sub> = 4 mA, V <sub>CC</sub> = Min		0.25	0.4	mA
I <sub>IH</sub>	HIGH Level Input Current	V <sub>CC</sub> = Max, V <sub>I</sub> = 2.7V			40	μA
I <sub>IL</sub>	LOW Level Input Current	V <sub>CC</sub> = Max, V <sub>I</sub> = 0.4V			-0.6	mA
I <sub>OS</sub>	Short Circuit Output Current	V <sub>CC</sub> = Max (Note 3)	-20		-100	mA
I <sub>CCH</sub>	Supply Current with Outputs HIGH	V <sub>CC</sub> = Max (Note 4)		6.1	10	mA
I <sub>CCL</sub>	Supply Current with Outputs LOW	V <sub>CC</sub> = Max (Note 5)		9	15	mA

**Note 2:** All typicals are at V<sub>CC</sub> = 5V, T<sub>A</sub> = 25°C.

**Note 3:** Not more than one output should be shorted at a time, and the duration should not exceed one second.

**Note 4:** I<sub>CCH</sub> is measured with all outputs OPEN, one input at each gate at 4.5V, and the other inputs grounded.

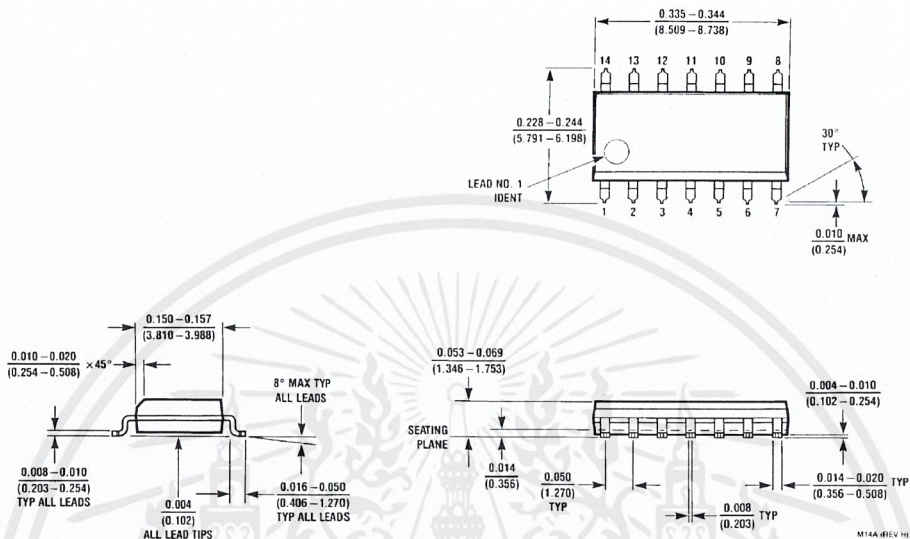
**Note 5:** I<sub>CCL</sub> is measured with all outputs OPEN and all inputs grounded.

**Switching Characteristics**

at V<sub>CC</sub> = 5V and T<sub>A</sub> = 25°C

Symbol	Parameter	Conditions	R <sub>L</sub> = 2 kΩ				Units
			C <sub>L</sub> = 15 pF		C <sub>L</sub> = 50 pF		
			Min	Max	Min	Max	
t <sub>PLH</sub>	Propagation Delay Time LOW-to-HIGH Level Output	Other Input		18		23	ns
t <sub>PHL</sub>	Propagation Delay Time HIGH-to-LOW Level Output	Low		17		21	ns
t <sub>PLH</sub>	Propagation Delay Time LOW-to-HIGH Level Output	Other Input		10		15	ns
t <sub>PHL</sub>	Propagation Delay Time HIGH-to-LOW Level Output	High		12		15	ns

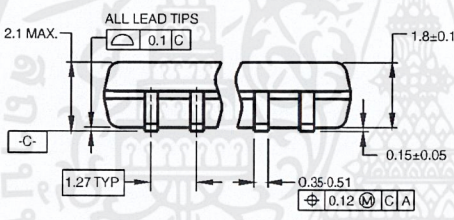
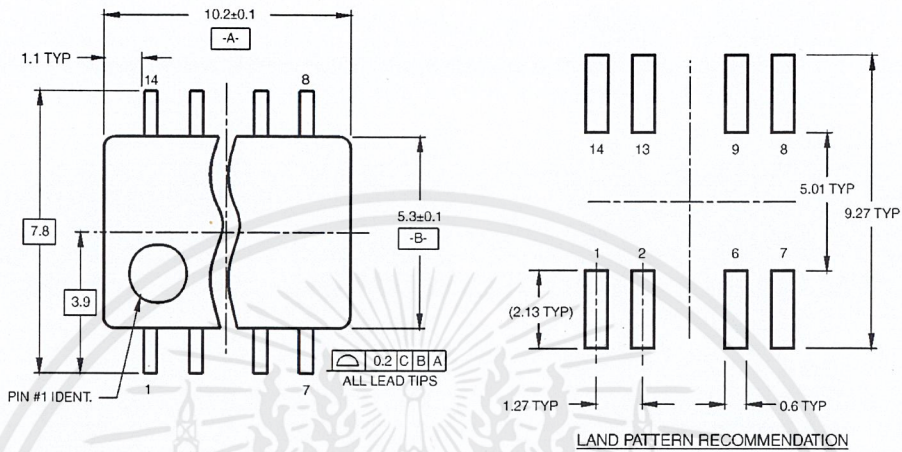
**Physical Dimensions** inches (millimeters) unless otherwise noted



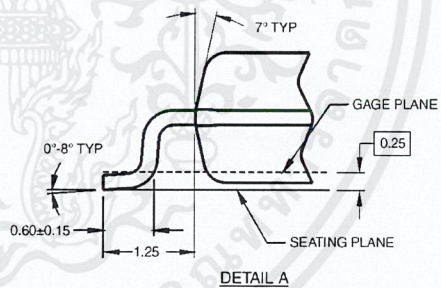
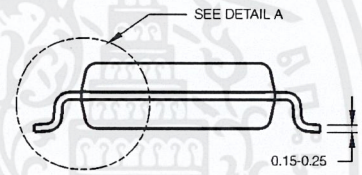
14-Lead Small Outline Integrated Circuit (SOIC), JEDEC MS-120, 0.150 Narrow  
Package Number M14A

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

**Physical Dimensions** inches (millimeters) unless otherwise noted (Continued)



DIMENSIONS ARE IN MILLIMETERS



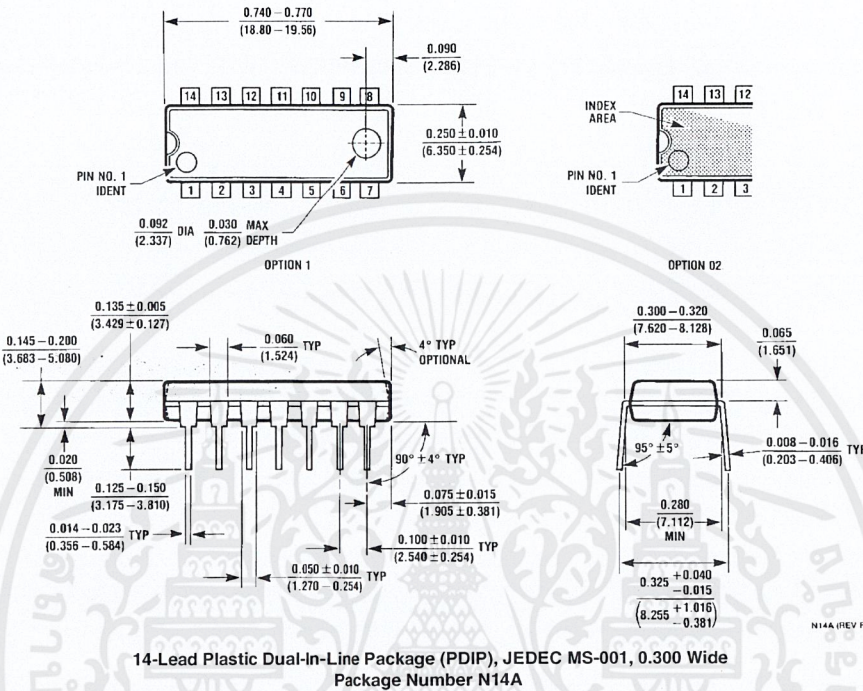
- NOTES:  
 A. CONFORMS TO EIAJ EDR-7320 REGISTRATION, ESTABLISHED IN DECEMBER, 1998.  
 B. DIMENSIONS ARE IN MILLIMETERS.  
 C. DIMENSIONS ARE EXCLUSIVE OF BURRS, MOLD FLASH, AND TIE BAR EXTRUSIONS.

M14DRevB1

**14-Lead Small Outline Package (SOP), EIAJ TYPE II, 5.3mm Wide  
 Package Number M14D**

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
 ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

**Physical Dimensions** inches (millimeters) unless otherwise noted (Continued)



Fairchild does not assume any responsibility for use of any circuitry described, no circuit patent licenses are implied and Fairchild reserves the right at any time without notice to change said circuitry and specifications.

**LIFE SUPPORT POLICY**

FAIRCHILD'S PRODUCTS ARE NOT AUTHORIZED FOR USE AS CRITICAL COMPONENTS IN LIFE SUPPORT DEVICES OR SYSTEMS WITHOUT THE EXPRESS WRITTEN APPROVAL OF THE PRESIDENT OF FAIRCHILD SEMICONDUCTOR CORPORATION. As used herein:

1. Life support devices or systems are devices or systems which, (a) are intended for surgical implant into the body, or (b) support or sustain life, and (c) whose failure to perform when properly used in accordance with instructions for use provided in the labeling, can be reasonably expected to result in a significant injury to the user.
2. A critical component in any component of a life support device or system whose failure to perform can be reasonably expected to cause the failure of the life support device or system, or to affect its safety or effectiveness.

[www.fairchildsemi.com](http://www.fairchildsemi.com)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



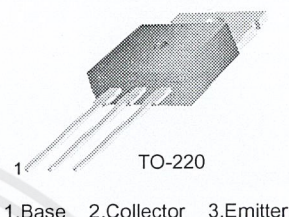
**ภาคผนวก ช:**  
**ดาต้าชีทของทรานซิสเตอร์ TIP120**

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## TIP120/121/122

### Medium Power Linear Switching Applications

- Complementary to TIP125/126/127

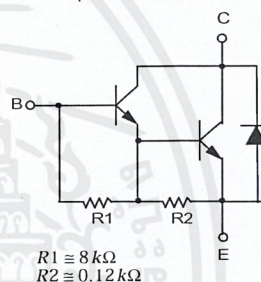


### NPN Epitaxial Darlington Transistor

#### Absolute Maximum Ratings $T_C=25^\circ\text{C}$ unless otherwise noted

Symbol	Parameter	Value	Units
$V_{CB0}$	Collector-Base Voltage : TIP120	60	V
	: TIP121	80	V
	: TIP122	100	V
$V_{CEO}$	Collector-Emitter Voltage : TIP120	60	V
	: TIP121	80	V
	: TIP122	100	V
$V_{EBO}$	Emitter-Base Voltage	5	V
$I_C$	Collector Current (DC)	5	A
$I_{CP}$	Collector Current (Pulse)	8	A
$I_B$	Base Current (DC)	120	mA
$P_C$	Collector Dissipation ( $T_a=25^\circ\text{C}$ )	2	W
	Collector Dissipation ( $T_C=25^\circ\text{C}$ )	65	W
$T_J$	Junction Temperature	150	$^\circ\text{C}$
$T_{STG}$	Storage Temperature	- 65 ~ 150	$^\circ\text{C}$

Equivalent Circuit



#### Electrical Characteristics $T_C=25^\circ\text{C}$ unless otherwise noted

Symbol	Parameter	Test Condition	Min.	Max.	Units
$V_{CE0(sus)}$	Collector-Emitter Sustaining Voltage	$I_C = 100\text{mA}, I_B = 0$	60		V
	: TIP121		80		V
	: TIP122		100		V
$I_{CEO}$	Collector Cut-off Current	$V_{CE} = 30\text{V}, I_B = 0$ $V_{CE} = 40\text{V}, I_B = 0$ $V_{CE} = 50\text{V}, I_B = 0$		0.5	mA
	: TIP120			0.5	mA
	: TIP121			0.5	mA
$I_{CBO}$	Collector Cut-off Current	$V_{CB} = 60\text{V}, I_E = 0$ $V_{CB} = 80\text{V}, I_E = 0$ $V_{CB} = 100\text{V}, I_E = 0$		0.2	mA
	: TIP120			0.2	mA
	: TIP121			0.2	mA
$I_{EBO}$	Emitter Cut-off Current	$V_{BE} = 5\text{V}, I_C = 0$		2	mA
$h_{FE}$	* DC Current Gain	$V_{CE} = 3\text{V}, I_C = 0.5\text{A}$	1000		
		$V_{CE} = 3\text{V}, I_C = 3\text{A}$	1000		
$V_{CE(sat)}$	* Collector-Emitter Saturation Voltage	$I_C = 3\text{A}, I_B = 12\text{mA}$		2.0	V
		$I_C = 5\text{A}, I_B = 20\text{mA}$		4.0	V
$V_{BE(on)}$	* Base-Emitter ON Voltage	$V_{CE} = 3\text{V}, I_C = 3\text{A}$		2.5	V
$C_{ob}$	Output Capacitance	$V_{CB} = 10\text{V}, I_E = 0, f = 0.1\text{MHz}$		200	pF

\* Pulse Test :  $PW \leq 300\mu\text{s}$ , Duty cycle  $\leq 2\%$

## Typical characteristics

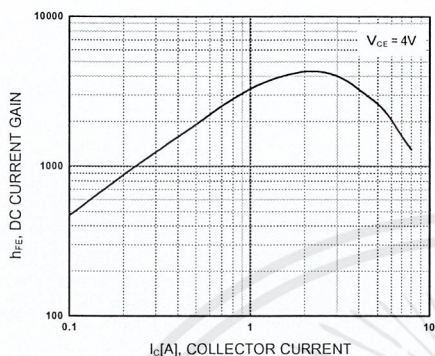


Figure 1. DC current Gain

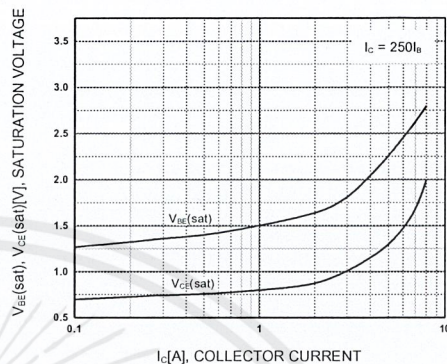


Figure 2. Base-Emitter Saturation Voltage  
Collector-Emitter Saturation Voltage

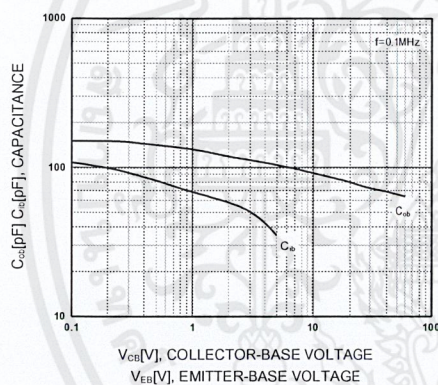


Figure 3. Output and Input Capacitance  
vs. Reverse Voltage

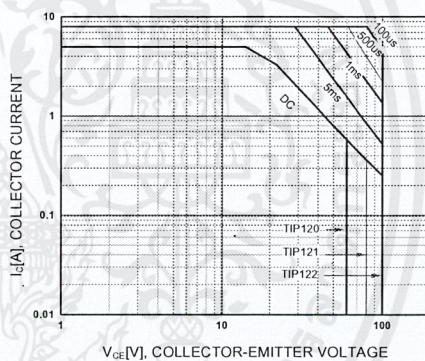


Figure 4. Safe Operating Area

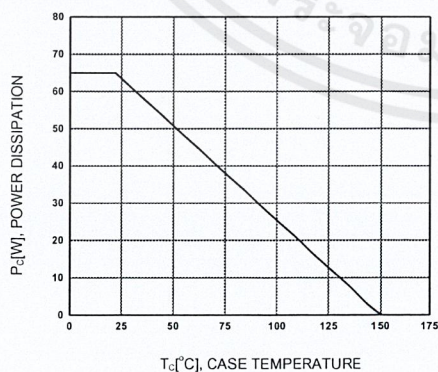


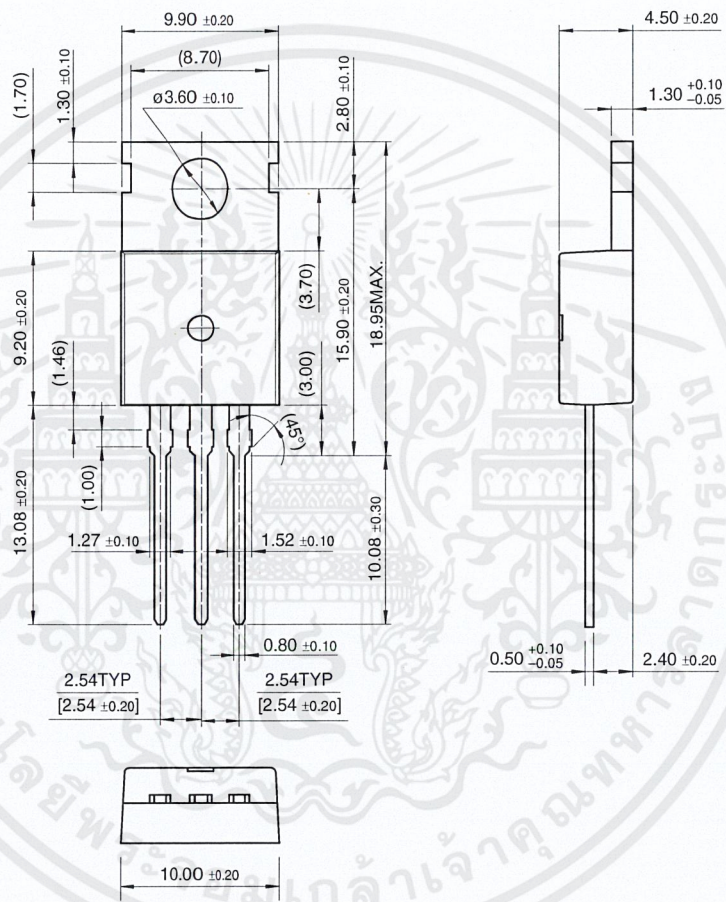
Figure 5. Power Derating

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

# Package Demensions

TIP120/121/122

## TO-220



Dimensions in Millimeters

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## TRADEMARKS

The following are registered and unregistered trademarks Fairchild Semiconductor owns or is authorized to use and is not intended to be an exhaustive list of all such trademarks.

ACE <sup>x</sup> ™	FAST <sup>®</sup>	OPTOPLANAR™	STAR*POWER™
Bottomless™	FASTr™	PACMAN™	Stealth™
CoolFET™	FRFET™	POP™	SuperSOT™-3
CROSSVOLT™	GlobalOptoisolator™	Power247™	SuperSOT™-6
DenseTrench™	GTO™	PowerTrench <sup>®</sup>	SuperSOT™-8
DOME™	HiSeC™	QFET™	SyncFET™
EcoSPARK™	ISOPLANAR™	QS™	TruTranslation™
E <sup>2</sup> CMOS™	LittleFET™	QT Optoelectronics™	TinyLogic™
EnSigna™	MicroFET™	Quiet Series™	UHC™
FACT™	MICROWIRE™	SLIENT SWITCHER <sup>®</sup>	UltraFET <sup>®</sup>
FACT Quiet Series™	OPTOLOGIC™	SMART START™	VCX™

STAR\*POWER is used under license

## DISCLAIMER

FAIRCHILD SEMICONDUCTOR RESERVES THE RIGHT TO MAKE CHANGES WITHOUT FURTHER NOTICE TO ANY PRODUCTS HEREIN TO IMPROVE RELIABILITY, FUNCTION OR DESIGN. FAIRCHILD DOES NOT ASSUME ANY LIABILITY ARISING OUT OF THE APPLICATION OR USE OF ANY PRODUCT OR CIRCUIT DESCRIBED HEREIN; NEITHER DOES IT CONVEY ANY LICENSE UNDER ITS PATENT RIGHTS, NOR THE RIGHTS OF OTHERS.

## LIFE SUPPORT POLICY

FAIRCHILD'S PRODUCTS ARE NOT AUTHORIZED FOR USE AS CRITICAL COMPONENTS IN LIFE SUPPORT DEVICES OR SYSTEMS WITHOUT THE EXPRESS WRITTEN APPROVAL OF FAIRCHILD SEMICONDUCTOR CORPORATION.

As used herein:

1. Life support devices or systems are devices or systems which, (a) are intended for surgical implant into the body, or (b) support or sustain life, or (c) whose failure to perform when properly used in accordance with instructions for use provided in the labeling, can be reasonably expected to result in significant injury to the user.
2. A critical component is any component of a life support device or system whose failure to perform can be reasonably expected to cause the failure of the life support device or system, or to affect its safety or effectiveness.

## PRODUCT STATUS DEFINITIONS

### Definition of Terms

Datasheet Identification	Product Status	Definition
Advance Information	Formative or In Design	This datasheet contains the design specifications for product development. Specifications may change in any manner without notice.
Preliminary	First Production	This datasheet contains preliminary data, and supplementary data will be published at a later date. Fairchild Semiconductor reserves the right to make changes at any time without notice in order to improve design.
No Identification Needed	Full Production	This datasheet contains final specifications. Fairchild Semiconductor reserves the right to make changes at any time without notice in order to improve design.
Obsolete	Not In Production	This datasheet contains specifications on a product that has been discontinued by Fairchild semiconductor. The datasheet is printed for reference information only.



**ภาคผนวก ญ:**  
**ฟังก์ชัน API ของ Logitech SDK 1.0**

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



**Logitech®**

**Logitech QuickCam® SDK 1.0  
Microsoft® MFC Programmer's Guide**

**Logitech Confidential**

**QuickCam Team  
Logitech, USA**

© 2000 Logitech. All rights reserved. Logitech, the Logitech logo, and other Logitech marks are owned by Logitech and may be registered. All other trademarks are the property of their respective owners.

# chapter one

## Introduction

With the emergence of new technology such as MPEG4 and Internet video broadcasting, computer cameras have finally become more of a tool than a novelty. Logitech, the world leader in computer camera sales, decided to provide a useful tool for software developers to easily control and utilize its family of QuickCam® cameras. From this decision, the QuickCam® SDK was created. In the following pages you will learn how to develop your own camera-based application, without the need to learn complex low-level camera interfaces. This document includes information about developing camera-based applications in MFC.

## 1.1 Purpose

The Logitech QuickCam® SDK is your backstage pass to developing digital camera applications for Logitech QuickCam® cameras. Traditional camera-based applications include complicated Video for Windows® (VFW) or DirectShow® programming instructions. Additionally, from these interfaces only one camera connection is allowed at any given time. The Logitech QuickCam® SDK solves this single-connection issue and allows any number of simultaneous camera connections through an easy-to-use COM interface.

## 1.2 Features

- **Window-based User Interface**  
A windowed user-interface with live video preview and easy to access camera settings is available. Additionally, each user-interface element can be hidden or shown programmatically.
- **Access to Camera Settings**  
Modify brightness, contrast, hue, exposure, image size, and many other camera settings programmatically, or with a provided camera settings dialog.
- **Access to Real-Time video data**  
For power users, access to real-time video data is available. This powerful feature allows fast, direct access to video data. You can develop your own motion detector, or stream video data through a video conferencing engine easily with this feature.
- **Movie Recording**  
Recording movies is easy with numerous recording properties at your disposal. Change the frame-rate, video compressor, audio compressor, and other properties with simple COM properties.
- **Taking Pictures**  
Pictures can be saved either to disk or to an application supplied memory buffer.
- **Text Overlay**  
Program-specified text can be overlaid on pictures and movies. You can select text color, opaque or transparent backgrounds, font, font size, and many other options.
- **Camera Notifications**  
Receive camera plug and un-plug notifications as well as a variety of other useful notification events. These events help insure that your application is up-to-date with camera activity at all times.
- **Multiple Simultaneous Camera Connections**  
This power feature is the primary reason the QuickCam® SDK exists. The QuickCam® SDK creates virtual camera instances for a single camera both efficiently and transparently. This feature allows you, as the developer, to create multiple simultaneous connections to a camera without worry.
- **Motion Detection**  
Easily incorporate motion detection into your application. This bonus feature, once enabled, notifies your application with motion analysis information.

### 1.3 Where Applicable

The Logitech QuickCam® SDK allows programmers working in C++, MFC, Microsoft Visual Basic® and other high-level languages to easily develop high-performance camera-based applications. There are basically three approaches to programming a digital video camera under Microsoft Windows®:

1. Video for Windows®
  - + De facto standard for video capture on Windows®.
  - No access to camera-specific features.
  - Not easy to learn.
  - Minimal: Many common tasks require additional programming.
2. DirectShow®
  - Not easy to learn.
  - Minimal: Many common tasks require additional programming.
3. Logitech QuickCam® SDK
  - + Full support of all camera features.
  - + Highly accessible from many languages.
  - + Simple interface to learn and use.
  - + Advanced features such as text overlaying and motion detection.
  - + Allows multiple simultaneous camera connections
  - Proprietary: Will only work with Logitech QuickCam® products.

### 1.4 Run-Time Requirements

The Logitech QuickCam® SDK can be used in the following environments:

- Windows® 98
- Windows® 2000
- Windows® Millenium

Minimum System Requirements:

- Logitech QuickCam® video camera
- Pentium® 200 Mhz MMX (266 MHz or higher and MMX technology recommended)
- 16 MB RAM
- Hard Drive with 100 MB free space
- 16-bit color displays (thousands of colors)
- Windows-compatible sound card (full duplex sound card recommended) and mouse
- Speakers required for receiving audio

## 1.5 QuickCam® SDK Software Architecture

The QuickCam® SDK depends on the interaction of several components at run-time, although your application only communicates with one of them.

- Logitech QuickCam® Video Portal Component**  
 This component implements the COM interface described in this document. It is a windowed in-process COM object with a video preview area and various UI elements any of which can be enabled or disabled at run-time. Complete camera control is available including contrast, brightness, and gain. Videos can be recorded using any Video for Windows installable compressor. Pictures can be saved to either memory or disk. Direct real-time access to video streaming data is also available. There are many other features and subsets of the above features available from the video portal.
- Logitech QuickCam® Video Server Component**  
 The Video Server is invisible to your application. It serves to coordinate between multiple instances of the Video Portal, and to manage and centralize their communication with the Video for Windows subsystem. The Video Server also communicates with another Logitech component called LVCom, which communicates directly with the WDM camera driver.

The Video Server interface is not described in this document. Future versions of the QuickCam® SDK may contain information on the Video Server interface.

LVCom is installed as part of the camera product software, you do not have to include it or install it with your application.

The following figure shows the various camera software components and how they are related at run-time:

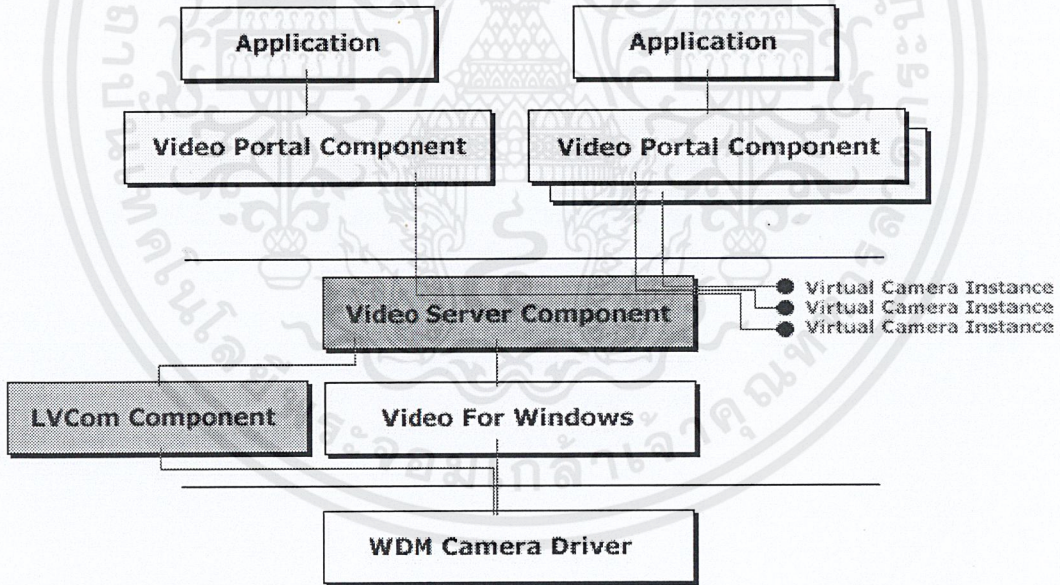


Figure 1 – Camera Software Components At Run-Time

# chapter two

## Getting Started with QuickCam® SDK

### 2.1 Installing the SDK

If you are reading this, there is a good chance you have already installed the QuickCam® SDK. If you haven't, you will need to obtain the SDK from Logitech. It is packaged as a downloadable self-extracting archive file: QCSDK1.EXE.

The QuickCam® SDK can be downloaded from the Logitech website <http://developer.logitech.com>. Follow the instructions from this web site to install the SDK. If you have any questions or problems installing the QuickCam® SDK, see the <http://developer.logitech.com> web site for assistance.

Once the QuickCam® SDK is installed the following folders are created:

```
\QCSDK1
  \Docs
  \Inc
  \Redist
  \Samples
    \MFC
      \BadgeMaker
      \HydraQC
      \StudioZero
      \TextOver
      \Vidbert
      \bin
    \Visual Basic
      \VBDemo
      \VBMotion
      \bin
    \Win32
      \Feedback
      \bin
```

### 2.2 Removing the SDK

To remove the QuickCam® SDK, use the **Control Panel** "Add/Remove Programs" feature of Windows.

### 2.3 Sample Code

The SDK includes sample code for Microsoft Visual C++® (Win32), Microsoft Visual C++® (MFC), and Microsoft Visual Basic® 5.0. All of this code is in \QCSDK1\Samples, in subdirectories named **WIN32**, **MFC** and **Visual Basic** respectively. Compiling the C++ (Win32) and MFC samples requires Microsoft Visual C++® 6.0. Compiling the Visual Basic samples require Microsoft Visual Basic® 5.0.

#### Microsoft Visual C++® (Win32) sample application

There is one **C++ (Win32)** sample that demonstrates using the QuickCam® SDK without a MFC (Microsoft Foundation Classes) dependency: Feedback. This sample includes header files from \QCSDK1\inc.

- **Feedback** - This C++ (Win32) sample demonstrates the QuickCam® SDK's motion detection capabilities. A simple motion magnitude graph is drawn depending on the amount of motion detected.

### Microsoft Visual C++ (MFC) sample applications

There are five **MFC** samples that demonstrate using the QuickCam® SDK via MFC (Microsoft Foundation Classes).

- **BadgeMaker** – This MFC sample demonstrates using the QuickCam® SDK to create photo ID badges. This sample uses the **PictureToMemory** method to save a picture into a memory buffer and overlay a text-stamp date onto the picture.
- **HydraQC** – This MFC sample demonstrates using the QuickCam® SDK to dynamically switch between multiple cameras.
- **StudioZero**– This MFC sample demonstrates using the QuickCam® SDK to record a movie.
- **TextOver**– This MFC sample demonstrates the QuickCam® SDK text overlay properties to apply a textual string to a picture.
- **Vidbert**– This MFC sample demonstrates the QuickCam® SDK video notification callback mechanism (see **StartVideoHook**). Various effects modify the live video image received in the video callback.

### Microsoft Visual Basic sample applications

There are two **Visual Basic** samples that demonstrate using the QuickCam SDK.

- **VBDemo** – This Visual Basic sample demonstrates the QuickCam SDK's motion detection capabilities. A simple motion magnitude graph is drawn depending on the amount of motion detected.
- **VBMotion** – This Visual Basic sample demonstrates several of the QuickCam SDK's features including movie recording, picture taking, and how to enable/disable the LED light on the camera.

## 2.4 Redistributable Files

The \QCSDK1\Redist folder of the QuickCam® SDK contains the installation program and necessary files to re-distribute the QuickCam® SDK. This setup program must be executed on all machines which use the QuickCam® SDK.

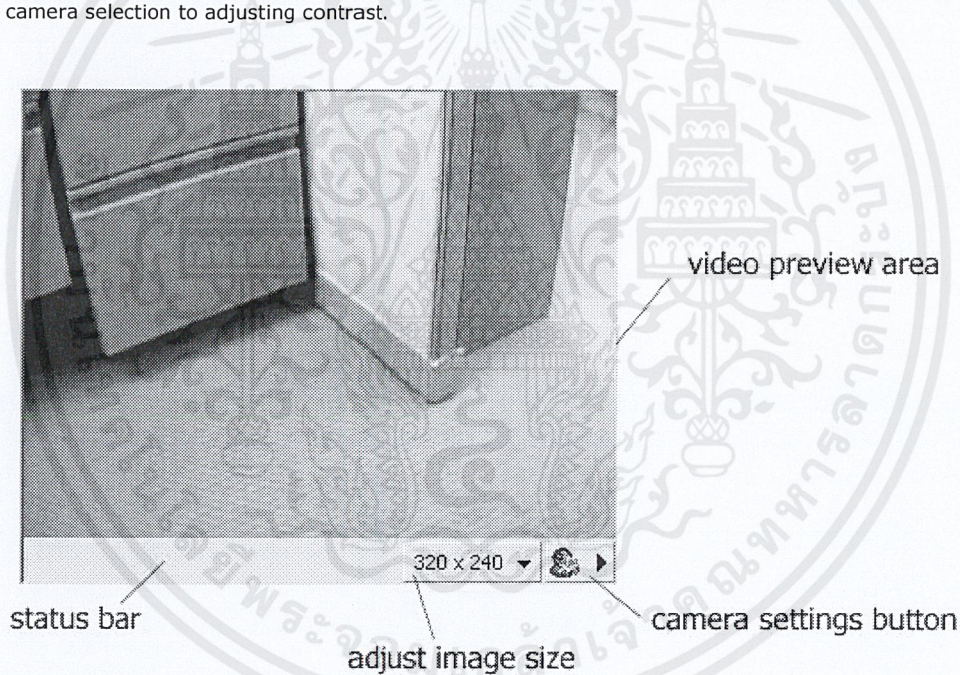
# chapter three

## Using the Video Portal API

This chapter begins with a tour of the visual aspects of the video portal control and continues on with some basic functionality. In this chapter you will learn how to create an instance of the video portal and start previewing live video. Examples of enabling and disabling user-interface elements are addressed. We also discuss saving images to disk and memory, recording movies, and creating text overlays.

### 3.1 User Interface Tour

The following diagram shows an example video portal window. There are only two main visual elements for the portal window: the preview area, and the status bar. The preview area is where live video is displayed. Within the status bar there are three main elements: a text area, an image size button, and a camera settings button. The text area of the status bar can show various actions in progress. Additionally, an application can send specific text messages to be displayed in this area. The image size button allows quick and easy access to modifying the image size received from the camera. Finally, the camera settings button allows users to quick access to camera settings from camera selection to adjusting contrast.



### 3.2 Getting Started

This is where we begin examining how to instantiate the video portal control and display live video on the screen. Getting video started with the video portal is easy.

1. Our first step however is to create a **MFC Appwizard.exe** project using Microsoft Visual Studio 6.0®. Create the project based upon a *dialog* box.
2. Select the "**Components and Controls...**" menu item (Project->Add to Project->Components and Controls...). The "Components and Controls Gallery" should appear.

3. From the Components and Controls Gallery dialog box, select "Registered ActiveX Controls" and a list of ActiveX controls appear. Choose the "VideoPortal Class" and click the "Insert" button. A prompt appears confirming the insertion of the control into the project, choose "OK".
4. Next, a "Confirm Classes" dialog box appears allowing you to customize the file names containing the source code for the video portal control. Accept the given filenames and choose "OK".
5. The video portal control is now added to your project. Next drag and drop the video portal control from the "Controls" resource dialog box into your dialog box resource.
6. Now we need to associate a member variable in your dialog box with the newly added video portal control. Using the MFC Classwizard, add a member variable named **m\_cVideo**.
7. The video control depends on including the "LVServerDefs.H" header file. LVServerDefs.H is located in \QCSDK1\inc. Specify the \QCSDK1\inc directory in the preprocessor "Additional include directories" setting. Select the Project->Settings menu item to display the "Project Settings" dialog box. Select the C/C++ tab and choose "Preprocessor" in the category combo box. In the "Additional include directories" edit box, add the following: **\QCSDK1\inc**
8. Insert the following line of code into your dialog box implementation file:

```
#include "LVServerDefs.H"
```

9. Next, we insert the code necessary to start using the video control. Insert the following lines into the **OnInitDialog** method of the dialog box implementation file:

```
char sReg[] = "HKEY_LOCAL_MACHINE\\Software\\MyApp";
m_cVideo.PrepareControl("MyApp1", sReg, 0 );

m_cVideo.EnableUIElements(UIELEMENT_STATUSBAR, 0, TRUE);
m_cVideo.ConnectCamera2();

m_cVideo.SetEnablePreview(TRUE);
```

10. The following summarizes what changes have been made to the dialog box implementation file:

```
#include "LVServerDefs.H"
.
.
.
.
CmyDialog::OnInitDialog()
{
    CDialog::OnInitDialog();

    char sReg[] = "HKEY_LOCAL_MACHINE\\Software\\MyApp";
    m_cVideo.PrepareControl("MyApp1", sReg, 0 );

    m_cVideo.EnableUIElements(UIELEMENT_STATUSBAR, 0, TRUE);
    m_cVideo.ConnectCamera2();

    m_cVideo.SetEnablePreview(TRUE);

    return TRUE;
}
```

11. Next, compile your project and execute it.

Let's take the major steps in this code one by one.

### Initialize the Video Portal Interface

The first video portal method called is **PrepareControl**. This method establishes a connection from the video portal to the video server. The **PrepareControl** method must always be called before any other video portal method. It specifies a name, a registry key, and initialization flags. The most interesting parameter is the name. All instances of the video portal with the same name share settings. For this reason, it is recommended that the name is carefully chosen. The video portal uses the registry key to save specific settings, such as image size and PictureSmart™ information. The video portal does not save global camera settings such as brightness, contrast and brightness.

### Configure the Portal User Interface

Next, we call the **EnableUIElements** method to enable the status bar. There are several visual elements, which can be enabled or disabled. For instance, by default the following images sizes are supported: 160x120, 320x240, and 640x480. (Note the QuickCam® Express camera does not support 640x480). You can forbid a user from switching to 320x240 by using the **EnableUIElements** method and passing UIELEMENT\_320x240, with a status of FALSE. For other examples, see the reference section on **EnableUIElements**.

### Connect to a camera

We are now ready to establish a connection to a QuickCam® camera. The recommended method to do this is **ConnectCamera2**. It establishes a camera connection with any camera that is currently in use by another portal instance. If there is not a current camera connection, the last camera used is chosen. There is an additional method **ConnectCamera** that allows connecting to a specific camera. However, this method should be used with caution, since all video portal instances *share the same camera*. Connecting to a specific camera will cause all video portals to start using that camera.

### Open the video preview window

Once a camera connection has been successfully established, we are ready to turn on the video preview display. The **SetEnablePreview** method enables video preview.

## 3.3 Taking Pictures

Next we examine how to use the video portal for saving pictures either to a file or into a memory buffer. First we will examine taking a picture and saving it to a file. The following code fragment demonstrates how this is accomplished.

```
#include "LVServerDefs.H"
.
.
BOOL CmyDialog::SaveImageToFile(void)
{
    char sFile[] = "c:\\image_test.bmp";
    return m_cVideo.PictureToFile( 0, 24, sFile, NULL );
}
```

Currently, the only supported format for saving images to both memory and disk is 24 bit RGB, see the reference section on **PictureToFile** for more information. Next we examine saving a picture to a memory buffer. The video portal saves into memory a.BMP file as it would exist on disk. In the prototype below, we first call **PictureToMemory** to receive the number of bytes required to store the image. After allocating a buffer of this size, we call **PictureToMemory** a second time to perform the actual save operation.

```
BOOL CmyDialog::SaveImageToMemory(void)
{
    long lSize;
    if ( !m_cVideo.PictureToMemory(0, 24, 0, &lSize, NULL) )
    {
        return FALSE;
    }
}
```

```

BYTE * pBuffer = new BYTE[lSize];

if ( !m_cVideo.PictureToMemory( 0, 24, (long)pBuffer, &lSize, NULL ) )
{
    delete []pBuffer; // delete the memory we allocated...
    return FALSE;
}
.
.
//pBuffer now contains a .BMP file in memory
.
.
delete []pBuffer;
return TRUE;
}

```

### 3.4 Overlay Text

Our next task is to demonstrate how to overlay text on an image. You can specify color, font, font size, indicate whether or not to use a text shadow, and finally whether the overlay is opaque or transparent.

```

#include "LVServerDefs.H"
.
.
BOOL CmyDialog::SaveImageToFile(void)
{
    char sFile[] = "c:\\image_test_overlay.bmp";
    char sTextStamp[] = "Monday April 4, 2000";

    m_cVideo.SetStampTextColor( RGB( 255,255,255 ) );
    m_cVideo.SetStampFontName( "Arial" );
    m_cVideo.SetStampPointSize( 10 );
    m_cVideo.SetStampTextShadow( FALSE );
    m_cVideo.SetStampTransparentBackground( TRUE );

    return m_cVideo.PictureToFile( 0, 24, sFile, sTextStamp );
}

```

The function prototype above demonstrates how to specify text stamp properties. Once the text stamp properties have been set, you do not need to re-set them for each picture. The image below depicts the image saved to disk with these parameters.



### 3.5 Recording Movies

Recording movies is a simple task when using the Video Portal. You have the option to specify a variety of recording properties. It should be noted here that only one movie can be recorded at any given time, and while a movie is being recorded, no pictures can be saved to either memory or to disk.

By default all movies are saved to disk at the fastest possible frame rate, which varies with the type of camera used.

The following codes illustrates how to start and stop a movie recording session. This example is fairly straightforward and uses the default recording properties supplied by the Video Portal.

```
#include "LVServerDefs.H"
.
void CMyDialog::OnStartMovie( void )
{
    char sMovieFile[] = "c:\\sample_movie_1.avi";
    if ( !m_cVideo.StartMovieRecording(sMovieFile, NULL ) )
        return;
}

void CMyDialog::OnStopMovie( void )
{
    m_cVideo.StopMovieRecording();
}
```

### 3.6 Animation (step-capture, time-lapse) Movie Recording

Our next example is slightly more difficult than the one above. We take a look at how to record a 'stop motion' or 'time-lapse' movie. Essentially, it's the same process as above, except a Video Portal method must be called for each frame to add to the movie. The Video Portal maintains a *recording-mode*. In the example above the mode was fast-as-possible, which is its default mode. For stop-motion and time-lapse recording, we use the *manual triggered mode*.

```
#include "LVServerDefs.H"
.
void CMyDialog::OnStartStepCapture( void )
{
    char sMovieFile[] = "c:\\sample_stepcapture.avi";

    // Select the manual triggered recording mode:
    m_cVideo.SetMovieRecordMode(STEPCAPTURE_MANUALTRIGGERED);

    // Specify the playback rate for this movie - 15 FPS:
    m_cVideo.SetMoviePlaybackFPS( 15 );

    // Doesn't make sense to record audio in this mode:
    m_cVideo.SetMovieRecordAudio(FALSE);

    if ( !m_cVideo.StartMovieRecording(sMovieFile, NULL ) )
        return;
}

void CMyDialog::OnAddFrame( void )
{
    m_cVideo.StepCaptureAddFrame();
}

void CMyDialog::OnStopMovie( void )
{
    m_cVideo.StopMovieRecording();
}
```

# chapter four

## Video Portal Interface Reference

The pages that follow list each method, property and notification event for the video portal. This is the first SDK release of the video portal interface. Future versions of the video portal will be backward compatible with this interface.

Methods

<ul style="list-style-type: none"><li><a href="#">PrepareControl</a></li><li><a href="#">GetCameraCount</a></li><li><a href="#">GetCameraDescription</a></li><li><a href="#">GetCameraType</a></li><li><a href="#">QueryCameraConnected</a></li><li><a href="#">QueryRegistryCameraIndex</a></li><li><a href="#">GetCameraState</a></li><li><a href="#">ConnectCamera</a></li><li><a href="#">ConnectCamera2</a></li><li><a href="#">DisconnectCamera</a></li><li><a href="#">LoadRegistrySettings</a></li><li><a href="#">SaveRegistrySettings</a></li><li><a href="#">SetCameraPropertyLong</a></li><li><a href="#">GetCameraPropertyLong</a></li><li><a href="#">PictureToFile</a></li><li><a href="#">PictureToMemory</a></li><li><a href="#">StartMovieRecording</a></li><li><a href="#">StopMovieRecording</a></li><li><a href="#">StepCaptureAddFrame</a></li><li><a href="#">MovieRecordWriteSingleFrame</a></li><li><a href="#">SetVideoFormat</a></li><li><a href="#">GetVideoFormat</a></li><li><a href="#">GetLastError</a></li><li><a href="#">ShowCameraDlg</a></li><li><a href="#">EnableUIElements</a></li><li><a href="#">StartVideoHook</a></li><li><a href="#">StopVideoHook</a></li></ul>	<ul style="list-style-type: none"><li><a href="#">CameraConnected (get)</a></li><li><a href="#">CameraIndex (get)</a></li><li><a href="#">PictureSound (get/put)</a></li><li><a href="#">StampTextColor (get/put)</a></li><li><a href="#">StampFontName (get/put)</a></li><li><a href="#">StampPointSize (get/put)</a></li><li><a href="#">StampTextShadow (get/put)</a></li><li><a href="#">StampTextShadowColor (get/put)</a></li><li><a href="#">StampTransparentBackground (get/put)</a></li><li><a href="#">StampBackgroundColor (get/put)</a></li><li><a href="#">EnablePreview (get/put)</a></li><li><a href="#">MovieVideoCompressionFOURCC (get/put)</a></li><li><a href="#">MovieVideoCompressionKeyFrameInterval (get/put)</a></li><li><a href="#">MovieVideoCompressionQuality (get/put)</a></li><li><a href="#">MoviePlaybackFPS (get/put)</a></li><li><a href="#">MovieAudioSamplesPerSecond (get/put)</a></li><li><a href="#">MovieAudioChannels (get/put)</a></li><li><a href="#">MovieAudioBitsPerSample (get/put)</a></li><li><a href="#">MovieAudioCompressionFOURCC (get/put)</a></li><li><a href="#">MovieAudioCompressionQuality (get/put)</a></li><li><a href="#">MovieRecordAudio (get/put)</a></li><li><a href="#">MovieRecordMode (get/put)</a></li><li><a href="#">MovieCreateFlags (get/put)</a></li><li><a href="#">MovieRecordingActiveLocal (get)</a></li><li><a href="#">MovieRecordingActiveGlobal (get)</a></li><li><a href="#">CameraState (get)</a></li><li><a href="#">EnableMovieRecordErrorPrompt (get/put)</a></li><li><a href="#">EnablePictureDiskErrorPrompt (get/put)</a></li><li><a href="#">StatusBarText (put)</a></li><li><a href="#">PreviewMaxWidth (get/put)</a></li><li><a href="#">PreviewMaxHeight (get/put)</a></li></ul>
---	--

Notifications

<p>PortalNotification</p> <ul style="list-style-type: none"><li>NOTIFICATIONMSG_MOTION</li><li>NOTIFICATIONMSG_MOVIERECORDERROR</li><li>NOTIFICATIONMSG_CAMERADETACHED</li><li>NOTIFICATIONMSG_CAMERAREATTACHED</li><li>NOTIFICATIONMSG_IMAGESIZECHANGE</li><li>NOTIFICATIONMSG_CAMERAPRECHANGE</li><li>NOTIFICATIONMSG_CAMERACHANGEFAILED</li><li>NOTIFICATIONMSG_POSTCAMERACHANGED</li><li>NOTIFICATIONMSG_CAMERBUTTONCLICKED</li><li>NOTIFICATIONMSG_VIDEOHOOK</li><li>NOTIFICATIONMSG_SETTINGDLGCLOSED</li><li>NOTIFICATIONMSG_QUERYPRECAMERAMODIFICATION</li><li>NOTIFICATIONMSG_MOVIESIZE</li></ul>
---

# Camera Properties

## Camera Properties

The QuickCam SDK provides mechanisms to control camera properties. Each camera supported by the QuickCam SDK has various properties. Some of the camera properties are available in all Logitech cameras, while others are not. **Appendix A** contains a matrix indicating camera properties and the cameras which support them. All of these camera properties are accessed by the **SetCameraPropertyLong** and **GetCameraPropertyLong** methods.

- **PROPERTY\_ORIENTATION**  
This property controls the orientation of the image. It can be any of the following values:
  - **ORIENTATIONMODE\_NORMAL**  
Indicates normal orientation
  - **ORIENTATIONMODE\_MIRRORED**  
Indicates a mirrored orientation
  - **ORIENTATIONMODE\_FLIPPED**  
Indicates a flipped orientation
  - **ORIENTATIONMODE\_FLIPPED\_AND\_MIRRORED**  
Indicates a flipped and mirrored orientation
- **PROPERTY\_BRIGHTNESSMODE**  
This property controls auto-brightness mode. When this property is set to **ADJUSTMENT\_AUTOMATIC**, auto-brightness is engaged, otherwise auto-brightness is turned off. This property can be either **ADJUSTMENT\_MANUAL** or **ADJUSTMENT\_AUTOMATIC**.
- **PROPERTY\_BRIGHTNESS**  
This property is used to adjust brightness and is only available when auto-brightness is turned off (see **PROPERTY\_BRIGHTNESSMODE**). This property can range between 0..255.
- **PROPERTY\_CONTRAST**  
This property is used to adjust contrast. This property can range between 0..255.
- **PROPERTY\_COLORMODE**  
This property controls auto-color mode. When this property is set to **ADJUSTMENT\_AUTOMATIC**, auto-color is engaged; otherwise auto-color mode is turned off. This property can be either **ADJUSTMENT\_MANUAL** or **ADJUSTMENT\_AUTOMATIC**.
- **PROPERTY\_REDGAIN**  
This property adjusts red saturation and is only available when auto-color is turned off (see **PROPERTY\_COLORMODE**). This property can range between 0..255.
- **PROPERTY\_BLUEGAIN**  
This property adjusts red saturation and is only available when auto-color is turned off (see **PROPERTY\_COLORMODE**). This property can range between 0..255.
- **PROPERTY\_SATURATION**  
This property adjusts saturation. This property can range between 0..255.
- **PROPERTY\_EXPOSURE**  
This property adjusts exposure. This property can range between 0..255.
- **PROPERTY\_RESET**  
This property is can only be set. It is used to reset a camera to its hardware defaults.
- **PROPERTY\_ANTIBLOOM**  
This property enables or disables anti-bloom. Blooming occurs in some Logitech cameras when bright lights appear to cover the entire video image. This property can be either **TRUE** or **FALSE**.
- **PROPERTY\_LOWLIGHTFILTER**

This property enables or disables low-light filtering within the image. This property can be either TRUE or FALSE.

- **PROPERTY\_HUE**  
This property adjusts hue. This property can range between 0..255.
- **PROPERTY\_PORT\_TYPE**  
This read-only property is used to retrieve the type of camera port connection.
- **PROPERTY\_PICTSMART\_MODE**  
This property enables or disables PictureSmart™ mode. This property can be either TRUE or FALSE.
- **PROPERTY\_PICTSMART\_LIGHT**  
This property is used to indicate the light-filtering algorithm when PictureSmart™ is enabled. This property can be any of the following:
  - **PICTSMART\_LIGHTCORRECTION\_NONE**  
Indicates no light correction.
  - **PICTSMART\_LIGHTCORRECTION\_COOLFLORESCENT**  
Indicates a "blue" or "cool" florescent light.
  - **PICTSMART\_LIGHTCORRECTION\_WARMFLORESCENT**  
Indicates a "red" or "warm" florescent light.
  - **PICTSMART\_LIGHTCORRECTION\_OUTSIDE**  
Indicates an outside light.
  - **PICTSMART\_LIGHTCORRECTION\_TUNGSTEN**  
Indicates a tungsten light.
- **PROPERTY\_MOTION\_DETECTION\_MODE**  
This property is used to enable/disable the motion detector. The motion detector sends notification events via the PortalNotification event (see **Notifications**). This property can be either TRUE or FALSE.
- **PROPERTY\_MOTION\_SENSITIVITY**  
This property is used to set the sensitivity of the motion detector. This property can range between 0..255.
- **PROPERTY\_WHITELEVEL**  
This property is used adjust the whiteness of an image and is only available when auto-white leveling is turned off (see PROPERTY\_AUTO\_WHITELEVEL).
- **PROPERTY\_AUTO\_WHITELEVEL**  
This property controls auto-white leveling mode. When this property is set to ADJUSTMENT\_AUTOMATIC, the auto-white level is engaged; otherwise auto-level mode is turned off. This property can be either ADJUSTMENT\_MANUAL or ADJUSTMENT\_AUTOMATIC.
- **PROPERTY\_ANALOGGAIN**  
This property controls analog gain and is only available when auto-analog gain is turned off (see PROPERTY\_AUTO\_ANALOGGAIN). This property can range between 0..255.
- **PROPERTY\_AUTO\_ANALOGGAIN**  
This property controls auto-analog gain mode. When this property is set to ADJUSTMENT\_AUTOMATIC, the auto-analog gain is engaged; otherwise auto-analog gain mode is turned off. This property can be either ADJUSTMENT\_MANUAL or ADJUSTMENT\_AUTOMATIC.
- **PROPERTY\_LOWLIGHTBOOST**  
This property enables/disables low-light boosting. This property can be either TRUE or FALSE.
- **PROPERTY\_COLORBOOST**  
This property enables/disables color boosting. This property can be either TRUE or FALSE.
- **PROPERTY\_ANTIFLICKER**

This property sets the anti-flicker mode. This property can be any of the following values:

- **ANTIFLICKER\_OFF**  
Turns off anti-flicker
- **ANTIFLICKER\_50Hz**  
Sets anti-flicker to 50Hz
- **ANTIFLICKER\_60Hz**  
Sets anti-flicker to 60Hz

▪ **PROPERTY\_OPTIMIZATION\_SPEED\_QUALITY**

This property sets the optimization setting. This property can be either **OPTIMIZE\_QUALITY**, which indicates a higher quality image (slower), or **OPTIMIZE\_SPEED**, which indicates a faster lower quality image.

▪ **PROPERTY\_LED**

This property sets current state of a camera's LED light. This property can be any of the following values:

- **LED\_OFF**  
Turns the LED light off.
- **LED\_ON**  
Turns the LED light on.
- **LED\_AUTO**  
The default LED light mode, which turns the LED light on when the camera is in use, and off otherwise.
- **LED\_MAX**

# Appendix A

Logitech QuickCam Web  
 Logitech QuickCam Home  
 Logitech QuickCam Pro  
 Logitech QuickCam Express

PROPERTY	VALUE				
PROPERTY_ORIENTATION	ORIENTATIONMODE_NORMAL		•	•	•
	ORIENTATIONMODE_MIRRORED		•	•	•
	ORIENTATIONMODE_FLIPPED		•	•	•
	ORIENTATIONMODE_FLIPPED_AND_MIRRORED		•	•	•
PROPERTY_BRIGHTNESSMODE	ADJUSTMENT_MANUALADJUSTMENT_AUTOMATIC				
PROPERTY_BRIGHTNESS	0..255		•	•	•
PROPERTY_CONTRAST	0..255		•	•	•
PROPERTY_COLORMODE	ADJUSTMENT_MANUAL				
	ADJUSTMENT_AUTOMATIC				
PROPERTY_REDGAIN	0..255				
PROPERTY_BLUEGAIN	0..255				
PROPERTY_SATURATION	0..255		•	•	•
PROPERTY_EXPOSURE	0..255		•	•	•
PROPERTY_RESET	NO FLAGS		•	•	•
PROPERTY_ANTIBLOOM	TRUE/FALSE				
PROPERTY_LOWLIGHTFILTER	TRUE / FALSE				
PROPERTY_HUE	0..255		•	•	•
PROPERTY_PORT_TYPE	PORTTYPE_UNKNOWNPORTTYPE_LPT_NIBBLE				
	PORTTYPE_LPT_BIDIRECTIONAL				
	PORTTYPE_LPT_ECP				
	PORTTYPE_LPT_ECPDMA				
	PORTTYPE_USB		•	•	•
PROPERTY_PICTSMART_MODE	TRUE/FALSE			•	
PROPERTY_PICTSMART_LIGHT	PICTSMART_LIGHTCORRECTION_NONE				
	PICTSMART_LIGHTCORRECTION_COOLFLORESCENT				
	PICTSMART_LIGHTCORRECTION_WARMFLORESCENT				
	PICTSMART_LIGHTCORRECTION_OUTSIDE				
PROPERTY_MOTION_DETECTION_MODE	TRUE/FALSE		•	•	•
PROPERTY_MOTION_SENSITIVITY	0.255		•	•	•
PROPERTY_WHITELEVEL	0.255		•	•	•
PROPERTY_AUTO_WHITELEVEL	ADJUSTMENT_MANUAL		•	•	•
	ADJUSTMENT_AUTOMATIC		•	•	•
PROPERTY_ANALOGGAIN	0.255		•	•	•
PROPERTY_AUTO_ANALOGGAIN	ADJUSTMENT_MANUAL		•	•	•
	ADJUSTMENT_AUTOMATIC		•	•	•
PROPERTY_LOWLIGHTBOOST	TRUE/FALSE		•	•	•
PROPERTY_COLORBOOST	TRUE/FALSE		•	•	•
PROPERTY_ANTIFLICKER	TRUE/FALSE		•	•	•
PROPERTY_OPTIMIZATION_SPEED_QUALITY	OPTIMIZE_QUALITYOPTIMIZE_SPEED				•
					•
PROPERTY_LED	LED_OFFLED_ONLED_AUTOLED_MAX				