



การบันทึกแพทเทิร์นของสายอากาศโดยคอมพิวเตอร์  
 COMPUTER BASE ANTENNA PATTERN RECORDER



โดย  
 นางสาวรจนา กิมฮง  
 นายวรรณฤทธิ์ พงศ์ประยูร

วัน เดือน ปี..... 17 ค.ค. 2541  
 เลขทะเบียน..... 039030  
 เลขเรียกหนังสือ..... T. A0271 7117 ก

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต

สาขาวิศวกรรมโทรคมนาคม

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2540

*Handwritten signature*

*Handwritten signature*

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
 ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

039030

การบันทึกแพทเทิร์นของสายอากาศโดยคอมพิวเตอร์  
COMPUTER BASE ANTENNA PATTERN RECORDER

โดย  
นางสาวรจนา กิมฮง 37014349  
นายวรรณฤทธิ์ พงศ์ประยูร 37014374



ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต  
สาขาวิศวกรรมโทรคมนาคม  
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง  
ปีการศึกษา 2540

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ปริญญาโทปีการศึกษา 2540

ภาควิชาวิศวกรรมโทรคมนาคม

คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เรื่อง การบันทึกแพทเทิร์นของสายอากาศโดยคอมพิวเตอร์

**COMPUTER BASE ANTENNA PATTERN RECORDER**

ผู้จัดทำ

- |                 |            |          |
|-----------------|------------|----------|
| 1. นางสาวรจนา   | กิมสง      | 37014349 |
| 2. นายวรรณฤทธิ์ | พงศ์ประยูร | 37014374 |



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การบันทึกแพทเทิร์นของสายอากาศโดยคอมพิวเตอร์

COMPUTER BASE ANTENNA PATTERN RECORDER

โดย นางสาวรณา กิมอง 37014349

นายวรรณฤทธิ์ พงศ์ประยูร 37014374

อาจารย์ที่ปรึกษา

อ.สุรพล บุญจันทร์

รศ.สมยศ จุณณะปิยะ

บทคัดย่อ

ปริญญานิพนธ์ฉบับนี้เป็นการประยุกต์นำเอาไมโครคอมพิวเตอร์มาช่วยในการพล็อตแพทเทิร์นของสายอากาศ ซึ่งจะทำให้การวัดและพล็อตแพทเทิร์นของสายอากาศเป็นไปอย่างรวดเร็วและมีประสิทธิภาพมากขึ้น โดยระบบการทำงานจะแบ่งเป็น 2 ส่วน คือ ส่วนของ hardware และ ส่วนของ software ในส่วนของ hardware จะเป็นการสร้างระบบปรับทิศทางของสายอากาศด้วยสเตปป์มอเตอร์ เพื่อให้สายอากาศหมุนไปที่มุมต่างๆ ได้ และทำการสร้างวงจรสำหรับแปลงค่าความแรงของสนามที่วัดได้ ให้เป็นข้อมูลดิจิทัล เพื่อที่จะสามารถส่งเข้าไปเก็บในไมโครคอมพิวเตอร์ได้ ในส่วนของ software จะเป็นการพัฒนาโปรแกรมสำหรับทำการพล็อตแพทเทิร์นของสายอากาศ โดยปรับทิศทางของสายอากาศ แล้วรับค่าความแรงของสนามเข้ามาทำการพล็อตแพทเทิร์นของสายอากาศ โดยแสดงผลบนจอภาพของคอมพิวเตอร์

ABSTRACT

This project is applied by microcomputer for test measure and record the antenna pattern. It has 2 part : hardware and software. The hardware which build the antenna positioner with stepping motor that controller by microcontroller and it has a changed field strength circuit for tested antenna that change analog signal to digital signal. In the part of software would develop program for plot the pattern of antenna and display it on monitor.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## สารบัญ

	หน้า
บทที่ 1 บทนำ	1
บทที่ 2 ทฤษฎีและหลักการ	3
บทที่ 3 การออกแบบและการสร้าง	35
บทที่ 4 การทดลองและผลการทดลอง	54
บทที่ 5 บทวิจารณ์และบทสรุป	65
ภาพผนวก	
บรรณานุกรม	



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.30 แสดง Input state ของ address line เพื่อเลือกชั้นแนล(Channel) ต่างๆ	32
รูปที่ 2.31 แสดง Connection Diagram	33
รูปที่ 2.32 แสดง Timing Diagram	33
รูปที่ 2.33 แสดง Application Information	34
รูปที่ 3.1 แสดงบล็อกโคอะแกรมการทำงานของระบบ	35
รูปที่ 3.2 แสดงบล็อกโคอะแกรมของระบบปรับทิศทางของสายอากาศด้วยสเตปปีงมอเตอร์	37
รูปที่ 3.3 แสดงวงจรของคอนโทรลเลอร์บอร์ด	38
รูปที่ 3.4 แสดงวงจรจับสเตปปีงมอเตอร์	40
รูปที่ 3.5 แสดงการต่อวงจร ADC0809 เข้ากับไมโครคอนโทรลเลอร์	41
รูปที่ 3.6 แสดงการต่อวงจร Serial Port	42
รูปที่ 3.7 แสดงไฟล์ชาร์ตของหน่วยควบคุมการทำงาน	43
รูปที่ 3.8 แสดงแบบฟอร์มการทำงานของโปรแกรม	47
รูปที่ 3.9 แสดงตารางกำหนด object , property , setting	48
รูปที่ 3.10 แสดงการกำหนดโครงเมนู	49
รูปที่ 3.11 a) แสดงโค้ดของออปชั่น 1 และออปชั่น 2	50
b) แสดงโค้ดของออปชั่น 3 และออปชั่น 4	50
c) แสดงโค้ดของเมนูต่างๆ	51
รูปที่ 3.12 แสดงส่วนประกอบเพิ่มเติมของโปรแกรม	52
รูปที่ 3.13 แสดงคุณสมบัติของ Mscomm1	52
รูปที่ 3.14 แสดงการเลือก chart ต่างๆ ใน Tee chart	53
รูปที่ 3.15 แสดงกราฟต่างๆ ใน Tee chart	53
รูปที่ 4.1 แสดงตาราง One-excitation (กำลังครึ่งเดียว)	54
รูปที่ 4.2 แสดงรูปสัญญาณของ one-excitation	55
รูปที่ 4.3 แสดงตาราง Two-excitation	55
รูปที่ 4.4 แสดงรูปสัญญาณ two-excitation	56

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 4.5 แสดงตาราง One-two excitation	56
รูปที่ 4.6 แสดงการวัดค่าความเป็น Linear ของ ADC0809	58
รูปที่ 4.7 แสดงชั้นแนล 1 เป็นสัญญาณที่เริ่มทำงานของ ADC0809	
ชั้นแนล 1 เป็นสัญญาณที่เริ่มทำงานของ ADC0809	60
รูปที่ 4.8 แสดงชั้นแนล 1 เป็นสัญญาณสิ้นสุดของ ADC0809	
ชั้นแนล 1 เป็นสัญญาณที่วัดจากพอร์ต 1	60
รูปที่ 4.9 แสดงชั้นแนล 1 เป็นสัญญาณที่วัดได้จากขา P1.0	
ชั้นแนล 1 เป็นสัญญาณที่วัดได้จากขา Rx ของ CPU	61
รูปที่ 4.10 แสดงผลการทดลองเป็นกราฟรูปโพลาร์ ของแพทเทิร์นของสายอากาศแบบโมโนโพล	
วัดที่ความถี่ 300 MHz	63
รูปที่ 4.11 แสดงผลการทดลองเป็นกราฟรูปเหลี่ยม ของแพทเทิร์นของสายอากาศแบบโมโนโพล	
วัดที่ความถี่ 300 MHz	64



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## บทที่ 1

### บทนำ

สิ่งสำคัญที่อธิบายถึงคุณสมบัติของสายอากาศคือ แพทเทิร์นของสายอากาศ (antenna pattern) ว่าจะมี ความสามารถในการแพร่กระจายคลื่นไปในทิศทางใดได้บ้าง เราจะทราบรูปแพทเทิร์นได้จากการวัดความเข้มของ สนามที่สายอากาศทดสอบสามารถรับได้ ด้วยเครื่องวัดความเข้มสนาม (field strength meter) ที่ต่ออยู่กับสาย อากาศทดสอบที่เราต้องการวัด โดยการหมุนสายอากาศไปให้ทำมุมต่างๆกับทิศทางของคลื่นที่ส่งออกมาจาก ต้น กำเนิด และทำการวัดความเข้มของสนามที่มุมต่างๆจนครบ 360 องศา จากนั้นนำค่าความเข้มของสนามที่วัดได้มา วาดลงในแผ่น polar plot เราก็จะได้รูปแพทเทิร์นของสายอากาศ

อย่างไรก็ตามการวัดและการพล็อตค่าความเข้มสนามที่มุมต่างๆจนครบ 360 องศา จะต้องใช้เวลานาน พอสสมควรและเป็นงานที่ไม่ค่อยจะรื่นรมย์นัก โดยเฉพาะในกรณีที่จะต้องทำการวัดในสถานที่ ที่มีอากาศร้อนจัด และแดดจัดในตอนกลางวัน

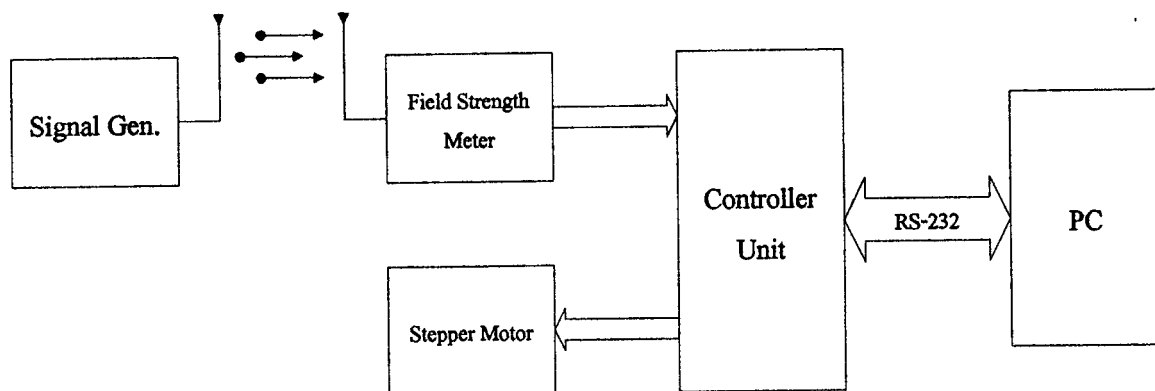
เนื่องจากในปัจจุบันนี้คอมพิวเตอร์ได้เข้ามามีบทบาทอย่างมากในงานวิศวกรรม ซึ่งในการออกแบบสาย อากาศก็เช่นกัน ได้มีการนำเอาไมโครคอมพิวเตอร์เข้ามาช่วยในการออกแบบอย่างกว้างขวาง รวมถึงการวัด แพทเทิร์นของสายอากาศ

ปรัชญาพื้นฐานนี้เป็นการประยุกต์นำเอาไมโครคอมพิวเตอร์มาช่วยในการพล็อตแพทเทิร์นของสาย อากาศโดยระบบการทำงานจะแบ่งเป็น 2 ส่วนคือ ส่วนของฮาร์ดแวร์ (hardware) และส่วนของซอฟต์แวร์ (software) ในส่วนของฮาร์ดแวร์ จะเป็นการสร้างระบบปรับทิศทางของสายอากาศด้วยสเตปป์ิงมอเตอร์ เพื่อให้ สายอากาศหมุนไปที่มุมต่างๆ ได้ และทำการสร้างวงจรสำหรับแปลงค่าความแรงของสนามไฟฟ้าที่วัดได้ เพื่อให้ เป็นข้อมูลดิจิทัล เพื่อที่จะสามารถส่งเข้าไปเก็บในไมโครคอมพิวเตอร์ได้ และในส่วนของซอฟต์แวร์เป็นการ พัฒนาโปรแกรมสำหรับการพล็อตแพทเทิร์นของสายอากาศ โดยแสดงผลบนจอภาพของคอมพิวเตอร์

ที่ผ่านมารเราจะทำการศึกษาทฤษฎีการวัดแพทเทิร์นของสายอากาศ และทำการสร้างระบบปรับทิศทาง ของสายอากาศโดยใช้สเตปป์ิงมอเตอร์เพื่อจะทำให้สายอากาศหมุนไปที่มุมต่างๆ ได้ เพื่อเป็นการสะดวกในการ ทำการวัดแพทเทิร์นของสายอากาศ เรานำเอาไมโครคอนโทรลเลอร์มาใช้ในการควบคุมสเตปป์ิงมอเตอร์

ในภาคการศึกษานี้เราได้ทำการเขียนโปรแกรมในการพล็อตแพทเทิร์นของสายอากาศโดยใช้โปรแกรม ภาษา Visual Basic และทำการปรับปรุงฮาร์ดแวร์ ให้ทำงานได้ดียิ่งขึ้น และทำการทดลองและวัดผลเพื่อให้ได้รูป แพทเทิร์นของสายอากาศออกมาแสดงบนจอคอมพิวเตอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 1.1 แสดง บล็อกไดอะแกรมของระบบ Computer Base Antenna Pattern Recorder



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## บทที่ 2

### ทฤษฎี

สายอากาศเป็นโครงสร้างซึ่งมักจะถูกสร้างจากวัสดุที่เป็นตัวนำที่ดี เพื่อออกแบบมาให้มีรูปร่างและขนาดที่จะแพร่กระจายคลื่นแม่เหล็กไฟฟ้าได้อย่างมีประสิทธิภาพ เป็นที่ทราบกันดีว่ากระแสซึ่งมีการเปลี่ยนแปลงตามเวลาจะแพร่กระจายคลื่นแม่เหล็กไฟฟ้า ดังนั้นสายอากาศจึงเป็นโครงสร้างซึ่งถูกกระตุ้นด้วยกระแสที่มีการเปลี่ยนแปลงตามเวลา ซึ่งมักจะทำได้โดยใช้สายส่งกำลังหรือท่อนำคลื่น อาจมีรูปร่างแตกต่างกันไปได้มากมาย สายอากาศที่ใช้กันแพร่หลายมากที่สุดคือสายอากาศไดโพลซึ่งเป็นส่วนประกอบพื้นฐานของสายอากาศซึ่งจะส่งหรือรับ (สายอากาศของเครื่องส่งและเครื่องรับต้องมีคุณสมบัติเหมือนกัน) คลื่นแม่เหล็กไฟฟ้าที่กระจายออกรอบตัวสายอากาศลักษณะรูปร่างของการกระจายพลังงานคลื่นแม่เหล็กไฟฟ้าเป็นรูปร่างต่างๆ เรียกว่ารูปแบบหรือแพทเทิร์นซึ่งจะเป็นตัวกำหนดค่าไดเรกทิวิตี (Directivity) คือการกระจายพลังงานไปในทิศทางใดทิศทางหนึ่งมากกว่าทิศทางอื่นๆ ซึ่งมีความสำคัญในการรับและส่งในทิศทางเดียวและป้องกันการรบกวนจากสถานีอื่นๆ

ดังนั้นจึงต้องมีการออกแบบสายอากาศในลักษณะการรับสัญญาณในทิศทางแ่งทิศทางใด จะทำให้ประสิทธิภาพในการรับสูงขึ้น และสามารถจัดสัญญาณรบกวนที่มาจากที่อื่นหรือมาจากการสะท้อนให้หมดไป สิ่งที่เราทราบได้ถึงทิศทางกรรับหรือส่งของสายอากาศ หรือรูปแบบการรับสัญญาณของสายอากาศ ทำได้โดยการวัดแพทเทิร์นของสายอากาศ

#### 2.1 หลักการวัดแพทเทิร์นของสายอากาศ

การที่จะทราบแพทเทิร์นของสายอากาศตัวหนึ่งเราต้องทราบความเข้มของสนาม (Field Strength) ที่สายอากาศตัวนั้นสามารถรับได้ ณ.ทุกๆ ทิศทางรอบตัวมัน ซึ่งจะเห็นว่ารูปแพทเทิร์นนั้นเป็นฟังก์ชันของมุมกวาด (Azimuth) และมุมเงย (Elevate) ซึ่งจะเป็น 3 มิติ แต่ในการใช้งานจริงนั้นเราต้องการรู้เพียงแพทเทิร์นในระนาบใดระนาบหนึ่งซึ่งเป็นระนาบที่ใช้งานจริงนั่นคือแพทเทิร์นที่ใช้งานจริงคือแพทเทิร์น 2 มิติ

ดังนั้นในการวัดแพทเทิร์นของสายอากาศตัวหนึ่ง จะวัดที่ความเข้มสนามที่สายอากาศตัวนั้นสามารถรับได้ว่ารับได้ดีเพียงใด ณ.ทุกๆ องศา (360 องศา) นั่นต้องมีสายอากาศส่ง (Source Antenna) ส่งคลื่นความถี่หนึ่งมายังสายอากาศทดสอบที่เราต้องการรู้แพทเทิร์น (Antenna under Test) แล้วเราก็จะวัดความเข้มสนามซึ่งสายอากาศสามารถรับได้ โดยการหมุนสายอากาศทดสอบไปในทิศทางรอบตัวมันทีละองศา

ในการพล็อตวัดแพทเทิร์นของสายอากาศนั้นวิธีที่การวัดอยู่ 3 แบบคือ

- เฟสแพทเทิร์น (Phase pattern) ซึ่งเป็นกราฟที่แสดงค่าเฟสของสัญญาณที่มุมต่างๆของสายอากาศ
- เพาเวอร์แพทเทิร์น (Power pattern) ซึ่งเป็นการวัดค่ากำลังงานที่มุมต่างๆที่สายอากาศหมุนไป
- ฟิลด์แพทเทิร์น (Field pattern) ซึ่งเป็นการพล็อตค่าความเข้มของสนามที่วัดได้ที่มุมต่างๆที่สายอากาศ

หมุนไป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ฟิลด์แพทเทิร์น และเพาเวอร์แพทเทิร์น เป็นสิ่งสำคัญที่จะบอกถึงลักษณะการแพร่กระจายของคลื่น แต่ฟิลด์แพทเทิร์นจะวัดได้ง่ายกว่าเพาเวอร์แพทเทิร์น ดังนั้นจึงนิยมวัดฟิลด์แพทเทิร์นมากกว่าวัดเพาเวอร์แพทเทิร์น ทั้งนี้เนื่องจากแพทเทิร์นทั้งสองแบบจะมีรูปร่างเหมือนกัน ส่วนเฟสแพทเทิร์นนั้นไม่ค่อยมีความสำคัญมากนัก

### 2.1.1 เกนของสายอากาศ (Antenna gain)

เป็นอัตราส่วนความเข้มของสนามหรือกำลังงานของคลื่นที่มีค่าสูงสุดที่ระยะห่างจากตัวสายอากาศ ระยะหนึ่งกับค่าความเข้มของสนามหรือกำลังงานของสายอากาศไอโซโทรปิกที่ระยะห่างเท่ากัน

$$\begin{aligned} \text{power gain} &= \frac{E_{\text{MAX}}^2}{E_{\text{MAX reference}}^2} \\ \text{field intensity gain} &= \frac{E_{\text{MAX}}}{E_{\text{MAX reference}}} \\ \text{gain(dB)} &= 10 \log \frac{E_{\text{MAX}}}{E_{\text{MAX ISO}}} \\ \text{power gain(dB)} &= 20 \log \frac{E_{\text{MAX}}}{E_{\text{MAX ISO}}} \end{aligned}$$

### 2.1.2 ระยะที่เหมาะสมในการวัดแพทเทิร์นของสายอากาศ

การวัดแพทเทิร์นของสายอากาศนั้นจะต้องทำการวัดที่ระยะฟาร์ฟิลด์ ซึ่งมีค่าเป็น

$$R = 2D^2 / \lambda$$

ซึ่งระยะนี้เป็นระยะที่ไกลพอที่จะประมาณได้ว่าสายอากาศที่ทำการทดสอบเป็นเสมือนจุดกำเนิด (Point source) หรือเสมือนว่าสายอากาศนั้นมีขนาดเล็กมาก จนกระทั่งพิจารณาเป็นจุดๆหนึ่งเท่านั้น และยังเป็นระยะที่คลื่นแผ่ ออกไปในอากาศมีลักษณะเป็นคลื่นตามขวางโดยสมบูรณ์

พื้นที่ที่ใช้ในการทดสอบสายอากาศ โดยทั่วไปแล้วแพทเทิร์นของสายอากาศไม่สามารถจะวัดได้ด้วย เครื่องมือที่ ติดตั้งบนพื้นผิวโลกได้ ทั้งนี้เนื่องจากการสะท้อนของคลื่นจากพื้นดินและวัตถุรอบข้าง แต่ที่ความ

ถึงสูงเราพอจะอนุโลมได้ และสำหรับการวัดในพื้นที่จำกัดเช่น ในห้องทดลองไร้แรงสะท้อน (Anechoic room) ซึ่งผนังห้องจะถูกบุด้วยสารที่มีคุณสมบัติดูดกลืนคลื่นได้ดี

### 2.1.3 พิสัยของการทดสอบสายอากาศ

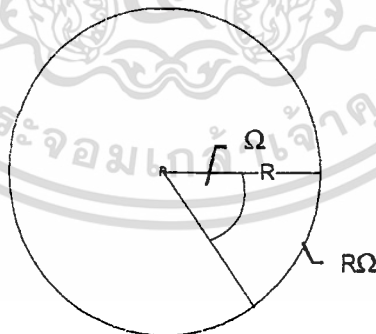
การทดสอบสายอากาศจะสามารถทำการทดสอบได้ ผลลัพธ์ที่ถูกต้องมากน้อยแค่ไหนขึ้นกับแฟกเตอร์หลักดังนี้

1. ขนาดของสายอากาศที่จะทำการวัด ซึ่งจะมีผลโดยตรงต่อระยะที่จะทำการทดสอบ โดยถ้าสายอากาศมีขนาดใหญ่ก็จะทำให้ระยะในการทดสอบต้องไกลขึ้น
2. ชนิดของสายอากาศที่จะทำการทดสอบ
3. ความแน่นอนของผลลัพธ์ที่ต้องการ
4. ระยะที่ใช้ในการทดสอบ ว่าเป็นระยะ ฟาร์ฟิลด์ หรือ เนียร์ฟิลด์

### 2.1.4 ขนาดของสายอากาศต้นกำเนิด

สายอากาศส่ง (Antenna source) ที่ทำการกระจายคลื่นมายังสายอากาศทดสอบ (Antenna under test) นั้นจะต้องมีขนาดใหญ่เพียงพอที่จะลดการสะท้อนของคลื่นที่มาจากสิ่งกีดขวางและพื้นผิวโลก อย่างไรก็ตาม ถ้าสายอากาศมีขนาดใหญ่เกินไป จะส่งผลให้เกิดคลื่นที่มีแอมพลิจูดไม่คงที่บนสายอากาศทดสอบ

ที่ระยะทาง  $R$  พื้นที่ที่ตกอยู่ภายใต้การส่งที่มี บีมวิดท์ (Beamwidth) เป็น  $\Omega$  จะเท่ากับ  $R\Omega$  พื้นที่นี้จะต้องใหญ่เป็น 4 เท่าของสายอากาศที่ใหญ่ที่สุด เพื่อเป็นการแกว่ง (Fluctuation) ของแอมพลิจูดไม่ให้เกิน 1/4 dB ดังรูปที่ 2.1



รูปที่ 2.1 แสดงขนาด  $R$  ซึ่งต้องใหญ่กว่าสายอากาศทดสอบ 4 เท่า

โดยที่  $\Omega \approx \lambda/d$  ดังนั้นขนาดของสายอากาศส่งควรเล็กกว่าครึ่งหนึ่งของสายอากาศทดสอบที่ใหญ่ที่สุด ถ้าทดสอบที่  $2D^2/\lambda$  สำหรับพิสัยการทดสอบที่สั้นเท่าไร ขนาดของสายอากาศส่งจะยิ่งเล็กลงเท่านั้น คำ  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดลอกเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## 2.1.5 การสะท้อนของคลื่นขณะทดสอบ

คลื่นที่สะท้อน (Reflect) กับสิ่งกีดขวางมาเข้าสายอากาศทดสอบ เป็นสาเหตุสำคัญของความผิดเพี้ยนของแพทเทิร์นที่เราต้องการวัด สาเหตุสำคัญที่ทำให้เกิดความผิดเพี้ยนในการวัดมีดังนี้

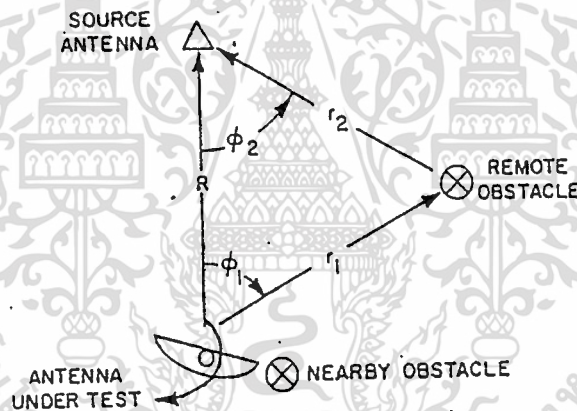
1. สิ่งกีดขวางที่อยู่ไกล (Remote Obstacle) เป็นตัวสะท้อนที่ทำให้เกิดความผิดเพี้ยน (Error) ที่สำคัญที่อยู่ไกลสายอากาศ เช่น สิ่งก่อสร้าง ตึก ต้นไม้ และอื่นๆ ค่าผิดพลาดจะสูงสุดเมื่อเมนบีม (Main beam) ของสายอากาศทดสอบหันไปยังสิ่งกีดขวางนั้น ซึ่งเป็นทิศที่รับสัญญาณสะท้อนมากที่สุด

ขนาดของความผิดพลาดจะบอกเป็นอัตราความเหลื่อมล้ำ (Interference ratio) ระหว่างพลังงานของคลื่นที่สายอากาศทดสอบได้รับจากสายอากาศส่งกับพลังงานของคลื่นที่สายอากาศทดสอบ

$$\text{Interference ratio (IR)} = G(\phi_2) / G(0) * (R\sigma / 4\pi r_1^2 r_2^2)$$

$G$  = directive gain of source ant

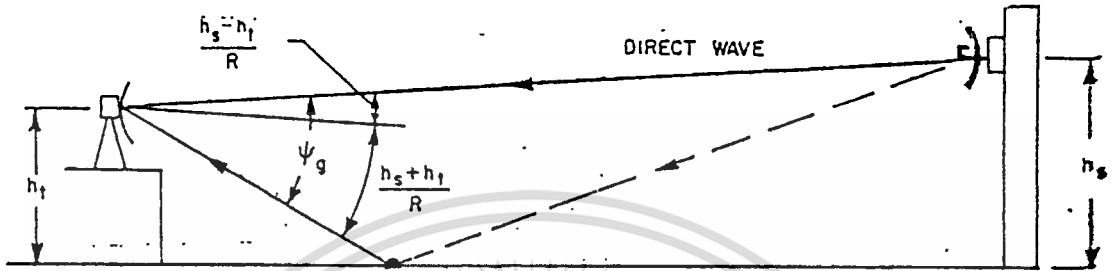
$\sigma$  = effective scattering cross section



รูปที่ 2.2 แสดงคลื่นสะท้อนจากสิ่งกีดขวางที่อยู่ไกล

2. สิ่งก่อสร้างที่อยู่ใกล้ (Nearby Obstacle) หมายถึงสิ่งก่อสร้างที่อยู่ใกล้สายอากาศ เกนหมุน บันได ขอบของหอคอยทดสอบ และอื่นๆ จะเป็นตัวสะท้อนให้เกิดการอินเตอร์เฟอเรนซ์ (Interference) ซึ่งจะมีผลต่อสายอากาศที่อัตราขยายต่ำ (Low gain Antenna) เท่านั้น จะไม่มีผลต่อสายอากาศที่มีโคเรคทีวิตีสูง เราสามารถแก้การเกิดอินเตอร์เฟอเรนซ์ ได้โดยการเอาวัสดุที่ไม่สะท้อนคลื่นไป คลุมสิ่งก่อสร้างที่สะท้อนคลื่น

3. สายอากาศที่มีการกระจายคลื่นกว้าง (Wide beam Antenna) หมายถึงสายอากาศที่มีทิศทางการรับคลื่นกว้างเช่น omnidirectional การวัดแพทเทิร์นของสายอากาศพวกนี้ต้องวัดในที่ที่มีการอินเตอร์เฟอเรนซ์น้อยกว่า 20 dB เช่นในห้องไร้การสะท้อนหรือคลุมปากสายอากาศ (Mount) ด้วยวัสดุที่สะท้อนคลื่นได้ต่ำ (Low reflection Material) เพื่อใช้ในการใช้งานเพื่อการศึกษาเท่านั้น ไม่นิยมนำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.3 แสดงการสะท้อนของคลื่นกับพื้นผิวเรียบ

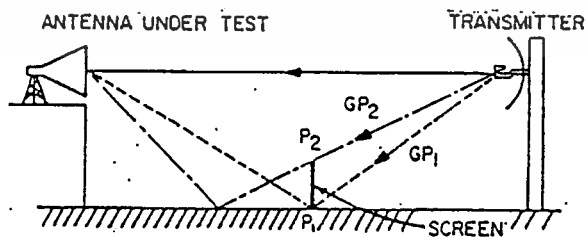
$h_s$  = ความสูงของสายอากาศส่งจากพื้นดิน

$h_t$  = ความสูงของสายอากาศรับจากพื้นดิน

โดยมุมระหว่างทิศทางของคลื่นตรงกับคลื่นสะท้อนคือ  $\psi = 2b/R$

การสะท้อนจะมีผลเมื่อ  $h_s$  และ  $h_t$  มีขนาดเล็กเมื่อเทียบกับ  $R$

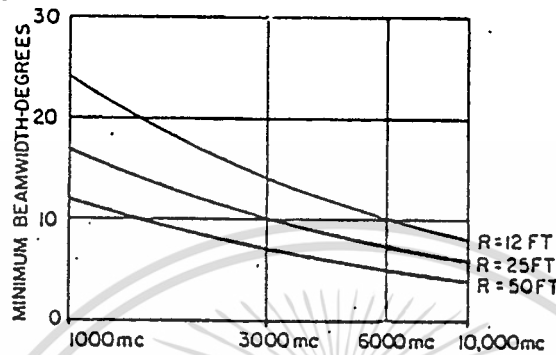
เราสามารถแก้การสะท้อนได้โดยนำเอาแผ่นscreen ที่มีความสูงมากกว่าไปกั้นที่พื้นผิว ณ จุดกึ่งกลางระหว่างสายอากาศส่งและสายอากาศรับดังรูป 2.4



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับรูป 2.4 แสดงการกั้นคลื่นสะท้อนจากพื้นผิว ญาติให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## 2.2.6 ห้องไร้การสะท้อน (Anechoic Chamber)

เป็นห้องเฉพาะใช้ทดสอบสายอากาศที่มีเกน(gain)ต่ำ ผนังห้องจะระบุด้วยวัสดุดูดคลื่นคลื่น ขนาดของห้องขึ้นอยู่กับพิสัยทดสอบ ความสัมพันธ์ระหว่างพิสัยทดสอบเป็นดังรูปที่ 2.5



รูปที่ 2.5 กราฟแสดงความสัมพันธ์ของ บีมวิคท์ และ R ในห้องไร้การสะท้อน

## 2.2 สเตปป์ิงมอเตอร์

สเตปป์ิงมอเตอร์เป็นอุปกรณ์เชิงกลทางไฟฟ้าที่มีอินพุทเป็นไบนารี โวลเตจ และเอาท์พุทเป็นลักษณะของการเคลื่อนที่เชิงมุมหรือหมุนไปเป็นสเตป การป้อนสัญญาณพัลส์ในลักษณะซีควเอนซ์ของเลขไบนารี ให้กับขดของสเตเตอร์ที่มีอยู่หลายขด แต่ละขดเรียกว่าเฟส จะเกิดแรงผลักต่อโรเตอร์ ทำให้โรเตอร์หมุนได้อย่างต่อเนื่อง

### 2.2.1 ข้อดีและข้อเสียของสเตปป์ิงมอเตอร์

#### ข้อดี

- สามารถขับเคลื่อนแบบ โอเพ่นลูป (Open loop) โดยไม่ต้องการการป้อนกลับ (Feedback) ได้ ไม่ว่าจะเป็นการควบคุมตำแหน่ง หรือความเร็ว โดยไม่มีปัญหาเรื่องเสถียรภาพ
- ความผิดพลาดเนื่องจากตำแหน่งน้อยมากเพราะการเคลื่อนที่เป็นแบบสเตป ซึ่งมีองศาที่แน่นอน
- สเตปป์ิงมอเตอร์ ครอบครองต่อ สัญญาณดิจิทัลทางด้านอินพุท ดังนั้นจึงเหมาะสำหรับใช้คอมพิวเตอร์ควบคุม
- ส่วนประกอบทางกลมีน้อยและไม่ต้องการบำรุงรักษา
- ติดตั้งได้ง่ายมีความทนทาน

#### ข้อเสีย

เอกสารนี้เป็นเอกสารที่จัดทำขึ้นเพื่อใช้ในการเรียนการสอนเท่านั้น ไม่สามารถนำไปใช้  
 - สเตปป์ิงมอเตอร์มีการขับเคลื่อนมุมสเตปที่แน่นอน ทำให้ขาดความยืดหยุ่นในสเตปรีโซลูชัน (Step Resolution) ไม่ควรใช้ทุกชิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- ประสิทธิภาพต่ำหากใช้วงจรขับแบบธรรมดา
- เกิด โอเวอร์ชูท (Overshoot) และ ออสซิลเลชัน (Oscillation) ในสเตปเรสปอนด์ (Step response)
- มีความสามารถจำกัด เมื่อใช้กับ โหลดที่มีความเฉื่อยสูง
- ฟริคชันโหลด (Friction load) จะเพิ่มความผิดพลาดของตำแหน่ง หากควบคุมแบบโอเพ่นลูป แต่

ความผิดพลาดจะเป็นแบบไม่สะสม (non- accumulative)

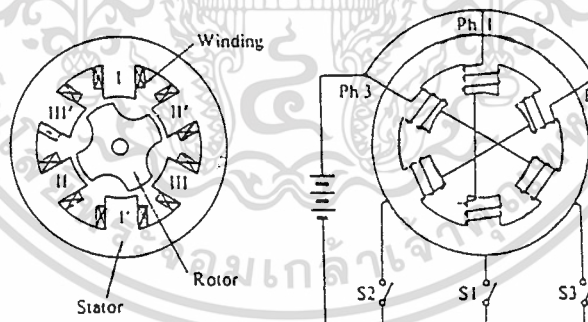
- กำลังทางค่านเอาท์พุทมีค่าจำกัด

### 2.2.1 สเตปปีงมอเตอร์ที่ใช้กันมากมี 3 ชนิดคือ

1. วาริเอเบิลรีลักแทนซ์สเตปปีงมอเตอร์ ( Variable Stepping Motor )
2. สเตปปีงมอเตอร์แบบแม่เหล็กถาวร ( Permanent Magnetic Stepping Motor )
3. สเตปปีงมอเตอร์แบบไฮบริด ( Hybrid Stepping Motor )

#### - วาริเอเบิลรีลักแทนซ์สเตปปีงมอเตอร์

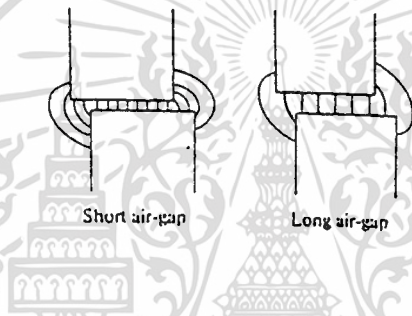
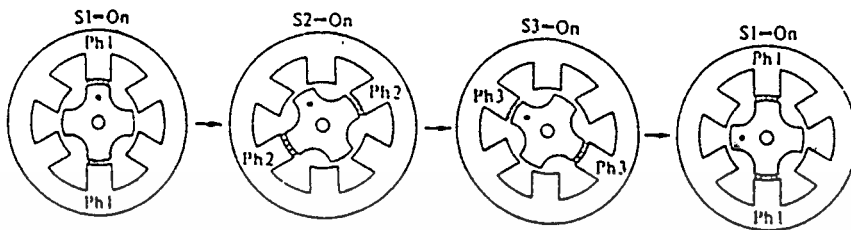
มอเตอร์แบบนี้โรเตอร์ทำด้วยเหล็กอ่อน ซึ่งค่าซึมซาบแม่เหล็ก (Permeability) สูง และสามารถให้ฟลักซ์แม่เหล็กผ่านได้มาก โดยโรเตอร์จะติดอยู่กับแกนมอเตอร์ และสเตเตอร์ติดอยู่กับโครงของตัวมอเตอร์ จากรูป 2.3 เป็นภาพตัดขวางของสเตปปีงมอเตอร์แบบนี้ซึ่งเป็นมอเตอร์ 3 เฟส มีซี่ฟัน 6 ซี่ ฟันของสเตเตอร์อยู่ตรงกันข้ามจะต่อกันเป็นอนุกรมหรือขนานก็ได้ (ในที่นี้คือแบบอนุกรม)



รูปที่ 2.6 โครงสร้างวาริเอเบิลรีลักแทนซ์มอเตอร์

เราจะเห็นว่าฟันของสเตเตอร์ 2 ซี่ ที่เฟสเดียวกันจะมีซี่แม่เหล็กตรงกันข้ามกันดังแสดงในรูป สมมติว่าฟัน I, II และ III มีซี่เป็นซี่เหนือ ฟัน I, II และ III จะเป็นซี่ใต้เมื่อถูกกระตุ้น กระแสแต่ละเฟสจะถูกกระตุ้น ฟลักซ์แม่เหล็กจะที่เกิดดังรูป 2.4 ฟันของ โรเตอร์ก็จะมีตำแหน่งในแนวเดียวกันกับฟันของสเตเตอร์ซึ่งมีผลให้แมกเนติกรีลักแทนซ์ (Magnetic Reluctance) น้อยที่สุด สภาวะนี้คือตำแหน่งสมดุล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.7 ฟลักซ์แม่เหล็กที่เกิดขึ้น

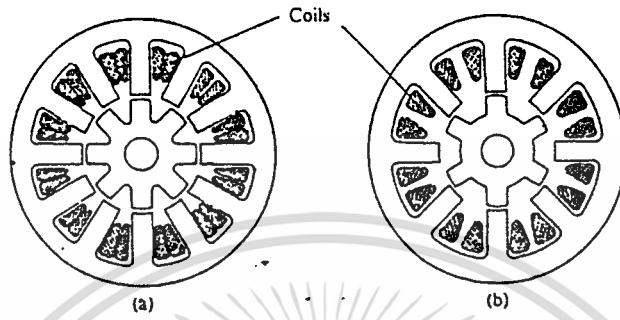
โครงสร้างเบื้องต้นของมอเตอร์แบบนี้ มีลักษณะดังนี้

1. ช่องว่างอากาศควรจะเล็กที่สุด เท่าที่จะเป็นไปได้ ช่องว่างอากาศระหว่างฟันของโรเตอร์ของสเตเตอร์ควรมีค่าความห่างกันน้อยมาก เพื่อที่จะได้ทอร์กสูงและตำแหน่งที่แน่นอนขึ้น
2. สำหรับมุมสเตปเล็กๆ จากรูป 2.8a แสดง 3 เฟสมอเตอร์ที่สเตเตอร์มีฟัน 12 ซี่ และ โรเตอร์มีฟัน 8 ซี่ รูปที่ 2.8 b เป็นรูป 4 เฟส มอเตอร์ที่สเตเตอร์มีฟัน 8 ซี่ และ โรเตอร์มีฟัน 6 ซี่ ซึ่งทั้งสองรูปนี้มีมุมสเตปเท่ากับ 15 องศา

ความสัมพันธ์ระหว่างมุมสเตป, จำนวนเฟส โดยจำนวนฟันของโรเตอร์  $N_r$  และจำนวนสเตปใน 1 รอบ  $S$  หาได้โดย

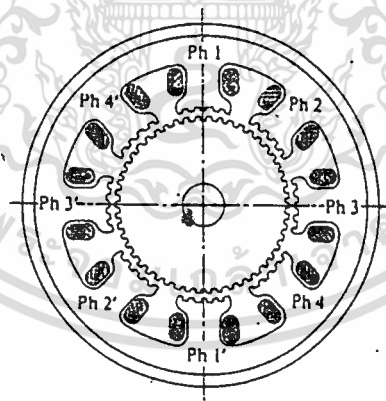
$$S = 360 / \theta_s = m (N_r)$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ เราสามารถลดมุมสเตปลงได้โดยเพิ่มจำนวนฟันบนโรเตอร์ ดังแสดงในรูปที่ 2.9 ที่มีการนำไปใช้



รูปที่ 2.8 แสดงโรเตอร์และสเตเตอร์

รูปที่ 2.9 หน้าตัดของ 4 เฟส มอเตอร์มีฟันโรเตอร์ 50 ซี่ มุมสเตป 1.8 องศา



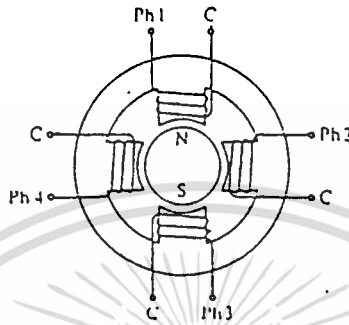
รูปที่ 2.9 แสดงหน้าตัดของสเตปมอเตอร์ 4 เฟส

- สเตปมอเตอร์แบบแม่เหล็กถาวร

สเตปมอเตอร์แบบนี้ใช้แม่เหล็กแบบถาวร รูปที่ 2.10 เป็นตัวอย่างของสเตปมอเตอร์แบบแม่เหล็กถาวร แบบ 4 เฟส โรเตอร์เป็นทรงกระบอก สเตเตอร์มีฟัน 4 ซี่ โดยที่แต่ละซี่มีขดลวดพันรอบ ถ้าจำนวนขั้วบนสเตเตอร์และขั้วแม่เหล็กบนโรเตอร์เพิ่มขึ้นเป็น 2 เท่า มุมแต่ละสเตปลดลงครึ่งหนึ่ง ดังนั้นเพื่อที่จะลดค่าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

มุมสเตปลงไปอีก ในสเตปป์มอเตอร์แบบแม่เหล็กถาวร ต้องเพิ่มจำนวนแม่เหล็กและซี่ฟัน อย่างไรก็ตามซี่แม่เหล็กที่จะเพิ่มขึ้นได้นั้นมีจำนวนจำกัด

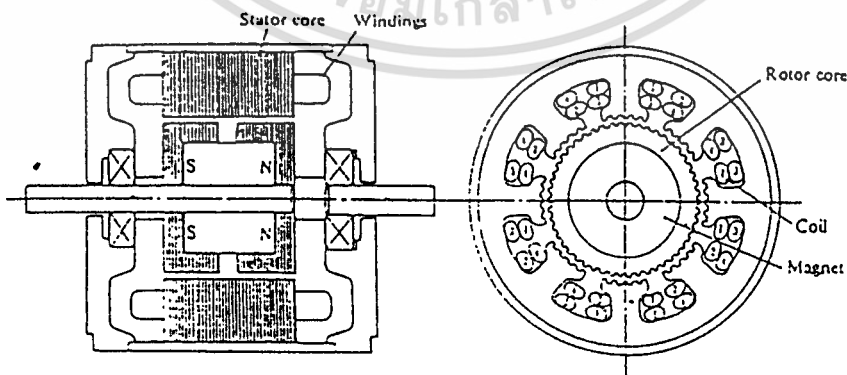
ลักษณะทั่วไปของมอเตอร์แบบนี้คือ โรเตอร์จะถูกยึดอยู่กับที่ แม้ว่าไม่มีการกระตุ้นเฟส ลักษณะเช่นนี้เรียกว่า ดีเทนท์ แมคคาไนซึ่ม (Detent Mechanism)



รูปที่ 2.10 แสดงโครงสร้างสเตปป์มอเตอร์ 4 เฟส

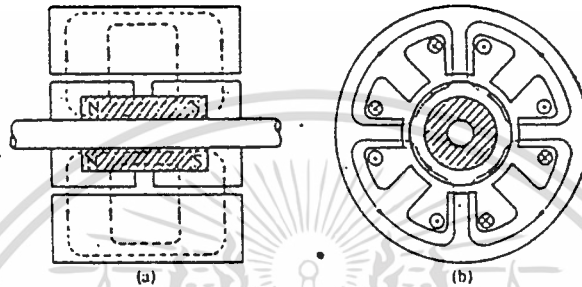
- สเตปป์มอเตอร์แบบไฮบริดจ์

เป็นสเตปป์มอเตอร์แบบหนึ่งที่มีโรเตอร์เป็นแม่เหล็กถาวรการใช้ชื่อไฮบริดจ์ได้มาจากการรวมหลักสำคัญของมอเตอร์แบบแม่เหล็กถาวรและแบบวาริเอเบิลรีลักแทนซ์ โครงสร้างของสเตเตอร์จะคล้ายกับแบบวาริเอเบิลรีลักแทนซ์ แต่การพันและการต่อขดลวดจะต่างจากแบบวาริเอเบิลรีลักแทนซ์มอเตอร์ ซึ่งมีเพียง 1 ขด จาก 2 ขด ของ 1 เฟส ที่ถูกพันเป็นขั้วเดียว ในขณะที่ 4 เฟสไฮบริดจ์มอเตอร์ขดขดลวด 2 เฟสที่แตกต่างกันจะถูกพันบนขั้วเดียวกันดังรูป 2.11 เพราะฉะนั้นขั้วหนึ่งจะไม่มีเพียงเฟสเดียว ขดลวด 2 ขด จะถูกพันเป็นขั้วเดียวกันแบบไบฟีลา (Bifilar Winding) ซึ่งจะทำให้ขั้วแม่เหล็กต่างกันขณะมีการกระตุ้น



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
รูปที่ 2.11 โครงสร้างของสเตปป์มอเตอร์แบบไฮบริดจ์  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกหนึ่งห้ามมิให้ตีแปลงเนื้อหาและต้องอ้างอิงแก่เจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ลักษณะที่สำคัญอย่างหนึ่งของไฮบริดมอเตอร์คือ โรเตอร์นั้นจะเป็นแม่เหล็กรูปร่างทรงกระบอกอยู่ในแกนเหล็กของโรเตอร์ มันถูกทำให้เป็นแม่เหล็กตามยาวเพื่อสร้างสนามขั้วเดียวดังรูปที่ 2.12 แต่ละขั้วของแม่เหล็กจะถูกล้อมรอบด้วยฟันเหล็กอ่อน ซึ่งฟันของโรเตอร์กับสเตเตอร์อยู่เหลื่อมกันอยู่ครึ่งช่วงฟัน

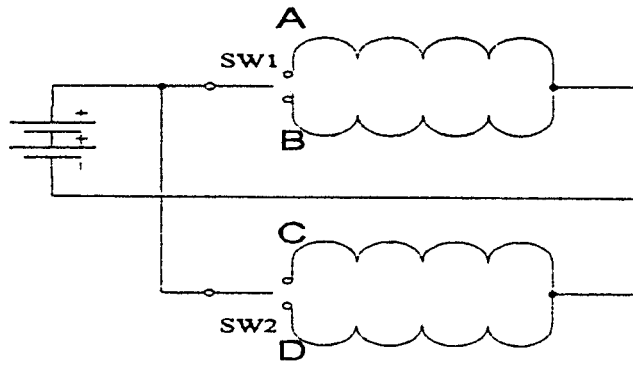


รูปที่ 2.12 การวางแม่เหล็กตามยาวเพื่อสร้างสนามขั้วเดียวกัน

### 2.2.2 การทำงานพื้นฐาน

สเตปป์ังมอเตอร์เป็นมอเตอร์ที่ถูกใช้งานโดยสัญญาณอินพุตที่เป็นพัลส์ทุกๆสัญญาณพัลส์ที่เข้ามาจะทำให้การเปลี่ยนแปลงสถานะสนามแม่เหล็กไฟฟ้าที่เกิดขึ้น และให้การหมุนของมอเตอร์เป็นมุมที่คงที่ ซึ่งจะแตกต่างกับการหมุนของมอเตอร์แบบธรรมดาในระบบควบคุมที่ใช้สัญญาณดิจิตอลนี้ เช่นในการส่งข้อมูล การควบคุมข้อมูล โดยที่ เป็นปริมาณค่าของสัญญาณดิจิตอลทั้งหมด จึงทำให้การควบคุมการทำงานของสเตปป์ังมอเตอร์โดยตรงได้เป็นอย่างดี การทำงานพื้นฐานของสเตปป์ังมอเตอร์แสดงในรูปที่ 2.13 ซึ่งแสดงบล็อกของวงจรขับทำให้กระแสไฟดีซี เรียงลำดับเข้าไปตามเฟสต่างๆของมอเตอร์หมุนไปตามองศาที่กำหนดไว้ การที่ทำให้กระแสไฟดีซีเรียงลำดับเข้าไปที่มอเตอร์ได้ ทำได้โดยการใช้งานของสวิทช์ และเพื่อให้มอเตอร์หมุนจำเป็นต้องมีวงจรขับสเตปป์ังมอเตอร์ที่จะควบคุมการไหลของกระแสไฟ คือ สวิทช์ 1 และสวิทช์ 2 ถ้าให้การทำงานของสวิทช์ตามตารางการทำงานที่แสดงไว้ในรูปที่ 2.14 ถ้าให้การทำงานโดยใช้รีเลย์ (Relay) หรือมือโยก สเตปป์ังมอเตอร์ก็จะหมุนเหมือนกันแต่หมุนช้า เนื่องจากรีเลย์ หรือมือโยกให้การเปลี่ยนแปลงของสวิทช์ช้า ซึ่งจะทำให้เร็วเท่ากับความเร็วสูงสุดของสัญญาณพัลส์ที่มอเตอร์ทำงานสามารถทำงานได้ (เป็นจำนวนพัลส์ต่อวินาที)

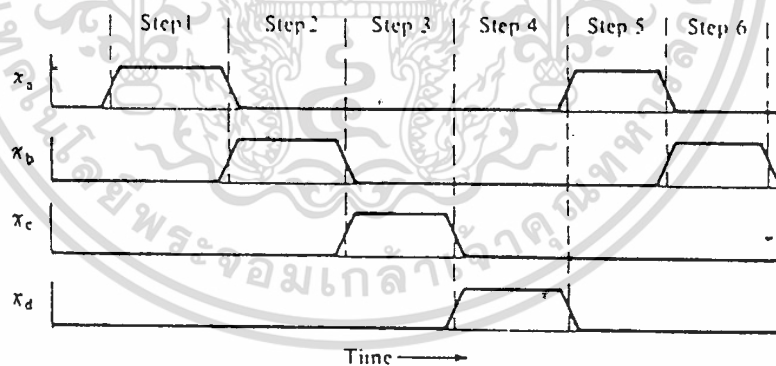
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.13 การทำงานพื้นฐานของสเต็ปมิงมอเตอร์

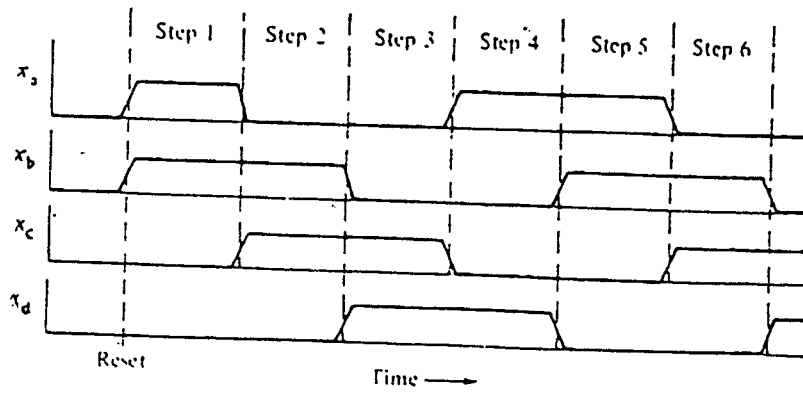
วงจรขับเคลื่อนมอเตอร์มีหลายชนิด ขึ้นอยู่กับชนิดของสเต็ปมิงมอเตอร์ที่มีอยู่ 3 ระบบคือ

1. ระบบการกระตุ้นสนามแม่เหล็กเฟสเดียว ( Single Phase Excitation )
2. ระบบการกระตุ้นสนามแม่เหล็ก 2 เฟส ( 2- Phase Excitation )
3. ระบบการกระตุ้นสนามแม่เหล็ก 1-2 เฟส ( Half-Step Excitation )

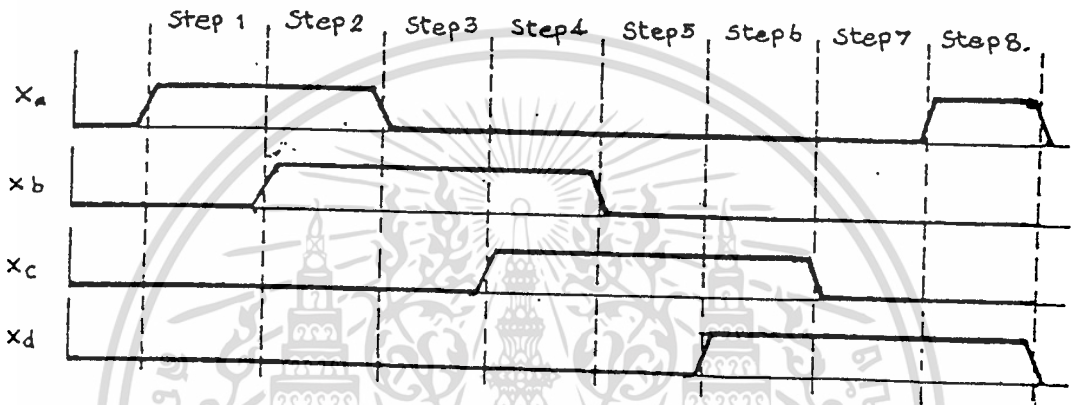


Single Phase Excitation

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



**2-Phase Excitation**

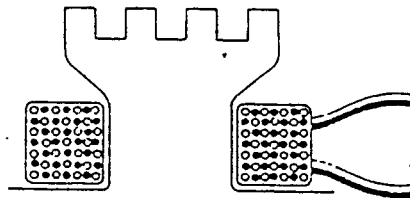


**Half-Step Excitation**

**รูปที่ 2.14** ชนิดการขับของสเตปป์ิงมอเตอร์

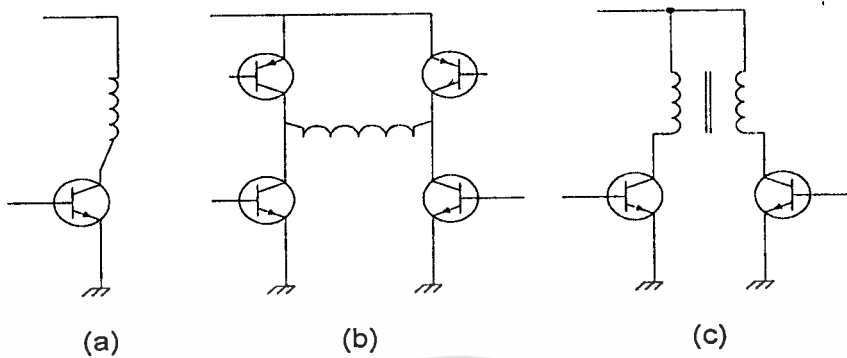
การพันลวดแบบโมนอฟิลลาและไบฟีลา (Monofilar and Bifilar Winding)

แบบโมนอฟิลลา เส้นลวดทั้ง 2 เส้นนี้จะถูกพันเหมือนกับเส้นเดียวกันดังรูปที่ 2.15 และขดลวดทั้ง 2 เส้นนี้ จะแยกกันที่ปลายเป็นลักษณะ 2 เส้น แยกกัน ถ้าขดลวดเป็นของเฟส 1 อีกขดหนึ่งจะเป็นของเฟส 3 และทำนองเดียวกันถ้าขดหนึ่งเป็นของเฟส 2 อีกขดหนึ่งจะเป็นของเฟส 4 (กรณีของมอเตอร์ 4 เฟส)



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดรูปที่ 2.15 การพันแบบไบฟีลาถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จุดประสงค์ของการพันแบบไบโพลาร์ก็คือเพื่อให้พลังงานกับขั้วแม่เหล็กสเตเตอร์โดยการสลับขั้วแม่เหล็ก การกระตุ้นแต่ละเฟสอาจเป็นแบบใดแบบหนึ่งใน 3 แบบในรูปที่ 2.16



รูปที่ 2.16 การพันแบบโมโนโพลาร์

ในวงจรรูปที่ 2.16a เป็นแบบโมโนโพลาร์ ขั้วแม่เหล็กจะถูกกระตุ้นเป็นขั้วเหนือขั้วใต้เสมอซึ่งแสดงว่าไม่สามารถกลับขั้วแม่เหล็กได้ การกระตุ้นแบบนี้เป็นการกระตุ้นแบบขั้วเดียว (Unipolar Excited)

ในวงจรรูปที่ 2.16b ทิศทางกระแสในขดลวดสามารถสลับกันได้เนื่องจากเป็นวงจรบริดจ์ (Bridge Excited)

อย่างไรก็ตามต้องใช้ทรานซิสเตอร์ถึง 4 ตัวต่อขดลวด 1 ขด แบบนี้จะทำให้ทอร์กที่ความเร็วต่ำมากกว่าแบบไบโพลาร์ แต่ก็มีโอกาสที่ทรานซิสเตอร์จะพังได้เนื่องจากจัดเวลาผิดพลาด

ในวงจรรูปที่ 2.16c เกี่ยวกับคู่สายไบโพลาร์ และทรานซิสเตอร์ 2 ตัว โดยทำให้สเตเตอร์ถูกกระตุ้นเป็นขั้วแบบไบโพลาร์ ก็จะเกิดสนามแม่เหล็กคัปปลิง (Coupling) เมื่อขั้วใดขั้วหนึ่งถูกกระตุ้น ถ้าแทนการพันแบบไบโพลาร์ด้วยเส้นลวด 2 เส้น ที่แยกจากกันตามความแตกต่างของอินดักแตนซ์ (Inductance) จะปรากฏระหว่างขด 2 ขด ทำให้ตำแหน่งผิดไป

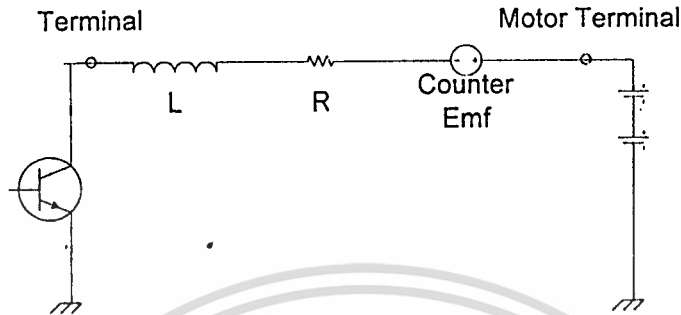
โดยทั่วไปประสิทธิภาพของมอเตอร์แบบแม่เหล็กถาวรกับแบบไฮบริดจ์ ที่ใช้แบบไบโพลาร์นี้จะได้ประสิทธิภาพดีกว่าแบบ โมโนโพลาร์

### 2.2.3 วงจรขับมอเตอร์ (MOTOR DRIVE CIRCUIT)

จุดประสงค์ของวงจรขับคือ เพื่อให้จะได้โวลต์เตจ และกระแสที่ถูกต้องไปยังมอเตอร์ในช่วงเวลาที่สั้น และลักษณะที่มีประสิทธิภาพ ทิศทางของกระแสในขดลวดของมอเตอร์แบบแม่เหล็กถาวร และแบบไฮบริดจ์ จะต้องใช้ในทิศทางที่เหมาะสมในเวลาต่างๆ สำหรับการขับมอเตอร์แบบวาริเอเบิลรีลัคแทนซ์ จะไม่ขึ้นกับทิศทางของกระแสที่ไหลในขดลวดเนื่องจากไม่มีแม่เหล็กอยู่ในตัวมันเลย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

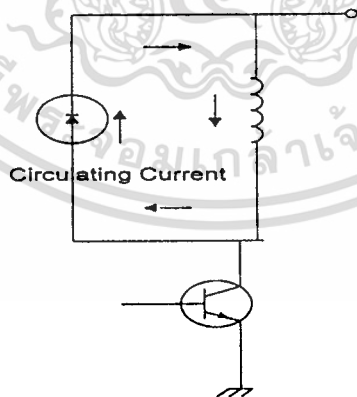
วงจรขับเคลื่อนมอเตอร์จะรับสัญญาณควบคุมมาจากวงจรจัดลำดับลอจิก ในการสร้างวงจรขับเคลื่อนมอเตอร์จะมี ปัญหาเกี่ยวกับอินดักแทนซ์และรีซิสแทนซ์ (Resistance) อนุกรมกันในวงจรสมมูลย์ ของมอเตอร์ดังรูปที่ 2.17 อีกทั้งในการหมุนยังมีปัญหาเพิ่มขึ้นเกี่ยวกับแรงดันไฟกลับ (Back Electromotive Force : Back EMF) นอกจากนี้ ยังต้องคำนึงถึงแหล่งจ่ายไฟดีซีและการใช้กับการป้องกัน เพาเวอร์ทรานซิสเตอร์ด้วย



รูปที่ 2.17 วงจรสมมูลย์ของสเตปปีงมอเตอร์

จากรูปที่ 2.17 เมื่อทรานซิสเตอร์หยุดทำงานจะเกิดสไปค์โวลต์เตจ (Spike Voltage) เนื่องจากขดลวด (ตัวเหนี่ยวนำ) ซึ่งอาจทำให้ทรานซิสเตอร์พังได้ จึงต้องมีวงจรป้องกันและลดลักษณะแบบนี้อย่างรวดเร็ว โดยวิธีต่อไปนี้

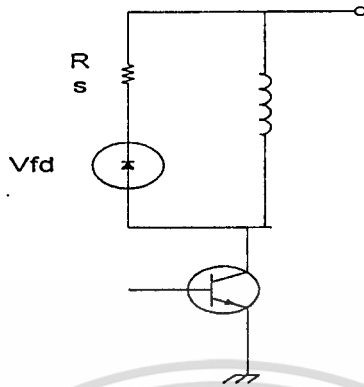
1. ลดโดยใช้ไดโอด (Diode Suppressor) ใช้ไดโอดต่อคร่อมขนานกับขดลวดดังรูปที่ 2.18 กระแสจะไหลอยู่ในวงจรขดลวดกับไดโอด หลังจากทรานซิสเตอร์หยุดทำงาน ซึ่งกระแสนี้จะลดลงไปตามเวลา แต่ก็ใช้เวลานานมากจึงจะเกิดทอร์กก็ต่ออยู่



รูปที่ 2.18 วงจร Diode Suppressor

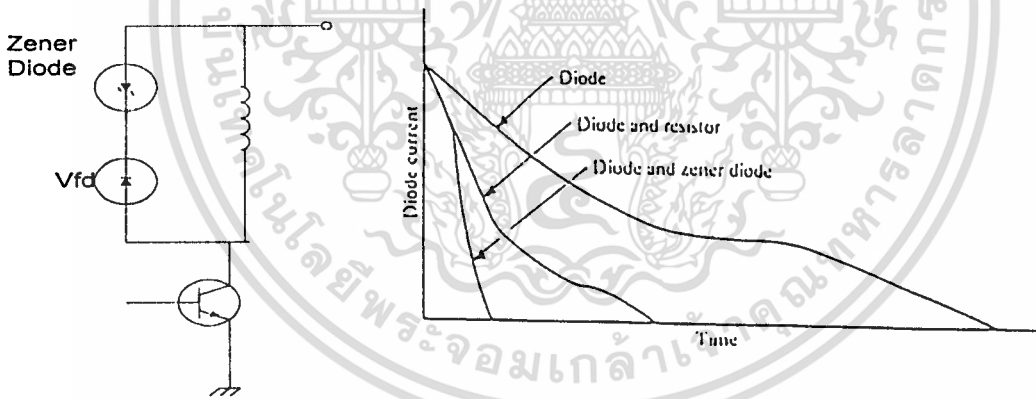
2. โดยใช้ไดโอดและตัวต้านทาน (Diode and Resistor Suppressor) โดยต่อความต้านทานอนุกรมกับ ไดโอดและต่อขนานคร่อมขดลวด ดังรูปที่ 2.19 วิธีนี้จะช่วยลดเวลา ในการที่จะทำให้กระแสหยุดไหลวน ซึ่งใช้ ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตัวต้านทานค่ามากกระแสก็จะหยุดไหลเร็วขึ้น แต่ทรานซิสเตอร์ต้องทนโวลต์เตจคร่อม คอลเลกเตอร์- อิมิเตอร์ เพิ่มขึ้นด้วย



รูปที่ 2.19 วงจร Diode and Resistor Suppressor

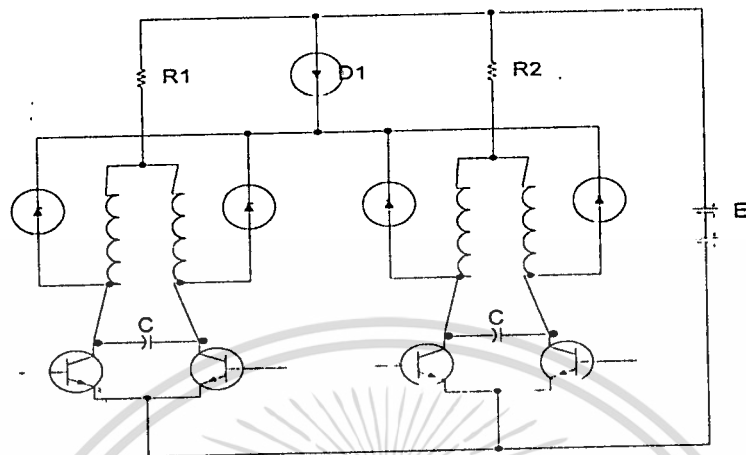
3. โดยใช้ซีเนอร์ไดโอด (Zener Diode Suppressor) โดยการต่อซีเนอร์ไดโอดดังรูปที่ 2.20 การต่อแบบนี้เมื่อเปรียบเทียบกับ 2 แบบแรก แล้วจะมีประสิทธิภาพมากกว่า ซึ่งสามารถลดกระแสที่เกิดขึ้นได้เร็วกว่า



รูปที่ 2.20 วงจร Zener Diode Suppressor

4. โดยใช้คอนเดนเซอร์ (Condensor suppressor) วิธีนี้มักจะใช้กับสแต็ปป์มอเตอร์แบบไบเฟลา การต่อวงจรดังรูป 2.21 ของมอเตอร์ 4 เฟส ตัวคอนเดนเซอร์ที่ต่ออยู่ระหว่างเฟส 1 กับเฟส 3 และเฟส 2 กับเฟส 4 จะทำหน้าที่ดังนี้ ในตอนที่ทรานซิสเตอร์ T1 หยุดทำงาน ในการทำงานแบบกระตุ้นเฟสเดียว T2 หรือ T4 เริ่มทำงานแต่ T3 ยังไม่ทำงาน เนื่องจากเฟส 1 เฟส 3 ต่ออยู่ในลักษณะไบเฟลา กระแสที่เกิดขึ้นหลังจาก T1 หยุดทำงานจะไหลวนไปตามเส้นไขปลา ดังแสดงในรูป ถ้า T3 หรือ T4 เริ่มทำงานแต่ T3 ยังไม่ทำงาน ประจุที่ชาร์จ เก็บไว้ในคอนเดนเซอร์จะช่วยเพิ่มทอร์กให้ โดยการปล่อยกระแสไหลผ่านเฟส 1 ทั้งนี้การต่อคอนเดนเซอร์สามารถใช้กับไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งยังมีให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การกระตุ้นแบบ 2 เฟส ได้ด้วย สำหรับค่า  $R_1, R_2$  ที่ใช้ในวงจรนี้ควรทำให้เหมาะสม นอกจากนั้นคอนเดนเซอร์ ยังช่วยลดการสั่นของมอเตอร์ โดยเปลี่ยนพลังงานกลเป็นพลังงานความร้อนแทน



รูปที่ 2.21 วงจร Condenser Suppressor

### 2.3 ส่วนควบคุม ( Controller )

วงจรขับเคลื่อนมอเตอร์ จะต้องมีส่วนควบคุมทำให้กระแสไฟดีซี เรียงลำดับเข้าไปตามเฟสต่างๆของมอเตอร์ให้หมุนไปตามองศาที่กำหนดไว้ การที่ทำให้กระแสไฟดีซีเรียงลำดับเข้าไปที่มอเตอร์ได้ ทำได้โดยการใช้งานของสวิตช์ ถ้าให้การทำงานโดยใช้รีเลย์ (Relay) หรือมือโยก สเตปมอเตอร์ก็จะหมุนเหมือนกันแต่หมุนช้า เนื่องจากรีเลย์ หรือมือโยกให้การเปลี่ยนแปลงของสวิตช์ช้า และไม่เป็นการสะดวกในการที่จะสั่งให้สเตปมอเตอร์หมุนได้ตามต้องการ ไมโครคอนโทรลเลอร์จึงถูกนำมาใช้งานควบคุมวงจรขับเคลื่อนมอเตอร์ได้เป็นอย่างดี และสามารถโปรแกรมการทำงานที่ละเอียดอ่อนซับซ้อนได้ตามต้องการ

ในการนำเอาไมโครคอนโทรลเลอร์ไปใช้งานนั้น จำเป็นจะต้องให้ไมโครคอนโทรลเลอร์สามารถติดต่อกับโลกภายนอกได้ คือ ให้นั้นสามารถส่งสัญญาณออกมาควบคุมอุปกรณ์ต่างๆ ได้ เช่น สเตปมอเตอร์ ส่วนที่ทำให้ไมโครคอนโทรลเลอร์สามารถติดต่อกับโลกภายนอกได้ที่รู้จักกันดีก็คือพอร์ต (Port) ซึ่งมีอยู่หลายลักษณะด้วยกัน เช่น เป็นไอซีแบบไตรสเตต (Tri-state) หรือพวกแลตช์ (Latch) เหล่านี้สามารถนำไปต่อใช้งานได้ง่ายที่สุด โดยตัวไมโครคอนโทรลเลอร์จะเป็นตัวควบคุมการอ่าน การเขียนพอร์ต หากเป็นการอ่านข้อมูลจากพอร์ตก็มักจะใช้ไอซีแบบไตรสเตตเป็นพอร์ตอินพุต โดยตัวไมโครคอนโทรลเลอร์จะส่งสัญญาณ ไปเปิดเกตของไตรสเตตนี้ให้ข้อมูลเข้ามาสู่สายข้อมูล (Data bus) และเข้าสู่ไมโครคอนโทรลเลอร์หรือ CPU ต่อไป แต่สำหรับมาแลตช์ไว้ที่ตัวมัน CPU จะรับสัญญาณมาทริก เพื่อให้อุปกรณ์ภายนอกนั้นรับสัญญาณจากตัวแลตช์นี้ไปอีกทีหนึ่ง ที่ทำเช่นนี้ก็เพราะว่าตัวไมโครคอนโทรลเลอร์หรือ CPU นี้ทำงานเร็วมากซึ่งช่วงของการส่งข้อมูลออกจากพอร์ตจะใช้เวลาไม่กี่ไมโครเซค ซึ่งอาจทำให้อุปกรณ์ภายนอกนั้นรับไม่ทัน

เอกสารนี้เป็นเอกสารลิขสิทธิ์ของสถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง ไม่ควรเผยแพร่โดยไม่ได้รับอนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

## 2.4 การสื่อสารข้อมูลของพอร์ตอนุกรม

การสื่อสารข้อมูลอนุกรมเป็นการรับหรือส่งข้อมูลในลักษณะกลุ่มของบิต คราวละหนึ่งบิต เรียงลำดับ เร็วไปจนถึงสิ้นสุดการสื่อสารแบบนี้จะมีข้อแตกต่างจากการสื่อสารแบบขนานเป็นอย่างมากเนื่องจากข้อมูลมีการ โอนย้ายมาพร้อมกันจึงมีความจำเป็นต้องใช้ จำนวนเส้นสัญญาณมากขึ้นตามจำนวนบิตของข้อมูลด้วย ในขณะที่ การสื่อสารแบบอนุกรมนั้นต้องการเส้นสัญญาณเพียงสอง หรือสามเส้นเท่านั้น ดังนั้นการสื่อสารแบบขนานจึง ไม่เหมาะสมในการสื่อสารกับอุปกรณ์ภายนอกเป็นระยะทางไกลๆ เพราะจะทำให้สิ้นเปลืองค่าใช้จ่ายมาก

### 2.4.1 ความเร็วของการสื่อสารข้อมูลอนุกรม

เนื่องจากการสื่อสารแบบอนุกรมเป็นการรับส่งข้อมูลในลักษณะของกลุ่มบิตข้อมูล (bit stream) ดังนั้น จึงต้องให้ความสนใจในการพิจารณาถึงเรื่องอัตราความเร็วในการรับส่งบิตเหล่านี้ เป็นลำดับแรก โดยทั่วไปมัก จะระบุกันในหน่วยของจำนวนบิตข้อมูลภายในเวลาหนึ่งวินาทีที่เรียกว่าอัตราบิตตามค่ามาตรฐานเหล่านี้ได้แก่ 110 150 300 1200 2400 4800 9600 19200 บิต ข้อมูลทั้ง 8 บิตนี้หากว่าถูกส่งออกมาด้วยอัตรา 2400 บิต จะใช้เวลาในการส่งข้อมูล 1 บิต มีค่าเท่ากับ  $1/2400$  หรือ  $416 \mu\text{s}$  และเวลาในการส่งข้อมูลทั้ง 8 บิตมีค่าเท่ากับ  $8 \times 416$  หรือ  $3328 \mu\text{s}$

### 2.4.2 รูปแบบของการส่งข้อมูลอนุกรม

การสื่อสารข้อมูลแบบอะซิงโครนัส จะใช้การแปลงข้อมูลขนานให้เป็นอนุกรมแล้วเพิ่มเติมบิตบางอย่าง ร่วมไปกับการส่งข้อมูลจริงได้แก่

บิตเริ่มต้น (Start Bit)

บิตเริ่มต้นมีหน้าที่สำหรับการบ่งบอกให้ทราบถึงตำแหน่งจุดเริ่มต้นก่อนบิตข้อมูล ตามปกติแล้วค่าของ บิตเริ่มต้นจะเป็นระดับลอจิกต่ำ

บิตแสดงสถานะความเป็นเลขคู่หรือเลขคี่ (Parity Bit)

บิตนี้มีหน้าที่เพื่อการตรวจสอบความถูกต้องของข้อมูล โดยทั่วไปเรียกว่าบิตพาริตี และจะนำไปต่อท้าย บิตข้อมูลค่าของบิตนี้จะขึ้นอยู่กับจำนวนค่าของบิตที่เป็น 1 ซึ่งจะเป็นได้สองลักษณะคือ พาริตีคู่ (Even Parity) หรือพาริตีคี่ (Odd Parity) ตัวอย่างเช่นในระบบที่ต่อกันโดยระบุว่าจะใช้พาริตีคี่ ทางด้านส่งจะนำค่าข้อมูลที่จะส่ง มาพิจารณาหาจำนวนของบิตที่มีค่า 1 เป็นเลขจำนวนคู่อยู่แล้วค่าของบิตพาริตีจะมีค่าเป็น 0 แต่หากว่าจำนวนบิต ที่มีค่าเป็น 1 เป็นเลขจำนวนคี่ ค่าของพาริตีก็จะมีค่าเป็น 1 การพิจารณาทางด้านรับเป็นการตรวจสอบจำนวนบิต ที่มีค่าเป็น 1 ของข้อมูลที่ได้รับทั้งหมดรวมทั้งบิตพาริตี ถ้ามีค่าเป็นเลขจำนวนคู่ แสดงว่าข้อมูลที่ได้รับเข้ามานี้ถูก ต้อง แต่หากไม่เป็นเลขจำนวนคู่แสดงว่าเกิดการผิดพลาดของข้อมูลขึ้นเป็นต้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### บิตสุดท้าย (Stop Bit)

บิตสุดท้ายเป็นบิตที่เพิ่มขึ้นเพื่อระบุถึงขอบเขตการสิ้นสุด ของกลุ่มบิตข้อมูลบิตสุดท้ายนี้สามารถโปรแกรมบิตได้คือ 1 บิต และ 2 บิต ดังนั้นกรณีของการส่งข้อมูลถูกส่งออกไปด้วยความเร็ว 2400 บอด เวลาโดยรวมในการส่งข้อมูล 1 ไบต์จะมีค่าเป็น  $12 \times 416 \mu\text{S}$  หรือ 4.99 mS

### 2.4.3 การส่งข้อมูลอนุกรมของ MCS-51

พอร์ตอนุกรมของ 8051 มีโครงสร้างการทำงานในแบบที่เรียกว่า ฟูลดูเพล็กซ์ (Full Duplex) ในการรับและส่งข้อมูลอนุกรมได้ในเวลาเดียวกันโดยทางด้านวงจรของตัวส่ง (Transmitter) ประกอบด้วยข้อมูลออกไปยังพอร์ตอนุกรมทางขาสัญญาณ TxD (พอร์ต 3.1) ส่วนวงจรด้านตัวรับ (Recciver) ประกอบด้วย SBUF เช่นเดียวกับสัญญาณข้อมูลอนุกรม ที่รับเข้ามาทางสัญญาณ RxD (พอร์ต3.0)

พอร์ตอนุกรมของ 8051สามารถโปรแกรมการทำงานได้หลายโหมดด้วยกันโดยเลือกที่บิต SM0 และ SM1 ซึ่งอยู่ในรีจิสเตอร์ควบคุม SCON การทำงานทั้ง 4 โหมด ของพอร์ตอนุกรม มีดังนี้

โหมด 1 : ใช้รับส่งข้อมูล 8 บิต โดยการส่งจะเลื่อนออกทีละบิต โดยส่งบิต D0 ออกไปก่อน ทางขา RxD และไม่มีการส่ง start bit แต่จะส่ง shift clock ทางขา TxD ความเร็ว 1/12 เท่าของ CPU CLOCK

โหมด 2 : ใช้สำหรับเชื่อมต่ออนุกรมแบบ UART (Universal Asynchronous Recciver/Transmitter) โดยส่งแบบ 10 บิต ข้อมูล 8 บิต 1 start bit และ 1 stop bit และสามารถเปลี่ยนแปลงอัตราความเร็วในการส่งข้อมูลได้โดยขึ้นกับบิต SMOD และ PCON

โหมด 3 : ใช้สำหรับการเชื่อมต่ออนุกรม แบบ UART โดยการใช้กลุ่มข้อมูลแบบ 11 บิต และสามารถเปลี่ยนแปลงอัตราความเร็วในการส่งข้อมูลได้ โดยควบคุมที่บิต SMOD และอัตราโอเวอร์โพล์ของ timer 1 นอกจากนี้โหมด 2 และ 3 ยังมีการดำเนินงานอีก แบบหนึ่ง โดยสามารถนำมาใช้ประโยชน์ในการสื่อสารข้อมูล แบบที่มีไมโครโปรเซสเซอร์หลายตัวทำงานร่วมกันได้ซึ่งมีชื่อเรียกว่า Multiprocessor Mode

### การอินเตอร์รัปต์ของพอร์ตสื่อสารอนุกรม

เนื่องจากการส่งหรือรับข้อมูลอนุกรมจะส่งทีละไบต์ 8051 จึงได้กำหนดให้บิตหรือแฟล็กซ์สถานะที่จัดรวมอยู่ในรีจิสเตอร์ SCON เช่นแฟล็กซ์ TI ซึ่งจะมีค่าเป็น 1 เมื่อข้อมูลได้ทำการส่งออกไปภายนอกเสร็จสิ้นแล้วเมื่อ แฟล็กซ์ RI ซึ่งมีค่าเป็น 1 เพื่อให้รู้ว่าได้รับข้อมูลผ่านเข้ามาทางพอร์ตอนุกรม เสร็จแล้วเมื่อ แฟล็กซ์ RI , TI นี้มีค่าเป็น 1 จะมีผลทำให้มีการอินเตอร์รัปต์ขึ้น ดังนั้นภายในโปรแกรมรับหรือส่งข้อมูลจะต้องทำการตรวจสอบจากสถานะของแฟล็กซ์เหล่านี้เองว่าเป็นการส่งข้อมูลหรือรับข้อมูล (vector ของ TI, RI อยู่ที่ 0023H)

### 2.4.4 กระบวนการรับและส่งข้อมูลอนุกรมของ 8051

การส่งข้อมูลออกทางพอร์ตอนุกรมของ 8051 จะเริ่มต้นขึ้นภายหลังเมื่อมีการเขียนข้อมูลลงใน SBUF ข้อมูลนี้จะถูกเลื่อนทีละบิต และส่งสัญญาณออกไปภายนอกโดยอัตโนมัติเมื่อข้อมูลเหล่านี้ได้ถูกส่งออกครบถ้วนแล้วจะทำให้ค่าของแฟล็กซ์ TI ให้เป็น 1 เพื่อแจ้งให้ทราบว่าขณะนี้ SBUF ว่างและพร้อมที่จะส่งข้อมูลไบต์ต่อ เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

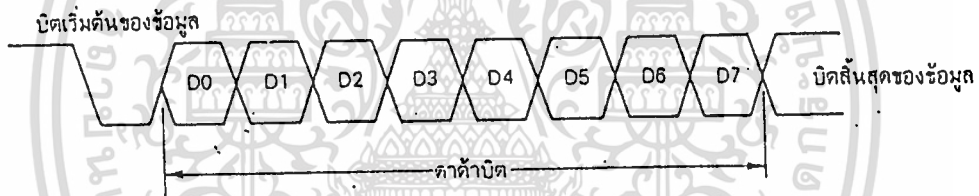
ไปแล้ว ในกรณีที่ผู้ใช้เขียนข้อมูลใหม่ลงในรีจิสเตอร์ SBUF โดยไม่รอให้แฟล็กซ์ T1 มีค่าเป็น 1 ก่อน จะมีผลทำให้ข้อมูลที่ส่งไปผิดพลาดได้

สำหรับการรับข้อมูลจากพอร์ตอนุกรมจะต้องเริ่มต้นโดยกำหนดเซ็ทค่าดังนี้ REN (Receiver Enable) ให้มีค่าเป็น 1 ก่อน หลังจากนั้นเมื่อมีข้อมูลภายนอกถูกส่งเข้ามาขัง 8051 ทีละบิต จนครบ และเมื่อบิตสุดท้ายถูกเลื่อนเข้ามาเรียบร้อยแล้ว ข้อมูลนั้นจะถูกย้ายมาเก็บไว้ในรีจิสเตอร์ SBUF แฟล็กซ์ RI จะมีค่าเป็น 1 (ถูกรีเซต) หลังจากนั้นจะเกิดการอินเตอร์รัปต์ขึ้น

ในโครงการนี้เราได้ทำการรับส่งข้อมูลโดยใช้วิธีการรับส่งข้อมูลอนุกรมโหมด 1 ดังนั้นจะอธิบายถึงการรับส่งข้อมูลอนุกรมโหมด 1 ซึ่งส่งข้อมูล 8 บิต, 1 start, 1 stop ดังนี้

#### 2.4.5 พอร์ตอนุกรมโหมด 1

การทำงานของพอร์ตสื่อสารข้อมูลแบบอนุกรมในโหมด 1 ข้อมูล 10 บิตถูกส่งออกไปภายนอกผ่านขา TXD และรับเข้ามาจากภายนอกผ่านขา RXD ข้อมูลที่รับหรือส่งจะประกอบด้วยบิตเริ่มต้นของข้อมูล 1 บิต (บิต 0) บิตข้อมูล 8 บิตที่เริ่มรับหรือส่งด้วยบิตต่ำสุดก่อน (LSB first) ตามด้วยบิตสิ้นสุดของข้อมูลอีกหนึ่งบิต (บิต 1) ดังรูปที่ 2.22



รูปที่ 2.22 แสดงข้อมูลที่รับและส่งในการทำงานของพอร์ตสื่อสารข้อมูลแบบอนุกรมในโหมด 1

ในการรับข้อมูล บิตสิ้นสุดของข้อมูลจะถูกนำไปไว้ในบิต RB8 ของรีจิสเตอร์ใช้งานเฉพาะ SCON ค่า baud rate ถูกกำหนดโดยอัตราการเกิด overflow ของ timer 1

การส่งข้อมูลจะเริ่มต้นโดยคำสั่งใดๆ ที่ใช้รีจิสเตอร์ SBUF เป็นรีจิสเตอร์ปลายทางสัญญาณ write to SBUF ที่เกิดขึ้นจะทำให้ 1 ถูกโหลดไปไว้ที่ D-flipflop ซึ่งเป็นตำแหน่งบิตที่ 9 ของรีจิสเตอร์สำหรับส่งข้อมูล ซึ่งประกอบขึ้นจากรีจิสเตอร์ SBUF (8 บิต) และ D-flipflop (1 บิต) ตำแหน่งบิตที่ 9 หรือ D-flipflop จะมีอินพุตต่อลงกราวด์และขา S ต่อกับสัญญาณ write to SBUF นอกจากนี้สัญญาณ write to SBUF จะถูกส่งไปบอกให้วงจรควบคุมการส่งข้อมูลทราบว่าขณะนี้มีความต้องการที่จะส่งข้อมูล การส่งข้อมูลจริงๆจะเริ่มขึ้นในช่วงสแตท 1 เฟส 1 ของเมทซินไซเคิลที่เกิดที่ถัดจากการเกิด overflow ครั้งถัดไปในเคาน์เตอร์ที่ถูกหารด้วย 16 ดังนั้นช่วงจังหวะการส่งข้อมูลแต่ละบิตจะสอดคล้อง (synchronized) กับสัญญาณที่ได้จากเคาน์เตอร์ที่ถูกหารด้วย 16 มีใช้กับสัญญาณ write to SBUF

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การส่งข้อมูลเริ่มต้นด้วยสัญญาณ SEND (แอกติฟที่สถานะลอจิก 0) ที่ถูกกระตุ้นให้เริ่มทำงานโดยเริ่มส่งบิตเริ่มต้นของข้อมูลอยู่ที่ตำแหน่งเอาต์พุทของรีจิสเตอร์สำหรับส่งข้อมูลแล้วค่า 1 ซึ่งไหลคไว้ในตำแหน่งที่ 9 ในตอนแรกจะอยู่ถัดจากบิตสูงสุดไปทางซ้าย ส่วนตำแหน่งอื่นๆทางซ้ายมีค่าเป็น 0 หหมด (ได้จาก D-flipflop ที่มีอินพุตต่อลงกราวด์) ในสถานะเช่นนี้วงจรตรวจจับศูนย์จะเริ่มทำงาน (ตรวจจับเพียง 7 บิต) โดยการส่งสัญญาณไปยังวงจรควบคุมการส่งข้อมูลให้ทำการเลื่อนข้อมูลเป็นครั้งสุดท้ายอีก 1 ครั้ง แล้วจึงหยุดการส่งข้อมูลให้สัญญาณ SEND หยุดแอกติฟ พร้อมทั้งเซตบิต T1 เหตุการณ์ที่เกิดขึ้นขณะส่งข้อมูลเสร็จสิ้นลงจะเกิดขึ้นในช่วงที่เกิด overflow ครั้งที่ 10 ของเคาน์เตอร์ที่ถูกหารด้วย 16 หลังจากสัญญาณ write to SBUF เกิดขึ้น

การรับข้อมูลเริ่มต้นเมื่อ MCS-51 ตรวจพบการเปลี่ยนสถานะจาก 1 เป็น 0 ที่ขา RXD ซึ่งการตรวจสอบขณะนี้ MCS-51 จะตรวจสอบด้วยความถี่ 16 เท่า ของค่า baud rate ในขณะนั้น เมื่อการเปลี่ยนสถานะของสัญญาณถูกตรวจพบเคาน์เตอร์หาร 16 จะถูกรีเซตในทันที และข้อมูล IFFH จะถูกไหลคไปไว้ในรีจิสเตอร์สำหรับรับข้อมูลซึ่งมีขนาด 9 บิต การรีเซตเคาน์เตอร์หาร 16 ก็เพื่อตั้งอัตราการ overflow ให้เริ่มตรงกับการเริ่มต้นรับข้อมูลของบิตถัดไปที่จะรับเข้ามา

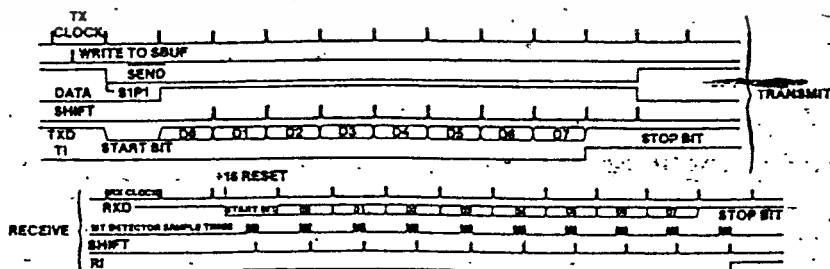
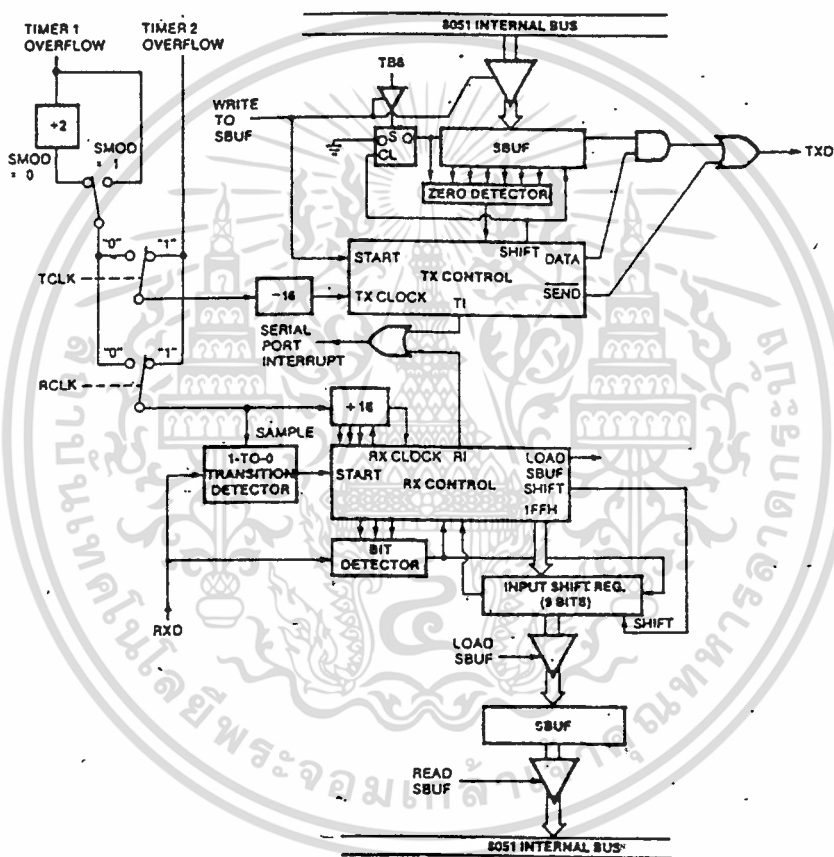
จากรูป เคาน์เตอร์จะถูกเพิ่มค่าอีกหนึ่งก็ต่อเมื่อ ไทม์เมอร์ 1 หรือ ไทม์เมอร์ 2 ที่ใช้เป็นตัวกำหนดค่า baud rate เกิด overflow ครบ 16 ครั้งช่วงเวลาในการรับข้อมูลในแต่ละบิตจะนำเอาอัตราการเกิด overflow มาเป็นตัวกำหนด โดยช่วงเวลาในการรับข้อมูลในแต่ละบิตจะถูกแบ่งออกเป็น 16 สเตทย่อย วงจรตรวจสอบข้อมูลในแต่ละบิต (bit detector) จะทำการตรวจสอบค่าข้อมูลในแต่ละบิตทุกๆสเตทที่ 7,8,9 ในช่วงเวลาของการรับข้อมูลในแต่ละบิต วงจรตรวจสอบข้อมูลจะตรวจสอบค่าสถานะที่ขา RXD ซึ่งมีข้อมูลรออยู่ ค่าที่รับมาจะถือว่าเป็นบิตข้อมูลที่ถูกดองก็ต่อเมื่อตรวจสอบว่าเป็นค่าเดียวกันอย่างน้อย 2 ครั้งจากการตรวจสอบทั้งหมด 3 ครั้ง การกระทำเช่นนี้เพื่อป้องกันสัญญาณรบกวน (noise rejection) ที่อาจเกิดขึ้นในระหว่างการรับและส่งข้อมูลแต่ในช่วงการรับข้อมูลของบิตแรก หากค่าที่ตรวจสอบในสเตทที่ 7,8,9 ไม่เป็น 0 วงจรที่ทำหน้าที่รับข้อมูลจะถูกรีเซตและระบบจะกลับไปเริ่มตรวจหาการเปลี่ยนสถานะจาก 1 เป็น 0 ที่ขา RXD ต่อไป ที่ทำเช่นนี้เพื่อยกเลิกการรับข้อมูล หากบิตเริ่มต้นของข้อมูลผิดพลาด แต่ถ้าบิตเริ่มต้นของข้อมูลถูกตรวจสอบแล้วมีค่าเป็น 0 การรับข้อมูลบิตต่อไปจะเริ่มขึ้นทันทีโดยบิตข้อมูลจะเลื่อนไปไว้ในรีจิสเตอร์สำหรับข้อมูล และบิตข้อมูลที่เหลือจะถูกรับตามมาเก็บไว้ในรีจิสเตอร์นี้เช่นกัน

ขณะที่บิตข้อมูลรับมาทางขวา บิตข้อมูลที่มีค่า 1 จะถูกเลื่อนออกไปทางซ้ายจนกระทั่งเมื่อบิตเริ่มต้นของข้อมูล(0) เลื่อนไปถึงตำแหน่งซ้ายสุดของรีจิสเตอร์ สำหรับรับข้อมูล มันจะส่งสัญญาณไปบอกวงจรควบคุมการรับข้อมูล (RX control block) ให้เลื่อนข้อมูลครั้งสุดท้ายอีก 1 ครั้ง เพื่อรับบิตสิ้นสุดของข้อมูล ไหลคค่าในรีจิสเตอร์ใช้งานเฉพาะ SBUF ด้วยข้อมูลในรีจิสเตอร์สำหรับรับข้อมูล ไหลคบิต RB8 ด้วยค่าบิตสิ้นสุดของข้อมูลและเซตบิต RI เป็นอันดับสุดท้าย สัญญาณที่จะไหลคข้อมูลไปไว้ใน SBUF และรับบิตสิ้นสุดของข้อมูลไปไว้ในบิต RB8 รวมทั้งการเซตบิต RI จะถูกสร้างขึ้นก็ต่อเมื่อเงื่อนไขต่อไปนี้เป็นจริงขณะสัญญาณ shift pulse ถูกสุดท้ายถูกสร้างขึ้น

บิต RI ถูกเคลียร์ (RI=0)

เอกสารนี้เป็นลิขสิทธิ์ของสถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง เพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ถ้าเงื่อนไขอย่างใดอย่างหนึ่งของทั้งสองอย่างนี้ไม่ตรง ข้อมูลที่รับเข้ามาจะหายไป โดยไม่สามารถเรียกคืนได้ แต่ถ้าเงื่อนไขทั้งสองถูกต้องบิตสิ้นสุดของข้อมูลจะถูกนำไปไว้ในบิต RB8 ของรีจิสเตอร์ SCON ส่วนบิตที่เป็นข้อมูลจำนวน 8 บิต จะถูกโหลดไปไว้ในรีจิสเตอร์ SBUF และบิต R1 จะถูกรีเซ็ต เมื่อถึงตอนนี้ไม่ว่าเงื่อนไขจะตรงหรือไม่ ระบบการรับส่งข้อมูลจะถือว่าการรับข้อมูลจำนวน 1 ไบต์ (รับข้อมูลเข้ามา 10 บิต แต่เป็นข้อมูลจริงๆเพียง 8 บิต) เสร็จสิ้นลงแล้วและเริ่มรีเซ็ตตัวเองเพื่อกลับไปรอรับการเปลี่ยนสถานะจาก 1 เป็น 0 ที่ขา RXD เพื่อรับข้อมูลไบต์ต่อไป (กลับไปรอรับบิตเริ่มต้นของข้อมูลใหม่)



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น รูปที่ 2.23 แสดงการทำงานของพอร์ตสื่อสารข้อมูลแบบอนุกรมโหมด 1 ทุกครั้งที่มีการนำไปใช้

## 2.5 การเชื่อมโยง MCS-51 กับหน่วยความจำ

การจัดหน่วยความจำสำหรับ MCS-51 แบ่งเป็น 2 ส่วนคือ

- Data Memory
- Program Memory

### 2.5.1 Data Memory

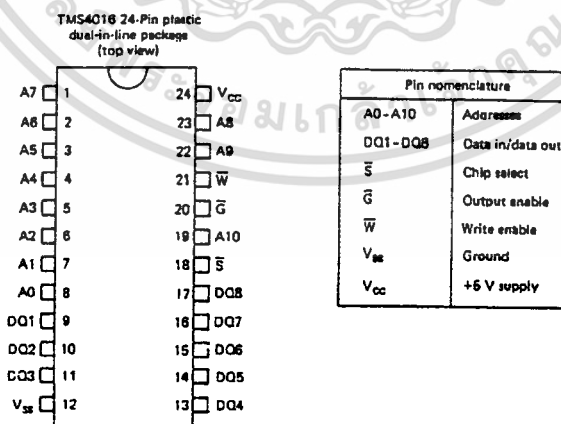
เป็นหน่วยความจำที่ใช้เก็บข้อมูล มีความจุได้ถึง 64-Kbytes เป็นหน่วยความจำประเภทอ่านและเขียนได้ (RAM) สัญญาจาก MCS-51 ที่ใช้ในการอ่านข้อมูลจาก RAM คือ RD และสัญญาที่ใช้ในการเขียนข้อมูลลง RAM คือ WR

**RAM (Random Access Memory)**

เป็นหน่วยความจำที่สามารถเขียนและอ่านข้อมูลได้ ข้อมูลจะสูญหายทันทีที่ขาดไฟเลี้ยง แบ่งเป็น 2 ชนิดคือ Static RAM และ Dynamic RAM

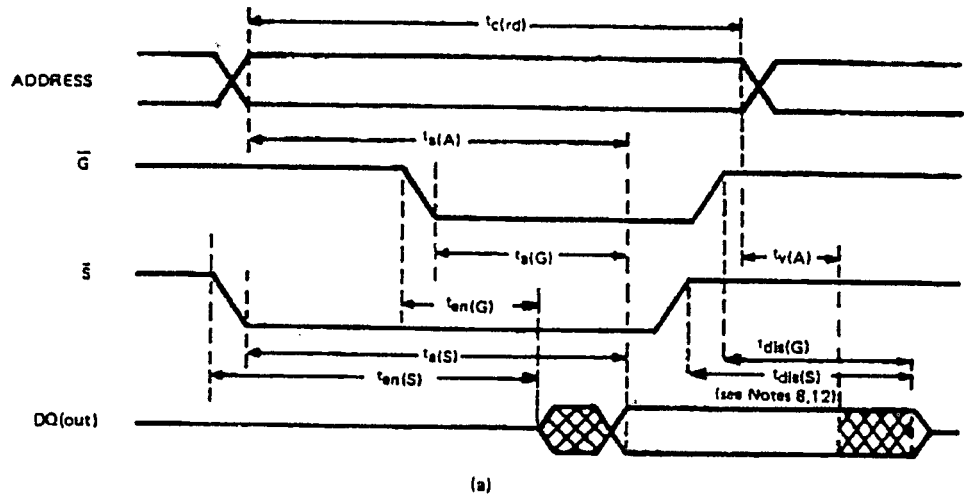
#### STATIC RAM

เป็นหน่วยความจำชนิดหนึ่งที่ย่อยในการนำมาใช้งานแต่ราคาสูงเมื่อเทียบกับ Dynamic RAM ซึ่งราคาถูกกว่าแต่ยุ่งยาก คือทำการ refresh หน่วยความจำตลอดเวลา Static RAM ที่มีข้อยุ่งยากหลายข้อด้วยกันหลายเบอร์ตามขนาดความจุของหน่วยความจำ เบอร์ที่นิยมมากสำหรับใช้งานที่ไม่ใหญ่มากนักคือเบอร์ TMS4016 หรือ 6116 ซึ่งมีขนาดความจุ 2K X8 ดังมีการวางขาตามรูป 2.24 และผังเวลาของ TMS4016 ดังแสดงในรูปที่ 2.24

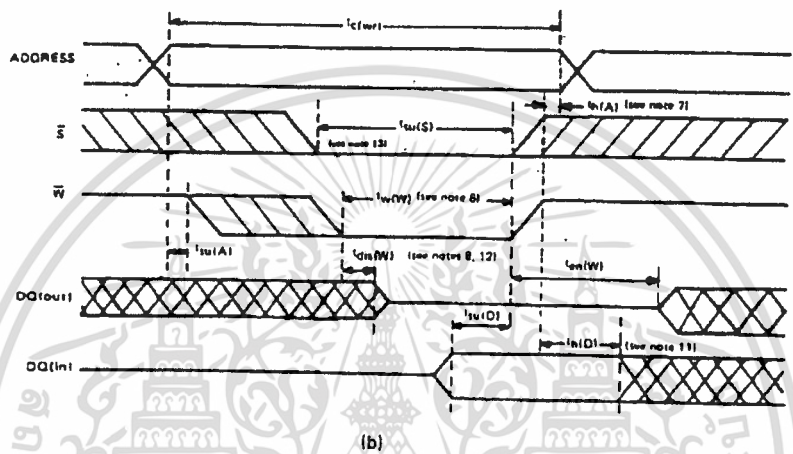


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น ผู้ที่นำข้อมูนี้เป็นต้นฉบับหรือดัดแปลงหรือต้องอ้างถึงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 2.24 แสดงการจัดวางขาของ Static RAM เบอร์ TMS4016



(a)



(b)

รูปที่ 2.25 แสดง (a) Read cycle timing for the TMS4016  
 (b) Write cycle timing

### 2.5.2 PROGRAM MEMORY

เป็นหน่วยความจำที่ใช้เก็บ โปรแกรมเป็นหน่วยความจำที่ใช้เก็บข้อมูลได้อย่างเดียว (ROM) สัญญาณจาก MCS-51 ที่ใช้ในการอ่าน ROM คือ PSEN

ROM (Read Only Memory)

เป็นหน่วยความจำแบบถาวร เมื่อขาดไฟเลี้ยงข้อมูลไม่สูญหาย เป็นหน่วยความจำชนิดอ่านข้อมูลได้อย่างเดียว

EPROM (Erasable Programmable read-only Memory)

เป็นหน่วยความจำแบบถาวรชนิดหนึ่งซึ่งผู้ใช้นำมาโปรแกรมเองได้และสามารถใช้แสง UV ลบข้อมูลได้

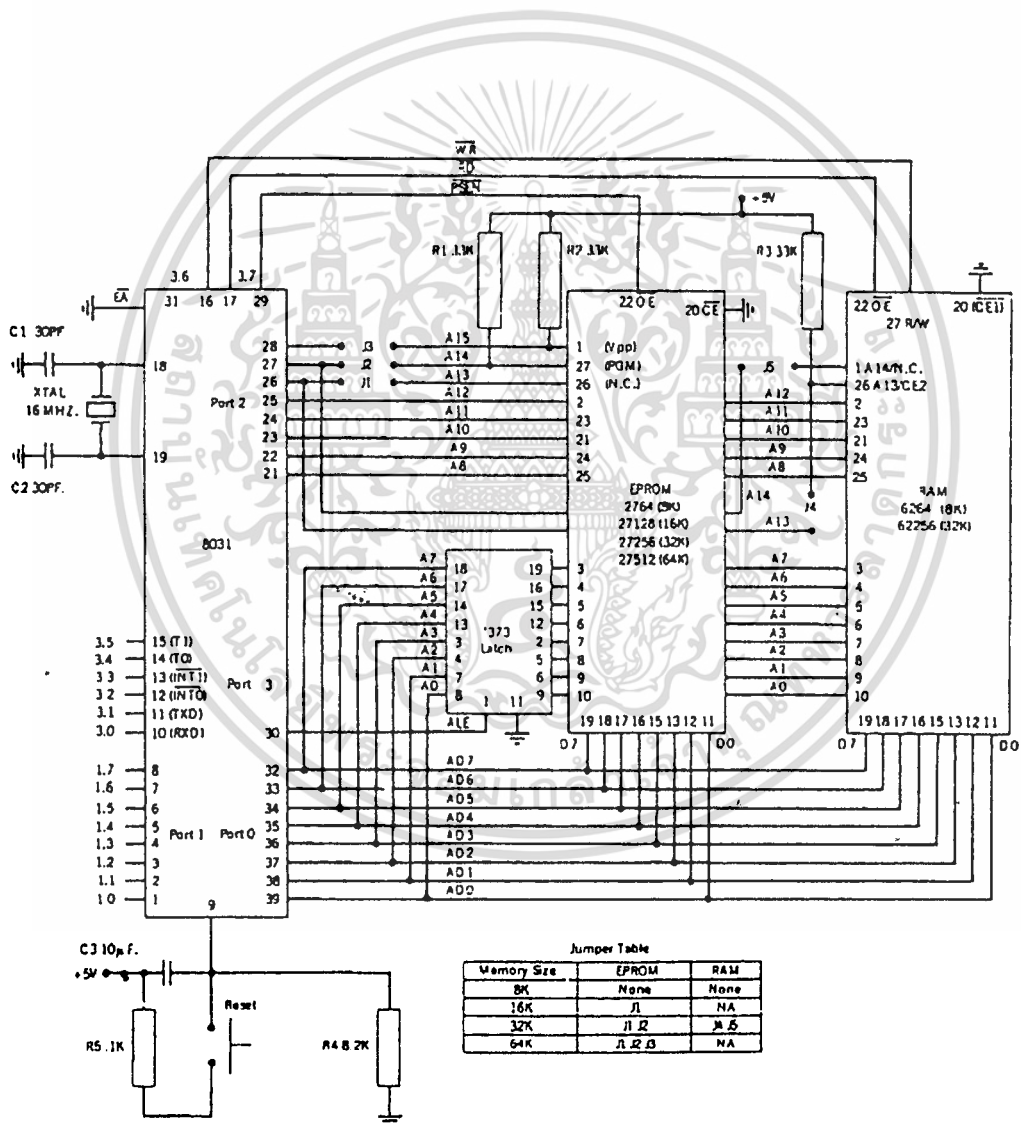
### 2.5.3 การถอดรหัสตำแหน่งของหน่วยความจำ

การถอดรหัสตำแหน่งของหน่วยความจำมีความจำเป็นอย่างไร เพราะว่าหน่วยความจำที่นำมาใช้มีขนาดเล็กกว่า 64 Kbytes ดังนั้นจึงต้องนำมาเรียงกันแล้วใช้วงจรถอดรหัสตำแหน่งของหน่วยความจำ แยกเป็นแต่ละช่วง ถ้าใช้หน่วยความจำขนาด 4 Kbytes จำนวน 8 ตัวจะเท่ากับ 32 Kbytes ถ้าให้ตัวแรกเริ่มที่แอดเดรส 8000H ตัวที่ 2 ก็จะมีที่แอดเดรส 9000H

เมื่อมีการแก้ไข ฟังก์ชัน อีกทั้งยังมีเหตุเปลี่ยนแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.5.4 การต่อ MCS-51 เข้ากับ EPROM และ RAM

การต่อใช้งานนี้อาจไม่ใช้ Decoder มาถอดรหัสตำแหน่งของหน่วยความจำก็ได้ ถ้าเราติดตั้งครั้งละ 1 ตัวเท่านั้น เพราะแอสติครสของ Program Memory และ Data Memory ใช้แอสติครสซ้อนกันอยู่ สัญญาณที่ใช้อ่านข้อมูลจาก Program Memory (EPROM) ก็คือ PSM และสัญญาณที่ใช้อ่านข้อมูลจาก Data Memory (RAM) ก็คือ RD ส่วนการเขียนข้อมูลลง RAM จะใช้ WR ดังในรูปที่ 2.26



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งรูปที่ 2.26 การต่อ MCS-51 เข้ากับ EPROM และ RAM เอกสารทุกครั้งที่มีการนำไปใช้

## 2.5.5 การเชื่อมโยง 8255 กับ MCS-51

การทำงานมีอยู่ 3 โหมด คือ

โหมด 0 มีการทำงานแบบ BASIC I/O ไม่มี Handshake

โหมด 1 โหมดนี้ใช้พอร์ท A,B ในการรับส่งข้อมูลและใช้พอร์ท C ในการตรวจสอบสัญญาณ (Handshake)

โหมด 2 โหมดนี้ใช้พอร์ท A ในการรับส่งข้อมูล 2 ทิศทางและพอร์ท B ในการระบุส่งข้อมูลและใช้พอร์ท C บิต 0,1,2 ในการรับส่งข้อมูลบิต และบิต 4,5,6 เป็นสัญญาณ Handshake

การถอดรหัสตำแหน่งพอร์ทของ 8255 ดังแสดงในรูป ได้เบอร์พอร์ทดังนี้

C000H-CFFFH

D000H-DFFFH

E000H-EFFFH

F000H-FFFFH

8000H-8FFFH

9000H-9FFFH

A000H-AFFFH

B000H-BFFFH

และใช้เอาต์พุตที่ตำแหน่ง (F000H-FFFFH) มาถอดรหัสร่วมกับ A9,A10,A11 โดยใช้ 74LS 138 ได้พอร์ท

F800H-F9FFFH

FA00H-FBFFFH

FC00H-FDFFFH

FE00H-FFFFH

F000H-F1FFFH

F200H-F3FFFH

F400H-F5FFFH

F600H-F7FFFH

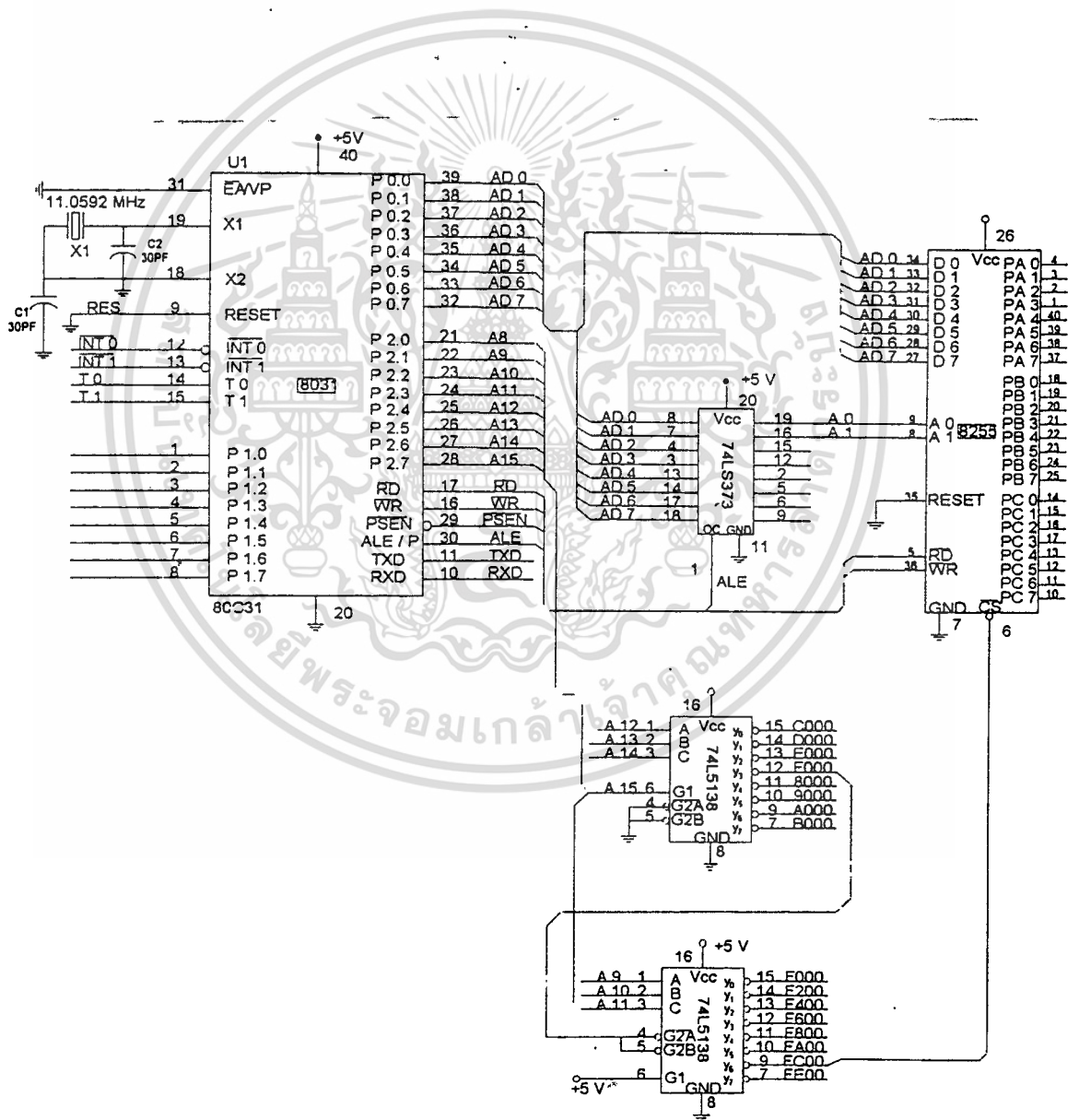
สัญญาณเลือกกริธ 8255 ใช้เบอร์พอร์ท (FC00H-FDFFFH) และ A0,A1 ของ 8255 จะได้เบอร์พอร์ทดังรูปที่

2.27

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

I/O ADDRESS	8255 PORT
FC00H	A
FC01H	B
FC02H	C
FC03H	CONTROL

รูปที่ 2.27 แสดงตาราง I/O Address ของ 8255  
 การเชื่อมโยง 8255 เข้ากับ CPU ดังแสดงการต่อในรูปที่ 2.28



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
 ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้า **รูปที่ 2.28 แสดงการเชื่อมโยง 8255 เข้ากับ CPU** เอกสารทุกครั้งที่มีการนำไปใช้

ตัวอย่าง การเขียนโปรแกรมให้พอร์ท C บิต 7 ส่งข้อมูล 0 และ 1 ออกมาอย่างต่อเนื่องเมื่อ พอร์ทของ 8255 อยู่ที่ FC00H-FC03H

จะได้ CONTROL WORD เมื่อ PC7 = 1 คือ 0XXX111b  
= 0 คือ 0XXX110b

ตัวอย่างโปรแกรม

```

ORG 800h      ;
START; MOV A,#0Fh      ; MODE CONTROL WORD PC7 = 1
MOV DPTR,# 0FC03h    ;
MOVX @ DPTR,A      ;
MOV A,#0EH      ; MODE CONTROL WORD PC7 = 0
MOV DPTR,# 0FC03h    ;
MOVX @ DPTR,A      ;
SJMP START
END

```

MODE1: STROBE I/O

โหมดนี้จะใช้พอร์ท A,B ในการส่งข้อมูล และใช้พอร์ท C ตรวจสอบความพร้อม เมื่อโปรแกรม(โหมด

1) แล้วพอร์ท C จะระบุเป็นขาสัญญาณดังนี้

เมื่อเป็นอินพุทพอร์ท

```

PC0 = INTRB
PC1 = IBFB
PC2 = INTEB
PC3 = INTRA
PC4 = INTEA
PC5 = IBFA
PC6 = I/O
PC7 = I/O

```

เมื่อเป็นเอาต์พุทพอร์ท

```

PC0 = INTRB
PC1 = OBFB
PC2 = INTEB
PC3 = INTRA

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

PC6 = INTEA

PC7 = OBFA

โดยที่ INTR คือ INTERRUPT REQUEST (การร้องขออินเทอร์รัพท์)  
 IBF คือ INPUT Buffer full (บัฟเฟอร์เต็ม)  
 STB คือ STROBE (สโตน)  
 I/O คือ INPUT/OUTPUT (อินพุท/เอาต์พุท) โปรแกรมได้  
 INTE คือ INTERRUPTS ENABLE (อินเทอร์รัพท์ อีนาเบิล)  
 ACK คือ ACKNOWLEDGE

## 2.6 การแปลงสัญญาณอนาลอกเป็นดิจิทัล

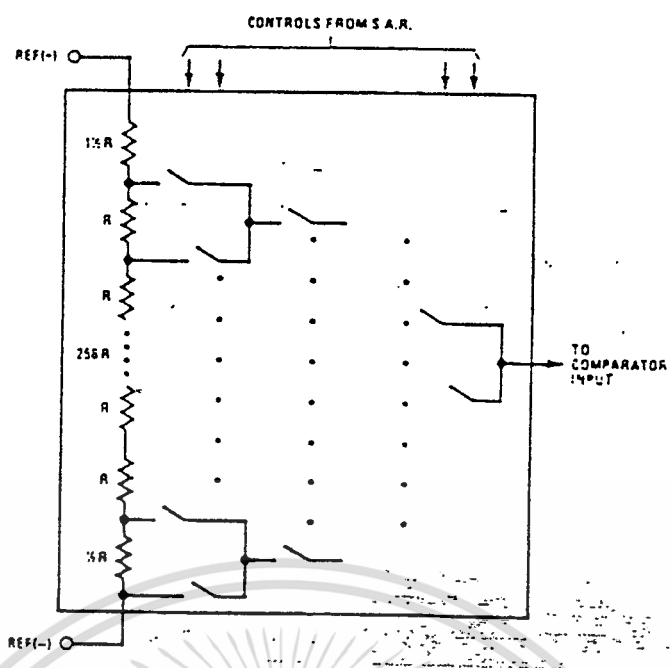
### ADC 0809

เป็น IC ประเภท CMOS แปลงสัญญาณอนาลอกเป็นดิจิทัล 8 บิต โดยใช้หลักการ successive approximation คอนเวอร์เตอร์มีลักษณะเป็น High Impedance จะเป็นตัวเปรียบเทียบกับ 256R โวลต์แดงดีไวเดอร์ (Voltage divider) โดยใช้การสับสวิตช์ (Switch tree) ต่อเนื่องกันไป 8 ช่องสัญญาณรวมจะเปลี่ยนเป็น 8 ช่องสัญญาณอนาลอกได้โดยตรง

#### ลักษณะของ ADC 0809

- Resolution 8 บิต
- Total error LSB และ +1 LSB
- ไม่มีรหัสผิดพลาด
- เวลาในการคอนเวอร์ชัน 100 us
- จ่ายไฟที่ 5 Vdc
- 8 ช่องสัญญาณรวมด้วย latched control logic
- ง่ายต่อการต่อเข้ากับไมโคร โปรเซสเซอร์
- เอาต์พุท TTL Voltage level
- No zero adjust required
- ใช้ power ค่า 15 mW
- latched TRI-STATE เอาต์พุท

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
 ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.29 แสดง Resistor Ladder and Switch Tree

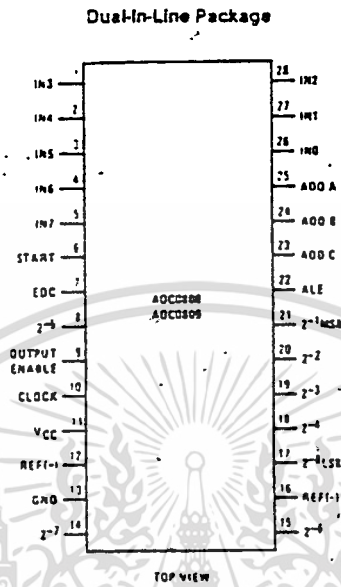
จากรูปที่ 2.30 ตาราง แสดง Input state ของ address line เพื่อเลือกชั้นแนล (Channel) ต่างๆ แอคเตรส จะถูกนำไปถอดรหัส เปลี่ยนจาก Low เป็น High ดังตาราง

SELECTED ANALOG CHANNEL	ADDRESS LINE		
	C	B	A
IN0	L	L	L
IN1	L	L	H
IN2	L	H	L
IN3	L	H	H
IN4	H	L	L
IN5	H	L	H
IN6	H	H	L
IN7	H	H	H

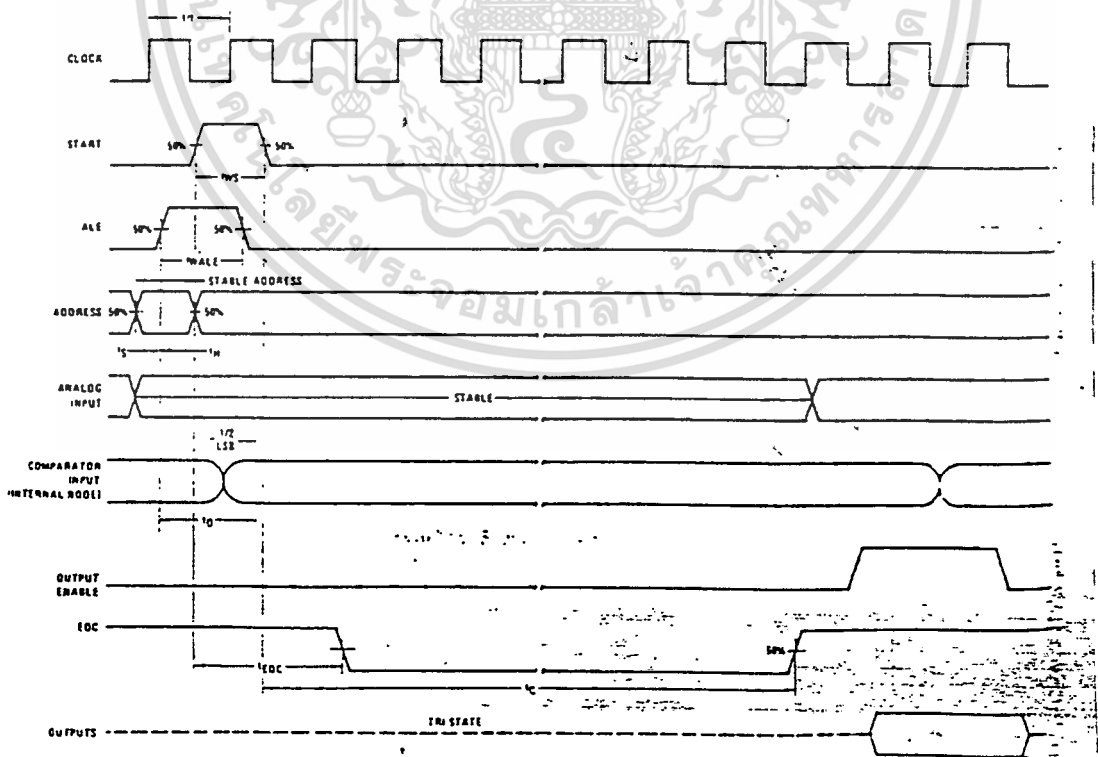
รูปที่ 2.30 แสดง Input state ของ address line เพื่อเลือกชั้นแนล (Channel) ต่างๆ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Connection Diagram



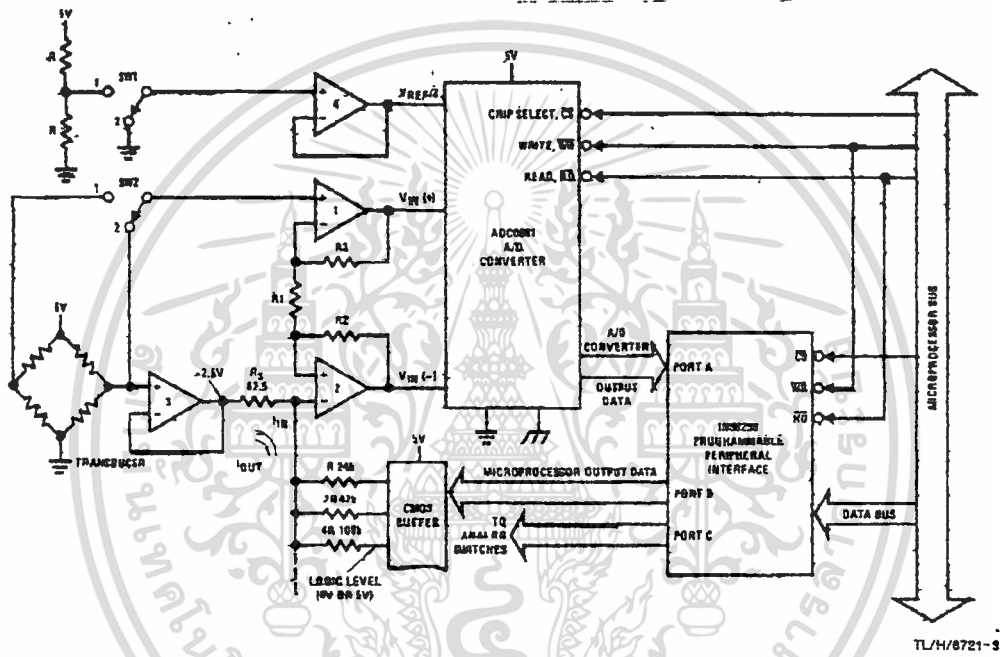
รูปที่ 2.31 แสดง Connection Diagram



รูปที่ 2.32 แสดง Timing Diagram

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Application Information



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับรูปที่ 2.33 แสดง Application Information ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
 ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### บทที่ 3 การออกแบบและการสร้าง

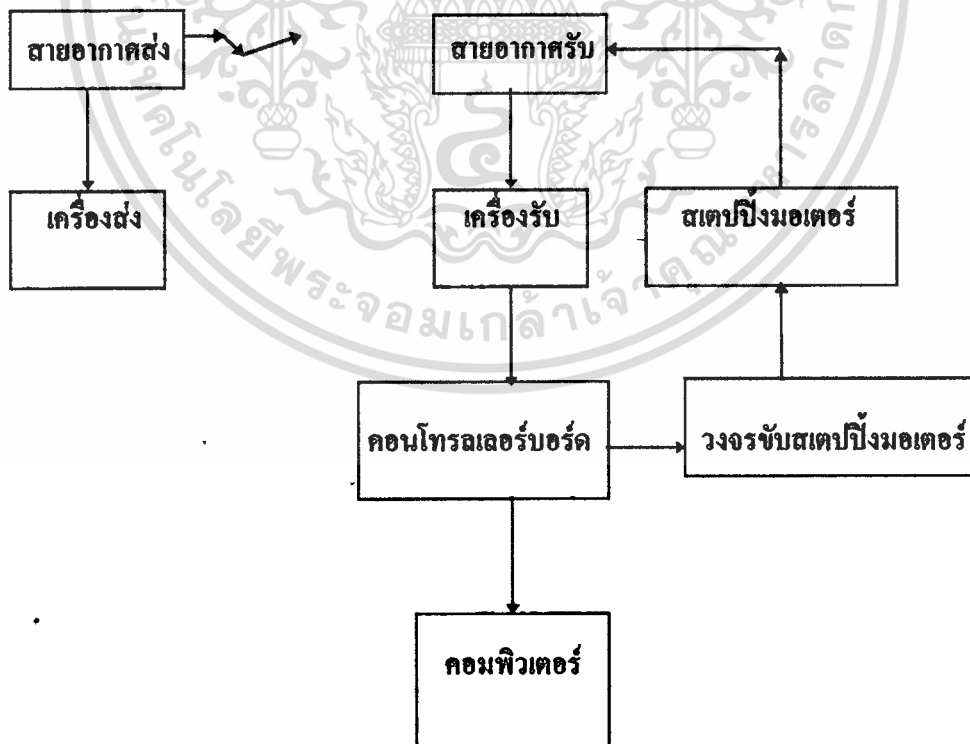
เราได้ทำการออกแบบการวัดแพทเทิร์นของสายอากาศ

#### 3.1 ระบบการวัดรูปแบบแพทเทิร์นของสายอากาศ

ระบบการวัดรูปแบบแพทเทิร์นของสายอากาศจะเป็นการวัดค่าสนามไฟฟ้า (E) โดยระบบจะประกอบ

ด้วย

- เครื่องวิเคราะห์โครงข่ายเป็นตัวส่งคลื่น 300 MHz
- เครื่องรับ Measuring Receiver ML521B เป็นตัวรับคลื่น ความถี่ 300 MHz
- สายอากาศความถี่ 300 MHz
- ชุดเฟืองและแท่นหมุนที่ต่อเข้ากับสายอากาศรับ
- แท่นจับสายอากาศส่ง
- ชุดขับเคลื่อนมอเตอร์
- ชุดคอนโทรลเลอร์บอร์ด
- คอมพิวเตอร์



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
รูปที่ 3.1 บล็อกไดอะแกรมแสดงการทำงานของระบบ  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 3.2 การสร้างอุปกรณ์ฮาร์ดแวร์ (hardware)

โครงการนี้เราได้ทำการสร้างอุปกรณ์ hardware ที่ใช้ในการวัดแพทเทิร์นของสายอากาศ โดยแบ่งเป็น 3

ส่วนคือ

- ชุดเฟืองและแท่นหมุน
- ชุดขับสเตปป์มอเตอร์
- ชุดคอนโทรลเลอร์บอร์ด

#### 3.2.1 ชุดเฟืองและแท่นหมุน

เนื่องจากเราใช้สเตปป์มอเตอร์ สเตปละ 1.8 องศา

ดังนั้น เราต้องใช้เฟืองยึดติดกับเพลา 90 ฟัน

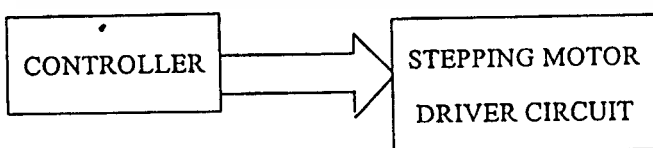
และใช้เฟืองยึดติดกับสเตปป์มอเตอร์ 25 ฟัน

$$\begin{aligned} \text{อัตราทดเฟือง} &= 25 : 90 \\ &= 0.5 : 1.8 \end{aligned}$$

ดังนั้น เราจะได้ สเตปละ 0.5 องศาต่อ 1 สเตป เมื่อใช้เฟืองทด

เราจะทำการออกแบบระบบปรับทิศทางของสายอากาศด้วยสเตปป์มอเตอร์ โดยหลักการทำงานนั้น เป็นการปรับทิศทางหรือตำแหน่งของสายอากาศโดยใช้ไมโครคอนโทรลเลอร์ เบอร์ 8031 CPU เป็นหัวใจในการทำงานทั้งหมด ทำหน้าที่สั่งงานให้สเตปป์มอเตอร์ทำงานตามขั้นตอนของโปรแกรม การควบคุมการทำงานของสเตปป์มอเตอร์ใช้การควบคุมแบบเปิด (open loop) ซึ่งการควบคุมแบบนี้ จะอาศัยความแน่นอนและเที่ยงตรงของการหมุนเป็นสเตป โดยการหมุนแต่ละครั้งจะหมุนไป 1.8 องศา (1.8/step) ฉะนั้นการสั่งงานสเตปป์มอเตอร์เราสามารถนับจำนวนองศาได้จากจำนวนพัลส์ ที่เราป้อนได้อย่างแม่นยำโดยไม่ต้องมีการป้อนกลับ ทั้งนี้ และทั้งนั้น จะต้องคำนึงถึงโหลดด้วย ในการเพิ่มการรับโหลดให้มากขึ้นจะใช้อัตราทดของเกียร์

ระบบของเครื่องปรับทิศทางของสายอากาศด้วยสเตปป์มอเตอร์ แสดงในรูปที่ 3.1



รูปที่ 3.2 บล็อกไดอะแกรมของระบบปรับทิศทางของสายอากาศด้วยสเตปป์มอเตอร์

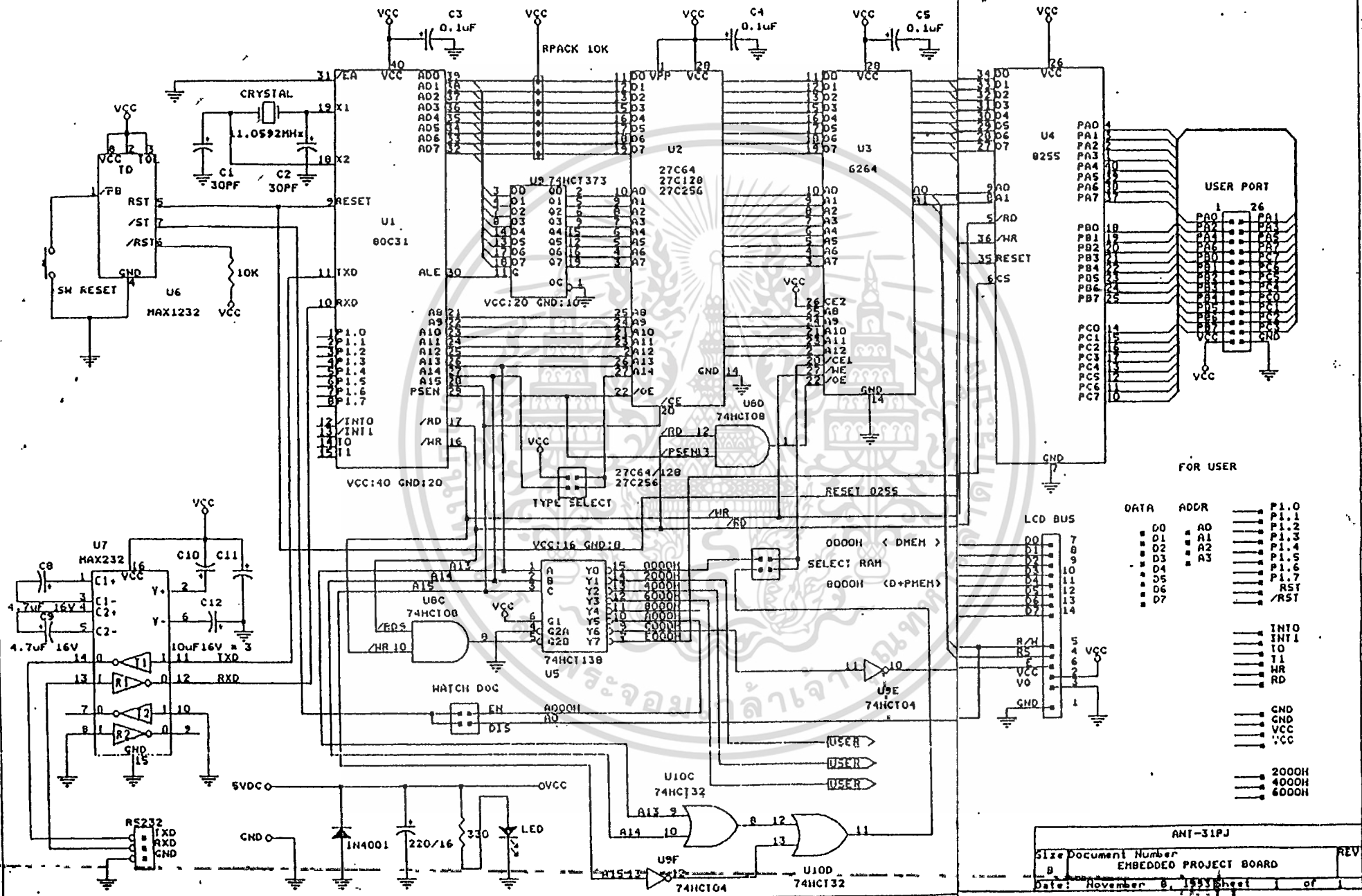
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 3.3 ชุดคอนโทรลเลอร์บอร์ด

คอนโทรลเลอร์บอร์ด ใช้ไมโครคอนโทรลเลอร์เบอร์ 8031 เป็น CPU ทำหน้าที่สั่งงานและประมวลผล โดยใช้หน่วยความจำแบบรอม (ROM) เบอร์ 27256 สำหรับเก็บโปรแกรมมอเตอร์ซึ่งเขียนด้วยภาษาแอสเซมบลี และมีหน่วยความจำแรม (RAM) เบอร์ 6264 เก็บค่าสแตกและบัฟเฟอร์ข้อมูลต่างๆ ส่วน ไอซี (IC) เบอร์ 8255 จะเป็นพอร์ตอินพุต / เอาต์พุตของคอนโทรลเลอร์บอร์ด



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



FOR USER

DATA	ADDR	Pin
00	A0	P1.0
01	A1	P1.1
02	A2	P1.2
03	A3	P1.3
04	A4	P1.4
05	A5	P1.5
06	A6	P1.6
07	A7	P1.7
		RST
		/RST

LCD BUS	Pin
D0	7
D1	8
D2	9
D3	10
D4	11
D5	12
D6	13
D7	14

	INTO
	INTI
	I0
	I1
	HR
	RD

	CND
	CND
	VCC
	VCC
	2000H
	4000H
	6000H

### 3.4 วงจรขับสเตปป์มอเตอร์

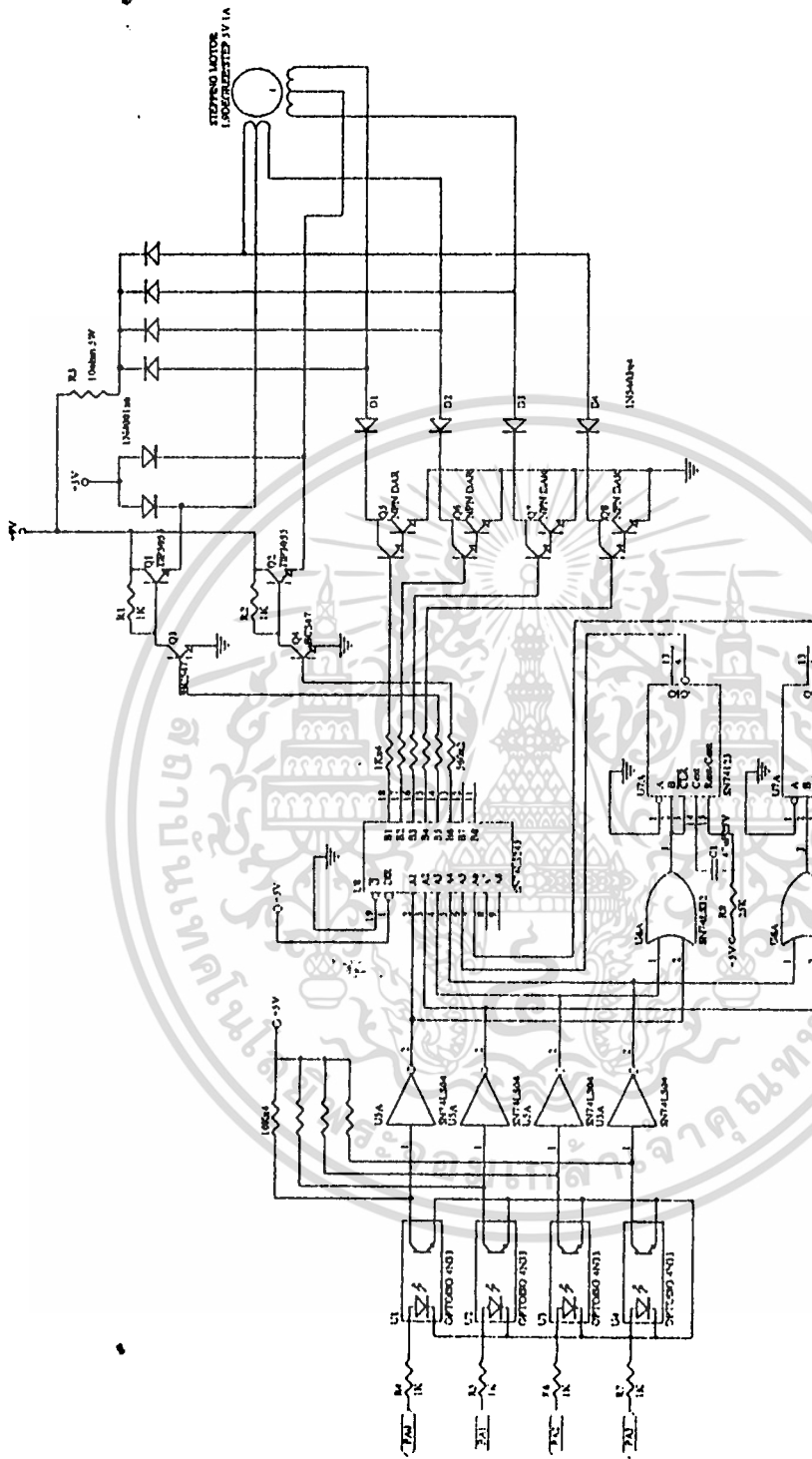
วงจรถับสเตปป์มอเตอร์ ซึ่งแสดงในรูป 3.3 ใช้อุปกรณ์หลักคือออปโตไอโซเรเตอร์ (Opto Isorator) เบอร์ 4N33 ซึ่งมีตัวรับเป็น โฟโตคาร์ลิ่งตัน (photo darlington) เป็นวงจรถับสเตปป์เพื่อแยกส่วนระหว่างวงจรถับสเตปป์ กับวงจรถับสเตปป์ เพื่อป้องกันการคิงกระแส จากวงจรถับสเตปป์ซึ่งจะทำให้วงจรถับสเตปป์เกิดการเสียหายขึ้นได้ R 10K 4 ตัวที่ต่อกับขา 5 ของออปโตไอโซเรเตอร์ จะทำหน้าที่ทำให้แรงดันอินพุทของ IC เบอร์ 74LS04 (not gate) มีค่าเป็น "1" ในกรณีที่ไม่มีพัลส์ป้อนให้ออปโตไอโซเรเตอร์ IC เบอร์ 74LS04 จะเปลี่ยนสถานะจาก "1" เป็น "0" แต่ถ้ามีพัลส์ป้อนเข้ามาก็จะให้แรงดันอินพุทของ IC เบอร์ 74LS04 เป็น "0" และเอาท์พุทจะเป็น "1"

IC เบอร์ 74LS245 เป็น IC ที่ทำหน้าที่เป็นบัฟเฟอร์และสามารถให้กระแสได้สูงสุดถึง 200 มิลลิแอมป์ ซึ่งเพียงพอที่จะขับทรานซิสเตอร์ที่จะต่อคาร์ลิ่งตันอยู่ในแต่ละเฟส โดยมีทรานซิสเตอร์ BC547 ต่อแบบ คาร์ลิ่งตันกับทรานซิสเตอร์ TIP3055 เพื่อที่ขยายกระแสให้สูงมากพอที่จะขับมอเตอร์ในแต่ละเฟสได้ เพราะกระแสที่ใช้ขับในแต่ละเฟสนั้นใช้ถึง 1 แอมป์ ที่แรงดัน 5 โวลต์ เนื่องจากทรานซิสเตอร์ TIP3055 นี้ต้องขับกระแสสูงเช่นนี้ ข้อมเกิดความร้อนขึ้นมากจึงจำเป็นต้องใช้ฮีตซิงค์ (Heat sink) เป็นตัวช่วยระบายความร้อนที่เกิดขึ้น

ในการควบคุมสเตปป์มอเตอร์ให้มีประสิทธิภาพสูงคุณนั้น จะต้องมีการคิงกระแสต่างๆไว้ 1 เฟส เพื่อจะลือคการหมุนของสเตปป์มอเตอร์ในขณะที่ไม่มีการสั่งงานให้สเตปป์มอเตอร์หมุน ซึ่งจะมีผลทำให้การหยุดสเตปป์มอเตอร์ เป็นไปอย่างมีประสิทธิภาพและแม่นยำมาก จึงใช้วงจรโมโนสเตเบิลไวเบเรเตอร์ (Monostable Multivibrator) มาช่วยโดยใช้ IC เบอร์ 74LS32 (or gate) และ IC เบอร์ 74LS123 ซึ่งเป็น IC โมโนสเตเบิลแบบรับการกระตุ้นซ้ำโดยจะมี 2 ตัวอยู่ในตัวเดียว ตัวแรกจะใช้ขับเฟส 1 และเฟส 3 ส่วนตัวที่ 2 จะใช้ขับเฟส 2 และเฟส 4 ถ้ามีการสั่งงานให้สเตปป์มอเตอร์ IC 74LS123 จะทำหน้าที่กำหนดให้ทรานซิสเตอร์ TIP3055 (Q1,Q2) ทำงานสลับกันตามพัลส์ที่ป้อนให้แต่ละเฟสซึ่งจ่ายแรงดันสูงให้สเตปป์มอเตอร์หรือจ่ายกระแสสูงขณะทำงาน (Run Current) ถ้ามีการหยุดป้อนพัลส์เมื่อใด ทรานซิสเตอร์ Q1,Q2 ก็จะหยุดจ่ายกระแสตามค่าเวลาของ โมโนสเตเบิล จากนั้นแรงต่ำ (3V) ก็จะป้อนเข้าที่สเตปป์มอเตอร์ซึ่งมีพัลส์บวกลือคอยู่ 1 เฟส กระแสที่ใช้ลือคนี้เรียกว่าโฮลด์เคอร์เรนท์ (Hold Current) ซึ่งมีค่าเท่ากับ 20% ของกระแสขณะทำงาน

เพื่อป้องกันการเสียหายที่จะเกิดขึ้นกับเพาเวอร์เนื่องจากปัญหาของแรงดันไฟกลับ (Back EMF) ที่เกิดขึ้นเนื่องจากการหยุดป้อนไฟให้ขดลวดสเตปป์มอเตอร์จึงใช้ไดโอด 1N4001 4 ตัว กับรีซิสเตอร์ 10 โอห์ม 5 วัตต์ เป็นตัวลคกระแสที่เกิดจากแรงดันไฟกลับได้ ไดโอด D1-D4 (1N5403) จะป้องกันสไปด์โวลต์เตจ เนื่องจากขดลวดที่เกิดขึ้นเมื่อทรานซิสเตอร์หยุดทำงาน

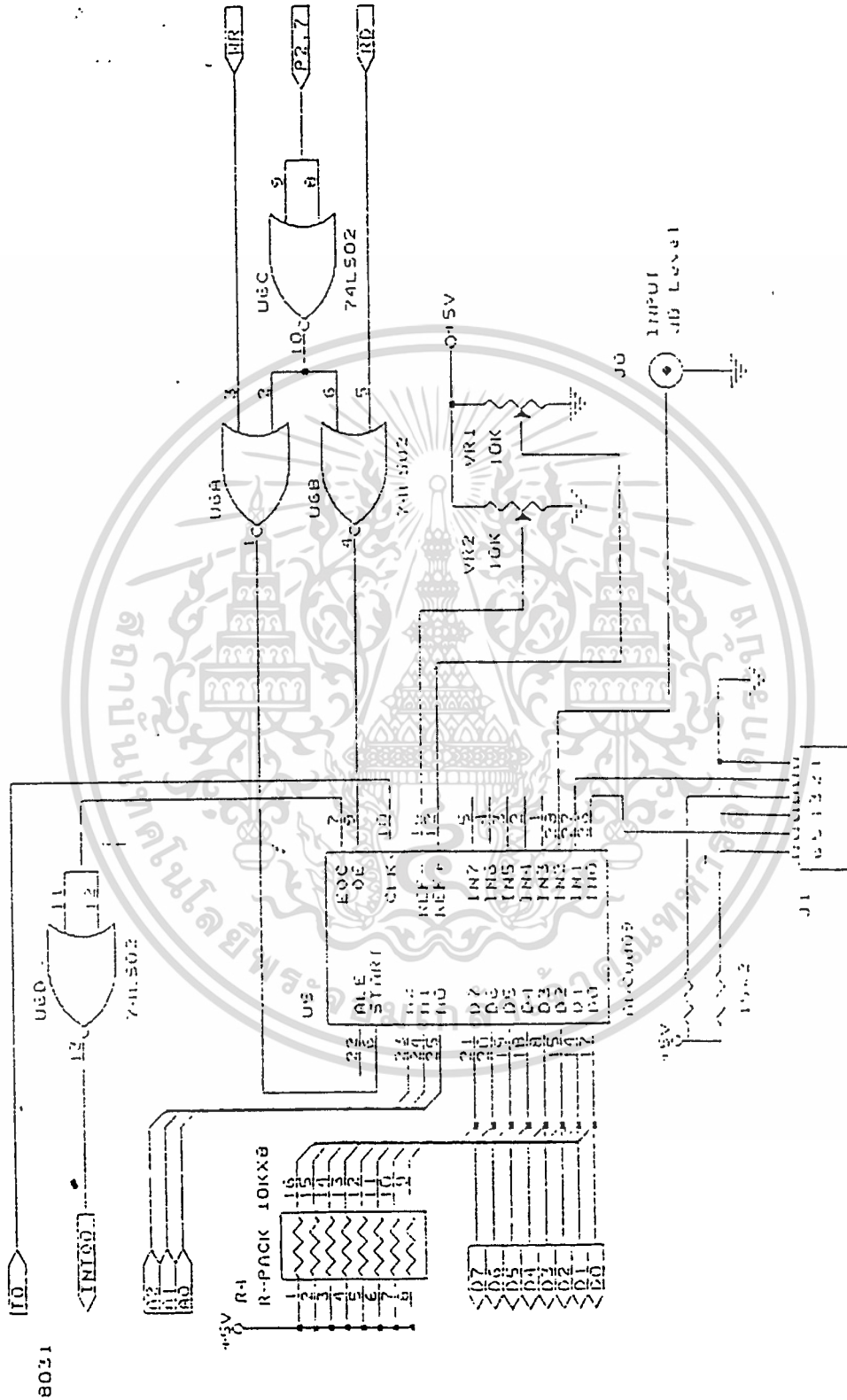
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
 ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมีรูปที่ 3.4 แสดงวงจรขับเคลื่อนสเตปป์มอเตอร์ของเอกสารทุกครั้งที่มีการนำไปใช้

### 3.5 การแปลงสัญญาณอนาลอกเป็นดิจิทัล

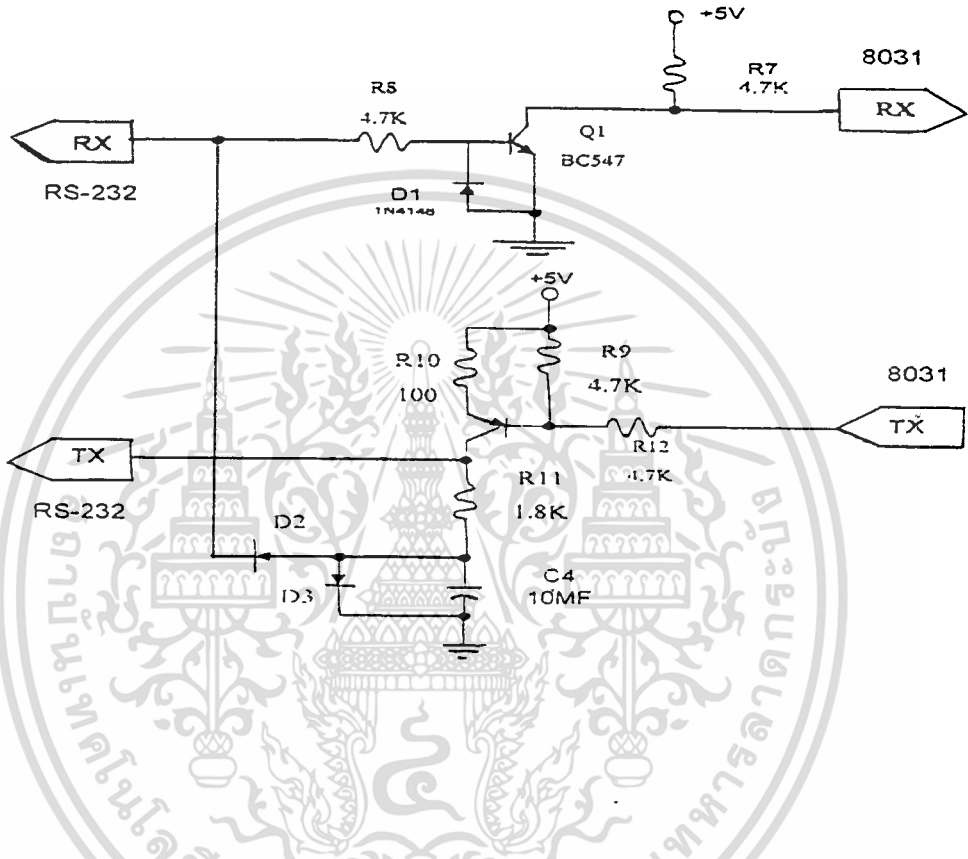
เราได้ทำการต่อวงจร ADC0809 เข้ากับไมโครคอนโทรลเลอร์ดังรูปที่ 3.6



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
 ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องแจ้งเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้  
 รูปที่ 3.5 แสดงการต่อ ADC0809 เข้ากับ ไมโครคอนโทรลเลอร์

### 3.6 พอร์ตอนุกรม RS-232

เราได้ทำการออกแบบโดยใช้ IC MAX232 เป็นตัวเปลี่ยนแรงดันจาก TTL ให้เข้ากับมาตรฐาน RS-232 โดยทำการต่อวงจรตามรูป



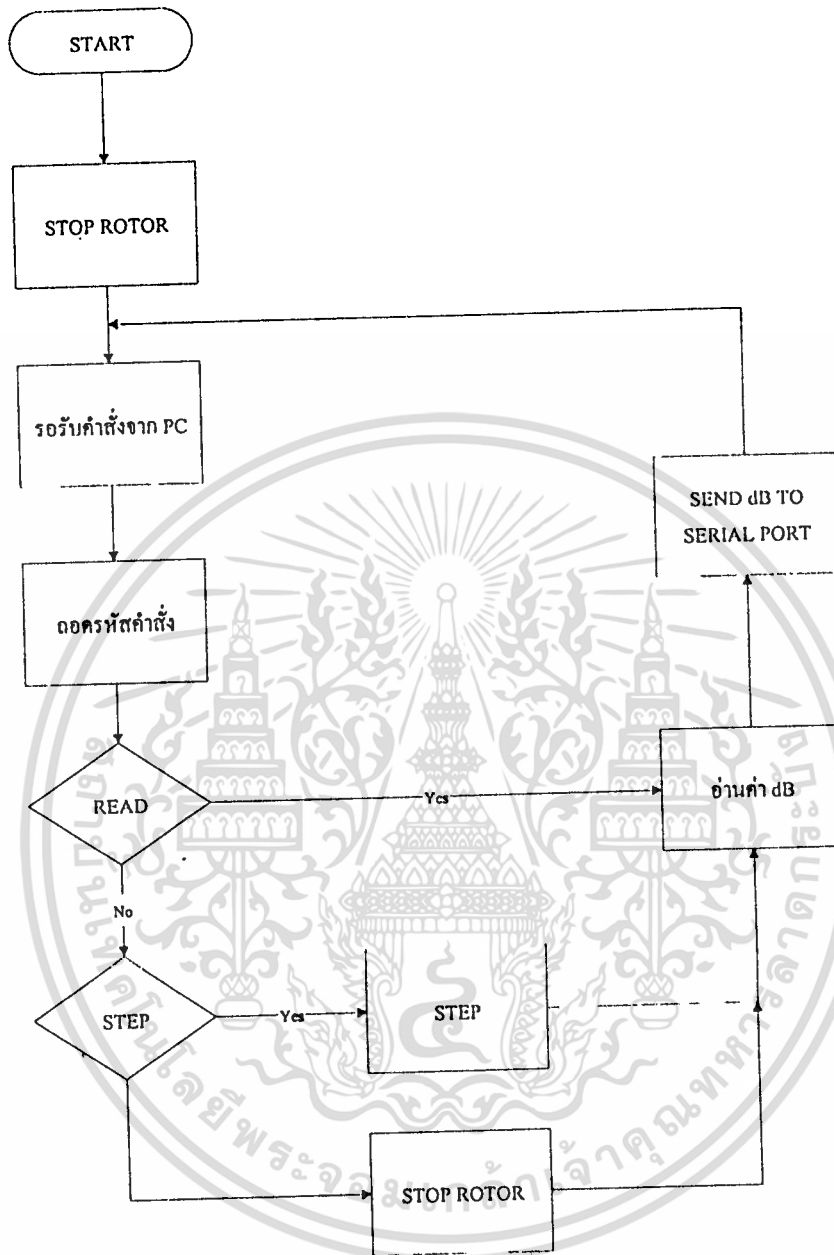
รูปที่ 3.6 แสดงการต่อวงจร Serial Port

### 3.7 หลักการเขียนโปรแกรมการทำงาน

การเขียนโปรแกรมในการพล็อตแพทเทิร์นของสายอากาศ เราได้นำโปรแกรมภาษาวิซวลเบสิก (Visual Basic) มาใช้ในการเขียนโปรแกรมในการพล็อตแพทเทิร์นของสายอากาศ และเขียนโปรแกรมควบคุมการหมุนของสายอากาศโดยใช้ภาษาแอสเซมบลี

การทำงานของระบบนั้นจะมีคอมพิวเตอร์เป็นตัวควบคุมการทำงานทั้งหมด โดยผ่านทางคอนโทรลเลอร์บอร์ด โดยการควบคุมวงจรรับ คอมพิวเตอร์จะทำการกำหนดคสเตรปที่จะทำการหมุนและกำหนดมุมทั้งหมดที่ต้องการแล้ว คอมพิวเตอร์จะทำการเก็บค่าที่อ่านออกมาจากวงจรแปลงสัญญาณอนาลอกเป็นดิจิตอล และนำมาแสดงผลเป็นกราฟแบบโพลาไรและกราฟแบบสี่เหลี่ยมบนจอคอมพิวเตอร์หน้าไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การทำงานของระบบควบคุมแสดงดังรูปที่ 3.8



รูปที่ 3.7 แสดงโฟลว์ชาร์ตของหน่วยควบคุมการทำงาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## 3.7.1 โปรแกรมการทำงานของคอนโทรลเลอร์บอร์ด

8051 Cross-Assembler (1.3) (C) 1987, 1989 Binary Technology

recrder.asm

```

0000                                ORG 0000H
0000 02006E                          LJMP SETUP          ;jump to setting serial & clock

000B                                ORG 000BH
000B 020050                          LJMP CLOCK         ;jump to generate clock for atod

0050                                ORG 0050H
0050 C0E0          CLOCK:  PUSH ACC
0052 C0D0          PUSH PSW
0054 C3           CLR C
0055 EC           MOV A,R4
0056 23           RL A
0057 FC           MOV R4,A
0058 33           RLC A
0059 92B4        MOV P3.4,C
005B D0D0          GO:   POP PSW
005D D0E0          POP ACC
005F 32           RETI          ;back to last interrupt

0060 C298          NODATA: CLR RI
0062 C299          CLR TI
0064 3098FD       JNB RI,$
0067 AA99          MOV R2,$BUF          ;keep data from pc

0069 120096       LCALL CHKEY        ;jump to check command
006C 80F2          SJMP NODATA

006E AF88          SETUP:  MOV R7,$8H          ;initial start motor
0070 90E003       MOV DPTR,$0E003H    ;stop motor
0073 7480          MOV A,$80H          ;set 8255
0075 F0           MOVX @DPTR,A
0076 90E000       MOV DPTR,$0E000H
0079 EF           MOV A,R7
007A F0           MOVX @DPTR,A

007B 7C55          MOV R4,$55H        ;initial clock value
007D 758700       MOV PCON,$00H     ;serial model
0080 759850       MOV SCON,$50H     ;set serial port model & enable receiver
0083 758922       MOV TMOD,$22H    ;timer mode
0086 758DE8       MOV TH1,$0e8H    ;baudrate 1200
0089 D28E          SETB TR1
008B D2AF          SETB IE.7        ;enable interrupt
008D D2A9          SETB IE.1        ;enable t0 overflow

interrupt
008F 758CFE       MOV TH0,$0FEH
0092 D28C          SETB TR0          ;start t0 generate clock

for atod
0094 80CA          SJMP NODATA

0096 EA           CHKEY:  MOV A,R2          ;check command
0097 B4530C       CJNE A,$'S',FORWARD ;stop or other
009A 90E003       MOV DPTR,$0E003H
009D 7480          MOV A,$80H          ;set 8255
009F F0           MOVX @DPTR,A
00A0 90E000       MOV DPTR,$0E000H
00A3 EF           MOV A,R7
00A4 F0           MOVX @DPTR,A
00A5 22           RET

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

8051 Cross-Assembler (1.3) (C) 1987, 1989 Binary Technology  
recrder.asm

```

00A6 B4460E FORWARD:CJNE A,#'F',BACK           ;step forward or other
00A9 90E003          MOV DPTR,#0E003H
00AC 7480            MOV A,#80H
00AE F0              MOVX @DPTR,A

00AF 90E000          MOV DPTR,#0E000h
00B2 EF              MOV A,R7
00B3 03              RR A                      ;rotate value
00B4 F0              MOVX @DPTR,A          ;step motor
00B5 FF              MOV R7,A              ;keep value step to r7

00B6 22              RET

00B7 B4420E          BACK: CJNE A,#'B',RETURN       ;step back or orther
00BA 90E003          MOV DPTR,#0E003H
00BD 7480            MOV A,#80H
00BF F0              MOVX @DPTR,A

00C0 90E000          MOV DPTR,#0E000h
00C3 EF              MOV A,R7
00C4 23              RL A
00C5 F0              MOVX @DPTR,A
00C6 FF              MOV R7,A

00C7 22              RET

00C8 B45211          RETURN: CJNE A,#'R',ATOD           ;return or other
00CB 90E003          MOV DPTR,#0E003H
00CE 7480            MOV A,#80H
00D0 F0              MOVX @DPTR,A
00D1 90E000          MOV DPTR,#0E000h
00D4 EF              MOV A,R7
00D5 23              AGAIN: RL A              ;rotate value back
00D6 F0              MOVX @DPTR,A          ;step back
00D7 7B9A            MOV R3,#666          ;back 666 times
00D9 DBFA            DJNZ R3,AGAIN        ;return to start point

00DB 22              RET

00DC B45019          ATOD: CJNE A,#'P',EXIT           ;plot or exit
00DF 906002          MOV DPTR,#6002H     ;atod address
00E2 7D03            MOV R5,#3           ;3 times to read data
00E4 F0              RDATA: MOVX @DPTR,A ;start atod
00E5 7E9B            MOV R6,#9BH         ;set conversion time delay for atod
00E7 DEFE            DJNZ R6,$           ;wait
00E9 E0              MOVX A,@DPTR        ;read data from atod
00EA DDF8            DJNZ R5,RDATA       ;read again

00EC C298            SDATA: CLR RI
00EE C299            CLR TI
00F0 F599            MOV SBUF,A          ;send data to pc
00F2 3099FD          JNB TI,$
00F5 C299            CLR TI

00F7 22              RET

00F8 22              EXIT: RET

0000-                END

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

8051 Cross-Assembler (1.3) (C) 1987, 1989 Binary Technology  
recrder.asm

gain	= 00D5	92	95	
atod	= 00DC	99	85	
back	= 00B7	72	59	
chke	= 0096	48	25	
clock	= 0050	8	5	
exit	= 00F8	115	99	
forward	= 00A6	59	49	
go	= 005B	16		
nodata	= 0060	20	26	46
rdata	= 00E4	102	106	
return	= 00C8	85	72	
sdata	= 00EC	107		
setup	= 006E	28	2	
stop	= 0097	49		



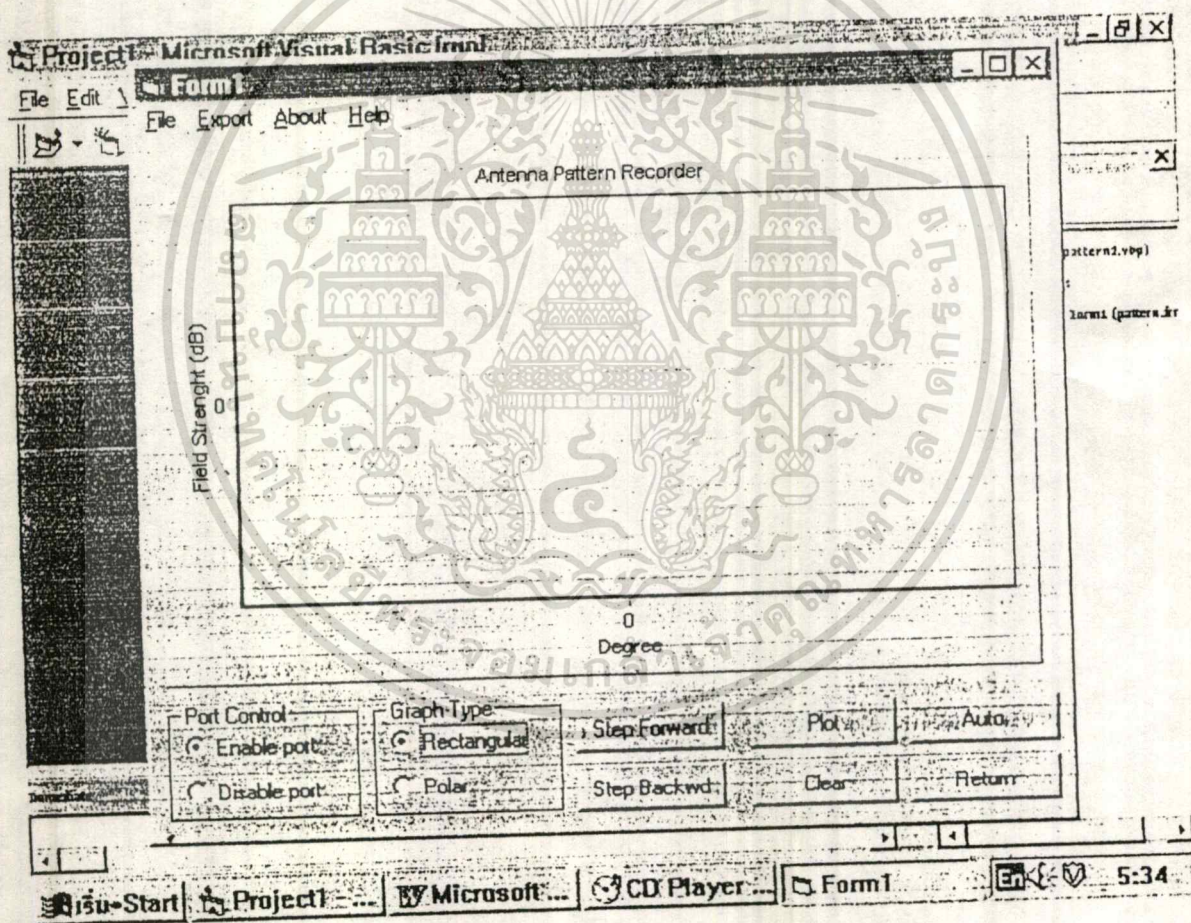
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เราได้นำ เอาโปรแกรมภาษาวิชวลเบสิก (Visual Basic) มาใช้ เนื่องจาก วิชวลเบสิกสามารถ รัน (Run) บนวินโดว์ ซึ่งสะดวกในการใช้และเขียนด้วยคำสั่งง่ายๆ และได้กราฟออกมาอย่างสวยงามบนหน้าจอ

### 3.7.2 อธิบายการทำงานของโปรแกรมที่ใช้พล็อตแพทเทิร์นของสายอากาศ

การสื่อสารกับคอนโทรลเลอร์บอร์ดในวิชวลเบสิกใช้ MSComm control ซึ่งใช้กับการส่งข้อมูลแบบพอร์ตอนุกรม (Serial Port) โดยใช้เวอร์ชัน VB4.0

รูปแบบฟอร์มของโปรแกรมการทำงานของวิชวลเบสิก มีดังนี้



รูปที่ 3.8 แสดงแบบฟอร์มการทำงานของโปรแกรม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โปรแกรมนี้ถูกสร้างขึ้นโดยใช้ command buttons , text box สำหรับแสดงข้อมูลต่อกัน, port control, MSComm control เป็นพอร์ตการสื่อสารแบบอนุกรม , tee chart เป็นตัวแสดงผลออกมาเป็นกราฟรูปโพลาร์และกราฟรูปสี่เหลี่ยม

การกำหนด object, property และการเซต ต่างๆ ดังต่อไปนี้

OBJECT	PROPERTY	SETTING
CommonDialog	Name	CommonDialog1
Frame	Name	Frame1
	Caption	Port Enable
Frame	Name	Frame2
	Caption	GraphType
OptionButton	Name	Option1
	Caption	Rectangular
OptionButton	Name	Option2
	Caption	Polar
OptionButton	Name	Option3
	Caption	Enable Port
OptionButton	Name	Option4
	Caption	Disable Port
CommandButton	Name	Command1
	Caption	Step Forward
CommandButton	Name	Command2
	Caption	Step Backwd
CommandButton	Name	Command3
	Caption	Plot
CommandButton	Name	Command4
	Caption	Clear
CommandButton	Name	Command5
	Caption	Auto
CommandButton	Name	Command6
	Caption	Return

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
รูปที่ 3.9 แสดงตารางการกำหนด object, property , setting  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามเผยแพร่ลงเนื้อหาและต้องอ้างอิงถึงชื่อเอกสารทุกครั้งที่มีการนำไปใช้

โครงสร้างเมนูแสดงดังต่อไปนี้

&File	mnuFileItem
—&Open	mnuOpenItem
—&Save	mnuSaveItem
—&Print	mnuPrintItem
—&Close	mnuCloseItem
—E&xit	mnuExitItem
&Export	mnuExportItem
&About	mnuAboutItem
—about T&hesis	mnuThesisItem
—about &Teechart	mnuaboutTchart

รูปที่ 3.10 เป็นการกำหนดโครงสร้างเมนู

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตัวอย่างโค้ดของอ็อบเจกต์ (Object's Code) บางส่วนที่ใช้ในการพล็อตแพทเทิร์นของสายอากาศโดยใช้ภาษาวิซวลเบสิก มีดังต่อไปนี้

```

Option1 Click
Private Sub mnuThesisItem_Click()
    MsgBox ("Thank you for our teachers and our friends"), _
        vbOKOnly, "This Thesis is manage by Pog & Kay."
End Sub

Private Sub Option1_Click()
    TChart1.ChangeSeriesType 0, scFastLine
    TChart1.Axis.Bottom.Visible = True
    TChart1.Axis.Left.Visible = True
End Sub

Private Sub Option2_Click()
    TChart1.ChangeSeriesType 0, scPolar
    TChart1.Series(0).asPolar.Circled = True
    TChart1.Series(0).asPolar.Pointer.Visible = False
    TChart1.Axis.Bottom.Automatic = False
    TChart1.Axis.Bottom.SetMinMax 0, 360
    TChart1.Axis.Bottom.Visible = False
    TChart1.Axis.Left.Visible = False
End Sub

```

รูปที่ 3.11 a) แสดงโค้ดของออปชั่น 1 และออปชั่น 2

```

Option3 Click
Private Sub Option3_Click()
    ' Enable port
    MSComm1.PortOpen = True
    Command1.Enabled = True
    Command2.Enabled = True
    Command3.Enabled = True
    Command5.Enabled = True
    Command6.Enabled = True
End Sub

Private Sub Option4_Click()
    ' Disable port
    MSComm1.PortOpen = False
    Command1.Enabled = False
    Command2.Enabled = False
    Command3.Enabled = False
    Command5.Enabled = False
    Command6.Enabled = False
End Sub

Private Sub mnuCloseItem_Click()

```

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์การเชิงงานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ต่อบ) แสดงโค้ดของออปชั่น 3 และออปชั่น 4 เอกสารทุกครั้งที่มีการนำไปใช้

```

mnuThesisItem Click
Private Sub mnuCloseItem_Click()
    TChart1.Series(0).Clear 'clear tchart
    mnuCloseItem.Enabled = False 'dim Close command
    mnuOpenItem.Enabled = True 'enable Open command
    'disable text box
End Sub

Private Sub mnuExitItem_Click()
    End 'quit program
End Sub

Private Sub TChart1_OnClick()
    Dim t As Integer
    For t = 0 To TChart1.SeriesCount - 1
        If TChart1.Series(t).GetMousePoint <> -1 Then
            MsgBox ("Clicked Series: "
                + TChart1.Series(t).Name + "at point: "
                + Str$(TChart1.Series(t).GetMousePoint))
        End If
    Next t
End Sub

```

```

mnuThesisItem Click
Private Sub mnuOpenItem_Click()
    CommonDialog1.Filter = "Teefiles (*.TEE)|*.TEE"
    CommonDialog1.ShowOpen
    If CommonDialog1.filename <> "" Then
        Form1.MousePointer = 11 'display hour glass
        TChart1.Import.LoadFromFile (CommonDialog1.filename)
        mnuCloseItem.Enabled = True
        Form1.MousePointer = 0 'reset mouse
    Else
        End If
End Sub

Private Sub mnuPrintItem_Click()
    TChart1.Printer.ShowPreview
End Sub

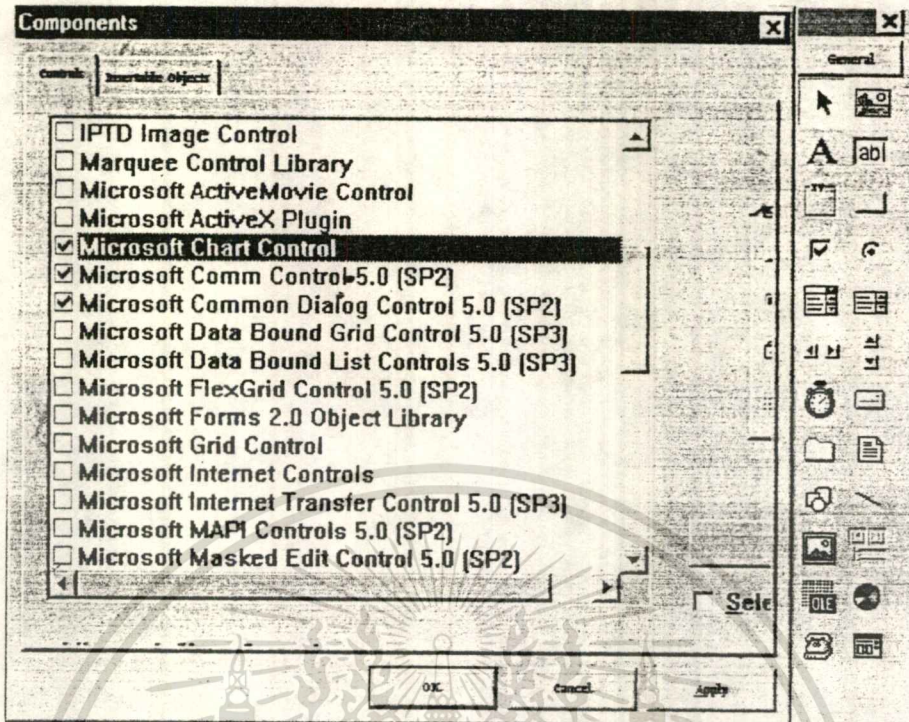
Private Sub mnuSaveItem_Click()
    CommonDialog1.ShowSave
    TChart1.Export.SaveToFile (CommonDialog1.filename)
End Sub

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่นอนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องขออนุญาตจากเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

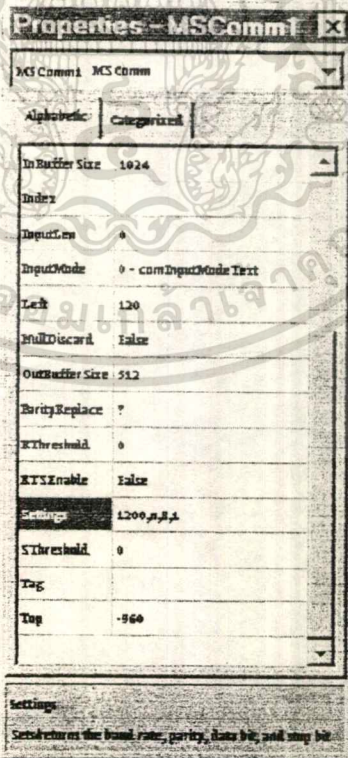
c) แสดงโค้ดของเมนูต่างๆ

ส่วนประกอบที่ใช้เพิ่มเติมใน โปรแกรมนี้



รูปที่ 3.12 แสดงส่วนประกอบเพิ่มเติมของโปรแกรม

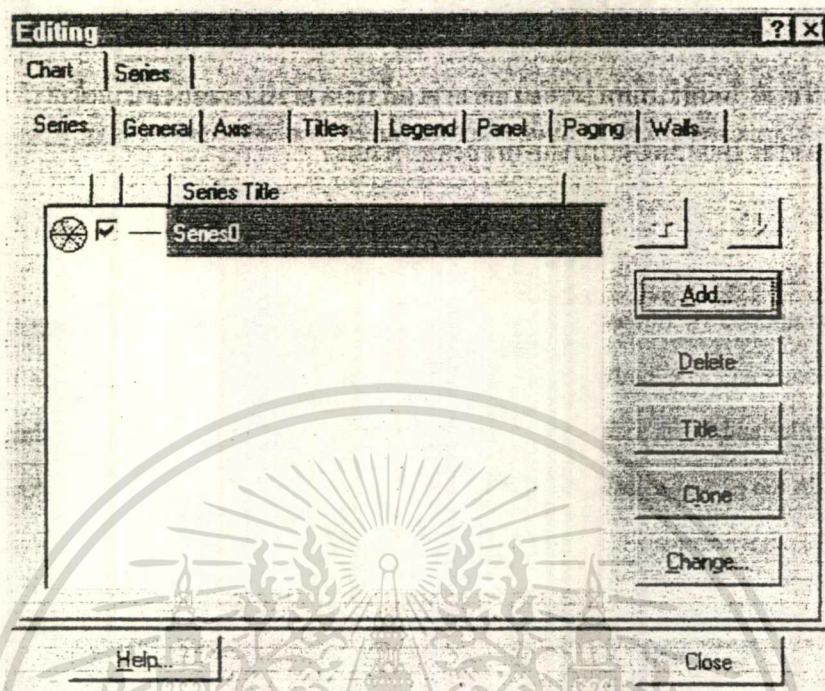
การกำหนดคุณสมบัติให้อุปเจ็ท MSComm1 MSComm



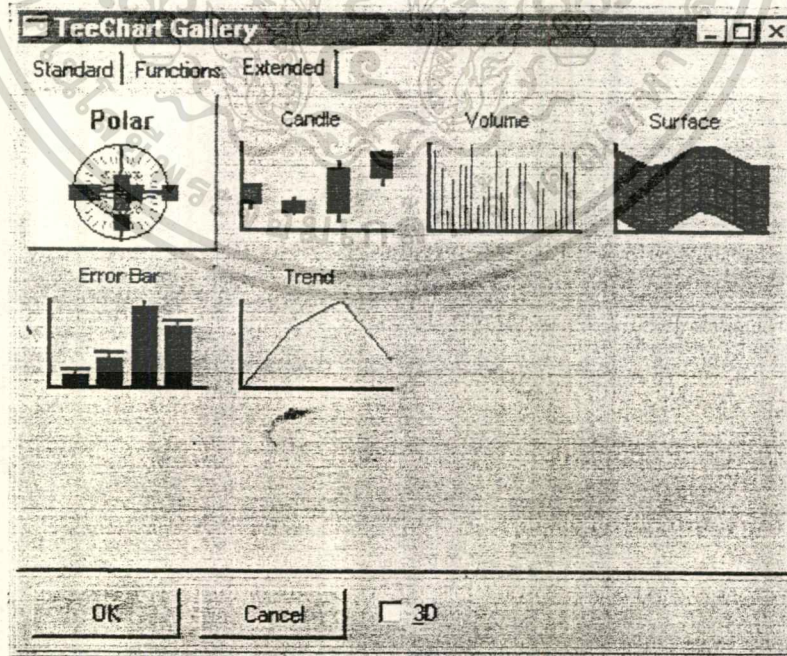
รูปที่ 3.13 แสดงคุณสมบัติของ MSComm1

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่แนะนำให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การเลือกชนิดของกราฟใน Tee chart มีดังรูปที่ 3.14



รูปที่ 3.14 แสดงการเลือก chart ต่างๆ ใน Tee chart



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
รูปที่ 3.15 แสดงกราฟต่างๆ ใน Tee chart  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## บทที่ 4

## การทดลองและผลการทดลอง

## วัตถุประสงค์

1. เพื่อศึกษาการขับสัญญาณการกระตุ้นจากไมโครคอนโทรลเลอร์ในแบบต่างๆ
2. เพื่อเปรียบเทียบการกระตุ้นสเตปป์ิงมอเตอร์ในแต่ละระบบ
3. เพื่อนำผลที่ได้ไปประยุกต์ใช้สำหรับการควบคุมตำแหน่งของการหมุนของสายอากาศ
4. เพื่อศึกษาการเขียนโปรแกรมภาษาวิซวลเบสิก (Visual Basic) และนำมาเขียนโปรแกรมในการพล็อตแพทเทิร์นของสายอากาศในรูปของโพลาร์พล็อต และแบบสี่เหลี่ยม แสดงบนจอคอมพิวเตอร์

## 4.1 สเตปป์ิงมอเตอร์

สเตปป์ิงมอเตอร์เป็นมอเตอร์ที่ทำงานโดยเปลี่ยนพลังงานไฟฟ้าแบบสวิตชิง ที่ป้อนให้กับขดลวดสเตเตอร์ ให้เป็นการเคลื่อนที่เชิงมุมบนแกนโรเตอร์ ในลักษณะการหมุนเวียนทีละสเตปตามสัญญาณอินพุทของเฟสต่างๆ การเคลื่อนที่และหยุดแต่ละสเตป จะมีความแม่นยำทางตำแหน่งสูงมากแต่มีข้อจำกัดอยู่เหมือนกันคือขนาดของแรงบิดน้อยจะทำให้ไม่สามารถใช้งานหนักมากได้ แบบของสเตปป์ิงมอเตอร์ที่พบบ่อยๆก็คือแบบไฮบริดจ์ โรเตอร์ทำด้วยแม่เหล็กถาวรและเป็นแบบชนิดขดลวด 4 เฟส ไฟที่จ่ายให้ขดลวดสเตเตอร์ในแต่ละเฟสจะมีผลทำให้เกิดสนามแม่เหล็กในทิศทางนั้นๆ เกิดแรงผลักรับแม่เหล็กถาวร ที่ตัวโรเตอร์ในทิศทางหนึ่ง การสั่งให้โรเตอร์หมุนในทิศทางใดก็สามารถทำได้โดยจ่ายกระแสไฟแก่ขดลวดสเตเตอร์ในเฟสต่างๆ อย่างถูกต้องต่อไป

## 4.1.1 การทดสอบสเตปป์ิงมอเตอร์

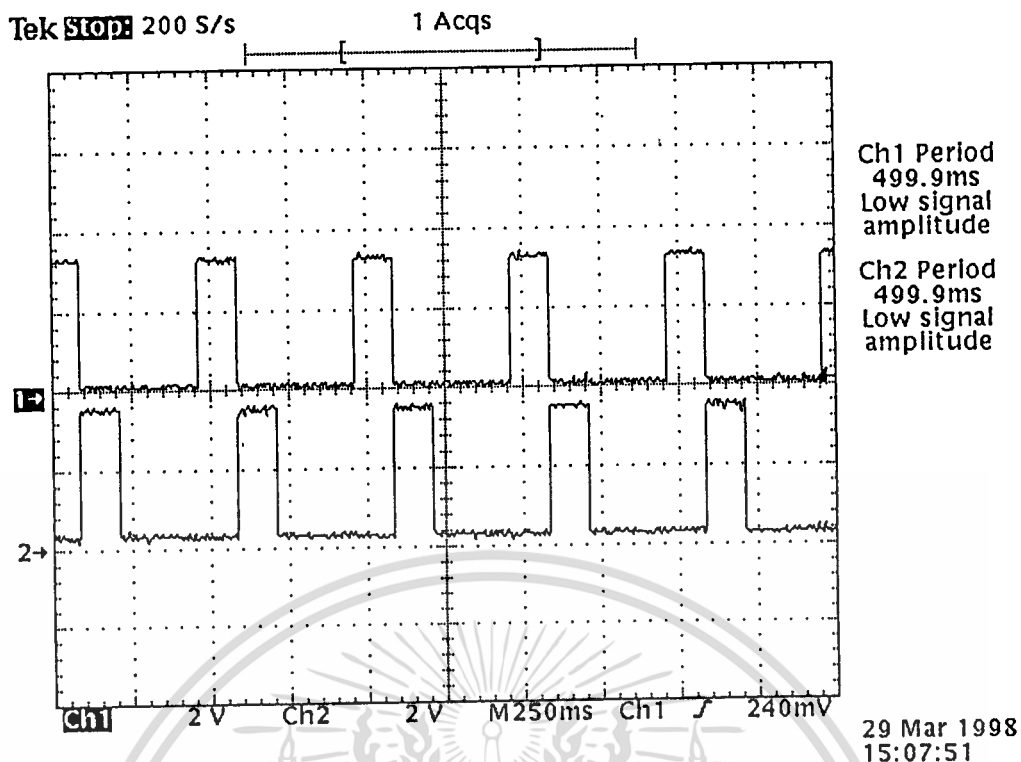
จากสเตปป์ิงมอเตอร์มีขนาด 4 เฟสนี้เราจะบังคับให้เกิดสเตปการหมุนได้เป็น 3 ลักษณะขึ้นอยู่กับการป้อนพัลส์ดังนี้

1. One-excitation หรือ half drive เป็นการจ่ายกระแสให้กับสเตปป์ิงมอเตอร์ครั้งละ 1 เฟส คือ  $\Phi_1, \Phi_2, \Phi_3, \Phi_4$  เรียงตามลำดับ หมุนเวียนกันไปแบบนี้ แรงบิดจะน้อยดังตารางในรูปที่ 4.1

No.	$\Phi_1$	$\Phi_2$	$\Phi_3$	$\Phi_4$
1	1	0	0	0
2	0	1	0	0
3	0	0	1	0
4	0	0	0	1

รูปที่ 4.1 One-excitation

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



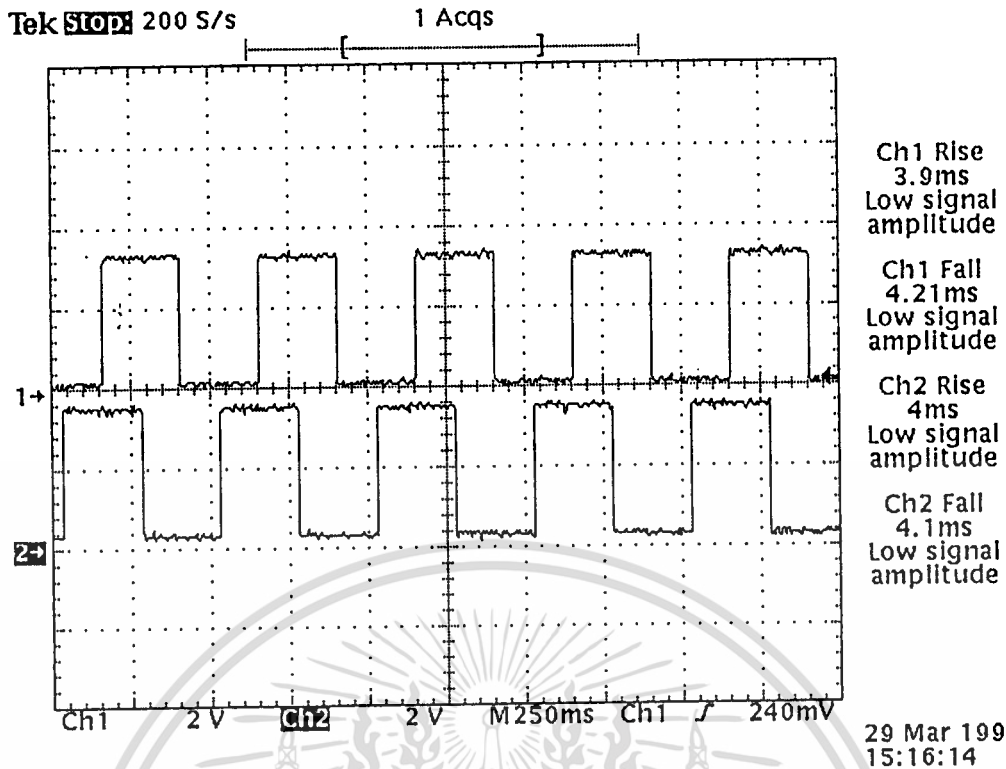
รูปที่ 4.2 แสดงสัญญาณที่วัดได้จากการกระตุ้นแบบ one-excitation

2. two-excitation หรือ full step เป็นการจ่ายกระแสให้กับสเตเตอร์ครั้งละ 2 เฟส พร้อมกันได้แก่  $\Phi_1$   $\Phi_2$ ,  $\Phi_2\Phi_3$ ,  $\Phi_3\Phi_4$ ,  $\Phi_4\Phi_1$  หมุนเวียนกันไปแบบนี้ แรงบิดที่ได้จะมากกว่าแบบแรก

No.	$\Phi_1$	$\Phi_2$	$\Phi_3$	$\Phi_4$
1	1	1	0	0
2	0	1	1	0
3	0	0	1	1
4	1	0	0	1

รูปที่ 4.3 แสดง two-excitation

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.4 แสดง full-step

3. one-two-excitation หรือ half step เป็นการจ่ายกระแสให้กับขดลวดสเตเตอร์ 1 เฟส สลับกันไปแบบนี้ ทำให้จำนวนของสเตปเพิ่มขึ้นเป็น 2 เท่า ของ 2 แบบแรกแต่แรงบิดจะน้อยกว่าแบบที่ 2 เล็กน้อย

No	$\Phi_1$	$\Phi_2$	$\Phi_3$	$\Phi_4$
1	1	0	0	0
2	1	1	0	0
3	0	1	0	0
4	0	1	1	0
5	0	0	1	0
6	0	0	1	1
7	0	0	0	1
8	1	0	0	1

รูปที่ 4.5 one-two-excitation

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จะเห็นว่า ในการควบคุมสเปคการทำงานของสเทปปีงมอเตอร์ทั้ง 3 รูปแบบ พบว่าการทดลองการขับสเทปปีงมอเตอร์โดยใช้การกระตุ้นแบบ 1 เฟส และ 2 เฟส (one-excitation and two-excitation) พบว่าที่ความเร็วต่ำจะเกิดการสั่นสะเทือนและเกิดเสียงดัง แต่จะให้ความหนาแน่นของสเทปได้ดี

การทดลองการขับสเทปปีงมอเตอร์โดยใช้การกระตุ้นแบบ 1-2 เฟส (one-two excitation) สามารถที่จะแก้ปัญหาการสั่นสะเทือนได้ แต่มีข้อเสียคือจะทำให้ความหนาแน่นของการสเทปได้ไม่ดี แต่มีความละเอียดของการสเทปได้มากกว่า การกระตุ้นแบบ 1 เฟส และ 2 เฟส ถึง 2 เท่า

จากการกระตุ้นทั้ง 3 แบบที่ทำการทดลองจะพบว่าที่ความเร็วเพิ่มขึ้นจะทำให้แรงบิดน้อยลง



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## 4.2 การทดสอบค่าความเป็น Linear ของ ADC0809

ระดับโวลต์แดง	ค่าที่ได้จาก ADC 0809 (V)
0.0	00000000
0.2	00000101
0.4	00010100
0.6	00011100
0.8	00100100
1.0	00100100
1.2	00111100
1.4	01010101
1.6	01010100
1.8	01011110
2.0	01100100
2.2	01101111
2.4	01111100
2.6	10000100
2.8	10001100
3.0	10011100
3.2	10100110
3.4	10101101
3.6	10110111
3.8	11000100
4.0	11001110
4.2	11010110
4.4	11100100
4.6	11101110
4.8	11110111
5.0	11111111

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์ 4.6 แสดงตารางการวัดค่าความเป็น Linear ของ ADC 0809 ใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

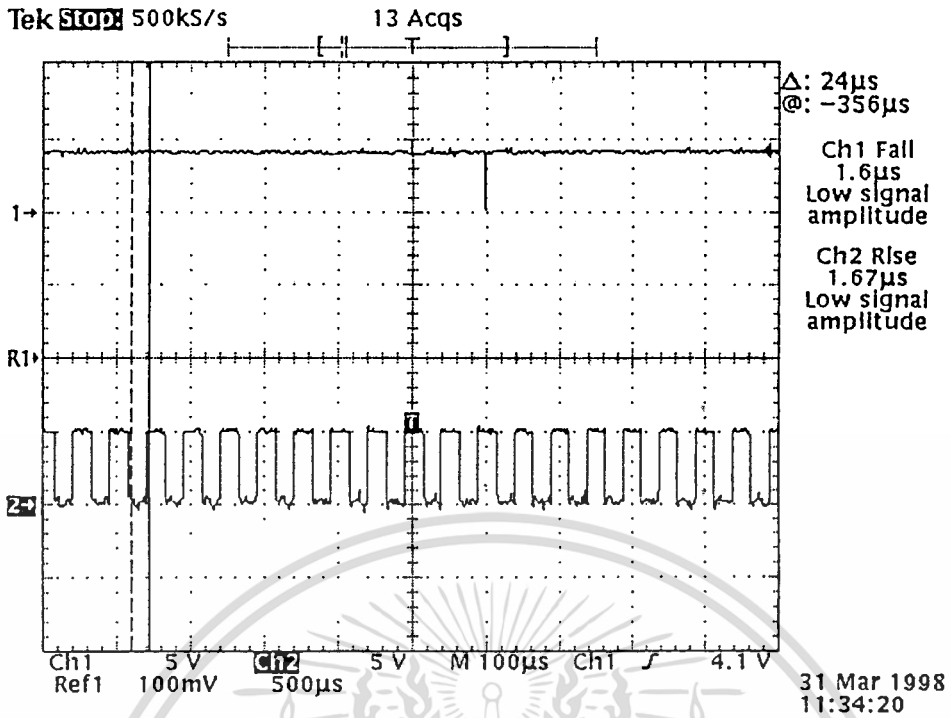
ได้มีการเขียนโปรแกรมทดลองใช้ ADC0809 ดังนี้

```

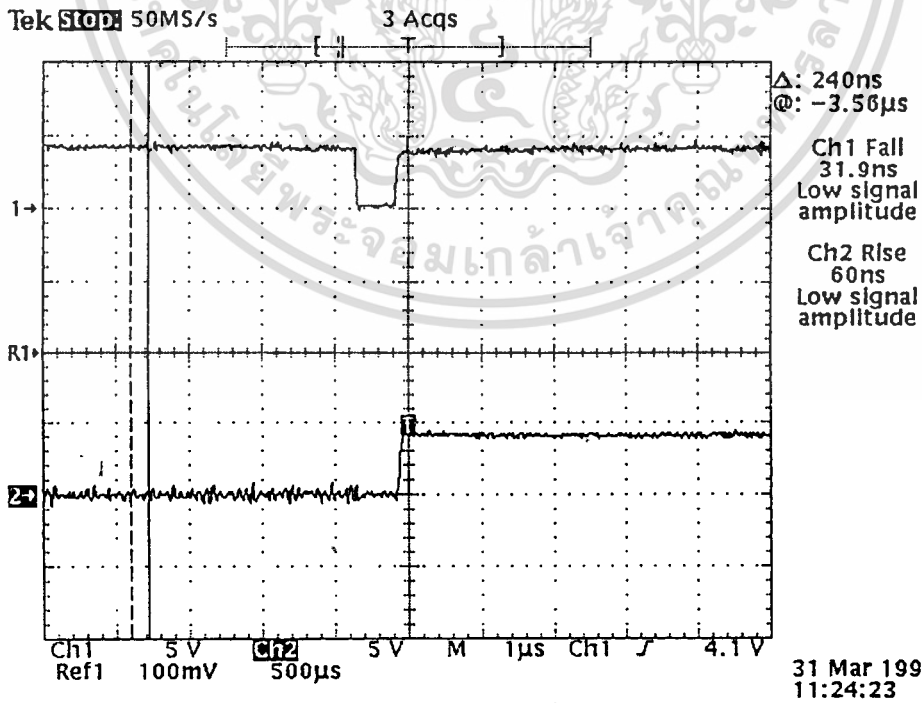
                ORG    0000H
                LJMP   SETUP        ; JUMP TO SETTING CLOCK
                ORG    000BH
                LJMP   SETUP        ; JUMP TO GENERATE CLOCK FOR ATOD
CLOCK:         ORG    0050H
                PUSH  ACC
                PUSH  PSW
                CLR   C
                MOV   A,R4
                RL    A
                MOV   R4,A
                RLC   A
                MOV   P3.4,C
GO:            POP   PSW
                POP   ACC
                RETI   ; BACK TO LAST INTERRUPT
NODATA:       MOV   DPTR,#6002H ; ATOD ADDRESS
                MOV   R5,#3      ; 3 TIMES TO READ DATA
RDATA:        MOVX  @DPTR,A      ; START ATOD
                MOV   R6,#9BH    ; SET CONVERSION TIME DELAY FOR ATOD
                DJNZ  R6,$       ; WAIT
                MOVX  A,@DPTR    ; READ DATA FROM ATOD
                MOV   P1,A       ; SEND DATA TO PORT 1
SETUP:        MOV   R4,#55H     ; INITIAL CLOCK VALUE
                MOV   PCON,#00H  ; SERIAL MODE 1
                MOV   TMOD,#22H  ; TIMER MODE
                SETB  IE.7       ; ENABLE INTERRUPT
                SETB  IE.1       ; ENABLE TO OVERFLOW INTERRUPT
                MOV   TH0,#0FEH
                SETB  TR0        ; START TO GENERATE CLOCK FOR ATOD
                SETJMP NODATA
                END

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.7 แสดง ชั้นแนล 1 เป็นสัญญาณเริ่มทำงานของ ADC0809  
ชั้นแนล 2 เป็นสัญญาณ Clock ที่ป้อนให้ ADC0809



รูปที่ 4.8 ชั้นแนล 1 เป็นสัญญาณสิ้นสุดของ ADC0809  
ชั้นแนล 2 เป็นสัญญาณที่วัดจากพอร์ต 1

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์หรือการสงวนลิขสิทธิ์อื่นใดไว้ก่อน กรุณาแจ้งให้ทราบก่อนนำเอกสารนี้ไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตีแบบลงนิตยสาร และต้องยกเงินของเอกสารทุกครั้งที่มีการนำไปใช้

### 4.3 การทดลองรับข้อมูลของพอร์ตอนุกรม (Serial port)

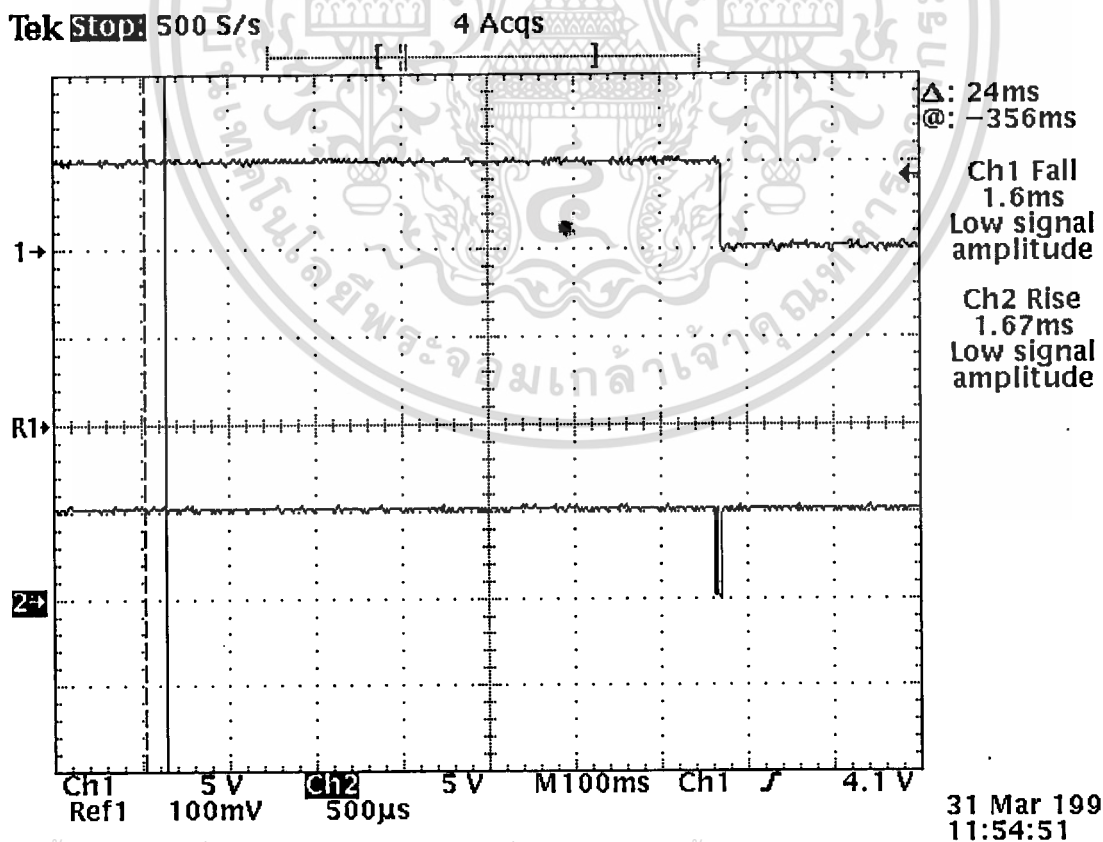
โปรแกรมรับข้อมูลในโหมด 1 ( 1 start bit , 1 stop bit, 8 bit data ) อัตราส่งข้อมูล 1200 บิตต่อวินาที และส่งข้อมูลนั้นออกจากพอร์ต 1

```

ORG      8000H
MOV      PCON,#00H      ; SMOD = 0
MOV      SCON,#50H      ; SERIAL MODE RECIEVE ENABLE
MOV      TMOD,#20H      ; TIMER 1 MODE 2
MOV      TH1,#0E8H      ; 1200 BAUD
SETB     TR1            ; START TIMER
WAIT:    JNB     RI,WAIT  ; RECIEVE DATA
CLR      RI
MOV      A,SBUF          ; DISPLAY AT P1
MOV      P1,A           ; DISPLAY AT P1
SJMB     WAIT
END

```

จากการทดลอง วัดสัญญาณที่ขา Rx ของ IC MAX232 จะได้สัญญาณดังนี้



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
 ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามใช้ซ้ำแปลงที่เอกสารที่วัดได้จากรูปที่ 4.9 ชั้นแนล 1 แสดงสัญญาณที่วัดได้จากขา P1.0

#### 4.4 ขั้นตอนการทดสอบการวัดแพทเทิร์นของสายอากาศ

เราได้ทำการทดสอบการวัดแพทเทิร์นของสายอากาศโดยทำการวัดที่ความถี่ 300 MHz โดยใช้สายอากาศแบบโมโนโคโพลความถี่ 300 MHz เป็นตัวทดสอบ

■ ทำการต่อสายอากาศส่งเข้ากับเครื่องส่ง โดยใช้ เครื่องเน็ตเวิร์คแอนาไลเซอร์ (Network Analyzer) เป็นตัวส่งคลื่นความถี่ 300 MHz

■ ทำการต่อสายอากาศรับเข้ากับเครื่องรับ (Measuring Receiver ML521B) ความถี่ 1-300 MHz

■ ทางด้านสายอากาศรับจะสามารถหมุนได้ 360 องศา ได้โดยทำการต่อเข้ากับชุดเฟืองและแท่นหมุน และสเตปปีงมอเตอร์และต่อวงจรขับสเตปปีงมอเตอร์ ซึ่งมีบอร์ดคอนโทรลเลอร์เป็นตัวควบคุมการทำงาน

■ เครื่องรับสัญญาณจะส่งสัญญาณที่รับมาจากเครื่องส่ง และส่งสัญญาณมายัง ADC0809 วงจรนี้จะทำการถอดรหัสสัญญาณอนาลอกเป็นดิจิทัล และส่งเข้าคอมพิวเตอร์ทางพอร์ตอนุกรม

■ คอมพิวเตอร์จะแสดงผลการทดลองออกมาเป็นกราฟรูปแบบโพลาร์ และแบบสี่เหลี่ยม

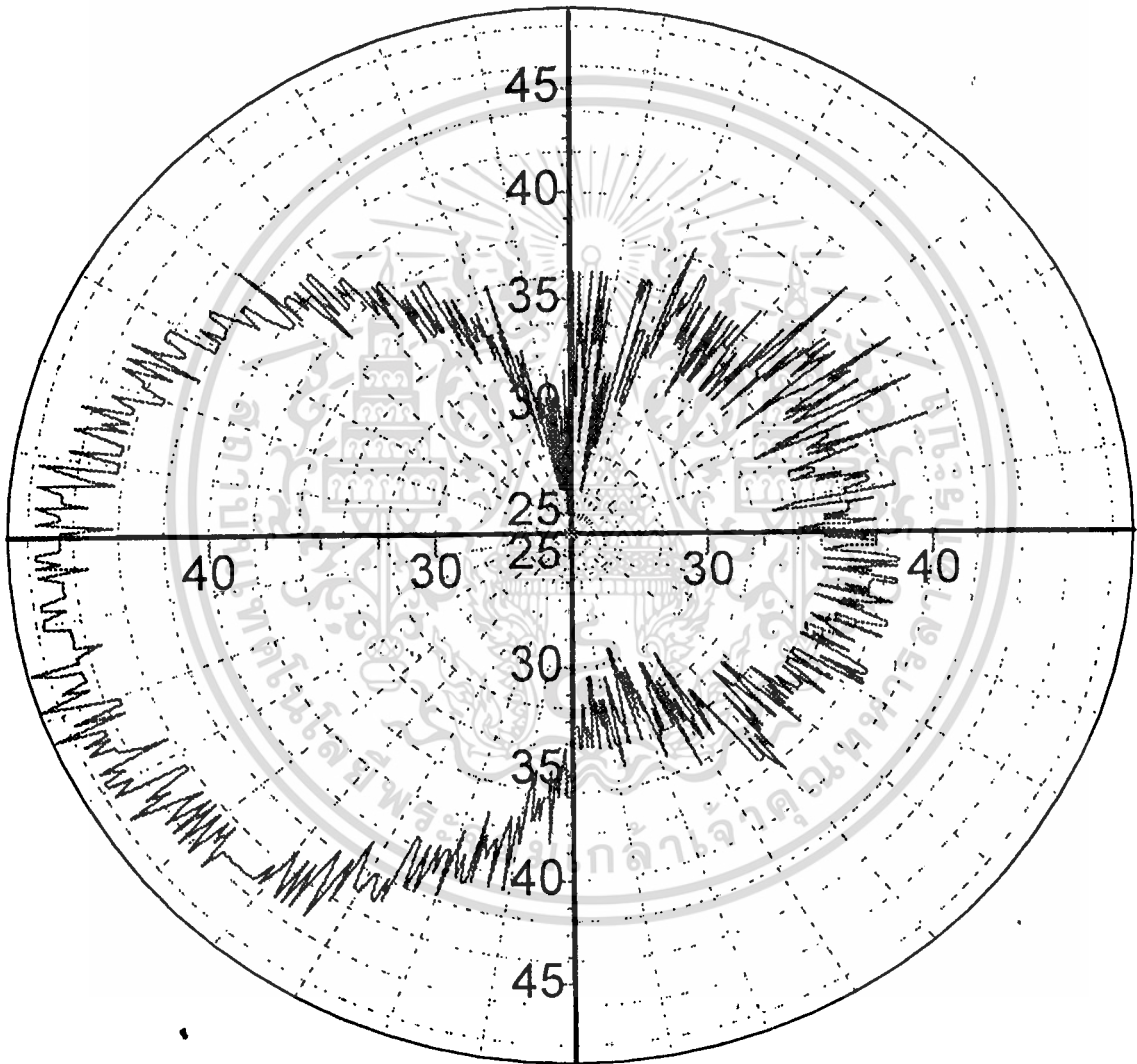


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

#### 4.5 ผลการทดลองการรันโปรแกรม

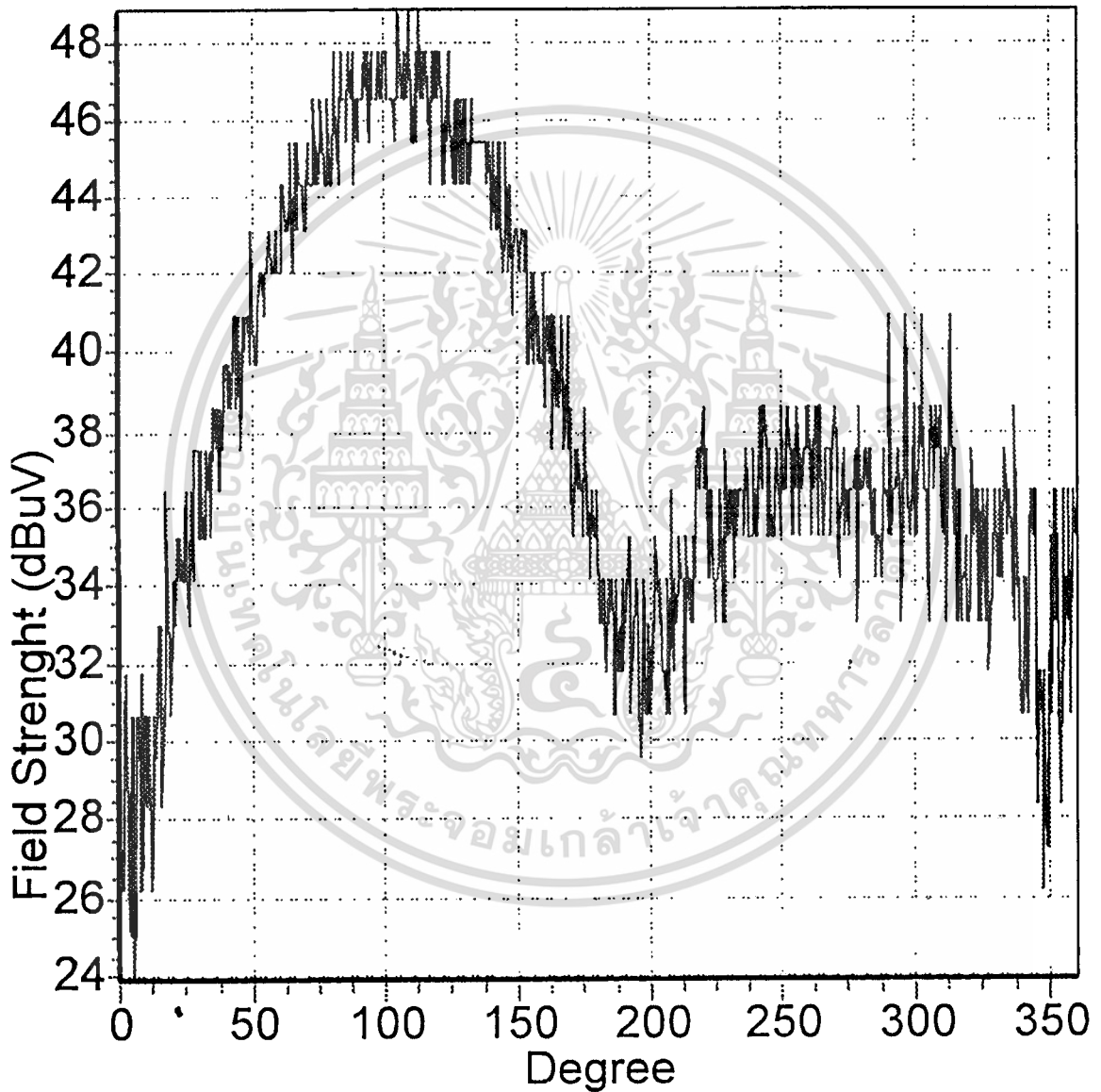
เมื่อทำการวัดแพทเทิร์นของสายอากาศโมโนโพลที่ความถี่ 300 MHz จะได้ผลการทดลองดังนี้

### Antenna Pattern Recorder



เอกสารนี้รูปที่ 4.10 แสดงผลการทดลองเป็นกราฟรูปโพลาร์ของแพทเทิร์นของสายอากาศแบบโมโนโพล  
ไม่ว่ากรณีใดๆ ทั้งทำการวัดที่ความถี่ 300 MHz นี้ขอและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## Antenna Pattern Recorder



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานที่การศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
 รูปที่ 4.11 แสดงผลการทดลองเป็นกราฟรูปสี่เหลี่ยมของแพทเทิร์นของสายอากาศแบบโมโนโพล  
 ไม่ว่าจะกรณีใดๆ ทั้งสิ้น ลิขสิทธิ์นี้เป็นของกองเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้  
 ทำการวัดที่ความถี่ 300 MHz

## บทที่ 5 บทวิจารณ์และบทสรุป

ระบบเครื่องปรับทิศทางของสายอากาศด้วยสเตปป์มอเตอร์ที่สร้างนี้สามารถนำมาใช้งานในการบันทึกแพทเทิร์นของสายอากาศเพราะมีความละเอียดและแน่นอนในการหมุนแต่ละสเตปป์

จากการศึกษาทฤษฎีสายอากาศและทดลองสร้างเครื่องปรับทิศทางของสายอากาศ ด้วยสเตปป์มอเตอร์ (Antenna Positional with Stepping Motor) ได้ตั้งแต่ 0-360 องศา ทางแนวราบจึงได้แนวคิดในการนำเครื่องปรับทิศทาง มาช่วยในการทดสอบแพทเทิร์นของสายอากาศได้ (antenna Pattern) เพราะสามารถโปรแกรมการทำงานลงในอีพროมของไมโครคอนโทรลเลอร์ ให้ใช้กับเครื่องคอมพิวเตอร์ส่วนบุคคล (Personal Computer) ได้ จึงสามารถปรับทิศทางของสายอากาศโดยผ่านทางซอฟต์แวร์ที่รันบนเครื่องพีซี ได้ และได้ทำการเขียนโปรแกรมในการพล็อตแพทเทิร์นของสายอากาศโดยใช้ภาษาวิจิวอลเบสิก ซึ่งสามารถรันบนวินโดวส์ได้และมีความสะดวกในการเขียนโปรแกรม และได้ผลการทดลองออกมาเป็นกราฟรูปโพลาร์และกราฟรูปสี่เหลี่ยม ได้ อย่างสวยงาม



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## หนังสืออ้างอิง

โชคชัย เดชพรุ่ง . Visual Basic 4 สำหรับ Windows 95 Step by Step . กรุงเทพฯ : บริษัท ซีเอ็ดดูเคชั่น จำกัด .

2539

สมยศ จุณณะปิยะ . การใช้งานไมโครคอนโทรลเลอร์ ตระกูล MCS-51. กรุงเทพฯ : โครงการตำราสถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง . 2537

ปรเมษฐ์ ประนายนันท์, ปิยะพงศ์ เผ่าวาณิช : คู่มือและการประยุกต์ใช้งานไมโครคอนโทรลเลอร์ ตระกูล MCS-51. กรุงเทพฯ : บริษัทซีเอ็ดดูเคชั่น จำกัด .

วิวัฒน์ กิรานนท์, โมนัย ไกรฤกษ์ . ทฤษฎีสายอากาศ . กรุงเทพฯ : โครงการตำราสถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ยีน ภู่วรรณ , สุธี จันทรัตน์วงศ์ . ทฤษฎีสายอากาศ . กรุงเทพฯ : บริษัท ซีเอ็ดดูเคชั่น จำกัด



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
Private Sub Command1_Click()  
'send "Step Forward" command  
MSComm1.Output = "F"
```

```
End Sub
```

```
Private Sub Command2_Click()  
Command2.Enabled = False  
Command1.Enabled = False  
Command3.Enabled = False  
Command5.Enabled = False  
Command6.Enabled = False  
Command4.Enabled = False
```

```
Dim n As Integer
```

```
For n = 1 To 666  
MSComm1.Output = "F"
```

```
'waits for data o be recieved  
'into serial buffer
```

```
Do
```

```
dummy = DoEvents()
```

```
Loop Until MSComm1.InBufferCount >= 1
```

```
'print value in text box and plot chart
```

```
TChart1.Series(0).AddXY 0.54 * n, Asc(MSComm1.Input) / 0.88, "",
```

```
vbRed
```

```
Next
```

```
Command1.Enabled = True
```

```
Command3.Enabled = True
```

```
Command5.Enabled = True
```

```
Command6.Enabled = True
```

```
Command2.Enabled = True
```

```
Command4.Enabled = True
```

```
' give time to Windows...
```

```
End Sub 'receive data and add to chart
```

```
Private Sub Command3_Click()
```

```
Do
```

```
dummy = DoEvents()
```

```
Loop Until MSComm1.InBufferCount >= 1
```

```
'print value in text box and plot chart
```

```
TChart1.Series(0).Add Asc(MSComm1.Input) / 0.88, "", vbRed
```

```
End Sub
```

```
Private Sub Command4_Click()
```

```
'clear chart
```

```
TChart1.Series(0).Clear
```

```
End Sub
```

```
Private Sub Command5_Click()
```

```
Dim i As Integer
```

```
Command1.Enabled = False
```

```
Command2.Enabled = False
```

```
Command3.Enabled = False
```

```
Command6.Enabled = False
```

```
For i = 1 To 165
```

```
MSComm1.Output = "B"
```

```
Next
```

```
Command1.Enabled = True
```

```
Command2.Enabled = True
```

```
Command3.Enabled = True
```

```
Command6.Enabled = True
```

```
End Sub
```

```
Private Sub Command6_Click()
'send "Step Backward" command
MSComm1.Output = "B"
End Sub
```

```
Private Sub Form_Load()
Option3.Value = True
TChart1.Axis.Bottom.Visible = False
TChart1.Axis.Left.Visible = False

End Sub
```

```
Private Sub mnuBitmapItem_Click()
CommonDialog1.Filter = "BitmapFiles (*.BMP)| *.BMP"
CommonDialog1.ShowSave
If CommonDialog1.filename <> "" Then
Form1.MousePointer = 11 'display hour glass
TChart1.Export.SaveToBitmapFile (CommonDialog1.filename)
Form1.MousePointer = 0 'reset mouse
Else
End If
End Sub
```

```
Private Sub mnuOpenItem_Click()
CommonDialog1.Filter = "Teefiles (*.Tee)| *.Tee"
CommonDialog1.ShowOpen
If CommonDialog1.filename <> "" Then
Form1.MousePointer = 11 'display hour glass
TChart1.Import.LoadFromFile (CommonDialog1.filename)
mnuCloseItem.Enabled = True
Form1.MousePointer = 0 'reset mouse
Else
End If
End Sub
```

```
Private Sub mnuPrintItem_Click()
TChart1.Printer.ShowPreview
End Sub
```

```
Private Sub mnuShowItem_Click()
TChart1.Export.ShowExport
End Sub
```

```
Private Sub Option1_Click()
TChart1.ChangeSeriesType 0, scFastLine
TChart1.Axis.Bottom.Visible = True
TChart1.Axis.Left.Visible = True

End Sub
```

```
Private Sub Option2_Click()
TChart1.ChangeSeriesType 0, scPolar
TChart1.Series(0).asPolar.Circled = True
TChart1.Series(0).asPolar.Pointer.Visible = False
```

```
TChart1.Series(0).asPolar.RotationAngle = 90
TChart1.Axis.Bottom.Automatic = False
TChart1.Axis.Bottom.SetMinMax 0, 360
TChart1.Axis.Bottom.Visible = True
TChart1.Axis.Left.Visible = True
```

```
End Sub
```

```
Private Sub Option3_Click()
    'Enable port
    MSComm1.PortOpen = True
    Command1.Enabled = True
    Command2.Enabled = True
    Command3.Enabled = True
    Command6.Enabled = True
    Command5.Enabled = True
End Sub
```

```
End Sub
```

```
Private Sub Option4_Click()
    'Disable port
    MSComm1.PortOpen = False
    Command1.Enabled = False
    Command2.Enabled = False
    Command3.Enabled = False
    Command6.Enabled = False
    Command5.Enabled = False
End Sub
```

```
End Sub
```

```
Private Sub mnuCloseItem_Click()
    TChart1.Series(0).Clear 'clear tchart
    mnuCloseItem.Enabled = False 'dim Close command
    mnuOpenItem.Enabled = True 'enable Open command
    'disable text box
End Sub
```

```
End Sub
```

```
Private Sub mnuExitItem_Click()
    End 'quit program
End Sub
```

```
End Sub
```

```
Private Sub TChart1_OnClick()
    Dim t As Integer
    For t = 0 To TChart1.SeriesCount - 1
        If TChart1.Series(t).GetMousePoint <> -1 Then
            MsgBox ("Clicked Series: " + TChart1.Series(t).Name + "at point: " +
                Str$(TChart1.Series(t).GetMousePoint))
        End If
    Next t
End Sub
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



**80C31BH/80C51BH/87C51**  
**MCS® 51**  
**CHMOS SINGLE-CHIP 8-BIT MICROCONTROLLER**  
*Automotive*

- Extended Automotive Temperature Range (-40°C to +125°C Ambient)
- High Performance CHMOS Process
- Power Control Modes
- 4 Kbyte-On-Chip ROM/EPROM
- 128 x 8-bit RAM
- 32 Programmable I/O Lines
- Two 16-Bit Timer/Counters
- 5 Interrupt Sources
- Quick-Pulse EPROM Programming
- 2-Level Program Memory Lock EPROM
- Boolean Processor
- Programmable Serial Port
- TTL- and CMOS-Compatible Logic Levels
- 64K External Program Memory Space
- 64K External Data Memory Space
- IDLE and POWER DOWN Modes
- ONCE Mode Facilitates System Testing
- Available in 12 MHz and 16 MHz Versions
- Available in PLCC and DIP Packages  
(See Packaging Specification, Order #231369)

The MCS® 51 CHMOS microcontroller products are fabricated on Intel's reliable CHMOS process and are functionally compatible with the standard MCS 51 HMOS microcontroller products. This technology combines the high speed and density characteristics of HMOS with the low power attributes of CHMOS. This combination expands the effectiveness of the powerful MCS 51 microcontroller architecture and instruction set.

Like the MCS 51 HMOS microcontroller versions, the MCS 51 CHMOS microcontroller products have the following features: 4 Kbytes of EPROM/ROM (87C51/80C51BH respectively); 128 bytes of RAM; 32 I/O lines; two 16-bit timer/counters; a five-source two-level interrupt structure; a full duplex serial port; and on-chip oscillator and clock circuitry. In addition, the MCS 51 CHMOS microcontroller products exhibit low operating power, along with two software selectable modes of reduced activity for further power reduction—Idle and Power Down.

The Idle mode freezes the CPU while allowing the RAM, timer/counters, serial port, and interrupt system to continue functioning. The Power Down mode saves the RAM contents but freezes the oscillator, causing all other chip functions to be inoperative.

The 87C51 is the EPROM version of the 80C51BH. It contains 4 Kbytes of on-chip program memory that can be electrically programmed, and can be erased by exposure to ultraviolet light. The 87C51 EPROM array uses a modified Quick-Pulse Programming algorithm, by which the entire 4 Kbyte array can be programmed in about 12 seconds.

**NOTICE:**

This datasheet contains information on products in full production. Specifications within this datasheet are subject to change without notice. Verify with your local Intel sales office that you have the latest datasheet before finalizing a design.

\*Other brands and names are the property of their respective owners.  
Information in this document is provided in connection with Intel products. Intel assumes no liability whatsoever, including infringement of any patent or copyright, for sale and use of Intel products except as provided in Intel's Terms and Conditions of Sale for such products. Intel retains the right to make changes to these specifications at any time, without notice. Microcomputer Products may have minor variations to this specification known as errata.  
COPYRIGHT © INTEL CORPORATION, 1995      January 1985      Order Number: 270418-007

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

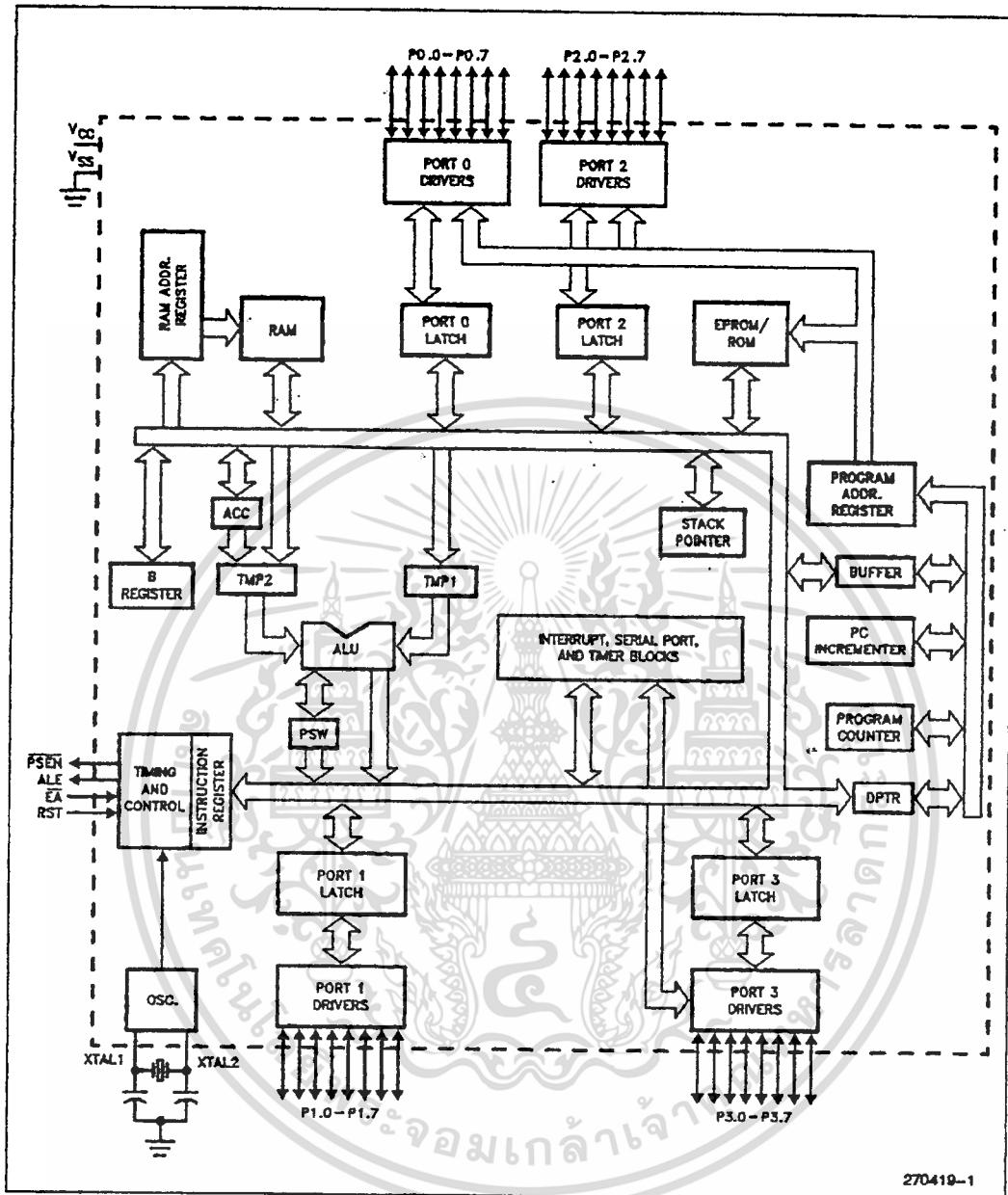


Figure 1. MCS<sup>®</sup> 51 Microcontroller Architectural Block Diagram

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



## AUTOMOTIVE 80C31BH/80C51BH/87C51

### 80C31BH/80C51BH/87C51 PRODUCT OPTIONS

Intel's extended and automotive temperature range products are designed to meet the needs of those applications whose operating requirements exceed commercial standards.

With the extended temperature range option, operational characteristics are guaranteed over the temperature range of  $-40^{\circ}\text{C}$  to  $+85^{\circ}\text{C}$  ambient. For the

automotive temperature range option, operational characteristics are guaranteed over the temperature range of  $-40^{\circ}\text{C}$  to  $+125^{\circ}\text{C}$  ambient.

The automotive and extended temperature versions of the MCS 51 microcontroller product families are available with or without burn-in options as listed in Table 1.

As shown in Figure 2, temperature, burn-in, and package options are identified by a one- or two-letter prefix to the part number.

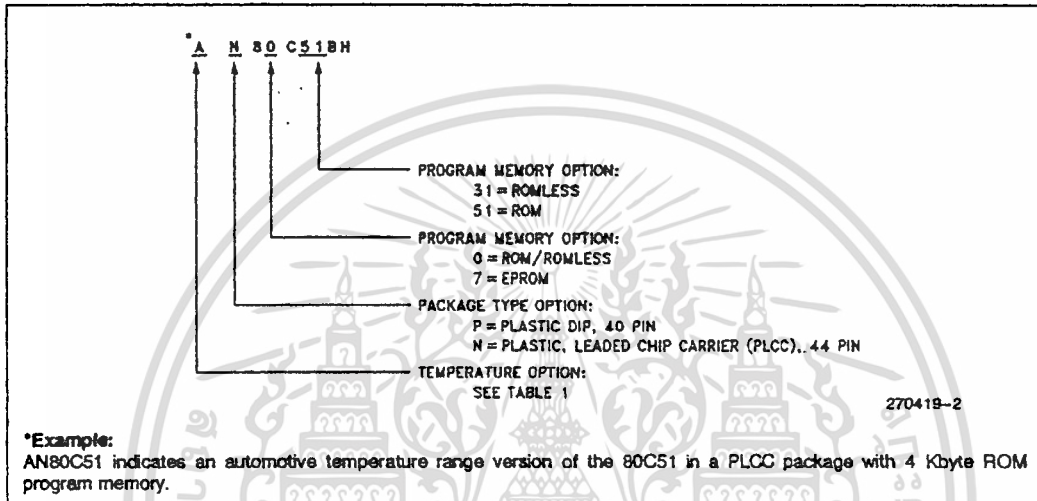
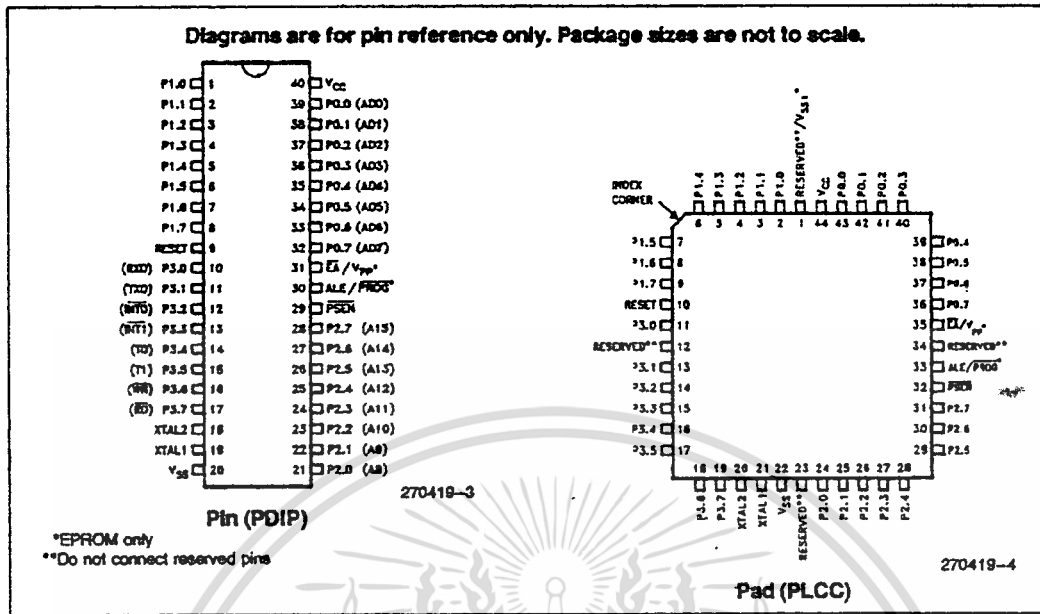


Figure 2. MCS<sup>®</sup> 51 Microcontroller Product Family Nomenclature

Table 1. Temperature Options

Temperature Classification	Temperature Designation	Operating Temperature $^{\circ}\text{C}$ Ambient	Burn-in Options
Extended	T	$-40$ to $+85$	Standard
	L	$-40$ to $+85$	Extended
Automotive	A	$-40$ to $+125$	Standard
	B	$-40$ to $+125$	Extended



Port 3 also serves the functions of various special features of the MCS 51 microcontroller family, as listed below:

Pin	Name	Alternate Function
P3.0	FXD	Serial Input Line
P3.1	TXD	Serial Output Line
P3.2	INT0	External Interrupt 0
P3.3	INT1	External Interrupt 1
P3.4	T0	Timer 0 External Input
P3.5	T1	Timer 1 External Input
P3.6	WR	External Data Memory Write Strobe
P3.7	RD	External Data Memory Read Strobe

Port 3 also receives some control signals for EPROM programming and program verification.

**RESET:** Reset input. A logic high on this pin for two machine cycles while the oscillator is running resets the device. An internal pulldown resistor permits a power-on reset to be generated using only an external capacitor to  $V_{CC}$ .

**ALE/PROG (EPROM Only):** Address Latch Enable output signal for latching the low byte of the address during accesses to external memory. This pin is also the program pulse input (PROG) during EPROM programming.

In normal operation ALE is emitted at a constant rate of 1/6 the oscillator frequency, and may be used for external timing or clocking purposes. Note, however, that one ALE pulse is skipped during each access to external Data Memory.

**PSEN:** Program Store Enable is the Read strobe to External Program Memory. When the 87C51/80C51BH is executing from Internal Program Memory, PSEN is inactive (high). When the device is executing code from External Program Memory, PSEN is activated twice each machine cycle, except that two PSEN activations are skipped during each access to External Data Memory.

**EA/Vpp:** External Access enable. EA must be strapped to  $V_{SS}$  in order to enable the 87C51/80C51BH to fetch code from External Program Memory locations starting at 0000H up to 0FFFFH. [Note, however, that if either of the Lock Bits is programmed, the logic level at EA is internally latched during reset.] (EPROM only.)

EA must be strapped to  $V_{CC}$  for internal program execution.

**Vpp (EPROM Only):** This pin also receives the 12.75V programming supply voltage ( $V_{pp}$ ) during EPROM programming.

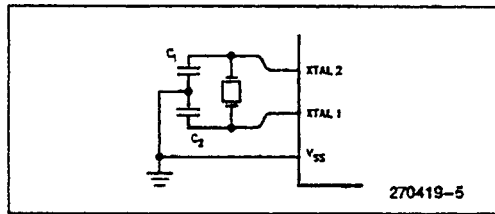


Figure 4. Using the On-Chip Oscillator

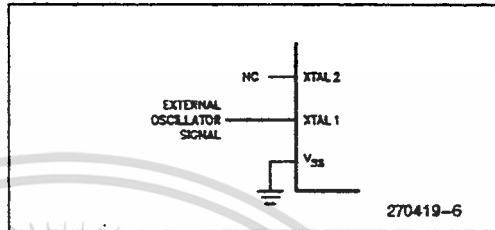


Figure 5. External Clock Drive

**XTAL1:** Input to the inverting oscillator amplifier and input to the internal clock generating circuits.

**XTAL2:** Output from the inverting oscillator amplifier.

### OSCILLATOR CHARACTERISTICS

XTAL1 and XTAL2 are the input and output, respectively, of an inverting amplifier which can be configured for use as an on-chip oscillator, as shown in Figure 4.

To drive the device from an external clock source, XTAL1 should be driven, while XTAL2 is left unconnected, as shown in Figure 5. There are no requirements on the duty cycle of the external clock signal, since the input to the internal clocking circuitry is through a divide-by-two flip-flop, but minimum and maximum high and low times specified on the Data-sheet must be observed.

### IDLE MODE

In Idle Mode, the CPU puts itself to sleep while all the on-chip peripherals remain active. The mode is invoked by software. The content of the on-chip RAM and all the Special Functions Registers remain unchanged during this mode. The Idle Mode can be terminated by any enabled interrupt or by a hardware reset.

It should be noted that when Idle is terminated by a hardware reset, the device normally resumes program execution, from where it left off, up to two machine cycles before the internal reset algorithm takes control. On-chip hardware inhibits access to

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Table 2. Status of the External Pins During Idle and Power Down

Mode	Program Memory	ALE	PSEN	PORT0	PORT1	PORT2	PORT3
Idle	Internal	1	1	Data	Data	Data	Data
Idle	External	1	1	Float	Data	Address	Data
Power Down	Internal	0	0	Data	Data	Data	Data
Power Down	External	0	0	Float	Data	Data	Data

**NOTE:**

For more detailed information on the reduced power modes refer to current Embedded Applications Handbook, and Application Note AP-252, "Designing with the 80C51BH."

internal RAM in this event, but access to the port pins is not inhibited. To eliminate the possibility of an unexpected write to a port pin when Idle is terminated by reset, the instruction following the one that invokes Idle should not be one that writes to a port pin or to external memory.

**POWER DOWN MODE**

In the Power Down mode the oscillator is stopped, and the instruction that invokes Power Down is the last instruction executed. The on-chip RAM and Special Function Registers retain their values until the Power Down mode is terminated.

The only exit from Power Down is a hardware reset. Reset redefines the SFRs but does not change the on-chip RAM. The reset should not be activated before V<sub>CC</sub> is restored to its normal operating level and must be held active long enough to allow the oscillator to restart and stabilize.

**DESIGN CONSIDERATIONS**

- At power on, the voltage on V<sub>CC</sub> and RST must come up at the same time for a proper start-up.
- Before entering the Power Down mode the contents of the Carry Bit and B.7 must be equal.
- When the Idle mode is terminated by a hardware reset, the device normally resumes program execution, from where it left off, up to two machine cycles before the internal reset algorithm takes control. On-chip hardware inhibits access to internal RAM in this event, but access to the port pins is not inhibited. To eliminate the possibility of an unexpected write when Idle is terminated by reset, the instruction following the one that invokes Idle should not be one that writes to a port pin or to external memory.
- An external oscillator may encounter as much as a 100 pF load at XTAL1 when it starts up. This is due to interaction between the amplifier and its feedback capacitance. Once the external signal meets the V<sub>IL</sub> and V<sub>IH</sub> specifications the capacitance will not exceed 20 pF.
- For EPROM versions exposure to light when the device is in operation may cause logic errors. For this reason, it is suggested that an opaque label be placed over the window when the die is exposed to ambient light.



### PROGRAM MEMORY LOCK (EPROM Only)

The 87C51 contains two program memory lock schemes: Encrypted Verify and Lock Bits.

**Encrypted Verify:** The 87C51 implements a 32-byte EPROM array that can be programmed by the customer, and which can then be used to encrypt the program code bytes during EPROM verification. The EPROM verification procedure is performed as usual, except that each code byte comes out logically X-NORed with one of the 32 key bytes. The key bytes are gone through in sequence. Therefore, to read the ROM code, one has to know the 32 key bytes in their proper sequence.

**Lock Bits:** Also on the chip are two Lock Bits which can be left unprogrammed (U) or can be programmed (P) to obtain the following additional features:

Bit 1	Bit 2	Additional Features
U	U	none
P	U	<ul style="list-style-type: none"> <li>Externally fetched code can not access internal Program Memory.</li> <li>Further programming disabled.</li> </ul>
U	P	(Reserved for Future definition.)
P	P	<ul style="list-style-type: none"> <li>Externally fetched code can not access internal Program Memory.</li> <li>Further programming disabled.</li> <li>Program verification is disabled.</li> </ul>

When Lock Bit 1 is programmed, the logic level at the  $\overline{EA}$  pin is sampled and latched during reset. If the device is powered up without a reset, the latch initializes to a random value, and holds that value until reset is activated. It is necessary that the latched value of  $\overline{EA}$  be in agreement with the current logic level at that pin in order for the device to function properly.

### ONCE MODE

The ONCE ("on-circuit emulation") mode facilitates testing and debugging of systems using the 87C51 without the 87C51 having to be removed from the circuit. The ONCE mode is invoked by:

1. Pull ALE low while the device is in reset and  $\overline{PSEN}$  is high;
2. Hold ALE low as RST is deactivated.

While the device is in ONCE mode, the Port 0 pins go into a float state, and the other port pins and ALE and  $\overline{PSEN}$  are weakly pulled high. The oscillator circuit remains active. While the 87C51 is in this mode, an emulator or test CPU can be used to drive the circuit. Normal operation is restored when a normal reset is applied.



**ABSOLUTE MAXIMUM RATINGS\***

Ambient Temperature Under Bias  $-40^{\circ}\text{C}$  to  $+125^{\circ}\text{C}$   
 Storage Temperature .....  $-65^{\circ}\text{C}$  to  $+150^{\circ}\text{C}$   
 Voltage on  $\overline{\text{EA}}/\text{V}_{\text{PP}}$  Pin to  $\text{V}_{\text{SS}}$  .....  $0\text{V}$  to  $+13.0\text{V}$   
 Voltage on Any Other Pin to  $\text{V}_{\text{SS}}$  ..  $-0.5\text{V}$  to  $+6.5\text{V}$   
 $\text{I}_{\text{OL}}$  per I/O pin .....  $15\text{ mA}$   
 Power Dissipation.....  $1.5\text{W}$   
 (Based on package heat transfer limitations, not device power consumption).  
 Typical Junction Temperature ( $\text{T}_{\text{J}}$ ) .....  $+135^{\circ}\text{C}$   
 (Based upon ambient temperature at  $+125^{\circ}\text{C}$ )  
 Typical Thermal Resistance Junction-to-Ambient ( $\theta_{\text{JA}}$ ):  
     PDIP .....  $75^{\circ}\text{C}/\text{W}$   
     PLCC.....  $46^{\circ}\text{C}/\text{W}$

NOTICE: This is a production data sheet. The specifications are subject to change without notice.

*\*WARNING: Stressing the device beyond the "Absolute Maximum Ratings" may cause permanent damage. These are stress ratings only. Operation beyond the "Operating Conditions" is not recommended and extended exposure beyond the "Operating Conditions" may affect device reliability.*

**DC CHARACTERISTICS:**

( $\text{T}_{\text{A}} = -40^{\circ}\text{C}$  to  $+125^{\circ}\text{C}$ ;  $\text{V}_{\text{CC}} = 5\text{V} \pm 10\%$  ( $5\text{V} \pm 20\%$  EPROM Only);  $\text{V}_{\text{SS}} = 0\text{V}$ )

Symbol	Parameter	Min	Typ(1)	Max (87C51/80C51BH)	Unit	Test Conditions
$\text{V}_{\text{IL}}$	Input Low Voltage (Except $\overline{\text{EA}}$ )	$-0.5$		$0.2\text{V}_{\text{CC}} - 0.25$	V	
$\text{V}_{\text{IL1}}$	Input Low Voltage to $\overline{\text{EA}}$	0		$0.2\text{V}_{\text{CC}} - 0.45$	V	
$\text{V}_{\text{IH}}$	Input High Voltage (Except XTAL1, RST)	$0.2\text{V}_{\text{CC}} + 1.0$		$\text{V}_{\text{CC}} + 0.5$	V	
$\text{V}_{\text{IH1}}$	Input High Voltage (XTAL1, RST)	$0.7\text{V}_{\text{CC}} + 0.1$		$\text{V}_{\text{CC}} + 0.5$	V	
$\text{V}_{\text{OL}}$	Output Low Voltage (Ports 1, 2, 3)			0.45(7)	V	$\text{I}_{\text{OL}} = 1.6\text{ mA}$ (2)
$\text{V}_{\text{OL1}}$	Output Low Voltage (Port 0, ALE, $\overline{\text{PSEN}}$ )			0.45(7)	V	$\text{I}_{\text{OL}} = 3.2\text{ mA}$ (2)
$\text{V}_{\text{OH}}$	Output High Voltage (Ports 1, 2, 3, ALE, $\overline{\text{PSEN}}$ )	2.4			V	$\text{I}_{\text{OH}} = -60\ \mu\text{A}$
		$0.9\text{V}_{\text{CC}}$			V	$\text{I}_{\text{OH}} = -10\ \mu\text{A}$
$\text{V}_{\text{OH1}}$	Output High Voltage (Port 0 in External Bus Mode)	2.4			V	$\text{I}_{\text{OH}} = -800\ \mu\text{A}$
		$0.9\text{V}_{\text{CC}}$			V	$\text{I}_{\text{OH}} = -80\ \mu\text{A}$ (3)
$\text{I}_{\text{IL}}$	Logical 0 Input Current (Ports 1, 2, 3)			-75	$\mu\text{A}$	$\text{V}_{\text{IN}} = 0.45\text{ V}$
$\text{I}_{\text{TL}}$	Logical 1-to-0 transition current (Ports 1, 2, 3)			-750	$\mu\text{A}$	(4)
$\text{I}_{\text{LI}}$	Input Leakage Current (Port 0)			$\pm 10$	$\mu\text{A}$	$\text{V}_{\text{IN}} = \text{V}_{\text{IL}}$ or $\text{V}_{\text{IH}}$
$\text{I}_{\text{CC}}$	Power Supply Current: Active Mode @ 12 MHz (5) Idle Mode @ 12 MHz (5) Power Down Mode		11.5	25/20	mA	(6) $\text{V}_{\text{CC}} = 2.2\text{V}$ to $5.5\text{V}$
			1.3	6/5	mA	
			3	100/75	$\mu\text{A}$	
RRST	Internal Reset Pulldown Resistor	50		300	k $\Omega$	
CIO	Pin Capacitance			10	pF	

**NOTES:**

1. "Typicals" are based on a limited number of samples taken from early manufacturing lots and are not guaranteed. The values listed are at room temp, 5V.
2. Capacitive loading on Ports 0 and 2 may cause spurious noise pulses to be superimposed on the  $\text{V}_{\text{OL}}$ s of ALE and Ports 1 and 3. The noise is due to external bus capacitance discharging into the Port 0 and Port 2 pins when these pins make 1-to-0 transitions during bus operations. In the worst cases (capacitive loading  $> 100\text{pF}$ ), the noise pulse on the ALE pin may exceed 0.6V. In such cases it may be desirable to qualify ALE with a Schmitt Trigger, or use an address latch with a Schmitt Trigger STROBE input.
3. Capacitive loading on Ports 0 and 2 may cause the  $\text{V}_{\text{OH}}$  on ALE and  $\overline{\text{PSEN}}$  to momentarily fall below the  $0.9\text{V}_{\text{CC}}$  specification when the address bits are stabilizing.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
 ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

**NOTES: (Continued)**

4. Pins of Ports 1, 2 and 3 source a transition current when they are being externally driven from 1 to 0. The transition current reaches its maximum value when  $V_{IN}$  is approximately 2V.

5. ICCMAX at other frequencies is given by:

Active Mode: 87C51  $ICCMAX = 0.94 \times FREQ + 13.71$   
 80Cx1BH  $ICCMAX = 1.47 \times FREQ + 2.36$

Idle Mode:  $ICCMAX = 0.14 \times FREQ + 3.81$

where FREQ is the external oscillator frequency in MHz. ICCMAX is given in mA. See Figure 6.

6. See Figures 7 through 10 for  $I_{CC}$  test conditions. Minimum  $V_{CC}$  for Power Down is 2.0V.

7. Under steady state (non-transient) conditions,  $I_{OL}$  must be externally limited as follows:

Maximum $I_{OL}$ per port pin:	10 mA
Maximum $I_{OL}$ per 8-bit port:	

Port 0:	26 mA
Ports 1, 2, and 3:	15 mA
Maximum total $I_{OL}$ for all output pins:	71 mA

If  $I_{OL}$  exceeds the test condition,  $V_{OL}$  may exceed the related specification. Pins are not guaranteed to sink current greater than the listed test conditions.

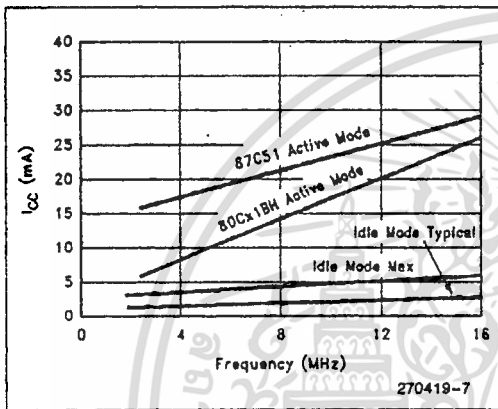


Figure 6.  $I_{CC}$  vs. FREQ. Valid only within frequency specifications of the device under test.

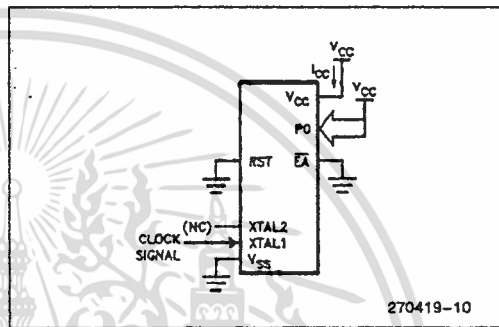


Figure 8.  $I_{CC}$  Test Condition, Idle Mode. All other pins are disconnected.

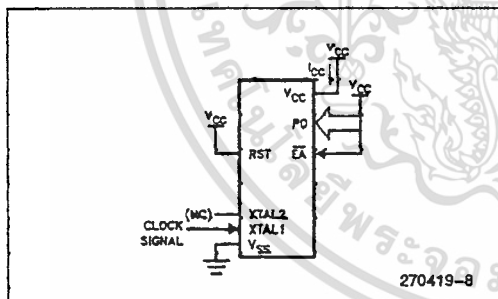


Figure 7.  $I_{CC}$  Test Condition, Active Mode. All other pins are disconnected.

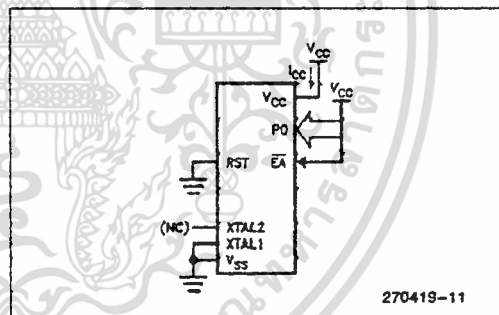


Figure 9.  $I_{CC}$  Test Condition, Power Down Mode. All other pins are disconnected.

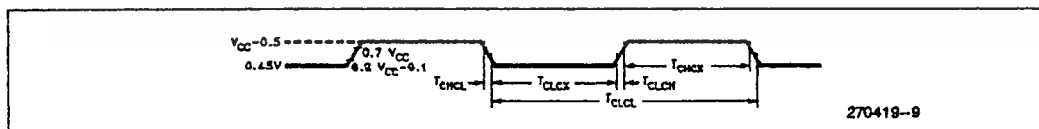


Figure 10. Clock Signal Waveform for  $I_{CC}$  tests in Active and Idle Modes.  $T_{CLCH} = T_{CHCL} = 5$  ns.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
 ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



**EXPLANATION OF THE AC SYMBOLS**

Each timing symbol has 5 characters. The first character is always a 'T' (stands for time). The other characters, depending on their positions, stand for the name of a signal or the logical status of that signal. The following is a list of all the characters and what they stand for.

- L: Logic level LOW, or ALE
- P: PSEN.
- Q: Output data.
- R: RD signal.
- T: Time.
- V: Valid.
- W: WR signal.
- X: No longer a valid logic level.
- Z: Float.

- A: Address.
- C: Clock.
- D: Input data.
- H: Logic level HIGH.
- I: Instruction (program memory contents).

For example,

- TAVLL = Time from Address Valid to ALE Low.
- TLLPL = Time from ALE Low to PSEN Low.

**AC CHARACTERISTICS:** (T<sub>A</sub> = -40°C to +125°C; V<sub>CC</sub> = 5V ±10% (5V ±20% EPROM Only); V<sub>SS</sub> = 0V; Load Capacitance for Port 0, ALE, and PSEN = 100 pF; Load Capacitance for All Other Outputs = 80 pF)

**EXTERNAL PROGRAM AND DATA MEMORY CHARACTERISTICS**

Symbol	Parameter	12 Mhz Osc		Variable Oscillator		Units
		Min	Max	Min	Max	
1/T <sub>CLCL</sub>	Oscillator Frequency 87C51/80C51BH/80C31BH			3.5	12-16	MHz
T <sub>LHLL</sub>	ALE Pulse Width	127		2T <sub>CLCL</sub> - 40		ns
T <sub>AVLL</sub>	Address Valid to ALE Low	28		T <sub>CLCL</sub> - 55		ns
T <sub>LLAX</sub>	Address Hold After ALE Low	48		T <sub>CLCL</sub> - 35		ns
T <sub>LLIV</sub>	ALE Low to Valid Instr In		224		4T <sub>CLCL</sub> - 110	ns
T <sub>LLPL</sub>	ALE Low to PSEN Low	43		T <sub>CLCL</sub> - 40		ns
T <sub>PLPH</sub>	PSEN Pulse Width	205		3T <sub>CLCL</sub> - 45		ns
T <sub>PLIV</sub>	PSEN Low to Valid Instr In		135		3T <sub>CLCL</sub> - 115	ns
T <sub>PXIX</sub>	Input Instr Hold After PSEN	0		0		ns
T <sub>PXIZ</sub>	Input Instr Float After PSEN		59		T <sub>CLCL</sub> - 25	ns
T <sub>AVIV</sub>	Address Valid to Valid Instr In		312		5T <sub>CLCL</sub> - 105	ns
T <sub>PLAZ</sub>	PSEN Low to Address Float		10		10	ns
T <sub>RLRH</sub>	RD Pulse Width	400		6T <sub>CLCL</sub> - 100		ns
T <sub>WLWH</sub>	WR Pulse Width	400		6T <sub>CLCL</sub> - 100		ns
T <sub>RLDV</sub>	RD Low to Valid Data In		252		5T <sub>CLCL</sub> - 165	ns
T <sub>RHDX</sub>	Data Hold After RD High	0		0		ns
T <sub>RHOZ</sub>	Data Float After RD High		97		2T <sub>CLCL</sub> - 70	ns
T <sub>LLDV</sub>	ALE Low to Valid Data In		517		8T <sub>CLCL</sub> - 150	ns
T <sub>AVDV</sub>	Address Valid to Valid Data In		585		9T <sub>CLCL</sub> - 165	ns
T <sub>LLWL</sub>	ALE Low to RD or WR Low	200	300	3T <sub>CLCL</sub> - 50	3T <sub>CLCL</sub> + 50	ns
T <sub>AVWL</sub>	Address Valid to RD or WR Low	203		4T <sub>CLCL</sub> - 130		ns
T <sub>QVWX</sub>	Data Valid to WR Transition	23		T <sub>CLCL</sub> - 60		ns
T <sub>WHQX</sub>	Data Hold After WR High	33		T <sub>CLCL</sub> - 50		ns
T <sub>RLAZ</sub>	RD Low to Address Float		0		0	ns
T <sub>WHLH</sub>	RD or WR High to ALE High	43	123	T <sub>CLCL</sub> - 40	T <sub>CLCL</sub> + 40	ns

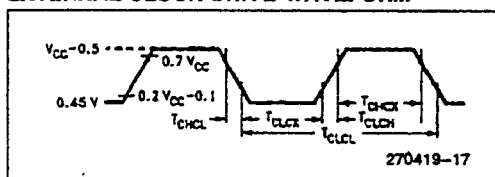
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



EXTERNAL CLOCK DRIVE

Symbol	Parameter	Min	Max	Units
1/T <sub>CLCL</sub>	Oscillator Frequency	3.5	12	MHz
T <sub>CHCX</sub>	High Time	20		ns
T <sub>CLCX</sub>	Low Time	20		ns
T <sub>CLCH</sub>	Rise Time		20	ns
T <sub>CHCL</sub>	Fall Time		20	ns

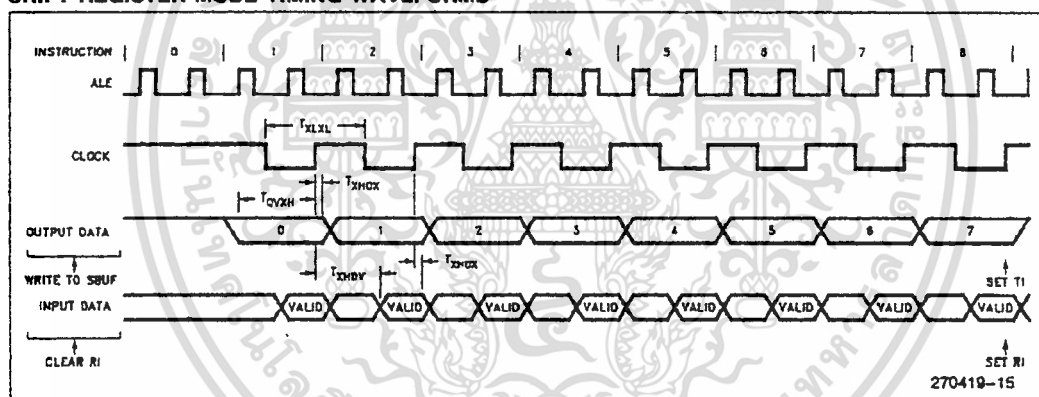
EXTERNAL CLOCK DRIVE WAVEFORM



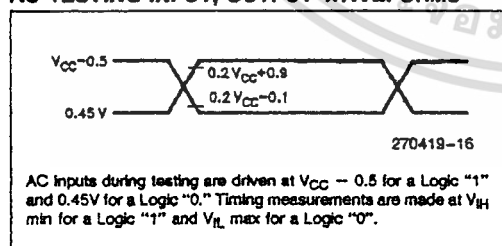
SERIAL PORT TIMING—SHIFT REGISTER MODE

Symbol	Parameter	12 MHz Oscillator		Variable Oscillator		Units
		Min	Max	Min	Max	
T <sub>XLXL</sub>	Serial Port Clock Cycle Time	1.0		12T <sub>CLCL</sub>		μs
T <sub>QVXH</sub>	Output Data Setup to Clock Rising Edge	700		10T <sub>CLCL</sub> - 133		ns
T <sub>XHQX</sub>	Output Data Hold After Clock Rising Edge	50		2T <sub>CLCL</sub> - 117		ns
T <sub>XHDX</sub>	Input Data Hold After Clock Rising Edge	0		0		ns
T <sub>XHDV</sub>	Clock Rising Edge to Input Data Valid		700		10T <sub>CLCL</sub> - 133	ns

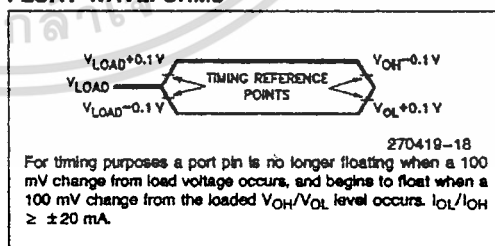
SHIFT REGISTER MODE TIMING WAVEFORMS



AC TESTING INPUT, OUTPUT WAVEFORMS



FLOAT WAVEFORMS



**EPROM CHARACTERISTICS  
(EPROM Only)**

The 87C51 is programmed by a modified Quick-Pulse Programming algorithm. It differs from older methods in the value used for  $V_{pp}$  (Programming Supply Voltage) and in the width and number of the ALE/PROG pulses.

The 87C51 contains two signature bytes that can be read and used by an EPROM programming system

to identify the device. The signature bytes identify the device as an 87C51 manufactured by Intel.

Table 3 shows the logic levels for reading the signature byte, and for programming the Program Memory, the Encryption Table, and the Lock Bits. The circuit configuration and waveforms for Quick-Pulse Programming are shown in Figures 11 and 12. Figure 13 shows the circuit configuration for normal Program Memory verification.

**Table 3. EPROM Programming Modes**

MODE	RST	PSEN	ALE/ PROG	EA/ V <sub>pp</sub>	P2.7	P2.6	P3.7	P3.6
Read Signature	1	0	1	1	0	0	0	0
Program Code Data	1	0	0*	V <sub>pp</sub>	1	0	1	1
Verify Code Data	1	0	1	1	0	0	1	1
Pgm Encryption Table	1	0	0*	V <sub>pp</sub>	1	0	1	0
Pgm Lock Bit 1	1	0	0*	V <sub>pp</sub>	1	1	1	1
Pgm Lock Bit 2	1	0	0*	V <sub>pp</sub>	1	1	0	0

**NOTES:**

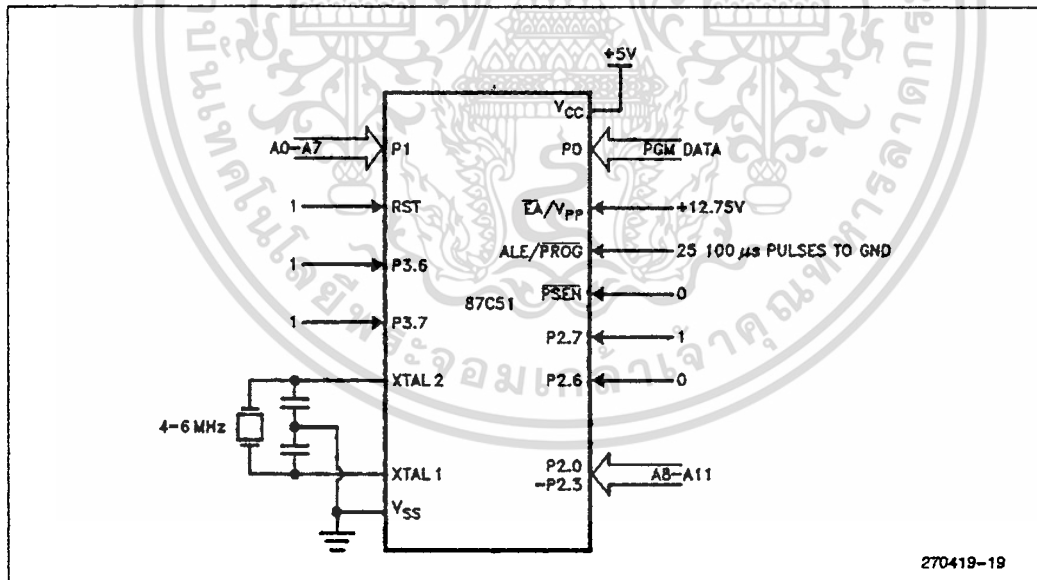
"1" = Valid high for that pin

"0" = Valid low for that pin

$V_{pp} = 12.75V \pm 0.25V$

$V_{CC} = 5V \pm 20\%$  during programming and verification

\*ALE/PROG receives 25 programming pulses while  $V_{pp}$  is held at 12.75V. Each programming pulse is low for 100  $\mu s$  ( $\pm 10 \mu s$ ) and high for a minimum of 10  $\mu s$ .



**Figure 11. Programming Configuration**

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

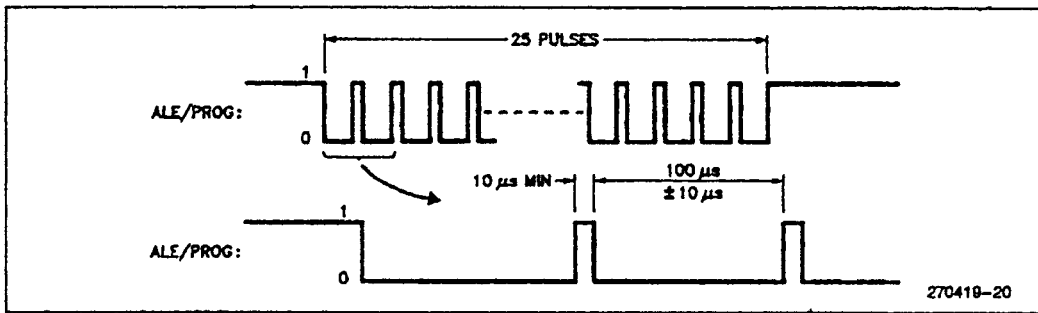


Figure 12. PROG Waveforms

### Quick-Pulse Programming (EPROM Only)

The setup for Microcontroller Quick-Pulse Programming is shown in Figure 11. Note that the 87C51 is running with a 4 to 6 MHz oscillator. The reason the oscillator needs to be running is that the device is executing internal address and program data transfers.

The address of the EPROM location to be programmed is applied to Ports 1 and 2, as shown in Figure 11. The code byte to be programmed into that location is applied to Port 0. RST, PSEN, and pins of Ports 2 and 3 specified in Table 3 are held at the "Program Code Data" levels indicated in Table 2. Then ALE/PROG is pulsed low 25 times as shown in Figure 12.

To program the Encryption Table, repeat the 25-pulse programming sequence for addresses 0

through 1FH, using the "Pgm Encryption Table" levels. Don't forget that after the Encryption Table is programmed, verify cycles will produce only encrypted data.

To program the Lock Bits, repeat the 25-pulse programming sequence using the "Pgm Lock Bit" levels. After one Lock Bit is programmed, further programming of the Code Memory and Encryption Table is disabled. However, the other Lock Bit can still be programmed.

Note that the  $\overline{EA}/V_{PP}$  pin must not be allowed to go above the maximum specified  $V_{PP}$  level for any amount of time. Even a narrow glitch above that voltage level can cause permanent damage to the device. The  $V_{PP}$  source should be well regulated and free of glitches and overshoot.

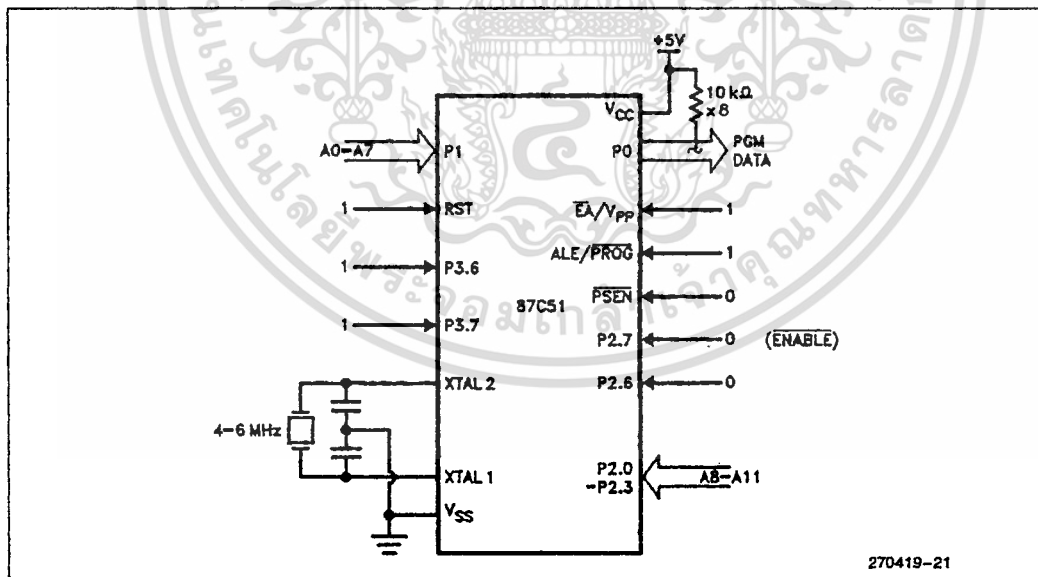


Figure 13. Program Verification

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

**Program Verification  
(EPROM Only)**

If Lock Bit 2 has not been programmed, the on-chip Program Memory can be read out for program verification. The address of the Program Memory location to be read is applied to Ports 1 and 2 as shown in Figure 13. The other pins are held at the "Verify Code Data" levels indicated in Table 3. The contents of the addressed location will be emitted on Port 0. External pullups are required on Port 0 for this operation. Detailed timing specifications are shown in later sections of this datasheet.

If the Encryption Table has been programmed, the data presented at Port 0 will be the Exclusive NOR of the program byte with one of the encryption bytes. The user will have to know the Encryption Table contents in order to correctly decode the verification data. The Encryption Table itself can not be read out.

**Reading the Signature Bytes  
(EPROM Only)**

The signature bytes are read by the same procedure as a normal verification of locations 030H and 031H, except that P3.6 and P3.7 need to be pulled to a logic low. The values returned are:

- (030H) = 89H indicates manufactured by Intel
- (031H) = 57H indicates 87C51

**Program/Verify Algorithms  
(EPROM Only)**

Any algorithm in agreement with the conditions listed in Table 3, and which satisfies the timing specifications, is suitable.

**Erase Characteristics  
(EPROM Only)**

Erase of the EPROM begins to occur when the chip is exposed to light with wavelengths shorter than approximately 4,000 Angstroms. Since sunlight and fluorescent lighting have wavelengths in this range, exposure to these light sources over an extended time (about 1 week in sunlight, or 3 years in room level fluorescent lighting) could cause inadvertent erasure. If an application subjects the device to this type of exposure, it is suggested that an opaque label be placed over the window.

The recommended erasure procedure is exposure to ultraviolet light (at 2537 Angstroms) to an integrated dose of at least 15 W-sec/cm<sup>2</sup>. Exposing the EPROM to an ultraviolet lamp of 12,000 μW/cm<sup>2</sup> rating for 30 minutes, at a distance of about 1 inch, should be sufficient.

Erase leaves the array in an all 1s state.

**EPROM PROGRAMMING AND VERIFICATION CHARACTERISTICS:**

(T<sub>A</sub> = 21°C to 27°C, V<sub>CC</sub> = 5V ±20%, V<sub>SS</sub> = 0V)

Symbol	Parameter	Min	Max	Units
V <sub>PP</sub>	Programming Supply Voltage	12.5	13.0	V
I <sub>PP</sub>	Programming Supply Current		50	mA
1/T <sub>CLCL</sub>	Oscillator Frequency	4	6	MHz
T <sub>AVGL</sub>	Address Setup to $\overline{\text{PROG}}$ Low	48T <sub>CLCL</sub>		
T <sub>GHAX</sub>	Address Hold After $\overline{\text{PROG}}$	48T <sub>CLCL</sub>		
T <sub>DVGL</sub>	Data Setup to $\overline{\text{PROG}}$ Low	48T <sub>CLCL</sub>		
T <sub>GHDX</sub>	Data Hold After $\overline{\text{PROG}}$	48T <sub>CLCL</sub>		
T <sub>EHS</sub>	P2.7 ( $\overline{\text{ENABLE}}$ ) High to V <sub>PP</sub>	48T <sub>CLCL</sub>		
T <sub>SHGL</sub>	V <sub>PP</sub> Setup to $\overline{\text{PROG}}$ Low	10		μs
T <sub>GHSL</sub>	V <sub>PP</sub> Hold After $\overline{\text{PROG}}$	10		μs
T <sub>GLGH</sub>	$\overline{\text{PROG}}$ Width	90	110	μs
T <sub>AVQV</sub>	Address to Data Valid		48T <sub>CLCL</sub>	
T <sub>ELQV</sub>	$\overline{\text{ENABLE}}$ Low to Data Valid		48T <sub>CLCL</sub>	
T <sub>EHQZ</sub>	Data Float After $\overline{\text{ENABLE}}$	0	48T <sub>CLCL</sub>	
T <sub>GHGL</sub>	$\overline{\text{PROG}}$ High to $\overline{\text{PROG}}$ Low	10		μs

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

**INTRODUCTION**

The 8051 is the original member of the MCS<sup>®</sup>-51 family, and is the core for all MCS-51 devices. The features of the 8051 core are:

- 8-bit CPU optimized for control applications
- Extensive Boolean processing (single-bit logic) capabilities
- 64K Program Memory address space
- 64K Data Memory address space
- 4K bytes of on-chip Program Memory
- 128 bytes of on-chip Data RAM
- 32 bidirectional and individually addressable I/O lines
- Two 16-bit timer/counters
- Full duplex UART
- 6-source/5-vector interrupt structure with two priority levels
- On-chip clock oscillator

The basic architectural structure of this 8051 core is shown in Figure 1.

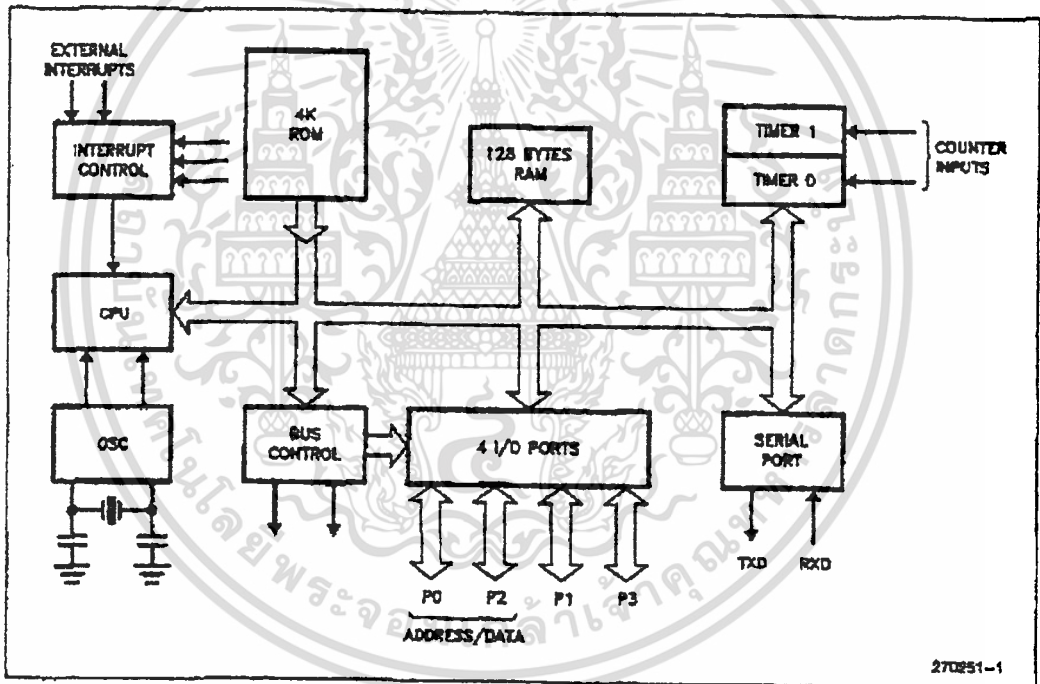


Figure 1. Block Diagram of the 8051 Core

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Table 1. The MCS-51 Family of Microcontrollers

DEVICE	ROM/EPROM (bytes)	Register RAM (bytes)	Speed (MHz)	IO Pins	IO Timers/Counters	UART	Interrupt Sources	PCA Channels	A/D Channels	S/P	QBC	DMA Channels	Look Dths	Power Down & Idle Modes
<b>8051 Product Line</b>														
8031AH	ROMLESS	128	12	32	2	1	6	0	0	0	0	0	0	-
8051AH	4K ROM	128	12	32	2	1	6	0	0	0	0	0	0	-
8051AP	4K ROM	128	12	32	2	1	5	0	0	0	0	0	P	-
8751H	4K EPROM	128	12	32	2	1	6	0	0	0	0	0	1	-
8751BH	4K EPROM	128	12	32	2	1	6	0	0	0	0	0	2	-
<b>8052 Product Line</b>														
8052AH	ROMLESS	256	12	32	3	1	8	0	0	0	0	0	0	-
8052AT	8K ROM	256	12	32	3	1	6	0	0	0	0	0	0	-
8752BH	8K EPROM	256	12	32	3	1	6	0	0	0	0	0	2	-
<b>80C51 Product Line</b>														
80C51BH	ROMLESS	128	12,16	32	2	1	5	0	0	0	0	0	0	Yes
80C51BH	4K ROM	128	12,16	32	2	1	5	0	0	0	0	0	0	Yes
80C51BHP	4K ROM	128	12,16	32	2	1	5	0	0	0	0	0	P	Yes
87C51	4K EPROM	128	12,16,20,24	32	2	1	5	0	0	0	0	0	3	Yes
<b>80C52-52 Product Line</b>														
80C52	ROMLESS	256	12,16,20,24	32	3	1	6	0	0	0	0	0	0	Yes
80C52	8K ROM	256	12,16,20,24	32	3	1	6	0	0	0	0	0	1*	Yes
87C52	8K EPROM	256	12,16,20,24	32	3	1	6	0	0	0	0	0	3	Yes
80C54	16K ROM	256	12,16,20,24	32	3	1	6	0	0	0	0	0	1	Yes
87C54	16K EPROM	256	12,16,20,24	32	3	1	6	0	0	0	0	0	3	Yes
80C58	32K ROM	256	12,16,20,24	32	3	1	6	0	0	0	0	0	1	Yes
87C58	32K EPROM	256	12,16,20,24	32	3	1	6	0	0	0	0	0	3	Yes
<b>80L52-58 Product Line</b>														
80L52	8K ROM	256	12,16,20*	32	3	1	6	0	0	0	0	0	1	Yes
87L52	8K OTP ROM	256	12,16,20*	32	3	1	6	0	0	0	0	0	3	Yes
80L54	16K ROM	256	12,16,20*	32	3	1	6	0	0	0	0	0	1	Yes
87L54	16K OTP ROM	256	12,16,20*	32	3	1	6	0	0	0	0	0	3	Yes
80L58	32K ROM	256	12,16,20*	32	3	1	6	0	0	0	0	0	1	Yes
87L58	32K OTP ROM	256	12,16,20*	32	3	1	6	0	0	0	0	0	3	Yes

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Table 1. The MCS-51 Family of Microcontrollers

DEVICE	ROM/EPROM (bytes)	RAM (bytes)	Speed (MHz)	I/O Pins	Timers/Counters	UART	Interrupt Sources	PCA Channels	A/D Channels	EEP	QBC	DMA Channels	Lock Bits	Power Down & Idle Modes
80C51FA/87C51FC Product Line														
80C51FA	256	256	12,16	32	3	1	7	5	0	0	0	0	0	Yes
80C51FA	ROMLESS	256	12,16	32	3	1	7	2	0	0	0	0	0	Yes
80C51FA	8K ROM	256	12,16	32	3	1	7	6	0	0	0	0	0	Yes
87C51FA	8K EPROM	256	12,16,20,24	32	3	1	7	6	0	0	0	0	0	Yes
89C51FB	16K ROM	256	12,16,20,24	32	3	1	7	6	0	0	0	0	0	Yes
87C51FB	16K EPROM	256	12,16,20,24	32	3	1	7	5	0	0	0	0	0	Yes
89C51FC	32K ROM	256	12,16,20,24	32	3	1	7	5	0	0	0	0	0	Yes
87C51FC	32K EPROM	256	12,16,20,24	32	3	1	7	5	0	0	0	0	0	Yes
80C510A/87C510C Product Line														
80C510A	ROMLESS	256	12,16,20*	32	3	1	7	5	0	0	0	0	0	Yes
80C510A	8K ROM	256	12,16,20*	32	3	1	7	6	0	0	0	0	0	Yes
87C510A	8K OTP ROM	256	12,16,20*	32	3	1	7	6	0	0	0	0	0	Yes
80C510B	16K ROM	256	12,16,20*	32	3	1	7	6	0	0	0	0	0	Yes
87C510B	16K OTP ROM	256	12,16,20*	32	3	1	7	6	0	0	0	0	0	Yes
80C510C	32K ROM	256	12,16,20*	32	3	1	7	6	0	0	0	0	0	Yes
87C510C	32K OTP ROM	256	12,16,20*	32	3	1	7	6	0	0	0	0	0	Yes
80C510X Product Line														
80C510X	ROMLESS	256	12,16	48	3	1	15	10	0	1	0	0	0	Yes
80C510X	8K ROM	256	12,16	48	3	1	15	10	0	1	0	0	0	Yes
87C510X	8K EPROM	256	12,16	48	3	1	15	10	0	1	0	0	0	Yes
80C512A Product Line														
80C512A	ROMLESS	256	18,6	40	2	1	11	0	0	1	1	2	0	Yes
80C512A	ROMLESS	256	18,6	56	2	1	11	0	0	1	1	2	0	Yes
80C512A	8K ROM	256	18,6	40	2	1	11	0	0	1	1	2	0	Yes
80C512L Product Line														
80C512L-BG	ROMLESS	256	16	24	2	1	10	0	4	0	1	0	0	Yes
80C512L-SG	8K ROM	256	16	24	2	1	10	0	4	0	1	0	0	Yes
80C512L-AG	8K ROM	256	16	24	2	1	10	0	4	0	1	0	0	Yes
80C512L-AH	ROMLESS	256	16	24	2	1	10	0	4	0	1	0	0	Yes
80C512L-AH	16K ROM	256	16	24	2	1	10	0	4	0	1	0	0	Yes
80C512L-AH	16K ROM	256	16	24	2	1	10	0	4	0	1	0	0	Yes
87C512L-AH	16K EPROM	256	16	24	2	1	10	0	4	0	1	0	0	Yes
80C518LAL	ROMLESS	256	16	24	2	1	10	0	4	0	1	0	0	Yes
81C518LAL	16K ROM	256	16	24	2	1	10	0	4	0	1	0	0	Yes
80C518LAL	16K ROM	256	16	24	2	1	10	0	4	0	1	0	0	Yes
87C518LAL	16K EPROM	256	16	24	2	1	10	0	4	0	1	0	0	Yes

\* ROM  
 24  
 20\*  
 1\*  
 P

ROM/OTP ROM/EPROM (bytes)  
 Speed (MHz)  
 Lock Bits  
 8XC12E Product Line\*  
 8XC512L Product Line\*

- SystemSoft Standard BIOS
- 24 MHz Internal-only operation
- 20MHz Available for Commercial Temperature Ranges Only
- 1 Lock Bit for 20MHz & 24MHz parts, no Lock Bit for 12 & 16MHz parts
- Program verification disabled, external memory access limited to 4K
- Communication Controller
- Keyboard Controller

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
 ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

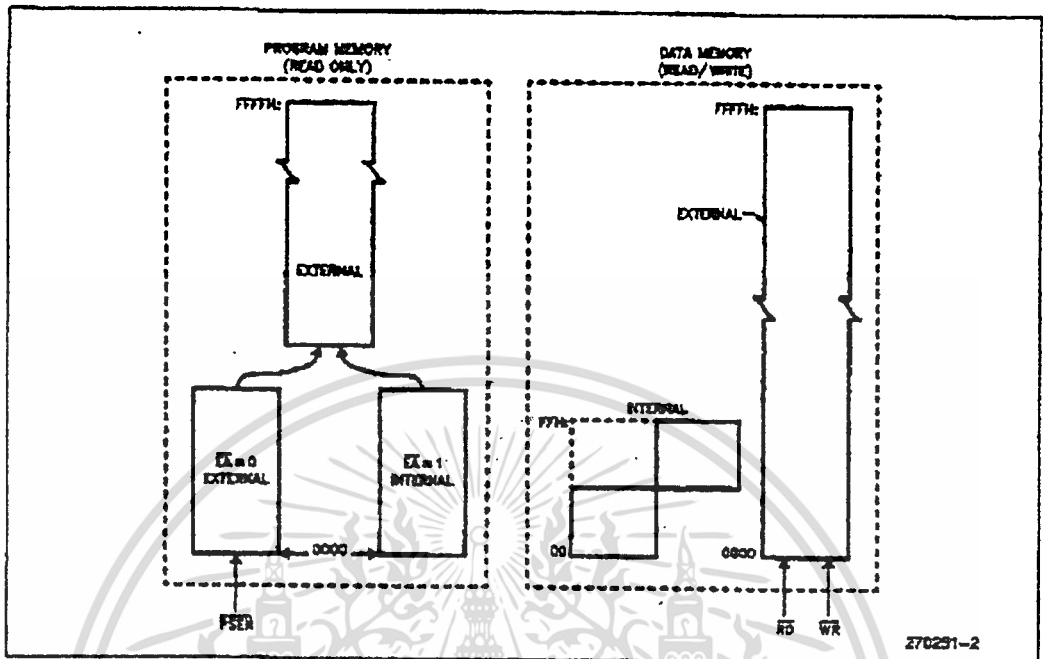


Figure 2. MCS<sup>®</sup>-51 Memory Structure

### CHMOS Devices

Functionally, the CHMOS devices (designated with "C" in the middle of the device name) are all fully compatible with the 8051, but being CMOS, draw less current than an HMOS counterpart. To further exploit the power savings available in CMOS circuitry, two reduced power modes are added:

- Software-invoked Idle Mode, during which the CPU is turned off while the RAM and other on-chip peripherals continue operating. In this mode, current draw is reduced to about 15% of the current drawn when the device is fully active.
- Software-invoked Power Down Mode, during which all on-chip activities are suspended. The on-chip RAM continues to hold its data. In this mode the device typically draws less than 10  $\mu$ A.

Although the 80C51BH is functionally compatible with its HMOS counterpart, specific differences between the two types of devices must be considered in the design of an application circuit if one wishes to ensure complete interchangeability between the HMOS and CHMOS devices. These considerations are discussed in the Application Note AP-252, "Designing with the 80C51BH".

For more information on the individual devices and features listed in Table 1, refer to the Hardware Descriptions and Data Sheets of the specific device.

### MEMORY ORGANIZATION IN MCS<sup>®</sup>-51 DEVICES

#### Logical Separation of Program and Data Memory

All MCS-51 devices have separate address spaces for Program and Data Memory, as shown in Figure 2. The logical separation of Program and Data Memory allows the Data Memory to be accessed by 8-bit addresses, which can be more quickly stored and manipulated by an 8-bit CPU. Nevertheless, 16-bit Data Memory addresses can also be generated through the DPTR register.

Program Memory can only be read, not written to. There can be up to 64K bytes of Program Memory. In the ROM and EPROM versions of these devices the lowest 4K, 8K or 16K bytes of Program Memory are provided on-chip. Refer to Table 1 for the amount of on-chip ROM (or EPROM) on each device. In the ROMless versions all Program Memory is external. The read strobe for external Program Memory is the signal PSEN (Program Store Enable).

Data Memory occupies a separate address space from Program Memory. Up to 64K bytes of external RAM can be addressed in the external Data Memory space. The CPU generates read and write signals, RD and WR, as needed during external Data Memory accesses.

External Program Memory and external Data Memory may be combined if desired by applying the RD and PSEN signals to the inputs of an AND gate and using the output of the gate as the read strobe to the external Program/Data memory.

### Program Memory

Figure 3 shows a map of the lower part of the Program Memory. After reset, the CPU begins execution from location 0000H.

As shown in Figure 3, each interrupt is assigned a fixed location in Program Memory. The interrupt causes the CPU to jump to that location, where it commences execution of the service routine. External Interrupt 0, for example, is assigned to location 0003H. If External Interrupt 0 is going to be used, its service routine must begin at location 0003H. If the interrupt is not going to be used, its service location is available as general purpose Program Memory.

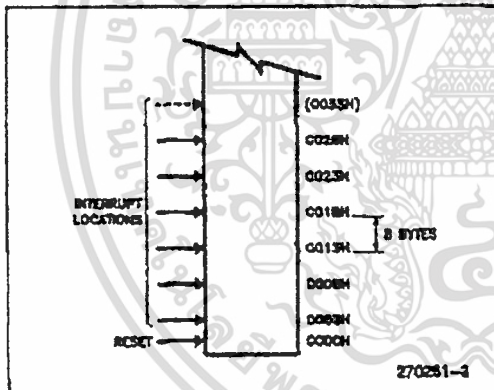


Figure 3. MCS<sup>®</sup>-51 Program Memory

The interrupt service locations are spaced at 8-byte intervals: 0003H for External Interrupt 0, 000BH for Timer 0, 0013H for External Interrupt 1, 001BH for Timer 1, etc. If an interrupt service routine is short enough (as is often the case in control applications), it can reside entirely within that 8-byte interval. Longer service routines can use a jump instruction to skip over subsequent interrupt locations, if other interrupts are in use.

The lowest 4K (or 8K or 16K) bytes of Program Memory can be either in the on-chip ROM or in an external ROM. This selection is made by strapping the EA (External Access) pin to either V<sub>CC</sub> or V<sub>SS</sub>.

In the 4K byte ROM devices, if the EA pin is strapped to V<sub>CC</sub> then program fetches to addresses 0000H through 0FFFH are directed to the internal ROM. Program fetches to addresses 1000H through FFFFH are directed to external ROM.

In the 8K byte ROM devices, EA = V<sub>CC</sub> selects addresses 0000H through 1FFFH to be internal, and addresses 2000H through FFFFH to be external.

In the 16K byte ROM devices, EA = V<sub>CC</sub> selects addresses 0000H through 3FFFH to be internal, and addresses 4000H through FFFFH to be external.

If the EA pin is strapped to V<sub>SS</sub>, then all program fetches are directed to external ROM. The ROMless parts must have this pin externally strapped to V<sub>SS</sub> to enable them to execute properly.

The read strobe to external ROM, PSEN, is used for all external program fetches. PSEN is not activated for internal program fetches.

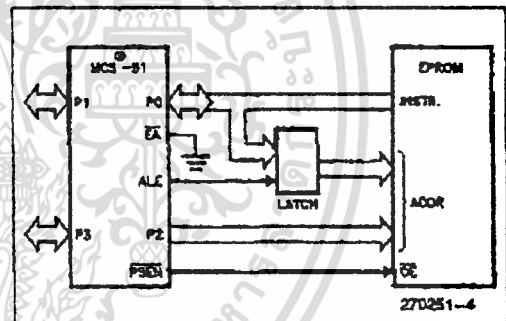


Figure 4. Executing from External Program Memory

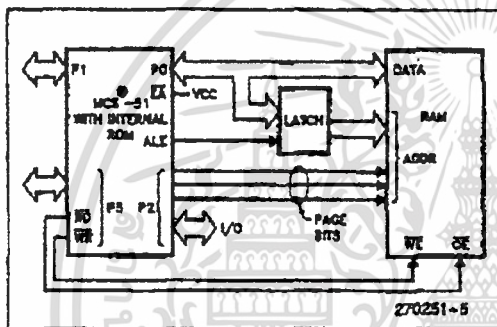
The hardware configuration for external program execution is shown in Figure 4. Note that 16 I/O lines (Ports 0 and 2) are dedicated to bus functions during external Program Memory fetches. Port 0 (P0 in Figure 4) serves as a multiplexed address/data bus. It emits the low byte of the Program Counter (PCL) as an address, and then goes into a float state awaiting the arrival of the code byte from the Program Memory. During the time that the low byte of the Program Counter is valid on P0, the signal ALE (Address Latch Enable) clocks this byte into an address latch. Meanwhile, Port 2 (P2 in Figure 4) emits the high byte of the Program Counter (PCH). Then PSEN strobes the EPROM and the code byte is read into the microcontroller.

Program Memory addresses are always 16 bits wide, even though the actual amount of Program Memory used may be less than 64K bytes. External program execution sacrifices two of the 8-bit ports, P0 and P2, to the function of addressing the Program Memory.

**Data Memory**

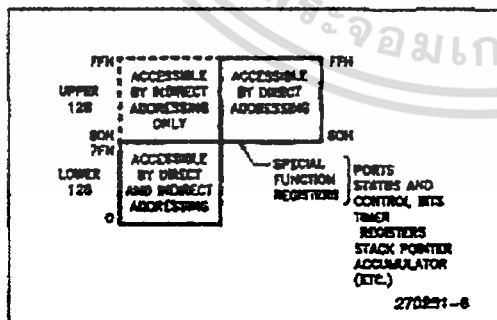
The right half of Figure 2 shows the internal and external Data Memory spaces available to the MCS-51 user.

Figure 5 shows a hardware configuration for accessing up to 2K bytes of external RAM. The CPU in this case is executing from internal ROM. Port 0 serves as a multiplexed address/data bus to the RAM, and 3 lines of Port 2 are being used to page the RAM. The CPU generates RD and WR signals as needed during external RAM accesses.



**Figure 5. Accessing External Data Memory.**  
If the Program Memory is internal, the Other Bits of P2 are Available as I/O.

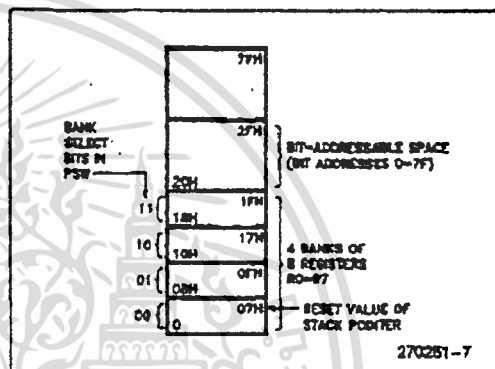
There can be up to 64K bytes of external Data Memory. External Data Memory addresses can be either 1 or 2 bytes wide. One-byte addresses are often used in conjunction with one or more other I/O lines to page the RAM, as shown in Figure 5. Two-byte addresses can also be used, in which case the high address byte is emitted at Port 2.



**Figure 6. Internal Data Memory**

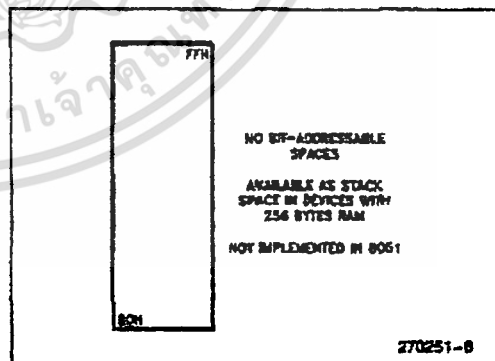
Internal Data Memory is mapped in Figure 6. The memory space is shown divided into three blocks, which are generally referred to as the Lower 128, the Upper 128, and SFR space.

Internal Data Memory addresses are always one byte wide, which implies an address space of only 256 bytes. However, the addressing modes for internal RAM can in fact accommodate 384 bytes, using a simple trick. Direct addresses higher than 7FH access one memory space, and indirect addresses higher than 7FH access a different memory space. Thus Figure 6 shows the Upper 128 and SFR space occupying the same block of addresses, 80H through FFH, although they are physically separate entities.



**Figure 7. The Lower 128 Bytes of Internal RAM**

The Lower 128 bytes of RAM are present in all MCS-51 devices as mapped in Figure 7. The lowest 32 bytes are grouped into 4 banks of 8 registers. Program instructions call out these registers as R0 through R7. Two bits in the Program Status Word (PSW) select which register bank is in use. This allows more efficient use of code space, since register instructions are shorter than instructions that use direct addressing.



**Figure 8. The Upper 128 Bytes of Internal RAM**

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่นอนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

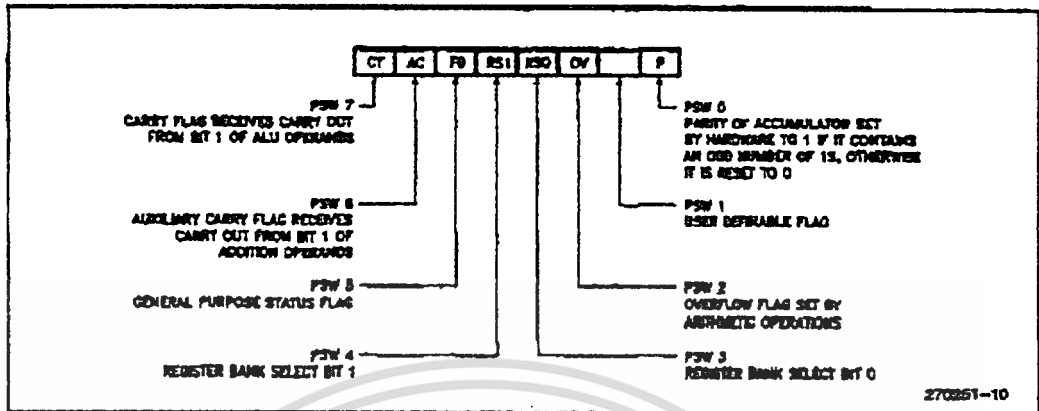


Figure 10. PSW (Program Status Word) Register in MCS<sup>®</sup>-51 Devices

The next 16 bytes above the register banks form a block of bit-addressable memory space. The MCS-51 instruction set includes a wide selection of single-bit instructions, and the 128 bits in this area can be directly addressed by these instructions. The bit addresses in this area are 00H through 7FH.

All of the bytes in the Lower 128 can be accessed by either direct or indirect addressing. The Upper 128 (Figure 8) can only be accessed by indirect addressing. The Upper 128 bytes of RAM are not implemented in the 8051, but are in the devices with 256 bytes of RAM. (See Table 1).

Figure 9 gives a brief look at the Special Function Register (SFR) space. SFRs include the Port latches, timers, peripheral controls, etc. These registers can only be accessed by direct addressing. In general, all MCS-51 microcontrollers have the same SFRs as the 8051, and at the same addresses in SFR space. However, enhancements to the 8051 have additional SFRs that are not present in the 8051, nor perhaps in other proliferations of the family.

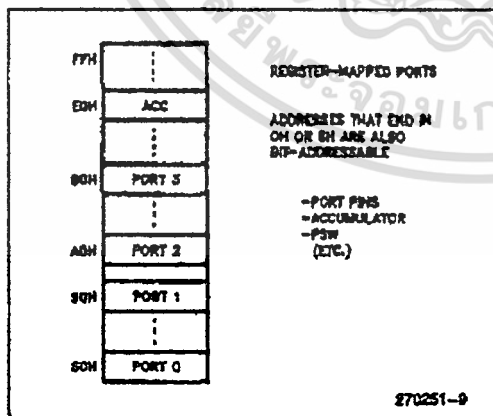


Figure 9. SFR Space

Sixteen addresses in SFR space are both byte- and bit-addressable. The bit-addressable SFRs are those whose address ends in 00BH. The bit addresses in this area are 80H through FFH.

### THE MCS<sup>®</sup>-51 INSTRUCTION SET

All members of the MCS-51 family execute the same instruction set. The MCS-51 instruction set is optimized for 8-bit control applications. It provides a variety of fast addressing modes for accessing the internal RAM to facilitate byte operations on small data structures. The instruction set provides extensive support for one-bit variables as a separate data type, allowing direct bit manipulation in control and logic systems that require Boolean processing.

An overview of the MCS-51 instruction set is presented below, with a brief description of how certain instructions might be used. References to "the assembler" in this discussion are to Intel's MCS-51 Macro Assembler, ASM51. More detailed information on the instruction set can be found in the MCS-51 Macro Assembler User's Guide (Order No. 9800937 for ISIS Systems, Order No. 122752 for DOS Systems).

### Program Status Word

The Program Status Word (PSW) contains several status bits that reflect the current state of the CPU. The PSW, shown in Figure 10, resides in SFR space. It contains the Carry bit, the Auxiliary Carry (for BCD operations), the two register bank select bits, the Overflow flag, a Parity bit, and two user-definable status flags.

The Carry bit, other than serving the functions of a Carry bit in arithmetic operations, also serves as the "Accumulator" for a number of Boolean operations.

The bits RS0 and RS1 are used to select one of the four register banks shown in Figure 7. A number of instructions refer to these RAM locations as R0 through R7. The selection of which of the four banks is being referred to is made on the basis of the bits RS0 and RS1 at execution time.

The Parity bit reflects the number of 1s in the Accumulator:  $P = 1$  if the Accumulator contains an odd number of 1s, and  $P = 0$  if the Accumulator contains an even number of 1s. Thus the number of 1s in the Accumulator plus  $P$  is always even.

Two bits in the PSW are uncommitted and may be used as general purpose status flags.

## Addressing Modes

The addressing modes in the MCS-51 instruction set are as follows:

### DIRECT ADDRESSING

In direct addressing the operand is specified by an 8-bit address field in the instruction. Only internal Data RAM and SFRs can be directly addressed.

### INDIRECT ADDRESSING

In indirect addressing the instruction specifies a register which contains the address of the operand. Both internal and external RAM can be indirectly addressed.

The address register for 8-bit addresses can be R0 or R1 of the selected register bank, or the Stack Pointer. The address register for 16-bit addresses can only be the 16-bit "data pointer" register, DPTR.

### REGISTER INSTRUCTIONS

The register banks, containing registers R0 through R7, can be accessed by certain instructions which carry a 3-bit register specification within the opcode of the instruction. Instructions that access the registers this way are code efficient, since this mode eliminates an address byte. When the instruction is executed, one of the eight registers in the selected bank is accessed. One of four banks is selected at execution time by the two bank select bits in the PSW.

### REGISTER-SPECIFIC INSTRUCTIONS

Some instructions are specific to a certain register. For example, some instructions always operate on the Accumulator, or Data Pointer, etc., so no address byte is needed to point to it. The opcode itself does that. Instructions that refer to the Accumulator as A assemble as accumulator-specific opcodes.

### IMMEDIATE CONSTANTS

The value of a constant can follow the opcode in Program Memory. For example,

```
MOV A, #100
```

loads the Accumulator with the decimal number 100. The same number could be specified in hex digits as 64H.

### INDEXED ADDRESSING

Only Program Memory can be accessed with indexed addressing, and it can only be read. This addressing mode is intended for reading look-up tables in Program Memory. A 16-bit base register (either DPTR or the Program Counter) points to the base of the table, and the Accumulator is set up with the table entry number. The address of the table entry in Program Memory is formed by adding the Accumulator data to the base pointer.

Another type of indexed addressing is used in the "case jump" instruction. In this case the destination address of a jump instruction is computed as the sum of the base pointer and the Accumulator data.

### Arithmetic Instructions

The menu of arithmetic instructions is listed in Table 2. The table indicates the addressing modes that can be used with each instruction to access the <byte> operand. For example, the ADD A, <byte> instruction can be written as:

```
ADD A,7FH      (direct addressing)
ADD A,@R0     (indirect addressing)
ADD A,R7      (register addressing)
ADD A,#127    (immediate constant)
```

The execution times listed in Table 2 assume a 12 MHz clock frequency. All of the arithmetic instructions execute in 1  $\mu$ s except the INC DPTR instruction, which takes 2  $\mu$ s, and the Multiply and Divide instructions, which take 4  $\mu$ s.

Note that any byte in the internal Data Memory space can be incremented or decremented without going through the Accumulator.

One of the INC instructions operates on the 16-bit Data Pointer. The Data Pointer is used to generate 16-bit addresses for external memory, so being able to increment it in one 16-bit operation is a useful feature.

The MUL AB instruction multiplies the Accumulator by the data in the B register and puts the 16-bit product into the concatenated B and Accumulator registers.

**Table 2. A List of the MCS<sup>®</sup>-51 Arithmetic Instructions**

Mnemonic	Operation	Addressing Modes				Execution Time (μs)
		Dir	Ind	Reg	Imm	
ADD A, <byte>	A = A + <byte>	X	X	X	X	1
ADDC A, <byte>	A = A + <byte> + C	X	X	X	X	1
SUBB A, <byte>	A = A - <byte> - C	X	X	X	X	1
INC A	A = A + 1	Accumulator only				1
INC <byte>	<byte> = <byte> + 1	X	X	X		1
INC DPTR	DPTR = DPTR + 1	Data Pointer only				2
DEC A	A = A - 1	Accumulator only				1
DEC <byte>	<byte> = <byte> - 1	X	X	X		1
MUL AB	B:A = B x A	ACC and B only				4
DIV AB	A = Int [A/B] B = Mod [A/B]	ACC and B only				4
DA A	Decimal Adjust	Accumulator only				1

The DIV AB instruction divides the Accumulator by the data in the B register and leaves the 8-bit quotient in the Accumulator, and the 8-bit remainder in the B register.

Oddly enough, DIV AB finds less use in arithmetic "divide" routines than in radix conversions and programmable shift operations. An example of the use of DIV AB in a radix conversion will be given later. In shift operations, dividing a number by 2<sup>n</sup> shifts its n bits to the right. Using DIV AB to perform the division

completes the shift in 4 μs and leaves the B register holding the bits that were shifted out.

The DA A instruction is for BCD arithmetic operations. In BCD arithmetic, ADD and ADDC instructions should always be followed by a DA A operation, to ensure that the result is also in BCD. Note that DA A will not convert a binary number to BCD. The DA A operation produces a meaningful result only as the second step in the addition of two BCD bytes.

**Table 3. A List of the MCS<sup>®</sup>-51 Logical Instructions**

Mnemonic	Operation	Addressing Modes				Execution Time (μs)
		Dir	Ind	Reg	Imm	
ANL A, <byte>	A = A .AND. <byte>	X	X	X	X	1
ANL <byte>, A	<byte> = <byte> .AND. A	X				1
ANL <byte>, #data	<byte> = <byte> .AND. #data	X				2
ORL A, <byte>	A = A .OR. <byte>	X	X	X	X	1
ORL <byte>, A	<byte> = <byte> .OR. A	X				1
ORL <byte>, #data	<byte> = <byte> .OR. #data	X				2
XRL A, <byte>	A = A .XOR. <byte>	X	X	X	X	1
XRL <byte>, A	<byte> = <byte> .XOR. A	X				1
XRL <byte>, #data	<byte> = <byte> .XOR. #data	X				2
CPL A	A = 00H	Accumulator only				1
CPL A	A = .NOT. A	Accumulator only				1
RL A	Rotate ACC Left 1 bit	Accumulator only				1
RLC A	Rotate Left through Carry	Accumulator only				1
RR A	Rotate ACC Right 1 bit	Accumulator only				1
RRC A	Rotate Right through Carry	Accumulator only				1
SWAP A	Swap Nibbles in A	Accumulator only				1

## Logical Instructions

Table 3 shows the list of MCS-51 logical instructions. The instructions that perform Boolean operations (AND, OR, Exclusive OR, NOT) on bytes perform the operation on a bit-by-bit basis. That is, if the Accumulator contains 00110101B and <byte> contains 01010011B, then

```
ANL  A, <byte>
```

will leave the Accumulator holding 0010001B.

The addressing modes that can be used to access the <byte> operand are listed in Table 3. Thus, the ANL A, <byte> instruction may take any of the forms

```
ANL  A, 7FH      (direct addressing)
ANL  A, @R1     (indirect addressing)
ANL  A, R6      (register addressing)
ANL  A, #53H    (immediate constant)
```

All of the logical instructions that are Accumulator-specific execute in 1  $\mu$ s (using a 12 MHz clock). The others take 2  $\mu$ s.

Note that Boolean operations can be performed on any byte in the lower 128 internal Data Memory space or the SFR space using direct addressing, without having to use the Accumulator. The XRL <byte>, #data instruction, for example, offers a quick and easy way to invert port bits, as in

```
XRL  P1, #0FFH
```

If the operation is in response to an interrupt, not using the Accumulator saves the time and effort to stack it in the service routine.

The Rotate instructions (RL A, RLC A, etc.) shift the Accumulator 1 bit to the left or right. For a left rotation, the MSB rolls into the LSB position. For a right rotation, the LSB rolls into the MSB position.

The SWAP A instruction interchanges the high and low nibbles within the Accumulator. This is a useful operation in BCD manipulations. For example, if the Accumulator contains a binary number which is known to be less than 100, it can be quickly converted to BCD by the following code:

```
MOV  B, #10
DIV  AB
SWAP A
ADD  A, B
```

Dividing the number by 10 leaves the tens digit in the low nibble of the Accumulator, and the ones digit in the B register. The SWAP and ADD instructions move the tens digit to the high nibble of the Accumulator, and the ones digit to the low nibble.

## Data Transfers

### INTERNAL RAM

Table 4 shows the menu of instructions that are available for moving data around within the internal memory spaces, and the addressing modes that can be used with each one. With a 12 MHz clock, all of these instructions execute in either 1 or 2  $\mu$ s.

The MOV <dest>, <src> instruction allows data to be transferred between any two internal RAM or SFR locations without going through the Accumulator. Remember the Upper 128 bytes of data RAM can be accessed only by indirect addressing, and SFR space only by direct addressing.

Note that in all MCS-51 devices, the stack resides in on-chip RAM, and grows upwards. The PUSH instruction first increments the Stack Pointer (SP), then copies the byte into the stack. PUSH and POP use only direct addressing to identify the byte being saved or restored.

Table 4. A List of the MCS-51 Data Transfer Instructions that Access Internal Data Memory Space

Mnemonic	Operation	Addressing Modes				Execution Time ( $\mu$ s)
		Dir	Ind	Reg	Imm	
MOV A, <src>	A = <src>	X	X	X	X	1
MOV <dest>, A	<dest> = A	X	X	X		1
MOV <dest>, <src>	<dest> = <src>	X	X	X	X	2
MOV DPTR, #data16	DPTR = 16-bit immediate constant.				X	2
PUSH <src>	INC SP; MOV "@SP", <src>	X				2
POP <dest>	MOV <dest>, "@SP"; DEC SP	X				2
XCH A, <byte>	ACC and <byte> exchange data.	X	X	X		1
XCHD A, @Ri	ACC and @Ri exchange low nibbles		X			1

but the stack itself is accessed by indirect addressing using the SP register. This means the stack can go into the Upper 128, if they are implemented, but not into SFR space.

In devices that do not implement the Upper 128, if the SP points to the Upper 128, PUSHed bytes are lost, and POPped bytes are indeterminate.

The Data Transfer instructions include a 16-bit MOV that can be used to initialize the Data Pointer (DPTR) for look-up tables in Program Memory, or for 16-bit external Data Memory accesses.

The XCH A, <byte> instruction causes the Accumulator and addressed byte to exchange data. The XCHD A,@Ri instruction is similar, but only the low nibbles are involved in the exchange.

To see how XCH and XCHD can be used to facilitate data manipulations, consider first the problem of shifting an 8-digit BCD number two digits to the right. Figure 11 shows how this can be done using direct MOVs, and for comparison how it can be done using XCH instructions. To aid in understanding how the code works, the contents of the registers that are holding the BCD number and the content of the Accumulator are shown alongside each instruction to indicate their status after the instruction has been executed.

		2A	2B	2C	2D	2E	ACC
MOV	A,2EH	00	12	34	56	78	78
MOV	2EH,2DH	00	12	34	56	56	78
MOV	2DH,2CH	00	12	34	34	56	78
MOV	2CH,2BH	00	12	12	34	56	78
MOV	2BH,#0	00	00	12	34	56	78
(a) Using direct MOVs: 14 bytes, 9 $\mu$ s							
		2A	2B	2C	2D	2E	ACC
CLR	A	00	12	34	56	78	00
XCH	A,2BH	00	00	34	56	78	12
XCH	A,2CH	00	00	12	56	78	34
XCH	A,2DH	00	00	12	34	78	56
XCH	A,2EH	00	00	12	34	56	78
(b) Using XCHs: 9 bytes, 5 $\mu$ s							

Figure 11. Shifting a BCD Number Two Digits to the Right

After the routine has been executed, the Accumulator contains the two digits that were shifted out on the right. Doing the routine with direct MOVs uses 14 code bytes and 9  $\mu$ s of execution time (assuming a 12 MHz clock). The same operation with XCHs uses less code and executes almost twice as fast.

To right-shift by an odd number of digits, a one-digit shift must be executed. Figure 12 shows a sample of code that will right-shift a BCD number one digit, using the XCHD instruction. Again, the contents of the registers holding the number and of the Accumulator are shown alongside each instruction.

		2A	2B	2C	2D	2E	ACC
MOV	R1,#2EH	00	12	34	56	78	XX
MOV	R0,#2DH	00	12	34	56	78	XX
loop for R1 = 2EH:							
LOOP:	MOV A,@R1	00	12	34	56	78	78
	XCHD A,@R0	00	12	34	56	78	78
	SWAP A	00	12	34	56	78	67
	MOV @R1,A	00	12	34	56	67	67
	DEC R1	00	12	34	56	67	67
	DEC R0	00	12	34	56	67	67
	CJNE R1,#2AH,LOOP						
loop for R1 = 2DH:							
		00	12	38	45	67	45
loop for R1 = 2CH:							
		00	18	23	45	67	23
loop for R1 = 2BH:							
		08	01	23	45	67	01
CLR	A	08	01	23	45	67	00
XCH	A,2AH	00	01	23	45	67	08

Figure 12. Shifting a BCD Number One Digit to the Right

First, pointers R1 and R0 are set up to point to the two bytes containing the last four BCD digits. Then a loop is executed which leaves the last byte, location 2EH, holding the last two digits of the shifted number. The pointers are decremented, and the loop is repeated for location 2DH. The CJNE instruction (Compare and Jump if Not Equal) is a loop control that will be described later.

The loop is executed from LOOP to CJNE for R1 = 2EH, 2DH, 2CH and 2BH. At that point the digit that was originally shifted out on the right has propagated to location 2AH. Since that location should be left with 0s, the lost digit is moved to the Accumulator.

**EXTERNAL RAM**

Table 5 shows a list of the Data Transfer instructions that access external Data Memory. Only indirect addressing can be used. The choice is whether to use a one-byte address, @RI, where RI can be either R0 or R1 of the selected register bank, or a two-byte address, @DPTR. The disadvantage to using 16-bit addresses is that 16-bit addresses use all 8 bits of Port 2 as address bus. On the other hand, 8-bit addresses allow one to address a few K bytes of RAM, as shown in Figure 5, without having to sacrifice all of Port 2.

All of these instructions execute in 2  $\mu$ s, with a 12 MHz clock.

**Table 5. A List of the MCS-51 Data Transfer Instructions that Access External Data Memory Space**

Address Width	Mnemonic	Operation	Execution Time ( $\mu$ s)
8 bits	MOVX A,@RI	Read external RAM @RI	2
8 bits	MOVX @RI,A	Write external RAM @RI	2
16 bits	MOVX A,@DPTR	Read external RAM @DPTR	2
16 bits	MOVX @DPTR,A	Write external RAM @DPTR	2

Note that in all external Data RAM accesses, the Accumulator is always either the destination or source of the data.

The read and write strobes to external RAM are activated only during the execution of a MOVX instruction. Normally these signals are inactive, and in fact if they're not going to be used at all, their pins are available as extra I/O lines. More about that later.

**LOOKUP TABLES**

Table 6 shows the two instructions that are available for reading lookup tables in Program Memory. Since these instructions access only Program Memory, the lookup tables can only be read, not updated. The mnemonic is MOVC for "move constant".

*(This table is not shown in the original document)*

**Table 6. The MCS-51 Lookup Table Read Instructions**

Mnemonic	Operation	Execution Time ( $\mu$ s)
MOVC A,@A+DPTR	Read Pgm Memory at (A+DPTR)	2
MOVC A,@A+PC	Read Pgm Memory at (A+PC)	2

The first MOVC instruction in Table 6 can accommodate a table of up to 256 entries, numbered 0 through 255. The number of the desired entry is loaded into the Accumulator, and the Data Pointer is set up to point to beginning of the table. Then

```
MOVC A,@A+DPTR
```

copies the desired table entry into the Accumulator.

The other MOVC instruction works the same way, except the Program Counter (PC) is used as the table base, and the table is accessed through a subroutine. First the number of the desired entry is loaded into the Accumulator, and the subroutine is called:

```
MOV A,ENTRY_NUMBER
CALL TABLE
```

The subroutine "TABLE" would look like this:

```
TABLE: MOVC A,@A+PC
      RET
```

The table itself immediately follows the RET (return) instruction in Program Memory. This type of table can have up to 255 entries, numbered 1 through 255. Number 0 can not be used, because at the time the MOVC instruction is executed, the PC contains the address of the RET instruction. An entry numbered 0 would be the RET opcode itself.

**Boolean Instructions**

MCS-51 devices contain a complete Boolean (single-bit) processor. The internal RAM contains 128 addressable bits, and the SFR space can contain up to 192 other addressable bits. All of the port lines are bit-addressable, and each one can be treated as a separate single-bit processor. The instructions that access these bits are not only logical operations, but a complete menu of operations, including complement, OR, and AND instructions. These kinds of bit operations are not available in other architectures with any amount of byte-oriented software.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

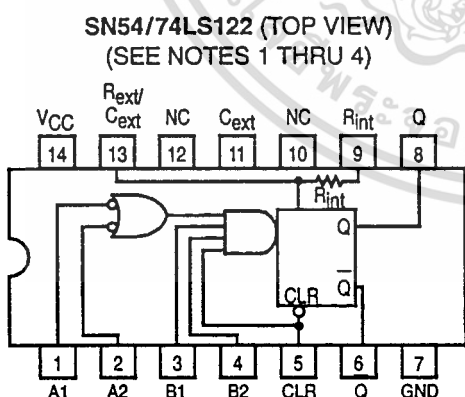
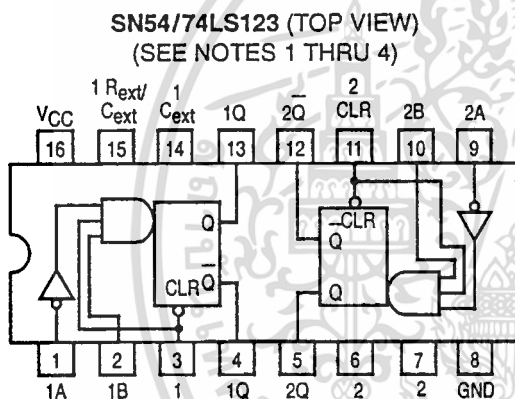


MOTOROLA

# RETRIGGERABLE MONOSTABLE MULTIVIBRATORS

These dc triggered multivibrators feature pulse width control by three methods. The basic pulse width is programmed by selection of external resistance and capacitance values. The LS122 has an internal timing resistor that allows the circuits to be used with only an external capacitor. Once triggered, the basic pulse width may be extended by retriggering the gated low-level-active (A) or high-level-active (B) inputs, or be reduced by use of the overriding clear.

- Overriding Clear Terminates Output Pulse
- Compensated for  $V_{CC}$  and Temperature Variations
- DC Triggered from Active-High or Active-Low Gated Logic Inputs
- Retriggerable for Very Long Output Pulses, up to 100% Duty Cycle
- Internal Timing Resistors on LS122



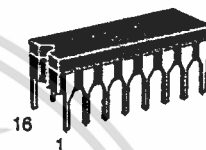
NC — NO INTERNAL CONNECTION.

### NOTES:

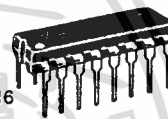
1. An external timing capacitor may be connected between  $C_{ext}$  and  $R_{ext}/C_{ext}$  (positive).
2. To use the internal timing resistor of the LS122, connect  $R_{int}$  to  $V_{CC}$ .
3. For improved pulse width accuracy connect an external resistor between  $R_{ext}/C_{ext}$  and  $V_{CC}$  with  $R_{int}$  open-circuited.
4. To obtain variable pulse widths, connect an external variable resistance between  $R_{int}/C_{ext}$  and  $V_{CC}$ .

## SN54/74LS122 SN54/74LS123

### RETRIGGERABLE MONOSTABLE MULTIVIBRATORS LOW POWER SCHOTTKY



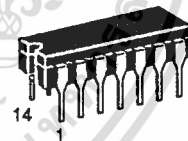
J SUFFIX  
CERAMIC  
CASE 620-09



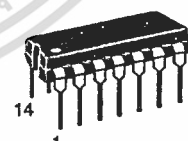
N SUFFIX  
PLASTIC  
CASE 648-08



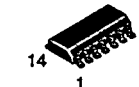
D SUFFIX  
SOIC  
CASE 751B-03



J SUFFIX  
CERAMIC  
CASE 632-08



N SUFFIX  
PLASTIC  
CASE 646-06



D SUFFIX  
SOIC  
CASE 751A-02

### ORDERING INFORMATION

SN54LSXXXJ	Ceramic
SN74LSXXXN	Plastic
SN74LSXXXD	SOIC

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

# SN54/74LS122 • SN54/74LS123

LS122  
FUNCTIONAL TABLE

CLEAR	INPUTS				OUTPUTS	
	A1	A2	B1	B2	Q	Q
L	X	X	X	X	L	H
X	H	H	X	X	L	H
X	X	X	L	X	L	H
X	X	X	X	L	L	H
H	L	X	↑	H	⌋	⌋
H	L	X	H	↑	⌋	⌋
H	X	L	↑	H	⌋	⌋
H	X	L	H	↑	⌋	⌋
H	H	↓	H	H	⌋	⌋
H	↓	↓	H	H	⌋	⌋
H	↓	H	H	H	⌋	⌋
↑	L	X	H	H	⌋	⌋
↑	X	L	H	H	⌋	⌋

LS123  
FUNCTIONAL TABLE

CLEAR	INPUTS		OUTPUTS	
	A	B	Q	Q
L	X	X	L	H
X	H	X	L	H
X	X	L	L	H
H	L	↑	⌋	⌋
H	↓	H	⌋	⌋
↑	L	H	⌋	⌋

## TYPICAL APPLICATION DATA

The output pulse  $t_W$  is a function of the external components,  $C_{ext}$  and  $R_{ext}$  or  $C_{ext}$  and  $R_{int}$  on the LS122. For values of  $C_{ext} \geq 1000$  pF, the output pulse at  $V_{CC} = 5.0$  V and  $V_{RC} = 5.0$  V (see Figures 1, 2, and 3) is given by

$$t_W = K R_{ext} C_{ext} \text{ where } K \text{ is nominally } 0.45$$

If  $C_{ext}$  is in pF and  $R_{ext}$  is in k $\Omega$  then  $t_W$  is in nanoseconds.

The  $C_{ext}$  terminal of the LS122 and LS123 is an internal connection to ground, however for the best system performance  $C_{ext}$  should be hard-wired to ground.

Care should be taken to keep  $R_{ext}$  and  $C_{ext}$  as close to the monostable as possible with a minimum amount of inductance between the  $R_{ext}/C_{ext}$  junction and the  $R_{ext}/C_{ext}$  pin. Good groundplane and adequate bypassing should be designed into the system for optimum performance to insure that no false triggering occurs.

It should be noted that the  $C_{ext}$  pin is internally connected to ground on the LS122 and LS123, but not on the LS221. Therefore, if  $C_{ext}$  is hard-wired externally to ground, substitution of a LS221 onto a LS123 socket will cause the LS221 to become non-functional.

The switching diode is not needed for electrolytic capacitance application and should not be used on the LS122 and LS123.

To find the value of K for  $C_{ext} \geq 1000$  pF, refer to Figure 4. Variations on  $V_{CC}$  or  $V_{RC}$  can cause the value of K to change, as can the temperature of the LS123, LS122. Figures 5 and 6 show the behavior of the circuit shown in Figures 1 and 2 if

separate power supplies are used for  $V_{CC}$  and  $V_{RC}$ . If  $V_{CC}$  is tied to  $V_{RC}$ , Figure 7 shows how K will vary with  $V_{CC}$  and temperature. Remember, the changes in  $R_{ext}$  and  $C_{ext}$  with temperature are not calculated and included in the graph.

As long as  $C_{ext} \geq 1000$  pF and  $5K \leq R_{ext} \leq 260K$  (SN74LS122/123) or  $5K \leq R_{ext} \leq 160K$  (SN54LS122/123), the change in K with respect to  $R_{ext}$  is negligible.

If  $C_{ext} \leq 1000$  pF the graph shown on Figure 8 can be used to determine the output pulse width. Figure 9 shows how K will change for  $C_{ext} \leq 1000$  pF if  $V_{CC}$  and  $V_{RC}$  are connected to the same power supply. The pulse width  $t_W$  in nanoseconds is approximated by

$$t_W = 6 + 0.05 C_{ext} (\text{pF}) + 0.45 R_{ext} (\text{k}\Omega) C_{ext} + 11.6 R_{ext}$$

In order to trim the output pulse width, it is necessary to include a variable resistor between  $V_{CC}$  and the  $R_{ext}/C_{ext}$  pin or between  $V_{CC}$  and the  $R_{ext}$  pin of the LS122. Figure 10, 11, and 12 show how this can be done.  $R_{ext}$  remote should be kept as close to the monostable as possible.

Retriggering of the part, as shown in Figure 3, must not occur before  $C_{ext}$  is discharged or the retrigger pulse will not have any effect. The discharge time of  $C_{ext}$  in nanoseconds is guaranteed to be less than  $0.22 C_{ext}$  (pF) and is typically  $0.05 C_{ext}$  (pF).

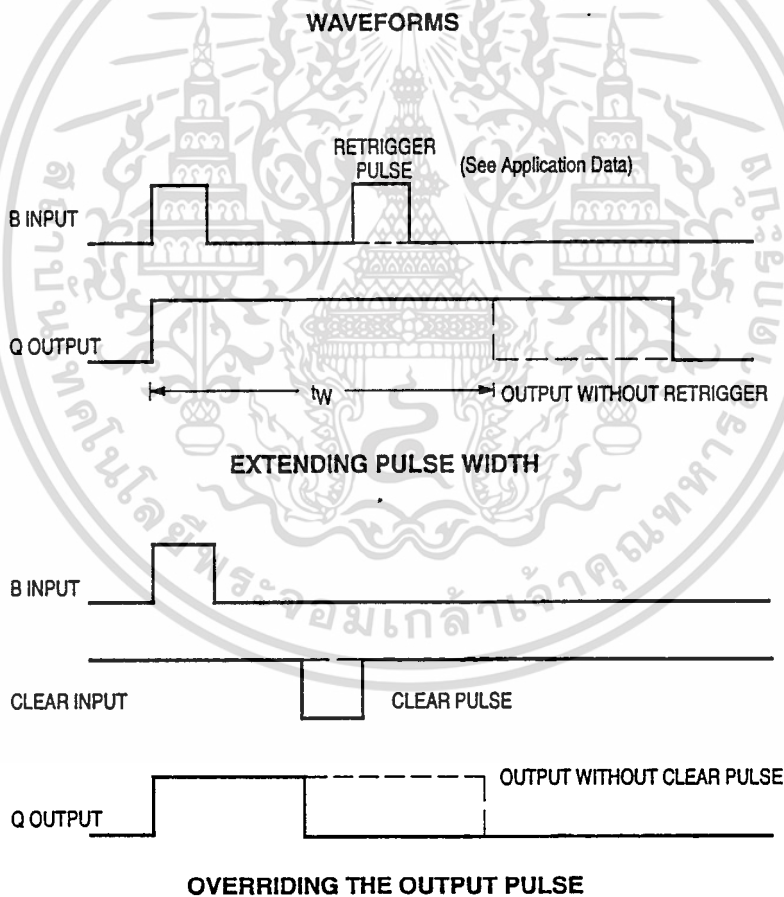
For the smallest possible deviation in output pulse widths from various devices, it is suggested that  $C_{ext}$  be kept  $\geq 1000$  pF.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

# SN54/74LS122 • SN54/74LS123

## 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	
R <sub>ext</sub>	External Timing Resistance	54	5.0		180	kΩ
		74	5.0		260	
C <sub>ext</sub>	External Capacitance	54, 74	No Restriction			
R <sub>ext</sub> /C <sub>ext</sub>	Wiring Capacitance at R <sub>ext</sub> /C <sub>ext</sub> Terminal	54, 74			50	pF



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

# SN54/74LS122 • SN54/74LS123

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

Symbol	Parameter		Limits			Unit	Test Conditions
			Min	Typ	Max		
$V_{IH}$	Input HIGH Voltage		2.0			V	Guaranteed Input HIGH Voltage for All Inputs
$V_{IL}$	Input LOW Voltage	54			0.7	V	Guaranteed Input LOW Voltage for All Inputs
		74			0.8		
$V_{IK}$	Input Clamp Diode Voltage			-0.65	-1.5	V	$V_{CC} = \text{MIN}$ , $I_{IN} = -18 \text{ mA}$
$V_{OH}$	Output HIGH Voltage	54	2.5	3.5		V	$V_{CC} = \text{MIN}$ , $I_{OH} = \text{MAX}$ , $V_{IN} = V_{IH}$ or $V_{IL}$ per Truth Table
		74	2.7	3.5		V	
$V_{OL}$	Output LOW Voltage	54, 74		0.25	0.4	V	$I_{OL} = 4.0 \text{ mA}$ , $V_{CC} = V_{CC} \text{ MIN}$ , $V_{IN} = V_{IL}$ or $V_{IH}$ per Truth Table
		74		0.35	0.5	V	$I_{OL} = 8.0 \text{ mA}$
$I_{IH}$	Input HIGH Current				20	$\mu\text{A}$	$V_{CC} = \text{MAX}$ , $V_{IN} = 2.7 \text{ V}$
					0.1	mA	$V_{CC} = \text{MAX}$ , $V_{IN} = 7.0 \text{ V}$
$I_{IL}$	Input LOW Current				-0.4	mA	$V_{CC} = \text{MAX}$ , $V_{IN} = 0.4 \text{ V}$
$I_{OS}$	Short Circuit Current (Note 1)		-20		-100	mA	$V_{CC} = \text{MAX}$
$I_{CC}$	Power Supply Current	LS122			11	mA	$V_{CC} = \text{MAX}$
		LS123			20		

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

## AC CHARACTERISTICS ( $T_A = 25^\circ\text{C}$ , $V_{CC} = 5.0 \text{ V}$ )

Symbol	Parameter		Limits			Unit	Test Conditions
			Min	Typ	Max		
$t_{PLH}$ $t_{PHL}$	Propagation Delay, A to Q Propagation Delay, A to Q		23	33		ns	$C_{ext} = 0$ $C_L = 15 \text{ pF}$ $R_{ext} = 5.0 \text{ k}\Omega$ $R_L = 2.0 \text{ k}\Omega$
			32	45			
$t_{PLH}$ $t_{PHL}$	Propagation Delay, B to Q Propagation Delay, B to Q		23	44		ns	
			34	56			
$t_{PLH}$ $t_{PHL}$	Propagation Delay, Clear to Q Propagation Delay, Clear to Q		28	45		ns	
			20	27			
$t_W \text{ min}$	A or B to Q		116	200		ns	$C_{ext} = 1000 \text{ pF}$ , $R_{ext} = 10 \text{ k}\Omega$ , $C_L = 15 \text{ pF}$ , $R_L = 2.0 \text{ k}\Omega$
$t_{WQ}$	A to B to Q	4.0	4.5	5.0		$\mu\text{s}$	

## AC SETUP REQUIREMENTS ( $T_A = 25^\circ\text{C}$ , $V_{CC} = 5.0 \text{ V}$ )

Symbol	Parameter		Limits			Unit	Test Conditions
			Min	Typ	Max		
$t_W$	Pulse Width		40			ns	

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่นิยมนำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

# SN54/74LS122 • SN54/74LS123

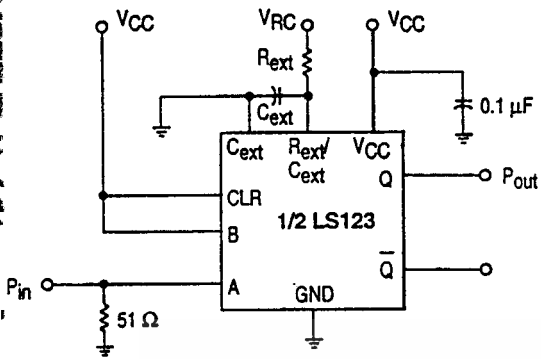


Figure 1

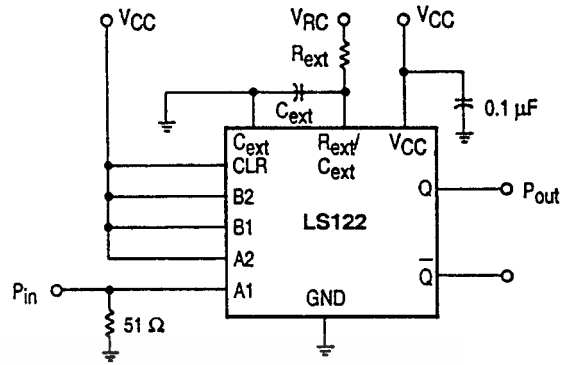


Figure 2

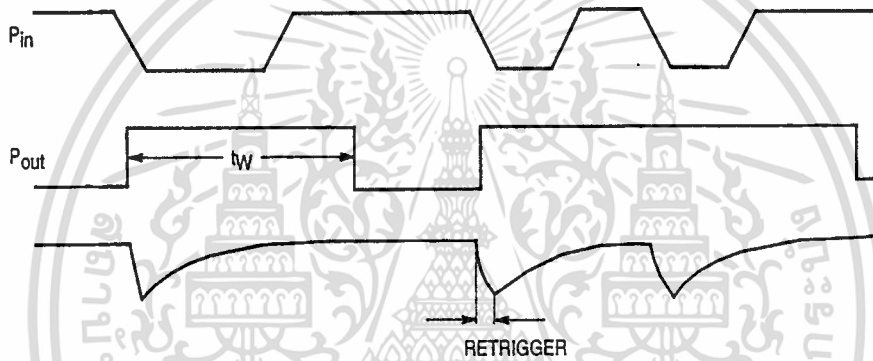


Figure 3

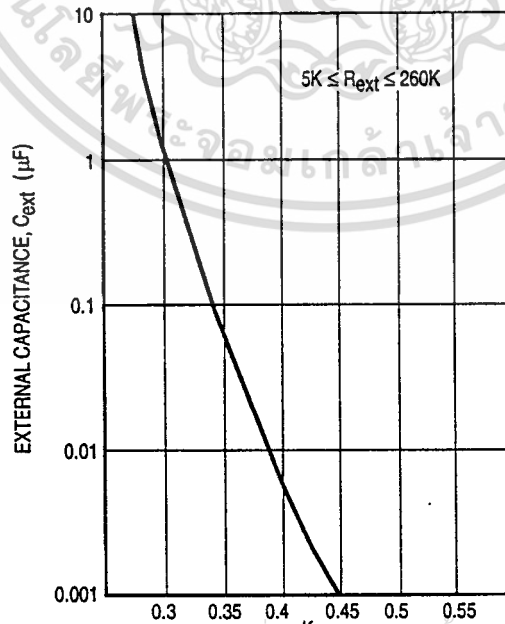


Figure 4

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

# SN54/74LS122 • SN54/74LS123

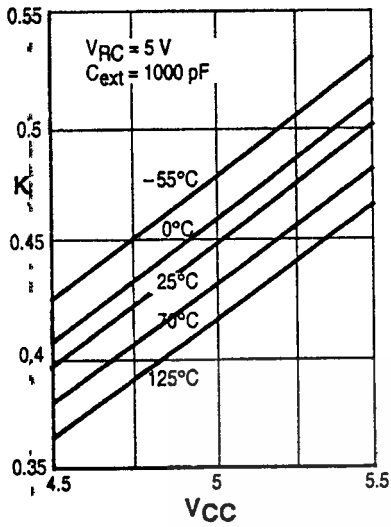


Figure 5. K versus VCC

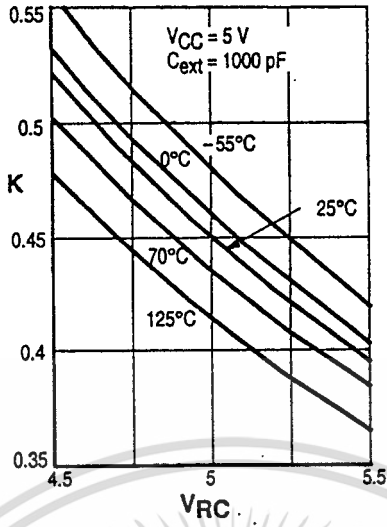


Figure 6. K versus VRC

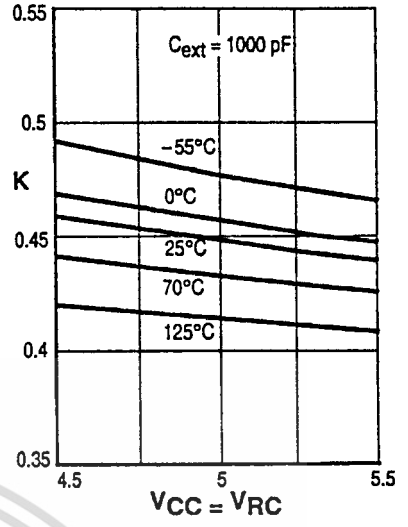


Figure 7. K versus VCC and VRC

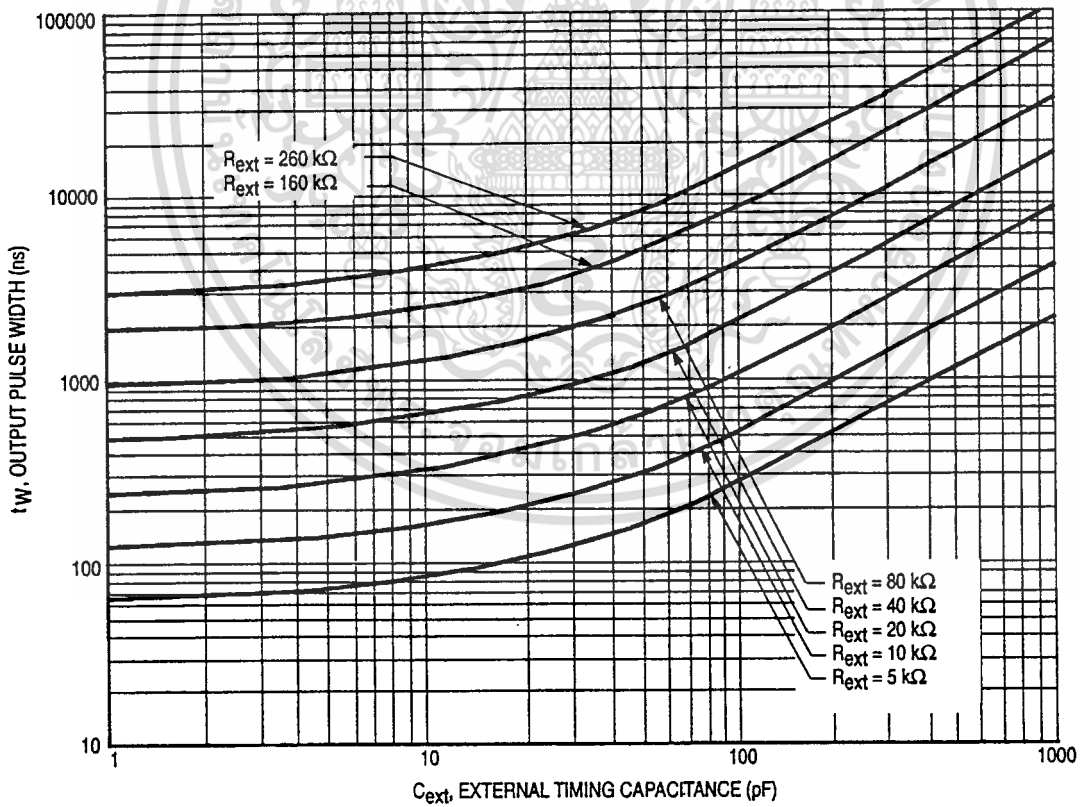


Figure 8

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

# SN54/74LS122 • SN54/74LS123

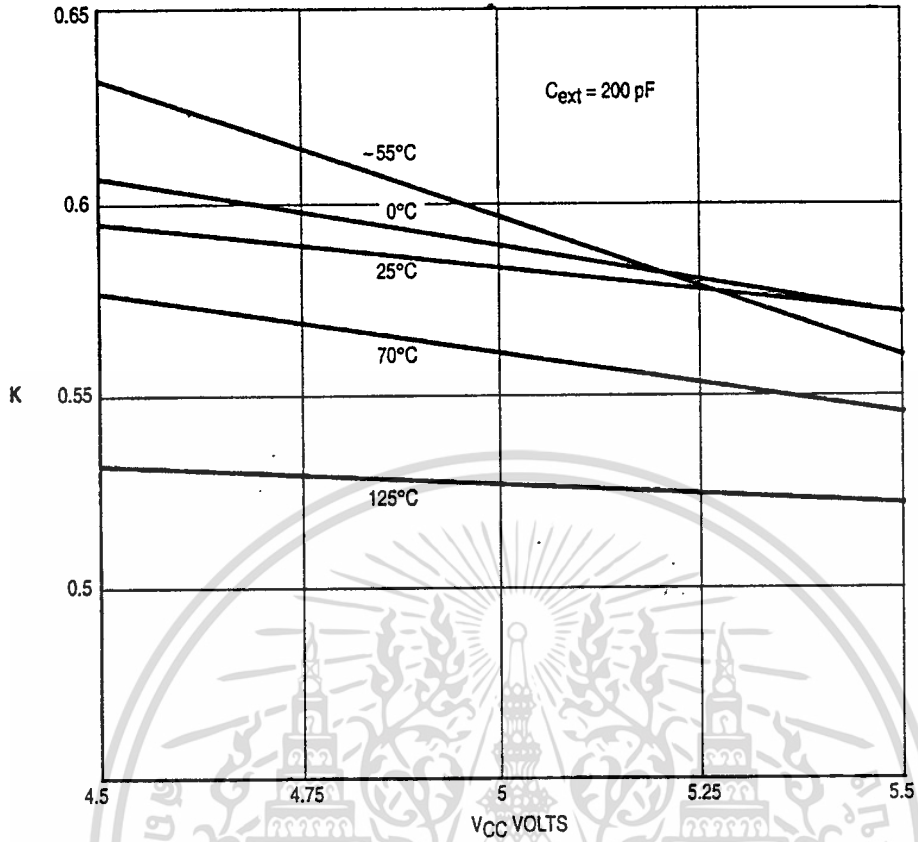


Figure 9

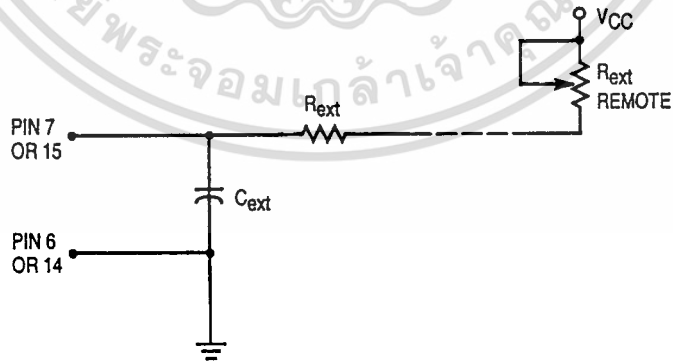


Figure 10. LS123 Remote Trimming Circuit

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## SN54/74LS122 • SN54/74LS123

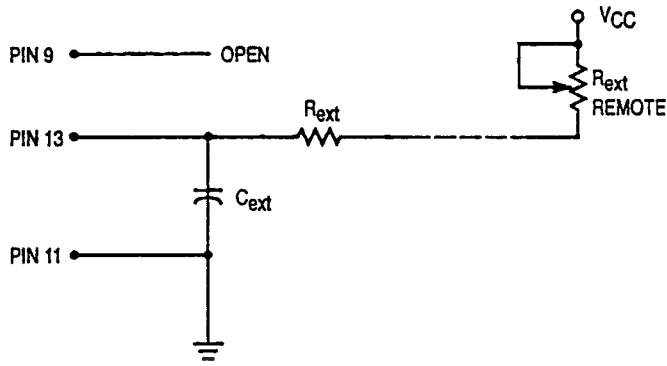


Figure 11. LS122 Remote Trimming Circuit Without  $R_{ext}$

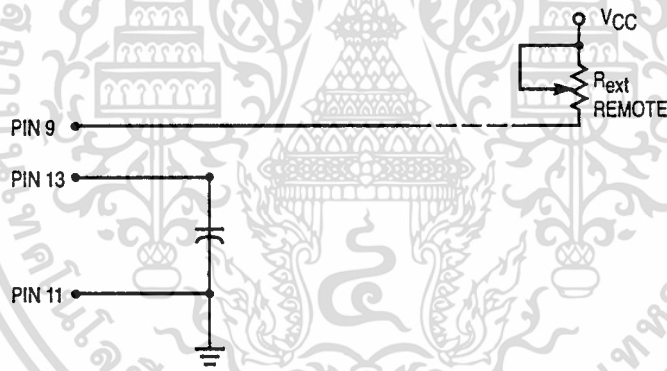


Figure 12. LS122 Remote Trimming Circuit with  $R_{int}$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

# Complementary Silicon Power Transistors

**NPN**  
**TIP3055**  
**PNP**  
**TIP2955**

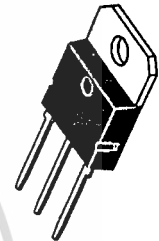
... designed for general-purpose switching and amplifier applications.

- DC Current Gain —  $h_{FE} = 20-70 @ I_C = 4.0 \text{ A dc}$
- Collector-Emitter Saturation Voltage —  $V_{CE(sat)} = 1.1 \text{ Vdc (Max) @ } I_C = 4.0 \text{ A dc}$
- Excellent Safe Operating Area

**15 AMPERE**  
**POWER TRANSISTORS**  
**COMPLEMENTARY**  
**SILICON**  
**60 VOLTS**  
**90 WATTS**

## MAXIMUM RATINGS

Rating	Symbol	Value	Unit
Collector-Emitter Voltage	$V_{CEO}$	60	Vdc
Collector-Emitter Voltage	$V_{CER}$	70	Vdc
Collector-Base Voltage	$V_{CB}$	100	Vdc
Emitter-Base Voltage	$V_{EB}$	7.0	Vdc
Collector Current — Continuous	$I_C$	1.5	A dc
Base Current	$I_B$	7.0	A dc
Total Power Dissipation @ $T_C = 25^\circ\text{C}$ Derate above $25^\circ\text{C}$	$P_D$	90 0.72	Watts W/ $^\circ\text{C}$
Operating and Storage Junction Temperature Range	$T_J, T_{stg}$	-65 to +150	$^\circ\text{C}$



CASE 340D-02

## THERMAL CHARACTERISTICS

Characteristic	Symbol	Max	Unit
Thermal Resistance, Junction to Case	$R_{\theta JC}$	1.39	$^\circ\text{C/W}$
Thermal Resistance, Junction to Ambient	$R_{\theta JA}$	35.7	$^\circ\text{C/W}$

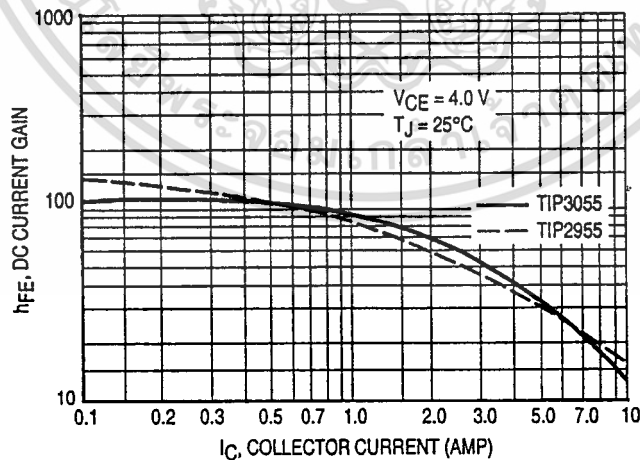


Figure 1. DC Current Gain

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

REV 1

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสาร



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

Characteristic	Symbol	Min	Max	Unit
<b>OFF CHARACTERISTICS</b>				
Collector-Emitter Sustaining Voltage (1) ( $I_C = 30\text{ mAdc}$ , $I_B = 0$ )	$V_{CE(sus)}$	60	—	Vdc
Collector Cutoff Current ( $V_{CE} = 70\text{ Vdc}$ , $R_{BE} = 100\text{ Ohms}$ )	$I_{CER}$	—	1.0	mAdc
Collector Cutoff Current ( $V_{CE} = 30\text{ Vdc}$ , $I_B = 0$ )	$I_{CEO}$	—	0.7	mAdc
Collector Cutoff Current ( $V_{CE} = 100\text{ Vdc}$ , $V_{BE(off)} = 1.5\text{ Vdc}$ )	$I_{CEV}$	—	5.0	mAdc
Emitter Cutoff Current ( $V_{BE} = 7.0\text{ Vdc}$ , $I_C = 0$ )	$I_{EBO}$	—	5.0	mAdc
<b>ON CHARACTERISTICS (1)</b>				
DC Current Gain ( $I_C = 4.0\text{ Adc}$ , $V_{CE} = 4.0\text{ Vdc}$ ) ( $I_C = 10\text{ Adc}$ , $V_{CE} = 4.0\text{ Vdc}$ )	$h_{FE}$	20 5.0	70 —	—
Collector-Emitter Saturation Voltage ( $I_C = 4.0\text{ Adc}$ , $I_B = 400\text{ mAdc}$ ) ( $I_C = 10\text{ Adc}$ , $I_B = 3.3\text{ Adc}$ )	$V_{CE(sat)}$	— —	1.1 3.0	Vdc
Base-Emitter On Voltage ( $I_C = 4.0\text{ Adc}$ , $V_{CE} = 4.0\text{ Vdc}$ )	$V_{BE(on)}$	—	1.8	Vdc
<b>SECOND BREAKDOWN</b>				
Second Breakdown Collector Current with Base Forward Biased ( $V_{CE} = 30\text{ Vdc}$ , $t = 1.0\text{ s}$ ; Nonrepetitive)	$I_{s/b}$	3.0	—	Adc
<b>DYNAMIC CHARACTERISTICS</b>				
Current Gain — Bandwidth Product ( $I_C = 0.5\text{ Adc}$ , $V_{CE} = 10\text{ Vdc}$ , $f = 1.0\text{ MHz}$ )	$f_T$	2.5	—	MHz
Small-Signal Current Gain ( $V_{CE} = 4.0\text{ Vdc}$ , $I_C = 1.0\text{ Adc}$ , $f = 1.0\text{ kHz}$ )	$h_{fe}$	15	—	kHz

(1) Pulse Test: Pulse Width = 300  $\mu\text{s}$ , Duty Cycle  $\leq 2.0\%$ .

NOTE: For additional design curves, refer to electrical characteristics curves of 2N3055.

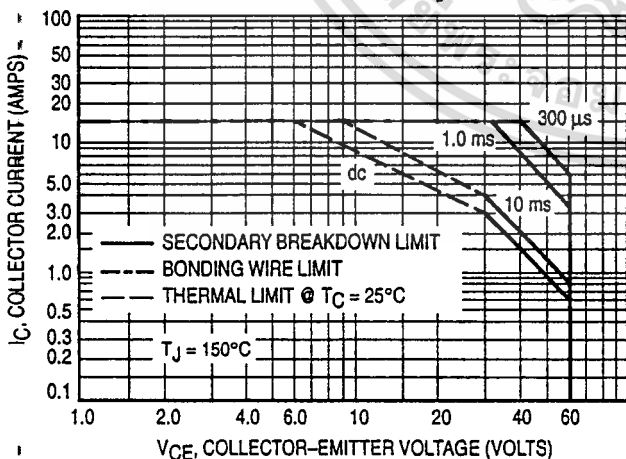


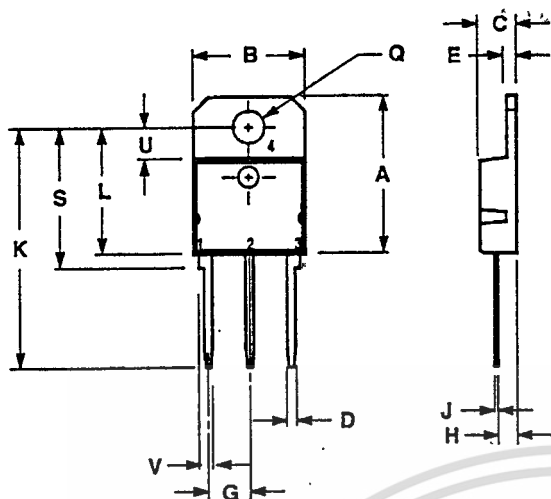
Figure 2. Maximum Rated Forward Bias Safe Operating Area

There are two limitations on the power handling ability of a transistor: average junction temperature and second breakdown. Safe operating area curves indicate  $I_C - V_{CE}$  limits of the transistor that must be observed for reliable operation; i.e., the transistor must not be subjected to greater dissipation than the curves indicate.

The data of Figure 2 is based on  $T_C = 25^\circ\text{C}$ ;  $T_J(pk)$  is variable depending on power level. Second breakdown pulse limits are valid for duty cycles to 10% but must be derated for temperature.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

PACKAGE DIMENSIONS

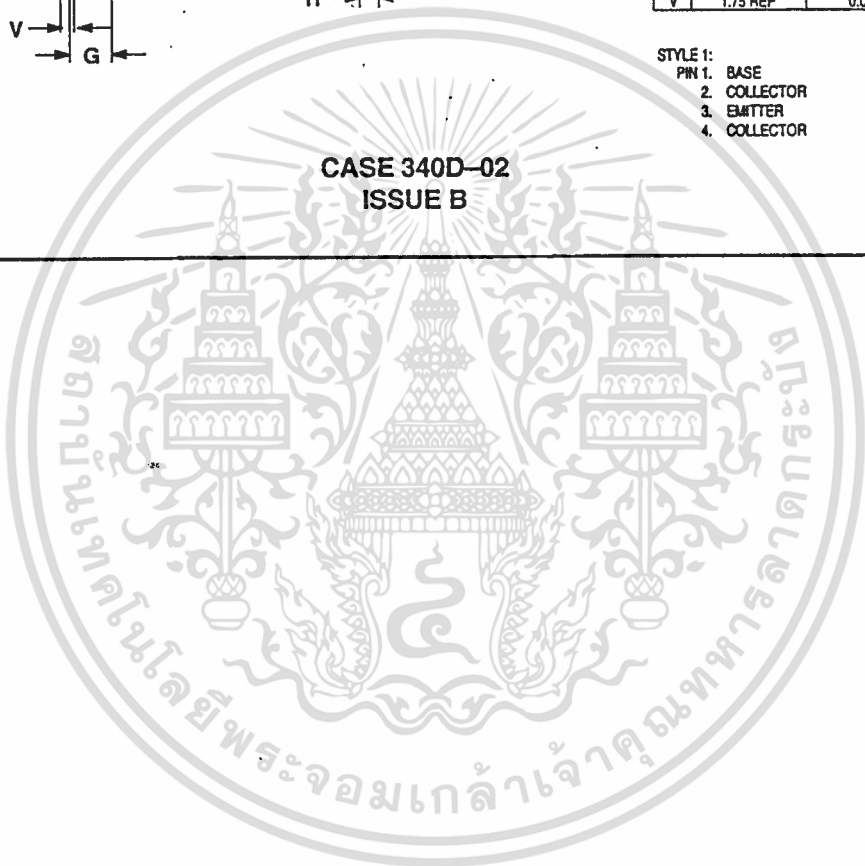


NOTES:  
 1. DIMENSIONING AND TOLERANCING PER ANSI Y14.5M, 1982.  
 2. CONTROLLING DIMENSION: MILLIMETER.

DIM	MILLIMETERS		INCHES	
	MIN	MAX	MIN	MAX
A	—	20.35	—	0.801
B	14.70	15.20	0.579	0.598
C	4.70	4.90	0.185	0.193
D	1.10	1.30	0.043	0.051
E	1.17	1.37	0.046	0.054
G	5.40	5.55	0.213	0.219
H	2.00	3.00	0.079	0.118
J	0.50	0.78	0.020	0.031
K	31.00 REF	—	1.220 REF	—
L	—	16.20	—	0.638
Q	4.00	4.10	0.158	0.161
S	17.80	18.20	0.701	0.717
U	4.00 REF	—	0.157 REF	—
V	1.75 REF	—	0.069	—


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

CASE 340D-02  
 ISSUE B



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
 ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



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

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

INTERNET: <http://Design-NET.com>



**MOTOROLA**

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



# SN54/74LS122 • SN54/74LS123

## 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>IN</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				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 1)		-20		-100	mA	V <sub>CC</sub> = MAX
I <sub>CC</sub>	Power Supply Current	LS122			11	mA	V <sub>CC</sub> = MAX
		LS123			20		

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		
t <sub>PLH</sub> t <sub>PHL</sub>	Propagation Delay, A to Q Propagation Delay, A to Q			23 32	33 45	ns	C <sub>ext</sub> = 0 C <sub>L</sub> = 15 pF
t <sub>PLH</sub> t <sub>PHL</sub>	Propagation Delay, B to Q Propagation Delay, B to Q			23 34	44 56		
t <sub>PLH</sub> t <sub>PHL</sub>	Propagation Delay, Clear to Q Propagation Delay, Clear to Q			28 20	45 27	ns	R <sub>ext</sub> = 5.0 kΩ R <sub>L</sub> = 2.0 kΩ
	t <sub>W min</sub>	A or B to Q			116		
t <sub>WQ</sub>	A to B to Q		4.0	4.5	5.0	μs	C <sub>ext</sub> = 1000 pF, R <sub>ext</sub> = 10 kΩ, C <sub>L</sub> = 15 pF, R <sub>L</sub> = 2.0 kΩ

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

Symbol	Parameter		Limits			Unit	Test Conditions
			Min	Typ	Max		
t <sub>W</sub>	Pulse Width		40			ns	

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

# SN54/74LS122 • SN54/74LS123

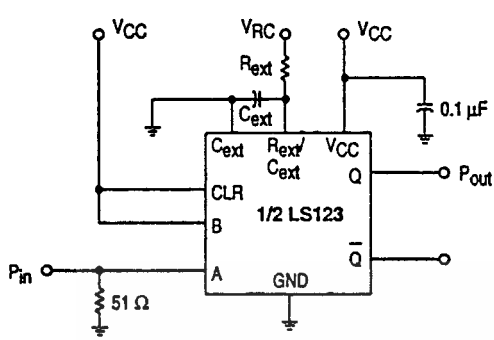


Figure 1

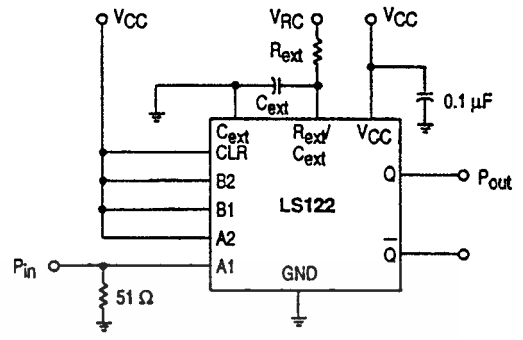


Figure 2

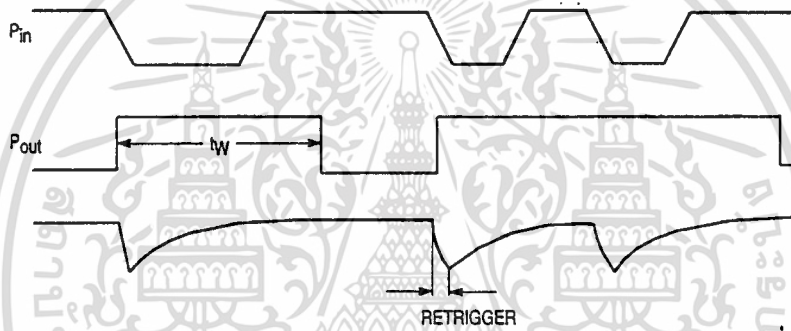


Figure 3

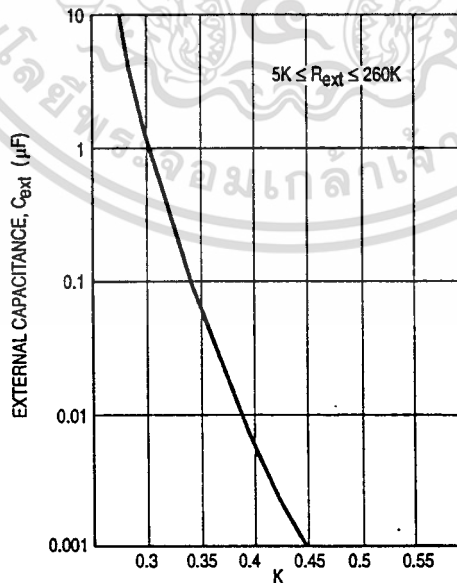


Figure 4

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์ไว้สำหรับการใช้งานที่ถูกต้องและเหมาะสม ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการศึกษา

## FAST AND LS TTL DATA

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาหรือต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

# SN54/74LS122 • SN54/74LS123

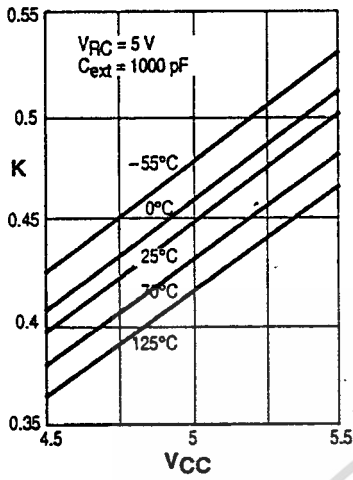


Figure 5. K versus VCC

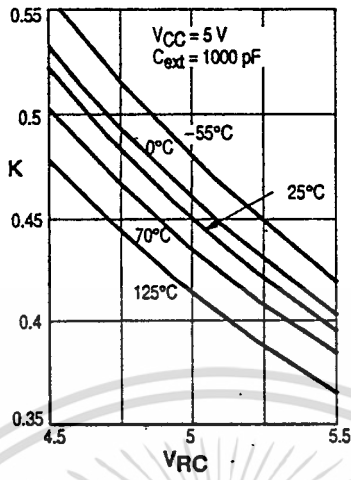


Figure 6. K versus VRC

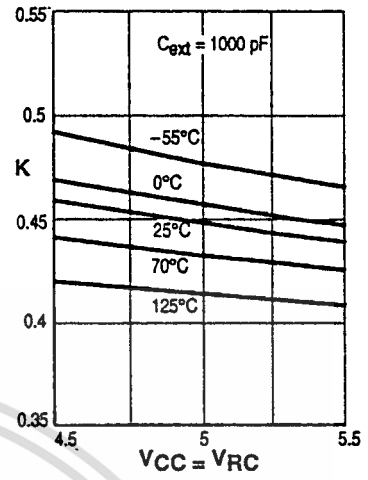


Figure 7. K versus VCC and VRC

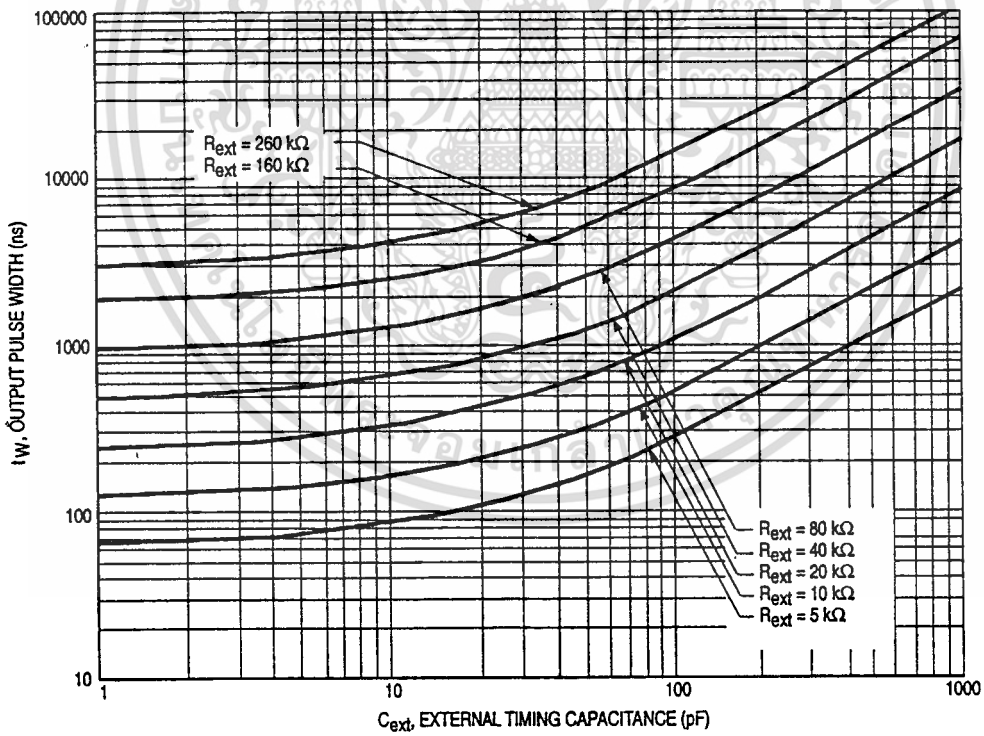


Figure 8

# SN54/74LS122 • SN54/74LS123

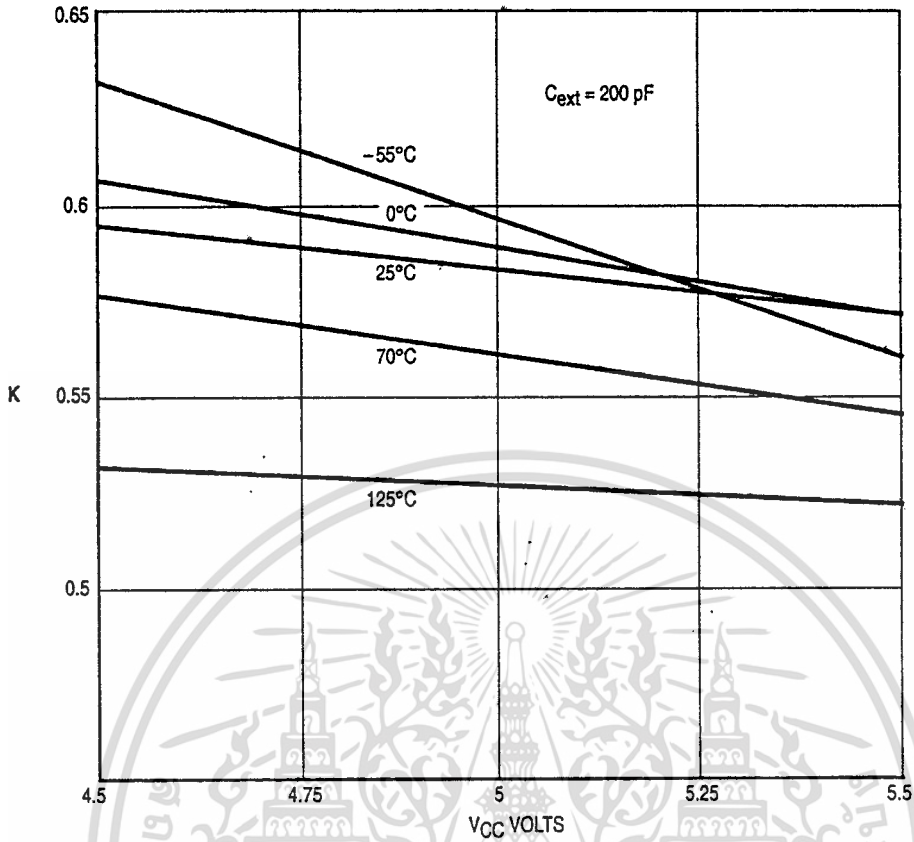


Figure 9

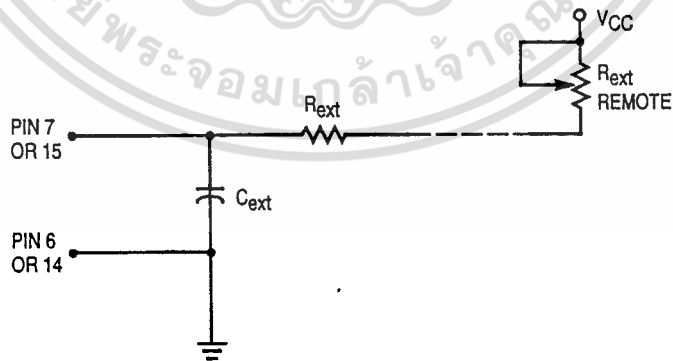


Figure 10. LS123 Remote Trimming Circuit

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## SN54/74LS122 • SN54/74LS123

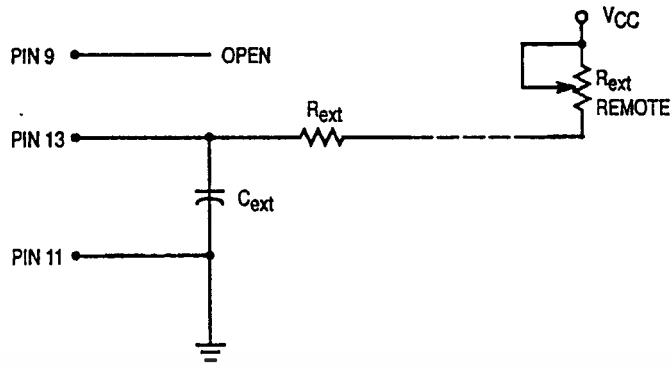


Figure 11. LS122 Remote Trimming Circuit Without  $R_{ext}$

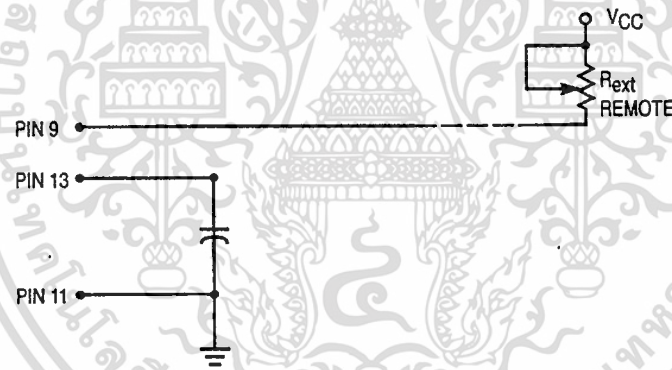


Figure 12. LS122 Remote Trimming Circuit with  $R_{int}$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

# Complementary Silicon Power Transistors

... designed for general-purpose switching and amplifier applications.

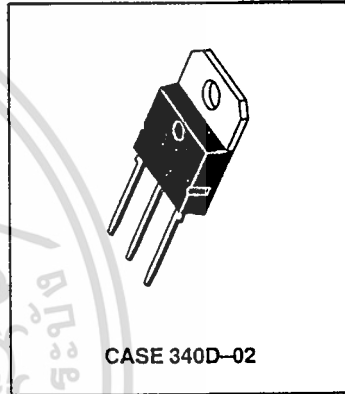
- DC Current Gain —  $h_{FE} = 20-70 @ I_C = 4.0 \text{ Adc}$
- Collector-Emitter Saturation Voltage —  $V_{CE(sat)} = 1.1 \text{ Vdc (Max) @ } I_C = 4.0 \text{ Adc}$
- Excellent Safe Operating Area

**NPN**  
**TIP3055**  
**PNP**  
**TIP2955**

**15 AMPERE**  
**POWER TRANSISTORS**  
**COMPLEMENTARY**  
**SILICON**  
**60 VOLTS**  
**90 WATTS**

## MAXIMUM RATINGS

Rating	Symbol	Value	Unit
Collector-Emitter Voltage	$V_{CEO}$	60	Vdc
Collector-Emitter Voltage	$V_{CER}$	70	Vdc.
Collector-Base Voltage	$V_{CB}$	100	Vdc
Emitter-Base Voltage	$V_{EB}$	7.0	Vdc
Collector Current — Continuous	$I_C$	1.5	A dc
Base Current	$I_B$	7.0	A dc
Total Power Dissipation @ $T_C = 25^\circ\text{C}$ Derate above $25^\circ\text{C}$	$P_D$	90 0.72	Watts W/ $^\circ\text{C}$
Operating and Storage Junction Temperature Range	$T_J, T_{stg}$	-65 to +150	$^\circ\text{C}$



## THERMAL CHARACTERISTICS

Characteristic	Symbol	Max	Unit
Thermal Resistance, Junction to Case	$R_{\theta JC}$	1.39	$^\circ\text{C/W}$
Thermal Resistance, Junction to Ambient	$R_{\theta JA}$	35.7	$^\circ\text{C/W}$

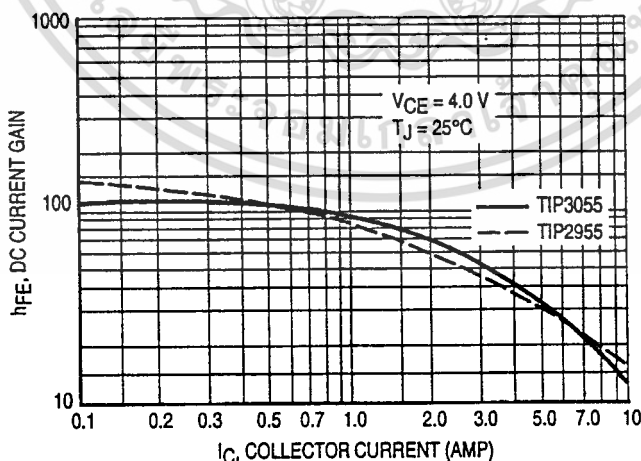


Figure 1. DC Current Gain

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

REV 1

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ELECTRICAL CHARACTERISTICS (T<sub>C</sub> = 25°C unless otherwise noted)

Characteristic	Symbol	Min	Max	Unit
<b>OFF CHARACTERISTICS</b>				
Collector-Emitter Sustaining Voltage (1) (I <sub>C</sub> = 30 mAdc, I <sub>B</sub> = 0)	V <sub>CEO(sus)</sub>	60	—	Vdc
Collector Cutoff Current (V <sub>CE</sub> = 70 Vdc, R <sub>BE</sub> = 100 Ohms)	I <sub>CER</sub>	—	1.0	mAdc
Collector Cutoff Current (V <sub>CE</sub> = 30 Vdc, I <sub>B</sub> = 0)	I <sub>CEO</sub>	—	0.7	mAdc
Collector Cutoff Current (V <sub>CE</sub> = 100 Vdc, V <sub>BE(off)</sub> = 1.5 Vdc)	I <sub>CEV</sub>	—	5.0	mAdc
Emitter Cutoff Current (V <sub>BE</sub> = 7.0 Vdc, I <sub>C</sub> = 0)	I <sub>EBO</sub>	—	5.0	mAdc

<b>ON CHARACTERISTICS (1)</b>				
DC Current Gain (I <sub>C</sub> = 4.0 Adc, V <sub>CE</sub> = 4.0 Vdc) (I <sub>C</sub> = 10 Adc, V <sub>CE</sub> = 4.0 Vdc)	h <sub>FE</sub>	20 5.0	70 —	—
Collector-Emitter Saturation Voltage (I <sub>C</sub> = 4.0 Adc, I <sub>B</sub> = 400 mAdc) (I <sub>C</sub> = 10 Adc, I <sub>B</sub> = 3.3 Adc)	V <sub>CE(sat)</sub>	— —	1.1 3.0	Vdc
Base-Emitter On Voltage (I <sub>C</sub> = 4.0 Adc, V <sub>CE</sub> = 4.0 Vdc)	V <sub>BE(on)</sub>	—	1.8	Vdc

<b>SECOND BREAKDOWN</b>				
Second Breakdown Collector Current with Base Forward Biased (V <sub>CE</sub> = 30 Vdc, t = 1.0 s; Nonrepetitive)	I <sub>s/b</sub>	3.0	—	Adc

<b>DYNAMIC CHARACTERISTICS</b>				
Current Gain — Bandwidth Product (I <sub>C</sub> = 0.5 Adc, V <sub>CE</sub> = 10 Vdc, f = 1.0 MHz)	f <sub>T</sub>	2.5	—	MHz
Small-Signal Current Gain (V <sub>CE</sub> = 4.0 Vdc, I <sub>C</sub> = 1.0 Adc, f = 1.0 kHz)	h <sub>fe</sub>	15	—	kHz

(1) Pulse Test: Pulse Width = 300 μs, Duty Cycle ≤ 2.0%.  
NOTE: For additional design curves, refer to electrical characteristics curves of 2N3055.

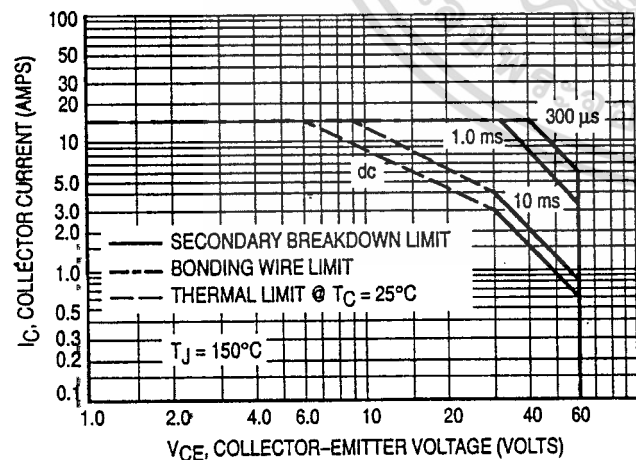


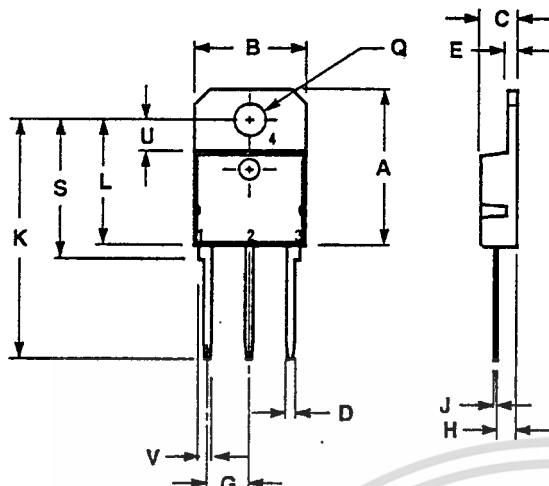
Figure 2. Maximum Rated Forward Bias Safe Operating Area

There are two limitations on the power handling ability of a transistor: average junction temperature and second breakdown. Safe operating area curves indicate I<sub>C</sub> – V<sub>CE</sub> limits of the transistor that must be observed for reliable operation; i.e., the transistor must not be subjected to greater dissipation than the curves indicate.

The data of Figure 2 is based on T<sub>C</sub> = 25°C; T<sub>J(pk)</sub> is variable depending on power level. Second breakdown pulse limits are valid for duty cycles to 10% but must be derated for temperature.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำสิ่งนี้ไปใช้

PACKAGE DIMENSIONS

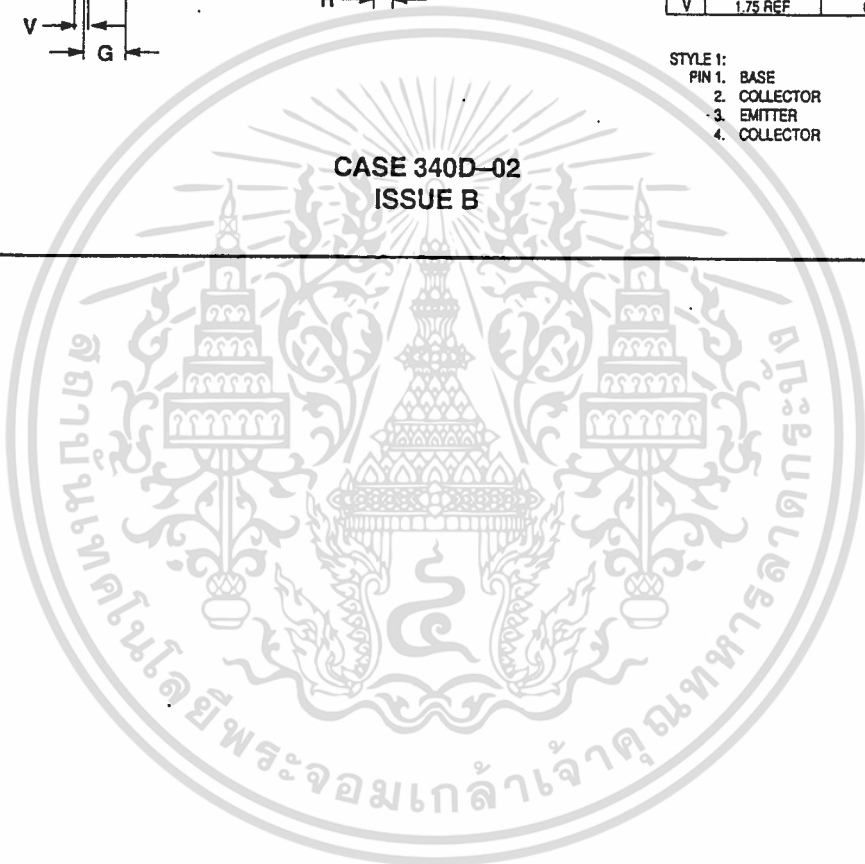


NOTES:  
 1. DIMENSIONING AND TOLERANCING PER ANSI Y14.5M, 1982.  
 2. CONTROLLING DIMENSION: MILLIMETER.

DIM	MILLIMETERS		INCHES	
	MIN	MAX	MIN	MAX
A	—	20.25	—	0.801
B	14.70	15.20	0.579	0.598
C	4.70	4.90	0.185	0.193
D	1.10	1.30	0.043	0.051
E	1.17	1.37	0.046	0.054
G	5.40	5.55	0.213	0.219
H	2.00	3.00	0.079	0.118
J	0.50	0.78	0.020	0.031
K	31.00 REF	—	1.220 REF	—
L	—	16.20	—	0.638
Q	4.00	4.10	0.158	0.161
S	17.80	18.20	0.701	0.717
U	4.00 REF	—	0.157 REF	—
V	1.75 REF	—	0.069	—

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

CASE 340D-02  
 ISSUE B

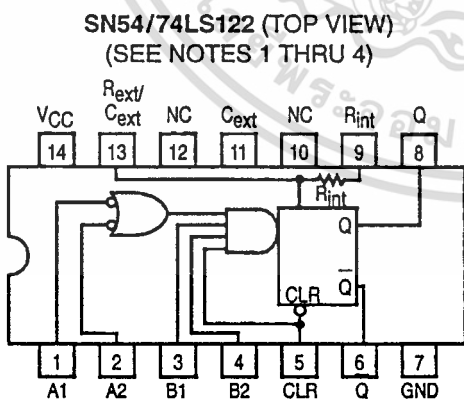
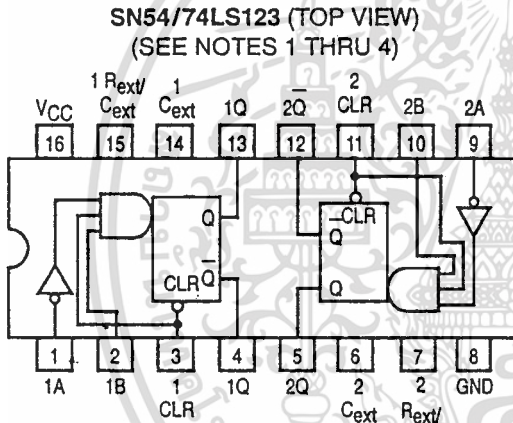


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
 ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

# RETRIGGERABLE MONOSTABLE MULTIVIBRATORS

These dc triggered multivibrators feature pulse width control by three methods. The basic pulse width is programmed by selection of external resistance and capacitance values. The LS122 has an internal timing resistor that allows the circuits to be used with only an external capacitor. Once triggered, the basic pulse width may be extended by retriggering the gated low-level-active (A) or high-level-active (B) inputs, or be reduced by use of the overriding clear.

- Overriding Clear Terminates Output Pulse
- Compensated for  $V_{CC}$  and Temperature Variations
- DC Triggered from Active-High or Active-Low Gated Logic Inputs
- Retriggerable for Very Long Output Pulses, up to 100% Duty Cycle
- Internal Timing Resistors on LS122



NC — NO INTERNAL CONNECTION.

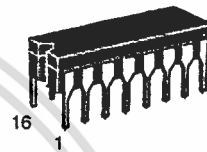
**NOTES:**

1. An external timing capacitor may be connected between  $C_{ext}$  and  $R_{ext}/C_{ext}$  (positive).
2. To use the internal timing resistor of the LS122, connect  $R_{int}$  to  $V_{CC}$ .
3. For improved pulse width accuracy connect an external resistor between  $R_{ext}/C_{ext}$  and  $V_{CC}$  with  $R_{int}$  open-circuited.
4. To obtain variable pulse widths, connect an external variable resistance between  $R_{int}/C_{ext}$  and  $V_{CC}$ .

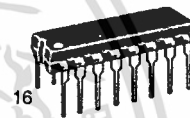
**SN54/74LS122**  
**SN54/74LS123**

**RETRIGGERABLE MONOSTABLE MULTIVIBRATORS**

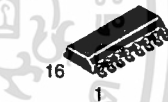
**LOW POWER SCHOTTKY**



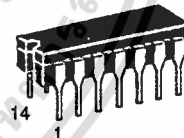
**J SUFFIX**  
**CERAMIC**  
CASE 620-09



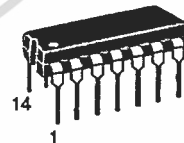
**N SUFFIX**  
**PLASTIC**  
CASE 648-08



**D SUFFIX**  
**SOIC**  
CASE 751B-03



**J SUFFIX**  
**CERAMIC**  
CASE 632-08



**N SUFFIX**  
**PLASTIC**  
CASE 646-06



**D SUFFIX**  
**SOIC**  
CASE 751A-02

**ORDERING INFORMATION**

SN54LSXXXJ Ceramic  
SN74LSXXXN Plastic  
SN74LSXXXD SOIC

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้