



สายอากาศปรับทิศทางอัตโนมัติ

AUTOMATIC ROTATING ANTENNA



|                                     |
|-------------------------------------|
| วัน เดือน ปี..... ๒๑ ก.ค. ๒๕๕๐      |
| เลขทะเบียน..... ๐๓๖๙๙๓              |
| เลขเรียกหนังสือ..... T.3๔๐๘๖ ก.๒๗๔๙ |

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาอุตสาหกรรมศาสตรบัณฑิต  
สาขาวิชาเทคโนโลยีโทรคมนาคม

คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานภายในเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

036993

หัวข้อปริญญานิพนธ์                   สายอากาศปรับทิศทางอัตโนมัติ  
AUTOMATIC ROTATING ANTENNA

จัดทำโดย                                   นาย กิตติพงศ์ แหวนทอง 37012040  
                                                  นาย พรเทพ ปิณฑพงษ์ 37012064  
                                                  นาย วิโรจน์ แสงสุวรรณ 37012066

อาจารย์ที่ปรึกษา                        อาจารย์ อุทัย ศรีธีระวิโรจน์  
                                                  อาจารย์ ขวลิศ เบญจางคประเสริฐ

ภาควิชา                                    เทคนิคอุตสาหกรรม

คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง  
อนุมัติให้นับเป็นส่วนหนึ่งของการศึกษาคตามหลักสูตรปริญญาอุตสาหกรรมศาสตรบัณฑิต

คณะกรรมการสอบปริญญานิพนธ์

..... ประธานกรรมการ  
( ..... )  
..... กรรมการ  
( ..... )  
..... กรรมการ  
( ..... )  
..... กรรมการ  
( ..... )  
..... กรรมการ  
( ..... )  
..... กรรมการ  
( ..... )

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ลิขสิทธิ์ของคณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## เรื่อง สายอากาศปรับทิศทางอัตโนมัติ

โดย

นาย กิตติพงศ์ แหวนทอง

นาย พรเทพ ปิญพงษ์

นาย วิโรจน์ แสงสุวรรณ

อาจารย์ที่ปรึกษา

อาจารย์อุทัย ศรีธีระวิโรจน์

อาจารย์ชวลิต เบญจางคประเสริฐ

ปีการศึกษา 2538

## บทคัดย่อ

ระบบควบคุมอัตโนมัติเป็นระบบที่ถูกนำมาใช้งานอย่างแพร่หลายในปัจจุบันและ  
 ปรินญาณิพนธ์ฉบับนี้ได้นำระบบควบคุมอัตโนมัติมาใช้ควบคุมการหมุนเสาอากาศของสาย  
 อากาศแบบยกลีแฉงรวมของเครื่องรับโทรทัศน์ทำให้เครื่องรับโทรทัศน์มีประสิทธิภาพในการรับ  
 สัญญาณภาพได้ดียิ่งขึ้น โครงการสายอากาศปรับทิศทางอัตโนมัตินี้เป็นการนำไมโครโปรเซส  
 เซอร์เบอร์ Z-80 มาประยุกต์ใช้งานเป็นตัวควบคุมสายอากาศให้หมุนไปในทิศทางที่รับสัญญาณ  
 ได้ดีที่สุด โดยนำสัญญาณ AGC ในตำแหน่งต่างๆ ในขณะที่สายอากาศหมุนจาก 0 ถึง 360 องศา  
 มาเปรียบเทียบกันและเก็บค่าแห่งที่มีค่าแรงดัน AGC น้อยที่สุดไว้ หลังจากนั้นไมโครโปรเซส  
 เซอร์ก็จะสั่งให้สายอากาศหมุนไปยังตำแหน่งที่รับสัญญาณได้ชัดเจนที่สุด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
 ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Title      Automatic Rotating Antenna

By

Mr. Kittipong Vanthong

Mr. Porntep panohapong

Mr. Wirote Sangsuwan

Adviser

Mr. Uthai Shreeteeravirote

Mr. Chawalit Benjangkaprasert

Year      1995

## ABSTRACT

Currently, automatic control systems are used widely for a multitude of functions. The primary objective of this project is to make use of automatic control systems for the control of the rotation of a YAKI T.V. antenna. This, in turn, improves the efficiency of the reception of televisions utilizing this system.

This project used microprocessor Z-80 and applied its, to control automatically rotating antenna in the best signal direction. Using AGC signal in several positions. While the antenna rotating 0-360 degree. Microprocessor will compare AGC signal for choose the position that have smallest AGC signal. After that the microprocessor will control antenna to the best position that have smallest AGC signal.

### กิตติกรรมประกาศ

ขอขอบคุณอาจารย์อุทัย ศรีธีระวิโรจน์ และ อาจารย์ชวลิต เบญจางคประเสริฐ ที่ได้ให้คำปรึกษาและคำแนะนำในการจัดทำโครงงานและปริญญานิพนธ์ในครั้งนี้และรวมทั้งเพื่อนๆที่ได้ให้ความช่วยเหลือและคำแนะนำในครั้งนี้ด้วยดีตลอดมา

ท้ายนี้ผู้จัดทำใคร่ขอกราบขอบพระคุณ บิดา-มารดา ซึ่งสนับสนุนในด้านการเงินและให้กำลังใจแก่ผู้จัดทำตลอดมาจนสามารถทำการวิจัยและสร้างเครื่องสายอากาศปรับทิศทางอัตโนมัติจนสำเร็จลุล่วงไปได้ด้วยดี



กิตติพงศ์ แหวนทอง

พรเทพ ปัญญพงษ์

วิโรจน์ แสงสุวรรณ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## สารบัญ

หน้าที่

|                                                              |    |
|--------------------------------------------------------------|----|
| บทคัดย่อ (ภาษาไทย) .....                                     | ก  |
| บทคัดย่อ (ภาษาอังกฤษ) .....                                  | ข  |
| กิตติกรรมประกาศ .....                                        | ค  |
| บทที่ 1                                                      |    |
| บทนำ .....                                                   | 1  |
| 1.1 วัตถุประสงค์ .....                                       | 1  |
| 1.2 ขอบเขตของปริิญยานิพนธ์ .....                             | 1  |
| 1.3 แนวคิดและที่มาของโครงการ .....                           | 1  |
| 1.4 ประโยชน์ที่คาดว่าจะได้รับจากโครงการเมื่อพัฒนาเสร็จ ..... | 2  |
| บทที่ 2                                                      |    |
| ทฤษฎีที่เกี่ยวข้อง .....                                     | 3  |
| 2.1 ระบบสายอากาศเครื่องรับโทรทัศน์ .....                     | 3  |
| 2.2 วงจรควบคุมการขยายสัญญาณอัตโนมัติ .....                   | 9  |
| 2.3 ไมโครโปรเซสเซอร์ Z80 .....                               | 12 |
| 2.3.1 การใช้ ROM กับ Z80 .....                               | 16 |
| 2.3.2 การใช้ STATIC RAM กับ Z80 .....                        | 24 |
| 2.3.3 Z80 กับ อินพุตและเอาต์พุต .....                        | 28 |
| 2.3.4 Z80 กับ การอินเทอร์รัพท์ .....                         | 35 |
| 2.3.5 การใช้ 8255 PIA กับ Z80 .....                          | 43 |
| 2.3.6 ไมโครโปรเซสเซอร์ Z80 CPA .....                         | 52 |
| 2.4 วงจรแปลงอะนาลอกเป็นดิจิทัล .....                         | 58 |
| 2.5 สเต็ปปีงมอเตอร์ .....                                    | 61 |
| 2.6 ออปแอมป์ .....                                           | 66 |
| 2.7 จอแสดงผล LCD .....                                       | 77 |

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 3

|                                                         |     |
|---------------------------------------------------------|-----|
| การออกแบบและกำหนดค่าอุปกรณ์ในแต่ละส่วน .....            | 84  |
| 3.1 แนวความคิดในการออกแบบ .....                         | 84  |
| 3.2 ส่วนประกอบของโครงการและการทำงานของแต่ละบล็อก .....  | 84  |
| 3.3 การออกแบบ .....                                     | 87  |
| 3.3.1 ชุดตรวจสอบแรงดัน AGC .....                        | 87  |
| 3.3.2 ชุดควบคุมการหมุนของสายอากาศ .....                 | 95  |
| 3.3.3 ชุดตรวจสอบการเปลี่ยนช่องและตรวจสอบระบบอื่นๆ ..... | 99  |
| 3.3.4 ชุด POWER SUPPLY .....                            | 101 |
| 3.3.5 การออกแบบโปรแกรมของเครื่องควบคุม .....            | 102 |

บทที่ 4

|                                   |     |
|-----------------------------------|-----|
| ผลการทดลองและสรุปผลการทดลอง ..... | 123 |
|-----------------------------------|-----|

บทที่ 5

|                                                      |     |
|------------------------------------------------------|-----|
| การใช้งานของเครื่องสายอากาศปรับทิศทางอัตโนมัติ ..... | 131 |
|------------------------------------------------------|-----|

หนังสืออ้างอิง

ภาคผนวก ก ลายวงจร

ภาคผนวก ข DATA SHEET

## บทที่ 1

### บทนำ

#### 1.1 วัตถุประสงค์

1. เพื่อจัดทำอุปกรณ์และหาวิวัฒนาการใหม่ๆ ที่จะเพิ่มประสิทธิภาพของสายอากาศที่ใช้ในการรับสัญญาณโทรทัศน์
2. เพื่อสร้างเสริมทักษะและแนวคิดในการนำไมโครโปรเซสเซอร์ไปประยุกต์ใช้งานในด้านต่างๆ ได้
3. เพื่อสร้างสายอากาศปรับทิศทางอัตโนมัติ

#### 1.2 ขอบเขตของปริิญาานิพนธ์

1. โครงการนี้ได้นำไมโครโปรเซสเซอร์มาประยุกต์ในการควบคุมการหมุนของเสาอากาศแบบยาก็ให้มีทิศทางตามทิศทางของสัญญาณที่แรงที่สุดจากสถานีส่งสัญญาณโทรทัศน์
2. โครงการนี้สามารถแสดงความแรงของสัญญาณ RF ที่รับเข้ามาได้
3. โครงการนี้จะแสดงมุมทิศทางของการหมุนของเสาอากาศเป็นองศา โดยใช้การแสดงผลแบบ LCD

#### 1.3 แนวคิดและที่มาของโครงการ

เนื่องจากในปัจจุบันตำแหน่งที่ตั้งของเสาอากาศที่มีขายค้ำห้องคลดมีการคิดตั้งค่าแห่งค่อนข้างแน่นอนเช่นสายอากาศแบบยาก็ซึ่งไม่สามารถที่จะติดตั้งสายอากาศเพื่อให้อาจารับสัญญาณโทรทัศน์ได้ชัดเจนทุกช่องความถี่ได้ดังนั้นถ้าต้องการให้มีการรับสัญญาณโทรทัศน์ได้ชัดเจนทุกช่องความถี่ทำโดยการแยกสายอากาศรับสัญญาณโทรทัศน์ของแต่ละช่องความถี่สัญญาณมาติดตั้งอีกซึ่งทำให้สิ้นเปลืองและในบางครั้งอาจไม่สามารถติดตั้งได้เนื่องจากความจำกัดของพื้นที่ซึ่งทำให้มองเห็นปัญหาคือจะทำอย่างไรให้สายอากาศมีขนาดกระทัดรัด( สายอากาศ ยากิ แบบแผงรวม ) สามารถรับสัญญาณโทรทัศน์ได้ชัดเจนทุกช่องความถี่

เนื่องจากตำแหน่งการติดตั้งของเสาอากาศโดยปกติจะติดตั้งในตำแหน่งที่คงที่ ดังนั้นการที่จะทำให้สายอากาศสามารถรับสัญญาณโทรทัศน์ได้ชัดเจนทุกครั้งที่มีการเปลี่ยนช่องความถี่ของโทรทัศน์ทำได้โดยการปรับจูนที่เครื่องรับโทรทัศน์ใหม่หรืออีกวิธีหนึ่งคือ

เอกสารนี้เป็นเอกสารที่จัดทำขึ้นเพื่อใช้ในการเรียนการสอนเท่านั้น เมื่อผู้ใดเห็นประโยชน์ในการนำเอกสารนี้ไปใช้โดยไม่ได้รับอนุญาตจากผู้จัดทำเอกสารนี้ถือว่าผิดกฎหมาย

การหมุนเสาอากาศไปในทิศทางที่มีสัญญาณชัดเจนที่สุด ถ้าให้ผู้ใช้งานไปทำการหมุนเสาอากาศทุกครั้งที่มีการเปลี่ยนช่องความถี่จะเป็นการยุ่งยาก ดังนั้นถ้าเราให้เสาอากาศปรับหมุนหาคลื่นเองทุกครั้งที่มีการเปลี่ยนช่องความถี่จะง่ายกว่าให้ผู้ใช้งานหมุนเสาอากาศเองทุกครั้ง ดังนั้นจึงได้มีแนวความคิดที่จะนำเอาไมโครโปรเซสเซอร์มาประยุกต์เป็นตัวที่คอย

หมุนเสาอากาศให้เวลาที่มีการเปลี่ยนช่องความถี่สมาชิกในกลุ่มจึงได้รวบรวมข้อมูลและเสนอโครงการเรื่องนี้

#### 1.4 ประโยชน์ที่คาดว่าจะได้รับจากโครงการเมื่อพัฒนาเสร็จ

1. ได้สายอากาศปรับทิศทางอัตโนมัติ โดยมีทิศทางตามทิศทางของสถานีส่งสัญญาณโทรทัศน์
2. สามารถนำโครงการนี้ไปประยุกต์ใช้กับสายอากาศโทรทัศน์แบบอื่นๆ ได้
3. โครงการนี้จะสะดวกสบายในการเคลื่อนย้ายเสาอากาศ
4. สายอากาศแบบปรับทิศทางอัตโนมัติสามารถติดตั้งได้ทั้งในกลางแจ้ง และใน
5. สามารถนำแนวความคิด ไปประยุกต์ใช้ในงานควบคุมด้านอื่นๆ

ตัวอาคาร

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สายอากาศปรับทิศทางอัตโนมัติ  
AUTOMATIC ROTATING ANTENNA



นาย กิตติพงศ์ แหวนทอง  
นาย พรเทพ ปัญงพงษ์  
นาย วิโรจน์ แสงสุวรรณ

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาอุตสาหกรรมศาสตรบัณฑิต  
สาขาวิชาเทคโนโลยีโทรคมนาคม  
คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ปีการศึกษา 2538  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## บทที่ 2

### ทฤษฎีที่เกี่ยวข้อง

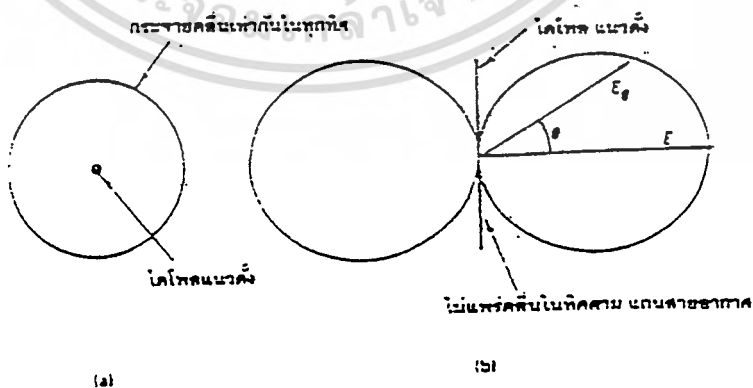
#### 2.1 ระบบสายอากาศเครื่องรับโทรทัศน์

การส่งสัญญาณความถี่วิทยุเป็นวิธีการแพร่กระจายคลื่นแม่เหล็กไฟฟ้าออกไปในอากาศซึ่งเป็นผลงานการค้นคว้าของ ไฮน์ริช เฮอร์ตซ์ นักวิทยาศาสตร์ชาวเยอรมัน โดยการนำเอาความถี่จากเครื่องส่งผ่านระบบสายส่ง ( TRANSMISSION LINE ) ออกไปสู่ระบบสายอากาศเหมือนระบบจ่ายกำลังทางไฟฟ้าออกไปยังผู้ที่ต้องการใช้งานเพียงแต่ระบบนี้ไม่มีสายส่งมีเพียงอากาศเป็นตัวกลาง

เครื่องรับโทรทัศน์จะรับพลังงานในรูปคลื่นแม่เหล็กไฟฟ้าอุปกรณ์ในการรับสัญญาณเป็นอุปกรณ์ในระบบพาสซีฟ ( PASSIVE DEVICE ) ซึ่งโดยตัวของมันเองนั้นมีการเสื่อมคุณภาพน้อยมากเมื่อเทียบกับอุปกรณ์แอคทีฟ ( ACTIVE DEVICE ) ในการรับสัญญาณโทรทัศน์ปัจจุบันโดยส่วนมากจะใช้สายอากาศแบบ ยากิ

#### สายอากาศแบบยากิ ( Yagi antenna )

เราสร้างสายอากาศแบบยากิจากไดโพลแบบ  $\lambda/2$  และพาราซิติก อีลิเมนต์ ( Parasitic element ) ความหมายของ พาราซิติกอีลิเมนต์ คือส่วนของสายอากาศที่ไม่ได้ต่อโดยตรงกับสายนำสัญญาณจากเครื่องรับหรือเครื่องส่ง แต่สามารถเหนี่ยวนำให้เกิดกระแสหรือแรงดันบนตัวมันได้และสายอากาศที่นำมาใช้งานร่วมกับ พาราซิติก เรียกว่า พาราซิติก อาร์เรย์ คุณสมบัติของไดโพลแบบ  $\lambda/2$  มีรูปแบบการแพร่คลื่นในระนาบแนวราบของไดโพลที่วางแนวตั้งเป็นวงกลมดังรูป

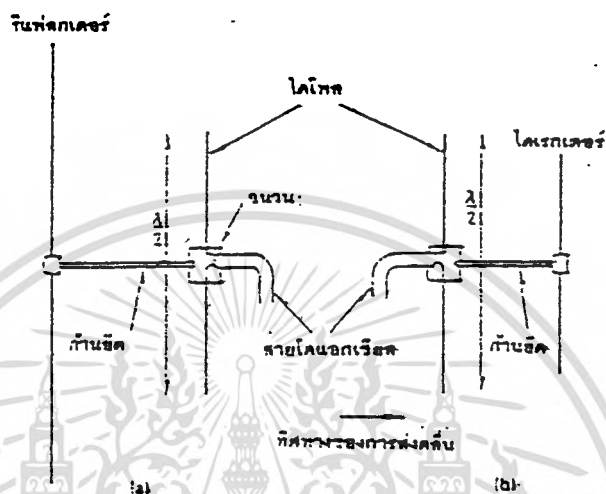


รูปที่ 2.1 แสดงแบบการแพร่คลื่นของสายอากาศไดโพลแบบแนวตั้ง (ขนาด  $\lambda/2$ )

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับ (a) รูปแบบระนาบแนวอน (b) รูปแบบระนาบแนวตั้ง ซึ่งประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

และในระนาบแนวตั้งพบว่า ไม่มีการแพร่หรือรับคลื่นเลยซึ่งในระบบสื่อสารวิทยุ  
ทั่วไปการประสิทธิภาพของสายอากาศที่มีโคเรคทีฟมากกว่าหนึ่ง

การเพิ่มพลังงานในโคเรคทีฟทำได้โดยการใช้โคโพลแบบ  $\lambda/2$  ร่วมกับพาราซี  
ติกไว้อีกด้านของสายอากาศในทิศทางตรงข้ามกับทิศที่มีการแพร่คลื่นมากที่สุดดังรูป



รูปที่ 2.2 แสดงสายอากาศโคโพลแบบ  $\lambda/2$  พร้อมกับ  
(a) รีเฟล็กเตอร์ (b) ไดเรกเตอร์

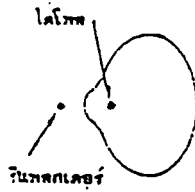
รีเฟล็กเตอร์มีผลต่อรูปแบบการแพร่คลื่นของโคโพลแบบ  $\lambda/2$  เนื่องจากมี  
สนามแม่เหล็กไฟฟ้าถูกเหนี่ยวนำที่มีผลให้ตัวรีเฟล็กเตอร์สามารถแพร่คลื่นเองได้

ตัวแปรที่มีผลต่อรูปแบบการแพร่คลื่น มี

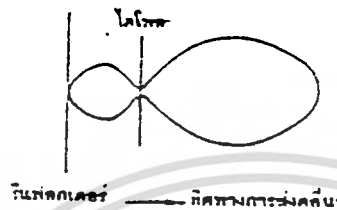
1. ความยาวของรีเฟล็กเตอร์
2. ระยะห่างจากโคโพล

พิจารณาในรูปที่ 2.3

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



แสดงรูปแบบการแพร่คลื่นของไดโพล  $\lambda/2$  และรีเฟลกเตอร์ในระนาบแนวราบ



แสดงรูปแบบการแพร่คลื่นของไดโพล  $\lambda/2$  และรีเฟลกเตอร์ในระนาบแนวตั้ง

รูปที่ 2.3

จากรูปทั้งสอง เห็นได้ชัดว่า ไดเรกทิวิตี ของอาร์เรย์ดีกว่า ไดโพลอย่างเดียว เหตุที่รีเฟลกเตอร์มีผลต่อการเปลี่ยนแปลงนี้ อธิบายได้ว่า เมื่อเราป้อนแรงดันไฟฟ้า (ที่ความถี่เรโซแนนซ์) และกระแสให้กับไดโพลจะมีการแพร่คลื่นแม่เหล็กไฟฟ้าไปทุกทิศในแนวตั้งฉากกับไดโพล พลังงานบางส่วนมาที่รีเฟลกเตอร์ และเหนี่ยวนำสนามแม่เหล็กไฟฟ้าเกิดขึ้น ซึ่งมีเฟสตามหลังแรงดันไฟฟ้าส่วนที่ป้อนให้ไดโพลอยู่โดยคิดจากระยะห่างของอีลีเมนต์ ถ้าความยาวของรีเฟลกเตอร์และระยะห่างระหว่างไดโพล/รีเฟลกเตอร์ถูกเลือกมาอย่างเหมาะสมแล้วพลังงานที่แพร่มาจากรีเฟลกเตอร์จะไปเสริมในส่วนของไดโพลในทิศทางที่ต้องการ

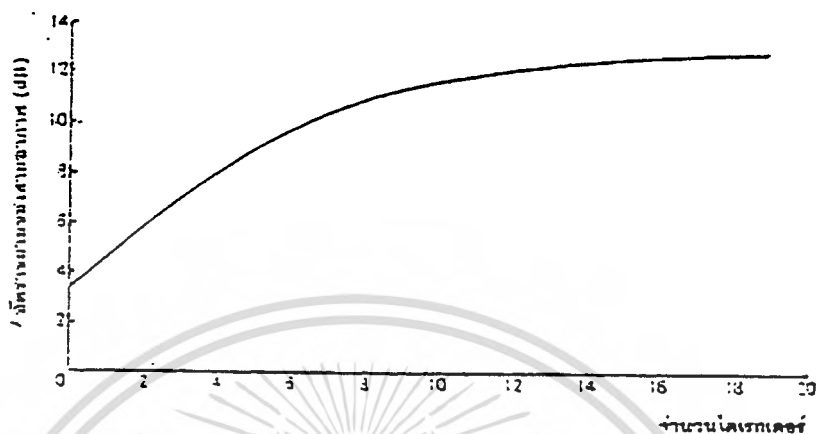
การเพิ่มค่าไดเรกทิวิตีและอัตราขยายของไดโพลสามารถทำได้โดยเพิ่มพาราซิติกอีลีเมนต์อันใหม่ลงไปโดยวางในตำแหน่งตรงข้ามกับรีเฟลกเตอร์เราเรียกอีลีเมนต์ใหม่ว่าไดเรกเตอร์ (Director) ที่มีขนาดสั้นกว่า  $\lambda/2$  อยู่ประมาณ 5% ขณะที่ไดโพลแพร่คลื่นจะมีบางส่วนเหนี่ยวนำให้ไดเรกเตอร์สามารถแพร่คลื่นได้เช่นเดียวกับรีเฟลกเตอร์

การพิจารณาเลือกความยาวของไดเรกเตอร์และระยะห่างระหว่างไดโพล/ไดเรกเตอร์ นับว่าสำคัญมากเพราะถ้าเลือกค่าที่ถูกต้องพลังงานที่แพร่จากไดเรกเตอร์จะไปเสริมส่วนของไดโพลเป็นการเพิ่มค่าไดเรกทิวิตีและอัตราขยายมากขึ้น

การเพิ่มค่าไดเรกทิวิตี หรืออัตราขยายของสายอากาศให้มากกว่านี้ไม่อาจทำได้โดยเพิ่มรีเฟลกเตอร์ตัวที่สองลงไปเพราะว่าสนามแม่เหล็กหลังรีเฟลกเตอร์ตัวแรกมีค่าอ่อน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น เมื่ออนุญาตให้มาใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

มากจนนำมาเหนียวน่า ไม่ได้แค่การเพิ่มโคเรกเตอร์ให้มากจำนวนขึ้นมีผลให้ค่าอัตราขยายของสายอากาศเพิ่มได้จริงดังรูปกราฟ



รูปที่ 2.4 กราฟแสดงความสัมพันธ์ของจำนวนโคเรกเตอร์กับอัตราขยายของสายอากาศภายในทิศที่มีการแพร่มากที่สุด

ในทางปฏิบัติการเลือกค่าระยะห่างระหว่างอีลีเมนต์ต้องคำนึงถึง 2 สิ่ง คือ

1. อัตราขยายที่ต้องการของอาร์เรย์
2. อัตราส่วนฟรอนต์ทูแบคที่ต้องการ

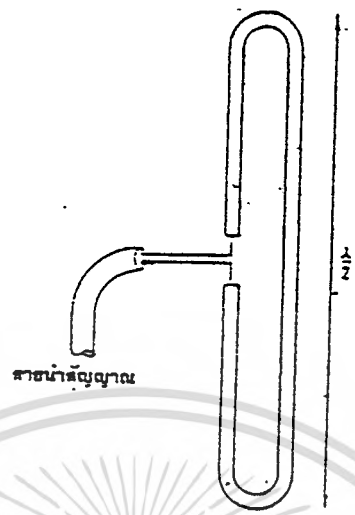
โดยทั่วไปแล้วระยะห่างระหว่างโคโพล/รีนฟลิกเตอร์ มีค่าระหว่าง  $0.15\lambda$  -  $0.25\lambda$  และ ระยะห่างระหว่างโคโพล/โคเรกเตอร์มีค่าระหว่าง  $0.1\lambda$  -  $0.15\lambda$

โคโพลแบบห่อ (Folded dipole)

ค่าอินพุทอิมพีแดนซ์ของโคโพลแบบ  $\lambda/2$  ที่รีโซแนนซ์มีค่าเท่ากับ 73 โอห์มในขณะเพิ่มพาราดิก อีลีเมนต์จะมีผลลดค่าอิมพีแดนซ์ลงอาจเหลือแค่ 50 โอห์มสำหรับการมีรีเฟลคเตอร์และโคเรกเตอร์อย่างละหนึ่งอัน หรือเหลือ 20 โอห์มถ้ามีโคเรกเตอร์หลายอัน แต่สายนำสัญญาณที่ใช้กับอาร์เรย์แบบยาก็จะมีค่าอิมพีแดนซ์มาตรฐานคือ 50 โอห์ม หรือ 75 โอห์มถ้าสายอากาศที่ใช้งานไม่แมชท์หรือมีค่าอิมพีแดนซ์ไม่เท่ากับสายนำสัญญาณ อาจเกิดคลื่นนิ่งหรือสแตนด์เวฟบนสายนำสัญญาณเป็นการสูญเสียพลังงานของระบบ ทางแก้ปัญหานี้คือต้องเพิ่มค่าอิมพีแดนซ์ของโคโพลให้มากขึ้นจากเดิมเพื่อว่าเวลาใช้งานกับพาราดิก อีลีเมนต์ค่าอิมพีแดนซ์ที่ลดลงมายังมีโอกาสเท่ากับค่า 50 โอห์ม หรือ 75 โอห์มของสายนำสัญญาณได้

เอกสารนี้เป็นเอกสารที่สโคโพลที่มีค่าอิมพีแดนซ์สูงกว่าค่าเดิมเราใช้โคโพลแบบห่อแก้ปัญหา  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

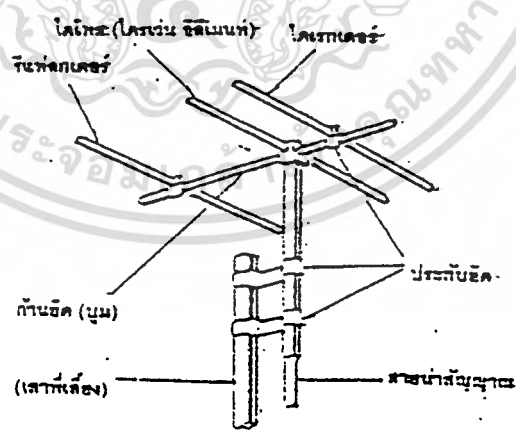
นี้แสดงในรูป



รูปที่ 2.5 แสดงไดโพลแบบห่วงขนาด  $\lambda/2$

ค่าอิมพีแดนซ์ของไดโพลแบบห่วงมีขนาดเป็นสี่เท่าของไดโพลธรรมดา นั่นคือมีค่าเท่ากับ  $4 \times 37 = 148$  โอห์ม นอกจากนี้ยังมีวิธีที่ทำให้จำนวนเท่าของอิมพีแดนซ์มากกว่านี้ได้โดยการให้แต่ละครึ่งของไดโพลแบบห่วงใช้ตัวนำที่มีเส้นผ่าศูนย์กลางต่างกัน

แถบความถี่ของสายอากาศแบบยาก็ยังมีค่าเพิ่มขึ้นด้วยเมื่อใช้ไดโพลแบบห่วง สายอากาศแบบยาก็ที่เป็นมาตรฐานแสดงไว้ดังรูป



รูปที่ 2.6 แสดงสายอากาศแบบยาก็ที่ใช้งานจริง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การทำงานของสายอากาศแบบยาก็มีตัวแปร 2 อย่างที่ควรคำนึงถึงอย่างมากคือ

1. ความยาวของอีลีเมนต์แต่ละอัน

2. ระยะห่างระหว่างอีลีเมนต์

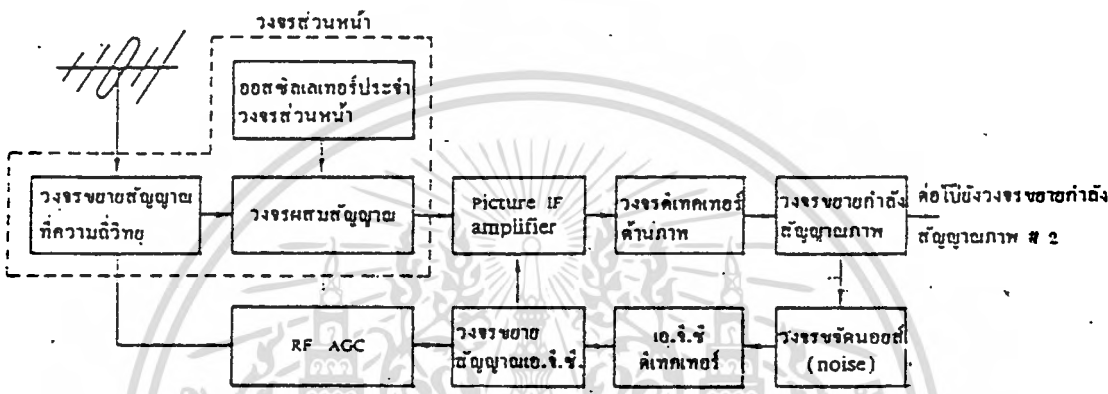
ทั้งสองข้อนี้มีหน่วยเป็นความยาวคลื่นดังนั้นการใช้งานควรอยู่ในย่านความถี่ VHF และ UHF ส่วนในย่าน HF และ MF ไม่ควรใช้สายอากาศแบบยาก็ เพราะขนาดโครงสร้างของสายอากาศจะใหญ่มาก สายอากาศยาก็มีใช้มากในงานรับสัญญาณโทรทัศน์ตามบ้านซึ่งเห็นได้ทั่วไปบนหลังคา หรือนำไปใช้ในการรับ/ส่ง ระบบวิทยุโทรทัศน์แบบจุดต่อจุด ย่าน VHF ได้



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.2 วงจรควบคุมการขยายสัญญาณอัตโนมัติ (AGC or automatic gain control circuit )

รูปที่ 2.7 แสดงแผนผังของวงจรควบคุมการขยายสัญญาณอัตโนมัติ โดยการนำสัญญาณจากเอาต์พุตของวงจรดีเทกเตอร์ด้านภาพผ่านวงจรจัดเสียงรบกวนไปยังวงจร เอ.จี.ซี. ในกรณีที่สัญญาณภาพมีระดับไม่ปกติ โดยอาจเพิ่มขึ้นหรือลดลงไม่แน่นอนซึ่งขึ้นอยู่กับความแรงของสัญญาณโทรทัศน์ที่รับได้ ก็จะทำให้เกิด เอ.จี.ซี. โวลเทจขึ้นเป็นบวกหรือเป็นลบแล้วแต่ลักษณะของ



รูปที่ 2.7 แผนผังของวงจร เอ.จี.ซี



รูปที่ 2.8 ความสัมพันธ์ระหว่างกระแสคอลเลกเตอร์กับค่าองการขยายสัมพัทธ์

การเปลี่ยนแปลงระดับของสัญญาณภาพ และจะใช้สัญญาณ เอ.จี.ซี. โวลเทจนี้เป็นตัวควบคุมการขยายของสัญญาณของวงจรขยายคลื่นพาห์ของภาพที่มีความถี่ไอ.เอฟ.กับวงจรขยายสัญญาณที่มีความถี่วิทยุ ซึ่งอยู่ในวงจรส่วนหน้า อันจะช่วยทำให้สามารถรักษาระดับของสัญญาณภาพให้คงที่หรือเกือบคงที่ตลอดเวลา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การทำงานของวงจรเอ.จี.ซี หรือวงจรควบคุมการขยายสัญญาณอัตโนมัติ อาศัยคุณลักษณะการขยายของทรานซิสเตอร์ตามที่แสดงไว้ในรูปที่ 2.7 ซึ่งเป็นคุณลักษณะความสัมพันธ์ของแกน หรือการขยายกับกระแสคอลเลคเตอร์ แกนจะมีค่าสูงสุดที่ค่าของกระแสคอลเลคเตอร์หนึ่ง ส่วนที่ค่ากระแสคอลเลคเตอร์อื่นๆ แกนจะลดลงเป็นส่วนโค้ง ในส่วนที่กระแสคอลเลคเตอร์มีค่าน้อยๆ แกนจะมีค่าสูงขึ้นจนถึงค่าสูงสุดที่ค่ากระแสคอลเลคเตอร์ค่าหนึ่ง ย่านการทำงานนี้เรียกว่า ย่านย้อนหลัง (reverse region) และการทำงานของวงจร เอ.จี.ซี ในย่านนี้ก็เรียกว่า reverse type AGC หลังจากแกนขึ้นสูงสุดที่กระแสคอลเลคเตอร์ค่าหนึ่งแล้ว เมื่อเพิ่มค่าของกระแสคอลเลคเตอร์ แกนก็จะลดลง การทำงานในย่านนี้เรียกว่า ย่านปกติ (forward region) และการทำงานของวงจร เอ.จี.ซี ในย่านนี้ก็ เรียกว่า forward type AGC โดยทั่วไป การทำงานของวงจร เอ.จี.ซี มักอยู่ในย่านปกติ

วงจรเอ.จี.ซี หรือวงจรควบคุมการขยายสัญญาณอัตโนมัติ มีอยู่ 3 ชนิด คือ

### 1. วงจรเอ.จี.ซี ที่ทำงานตามค่าเฉลี่ยของสัญญาณภาพ

วงจรนี้มีลักษณะเช่นเดียวกับวงจรเรกติไฟเออร์แบบครึ่งคลื่น (half wave rectifier) ซึ่งใช้แอล.ซี. ฟิลเตอร์ (LC filter) การทำงานของวงจรขึ้นอยู่กับค่าเฉลี่ยของสัญญาณภาพ เนื่องด้วยเป็นวงจรร่ายๆ จึงมีนอยส์พัลส์ (noise pulses) เข้ามารบกวนสัญญาณภาพได้ง่าย และหากสัญญาณภาพในห้องส่งโทรทัศน์เปลี่ยนแปลง ก็จะทำให้ค่าของเอ.จี.ซี โวลเทจเปลี่ยนแปลงไปด้วย อันจะมีผลทำให้คอนทราสต์ (contrast) ของภาพไม่คงที่ตลอดเวลา ในปัจจุบันนี้ วงจรแบบนี้จึงไม่นิยมใช้

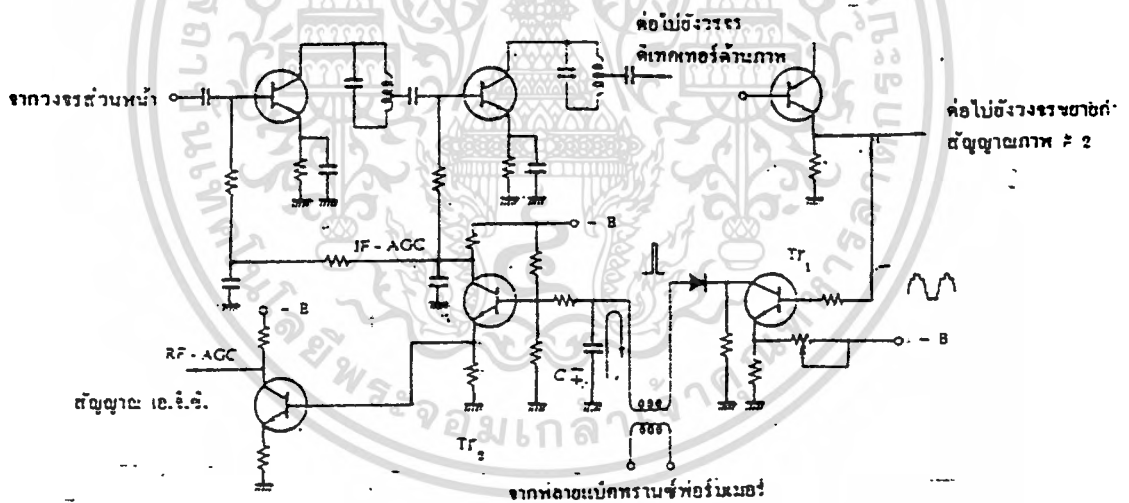
### 2. วงจรเอ.จี.ซี ที่ทำงานตามค่าสูงสุดของสัญญาณภาพ

ในขณะที่สัญญาณภาพเปลี่ยนแปลงไปตามภาพจริงๆ ในห้องส่งโทรทัศน์ ค่าของสัญญาณซึ่งจะยังคงที่ในระดับเดิมไม่เปลี่ยนแปลง เนื่องจากระดับของสัญญาณซึ่งนี้ มีระดับตรงกับระดับค่าสูงสุดของสัญญาณภาพ ฉะนั้น เพื่อป้องกันไม่ให้เอ.จี.ซี โวลเทจ เปลี่ยนแปลงไปตามสัญญาณภาพ จึงหาวิธีทำให้เอ.จี.ซี โวลเทจนี้ เป็นสัดส่วนโดยตรงกับค่าสูงสุดของสัญญาณภาพซึ่งเป็นค่าคงที่ แทนที่จะใช้ค่าเฉลี่ยมาคำนวณการ แต่อย่างไรก็ตาม หากปรากฏมีนอยส์พัลส์รวมอยู่ด้วย และนอยส์พัลส์นี้มีค่าสูงกว่าระดับของสัญญาณซึ่งค์ จะทำให้รบกวนสัญญาณของเอ.จี.ซี โวลเทจได้เหมือนกัน จึงจำเป็นต้องเพิ่มวงจรจับนอยส์ขึ้นในเส้นทางเดินของสัญญาณก่อนถึงวงจรเอ.จี.ซี ดีเทคเตอร์

### 3. วงจรคีย์ เอ.จี.ซี

วงจรนี้แสดงไว้ในรูปที่ 2.9 โดยใช้สัญญาณพัลส์เส้นสะแกนสะบัดกลับทางแนว  
นอกจากฟลายแบ็คทรานส์ฟอर्मเมอร์ มาายังยังคอลเลคเตอร์ของทรานซิสเตอร์ TR1 เพื่อให้  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ทรานซิสเตอร์ TR1 ทำงานในขณะที่มีพัลส์เส้นสะแกนสะบัดกลับทางแนวนอน ซึ่งจังหวะที่มีพัลส์เส้นสะแกนสะบัดกลับทางแนวนอนนี้ เป็นช่วงเวลาที่มิสัญญาณเชิงพัลส์ทางแนวนอนเนื่องจากวงจรนี้จะทำงานพร้อมกันกับการเกิดซิงค์พัลส์ทางแนวนอนจึงเรียกววงจรนี้ว่า วงจรคีย์เอ.จี.ซี (keyed AGC circuit) การทำงานวงจรคีย์เอ.จี.ซี มีดังนี้คือ ในขณะที่ทรานซิสเตอร์ TR1 ทำงานกระแสคอลเลคเตอร์ของทรานซิสเตอร์ TR1 จะไหลผ่านตัวเก็บประจุ C ตามที่แสดงไว้ในรูป โวลเทจของตัวเก็บประจุ C นี้จะป้อนเข้าขาเบสของทรานซิสเตอร์ TR2 ซึ่งจะทำให้เกิดเอ.จี.ซี โวลเทจขึ้นทางด้าน อิมิตเตอร์หากเอาท์พุทโวลเทจของวงจรดีเทคเตอร์ทางด้านภาพมีค่าสูงขึ้น จะทำให้กระแสคอลเลคเตอร์ของทรานซิสเตอร์ TR1 มีค่าสูงขึ้น และโวลเทจที่ตัวเก็บประจุจะมากขึ้นด้วย ซึ่งจะทำให้ที่ค่า เอ.จี.ซี โวลเทจมากขึ้น เมื่อป้อนสัญญาณเอ.จี.ซี โวลเทจให้วงจรขยายสัญญาณ คลื่นพาห์ของภาพที่มีความถี่ ไอ.เอฟ ซึ่งมีทรานซิสเตอร์ทำงานในย่านปกติ สัญญาณเอ.จี.ซี โวลเทจที่เพิ่มขึ้นก็จะไปทำให้แกนของวงจรขยายสัญญาณลดต่ำลง อันจะช่วยทำให้เอาท์โวลเทจของวงจรดีเทคเตอร์มีค่าลดลงเป็นปกติ วงจรนี้นิยมใช้กันมากอย่างแพร่หลาย เพราะไม่มีน้อยสร์บกวน



รูปที่ 2.9 วงจร คีย์ เอ.จี.ซี

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

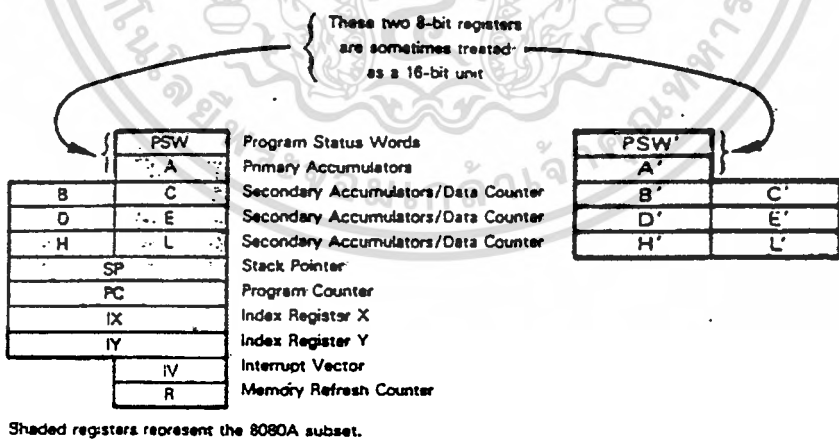
## 2.3 ไมโครโปรเซสเซอร์ Z80

ไมโครโปรเซสเซอร์ Z80 เป็น CPU ที่ผลิตจากบริษัท ZILOG INC. โดยกลุ่มวิศวกรชุดเดียวกับที่ผลิต 8080 CPU ของบริษัท INTEL COOPERATION., Z80 CPU ได้รับการพัฒนาให้มีข้อดีเหนือกว่า 8080 เช่น มีชุดคำสั่งมากถึง 158 คำสั่งโดยรวมชุดคำสั่งเดิมของ 8080 ไว้ 80 คำสั่ง นอกจากนี้ Z80 ยังมีรีจิสเตอร์มากกว่าใน 8080 ถึง 12 ตัว และ 8080 เพียงตัวเดียวก็ยังไม่สามารถที่จะนำไปใช้งานได้ต้องต่อกับอุปกรณ์สนับสนุนอีก 2 ตัวคือ CLOCK GENERATOR CHIP, SYSTEM CONTROLER CHIP รวมกันเรียกว่า THREE CHIP PROCESSOR แต่ใน Z80 CPU ได้รวมเอาลักษณะพื้นฐานเหล่านี้ไว้ในชิพเดียวกัน และเพิ่มประสิทธิภาพทาง HARDWARE, SOFTWARE และการ INTERFACE ให้สูงขึ้น

Z80 ไมโครโปรเซสเซอร์เพียงชิ้นเดียวไม่สามารถทำงานเป็นระบบคอมพิวเตอร์ได้ ต้องอาศัยอุปกรณ์อื่นๆอีก 2 ส่วนคือ หน่วยความจำ (MEMORY) และหน่วยรับส่งข้อมูลเข้าออก (I/O DEVICE) ซึ่งในการทำงานตามคำสั่งจากโปรแกรมที่ป้อนเข้ามา Z80 CPU จะต้องทำการโอนย้ายคำสั่งหรือข้อมูลระหว่างหน่วยความจำ กับรีจิสเตอร์ (REGISTER) ก่อนอื่นเราจะกล่าวถึงรีจิสเตอร์ภายในของ Z80 เสียก่อน

รีจิสเตอร์ต่างๆใน Z80 CPU.

Z80 CPU จะประกอบไปด้วยรีจิสเตอร์ถึง 22 ตัวดังรูป 2.10 รีจิสเตอร์เหล่านี้จะแบ่งได้เป็น 2 กลุ่มคือ รีจิสเตอร์ที่ทำหน้าที่ทั่วไป และ รีจิสเตอร์ที่ทำหน้าที่เฉพาะงาน



รูปที่ 2.10 แสดงรีจิสเตอร์ต่างๆภายใน Z80

1. รีจิสเตอร์ที่ทำหน้าที่ทั่วไป แบ่งเป็นรีจิสเตอร์หลัก ได้แก่ A, B, C, D, E, H เอกสและ L มีความจุขนาด 8 บิต รีจิสเตอร์เหล่านี้ใช้เก็บข้อมูลชั่วคราว นอกจากนี้ยังสามารถรับค่าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ข้อมูลจากหน่วยความจำหรืออาจจะทำการย้ายข้อมูลไปเก็บไว้ในหน่วยความจำก็ได้ และ รีจิสเตอร์สำรองได้แก่ A', B', C', D', H' และ L' ซึ่งเป็นรีจิสเตอร์ที่ทำหน้าที่เก็บข้อมูลที่มาจากรีจิสเตอร์หลักในกรณีที่ต้องใช้รีจิสเตอร์หลักในการทำงานอย่างอื่นก่อน ดังนั้นรีจิสเตอร์กลุ่มนี้จึงไม่สามารถกระทำการทางคณิตศาสตร์และลอจิกได้

รีจิสเตอร์ A เรียกว่า แอคคิวมูเลเตอร์ (ACCUMULATOR) ทำหน้าที่เก็บข้อมูลชั่วคราวที่ได้จากการทำการทางคณิตศาสตร์ เช่น บวกหรือลบข้อมูล 2 จำนวน ผลลัพธ์ที่ได้จะเก็บไว้ในรีจิสเตอร์ A นี้ นอกจากนี้ ในการปฏิบัติตามคำสั่งที่ใช้กับข้อมูลขนาด 16 บิต Z80 จะนำเอารีจิสเตอร์แฟลก "F" (FLAG REGISTER) มาใช้ร่วมกับรีจิสเตอร์ A เรียกว่า คู่รีจิสเตอร์ AF ซึ่งมีขนาด 16 บิต นอกจากนี้ยังมีรีจิสเตอร์ 16 บิต อื่นๆอีก คือ BC, DE และ HL

2. รีจิสเตอร์ที่ใช้งานเฉพาะอย่าง ได้แก่ รีจิสเตอร์ I, R, IX, IY, SP และ PC ซึ่งทำหน้าที่ต่างๆดังนี้

รีจิสเตอร์ I (INTERRUPT PAGE ADDRESS REGISTER) เมื่อมีการอินเทอร์รัพท์เกิดขึ้นจำเป็นต้องบอกตำแหน่งของหน่วยความจำที่เก็บโปรแกรมตอบสนองการอินเทอร์รัพท์ รีจิสเตอร์ I จะทำหน้าที่เก็บค่า 8 บิตบนของตำแหน่งข้อมูลในหน่วยความจำ ส่วนค่า 8 บิตล่างจะป้อนจากภายนอกให้แก่ CPU ค่าทั้งสองจะประกอบกันเป็นค่าแอดเดรสที่ระบุตำแหน่งของโปรแกรมการตอบสนองการอินเทอร์รัพท์

รีจิสเตอร์ R (MEMORY REFRESH REGISTER) เป็นรีจิสเตอร์ขนาด 7 บิตที่ถูกใช้ในการรีเฟรช (REFRESH) DYNAMIC RAM และค่ารีจิสเตอร์ R จะเพิ่มขึ้นเองโดยอัตโนมัติ ในทุกๆครั้งที่มีการเฟรชคำสั่งจากหน่วยความจำ

รีจิสเตอร์ IX และ IY (INDEX REGISTER) เป็นรีจิสเตอร์ที่มีขนาด 16 บิต มีประโยชน์ใช้บ่งบอกตำแหน่งในหน่วยความจำแบบ INDEX ADDRESSING MODE โดยจะกำหนดให้ค่าใน INDEX REGISTER เป็นค่าอ้างอิง แล้วใช้คำสั่งบ่งบอกว่าตำแหน่งของข้อมูลที่ต้องการอยู่ห่างจากค่าอ้างอิงนี้เท่าใด โดยจะบอกค่าระยะห่างในรูปของ TWO COMPLEMENT รีจิสเตอร์ SP (STACK POINTER) มีขนาด 16 บิต ในหน่วยความจำชนิด RAM จะมีส่วนหนึ่งที่ถูกกำหนดให้เป็นที่ยกเก็บข้อมูลชั่วคราว ส่วนนี้เรียกว่าสแตค (STACK) ซึ่งมีลักษณะการเก็บข้อมูลแบบ LIFO (LAST IN FIRST OUT) เราสามารถที่จะเก็บข้อมูลลงบนสแตคโดยใช้คำสั่ง PUSH และเมื่อต้องการดึงข้อมูลออกจากสแตคต้องใช้คำสั่ง POP

รีจิสเตอร์ PC (PROGRAM COUNTER) เป็นรีจิสเตอร์ขนาด 16 บิตที่ใช้ในการเก็บตำแหน่งของหน่วยความจำที่ CPU จะเฟรช (FETCH) คำสั่งหลังจากที่เฟรชคำสั่งเรียบร้อยแล้ว

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์การใช้งานเพื่อการศึกษาเท่านั้น เมื่อผู้ใดเห็นประโยชน์ในการนำเอกสารนี้ไปใช้โดยไม่ได้รับอนุญาตให้สงวนลิขสิทธิ์ไว้ให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ค่าในรีจิสเตอร์ PC จะเพิ่มขึ้น และจะชี้ไปยังตำแหน่งของคำสั่งถัดไป เราจะสามารถเปลี่ยนแปลงค่าใน PC ได้โดยใช้คำสั่ง CALL หรือ JUMP

รีจิสเตอร์ F (FLAG REGISTER) ประกอบด้วย

SIGN FLAG (S) : แฟล็กเครื่องหมาย

ZERO FLAG (Z) : แฟล็กศูนย์

HALF CARRY FLAG (H) : แฟล็กทศครึ่ง

PARITY/OVERFLOW FLAG (P/V) : แฟล็กพาริตีหรือโอเวอร์โฟลว์

SUBTRACT FLAG (N) : แฟล็กลบ

CARRY FLAG (C) : แฟล็กตัวทด

ผู้ผลิต Z80 ได้นำเอาแฟล็กเหล่านี้มาประกอบร่วมกับบิตว่าง (X : ไม่มีความหมาย) อีก 2 บิตเพื่อทำเป็นรีจิสเตอร์ขนาด 8 บิต สำหรับรายละเอียดของแฟล็กเหล่านี้จะไม่บอกกล่าวถึง รายละเอียดของขา Z80 (Z80 PIN OUTS)

A0-A15 (ADDRESS BUS) : เป็นขาสัญญาณเอาต์พุตแบบ TRI-STATE ใช้บ่งบอกตำแหน่งหน่วยความจำได้ถึง  $2^{16} = 65536$  ตำแหน่ง A0-A7 จะแสดงตำแหน่งของพอร์ทที่ Z80 ต้องการติดต่อด้วย นอกจากนี้ขา A0-A6 จะให้ค่ารีเฟรชแอดเดรสออกมาขณะที่ Z80 ให้สัญญาณรีเฟรช

D0-D7 (DATA BUS) : เป็นขาสัญญาณอินพุต/เอาต์พุต TRI-STATE แบบสองทิศทางซึ่งเป็นทางผ่านของข้อมูลระหว่าง Z80 กับหน่วยความจำและอุปกรณ์ I/O

/M1 (MACHINE CYCLE ONE) : เป็นขาเอาต์พุตแอคทีฟที่ล่อจิก "0" ขา /M1 นี้จะแอคทีฟขณะที่ Z80 ทำการเฟตช์ออฟโกลด์ของคำสั่ง ในกรณีที่คำสั่งที่จะเฟตช์เข้ามานั้นมีขนาด 2 ไบต์ M1 จะแอคทีฟในทุกๆ ไซเคิลการเฟตช์แต่ละไบต์

/MREQ (MEMORY REQUEST) : เป็นสายเอาต์พุตแบบ TRI-STATE แอคทีฟที่ล่อจิก "0" เพื่อเป็นการบ่งบอกว่า Z80 กำลังกระทำการติดต่อกับหน่วยความจำ

/IORQ (INPUT/OUTPUT REQUEST) : เป็นสายเอาต์พุตแบบ TRI-STATE จะแอคทีฟที่ล่อจิก "0" เพื่อเป็นการบ่งบอกว่า Z80 กำลังทำการติดต่อกับอุปกรณ์ I/O และเมื่อ /IORQ และ /M1 แอคทีฟทั้งคู่จะเป็นการบ่งบอกการตอบรับการอินเทอร์รัพท์ (INTERRUPT ACKNOWLEDGE)

/RD (MEMORY READ) : เป็นขาเอาต์พุต TRI-STATE จะแอคทีฟที่ล่อจิก "0" เมื่อ Z80 ต้องการอ่านข้อมูลจากหน่วยความจำหรืออุปกรณ์ I/O และ Z80 จะรับข้อมูลจากบัสข้อมูลเข้าไปเมื่อสัญญาณนี้เปลี่ยนระดับลอจิกจาก "0" เป็น "1" ไม่นานนักให้นำไปใช้ประโยชน์ด้านการคำนวณที่ถี่ขึ้น อีกทั้งยังมีให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

/WR (MEMORY WRITE) : เป็นขาเอาต์พุตแบบ TRI-STATE จะแอกทีฟที่ลอจิก "0" เมื่อ Z80 ต้องการส่งข้อมูลออกไปให้หน่วยความจำหรืออุปกรณ์ I/O

/RFSH (REFRESH) : เป็นขาเอาต์พุต จะแอกทีฟเมื่อ 7 บิตล่าง (A0-A6) ของบัสแอดเดรสให้ค่ารีเฟรชออกมา

/HALT (HALT STATE) : เป็นขาเอาต์พุต จะแอกทีฟที่ลอจิก "0" เมื่อ Z80 อยู่ในสถานะของการ /HALT คือ CPU จะทำคำสั่ง NOP (NO OPERATION) เพื่อให้เกิดการรีเฟรชได้และ Z80 จะหลุดพ้นจากสถานะการ HALT เมื่อได้รับการรีเซ็ตหรือถูกอินเทอร์รัพท์

/WAIT : เป็นขาอินพุตแอกทีฟที่ลอจิก "0" และจะมีการตรวจสอบสัญญาณนี้ที่ขอบล่างของคล็อกลูกที่ 2 ของทุกๆ MACHINE CYCLE เมื่อมีการตรวจพบว่าขาอินพุตนี้แอกทีฟจะมีการแทรก WAIT STATE ให้กับแต่ละ MACHINE CYCLE เพื่อเป็นการรออุปกรณ์ภายนอกทำงานให้ทันกับการทำงานของ Z80 และ Z80 จะแทรก WAIT STATE จนกว่าจะมีการตรวจสอบพบว่าขา WAIT จะมีลอจิกเป็น "1"

/INT (INTERRUPT REQUEST) : เป็นขาอินพุตแอกทีฟที่ลอจิก "0" Z80 จะตรวจสอบสัญญาณที่ขานี้ทุกๆการสิ้นสุดของ INSTRUCTION CYCLE (LAST STATE)

/NMI (NON MASKABLE INTERRUPT) : เป็นขาอินพุตแอกทีฟที่ลอจิก "0" สัญญาณ NON MASKABLE INTERRUPT ที่มีระดับความสำคัญในการขออินเทอร์รัพท์สูงกว่าสัญญาณ INTERRUPT REQUEST Z80 จะตอบรับการอินเทอร์รัพท์ชนิดนี้เสมอ โดยที่ไม่สามารถ DISABLE ได้ด้วย SOFTWARE

/RESET : เป็นขาอินพุตที่แอกทีฟที่ลอจิก "0" สัญญาณนี้จะทำการ INITIALIZE CPU โดยทำการรีเซ็ต INTERRUPT FLIP-FLOP และ เซ็ตค่าในโปรแกรมเคาท์เตอร์ให้เป็น 0000H และในสถานะการรีเซ็ตนี้บัสแอดเดรสและบัสข้อมูลจะอยู่ในสถานะ HIGH IMPEDENCE และสัญญาณควบคุมต่างๆจะอยู่ในสถานะ INACTIVE

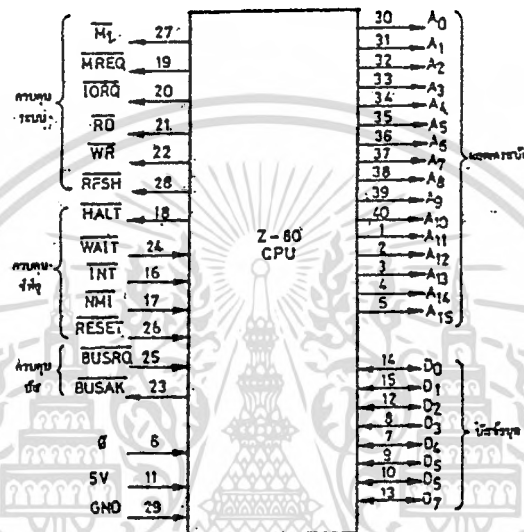
/BUSRQ (BUS REQUEST) : เป็นขาอินพุตแอกทีฟที่ลอจิก "0" สัญญาณ BUS REQUEST เป็นสัญญาณที่มีระดับความสำคัญสูงกว่าสัญญาณ NON MASKABLE INTERRUPT และมีการตรวจสอบสัญญาณนี้ทุกๆ การสิ้นสุดของ MACHINE CYCLE อุปกรณ์ภายนอกจะให้สัญญาณนี้แก่ Z80 เมื่อต้องการใช้บัสข้อมูลและบัสแอดเดรสโดยเปรียบเสมือนว่าเป็นการถอด Z80 ออกจากระบบบัส

/BUSAK (BUS ACKNOWLEDGE) : เป็นขาเอาต์พุตแอกทีฟที่ลอจิก "0" ขานี้จะแอกทีฟเมื่อ Z80 ตอบสนองการต่อสัญญาณ BUS REQUEST และจะทำให้บัสข้อมูล, บัสควบคุม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

และบัสแอดเดรสมีสถานะเป็น HIGH IMPEDENCE ซึ่งทำให้อุปกรณ์ภายนอกใช้บัสเหล่านี้โดยไม่มีผลต่อ CPU

จากที่กล่าวมาทั้งหมดเป็นรายละเอียดที่ควรทราบ และนำ Z80 ไปใช้งานร่วมกับอุปกรณ์สนับสนุนอื่นๆ(CHIP SUPPORT) ในหัวข้อต่อไป



รูปที่ 2.11 ลักษณะของขาไอซี Z-80 CPU

### 2.3.1 การใช้ ROM กับ Z80

ในหัวข้อนี้จะเป็นการศึกษาว่าจะใช้ไมโครโปรเซสเซอร์ Z80 ร่วมกับหน่วยความจำชนิด ROM ได้อย่างไร โดยจะกล่าวถึงชนิดของ ROM และการใช้งานก่อนต่อจากนั้นจะกล่าวถึงการนำ ROM มาต่อร่วมกับไมโครโปรเซสเซอร์ Z80

ในระบบไมโครโปรเซสเซอร์ จะมีหน่วยความจำชนิดหนึ่งที่ใช้สำหรับเก็บข้อมูลโดยที่ข้อมูลเหล่านี้ไม่สูญหายเมื่อไม่มีการจ่ายไฟเลี้ยงแก่ระบบ หน่วยความจำชนิดนี้ก็คือ ROM (READ ONLY MEMORY) นั่นเองซึ่งข้อมูลที่เก็บอยู่ใน ROM สามารถอ่านออกมาได้ แต่ไม่สามารถเขียนข้อมูลเข้าไปได้ ประโยชน์ของ ROM ในระบบไมโครโปรเซสเซอร์ก็คือ ทำให้ CPU ของระบบสามารถที่จะเริ่มทำงานโดยการกำหนดสถานะของอุปกรณ์ฮาร์ดแวร์ต่างๆให้อยู่ในสถานะที่พร้อมจะทำงานได้ เมื่อเราเริ่มจ่ายไฟเลี้ยงให้แก่ระบบ

เอกสารนี้เป็นเอกสารลิขสิทธิ์ของสถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หน่วยความจำที่เก็บข้อมูลไว้ได้โดยไม่สูญหาย ที่ใช้ในระบบไมโครโปรเซสเซอร์มีหลายชนิดเช่น

ROM (READ ONLY MEMORY) : ข้อมูลทั้งหมดที่อยู่ใน ROM จะถูกโปรแกรมโดยผู้ผลิต เราจะใช้ ROM เมื่อมีข้อมูลที่ไม่เปลี่ยนแปลงและมีความต้องการใช้งานเป็นจำนวนมาก

PROM (PROGRAMMABLE READ ONLY MEMORY) : ข้อมูลที่ต้องการโปรแกรมจะถูกโปรแกรมโดยผู้ใช้เองโดยป้อนพัลส์ที่มีแรงดันสูง (HIGH VOLTAGE PULSED) ทำให้ METAL STRIPS หรือ POLYCRYSTALLINE SILICON ที่อยู่ในตัว IC ขาดออกจากกันทำให้เกิดเป็นลอจิก "1" หรือ "0" ตามตำแหน่งที่กำหนดในหน่วยความจำนั้นๆ เมื่อ PROM ถูกโปรแกรมแล้วข้อมูลภายในจะไม่สามารถเปลี่ยนแปลงได้ หน่วยความจำชนิดนี้จะใช้เมื่อต้องการความเร็วในการทำงานสูง

EPROM (ERASABLE PROGRAMMABLE ROM) : ข้อมูลจะถูกโปรแกรมโดยผู้ใช้โดยการให้สัญญาณที่มีแรงดันสูงผ่านเข้าไปในตัว EPROM ซึ่งเป็นวิธีเดียวกับที่ใช้ใน PROM แต่ข้อมูลที่อยู่ใน EPROM เปลี่ยนแปลงได้ โดยการลบข้อมูลเดิมที่มีอยู่ใน EPROM ออกก่อนแล้วค่อยโปรแกรมเข้าไปใหม่ การลบข้อมูลนี้ทำได้โดยการฉายแสงอุลตราไวโอเลตเข้าไปในตัวไอซี โดยผ่านทางกระจกใสที่อยู่บนตัว IC

EAROM (ELECTRICALLY ALTERABLE ROM) : ข้อมูลจะถูกโปรแกรมโดยผู้ใช้เหมือนใน EPROM แต่สิ่งที่แตกต่างกันก็คือข้อมูลของ EAROM สามารถลบได้โดยทางไฟฟ้า ไม่ใช่ด้วยการฉายแสงเหมือนแบบ EPROM

หน่วยความจำที่ได้กล่าวมาแล้วนี้เป็นหน่วยความจำที่ข้อมูลไม่สูญหายเมื่อไม่มีไฟเลี้ยงและสามารถอ่านข้อมูลมาใช้ได้เพียงอย่างเดียวเท่านั้น

### ขั้นตอนในการอ่านข้อมูลจาก ROM

ลำดับขั้นตอนในการอ่านข้อมูลแต่ละครั้งออกจาก ROM ดังแสดงไว้ข้างล่างนี้

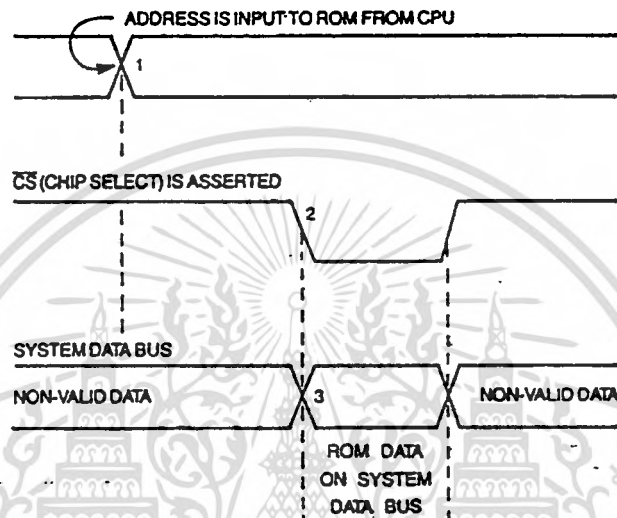
1. คำแอดเดรสจะถูกป้อนเข้าไปยัง ROM (โดย CPU) คำแอดเดรสดจะกำหนดตำแหน่งของข้อมูลที่ต้องการอ่าน โดยข้อมูลจะถูกอ่านออกมาเพียงครั้งละ 1 ไบต์เท่านั้น

2. CPU จะคอยอยู่ช่วงเวลาหนึ่ง (WAIT STATE) เรียกว่า ACCESS TIME ประมาณ 100-300 nanoseconds ขึ้นอยู่กับชนิดของ ROM ซึ่งเป็นเวลาที่ ROM ใช้ในการถอดรหัสของแอดเดรส ของข้อมูลที่ต้องการจะอ่านออกมาทางเอาต์พุตของ ROM

3. CHIP SELECT LINE จะถูกทำให้แอคทีฟ เพื่อให้ข้อมูลออกมาที่บัสของระบบได้ ต่อจากนั้น CPU จะสตrobe (STROBE) ข้อมูลเข้าไปเก็บไว้ในรีจิสเตอร์ภายใน CPU ไม่ว่าจะเป็นกรณีใด ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4. CHIP SELECT LINE จะถูกสั่งให้เลิกทำงาน(สถานะ INACTIVE) เพื่อให้ข้อมูลที่อยู่บนบัสข้อมูลของระบบหายไป

ลำดับการทำงานที่หายไปทีกล่าวนานี้ จะเกิดทุกครั้งที CPU อ่านข้อมูลจาก ROM ในรูปที่ 2.12 แสดงไคอะแกรมเวลา (TIMING DIAGRAM) ที่หายไปของลำดับการทำงานนี้



รูปที่ 2.12 แสดงไคอะแกรมเวลาของสัญญาณที่ใช้ในการอ่านข้อมูลจากหน่วยความจำประเภท ROM

1. CPU ส่งค่าแอดเดรสให้กับ ROM
2. สัญญาณ CHIP SELECT ถูกส่งให้กับ ROM
3. ข้อมูลจาก ROM ถูกส่งออกมาบนบัสข้อมูล

การต่อ ROM เข้ากับ Z80 บัส

รูปที่ 2.13 แสดงถึงการต่อบัสแอดเดรสและบัสข้อมูลของ Z80 เข้ากับ ROM โดยตรง จะสังเกตได้ว่าเรายังไม่ต่อ CHIP SELECT LINE ของ EPROM เบอร์ 2716 เข้ากับระบบ ทั้งนี้เพราะเราจะต้องพิจารณาผังแอดเดรส (ADDRESS MAPPING) เสียก่อน

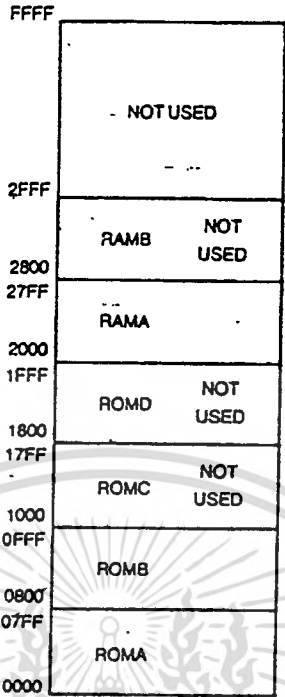
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



### ผังแอดเดรส (ADDRESS MAPPING)

ก่อนที่จะทำการสร้างระบบทางฮาร์ดแวร์ผู้ออกแบบจำเป็นต้องกำหนดโครงสร้างของผังหน่วยความจำเสียก่อน เนื่องจากว่าผังหน่วยความจำเป็นสิ่งที่บอกให้ทราบว่าตำแหน่งของ ROM, RAM และอุปกรณ์ I/O อยู่ที่ไหน ในรูปที่ 2.14 ได้แสดงผังหน่วยความจำ จะเห็นว่าในรูปนี้เราแบ่งตำแหน่งใน 65,536 ตำแหน่งออกเป็นกลุ่มตามหน้าที่ของการทำงานและในกลุ่มของ ROM และ RAM ที่แสดงไว้นี้ ได้เพื่อไว้สำหรับที่เฉพาะที่อาจจำเป็นต้องใช้ด้วย

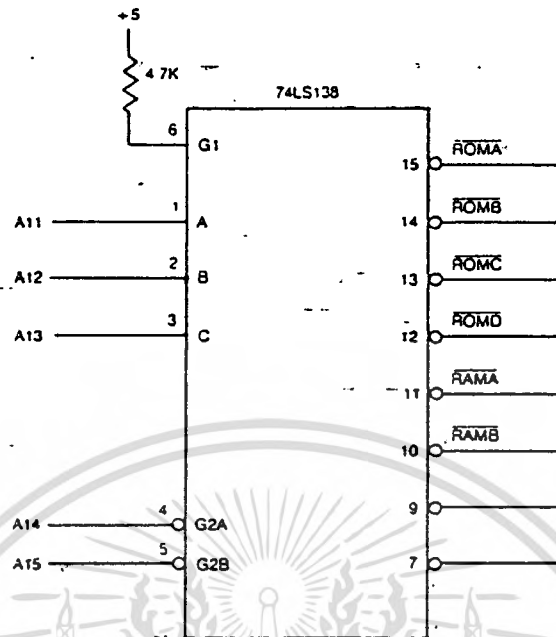
036993



รูปที่ 2.14 ผังหน่วยความจำของระบบไมโครโปรเซสเซอร์ ซึ่งเราใช้กำหนดพื้นที่ของหน่วยความจำแต่ละตัว โดยทั่วไปจะถูกออกแบบไว้ก่อนที่จะสร้างฮาร์ดแวร์ของระบบ

การสร้างสัญญาณ CHIP SELECT สำหรับ ROM

รูปที่ 2.14 จะเห็นได้ว่า ROM A จะถูกเลือกเมื่อ Z80 มีเอาต์พุตแอดเดรสอยู่ระหว่าง 0000H กับ 07FFH ฉะนั้นระบบของเราจึงจำเป็นต้องมีสัญญาณที่ใช้เป็นตัวเลือกแอดเดรสให้อยู่ในช่วงที่เราต้องการ



รูปที่ 2.15 วงจรที่ใช้เลือกอุปกรณ์หน่วยความจำที่กำหนดบนแผงหน่วยความจำ

จากรูปที่ 2.15 ได้แสดงวงจรที่ใช้เลือกแอดเดรสของหน่วยความจำซึ่งเมื่อ A11,A12, A13,A14 และ A15 มีลอจิก “0” จะทำให้ขา 15 ของ IC 74LS138 มีลอจิกเป็น “0” (ROM A ถูกเลือก) และขา 15 จะเป็น “0” ภายในช่วงของแอดเดรสที่กำหนดไว้เท่านั้น ในตารางรูปที่ 2.16 ได้แสดงถึงการตอบสนองของขาเอาต์พุต IC 74LS138 เมื่อมีค่าแอดเดรสบนบัสแอดเดรสในช่วงต่างๆกัน

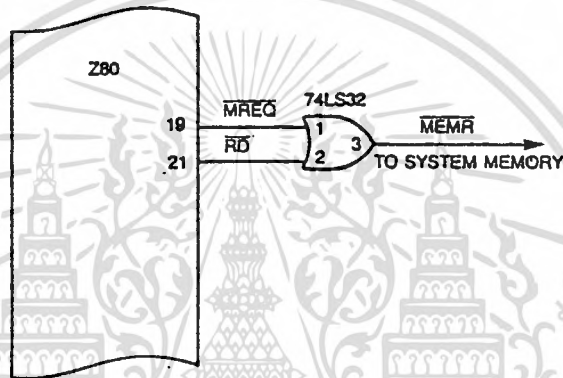
| A15 | A14 | A13 | A12 | A11 | A10 | A0 | HEX  | PIN #=0 |
|-----|-----|-----|-----|-----|-----|----|------|---------|
| 0   | 0   | 0   | 0   | 0   | 0   | 0  | 0000 | 15      |
| 0   | 0   | 0   | 0   | 0   | 1   | 1  | 07FF | 15      |
| 0   | 0   | 0   | 0   | 1   | 0   | 0  | 0800 | 14      |
| 0   | 0   | 0   | 0   | 1   | 1   | 1  | 0FFF | 14      |
| 0   | 0   | 0   | 1   | 0   | 0   | 0  | 1000 | 13      |
| 0   | 0   | 0   | 1   | 0   | 1   | 1  | 17FF | 13      |
| 0   | 0   | 0   | 1   | 1   | 0   | 0  | 1800 | 12      |
| 0   | 0   | 0   | 1   | 1   | 1   | 1  | 1FFF | 12      |
| 0   | 0   | 1   | 0   | 0   | 0   | 0  | 2000 | 11      |
| 0   | 0   | 1   | 0   | 0   | 1   | 1  | 27FF | 11      |
| 0   | 0   | 1   | 0   | 1   | 0   | 0  | 2800 | 10      |
| 0   | 0   | 1   | 0   | 1   | 1   | 1  | 2FFF | 10      |

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับรูปที่ 2.16 แสดงผังหน่วยความจำของรูป 2.15 ให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### การสร้างสัญญาณสำหรับอ่านหน่วยความจำ

ในส่วนนี้เราจะมาดูวิธีการที่จะต่อขา CHIP SELECT INPUT เข้ากับ Z80 ในขั้นแรกจะต้องศึกษาเกี่ยวกับสัญญาณเวลาที่ใช้ในการควบคุมที่ส่งจาก Z80 เพื่อใช้ในการเริ่มและหยุดการส่งข้อมูล และเรียกสัญญาณนี้ว่า MEMORY READ (/MEMR) สัญญาณ /MEMR และสัญญาณ MEMORY SELECT จะทำงานร่วมกันเพื่อที่จะนำข้อมูลที่อยู่ใน ROM ออกไปที่บัตข้อมูลในช่วงเวลาที่ถูกต้อง

ในรูปที่ 2.17 เป็นวิธีหนึ่งที่ใช้ในการสร้างสัญญาณ /MEMR

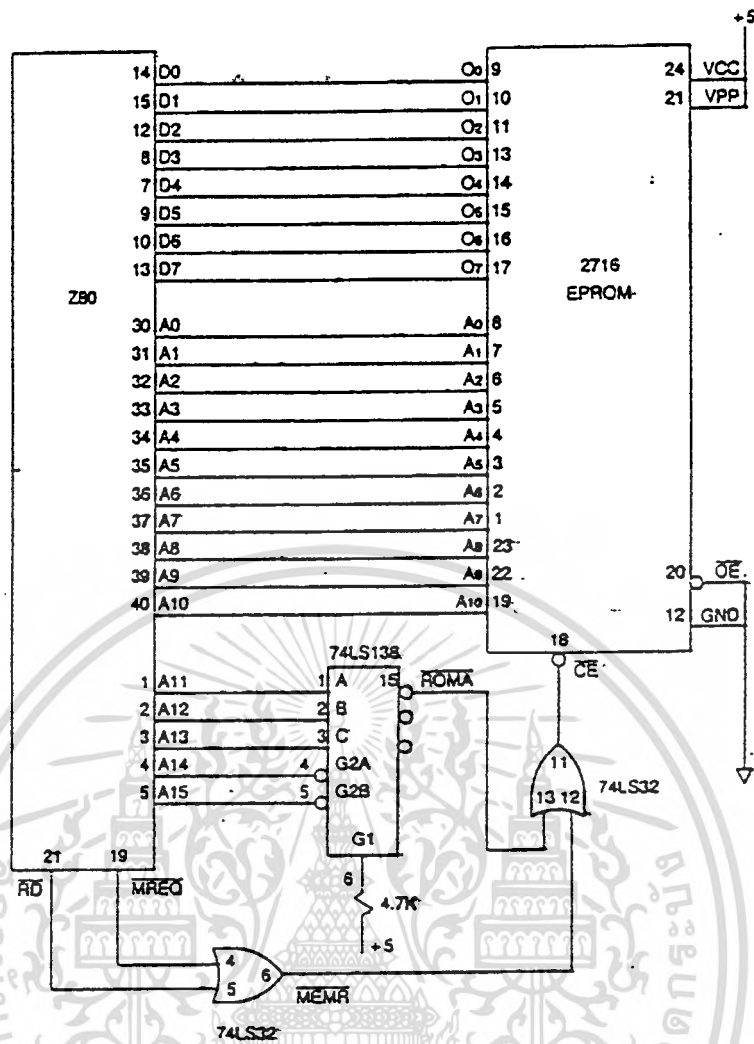


รูปที่ 2.17 แสดงวิธีการสร้างสัญญาณควบคุมในขบวนการอ่านข้อมูลจากหน่วยความจำ โดยได้มาจากการนำสัญญาณ /MREQ และ /RD มา OR กัน

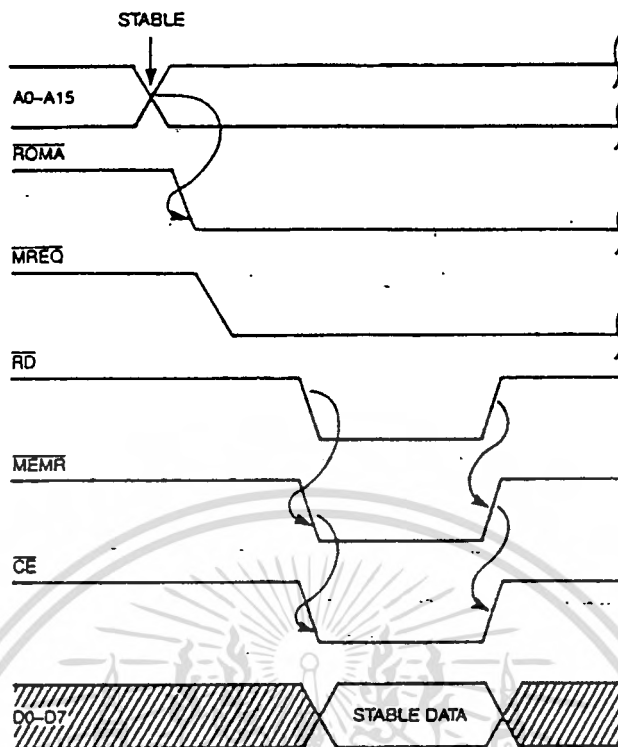
### การต่อ CHIP SELECT LINES

รูปที่ 2.18 แสดงให้เห็นลักษณะการต่อ Z80 กับ ROM อย่างสมบูรณ์ เมื่อสัญญาณ MEMORY SELECT และ MEMORY READ (/MEMR) เป็นลอจิก "0" ทั้งคู่ แล้วสัญญาณ CHIP SELECT ที่ต่อเข้ากับ EPROM เบอร์ 2716 จะทำงาน

รูปที่ 2.19 แสดงไคอะแกรมเวลาความสัมพันธ์ของสัญญาณที่สำคัญในระหว่างที่มีการอ่านข้อมูลจาก ROM



รูปที่ 2.18 ผังวงจรสมบูรณ์แสดงการต่อ 2716 EPROM กับ Z80 ไมโครโปรเซสเซอร์สัญญาณลอจิกที่ใช้เลือกอุปกรณ์หน่วยความจำ ถูกออกแบบไว้ตามผังหน่วยความจำ

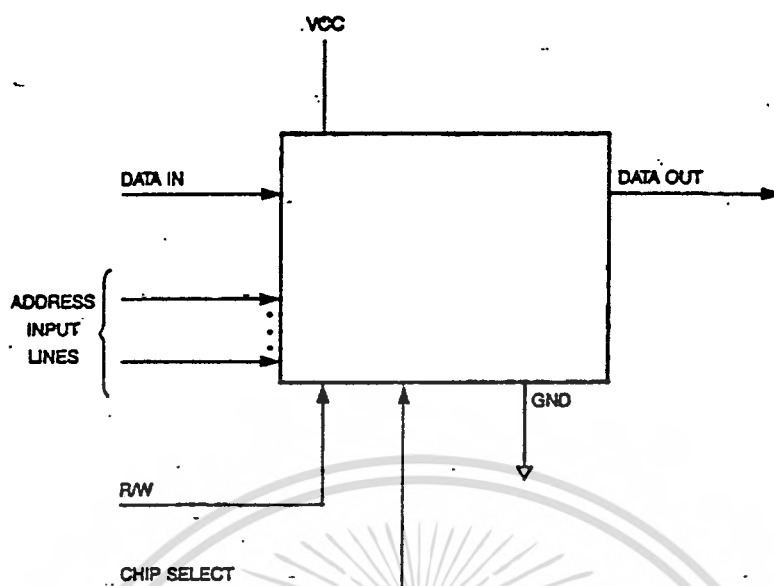


รูปที่ 2.19 โค้ดแอมเวลแสดงความสัมพันธ์ของสัญญาณที่สำคัญ ซึ่งใช้ในการอ่านข้อมูลจากหน่วยความจำตามรูป 2.18

### 2.3.2 การใช้ STATIC RAM กับ Z80

ในหัวข้อนี้จะกล่าวถึงการนำ RAM (RANDOM ACCESS MEMORY) มาเชื่อมต่อกับไมโครโปรเซสเซอร์ Z80 RAM ถูกใช้เก็บโปรแกรม,ข้อมูล หรือ ตัวแปร เป็นการชั่วคราว (ซึ่งแตกต่างจาก ROM) และ ข้อมูลที่เก็บไว้ใน RAM จะสูญหายไป เมื่อ ไม่มีการจ่ายไฟให้

RAM เป็นหน่วยความจำที่ใช้ในระบบไมโครโปรเซสเซอร์ซึ่งสามารถทั้งเขียนข้อมูลเข้าไปเก็บในตัวมันและอ่านออกมาได้ โดย CPU ต้องมีสัญญาณทางไฟฟ้าที่มีลักษณะเฉพาะสำหรับขบวนการเหล่านี้ ในรูปที่ 2.20 แสดงบล็อกโค้ดแอมของ RAM และสัญญาณทางไฟฟ้าที่สำคัญ



รูปที่ 2.20 แสดงสัญญาณทางไฟฟ้าที่จำเป็นในการใช้อุปกรณ์จำพวก RAM

### ขั้นตอนที่เกิดขึ้นเมื่อทำการอ่าน RAM

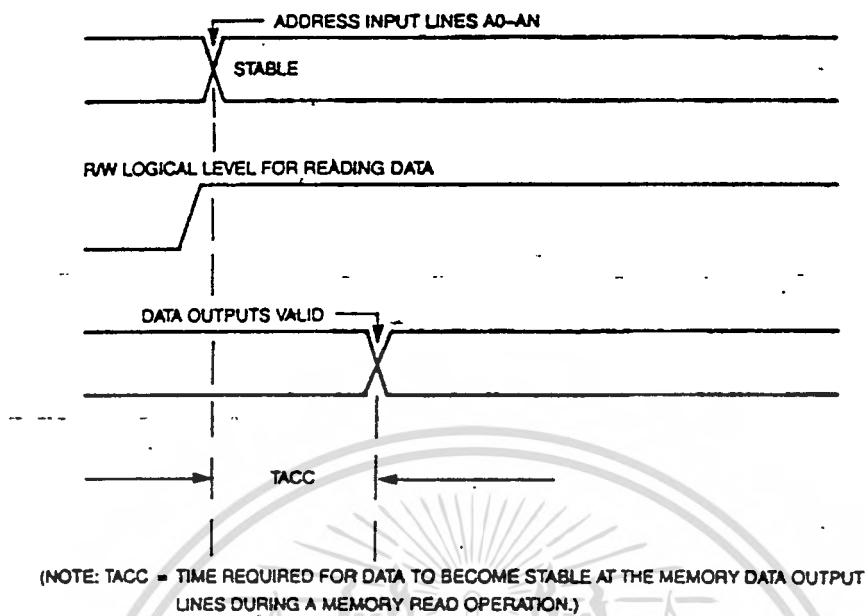
1. สัญญาณที่บ่งตำแหน่งแอดเดรสจะเข้ามาที่หน่วยความจำทางบัสแอดเดรสในเวลาเดียวกัน RAM จะนำเอาสัญญาณนี้มาทำการถอดรหัสตำแหน่งภายใน ซึ่งเป็นตำแหน่งที่อยู่ของข้อมูลที่จะอ่าน

2. สัญญาณบนเส้นควบคุม R/W จะถูกทำให้อยู่ในตำแหน่งลอจิกที่ถูกต้องซึ่งจะเป็นลอจิก "1" หรือ "0" ขึ้นอยู่กับหน่วยความจำนั้น(สามารถดูได้จาก DATA SHEET จากผู้ผลิต)

3. ระบบจะคอยอยู่ช่วงเวลาหนึ่ง เรียกช่วงเวลานี้ว่า READ ACCESS TIME เพื่อให้วงจรภายใน MEMORY CHIP ถอดรหัสตำแหน่งและเลือกข้อมูลในตำแหน่งนั้นๆ

4. หลังจากที่ยังคอยอยู่ช่วงเวลาหนึ่งแล้ว ข้อมูลจะถูกส่งออกมาบน DATA OUT LINE และถูกอ่านโดยไมโครโปรเซสเซอร์ของระบบ ถ้าหากไมโครโปรเซสเซอร์อ่านข้อมูลทันทีโดยไม่มี ACCESS TIME ข้อมูลที่ไม่ถูกต้องอาจจะถูกอ่านเข้าไปได้

ขั้นตอนเหล่านี้แสดงในรูปที่ 2.21 และเราจะกลับมาดูกันอีกครั้งหนึ่งเมื่อกล่าวถึงการติดต่อระหว่าง Z80 กับ RAM ซึ่งขั้นตอนในการอ่านข้อมูลของ RAM เหมือนกับ ROM ที่จริงแล้วระบบไมโครโปรเซสเซอร์ไม่สามารถที่จะแยกออกได้ว่า ข้อมูลที่อ่านมานั้นมาจาก ROM หรือ RAM

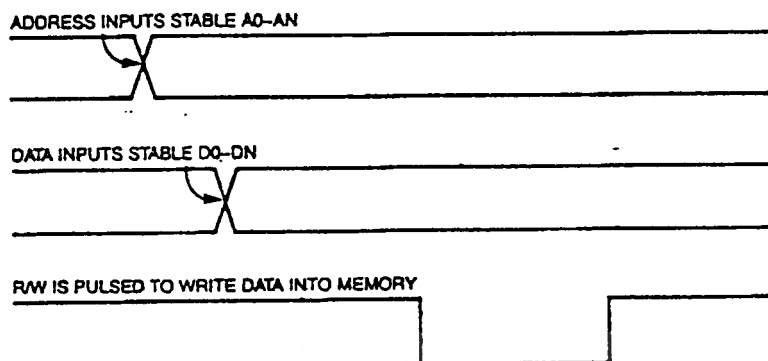


รูปที่ 2.21 ไคอะแกรมเวลาแสดงลำดับของขบวนการอ่านข้อมูลจากหน่วยความจำ

#### ขั้นตอนที่เกิดขึ้นเมื่อทำการเขียนข้อมูลเข้าไปใน RAM

1. บน ADDRESS INPUT LINE จะมีสัญญาณลอจิกที่บ่งตำแหน่งที่อยู่ภายในหน่วยความจำ ซึ่งเป็นตำแหน่งที่จะนำข้อมูลเข้าไปเขียน
2. ข้อมูลที่ต้องการเขียนจะปรากฏอยู่บน DATA INPUT LINE
3. ระบบจะคอยอยู่ช่วงเวลาหนึ่ง ช่วงเวลานี้เรียกว่า WRITE ACCESS TIME (ซึ่งใช้เวลาประมาณ 200-300 nanoseconds) เพื่อที่จะทำให้วงจรลอจิกที่ภายในอยู่ในสภาวะคงที่เสียก่อน
4. หลังจากคอยอยู่ช่วงเวลาหนึ่งแล้ว เส้นควบคุม R/W ของหน่วยความจำก็จะทำให้เป็นระดับของลอจิกที่ทำให้เกิดการเขียนเกิดขึ้น หรือเป็นพัลส์ เพื่อที่จะทำให้ข้อมูลที่ DATA INPUT LINE ถูกเขียนเข้าไปใน RAM

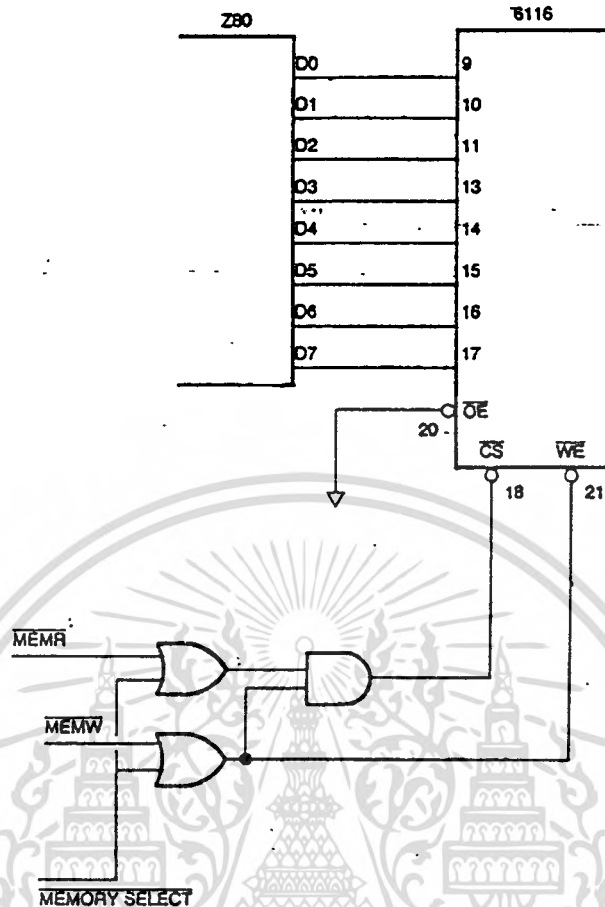
ในรูปที่ 2.22 ได้แสดงไคอะแกรมเวลาของขั้นตอนที่เกิดขึ้นเหล่านี้แล้วซึ่งอุปกรณ์ฮาร์ดแวร์ที่ต้องการให้เป็นไปตามขั้นตอนนี้ ขึ้นอยู่กับไมโครโปรเซสเซอร์ที่ใช้



รูปที่ 2.22 ไคอะแกรมเวลาแสดงลำดับของขบวนการที่สำคัญในการเขียนข้อมูลลงในหน่วยความจำ

#### หน่วยความจำ STATIC RAM เบอร์ 6116

ยังมีหน่วยความจำ STATIC RAM อื่นๆ อีกที่สามารถนำมาใช้ได้โดยง่ายกับระบบการควบคุมของ Z80 ที่น่าสนใจก็คือ เบอร์ 6116 ซึ่งเป็นแบบ 2K\*8 COMMON I/O STATIC RAM ดังรูปที่ 2.23 เป็นวิธีหนึ่งที่ใช้ในการต่อ 6116 RAM เข้ากับระบบควบคุมของ Z80 ซึ่งใช้ 6116 เพียงตัวเดียวก็สามารถทำให้เป็นระบบ RAM ที่สมบูรณ์ได้ (2114 ต้องใช้ถึง 2 ตัว) เนื่องจากมีโครงสร้างเป็นแบบ 2K\*8 จึงทำให้ 6116 RAM เหมาะสมกับการใช้งานกว่าไม่ว่าจะเป็นระบบใหญ่หรือเล็ก



รูปที่ 2.23 แผนผังแสดงการต่อ 6116 เข้ากับ Z80 CPU

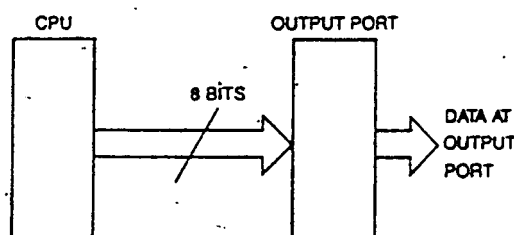
### 2.3.3 Z80 กับอินพุตและเอาต์พุต

ในกระบวนการทำงานของระบบไมโครโปรเซสเซอร์นั้น มีความจำเป็นอย่างยิ่งที่จะต้องทำการติดต่อกับอุปกรณ์อื่น ๆ เสมอ นอกเหนือจากหน่วยความจำแล้ว ในบางครั้ง CPU ยังมีความจำเป็นที่จะต้องทำการอ่านข้อมูลจากอุปกรณ์อินพุตและส่งข้อมูลออกไปทางอุปกรณ์เอาต์พุต สำหรับหัวข้อนี้ เราจะศึกษาวิธีการเชื่อมต่อระหว่าง Z80 เข้ากับอุปกรณ์ I/O (INPUT/OUTPUT DEVICE)

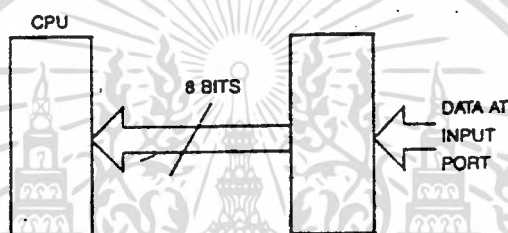
#### ขบวนการอินพุตและเอาต์พุตของ Z80

สัญญาณทางไฟฟ้าที่ใช้ในระบบฮาร์ดแวร์ในขณะที่ Z80 ทำการติดต่อกับอุปกรณ์ I/O นั้นจะประกอบไปด้วย สัญญาณจากบัสแอดเดรส (ADDRESS BUS) สัญญาณจากบัสข้อมูล (DATA BUS) และสัญญาณจากบัสควบคุม (CONTROL BUS)

รูป 2.24A และ 2.24B จะแสดงบล็อกไดอะแกรมของการส่งข้อมูลขนาด 8 บิตให้กับอุปกรณ์เอาต์พุตและรับข้อมูลจากอุปกรณ์อินพุต สำหรับรายละเอียดจะกล่าวถึงในภายหลัง เอกสารฉบับนี้ไม่ได้มีจุดประสงค์เพื่อแสดงวิธีการเชื่อมต่อที่ถูกต้อง แต่เพียงอย่างเดียว การนำเอาเอกสารนี้ไปใช้โดยไม่ปรึกษาผู้เกี่ยวข้อง อาจทำให้เกิดความเสียหายได้



รูปที่ 2.24A แสดงบล็อกไดอะแกรมของพอร์ทเอาต์พุต



รูปที่ 2.24B แสดงบล็อกไดอะแกรมของพอร์ทอินพุต

### การบ่งบอกตำแหน่งของพอร์ท (PORT ADDRESS)

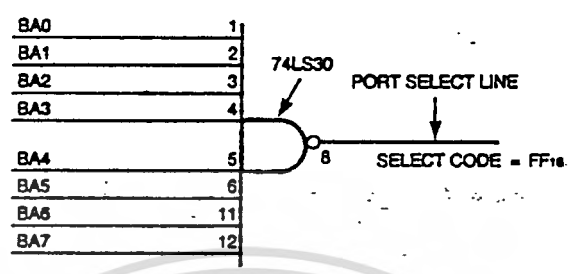
ในระบบของ Z80 นั้นจะมีบัสแอดเดรสอยู่ 16 เส้น แต่จะถูกใช้ในการบ่งบอกตำแหน่งของพอร์ท I/O เพียง 8 เส้นเท่านั้น ซึ่งก็คือ Z80 สามารถจะติดต่อกับ I/O ขนาด 8 บิตได้ถึง 256 พอร์ท ( $2^8 = 256$ )

เนื่องจากการบ่งตำแหน่งของ I/O และของหน่วยความจำในระบบของ Z80 ใช้บัสแอดเดรสชุดเดียวกันไมโครโปรเซสเซอร์จึงต้องมีสัญญาณ/IORQ (INPUT/OUTPUT REQUEST) แยกออกจาก /MREQ (MEMORY REQUEST) เราทราบมาแล้วว่าในระบบหน่วยความจำนั้นไม่จำเป็นต้องใช้บัสแอดเดรสทั้ง 16 เส้น (A0-A15) ในระบบของ I/O ก็ไม่จำเป็นที่จะต้องใช้สายแอดเดรสทั้ง 8 เส้นเช่นกัน

แต่ละพอร์ท I/O ในระบบไมโครโปรเซสเซอร์จะตอบสนองเพียงตำแหน่งเดียวของระบบบัสแอดเดรส (A0-A7) ตำแหน่งของพอร์ท I/O ที่ไมโครโปรเซสเซอร์ติดต่อด้านนั้นจะถูกเรียกว่า “พอร์ทแอดเดรส (PORT ADDRESS)” สำหรับในบทนี้เราจะใช้ตำแหน่ง 0FFH เป็น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

พอร์ทแอดเดรสของอุปกรณ์ I/O รูป 2.25 แสดงวงจรที่ใช้ในการถอดรหัสเลือกพอร์ทแอดเดรส 0FFH นี้

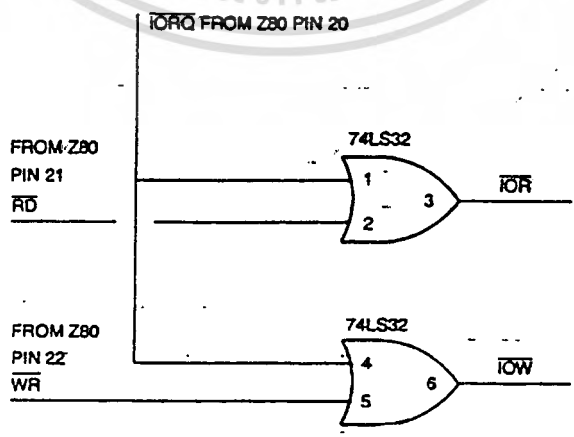


รูปที่ 2.25 แสดงวงจรถอดรหัสเพื่อสร้างสัญญาณ PORT SELECT โดยมีตำแหน่ง PORT อยู่ที่ 0FFH

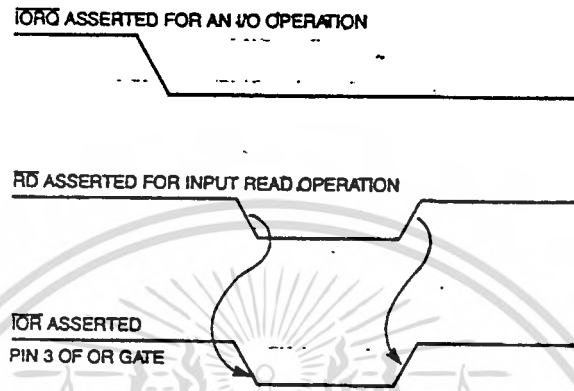
อย่างไรก็ตาม PORT SELECT LINE สามารถที่จะแอดทิฟได้ ในขณะที่ Z80 ไม่ต้องการติดต่อกับ I/O นั่นเอง เนื่องจากระบบหน่วยความจำใช้บัสแอดเดรสชุดเดียวกับระบบของ I/O นั่นเอง

**การสร้างสัญญาณควบคุม /IOW และ /IOR**

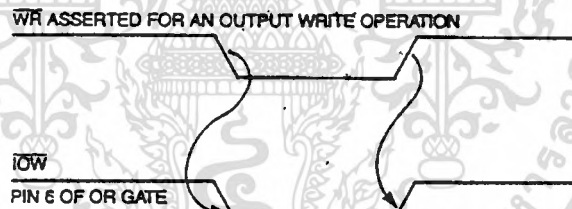
ในขณะที่ Z80 กระทำขบวนการติดต่อกับ I/O ขา /IORQ ของ Z80 จะให้ลอจิก “0” ออกมา เราจะใช้สัญญาณนี้ร่วมกับสัญญาณ /RD และ /WR เพื่อสร้างสัญญาณ /IOR (I/O READ : เมื่อต้องการรับข้อมูลจากอุปกรณ์อินพุต) และสัญญาณ /IOW (I/O WRITE : เมื่อต้องการส่งข้อมูลออกไปทางอุปกรณ์เอาต์พุต)



รูปที่ 2.26 แสดงให้เห็นว่าเราจะสร้างสัญญาณ  $\overline{IOR}$  และ  $\overline{IOW}$  ในระบบของ Z80 สังเกตว่าสัญญาณเหล่านี้จะแอกทีฟ ระหว่างกระทำการติดต่อกับอุปกรณ์ I/O เท่านั้น รูป 2.27A และ 2.27B แสดงไคอะแกรมเวลาของขบวนการดังกล่าว



รูปที่ 2.27A แสดงไคอะแกรมเวลาของการเกิดสัญญาณ  $\overline{IOR}$

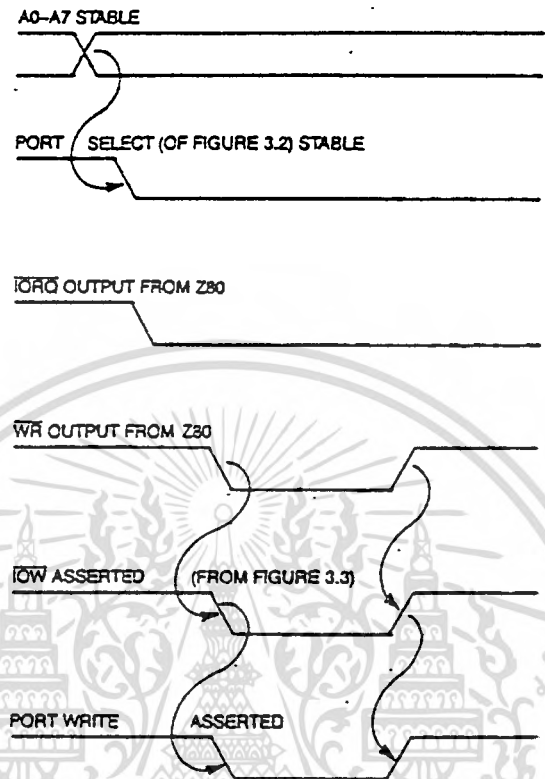


รูปที่ 2.27B แสดงไคอะแกรมเวลาของการเกิดสัญญาณ  $\overline{IOW}$

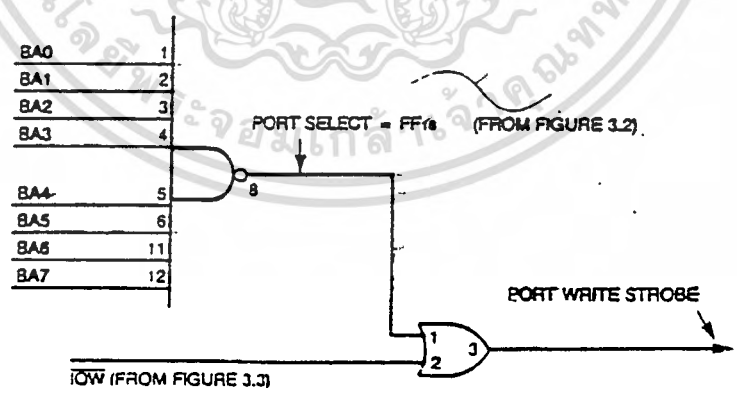
สัญญาณการเขียนพอร์ท (PORT WRITE) ของระบบ Z80 จะถูกใช้เป็นสัญญาณ WRITE ENABLE STROBE ของพอร์ทเอ้าพุทที่ถูกเลือก สัญญาณการเขียนพอร์ทนี้จะเกิดขึ้นได้ก็ต่อเมื่อ PORT SELECT LINE และ  $\overline{IOW}$  แอกทีฟ พร้อมกัน สัญญาณการเขียนพอร์ทจะเป็นสัญญาณที่กระตุ้นให้อุปกรณ์เอ้าพุทรับข้อมูลจากไมโครโปรเซสเซอร์ไปแลทช์ (LATCH) เอาไว้

ไคอะแกรมเวลาในรูป 2.28 แสดงลำดับเวลาของสัญญาณที่ปรากฏขึ้นในขณะที่ข้อมูลถูกส่งออกไปยังพอร์ทเอ้าพุท ในรูปนี้ จะเห็นว่าทุกๆสัญญาณยกเว้นสัญญาณ  $\overline{WR}$  จะเป็นเอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สัญญาณซึ่งจะคงอยู่นานกว่าขบวนการติดต่อกับ I/O จะสิ้นสุดลง ดังนั้นเราจะเขียนวงจรได้ดังรูป 2.29 วงจรนี้แสดงวิธีหนึ่งที่ใช้ในการสร้างสัญญาณการเขียนพอร์ทเท่านั้น



รูปที่ 2.28 แสดงไคอะแกรมเวลาของการเกิดสัญญาณ PORT WRITE



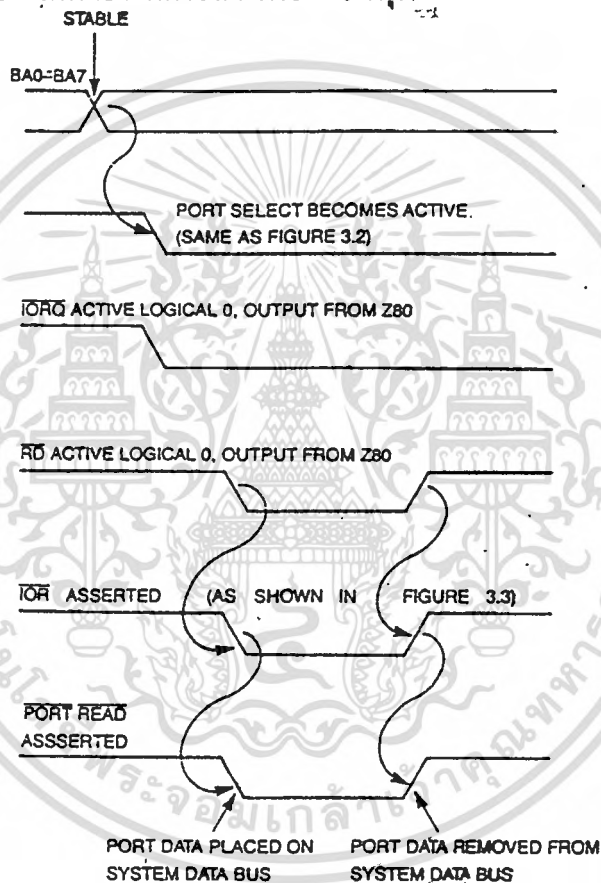
รูปที่ 2.29 แสดงวงจรการสร้างสัญญาณ PORT WRITE

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### การสร้างสัญญาณการอ่านพอร์ท

ที่จะกล่าวต่อไปนี้เราจะศึกษาขบวนการอ่านข้อมูลจากพอร์ทอินพุตของ Z80 ไคอะแกรมเวลาในรูป 2.30 แสดงลำดับของสัญญาณที่ปรากฏขึ้นระหว่างขบวนการอ่านข้อมูลจากพอร์ทอินพุต ในไคอะแกรมนี้ สัญญาณ  $/IOR$  จะมีลักษณะคล้ายกับสัญญาณ  $/LOW$  ในขณะทำการส่งข้อมูลไปยังพอร์ทเอาต์พุต

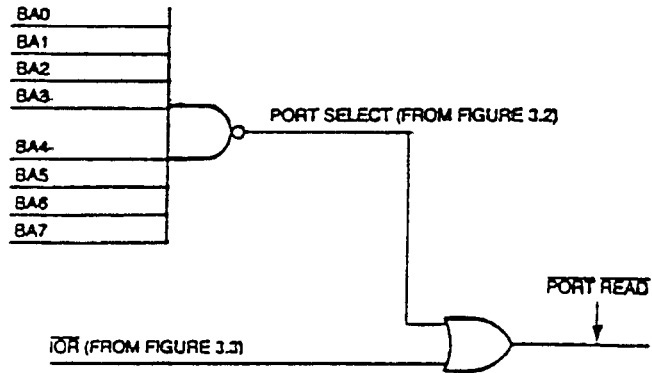
เราจะมาพิจารณาผลของสัญญาณ  $/RD$  ซึ่งแสดงในรูป 2.30 สัญญาณ  $/RD$  นี้จะนำไปสร้างสัญญาณ  $/IOR$  ให้แก่ฮาร์ดแวร์ส่วนพอร์ทอินพุต



รูปที่ 2.30 แสดงไคอะแกรมเวลาของสัญญาณที่เกิดขึ้นในรูปที่ 2.29

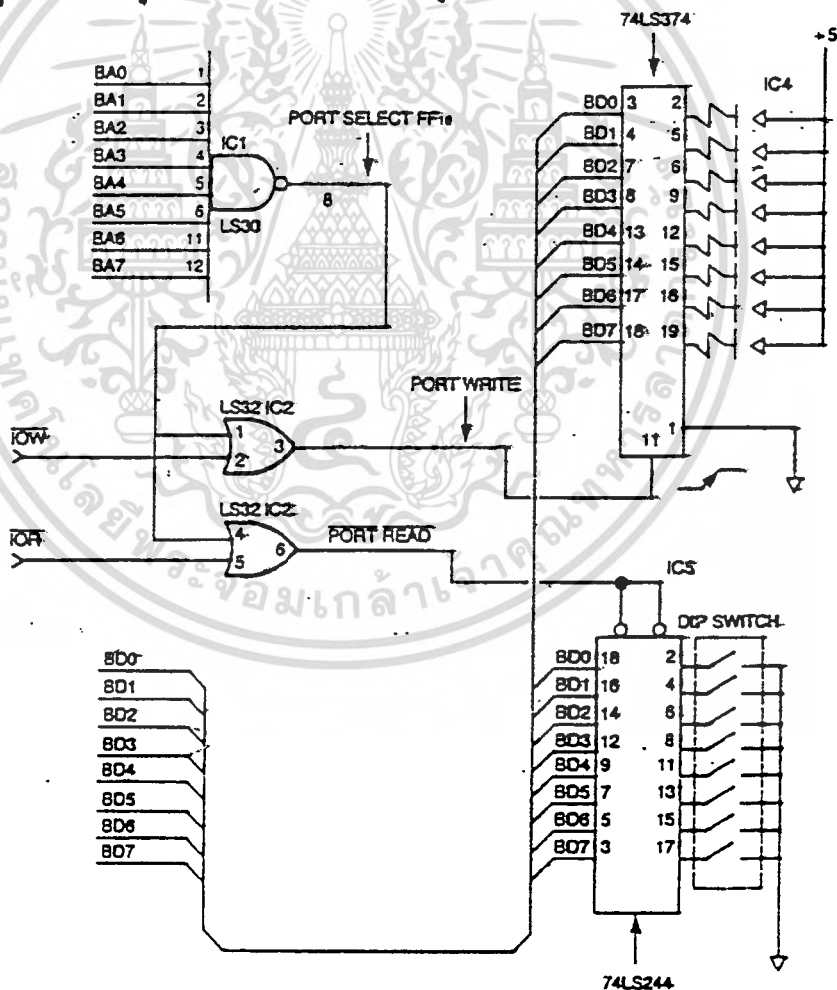
สัญญาณนี้จะทำให้เกิดการเคลื่อนย้ายข้อมูลจากพอร์ทอินพุตเข้าสู่ Z80 เมื่อ  $/RD$  เปลี่ยนลอจิกมาเป็น “0” ข้อมูลจากพอร์ทอินพุตจะปรากฏบนระบบบัสข้อมูล และจะถูก STROBE เข้าไปในรีจิสเตอร์ภายใน (INTERNAL REGISTER) ของ Z80 สัญญาณ  $/RD$  ก็จะกลับมาเป็น ลอจิก “1” ใหม่ ขบวนการเคลื่อนย้ายข้อมูลก็จะสิ้นสุดลง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.31 แสดงไคอะแกรมเวลาการสร้างสัญญาณ PORT READ

รูป 2.31 แสดงวิธีหนึ่งในหลายวิธี ที่จะสร้างสัญญาณอ่านพอร์ท ซึ่งมาจาก  
เงื่อนไขลอจิกในรูป 2.30 ในรูป 2.32 แสดงวงจรที่สมบูรณ์ในการสร้างพอร์ท I/O ขนาด  
8 บิตทั่วๆไป



รูปที่ 2.32 แสดงวงจรสมมูลของพอร์ทอินพุต/เอาต์พุตขนาด 8 บิต

ซึ่งมีตำแหน่ง I/O อยู่ที่ OFFH

ซึ่งมีตำแหน่ง I/O อยู่ที่ OFFH นั้น เอกสารนี้เป็นเอกสารที่ส่งมอบสำหรับใช้งานเพื่อวัตถุประสงค์เท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### ขบวนการอ่านข้อมูลจากพอร์ทอินพุท

ต่อไปเราจะพิจารณาขบวนการอ่านข้อมูลของ Z80 จากพอร์ทอินพุท ดังแสดงในรูป 2.32 ขบวนการส่วนใหญ่จะมีลักษณะคล้ายๆกับการส่งข้อมูลไปยังพอร์ทเอาต์พุท ยกเว้นสัญญาณ /RD ซึ่งจะถูกใช้แทนสัญญาณ /WR

1. สัญญาณ A0-A7 จะปรากฏขึ้นบนระบบบัสแอดเดรส IC1 จะทำการถอดรหัสขา 8 เอาต์พุทของ IC1 จะมีลอจิกเป็น "0" ข้อมูลต่างๆ ของพอร์ทอินพุทจะยังไม่ปรากฏบนบัสข้อมูล จะรอจนกว่าสัญญาณ /IORQ และสัญญาณ /RD เกิดขึ้นพร้อมกันก่อน

2. /IORQ จะถูกเช็ทให้มีลอจิกเป็น "0" โดย Z80 การกระทำนี้เป็นการบ่งบอกแก่ระบบว่า Z80 ต้องการติดต่อกับ I/O ไม่ใช่หน่วยความจำ

3. สัญญาณ /RD จะถูกเช็ทให้มีลอจิกเป็น "0" โดย Z80 ซึ่งจะทำให้เกิดสัญญาณ IOR ขึ้นในระบบ

4. เมื่อสัญญาณ /IOR ปรากฏขึ้นไมโครโปรเซสเซอร์จะทำการรับข้อมูลจากพอร์ทอินพุท เพราะว่าขา /RD จาก Z80 มีลอจิกเป็น "0" ขา ENABLE (ขา 1,19) ของ IC5 (74LS244) จะถูกเช็ทให้มีลอจิกเป็น "0" ลอจิก "0" บนขา ENABLE ของ 74LS244 จะทำให้ขาข้อมูลออก (DATA OUTPUT) มีข้อมูลส่งออกไปให้ระบบบัสข้อมูล

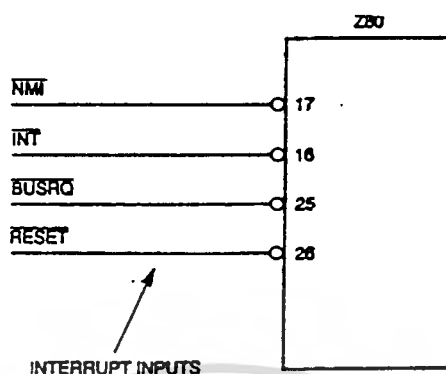
5. ท้ายสุดขา /RD จะถูกเช็ทให้มีลอจิกเป็น "1" อีกครั้งหนึ่ง ภายใต้การควบคุมของ Z80 เมื่อ /RD มีลอจิกเป็น "1" ข้อมูลจากพอร์ทอินพุทก็จะถูกเคลื่อนย้ายจากระบบบัสข้อมูลเข้าสู่ CPU ขบวนการอ่านข้อมูลจากพอร์ทอินพุทก็จะสิ้นสุดลง

#### 2.3.4 Z80 กับการอินเทอร์รัพท์

การใช้ไมโครโปรเซสเซอร์นั้น สิ่งหนึ่งที่ผู้ออกแบบระบบมักจะต้องคำนึงถึงเสมอก็คือ เรื่องของการอินเทอร์รัพท์ ดังนั้นในการที่จะเอาไมโครโปรเซสเซอร์ไปใช้งานได้อย่างมีประสิทธิภาพนั้นจึงจำเป็นต้องทำความเข้าใจเกี่ยวกับลักษณะและวิธีการอินเทอร์รัพท์ เสียก่อน

##### การอินเทอร์รัพท์ของ Z80

จุดประสงค์ของการทำขบวนการอินเทอร์รัพท์ก็คือ การทำให้ CPU พักจากการทำงานในโปรแกรมหลักไว้ชั่วคราว แล้วกระโดดไปทำงานในส่วนของโปรแกรมที่ตอบสนองต่อการอินเทอร์รัพท์ (INTERRUPT SERVICE ROUTINE) นั้นโดยขึ้นกับชนิดและโหมดของการอินเทอร์รัพท์ ซึ่งจะกล่าวถึงต่อไป



รูปที่ 2.33 สัญญาณที่ใช้เกี่ยวกับการอินเทอร์รัพท์บน Z80

Z80 CPU ได้จัดการอินเทอร์รัพท์ ไว้ใน 2 ลักษณะ คือการอินเทอร์รัพท์ แบบไม่มีเงื่อนไข (NON-MASKABLE INTERRUPT) และแบบมีเงื่อนไข (MASKABLE INTERRUPT) ซึ่ง Z80 จะทำการตอบสนองต่อการอินเทอร์รัพท์ทั้ง 2 แบบในลักษณะที่แตกต่างกันดังที่จะกล่าวถึงในภายหลัง ยังมีวิธีอื่นอีก 2 วิธี ซึ่งอาจจะกล่าวได้ว่าเป็นการอินเทอร์รัพท์ Z80 เช่นกันก็คือ DMA (DIRECT MEMORY ACCESS) และการรีเซ็ต

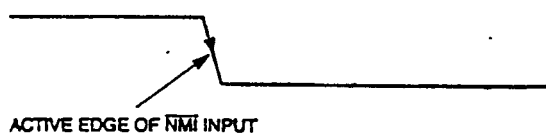
การรีเซ็ต Z80 นั้นสามารถที่จะทำได้โดยการทำให้ระดับลอจิกที่ขารีเซ็ตของ Z80 เป็นลอจิก "0" ซึ่งจะเป็นการทำให้วงจรภายใน Z80 บางส่วนถูกปรับสถานะใหม่ จากนั้น Z80 ก็จะกระโดดไปปฏิบัติตามส่วนของโปรแกรมที่แอดเดรส 0000H โดยทั่วไปแล้วเป็นโปรแกรมที่ใช้ในการควบคุมระบบ (MONITOR PROGRAM)

สำหรับขบวนการ DMA นั้นเป็นขบวนการที่ค่อนข้างจะแตกต่างจากการอินเทอร์รัพท์และการรีเซ็ตอยู่มาก เนื่องจากขบวนการ DMA นี้แทนที่จะทำให้ Z80 กระโดดไปทำงานในโปรแกรมที่อยู่ในแอดเดรสอื่น กลับทำให้ Z80 หยุดทำงานไว้ชั่วขณะหนึ่ง

#### การอินเทอร์รัพท์แบบไม่มีเงื่อนไข (NON-MASKABLE INTERRUPT)

Z80 จะทำการตรวจสอบสัญญาณที่ขา /NMI อยู่ตลอดเวลา ว่ามีการเปลี่ยนระดับลอจิกจาก "1" เป็น "0" หรือไม่ ถ้าพบว่าการเปลี่ยนระดับดังกล่าวเกิดขึ้น FLIP-FLOP ภายใน Z80 จะถูกเซ็ตไว้ และ Z80 จะตรวจสอบค่าของ FLIP-FLOP ใน STATE สุดท้ายของการทำงานในทุกๆ INSTRUCTION CYCLE ดังรูปที่ 2.34 ในกรณีที่ Z80 พบว่าค่าใน FLIP-FLOP นี้ถูกเซ็ตไว้ ก็จะเข้าสู่การทำงานในขบวนการที่ตอบสนองต่อการอินเทอร์รัพท์ในแบบนี้ ซึ่งจะกล่าวถึงในภายหลัง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.34 รูปสัญลักษณ์ที่ใช้ในการขออินเทอร์รัพท์แบบไม่มีเงื่อนไข

การขออินเทอร์รัพท์แบบนี้ Z80 จะทำการตอบสนองในทุกกรณี (ผู้ออบแบบไม่สามารถที่จะทำให้ Z80 DISABLE ต่อการอินเทอร์รัพท์แบบ นอน-มาสค์เคเบิลนี้ได้) ดังนั้น อุปกรณ์ภายนอกที่จะทำการขออินเทอร์รัพท์แบบนี้ได้นั้นจึงมักจะเป็นอุปกรณ์ที่มีความสำคัญมาก ๆ เท่านั้น

หลังจากที่ตรวจพบการอินเทอร์รัพท์แบบนี้แล้ว Z80 จะมีการทำงานดังนี้

1. นำเอาค่าแอดเดรสในรีจิสเตอร์ PC ไปเก็บไว้ในสแตค
2. นำเอาค่าสถานะของ IFF1 ไปเก็บไว้ใน IFF2
3. ปรับสถานะของ IFF1 ให้เป็นลอจิก "0" เพื่อให้ Z80 ทำการ DISABLE ต่อการขออินเทอร์รัพท์จากภายนอก
4. Z80 จะกระโดดไปปฏิบัติตามโปรแกรมที่แอดเดรส 0066H

สิ่งสำคัญสิ่งหนึ่งที่จะต้องคำนึงถึงก็คือ ในขณะที่ Z80 ถูกอินเทอร์รัพท์นั้นค่าในรีจิสเตอร์ต่างๆจะไม่ถูกเก็บเอาไว้ ดังนั้นในส่วนต้นของโปรแกรมตอบสนองการอินเทอร์รัพท์จึงควรที่จะทำการเก็บค่ารีจิสเตอร์ต่างๆเอาไว้ เนื่องจาก Z80 อาจจะถูกอินเทอร์รัพท์ที่ส่วนใดของโปรแกรมหลักก็ได้ ถ้าเราไม่ทำการเก็บค่ารีจิสเตอร์ต่างๆไว้ ค่ารีจิสเตอร์ต่างๆเหล่านั้นอาจจะถูกเปลี่ยนแปลงไปทำให้เกิดความผิดพลาดขึ้นในระบบได้ และอีกสิ่งหนึ่งที่สำคัญมากก็คือที่ส่วนท้ายของโปรแกรมตอบสนองการอินเทอร์รัพท์นั้น จะต้องทำการดึงเอาค่าที่เก็บไว้กลับคืนมาให้กับรีจิสเตอร์ของ Z80 ด้วย ถ้าไม่ทำการดึงเอาค่าที่เก็บไว้กลับคืนมา นอกจากจะทำให้ค่าในรีจิสเตอร์เกิดความผิดพลาดแล้วยังจะทำให้ Z80 กลับมาทำงานที่แอดเดรสเดิมก่อนที่จะถูกการอินเทอร์รัพท์ไม่ได้อีกด้วย เพราะค่าที่ถูกนำมาใส่คืนให้กับ PC นั้นไม่ใช่ค่าเดิม แต่จะเป็นค่ารีจิสเตอร์แทน

จากรูปที่ 2.35 จะเห็นว่าถ้าเราไม่ใช่คำสั่ง POP ก่อนคำสั่ง RETN ค่าของ PC จะกลายเป็นค่าของรีจิสเตอร์ HL แทนที่จะเป็นค่าเดิมของ PC ที่เก็บเอาไว้ ซึ่งเป็นสาเหตุใหญ่ที่ทำให้เกิดความผิดพลาดขึ้นในระบบได้

เอกสารนี้เป็นเอกสารลิขสิทธิ์ของสถาบันเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

PUSH AF

PUSH BC

PUSH DE

PUSH HL

-----  
! เซอวิสรูทีน !  
-----

POP HL

POP DE

POP BC

POP AF

RETN

รูปที่ 2.35 การใช้คำสั่ง PUSH และ POP ในโปรแกรมตอบสนองการอินเทอร์รัพท์

การสร้างสัญญาณ /NMI (สัญญาณ NON-MASKABLE INTERRUPTS)

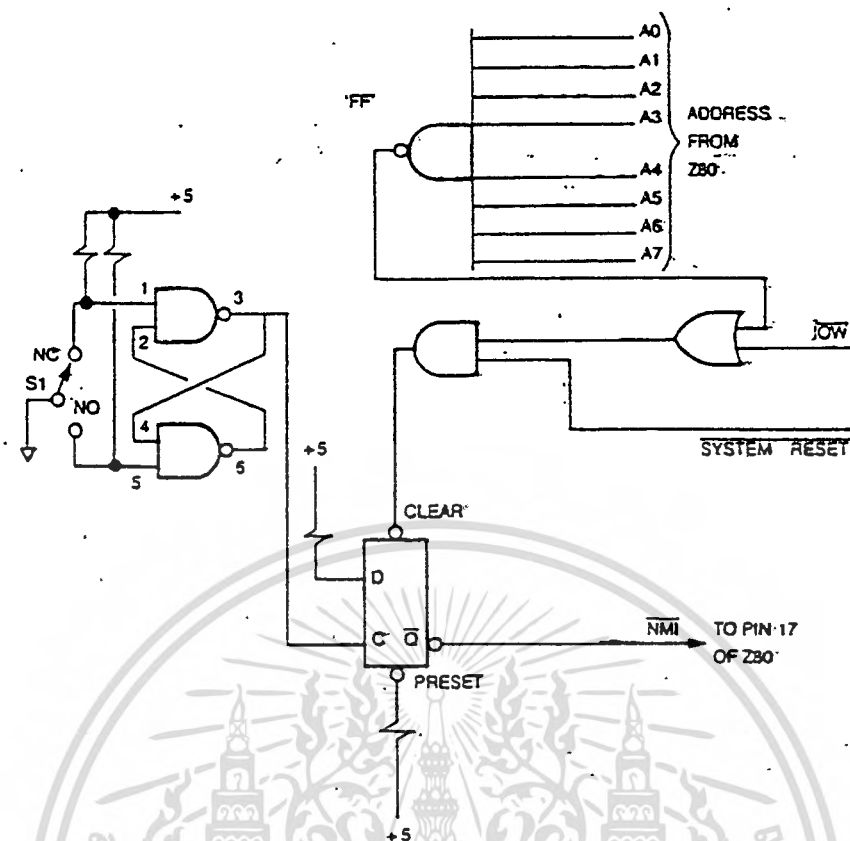
วิธีการในการสร้างสัญญาณ /NMI นั้นมีอยู่หลายวิธี แต่ในทุกวิธีต่างก็ต้องยึดหลักที่ว่า เมื่อต้องการที่จะขออินเทอร์รัพท์ Z80 นั้น จะต้องทำให้ระดับสัญญาณ ที่ขา /NMI ของ Z80 เปลี่ยนจากลอจิก “1” เป็น “0” และเมื่อเสร็จสิ้นจากขบวนการอินเทอร์รัพท์แล้วก็ต้องทำการถอนสัญญาณ /NMI โดยการทำให้ระดับสัญญาณที่ขา /NMI ของ Z80 กลับเป็นลอจิก “1” อีกครั้ง

วงจรข้างล่างนี้เป็นวิธีง่ายวิธีหนึ่งในการที่จะสร้างสัญญาณ /NMI ให้กับ Z80 ได้ โดยเพียงแต่กดสวิทช์เท่านั้น

เมื่อสวิทช์ S1 ถูกกด เอาท์พุทที่ขา 3 ของ NAND GATE จะกลายเป็น “0” ทำให้ขาคล็อกของ D FLIP-FLOP กลายเป็น “0” ตามเมื่อเราปล่อยสวิทช์ ขาคล็อกของ D FLIP-FLOP จะกลายเป็น “1” ซึ่งจะมีผลให้อาท์พุท /Q กลายเป็น “0” ซึ่งจะทำให้ขา /NMI ของ Z80 ได้รับสัญญาณอินเทอร์รัพท์ หลังจากที Z80 ทำงานในส่วนที่เป็นโปรแกรมตอบสนองการอินเทอร์รัพท์จนเสร็จแล้วนั้น ก่อนที่ Z80 จะกลับเข้าสู่โปรแกรมหลัก เราจะต้องเขียนโปรแกรมให้ Z80 ส่งข้อมูลไปที่พอร์ท 0FFH ซึ่งจะทำให้ /Q ของ D FLIP-FLOP กลับเป็น “1” อีกครั้ง

ผังรูปที่ 2.37

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.36 ตัวอย่างวงจรสำหรับสร้างสัญญาณ NMI ให้กับ Z80

ORG 0066H

PUSH AF ; เก็บค่า رجิสเตอร์ไว้ใน

PUSH BC ; สแต็ก

PUSH DE

PUSH HL

ส่วนที่เป็นโปรแกรมตอบสนองการอินเทอร์รัพท์

POP HL ; นำค่าที่เก็บไว้มาคืนให้

POP DE ; กับ رجิสเตอร์

POP BC

POP AF

OUT (OFFH), A ; ทำการถอนสัญญาณ NMI

RETN

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์ 2.37 ตัวอย่างโปรแกรมสำหรับวงจรในรูปที่ 2.36 ไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## มาส์ค-เคเบิลอินเทอร์รัพท์ (MASKABLE INTERRUPT)

การอินเทอร์รัพท์แบบนี้เราสามารถที่จะสั่งให้ Z80 ENABLE หรือ DISABLE ต่อการอินเทอร์รัพท์ได้ด้วยวิธีทางซอฟต์แวร์ โดยใช้คำสั่ง EI (ENABLE INTERRUPT) หรือ DI (DISABLE INTERRUPT)

สำหรับวิธีการที่จะขอการอินเทอร์รัพท์แบบนี้เราสามารถที่จะทำได้โดยการทำให้ระดับลอจิกที่ขา  $\overline{INT}$  เป็น "0" อยู่จนกระทั่ง Z80 สามารถที่จะตรวจจับสัญญาณได้ โดยที่ Z80 จะทำการตรวจสอบระดับสัญญาณที่ขา  $\overline{INT}$  ใน STATE สุดท้ายของการทำงานในทุกๆ INSTRUCTION CYCLE เมื่อ Z80 ตรวจพบสัญญาณ  $\overline{INT}$  แล้วก็จะทำการตรวจสอบสถานะของ IFF1 ถ้าพบว่าสถานะของ IFF1 เป็น "0" Z80 ก็จะ DISABLE ต่อการอินเทอร์รัพท์ แต่ในกรณีที่สถานะของ IFF1 เป็น "1" Z80 ก็จะทำการตอบสนองต่อการอินเทอร์รัพท์ และก่อนที่ Z80 จะกลับเข้าสู่การทำงานในโปรแกรมหลักนั้น เราจะต้องทำการถอนสัญญาณอินเทอร์รัพท์ที่ขา  $\overline{INT}$  ของ Z80 ออกด้วย โดยการทำให้ระดับสัญญาณที่ขา  $\overline{INT}$  ของ Z80 กลับเป็น "1" ใหม่

ในตอนนี้เราสามารถสรุปการอินเทอร์รัพท์แบบนี้ได้เป็น 2 หัวข้อ คือ

1. การขอการอินเทอร์รัพท์แบบมาส์คเคเบิล สามารถที่จะถูก DISABLE ได้โดยใช้คำสั่ง DI ซึ่งจะทำให้สถานะของ IFF1 กลายเป็น "0"
2. สัญญาณ  $\overline{INT}$  จะต้องมิลลอจิกเป็น "0" อยู่จนกระทั่ง Z80 สามารถที่จะจับสัญญาณ  $\overline{INT}$  ได้ และก่อนที่ Z80 จะกลับเข้าสู่การทำงานในโปรแกรมหลักนั้น จะต้องทำการถอนสัญญาณ  $\overline{INT}$  ออกด้วย โดยการทำให้สัญญาณที่ขา  $\overline{INT}$  กลับเป็น "1" อีกครั้ง

การอินเทอร์รัพท์แบบมาส์คเคเบิลนี้ มีการทำงานในลักษณะต่างๆกันถึง 3 โหมด คือ โหมด 0, โหมด 1, และ โหมด 2 ซึ่งเราจะต้องทำการโปรแกรมเลือกโหมดก่อนที่ Z80 จะถูกอินเทอร์รัพท์โดยใช้คำสั่ง IM0, IM1 หรือ IM2 เพื่อเลือกการทำงานในโหมด 0, 1 หรือ 2 ตามลำดับ

### การอินเทอร์รัพท์ในโหมด 1

การเลือกการอินเทอร์รัพท์ในโหมดนี้ทำได้โดยใช้คำสั่ง IM1 หลังจาก that Z80 ได้ถูกโปรแกรมให้ทำงานในโหมด 1 แล้ว Z80 จะทำงานตามขั้นตอนต่อไปนี้

1. สถานะของ IFF1 จะถูกปรับให้เป็นลอจิก "0" ซึ่งจะเป็นการทำให้ Z80 DISABLE ต่อการอินเทอร์รัพท์ที่ขา  $\overline{INT}$
- 2 สถานะของ IFF2 จะถูกปรับให้เป็นลอจิก "0" เช่นเดียวกับ IFF1

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3. ค่าของ PC จะถูกนำไปเก็บไว้บนสแตค ซึ่งจะถูกนำกลับมาใช้เป็นแอดเดรส (ก็นำกลับมาคืนให้กับ PC นั่นเอง) ของชุดคำสั่งที่ Z80 จะทำหลังจากที่เสร็จจากการทำงานในโปรแกรมตอบสนองการอินเทอร์รัพต์แล้ว

4. Z80 จะกระโดดไปทำงานที่แอดเดรส 0038H

#### การอินเทอร์รัพต์ในโหมด 0

การอินเทอร์รัพต์แบบนี้สามารถทำได้โดยการรีเซตบน Z80 ดังนั้น โหมดของการอินเทอร์รัพต์จะถูกจัดให้อยู่ในโหมด 0 หรือ อีกวิธีหนึ่งที่จะทำให้ Z80 ทำงานในโหมดนี้ก็คือการใช้คำสั่ง IM0

วิธีการขออินเทอร์รัพต์ในโหมดนี้ก็สามารถที่จะทำได้โดยใช้วิธีเดียวกันกับการอินเทอร์รัพต์ในโหมด 1 อย่างไรก็ตามในโหมด 0 นี้จะมีข้อได้เปรียบกว่าในโหมด 1 อยู่ก็คือเราสามารถที่จะสั่งให้ Z80 กระโดดไปทำงานที่แอดเดรสใดๆหรือทำคำสั่งใดก็ได้ซึ่งจะได้กล่าวต่อไป

เมื่อได้รับการขออินเทอร์รัพต์จากอุปกรณ์ภายนอกและ Z80 ก็จะถูกสั่งให้ ENABLE การขออินเทอร์รัพต์แล้ว Z80 จะทำงานตามขั้นตอนต่อไปนี้

1. IFF1 และ IFF2 จะถูกปรับให้มีลอจิก "0" ซึ่งจะเป็นการทำให้ Z80 DISABLE ต่อการขออินเทอร์รัพต์ที่อาจเกิดขึ้นได้อีก

2. เอาท์พุทจากขา /M1 และ /IORQ จะกลายเป็น "0" ทั้งคู่ซึ่งจะเกิดขึ้นเฉพาะในขณะที่ Z80 ได้รับทราบการขออินเทอร์รัพต์แล้วเท่านั้น โดยทั่วไปแล้วสัญญาณจากขา /M1 และ /IORQ นี้จะถูกนำมาผ่าน OR GATE (ซึ่งจะทำให้เอาท์พุทเป็น "0" เฉพาะในช่วงที่สัญญาณจากขา /M1 และ /IORQ เป็น "0" พร้อมกันเท่านั้น) เพื่อนำไปใช้สำหรับการทำให้อุปกรณ์ภายนอกที่ขออินเทอร์รัพต์ทราบว่า Z80 ได้ทำการตอบรับการขออินเทอร์รัพต์แล้ว

3. เมื่ออุปกรณ์ภายนอกได้ตรวจสอบพบว่า Z80 ได้ทำการตอบรับการขออินเทอร์รัพต์มาแล้วก็จะส่งข้อมูล 1 ไบต์ ออกมาที่บัสข้อมูล (DATA BUS) ให้กับ Z80 จะถือว่าข้อมูลนี้เป็นออฟโคดของคำสั่ง และทำงานตามออฟโคดนั้นทันที โดยทั่วไป แล้วข้อมูลไบต์นี้จะอยู่ในชุดคำสั่ง RST (RESTART) 0-7 ซึ่งจะมีผลทำให้ Z80 กระโดดไปทำงานในแอดเดรสต่างๆ

4. Z80 อ่านข้อมูลเข้ามาแล้วทำการถอดรหัสเช่นเดียวกับออฟโคดของคำสั่ง

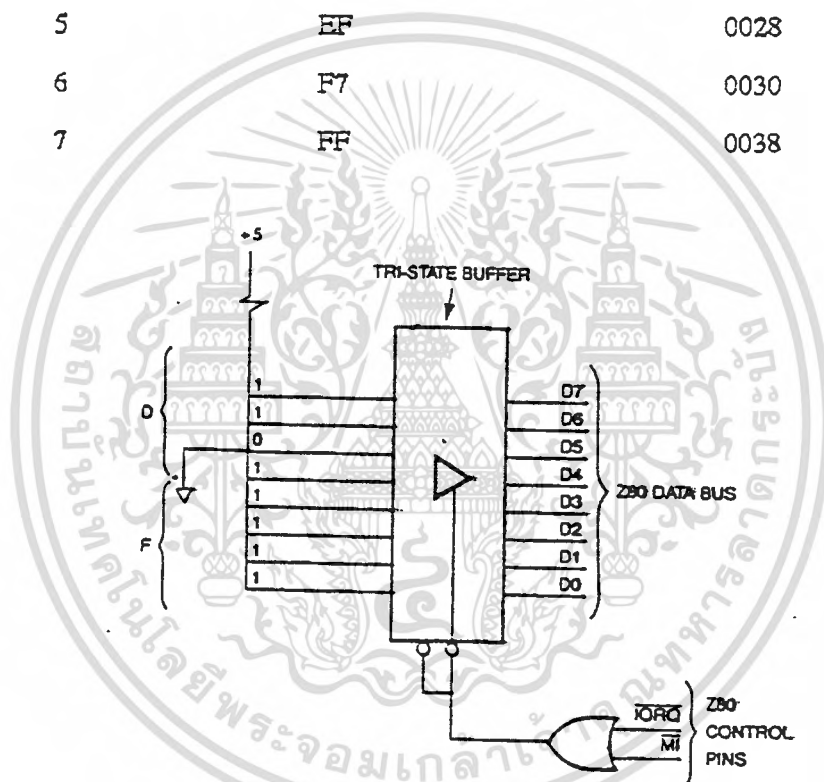
5. ค่า PC จะถูกเก็บบนสแตค

6. Z80 จะกระโดดไปทำงานยังตำแหน่งแอดเดรสที่ถูกกำหนดโดยชุดคำสั่ง RST

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตำแหน่งแอดเดรสที่ Z80 จะกระโดดไปทำเมื่อเทียบกับข้อมูลที่ได้รับดังนี้

| RST | ข้อมูลที่ส่งให้กับ Z80 (ฐาน 16) | ตำแหน่งแอดเดรส(ฐาน 16) |
|-----|---------------------------------|------------------------|
| 0   | C7                              | 0000                   |
| 1   | CF                              | 0008                   |
| 2   | D7                              | 0010                   |
| 3   | DF                              | 0018                   |
| 4   | E7                              | 0020                   |
| 5   | EF                              | 0028                   |
| 6   | F7                              | 0030                   |
| 7   | FF                              | 0038                   |



รูปที่ 2.38 วงจรง่าย ๆ สำหรับส่งข้อมูล RST ให้กับ Z80

รูปที่ 2.38 นี้แสดงถึง วิธีการง่าย ๆ วิธีหนึ่งที่ใช้ในการส่งข้อมูล RST ให้กับ Z80

#### การอินเทอร์รัพท์ในโหมด 2

Z80 จะเลือกโหมด 2 เป็นโหมดในการอินเทอร์รัพท์เมื่อ Z80 ได้รับคำสั่ง IM2 ( INTERRUPT MODE2 ) สำหรับวิธีการขออินเทอร์รัพท์ในโหมดนี้ ก็ใช้วิธีเดียวกันกับการอินเทอร์รัพท์ในโหมด 1 และ 0 นั่นเอง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การทำงานของ Z80 เมื่อถูกอินเทอร์รัพท์ในโหมด 2 นี้ก็คือ เมื่อ Z80 ตอบรับการขออินเทอร์รัพท์โดยการส่งสัญญาณ /M1 และ /IORQ (ลอจิก "0") ให้กับอุปกรณ์ภายนอกที่ทำการขออินเทอร์รัพท์ อุปกรณ์ภายนอกก็จะทำการส่งข้อมูล 1 ไบต์ผ่านบัสข้อมูลให้กับ Z80 ซึ่งในขั้นตอนนี้จะมีลักษณะคล้ายกับการอินเทอร์รัพท์ในโหมด 0 นั่นเอง เพียงแต่ว่าในโหมด 2 นี้เมื่อ Z80 ได้รับข้อมูลแล้วจะนำเอามาประกอบกับข้อมูลที่ถูกเก็บไว้ในรีจิสเตอร์ I โดยใช้ข้อมูลในรีจิสเตอร์ I เป็นแอดเดรส 8 บิตบน (A15-A18) และข้อมูลที่ได้รับเข้ามาเป็นแอดเดรสที่ N นี้มาประกอบกับข้อมูลในแอดเดรสที่ N+1 เพื่อใช้เป็นแอดเดรสของ ROUTINE ที่ Z80 จะกระโดดไปทำ โดยจะใช้ข้อมูลในแอดเดรสที่ N เป็นแอดเดรส 8 บิตล่าง และใช้ข้อมูลในแอดเดรสที่ N+1 เป็นแอดเดรส 8 บิตบน

### 2.3.5 การใช้ 8255 PIA กับ Z80

เป็นอุปกรณ์ซึ่งช่วยในการติดต่อกับระบบภายนอกของระบบไมโครโปรเซสเซอร์ให้มีประสิทธิภาพยิ่งขึ้น ซึ่งจะกล่าวดังต่อไปนี้

#### รายละเอียดเกี่ยวกับ 8255

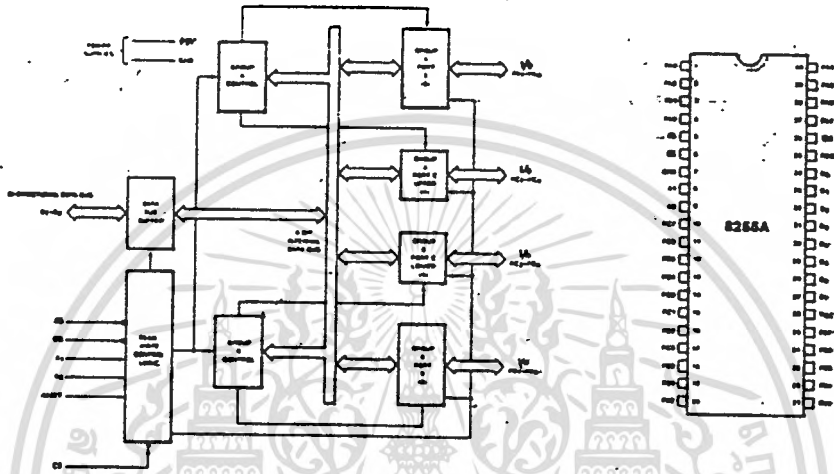
8255 เป็นอุปกรณ์ LSI (LARGE SCALE INTEGRATED CIRCUIT) บรรจุอยู่ใน PACKAGE 40 ขา ผลิตโดยบริษัท INTEL COORPERATION

รูปที่ 2.39 นี้จะแสดงบล็อกไดอะแกรมของ 8255 ซึ่งหน้าที่ของแต่ละบล็อกมีดังนี้

## 8255A/8255A-5 PROGRAMMABLE PERIPHERAL INTERFACE

- MCS-85™ Compatible 8255A-5
- 24 Programmable I/O Pins
- Completely TTL Compatible
- Fully Compatible with Intel® Micro-processor Families
- Improved Timing Characteristics
- Direct BH Set/Reset Capability Easing Control Application Interface
- 40-Pin Dual In-Line Package
- Reduces System Package Count
- Improved DC Driving Capability

The Intel® 8255A is a general purpose programmable I/O device designed for use with Intel® microprocessors. It has 24 I/O pins which may be individually programmed in 2 groups of 12 and used in 3 major modes of operation. In the first mode (MODE 0), each group of 12 I/O pins may be programmed in sets of 4 to be input or output. In MODE 1, the second mode, each group may be programmed to have 8 lines of input or output. Of the remaining 4 pins, 3 are used for handshaking and interrupt control signals. The third mode of operation (MODE 2) is a bidirectional bus mode which uses 8 lines for a bidirectional bus, and 5 lines, borrowing one from the other group, for handshaking.



รูปที่ 2.39 แสดงบล็อกไดอะแกรมและการวางตำแหน่งขาของ 8255

บล็อกกลุ่มแรกที่จะพูดถึงนี้ได้แก่ บล็อกจำนวน 4 บล็อก ที่อยู่ทางขวามือของรูป ซึ่งจะ有一部分ที่ใช้เชื่อมต่อกับอุปกรณ์ภายนอกอื่นๆ โดยมีสาย PA0-PA7, PB0-PB7 และ PC0-PC7 เป็นทางผ่านข้อมูลระหว่างอุปกรณ์ภายนอกกับ 8255

บล็อกกลุ่มถัดมาได้แก่ GROUP A CONTROL และ GROUP B CONTROL ซึ่งจะเป็นตัวกำหนดลักษณะการทำงานของทั้ง 3 I/O พอร์ต

บล็อกกลุ่มสุดท้ายที่จะกล่าวถึง ได้แก่ DATA BUS BUFFER และ READ/WRITE CONTROL LOGIC ซึ่งบล็อกเหล่านี้จะเป็นส่วนที่ติดต่อกับ CPU, DATA BUS BUFFER จะเป็นบัฟเฟอร์ให้กับบัสข้อมูลของ CPU ส่วน READ/WRITE CONTROL LOGIC จะเป็นส่วนที่ควบคุมให้ข้อมูลเข้าหรือออกจากรีจิสเตอร์ภายใน

### รายละเอียดการจัดเรียงขาของ 8255

การจัดขาได้แสดงไว้ในรูปที่ 2.39 รายละเอียดของแต่ละขามิดังนี้คือ

D0-D7 : เป็นสายข้อมูลอินพุต/เอาต์พุตแบบสองทิศทาง (BI DIRECTIONAL BUS)

จะเป็นทางผ่านของข้อมูลระหว่างพอร์ตต่างๆของ 8255 กับบัสข้อมูลของ Z80 โดยใช้ประโยชน์ด้านการคำนวณร่วมกันทั้งสี่ชิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

/CS (CHIP SELECT INPUT) : เมื่อขานี้มีสถานะลอจิกเป็น “0” CPU จะสามารถที่จะอ่านหรือเขียนข้อมูลกับ 8255 ได้

/RD (READ INPUT) : เมื่อขานี้มีสถานะลอจิกเป็น “0” และสัญญาณ /CS มีลอจิกเป็น “0” ข้อมูลจาก 8255 จะปรากฏสู่ระบบบัสข้อมูล CPU ก็จะสามารถอ่านข้อมูลออกไปได้

/WR (WRITE INPUT) : เมื่อขานี้มีสถานะลอจิกเป็น “0” และสัญญาณ /CS มีลอจิกเป็น “0” ข้อมูลจากระบบบัสข้อมูลจะถูกเขียนเข้าไปยัง 8255 ได้

A0-A1 (ADDRESS INPUT) : จะเป็นตัวกำหนดการเลือกใช้รีจิสเตอร์ภายในของ 8255

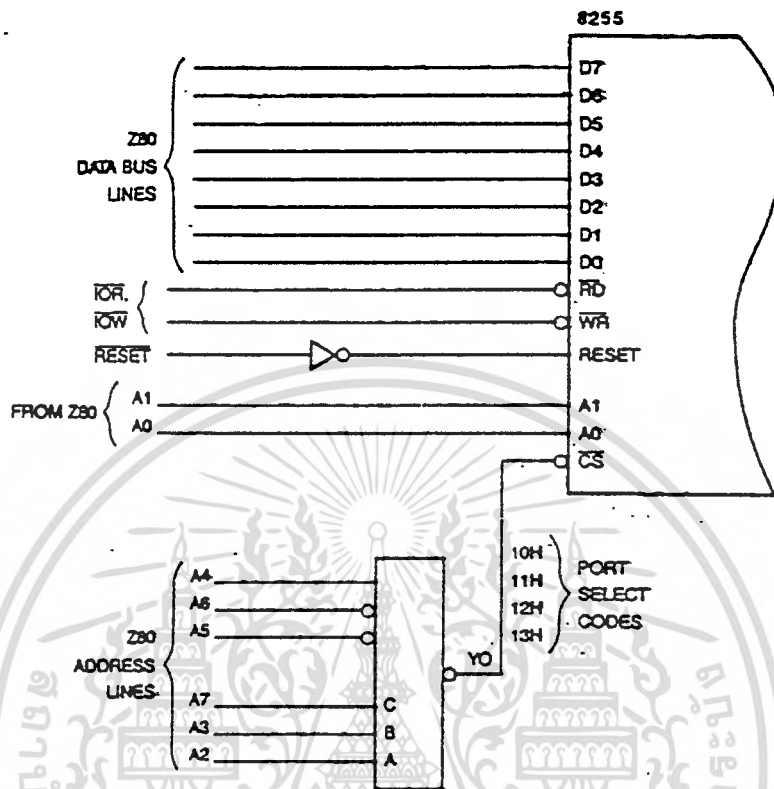
RESET : เมื่อขานี้มีสถานะเป็น “1” 8255 จะอยู่ในสภาวะรีเซตทุกๆพอร์ทของ 8255 จะถูกเซตให้อยู่ในโหมดอินพุท

PA0-PA7,PB0-PB7 : ขาสัญญาณเหล่านี้จะถูกใช้เป็นพอร์ท I/O ขนาด 8 บิตใช้ต่อกับอุปกรณ์ภายนอกอื่นๆ

PC0-PC7 : ขาสัญญาณนี้ถูกใช้เป็นพอร์ท I/O ขนาด 8 บิต ซึ่งกลุ่มของขาสัญญาณเหล่านี้สามารถแบ่งออกเป็น 2 กลุ่มโดยแต่ละกลุ่มมีขนาด 4 บิตได้ กลุ่มแรกจะควบคุม PB0-PB7 และกลุ่มที่ 2 ใช้ควบคุม PA0-PA7

การต่อ 8255 เข้ากับ Z80

ในการต่อ 8255 เข้ากับระบบของ Z80 นั้น สัญญาณต่างๆที่เกิดขึ้นจะเหมือนกับขบวนการติดต่อกับ I/O ดังที่เคยกล่าวมาแล้ว โดยจะต้องเอาสัญญาณ A0-A7 จาก Z80 มาถอดรหัสเพื่อสร้างสัญญาณเลือกพอร์ท แต่เนื่องจาก 8255 มีขา ADDRESS INPUT อยู่แล้ว 2 ขา (A0,A1) ซึ่งโดยปกติแล้วขา A0,A1 นี้จะต่อเข้าโดยตรงกับ A0,A1 จากบัสแอดเดรส นั่นคือ 8255 หนึ่งตัวจะใช้ค่าพอร์ทแอดเดรสถึง 4 ค่า ( $2^2$ ) ส่วนสัญญาณอีก 6 เส้น (A2-A7) จะนำไปถอดรหัสเพื่อทำสัญญาณเลือกชิพให้แก่ 8255 ดังแสดงในรูปที่ 2.40

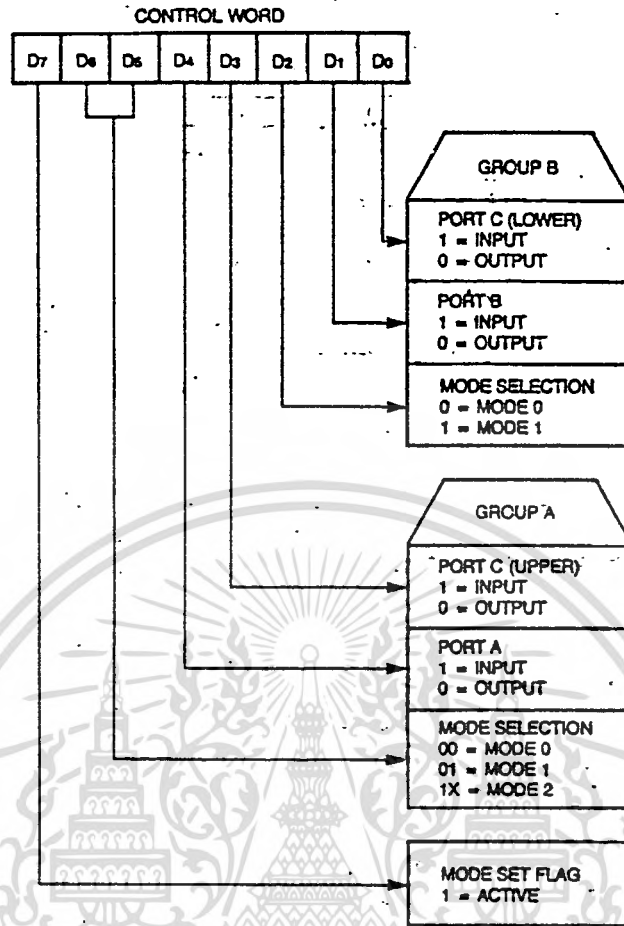


รูปที่ 2.40 แสดงผังวงจรสมบูรณ์ของการเชื่อมต่อ 8255 เข้ากับระบบของ Z80

### 8255 READ และ WRITE REGISTER

ขณะนี้เราได้ทำการต่อ 8255 เข้ากับระบบของ Z80 แล้ว ต่อไปเราจะศึกษาการโปรแกรมใช้งาน 8255 เพื่อให้ทำงานตามที่เรต้องการได้

รูปที่ 2.41 จะแสดงรายละเอียดของแต่ละบิตของรีจิสเตอร์ควบคุมนี้ ต่อไปเราจะกล่าวถึงลักษณะการทำงานของ 8255 ทั้ง 3 โหมด และการโปรแกรมให้อยู่ในโหมดต่างๆได้ดังต่อไปนี้คือ



รูปที่ 2.41 แสดงรายละเอียดแต่ละบิตของรีจิสเตอร์ควบคุมของ 8255

### โหมด 0 : BASIC REGISTER I/O

ในการใช้ 8255 ให้อยู่ในโหมด 0 นั้นเราจะต้องส่งคำสั่งควบคุมให้แก่รีจิสเตอร์ควบคุม (CONTROL WORD) ก่อน คำสั่งควบคุมนี้จะกำหนดลักษณะการทำงานให้แก่แต่ละพอร์ทของ 8255 ตัวอย่างหนึ่งของคำสั่งควบคุมที่จะส่งให้ 8255 ทำงานอยู่ในโหมด 0 นี้ได้แก่

| D7 | D6 | D5 | D4 | D3 | D2 | D1 | D0 |
|----|----|----|----|----|----|----|----|
| 1  | 0  | 0  | 0  | 0  | 0  | 0  | 0  |

จากรูปที่ 2.41 เราจะเห็นว่า

บิต D7 เป็นตัวกำหนดว่าเป็นคำสั่งควบคุม (CONTROL WORD)

บิต D6 และ D5 กำหนดโหมดการทำงานของพอร์ท A D6,D5 มีค่าเป็น "0" แสดงว่าอยู่ในโหมด 0

บิต D4 = "0" กำหนดให้พอร์ท A เป็นพอร์ทเอาต์พุต

บิต D3 = "0" ใช้พอร์ท C 4 บิตบนเป็นพอร์ทเอาต์พุต

เอกสารนี้เป็นเอกสารลิขสิทธิ์ของมหาวิทยาลัยเทคโนโลยีพระจอมเกล้าธนบุรี หากมีการนำเอกสารนี้ไปใช้โดยไม่ได้รับอนุญาตให้มหาวิทยาลัยเทคโนโลยีพระจอมเกล้าธนบุรี ขอสงวนสิทธิ์ในการดำเนินคดีตามกฎหมาย

บิต D2= "0" เซ็ตโหมดของพอร์ท B ให้พอร์ท B อยู่ในโหมด 0

บิต D1= "0" เซ็ตพอร์ท B เป็นพอร์ทเอาต์พุต

บิต D0= "0" เซ็ตพอร์ท C ให้ 4 บิตล่างเป็นพอร์ทเอาต์พุต

คำสั่งควบคุมนี้จะกำหนดให้พอร์ททั้ง 3 ของ 8255 ทำงานอยู่ในโหมด 0 และเป็นพอร์ทเอาต์พุตซึ่งจะได้สายสัญญาณซึ่งสามารถติดต่อกับอุปกรณ์ภายนอกได้ถึง 24 สาย คำสั่ง Z80 ที่จะเซ็ทให้ 8255 อยู่ในลักษณะดังกล่าวได้แก่

LD A, 80H : เซ็ตคำสั่งควบคุม  
OUT (13H), A : ส่งคำสั่งควบคุมให้ 8255

เมื่อ Z80 ทำการ EXECUTE คำสั่งข้างต้นแล้ว 8255 จะถูกเซ็ทให้พอร์ททุกพอร์ทเป็นพอร์ทเอาต์พุต และอยู่ในโหมด 0 เราจะสามารถส่งข้อมูลออกไปยังพอร์ทต่างๆได้ด้วยคำสั่ง OUT ของ Z80 ตัวอย่างเช่น เราต้องการส่ง 23H ไปยังพอร์ท A, 41H ไปยังพอร์ท B และ 73H ไปยังพอร์ท C เราจะต้องให้ Z80 ทำตามโปรแกรมลักษณะดังนี้

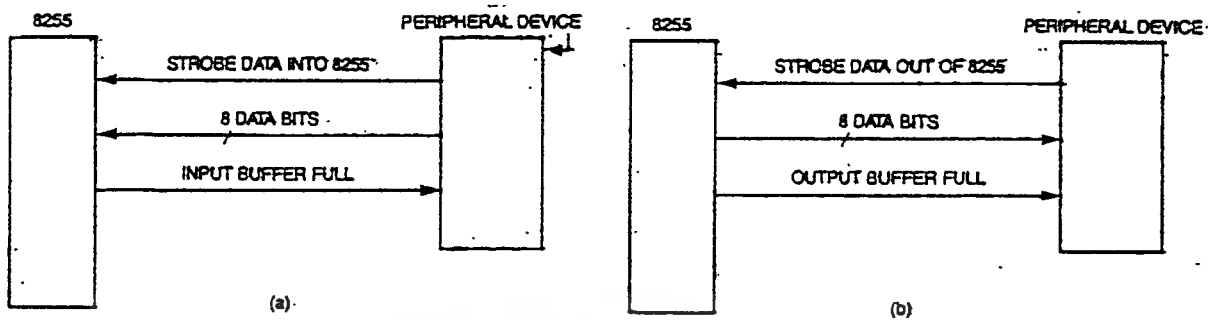
LD A, 32H : เซ็ตข้อมูลให้พอร์ท A  
OUT (10H), A : ส่งข้อมูลให้พอร์ท A  
LD A, 41H : เซ็ตข้อมูลให้พอร์ท B  
OUT (11H), A : ส่งข้อมูลให้พอร์ท B  
LD A, 73H : เซ็ตข้อมูลให้พอร์ท C  
OUT (12H), A : ส่งข้อมูลให้พอร์ท C

หลังจากที่คำสั่งเหล่านี้ถูก EXECUTE แล้วพอร์ท A,B และ C ของ 8255 จะมีข้อมูลต่างๆที่ส่งไปให้ปรากฏอยู่

#### การใช้งาน 8255 ในโหมด 1

การทำงานของ 8255 ในโหมด 1 นี้เป็นการทำงานในลักษณะของการ HANDSHAKE, พอร์ท A และ พอร์ท B จะเป็นพอร์ทข้อมูลส่วนพอร์ท C นี้จะถูกใช้เป็นสัญญาณ HANDSHAKE โดย 4 บิตบนจะเป็นสัญญาณ HANDSHAKE ให้กับพอร์ท A และ 4 บิตล่างจะเป็นสัญญาณ HANDSHAKE ให้กับพอร์ท B

หลักการรับส่งข้อมูลในวิธีการของ HANDSHAKE นี้ คือการให้อุปกรณ์ภายนอกส่งสัญญาณแสดงสถานะความพร้อมให้กับ 8255 ดังแสดงในรูปที่ 2.42



รูปที่ 2.42 บล็อกไดอะแกรมแสดงลักษณะการทำงานการติดต่อระหว่าง 8255 กับอุปกรณ์ภายนอกในลักษณะ HANDSHAKE

ในรูปที่ 2.42A นี้ข้อมูลจะถูกส่งออกจากอุปกรณ์ภายนอกเข้าสู่ 8255 ก่อนที่อุปกรณ์ภายนอกจะเขียนข้อมูลให้แก่ 8255

ในรูปที่ 2.42B 8255 จะทำหน้าที่เป็นตัวส่งข้อมูลให้กับอุปกรณ์ภายนอก ก่อนที่ 8255 จะส่งข้อมูลให้กับอุปกรณ์ภายนอกนั้นจะต้องเช็ค OUTPUT BUFFER FULL FLAG เสียก่อนเพื่อบ่งบอกให้อุปกรณ์ภายนอกทราบว่าขณะนี้ 8255 มีข้อมูลพร้อมที่จะส่งออกไปแล้ว เมื่ออุปกรณ์ภายนอกส่งสัญญาณ STROBE รับเอาข้อมูลเข้าไปแล้ว INPUT BUFFER FULL FLAG จะเปลี่ยนเป็นเท็จเพื่อบ่งบอกให้อุปกรณ์ภายนอกรู้ว่าขณะนี้ไม่มีข้อมูลอยู่ใน 8255, Z80 สามารถส่งข้อมูลใหม่ออกไปให้ 8255 ได้

รายละเอียดของแต่ละบิตของพอร์ท C ในขณะที่ 8255 ทำงานในโหมด 1 มีดังรูปที่ 2.43

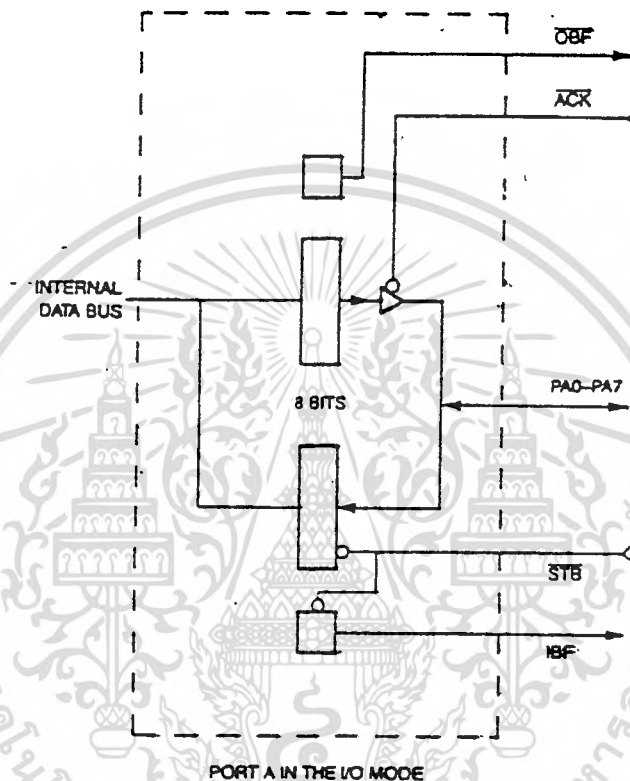
|     | OUT               | IN                |
|-----|-------------------|-------------------|
| PC0 | INTR <sub>B</sub> | INTR <sub>B</sub> |
| PC1 | IBF <sub>B</sub>  | OBF <sub>B</sub>  |
| PC2 | STB <sub>B</sub>  | ACK <sub>B</sub>  |
| PC3 | INTR <sub>A</sub> | INTR <sub>A</sub> |
| PC4 | STB <sub>A</sub>  | IO                |
| PC5 | IBF <sub>A</sub>  | IO                |
| PC6 | IO                | ACK <sub>A</sub>  |
| PC7 | IO                | OBFA              |

รูปที่ 2.43 แสดงรายละเอียดและหน้าที่ของแต่ละขาของพอร์ท C ในโหมด 1

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## การใช้งาน 8255 ในโหมด 2

การใช้งานในโหมดนี้จะเป็นการใช้งานในลักษณะที่ให้พอร์ท A เป็นพอร์ทข้อมูลแบบสองทิศทาง เมื่อ 8255 ถูกโปรแกรมให้พอร์ท A อยู่ในโหมด 2 นี้แล้วพอร์ท A จะมีลักษณะการทำงานตามบล็อกไดอะแกรมรูปที่ 2.44



รูปที่ 2.44 บล็อกไดอะแกรมแสดงการทำงานของพอร์ท A ในโหมด 2

การทำงานในโหมดนี้ก็คือ การใช้พอร์ท A เป็นอินพุตและเอาต์พุตแลตช์

เอาต์พุตแลตช์ ( OUTPUT LATCH) หมายถึง การเก็บเอาข้อมูลไว้เพื่อรออุปกรณ์ภายนอกมารับเอาข้อมูลออกไป

อินพุตแลตช์ ( INPUT LATCH) หมายถึง การเก็บข้อมูลที่อุปกรณ์ภายนอกส่งเข้ามาเพื่อรอให้ CPU อ่านเข้าไป

ต่อไปเราจะมาพิจารณาการทำงานของบล็อกไดอะแกรมดังรูปที่ 2.44 และศึกษาว่า จะมีการรับ /ส่ง ข้อมูลกับอุปกรณ์ภายนอกได้อย่างไร

ในการส่งข้อมูลให้อุปกรณ์ภายนอกนั้น ขั้นตอนแรกที่เกิดขึ้นก็คือ CPU จะต้องส่งข้อมูลไปแลตช์ไว้ในพอร์ท A ซึ่งจะทำให้ OUTPUT BUFFER FULL (OBF) ถูกเจ็ทให้เป็นจริง

เอกสารนี้เป็นเอกสารลิขสิทธิ์ของมหาวิทยาลัยเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สัญญาณนี้จะบอกให้อุปกรณ์ภายนอกทราบว่าขณะนี้พอร์ท A มีข้อมูลอยู่และบอกให้ CPU รู้ว่าข้อมูลของพอร์ท A ที่ส่งไปให้มันยังไม่ถูกอุปกรณ์ภายนอกอ่านออกไป อุปกรณ์จะต้องส่งสัญญาณ /ACK (ACKNOWLEDGE) ให้กับ 8255 สัญญาณนี้จะเป็นการ ENABLE ให้ข้อมูลที่อยู่ในพอร์ท A ส่งออกไปยัง PA0-PA7 และเป็นการรีเซ็ต OBF เพื่อเป็นการบ่งบอกให้ CPU รู้ว่าข้อมูลที่อยู่ในพอร์ท A ถูกอุปกรณ์ภายนอกนั้นอ่านออกไปแล้ว และสามารถส่งข้อมูลใหม่ให้พอร์ท A ได้อีก

ในการรับส่งข้อมูลจากอุปกรณ์ภายนอกนั้น อุปกรณ์ภายนอกต้องตรวจสอบสถานะของ INPUT BUFFER FULL (/IBF) เสียก่อน ถ้า /IBF มีลอจิกเป็น "1" แสดงว่าขณะนี้พอร์ท A มีข้อมูลอยู่ CPU ยังไม่ได้อ่านข้อมูลเข้าไป แต่ถ้ามีลอจิกเป็น "0" แสดงว่าไม่มีข้อมูลอยู่ในพอร์ท A อุปกรณ์ภายนอกจะส่งข้อมูลและสัญญาณ STROBE (/STB) ให้ 8255 สัญญาณนี้จะสั่งให้มีการส่งข้อมูลไปแลตช์ไว้ในพอร์ท A และเซตให้ /IBF มีสถานะเป็นจริง และ CPU จะตรวจสอบสถานะของ /IBF ได้โดยการอ่านข้อมูลจากพอร์ท C เมื่อ /IBF มีสถานะเป็นจริงแสดงว่ามีข้อมูลที่พร้อมที่จะให้ CPU อ่านออกไป เมื่อ CPU อ่านข้อมูลออกไปแล้ว /IBF จะมีสถานะเป็นลอจิก "0" และอุปกรณ์ภายนอกสามารถส่งข้อมูลใหม่เข้ามาได้

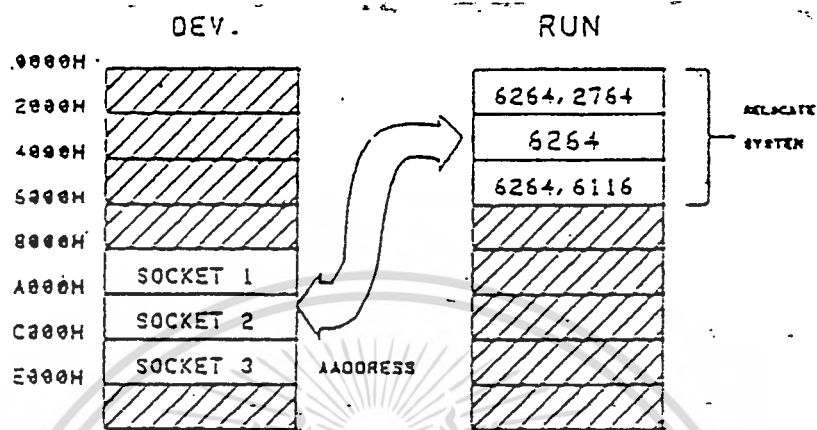
ในการใช้งานในโหมด 2 พอร์ท C จะเป็นตัวแสดงสถานะของสัญญาณดังกล่าว และรายละเอียดของแต่ละบิตจะแสดงดังรูป 2.45

| PORT C LINE | DEFINITION          |
|-------------|---------------------|
| PC0         | IO/ $\overline{IO}$ |
| PC1         | $\overline{IO}$     |
| PC2         | $\overline{IO}$     |
| PC3         | INTRA               |
| PC4         | STB <sub>A</sub>    |
| PC5         | IBF <sub>A</sub>    |
| PC6         | ACK <sub>A</sub>    |
| PC7         | OBF <sub>A</sub>    |

รูปที่ 2.45 แสดงรายละเอียดของแต่ละขาของพอร์ท C ในการใช้งานในโหมด 2



DIP 4 เลือก SOCKET 4 ให้ใช้กับ 6116 (ตัวที่ 3,4 นี้จะ ON ตัวใดตัวหนึ่งเท่านั้น)  
 รายละเอียดการจัดหน่วยความจำและ PORT ของ Z80 CP-A



รูปที่ 2.47

การจัดหน่วยความจำของ CP-A นี้จะอ้าง Address เริ่มต้นที่ 8000H เพื่อให้ทำโปรแกรมได้สะดวกจากเครื่องอีทีบอร์ด ( ET BROAD ) และเมื่อนำโปรแกรมไปใช้งาน CP-A จะมีระบบ Relocate เพื่อให้หน่วยความจำไปเริ่มต้นที่ 0000H ได้ (ในโหมด RUN) เพื่อให้เป็น Address ที่ทำงานเมื่อเริ่มเปิดเครื่องหรือกด RESET

ระบบ Port จะทำการ Decode ไว้ทั้งหมด 8 ช่วง และจะสามารถเลือกเบอร์ได้ 7 ตำแหน่ง โดยการ jump สายบนแผ่นปรีน ซึ่งมี เบอร์ Port ดังนี้

| ตำแหน่ง | เบอร์ Port                  |
|---------|-----------------------------|
| 1       | 00-1F (จริงๆใช้เพียง 00-03) |
| 2       | 20-3F                       |
| 3       | 60-7F                       |
| 4       | 80-9F                       |
| 5       | A0-BF                       |
| 6       | C0-DF                       |
| 7       | E0-FF                       |

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

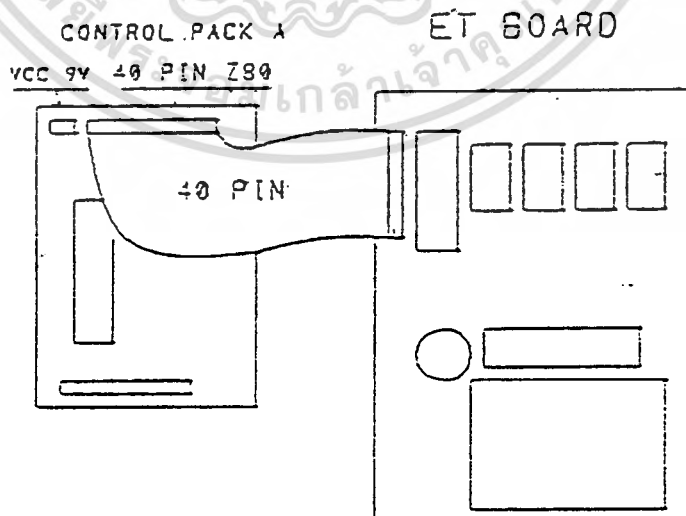
ส่วนเบอร์ Port 40-5F นั้น ไว้สำหรับการขยายได้อีกทางด้าน 8255 Connector ใน โหมด Dev นี้ ผู้ใช้จะเห็น Address ในตำแหน่ง 8000H A000H C000H ตามลำดับและถ้า นำ Ram 6264 มาเสียบที่ Socket 1 ก็จะสามารถทำโปรแกรมได้เลย โดยใช้การทำงานต่างๆ ทั้งหมดของ อธิบอร์ค การนำ Ram 6264 มาเสียบ จะต้องเปลี่ยน JP1 ให้อยู่ตำแหน่ง 62 ด้วย ซึ่ง จะต้องคัดลายปรี้นที่ Default ไว้ที่ตำแหน่ง 27 ก่อน แล้วจึงต่อสายใหม่

### การนำไปใช้งาน

1. ทำการปรับ DIP Switch ตัวที่ 1 ให้อยู่ในตำแหน่ง Run และตัวที่ 2 ให้อยู่ในตำแหน่ง ON
2. ใส่ตัว CPU Z80 ลงใน Socket ให้เรียบร้อย
3. นำโปรแกรมที่พัฒนาเรียบร้อยแล้วลงในตัว EPROM โดยใช้เครื่อง EP Burner หรือจะใช้เครื่องอื่นๆก็ได้ โดยโปรแกรมนี้จะต้องแปลงให้ทำงานที่ Address 0000H จากนั้นก็นำมาใส่ลงบอร์ดที่ Socket 1
4. ถ้ามีการแก้ไขที่ JP1 ในขณะพัฒนา ก็ต้องเปลี่ยนกลับเป็นตำแหน่ง 27 เหมือนเดิม

### การใช้งานร่วมกับอธิบอร์ค

Z80 CP-A มีโหมดในการทำงาน 2 ลักษณะคือ การพัฒนาและการนำไปใช้งาน ในขั้นตอนของการพัฒนานั้นจะทำได้โดยเสียบเข้ากับอธิบอร์คได้เลย แล้วจึงนำโปรแกรมาทำงานโดยผ่าน EPROM หรือ RAM-PACK ซึ่งผู้ใช้งานจะต้องเขียนโปรแกรมโดยอ้าง Address ตามลักษณะที่กล่าวไว้ ในการเสียบกับอธิบอร์คนั้นจะแสดงได้ดังภาพต่อไปนี้



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

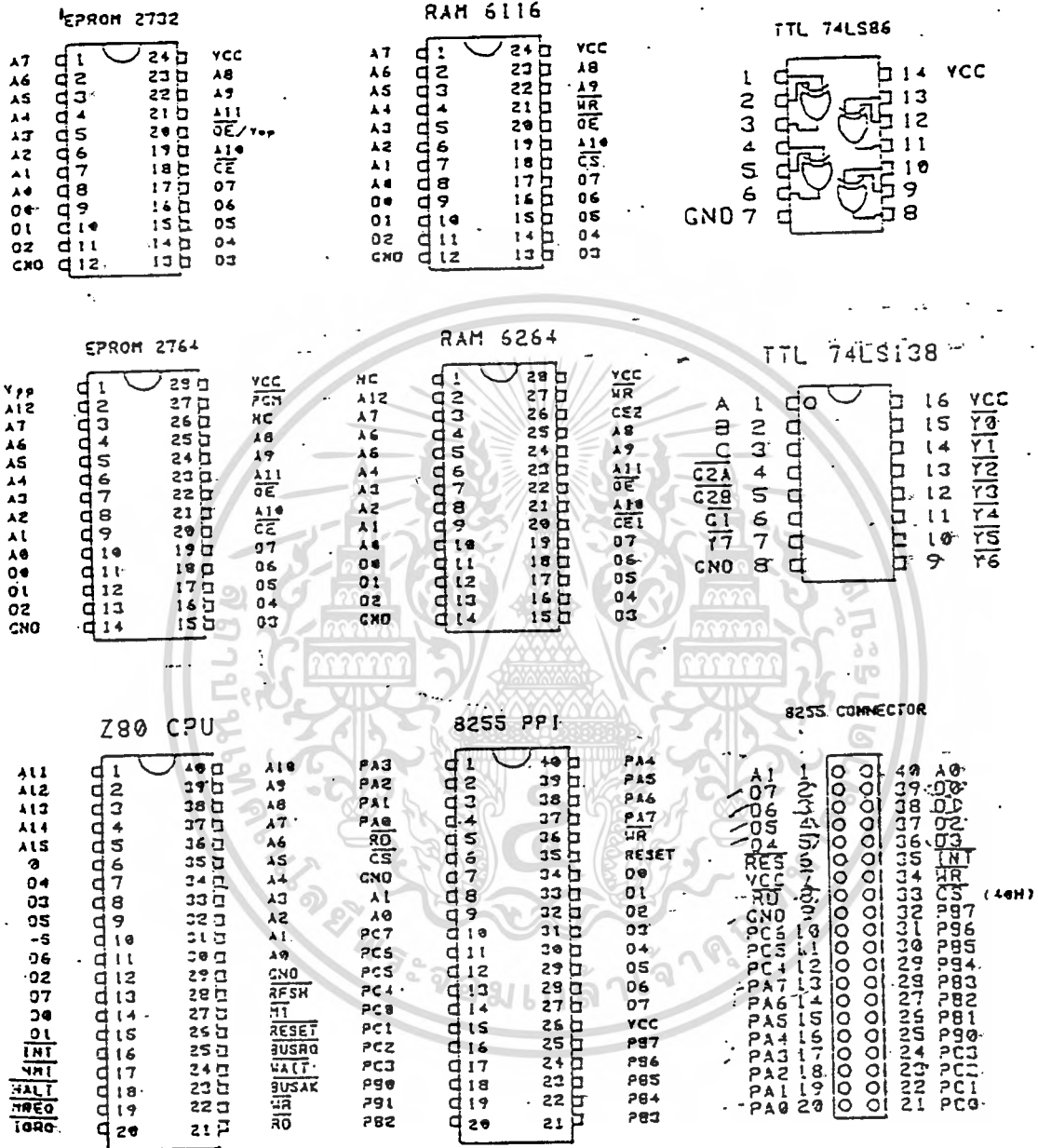
ซึ่งจะพอสรุปขั้นตอนได้ดังนี้

1. ทำการปรับ DIP Switch ตัวที่ 1 ให้อยู่ในตำแหน่ง Dev. และตัวที่ 2 ให้อยู่ในตำแหน่ง off ซึ่งเป็นการตัดความถี่ Clock ให้เหลือเพียง Clock จากอีทีบอร์ด
2. ผู้ใช้จะต้องเอา CPU Z80 ตัวใดตัวหนึ่งออกจากบอร์ด คือเอาบอร์ด CP-A หรือบนอีทีบอร์ดก็ได้ เพื่อที่จะไม่ซ้อนกัน
3. แหล่งจ่ายไฟ 5 โวลต์ ของทั้งสองบอร์ดจะต่อถึงกันหมด และเนื่องจากใช้กระแสน้ำขึ้น ผู้ใช้ควรจะใช้ Adapter ที่มีขนาดใหญ่พอสมควร และก็เปรียบกับบอร์ดใดบอร์ดหนึ่งได้เลย ทั้งนี้ถ้ามีการต่อ Port ออกไปใช้งาน ก็ต้องดูว่ากระแสจ่ายได้ เพียงพอหรือไม่ ถ้าไม่พอก็ควรแยกแหล่งจ่ายไฟออกเป็นอีกชุดหนึ่ง



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## รายละเอียดขา IC และ Connector



รูปที่ 2.49

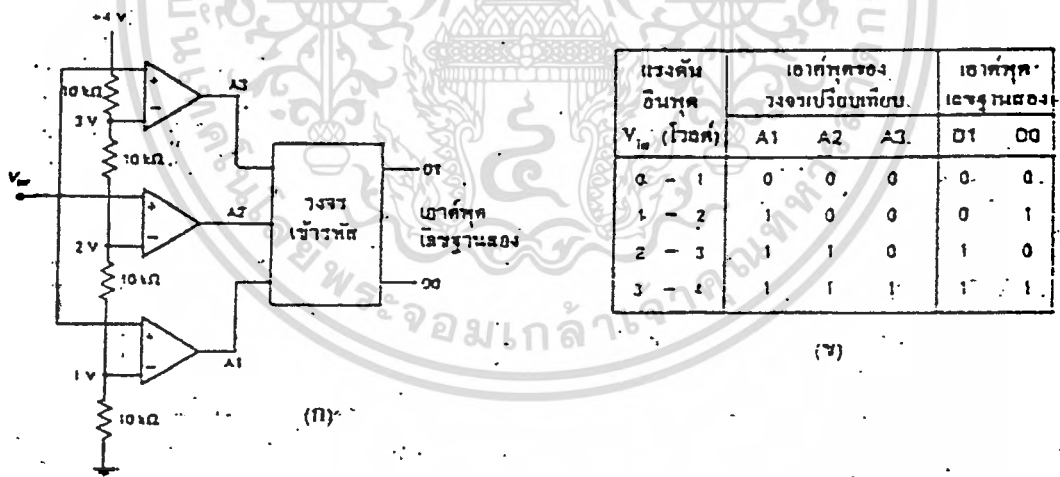
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



## 2.4 วงจรแปลงขนาดออกเป็นดิจิทัล (Analog to Digital converter)

วงจรเปลี่ยนสัญญาณอนาลอกเป็นดิจิทัล ที่ใช้กันอยู่ทั่วไปมีหลายแบบด้วยกันคือ  
แบบใช้วงจรเปรียบเทียบขนานหรือแบบแฟลช (Parallel Comparator  
Simultaneous "Flash" A/D CONVERTER)

วงจรแบบนี้จะใช้ วงจรเปรียบเทียบที่ต่อขนานกันดังรูปที่ 2.51(ก) ประกอบด้วย  
ออปแอมป์ที่ต่อเป็นวงจรเปรียบเทียบ และตัวต้านทานต่อไว้เพื่อแบ่งแรงดันที่ขาอินพุตแบบกลับ  
(inverting) หรือขาลบให้มีขนาดต่างๆ กัน จากหลักการของวงจรเปรียบเทียบทั่วไป เมื่อแรงดัน  
อินพุตที่ขาอินพุตแบบไม่กลับ(noninverting) หรือขาบวก มีค่าสูงกว่าที่ขาลบเอาต์พุตจะได้แรง  
ดันค่าสูง ดูได้จากตารางที่รูป 2.51(ข) ซึ่งเอาต์พุตที่ได้จากวงจรเปรียบเทียบนี้จะนำไปเข้ารหัส  
ให้เป็นเลขฐานสองต่อไปเมื่อต้องการวงจรที่ความละเอียดสูงขึ้น จำเป็นต้องใช้วงจรเปรียบเทียบ  
เพิ่มขึ้นโดยหาจำนวนวงจรเปรียบเทียบได้จาก สูตร  $2^N - 1$  เมื่อ N แทนจำนวนบิตหรือความ  
ละเอียดที่ต้องการ แต่จะเห็นว่าที่ความละเอียด 8 บิตต้องใช้วงจรเปรียบเทียบมากถึง 255 ตัวซึ่ง  
เป็นข้อเสียของวงจรแบบนี้ ข้อเสียอีกประการหนึ่งคือเอาต์พุตไม่เป็นเลขฐานสองต้องมีวงจรเพิ่ม  
เติมไปทำการเข้ารหัส ข้อดีของวงจรคือมีความเร็วสูงมาก บางครั้งจึงต้องเรียกววงจรเอชดีแบบนี้  
ว่าแบบ แฟลช วงจรชนิดนี้ใช้เวลาในการแปลงได้เร็วในระดับนาโนวินาที ที่เดียว



รูปที่ 2.51 (ก) แสดงการต่อวงจร PARALLEL COMPARATOR A/D CONVERTER

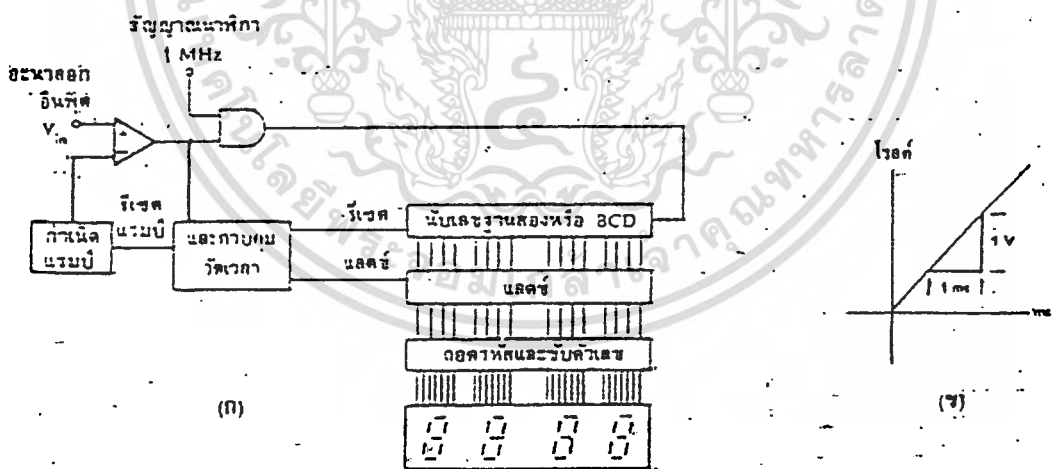
(ข) ตารางความสัมพันธ์ระหว่างแรงดันอินพุตที่เป็นอนาลอกกับเอาต์พุตที่เป็นดิจิทัล

### วงจรเอชดีที่ใช้การอินทิเกรต

เอกสารนี้เป็นเอกสารของวงจรเปลี่ยนสัญญาณเอชดีที่ใช้เทคนิคการอินทิเกรตสัญญาณมีหลายแบบ คือขั้นตอนการคำนวณ  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดลอกเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 1. แบบสโโลปเดี่ยวหรือแบบแรมป์ (Single Ramp หรือ Single Slope A/D Converter)

วงจรเอทิดีแบบนี้แสดงไว้ดังรูปที่ 2.52 ประกอบไปด้วย วงจรกำหนดสัญญาณแรมป์, วงจรเปรียบเทียบ, วงจรรนับ BCD เมื่อเริ่มทำการเปลี่ยนสัญญาณ สัญญาณแรมป์และวงจรรนับจะถูกรีเซ็ตให้เป็น 0 แรงดันอะนาล็อกถูกป้อนไปยังวงจรเปรียบเทียบทางขาอินพุตบวก เมื่อแรงดันที่ขานี้มีค่าเป็นบวกมากกว่าลบ วงจรเปรียบเทียบจะให้เอาต์พุตออกมาเป็นระดับ high ทำให้แอนคเกทปล่อยสัญญาณนาฬิกาผ่านไปยังวงจรรนับได้และทำให้เกิดสัญญาณแรมป์เมื่อสัญญาณแรมป์มีแรงดันเป็นบวกขึ้นเรื่อยๆจนมากกว่าระดับแรงดันอินพุตเอาต์พุตของวงจรเปรียบเทียบก็ตกลงมาเป็นระดับ LOW ปิดแอนคเกทไม่มีสัญญาณผ่านไปให้วงจรรนับ วงจรรนับจะหยุดนับและเก็บค่าไว้ที่วงจรแลตช์จากนั้นจึงทำการรีเซ็ตวงจรรนับและวงจรกำหนดสัญญาณแรมป์ สมมุติให้สัญญาณนาฬิกามีความถี่ 1 MHz วงจรรนับ BCD 4 หลักแรงดันอินพุต  $V_{IN} = 2V$  สัญญาณแรมป์มีความชัน 1 V/ms ดังแสดงในรูป 2.52(ข) จากจุดเริ่มต้นจนถึงแรงดันสูงสุด(2 โวลต์)สัญญาณแรมป์ใช้เวลา 2 ms หลังจากนั้นจึงปิดสัญญาณที่ส่งไปให้วงจรรนับ ซึ่งวงจรรนับทำการนับถึง 200 ลูก วงจรลักษณะนี้นักนำไปใช้งานในการเปลี่ยนเวลาเป็นขนาดของสัญญาณหรืออาจใช้ในดิจิตอลโวลท์มิเตอร์ แต่ไม่ใช้กับงานที่ต้องการความถูกต้อง เนื่องจากการเปลี่ยนแปลงในแหล่งกำเนิดสัญญาณแรมป์ขึ้นกับอุณหภูมิและผลตอบสนองต่อสัญญาณอินพุตทำให้ไม่มีความถี่ที่คงที่ดังนั้นจึงมีการปรับปรุงให้ดีขึ้นกลายเป็นแบบสโโลปคู่

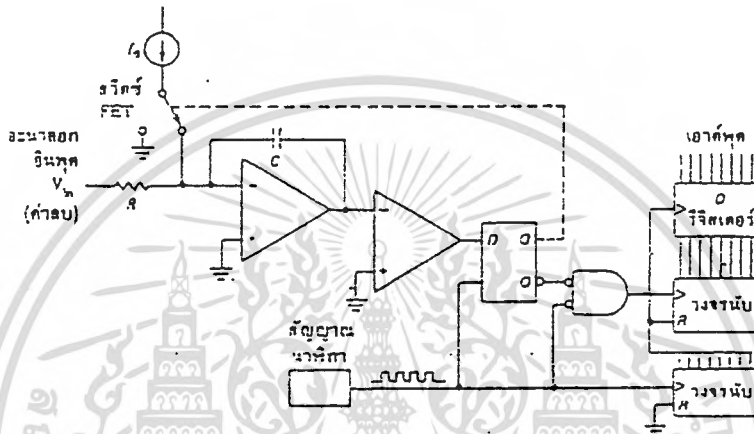


รูปที่ 2.52 วงจรเปลี่ยนสัญญาณเอทิดีแบบสโโลปเดี่ยว

(ก) แสดงบล็อกไดอะแกรม (ข) ความชันของสัญญาณแรมป์

## 2. แบบเดลต้า-ซิกม่า (Delta-Sigma A/D Converters)

จากวงจรรูปที่ 2.54 เมื่อมีแรงดันอินพุตป้อนเข้ามาที่วงจรอินทิเกรเตอร์จะให้เอาต์พุตไปเข้าวงจรเปรียบเทียบ เปรียบเทียบกับแรงดันคงที่ (จากรูปคือกราวด์) พัลส์ของกระแสที่ได้ขึ้นอยู่กับเอาต์พุตของวงจรเปรียบเทียบ โดยการสวิทช์ที่ทำงานจากเฟดจะควบคุมให้กระแสเข้าไปยังที่จุลรวมหรือลงกราวด์ไป ส่วนวงจรนับจะนับจำนวนพัลส์ด้วยหลักการที่คล้ายกัน



รูปที่ 2.54 วงจรเปลี่ยนสัญญาณแบบเดลต้า-ซิกม่า

### ข้อสรุปของเหตุκιแบบอินทิเกรตสัญญาณ

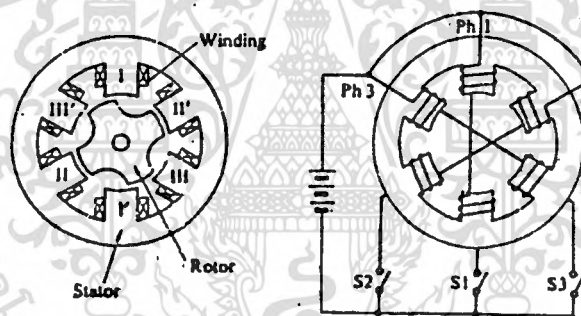
จุดสำคัญของอินทิเกรตคือเทคนิคคืออินพุตที่ให้กับวงจรอินทิเกรเตอร์ต้องเป็นกระแส ไอซีคอนเวอร์เตอร์บางตัวอาจมีอินพุตให้สองขา แต่จะมีขาหนึ่งต่อตรงกับจุด summing point ใช้กับอุปกรณ์ที่เป็นแหล่งจ่ายกระแสโดยตรง ถ้าให้อินพุตเป็นกระแสก็ไม่ต้องคำนึงถึงแรงดันออฟเซตของวงจรอินทิเกรเตอร์ แต่ถ้าใช้กับอินพุตที่เป็นแรงดันต้องปรับออฟเซตของออฟแอมป์เสียก่อน การใช้อินพุตเป็นกระแสทำให้ย่านการใช้งานทางไฟสลับกว้าง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## 2.5 สเต็ปป์มอเตอร์ (Stepping Motors)

สเต็ปป์มอเตอร์ แบ่งออกได้หลายชนิดด้วยกัน เช่น Variable-reluctance, Permanent Magnet, Hybrid, Linear และอื่นๆ แต่นิยมใช้มี 2 ชนิด คือแบบ Variable Reluctance และ Permanent Magnet

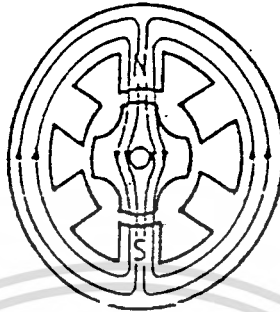
**Variable-Reluctance** หรือเรียกสั้นๆ ว่า VR มอเตอร์จะเป็นพื้นฐานสำคัญในการทำงานของสเต็ปป์มอเตอร์ ซึ่งจะช่วยให้เข้าใจการทำงานของสเต็ปป์มอเตอร์ชนิดอื่นๆ ได้ง่ายขึ้น รูปที่ 5.1 เป็นภาพหน้าตัดและยังแสดงถึงการพันขดลวดของ VR มอเตอร์ แบบ 3 เฟส มีขั้วเหนือและขั้วใต้อยู่ตรงข้ามกัน 3 คู่ โดยจะพันขดลวดแบบอนุกรมกันในแต่ละชุด ถ้ามีการกระตุ้นเฟสเกิดขึ้น ขั้ว I, II, III จะเป็นขั้วได้ และ ขั้ว I, II, III จะเป็นขั้วเหนือ ทั้งโรเตอร์และสเตเตอร์จะทำจากเหล็กผสมซิลิกอนซึ่งเป็นวัสดุที่มีความซึมซับ (Permeability) สูง สามารถให้เส้นแรงแม่เหล็กไหลผ่านได้มาก



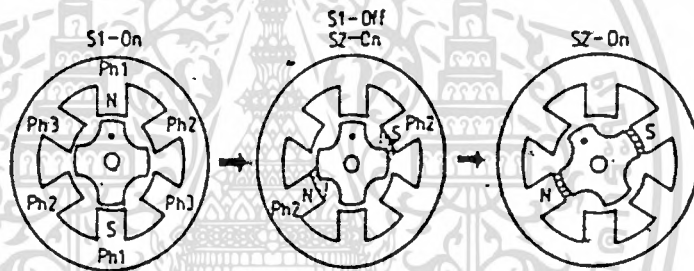
รูปที่ 2.56 ภาพหน้าตัดและการพันขดลวด VR สเต็ปป์มอเตอร์แบบ 3 เฟส

การทำงานจะเริ่มจากการกระตุ้นที่เฟส 1 ก่อน (S1 “ON”) ทำให้เส้นแรงแม่เหล็กเกิดขึ้น ดังรูปที่ 2.57 ตัวโรเตอร์จะพยายามวางตำแหน่งตัวเอง ให้อยู่ในทิศทางที่ทำให้เกิดค่าความต้านทานแม่เหล็กที่น้อยที่สุด ในขั้ว I-I’ ในขณะที่เริ่มต้นที่จะกระตุ้นที่เฟส II (S1 “OFF”, S2 “ON”) ดังรูปที่ 2.58 เส้นแรงแม่เหล็กจะไม่อยู่ในแนวทาง เดิมที่สะดวก จึงทำให้ค่าความต้านทานแม่เหล็กมีค่าสูง ตัวโรเตอร์ก็จะพยายามปรับตัวเองเพื่อให้ค่าความต้านทานแม่เหล็กน้อยที่สุด ด้วยการหมุนในทิศทางทวนเข็มนาฬิกา ซึ่งแรงบิดที่ใช้หมุนเกิดจากแรงของเส้นแรงแม่เหล็ก แล้วจะไปหยุดที่ตำแหน่งความต้านทานแม่เหล็กน้อยที่สุด นั่นคือ จะหมุนไป 1 สเต็ป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



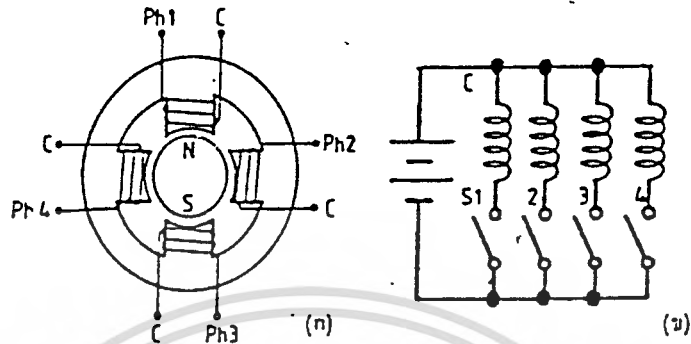
รูปที่ 2.57 แสดงเส้นแรงแม่เหล็กขณะกระตุ้นเฟส 1



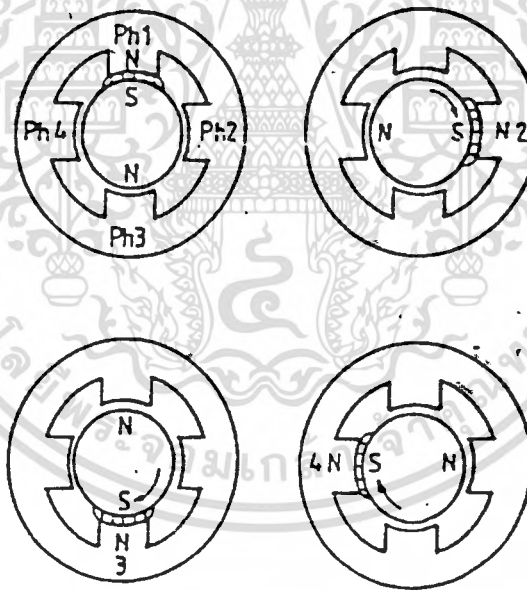
รูปที่ 2.58 แสดงขั้นตอนการกระตุ้นเฟสจาก เฟส 1 ไปยังเฟส 2

Permanent Magnet หรือเรียกสั้นๆว่า PM มอเตอร์ จะมีข้อแตกต่างที่สำคัญจาก VR มอเตอร์ ก็คือ โรเตอร์จะเป็นแม่เหล็กถาวร จึงทำให้การพันขดลวดจึงต้องแตกต่างกันไปด้วย ดังแสดงใน รูปที่ 2.59(ก) จะเห็นว่าสเตเตอร์ในแต่ละขั้วจะมีขดลวดพันอยู่ ซึ่งถือว่าแต่ละขั้วคือหนึ่งเฟส ดังนั้นจากรูปจึงมีทั้งหมด 4 เฟสด้วยกัน สำหรับการต่อวงจรการกระตุ้นเฟสมอเตอร์อย่างง่าย ดังแสดงไว้ในรูป 2.59(ข) จะเห็นว่าปลายขดลวด (C) ของทุกเฟสจะต้องร่วมกันเข้ากับแหล่งจ่ายไฟ ดังนั้นเมื่อเกิดการกระตุ้นที่เฟสใดแล้วขั้วสเตเตอร์ที่เฟสนั้นก็จะกลายเป็นขั้วเหนือ รูปที่ 2.60 จะเป็นการแสดงตำแหน่งของโรเตอร์ในแต่ละสเต็ป หลังจากถูกกระตุ้นที่เฟส 1-2-3-4 ตามลำดับ และจะหมุนไปในทิศทางตามเข็มนาฬิกาทุก 90 องศาต่อสเต็ป ถ้าต้องการจะให้มุมองศาต่อสเต็ปมีค่าลดลงหรือมีความละเอียดในตำแหน่งมากขึ้นจะต้องเพิ่มจำนวนเฟสของสเตเตอร์และจำนวนขั้วแม่เหล็ก ของโรเตอร์ให้มากขึ้น ข้อเสียของ PM มอเตอร์ คือ มีราคาแพง เอกสารนี้เป็นเอกสารสงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่นิยมนำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

และความหนาแน่นของเส้นแรงแม่เหล็กจะถูกจำกัดโดยเส้นแรงแม่เหล็กภายใน(Magnetic Remanence) ของแม่เหล็กถาวร ทำให้ไม่สามารถผลิตแรงบิดให้มากได้



รูปที่ 2.59 (ก) ภาพหน้าตัดของ PM สเต็ปป์มอเตอร์ แบบ 4 เฟส  
(ข) วงจรกระตุ้นเฟสพื้นฐาน สำหรับ PM มอเตอร์ 4 เฟส



รูปที่ 2.60 ลำดับขั้นตอนการหมุนในมอเตอร์ 4 เฟส

จากสเต็ปป์มอเตอร์มีขนาด 4 เฟส นี้เราจะบังคับให้เกิดสเต็ปการหมุนได้เป็น 3 ลักษณะขึ้นอยู่กับ การป้อนพัลส์ ดังนี้

1. One - excitation เป็นการจ่ายกระแสให้กับสเตเตอร์ครั้งละ 1 เฟส

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

|   | $\phi_1$ | $\phi_2$ | $\phi_3$ | $\phi_4$ |
|---|----------|----------|----------|----------|
| 1 | 1        | 0        | 0        | 0        |
| 2 | 0        | 1        | 0        | 0        |
| 3 | 0        | 0        | 1        | 0        |
| 4 | 0        | 0        | 0        | 1        |

ตารางที่ 2.61

2. Two - excitation หรือ full step เป็นการจ่ายกระแสให้กับสเตเตอร์ครั้งละ 2 เฟสพร้อมกัน ได้แก่  $\phi_1\phi_2$ ,  $\phi_2\phi_3$ ,  $\phi_3\phi_4$ ,  $\phi_4\phi_1$  หมุนเวียนกันไปแบบนี้แรงบิดที่ได้จะมากกว่าแบบแรม  
ดังตารางที่ 2.62

|   | $\phi_1$ | $\phi_2$ | $\phi_3$ | $\phi_4$ |
|---|----------|----------|----------|----------|
| 1 | 1        | 1        | 0        | 0        |
| 2 | 0        | 1        | 1        | 0        |
| 3 | 0        | 0        | 1        | 1        |
| 4 | 1        | 0        | 0        | 1        |

ตารางที่ 2.62

3. One-two excitation หรือ half step เป็นการจ่ายกระแสให้กับขดลวดสเตเตอร์ 1 เฟส สลับกันไปแบบนี้ ทำให้จำนวนสเต็ปเพิ่มขึ้นเป็น 2 เท่า ของ 2 แบบแรก แต่แรงบิดจะน้อยกว่าแบบที่ 2 เล็กน้อย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

|   | $\emptyset_1$ | $\emptyset_2$ | $\emptyset_3$ | $\emptyset_4$ |
|---|---------------|---------------|---------------|---------------|
| 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             |

ตารางที่ 2.63

ในการควบคุมสเคปการทำงานของสเคปโป๊วทั้งสามรูปแบบ ที่ทราบมานั้นเราสามารถทำได้เป็น สองลักษณะคือ ใช้วงจรซีคอนลอจิกและการใช้ไมโครโปรเซสเซอร์ควบคุมแต่ที่นิยมใช้กัน คือจะใช้ไมโครโปรเซสเซอร์ได้ถูกลงมากในปัจจุบันและ ผู้ใช้สามารถพลิกแพลงการใช้ได้ตามต้องการ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## 2.6 ออปแอมป์ (OP AMP)

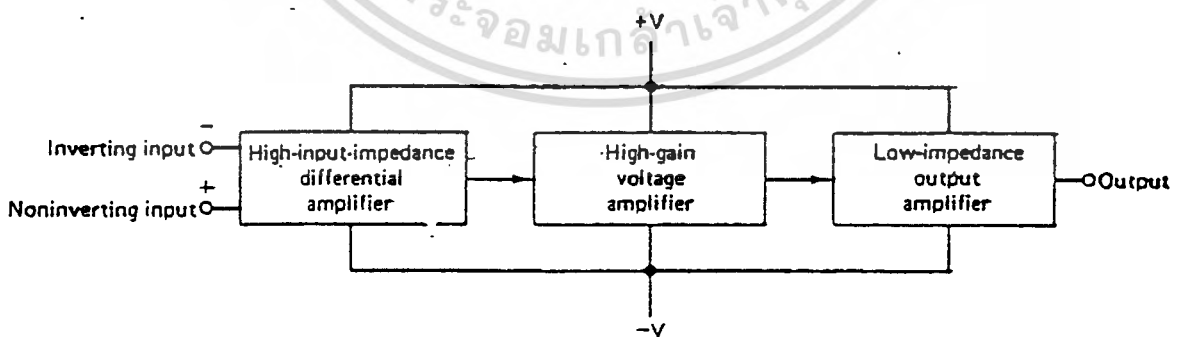
ออปแอมป์เป็นอุปกรณ์ชนิดหนึ่งซึ่งถูกออกแบบมาให้สามารถทำงานได้หลายรูปแบบและยังเน้นความสะดวกในการนำไปใช้งานอีกด้วย โดยสามารถประกอบเป็นวงจรได้โดยการต่อร่วมกับอุปกรณ์ภายนอกเพียงไม่กี่ตัวเท่านั้น ในอดีตออปแอมป์จะประกอบขึ้นจากอุปกรณ์หลายตัวที่ถูกนำมาบรรจุไว้ในภาชนะขึ้นเดียวกันซึ่งนอกจากจะทำให้ออปแอมป์มีขนาดใหญ่แล้วยังอาจมีประสิทธิภาพค่อนข้างต่ำด้วย แต่ปัจจุบัน เราสามารถซื้อออปแอมป์ในรูปแบบของวงจรรวม (IC) ได้ตามท้องตลาด และจากคุณสมบัติ IC ออปแอมป์ที่ได้พัฒนาขึ้นทำให้อุปกรณ์ชนิดนี้เป็นที่รู้จักกันอย่างแพร่หลาย

### คุณสมบัติทั่วไป

โดยทั่วไปแล้ว เราสามารถกล่าวได้ว่า ไอซีออปแอมป์ คือ อุปกรณ์โซลิดสเตท (SOLID STATE) ชนิดหนึ่ง ซึ่งสามารถตรวจวัดระดับสัญญาณไฟตรง และ ไฟสลับได้ และยังสามารถนำไปใช้ขยายสัญญาณได้อีกด้วย ไอซีออปแอมป์พื้นฐานจะต้องประกอบด้วยวงจรภายในภาคต่างๆ ดังนี้ คือ:

1. คิฟเฟอเร็นเชียลแอมป์ หรือ วงจรขยายผลต่าง (DIFFERENTIAL AMPLIFIER) ที่มีอินพุตอิมพีแดนซ์สูงมาก
2. วงจรขยายแรงดันซึ่งมีอัตราขยายสูงมาก
3. วงจรขยายภาคเอาต์พุตที่มี เอาต์พุตอิมพีแดนซ์ต่ำมาก

รูป 2.64 แสดงบล็อกไดอะแกรมของวงจรภายในภาคต่างๆ ของออปแอมป์ดังกล่าว และจากรูปจะสังเกตเห็นว่า แรงดันไฟตรงที่จ่ายให้แก่ออปแอมป์มักประกอบด้วยไฟบวกและลบ เพื่อให้เอาต์พุตสามารถสวิงได้ทั้งซิกบวกและลบเทียบกับกราวด์



รูปที่ 2.64

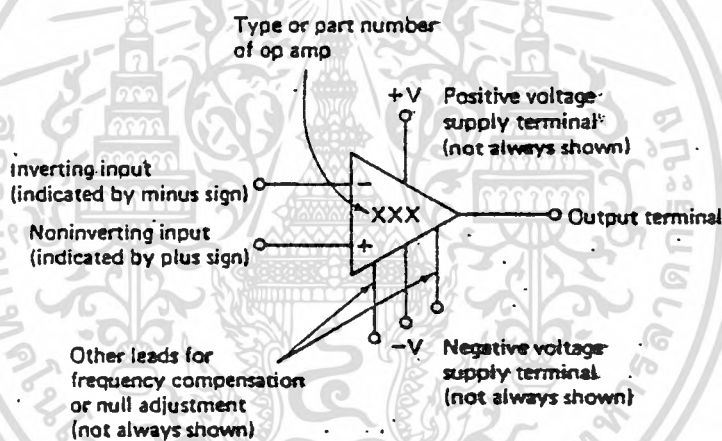
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากคุณสมบัติของออปแอมป์ที่กล่าวมาทำให้เราสามารถสรุปคุณสมบัติที่สำคัญบางประการของออปแอมป์ในอุดมคติ ดังนี้

1. เนื่องจากอินพุตอิมพีแดนซ์ของออปแอมป์มีค่าสูงเป็นอนันต์ กระแสเข้าที่อินพุตจะต่ำจนเกือบเท่ากับศูนย์ หรืออีกนัยหนึ่ง ไม่มีกระแสอินพุตเข้าสู่ออปแอมป์เลย
2. อัตราขยายขณะเปิดรูป A (ขณะที่ยังไม่มีกระแสย้อนกลับ) จะมีค่าสูงมาก ซึ่งหมายความว่า แรงดันระหว่างขั้วอินพุตควรมีค่าใกล้เคียงศูนย์ (เนื่องจาก  $V_{out} / A = V_{in}$ )
3. เอาต์พุตอิมพีแดนซ์มีค่าต่ำมากจนไม่ทำตัวเป็นโหลดต่อภาคเอาต์พุตของวงจรขยาย

จรรยา

รูป 2.65 แสดงสัญลักษณ์ทั่วไปของออปแอมป์ ซึ่งประกอบด้วยขั้วอินพุต 2 ขั้ว ขั้วสำหรับแหล่งจ่ายไฟ 2 ขั้ว ขั้วเอาต์พุต 1 ขั้ว และขั้วสำหรับปรับออฟเซต หรือการชดเชยความถี่ อีก 2 ขั้ว



รูปที่ 2.65

ขั้วอินพุตทั้งสองของออปแอมป์มีลักษณะต่างกันดังนี้ คือ สำหรับขั้วลบ เมื่อป้อนไฟตรงหรือไฟสลับเข้าไป ในขณะที่ขั้วบวกต่อกับจุดอ้างอิงจุดหนึ่ง สัญญาณที่ออกมาที่เอาต์พุตจะกลับเฟสกับอินพุต  $180^\circ$  ส่วนการป้อนสัญญาณที่ขั้วบวก เอาต์พุตจะมีเฟสตรงกับอินพุต ดังนั้นจึงกล่าวได้ว่าเครื่องหมายที่อินพุต คือ การแสดงเฟสของเอาต์พุตเทียบกับอินพุต ส่วนขั้วสำหรับปรับออฟเซต หรือชดเชยความถี่นั้นโดยมากมักจะ ไม่ถูกแสดง ในวงจรทั่วไป

## คุณสมบัติและพารามิเตอร์บางชนิดของออปแอมป์

### 1. อินพุตอิมพีแดนซ์

ในทางอุดมคติควรมีค่าเท่ากับอนันต์ แต่ในความเป็นจริง อินพุตอิมพีแดนซ์จะมีค่าประมาณ 1 เมกกะโอห์ม ( $1 \cdot 10^6$  โอห์ม) ค่าอินพุตอิมพีแดนซ์นี้ยังมีค่ามากขึ้นเท่าใด ออปแอมป์ตัวนั้นก็ทำงานได้ดียิ่งขึ้น นอกจากนี้ เมื่อนำออปแอมป์ไปใช้งานในย่านความถี่สูง ควรระวังผลจากอินพุตคาปาซิแตนซ์ของวงจรด้วย ซึ่งมักมีค่าประมาณ 2 พิโกฟารัด ( $2 \cdot 10^{-12}$  ฟารัด) เมื่อขั้วอินพุตขั้วหนึ่งต่อกับกราวด์

### 2. เอาท์พุตอิมพีแดนซ์

ดังที่ได้กล่าวมาแล้วว่า ออปแอมป์ในอุดมคติจะมีเอาท์พุตอิมพีแดนซ์เป็นศูนย์ แต่ในความเป็นจริง ค่านี้อาจมีได้ตั้งแต่ 25 ถึงหลายพันโอห์มขึ้นไป แต่อย่างไรก็ตาม เรามักสมมติให้เอาท์พุตอิมพีแดนซ์ในวงจรมีค่าเป็นศูนย์ เพื่อง่ายต่อการคำนวณ และวิเคราะห์

### 3. กระแสไบอัสด้านอินพุต

เนื่องจากอินพุตอิมพีแดนซ์ของออปแอมป์ไม่เป็นอนันต์ ดังนั้น จึงมีกระแสค่าน้อยๆ (มีหน่วยเป็น นาโน ( $10^{-9}$ ) ถึง ไมโคร ( $10^{-6}$ ) แอมแปร์) ไหลผ่านขั้วอินพุตทั้งสอง ซึ่งค่าเฉลี่ยของกระแสดังกล่าวถูกเรียกว่าเป็น 'กระแสไบอัสด้านอินพุต' กระแสจะก่อให้เกิดความไม่สมดุลในวงจรภายใน ซึ่งจะเป็ผลกระทบบต่อภาคเอาท์พุตด้วย ดังนั้น กระแสนี้ควรถูกกำจัดให้มีค่าต่ำสุด (อาจทำได้โดยการใช้ออปแอมป์ที่มีอินพุตเป็น FET)

### 4. แรงดันออฟเซต (OFFSET) ที่เอาท์พุต

แรงดันออฟเซตที่เอาท์พุตเกิดขึ้นจากกระแสไบอัสด้านอินพุต ซึ่งในทางอุดมคติ เมื่อแรงดันอินพุตระหว่างขั้วทั้งสองมีค่าเท่ากัน แรงดันที่เอาท์พุตควรเป็นศูนย์ แต่โดยทั่วไปมักไม่เป็นเช่นนั้น คือ มักมีแรงค่านึงปรากฏที่เอาท์พุตขณะที่อินพุตเป็นศูนย์ ซึ่งเราสามารถแก้ไขได้โดย การป้อนแรงดัน หรือ กระแสออฟเซตที่อินพุต แล้วปรับจนได้  $V_{out} = 0$  โวลต์

### 5. กระแสออฟเซตที่อินพุต

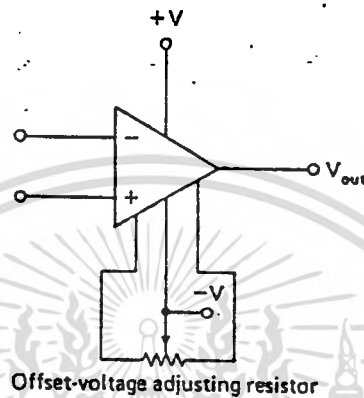
ในการปรับแรงดันออฟเซตที่เอาท์พุตให้มีค่าเป็นศูนย์ กระแสอินพุตทั้งสองขั้วควรมีค่าเท่ากัน แต่ในทางปฏิบัติ จะพบว่า เราต้องจ่ายกระแสให้แก่อินพุตขั้วหนึ่งมากกว่าอีกขั้วหนึ่งเสมอ เพื่อให้แรงดันเอาท์พุตมีค่าเป็นศูนย์ ซึ่งกระแสออฟเซตนี้อาจมีค่าประมาณ 20 มิลลิแอมป์

### 6. แรงดันออฟเซตที่อินพุต

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ในอุดมคติ แรงดันเอาต์พุตจะเท่ากับศูนย์ก็ต่อเมื่อแรงดันระหว่างขั้วอินพุตมีค่าเป็นศูนย์เช่นกัน แต่ในทางปฏิบัติ ความไม่สมดุลภายในวงจรมักทำให้เราต้องป้อนแรงดันค่าหนึ่งแก่อินพุตใดๆเสมอ เพื่อให้แรงดันเอาต์พุตเป็นศูนย์

## 7. การปรับออฟเซตให้เป็นศูนย์ (OFFSET NULLING)



รูปที่ 2.66

ในการปรับแรงดันเอาต์พุตให้เป็นศูนย์ หรือ การปรับแรงดันออฟเซตที่อินพุตให้เป็นศูนย์นั้น เราอาจใช้ขั้ว OFFSET NULLING ที่ผู้ผลิตได้เตรียมไว้ให้แล้ว โดยให้ศึกษาข้อมูลรายละเอียดของออปแอมป์เบอร์นั้นๆ รูป 2.66 แสดงวงจรที่ใช้ในการตั้งค่าศูนย์ให้แก่ออปแอมป์โดยมีขั้นตอนดังนี้

1. ตรวจสอบวงจรให้เรียบร้อยและเช็คอุปกรณ์ที่ต้องนำมาต่อ
2. ลดระดับสัญญาณอินพุตจนเหลือศูนย์ ถ้ามีตัวต้านทานต่ออนุกรมกับอินพุตอยู่ ให้ตรวจสอบดังนี้

ก) ถ้าตัวต้านทานนั้นมีค่าสูงกว่าอิมพีแดนซ์ของแหล่งกำเนิดสัญญาณอินพุตตั้งแต่ 1% ขึ้นไป ปลดตัวต้านทานไว้เช่นเดิม

ข) หากตัวต้านทานดังกล่าวมีค่าน้อยกว่าหรือเท่ากัน ให้นำแหล่งกำเนิดสัญญาณอินพุตนั้นออกแล้วต่อตัวต้านทานที่มีขนาดเท่ากับอินพุตอิมพีแดนซ์ของแหล่งกำเนิดอินพุตเข้าแทน

ค) ต่อโหนดเข้ายังขั้วเอาต์พุต

ง) ป้อนไฟตรงให้วงจร

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- จ) วัดแรงดันที่เอาท์พุทด้วย โวลท์มิเตอร์ หรือ ออสซิลโลสโคป (แรงดันเอาท์พุทอาจมีค่าต่ำเพียงไม่กี่มิลลิโวลต์ ดังนั้น ควรใช้อุปกรณ์สำหรับวัดที่เหมาะสม)
- ณ) ปรับตัวต้านทานชนิดปรับค่าได้จน  $V_{out}$  มีค่าเป็นศูนย์
- ข) ถอดอุปกรณ์ที่เพิ่มเข้าไป แล้วต่อวงจรดังเดิม แต่ห้ามแตะต้องตัวต้านทานปรับค่าได้

### 8. ผลของอุณหภูมิ

อุณหภูมิมีผลต่ออุปกรณ์โซลิตสเตททุกชนิด รวมทั้งออปแอมป์ ดังนั้น ผลกระทบจากอุณหภูมิจะทำให้กระแสและแรงดันออฟเซตเปลี่ยนแปลงไป ซึ่งเราเรียกการเปลี่ยนแปลงอันเกิดขึ้นจากอุณหภูมิว่า 'ดริฟท์' ดังนั้น ในขณะที่ทำงาน ควรตรวจสอบด้วยว่าออปแอมป์จะมีเปอร์เซ็นต์การผิดพลาดมากน้อยเพียงใด หากอุณหภูมิเปลี่ยนไป

### 9. การรบกวนความถี่

ปัญหาที่เกิดขึ้นกับออปแอมป์ที่ใช้ในย่านความถี่สูงก็คือ การออสซิลเลท ซึ่งเกิดจากอัตราขยายที่มีค่อนข้างสูงของออปแอมป์เอง และยังเกิดการเลื่อนเฟส ณ จุดต่างๆ ภายในวงจร เป็นผลให้เราไม่สามารถควบคุมอัตราขยายของสัญญาณป้อนกลับได้ วิธีที่แก้ปัญหาก็คือต่อตัวเก็บประจุให้แก่วงจร ซึ่งจะทำให้อัตราการขยายลดลงเมื่อความถี่สูงขึ้น วิธีที่แก้ปัญหาก็คือต่อตัวเก็บประจุให้แก่วงจร ซึ่งจะทำให้อัตราการขยายลดลงเมื่อความถี่สูงขึ้น

### 10. อัตราสลับ (SLEW RATE)

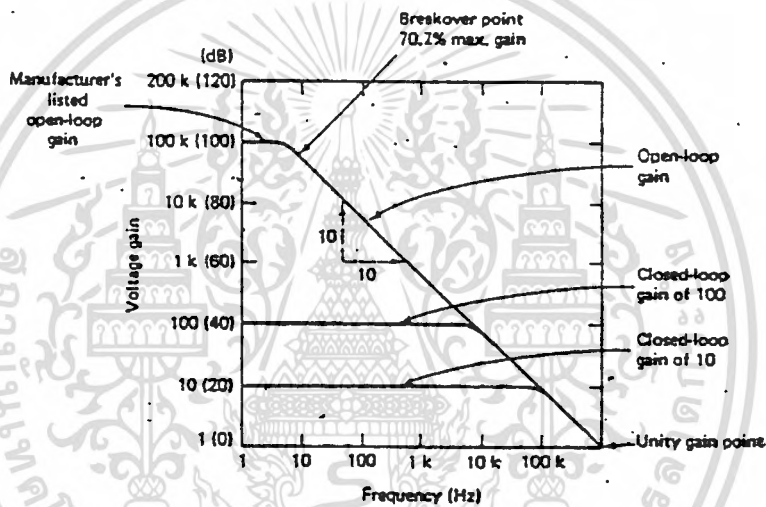
อัตราสลับ คือ อัตราการเปลี่ยนแปลงสูงสุดของแรงดันเอาท์พุทเทียบกับเวลา ดังสมการ

$$\begin{aligned}\text{อัตราสลับ} &= \frac{\text{การเปลี่ยนแปลงสูงสุดของแรงดันเอาท์พุท}}{\text{การเปลี่ยนแปลงเวลา}} \\ &= \frac{\Delta V_{out} (\text{max})}{\Delta t}\end{aligned}$$

### 11. การตอบสนองต่อความถี่

อัตราขยายของออปแอมป์จะลดลงเมื่อความถี่สูงขึ้น ดังแสดงในรูป 2.67 จะพบว่า อัตราขยายที่ผู้ผลิตแสดงไว้ในคาต้าชีท (DATE SHEET) จะเป็นอัตราขยายที่ความถี่ 0 เฮิรตซ์ หรือ ไฟตรง (ซึ่งแสดงให้เห็นความถี่สูงสุดของออปแอมป์) ในการทำงานแบบลูปเปิดไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จะเห็นได้ว่า วงจรมีเสถียรภาพต่ำมาก นั่นคือ เมื่อความถี่เพิ่มขึ้นสิบเท่า อัตราขยายจะลดลง 10 เท่าทันทีโดยทั่วไป แบนด์วิดธ์ของวงจรจะหมายถึงช่วงความถี่ที่อัตราขยายมีค่า 70.7% ของอัตราขยายสูงสุด จากตัวอย่างในรูป แบนด์วิดธ์ของออปแอมป์แบบเปิดรูป จะมีค่าประมาณ 10 เฮิรตซ์ ดังนั้น หากต้องการให้วงจรที่ออกแบบขึ้นสามารถทำงานได้ที่ความถี่อินพุตสูงๆ จะต้องป้อนสัญญาณจากเอาต์พุตกลับมายังอินพุต (การป้อนกลับแบบลบ) ซึ่งจะทำให้อัตราขยายของรูป ปกติลดลง แต่ในขณะเดียวกัน ความถี่ที่อัตราขยายเท่ากับ 70.7% ของอัตราขยายสูงสุดก็จะเพิ่มขึ้น ด้วยอัตราเท่ากัน ส่วนจุดที่มีอัตราขยายเท่ากับหนึ่ง (UNITY-GAIN POINT) จะแสดงความถี่สูงสุดของออปแอมป์เมื่ออัตราขยายมีค่าเป็นหนึ่ง



รูปที่ 2.67

## 12. ผลคูณของอัตราขยายและแบนด์วิดธ์

จากที่ได้อธิบายมาในหัวข้อที่แล้วว่า เมื่ออัตราขยายลดลงเท่าใด ความถี่ก็จะสูงขึ้นด้วยจำนวนเท่าของค่านั้นด้วย ซึ่งแสดงว่า ผลคูณระหว่างอัตราขยายและแบนด์วิดธ์จะมีค่าคงที่เสมอไป และ เราสามารถหาค่านั้นได้จากจุดซึ่งมีอัตราขยายเท่ากับหนึ่ง (ค่านี้จะหาได้จากคาตาคีพ) ผลคูณนี้มีประโยชน์อย่างมากในการประมาณความถี่สูงสุดที่วงจรสามารถทำงานได้ ตัวอย่างเช่น หากเราพบว่า ความถี่ที่อัตราขยายมีค่าเท่ากับหนึ่ง (หาจาก DATA SHEET) มีค่าเป็น 1 เมกกะเฮิรตซ์ (ดังแสดงในรูป 2.67) และอัตราขยายของวงจรที่ออกแบบมีค่าเท่ากับ 100 ดังนั้น ความถี่สูงสุดที่เราประมาณไว้ควรมีค่าอยู่ในราว

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์ (แบนด์วิดธ์)  $= (1,000,000 / 100) = 10 \text{ kHz}$  ดึงนำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 13. อัตราการลดสัญญาณชนิดคอมมอนโหมด (CMRR)

COMMON-MODE REJECTION RATIO เป็นคุณสมบัติอย่างหนึ่งของออปแอมป์ที่ได้มาจากภาคอินพุตที่เป็นดิฟเฟอเรนเชียลแอมป์ หมายความว่า หากสัญญาณที่เข้ามายังขั้วอินพุตทั้งสองนี้มีเฟสตรงกัน และยังมีขนาดเท่ากัน (สัญญาณแบบนี้เรียกว่าอยู่ในคอมมอนโหมด) สัญญาณทั้งสองนี้จะถูกกำจัดทิ้งไป ดังนั้น เอาต์พุตจะเป็นศูนย์ ส่วนสัญญาณที่มีขนาดต่างกันและมีเฟสต่างกันจะเรียกว่าอยู่ในดิฟเฟอเรนเชียลโหมด

$$\text{CMRR} = A_D / A_{\text{cm}}$$

โดยที่  $A_D$  คืออัตราขยายสำหรับสัญญาณแบบดิฟเฟอเรนเชียล ส่วน  $A_{\text{cm}}$  คืออัตราขยายสำหรับสัญญาณคอมมอนโหมด

### 14. หน่วยเดซิเบล (dB)

การคำนวณอัตราขยายของวงจรใดๆสามารถหาได้จากสมการ

$$A_x = X_{\text{out}} / X_{\text{in}}$$

โดยที่ X อาจเป็นแรงดัน หรือ กระแสก็ได้ และโดยทั่วไปนิยามหาอัตราขยายจากสมการ

$$\text{dB} = 20 \log A_v$$

### ค่าใช้งานต่างๆที่ควรเข้าใจ

การนำไอซีออปแอมป์ไปใช้งาน ผู้ใช้ควรศึกษาค่าชี้ของบริษัทผู้ผลิตให้ละเอียดเสียก่อนเพื่อป้องกันออปแอมป์ชำรุด เสียหาย ค่าต่างๆที่ควรรู้ได้แก่

ก) แรงดันจากแหล่งจ่ายไฟเลี้ยง  $\pm V$  (SUPPLY VOLTAGE) .

ข) กำลังที่สูญเสียในตัวไอซี (POWER DISSIPATION) ซึ่งแสดงกำลังสูงสุดที่อุปกรณ์สามารถปลดปล่อยออกมาโดยที่อุณหภูมิไม่สูง จนทำให้ไอซีพัง

ค) แรงดันสูงสุดระหว่างขั้วอินพุต (DIFFERENTIAL INPUT VOLTAGE)

ง) แรงดันสูงสุดที่อินพุตแต่ละขั้ว (INPUT VOLTAGE)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จ) ความสามารถในการทนภาวะลัดวงจรที่เอาต์พุต (OUTPUT SHORT-CIRCUIT DURATION)

ด) อุณหภูมิที่ใช้งาน (OPERATING TEMPERATURE)

ข) อุณหภูมิที่ใช้เก็บอุปกรณ์ (STORAGE-TEMPERATURE)

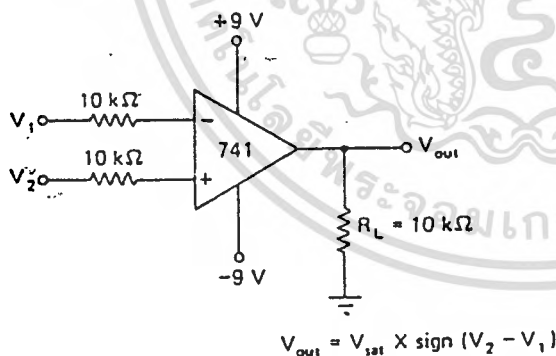
ณ) อุณหภูมิของขาอุปกรณ์ (LEAD TEMPERATURE) แสดงอุณหภูมิที่ไอซีสามารถทนได้ในขณะที่ถูกบัดกรี

### วงจรออปแอมป์พื้นฐาน

ในหัวข้อนี้จะกล่าวถึงวงจรพื้นฐานของการนำออปแอมป์ไปใช้งานซึ่งวงจรเหล่านี้จะเป็นพื้นฐานที่ดีในการนำไปประกอบวงจรที่สลับซับซ้อนมากขึ้นไปโดยจะแบ่งออกเป็นชนิดการทำงานแบบรูปเปิดเช่น คอมพาราเตอร์ ตัวตรวจวัดระดับสัญญาณ เป็นต้น และแบบรูปปิดเช่น วงจรขยายแรงดัน เป็นต้น

#### 1. วงจรเปรียบเทียบแรงดัน(คอมพาราเตอร์)

การทำงานเป็นคอมพาราเตอร์ในขณะเปิดดูป็น ออปแอมป์จะสามารถเปรียบเทียบระดับสัญญาณระหว่างขั้วอินพุตทั้งสองได้ค่อนข้างแม่นยำ โดยใช้หลักเรื่องความสัมพันธ์ระหว่างขั้วอินพุตที่ได้ เมื่อแรงดันที่ขั้วลบมีค่าเป็นบวกสูงกว่าแรงดันอินพุตที่ขั้วบวกสัญญาณที่เอาต์พุตจะเป็นลบ และมีขนาดเกือบเท่ากับ  $-V_{sat}$  จากแหล่งจ่ายไฟดังในรูปที่ 2.68



| Input voltage |       | Output voltage |
|---------------|-------|----------------|
| $V_1$         | $V_2$ | $\pm V_{sat}$  |
| +1            | +2    | +8             |
| +2            | +1    | -8             |
| 0             | 0     | 0              |
| +1            | -1    | -8             |
| -1            | +1    | +8             |
| -1            | -2    | -8             |
| -2            | -1    | +8             |

รูปที่ 2.68

นอกจากนี้ เราสามารถใช้วงจรคอมพาราเตอร์สำหรับตรวจวัดระดับสัญญาณว่ามีค่าเท่ากับแรงดันอ้างอิงหรือยัง โดยใช้หลักการเช่นเดียวกันกับที่ได้ศึกษามา ต่างกันเพียงการป้อนแรงดันอ้างอิงเท่านั้นแทนที่จะป้อนศักดากราวด์เช่นเดิม โดยที่แรงดันอ้างอิง ( $V_{ref}$ ) จะมีค่า

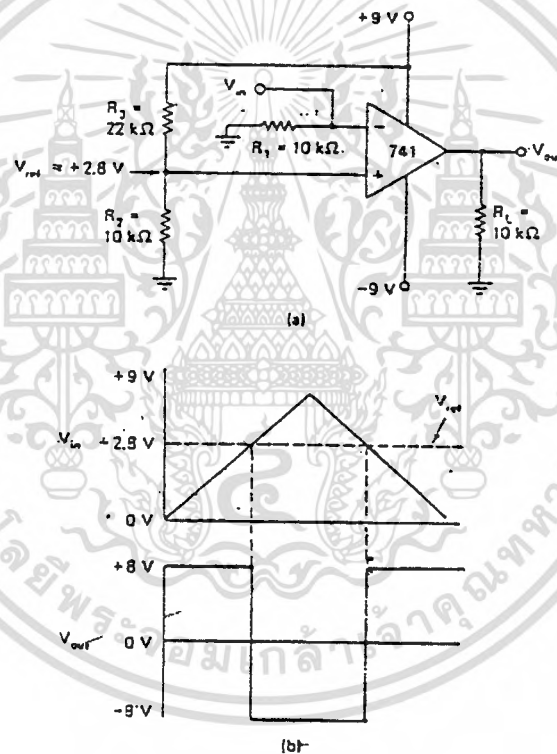
เอกสารนี้เป็นเอกสารของมหาวิทยาลัยเทคโนโลยีพระจอมเกล้าธนบุรี ไม่อนุญาตให้เผยแพร่หรือใช้โดยไม่ได้รับอนุญาตจากทางมหาวิทยาลัย หากฝ่าฝืนอาจมีความผิดตามกฎหมายว่าด้วยการคุ้มครองสิทธิบัตร

$$V_{ref} = R_2/(R_2+R_3) (+V)$$

กรณีในรูปที่ 2.69  $V_{ref}$  จะมีค่าเท่ากับ

$$\begin{aligned} V_{ref} &= (10k/(10k+22k))(+9V) \\ &= +2.8 \text{ V} \end{aligned}$$

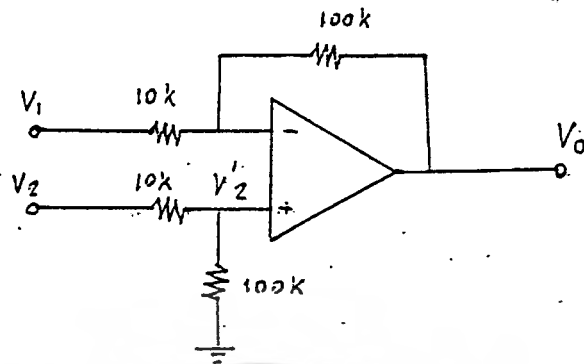
ในกรณีนี้ ป้อนแรงดันอ้างอิงที่ค่าบวก และป้อนสัญญาณที่ต้องการจะวัด ณ ขั้วอินพุตลบ ดังนั้น สัญญาณที่เอาต์พุตจึงมีลักษณะกลับเฟสกับอินพุต 180 องศา ส่วนรูปที่ 2.70 เป็นการตรวจระดับสัญญาณอ้างอิงที่เป็นลบค่าหนึ่ง



รูปที่ 2.69 Voltage Detector Circuit

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้





รูปที่ 2.72 Difference Amplifier

ใช้หลักการของ Superposition คือคิดว่า  $V$  ลักวงจรกราวด์ ดังนั้นแรงดันเอาต์พุตจะเท่ากับ

$$V_{o1} = \frac{-R_f}{R_1} (V_1)$$

และถ้า  $V_1$  ลักวงจรกราวด์ จะได้แรงดันเอาต์พุตเท่ากับ

$$V_{o2} = \left(1 + \frac{R_2}{R_1}\right) V_2$$

ดังนั้น

$$V_o = V_{o1} + V_{o2}$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## 2.7 DOT MATRIX LCD MODULE

อุปกรณ์ในปัจจุบันนี้ในส่วนแสดงผลนั้นจะใช้ LCD เสียเป็นส่วนใหญ่ไม่ว่าจะเป็น เครื่องเล่น VEDIO, เครื่องถ่ายภาพเอกสาร, เครื่องมือวัดคุมต่างๆ, เครื่องคอมพิวเตอร์ เราจะพอแบ่ง DOT MATRIX LCD MODULE นี้ออกได้เป็นพวกดังนี้:-

1. CHARECTER LCD MODULE

2. GRAPHIC LCD MODULE

3. SEGMENT DISPLAY TYPE LCD MODULE

โดยในแต่ละแบบนี้ก็จะมีส่วนประกอบใหญ่ๆแบ่งได้เป็น

1. DOT MATRIX LCD เป็นตัวแสดงผลให้เรามองเห็นในลักษณะการปิด และเปิดตัวเองกับแสงก็คือ ส่วนของที่เป็นตัวกระจกบรรจุผลึก

2. DRIVER เป็นตัวรับสัญญาณจากตัวควบคุมมาขับผลึก LCD อีกทีหนึ่งโดยมี เบอร์ที่นิยมใช้ใน LCD MODULEG เช่น HD44100H , MSM5259

3. CONTROLLER เป็นตัวรับข้อมูลจากอุปกรณ์ภายนอกมาและจัดการควบคุม LCD MODULE ให้ทำงานแสดงผลต่างๆ เช่น การลบจอภาพ,การเกิดตัวอักษร, เป็นต้น โดยมี เบอร์ IC ที่นิยมใช้กันคือ HD4478 ซึ่งจะใช้ในแบบ CHARECTER LCD MODULE เป็นส่วนใหญ่และ เบอร์ IC HD61830 จะใช้แบบ GRAPHIC LCD MODULE

ในการศึกษาการทำงานและใช้งาน LCD MODULE นั้นไม่ใช่เรื่องยากเลยถ้าเราสามารถทำความเข้าใจในส่วนของ CONTROLLER ได้ก็เพียงพอแล้วและโดยมาก LCD MODULE ในแต่ละบริษัทแล้วจะใช้ตัวอักษรหรือจำนวนบรรทัดก็มีหลักการทำงานแบบเดียวกัน ทั้งหมด IC ที่นิยมกันมากที่สุดตัวหนึ่งที่เป็น CONTROLLER LCD ก็คือเบอร์ HD44780 โดยรูปแบบการทำงานของมันได้เป็นมาตรฐานให้กับ CONTROLLER LCD ตัวอื่นๆด้วย

HD44780 เป็นไอซี LSI ตัวหนึ่งใช้ควบคุม LCD โดยแสดงผลในรูปแบบตัวอักษรหรือสัญลักษณ์ต่างๆตัวมันเองสามารถต่อใช้งานแบบ 4BIT หรือ 8BIT โดยถ้าเราต่อแบบ 4BIT จะต่อใช้งานที่ DB7-DB4 เท่านั้นโดยข้อมูลครั้งแรกที่ส่งนั้น HD44780 จะถือเป็นข้อมูล 4 BIT บน และข้อมูลที่ส่งต่อมานั้นเป็นข้อมูล 4 BIT ล่าง

รายละเอียดของคำสั่ง HD44780

1. CLEAR DISPLAY

Clear display

RS      R/W      DB7 ..... DB0

Code

|   |   |   |   |   |   |   |   |   |   |
|---|---|---|---|---|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |
|---|---|---|---|---|---|---|---|---|---|

คำสั่งนี้จะเป็นการเขียนช่องว่างหรือ SPACE (ASCII 20H) เข้าไปใน DD RAM ทั้งหมดและทำการ SET DD RAM ADDRESSER เป็นศูนย์ ตัว CURSOR จะกลับไปอยู่ตำแหน่งบนสุดซ้ายมือของจอภาพ SET I/D = 1,S ไม่มีการเปลี่ยน

2. RETURN HOME

Return Home

RS      R/W      DB7 .....DB0

Code

|   |   |   |   |   |   |   |   |   |   |
|---|---|---|---|---|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | * |
|---|---|---|---|---|---|---|---|---|---|

\* NO effect

คำสั่งนี้จะทำการ SET DD RAM ADDRESSER เป็นศูนย์ตัว CURSOR จะกลับไปอยู่ตำแหน่งบนสุดซ้ายมือของจอภาพข้อมูลในจอภาพจะไม่เปลี่ยน

3. ENTRY MODE SET

Entry mode set

RS      R/W      DB7 .....DB0

Code

|   |   |   |   |   |   |   |   |     |   |
|---|---|---|---|---|---|---|---|-----|---|
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | I/D | S |
|---|---|---|---|---|---|---|---|-----|---|

BIT I/D : โดยจะเป็นตัวกำหนดให้ว่าเมื่อเขียนหรืออ่านข้อมูลแล้วจะทำให้ DD RAM ADDRESSER เพิ่มขึ้นหนึ่งหรือลดลงหนึ่งโดย

1 = เพิ่ม

0 = ลดลงหนึ่ง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

BIT : เป็นตัวกำหนดแสดงผลโดยถ้า S=1 จะเป็นการใส่ข้อมูลแล้วตัว CURSOR อยู่กับที่ ข้อมูลจะถูกดันไปทางซ้าย ถ้า S=0 ข้อมูลจะอยู่กับที่ ตัว CURSOR จะถูกดันไปทางขวามือ

#### 4. DISPLAY ON/OFF CONTROL

RS R/W DB7 .....DB0

Code

|   |   |   |   |   |   |   |   |   |   |
|---|---|---|---|---|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 | 0 | 1 | D | C | B |
|---|---|---|---|---|---|---|---|---|---|

SET D : เป็น BIT ให้เปิดปิดหน้าจอภาพโดยถ้า

D = 1 จะ ON และ

D = 0 จะ OFF

BIT C : จะให้แสดง CURSOR ให้ BIT C=1 และถ้าไม่ต้องการแสดง CURSOR BIT C = 0 โดยตัว CURSOR จะอยู่กับที่ LINE ที่ 8 ในแบบ 5X7 DOT และจะอยู่ LINE ที่ 11 ในแบบ 5X10 DOT

BIT B : เป็นบิตเซ็ทกระพริบของ CURSOR โดย B=1 การกระพริบ B=0 ไม่มีการกระพริบโดย มีระยะเวลาการกระพริบประมาณ 379.2 ms

#### 5. CURSOR OR DISPLAY SHIFT

Cursor or display shift

RS R/W DB7 .....DB0

Code

|   |   |   |   |   |   |     |     |   |   |
|---|---|---|---|---|---|-----|-----|---|---|
| 0 | 0 | 0 | 0 | 0 | 1 | S/R | R/L | * | * |
|---|---|---|---|---|---|-----|-----|---|---|

\* NO effect

เป็นคำสั่งกำหนดให้ตำแหน่ง CURSOR หรือข้อมูลไปเกิดทางซ้ายหรือขวาโดยไม่ต้องใช้คำสั่งเขียนหรืออ่าน โดย

S/C R/L

- |   |   |                                                         |
|---|---|---------------------------------------------------------|
| 0 | 0 | ทำการย้าย CURSOR ไปจากตำแหน่งเดิมไปทางซ้ายมือ 1 ตำแหน่ง |
| 0 | 1 | ทำการย้าย CURSOR ไปจากตำแหน่งเดิมไปทางขวามือ 1 ตำแหน่ง  |
| 1 | 0 | เป็นการดันตัวอักษรที่เกิดไปทางซ้ายมือ                   |

เอกสารนี้เป็นเอกสารที่ 1 ในจำนวน 1 ฉบับ ใช้สำหรับการค้นตัวอักษรที่เกิดไปทางขวามือ อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## 6. FUNCTION SET

Function Set

RS      R/W      DB7 .....DB0

Code

|   |   |   |   |   |    |   |   |   |   |
|---|---|---|---|---|----|---|---|---|---|
| 0 | 0 | 0 | 0 | 1 | DL | N | F | * | * |
|---|---|---|---|---|----|---|---|---|---|

\* NO effect

BIT DL : เป็นการเซตการติดต่อกว่าจะให้เป็นแบบ 8 บิต หรือ 4 บิต โดยถ้าต้องการติดต่อก 4 BIT DL = 0 และ BIT DL = 1

N : เป็นการเซตบรรทัดการแสดงผล N=0 แสดง 1 บรรทัด

N : 1 แสดง 2 บรรทัด ในกรณีที่มากกว่า 2 บรรทัดก็ให้เซต N = 1

F : เป็นการเซตขนาด DOT การแสดงผล 5X7 หรือ 5X10 โดย

F = 0 เป็นแบบ 5X7 และ F =1 เป็นแบบ 5X10

| N F | No. of display line | Character font | Duty factor | Remarks                                               |
|-----|---------------------|----------------|-------------|-------------------------------------------------------|
| 0 0 | 1                   | 5 x 7 dots     | 1/8         |                                                       |
| 0 1 | 1                   | 5 x 10 dots    | 1/11        |                                                       |
| 1 * | 2                   | 5 x 7 dots     | 1/16        | Cannot display 2 lines with 5 x 10 dot character font |

\* NO effect

## 7. SET CG RAM ADDRESS

Set CG RAM address.

RS      R/W      DB7 .....DB0

Code

|   |   |   |   |   |   |   |   |   |   |
|---|---|---|---|---|---|---|---|---|---|
| 0 | 0 | 0 | 1 | A | A | A | A | A | A |
|---|---|---|---|---|---|---|---|---|---|

<--- Higher

Lower

--->

Order Bits

Order Bits

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ใน HD44780 นั้นจะมีหน่วยความจำอยู่ 2 ชุด คือ DISPLAY DATA RAM (DD RAM) จำนวน 80X8 BIT และ CHARECTER GENERATOR ROM CG RAM จำนวน 512 BIT และ 7200 BIT คำสั่งนี้จะเป็นการ SET ADDRESS ใน CG RAM โดยต้องทำการ SET ADDRESS ก่อนเขียนหรือ อ่านข้อมูลจาก CG RAM ด้วย

8. SET DD RAM ADDRESS

Set CG RAM address

RS     R/W     DB7 ..... DB0

Code

|             |   |   |           |   |   |   |            |   |       |
|-------------|---|---|-----------|---|---|---|------------|---|-------|
| 0           | 0 | 1 | A         | A | A | A | A          | A | A     |
| <... Higher |   |   | Order Bit |   |   |   | lower      |   | >...> |
|             |   |   |           |   |   |   | Order Bits |   |       |

เป็นคำสั่ง SET ค่า ADDRESS ใน DD RAM ในการเขียนหรืออ่านค่าจาก DD RAM (DD RAM คือ ส่วนที่จะแสดงผลหน้าจอ LCD) โดยจำนวน DDRESS ที่จะเกิดขึ้นบนจอ LCD จะอยู่กับ SET ค่า N ด้วย

N = 0 (1บรรทัด) ADDRESS จะอยู่ที่ 00H - 4FH

N = 1 (2บรรทัด) ADDRESS จะอยู่ที่ 00H - 27H สำหรับบรรทัดที่ 1 และ 40H - 67H สำหรับบรรทัดที่ 2

แบบการจัด ADDRESS ของ DD RAM หน้าจอ LCD แบบ 16 ตัวอักษร 1 บรรทัด,16 ตัวอักษร 2 บรรทัด,16 ตัวอักษร 4 บรรทัด , 20 ตัวอักษร 1บรรทัด,20 ตัวอักษร 2บรรทัด และ 40 ตัวอักษร 2บรรทัด

16 ตัวอักษร 1 บรรทัด

|    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |
|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|
| 00 | 01 | 02 | 03 | 04 | 05 | 06 | 07 | 40 | 41 | 42 | 43 | 44 | 45 | 46 | 47 |
|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|

16 ตัวอักษร 2 บรรทัด

|    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |
|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|
| 00 | 01 | 02 | 03 | 04 | 05 | 06 | 07 | 08 | 09 | 0A | 0B | 0C | 0D | 0E | 0F |
| 40 | 41 | 42 | 43 | 44 | 45 | 46 | 47 | 48 | 49 | 4A | 4B | 4C | 4D | 4E | 4F |

16 ตัวอักษร 4 บรรทัด

|    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |
|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|
| 00 | 01 | 02 | 03 | 04 | 05 | 06 | 07 | 08 | 09 | 0A | 0B | 0C | 0D | 0E | 0F |
|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|

|    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |
|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|
| 40 | 41 | 42 | 43 | 44 | 45 | 46 | 47 | 48 | 49 | 4A | 4B | 4C | 4D | 4E | 4F |
| 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 1A | 1B | 1C | 1D | 1E | 1F |
| 50 | 51 | 52 | 53 | 54 | 55 | 56 | 57 | 58 | 59 | 5A | 5B | 5C | 5D | 5E | 5F |

20 ตัวอักษร 1 บรรทัด

|    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |
|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|
| 00 | 01 | 02 | 03 | 04 | 05 | 06 | 07 | 08 | 09 | 0A | 0B | 0C | 0D | 0E | 0F | 10 | 11 | 12 | 13 |
|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|

20 ตัวอักษร 2 บรรทัด

|    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |
|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|
| 00 | 01 | 02 | 03 | 04 | 05 | 06 | 07 | 08 | 09 | 0A | 0B | 0C | 0D | 0E | 0F | 10 | 11 | 12 | 13 |
| 40 | 41 | 42 | 43 | 44 | 45 | 46 | 47 | 48 | 49 | 4A | 4B | 4C | 4D | 4E | 4F | 50 | 51 | 52 | 53 |

9. READ BUSY FLAG AND ADDRESS

Read busy flag & address

RS

R/W

DB7.....DB0

Code

|   |   |    |   |   |   |   |   |   |   |
|---|---|----|---|---|---|---|---|---|---|
| 0 | 1 | BF | A | A | A | A | A | A | A |
|---|---|----|---|---|---|---|---|---|---|

<--- Higher

Lower --->

Order Bits

Order Bits

เป็นคำสั่งอ่านค่า BUSY FLAG ซึ่งจะเป็นตัวบอกว่าตัว HD44780 นี้อยู่ในขบวนการทำงานภายในอยู่หรืออยู่ในสภาพพร้อมจะรับข้อมูล โดย

BF = 1 อยู่ในขบวนการทำงานภายในไม่พร้อมจะรับข้อมูลหรือคำสั่ง

BF = 0 พร้อมจะรับข้อมูลหรือคำสั่งได้

และนอกจากนี้ยังเป็นคำสั่งอ่านค่าข้อมูล ADDRESS ของ CG RAM หรือ DD RAM ด้วย

10. WRITE DATA TO CG RAM หรือ DD RAM

Write data to CG or DD RAM

RS    R/W   DB7 .....DB0

### Code

|   |   |   |   |   |   |   |   |   |   |
|---|---|---|---|---|---|---|---|---|---|
| 1 | 0 | D | D | D | D | D | D | D | D |
|---|---|---|---|---|---|---|---|---|---|

<... Higher

Lower

...

## Order Bits

## Order Bits

เป็นคำสั่งเขียนข้อมูลเข้าไปใน CG หรือ DD RAM โดยเมื่อเขียนข้อมูลและ ADDRESS จะเพิ่มหรือลดโดยอัตโนมัติตามคำสั่งที่ SET ใน ENTRY MODE ข้อกำหนดที่จะรู้ว่าเป็นการเขียนข้อมูลของ CG RAM หรือ DD RAM ทำโดยการ SET ADDRESS ของ CG RAM หรือ DD RAM ขึ้นมาก่อนจะเขียนข้อมูล

## 11. READ DATA FROM CG OR DD RAM

Read data CG or DD RAM

RS      R/W      DB7 ..... DB0

**Code**

|   |   |   |   |   |   |   |   |   |   |
|---|---|---|---|---|---|---|---|---|---|
| 1 | 1 | D | D | D | D | D | D | D | D |
|---|---|---|---|---|---|---|---|---|---|

<... Higher

Lower

...

### Order Bits

## Order Bits

เป็นคำสั่งอ่านข้อมูลจาก CG RAM หรือ DD RAM โดยก่อนอ่านค่าจาก DD RAM หรือ CG RAM นี้ควรจะใช้คำสั่ง SET ADDRESS ก่อนเพื่อให้รู้ว่าข้อมูลที่อ่านได้นั้นเป็น DD หรือ CG RAM จากตารางการทำงานจะเห็นว่าการใช้งาน LCD MODULE นั้นง่ายเพียงแต่เราส่งคำสั่งเริ่มแรกและ SET ความต้องการในขนาดตัวอักษร, CURSOR หลังจากนั้นเราก็สามารถเขียนตัวอักษรเข้าไปใน DD RAM ตามตารางตัวอักษรที่ให้มานั้นก็จะเกิดอักษรในจอภาพ LCD เรายังสามารถกำหนดตำแหน่งตัวอักษรที่จะให้เกิดบนจอได้โดยการ SET DD RAM ADDRESS ตามตารางที่ให้มาในหัวข้อ SET DD RAM ADDRESS ขอให้ทดสอบทำความเข้าใจกับตัวโปรแกรมที่ใช้กับ ET BOARD V 3.0, V3.5 นี้ที่ให้มาจะเห็นว่าจะมีส่วนเริ่มต้นก็คือ ส่วนการ INITIAL LCD เพื่อกำหนดหน้าที่การทำงานต่าง ๆ

## **บทที่ 3**

### **การออกแบบและกำหนดค่าอุปกรณ์ในแต่ละส่วน**

#### **3.1 แนวความคิดในการออกแบบ**

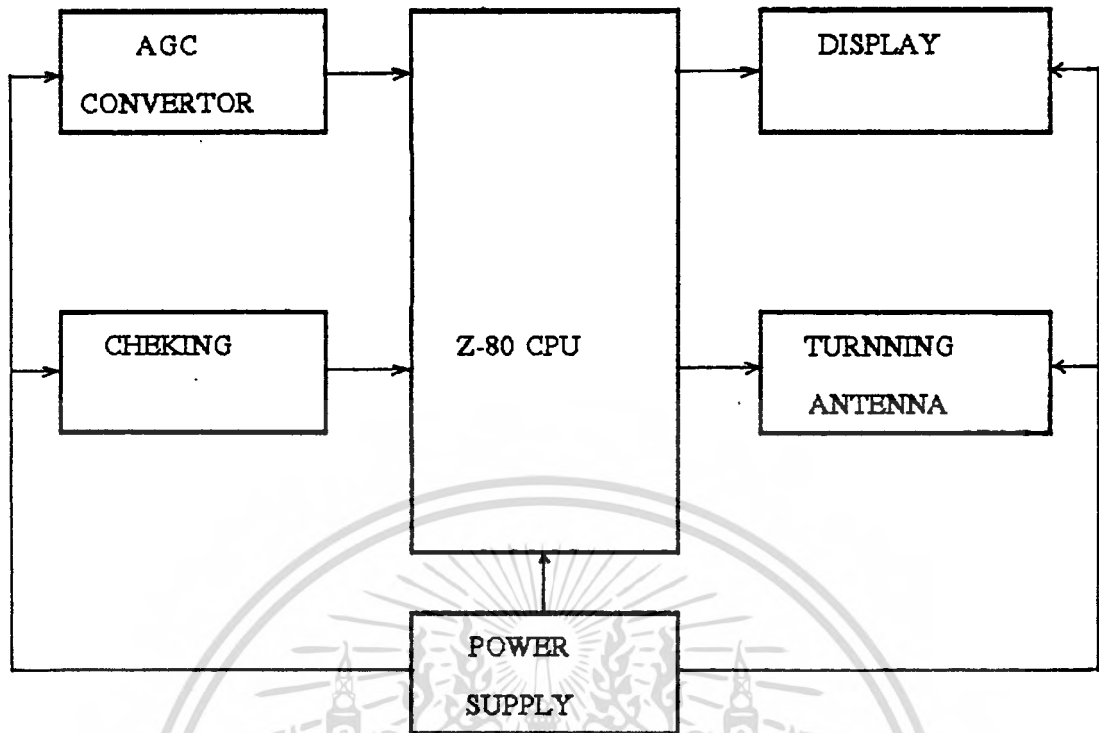
สัญญาณโทรทัศน์จะถูกรับเข้ามาที่สายอากาศ แล้วทำการขยายสัญญาณนั้นในภาค Pre-Amp ก่อนที่จะส่งไปยังภาคเลือกช่องความถี่ (TUNNER) และ IF AMP DETECTOR ต่อไปในภาค IF นี้จะมีแรงดันอยู่อันหนึ่งคอยควบคุมอัตราขยายของสัญญาณโดยอัตโนมัติ ซึ่งเรียกว่า AGC (Automatic Gain Control) หรือ ACC (Automatic Color Control) ถ้าสัญญาณเข้ามาแรงเกินไป AGC จะผลิตแรงไฟควบคุมให้น้อยลง เพื่อจะลดอัตราขยายของภาค IF Amp หรือถ้าสัญญาณมาอ่อน AGC จะเพิ่มแรงดัน เพื่อให้ GAINS การขยายมากขึ้นทำให้ Amplitude ทางออกของสัญญาณคงที่ ซึ่งสัญญาณที่ใช้ควบคุมแรงดันของ AGC นี้ถูกส่งย้อนกลับมาจากภาค DETECTOR นั้นเอง

เพราะฉะนั้น เมื่อหมุนสายอากาศไป 360 องศา แรงดัน AGC ที่วัดได้ในแต่ละตำแหน่งจะไม่เท่ากัน และจุดที่ชัดเจนที่สุดคือ จุดที่สัญญาณเข้ามาแรงที่สุด ซึ่งจุดนี้แรงดันของ AGC มีค่าน้อยด้วย

สรุปได้ว่า ในการหมุนสายอากาศไป 1 รอบ จะสามารถหาจุดแรงที่สุดของสัญญาณได้ซึ่งสามารถใช้ Microprocessor ในการควบคุมการหมุนของสายอากาศและการประมวลผลได้

#### **3.2 ส่วนประกอบของโครงงานและการทำงานของแต่ละบล็อก**

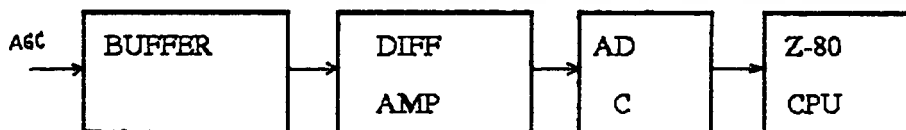
โครงงานนี้ประกอบด้วยส่วนหลักๆดังรูปบล็อกไดอะแกรมรูปที่ 3.1 ข้างล่างนี้



รูปที่ 3.1 เป็นบล็อกไดอะแกรมโครงสร้างทั้งหมด

ซึ่งหน้าที่ของแต่ละบล็อกมีดังนี้

AGC CONVERTOR เป็นส่วนที่ทำหน้าที่แปลงแรงดัน AGC ให้เป็น BINARY 8 BIT เพื่อให้เป็นข้อมูลในการประมวลผลของ Z-80 CPU ซึ่งภายในบล็อกนี้จะประกอบด้วยวงจร ดังรูปที่ 3.2 ข้างล่างนี้

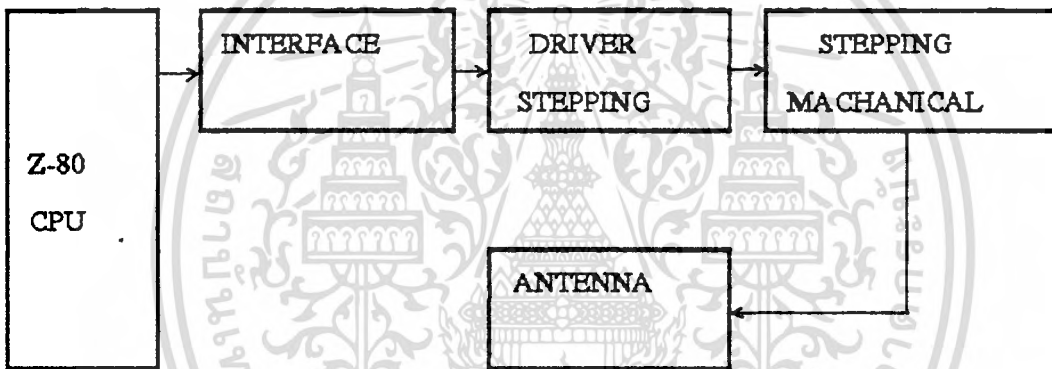


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
รูปที่ 3.2  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โดย BUFFER จะทำหน้าที่ป้องกันการเพิ่ม LOAD ให้กับวงจรเครื่องรับโทรทัศน์ จากนั้นก็จะส่งแรงดัน AGC เข้ายังภาค DIFF AMP ซึ่งทำหน้าที่ทำ SCALE แล้วส่งเข้าเป็นอินพุตของภาค ADC ซึ่งภาค ADC จะทำการแปลงแรงดัน AGC ให้เป็นข้อมูลดิจิทัลแล้วนำเข้า ไปประมวลผลใน Z-80 CPU

DISPLAY เป็นภาคในการแสดงผลค่ามุมที่มอเตอร์หมุนเป็นองศาและระดับความแรงของสัญญาณโดยใช้การแสดงผลเป็นแบบ LCD

TURNING ANTENNA ภาคนี้เป็นภาคที่ทำหน้าที่หมุนเสาอากาศไปในทิศทางและตามตำแหน่งที่ต้องการซึ่งประกอบด้วยวงจรดังรูปที่ 3.3 ข้างล่างนี้



รูปที่ 3.3

เมื่อ Z-80 CPU ทำการประมวลผลค่าแรงดัน AGC แล้วก็จะทำการสั่งให้มอเตอร์หมุนเสาอากาศโดยผ่านทางบล็อก INTERFACE ซึ่งในที่นี้จะใช้ 8255 เป็น PROT อินพุตเข้าพุตในการติดต่อกับภายนอกของ Z-80 CPU ส่วนวงจรบล็อก DRIVER STEPPING จะเป็นตัวที่ทำหน้าที่ขับมอเตอร์ให้หมุนไปตามทิศทางที่ Z-80 CPU สั่งงาน

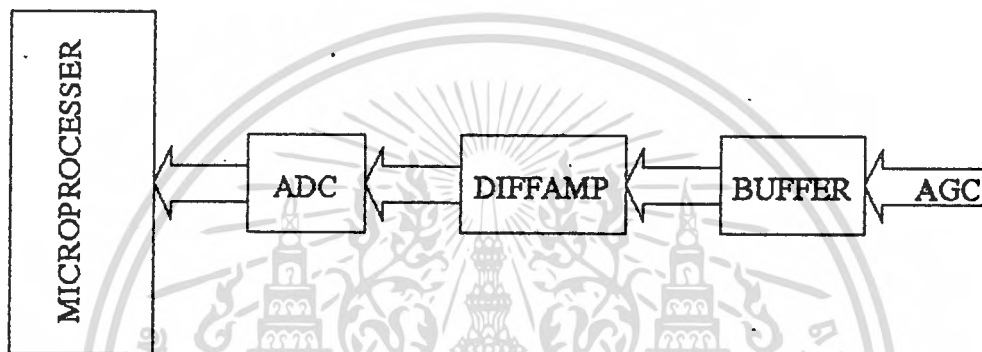
CHECKING บล็อกนี้จะประกอบด้วยวงจร SENSOR ที่คอยตรวจสอบการเปลี่ยนช่องสัญญาณของเครื่องรับโทรทัศน์และการปิด/เปิดเครื่องรับโทรทัศน์และจะประกอบด้วยวงจร SENSOR ที่คอยตรวจสอบตำแหน่งเริ่มต้นในการหมุนของเสาอากาศ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

POWER SUPPLY ภาคจ่ายไฟเป็นภาคที่จ่ายแรงดันเลี้ยงวงจรทั้งระบบโดยภาคจ่ายไฟจะแบ่งออกเป็น 2 ชุดโดยชุดแรกจะจ่ายให้กับ Z-80 CPU , AGC CONVERTOR , CHECKING CONTROL และอีกชุดหนึ่งซึ่งต้องจ่ายกระแสได้สูงจะจ่ายให้กับวงจรภาค TURNING ANTENNA

### 3.3 การออกแบบ

#### 3.3.1 ชุดตรวจสอบแรงดัน AGC



รูปที่ 3.4 BLOCK DIAGRAM ชุดตรวจสอบแรงดัน AGC

#### การทำงาน

แรงดัน AGC จะถูกป้อนผ่าน BUFFER เพื่อป้องกันการเพิ่ม LOAD ให้กับวงจรเครื่องรับโทรทัศน์ และแรงดัน AGC จะถูกส่งมาทำ SCALE 0 - 5 V ที่ DIFF AMP ก่อนจะถูกส่งเข้าวงจร ADC แล้วหลังจากนั้น วงจร ADC ก็จะไปแปลงแรงดัน AGC ให้เป็นสัญญาณข้อมูลที่เป็นดิจิทัล และถูกส่งเข้าไปประมวลผลใน MICROPROCESSER ต่อไป

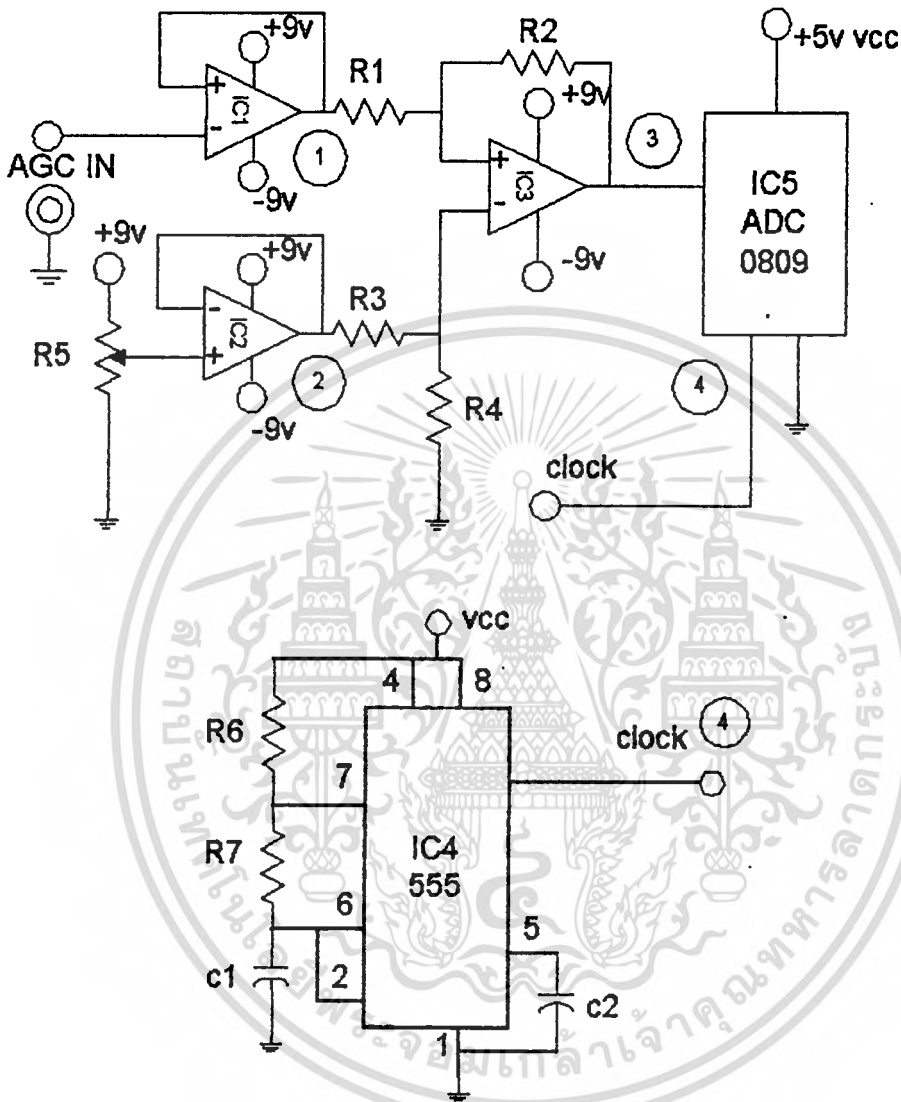
ในการออกแบบภาค AGC CONVERTER จะใช้ IC ดังนี้

IC เบอร์ ADC 0809

IC เบอร์ MC 1458 ใช้ออกแบบเป็น BUFFER และ DIFF AMP

ซึ่งมีวงจรดังนี้

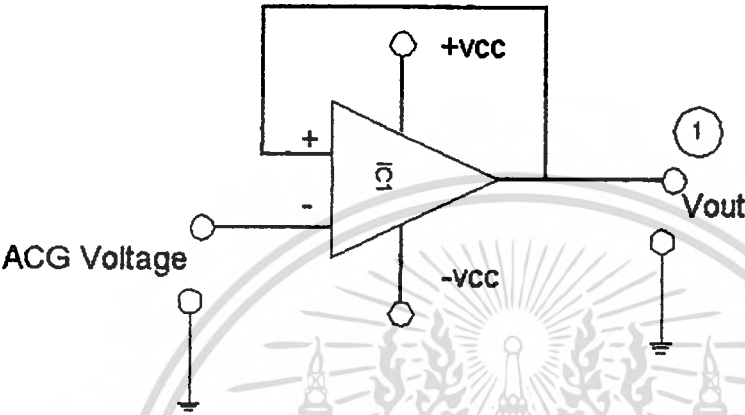
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.5 วงจรชุดตรวจสอบแรงดัน AGC

การวิเคราะห์การทำงานและการออกแบบแต่ละส่วนในชุดตรวจสอบแรงดัน AGC

IC1 ( BUFFER)



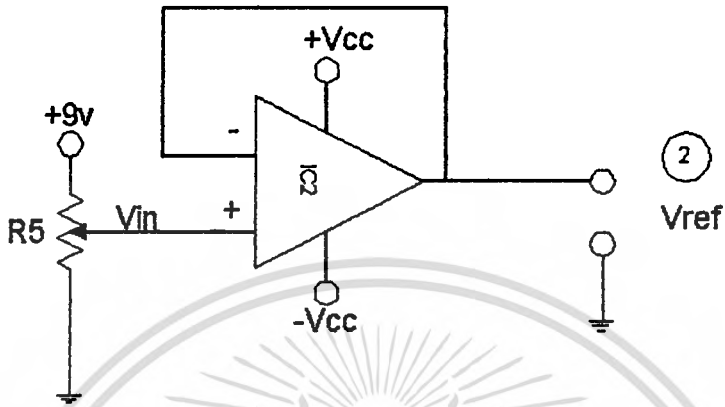
รูปที่ 3.6

จากรูปเป็นวงจร DC VOLTAGE FOLLOWER AMPLIFIER จุดประสงค์สำคัญของวงจรนี้คือ ส่งถ่าย DC VOLTAGE จาก HIGH IMPEDANCE ไปยัง LOW IMPEDANCE ซึ่ง GAIN ของวงจรเท่ากับ 1 เท่า จึงนิยมนำไปใช้เป็น BUFFER หากต้องการให้การทำงานของวงจรได้ผลดีที่สุดแล้ว OUTPUT VOLTAGE ควรไม่เกิน 80% ของ SUPPLY VOLTAGE การออกแบบ

จากวงจร VOLTAGE FOLLOWER ข้างบนเลือกใช้อุปกรณ์ดังนี้

- จากการใช้ AGC(max) = 6 Volte ( 5 - 6 volte )
- เลือกใช้ OPAMP เบอร์ MC1458
- เลือกใช้ SUPPLY = + 9Volte , - 9Volte

## IC2 (REFERENCE VOLTAGE)



รูปที่ 3.7

จากวงจรเป็นวงจร BUFFER ที่  $V_{in}$  สามารถปรับได้เพื่อที่จะให้  $V_{ref}$  ตามต้องการ

- ในที่นี้ต้องปรับ  $R5$  เพื่อที่จะให้ได้  $V_{ref} = 6\text{ V}$  ต้องปรับ  $V_{in} = 6\text{ volts}$

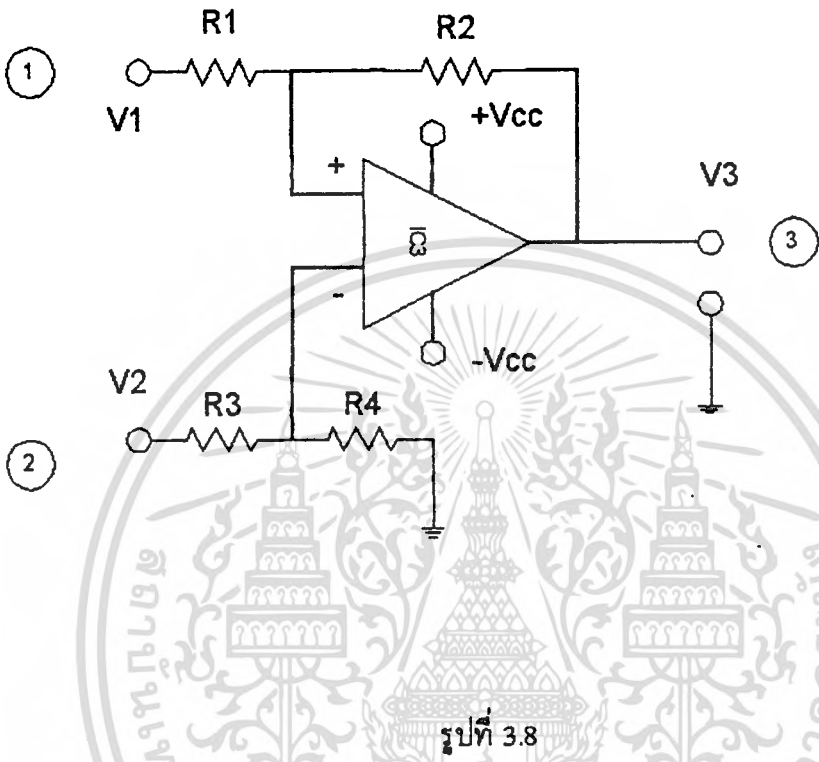
### การออกแบบ

จากวงจรเลือกใช้อุปกรณ์ดังนี้

- เลือก POWER SUPPLY = + 9 volts , - 9 volts
- เลือก  $R5 = R$  ปรับค่า  $10\text{ K}\Omega$
- OPAMP เบอร์ MC 1458

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

IC3 (SCALE 0.5 V)



จากวงจรเป็นวงจร DIFFERENTIAL DC AMPLIFIER

โดยที่  $V3 = V2 * [ ( R4 / (R3 + R4) ) * ( (R1 + R2) / R1 ) ] - V2 * [ ( R2 / R1 ) ]$

การออกแบบ

จากวงจรเลือกใช้อุปกรณ์ดังต่อไปนี้

- เลือก  $R1 = R3$
- เลือก  $R2 = R4$

หาค่า  $R1, R2, R3, R4$  ดังนี้

จาก

$$\begin{aligned} V3 &= V2 * [ ( R4 / (R3 + R4) ) * ( (R1 + R2) / R1 ) ] - V2 * [ ( R2 / R1 ) ] \\ V3 &= V2 * [ ( R2 / (R1 + R2) ) * ( (R1 + R2) / R1 ) ] - V2 * [ ( R2 / R1 ) ] \\ &= [ ( V2 - V1 ) * ( R2 / R1 ) ] \end{aligned}$$

เอกสารนี้เป็นเอกสารตัวอย่างสำหรับใช้ในการเรียนการสอนเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้เผยแพร่ข้อมูลนี้ออกนอกห้องเรียนและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้  
V1 = 5 ~ 6 VLOTS

$$V_2 = 6 \text{ VOLTS}$$

โดยการให้  $V_1 = 5 \text{ Volts}$

$$V_3 = [6 - 5] * (R_2/R_1) = 5 \text{ Volts}$$

$$(R_2/R_1) = 5 = A_v$$

ดังนั้นจึงเลือกค่า  $R_1$  และ  $R_2$  ดังนี้

$$\text{- เลือก } R_1 = 25 \text{ K}\Omega$$

$$\text{- เลือก } R_2 = 125 \text{ K}\Omega$$

จากการให้  $R_1 = R_3$  และ  $R_2 = R_4$  ดังนั้นค่า  $R_1$ ,  $R_2$ ,  $R_3$  และ  $R_4$  จึงมีค่าดังนี้

$$R_1 = R_3 = 25 \text{ K}\Omega$$

$$R_2 = R_4 = 125 \text{ K}\Omega$$

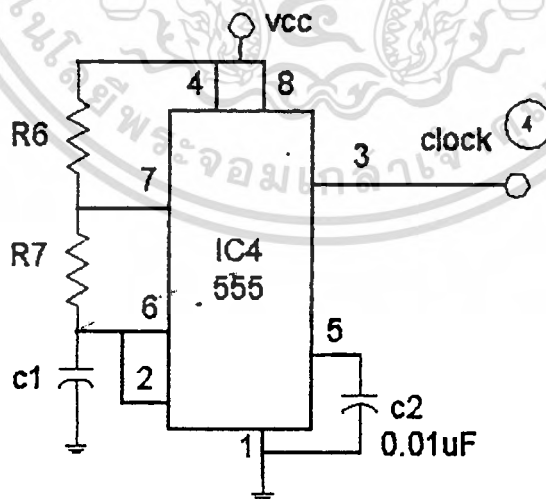
IC3 ใช้เบอร์ MC 1458

เลือก  $V_{CC} = +9 \text{ Volts}$ ,  $-9 \text{ Volts}$

เมื่อ  $V_1 = 6 \text{ Volts}$  จะทำให้  $V_3 = 0 \text{ Volts}$

Range ของแรงดัน  $V_3 = 0 \sim 5 \text{ Volts}$

#### IC4 CLOCK GENERATER



รูปที่ 3.9

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากวงจรเป็นวงจร Astable ซึ่งนำมาสร้างสัญญาณ Clock เพื่อที่จะป้อนให้กับวงจร ADC จากสมการทางคณิตศาสตร์ของวงจรมีดังนี้

$$T_H = -(R_6 + R_7) C_1 \ln \left[ \frac{V_{cc} - 2/3 V_{cc}}{V_{cc} - 1/3 V_{cc}} \right]$$

$$T_L = 0.693 R_7 C_1$$

$$T = T_H + T_L$$

$$= 0.693 (R_6 + 2R_7) C_1$$

$$f = 1/T$$

$$= 1.443 / ((R_6 + 2R_7) C_1)$$

$$\text{DUTY CYCLE} = T_H / T$$

$$= (R_6 + R_7) / (R_6 + 2R_7)$$

#### การออกแบบ

$$\text{- เลือกใช้ความถี่} = 4 \text{ KHz}$$

$$\text{- DUTY CYCLE} = 50\%$$

$$\text{- เลือก } R_7 = 1 \text{ K}\Omega$$

$$\text{จาก DUTY CYCLE \%} = [(R_6 + R_7) / (R_6 + 2R_7)] \times 100$$

$$0.5 (R_6 + 2 \text{ K}\Omega) = R_6 + 1 \text{ K}\Omega$$

$$0.5 R_6 = 1 \text{ K}\Omega$$

$$R_6 = 2 \text{ K}\Omega$$

$$\text{จาก } f = 1.443 / ((R_6 + 2R_7) C_1)$$

$$4 \text{ kHz} = 1.443 / ((2 \text{ K} + 2 \text{ K}) C_1)$$

$$\text{ดังนั้น } C_1 = 1.443 / 16 \text{ M}$$

$$= 0.09 \text{ }\mu\text{F}$$

ดังนั้นในวงจรสร้าง Clock จึงเลือกใช้อุปกรณ์ต่าง ๆ ดังนี้

$$\text{- } R_6 = 2 \text{ K}\Omega$$

$$\text{- } R_7 = 1 \text{ K}\Omega$$

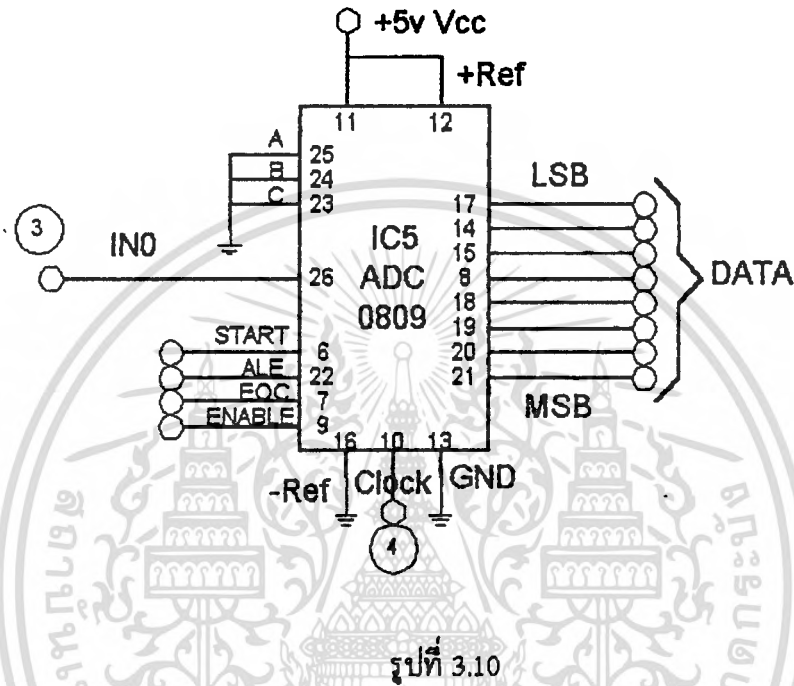
$$\text{- } C_1 = 0.1 \text{ }\mu\text{F}$$

$$\text{- } C_2 = 0.01 \text{ }\mu\text{F}$$

$$\text{- IC4 ใช้เบอร์ 555}$$

เอกสารนี้เป็นเอกสารที่สงวนไว้ Power Supply + 9 Volts ศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

IC5 ( AGC VOLTAGE TO BINARY DATA 8 BIT )



จากรูปข้างบนจะใช้ IC 0809 เพื่อที่จะแปลง AGC VOLTAGE ให้เป็น BINARY DATA 8 BIT

$+Ref = 5\text{ V}$   
 $-Ref = 0\text{ V}$   
 $IN0 = 0 \sim 5\text{ V}$   
 $255\text{ ( STEP ของ OUTPUT ) } = 5\text{ V}$   
 $1\text{ STEP} = 5/255\text{ V}$   
 $= 0.02\text{ V}$

ซึ่งมีความหมายว่า เมื่อแรงดันที่ IN0 เปลี่ยนไป 0.02 V จะทำให้ DATA ที่ OUTPUT เปลี่ยนไป 1 ค่า

**สรุปการทำงานชุด AGC CONVERT**

เอกสารนี้เป็นเอกสารที่สแกนจากเว็บไซต์ของกรมส่งเสริมการค้าระหว่างประเทศ กระทรวงพาณิชย์ ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น ผ่าน IC1 มิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Voltage ที่จุด 1 = 5 V

ผ่าน IC3

Voltage ที่จุด 3 = [ 6 - 5 ]5 = 5 V

ผ่าน IC5

AGC STEP = 5/0.02 = 250 STEP

เปลี่ยนจาก DECIMAL เป็น BINARY

250 เป็น 11111010

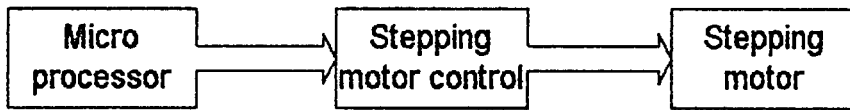
ดังนั้น AGC ที่เข้ามา 5 V จะเป็นข้อมูลดิจิทัลคือ 11111010

สรุปอุปกรณ์ที่ใช้ในวงจร AGC CONVERT

|                 |     |                 |
|-----------------|-----|-----------------|
| IC1 , IC2 , IC3 | ใช้ | MC 1458         |
| IC4             | ใช้ | 555             |
| IC5             | ใช้ | ADC 0809        |
| R1 , R3         | ใช้ | 25KΩ            |
| R2 , R4         | ใช้ | 125KΩ           |
| R5              | ใช้ | R เกือกม้า 10KΩ |
| R6              | ใช้ | 2KΩ             |
| R7              | ใช้ | 1KΩ             |
| C1              | ใช้ | 0.1uF           |
| C2              | ใช้ | 0.01uF          |

3.3.2 ชุดควบคุมการหมุนของสายอากาศ

ในการหมุนสายอากาศเราจะใช้ STEPPING MORTOR เป็นตัวขับเคลื่อนที่ ไมโครโปรเซสเซอร์จะส่งข้อมูลให้ชุดควบคุมการหมุน STEPPING MORTOR แสดงบล็อกการ ควบคุมการหมุนของสายอากาศดังนี้

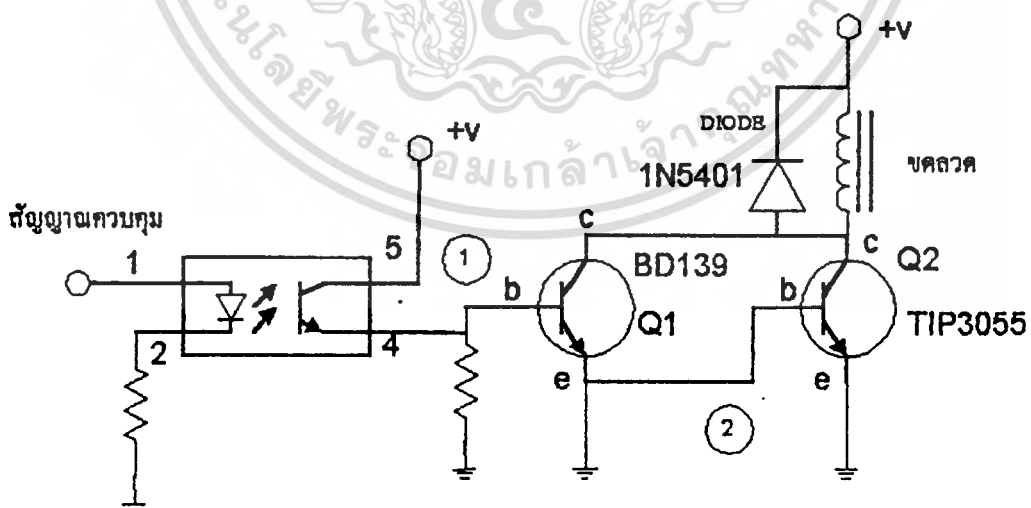


BLOCK DIAGRAM ชุดควบคุมการหมุนของสายอากาศ

รูปที่ 3.11

#### การออกแบบชุด Stepping motor control

ในการออกแบบวงจรขับเคลื่อน Stepping motor สิ่งที่ต้องคำนึงถึงคือ ขดลวดของตัว Stepping motor ต้องการกระแสและแรงดันเท่าไร เพื่อที่จะทำให้ Stepping motor สามารถทำงานได้ การออกแบบวงจรให้หลักการ การ เปิด - ปิด กระแส ผ่านขดลวด Stepping motor ดังแสดงในรูปข้างล่าง



รูปที่ 3.12

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากรูปวงจรเมื่อสัญญาณควบคุมเป็น ' H ' เอาท์พุทที่จุด 1 จะเป็น ' H ' ซึ่งจะทำให้ขา B ของทรานซิสเตอร์ Q1 ได้รับการไบอัส ทำให้มีกระแสคอลเลกเตอร์ [ IC1 ] และทำให้ทรานซิสเตอร์ Q2 ได้รับการไบอัสที่ขา B ที่จุด 2 ทำให้ทรานซิสเตอร์ Q2 นำกระแสทำให้มีกระแสไหลผ่านขดลวดของ Stepping motor ซึ่ง Q2 ทำหน้าที่เสมือนเป็นสวิตช์ ON - OFF เพื่อให้มีกระแสไหลผ่านขดลวดของ Stepping motor

เมื่อสัญญาณควบคุมเป็น ' L ' เอาท์พุทที่จุด 1 จะเป็น ' L ' ทำให้ที่ขา B ของทรานซิสเตอร์ Q1 เป็น 0 ทำให้ทรานซิสเตอร์ Q1 ไม่มีการนำกระแสเป็นผลให้ Q2 ไม่นำกระแสด้วยและส่งผลให้ไม่มีกระแสไหลผ่านขดลวดของ Stepping motor ด้วย

#### หลักสำคัญในการออกแบบวงจรขับเคลื่อน

##### 1. แรงดันคอลเล็กเตอร์ - อิมิตอร์ [ $V_{ce}$ ]

เป็นแรงดันสูงสุดระหว่างขาคอลเล็กเตอร์กับอิมิตอร์ ของทรานซิสเตอร์ แต่ในการใช้งานจริงแล้วจะไม่ใช้แรงดันของแหล่งจ่ายสูงกว่าค่า  $V_{ce}$  เนื่องจากคุณสมบัติของขดลวด เมื่อตัดกระแสที่จ่ายเข้าจะเกิดแรงดันย้อนกลับคือแรงดัน EMF จากเหตุผลนี้เองจึงต้องมีการต่อไดโอดขนานกับขดลวดเพื่อบายพาสกระแสที่เกิดจากแรงดัน EMF เมื่อต่อไดโอดเข้าในวงจรแรงดันคอลเล็กเตอร์จะมีค่าเป็น  $V_1 + V_2$

โดยที่  $V_1$  คือ แรงดันแหล่งจ่าย

$V_2$  คือ แรงดัน Forward ของไดโอด

ดังนั้นในการออกแบบใช้งานจริงควรจะต้องเลือกใช้ทรานซิสเตอร์ที่มีค่า  $V_{ce}$  ตั้งแต่ 2 เท่าของแรงดันแหล่งจ่ายไฟ [  $V_1$  ] จึงจะไม่มีปัญหา

##### 2. กระแสคอลเล็กเตอร์ [ IC ]

กระแสที่ไหลผ่านขดลวดนั้นจะต้องไม่เกินค่ากระแสคอลเล็กเตอร์สูงสุดของทรานซิสเตอร์

##### 3. แรงดันอิ่มตัวของทรานซิสเตอร์ - อิมิตอร์ [ $V_{ce(sat)}$ ]

$V_{ce(sat)}$  ของทรานซิสเตอร์จะเป็นแรงดันตกคร่อมระหว่างคอลเล็กเตอร์กับอิมิตอร์เมื่อทรานซิสเตอร์ " ON " เต็มที่ซึ่งถ้าค่า  $V_{ce(sat)}$  มีค่าน้อยประสิทธิภาพในการขับเคลื่อนขดลวดจะดี

##### 4. ค่าสูญเสียของทรานซิสเตอร์ [ PC ]

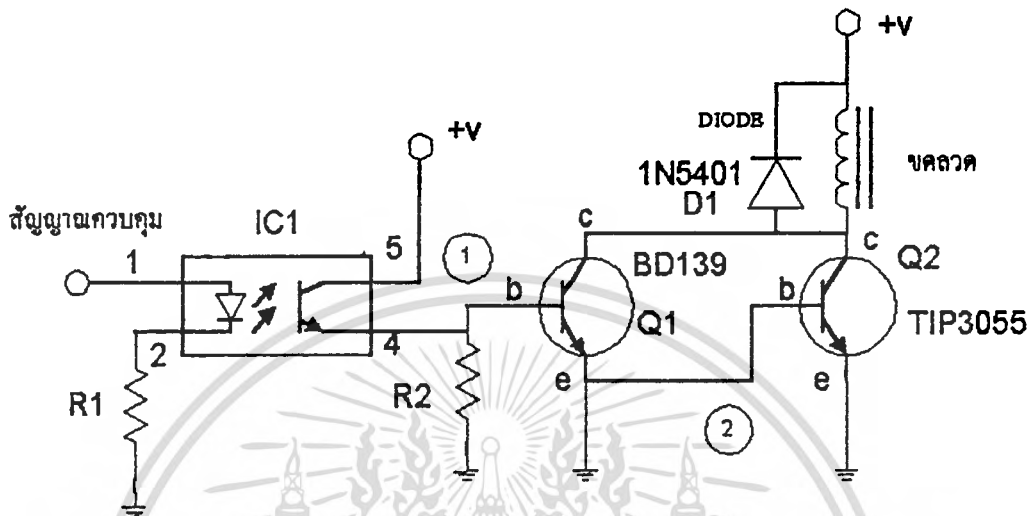
เป็นค่ากำลังงานสูญเสียของทรานซิสเตอร์ ซึ่งมีค่าเท่ากับ  $V_{ce(sat)} * IC$

ซึ่งจะเห็นได้ว่าเมื่อ  $V_{ce(sat)}$  มีค่ายิ่งต่ำค่าสูญเสียก็จะยิ่งน้อย และถ้าใช้งานทรานซิสเตอร์ให้มีค่า  $P_o$  สูงกว่าค่าที่กำหนด จะทำให้ทรานซิสเตอร์ชำรุดได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## การออกแบบวงจรขับ Stepping motor

วงจรขับ Stepping motor ดังรูปข้างล่างนี้



รูปที่ 3.13 แสดงวงจรขับ Stepping 1 เฟส

จากวงจรจะประกอบด้วยอุปกรณ์ต่าง ๆ ดังนี้

- IC1 เป็น OPTOTRANSISTOR ซึ่งจะทำหน้าที่เชื่อมต่อสัญญาณควบคุมจาก Z80 กับชุดขับ Stepping motor เหตุที่ใช้ตัวเชื่อมต่อทางแสงเพื่อเป็นการลดภาระให้กับ Z80
- Q1 และ Q2 เป็นทรานซิสเตอร์ซึ่งต่อในลักษณะเป็นคาลิตัน
- D1 เป็นไดโอดที่ต่อเพื่อป้องกันแรงดัน EMF ย้อนกลับป้องกันไม่ให้ทรานซิสเตอร์เสียหาย ซึ่งไดโอดนี้จะต้องทนกระแสได้ไม่น้อยกว่า 2 เท่า ของกระแสที่ใช้กับขดลวด Stepping motor เนื่องจากข้างบนเป็นวงจรขับ Stepping motor เพียง 1 เฟส ดังนั้นหากใช้มอเตอร์ 4 เฟสก็ต้องทำชุดขับทั้ง 4 เฟส

### สรุปอุปกรณ์ที่ใช้ในภาคขับ Stepping motor

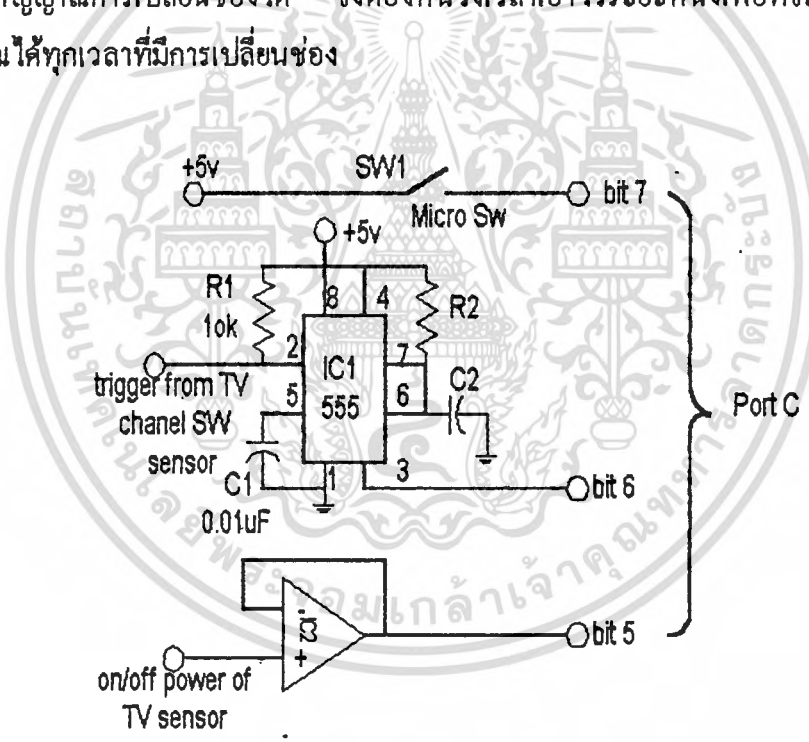
|       |     |                   |
|-------|-----|-------------------|
| - R1  | ใช้ | 150K $\Omega$ * 4 |
| - R2  | ใช้ | 47K $\Omega$ * 4  |
| - D1  | ใช้ | 1N5401 * 4        |
| - IC1 | ใช้ | 4N26 * 4          |
| - Q1  | ใช้ | BD139 * 4         |
| - Q2  | ใช้ | TIP3055 * 4       |

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.3.3 ชุดตรวจสอบการเปลี่ยนช่องความถี่และตรวจสอบระบบอื่น ๆ

ในการเปลี่ยนช่องความถี่ และการเริ่มเปิดเครื่องรับโทรทัศน์เราจะให้สายอากาศหมุนไปยังจุดเริ่มต้นโดยการตรวจสอบที่สวิทช์ที่ติดไว้ที่จุดเริ่มต้นของการหมุนแล้วเริ่มทำการหมุนสายอากาศไปจนครบ 360 องศา เพื่อค้นหาตำแหน่งที่ชัดเจนที่สุดของสัญญาณและเมื่อมีการปิดเครื่องรับโทรทัศน์สายอากาศก็จะหมุนกลับมายังตำแหน่งเริ่มต้นอีก แล้วทำการตรวจสอบว่ามีการเปิดเครื่องรับโทรทัศน์หรือไม่ ถ้ามีการเปิดเครื่องรับโทรทัศน์ก็จะเริ่มต้นการทำงานเหมือนข้างต้นอีก

ชุดตรวจสอบในการเปลี่ยนช่องความถี่จะมีสัญญาณ TRIGGER ที่เกิดจากการเปลี่ยนช่องเพียงช่วงเวลาสั้น ๆ เพราะฉะนั้นในขณะที่กำลังหมุนมอเตอร์อาจจะทำให้ไม่สามารถตรวจจับสัญญาณการเปลี่ยนช่องได้ จึงต้องหน่วงเวลาเอาไว้ระยะหนึ่งเพื่อที่จะสามารถตรวจจับสัญญาณได้ทุกเวลาที่มีการเปลี่ยนช่อง



รูปที่ 3.14 วงจรชุดระบบ SENSOR

IC1 [ TV CHANNEL SW. SENSOR ]

เป็นวงจร Monostable มีไว้เพื่อทำการหน่วงเวลา Pluse ที่ Channel Sensor รับจากการเปลี่ยน Channel ของ เครื่องรับ TV เพื่อให้ Microprocessor สามารถตรวจจับสัญญาณการเปลี่ยนช่องมาตรวจสอบได้ ซึ่งตั้งเวลาในการหน่วงไว้ 0.5 Sec

ช่วงหน่วงเวลาของวงจร Monostable โดยใช้ 555 ขึ้นอยู่กับผลคูณของ R2 และ C2

ซึ่งเวลาในการหน่วงจะเท่ากับ

$$T [\text{delay}] = 1.1R_2 \cdot C_2$$

การออกแบบ

เนื่องจากต้องการเวลาหน่วง 0.5 Sec

$$T [\text{delay}] = 0.5 \text{ Sec}$$

$$\text{เลือกใช้ } R_2 = 100 \text{ K}\Omega$$

คำนวณหาค่า  $C_2$

$$\begin{aligned} C_2 &= T [\text{delay}] / 1.1R_2 \\ &= 0.5 / (1.1 \cdot 100 \cdot 10000) \\ &= 4.5 \text{ }\mu\text{F} \end{aligned}$$

เพราะฉะนั้นจึงเลือกใช้

$$R_2 = 100 \text{ K}\Omega$$

$$C_2 = 5 \text{ }\mu\text{F}$$

#### IC2 [ ON/OFF TV SENSOR BUFFER ]

โดยการใช้ IC เบอร์ MC 1458 คือเป็นวงจร Voltage Follower amplifier

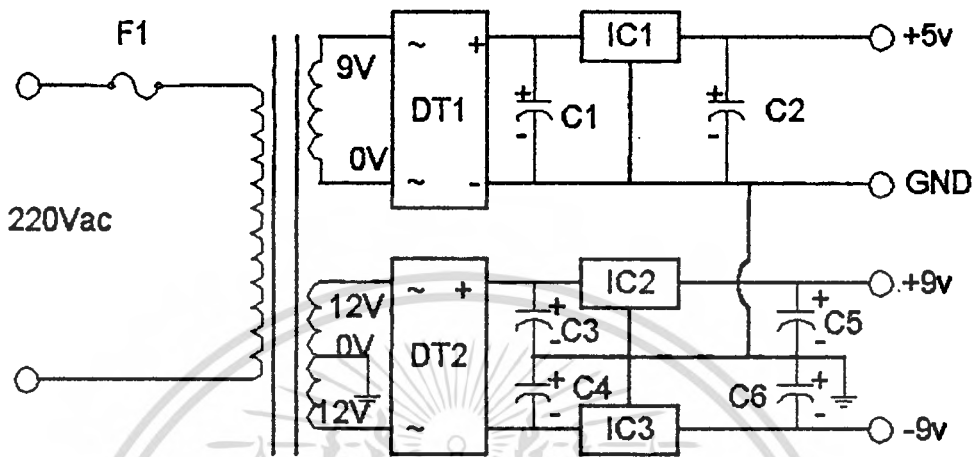
สรุปอุปกรณ์ที่ใช้ในชุด SENSOR

|       |     |                                 |
|-------|-----|---------------------------------|
| - IC1 | ใช้ | 555                             |
| - IC2 | ใช้ | MC1458                          |
| - R1  | ใช้ | 10 K $\Omega$                   |
| - R2  | ใช้ | 100K $\Omega$                   |
| - C1  | ใช้ | 0.01 $\mu\text{F}$ ( ceramic )  |
| - C2  | ใช้ | 5 $\mu\text{F}$ ( Elcotrolize ) |

๙

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 3.3.4 ชุด POWER SUPPLY



รูปที่ 3.15

เป็น Power Supply สำหรับจ่ายไฟ +5V และ +9V, -9V

สำหรับ +5V จะจ่ายให้กับทุกภาคในระบบ

สำหรับ +9V และ -9V จะจ่ายให้กับ OPAMP ในชุดตรวจสอบแรงดัน AGC

#### การออกแบบ

+ 5V ต้องจ่ายกระแสประมาณ 5 A ให้กับระบบ เพราะฉะนั้น จึงใช้ IC Regulator +5 V,8A เพื่อให้สูงกว่ากระแสที่ใช้งาน

DT1 เลือกใช้ BRIDGE RECTIFIER ขนาด 10 A

ขด SECONDARY 9V ของ T1 ต้องจ่ายกระแสได้ประมาณ 5A

+9V และ -9V do จ่ายกระแสให้ระบบประมาณ 50 mA จึงเลือกใช้ IC2 เป็น IC REGULATOR +9V,100mA และ IC3 เป็น IC REGULATOR -9V,100mA

DT2 เลือกใช้ BRIDGE RECTIFIER ขนาด 0.5 A

ขด SECONDARY 12V ต้องจ่ายกระแสได้ประมาณ 100mA

#### สรุปอุปกรณ์ชุด POWER SUPPLY

-T1 ใช้ trap 0,9v,5A และ trap 15V , 0 , 15V ,0,5A

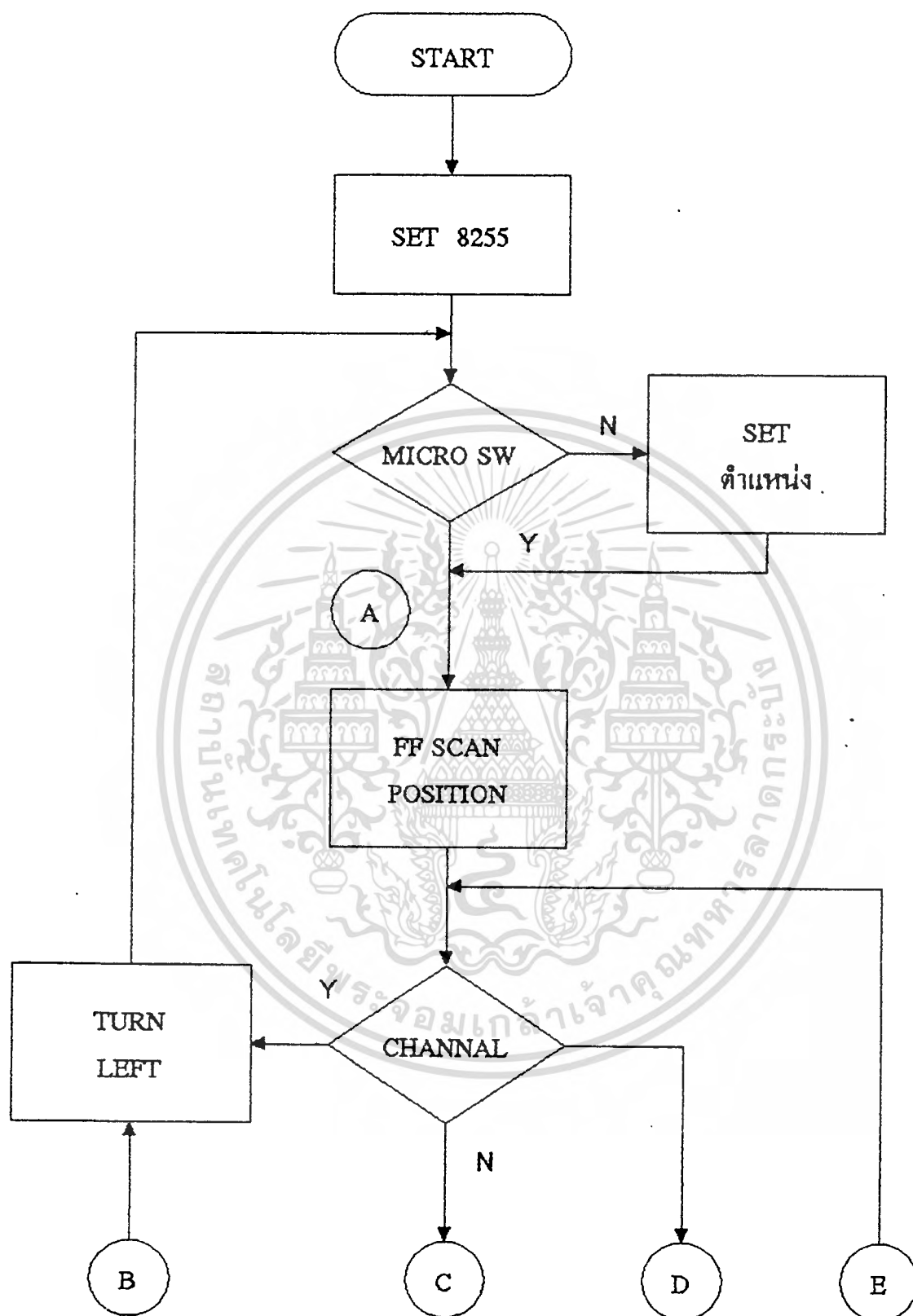
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับใช้ภายในเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้ง IC1 ที่ตัดใช้ลงนี้ 3052 x2 ต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- IC2     ใช้     7809
- IC3     ใช้     7909
- C1,C3,C4     ใช้     2200uF x 3
- C2,C5,C6     ใช้     1000uF x 3

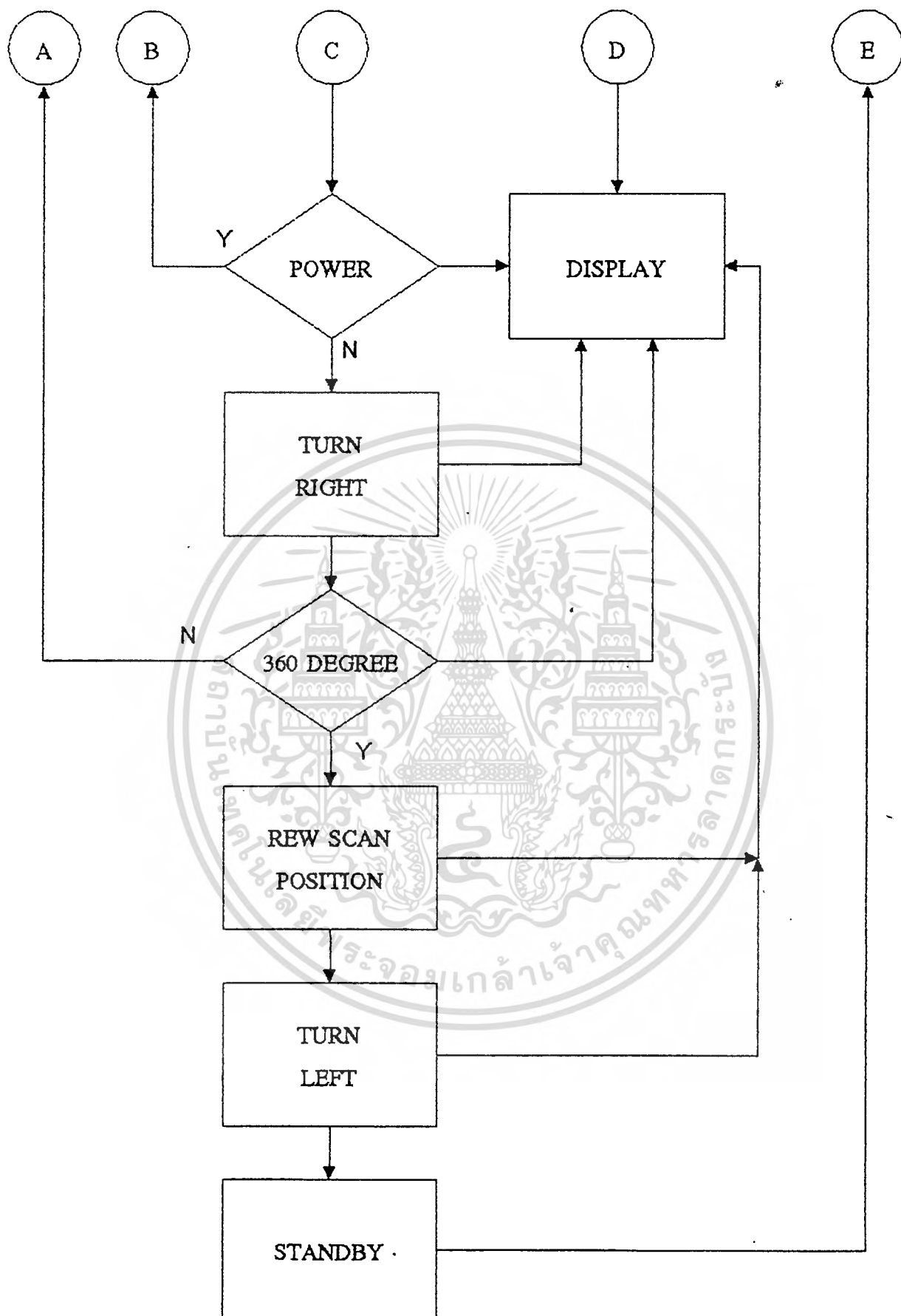
### 3.3.5 การออกแบบโปรแกรมของเครื่องควบคุม

ในการออกแบบการเขียนโปรแกรมของเครื่องสายอากาศปรับทิศทางอัตโนมัติ นั้นจะนำหลักการของเครื่องสายอากาศปรับทิศทางอัตโนมัติมาใช้ในการออกแบบการเขียนโปรแกรมให้ได้ตามขอบเขตที่กำหนดไว้โดยเริ่มต้นจากการออกแบบ FLOW CHART ให้มีเงื่อนไขตามที่กำหนดครบถ้วน ต่อจากนั้นก็ทำการเขียนโปรแกรมการทำงานของเครื่องสายอากาศปรับทิศทางอัตโนมัติตาม FLOW CHART ที่ได้ออกแบบไว้ดังแสดงในรูปที่ 3.16





เอกสารนี้เป็นเอกสารที่เผยแพร่ไว้สำหรับเอกสารใช้เฉพาะเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
 ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
 รูปที่ 3.17 แสดง FLOW CHART โปรแกรมของเครื่องควบคุม (ต่อ)  
 ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## โปรแกรมของเครื่องควบคุม

```

1          ;*****
2          ;**  MAIN PROGRAM FINE AGC LEST THAN  **
3          ;**          SUM LCD DISPLAY          **
4          ;*****
5          0000          ORG 0000H
6          00E0          PORTA EQU 0E0H
7          00E1          PORTB EQU 0E1H
8          00E2          PORTC EQU 0E2H
9          00E3          CONTP EQU 0E3H
10         00A0          S_AGC EQU 0A0H
11         00C0          R_AGC EQU 0C0H
12         2600          DAGC EQU 2600H
13         2700          DAMOT EQU 2700H
14         2800          DIF EQU 2800H
15         2900          DALOG EQU 2900H
16         2A00          CBUF EQU 2A00H
17 0000 3E 00          LD A,00H
18 0002 3D          PRE DEC A
19 0003 20 FD          JR NZ,PRE
20 0005 31 FF 2E          LD SP,2EFFFH
21 0008 3E 88          LD A,88H
22 000A D3 E3          OUT (CONTP),A ;INITIAL 8255
23 000C CD A8 01          CALL DIS1 ;DISPLAY NAME PROJECT
24 000F 3E 33          LD A,33H
25 0011 32 00 27          LD (DAMOT),A
26 0014 AF          STAT XOR A
27 0015 DB E2          IN A,(PORTC)
28 0017 CB 67          BIT 4,A

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

29 0019 C4 A8 00      CALL NZ,ZEPOINT
30 001C CD C9 01      CALL DIS2
31 001F AF            XOR A
32 0020 32 00 29      LD (DALOG),A
33 0023 32 01 29      LD (DALOG+1),A
34 0026 06 C8         LD B,C8H      ;VALUE STEP TRUNNING
35 0028 11 00 00      LD DE,0000H
36 002B 3E FF         LD A,FFH
37 002D 4F            LD C,A
38 002E D3 A0         LOOP OUT (S_AGC),A ;ENABLE AGC CONVERTER
39 0030 CD FB 00      CALL DELAYT
40 0033 DB C0         IN A,(R_AGC) ;READ VALUE AGC
41 0035 32 00 26      LD (DAGC),A
42 0038 1C            INC E
43 0039 91            SUB C
44 003A 28 07         JR Z,DRIVE
45 003C 30 05         JR NC,DRIVE
46 003E 3A 00 26      LD A,(DAGC)
47 0041 4F            LD C,A      ;KEEP VALUE LEST THAN AGC
48 0042 53            LD D,E      ;KEEP POTTION LEST THAN AGC
49 0043 CD C7 00      DRIVE CALL RIGHT ;TURNING ONE STEP
50 0046 DB E2         IN A,(PORTC)
51 0048 CB 6F         BIT 5,A      ;CHECK CH
52 004A 20 38         JR NZ,CHDI
53 004C CB 77         BIT 6,A      ;CHECK POWER
54 004E 28 46         JR Z,PODI
55 0050 05            DEC B
56 0051 20 DB         JR NZ,LOOP
57 0053 C5            PUSH BC
58 0054 E5            PUSH HL

```

เอกสารนี้เป็นเอกสารสงวนลิขสิทธิ์สำหรับการใช้งานเพื่อ  
 ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

59 0055 D5          PUSH DE
60 0056 CD DA 01     CALL DIS3;DISPLAY SCAN POSITION REVERT
61 0059 D1          POP DE
62 005A E1          POP HL
63 005B C1          POP BC
64 005C 3E C8       LD A,C8H
65 005E 92          SUB D      ;FINE POSITION REVERTS
66 005F 47          LD B,A
67 0060 CD B8 00     REVRES CALL LEFT
68 0063 DB E2       IN A,(PORTC)
69 0065 CB 6F       BIT 5,A      ;CHECK CH
70 0067 20 1B       JR NZ,CHDI
71 0069 CB 77       BIT 6,A      ;CHECK POWER
72 006B 28 29       JR Z,PODI
73 006D 05          DEC B
74 006E 20 F0       JR NZ,REVRES
75 0070 CD EB 01     CALL DIS4 ;DISPLY ANGLE STANDBY
76 0073 AF          XOR A
77 0074 AF          STANDBY XOR A
78 0075 DB E2       IN A,(PORTC)
79 0077 CB 6F       BIT 5,A      ;CHECK CH
80 0079 C2 84 00     JP NZ,CHDI
81 007C CB 77       BIT 6,A      ;CHECK POWER
82 007E CA 96 00     JP Z,PODI
83 0081 18 F1       JR STANDBY
84 0083 C9          RET
85 0084 F5          CHDI  PUSH AF
86 0085 C5          PUSH BC
87 0086 E5          PUSH HL
88 0087 D5          PUSH DE

```

เอกสารนี้เป็นเอกสารสงวนลิขสิทธิ์สำหรับการใช้งานเพื่อ Push DE เท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

|     |      |          |                                 |
|-----|------|----------|---------------------------------|
| 89  | 0088 | CD FA 01 | CALL DISCH                      |
| 90  | 008B | CD E4 00 | CALL DELAYD                     |
| 91  | 008E | D1       | POP DE                          |
| 92  | 008F | E1       | POP HL                          |
| 93  | 0090 | C1       | POP BC                          |
| 94  | 0091 | F1       | POP AF                          |
| 95  | 0092 | C3 14 00 | JP STAT                         |
| 96  | 0095 | C9       | RET                             |
| 97  | 0096 | F5       | PODI PUSH AF                    |
| 98  | 0097 | C5       | PUSH BC                         |
| 99  | 0098 | E5       | PUSH HL                         |
| 100 | 0099 | D5       | PUSH DE                         |
| 101 | 009A | CD 0B 02 | CALL DISPO                      |
| 102 | 009D | CD E4 00 | CALL DELAYD                     |
| 103 | 00A0 | D1       | POP DE                          |
| 104 | 00A1 | E1       | POP HL                          |
| 105 | 00A2 | C1       | POP BC                          |
| 106 | 00A3 | F1       | POP AF                          |
| 107 | 00A4 | C3 14 00 | JP STAT                         |
| 108 | 00A7 | C9       | RET                             |
| 109 |      |          | *****                           |
| 110 |      |          | ;** SUB TRUNING TO ZEROPOINT ** |
| 111 |      |          | *****                           |
| 112 | 00A8 | CD 1C 02 | ZEPOINT CALL DISRE              |
| 113 | 00AB | CD E4 00 | CALL DELAYD                     |
| 114 | 00AE | CD B8 00 | ZEPO CALL LEFT                  |
| 115 | 00B1 | DB E2    | IN A,(PORTC)                    |
| 116 | 00B3 | CB 67    | BIT 4,A                         |
| 117 | 00B5 | 20 F7    | JR NZ,ZEPO                      |
| 118 | 00B7 | C9       | RET                             |

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

119      ,*****
120      ,**   SUB TRUNING MORTER REVERTS   **
121      ,*****
122 00B8 3A 00 27    LEFT LD A,(DAMOT)
123 00BB 0F          RRCA
124 00BC 32 00 27    LD (DAMOT),A
125 00BF D3 E0       OUT (PORTA),A
126 00C1 3E 02       LD A,02H
127 00C3 CD 08 01    CALL DISANG
128 00C6 C9          RET
129      ,*****
130      ,**   SUB TRUNING MORTER FORWORD   **
131      ,*****
132 00C7 3A 00 27    RIGHT LD A,(DAMOT)
133 00CA 07          RLCA
134 00CB 32 00 27    LD (DAMOT),A
135 00CE D3 E0       OUT (PORTA),A
136 00D0 3E 01       LD A,01H
137 00D2 CD 08 01    CALL DISANG
138 00D5 AF          XOR A
139 00D6 C9          RET
140      ,*****
141      ,**   SUB DELAY FOR STEPING       **
142      ,*****
143 00D7 08          DELAYM EX AFAF
144 00D8 D9          EXX
145 00D9 21 FF 06    LD HL,06FFH
146 00DC 2B          DM1 DEC HL
147 00DD 7C          LD A,H
148 00DE B5          OR L

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
149 00DF 20 FB          JR NZ,DM1
150 00E1 D9             EXX
151 00E2 08             EX AF,AF
152 00E3 C9             RET
153                      ;*****
154                      ;**   SUB DELAY FOR DISSHOW   **
155                      ;*****
156 00E4 08             DELAYD EX AF,AF
157 00E5 D9             EXX
158 00E6 21 FF 9F       LD HL,9FFFH
159 00E9 2B             DMD  DEC HL
160 00EA 7C             LD A,H
161 00EB B5             OR L
162 00EC 20 FB          JR NZ,DMD
163 00EE D9             EXX
164 00EF 08             EX AF,AF
165 00F0 C9             RET
166 00F1 CD E4 00       WAIT  CALL DELAYD
167 00F4 DB E2          IN A,(PORTC)
168 00F6 CB 77          BIT 6,A
169 00F8 20 F7          JR NZ,WAIT
170 00FA C9             RET
171                      ;*****
172                      ;**   SUB DELAY FOR DISPLAY   **
173                      ;*****
174 00FB 08             DELAYT EX AF,AF
175 00FC D9             EXX
176 00FD 21 FF 06       LD HL,06FFH
177 0100 2B             DT   DEC HL
178 0101 7C             LD A,H
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

|     |      |    |        |                   |
|-----|------|----|--------|-------------------|
| 179 | 0102 | B5 |        | OR L              |
| 180 | 0103 | 20 | FB     | JR NZ,DT          |
| 181 | 0105 | D9 |        | EXX               |
| 182 | 0106 | 08 |        | EX AF,AF          |
| 183 | 0107 | C9 |        | RET               |
| 184 | 0108 | F5 | DISANG | PUSH AF           |
| 185 | 0109 | C5 |        | PUSH BC           |
| 186 | 010A | E5 |        | PUSH HL           |
| 187 | 010B | D5 |        | PUSH DE           |
| 188 | 010C | FE | 01     | CP 01H            |
| 189 | 010E | CC | 1B 01  | CALL Z,ADD        |
| 190 | 0111 | FE | 02     | CP 02H            |
| 191 | 0113 | CC | 32 01  | CALL Z,SUB        |
| 192 | 0116 | D1 |        | POP DE            |
| 193 | 0117 | E1 |        | POP HL            |
| 194 | 0118 | C1 |        | POP BC            |
| 195 | 0119 | F1 |        | POP AF            |
| 196 | 011A | C9 |        | RET               |
| 197 | 011B | 2A | 00 29  | ADD LD HL,(DALOG) |
| 198 | 011E | 11 | 12 00  | LD DE,0012H       |
| 199 | 0121 | 19 |        | ADD HL,DE         |
| 200 | 0122 | 22 | 00 2A  | LD (CBUF),HL      |
| 201 | 0125 | 22 | 00 29  | LD (DALOG),HL     |
| 202 | 0128 | CD | 4B 01  | CALL HTD          |
| 203 | 012B | CD | 6D 01  | CALL DISF         |
| 204 | 012E | CD | 2D 02  | CALL ANGLE        |
| 205 | 0131 | C9 |        | RET               |
| 206 | 0132 | AF | SUB    | XOR A             |
| 207 | 0133 | 2A | 00 29  | LD HL,(DALOG)     |
| 208 | 0136 | 11 | 12 00  | LD DE,0012H       |

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

|     |      |          |                 |
|-----|------|----------|-----------------|
| 209 | 0139 | ED 52    | SBC HL,DE       |
| 210 | 013B | 22 00 2A | LD (CBUF),HL    |
| 211 | 013E | 22 00 29 | LD (DALOG),HL   |
| 212 | 0141 | CD 4B 01 | CALL HTD        |
| 213 | 0144 | CD 6D 01 | CALL DISF       |
| 214 | 0147 | CD 2D 02 | CALL ANGLE      |
| 215 | 014A | C9       | RET             |
| 216 | 014B | AF       | HTD XOR A       |
| 217 | 014C | 21 05 2A | LD HL,CBUF+5    |
| 218 | 014F | 06 03    | LD B,03H        |
| 219 | 0151 | 2B       | HTD1 DEC HL     |
| 220 | 0152 | 77       | LD (HL),A       |
| 221 | 0153 | 10 FC    | DJNZ HTD1       |
| 222 | 0155 | 0E 10    | LD C,10H        |
| 223 | 0157 | 21 00 2A | HTD2 LD HL,CBUF |
| 224 | 015A | CB 16    | RL (HL)         |
| 225 | 015C | 23       | INC HL          |
| 226 | 015D | CB 16    | RL (HL)         |
| 227 | 015F | 23       | INC HL          |
| 228 | 0160 | 06 03    | LD B,03H        |
| 229 | 0162 | 7E       | HTD3 LD A,(HL)  |
| 230 | 0163 | 8F       | ADC A,A         |
| 231 | 0164 | 27       | DAA             |
| 232 | 0165 | 77       | LD (HL),A       |
| 233 | 0166 | 23       | INC HL          |
| 234 | 0167 | 10 F9    | DJNZ HTD3       |
| 235 | 0169 | 0D       | DEC C           |
| 236 | 016A | 20 EB    | JR NZ,HTD2      |
| 237 | 016C | C9       | RET             |
| 238 | 016D | D5       | DISF PUSH DE    |

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษานั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

|     |      |          |      |               |
|-----|------|----------|------|---------------|
| 239 | 016E | B5       |      | PUSH HL       |
| 240 | 016F | C5       |      | PUSH BC       |
| 241 | 0170 | 11 0B 2B |      | LD DE,2B0BH   |
| 242 | 0173 | 0E 02    |      | LD C,02H      |
| 243 | 0175 | 21 03 2A |      | LD HL,CBUF+3  |
| 244 | 0178 | 7E       | BCD1 | LD A,(HL)     |
| 245 | 0179 | 47       |      | LD B,A        |
| 246 | 017A | 1F       |      | RRA           |
| 247 | 017B | 1F       |      | RRA           |
| 248 | 017C | 1F       |      | RRA           |
| 249 | 017D | 1F       |      | RRA           |
| 250 | 017E | E6 0F    |      | AND 00001111B |
| 251 | 0180 | C6 30    |      | ADD A,30H     |
| 252 | 0182 | 12       |      | LD (DE),A     |
| 253 | 0183 | 78       |      | LD A,B        |
| 254 | 0184 | E6 0F    |      | AND 00001111B |
| 255 | 0186 | 13       |      | INC DE        |
| 256 | 0187 | C6 30    |      | ADD A,30H     |
| 257 | 0189 | 12       |      | LD (DE),A     |
| 258 | 018A | 13       |      | INC DE        |
| 259 | 018B | 21 02 2A |      | LD HL,CBUF+2  |
| 260 | 018E | 7E       |      | LD A,(HL)     |
| 261 | 018F | 47       |      | LD B,A        |
| 262 | 0190 | 1F       |      | RRA           |
| 263 | 0191 | 1F       |      | RRA           |
| 264 | 0192 | 1F       |      | RRA           |
| 265 | 0193 | 1F       |      | RRA           |
| 266 | 0194 | E6 0F    |      | AND 00001111B |
| 267 | 0196 | C6 30    |      | ADD A,30H     |
| 268 | 0198 | 12       |      | LD (DE),A     |

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

269 0199 13          INC DE
270 019A 3E 2E      LD A,2EH
271 019C 12          LD (DE),A
272 019D 78          LD A,B
273 019E E6 0F      AND 00001111B
274 01A0 13          INC DE
275 01A1 C6 30      ADD A,30H
276 01A3 12          LD (DE),A
277 01A4 C1          POP BC
278 01A5 E1          POP HL
279 01A6 D1          POP DE
280 01A7 C9          RET
281                  ;*****
282                  ;** PROGRAM TEST LCD DISPLAY FORWARD **
283                  ;*****
284 01A8 CD 4D 03 DIS1 CALL INTLCD
285 01AB 21 5D 02      LD HL,TAB0
286 01AE 3E 01          LD A,1
287 01B0 CD 74 03      CALL WRLINE
288 01B3 3E 02          LD A,2
289 01B5 CD 74 03      CALL WRLINE
290 01B8 CD 4D 03      CALL INTLCD
291 01BB 21 7D 02      LD HL,TAB1
292 01BE 3E 01          LD A,1
293 01C0 CD 74 03      CALL WRLINE
294 01C3 3E 02          LD A,2
295 01C5 CD 74 03      CALL WRLINE
296 01C8 C9          RET
297 01C9 CD 4D 03 DIS2 CALL INTLCD
298 01CC 21 9D 02      LD HL,TAB2

```



|     |      |          |                    |
|-----|------|----------|--------------------|
| 329 | 0218 | CD 74 03 | CALL WRLINE        |
| 330 | 021B | C9       | RET                |
| 331 | 021C | CD 4D 03 | DISRE CALL INITLCD |
| 332 | 021F | 21 2D 03 | LD HL,TAB7         |
| 333 | 0222 | 3E 01    | LD A,1             |
| 334 | 0224 | CD 74 03 | CALL WRLINE        |
| 335 | 0227 | 3E 02    | LD A,2             |
| 336 | 0229 | CD 74 03 | CALL WRLINE        |
| 337 | 022C | C9       | RET                |
| 338 | 022D | E5       | ANGLE PUSH HL      |
| 339 | 022E | C5       | PUSH BC            |
| 340 | 022F | 21 0B 2B | LD HL,2B0BH        |
| 341 | 0232 | 3E 4B    | LD A,4BH           |
| 342 | 0234 | CD A5 03 | CALL GOTO          |
| 343 | 0237 | 06 05    | LD B,05H           |
| 344 | 0239 | 56       | ANG1 LD D,(HL)     |
| 345 | 023A | C5       | PUSH BC            |
| 346 | 023B | CD B5 03 | CALL WRBYTE        |
| 347 | 023E | C1       | POP BC             |
| 348 | 023F | 23       | INC HL             |
| 349 | 0240 | 10 F7    | DJNZ ANG1          |
| 350 | 0242 | C1       | POP BC             |
| 351 | 0243 | E1       | POP HL             |
| 352 | 0244 | C9       | RET                |
| 353 | 0245 | E5       | ANGLE2 PUSH HL     |
| 354 | 0246 | C5       | PUSH BC            |
| 355 | 0247 | 21 0B 2B | LD HL,2B0BH        |
| 356 | 024A | 3E 0B    | LD A,0BH           |
| 357 | 024C | CD A5 03 | CALL GOTO          |
| 358 | 024F | 06 05    | LD B,05H           |

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

359 0251 56          ANG2  LD D,(HL)
360 0252 C5          PUSH BC
361 0253 CD B5 03     CALL WRBYTE
362 0256 C1          POP BC
363 0257 23          INC HL
364 0258 10 F7       DJNZ ANG2
365 025A C1          POP BC
366 025B E1          POP HL
367 025C C9          RET
368 025D 41 55 54 4F 4D
TAB0 DB 41H,55H,54H,4FH,4DH,41H,54H,49H,43H,41H,4EH,54H,45H,4EH,4EH,41H
      0262 41 54 49 43 41 H,49H,43H,41H,4EH,54H,45H,4EH,4EH,41H
      0267 4E 54 45 4E 4E
      026C 41
369 026D 20 4B 49 4E 47
      DB 20H,4BH,49H,4EH,47H,20H,4DH,4FH,4EH,47H,4BH,55H,54H,27H,53H,20H
      0272 20 4D 4F 4E 47
      0277 4B 55 54 27 53
      027C 20
370 027D 20 49 4E 53 54
TAB1 DB 20H,49H,4EH,53H,54H,49H,54H,55H,54H,45H,20H,20H,20H,4FH,46H,20H
      0282 49 54 55 54 45
      0287 20 20 20 4F 46
      028C 20
371 028D 20 20 20 54 45
      DB 20H,20H,20H,54H,45H,43H,48H,4EH,4FH,4CH,4FH,47H,59H,20H,20H,20H
      0292 43 48 4E 4F 4C
      0297 4F 47 59 20 20
      029C 20

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

372 029D 53 43 41 4E 49

TAB2 DB 53H,43H,41H,4EH,49H,4EH,47H,20H,50H,4FH,53H,49H,54H,49H,4FH,4EH

02A2 4E 47 20 50 4F

02A7 53 49 54 49 4F

02AC 4E

373 02AD 2A 46 46 20 41

DB 2AH,46H,46H,20H,41H,4EH,47H,4CH,45H,20H,20H,20H,20H,20H,20H,20H

02B2 4E 47 4C 45 20

02B7 20 20 20 20 20

02BC 20

374 02BD 53 43 41 4E 49

TAB3 DB 53H,43H,41H,4EH,49H,4EH,47H,20H,50H,4FH,53H,49H,54H,49H,4FH,4EH

02C2 4E 47 20 50 4F

02C7 53 49 54 49 4F

02CC 4E

375 02CD 2A 52 45 57 20

DB 2AH,52H,45H,57H,20H,41H,4EH,47H,4CH,45H,20H,20H,20H,20H,20H,20H

02D2 41 4E 47 4C 45

02D7 20 20 20 20 20

02DC 20

376 02DD 53 54 41 4E 44

TAB4 DB 53H,54H,41H,4EH,44H,42H,59H,41H,4EH,47H,20H,20H,20H,20H,20H,20H

02E2 42 59 41 4E 47

02E7 20 20 20 20 20

02EC 20

377 02ED 43 48 41 4E 47

TAB5 DB 43H,48H,41H,4EH,47H,45H,20H,20H,43H,48H,41H,4EH,4EH,45H,4CH,53H

02F2 45 20 20 43 48

02F7 41 4E 4E 45 4C

02FC 53

เอกสารนี้เป็นเอกสารที่งานวิจัยสำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

378 02FD 52 45 53 54 41

DB 52H,45H,53H,54H,41H,52H,53H,20H,50H,52H,4FH,47H,52H,41H,4DH,52H

0302 52 53 20 50 52

0307 4F 47 52 41 4D

030C 52

379 030D 20 50 4F 57 45

TAB6 DB 20H,50H,4FH,57H,45H,52H,20H,20H,54H,56H,20H,20H,4FH,46H,46H,20H

0312 52 20 20 54 56

0317 20 20 4F 46 46

031C 20

380 031D 52 45 53 54 41

DB 52H,45H,53H,54H,41H,52H,53H,20H,50H,52H,4FH,47H,52H,41H,4DH,52H

0322 52 53 20 50 52

0327 4F 47 52 41 4D

032C 52

381 032D 20 52 45 54 55

TAB7 DB 20H,52H,45H,54H,55H,52H,4EH,20H,54H,4FH,20H,5AH,45H,52H,4FH,20H

0332 52 4E 20 54 4F

0337 20 5A 45 52 4F

033C 20

382 033D 2A 52 45 57 20

DB 2AH,52H,45H,57H,20H,41H,4EH,47H,4CH,45H,20H,20H,20H,20H,20H,20H

0342 41 4E 47 4C 45

0347 20 20 20 20 20

034C 20

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

383      ,*****
384      ;**          INITIAL LCD          **
385      ;** PB0-PB7 :PIN D0-D7 (DATA READ/WRITE LCD) **
386      ;** PA4      :PIN RS   (REGISTER SELECTION)    **
387      ;** PA5      :PIN R/W  (READ/WRITW)           **
388      ;** PA6      :PIN E    (ENABLE SIGNAL PULSE)   **
389      ,*****
390 034D 3E 00      INITLCD LD A,0
391 034F D3 E2      OUT (PORTC),A
392 0351 3E 38      LD A,00111000B ;Funtion set 38h
393                                     ;DL=1 8 bit,N=1 1/16 duty,F=0 5x7
394 0353 D3 E1      OUT (PORTB),A
395 0355 CD 96 03    CALL EPLUSE ;enable signal pluse
396 0358 CD FB 00    CALL DELAYT ;DELAY > 4.1 ms
397 035B 3E 0C      LD A,00001100B ;display on/off control
398                                     ;D=1 off,C=1 ursor on,B=1 blink
399 035D D3 E1      OUT (PORTB),A
400 035F CD 96 03    CALL EPLUSE
401 0362 3E 19      LD A,00011001B ;entry mode set
402                                     ;I/D=1 increment,S=0 right
403 0364 D3 E1      OUT (PORTB),A
404 0366 CD 96 03    CALL EPLUSE
405 0369 3E 01      LD A,00000001B
406 036B D3 E1      OUT (PORTB),A
407 036D CD 96 03    CALL EPLUSE
408 0370 CD FB 00    CALL DELAYT
409 0373 C9          RET

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

410      ;*****
411      ;**   WRIT LINE 16 CHAR      **
412      ;**   INPUT (HL)=DATA      **
413      ;**   INPUT  A =LINE      **
414      ;*****
415 0374 FE 01      WRLINE CP 1
416 0376 28 05      JR Z,WRL1
417 0378 FE 02      CP 2
418 037A 28 08      JR Z,WRL2
419 037C C9      RET
420 037D 3E 00      WRL1 LD A,00H
421 037F CD A5 03      CALL GOTO
422 0382 18 05      JR WRLM
423 0384 3E 40      WRL2 LD A,40H
424 0386 CD A5 03      CALL GOTO
425 0389 06 10      WRLM LD B,16
426 038B 56      WRL LD D,(HL)
427 038C C5      PUSH BC
428 038D CD B5 03      CALL WRBYTE
429 0390 C1      POP BC
430 0391 23      INC HL
431 0392 3D      DEC A
432 0393 10 F6      DJNZ WRL
433 0395 C9      RET
434      ;*****
435      ;**   ENABLE PLUSE SUB-      **
436      ;*****
437- 0396 DB E2      EPLUSE IN A,(PORTC)
438 0398 CB D7      SET 2,A
439 039A D3 E2      OUT (PORTC),A ;enable bit 6=1

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้ทำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

440 039C 06 00          LD B,00H
441 039E 10 FE      EP1    DJNZ EP1
442 03A0 CB 97          RES 2,A
443 03A2 D3 E2          OUT (PORTC),A
444 03A4 C9            RET
445                      ;*****
446                      ;**      GOTO POSITION      **
447                      ;**      INPUT REG D=DATA    **
448                      ;*****
449 03A5 C5            GOTO  PUSH BC
450 03A6 CB FF          SET 7,A      ;set dd ram
451 03A8 D3 E1          OUT (PORTB),A
452 03AA CD FB 00       CALL DELAYT
453 03AD AF            XOR A
454 03AE D3 E2          OUT (PORTC),A ;set RS=0,R/W=0
455 03B0 CD 96 03       CALL EPLUSE
456 03B3 C1            POP BC
457 03B4 C9            RET
458                      ;*****
459                      ;**      WRITE DATA SUB    **
460                      ;**      INPUT REG D=DATA    **
461                      ;*****
462 03B5 3E 01      WRBYTELD A,00000001B ;data write
463 03B7 D3 E2          OUT (PORTC),A
464 03B9 7A            LD A,D      ;data byte
465 03BA D3 E1          OUT (PORTB),A
466 03BC CD FB 00       CALL DELAYT
467 03BF CD 96 03       CALL EPLUSE
468 03C2 C9            RET

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## บทที่ 4

### ผลการทดลองและสรุปผลการทดลอง

ในบทนี้จะกล่าวถึงผลการทดลองของแต่ละส่วนในโครงงานนี้และสรุปผลการทดลองพร้อมทั้งปัญหาที่พบในการทดลองและข้อเสนอแนะต่างๆ

#### 4.1 ผลการทดลอง

##### 4.1.1 ผลการทดลองการตรวจวัดแรงดัน AGC

ในภาค IF AMP ของเครื่องรับโทรทัศน์จะมีแรงดันอยู่ชนิดหนึ่งซึ่งคอยควบคุมอัตราการขยายของสัญญาณโดยอัตโนมัติซึ่งเรียกว่า AGC (Automatic Gain Control) ถ้าสัญญาณขาเข้ามาน้อยเกินไป AGC จะผลิตแรงดันควบคุมให้มากขึ้นเพื่อเพิ่มอัตราการขยายให้มากขึ้น หรือถ้าสัญญาณขาเข้ามามากเกินไป AGC จะผลิตแรงดันควบคุมให้น้อยลงเพื่อลดอัตราการขยายของภาค IF AMP ทำให้ Amplitude ทางด้าน Output ของสัญญาณคงที่

ดังนั้นในการตรวจวัดแรงดัน AGC จะได้ว่า ในช่วงที่สัญญาณภาพมีความคมชัดจนมากที่สุด คือช่วงที่แรงดัน AGC จะมีค่าน้อยที่สุด ในทางตรงกันข้าม ถ้าในช่วงที่สัญญาณภาพมีความคมชัดน้อยมาก คือช่วงที่แรงดัน AGC จะมีค่ามากที่สุดเช่นกัน



รูปที่ 4.1 แสดงรูปสัญญาณภาพที่มีความคมชัดจนที่สุด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.2 แสดงรูปสัญญาณภาพที่มีความชัดเจนน้อย

#### 4.1.2 ผลการทดลองวงจร Analog To Digital Converter

วงจร Analog To Digital Converter (ADC) ทำหน้าที่ แปลงแรงดัน DC ทางอินพุตไปเป็น Digital ออกทางเอาต์พุตโดยมีขนาด 8 bit เป็นเลขฐาน 16 ดังนั้นผลแรงดัน AGC ที่ผ่านวงจร Analog To Digital Converter จะถูกเปลี่ยนเป็นดิจิตอลโดย IC ADC0809 ดังตารางที่ 4.1 แล้วสัญญาณดิจิตอลนี้จะถูกส่งไปยังภาคประมวลผลซึ่งใช้ Z80 CP-A ในการประมวลผลต่อไป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 4.1 แสดงผลการแปลงแรงดัน AGC เป็นดิจิตอลขนาด 8 bit

| AGC (V) | AGC (Digital) |
|---------|---------------|
| 0.0     | 00 H          |
| 0.5     | 19 H          |
| 1.0     | 33 H          |
| 1.5     | 4C H          |
| 2.0     | 66 H          |
| 2.5     | 7F H          |
| 3.0     | 99 H          |
| 3.5     | B2 H          |
| 4.0     | CC H          |
| 4.5     | B5 H          |
| 5.0     | FF H          |

4.1.3 ผลการทดลองภาคประมวลผล

ในปริญญานิพนธ์ฉบับนี้ได้ใช้ไมโครโปรเซสเซอร์เบอร์ Z80 มาต่อร่วมกับ ADC 0809, EPROM เบอร์ 2764, RAM เบอร์ 6264 และ 8255 โดยมี EPROM ทำหน้าที่เก็บโปรแกรม มอนิเตอร์ให้ Z80 โดยมี ADC 0809 เป็นอินพุตพอร์ท ข้อมูลจากอินพุตพอร์ทจะถูกอ่านเข้ามา เก็บไว้ที่ RAM เบอร์ 6264 ส่วนการส่งข้อมูลออกไปควบคุมเพื่อหมุนมอเตอร์จะส่งออกที่พอร์ท ของ 8255 การทำงานของภาคประมวลผลจะขึ้นอยู่กับ การเขียนโปรแกรมการทำงานนั่นเอง

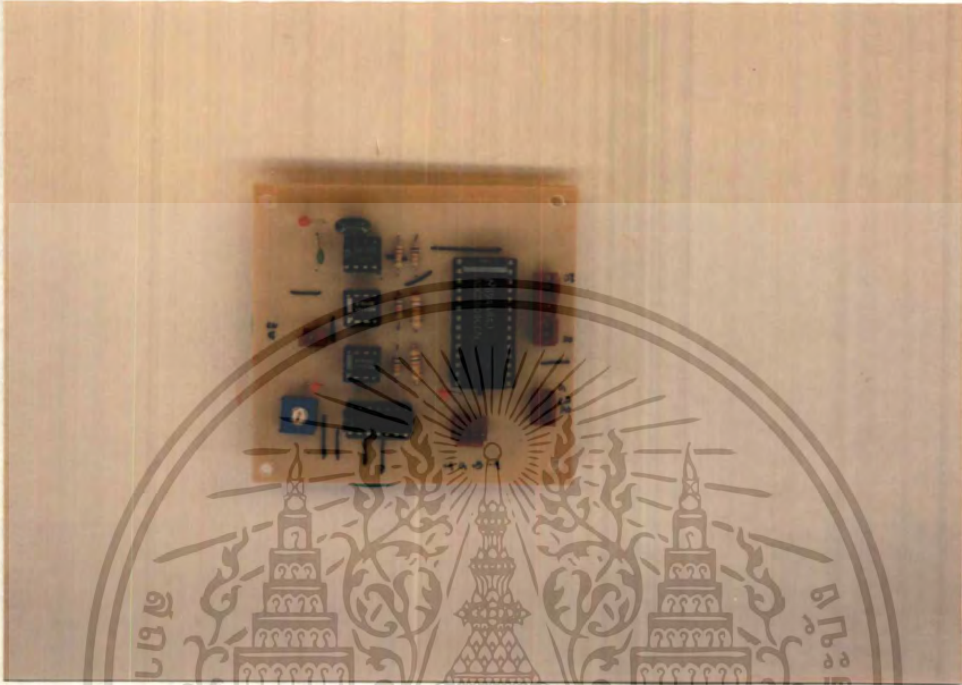
4.1.4 ผลการทดลองภาค SENSOR ต่างๆ

SENSOR ที่ใช้ในโครงงานนี้โดยมีการ SENSOR 3 อย่างคือ การเปลี่ยนช่อง โทรทัศน์ การเปิด-ปิดเครื่องรับโทรทัศน์ และ SENSOR ตำแหน่งเริ่มต้นของมอเตอร์โดยการ CHECK เปลี่ยนช่องโทรทัศน์ใช้สัญญาณจากขา 6 ของ IC MPU ของเครื่องรับโทรทัศน์ การ CHECK การเปิด - ปิด เครื่องรับโทรทัศน์ ใช้สัญญาณจากขา 9 ของ IC BAND SW ของเครื่องรับโทรทัศน์ และตำแหน่งเริ่มต้นของมอเตอร์ใช้ลิ้มิตสวิชในการ CHECK จากการทดลองได้

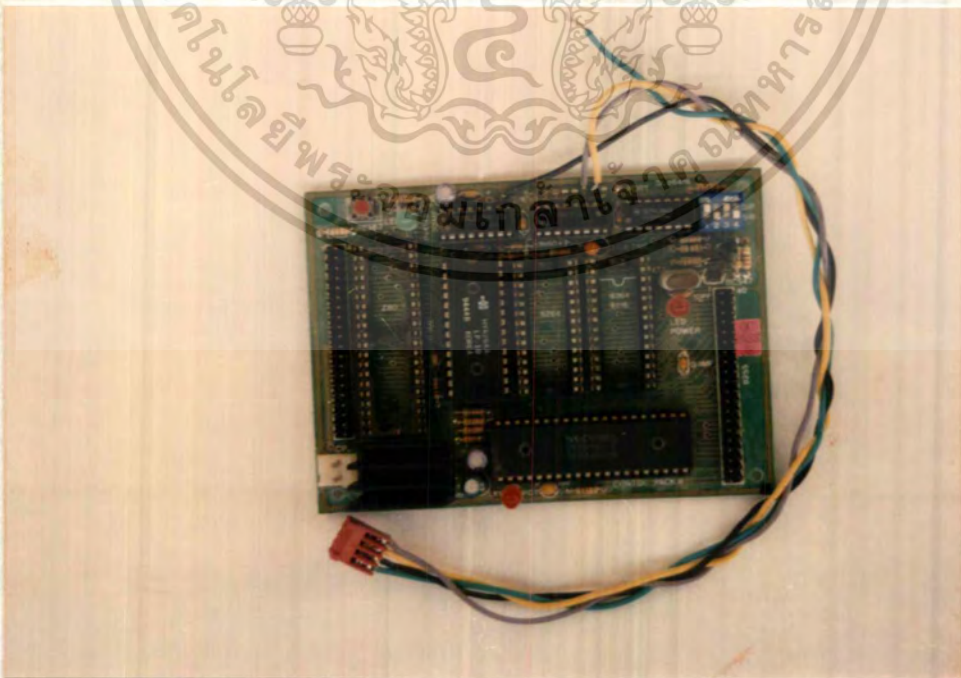
ผลดีพอสมควร

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

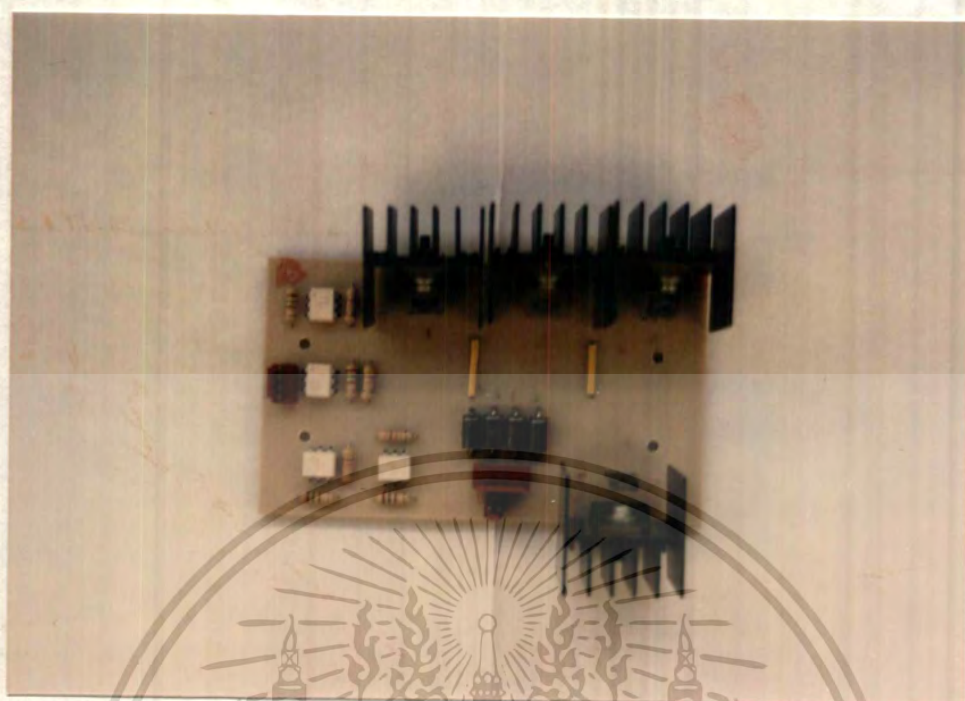
#### 4.1.5 แสดงรูปภาพของแต่ละภาค



4.3 แสดงรูปภาพ ภาค AGC CONVERTER



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับใช้ภายในเท่านั้น ไม่ควรเผยแพร่สู่สาธารณะโดยไม่ได้รับอนุญาต  
**4.4 แสดงรูปภาพ ภาค การประมวลผล ( Z-80 CPU )** ใช้ประโยชน์ด้านการค้า  
 ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

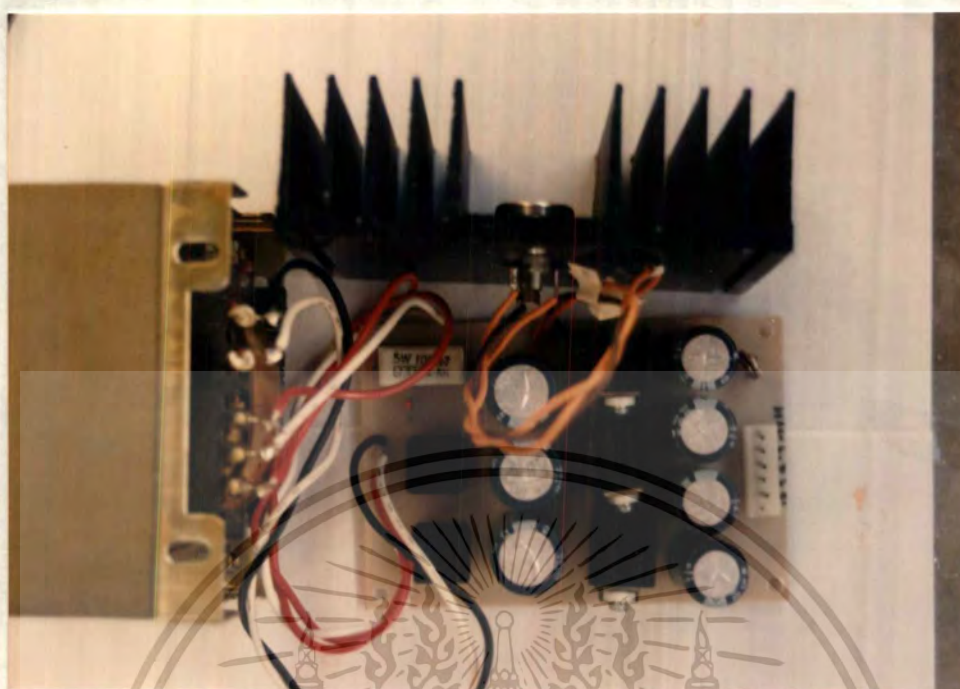


4.5 แสดงรูปภาพ ภาค DRIVER STEPPING MOTOR



4.6 แสดงรูปภาพ ภาค SENSOR

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



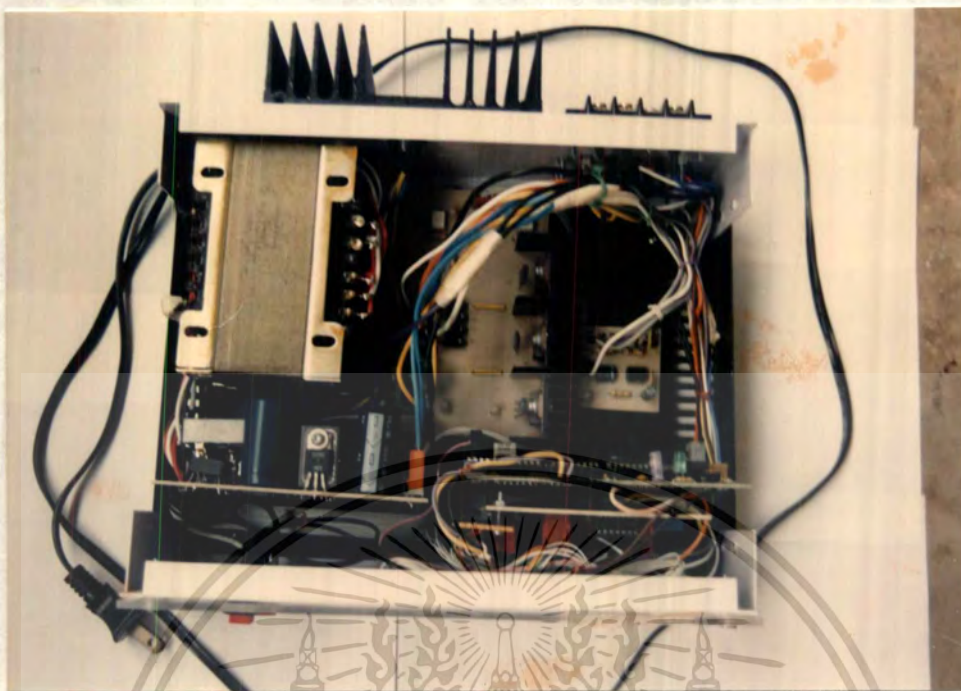
4.7 แสดงรูปภาพ ภาค POWER SUPPLY

#### 4.1.6 ผลการทดลองเมื่อนำทุกส่วนมาประกอบลงกล่อง

จากการที่ทดลองในการทำงานของภาควงจรต่างๆ ได้นำวงจรดังกล่าวมาประกอบเข้าด้วยกันแล้วทำการนำชิ้นงานประกอบลงกล่องดังแสดงในรูปที่ 4.8 แล้วทำการทดสอบกับโปรแกรมที่เขียนไว้ตามวัตถุประสงค์ของโครงการที่ตั้งไว้ โดยการทดสอบ ให้ CPU ทำงานตามโปรแกรมโดยในการทดลองนั้นได้จ่ายไฟ DC 5 Volt ให้แก่ Input AGC แทนสัญญาณ AGC จากเครื่องรับโทรทัศน์ เมื่อ CPU เริ่มทำงานโดยสั่งให้ Stepping Motor หมุนไป 1 Step หรือ 1.8 องศา พร้อมทั้งเก็บค่า AGC ไว้ และทำไปจนถึงประมาณ 180 องศา แล้วทำการจ่ายแรงดัน AGC เท่ากับ 0 Volt (วงจร AGC จะมีค่าสูงสุด) ต่อจากนั้นให้ AGC มีค่าเท่ากับ 5 Volt อีกครั้งปรากฏว่า CPU สั่งให้ Stepping Motor หยุดหมุนที่ประมาณ 180 องศาพร้อมกับการทำงานในเงื่อนไขอื่นๆต่อไป เช่น การหมุน ซ้าย-ขวา, การเปลี่ยนช่องความถี่ หรือการปิด POWER ของเครื่องรับโทรทัศน์

จากนั้นได้นำสัญญาณจากเครื่องรับโทรทัศน์ ซึ่งประกอบด้วย AGC, IF, POWER และ CS มาต่อเข้ากับเครื่องแล้วให้ทำงานตามโปรแกรมที่กำหนดไว้ ปรากฏว่าได้ผลตามที่ต้องการไว้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.8 แสดงการประกอบอุปกรณ์ทดลอง

## 4.2 สรุปผลการทดลอง

เนื่องจากโครงงานนี้ได้นำ ไมโครโปรเซสเซอร์มาใช้ในการควบคุมให้เป็นแบบระบบควบคุมอัตโนมัติจึงทำให้ง่ายต่อการใช้งานและง่ายต่อการเปลี่ยนแปลงหรือแก้ไข โดยการเปลี่ยนแปลงหรือแก้ไขโปรแกรมให้สอดคล้องกับฮาร์ดแวร์ ซึ่งจากโครงงานนี้เมื่อทำการหมุนหาคะลินของสัญญาณโทรทัศน์ที่ชัดเจน ปรากฏผลได้ว่า เครื่องสามารถที่จะค้นหาสัญญาณโทรทัศน์ได้ชัดเจน แต่ในบางครั้งสัญญาณที่ได้อาจจะ ไม่ชัดเจนอยู่บ้าง แต่เมื่อได้ทำการทดลองหลายๆครั้งแล้ว ปรากฏว่าส่วนมากจะรับได้ชัดเจนคือ

## 4.3 ปัญหาที่พบในการทดลอง

4.3.1 ในบางครั้งสัญญาณ AGC ที่จุดของสัญญาณภาพโทรทัศน์ที่ชัดเจน อาจไม่ใช่จุดที่สัญญาณ AGC มีค่าต่ำสุด จึงทำให้การหาตำแหน่งผิดพลาดได้

4.3.2 ในบางครั้งสัญญาณ AGC ที่จุดของสัญญาณภาพโทรทัศน์ที่ชัดเจนไม่ใช่จุดที่สัญญาณ AGC ต่ำที่สุด อาจทำให้การหาทิศทางของสัญญาณผิดพลาดได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.3.3 จากการที่มีการสะท้อนของสัญญาณโทรทัศน์ กับอาคารต่างซึ่งทำให้ทิศทางของสัญญาณเปลี่ยนไปจึงทำให้ทิศทางของสายอากาศไม่ตรงกับสถานีส่ง

4.3.4 ปัญหาความร้อนของภาค Power Supply ของส่วนที่จ่ายให้กับ Stepping Motor เพราะเนื่องจาก Stepping Motor ต้องการกระแสสูงถึง 5 แอมป์ จึงทำให้เกิดความร้อน

#### 4.4 ข้อเสนอแนะ

4.4.1 ควรใช้ Motor ที่มีโวลท์สูงขึ้นกระแสในขดลวดของ Motor จึงจะน้อยลงทำให้ขั้วพลาขมีความร้อนน้อยลงเนื่องจาก Motor กินกระแสต่ำ



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## บทที่ 5

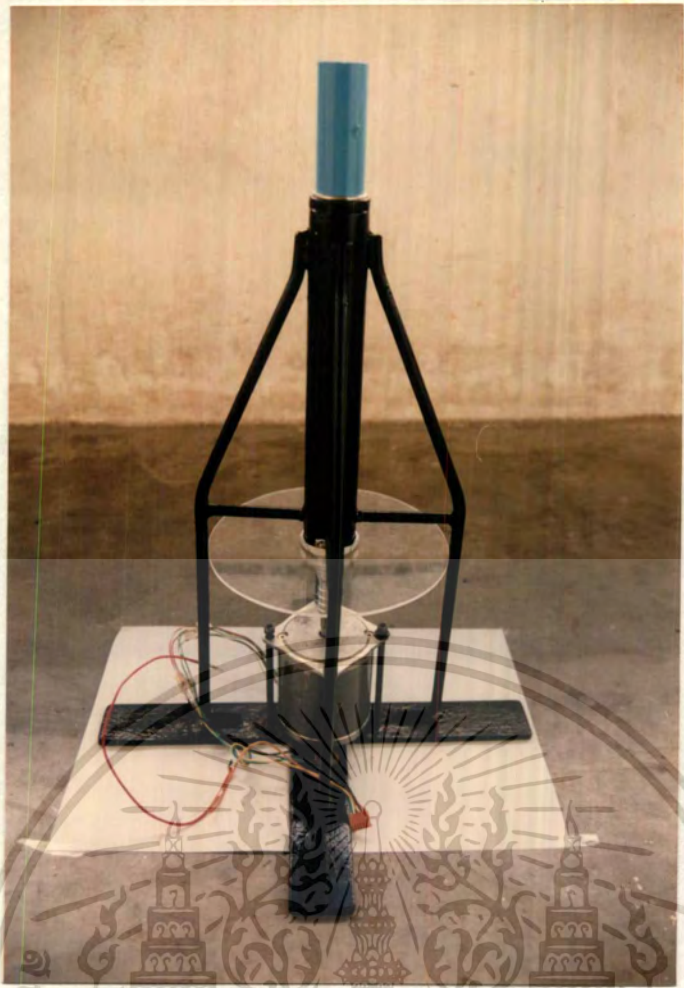
### การใช้งานของเครื่อง สายอากาศปรับทิศทางอัตโนมัติ

#### วิธีการใช้งาน

ทำการต่อสายสัญญาณ AGC, IF, POWER และ CS เข้าที่โทรทัศน์ และ ต่อสาย Motor เข้ากับ Motor ที่ส่วนฐานหมุนสายอากาศ เสร็จแล้วทำการเปิด SW POWER ที่เครื่องสายอากาศปรับทิศทางอัตโนมัติ เครื่องจะสั่งให้ Motor หมุนไปยังจุดเริ่มต้นที่ 0 องศา (ถ้าตำแหน่งของสายอากาศไม่ได้อยู่ในจุดเริ่มต้นที่ 0 องศา) แล้วรอการเปิด Switch Power ของเครื่องรับโทรทัศน์ เมื่อเปิด Switch Power เครื่องรับโทรทัศน์ Motor จะทำการหมุนสายอากาศไป 1 รอบ หรือ 360 องศา พร้อมทั้งเก็บค่าแรงดัน AGC ไว้ทุก Step การหมุน เมื่อหมุนครบ 1 รอบแล้ว Motor ก็จะทำาการหมุนสายอากาศในทิศทางตรงกันข้าม (เพื่อป้องกันสายพันรอบเสาสายอากาศ) ไปยังตำแหน่งที่ชัดเจนที่สุดของภาพแล้วหยุดหมุน เมื่อมีการเปลี่ยนช่องความถี่ Motor ก็จะทำาการหมุนสายอากาศไปยังตำแหน่งที่ชัดเจนของภาพมากที่สุด แต่ถ้ามีการปิด Switch Power ของโทรทัศน์ Motor ก็จะทำาการหมุนสายอากาศไปยังจุดเริ่มต้นที่ 0 องศาใหม่ การประกอบส่วนต่างๆทั้งหมดของชุดเครื่องสายอากาศปรับทิศทางอัตโนมัติสามารถแสดงได้ในรูปที่ 5.1



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาค้นคว้าเป็นเอกสารเท่านั้นไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 5.2 แสดงการติดตั้งฐานเสาอากาศ



เอกสารนี้เป็นเอกสารที่สงวนไว้รูปที่ 5.3 แสดงรูปค่านหน้ากล่องของเครื่อง  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

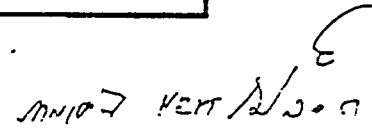
## การต่อสัญญาณจากเครื่องรับโทรทัศน์มายังเครื่อง สายอากาศปรับทิศทางอัตโนมัติ

จากรูปวงจรเครื่องรับโทรทัศน์สัญญาณที่จะนำมาต่อเข้ายังเครื่อง สายอากาศปรับทิศทางอัตโนมัติ คือ

- สัญญาณ AGC ต่อจากขา 4 ของ IC 101 ( VIF/SIF ) จุดจุดการต่อจากวงจรเครื่องรับโทรทัศน์
- สัญญาณ GND ต่อจากขา 7 ของ IC 101 ( VIF/SIF ) จุดจุดการต่อจากวงจรเครื่องรับโทรทัศน์
- สัญญาณการ ON/OFF TV ต่อจากขา 9 ของ IC 1103 ( BAND SW ) จุดจุดการต่อจากวงจรเครื่องรับโทรทัศน์
- สัญญาณการเปลี่ยน CH ต่อจากขา 6 ของ IC 1102 ( MPU ) จุดจุดการต่อจากวงจรเครื่องรับโทรทัศน์



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

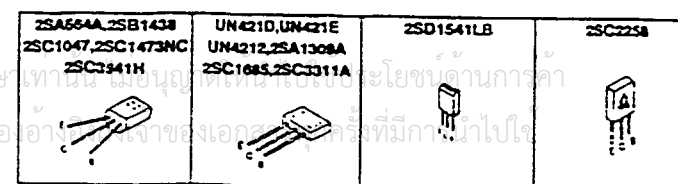


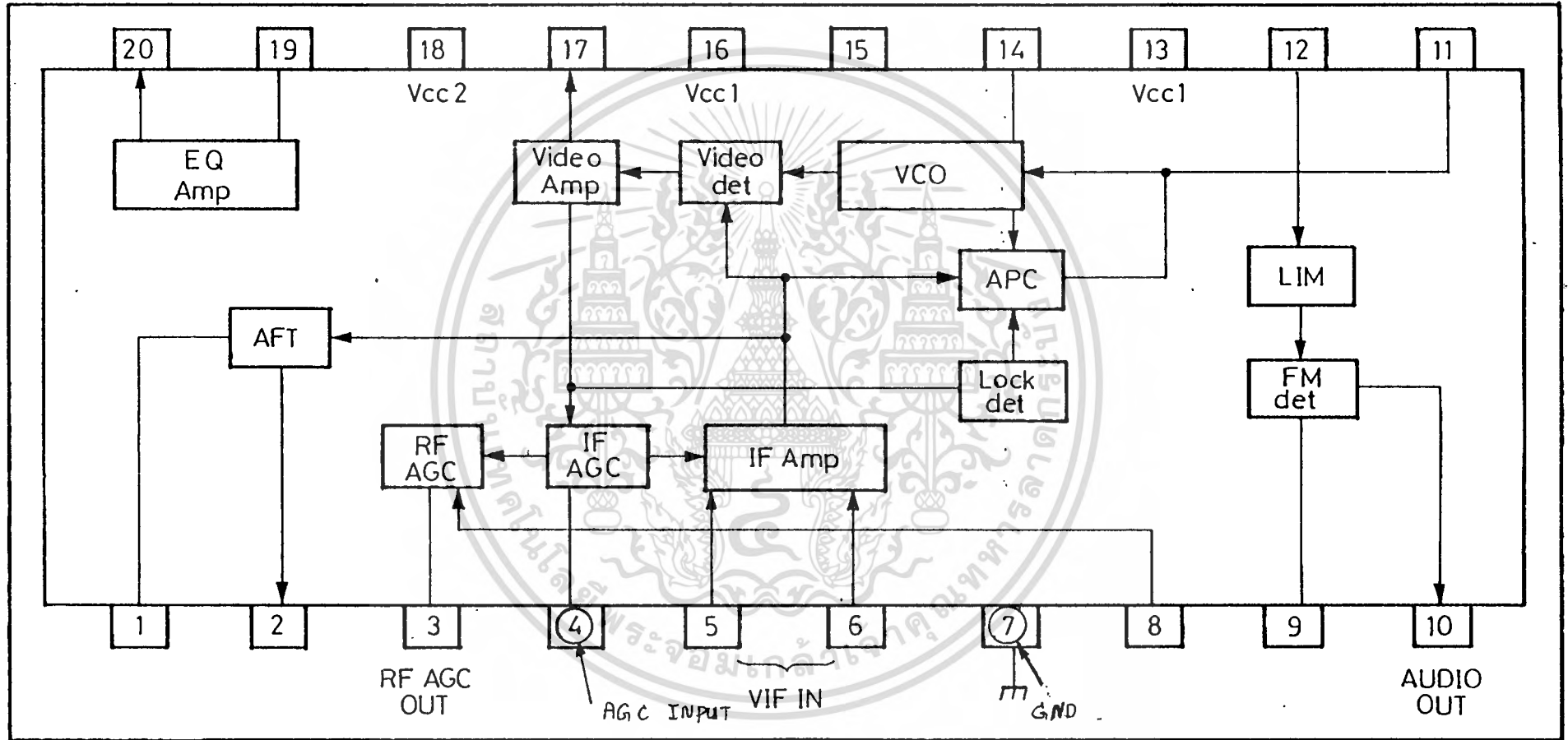
|             | TC-14L3    | TC-14Z2   | TC-14Z3                   |
|-------------|------------|-----------|---------------------------|
| AV Terminal | TJB4G602   | TJB4G601  | TJB4G602                  |
| C 621       | 50V 18     | -         | 50V 18                    |
| C 604       | 50V 470    | -         | 50V 470                   |
| C3118       | 16V 47     | -         | 16V 47                    |
| C3107       | 16V 470    | -         | 16V 470                   |
| C3101       | 50V 3.3    | -         | 50V 3.3                   |
| C 302       | 50V 0.01   | -         | 50V 0.01                  |
| C 608       | 50V 0.01   | -         | 50V 0.01                  |
| C3108       | 50V 470    | -         | 50V 470                   |
| C31C2       | 50V 2200   | -         | 50V 2200                  |
| J 190       | -          | T3A205016 | -                         |
| JX2         | -          | T3A205016 | -                         |
| LC302       | ELB5A085   | -         | ELB5A085 or<br>TLK156090E |
| LC602       | TAXSAB4250 | -         | TAXSAB4250                |
| Q 302       | 2SC1685R   | -         | 2SC1685R                  |
| Q3110       | 2SC1685QR  | -         | 2SC1685CR<br>or 2SC3311A  |
| R3120       | 100        | -         | 100                       |
| R305        | 10K        | -         | 10K                       |
| R319        | 10K        | -         | 10K                       |
| R1131       | 10K        | -         | 10K                       |
| R3105       | 10K        | -         | 10K                       |
| R3103       | 100K       | -         | 100K                      |
| R3114       | 12K        | -         | 12K                       |
| R612        | 220        | -         | 220                       |
| R1130       | 12.2K      | -         | 12.2K                     |
| R3112       | 3.3K       | -         | 3.3K                      |
| R3109       | 390        | -         | 390                       |
| R232        | 820        | T3A205016 | T3A205016                 |
| S1102       | EVQQBH06T  | EVQQBH12G | EVQQBH12G                 |
| S1103       | EVQQBH06T  | EVQQBH12G | EVQQBH12G                 |
| S1104       | EVQQBH06T  | EVQQBH12G | EVQQBH12G                 |
| S1106       | EVQQBH06T  | EVQQBH12G | EVQQBH12G                 |
| S1107       | EVQQBH06T  | EVQQBH12G | EVQQBH12G                 |
| S1113       | EVQQBH06T  | EVQQBH12G | EVQQBH12G                 |

## ตารางวัดไฟ IC601

| 27  | 27  | 27  | 27  |
|-----|-----|-----|-----|
| 0   | 27  | 0   | 0   |
| 1   | 28  | 1   | 1   |
| 2   | 29  | 2   | 2   |
| 3   | 30  | 3   | 3   |
| 4   | 31  | 4   | 4   |
| 5   | 32  | 5   | 5   |
| 6   | 33  | 6   | 6   |
| 7   | 34  | 7   | 7   |
| 8   | 35  | 8   | 8   |
| 9   | 36  | 9   | 9   |
| 10  | 37  | 10  | 10  |
| 11  | 38  | 11  | 11  |
| 12  | 39  | 12  | 12  |
| 13  | 40  | 13  | 13  |
| 14  | 41  | 14  | 14  |
| 15  | 42  | 15  | 15  |
| 16  | 43  | 16  | 16  |
| 17  | 44  | 17  | 17  |
| 18  | 45  | 18  | 18  |
| 19  | 46  | 19  | 19  |
| 20  | 47  | 20  | 20  |
| 21  | 48  | 21  | 21  |
| 22  | 49  | 22  | 22  |
| 23  | 50  | 23  | 23  |
| 24  | 51  | 24  | 24  |
| 25  | 52  | 25  | 25  |
| 26  | 53  | 26  | 26  |
| 27  | 54  | 27  | 27  |
| 28  | 55  | 28  | 28  |
| 29  | 56  | 29  | 29  |
| 30  | 57  | 30  | 30  |
| 31  | 58  | 31  | 31  |
| 32  | 59  | 32  | 32  |
| 33  | 60  | 33  | 33  |
| 34  | 61  | 34  | 34  |
| 35  | 62  | 35  | 35  |
| 36  | 63  | 36  | 36  |
| 37  | 64  | 37  | 37  |
| 38  | 65  | 38  | 38  |
| 39  | 66  | 39  | 39  |
| 40  | 67  | 40  | 40  |
| 41  | 68  | 41  | 41  |
| 42  | 69  | 42  | 42  |
| 43  | 70  | 43  | 43  |
| 44  | 71  | 44  | 44  |
| 45  | 72  | 45  | 45  |
| 46  | 73  | 46  | 46  |
| 47  | 74  | 47  | 47  |
| 48  | 75  | 48  | 48  |
| 49  | 76  | 49  | 49  |
| 50  | 77  | 50  | 50  |
| 51  | 78  | 51  | 51  |
| 52  | 79  | 52  | 52  |
| 53  | 80  | 53  | 53  |
| 54  | 81  | 54  | 54  |
| 55  | 82  | 55  | 55  |
| 56  | 83  | 56  | 56  |
| 57  | 84  | 57  | 57  |
| 58  | 85  | 58  | 58  |
| 59  | 86  | 59  | 59  |
| 60  | 87  | 60  | 60  |
| 61  | 88  | 61  | 61  |
| 62  | 89  | 62  | 62  |
| 63  | 90  | 63  | 63  |
| 64  | 91  | 64  | 64  |
| 65  | 92  | 65  | 65  |
| 66  | 93  | 66  | 66  |
| 67  | 94  | 67  | 67  |
| 68  | 95  | 68  | 68  |
| 69  | 96  | 69  | 69  |
| 70  | 97  | 70  | 70  |
| 71  | 98  | 71  | 71  |
| 72  | 99  | 72  | 72  |
| 73  | 100 | 73  | 73  |
| 74  | 101 | 74  | 74  |
| 75  | 102 | 75  | 75  |
| 76  | 103 | 76  | 76  |
| 77  | 104 | 77  | 77  |
| 78  | 105 | 78  | 78  |
| 79  | 106 | 79  | 79  |
| 80  | 107 | 80  | 80  |
| 81  | 108 | 81  | 81  |
| 82  | 109 | 82  | 82  |
| 83  | 110 | 83  | 83  |
| 84  | 111 | 84  | 84  |
| 85  | 112 | 85  | 85  |
| 86  | 113 | 86  | 86  |
| 87  | 114 | 87  | 87  |
| 88  | 115 | 88  | 88  |
| 89  | 116 | 89  | 89  |
| 90  | 117 | 90  | 90  |
| 91  | 118 | 91  | 91  |
| 92  | 119 | 92  | 92  |
| 93  | 120 | 93  | 93  |
| 94  | 121 | 94  | 94  |
| 95  | 122 | 95  | 95  |
| 96  | 123 | 96  | 96  |
| 97  | 124 | 97  | 97  |
| 98  | 125 | 98  | 98  |
| 99  | 126 | 99  | 99  |
| 100 | 127 | 100 | 100 |

## ตำแหน่งชาทรานซิสเตอร์





IC101 (TVSM52034SP)

## หนังสืออ้างอิง

นรินทร์ เนาวประทีป ออปแอมป์ 741 กรุงเทพฯ ฟิสิกส์เซ็นเตอร์, 2531.

นรินทร์ เนาวประทีป การใช้งาน Z80 กรุงเทพฯ ฟิสิกส์เซ็นเตอร์,

สุพันธ์ อมรเชิดชู “สเต็ปเปอร์มอเตอร์” เซมิคอนดักเตอร์อิเล็กทรอนิกส์. (126) :

83-89, เมษายน 2536.

คู่มือเทียบเบอร์ ไอซี TTL กรุงเทพฯ บริษัทซีเอ็ดยูเคชั่น จำกัด, 2521.

คู่มือไอซีชัพพอร์ทและหน่วยความจำ กรุงเทพฯ บริษัทซีเอ็ดยูเคชั่น จำกัด, 2529.

ก้องเกียรติ ณ สิวา ทฤษฎีการใช้ไมโครไอซี 555 กรุงเทพฯ อิเล็กทรอนิกส์วิลด์,  
2528.



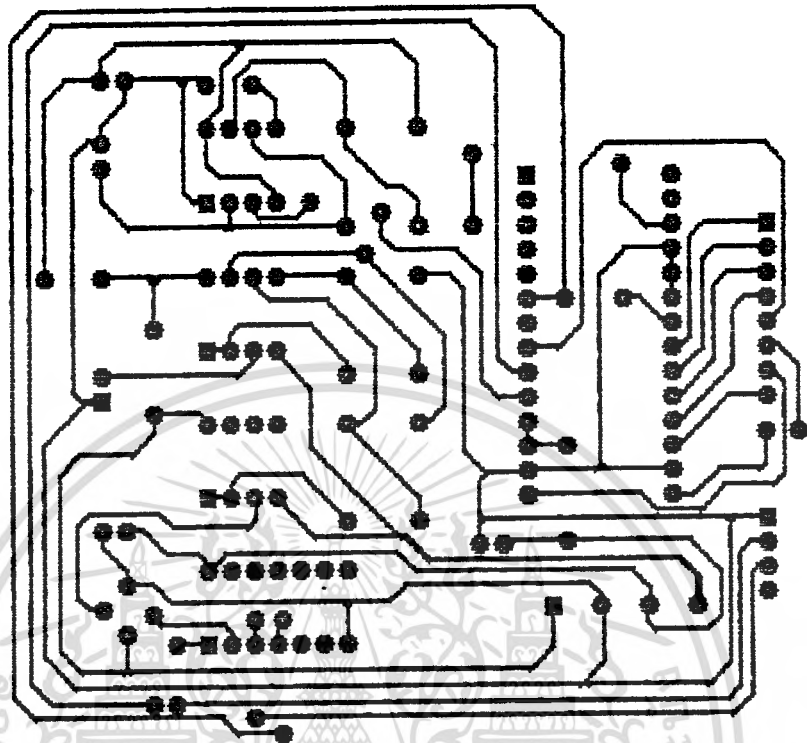
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



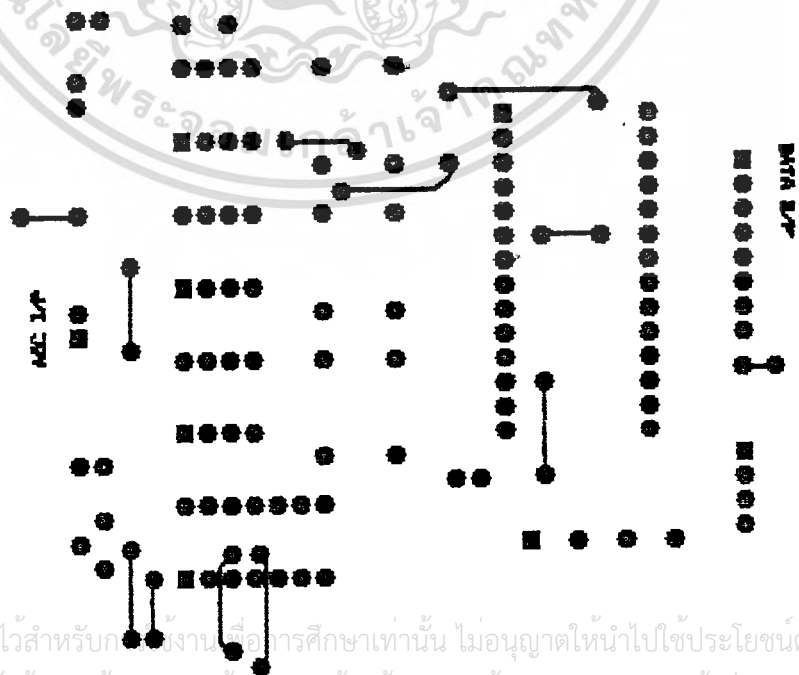
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



# ลายวงจรด้านหน้าและด้านหลังของวงจร AGC

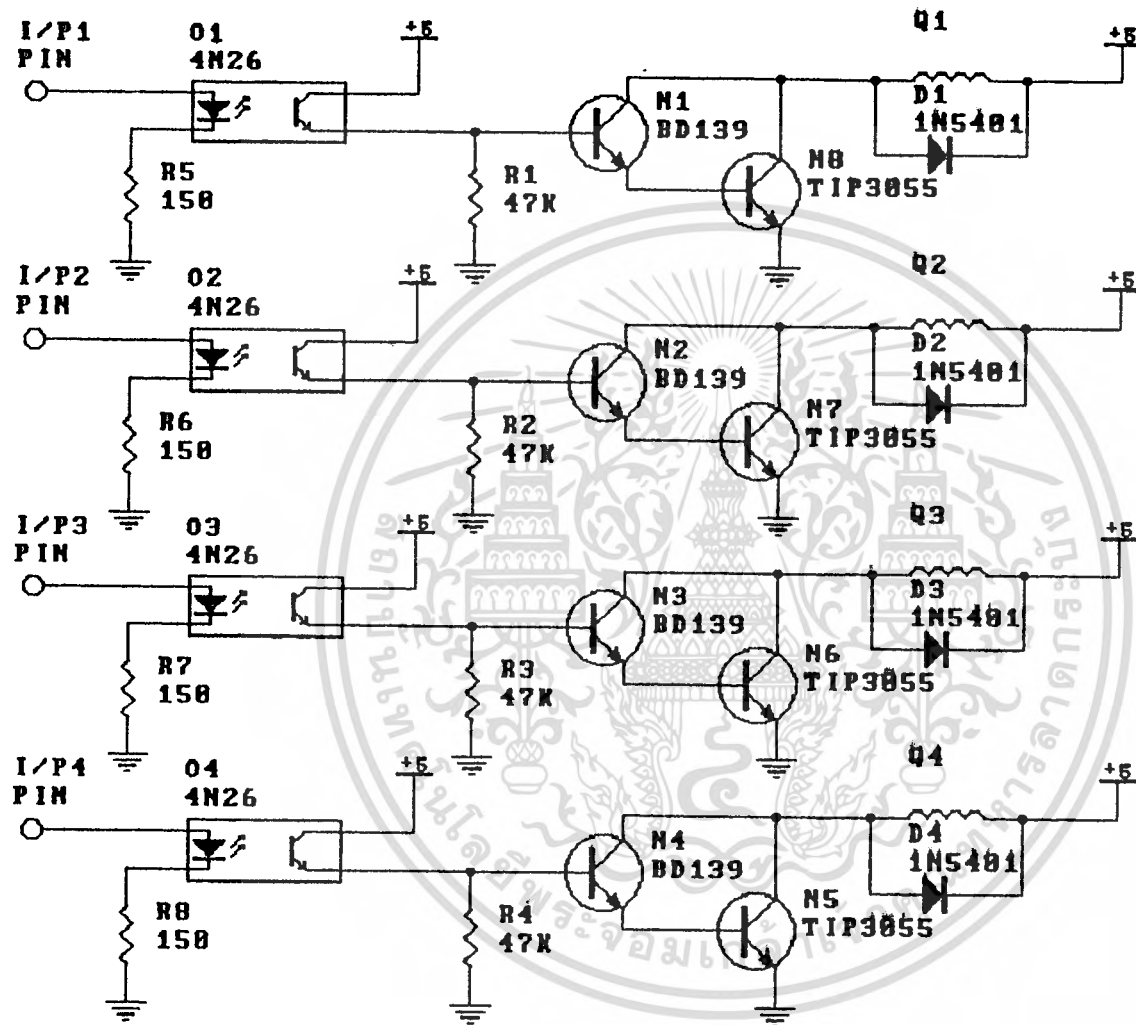


AGC Bottom Layer



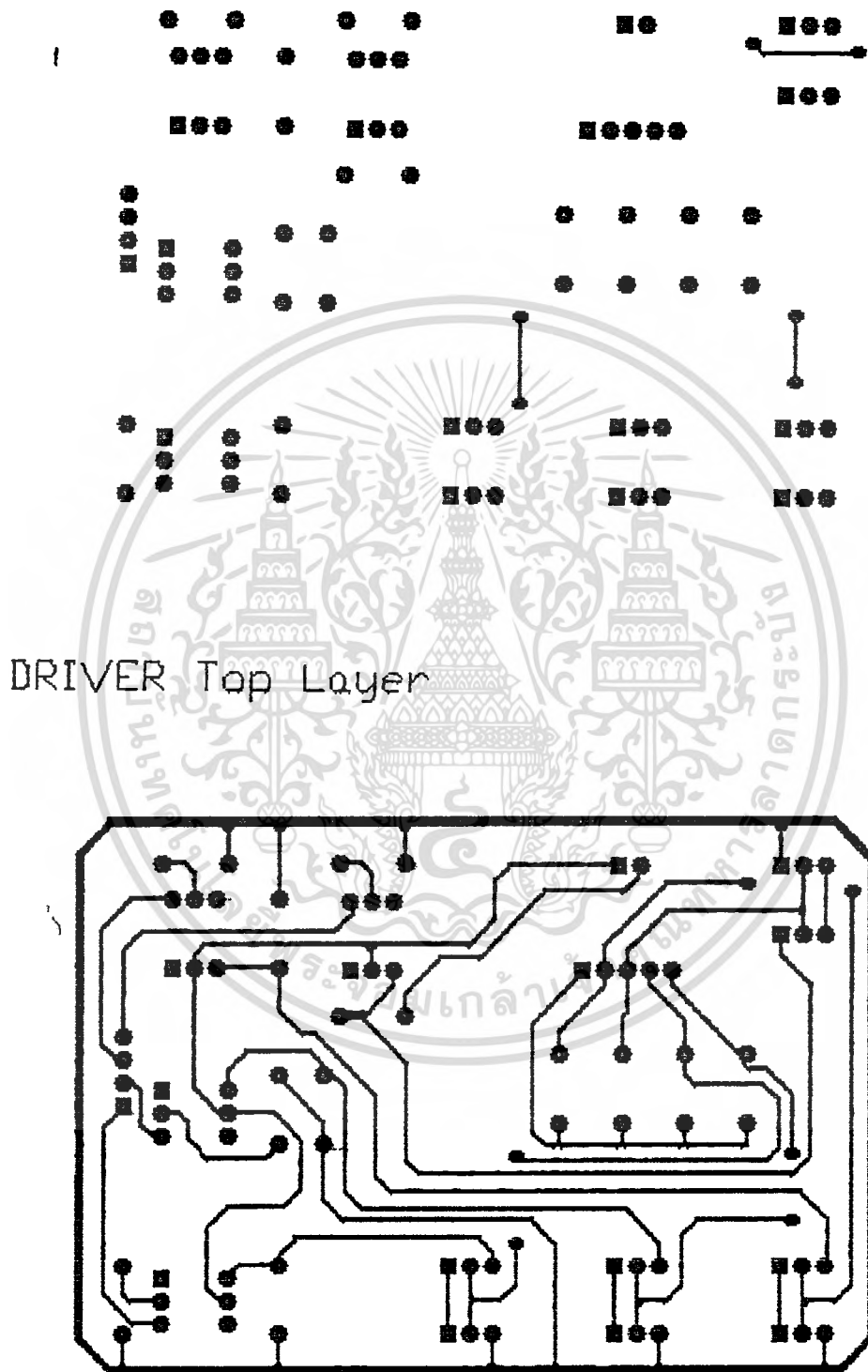
AGC Top Layer

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับคนงานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



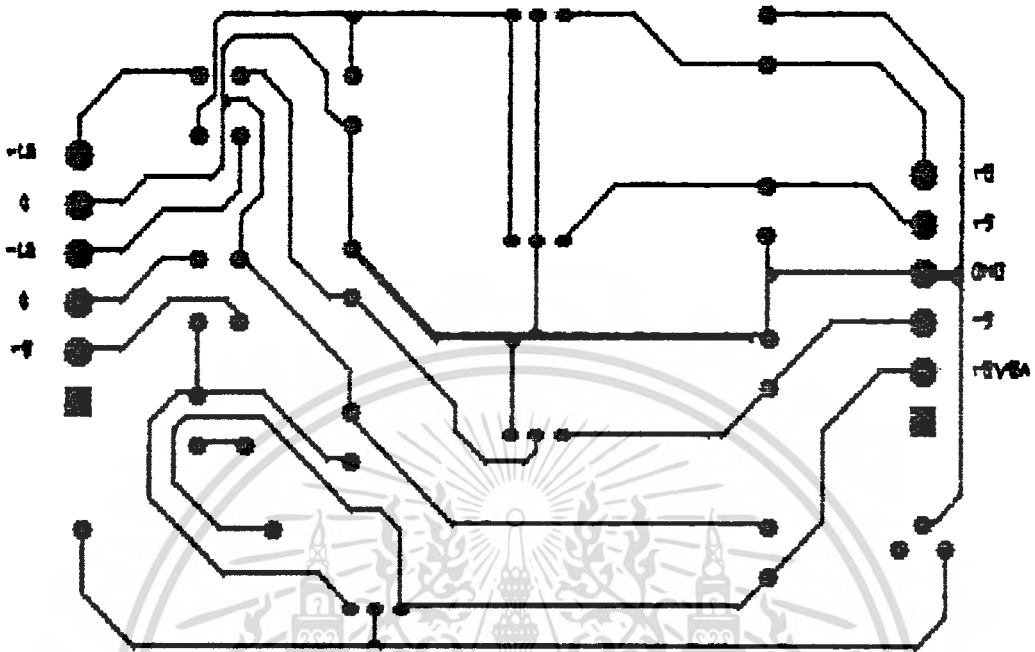
**DRIVER STEPPINGMOTOR**

## ลายวงจรด้านหน้าและด้านหลังของวงจร DRIVE STEPPING MOTOR



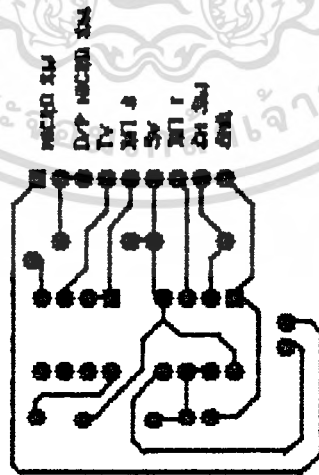
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ หากต้องการนำเอกสารนี้ไปใช้ ต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ลายวงจร POWER SUPPLY



POWER Top Layer

ลายวงจรภาค SENSOR



SENSOR Top Layer

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับใช้เฉพาะในโครงการเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

# Zilog

## FEATURES

- The instruction set contains 158 instructions. The 78 instructions of the 8080A are included as a subset; 8080A software compatibility is maintained.
- Eight MHz, 6 MHz, 4 MHz, and 2.5 MHz clocks for the Z8CH, Z8CB, Z8CA, and Z80 CPU result in rapid instruction execution with consequent high data throughput.
- The extensive instruction set includes string, bit, byte, and word operations. Block searches and block transfers, together with indexed and relative addressing, result in the most powerful data handling capabilities in the microcomputer industry.
- The Z80 microprocessors and associated family of peripheral controllers are linked by a vectored interrupt

system. This system may be daisy-chained to allow implementation of a priority interrupt scheme. Little, if any, additional logic is required for daisy-chaining.

- Duplicate sets of both general-purpose and flag registers are provided, easing the design and operation of system software through single-context switching, background-foreground programming, and single-level interrupt processing. In addition, two 16-bit index registers facilitate program processing of tables and arrays.
- There are three modes of high speed interrupt processing: 8080 similar, non-Z80 peripheral device, and Z80 Family peripheral with or without daisy chain.
- On-chip dynamic memory refresh counter.

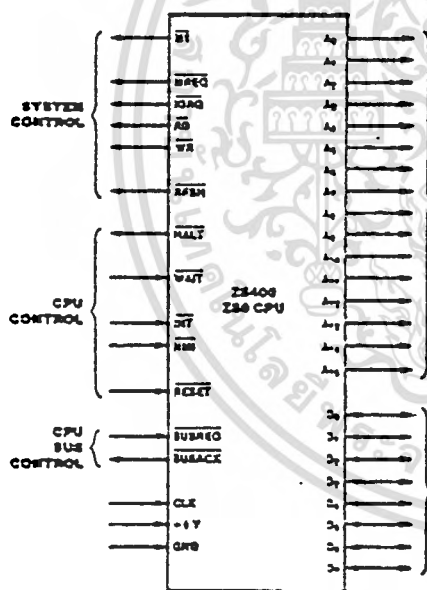


Figure 1. Pin Functions

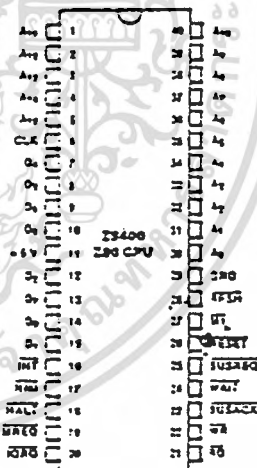


Figure 2a. 40-Pin Dual-In-Line Package (DIP) Pin Assignments

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

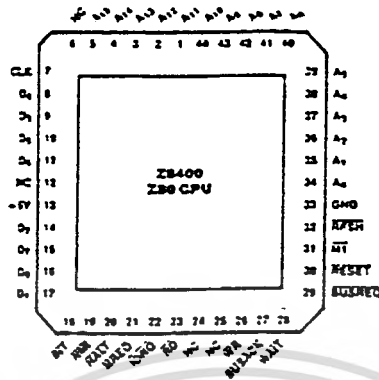


Figure 2b. 44-Pin Chip Carrier Pin Assignments

## GENERAL DESCRIPTION

The Z80, Z80A, Z80B, and Z80H CPUs are third-generation single-chip microprocessors with exceptional computational power. They offer higher system throughput and more efficient memory utilization than comparable second- and third-generation microprocessors. The internal registers contain 208 bits of read/write memory that are accessible to the programmer. These registers include two sets of six general-purpose registers which may be used individually as either 8-bit registers or as 16-bit register pairs. In addition, there are two sets of accumulator and flag registers. A group of "Exchange" instructions makes either set of main or alternate registers accessible to the programmer. The alternate set allows operation in foreground-background mode or it may be reserved for very fast interrupt response.

The Z80 also contains a Stack Pointer, Program Counter, two index registers, a Refresh register (counter), and an Interrupt register. The CPU is easy to incorporate into a system since it requires only a single +5V power source. All output signals are fully decoded and timed to control standard memory or peripheral circuits; the CPU is supported by an extensive family of peripheral controllers. The internal block diagram (Figure 3) shows the primary functions of the Z80 processors. Subsequent text provides more detail on the Z80 I/O controller family, registers, instruction set, interrupts and busy chaining, and CPU timing.

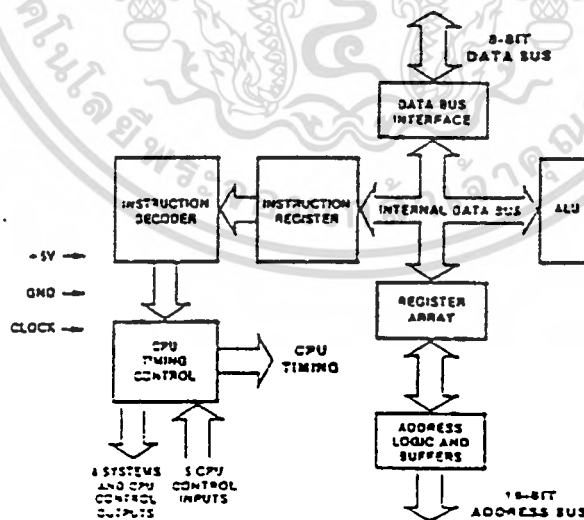


Figure 3. Z80 CPU Block Diagram

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

interrupt has been detected by the CPU (provided that interrupts are enabled and  $\overline{\text{SUSREQ}}$  is not active) a special interrupt processing cycle begins. This is a special fetch (M1) cycle in which  $\overline{\text{ICRQ}}$  becomes active rather than  $\overline{\text{MREQ}}$ , as in a normal M1 cycle. In addition, this special M1 cycle is automatically extended by two WAIT states, to allow for the time required to acknowledge the interrupt request.

**Mode 0 Interrupt Operation.** This mode is similar to the 8080 microprocessor interrupt service procedures. The interrupting device places an instruction on the data bus. This is normally a Restart instruction, which will initiate a call to the selected one of eight restart locations in page zero of memory. Unlike the 8080, the Z80 CPU responds to the Call instruction with only one interrupt acknowledge cycle followed by two memory read cycles.

**Mode 1 Interrupt Operation.** Mode 1 operation is very similar to that for the NMI. The principal difference is that the Mode 1 interrupt has only one restart location, 0035H.

**Mode 2 Interrupt Operation.** This interrupt mode has been designed to utilize most effectively the capabilities of the Z80 microprocessor and its associated peripheral family. The interrupting peripheral device selects the starting address of the interrupt service routine. It does this by placing an 8-bit vector on the data bus during the interrupt acknowledge cycle. The CPU forms a pointer using this byte as the lower 8 bits and the contents of the I register as the upper 8 bits. This points to an entry in a table of addresses for interrupt service routines. The CPU then jumps to the routine at that address. This flexibility in selecting the interrupt service routine address allows the peripheral device to use several different types of service routines. These routines may be located at any available location in memory. Since the interrupting device supplies the low-order byte of the 2-byte vector, bit 0 ( $A_0$ ) must be a zero.

**Interrupt Priority (Daisy Chaining and Nested Interrupts).** The interrupt priority of each peripheral device is determined by its physical location within a daisy-chain configuration. Each device in the chain has an interrupt enable input line (IEI) and an interrupt enable output line (IEO), which is fed to the next lower priority device. The first device in the daisy chain has its IEI input hardwired to a High

level. The first device has highest priority, while each succeeding device has a corresponding lower priority. This arrangement permits the CPU to select the highest priority interrupt from several simultaneously interrupting peripherals.

The interrupting device disables its IEO line to the next lower priority peripheral until it has been serviced. After servicing, its IEO line is raised, allowing lower priority peripherals to demand interrupt servicing.

The Z80 CPU will nest (queue) any pending interrupts or interrupts received while a selected peripheral is being serviced.

**Interrupt Enable/Disable Operation.** Two flip-flops, IFF<sub>1</sub> and IFF<sub>2</sub>, referred to in the register description, are used to signal the CPU interrupt status. Operation of the two flip-flops is described in Table 2. For more details, refer to the Z80 CPU Technical Manual (C3-002S-31) and Z80 Assembly Language Programming Manual (C3-0002-01).

Table 2. State of Flip-Flops

| Action                       | IFF <sub>1</sub> | IFF <sub>2</sub> | Comments                                                                              |
|------------------------------|------------------|------------------|---------------------------------------------------------------------------------------|
| CPU Reset                    | 0                | 0                | Maskable interrupt<br>INT disabled                                                    |
| DI instruction execution     | 0                | 0                | Maskable interrupt<br>INT disabled                                                    |
| EI instruction execution     | 1                | 1                | Maskable interrupt<br>INT enabled                                                     |
| LD A,I instruction execution | •                | •                | IFF <sub>2</sub> → Parity flag                                                        |
| LD A,R instruction execution | •                | •                | IFF <sub>2</sub> → Parity flag                                                        |
| Accept NMI                   | 0                | IFF <sub>1</sub> | IFF <sub>1</sub> = IFF <sub>2</sub><br>(Maskable interrupt<br>INT disabled)           |
| RETN instruction execution   | IFF <sub>2</sub> | •                | IFF <sub>2</sub> = IFF <sub>1</sub> at<br>completion of an<br>NMI service<br>routine. |

## Z80 MICROPROCESSOR FAMILY

The Z80 microprocessor is the central element of a comprehensive microprocessor product family. This family works together in most applications with minimum requirements for additional logic, facilitating the design of efficient and cost-effective microcomputer-based systems.

Zilog has designed five components to provide extensive support for the Z80 microprocessor. These are:

- The PIO (Parallel Input/Output) operates in both data-byte I/O transfer mode (with handshaking) and in bit mode (without handshaking). The PIO may be configured to interface with standard parallel peripheral devices such as printers, tape punches, and keyboards.
- The CTC (Counter/Timer Circuit) features four programmable 8-bit counter/timers, each of which has an

8-bit prescaler. Each of the four channels may be configured to operate in either counter or timer mode.

- The DMA (Direct Memory Access) controller provides dual port data transfer operations and the ability to terminate data transfer as a result of a pattern match.
- The SIO (Serial Input/Output) controller offers two channels. It is capable of operating in a variety of programmable modes for both synchronous and asynchronous communication, including Bi-Synch and SCLC.
- The DART (Dual Asynchronous Receiver/Transmitter) device provides low cost asynchronous serial communication. It has two channels and a full modem control interface.

## Z80 CPU REGISTERS

Figure 4 shows three groups of registers within the Z80 CPU. The first group consists of duplicate sets of 8-bit registers: a principal set and an alternate set (designated by 'prime', e.g., A'). Both sets consist of the Accumulator Register, the Flag Register, and six general-purpose registers. Transfer of data between these duplicate sets of registers is accomplished by use of "Exchange" instructions. The result is faster response to interrupts and easy, efficient implementation of such versatile pro-

gramming techniques as background-foreground data processing. The second set of registers consists of six registers with assigned functions. These are the I (Interrupt Register), the R (Refresh Register), the IX and IY (Index Registers), the SP (Stack Pointer), and the PC (Program Counter). The third group consists of two interrupt status flip-flops, plus an additional pair of flip-flops which assist in identifying the interrupt mode at any particular time. Table 1 provides further information on these registers.

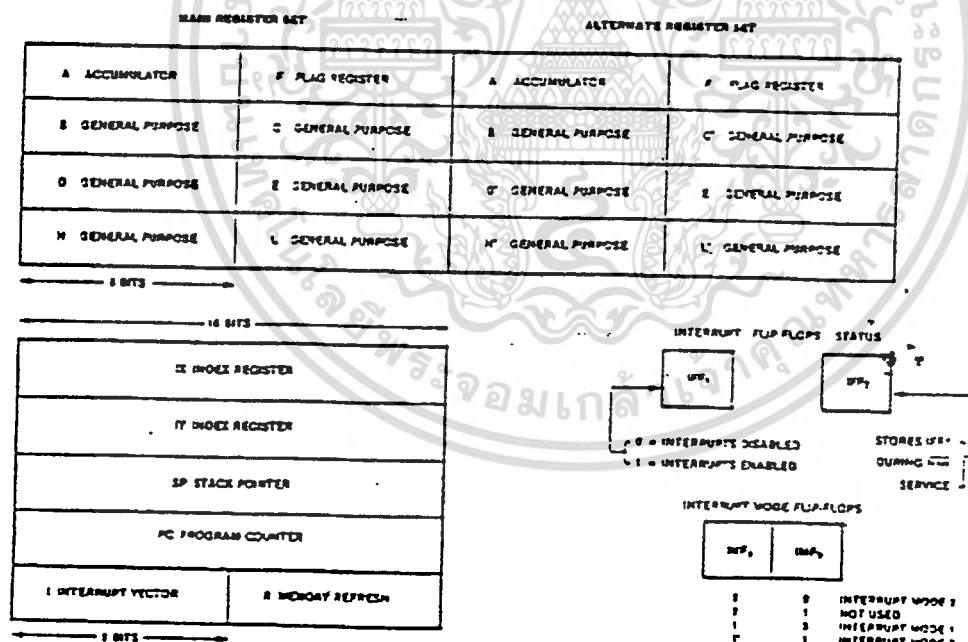


Figure 4. CPU Registers

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## Z80 CPU REGISTERS (Continued)

Table 1. Z80 CPU Registers

| Register                                                                                                                                                           | Size (Bits)        | Remarks                                                                                                                                             |
|--------------------------------------------------------------------------------------------------------------------------------------------------------------------|--------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------|
| A, A'                                                                                                                                                              | Accumulator        | 8<br>Stores an operand or the results of an operation.                                                                                              |
| F, F'                                                                                                                                                              | Flags              | 8<br>See Instruction Set.                                                                                                                           |
| B, B'                                                                                                                                                              | General Purpose    | 8<br>Can be used separately or as a 16-bit register with C.                                                                                         |
| C, C'                                                                                                                                                              | General Purpose    | 8<br>See B, above.                                                                                                                                  |
| D, D'                                                                                                                                                              | General Purpose    | 8<br>Can be used separately or as a 16-bit register with E.                                                                                         |
| E, E'                                                                                                                                                              | General Purpose    | 8<br>See D, above.                                                                                                                                  |
| H, H'                                                                                                                                                              | General Purpose    | 8<br>Can be used separately or as a 16-bit register with L.                                                                                         |
| L, L'                                                                                                                                                              | General Purpose    | 8<br>See H, above.                                                                                                                                  |
| Note: The (B,C), (D,E), and (H,L) sets are combined as follows:<br>B — High byte    C — Low byte<br>D — High byte    E — Low byte<br>H — High byte    L — Low byte |                    |                                                                                                                                                     |
| I                                                                                                                                                                  | Interrupt Register | 8<br>Stores upper eight bits of memory address for vectored Interrupt processing.                                                                   |
| R                                                                                                                                                                  | Refresh Register   | 8<br>Provides user-transparent dynamic memory refresh. Automatically incremented and placed on the address bus during each instruction fetch cycle. |
| IX                                                                                                                                                                 | Index Register     | 16<br>Used for indexed addressing.                                                                                                                  |
| IY                                                                                                                                                                 | Index Register     | 16<br>Used for indexed addressing.                                                                                                                  |
| SP                                                                                                                                                                 | Stack Pointer      | 16<br>Holds address of the top of the stack. See Push or Pop in instruction set.                                                                    |
| PC                                                                                                                                                                 | Program Counter    | 16<br>Holds address of next instruction.                                                                                                            |
| IFF <sub>1</sub> -IFF <sub>2</sub>                                                                                                                                 | Interrupt Enable   | Flip-Flops<br>Set or reset to indicate interrupt status (see Figure 4).                                                                             |
| IMF <sub>0</sub> -IMF <sub>7</sub>                                                                                                                                 | Interrupt Mode     | Flip-Flops<br>Reflect interrupt mode (see Figure 4).                                                                                                |

## INTERRUPTS: GENERAL OPERATION

The CPU accepts two interrupt input signals:  $\overline{\text{NMI}}$  and  $\overline{\text{INT}}$ . The  $\overline{\text{NMI}}$  is a non-maskable interrupt and has the highest priority.  $\overline{\text{INT}}$  is a lower priority interrupt and it requires that interrupts be enabled in software in order to operate.  $\overline{\text{INT}}$  can be connected to multiple peripheral devices in a wired-OR configuration.

The Z80 has a single response mode for interrupt service for the non-maskable interrupt. The maskable interrupt,  $\overline{\text{INT}}$ , has three programmable response modes available. These are:

- Mode 0 — similar to the 8080 microprocessor.
- Mode 1 — Peripheral Interrupt service, for use with non-8080/Z80 systems.
- Mode 2 — a vectored interrupt scheme, usually daisy-chained, for use with Z80 Family and compatible peripheral devices.

The CPU services interrupts by sampling the  $\overline{\text{NMI}}$  and  $\overline{\text{INT}}$  signals at the rising edge of the last clock of an instruction. Further interrupt service processing depends upon the type of interrupt that was detected. Details on interrupt responses are shown in the CPU Timing Section.

**Non-Maskable Interrupt ( $\overline{\text{NMI}}$ ).** The nonmaskable interrupt cannot be disabled by program control and therefore will be accepted at all times by the CPU.  $\overline{\text{NMI}}$  is usually reserved for servicing only the highest priority type interrupts, such as that for orderly shutdown after power failure has been detected. After recognition of the  $\overline{\text{NMI}}$  signal (provided  $\overline{\text{BUSREQ}}$  is not active), the CPU jumps to restart location 0065H. Normally, software starting at this address contains the interrupt service routine.

**Maskable Interrupt ( $\overline{\text{INT}}$ ).** Regardless of the interrupt mode set by the user, the Z80 response to a maskable interrupt input follows a common timing cycle. After the

## PIN DESCRIPTIONS

**A<sub>0</sub>-A<sub>15</sub>.** Address Bus (output, active High, 3-state). A<sub>0</sub>-A<sub>15</sub> form a 16-bit address bus. The Address Bus provides the address for memory data bus exchanges (up to 64K bytes) and for I/O device exchanges.

**BUSACK.** Bus Acknowledge (output, active Low). Bus Acknowledge indicates to the requesting device that the CPU address bus, data bus, and control signals MREQ, IORQ, RD, and WR have entered their high-impedance states. The external circuitry can now control these lines.

**BUSREQ.** Bus Request (input, active Low). Bus Request has a higher priority than NMI and is always recognized at the end of the current machine cycle. BUSREQ forces the CPU address bus, data bus, and control signals MREQ, IORQ, RD, and WR to go to a high-impedance state so that other devices can control these lines. BUSREQ is normally wired-OR and requires an external pullup for these applications. Extended BUSREQ periods due to extensive DMA operations can prevent the CPU from properly refreshing dynamic RAMs.

**D<sub>0</sub>-D<sub>7</sub>.** Data Bus (input/output, active High, 3-state). D<sub>0</sub>-D<sub>7</sub> constitute an 8-bit bidirectional data bus, used for data exchanges with memory and I/O.

**HALT.** Halt State (output, active Low). HALT indicates that the CPU has executed a Halt instruction and is awaiting either a nonmaskable or a maskable interrupt (with the mask enabled) before operation can resume. While halted, the CPU executes NOPs to maintain memory refresh.

**INT.** Interrupt Request (input, active Low). Interrupt Request is generated by I/O devices. The CPU honors a request at the end of the current instruction if the internal software-controlled interrupt enable flip-flop (IEF) is enabled. INT is normally wired-OR and requires an external pullup for these applications.

**IORQ.** Input/Output Request (output, active Low, 3-state). IORQ indicates that the lower half of the address bus holds a valid I/O address for an I/O read or write operation. IORQ is also generated concurrently with M1 during an interrupt acknowledge cycle to indicate that an interrupt response vector can be placed on the data bus.

**M1.** Machine Cycle One (output, active Low). M1, together with MREQ, indicates that the current machine cycle is the opcode fetch cycle of an instruction execution. M1, together with IORQ, indicates an interrupt acknowledge cycle.

**MREQ.** Memory Request (output, active Low, 3-state). MREQ indicates that the address bus holds a valid address for a memory read or memory write operation.

**NMI.** Non-Maskable Interrupt (input, negative edge-triggered). NMI has a higher priority than INT. NMI is always recognized at the end of the current instruction, independent of the status of the interrupt enable flip-flop, and automatically forces the CPU to restart at location 0066H.

**RD.** Read (output, active Low, 3-state). RD indicates that the CPU wants to read data from memory or an I/O device. The addressed I/O device or memory should use this signal to gate data onto the CPU data bus.

**RESET.** Reset (input, active Low). RESET initializes the CPU as follows: it resets the interrupt enable flip-flop, clears the PC and Registers I and R, and sets the interrupt status to Mode 0. During reset time, the address and data bus go to a high-impedance state, and all control output signals go to the inactive state. Note that RESET must be active for a minimum of three full clock cycles before the reset operation is complete.

**RFSH.** Refresh (output, active Low). RFSH, together with MREQ, indicates that the lower seven bits of the system's address bus can be used as a refresh address to the system's dynamic memories.

**WAIT.** Wait (input, active Low). WAIT indicates to the CPU that the addressed memory or I/O devices are not ready for a data transfer. The CPU continues to enter a Wait state as long as this signal is active. Extended WAIT periods can prevent the CPU from refreshing dynamic memory properly.

**WR.** Write (output, active Low, 3-state). WR indicates that the CPU data bus holds valid data to be stored at the addressed memory or I/O location.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## SUMMARY OF FLAG OPERATION

| Instructions                        | D <sub>7</sub> | S | Z | H | P/V | N   | C | D <sub>0</sub> | Comments                                                                                              |
|-------------------------------------|----------------|---|---|---|-----|-----|---|----------------|-------------------------------------------------------------------------------------------------------|
| ADD A, $\pm$ AOC A, s               | :              | : | X | : | X   | V   | 0 | :              | 8-bit add or add with carry.                                                                          |
| SUB $\pm$ SBC A, $\pm$ CP $\pm$ NEG | :              | : | X | : | X   | V   | 1 | :              | 8-bit subtract, subtract with carry, compare and negate accumulator.                                  |
| AND s                               | :              | : | X | 1 | X   | P   | 0 | 0              | Logical operation.                                                                                    |
| OR $\pm$ XCR s                      | :              | : | X | 0 | X   | P   | 0 | 0              | Logical operation.                                                                                    |
| INC s                               | :              | : | X | : | X   | V   | 0 | *              | 8-bit increment.                                                                                      |
| DEC s                               | :              | : | X | : | X   | V   | 1 | *              | 8-bit decrement.                                                                                      |
| ADD DD, ss                          | *              | * | X | X | X   | *   | 0 | :              | 16-bit add.                                                                                           |
| ADC HL, ss                          | :              | : | X | X | X   | V   | 0 | :              | 16-bit add with carry.                                                                                |
| SBC HL, ss                          | :              | : | X | X | X   | V   | 1 | :              | 16-bit subtract with carry.                                                                           |
| RLA; RLCA; RRA; RRCA                | *              | * | X | 0 | X   | *   | 0 | :              | Rotate accumulator.                                                                                   |
| RL m; RLC m; RR m;                  | :              | : | X | 0 | X   | P   | 0 | :              | Rotate and shift locations.                                                                           |
| RRC m; SRA m;                       |                |   |   |   |     |     |   |                |                                                                                                       |
| SRL m                               |                |   |   |   |     |     |   |                |                                                                                                       |
| RLD; RRD                            | :              | : | X | 0 | X   | P   | 0 | *              | Rotate 8-bit left and right.                                                                          |
| DAA                                 | :              | : | X | : | X   | P   | * | :              | Decimal adjust accumulator.                                                                           |
| CPL                                 | *              | * | X | 1 | X   | *   | 1 | *              | Complement accumulator.                                                                               |
| SCF                                 | *              | * | X | 0 | X   | *   | 0 | 1              | Set carry.                                                                                            |
| CCF                                 | *              | * | X | X | X   | *   | 0 | :              | Complement carry.                                                                                     |
| IN r (C)                            | :              | : | X | 0 | X   | P   | 0 | *              | Input register indirect.                                                                              |
| INI; INO; OUT; OUTD                 | X              | : | X | X | X   | X   | 1 | *              | Block input and output. Z = 1 if B = 0, otherwise Z = 0.                                              |
| INIR; INCR; OTIR; OTDR              | X              | 1 | X | X | X   | X   | 1 | *              | Block input and output. Z = 1 if B = 0, otherwise Z = 0.                                              |
| LDI; LDD                            | X              | X | X | 0 | X   | :   | 0 | *              | Block transfer instructions. P/V = 1 if SC = 0, otherwise P/V = 0.                                    |
| LDIR; LDDR                          | X              | X | X | 0 | X   | 0   | 0 | *              | Block transfer instructions. P/V = 1 if SC = 0, otherwise P/V = 0.                                    |
| CPI; CPRI; CPD; CPDR                | X              | : | X | X | X   | :   | 1 | *              | Block search instructions. Z = 1 if A = (HL), otherwise Z = 0. P/V = 1 if SC = 0, otherwise P/V = 0.  |
| LD A; I, LD A, R                    | :              | : | X | 0 | X   | IFF | 0 | *              | IFF, the content of the interrupt enable flip-flop, (IFF <sub>2</sub> ), is copied into the P/V flag. |
| BIT b, s                            | X              | : | X | 1 | X   | X   | 0 | *              | The state of bit b of location s is copied into the Z flag.                                           |

## SYMBOLIC NOTATION

| Symbol | Operation                                                                                                                                                                                                                                                                                                                                                                                                                                            | Symbol | Operation                                                                               |
|--------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|--------|-----------------------------------------------------------------------------------------|
| S      | Sign flag. S = 1 if the MSB of the result is 1.                                                                                                                                                                                                                                                                                                                                                                                                      | :      | The flag is affected according to the result of the operation.                          |
| Z      | Zero flag. Z = 1 if the result of the operation is 0.                                                                                                                                                                                                                                                                                                                                                                                                | *      | The flag is unchanged by the operation.                                                 |
| P/V    | Parity or overflow flag. Parity (P) and overflow (V) share the same flag. Logical operations affect this flag with the parity of the result while arithmetic operations affect this flag with the overflow of the result. If P/V holds parity, P/V = 1 if the result of the operation is even; P/V = 0 if result is odd. If P/V holds overflow, P/V = 1 if the result of the operation produced an overflow. If P/V does not hold overflow, P/V = 0. | 0      | The flag is reset by the operation.                                                     |
| H*     | Half-carry flag. H = 1 if the add or subtract operation produced a carry into, or borrow from, bit 4 of the accumulator.                                                                                                                                                                                                                                                                                                                             | 1      | The flag is set by the operation.                                                       |
| N*     | Add/Subtract flag. N = 1 if the previous operation was a subtract.                                                                                                                                                                                                                                                                                                                                                                                   | X      | The flag is indeterminate.                                                              |
| C      | Carry/Link flag. C = 1 if the operation produced a carry from the MSB of the operand or result.                                                                                                                                                                                                                                                                                                                                                      | V      | P/V flag affected according to the overflow result of the operation.                    |
|        |                                                                                                                                                                                                                                                                                                                                                                                                                                                      | P      | P/V flag affected according to the parity result of the operation.                      |
|        |                                                                                                                                                                                                                                                                                                                                                                                                                                                      | r      | Any one of the CPU registers A, B, C, D, E, H, L.                                       |
|        |                                                                                                                                                                                                                                                                                                                                                                                                                                                      | s      | Any 8-bit location for all the addressing modes allowed for the particular instruction. |
|        |                                                                                                                                                                                                                                                                                                                                                                                                                                                      | ss     | Any 16-bit location for all the addressing modes allowed for that instruction.          |
|        |                                                                                                                                                                                                                                                                                                                                                                                                                                                      | i      | Any one of the two index registers IX or IY.                                            |
|        |                                                                                                                                                                                                                                                                                                                                                                                                                                                      | R      | Refresh counter.                                                                        |
|        |                                                                                                                                                                                                                                                                                                                                                                                                                                                      | n      | 8-bit value in range < 0, 255 >.                                                        |
|        |                                                                                                                                                                                                                                                                                                                                                                                                                                                      | nn     | 16-bit value in range < 0, 65535 >.                                                     |

\* H and N flags are used in conjunction with the decimal adjust instruction (DAA) to correctly correct the result and packed BCD format following addition or subtraction using operands with packed BCD format.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



## 8255A/8255A-5 PROGRAMMABLE PERIPHERAL INTERFACE

- MCS-35™ Compatible 8255A-5
  - 24 Programmable I/O Pins
  - Completely TTL Compatible
  - Fully Compatible with Intel Microprocessor Families
  - Improved Timing Characteristics
  - Direct Bit Set/Reset Capability Easing Control Application Interface
  - Reduces System Package Count
  - Improved DC Driving Capability
  - Available in EXPRESS
    - Standard Temperature Range
    - Extended Temperature Range
  - 40 Pin DIP Package or 44 Lead PLCC
- (See Intel Packaging: Order Number: 231268)

The Intel 8255A is a general purpose programmable I/O device designed for use with Intel microprocessors. It has 24 I/O pins which may be individually programmed in 2 groups of 12 and used in 3 major modes of operation. In the first mode (MODE 0), each group of 12 I/O pins may be programmed in sets of 4 to be input or output. In MODE 1, the second mode, each group may be programmed to have 8 lines of input or output. Of the remaining 4 pins, 3 are used for handshaking and interrupt control signals. The third mode of operation (MODE 2) is a bidirectional bus mode which uses 8 lines for a bidirectional bus, and 5 lines, borrowing one from the other group, for handshaking.

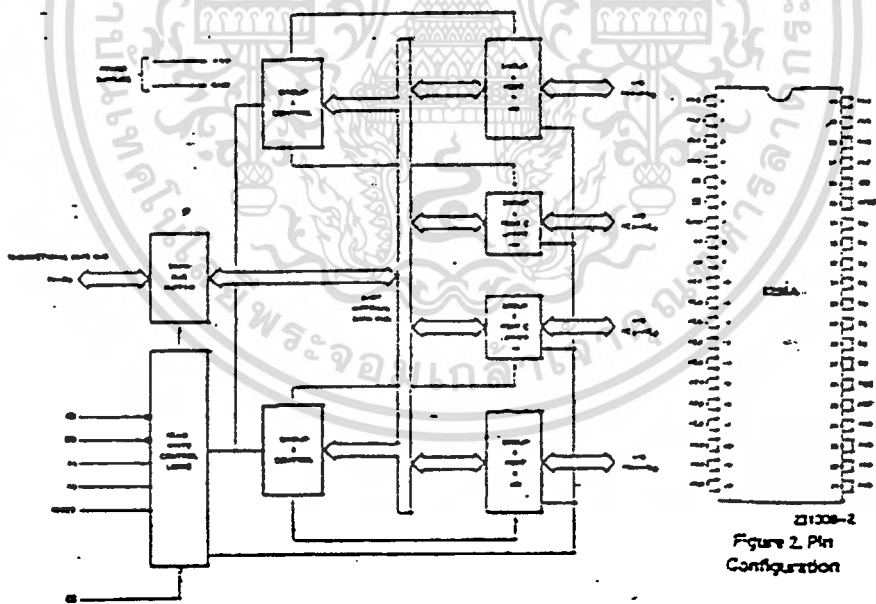


Figure 1. 8255A Block Diagram

Figure 2. Pin Configuration

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## 8255A FUNCTIONAL DESCRIPTION

CPU Address and Control busses and in turn, issues commands to both of the Control Groups.

## General

The 8255A is a programmable peripheral interface (PPI) device designed for use in Intel microcomputer systems. Its function is that of a general purpose I/O component to interface peripheral equipment to the microcomputer system bus. The functional configuration of the 8255A is programmed by the system software so that normally no external logic is necessary to interface peripheral devices or structures.

## Data Bus Buffer

This 3-state bidirectional 8-bit buffer is used to interface the 8255A to the system data bus. Data is transmitted or received by the buffer upon execution of input or output instructions by the CPU. Control words and status information are also transferred through the data bus buffer.

## Read/Write and Control Logic

The function of this block is to manage all of the internal and external transfers of both Data and Control or Status words. It accepts inputs from the

 $(\overline{CS})$ 

Chip Select. A "low" on this input pin enables the communication between the 8255A and the CPU.

 $(\overline{RD})$ 

Read. A "low" on this input pin enables the 8255A to send the Data or status information to the CPU on the data bus. In essence, it allows the CPU to "read from" the 8255A.

 $(\overline{WR})$ 

Write. A "low" on this input pin enables the CPU to write data or control words into the 8255A.

 $(A_0 \text{ and } A_1)$ 

Port Select 0 and Port Select 1. These input signals, in conjunction with the  $\overline{RD}$  and  $\overline{WR}$  inputs, control the selection of one of the three ports or the control word registers. They are normally connected to the least significant bits of the address bus ( $A_0$  and  $A_1$ ).

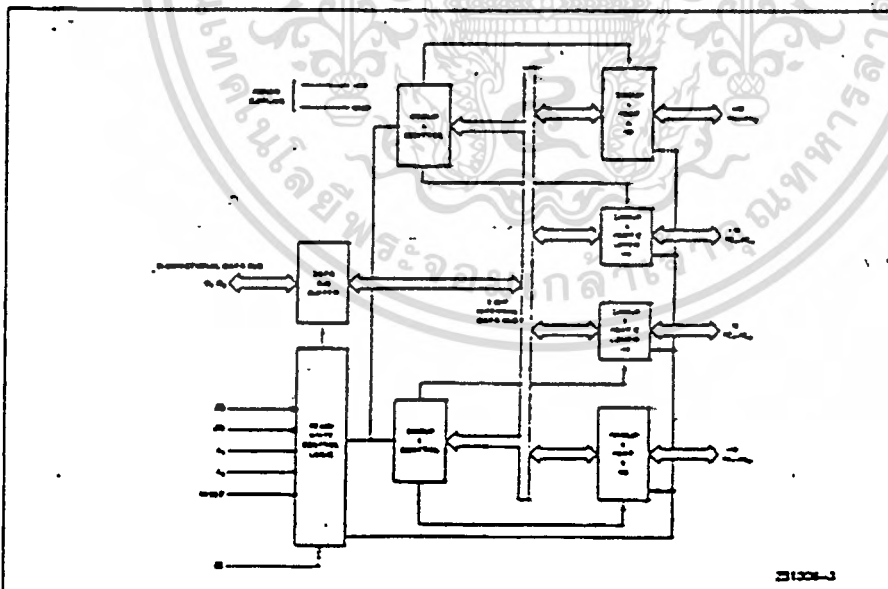


Figure 3. 8255A Block Diagram Showing Data Bus Buffer and Read/Write Control Logic Functions

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## 8255A BASIC OPERATION

| A <sub>1</sub> | A <sub>0</sub> | RD | WR | CS | Input Operation (READ)   |
|----------------|----------------|----|----|----|--------------------------|
| 0              | 0              | 0  | 1  | 0  | Port A → Data Bus        |
| 0              | 1              | 0  | 1  | 0  | Port B → Data Bus        |
| 1              | 0              | 0  | 1  | 0  | Port C → Data Bus        |
|                |                |    |    |    | Output Operation (WRITE) |
| 0              | 0              | 1  | 0  | 0  | Data Bus → Port A        |
| 0              | 1              | 1  | 0  | 0  | Data Bus → Port B        |
| 1              | 0              | 1  | 0  | 0  | Data Bus → Port C        |
| 1              | 1              | 1  | 0  | 0  | Data Bus → Control       |
|                |                |    |    |    | Disable Function         |
| X              | X              | X  | X  | 1  | Data Bus → 3-State       |
| 1              | 1              | 0  | 1  | 0  | Illegal Condition        |
| X              | X              | 1  | 1  | 0  | Data Bus → 3-State       |

## (RESET)

Reset. A "high" on this input clears the control register and all ports (A, B, C) are set to the input mode.

## Group A and Group B Controls

The functional configuration of each port is programmed by the systems software. In essence, the CPU "outputs" a control word to the 8255A. The control word contains information such as "mode", "bit set", "bit reset", etc., that initializes the functional configuration of the 8255A.

Each of the Control blocks (Group A and Group B) accepts "commands" from the Read/Write Control Logic, receives "control words" from the internal data bus and issues the proper commands to its associated ports.

Control Group A—Port A and Port C upper (C<sub>4</sub>—C<sub>7</sub>)  
Control Group B—Port B and Port C lower (C<sub>3</sub>—C<sub>0</sub>)

The Control Word Register can Only be written into. No Read operation of the Control Word Register is allowed.

## Ports A, B, and C

The 8255A contains three 8-bit ports (A, B, and C). All can be configured in a wide variety of functional characteristics by the system software but each has its own special features or "personality" to further enhance the power and flexibility of the 8255A.

Port A. One 8-bit data output latch/buffer and one 8-bit data input latch.

Port B. One 8-bit data input/output latch/buffer and one 8-bit data input buffer.

Port C. One 8-bit data output latch/buffer and one 8-bit data input buffer (no latch for input). This port can be divided into two 4-bit ports under the mode control. Each 4-bit port contains a 4-bit latch and it can be used for the control signal outputs and status signal inputs in conjunction with ports A and B.

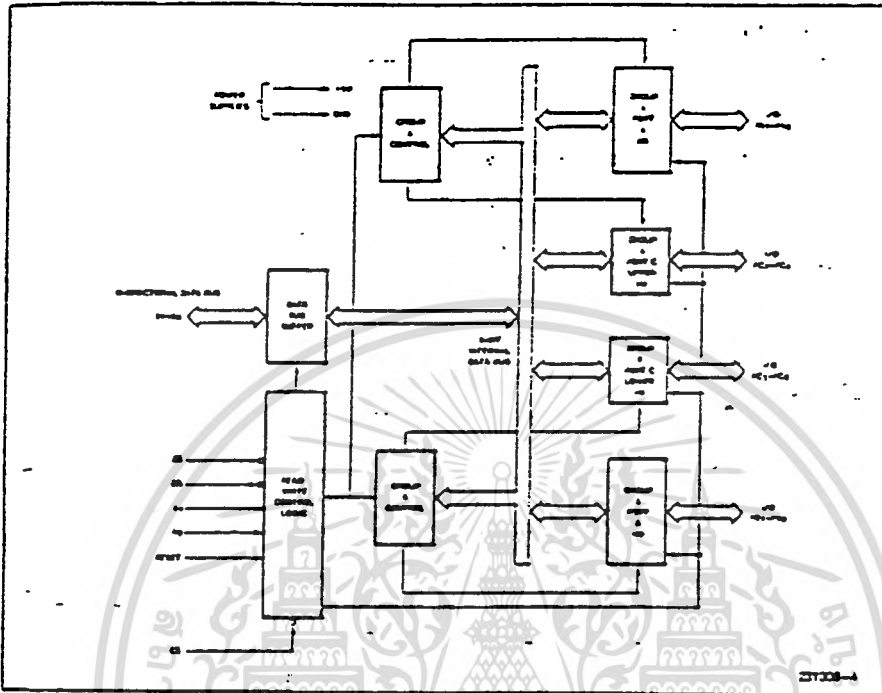
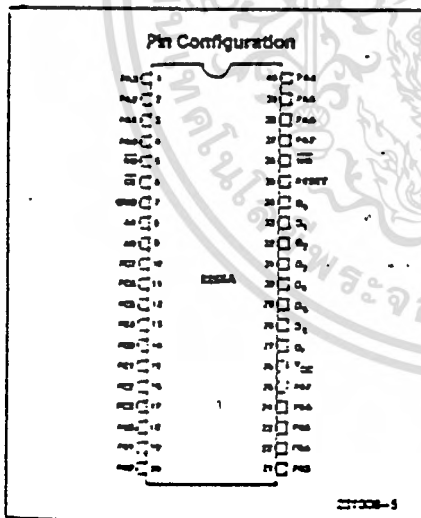


Figure 4. 8255A Block Diagram Showing Group A and Group B Control Functions



## Pin Names

|                                |                           |
|--------------------------------|---------------------------|
| C <sub>0</sub> -C <sub>7</sub> | Data Bus (Bi-Directional) |
| RESET                          | Reset Input               |
| CS                             | Chip Select               |
| RD                             | Read Input                |
| WR                             | Write Input               |
| A0, A1                         | Port Address              |
| PA7-PA0                        | Port A (BIT)              |
| PS7-PS0                        | Port B (BIT)              |
| PC7-PC0                        | Port C (BIT)              |
| V <sub>CC</sub>                | +5 Volts                  |
| GND                            | 0 Volts                   |

## 8255A OPERATIONAL DESCRIPTION

## Mode Selection

There are three basic modes of operation that can be selected by the system software:

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



## 2764 (8K x 8) UV ERASABLE PROM

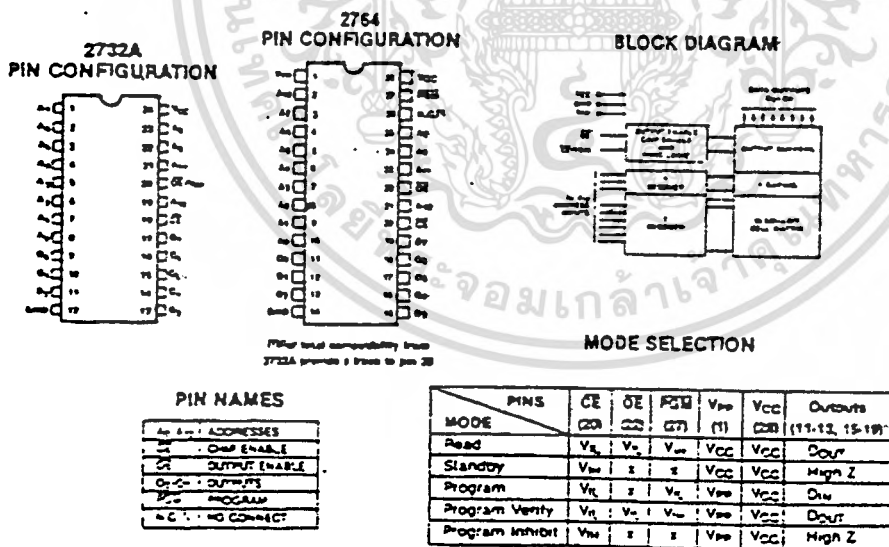
- 200 ns (2764-2) Maximum Access Time... HMOS<sup>®</sup>-E Technology
- Compatible to high speed 8MHz 8086-2 MPU ... Zero WAIT State
- Two-Line Control
- Pin Compatible to 2732A EPROM
- Industry Standard Pinout ... JEDEC Approved
- Low Standby Current ... 35mA Max.

The Intel<sup>®</sup> 2764 is a 5V only 65,536 bit ultraviolet erasable and electrically programmable read-only memory (EPROM). The standard 2764 access time is 250ns with speed selection available at 200ns. The access time is compatible to high performance microprocessors, such as Intel's 8MHz 8086-2. In these systems, the 2764 allows the microprocessor to operate without the addition of WAIT states.

An important 2764 feature is the separate output control, Output Enable ( $\overline{OE}$ ) from the Chip Enable control ( $\overline{CE}$ ). The  $\overline{OE}$  control eliminates bus contention in multiple bus microprocessor systems. Intel's Application Note AP-72 describes the microprocessor system implementation of the  $\overline{OE}$  and  $\overline{CE}$  controls on Intel's EPROMs. AP-72 is available from Intel's Literature Department.

The 2764 has a standby mode which reduces the power dissipation without increasing access time. The active current is 150mA, while the standby current is only 35mA, a 75% savings. The standby mode is achieved by applying a TTL-high signal to the  $\overline{CE}$  input.

The 2764 is fabricated with HMOS<sup>®</sup>-E technology, Intel's high-speed N-channel MOS Silicon Gate technology.



<sup>®</sup>HMOS is a registered trademark of Intel Corporation.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



## 2732 32K (4K x 8) UV ERASABLE PROM

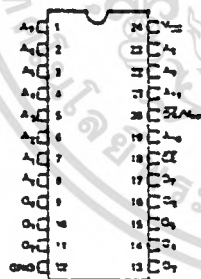
- Fast Access Times:
  - 450 ns Max. 2732
  - 550 ns Max. 2732-5
- Single +5V  $\pm$  5% Power Supply
- Output Enable for MCS-85™ and MCS-86™ Compatibility
- Low Power Dissipation:
  - 150mA Max. Active Current
  - 30mA Max. Standby Current
- Pin Compatible to Intel® 2716 EPROM
- Completely Static
- Simple Programming Requirements
  - Single Location Programming
  - Programs with One 50ms Pulse
- Three-State Output for Direct Bus Interface

The Intel® 2732 is a 32,768-bit ultraviolet erasable and electrically programmable read-only memory (EPROM). The 2732 operates from a single 5-volt power supply, has a standby mode, and features an output enable control. The total programming time for all bits is three and a half minutes. All these features make designing with the 2732 in microcomputer systems faster, easier, and more economical.

An important 2732 feature is the separate output control, Output Enable (OE) from the Chip Enable control (CE). The OE control eliminates bus contention in multiple bus microprocessor systems. Intel's Application Note AP-72 describes the microprocessor system implementation of the OE and CE controls on Intel's 2716 and 2732 EPROMs. AP-72 is available from Intel's Literature Department.

The 2732 has a standby mode which reduces the power dissipation without increasing access time. The maximum active current is 150mA, while the maximum standby current is only 30mA, an 80% savings. The standby mode is achieved by applying a TTL-high signal to the CE input.

### PIN CONFIGURATION

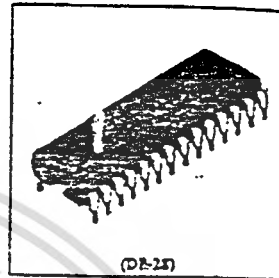


# HM6264P-10, HM6264P-12, HM6264P-15

8192-word x 8-bit High Speed Static CMOS RAM

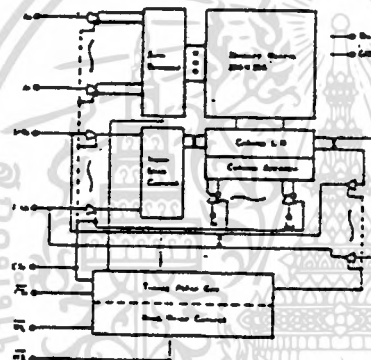
## ■ FEATURES

- Fast access Time 100ns/120ns/150ns (max.)
- Low Power Standby Standby: 0.1mW (typ.)
- Low Power Operation Operating: 200mW (typ.)
- Single +5V Supply
- Completely Static Memory..... No clock or Timing Strobe Required
- Equal Access and Cycle Time
- Common Data Input and Output, Three State Output
- Directly TTL Compatible: All Input and Output
- Standard 28pin Package Configuration
- Pin Out Compatible with 64K EPROM HM482764



(DZ-28)

## ■ BLOCK DIAGRAM



## ■ PIN ARRANGEMENT



(Top View)

## ■ ABSOLUTE MAXIMUM RATINGS

| Item                              | Symbol     | Rating          | Unit |
|-----------------------------------|------------|-----------------|------|
| Terminal Voltage *                | $V_T$      | -0.5 ** to +7.0 | V    |
| Power Dissipation                 | $P_T$      | 1.0             | W    |
| Operating Temperature             | $T_{op}$   | 0 to +70        | °C   |
| Storage Temperature               | $T_{stg}$  | -55 to +125     | °C   |
| Storage Temperature (Under Bias): | $T_{bias}$ | -10 to +85      | °C   |

\* With respect to GND. \*\* Pulse width 50ms: -3.0V

## ■ TRUTH TABLE

| WE | CS | CS | OE | Mode                      | V0 Pin | $V_{CC}$ Current | Note            |
|----|----|----|----|---------------------------|--------|------------------|-----------------|
| X  | H  | X  | X  | Not Selected (Power Down) | High Z | /SR, /SR2        | —               |
| X  | X  | L  | X  | Output Disabled           | High Z | /SR, /SR2        |                 |
| H  | L  | H  | H  | Output Disabled           | High Z | /OC, /OC2        |                 |
| H  | L  | H  | L  | Read                      | Don't  | /OC, /OC2        |                 |
| L  | L  | H  | H  | Write                     | Don't  | /OC, /OC2        | Write Cycle (1) |
| L  | L  | H  | L  | Write                     | Don't  | /OC, /OC2        | Write Cycle (2) |

• : Don't care

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

# ADC 0809 (8-Bit up Compatible A/D Converters with 8-Channel Multiplexer)

ADC0808/ADC0809



National  
Semiconductor  
Corporation

## ADC0808, ADC0809 8-Bit $\mu$ P Compatible A/D Converters with 8-Channel Multiplexer

### General Description

The ADC0808, ADC0809 data acquisition component is a monolithic CMOS device with an 8-bit analog-to-digital converter, 8-channel multiplexer and microprocessor compatible control logic. The 8-bit A/D converter uses successive approximation as the conversion technique. The converter features a high impedance chopper stabilized comparator, a 256R voltage divider with analog switch tree and a successive approximation register. The 8-channel multiplexer can directly access any of 8 single-ended analog signals.

The device eliminates the need for external zero and full-scale adjustments. Easy interfacing to microprocessors is provided by the latched and decoded multiplexer address inputs and latched TTL TRI-STATE<sup>®</sup> outputs.

The design of the ADC0808, ADC0809 has been optimized by incorporating the most desirable aspects of several A/D conversion techniques. The ADC0808, ADC0809 offers high speed, high accuracy, minimal temperature dependence, excellent long-term accuracy and repeatability, and consumes minimal power. These features make this device ideally suited to applications from process and machine control to consumer and automotive applications. For 16-channel multiplexer with common output (sample/hold port) see ADC0816 data sheet. (See AN-247 for more information.)

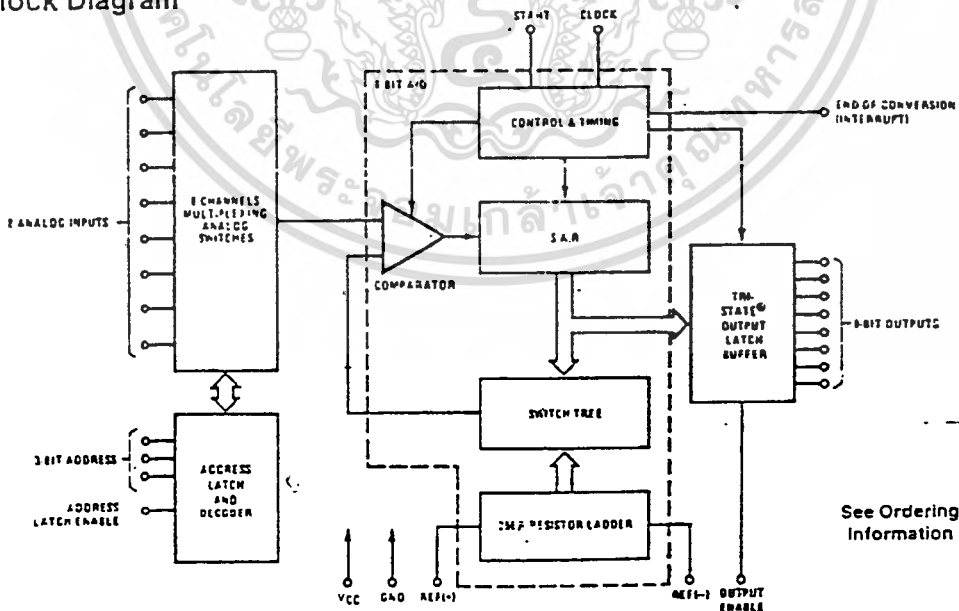
### Features

- Easy interface to all microprocessors
- Operates ratiometrically or with  $\pm V_{CC}$  or analog span adjusted voltage reference
- No zero or full-scale adjust required
- 8-channel multiplexer with address logic
- 0V to 5V input range with single 5V power supply
- Outputs meet TTL voltage level specifications
- Standard hermetic or molded 28-pin DIP package
- 28-pin molded chip carrier package

### Key Specifications

- |                          |                                       |
|--------------------------|---------------------------------------|
| ■ Resolution             | 8 Bits                                |
| ■ Total Unadjusted Error | $\pm \frac{1}{2}$ LSB and $\pm 1$ LSB |
| ■ Single Supply          | 5 V <sub>DC</sub>                     |
| ■ Low Power              | 15 mW                                 |
| ■ Conversion Time        | 100 $\mu$ s                           |

### Block Diagram



See Ordering  
Information

7.1/4/88:2-1

**Absolute Maximum Ratings** (Notes 1 & 2)

If Military/Aerospace specified devices are required, contact the National Semiconductor Sales Office/ Distributors for availability and specifications.

|                                              |                                   |
|----------------------------------------------|-----------------------------------|
| Supply Voltage (V <sub>CC</sub> ) (Note 3)   | 6.5V                              |
| Voltage at Any Pin                           | -0.3V to (V <sub>CC</sub> + 0.3V) |
| Except Control Inputs                        |                                   |
| Voltage at Control Inputs                    | -0.3V to +15V                     |
| (START, OE, CLOCK, ALE, ADD A, ADD B, ADD C) |                                   |
| Storage Temperature Range                    | -65°C to +150°C                   |
| Package Dissipation at T <sub>A</sub> = 25°C | 875 mW                            |
| Lead Temp. (Soldering, 10 seconds)           |                                   |
| Dual-In-Line Package (plastic)               | 260°C                             |
| Dual-In-Line Package (ceramic)               | 300°C                             |
| Molded Chip Carrier Package                  |                                   |
| Vapor Phase (60 seconds)                     | 215°C                             |
| Infrared (15 seconds)                        | 220°C                             |
| ESD Susceptibility (Note 11)                 | 400V                              |

**Operating Conditions** (Notes 1 & 2)

|                                   |                                                      |
|-----------------------------------|------------------------------------------------------|
| Temperature Range (Note 1)        | T <sub>MIN</sub> ≤ T <sub>A</sub> ≤ T <sub>MAX</sub> |
| ADC0808CJ                         | -55°C ≤ T <sub>A</sub> ≤ +125°C                      |
| ADC0808CCJ, ADC0808CCN,           |                                                      |
| ADC0809CCN                        | -40°C ≤ T <sub>A</sub> ≤ +85°C                       |
| ADC0809CCV, ADC0809CCV            | -40°C ≤ T <sub>A</sub> ≤ +85°C                       |
| Range of V <sub>CC</sub> (Note 1) | 4.5 V <sub>DC</sub> to 6.0 V <sub>DC</sub>           |

**Electrical Characteristics**

Converter Specifications: V<sub>CC</sub> = 5 V<sub>DC</sub> = V<sub>REF</sub> +, V<sub>REF</sub> (-) = GND, T<sub>MIN</sub> ≤ T<sub>A</sub> ≤ T<sub>MAX</sub> and f<sub>CLK</sub> = 640 kHz unless otherwise stated.

| Symbol                              | Parameter                  | Conditions                           | Min                      | Typ                | Max                      | Units           |
|-------------------------------------|----------------------------|--------------------------------------|--------------------------|--------------------|--------------------------|-----------------|
|                                     | ADC0808                    |                                      |                          |                    |                          |                 |
|                                     | Total Unadjusted Error     | 25°C                                 |                          |                    | ± 1/2                    | LSB             |
|                                     | (Note 5)                   | T <sub>MIN</sub> to T <sub>MAX</sub> |                          |                    | ± 3/4                    | LSB             |
|                                     | ADC0809                    |                                      |                          |                    |                          |                 |
|                                     | Total Unadjusted Error     | 0°C to 70°C                          |                          |                    | ± 1                      | LSB             |
|                                     | (Note 5)                   | T <sub>MIN</sub> to T <sub>MAX</sub> |                          |                    | ± 1 1/4                  | LSB             |
|                                     | Input Resistance           | From Ref(+) to Ref(-)                | 1.0                      | 2.5                |                          | kΩ              |
|                                     | Analog Input Voltage Range | (Note 4) V(+) or V(-)                | GND - 0.10               |                    | V <sub>CC</sub> - 0.10   | V <sub>DC</sub> |
| V <sub>REF</sub> (+)                | Voltage, Top of Ladder     | Measured at Ref(+)                   |                          | V <sub>CC</sub>    | V <sub>CC</sub> - 0.1    | V               |
| $\frac{V_{REF}(-) + V_{REF}(+)}{2}$ | Voltage, Center of Ladder  |                                      | V <sub>CC</sub> /2 - 0.1 | V <sub>CC</sub> /2 | V <sub>CC</sub> /2 + 0.1 | V               |
| V <sub>REF</sub> (-)                | Voltage, Bottom of Ladder  | Measured at Ref(-)                   | -0.1                     | 0                  |                          | V               |
| I <sub>IN</sub>                     | Comparator Input Current   | f <sub>C</sub> = 640 kHz, (Note 6)   | 2                        | ± 0.5              | 2                        | μA              |

**Electrical Characteristics**

Digital Levels and DC Specifications: ADC0808CJ 4.5V ≤ V<sub>CC</sub> ≤ 5.5V, -55°C ≤ T<sub>A</sub> ≤ +125°C unless otherwise noted  
 ADC0808CCJ, ADC0808CCN, ADC0808CCV, ADC0809CCN and ADC0809CCV, 4.75V ≤ V<sub>CC</sub> ≤ 5.25V, -40°C ≤ T<sub>A</sub> ≤ +85°C unless otherwise noted

| Symbol                    | Parameter                   | Conditions                                                           | Min  | Typ | Max | Units |
|---------------------------|-----------------------------|----------------------------------------------------------------------|------|-----|-----|-------|
| <b>ANALOG MULTIPLEXER</b> |                             |                                                                      |      |     |     |       |
| I <sub>OFF</sub> (-)      | OFF Channel Leakage Current | V <sub>CC</sub> = 5V, V <sub>IN</sub> = 5V,<br>T <sub>A</sub> = 25°C |      | 10  | 200 | nA    |
|                           | (                           | T <sub>MIN</sub> to T <sub>MAX</sub>                                 |      |     | 1.0 | μA    |
| I <sub>OFF</sub> (-)      | OFF Channel Leakage Current | V <sub>CC</sub> = 5V, V <sub>IN</sub> = 0,<br>T <sub>A</sub> = 25°C  | -200 | -10 |     | nA    |
|                           |                             | T <sub>MIN</sub> to T <sub>MAX</sub>                                 | -1.0 |     |     | μA    |

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
 ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Electrical Characteristics (Continued)

Digital Levels and DC Specifications: ADC0808CJ  $4.5V \leq V_{CC} \leq 5.5V$ ,  $-55^{\circ}C \leq T_A \leq +125^{\circ}C$  unless otherwise noted  
ADC0808CCJ, ADC0808CCN, ADC0808CCV, ADC0809CCN and ADC0809CCV,  $4.75 \leq V_{CC} \leq 5.25V$ ,  $-40^{\circ}C \leq T_A \leq +85^{\circ}C$  unless otherwise noted

| Symbol                           | Parameter                                         | Conditions                 | Min          | Typ | Max  | Units              |
|----------------------------------|---------------------------------------------------|----------------------------|--------------|-----|------|--------------------|
| CONTROL INPUTS                   |                                                   |                            |              |     |      |                    |
| $V_{IN(1)}$                      | Logical "1" Input Voltage                         |                            | $V_{CC}-1.5$ |     |      | V                  |
| $V_{IN(0)}$                      | Logical "0" Input Voltage                         |                            |              |     | 1.5  | V                  |
| $I_{IN(1)}$                      | Logical "1" Input Current<br>(The Control Inputs) | $V_{IN} = 15V$             |              |     | 1.0  | $\mu A$            |
| $I_{IN(0)}$                      | Logical "0" Input Current<br>(The Control Inputs) | $V_{IN} = 0$               | -1.0         |     |      | $\mu A$            |
| $I_{CC}$                         | Supply Current                                    | $f_{CLK} = 640\text{ kHz}$ |              | 0.3 | 3.0  | mA                 |
| DATA OUTPUTS AND EOC (INTERRUPT) |                                                   |                            |              |     |      |                    |
| $V_{OUT(1)}$                     | Logical "1" Output Voltage                        | $I_O = -360\text{ }\mu A$  | $V_{CC}-0.4$ |     |      | V                  |
| $V_{OUT(0)}$                     | Logical "0" Output Voltage                        | $I_O = 1.6\text{ mA}$      |              |     | 0.45 | V                  |
| $V_{OUT(0)}$                     | Logical "0" Output Voltage EOC                    | $I_O = 1.2\text{ mA}$      |              |     | 0.45 | V                  |
| $I_{OUT}$                        | TRI-STATE Output Current                          | $V_O = 5V$<br>$V_O = 0$    | -3           |     | 3    | $\mu A$<br>$\mu A$ |

Electrical Characteristics

Timing Specifications  $V_{CC} = V_{REF(+)} = 5V$ ,  $V_{REF(-)} = GND$ ,  $t_r = t_f = 20\text{ ns}$  and  $T_A = 25^{\circ}C$  unless otherwise noted.

| Symbol           | Parameter                         | Conditions                                    | Min | Typ | Max                    | Units            |
|------------------|-----------------------------------|-----------------------------------------------|-----|-----|------------------------|------------------|
| $t_{WS}$         | Minimum Start Pulse Width         | (Figure 5)                                    |     | 100 | 200                    | ns               |
| $t_{WALE}$       | Minimum ALE Pulse Width           | (Figure 5)                                    |     | 100 | 200                    | ns               |
| $t_s$            | Minimum Address Set-Up Time       | (Figure 5)                                    |     | 25  | 50                     | ns               |
| $t_H$            | Minimum Address Hold Time         | (Figure 5)                                    |     | 25  | 50                     | ns               |
| $t_D$            | Analog MUX Delay Time<br>From ALE | $R_S = 0\Omega$ (Figure 5)                    |     | 1   | 2.5                    | $\mu S$          |
| $t_{HL}, t_{HO}$ | OE Control to O Logic State       | $C_L = 50\text{ pF}$ , $R_L = 10k$ (Figure 8) |     | 125 | 250                    | ns               |
| $t_{HL}, t_{OH}$ | OE Control to Hi-Z                | $C_L = 10\text{ pF}$ , $R_L = 10k$ (Figure 8) |     | 125 | 250                    | ns               |
| $t_c$            | Conversion Time                   | $f_c = 640\text{ kHz}$ , (Figure 5) (Note 7)  | 90  | 100 | 116                    | $\mu S$          |
| $f_c$            | Clock Frequency                   |                                               | 10  | 640 | 1280                   | kHz              |
| $t_{EOC}$        | EOC Delay Time                    | (Figure 5)                                    | 0   |     | $6 \pm 2\text{ }\mu S$ | Clock<br>Periods |
| $C_{IN}$         | Input Capacitance                 | At Control Inputs                             |     | 10  | 15                     | pF               |
| $C_{OUT}$        | TRI-STATE Output<br>Capacitance   | At TRI-STATE Outputs, (Note 12)               |     | 10  | 15                     | pF               |

Note 1: Absolute Maximum Ratings indicate limits beyond which damage to the device may occur. DC and AC electrical specifications do not apply when operating the device beyond its specified operating conditions.

Note 2: All voltages are measured with respect to GND, unless otherwise specified.

Note 3: A zener diode exists, internally, from  $V_{CC}$  to GND and has a typical breakdown voltage of  $7 V_{CC}$ .

Note 4: Two on-chip diodes are tied to each analog input which will forward conduct for analog input voltages one diode drop below ground or one diode drop greater than the  $V_{CC}$  supply. The spec allows 100 mV forward bias of either diode. This means that as long as the analog  $V_{IN}$  does not exceed the supply voltage by more than 100 mV, the output code will be correct. To achieve an absolute  $0V_{CC}$  to  $5V_{CC}$  input voltage range will therefore require a minimum supply voltage of 4.900V over temperature variations, initial tolerance and loading.

Note 5: Total unadjusted error includes offset, full-scale, linearity, and multiplexer errors. See Figure 3. None of these A/Ds requires a zero or full-scale adjust. However, if an absolute zero code is desired for an analog input other than 0V, or if a narrow full-scale span exists (for example: 0.5V to 4.5V full-scale) the reference voltages can be adjusted to achieve this. See Figure 13.

Note 6: Comparator input current is a bias current into or out of the chopper stabilized comparator. The bias current varies directly with clock frequency and has little temperature dependence (Figure 6). See paragraph 4.0.

Note 7: The outputs of the data register are updated one clock cycle before the rising edge of EOC.

Note 8: Human body model, 100 pF discharged through a 1.5 k $\Omega$  resistor.

## Functional Description

**Multiplexer.** The device contains an 8-channel single-ended analog signal multiplexer. A particular input channel is selected by using the address decoder. Table I shows the input states for the address lines to select any channel. The address is latched into the decoder on the low-to-high transition of the address latch enable signal.

TABLE I

| SELECTED<br>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 |

## CONVERTER CHARACTERISTICS

### The Converter

The heart of this single chip data acquisition system is its 8-bit analog-to-digital converter. The converter is designed

to give fast, accurate, and repeatable conversions over a wide range of temperatures. The converter is partitioned into 3 major sections; the 256R ladder network, the successive approximation register, and the comparator. The converter's digital outputs are positive true.

The 256R ladder network approach (Figure 1) was chosen over the conventional R/2R ladder because of its inherent monotonicity, which guarantees no missing digital codes. Monotonicity is particularly important in closed loop feedback control systems. A non-monotonic relationship can cause oscillations that will be catastrophic for the system. Additionally, the 256R network does not cause load variations on the reference voltage.

The bottom resistor and the top resistor of the ladder network in Figure 1 are not the same value as the remainder of the network. The difference in these resistors causes the output characteristic to be symmetrical with the zero and full-scale points of the transfer curve. The first output transition occurs when the analog signal has reached  $-\frac{1}{2}$  LSB and succeeding output transitions occur every 1 LSB later up to full-scale.

The successive approximation register (SAR) performs 8 iterations to approximate the input voltage. For any SAR type converter, n-iterations are required for an n-bit converter. Figure 2 shows a typical example of a 3-bit converter. In the ADC0808, ADC0809, the approximation technique is extended to 8 bits using the 256R network.

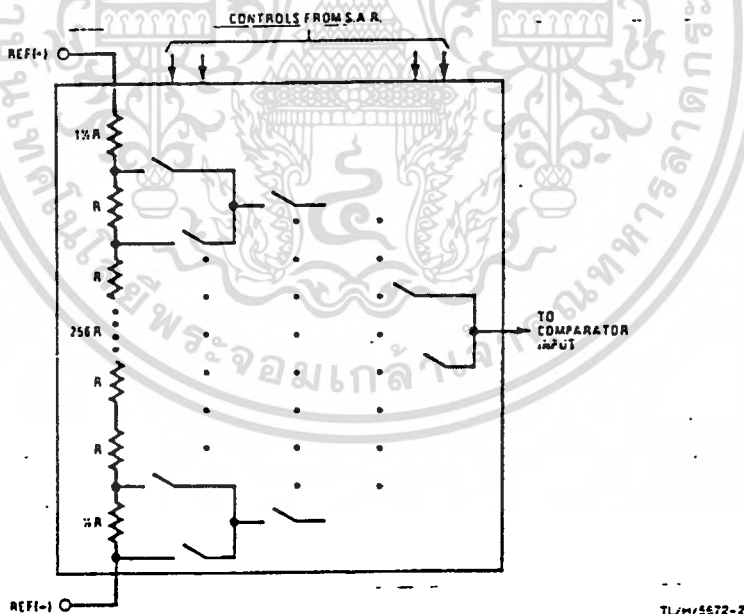


FIGURE 1. Resistor Ladder and Switch Tree

### Functional Description (Continued)

The A/D converter's successive approximation register (SAR) is reset on the positive edge of the start conversion (SC) pulse. The conversion is begun on the falling edge of the start conversion pulse. A conversion in process will be interrupted by receipt of a new start conversion pulse. Continuous conversion may be accomplished by tying the end-of-conversion (EOC) output to the SC input. If used in this mode, an external start conversion pulse should be applied after power up. End-of-conversion will go low between 0 and 8 clock pulses after the rising edge of start conversion. The most important section of the A/D converter is the comparator. It is this section which is responsible for the ultimate accuracy of the entire converter. It is also the

comparator drift which has the greatest influence on the repeatability of the device. A chopper-stabilized comparator provides the most effective method of satisfying all the converter requirements.

The chopper-stabilized comparator converts the DC input signal into an AC signal. This signal is then fed through a high gain AC amplifier and has the DC level restored. This technique limits the drift component of the amplifier since the drift is a DC component which is not passed by the AC amplifier. This makes the entire A/D converter extremely insensitive to temperature, long term drift and input offset errors.

Figure 4 shows a typical error curve for the ADC0808 as measured using the procedures outlined in AN-179.

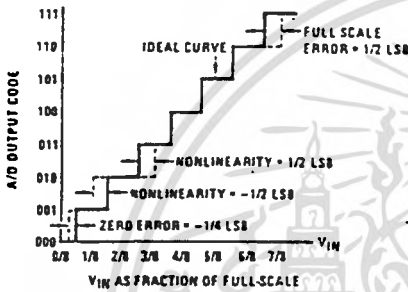


FIGURE 2. 3-Bit A/D Transfer Curve

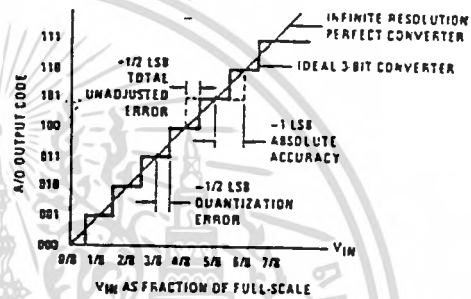
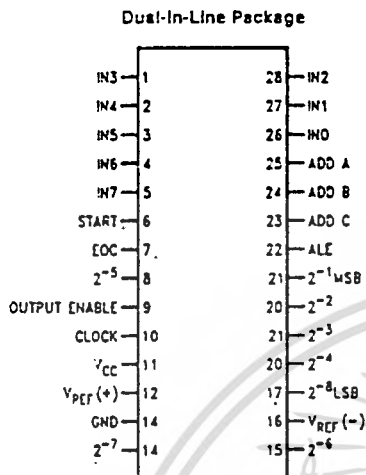


FIGURE 3. 3-Bit A/D Absolute Accuracy Curve



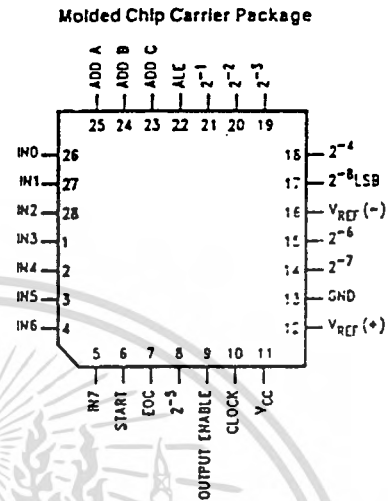
FIGURE 4. Typical Error Curve

# Connection Diagrams



TL/M/5672-11

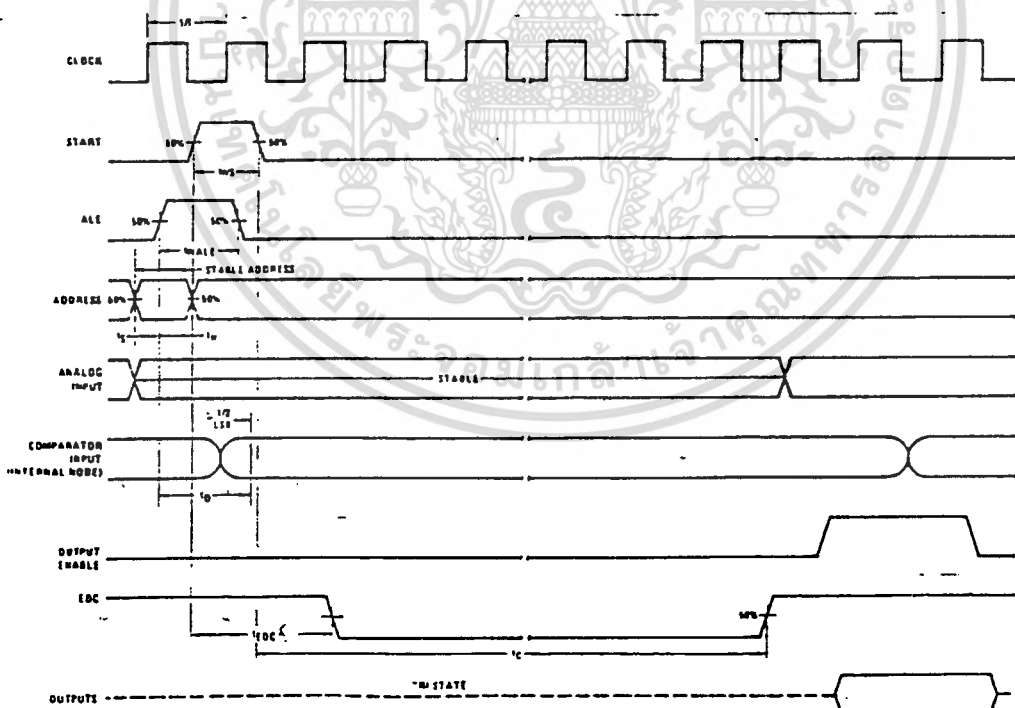
Order Number ADC0808CCN, ADC0809CCN,  
ADC0808CCJ or ADC0808CJ  
See NS Package J28A or N28A



TL/M/5672-12

Order Number ADC0808CCV or ADC0809CCV  
See NS Package V28A

## Timing Diagram



TL/M/5672-4

FIGURE 5

## Typical Performance Characteristics

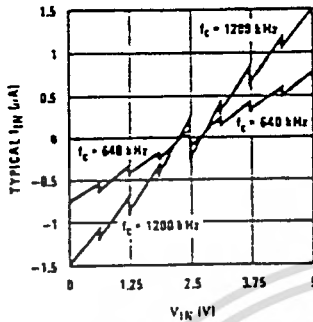


FIGURE 6. Comparator  $I_{IN}$  vs  $V_{IN}$   
( $V_{CC} = V_{REF} = 5V$ )

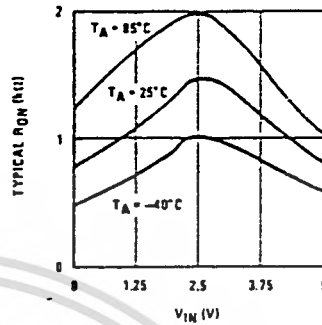


FIGURE 7. Multiplexer  $R_{ON}$  vs  $V_{IN}$   
( $V_{CC} = V_{REF} = 5V$ )

TL/—542—5

## TRI-STATE Test Circuits and Timing Diagrams

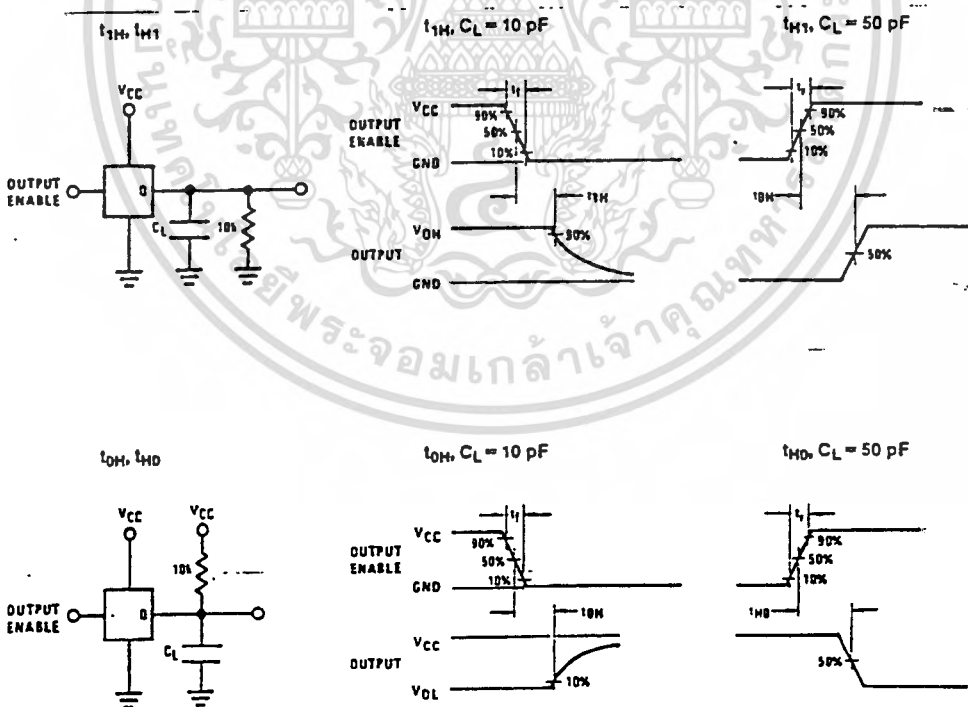


FIGURE 8

TL/—542—6

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

# Applications Information

## OPERATION

### 1.0 RATIOMETRIC CONVERSION

The ADC0808, ADC0809 is designed as a complete Data Acquisition System (DAS) for ratiometric conversion systems. In ratiometric systems, the physical variable being measured is expressed as a percentage of full-scale which is not necessarily related to an absolute standard. The voltage input to the ADC0808 is expressed by the equation

$$\frac{V_{IN} - V_Z}{V_{FS} - V_Z} = \frac{D_x}{D_{MAX} - D_{MIN}} \quad (1)$$

$V_{IN}$  = Input voltage into the ADC0808  
 $V_{FS}$  = Full-scale voltage  
 $V_Z$  = Zero voltage  
 $D_x$  = Data point being measured  
 $D_{MAX}$  = Maximum data limit  
 $D_{MIN}$  = Minimum data limit

A good example of a ratiometric transducer is a potentiometer used as a position sensor. The position of the wiper is directly proportional to the output voltage which is a ratio of the full-scale voltage across it. Since the data is represented as a proportion of full-scale, reference requirements are greatly reduced, eliminating a large source of error and cost for many applications. A major advantage of the ADC0808, ADC0809 is that the input voltage range is equal to the supply range so the transducers can be connected directly across the supply and their outputs connected directly into the multiplexer inputs. (Figure 9).

Ratiometric transducers such as potentiometers, strain gauges, thermistor bridges, pressure transducers, etc., are suitable for measuring proportional relationships; however, many types of measurements must be referred to an absolute standard such as voltage or current. This means a system reference must be used which relates the full-scale voltage to the standard volt. For example, if  $V_{CC} = V_{REF} = 5.12V$ , then the full-scale range is divided into 256 standard steps. The smallest standard step is 1 LSB which is then 20 mV.

### 2.0 RESISTOR LADDER LIMITATIONS

The voltages from the resistor ladder are compared to the selected into 8 times in a conversion. These voltages are coupled to the comparator via an analog switch tree which is referenced to the supply. The voltages at the top, center and bottom of the ladder must be controlled to maintain proper operation.

The top of the ladder,  $REF(+)$ , should not be more positive than the supply, and the bottom of the ladder,  $REF(-)$ , should not be more negative than ground. The center of the ladder voltage must also be near the center of the supply because the analog switch tree changes from N-channel switches to P-channel switches. These limitations are automatically satisfied in ratiometric systems and can be easily met in ground referenced systems.

Figure 10 shows a ground referenced system with a separate supply and reference. In this system, the supply must be trimmed to match the reference voltage. For instance, if a 5.12V is used, the supply should be adjusted to the same voltage within 0.1V.

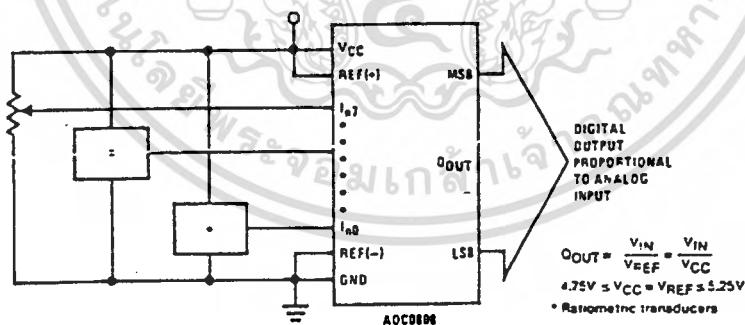


FIGURE 9. Ratiometric Conversion System

## Applications Information (Continued)

The ADC0806 needs less than a milliamp of supply current so developing the supply from the reference is readily accomplished. In Figure 11 a ground referenced system is shown which generates the supply from the reference. The buffer shown can be an op amp of sufficient drive to supply the milliamp of supply current and the desired bus drive, or if a capacitive bus is driven by the outputs a large capacitor will supply the transient supply current as seen in Figure 12. The LM301 is overcompensated to insure stability when loaded by the 10  $\mu$ F output capacitor.

The top and bottom ladder voltages cannot exceed  $V_{CC}$  and ground, respectively, but they can be symmetrically less than  $V_{CC}$  and greater than ground. The center of the ladder voltage should always be near the center of the supply. The sensitivity of the converter can be increased, (i.e., size of the LSE steps decreased) by using a symmetrical reference system. In Figure 13, a 2.5V reference is symmetrically centered about  $V_{CC}/2$  since the same current flows in identical resistors. This system with a 2.5V reference allows the LSB bit to be half the size of a 5V reference system.

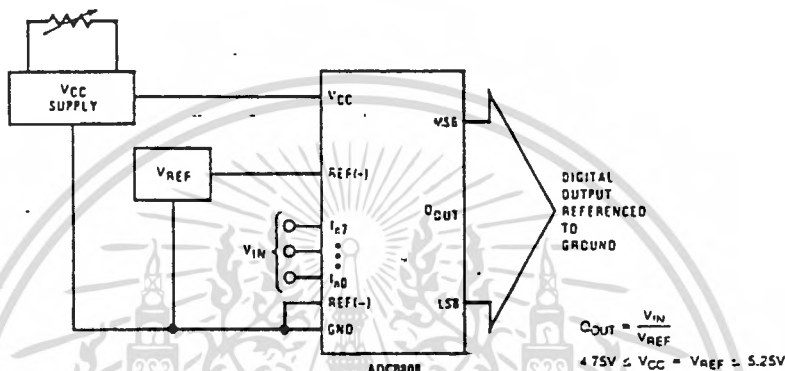


FIGURE 10. Ground Referenced Conversion System Using Trimmed Supply

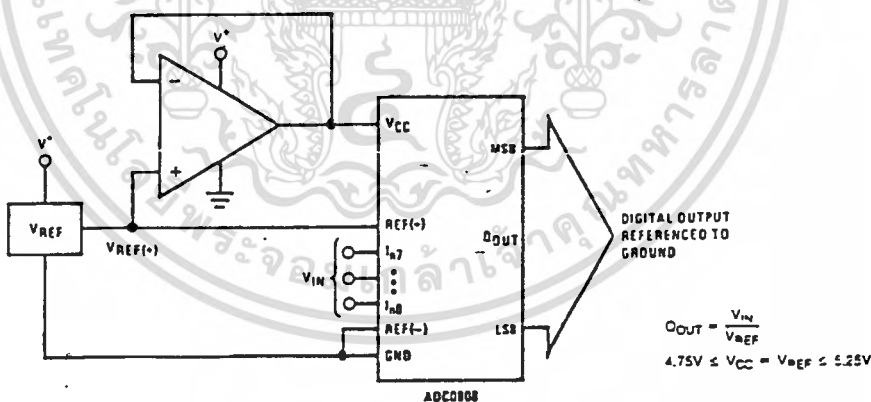


FIGURE 11: Ground Referenced Conversion System with Reference Generating  $V_{CC}$  Supply

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## Applications Information (Continued)

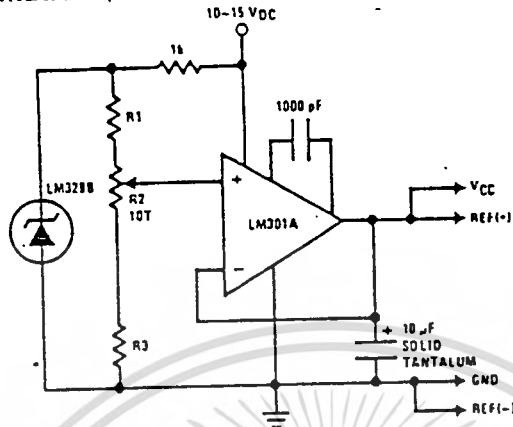


FIGURE 12. Typical Reference and Supply Circuit

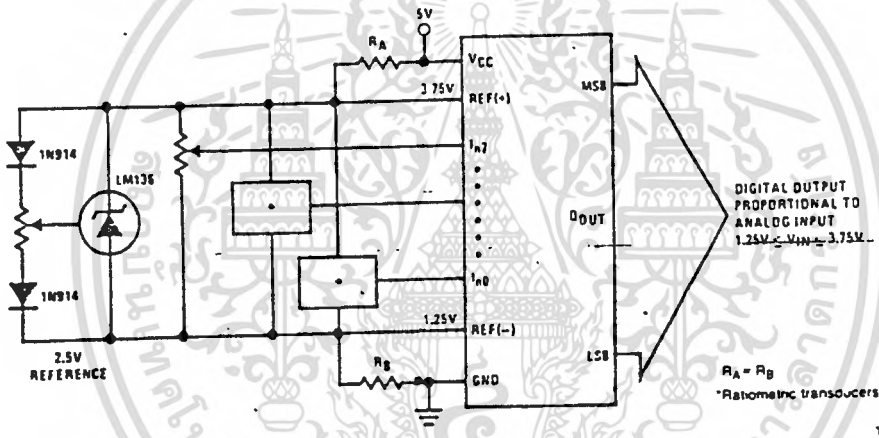


FIGURE 13. Symmetrically Centered Reference

### 3.0 CONVERTER EQUATIONS

The transition between adjacent codes  $N$  and  $N + 1$  is given by:

$$V_{IN} = \left\{ (V_{REF(+)} - V_{REF(-)}) \left[ \frac{N}{256} + \frac{1}{512} \right] \pm V_{TUE} \right\} - V_{REF(-)} \quad (2)$$

The center of an output code  $N$  is given by:

$$V_{IN} \left\{ (V_{REF(+)} - V_{REF(-)}) \left[ \frac{N}{256} \right] \pm V_{TUE} \right\} + V_{REF(-)} \quad (3)$$

The output code  $N$  for an arbitrary input are the integers within the range:

$$N = \frac{V_{IN} - V_{REF(-)}}{V_{REF(+)} - V_{REF(-)}} \times 256 \pm \text{Absolute Accuracy} \quad (4)$$

where:  $V_{IN}$  = Voltage at comparator input

$V_{REF(+)}$  = Voltage at Ref(+)

$V_{REF(-)}$  = Voltage at Ref(-)

$V_{TUE}$  = Total unadjusted error voltage (typically

$V_{REF(+)} \pm 512$ )

### 4.0 ANALOG COMPARATOR INPUTS

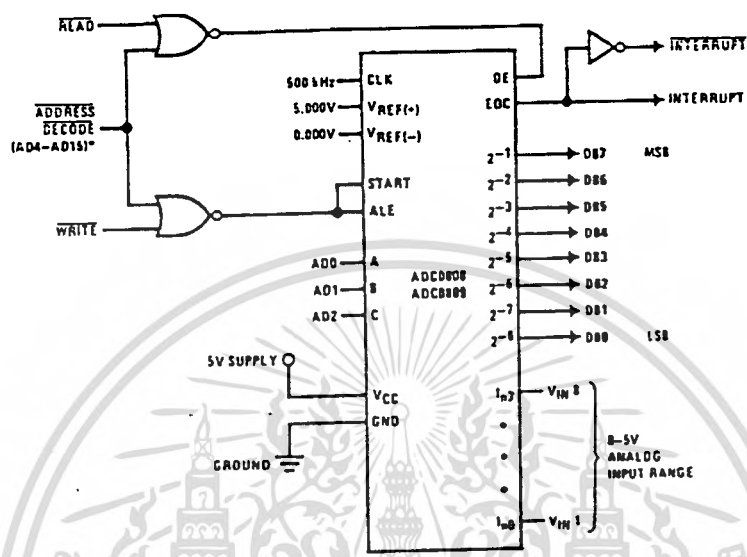
The dynamic comparator input current is caused by the periodic switching of on-chip stray capacitances. These are connected alternately to the output of the resistor ladder/switch tree network and to the comparator input as part of the operation of the chopper stabilized comparator.

The average value of the comparator input current varies directly with clock frequency and with  $V_{IN}$  as shown in Figure 6.

If no filter capacitors are used at the analog inputs and the signal source impedances are low, the comparator input current should not introduce converter errors, as the transient created by the capacitance discharge will die out before the comparator output is strobed.

If input filter capacitors are desired for noise reduction and signal conditioning they will tend to average out the dynamic comparator input current. It will then take on the characteristics of a DC bias current whose effect can be predicted conventionally.

Typical Application



TL/M/5672-10

\*Address latches needed for 8085 and SC/MP interfacing the ADC0808 to a microprocessor

MICROPROCESSOR INTERFACE TABLE

| PROCESSOR | READ       | WRITE     | INTERRUPT (COMMENT)            |
|-----------|------------|-----------|--------------------------------|
| 8080      | MFMR       | MEMW      | INTR (Thru RST Circuit)        |
| 8085      | RD         | WR        | INTR (Thru RST Circuit)        |
| Z-80      | RD         | WR        | INT (Thru RST Circuit, Mode 0) |
| SC/MP     | NRDS       | NWDS      | SA (Thru Sense A)              |
| 6800      | VMA*φ2•R/W | VMA*φ•R/W | IRQA or IRQB (Thru PIA)        |

Ordering Information

| TEMPERATURE RANGE |                    | - 40°C to + 85°C |                        |                  | - 55°C to + 125°C |
|-------------------|--------------------|------------------|------------------------|------------------|-------------------|
| Error             | = ½ LSB Unadjusted | ADC0808CCN       | ADC0808CCV             | ADC0808CCJ       | ADC0808CJ         |
|                   | = 1 LSB Unadjusted | ADC0809CCN       | ADC0809CCV             |                  |                   |
| Package Outline   |                    | N28A Molded DIP  | V28A Molded Chp Carier | J28A Ceramic DIP | J28A Ceramic DIP  |

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

# MOTOROLA SEMICONDUCTOR TECHNICAL DATA

## 6-Pin DIP Optoisolators Transistor Output

These devices consist of a gallium arsenide infrared emitting diode optically coupled to a monolithic silicon phototransistor detector.

- Convenient Plastic Dual-In-Line Package
- High Current Transfer Ratio — 100% Minimum at Spec Conditions
- Guaranteed Switching Speeds
- High Input-Output Isolation Guaranteed — 7500 Volts Peak
- UL Recognized. File Number E54915 
- VDE approved per standard G883 6.80 (Certificate number 41853), with additional approval to DIN IEC380 VDE0906, IEC435 VDE0805, IEC65 VDE0860, VDE0110b, covering all other standards with equal or less stringent requirements, including IEC204/VDE0113, VDE0160, VDE0832, VDE0833, etc. 
- Meets or Exceeds All JEDEC Registered Specifications
- Special lead form available (add suffix "T" to part number) which satisfies VDE0883/6.80 requirement for 8 mm minimum creepage distance between input and output solder pads.
- Various lead form options available. Consult "Optoisolator Lead Form Options" data sheet for details.

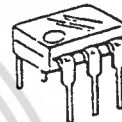
MAXIMUM RATINGS ( $T_A = 25^\circ\text{C}$  unless otherwise noted)

| Rating                                                                                                                           | Symbol    | Value       | Unit        |
|----------------------------------------------------------------------------------------------------------------------------------|-----------|-------------|-------------|
| <b>INPUT LED</b>                                                                                                                 |           |             |             |
| Reverse Voltage                                                                                                                  | $V_R$     | 6           | Volts       |
| Forward Current — Continuous                                                                                                     | $I_F$     | 60          | mA          |
| LED Power Dissipation in $T_A = 25^\circ\text{C}$<br>with Negligible Power in Output Detector<br>Derate above $25^\circ\text{C}$ | $P_D$     | 120<br>1.41 | mW<br>mW/°C |
| <b>OUTPUT TRANSISTOR</b>                                                                                                         |           |             |             |
| Collector-Emitter Voltage                                                                                                        | $V_{CE0}$ | 30          | Volts       |
| Emitter-Base Voltage                                                                                                             | $V_{EB0}$ | 7           | Volts       |
| Collector-Base Voltage                                                                                                           | $V_{CB0}$ | 70          | Volts       |
| Collector Current — Continuous                                                                                                   | $I_C$     | 150         | mA          |
| Detector Power Dissipation in $T_A = 25^\circ\text{C}$<br>with Negligible Power in Input LED<br>Derate above $25^\circ\text{C}$  | $P_D$     | 150<br>1.76 | mW<br>mW/°C |
| <b>TOTAL DEVICE</b>                                                                                                              |           |             |             |
| Isolation Source Voltage (1)<br>(Peak ac Voltage, 60 Hz, 1 sec Duration)                                                         | $V_{ISO}$ | 7500        | Vac         |
| Total Device Power Dissipation in $T_A = 25^\circ\text{C}$<br>Derate above $25^\circ\text{C}$                                    | $P_D$     | 250<br>2.94 | mW<br>mW/°C |
| Ambient Operating Temperature Range                                                                                              | $T_A$     | -55 to +100 | °C          |
| Storage Temperature Range                                                                                                        | $T_{stg}$ | -55 to +150 | °C          |
| Soldering Temperature (10 seconds, 1"16" from case)                                                                              | —         | 260         | °C          |

(1) Isolation surge voltage is an internal device dielectric breakdown rating.  
For this test, Pins 1 and 2 are common, and Pins 4, 5 and 6 are common

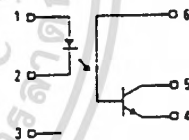
4N35  
4N36  
4N37

6-PIN DIP  
OPTOISOLATORS  
TRANSISTOR  
OUTPUT



CASE 730A-02  
PLASTIC

### SCHEMATIC



- 1 LED ANODE
- 2 LED CATHODE
- 3 NC
- 4 EMITTER
- 5 COLLECTOR
- 6 BASE

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4N35, 4N36, 4N37

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

| Characteristic                                                                                                                                         | Symbol               | Min               | Typ                 | Max               | Unit     |
|--------------------------------------------------------------------------------------------------------------------------------------------------------|----------------------|-------------------|---------------------|-------------------|----------|
| INPUT LED                                                                                                                                              |                      |                   |                     |                   |          |
| Forward Voltage (I <sub>F</sub> = 10 mA)<br>T <sub>A</sub> = 25°C<br>T <sub>A</sub> = -55°C<br>T <sub>A</sub> = 100°C                                  | V <sub>F</sub>       | 0.8<br>0.9<br>0.7 | 1.15<br>1.3<br>1.05 | 1.5<br>1.7<br>1.4 | V        |
| Reverse Leakage Current (V <sub>R</sub> = 6 V)                                                                                                         | I <sub>R</sub>       | —                 | —                   | 10                | μA       |
| Capacitance (V = 0 V, f = 1 MHz)                                                                                                                       | C <sub>J</sub>       | —                 | 18                  | —                 | pF       |
| OUTPUT TRANSISTOR                                                                                                                                      |                      |                   |                     |                   |          |
| Collector-Emitter Dark Current (V <sub>CE</sub> = 10 V, T <sub>A</sub> = 25°C)<br>(V <sub>CE</sub> = 30 V, T <sub>A</sub> = 100°C)                     | I <sub>CEO</sub>     | —                 | 1<br>50             | —<br>500          | nA<br>μA |
| Collector-Base Dark Current (V <sub>CB</sub> = 10 V)<br>T <sub>A</sub> = 25°C<br>T <sub>A</sub> = 100°C                                                | I <sub>CBO</sub>     | —                 | 0.2<br>100          | 20<br>—           | nA       |
| Collector-Emitter Breakdown Voltage (I <sub>C</sub> = 1 mA)                                                                                            | V <sub>BRICEO</sub>  | 30                | 45                  | —                 | V        |
| Collector-Base Breakdown Voltage (I <sub>C</sub> = 100 μA)                                                                                             | V <sub>BRICBO</sub>  | 70                | 100                 | —                 | V        |
| Emitter-Base Breakdown Voltage (I <sub>E</sub> = 100 μA)                                                                                               | V <sub>BRIEBO</sub>  | 7                 | 7.8                 | —                 | V        |
| DC Current Gain (I <sub>C</sub> = 2 mA, V <sub>CE</sub> = 5 V)                                                                                         | h <sub>FE</sub>      | —                 | 400                 | —                 | —        |
| Collector-Emitter Capacitance (f = 1 MHz, V <sub>CE</sub> = 0)                                                                                         | C <sub>CE</sub>      | —                 | 7                   | —                 | pF       |
| Collector-Base Capacitance (f = 1 MHz, V <sub>CB</sub> = 0)                                                                                            | C <sub>CB</sub>      | —                 | 19                  | —                 | pF       |
| Emitter-Base Capacitance (f = 1 MHz, V <sub>EB</sub> = 0)                                                                                              | C <sub>EB</sub>      | —                 | 9                   | —                 | pF       |
| COUPLED                                                                                                                                                |                      |                   |                     |                   |          |
| Output Collector Current (I <sub>F</sub> = 10 mA, V <sub>CE</sub> = 10 V)<br>T <sub>A</sub> = 25°C<br>T <sub>A</sub> = -55°C<br>T <sub>A</sub> = 100°C | I <sub>C</sub>       | 10<br>4<br>4      | 30<br>—<br>—        | —<br>—<br>—       | mA       |
| Collector-Emitter Saturation Voltage (I <sub>C</sub> = 0.5 mA, I <sub>F</sub> = 10 mA)                                                                 | V <sub>CE(sat)</sub> | —                 | 0.14                | 0.3               | V        |
| Turn-On Time                                                                                                                                           | I <sub>on</sub>      | —                 | 7.5                 | 10                | μs       |
| Turn-Off Time                                                                                                                                          | I <sub>off</sub>     | —                 | 5.7                 | 10                | μs       |
| Rise Time                                                                                                                                              | t <sub>r</sub>       | —                 | 3.2                 | —                 | μs       |
| Fall Time                                                                                                                                              | t <sub>f</sub>       | —                 | 4.7                 | —                 | μs       |
| Isolation Voltage (f = 60 Hz, t = 1 sec)                                                                                                               | V <sub>ISO</sub>     | 7500              | —                   | —                 | Vac(pk)  |
| Isolation Current (V <sub>I/O</sub> = 3550 Vpk)<br>(V <sub>I/O</sub> = 2500 Vpk)<br>(V <sub>I/O</sub> = 1500 Vpk)                                      | I <sub>ISO</sub>     | —<br>—<br>—       | —<br>—<br>8         | 100<br>100<br>100 | μA       |
| Isolation Resistance (V = 500 V)                                                                                                                       | R <sub>ISO</sub>     | 10 <sup>11</sup>  | —                   | —                 | Ω        |
| Isolation Capacitance (V = 0 V, f = 1 MHz)                                                                                                             | C <sub>ISO</sub>     | —                 | 0.2                 | 2                 | pF       |

TYPICAL CHARACTERISTICS

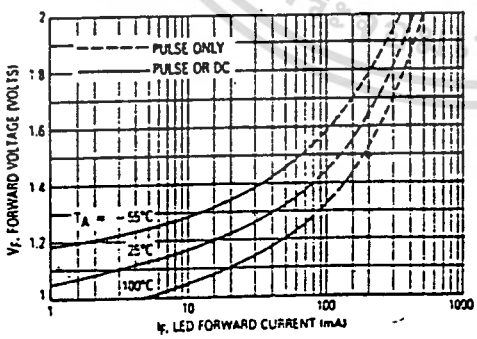


Figure 1. LED Forward Voltage versus Forward Current

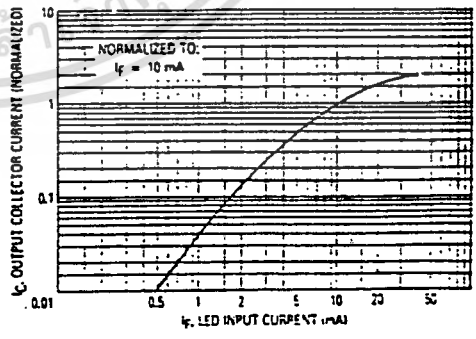


Figure 2. Output Current versus Input Current

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4N35, 4N36, 4N37

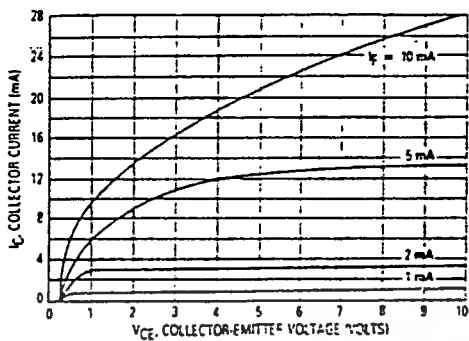


Figure 3. Collector Current versus Collector-Emitter Voltage

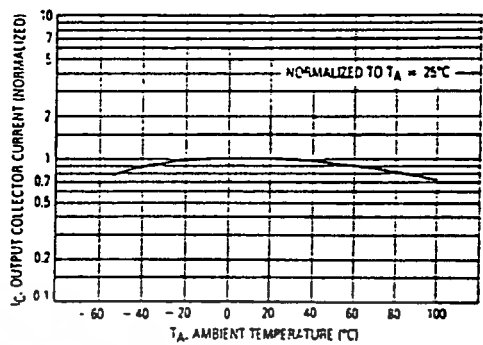


Figure 4. Output Current versus Ambient Temperature

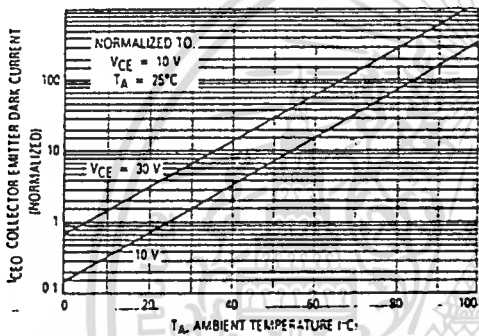


Figure 5. Dark Current versus Ambient Temperature

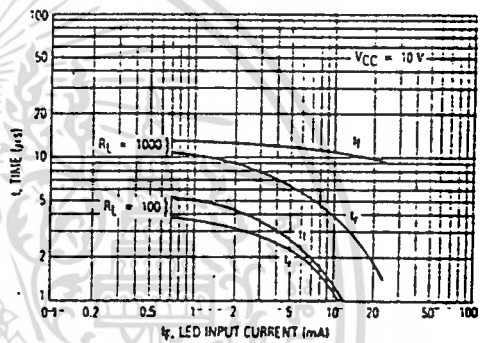


Figure 6. Rise and Fall Times

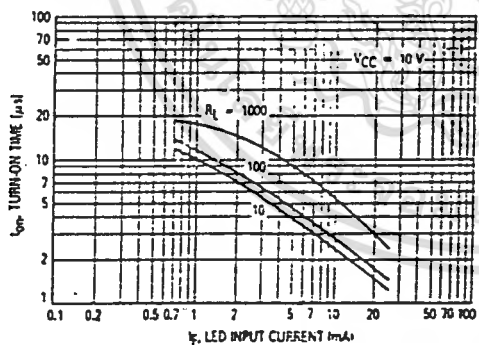


Figure 7. Turn-On Switching Times

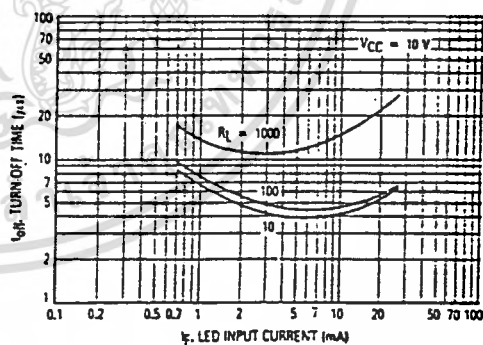


Figure 8. Turn-Off Switching Times

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4N35, 4N36, 4N37

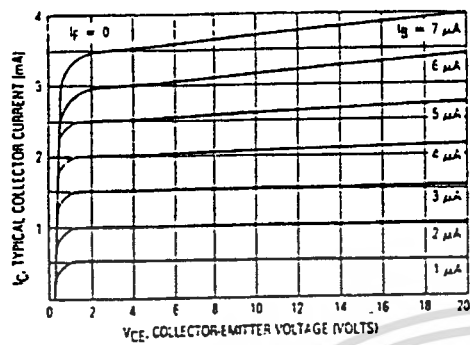


Figure 9. DC Current Gain (Detector Only)

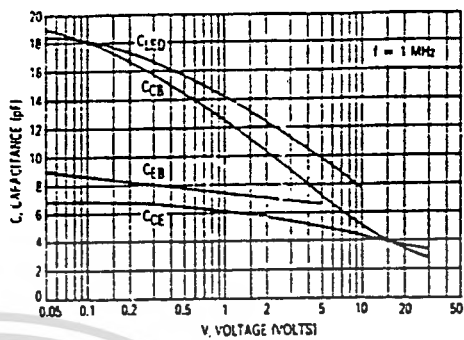


Figure 10. Capacitances versus Voltage

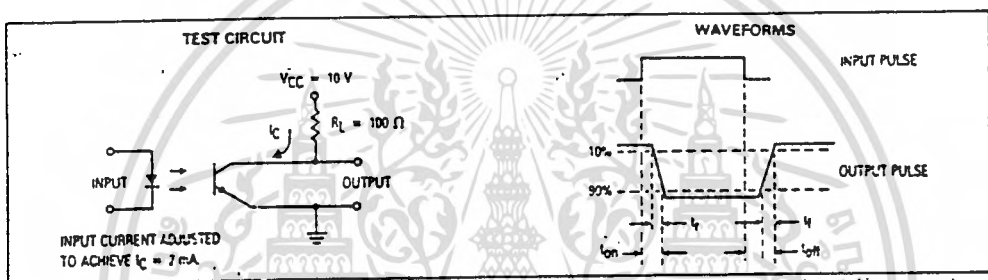
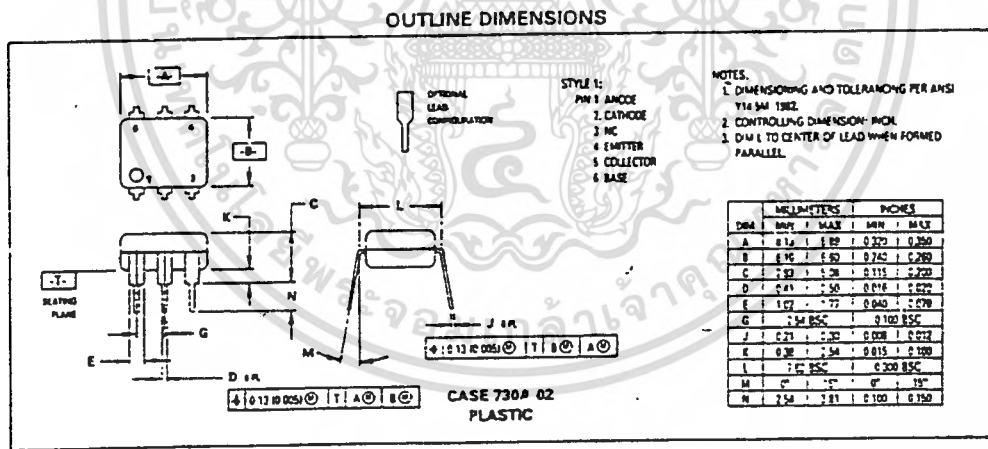


Figure 11. Switching Times



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้