การใช้ PC ควบคุม วงจรเพาว์เวอร์ซัพพลาย, วงจรชาร์จแบตเตอรี่ และ ดิซี.โวลท์มิเตอร์

นาย กิตติ โกสิทธิ์ 34-131201

นาย ชัชชัย อินทรศักดิ์ 34-131206

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาอุตสาหกรรมศาสตร์บัณฑิต

ภาควิชาเทคนิคอุตสาหกรรม คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหาร ลาดกระบัง

ปีการศึกษา 2536

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

033247

PC CONTROL POWER SUPPLY CIRCUIT, CHARGE BATTERY CIRCUIT AND DC. VOLTMETER

Mr. Kitti Kosit 34-131201

Mr. Chatchai Intarasak 34-131206

Project Report Submitted in Partial Fulfillment of the Requirement

for the Bachelor's Degree

Department of Industril technology

Faculty of Engineering

King Mongkut's Institute of Technology

1993

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หัวข้อปริญญานิพนธ์	การใช้ PC ควบคุมวงจรเพาว์เวอร์ซัพพลาย, วงจรชาร์จแบตเตอร์รี่ และ ดีซี. โวลท์มิเตอร์
ชื่อนักศึกษา	นาย กิตติ โกสิทธิ์
	นาย ชัชชัย อินทรศักดิ์
อาจารย์ที่ปรึกษา	อ.ไพศาล สิทธิโยภาสกุล
ภาควิชา	เทคนิคอุตสาหกรรม
ปีการศึกษา	2536

ภาคเทคนิคอุตสาหกรรม คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง อนุมัติให้นับปริญญานิพนธ์ฉบับนี้ เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาอุตสาหกรรมศาสตร์บัณฑิต

.....หัวหน้าภาควิชาเทคนิคอุตสาหกรรม

( )

คณะกรรมการสอบ

.....ประธานคณะกรรมการ

( )

.....กรรมการ

( )

.....กรรมการ

( )

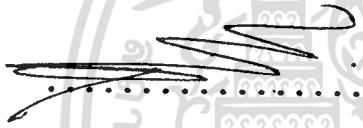
.....กรรมการ

( )

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Project Report	PC Control Power Supply circuit, Battery Charge Circuit And DC. Voltmeter
By	Mr. Kitti Kosit
	Mr. Chatchai Intarasak
Department	Industrial Technology
Project Report Advisor	Mr. Pisan Sittiyophasakul

Accepted by the Faculty of Engineering, King Mongkut's Institute of Technology, Ladgrabang in partial fulfillment of the requirements for the bachelor's degree



.....Chairman

( )

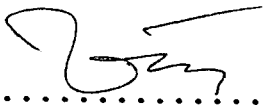
Project Report Committee

.....Member

( )

.....Member

( )



.....Member

( )

.....Member

( )

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หัวข้อปริญญานิพนธ์	การใช้ PC ควบคุมวงจรเพาว์เวอร์ซัพพลาย, วงจรชาร์จแบตเตอร์รี่ และ ดีซี. โวลท์มิเตอร์
ชื่อนักศึกษา	นาย กิตติ โกสิทธิ์
	นาย ชัชชัย อินทรศักดิ์
อาจารย์ที่ปรึกษา	อ.ไพศาล สิทธิโยภาสกุล
ภาควิชา	เทคนิคอุตสาหกรรม
ปีการศึกษา	2536

บทคัดย่อ

การใช้ไมโครคอมพิวเตอร์ ควบคุมวงจร Power Supply วงจรชาร์จแบตเตอร์รี่ และวงจร DC.Voltmeter นี้ สามารถแบ่งเป็น 2 ส่วนใหญ่ ๆ คือ ส่วนของซอร์ฟแวร์และส่วนของฮาร์ดแวร์ ส่วนของซอร์ฟแวร์นั้นจะใช้ภาษาเบสิคในการเขียนโปรแกรม ซึ่งในโปรแกรมจะมี sub program ต่างๆ เพื่อใช้ควบคุมฮาร์ดแวร์ภายนอก และ รับข้อมูลจากฮาร์ดแวร์ภายนอกส่วนของฮาร์ดแวร์ จะประกอบด้วยวงจร 4 วงจร คือ

ส่วนที่ 1 จะเป็นวงจร 8255 I/O Port ซึ่งจะมีหน้าที่ใช้ Interface กับวงจรอื่นๆ ภายนอก

ส่วนที่ 2 จะเป็นวงจร Power Supply ซึ่งจะใช้หลักการของ D/A Convertor ซึ่งควบคุมจาก DATA ของ 8255 โดย Power Supply สามารถนำไปใช้เป็นประโยชน์กับวงจรภายนอก จะสามารถ เลือก Voltage ได้ 0-25.0 V ได้ โดยมีความละเอียดขั้นละ 0.1 V

ส่วนที่ 3 เป็นวงจร DC. Voltmeter จะใช้หลักการของ A/D Convertorโดยจะรับ Voltage ทางภายนอก มาแปลงเป็นสัญญาณ DIGITAL เพื่อส่งเข้าทาง 8255 แล้วไปแสดงผลทางจอภาพ

ส่วนที่ 4 เป็นวงจรชาร์จแบตเตอร์รี่โดยเมื่อเราเลือกใช้การชาร์จแบตเตอร์รี่แล้วเราจะใช้ แรงดันจากวงจร Power Supply ค่าต่าง ๆ ไปควบคุมกระแสในการชาร์จแบตเตอร์รี่ของชนิดต่าง ๆ และวงจร DC. Voltmeter ก็จะรับ Voltage ตรงจุด Senser เพื่อใช้ในการหยุดชาร์จแบตเตอร์รี่ เมื่อแบตเตอร์รี่เต็มแล้ว

ลิขสิทธิ์ของวิทยานิพนธ์ที่เผยแพร่ในเว็บไซต์นี้เป็นของจุฬาลงกรณ์มหาวิทยาลัย ห้ามมิให้ทำซ้ำ ดัดแปลง เผยแพร่ หรือกระทำการใดๆ ที่ละเมิดลิขสิทธิ์ ก่อนได้รับอนุญาตจากเจ้าของลิขสิทธิ์เท่านั้น ผู้ใช้งานสามารถนำไปใช้ประโยชน์ด้านการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่อนุญาตให้นำไปเผยแพร่ต่อ ทั้งนี้ต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Project Report	PC Control Power Supply circuit,Battery Charge Circuit And DC. Voltmeter
Name	Mr. Kitti Kosit
	Mr. Chatchai Intarasak
Project Report Advisor	Mr. Pisan
Department	Industrial Technology
Academic Year	1993

Abstract

USING A MICROCOMPUTER CONTROL POWER SUPPLY CIRCUIT ,BATTERY CHARGE CIRCUIT AND DC. VOLTMETER. THIS PROJECT CONSIST THE SOFTWARE AND THE HARDWARE. THE SOFTWARE USED BASIC LANGUAGE FOR PROGRAMMING. THIS PROGRAM HAVE SUB PROGRAM USED FOR CONTROL THE EXT. HARDWARE AND RECEIVE THE DATA FROM EXT.HARDWARE. THE HARDWARE CONSIST 4 CIRCUITS ARE:

1. 8255 I/O PORT CIRCUIT USED FOR INTERFACE WITH THE EXT. CIRCUIT

2. POWER SUPPLY CIRCUIT USED D/A CONVERTOR CONTROL DATA OF 8255. THIS POWER SUPPLY CAN USED WITH THE EXT.CIRCUIT WHICH CAN SELECTED OUTPUT VOLTAGE FROM 0.0 TO 25.0 V.(0.1/STEP)

3. DC.VOLTMETER CIRCUIT USED A/D CONVERTOR BY RECEIVE EXT. VOLTAGE FOR CONVERT TO DIGITAL SIGNAL AND SEND THIS DATA TO 8255 FOR DISPLAY ON MONITOR.

4. BATTERY CHARGE CIRCUIT USED VOLTAGE FROM POWER SUPPLY CIRCUIT TO CONTROL THE CURRENT FOR CHARGE THE BATTERY EACH TYPE AND DC. VOLTMETER CIRCUIT WILL BE RECEIVE VOLTAGE AT SENSOR POINT FOR STOP CHARGE BATTERY (WHEN POWER FULL)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กิตติกรรมประกาศ

เนื่องจากชุดการใช้ไมโครคอมพิวเตอร์ควบคุมวงจร Power Supply วงจรชาร์จแบตเตอร์รี่ และ DC.Voltmeter นี้ จำเป็นจะต้องใช้ความรู้ทางด้าน คอมพิวเตอร์ และเทคนิคต่าง ๆ ซึ่งข้อมูล และ เอกสารอ้างอิงต่าง ๆ ไม่สามารถหาได้ง่ายนัก ซึ่งความอนุเคราะห์ที่ผู้จัดทำ ได้รับอย่างมากนี้ สืบเนื่องมาจาก ความตั้งใจจริงในการอบรมสั่งสอนและแนะนำใช้แนวทางที่ถูกต้องของอาจารย์ ผู้มีพระคุณ ซึ่งผู้จัดทำนอกจากจะได้รับความรู้ต่าง ๆ จากการถ่ายทอดแล้ว ยังได้รับความห่วงใย ที่มีต่อผู้จัดทำฉันท์ศิษย์ และ อาจารย์อีกด้วย

ฉะนั้นในโอกาสอันเหมาะสมนี้ ผู้จัดทำขอได้กล่าวคำขอบพระคุณในความรู้ ความห่วงใย และความปราถนาดี ที่ผู้จัดทำได้รับจากท่าน อาจารย์ ไพศาล สิทธิโยภาสกุล มา ณ. ที่นี้ด้วยความ จริงใจ โดยส่วนดีของผลงานที่ อาจมีโอกาสได้เผยแพร่ไปยังผู้แสวงหาความรู้ทางด้านนี้ ผู้จัดทำ ขอมอบให้เป็นเกียรติแก่ท่านอาจารย์ ส่วนข้อผิดพลาดหรือแนวทางที่ผิดนั้น ผู้จัดทำ น้อมรับไว้เพื่อแก้ไขต่อไปในอนาคตมา ณ. ที่นี้ด้วย

คณะผู้จัดทำ

15 ธันวาคม 2536

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ

หน้า

บทคัดย่อ

กิตติกรรมประกาศ

สารบัญ

บทที่ 1 บทนำ 1

1.1 ความสำคัญและที่มาของโครงงาน 1

1.2 วัตถุประสงค์ของโครงงาน 2

1.3 ขอบเขตความสามารถของโครงงาน 2

1.4 ประโยชน์ที่ได้รับ 2

1.5 ระเบียบวิธีดำเนินโครงงาน 3

1.6 อุปกรณ์ที่จำเป็น 3

1.7 งบประมาณในการดำเนินโครงงาน 3

บทที่ 2 ทฤษฎี และ หลักการ 4

2.1 พื้นฐานทั่วไปของ ไมโครคอมพิวเตอร์ 5

2.1.1 ไมโครโปรเซสเซอร์ 6

2.1.1.1 หน่วยคำนวน 6

2.1.1.2 รีจีสเตอร์ที่ใช้เก็บข้อมูล 6

2.1.1.3 รีจีสเตอร์ที่ใช้เก็บข้อมูล ของแอดเดรส 7

2.1.1.4 หน่วยควบคุม 7

2.1.1.5 สัญญาณนาฬิกา 7

2.1.2 หน่วยความจำ 7

2.1.2.1 รอม 7

2.1.2.2 แรม 8

2.1.3 อุปกรณ์ อินพุท-เอาท์พุท 8

2.1.3.1 แอดเดรสบัส 8

2.1.3.2 บัสควบคุม 8

2.1.3.3 บัสข้อมูล 8

2.2 ตำแหน่งต่าง ๆ ของสล๊อต IBM PC 9

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เรื่อง	หน้า
2.3 ตำแหน่งต่าง ๆ เกี่ยวกับสัญญานบนสล็อต IBM PC	11
2.3.1 A0-A19	11
2.3.2 D0-D7	11
2.3.3 RESET DRV.	12
2.3.4 ALE	12
2.3.5 MEMW	13
2.3.6 MEMR	13
2.3.7 IOR	13
2.3.8 IOW	14
2.3.9 +5 Vdc.	14
2.3.10 +12 Vdc.	14
2.3.11 -5 Vdc.	15
2.3.12 -12 Vdc.	15
2.3.13 GND	15
2.4 การกำหนดหมายเลขพอร์ตบน IBM PC	16
2.5 ไอซี 8255 โปรแกรมมาเบิ้ลเพอริเพอรัล อินเตอร์เฟสชิ่ง	17
2.5.1 ลักษณะทั่วไปของ 8255	18
2.5.2 การต่อใช้งาน 8255	20
2.5.2.1 การโปรแกรม 8255	21
2.5.2.2 กรณีพิเศษของรหัสควบคุม	24
2.5.2.3 การทำงานในโหมด 0	25
2.5.2.4 การทำงานในโหมด 1	25
2.5.2.5 การทำงานในโหมด 2	29
2.6 A/D คอนเวอร์เตอร์	33
2.6.1 ทฤษฎีการสุ่มตัวอย่าง	34
2.6.2 A/D แบบแฟลช์	35
2.6.3 ADC แบบสโลปเดียว	37
2.6.4 A/D แบบสโลปคู่	38

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เรื่อง	หน้า
2.6.5 DAC แบบมีการป้อนกลับ	40
2.6.6 ADC แบบประมาณค่าหลายๆ ครั้ง	43
2.6.7 Quantizing Error	44
2.7 D/A คอนเวอร์เตอร์	45
2.7.1 Binary-Weighed Resistor D/A	46
2.7.2 Ladder Network D/A	48
บทที่ 3 การใช้ พีซี ควบคุมแหล่งจ่ายไฟ ,วงจรชาร์จแบตเตอรี่ และ ดีซี. โวลท์มิเตอร์	51
- POWER SUPPLY	53
- DC. VOLTMETER	53
- BATTERY CHARGER	53
3.1 วงจร 8255 I/O Port	54
3.1.1 ลักษณะของ 8255 I/O Port Card	54
3.1.2 การใช้งาน IC 8255	54
3.1.3 การทำงานของ 8255 I/O Card	55
3.1.4 การ Decode Port	55
3.2 วงจร D/A Convertor และ Power Supply	58
3.2.1 การใช้งาน	60
3.3 วงจร A/D Convertor	60
3.3.1 การใช้งานของวงจร A/D Card	60
3.3.2 การทำงานของวงจร	61
3.3.3 เครื่องวัดแรงดัน	62
3.3.4 วัดแรงดันไฟฟ้า DC.	63
3.4 วงจรชาร์จแบตเตอรี่	66
3.4.1 เซลแบบนิเกิล-แคดเมียม	66
3.4.2 ปัญหาที่เกิดขึ้น	67
3.4.3 ราคา	68
3.4.4 เซลแบบกระดุม	71

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เรื่อง	หน้า
3.4.5 หลักการทำงานของวงจร	73
บทที่ 4 ภาษาคอมพิวเตอร์	75
4.1 ภาษาเครื่อง	75
4.2 ภาษาแอซเซมบลี	76
4.3 ภาษาระดับสูง	77
4.3.1 Procedure Orlented Language	78
4.3.2 Problem Oriented Language	79
4.4 ภาษาเบสิค	79
4.4.1 ข้อได้เปรียบของภาษาเบสิค	80
4.4.2 คำสั่ง	80
4.4.2.1 คำสั่งตรง	83
4.4.2.2 คำสั่งตามขั้นตอน	89

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 1

บทนำ

ในปัจจุบัน การใช้ไมโครคอมพิวเตอร์ มีใช้กันอย่างแพร่หลายไม่ว่า จะเป็น การใช้งานทางธุรกิจ วิศวกรรม หรือ ใช้ส่วนตัว ก็ตาม เครื่องไมโครคอมพิวเตอร์ นั้น สามารถขยายความสามารถ ของระบบเพื่อประสิทธิภาพ การทำงาน หรือเพิ่มฟังก์ชั่นการทำงานของระบบได้โดย การเชื่อมต่อ เครื่องไมโครคอมพิวเตอร์ กับ วงจรภายนอก โดยใช้ไอซีเบอร์ 8255 เป็น Input Portเพื่อรับ DATA จาก วงจรภายนอก และ Output Port เพื่อส่ง DATA ไปควบคุมวงจรภายนอก

ในส่วนของวงจรภายนอกที่ออกแบบนี้จะประกอบด้วยวงจร Power Supply ซึ่งอาศัยวงจร DIGITAL TO ANALOG CONVERTOR วงจร DC. Voltmeter ซึ่งอาศัยวงจร ANALOG TO DIGITAL CONVERTOR และ วงจรชาร์จแบตเตอร์รี่

การออกแบบของวงจรภายนอก จะคำนึงถึง ความแม่นยำ เป็นหลัก โดยการออกแบบจะแบ่งการใช้งานเป็น 2 ลักษณะ คือการใช้วงจร Power Supply และ DC.Voltmeter ไปต่อ ร่วมกับ วงจร ชาร์จแบตเตอร์รี่ กับ การใช้วงจร Power Supply และ DC. Voltmeter ไปใช้งานในการทดลองได้ ซึ่งหวังเป็นอย่างยิ่งว่า คงเป็นประโยชน์ต่อท่านที่สนใจไม่มากก็น้อย

1.1 ความสำคัญและที่มาของโครงงาน

ในปัจจุบัน การใช้ไมโครคอมพิวเตอร์ ไปประยุกต์ใช้งานนั้นมีกันอย่างแพร่หลาย และส่วน หนึ่งในด้านการทดลองนั้นจำเป็นจะต้องมีชุด Power Supply และ DC. voltmeter ที่ดี และ เที่ยงตรง ไมโครคอมพิวเตอร์ ก็สามารถประยุกต์ ใช้งานทางด้าน การทดลองได้เช่นกัน รวมทั้งใช้ใน การควบคุม การชาร์จแบตเตอร์รี่ได้ อีกด้วย โครงงานนี้จึงเกิดขึ้นมาเพื่อศึกษา การประยุกต์ การใช้งาน ของ ไมโครคอมพิวเตอร์ และสนองความต้องการของผู้ที่ทำงานและกำลังศึกษาอยู่ขณะนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1.2 วัตถุประสงค์ของโครงงาน

จุดมุ่งหมายของโครงงาน นั้น จะเน้นทางด้าน การเพิ่มความสามารถของไมโครคอมพิวเตอร์ โดยการ ขยายระบบ เพื่อเป็นประโยชน์ให้มากยิ่งขึ้น อย่างไรก็ตาม การทำโครงการนี้จึงมีวัตถุประสงค์ดังนี้

1. เพื่อออกแบบ และสามารถใช้ไมโครคอมพิวเตอร์ไปควบคุมวงจรPower Supply วงจร ชาร์จแบตเตอรรี่ และ เป็นส่วนรับข้อมูลจากวงจร DC. Voltmeter

2. เพื่อให้เป็น ต้นแบบ สำหรับนักศึกษา หรือ บุคคลที่สนใจจะนำไปศึกษา เพื่อความเข้าใจในการใช้ ไมโครคอมพิวเตอร์ เชื่อมต่อใช้งานกับวงจรภายนอกได้

3. เพื่อใช้ วิชาความรู้ที่ได้ ศึกษาในหลักสูตร ให้เกิดประโยชน์ ทางด้าน การศึกษาเล่าเรียน และ อื่น ๆ ได้

4. เพื่อเป็นการ ประยุกต์ ไมโครคอมพิวเตอร์ และอุปกรณ์อิเล็กทรอนิคส์ ให้มีความสามารถ และ ประสิทธิภาพดียิ่งขึ้น

5. สามารถนำไป ประยุกต์ และ ปรับปรุง ใช้งานในลักษณะต่าง ๆ กัน

1.3 ขอบเขตความสามารถของโครงงาน

1. สามารถใช้ ชาร์จแบตเตอร์รี่ได้ หลายชนิด เช่น เซลขนาด AAA, AA, C,และ PP-3 ซึ่งสามารถ ชาร์จแบตเตอร์รี่ได้ สูงสุด 8 ก้อน

2. สามารถใช้เป็น Power Supply ที่เลือก Voltage ได้ 0 - 25 V มีความละเอียด ขั้นละ 0.1 V โดยจ่ายกระแสได้ 2 A.

3. สามารถใช้เป็น DC. Voltmeter ได้ โดยเลือกย่านวัดได้ 4 RANGE คือ 200 mV, 2 V, 20 V, 200 V

1.4 ประโยชน์ที่ได้รับ

1. สามารถนำเครื่องคอมพิวเตอร์ ประยุกต์ ใช้งานภายนอกได้โดยการInterface

2. เป็นแนวทางของ การศึกษา ต่อไป ของการนำไมโครคอมพิวเตอร์ (PC)ไปประยุกต์ใช้ งานอื่นๆ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3. รู้เทคนิค ในการเขียนโปรแกรม และ ภาษาในการสั่งงาน หรือ ติดต่อให้คอมพิวเตอร์ ทำงานได้อย่างเหมาะสม

4. รู้จักการ เชื่อมต่อ (Interface) ของไมโครคอมพิวเตอร์ และ ส่วนของ โปรแกรม เพื่อควบคุม Hard Ware

1.5 ระเบียบวิธีดำเนินโครงงาน

1. ศึกษาการทำงาน และ การใช้งานเครื่อง ไมโครคอมพิวเตอร์ (PC)

2. ศึกษาการทำงาน และ การใช้งานในส่วนของ hard ware ส่วนของการinter face วงจร 8255 I/O port , ชุดวงจร power supply , ชุดวงจร charge battery และ ส่วนของวงจร A/D Converter และ D/A Converter

3. ศึกษา ส่วนของ Soft ware โดยใช้ภาษาเบสิก

4. ทำการออกแบบส่วนประกอบ ต่าง ๆ ของ Hard Ware และ Soft Ware

5. ดำเนินการสร้างชุดของ Hard ware และเขียนโปรแกรมทางด้าน Soft Ware

6. ทดลองการทำงานร่วมกันระหว่างส่วน Hard ware และ Soft ware

7. สรุป ผลการทำโครงงาน ข้อเสนอแนะ ปัญหาในการทำงาน รวบรวมข้อมูลทั้งหมด พิมพ์ ตรวจ แก้ไข และเสนอรายงาน

1.6 อุปกรณ์ที่จำเป็นในการทำโครงงาน(ที่มีอยู่แล้วและต้องจัดหา)

1. Microcomputer

2. Multimeter

3. Photo Board

4. อุปกรณ์อิเล็กทรอนิกส์ ที่เกี่ยวกับการประกอบ Hard Ware เช่น IC ,Resistor ฯลฯ

5. อุปกรณ์ในการเชื่อมต่อ Microcomputer กับ Hard Ware

1.7 งบประมาณในการดำเนินโครงงานประมาณ 5,000 บาท

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์ ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

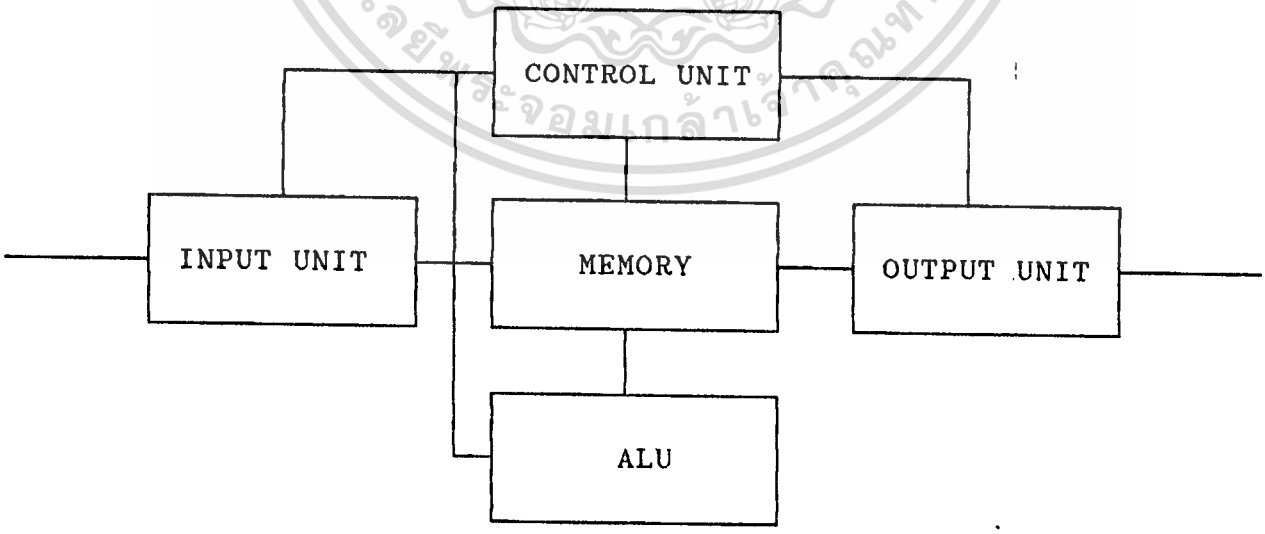
บทที่ 2
ทฤษฎี
และ
หลักการ

การใช้ไมโครคอมพิวเตอร์เข้าไปเชื่อมต่อกับระบบภายนอกต่าง ๆ นั้น เช่นการควบคุมการชาร์จแบตเตอรี่ ,แหล่งจ่ายไฟ(Power Supply) และใช้เป็น DC.Voltmeter ในระบบไมโครคอมพิวเตอร์ มีการทำงานเป็นแบบดิจิตอล (Digital) การจะเชื่อมต่อเข้ากับ สัญญาณอนาล็อก (Analog) จำเป็นต้องมีการเปลี่ยน สัญญาณดิจิตอล (Digital Signal) ให้ เป็น สัญญาณอนาล็อกเสียก่อน เพื่อทำการควบคุมอุปกรณ์ภายนอกแบบอนาล็อก นั้น อุปกรณ์ภายนอก อนาลอกที่ถูกควบคุมโดยสัญญาณ ดิจิตอล จากคอมพิวเตอร์ในงานชิ้นนี้ก็คือ วงจรแหล่งจ่ายไฟ (Power Supply) และ วงจรการชาร์จแบตเตอรี่ เราสามารถปรับปรุงให้ ควบคุมการจ่ายไฟ และ การชาร์จแบตเตอรี่ได้ ส่วนการเชื่อมต่อแบบอนาล็อกเข้าสู่ระบบคอมพิวเตอร์ จำเป็นต้องมี ตัวกลางในการแปลงสัญญาณอนาล็อกนั้นให้เป็นสัญญาณทางอิเลคทรอนิกส์ ตัวแปลงที่ใช้ก็คือ วงจร เปลี่ยนสัญญาณอนาล็อก เป็นสัญญาณดิจิตอล และพอร์ต แบบขนานที่เราใช้คือ ไอซี เบอร์ 8255 โปรแกรมมาเบิลเพอริเพอรัล อินเตอร์เฟสชิ่ง (Programmable Peripheral Interface) ซึ่งทำงานได้ง่าย การศึกษาได้เสนอระบบที่ใช้ ควบคุมแหล่งจ่ายไฟ และ การชาร์จแบตเตอรี่ โดย ใช้การแปลงผันสัญญาณอนาล็อกให้เป็นดิจิตอล และ ในทางกลับกันสามารถ เปลี่ยนสัญญาณ ดิจิตอลให้อยู่ ในรูปสัญญาณอนาล็อก โดยใช้วงจรที่เรียกว่า การแปลงสัญญาณดิจิตอลเป็นสัญญาณอนาล็อก โดย เราน่าไมโครคอมพิวเตอร์ มาควบคุม แหล่งจ่ายไฟ และ การชาร์จแบตเตอรี่ โดยใช้ส่วนอินเตอร์เฟส (Interface) ที่เราสร้างขึ้นมาทำการต่อเข้ากับไมโครพิวเตอร์ และต่อเข้ากับอุปกรณ์ภายนอก ก็คือชุดแหล่งจ่ายไฟตรง โดยที่เราจะใช้ในส่วนของซอฟต์แวร์(Software) มาควบคุม ไมโครคอมพิวเตอร์อีกครั้งหนึ่ง โดยที่เราสามารถจัด (Set)ได้ โดยที่ระบบควบคุม จะต้องมีการ ผิดเพี้ยน ของสัญญาณ การควบคุมน้อยที่สุด และ แสดงผลการควบคุมได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.1 พื้นฐานทั่วไปของไมโครคอมพิวเตอร์

จากอดีต ที่ผ่านมา ระบบคอมพิวเตอร์ ได้รับการพัฒนา และเปลี่ยนแปลงไปอย่างมาก และ ยังมี แนวโน้มที่จะ พัฒนาต่อไปอีกในอนาคต ซึ่งในปัจจุบันอุปกรณ์ต่าง ๆ ที่เกี่ยวกับคอมพิวเตอร์ ได้มีขนาดเล็กลง ทำงานรวดเร็วและมีความเชื่อถือได้มากขึ้น ดังนั้นจึงทำให้คอมพิวเตอร์ได้เข้า มามีส่วนเกี่ยวข้อง กับชีวิตประจำวัน ในสังคมมนุษย์ไปอย่างหลีกเลี่ยงไม่ได้ จะเห็นได้จากนำเอา คอมพิวเตอร์ เข้าไปใช้ในระบบการ จัดการเกี่ยวกับธุรกิจระบบการควบคุมต่าง ๆ ภายในโรงงานอุตสาหกรรม เป็นต้น และมีแนวโน้มที่จะนำมาประยุกต์ใช้เพิ่มมากขึ้นเรื่อยๆ การใช้ไมโครโปรเซสเซอร์ (Microprocessor)ควบคุมความเร็วมอเตอร์ก็เป็นอีกวิธีหนึ่ง ที่นำเอา ระบบคอมพิวเตอร์ เข้ามาประยุกต์ใช้ อย่างมีประสิทธิภาพซึ่งมีผลดี คือ ให้ความแม่นยำใน การควบคุมสูง มีการตอบต่อปฏิกิริยาที่เกิดขึ้นค่อนข้างรวดเร็วและยังให้ความสะดวกสบาย ในการทำงานเพราะวิธีการควบคุม โดย ซอฟท์แวร์ (Software) นั้นเมื่อต้องการ แก้ไขคำสั่ง ก็สามารถกระทำ ได้ง่าย โดยการเปลี่ยนแปลงซอฟท์แวร์เท่านั้นไม่ต้องไปเปลี่ยนโครงสร้าง ทางด้านฮาร์ดแวร์ (Hardware)ซึ่งเป็นความยุ่งยากในทางปฏิบัติ ส่วนประกอบต่าง ๆ ของไมโครคอมพิวเตอร์จะเชื่อมต่อกันด้วยบัส (Bus) ส่วนประกอบ เหล่านี้ จะมีลักษณะเป็น ไอซีแบบ LSI (Large Scale Intregrated Circuit) ในรูปที่ 2.1.1



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 2.1.1 โครงสร้างพื้นฐานของไมโครคอมพิวเตอร์

จะแสดงถึง โครงสร้างพื้นฐานของ ไมโครคอมพิวเตอร์ทั่ว ๆ ไป และ เราสามารถอธิบายถึง ส่วนประกอบของไมโครคอมพิวเตอร์แยกเป็นส่วน ๆ ได้ 3 ส่วนคือ

- ไมโครโปรเซสเซอร์ (Microprocessor)
- หน่วยความจำ (Memory)
- อุปกรณ์อินพุท-เอาท์พุท (Input-Output)

2.1.1 ไมโครโปรเซสเซอร์ (Microprocessor)

ส่วนไมโครโปรเซสเซอร์นี้ ถือว่า เป็นหัวใจ และ สมองของระบบคอมพิวเตอร์ จะทำหน้าที่ตามคำสั่ง หรือ โปรแกรมที่ผู้ใช้เขียนขึ้นโดยมีซีพียู (CPU :Central Processor Unit) เป็นตัวควบคุม การทำงานของอุปกรณ์อื่น ๆ ที่ทำหน้าที่ต่างกันให้ทำงาน สอดคล้องกัน ซีพียูนี้จะทำหน้าที่ จัดลำดับการทำงาน ก่อน-หลัง ตามความสำคัญ ของโปรแกรม ซึ่งขณะทำงานโปรแกรมเหล่านี้จะถูกเก็บ เข้าไว้ในหน่วยความจำ (Memory Unit) ซึ่งมีสาย ข้อมูล (Data Bus) และ สายตำแหน่ง (Address Bus) ที่ต่อเข้ากับสายข้อมูลและสานตำแหน่งของไมโครโปรเซสเซอร์ ภายในไมโครโปรเซสเซอร์ จะมีส่วนประกอบสำคัญอยู่ 3 ส่วนคือ

2.1.1.1 หน่วยคำนวณ (ALU:Aritgermatec and Logical Unit)

เป็นหน่วยความจำ ที่ทำหน้าที่ คำนวณผลทางคณิตศาสตร์ และ ตรรกศาสตร์ เช่นการบวก ลบ AND หรือ OR กันในแต่ละบิทของข้อมูล ซึ่งผลลัพธ์ของการคำนวณทุกครั้งจะถูกเก็บเอาไว้ใน หน่วยความจำชั่วคราว หรือ รีจิสเตอร์ (Registor) เรียกว่าแอคคิวมูเลเตอร์ (Accumulater : A Registor) และ จะแสดงสภาวะ ของผลลัพธ์ที่คำนวณได้ ที่แฟลกรีจิสเตอร์ (Flag : F Registor)

2.1.1.2 รีจิสเตอร์ที่ใช้เก็บข้อมูล (Data Registor)

เป็นรีจิสเตอร์ที่ใช้สำหรับ เก็บข้อมูลชั่วคราว หลังจากทำ การคำนวณ หรือ เป็นที่รอพักข้อมูล ระหว่างการเคลื่อนย้าย ตำแหน่งภายใน ของ CPU เอง หรือหน่วยความจำอื่น ๆ CPU เบอร์ Z-80 นั้นจะมีรีจิสเตอร์ที่ใช้ เก็บข้อมูลหลัก ๆ คือ AF,BC,DE และ HLจะเป็นรีจิสเตอร์ขนาด 16 บิท (Bit) สามารถใช้แยกกันทีละ 8 บิทได้ รีจิสเตอร์หลักทั้งหมดนี้จะมี บัสข้อมูล (Data Bus) 8 สาย ต่อถึงกันหมด และ ยังต่อกับ บัสข้อมูล ขนาด 8 บิทจากภายนอกอีกด้วย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.1.1.3 รีจิสเตอร์ที่ใช้เก็บข้อมูลของแอดเดรส (Address Registor)

เป็นรีจิสเตอร์ ที่ใช้สำหรับ เก็บข้อมูลชั่วคราว หลังจากการทำงาน หรือ เป็นที่รอพักข้อมูลระหว่างการย้ายตำแหน่งภายใน ของ CPU อ้างถึงตำแหน่ง ของอุปกรณ์ เหล่านี้ ได้รีจิสเตอร์ที่ใช้เก็บตำแหน่งข้อมูลใน Z-80 คือ โปรแกรมเคาท์เตอร์ (PC : Program Counter) และ แสตคพอยท์เตอร์ (SP : Stack Pointer)

2.1.1.4 หน่วยควบคุม (Control Unit)

ถือว่าเป็นหัวใจของขบวนการทั้งหมด โดยจะส่งสัญญาณคอยควบคุมให้จังหวะแก่หน่วยอื่น ๆ ให้ทำงานไม่ซับซ้อน หน่วยควบคุม จะรับคำสั่งมาจากหน่วยความจำ และ แปลคำสั่งนั้น แล้วส่งสัญญาณเท่าที่จำเป็น ไปควบคุมหน่วยอื่น ๆ ให้เป็นไปตามคำสั่ง หน่วยควบคุมจะรู้ว่าคำสั่งต่อไปอยู่ที่ไหน ในหน่วยความจำ โดยมีตัวบอกตำแหน่ง คือ โปรแกรมเคาท์เตอร์

2.1.1.5 สัญญาณนาฬิกา (Clock)

เป็นฐานเวลาที่ป้อนให้กับชุดของ CPU แล้วจะใช้ฐานเวลานี้เป็นตัวควบคุมจังหวะการทำงานความถี่ของสัญญาณนาฬิกานี้ จะเป็นตัวกำหนดความเร็วในการทำงานของไมโครโปรเซสเซอร์

2.1.2 หน่วยความจำ (Memory)

หน่วยความจำ เป็นหน่วยที่เก็บคำสั่ง และ ข้อมูลต่าง ๆ ทั้งหมดขนาดของหน่วยความจำจะขึ้น อยู่กับจำนวนตำแหน่ง ที่หน่วยความจำ จะสามารถเก็บข้อมูลได้ หน่วยความจำแบ่งออกเป็น 2 ชนิด

2.1.2.1 รอม (ROM :Read Only Memory)

การใช้งานในคอมพิวเตอร์นั้นข้อมูลบางอย่างเรานำไปเก็บไว้เพียงครั้งเดียวนั้น หลังจากนั้นก็อ่านออกมา ใช้แต่เพียงอย่างเดียว หน่วยความจำที่เหมาะกับงานนี้ก็ควรจะเป็นแบบเขียนได้ และ ไม่สูญหายไป แม้ว่าจะไม่มีไฟเลี้ยงตามซึ่งก็คือรอมนั้นเอง มีอยู่ 3 แบบคือ

- รอม (ROM) เป็นหน่วยความจำ ที่โปรแกรมมาจากโรงงานผู้ผลิตเลย
- พรอม (PROM) เป็นหน่วยความจำที่ ผู้ใช้งานมาโปรแกรมเอง ตอนจะใช้งานตาม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น ไม่อนุญาตให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ต้องการ

- อีพรอม (EPROM) เป็นหน่วยความจำที่ผู้ใช้งานมาโปรแกรมเองแต่สามารถลบออกได้ด้วยวิธีการอันเหมาะสม เช่น การฉายแสงอุลตร้าไวโอเลต หน่วยความจำแบบ ROM นี้จะใช้เก็บโปรแกรมสำหรับ ระบบโปรแกรมมอนิเตอร์

2.1.2.2 แรม (RAM :Read Access Memory)

เป็นหน่วยความจำที่สามารถอ่านและเขียนข้อมูลลงไปได้ มีข้อดีคือใช้แรมในการพัฒนาโปรแกรม เพราะสามารถเปลี่ยนแปลงแก้ไขได้ง่าย เมื่อพัฒนาโปรแกรมเสร็จแล้วจึงจัดเก็บข้อมูลลงไปในรอมอีกที เพื่อเป็นการเก็บข้อมูลอย่างถาวรต่อไป ข้อเสีย คือขณะที่ใช้งานอยู่ถ้าเกิดไฟดับจะทำให้ข้อมูลต่าง ๆ ถูกลบไปด้วย แต่จะมีแรมอีกแบบหนึ่งคือ แรมแพค (RAM Pack) ซึ่งสามารถเก็บข้อมูลไว้ได้เมื่อไฟดับไป เพราะแรมชนิดนี้จะมีแบตเตอรี่คอยจ่ายไฟเลี้ยงไว้จนกว่าแบตเตอรี่จะหมดไป

2.1.3 อุปกรณ์อินพุท-เอาท์พุท (Input-Output)

เป็นอุปกรณ์ภายนอก ที่ทำหน้าที่ ติดต่อกับไมโครคอมพิวเตอร์ อุปกรณ์อินพุทก็คือแหล่งกำเนิดสัญญาณข้อมูล ให้กับ ไมโครโปรเซสเซอร์ อุปกรณ์เอาท์พุทก็ คือ แหล่งที่รับข้อมูลมาจากไมโครโปรเซสเซอร์โดย CPU จะส่งสัญญาณเลือกอุปกรณ์ตัวใดตัวหนึ่งพร้อมกับข้อมูลที่จะส่งหรือรับซึ่งจะเชื่อมต่อกับ CPU และ บัสต่าง ๆ ดังนี้

2.1.3.1 แอดเดรสบัส (Address Bus)

เป็นบัสทางเดียวใช้ส่งผ่านค่าแอดเดรสจาก CPU ออกไปจากหน่วยความจำ เพื่อระบุตำแหน่งที่ต้องการ รับ หรือ ส่งข้อมูล หรือ ใช้ระบุตำแหน่งของพอร์ท อินพุท-เอาท์พุท ที่ CPU ต้องการติดต่อด้วย

2.1.3.2 บัสควบคุม (Control Bus)

เป็นบัสทางเดียวที่ใช้ส่งผ่านสัญญาณควบคุมให้กับอุปกรณ์ต่าง ๆ ในระบบ

2.1.3.3 บัสข้อมูล (Data Bus)

เป็นบัส 2 ทิศทางที่ใช้ในการส่งผ่านข้อมูลระหว่าง CPU กับอุปกรณ์อื่น ๆ ในระบบจำ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับใช้เพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น ทั้งนี้ห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

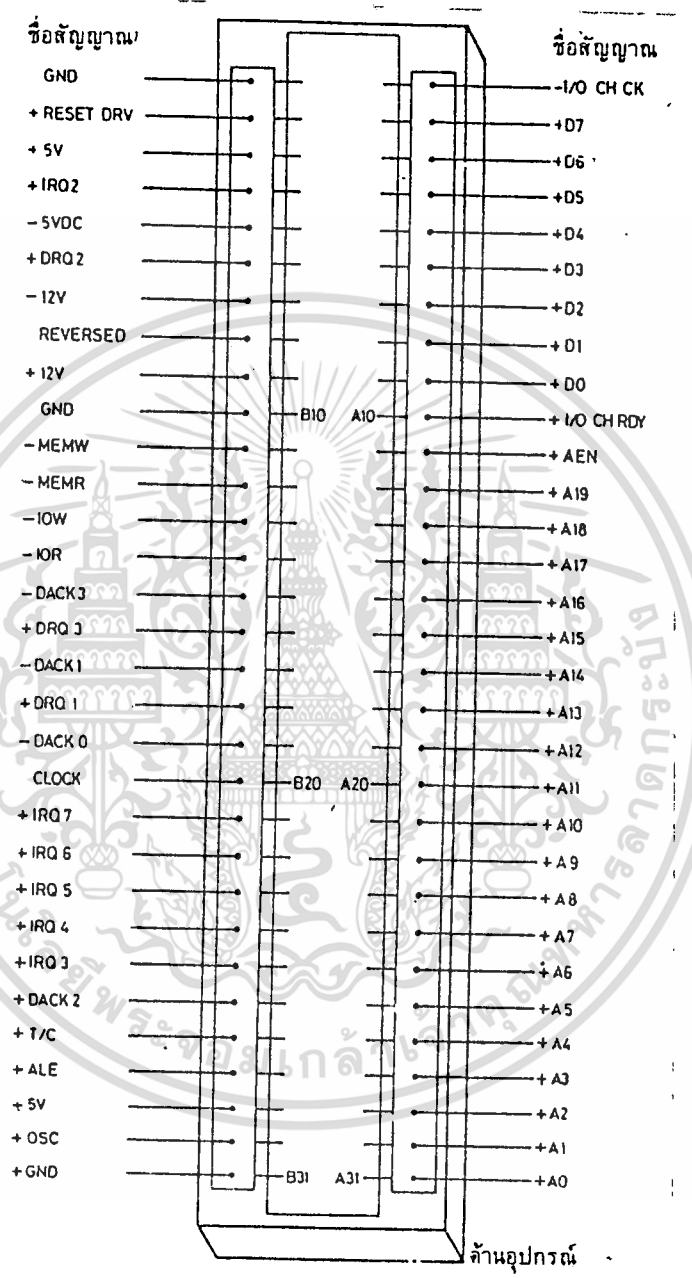
นวนเส้นของบัสข้อมูล จะขึ้นอยู่กับในกรณีของ 8088 นั้นCPU จะส่งผ่านข้อมูลทีละ 8 บิท ดังนั้นจะมีจำนวนเส้นของชนิดของ CPU บัสข้อมูล 8 เส้น

2.2 ตำแหน่งของสล็อต IBM PC

ภายใน IBM/PC ได้มีการออกแบบ ให้สามารถเพิ่มเติมวงจรอินเตอร์เฟส เข้าไปในภายหลังได้ โดยผ่านทางสล็อตที่อยู่บนเมนบอร์ด (Main Board) สำหรับสล็อตบนเมนบอร์ดมีจำนวน 5 สล็อต (IBM/PC มี 8 สล็อต) แต่ละสล็อตมีจำนวนขาทั้งสิ้น 62 ขา แบ่งออกเป็น 2 ข้างๆละ 31 ขา ตำแหน่งขาของสล็อตที่อยู่ทางด้านซ้ายของสล็อตจะเรียกโดยใช้อักษร B นำหน้าเลข ตำแหน่งของขา เช่น ขา B10 ก็คือขาทางด้านซ้ายของสล็อตขาที่ 10 นับจากทางด้านท้ายของเครื่อง ส่วนขาที่อยู่ทางด้านขวา ของสล็อต จะเรียก โดยใช้อักษร A นำหน้าเลขตำแหน่งของขา เช่น ขา A24 ก็คือ ขาทางด้านขวาของสล็อต ขาที่ 24 (นับจากทางด้านท้ายของเครื่อง) แต่ละขาของสล็อตเหล่านี้ เชื่อมต่อกับเส้นสัญญาณต่างบนเมนบอร์ด ทำให้การสร้างวงจรอินเตอร์เฟส กับ IBM/PC สามารถกระทำได้โดยสะดวกเส้นสัญญาณที่เชื่อมต่อกับขาสล็อตเหล่านี้ ประกอบไปด้วย เส้นสัญญาณของแอดเดรส (Address Bus) ,บัสข้อมูล (Data Bus) ,บัสควบคุม (Control Bus) สำหรับการเขียน / อ่านข้อมูลจากหน่วยความจำ หรือ พอร์ต I/O ,เส้นสัญญาณสำหรับการขออินเตอร์รัพท์ของ วงจรอินเตอร์เฟส, เส้นสัญญาณสำหรับการของ DMA ,สัญญาณเวลา (Timming Signal) ต่าง ๆที่ใช้ในระบบเส้นสัญญาณแสดงการรีเฟรชหน่วยความจำ และสัญญาณสำหรับตรวจสอบความผิดพลาด (I/O Check) นอกจากเส้นสัญญาณเหล่านี้แล้ว สล็อตบนเมนบอร์ดยังเชื่อมต่อกับแหล่งจ่ายไฟต่าง ๆ คือ + 5Vdc , - 5Vdc , + 12Vdc และ - 12Vdc ตำแหน่ง ต่าง ๆ บนสล็อตของ IBM/PC แสดงดังรูปที่ 2.2.1

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

033247



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 2.2.1 แสดงตำแหน่งขาบนสล็อตของ IBM/PC

2.3 ตำแหน่งต่าง ๆ เกี่ยวกับสัญญาณบนสล็อต IBM/PC

สัญญาณต่างๆ บนสล็อตของ IBM/PC มีหลายสัญญาณแต่ในที่นี้จะกล่าวถึงเฉพาะสัญญาณต่าง ๆ ที่เกี่ยวกับโครงงานนี้ ซึ่งเป็นสัญญาณที่จำเป็นต้องใช้เกี่ยวกับการเขียน/อ่านข้อมูลกับหน่วยความจำและพอร์ต I/O เท่านั้น

2.3.1 A0-A19 (Address Bus ; ขา A31-A12) :

ขาสัญญาณทั้ง 20 ขานี้เป็นเอาท์พุต ซึ่งใช้ กำหนดแอดเดรส ของหน่วยความจำ หรืออุปกรณ์ I/O ที่ 8088 ต้องการจะติดต่อด้วยโดยที่สัญญาณ A0 จะมีนัยสำคัญต่ำสุด สำหรับแอดเดรสบนบัส แอดเดรส A0 - A19 นี้จะถูกกำหนดโดย 8088 ในระหว่างขบวนการอ่าน / เขียนข้อมูลลงในหน่วย ความจำหรืออุปกรณ์ I/O แต่ในช่วงของขบวนการ DMA นั้น DMA - Controller จะเป็นผู้ กำหนดค่าแอดเดรสบนบัสแอดเดรสเอง (ในระหว่างนี้ 8088 จะถูกตัดออกจากระบบ) จะเห็นได้ ว่าจำนวนเส้นแอดเดรสนี้จะมีอยู่ 20 เส้น ซึ่งสามารถที่จะ อ้างแอดเดรสของหน่วยความจำได้ถึง 1 Mbyte แต่อย่างไรก็ตาม จะมีแอดเดรสบางแอดเดรสที่ถูกใช้งานโดย IBM/PC อยู่ก่อนแล้ว คือแอดเดรสของหน่วยความจำ RAM บนเมนบอร์ดที่ถูกใช้โดยระบบจำนวน 64 Kbyte (สำหรับ IBM PC/XT จะเป็นจำนวน 256 KByte) และแอดเดรสสำหรับหน่วยความจำ ROM อีก 48 Kbyte ซึ่งถูกจัดในช่วงของแอดเดรสบนสุดใน 1 Mbyte คือ 0FC00H (สำหรับ IBM PC/XT จะเป็น 64 Kbyte) สำหรับ การอ้างแอดเดรสของพอร์ต I/O นั้นจะใช้เส้นแอดเดรสเพียง 16 เส้นคือ A0-A15 ซึ่งจะทำให้อ้างแอดเดรสของพอร์ตได้ 64 กิโลพอร์ต โดยผ่านทางชุดคำสั่ง IN และ OUT ส่วนเส้นแอดเดรสที่เหลือ คือ A16-A19 นั้นจะไม่ถูกใช้งาน อย่างไรก็ตามภายใน IBM/PC จะใช้แอดเดรสในการ อ้างแอดเดรสของพอร์ตเพียง 10 เส้น คือจาก A0-A9 และค่าแอดเดรสที่ใช้งานจะต้องอยู่ในช่วง 0200H-03FFH เท่านั้น

2.3.2 D0-D7 (Data Bus ; ขา A9-A2) :

ขาสัญญาณนี้จะเป็นแบบ Bi - Directional ซึ่งต่อกับบัสข้อมูล ของระบบเพื่อทำหน้าที่ในการ ส่งผ่านข้อมูล ระหว่างพอร์ต I/O กับ IBM/PC โดยบิต D0 จะมีนัยสำคัญต่ำ

สุด และ บิต D7 จะมีนัยสำคัญสูงสุด สำหรับในบัสไซเคิลของการเขียนข้อมูล ที่สร้างขึ้นโดย 8088 นั้น ข้อมูลจะถูกส่งผ่านข้อมูลออกมา บนบัสข้อมูลก่อนที่สัญญาณ IOW (ในกรณีที่ต้องการส่งข้อมูลให้กับพอร์ต) หรือ MEMW (ในกรณีที่ต้องการส่งข้อมูลให้กับหน่วยความจำ) จะเปลี่ยนจากลอจิก "0"เป็นลอจิก "1" (ขอบขาขึ้น) ซึ่งโดยทั่วไปขอบขาขึ้นของสัญญาณ IOW หรือ MEMW นี้จะถูกใช้เพื่อสั่งให้พอร์ต I/O หรือหน่วยความจำที่มี แอดเดรสตรง กับค่าแอดเดรสบนบัสแอดเดรสนั้น รับข้อมูลไปเก็บไว้ สำหรับในบัสไซเคิล ของการอ่านข้อมูลที่สร้างขึ้นโดย 8088 นั้น พอร์ต I/O หรือหน่วยความจำที่ถูกอ้างถึงจะต้องส่งข้อมูลออกมาบนบัสข้อมูล ก่อนที่สัญญาณ IOR (ในกรณีที่ ต้องการอ่านข้อมูลจากพอร์ต) หรือ MEMR (ในกรณีที่ต้องการอ่าน ข้อมูลจากหน่วยความจำ) จะเปลี่ยนจากลอจิก "0" เป็นลอจิก "1"(ขอบขาขึ้น)

2.3.3 RESET DRV (ขา B2) :

ขาสัญญาณนี้เป็นเอาท์พุต ซึ่งจะแอคทีฟ (ลอจิก "1") ในช่วงที่เราเริ่มจ่ายไฟให้กับระบบ และ จะยังคงแอคทีฟไปจนกว่าระบบต่าง ๆ ภายใน IBM/PC จะพร้อมที่จะทำงาน ถ้าระดับแรงดันของแหล่งจ่ายไฟตกลง สัญญาณนี้ก็จะถูกทำให้ แอคทีฟเช่นกัน โดยทั่วไปแล้วสัญญาณนี้ จะถูกนำไปใช้ใน การรีเซ็ทวงจรอินเตอร์เฟส หรือ อุปกรณ์ I/O ต่าง ๆ ในช่วงที่เริ่มจ่ายไฟให้กับระบบ ซึ่งจะเป็นการทำให้วงจร หรือ อุปกรณ์เหล่านั้น ถูกปรับให้อยู่ ในสภาวะที่แน่นอน ก่อนที่จะเริ่มต้น การทำงานในระบบ (สภาวะนี้เป็นสภาวะที่เราทราบ และต้องการให้วงจรทำงานในขณะที่ระบบถูก รีเซ็ท)

2.3.4 ALE (Address Latch Enable; ขา B28) :

ขาสัญญาณนี้เป็นสัญญาณเอาท์พุตที่ 8288 Bus Controller สร้างขึ้นเพื่อใช้สำหรับแสดงการ เริ่มต้นของบัสไซเคิล และ แสดงให้อุปกรณ์ภายนอกทราบว่า แอดเดรสที่ 8088 ต้องการจะติดต่อด้วยนั้น ถูกส่งออกมาบนบัสแอดเดรสแล้ว โดยที่สัญญาณ ALE นี้จะเปลี่ยนจากลอจิก "1" เป็นลอจิก "0" เมื่อ ค่าแอดเดรสที่ถูกต้อง ถูกส่งออกมาบนบัสข้อมูลเรียบร้อยแล้ว ดังนั้น ขอบขาลงของสัญญาณ ALE นี้จะถูกใช้ใน การแลทช์ค่าแอดเดรสจากบัสแอดเดรส/ข้อมูล (Address/Data Bus ; AD0-AD7) ของ 8088 ทำให้ สามารถแยกค่าแอดเดรส (A0-A19) และข้อมูล (A0-A7) ออกจากกันได้ อย่างไรก็ตาม สัญญาณ

ALE จะแอคทีฟเฉพาะ ในบัสไซเคิลที่สร้างขึ้นโดย 8088 เท่านั้น โดยจะไม่ แอคทีฟในระหว่างขบวนการ DMA

2.3.5 MEMW (Memory Write; ขา B11) :

ขานี้เป็นเอ้าท์พุตแอคทีฟที่ลอจิก "0" ซึ่ง 8288 Bus Controller สร้างขึ้นในระหว่าง บัสไซเกิลในการ เขียนข้อมูล ลงในหน่วยความจำของ 8088 สัญญาณ MEMW นี้จะ ถูกส่งออกมา เพื่อให้หน่วยความจำ ที่แอดเดรส ตรงกับค่าแอดเดรสบนบัสแอดเดรสนั้น ทำการรับข้อมูลที่อยู่ บนบัสข้อมูลไปเก็บไว้ โดยทั่วไปหน่วยความจำจะรับข้อมูลในช่วงของขาขึ้นของสัญญาณ MEMW สำหรับในระหว่างขบวนการ DMA นั้น 8237A-5 DMA-Controller จะทำการควบคุมบัสต่าง ๆ ของระบบแทน 8088 และ สัญญาณ MEMW จะถูกใช้บัสไซเคิลของการ เขียนข้อมูลลง ในหน่วยความจำ (ข้อมูลถูกส่งจากอุปกรณ์ I/O ไปให้กับหน่วยความจำ)

2.3.6 MEMR (Memory Read ; ขา B12) :

ขานี้เป็นเอาท์พุตจาก 8288 ซึ่งสัญญาณนี้จะแอคทีฟ ลอจิก "0" ในระหว่างบัสไซเคิลของการอ่านข้อมูลจากหน่วยความจำของ 8088 เพื่อให้หน่วยความจำที่มีแอดเดรสตรงกับค่าแอดเดรส บนบัสแอดเดรสนั้น ทำการส่งข้อมูลออกมา บนบัสข้อมูล โดยหน่วยความจำ นั้นจะต้องส่งข้อมูลออกมาในช่วงเวลา 30 nanosec ก่อนที่สัญญาณ MEMW จะกลับเป็นลอจิก "1" ทั้งนี้ก็เพื่อให้ 8088 ได้รับข้อมูลที่ถูกต้อง สำหรับ ในระหว่างขบวนการ DMA นั้น DMA-Controller จะควบคุมบัสต่าง ๆ ของระบบแทน 8088 และ สัญญาณ MEMR จะถูกใช้ในบัสไซเคิล ของการอ่านข้อมูล จาก หน่วยความจำ (ข้อมูล ถูกส่งจากหน่วย ความจำไปให้กับอุปกรณ์)

2.3.7 IOR (I/O Read; ขา B14) :

ขาสัญญาณนี้เป็นเอาท์พุตแอคทีฟที่ลอจิก"0" ที่สร้างขึ้นโดย 8288 Bus Controller เพื่อใช้ในการแสดงว่า บัสไซเคิลที่เกิดขึ้นนี้ เป็นบัสไซเคิล ของการ อ่านข้อมูล จากพอร์ต I/O เพื่อให้พอร์ต I/O ที่มีแอดเดรส ตรงกับแอดเดรสบนบัส แอดเดรสนั้นส่งข้อมูลออกมา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารนี้ทุกครั้งที่มีการนำไปใช้

บนบัสข้อมูล โดย ข้อมูลจะต้องถูกส่งออกมา บนบัสข้อมูลก่อนขอบขาขึ้นของสัญญาณ IOR ประมาณ 30 nano sec เพื่อให้มั่นใจว่า 8088 สามารถรับข้อมูลได้ถูกต้อง สำหรับในขบวนการ DMA 8237A - 5 DMA Controller จะทำการสร้างสัญญาณ IOR เอง โดยที่ค่าแอดเดรสที่อยู่บนบัสแอดเดรส จะเป็นค่า แอดเดรสของหน่วยความจำ (แทนที่จะเป็นแอดเดรสของพอร์ต I/O) ที่พอร์ต I/O ที่ขอ DMAต้องการจะนำข้อมูลไปเก็บ การที่พอร์ตใดจะส่งข้อมูลออกมาบนบัส ข้อมูลนั้นจะอาศัยสัญญาณ DACK จาก DMA Controller เป็นตัวกำหนด เช่น กรณีที่สัญญาณ DACK แอคทีฟก็จะแสดงว่า พอร์ต I/O ที่ ต้องส่งข้อมูลออกมาบนบัสข้อมูลก็คือ พอร์ต I/O ที่ขอ DMA ผ่านทางแชนแนลที่ 1 (DRQ1) เป็นต้น

2.3.8 IOW (I/O Write: ขา B13) :

ขาสัญญาณนี้เป็นเอาท์พุตแอคทีฟ ที่ลอจิก "0" ซึ่งถูกสร้างขึ้น โดย 8288 Bus Controller เพื่อใช้แสดงว่า บัสไซเคิลที่เกิดขึ้นนี้ เป็นบัสไซเคิลของการ เขียนข้อมูลลงบนพอร์ต I/O เพื่อให้ I/O ที่มีแอดเดรสตรงกับ แอดเดรสบนบัสแอดเดรสนั้น รับข้อมูลที่อยู่บนบัสข้อมูล ไปเก็บไว้ อย่างไรก็ตาม เนื่องจากในช่วงเวลาที่สัญญาณ IOW นี้ แอคทีฟ ลอจิก "0" นั้นข้อมูล บนบัสข้อมูลอาจจะยัง ไม่สมบูรณ์ ดังนั้นในการออกแบบจึงควรใช้ขอบขาขึ้นของสัญญาณ IOW แทนขอบขาลงในการทำให้พอร์ต I/O ที่เกี่ยวข้องรับข้อมูลไปเก็บไว้ เพื่อให้ข้อมูลบนบัสข้อมูลสมบูรณ์เสียก่อน สำหรับในขบวนการ DMA นั้น DMA-Controller จะทำการสร้าง สัญญาณ IOW เอง โดยที่ค่าแอดเดรสที่อยู่บนบัส แอดเดรสจะเป็นค่าแอดเดรสของหน่วยความจำที่พอร์ต I/O ที่ขอ DMA ต้องการจะอ่านข้อมูล

2.3.9 +5Vdc (ขา B3 และ B29) :

ขาทั้งสองนี้ ต่อกับ แหล่งจ่ายไฟ DC. + 5V ของระบบ โดยจะมีค่าความเที่ยงตรง (Regulate) + 5% คือ อยู่ในช่วง +4.75 ถึง +5.25 Vdc.

2.3.10 +12Vdc (ขา B9) :

ขานี้จะ ต่อกับ แหล่งจ่ายไฟ DC. + 12 V. ของระบบ โดยจะมีค่าความเที่ยงตรง (Regulate) + 5% คือ อยู่ในช่วง +11.4 ถึง +12.6 Vdc.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.3.11 -5Vdc (ขา B5) :

ขานี้จะ ต่อกับแหล่งจ่ายไฟ DC. - 5V. ของระบบ โดยจะมีค่า ความเที่ยงตรง (Regulate) + 10% คือ อยู่ในช่วง -5.5 ถึง -4.5 Vdc.

2.3.12 -12Vdc (ขา B7) :

ขานี้จะ ต่อกับ แหล่งจ่ายไฟ DC. - 12 V. ของระบบ โดยจะมีค่าความเที่ยงตรง (Regulate) + 10% คือ อยู่ในช่วง -13.2 ถึง -10.8 Vdc.

2.3.13 GND (ขา B1,B10 และ B13) :

ขาทั้ง 3 นี้จะต่อเข้ากับกราวด์ (Ground) ของระบบ การจัดสัญญาณบนสล็อตของ IBM PC/XT นั้นจะมี สล็อตสำหรับเชื่อมต่อกับวงจรภายนอกได้มากขึ้นคือใน IBM PC/XT จึงทำการเพิ่ม จำนวนสล็อต บนเมนบอร์ดขึ้นเป็น 8 สล็อต จะยังคงเหมือนกันใน IBM PC เพียงแต่สัญญาณต่าง ๆ ที่จะถูกส่งออกมายังขาของสล็อตทั้ง 8 นั้น จะถูกต่อผ่านวงจรขับกระแส (Buffer) ก่อน และ ในสล็อตที่ 8 นี้ขา B8 จะถูกใช้งานด้วย โดยจะถูกใช้ เป็นขา CARDSLCTD (หรือ Card Select) ซึ่งขาสัญญาณนี้จะเป็นสัญญาณอินพุตจากวงจรภายนอกที่เสียบอยู่บน สล็อตที่ 8 เพื่อให้วงจรบนเมนบอร์ดทราบว่าการ์ดที่อยู่บนสล็อตนี้ ถูกเลือกใช้งานอยู่ซึ่งจะทำให้ Driver บนเมนบอร์ดทำการ อ่าน หรือ ส่งข้อมูล ไปยังสล็อตที่ 8 เนื่องจากใน IBM/PC ได้ใช้งานเส้นแอดเดรสเพียง 10 เส้น (คือ A0-A9) ดังนั้นจึงสามารถที่จะอ้าง แอดเดรสของพอร์ตได้สูงสุดเพียง 1024 พอร์ต (จากจำนวน 64 K. พอร์ต) เท่านั้น นอกจากนี้ ในกรณีที่เป็น การอ่านข้อมูลจากพอร์ตของ IBM/PC ข้อมูล ในบิต A9 จะถูกจัดให้มีหน้าที่ในการแบ่งพอร์ตทั้ง 1024 พอร์ต ออกเป็น 2 ส่วน (ส่วนละ 512 พอร์ต) อีกด้วย กล่าวคือถ้าข้อมูลในบิต A9 เป็น "0" แล้วเราจะทำการอ่านข้อมูลได้เฉพาะ จากพอร์ตของอุปกรณ์ หรือ ชิพซัพพอร์ตต่าง ๆ ที่อยู่บนเมนบอร์ด(Mainboard) ของ IBM/PC เช่น 8253-5,8237-5 หรือ 8259A เท่านั้น แต่ถ้าข้อมูลในบิต A9 นี้เป็น "1" ก็จะทำการอ่านข้อมูลได้เฉพาะ จากพอร์ตที่อยู่บนการ์ดต่าง ๆ เท่านั้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.4 การกำหนดหมายเลขพอร์ตใน IBM/PC

สำหรับไมโครโปรเซสเซอร์ 16 บิตนี้จะต้องมีความเข้ากันได้ หรือ คอมพาทิเบิ้ล (Compatile) ทางฮาร์ดแวร์กับระบบไมโครคอมพิวเตอร์ 16 บิต IBM/PC ดังนั้นส่วนของฮาร์ดแวร์ จะต้องกำหนดให้ได้หมายเลขพอร์ต เหมือนกัน หมายเลขพอร์ตที่ใช้จะมีลักษณะเป็นแบบกำหนด ตายตัว จะเคลื่อนย้ายไปที่อื่นไม่ได้ เพราะอาจจะมีปัญหาสำหรับโปรแกรมหรือซอฟต์แวร์ ในบางระดับที่จัดการเกี่ยวกับพอร์ตโ ดยตรงได้

หมายเลขพอร์ตที่ใช้ได้รับการกำหนดดังนี้

000H-00FH	พอร์ตของชิฟ DMA 8237A
020H-021H	พอร์ตของชิฟอินเตอร์รัพต์คอนโทรลเลอร์ 8259A
040H-043H	พอร์ตของไอซีไทม์เมอร์ 8253
060H-063H	พอร์ตของ 8255A ที่อยู่บนเมนบอร์ด
080H-083H	พอร์ตของดีเอ็มเอที่ใช้กำหนดเพจ (Page Register)
0AX	รีจิสเตอร์ที่ใช้สำหรับ MNI
0CX	สงวนไว้
0EX	สงวนไว้
200H-20FH	พอร์ตที่ใช้ในการควบคุม
210H-217H	ส่วนขยายเพิ่มต่อ
220H-24FH	สงวนไว้
278H-27FH	สงวนไว้
2F0H-2F7H	สงวนไว้
2F8H-2FFH	พอร์ตสื่อสาร COM 2
300H-31FH	โปรโตไทม์การ์ด
320H-32FH	วงจรควบคุมฮาร์ดดิสก์
378H-37FH	เครื่องพิมพ์แบบขนาน
380H-38FH	วงจรสื่อสาร SDLC
3A0H-3AFH	สงวนไว้
3B0H-3BFH	วงจรควบคุมการแสดงผลบน CRT แบบโมโนโครม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น รวมทั้งไม่ให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3C0H-3CFH	สงวนไว้
3D0H-3DFH	วงจรควบคุมการแสดงผลบน CRT แบบสี
3E0H-3EFH	สงวนไว้
3F0H-3F7H	วงจรควบคุมดิสก์ไดร์ฟ
3F8H-3FFH	วงจรควบคุมพอร์ตสื่อสาร COM 1

2.5 ไอซี 8255 โปรแกรมมาเบิ้ล เพอริเพอรัล อินเตอร์เฟสซิ่ง (Progammable Pheripheral Interfacing)

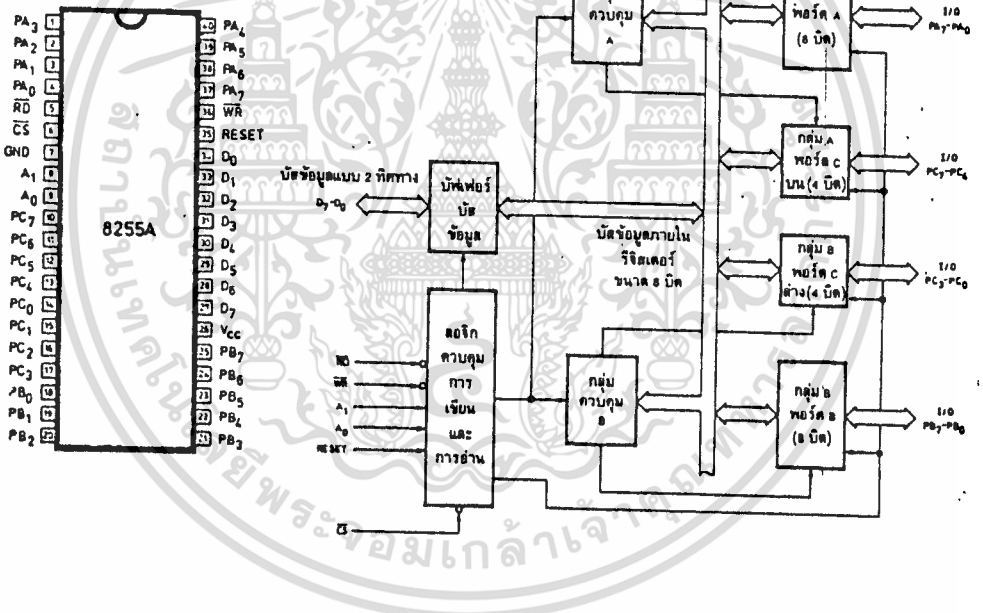
เป็นไอซี พอร์ทที่สามารถโปรแกรมการทำงานได้ ในการนำเอาไมโครโปรเซสเซอร์ไปใช้งานนั้น จำเป็นต้องให้ ไมโครโปรเซสเซอร์สามารถติดต่อกับโลกภายนอกได้ ซึ่งก็คือให้มันสามารถ ส่งสัญญาณมาควบคุม อุปกรณ์ต่าง ๆได้ เช่น สเต็ปปิ้งมอเตอร์,ควบคุมอุปกรณ์ไฟฟ้า ต่าง ๆ ส่วนที่ทำได้ไมโครโปรเซสเซอร์สามารถติดต่อกับโลกภายนอกได้ที่รู้จักกันดีคือ พอร์ท (Port) ซึ่งก็มีอยู่หลายลักษณะด้วยกัน เช่นเป็นตัวไอซีแบบไดรสเตท (Tri-state) เบอร์ 74LS244 หรือ พวก แลทต์ (Latch) เช่น 74LS374 เหล่านี้สามารถนำเอามาต่อใช้งานกับ CPU ได้ง่ายที่สุด โดย ตัว CPU จะเป็นตัวควบคุมการอ่านเขียนพอร์ท หากเป็นการอ่านข้อมูล จากพอร์ทก็มักจะใช้ ไอซีแบบ ไดรสเตท เป็นอินพุทโดยตัว CPU จะส่งสัญญาณไปเปิดเกทของ ไดรสเตท นี้ให้ข้อมูลเข้ามาสู่สายข้อมูล (Data Bus) และ เข้าสู่ไมโครโปรเซสเซอร์ หรือ CPU ต่อไป แต่สำหรับ พอร์ทเอ้าพุทก็จะใช้ แลทต์ฟลิบ-ฟลอบทำหน้าที่ รับสัญญาณข้อมูลจาก ไมโครโปรเซสเซอร์มาแลทต์ไว้ที่ตัวมัน (CPU ส่ง สัญญาณมาทริก) เพื่อให้อุปกรณ์ภายนอก นั้นรับสัญญาณจากตัวแลทต์นี้ไปอีกทีหนึ่ง ที่ทำเช่นนี้เพราะตัวไมโครโปเซสเซอร์ หรือ CPU นี้ทำงานเร็วมากซึ่งในช่วงของการส่งข้อมูลออกพอร์ทจะใช้เวลาไม่กี่ไมโครเซท (μS) ซึ่งอาจจะทำให้ อุปกรณ์ภายนอกรับไม่ทัน มีบริษัทต่าง ๆ เล็งเห็นความสำคัญ ของการติดต่อระหว่าง CPU กับอุปกรณ์ภายนอก นี้จึงทำได้ ทำไอซีสำเร็จรูปขึ้นหลายเบอร์ และ ที่จะขอนำมากล่าวในที่นี้ก็ คือ ไอซีเบอร์ 8255 ซึ่งเป็นของบริษัท อินเทล ซึ่งได้ออกแบบมาใช้กับ CPU เบอร์ 8080 แต่เราสามารถนำมาประยุกต์ใช้กับเบอร์อื่น ๆ ได้โดยไม่ยาก สาเหตุที่ 8255 เป็นที่นิยมมาก ก็เพราะว่ามันสามารถถูก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

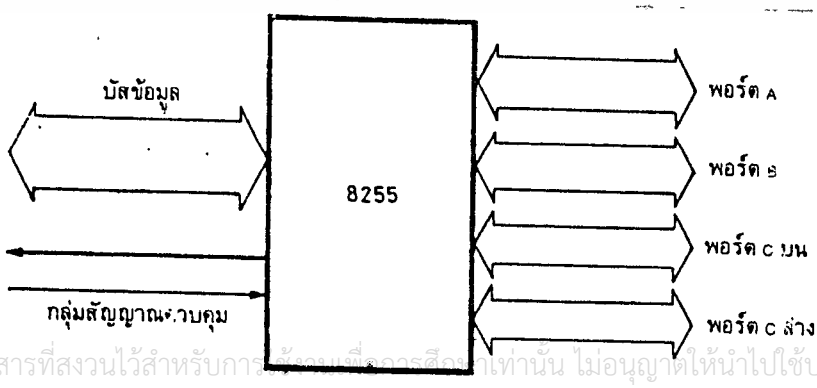
โปรแกรมให้ทำงานใน ลักษณะต่าง ๆ ไม่ว่าจะเป็น อินพุท , เอ้าท์พุท หรือ แม้แต่แบบ แฮนด์เชคกิ้ง (Handshaking) ได้ทั้ง ยังราคาถูกอีกต่างหาก

2.5.1 ลักษณะทั่วไปของ 8255

เป็นไอซีขนาด 40 ขา ตัวแบนโดยแยกเป็นลักษณะของบล็อกง่าย ๆ ดังรูปที่ 2.5.1 คือ จะมี พอร์ทให้ใช้งานได้ถึง 3 พอร์ท (เป็นขนาด 8 บิท) พอร์ท A ,พอร์ท B,พอร์ท และ พอร์ท C โดย พอร์ท C นี้สามารถแยกได้เป็น 2 ส่วนคือพอร์ท Cบนตั้งแต่ PC_4-PC_7 จำนวน 4 บิทและพอร์ท C ล่างตั้งแต่ PC_0-PC_3 โดยพอร์ททุกพอร์ท (A,B,C) สามารถโปรแกรมได้ให้เป็นอินพุทหรือเอ้าท์พุทซึ่งจะได้กล่าวถึงการ โปรแกรมในรายละเอียดต่อไป



รูปที่ 2.5.1 บล็อกไดอะแกรมของไอซีเบอร์ 8255



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น ไม่อนุญาตให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 2.5.2 แสดงโครงสร้างภายในของไอซี 8255

ในรูปที่ 2.5.2 จะเห็นโครงสร้างภายในที่แสดงถึงกลุ่มควบคุมที่มีอยู่ 3 กลุ่มคือ

กลุ่มควบคุมชุด A จะควบคุมพอร์ท A และพอร์ท C บน
กลุ่มควบคุมชุด B จะควบคุมพอร์ท B และพอร์ท C ล่าง
กลุ่มควบคุมโลจิกการเขียนและอ่าน

การทำงานของ 8255 จะใช้สัญญาณควบคุมจากตัวไมโครโปรเซสเซอร์มาควบคุมโดยจะมีการส่งคำสั่ง (Control) มาที่กลุ่มควบคุมชุด A,B แล้วกลุ่มที่ควบคุมชุดนี้ก็จะสั่งต่อไปที่พอร์ท เพื่อเป็นไปตามข้อกำหนดของคำสั่งนั้น ๆ เช่น ให้พอร์ท A เป็นอินพอร์ท B เป็นเอ้าท์พุทเหล่านี้เป็นต้น ส่วนกรณีเมื่อการอ่านเขียนพอร์ทจาก CPU กลุ่มควบคุมโลจิกการเขียนอ่าน จะเป็นตัวส่งสัญญาณไป บอกแก่กลุ่มที่ควบคุม ในแต่ละชุด อีกทีทั้งนี้แล้วแต่ว่า CPU จะมีการอ่าน และ การเขียนพอร์ทของกลุ่ม ควบคุมชุดใดต่อไปเรามาดูถึงความหมายขาต่าง ๆ ของไอซี 8255 เพื่อที่จะได้ใช้งานได้อย่างถูกต้องต่อไป

D0-D7 เป็นขาข้อมูลของ 8255 ที่ใช้ติดต่อกับตัวไมโครโปรเซสเซอร์ซึ่งข้อมูลที่จะเข้าออกสู่พอร์ทต่าง ๆ ของ 8255 จะต้องผ่านขาข้อมูลนี้

CS เป็นขาอินพุทที่รับสัญญาณลอจิก "0" จากภายนอกเพื่อแสดงว่าต้องการเลือกใช้ไอซีเบอร์นี้ หากได้รับโลจิก "1" ก็จะทำให้ไอซีตัวนี้ไม่ทำงานคือไม่รับสัญญาณใด ๆ ทั้งสิ้น

RD เป็นขาอินพุทที่รับสัญญาณจากตัวไมโครโปรเซสเซอร์ โดยหากมีโลจิกเป็น "0" จะเป็นการแสดงว่า CPU ต้องการที่จะอ่านข้อมูลจากตัว 8255

WR เป็นขาอินพุทที่รับสัญญาณจากตัวไมโครโปรเซสเซอร์ โดยหากมีโลจิกเป็น "0" ก็จะเป็นการแสดงว่า CPU ต้องการที่จะเขียนข้อมูลจากตัว 8255

A0-A1 เป็นอินพุทที่รับแอดเดรสจากตัวไมโครโปรเซสเซอร์ที่ถอดระหัสตำแหน่งของ 8255 เรียบร้อยแล้วโดยจะมีตำแหน่งใช้งาน 4 ตำแหน่ง เพื่ออ่านเขียนรีจิสเตอร์ (พอร์ท) ของ 8255 ที่มีอยู่ด้วยกัน 4 ตัว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

RESET เป็นขาอินพุทที่รับสัญญาณจากภายนอกเข้ามาทำการรีเซทตัว 8255 โดยหากได้รับโลจิก "1" จะทำให้พอร์ททุกพอร์ทเป็น อินพุทพอร์ทหมดทั้งนี้เพื่อไม่ต้องการให้มีสัญญาณออกไปกวนต่อ ระบบภายนอกเพื่อ 8255 ได้รับสัญญาณรีเซท

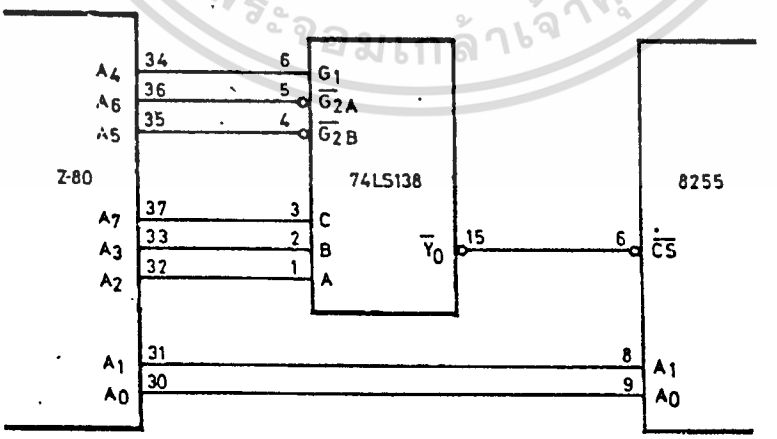
PA_0_PA_7 เป็นขาสัญญาณพอร์ท A ที่ใช้ติดต่อกับโลกภายนอก

PB_0-PB_7 เป็นขาสัญญาณพอร์ท B ที่ใช้ติดต่อกับโลกภายนอก

PC_0_PC_7 เป็นขาสัญญาณพอร์ท C ที่ใช้ติดต่อกับโลกภายนอก ซึ่งพอร์ทนี้จะแบ่งออกเป็น 2 กลุ่ม คือ PC_0_PC_3 และPC_4_PC_7 ซึ่งสามารถโปรแกรมแยกกันได้อีกต่างหาก

2.5.2 การต่อใช้งาน 8255

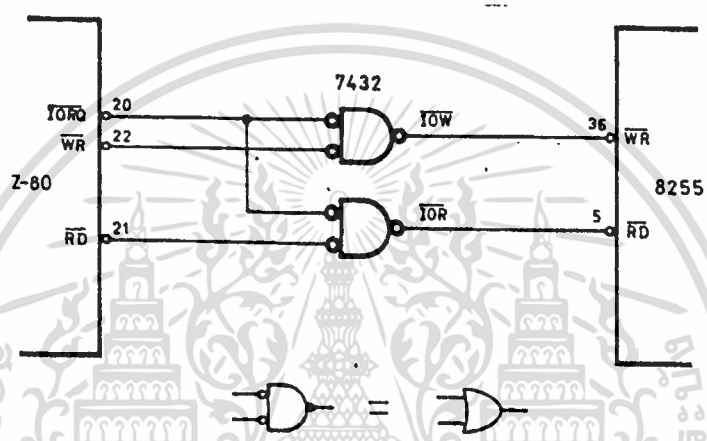
หากดูที่ขาของ 8255 ที่ รูปที่ 2.5.2 แล้วสังเกตเห็นได้ว่าเราสามารถต่อขา 8255 บางส่วนได้โดยตรงกับขาไมโครโปรเซสเซอร์เลข (Z-80) เช่นขา D0-D7,A0-A1เป็นต้น หากแต่บางขา เราจำเป็นต้องมีการ ดัดแปลงสัญญาณที่ได้จาก CPU (ซึ่งกรณีนี้เราใช้เบอร์ Z-80) เสียก่อนโดยหากเราถอดรหัสแอดเดรสของ 8255 ให้เป็นพอร์ทที่แอดเดรส 10H, 11H ,12H ,13H เราจะสามารถทำ ดีโค็ดเดอร์ได้โดยง่ายดังรูปที่ 2.5.3



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น ทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 2.5.3 การกำหนดแอดเดรสให้กับ 8255

สังเกตได้ว่า CS จะแอดดีฟทุกครั้งหาก Z-80 อ้างพอร์ทที่แอดเดรส 000100 XX โดยค่าของ XX คือ A0,A1 ที่เราจะต่อเข้ากับ 8255 เพื่อทำการเลือกรีจีสเตอร์ควบคุม และ พอร์ททั้งสาม ของ 8255 สัญญาณการควบคุมอีกส่วนที่สำคัญก็ คือสัญญาณควบคุมการอ่านเขียนพอร์ท ของ 8255 โดยหาก WRได้รับแอดดีฟ "0" ก็จะเป็นการเขียนข้อมูลจาก CPU เข้าสู่ตัว 8255 หรือ ออกสู่พอร์ทที่ต้องการ และ เช่นเดียวกันหาก RD ได้รับโลจิก "0" ก็จะเป็นการอ่านข้อมูลพอร์ทจาก ตัว 8255 เข้าสู่ตัว CPU ซึ่งหากต่อ 8255 เข้ากับตัว Z-80 แล้ว เราจะเป็นต้องทำสัญญาณ การอ่าน เขียนพอร์ทของ Z-80 ให้ถูกต้องเสียก่อน โดยต่อดัง รูปที่ 2.5.4



รูปที่ 2.5.4 วงจรการต่อสายสัญญาณควบคุมการเขียนและการอ่าน 8255

และสุดท้ายคือสัญญาณ RESET ซึ่งของ 8255 จะรับแอดดีฟที่โลจิก "1" ซึ่งตัว Z-80 จะให้สัญญาณ RESET แอดดีฟ "0" ฉะนั้นเราจะต้องนำมาผ่านอินเวอร์เตอร์ก่อนที่จะได้ ลักษณะการต่อร่วมกับ CPU ดังนี้

2.5.2.1 การโปรแกรม 8255

เราได้ทราบมาแล้วในรูปที่ 2.5.2 ว่าโครงสร้างภายในของ 8255 มีกลุ่มควบคุมอยู่ 3 กลุ่ม ซึ่งทั้งสามกลุ่มนี้จะทำงานร่วมกัน ดังที่กล่าวมา และเราสามารถจะควบคุมการทำงานของพอร์ท จาก CPU ได้โดยสั่งงานมาที่กลุ่มควบคุมดังกล่าว แต่ตัว CPU จะมองเห็น 8255 เป็นพอร์ทด้วยกัน โดยแต่ละพอร์ท เสมือนเป็น รีจีสเตอร์ที่ CPU สามารถจะทำการอ่าน/เขียนได้ แต่ละพอร์ท จะอยู่คนละแอดเดรสกัน ดังที่เราได้ทำการดีโค๊ดให้ 8255 ที่แอดเดรส 10H ,11H ,12H ,13H (ตามสัญญาณ A0-A1) และ เราจะได้ตำแหน่งของ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

พอร์ท 8255 แต่ละตัวดังนี้

10H ==========} พอร์ท A

11H ==========} พอร์ท B

12H ==========} พอร์ท C

13H ==========} พอร์ท Control

ซึ่งหากมีการอ่านเขียนไปยังพอร์ทดังกล่าวก็จะใช้ร่วมกับสัญญาณ RD,WR โดย WR หมายถึง เอ้าท์พุทข้อมูล และ RD แอดตีฟ หมายถึง อินพุทข้อมูล ดังนั้นเราจะโลจิกที่ขาของ 8255 ในลักษณะต่าง ๆ ดังนี้

AD	WR	AAI	AO	ความหมาย
1	0	0	0	เขียนพอร์ด A ซึ่งเป็นข้อมูล
0	1	0	0	อ่านพอร์ด A ซึ่งเป็นข้อมูล
1	0	0	1	เขียนพอร์ด B ซึ่งเป็นข้อมูล
0	1	0	1	อ่านพอร์ด B ซึ่งเป็นข้อมูล
1	0	1	0	เขียนพอร์ด C ซึ่งเป็นข้อมูล
0	1	1	0	อ่านพอร์ด C ซึ่งเป็นข้อมูล
1	0	1	1	เขียนข้อมูลซึ่งเป็นรหัสข้อมูล
0	1	1	1	อ่านเข้ามาซึ่งไม่มีความหมาย ใด

รูปที่ 2.5.5 แสดงตารางของโลจิกเมื่อทำการติดต่อกับ 8225

การใช้งานเราจะต้องส่งรหัสควบคุม (Control code) เข้าไปยังพอร์ทควบคุมหรือเรียกอีกอย่างหนึ่งว่า รีจีสเตอร์ควบคุม ซึ่งจะเป็นข้อมูลขนาด 1 ไบท์ ส่งไปที่ แอดเดรส 13H กรณีนี้เราถอดรหัสไว้ที่ 13H โดยความหมายของแต่ละบิทที่เราส่งไปเปรแกรมทำงานเป็นดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บิต D7 เป็นบิตที่แสดงว่าในไบท์นี้เป็นรหัสควบคุมถ้าเป็น "1" โดยแต่ละบิตจะมีผลต่อการเปลี่ยนแปลงโหมดต่าง ๆ ของ 8255 หากเป็น "0" จะเป็นเซตบิทของพอร์ท C

บิต D6,D5 เป็นการเลือกโหมดของพอร์ท A ซึ่งจะมีอยู่ด้วยกัน 3 โหมด คือ 0,1,2

บิต D4 เป็นการกำหนดให้พอร์ท A ให้เป็นอินพุทหรือเอ้าพุท โดยหากเป็น "1" ก็จะเป็นอินพุท หากเป็น"0"ก็แสดงว่าเป็นเอ้าท์พุท

บิต D3 เป็นการกำหนดให้พอร์ท C บนให้เป็นอินพุทหรือเอ้าท์พุทโดยหากเป็น "1" ก็จะเป็นอินพุท หากเป็น "0" ก็แสดงว่าให้เป็นเอ้าท์พุท

บิต D2 เป็นการกำหนดโหมดการทำงานของพอร์ท B โดยหากเป็น "0" หมายถึงเลือกให้พอร์ท B ทำงานในโหมด 0 หากเป็น "1" เป็นการเลือกให้พอร์ท B ทำงานในโหมด 1

บิต D1 เป็นการกำหนดให้พอร์ท B ให้เป็นอินพุทหรือเอ้าพุทโดยหากเป็น "1" ก็จะเป็นอินพุท หากเป็น "0" ก็แสดงว่าให้เป็นเอ้าท์พุท

บิต D0 เป็นการกำหนดให้พอร์ท C ล่าง ให้เป็นอินพุทหรือเอ้าพุทโดยหากเป็น "1" ก็จะเป็นอินพุทหากเป็น "0" ก็แสดงว่าให้เป็นเอ้าท์พุท

การโปรแกรมจะเริ่มจากการส่งค่ารหัสควบคุม 1 ไบท์ ดังที่กล่าวนี้ไปสู่พอร์ทควบคุม หลังจากนั้น หากต้องการเรียกไปถึงพอร์ทใด ก็สามารถอ้างได้ตามแอดเดรสทันที เช่น ต้องการโปรแกรมให้พอร์ท A ,B ,C ทั้งหมดให้เป็น เอ้าท์พุทพอร์ท เราก็จะได้รหัสควบคุมเป็น 10000000 หรือ 80H เราก็จะสั่งเป็น

```
LD A,080H        ; กำหนดรหัสควบคุม
OUT (013H),A     ; เป็นการส่งรหัสควบคุมสู่รีจิสเตอร์ควบคุม 8255
```

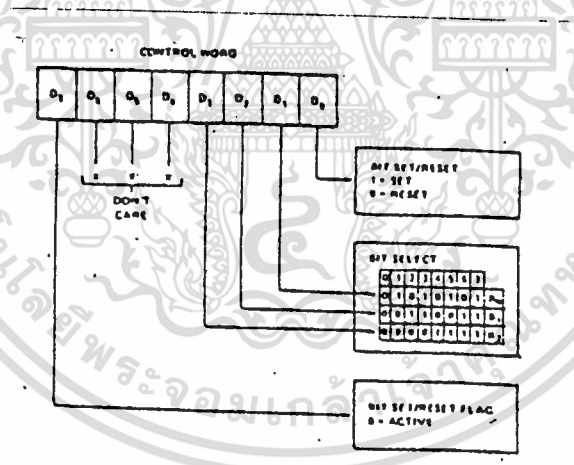
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากนี้ทั้งสามพอร์ทก็จะเป็นเอ้าท์พุทพอร์ทให้เราตามต้องการเมื่อเราจะส่งค่า ออกไปก็จะสามารถจะทำได้โดยง่ายเช่น ต้องการส่งค่า 088H ออกไปที่พอร์ท B,C เราจะทำดังนี้

```
LD A,088H       ;ค่าข้อมูลต้องการส่ง
OUT (011H),A    ;ส่งออกไปพอร์ท B
OUT (012H),A    ;ส่งออกไปพอร์ท C
```

2.5.2.2 กรณีพิเศษของรหัสควบคุม

ปกติรหัสที่เราต้องส่งไปถึงพอร์ทควบคุมหรือรีจิสเตอร์ควบคุมจะต้องเป็นการเซทโหมดพอร์ทอินพุท เอ้าท์พุท และ ให้รหัสนั้นบิต 7 จะต้องเป็น "1" เสมอที่นี้หากบิต 7 นั้นเป็น "0" บ้าง จะเกิดอะไรขึ้น ถ้าหากบิต 7 เป็น "0" จะถูกส่งไปที่แอดเดรสของพอร์ทควบคุมแล้ว 8255 จะถือว่าเป็นคำสั่งของการเซท/รีเซทโดยจะมี ฟอร์แมทดังรูปที่ 2.5.6 นี้



รูปที่ 2.5.6 แสดงถึงระหัสควบคุมที่ใช้ในการเซท/รีเซท บิต

<u>ตัวอย่างเช่น</u>

```
LD A 80H            ;ระหัสควบคุมให้ทั้งสามพอร์ทเป็น OUTPUT
OUT (13H),A         ;ส่งสู่รีจีสเตอร์ควบคุม
LD A,00001001B      ;ระหัสควบคุมใช้เซทบิตที่ 4 ของพอร์ท C เป็น "1"
OUT (013H),A        ;บิต 4 พอร์ท C เป็น "1"
DEC A               ;เปลี่ยนเป็นรีเซท
OUT (013H),A        ;กำหนดให้บิต 4 พอร์ท C เป็น "0"
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จะเห็นว่าสามารถนำไปประยุกต์ใช้งานได้ เช่น เป็นตัวสร้างพัลซ์, และ กำหนด ใช้งาน เปิด-ปิด อุปกรณ์ด้วย พอร์ท C ที่มีค่าสั่งไม่ยุ่งยาก และ เป็นอิสระเป็นต้น

2.5.2.3 การทำงานในโหมด 0

จัดว่าเป็นโหมดพื้นฐาน ที่นิยมใช้กันมากที่สุดเนื่องด้วยความง่ายตรงไปตรงมา คือทั้งสามพอร์ทเราสามารถจะใช้พอร์ทใดเป็น อินพุท, เอ้าท์พุทได้ โดยเฉพาะพอร์ท C ยังแยกให้เป็น 2 ชุด ๆ ละ 4 บิต ซึ่งในแต่ละชุดนี้ก็สามารถจะโปรแกรมให้ชุดใดชุดหนึ่งเป็นอินพุทหรือ เอ้าท์พุทได้อีก ฉะนั้นโดย สรุปแล้ว ก็จะมีพอร์ทที่จะโปรแกรมให้เป็นอินพุทหรือเอ้าท์พุทได้เสมือน 4 พอร์ทคือ พอร์ท A ,พอร์ท B ,พอร์ท C บน และ พอร์ท C ล่าง (แต่โปรแกรมแยกเฉพาะบิตในแต่ละพอร์ทไม่ได้) ซึ่งหากเราดูที่ รหัสคำสั่งแล้วจะเห็นว่ามีอยู่ 4 บิต ที่ถูกกำหนดตายตัว คือ บิต 7 ,บิต 5 ,บิต 6 และ บิต 2 ส่วนที่เหลืออีก 4 บิต ก็คือข้อกำหนดว่าจะให้พอร์ทใดเป็น อินพุท/เอ้าท์พุท นั้นเอง ซึ่งหากเราให้พอร์ทใดเป็น อินพุท เราก็ใส่โลจิก "1" ที่บิตนั้น หรือ หากต้องการให้พอร์ทใดเป็น เอ้าท์พุทก็ให้ใส่ "0" ที่บิตนั้นจะเห็นได้ว่า จะมีความเป็นไปได้ในการ กำหนดลักษณะของพอร์ทในโหมดนี้อยู่ 16 อย่างด้วยกัน

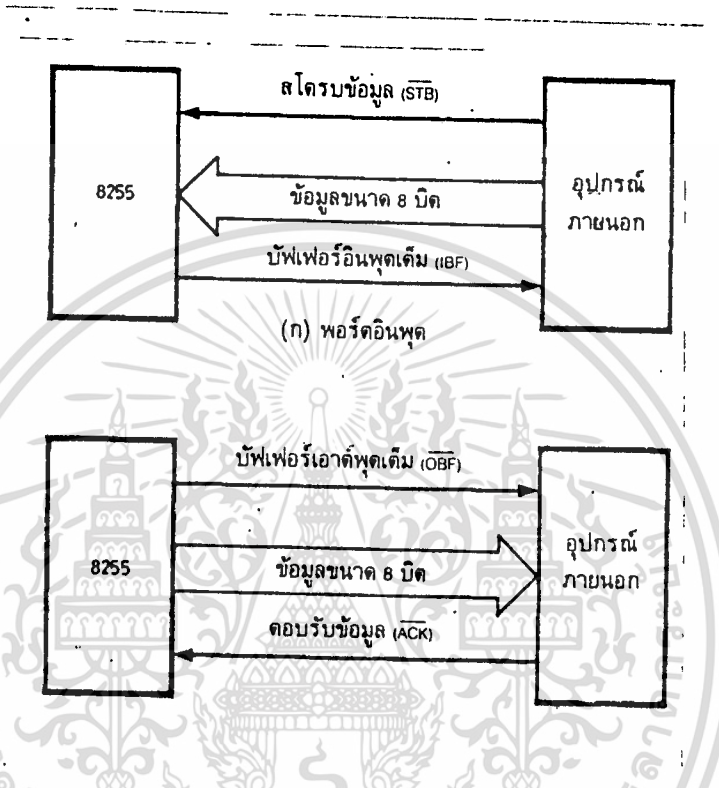
BIT	7	6	5	4	3	2	1	0
	1	0	0	X	X	0	X	X

รูปที่ 2.6.7 แสดงระหัสคำสั่งของ โหมด 0

2.5.2.4 การทำงานในโหมด 1

เป็นความสามารถพิเศษที่บริษัทผู้ผลิตเพิ่มขึ้นมาให้แก่ผู้ใช้คือ นอกจาก 8255 จะถูกใช้เป็น พอร์ทแบบขนาน ที่สามารถโปรแกรมให้เป็น อินพุท-เอ้าท์พุทพอร์ทได้แล้วยังสามารถให้ทำงานในเรื่องของการสื่อสารอัตโนมัติได้อีกด้วย นั่นคือในโหมด เป็นจะใช้ในการรับ-ส่งข้อมูล แบบมีการตรวจสอบสัญญาณก่อน (Handshakig) โดยใช้อินพุท เอ้าท์พุทของ

พอร์ท A และ พอร์ท B เป็นหลัก ส่วน พอร์ท C บนจะใช้เป็น ตัวตรวจสอบสัญญาณ (Handshaking) ของพอร์ท A ,พอร์ท C ล่าง ใช้เป็นตัวตรวจสอบสัญญาณสำหรับพอร์ท B ฉะนั้น การรับ - การส่งข้อมูล นอกจากมีข้อมูลแล้ว ก็จะมีสัญญาณควบคุมเพิ่มด้วยดังแสดงในรูปที่ 2.5.8



รูปที่ 2.5.8 โครงสร้างตัวตรวจสอบสัญญาณของพอร์ตอินพุตและพอร์ตเอาท์พุต

วิธีการทำ Handshake นี้จะมีประโยชน์มากเพราะปกติอุปกรณ์ภายนอกมักจะทำงานได้ช้ากว่า ตัวไมโครโปรเซสเซอร์อยู่แล้ว ด้วยวิธีนี้จะทำให้ตัวไมโครโปรเซสเซอร์สามารถที่จะติดต่อกับ อุปกรณ์ภายนอกด้วย 8255 ได้อย่างมีประสิทธิ์ภาพ

ในรูปที่ 2.5.8 นั้นจะแสดงให้เราเห็นถึงบล็อกของการติดต่อระหว่างอุปกรณ์ภายนอกกับ 8255 ซึ่งได้กำหนดให้พอร์ท A เป็นเอ้าท์พุทพอร์ท และ พอร์ท B เป็นอินพุท (เราอาจกำหนดให้อยู่ในลักษณะอื่นก็ได้) และ เช่นเดียวกับกับโหมด 0 ที่เราได้กล่าวมาแล้ว คือ ก่อนที่เราจะใช้งาน 8255 อันดับแรกเลยเราต้องการทำโปรแกรมมันก่อนโดยการส่งคำสั่งควบคุม ขนาด 1 ไบท์ ที่รีจีสเตอร์ควบคุมของ 8255 ก่อนซึ่งเรากำหนดให้พอร์ทเป็นดังรูปที่ 25 เราจะได้คำสั่งควบคุมเป็น ในบิทอื่น ๆ นอกจากบิทที่ 3,0 นั้นเราสามารถเข้าใจได้เพราะเคยกล่าวมาแล้ว ส่วนบิทที่ใส่เครื่องหมาย X นั้น เราต้องมาพิจารณาเพื่อใส่ค่าต่อไป คือ ลักษณะการทำงานโหมด 1 นี้ เราจะมีตารางที่แสดงถึงการใช้พอร์ท C เพื่อเป็นสัญญาณ

สงวนลิขสิทธิ์ ใช้ได้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ควบคุมทั้งพอร์ท A และ B ในกรณีอินพุท และเอ้าท์พุทดังนี้ฉะนั้นจากตารางเราจะเห็นได้ว่าในบิทที่ 3 นั้นขึ้นอยู่ว่าผู้ใช้ต้องการที่จะกำหนดให้ PC6,PC7 เป็นอินพุทหรือเอ้าท์พุท (โดยใส่ 1 เมื่อต้องการที่จะเป็นอินพุทและใส่ 0 เมื่อต้องการ ให้เป็นเอ้าท์พุท)ส่วนในบิทที่ 0 นั้นไม่สนใจเพราะ เราไม่สามารถกำหนดอะไรได้ (PC0-PC2 ถูกนำไปเป็นสัญญาณควบคุมแก่พอร์ท B แล้ว และ PC3 ถูกนำไปเป็นสัญญาณควบคุมแก่พอร์ท A แล้วเช่นกัน)

1.ลำดับสัญญาณในกรณีเอ้าท์พุทพอร์ทโหมด 1

การเอ้าท์พุทข้อมูลหมายถึงการส่งข้อมูลจากตัว CPU ออกมาสู่ตัว 8255 ที่พอร์ทนั้น ๆ เพื่อรอสัญญาณ การรับข้อมูลจากอุปกรณ์ภายนอก รับข้อมูลนี้ไป สัญญาณที่ใช้ในกรณีเอ้าท์พุทข้อมูลที่ตัว 8255 มีดังนี้

- OBF (OUTPUT BUFFER FULL) เป็นเอ้าท์พุทจะแอดตีฟที่โลจิก 0 เป็นตัวบอกว่าขณะนี้ที่ตัว 8255 มีข้อมูลจาก CPU อยู่ยัง ไม่ถูกอ่านจากอุปกรณ์ภายนอก

- ACk ACKNOWLEDGE) เป็นอินพุทแอดตีฟที่โลจิก0 เป็นสัญญาณจากอุปกรณ์ภายนอกที่ส่งมาเพื่อรับเอาข้อมูลจาก 8255 ไป

- INTR (INTERRUPT REQUES) เป็นสัญญาณเอ้าท์พุทจากตัว 8255 แอดตีฟที่โลจิก 0 ปกติเราจะใช้ไปทริกให้กับ CPU ที่ขา INT เพื่อบอกให้ CPU ทราบว่าข้อมูลได้อ่านจาก 8255 ไปแล้วซึ่งเราสามารถ เซท/รีเซท ว่าต้องการให้เกิดสัญญาณที่ขา INTR นี้หรือไม่ก็ได้โดยซอฟร์แวร์

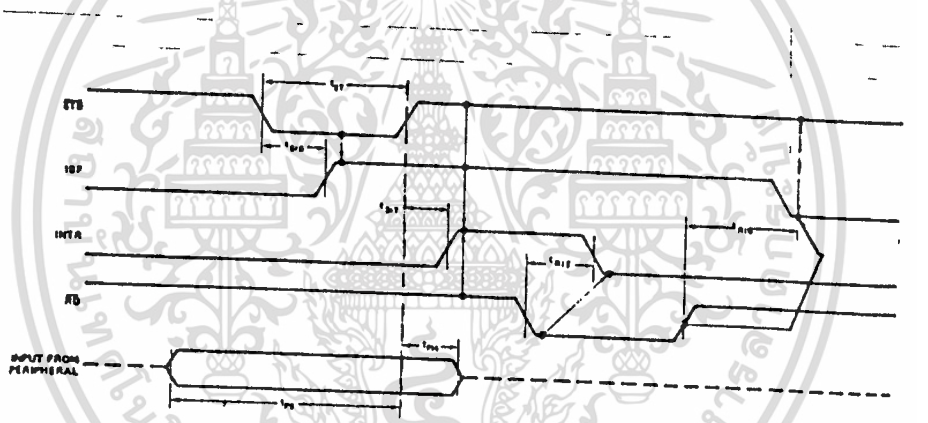
อธิบายได้ดังนี้

เมื่อ CPU สัญญาณการเขียนข้อมูลเข้ามาเก็บไปที่ตัว 8255 (WR) จะเป็นผลทำให้ OBF แสดงสภาวะว่า มีข้อมูลเข้ามาแลทช์ ไว้ที่พอร์ท A (อาจเป็นพอร์ทอื่นตามที่เราโปรแกรมไว้) โดยจะให้โลจิกออกเป็น 0 และสัญญาณ INTR ก็จะเป็น 0 ด้วย (หากเราทำการเซทไว้ก่อน) ทีนี้สัญญาณก็จะอยู่อย่างนี้ต่อไปเรื่อย ๆ หากไม่มีสัญญาณการ อ่านข้อมูลจากอุปกรณ์ภายนอกมา อ่านสัญญาณนั้น คือ ACK เมื่ออุปกรณ์ภายนอกต้ องการ อ่านเอาข้อ

มูลจากตัว 8255 ก็จะทำการตรวจดูสัญญาณที่ขา OBF ของ 8255 ก่อนว่าเป็น 0 หรือไม่ เพราะหากได้ 0 เป็นการแสดงว่าข้อมูลมีอยู่ในพอร์ท พร้อมที่จะทำการอ่านได้ จึงส่งสัญญาณแอคตีฟโลจิกศูนย์มาที่ขา ACK ดังแสดงในรูปที่ 26 เป็นการอ่านเอาข้อมูลจาก 8255 ออกไปนั้นเองจากนั้นสัญญาณ OBF , INTR ก็จะกลับคืนเป็น โลจิก 1 ดังเดิมซึ่งจะทำให้ CPU ทราบได้ว่า มีการอ่านเอาข้อมูล ที่พอร์ทไปเรียบร้อยแล้ว CPU สามารถที่จะทำการส่งข้อมูลเข้ามาใหม่ได้

2.ลำดับสัญญาณในการอินพุท โหมด 1

การอินพุทพอร์ทหมายถึงการที่อุปกรณ์ภายนอกส่งข้อมูลเข้ามาเก็บไว้ที่ตัว 8255 แต่ต้องตรวจดูก่อนเช่นกันว่า 8255 ว่างหรือไม่ เพื่อให้ตัว CPU มาอ่านเอาข้อมูลนั้นไปสู่ตัว CPU ต่อไป รูปลำดับสัญญาณและความหมายที่ต้องใช้ในกรณีอินพุทมีดังนี้



รูปที่ 2.5.9 แสดงลำดับสัญญาณที่เกิดขึ้นในกรณีที่การทำงานโหมด 1 อินพุท

- IBF (INPUT BUFFER FULL) เป็นสัญญาณเอ้าท์พุท ที่แอคตีฟโลจิก 1 จะเป็นตัวแสดงว่าขณะนี้ข้อมูลในตัว 8255 เต็มอยู่หรือไม่ หมายความว่าข้อมูลถูกอ่านด้วย CPU ไปหรือยัง หากยัง ที่ขาสัญญาณนี้ก็จะแสดงโลจิกเป็น 1 อยู่ แต่หากมีการอ่านเอาข้อมูลไปแล้วก็จะแสดงโลจิกเป็น 0 เพื่อให้อุปกรณ์ภายนอกสามารถส่งข้อมูลตัวต่อไปเข้ามาได้ทันที

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- STB (STROBE INPUT) เป็นสัญญาณ อินพุท ที่แอดตีฟที่โลจิก 0 จะเป็นขาที่รับสัญญาณจากอุปกรณ์ภายนอก เพื่อแลทช์ข้อมูลที่ส่งมาจากภายนอกเข้าสู่ตัว 8255

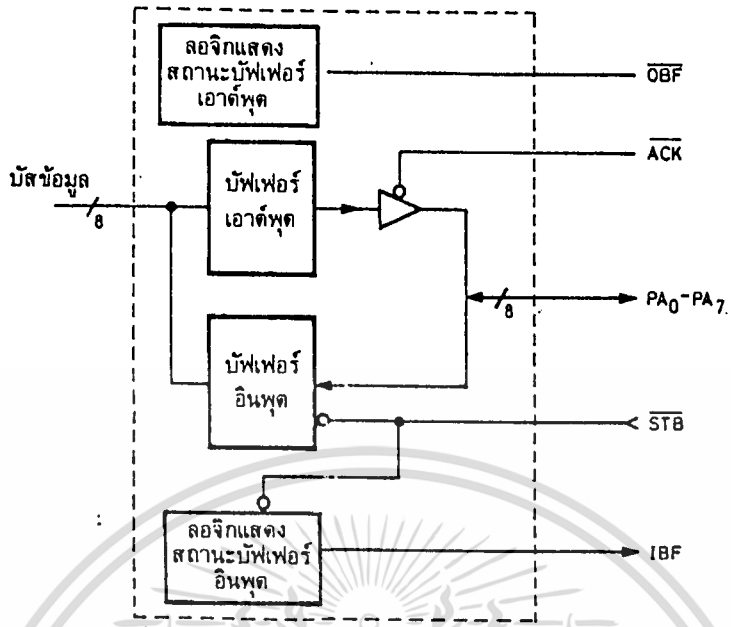
- INTR (INTERRUPT REQUEST) เป็นสัญญาณเอ้าท์พุทแอดตีฟที่โลจิก 1 มักใช้เป็นสัญญาณทริกอินเตอร์รัฟท์ให้กับ CPU เพื่อบอกให้ทราบว่าอุปกรณ์ภายนอกส่งข้อมูลมาแล้ว

อธิบายได้ดังนี้

เมื่ออุปกรณ์ภายนอกต้องการส่งข้อมูลเข้าสู่ตัว 8255 ก็ต้องตรวจดูก่อนว่าที่พอร์ทนั้นมีข้อมูลตกค้างอยู่ หรือว่าง หรือไม่ โดยการตรวจที่ขาสัญญาณ IBF ว่าเป็นโลจิก 0 หรือไม่เพราะหากเป็นโลจิก 0 จะหมายถึง ว่าง แต่หากเป็นโลจิก 1 จะหมายถึงว่าข้อมูลยังคงมีอยู่ เมื่อตรวจดู และ ทราบว่าเป็น 0 ก็สามารถส่งข้อมูลไปได้โดย ส่งสัญญาณพัลซ์โลจิก 0 ไปที่ขา STB เพื่อบอกให้ 8255 ได้ทำการแลทช์ข้อมูลที่ส่งให้นั้น และเพื่อแลทช์ข้อมูลไว้แล้วก็เป็นผลทำให้ IBF มีโลจิกไปเป็น 1 เพื่อแสดงถึงว่าได้แลทช์ข้อมูลเข้าสู่ พอร์ทแล้วพร้อมกันนั้นก็จะให้สัญญาณ INTR เป็นโลจิก 1 เพื่อแสดงให้ตัว CPU ทราบว่าได้แลทช์ข้อมูลจากอุปกรณ์ภายนอกได้แล้ว หากเราต่อเป็นสัญญาณอินเตอร์รัฟท์สู่ตัว CPU และสัญญาณก็จะค้างลักษณะนี้ต่อไป จนกว่าจะมีสัญญาณการอ่านข้อมูลจาก CPU มาอ่าน เอาข้อมูลจากพอร์ท 8255 ไปคือสัญญาณ RD และ เมื่อ CPU ส่งสัญญาณ RD มาอ่านแล้วก็จะเป็นผลให้สัญญาณ IBF กลับเป็นโลจิก 1 และ INTR กลับเป็นโลจิก 0 ดังเดิมซึ่งหากอุปกรณ์ภายนอกตรวจดูที่สัญญาณ IBF ก็จะทราบว่า มันสามารถส่งข้อมูลตัวต่อ ไปมาที่ 8255 ได้แล้ว ขอให้สังเกตุสัญญาณในรูปที่ 2.5.9 ประกอบด้วย

2.5.2.5 การทำงานในโหมด 2

การทำงานในโหมดที่ 2 นี้ จะสามารถได้เฉพาะพอร์ท A เท่านั้นเนื่องจาก ว่าการทำงานในโหมดนี้ คือการที่กำหนดให้พอร์ท A เป็นได้ทั้งอินพุท และเอ้าท์พุทได้ในพอร์ทเดียวซึ่งทำให้ต้อง ให้สายสัญญาณควบคุมมากขึ้นเป็น 5เส้น ซึ่งก็ใช้พอร์ท C เป็นขาสัญญาณควบคุมนี้ทำให้ไม่เพียงพอแก่พอร์ท B ที่จะให้เป็นโหมด 2 ฉะนั้นพอร์ท B จึงสามารถทำงานได้เฉพาะโหมด 0 ,1 เท่านั้น



รูปที่ 2.5.10 แสดงถึงบล๊อกไดอะแกรมการทำงานในโหมด 2 ของพอร์ท A

การทำงานในโหมดนี้ คือ การใช้พอร์ท A เป็นทั้งอินพุทแลทช์ข้อมูลและเอ้าท์พุทแลทช์ข้อมูลโดย เอ้าท์พุทแลทช์ก็จะหมายถึงการที่พอร์ท A รับข้อมูลจาก CPU มาทำการแลทช์ไว้เพื่อรอการอ่านข้อมูลนี้ไปด้วยอุปกรณ์ภายนอก ส่วนกรณีของอินพุทแลทช์ก็หมายถึงการเก็บข้อมูลที่อุปกรณ์ภายนอก ส่งมาแลทช์ไว้เพื่อรอให้ CPU ทำการอ่านข้อมูลนี้ไปนั่นเอง

การทำงาน โดยทั่วไปก็เหมือนกันกับการทำงานในโหมด 1 ที่ได้กล่าวมาเพียงแต่เป็นการรวมเอาพอร์ทการรับส่งไว้เป็นช่องเดียวนั่น คือเมื่อกระทำการเอ้าท์พุทก็จะมีสัญญาณOBF ,ACK ,INTR ใช้ติดต่อควบคุมและเพื่อกระทำการอินพุทก็จะมีสัญญาณ IBF ,STB ,INTR ใช้ในการติดต่อควบคุมการทำงาน โดยจะขอกล่าวเป็นลำดับดังนี้

การส่งข้อมูลจาก CPU ไปสู่ 8255 เพื่อออกสู่ภายนอกนั้นขั้นแรกก็ต้องตรวจดูก่อนว่าพอร์ท A ว่าว่างหรือไม่โดยการอ่านค่า บิท PC7 (OBF) มาดูว่าเป็น 1(ว่าง)ก็สามารถส่งข้อมูลออกไปแลทช์ไว้ที่ 8255 ได้ และเมื่ออุปกรณ์ภายนอกต้องการรับข้อมูลไปก็จะตรวจที่ PC7 (OBF) นี้ว่าได้โลจิกเป็น 0 หรือไม่ หากเป็น 0 หมายถึงมีข้อมูลอยู่ ก็จะทำการอ่านเอาไปโดยการส่งสัญญาณ ACK มาที่บิท PC6(ACK)เพื่ออ่านเอาข้อมูลไปเป็นผลทำให้สถานะของ PC7 (OBF) กลับคืนเป็น 1 อีกครั้งเพื่อแสดงตัวว่า 8255 ว่างที่จะรับข้อมูลตัวต่อไปจาก CPU แล้วการรับข้อมูลจาก อุปกรณ์ภายนอกนั้น ก่อนที่จะทำการส่ข้อมูลจากอุปกรณ์ภายนอกที่พอร์ท A นั้นอุปกรณ์ภายนอกจะทำการตรวจดูก่อนว่าพอร์ท A ว่างหรือไม่ โดยการ

ตรวจที่บิท PC5 (IBF) ว่าเป็นโลจิก 0 หรือไม่ หากเป็น 0 แสดงว่าก็จะส่งสัญญาณ STB มาที่ขา PC4 เป็นการบอกให้ 8255 ได้ทำการแลทช์ข้อมูลของอุปกรณ์ภายนอกที่ส่งมาไว้ที่พอร์ท A เมื่อแลทช์แล้ว PC4 (IBF) จะเปลี่ยนโลจิกไปเป็น 1 ทันทีเพื่อบอกให้อุปกรณ์ภายนอกทราบว่าข้อมูลถูกแลทช์สู่พอร์ท A เรียบร้อยแล้ว อย่าเพิ่งส่งข้อมูลมาอีก ถึงตอนนี้เมื่อ CPU ตรวจดูที่ PC4 (IBF) ดูโดยการอ่าน พอร์ท C ก็จะทราบว่ามี ข้อมูลส่งมาแล้ว จึงทำการอ่านข้อมูลไปได้ ส่ง RD มาอ่าน เป็นผลทำให้PC4 (IBF) กลับโลจิกเป็น 0 อีกครั้งเพื่อ ให้อุปกรณ์ภายนอกส่งข้อมูล ตัวต่อไปมาที่ 8255 อีกขาควบคุมของพอร์ท C ที่ใช้ในการทำสัญญาณ ควบคุมต่าง ๆ นั้น แสดงดังรูปที่ 2.5.9 สังเกตุเห็นได้ว่า PC0-PC2 นั้นเราสามารถกำหนดให้เป็น อินพุท หรือ เอ้าท์พุทได้อีกต่างหาก ในกรณีที่ พอร์ท B ถูกโปรแกรมในโหมด 0 (เพราะไม่ต้องมีสัญญาณควบคุม) แต่หากพอร์ท B ถูกโปรแกรมในโหมด 1 จะทำให้ต้องใช้ PC0-PC2 นี้เป็นสัญญาณควบคุมของพอร์ท B

พอร์ต C	ความหมาย
PC0	I/O
PC1	I/O
PC2	I/O
PC3	$\overline{INTR}$
PC4	$\overline{STB}$
PC5	$\overline{IBF}$
PC6	$\overline{ACK}$
PC7	$\overline{OBF}$

รูปที่ 2.5.11 แสดงถึงพอร์ท C ที่ถูกใช้เป็นสัญญาณควบคุมในโหมด 2

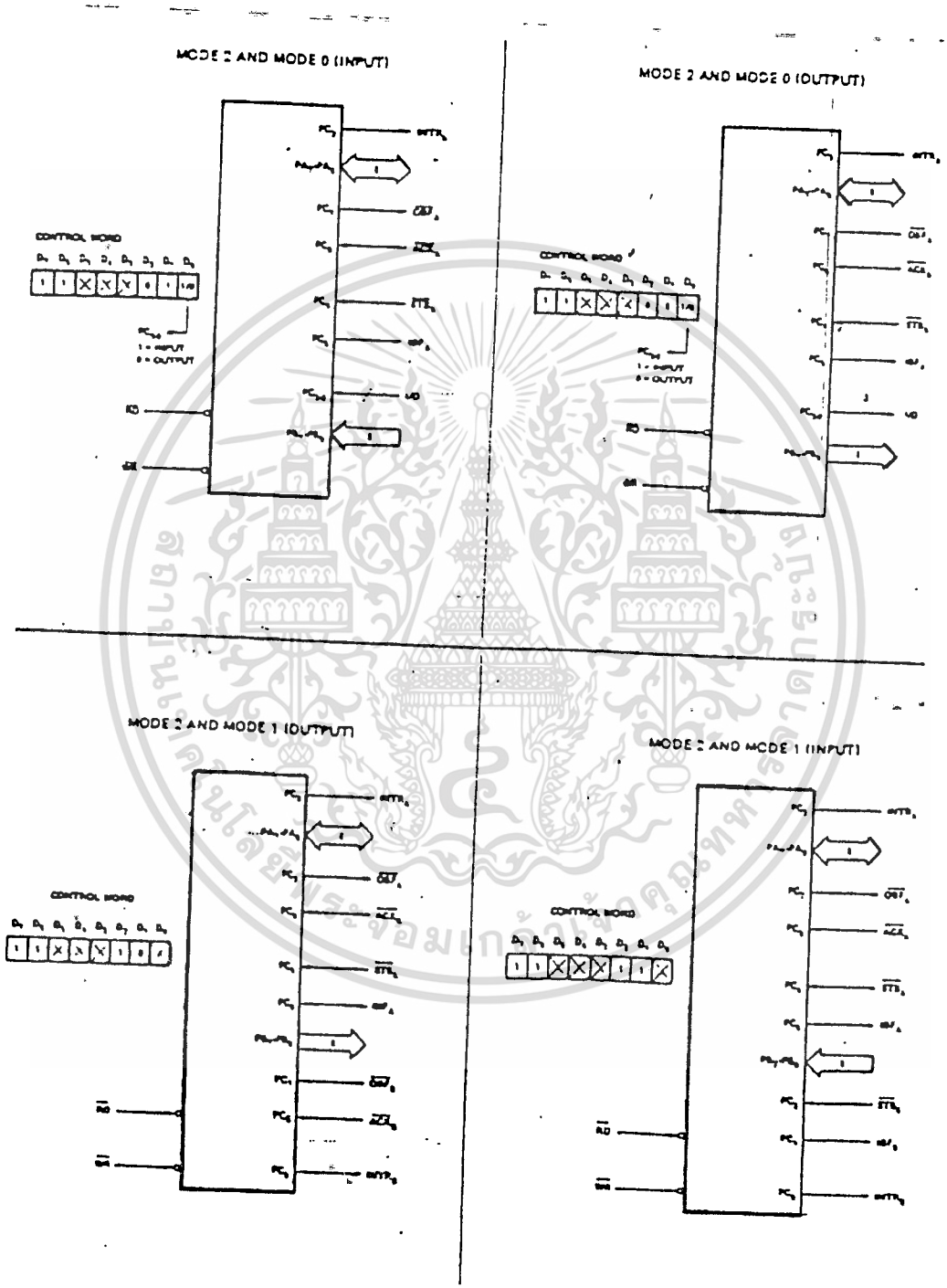
ส่วนสัญญาณ INTR นั้นเราก็สามารถกำหนดให้มีการเกิดอินเตอร์รัพ์หรือไม่ก็ได้ โดยการ เซท/รีเซท บิท ดังนี้

กรณี อินพุท สามารรถที่จะทำการ เซท/รีเซท ได้ที่ บิท PC4(STB)

กรณี เอ้าท์พุท สามารถที่จะทำการ เซท/รีเซท ได้ที่ บิท PC6 (ACK)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เราจะสามารถโปรแกรมให้พอร์ท A ในโหมด 2 ทำงานร่วมกับพอร์ท B ซึ่งเป็นโหมด 0 , 1 ก็ได้ดังนี้

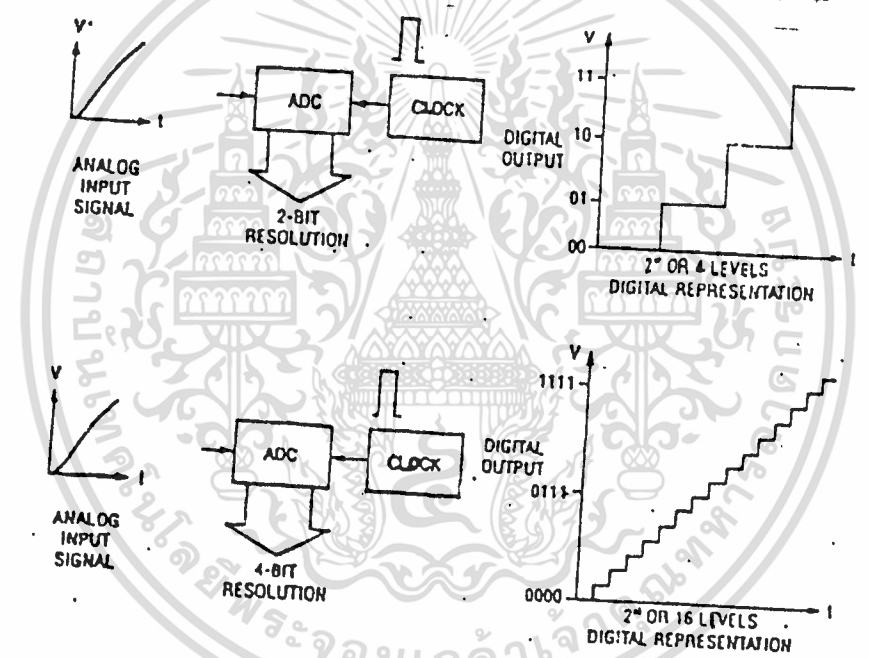


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 2.5.12 แสดงการโปรแกรมโหมด 2 ร่วมกับโหมด 0,1

2.6 A/D คอนเวอร์เตอร์ (Analog to Digital Convertor

A/D คอนเวอร์เตอร์ หรือ ADC ใช้สำหรับการแปลงสัญญาอินพุทที่เป็นอะนาล็อก ให้เป็นจำนวนจำกัดของ ดิจิตอลบิทผลลัพท์ที่ได้จะอยู่ในรูปของ "WORD" ทางดิจิตอลซึ่งจะกลายเป็นรหัสเลขฐานสองที่แทนระดับ แต่ละระดับของสัญญาอะนาล็อกในขณะที่ DAC กำลังทำงานแปลงสัญญาณอยู่ ความละเอียดของ ADC จะคล้ายกับความละเอียดของ DAC อย่างมาก กล่าวคือ จำนวนบิท ทางเอ้าท์พุทมีหลาย ๆ บิท ความละเอียดของ ADC ตัวนั้นก็จะมีมากขึ้นเช่น ADC ขนาด 12 บิท จะมีความละเอียดเท่ากับ 12 เป็นต้น ดังแสดงในรูปที่ 2.6.1



รูปที่ 2.6.1 ความละเอียดของ ADC จะแปรผันตรงกับจำนวนบิททางด้านเอ้าท์พุท

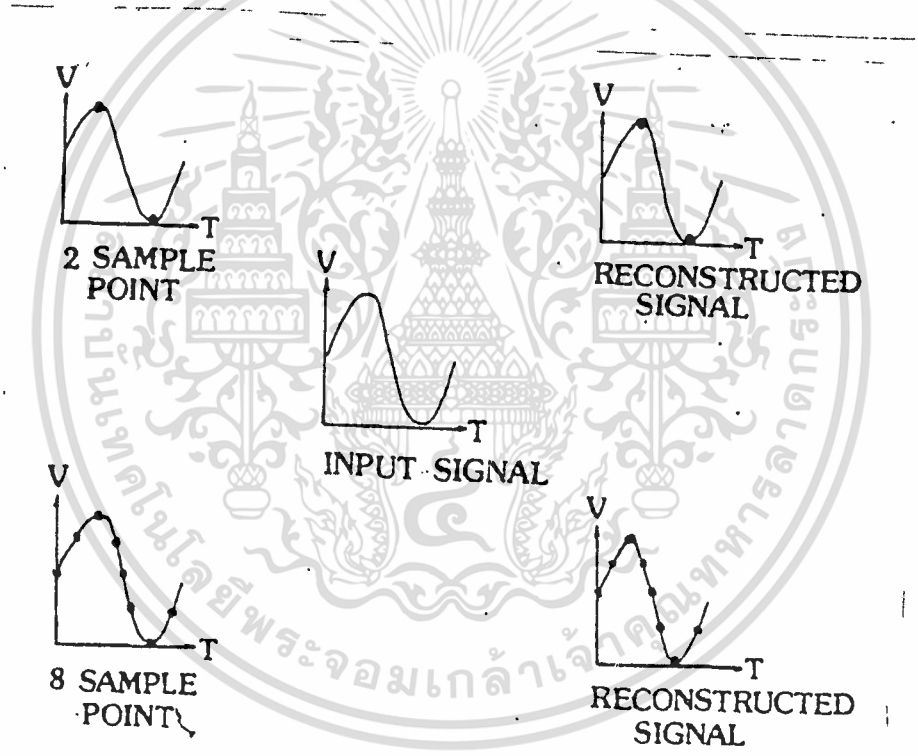
ค่าเวลาในการแปรผัน (CONVERSION TIME) เป็นเกณฑ์ที่สำคัญอีกตัวหนึ่งของ ADC ตามที่คุณได้เห็นมาว่า การแปลงสัญญาอะนาล็อกให้กลายเป็นสัญญาณทางดิจิตอลไม่ได้เกิดขึ้นโดยทันทีทันใด แต่ต้องมีการผ่านขบวนการต่าง ๆ ด้วยเหตุที่ผลลัพท์ต้องเวลาค่าเวลาขณะหนึ่งที่จะทำการสุ่ม(Sample) สัญญาอินพุทและสัญญาดิจิตอลที่เป็นรหัสไบนารีออกมาที่เอ้าท์พุท ดังนั้น ค่าเวลาการแปรผัน คือ ช่วงเวลาที่ต้องการกระทำขบวนการให้เสร็จสิ้น ซึ่งมีค่าอยู่ประมาณ μs หรือ DAC ความเร็วสูงและเป็น ms สำหรับ DAC แบบธรรมดาเนื่องจากการเปลี่ยน A/D นั้นต้องการกระบวนการ ชิงโครไนส์ ที่แน่นอน และ แม่นยำ แหล่งกำเนิดสัญญาณนาฬิกาจึงจำเป็นต้องมีในวงจร

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเอกสารนี้ทุกครั้งที่มีการนำไปใช้

2.6.1 ทฤษฎีการสุ่มตัวอย่าง (Sampling.theory)

เนื่องจาก ADC ต้องการค่าเวลาขณะหนึ่ง ใช้ในขบวนการ เปลี่ยนแปลง สัญญาณอะนาล็อกเป็น สัญญาณดิจิตอล ช่วงเวลาช่วงหนึ่งจะใช้สำหรับการสุ่มตัวอย่าง ของสัญญาณตัวอย่างเช่น ADC สามารถเปลี่ยนสัญญาณเสร็จสมบูรณ์ได้ผ่านในเวลา 1ms ดังนั้นมันจึงเปลี่ยนสัญญาณได้ 1000 ครั้งใน 1 วินาที (ในทางทฤษฎี) อัตราการเปลี่ยนแปลงสัญญาณสูงสุดมีค่าเท่ากับส่วนกับของเวลาการเปลี่ยน (COMVERSION RATS=1/CONVERSION TIME)

ตัวคอนเวอร์ทเตอร์ จะสุ่มตัวอย่างของสัญญาณด้วยอัตราต่ำสุดเป็น 2 เท่าของความถี่สูงสุดของสัญญาณอินพุทที่เข้ามา อัตราการสุ่มนี้เรียกว่า" Nyquist rate "พิจารณาสัญญาณอะนาล็อก ที่เป็นรูปคลื่นซายน์ 10 Hz จ่ายในกับตัว ADC ตามรูปที่ 2.6.2



รูปที่ 2.6.2 การสุ่มหลาย ๆ ช่วงจากสัญญาณอินพุทอะนาล็อกจะมีลักษณะใกล้เคียงกับสัญญาณดังเดิมเมื่อมีการสร้างสัญญาณขึ้นมาใหม่ DAC

อัตราต่ำสุดของการสุ่มตัวอย่างเป็น 2f หรือ 20 Hz ซึ่งจะให้ข้อมูลดิจิตอลขนาด 2 บิท ออกมาในแต่ละไซเคิ้ล เมื่อข้อมูลดิจิตอลถูกนำมาสร้างเป็นสัญญาณอะนาล็อกขึ้นมาใหม่โดย DAC สัญญาณอะนาล็อกตัวใหม่มีลักษณะคล้ายกับสัญญาณดังเดิม (ตัวฟิลเตอร์บนตัว DAC

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

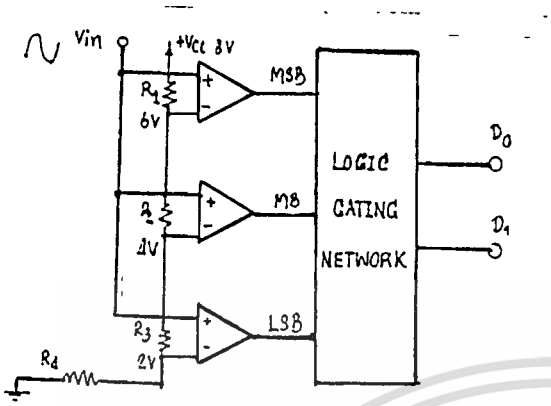
จะทำให้รูปร่างของสัญญาณเอ้าท์พุทเรียบขึ้น) ค่าความถี่ 10 Hz เป็นความถี่สูงสุดที่เข้ามายังตัว ADC ค่าเวลาที่ใช้ในการเปลี่ยนสัญญาณสูงสุดเป็น 1/20 Hz หรือ 500 ms เป็นต้น การที่เราจะปรับปรุงประสิทธิภาพของ ADC ในแง่ความเหมือนจริงของสัญญาณการแปลงให้อยู่ในรูปของ ดิจิตอล เราจะต้องเพิ่มอัตราการสุ่มขึ้น ในค่าเวลาเท่าเดิม อัตราการสุ่ม จุดต่อไซเคิ้ล ต้องการอัตราสุ่มของส่วนประกอบความถี่สูงสุดอินพุทครั้งเช่น ความถี่อินพุท 10 Hz จะต้องสุ่มตัวอย่างที่ 10 Hz ดังนั้น ตัวคอน์ทเวอร์เตอร์ควรจะมีค่าการเปลี่ยนเป็น 1/0 Hz หรือ 12.5 ms ถ้าตัว ADC ไม่สามารถสุ่มตัวอย่างได้เร็วพอต่อสัญญาณอินพุทที่เปลี่ยนแปลงไปข่าวสาร ที่บรรจุใน สัญญาณอะนาล็อก ทางอินพุท จะสูญหายไปค่า ความสัมพันธ์ระหว่างความถี่ทางอินพุท ค่าเวลาในการเปลี่ยนสัญญาณและอัตราการสุ่ม เป็น พารามิเตอร์ของ ADC ที่สำคัญตัวหนึ่งวิธีการหลาย ๆ วิธีได้ถูกพัฒนาหลาย ๆ ปีผ่านมา เพื่อที่จะ ทำการเปลี่ยนสัญญาณอะนาล็อก ให้อยู่ในรูปของ สัญญาณดิจิตอล หลาย ๆ วิธีที่ยังใช้อยู่ทุกวันนี้มี 6 วิธีด้วยกันคือ

2.6.1. Flash techniques เทคนิคแบบแฟลช์
2.6.2. Single slope techniques เทคนิคแบบสโลปเดียว
2.6.3. Double slope tachniques เทคนิคแบบสโลปคู่
2.6.4. Single counter techniques เทคนิคแบบเคาเตอร์เดียว
2.6.5. Tracking counter techniques เทคนิคแบบเทร็ดกิ้งเค้าเตอร์
2.6.6. Ssccessive sporoximation techniques เทคนิคการประเมินค่าหลายครั้ง

2.6.2 A/D แบบแฟลช์ (FLASH CONVERSTOR)

แฟลช์คอนเวอร์ทเตอร์ เป็น ADC ที่เร็วที่สุดในบรรดา ADC ที่ใช้เทคนิคแบบอื่น ๆ ลักษณะของแฟลช์คอนเวอร์ทเตอร์ จะใช้ชุดของตัวเปรียบเทียบ (comparator) ที่ก่อในลักษณะขนานกันเพื่อทำการแปลง สัญญาณอะนาล็อกทางอินพุทให้เป็นรหัสดิจิตอลดังนั้น แฟลช์คอนเวอร์ทเตอร์เป็นแบบขนาน พิจารณาในรูปที่ 2.6.3

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



Vin (V)	Binary Bit o/p		Comparator o/p		
	D1	DO	MSB	MB	LSB
0-2	0	0	0	0	0
2-4	0	1	0	0	1
4-6	1	0	0	1	1
6-8	1	1	1	1	1

รูปที่ 2.6.3 แฟลช์ A/D คอนเวอร์ทเตอร์เป็นคอนเวอร์ทเตอร์ที่ความเร็วในการเปลี่ยนแปลงสัญญาณสูงและมีลักษณะวงจรง่ายความละเอียดต่ำ

จากรูปที่ 2.6.3 ตัวต้านทานที่ต่ออนุกรมกัน จะอยู่ในรูปวงจรแบ่งแรงดันที่ตกคร่อมตัวเปรียบเทียบแต่ละตัว แรงดันอินพุทสูงสุดจะขึ้นอยู่กับค่า VCC สัญญาณเอ้าท์พุทจากตัวเปรียบเทียบแต่ละตัวจะเป็น 1 หรือ 0 ซึ่งเป็นระดับสัญญาณโลจิกของวงจรดิจิตอล เมื่อไม่มีแรงดันอินพุทเพิ่มขึ้น เอ้าท์พุทของตัวเปรียบเทียบแต่ละตัวจะเป็นโลจิก " 0 " ต่อมา แรงดันอินพุทเพิ่มขึ้น เอ้าท์พุท ของตัวเปรียบเทียบแต่ละตัวจะเป็น โลจิก " 1 " ได้ ตามลำดับขึ้นไปเมื่อแรงดันอินพุทมีมากกว่าแรงดันอ้างอิง แต่ละค่าที่ถูกเซท โดยวงจรแบ่งแรงดันเน็ทเวอร์ค การสร้างรหัสที่เอ้าท์พุทของคอนเวอร์ทเตอร์

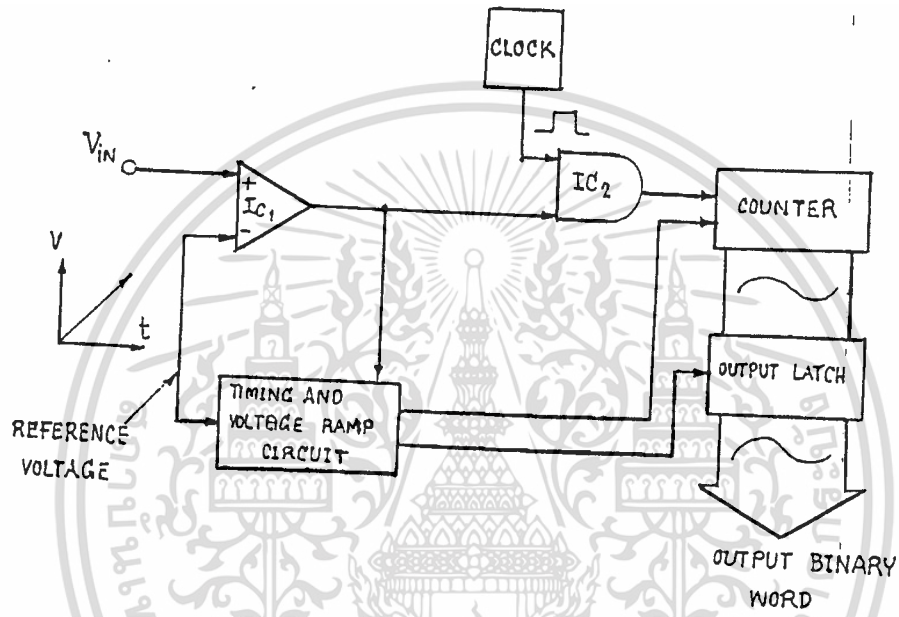
ถ้าเราสังเกตุวงจรในรูปที่ 2.6.3 ให้ดีจะพบว่า วงจรที่ใช้ตัวเปรียบเทียบ 2-1เป็นตัวแสดงความละเอียดของ คอนเวอร์ทเตอร์ จากตัวอย่างคอนเวอร์ทเตอร์ขนาด ๒ บิทของเราต้องการ 2 - 1 เท่ากับ 3 และคอนเวอร์ทเตอร์ขนาด บิท ต้องใช้ตัวเปรียบเทียบถึง 2-1 ตัว หรือ 255 ตัว

แฟลช์คอนเวอร์ทเตอร์ ได้มีเพียง ข้อเสีย ของมันเพียงอย่างเดียว แต่มันยังมีของดีที ADC แบบอื่น ๆ ไม่มี หรือสู้ไม่ได้คือความเร็วช่วงเวลาในการเปลี่ยนจึงมีความเท่ากับเวลาหน่วง ในตัวเปรียบเทียบแต่ละตัว และ วงจรเกทในวงจร เท่านั้น ซึ่งในเวลา ไม่เพียง ไมโครเซค

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์ ไม่อนุญาตให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.6.3 ADC แบบสโลปเดียว (single-Slope ADC)

การแปลงสัญญาณ อะนาล็อกให้เป็นสัญญาณ ดิจิตอลที่มีประสิทธิสูงสุดวิธีหนึ่งคือ วิธีแบบ A/D สโลปเดียวหรือเรียกว่า A/D แรมป์เดียว (Single-ramp A/D ตัวแสดงไว้ในรูปที่ 2.6.4



รูปที่ 2.6.4 ตัวผลิตสัญญาณแรมป์ ตัวเปรียบเทียบและตัวนับ เป็นส่วนประกอบบทที่สำคัญของ A/D คอนเวอร์ทเตอร์แบบสโลป์เดียว

การทำงานของวงจร A/D สโลปเดียวเริ่มต้นการรีเซทวงจรนับและแรงดันแรมป์อยู่ที่ 0 เอ้าท์พุทของตัวเปรียบเทียบที่จุดเริ่มต้นเป็น "0" ดังนั้นจึงไม่มีสัญญาณนาฬิกาจ่ายให้กับวงจรนับ เมื่อแรงดันอินพุท ถูกจ่ายให้กับตัว คอนเวอร์ทเตอร์ ขาอินพุท นอนอินเวอติ้ง (+) จะมีค่าแรงดันเกินกว่าแรงดันขาอินพุทอินเวอติ้ง (-) ดังนั้น เอ้าท์พุทของตัวเปรียบเทียบจึงเป็นโลจิก "High" สัญญาณโลจิก " High " นี้ไปเอนาเบิ้ลให้ and เกททำงานให้ยอมรับพัล์ส์ของสัญญาณนาฬิกา ผ่านตัวมันเข้าไปยังวงจรนับเลขฐานสองให้ทำงานขณะเดียวกัน วงจรไทม์มิ่ง จะขับให้แรงแรมป์ เพิ่มขึ้น ซึ่งทำให้แรงดันอินพุทที่ขาอินเวอติ้งของตัวเปรียบเทียบเพิ่มขึ้นอย่างรวดเร็ว เมื่อแรงดันแรมป์อ้างอิงเริ่มมากกว่าแรงดันอินพุทเอ้าท์

พุทของตัวเปรียบเทียบ จะตกลงเป็น โลจิก " low " อีกครั้ง วงจรนับจุเกิดการ ค้าง (latch) ค่าที่นับขณะหนึ่งต่อมาทำการ รีเซท ตัวนับสำหรับ วัฏจักรการแปลงสัญญาณช่วงต่อไปเมื่อแรงดัน แรมป์ อ้างอิงมีค่าเท่ากับ แรงดันอินพุท ที่จ่ายเข้ามาวงจรนับจะถูกกระตุ้นให้นับเลขฐานสอง ขณะเดียวกันค่าที่นับได้ จึงเป็นสัญญาณ ดิจิตอล ของสัญญาณอะนาล็อกทางด้านอินพุทที่เข้ามาในขณะนั้น จะสังเกตุได้ว่าความเร็วของสัญญาณนาฬิกาและอัตราการเพิ่มขึ้นในลักษณะเป็นแรงดัน แรมป์ จะต้องมีความสัมพันธ์กันอย่าง ถูกต้องเพื่อให้วงจรนับทำงานตามหน้าที่ ได้อย่างถูกต้อง แน่นอน ค่าเวลาที่ต้องการ ทำการเปลี่ยนขึ้น อยู่ระดับ สัญญาณอะนาล็อกทางอินพุท เพราะว่าวงจรนับ และ แรงดันแรมป์ อ้างอิงทั้งคู่เริ่มต้นจากศูนย์ที่ทุก ๆ วัฏจักรการแปรผัน มันจึงใช้เวลาค่อนข้างนาน ที่จะทำให้แรงดันอ้างอิงเท่ากับแรงดันอินพุท ในทางตรงกันข้าม ถ้าแรงดันอินพุทมีค่าน้อย ช่วงเวลา ที่แรงดันแรมป์ที่อ้างอิงเพิ่มขึ้นจะเท่ากับแรงดันอินพุท จึงใช้เวลาน้อยกว่ากรณีแรงดันอินพุท มีค่ามาก ๆ แรงดันแรมป์อ้างอิง สามารถเปลี่ยนแปลงมากขึ้น เท่ากับแรงดันอินพุทได้ เร็วกว่า 1 โวลท์ ต่อ 1/1000 วินาที เช่นถ้าแรงดันอินพุทเป็น 2 โวลท์ ถูกจ่ายให้กับวงจรในรูป วงจรจะใช้ เวลา 2 x 1 Volt/ms ซึ่งเท่ากับ 2 ms สำหรับแรงดันแรมป์ ที่จะเพิ่มขึ้นจะมีแรงดันเท่ากับ แรงดันอินพุท การนับเลขฐานสอง จะกระทำหลังจาก 2 ms ไปแล้วความเร็วในช่วงนี้ขึ้นอยู่กับความเร็วสัญญาณนาฬิกามีค่าสูงจะทำให้จังหวะในการนับเร็วขึ้นด้วย

เนื่องจากการทำงาน ของสัญญาณนาฬิกา ขึ้นอยู่กับแรงดันแรมป์ จึงเป็นลักษณะพิเศษของ A/D สโลปเดียว ที่มีสัญญาณเอ้าท์พุทออกมาเป็นเลขฐานสองโดยตรง ไอซีเครื่องมือวัดบางตัวที่ใช้เทคนิค แบบสโลปเดียว นี้จะแปลงรหัส BCD ไปขับภาคแสดงผล 7 เซ็กเม้นท์ได้โดยตรง ซึ่งทำให้สะดวก และ ข้อได้เปรียบกว่าเทคนิค A/D แบบอื่นอย่างมาก

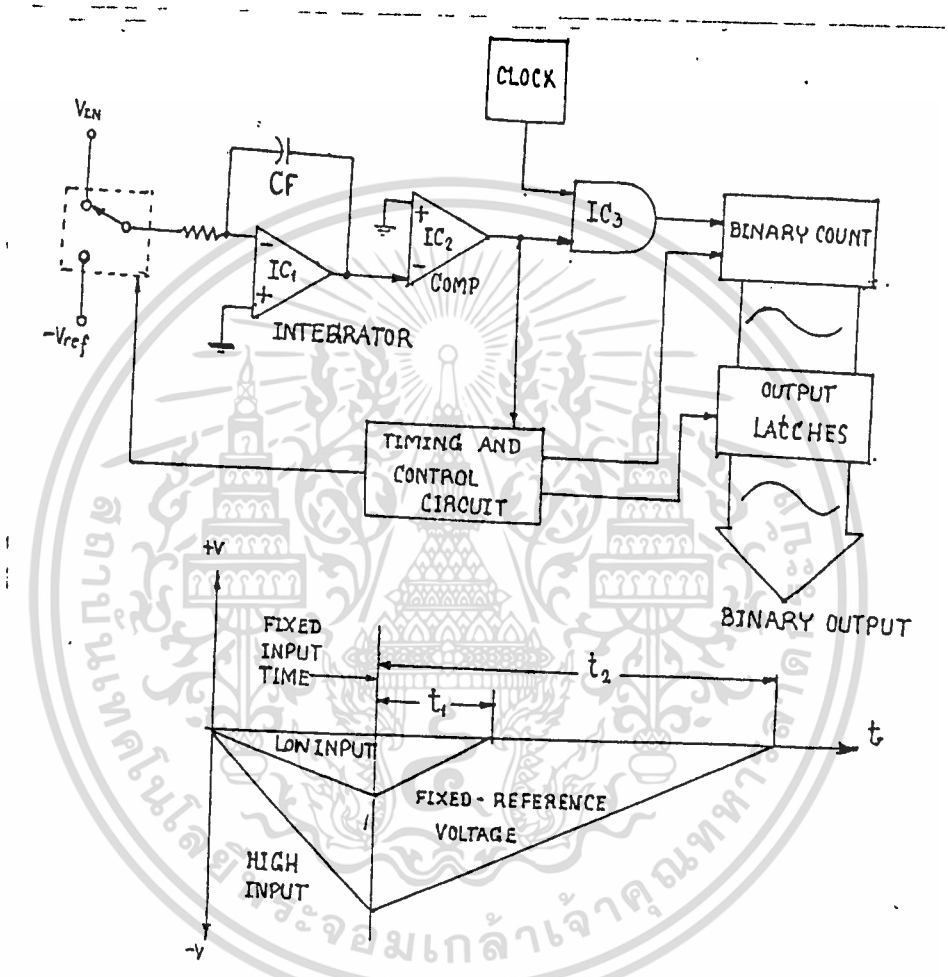
ข้อเสียของ A/D สโลปเดี่ยวคือ การทำงานที่ไม่ค่อยมีเสถียรภาพเมื่อใช้งาน A/D เป็นเวลานาน ๆ โดยปราศจากการประสานจังหวะ (Synchroninzation)ระหว่างวงจรผลิตสัญญาณนาฬิกา และ วงจรสร้างสัญญาณแรมป์ทุก ๆการเลื่อนของความเร็วสัญญาณนาฬิกาหรือ แรงดันแรมป์ เป็นเหตุทำให้เกิดการ ผิดพลาดที่รหัสทางเอ้าท์พุท จึงเป็นสาเหตุที่สำคัญที่ทำให้ A/D สโลปเดี่ยวไม่น่าไปใช้งานที่ต้องการความถูกต้องสูง

2.6.4 A/D แบบสโลปคู่ (Double-Slope ADC)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เทคนิคการเปลี่ยนสัญญาณ อะนาล็อก ไปเป็นสัญญาณ ดิจิตอลแบบสโลปคู่ เป็นเทคนิคที่

ให้ข้อดีทางเสถียรภาพของการแปลง สัญญาณ เมื่อสัญญาณอินพุทมีการเปลี่ยนแปลง อย่างรวดเร็ววงจรผลิต สัญญาณแรมป์อ้างอิง ได้ปรับปรุงขึ้นโดยตัดเอา ผลกระทบ ของการเลื่อนไหล เมื่อใช้วงจรไปนาน ๆ สัญญาณอินพุท ของตัวคอนเวอร์ทเตอร์ แบบสโลปคู่จะป้อนให้กับวงจรอินทิเกรดเตอร์ เมื่อสัญญาณอินพุทที่เป็นบวกถูกป้อนเข้ามายังตัว ADC



รูปที่ 2.6.5 ADC แบบสโลปคู่ เป็นคอนเวอร์ทเตอร์ที่ให้ประสิทธิภาพการทำงานสูงกว่า ADC แบบสโลปเดียว

ความชันของแรงดันแรมป์ทางด้านเอ้าท์พุทของวงจรอินทิเกทเตอร์มีทิศทางเป็นลบและจะมีค่าเป็นลบด้วย แรงดันลบ ที่ได้นี้ทำ ให้เอ้าท์พุทวงจรเปรียบเทียบเป็น " High " ด้วยเหตุนี้จึงเป็นการ กระตุ้นให้เกิดสัญญาณนาฬิกา ป้อนเข้าไปยังขั้วอินพุทของ วงจรนับ ซึ่งเป็นการเริ่มต้นนับขึ้นไปเรื่อย ๆ วงจรอินทิเกทเตอร์จะให้สัญญาณแรมป์เพียงคาบเวลาที่คงที่ขณะ

หนึ่งเท่านั้น หลังจากช่วงเวลานี้แล้ว วงจรควบคุมทำการเคลียวงจรนับ เอ้าท์พุทของวงจรเปรียบเทียบจะกลายเป็น "low" ซึ่งเป็นสัญญาณนาฬิกาที่ป้อนวงจรนับหยุดลง วงจรควบคุมจะทำการตรวจสอบซึ่งเปลี่ยน และ แลทช์ การนับที่เอ้าท์พุทไว้เราทำการเคลียวงจรอีกครั้ง

อัตราอินทิเกท ขึ้นอยู่กับขนาดของ แรงดันอินพุท เช่นเดียวค่าของ R และ C ดังนั้นแรงดันอินพุทที่ต่ำ ๆ จะสงของวงจรอินทิเกทเตอร์ให้น้อยกว่าแรงดันดินพุทที่มีค่าสูง ๆในช่วงคาบเวลาอินพุท ที่แน่นอนของวัฏจักรการแปรผัน (Conversion eyele)ดังนั้นคอนเวอร์ทเตอร์แบบสโลปคู่ มีประสิทธิภาพเหมาะสำหรับการประยุกต์ใช้งานที่มีความแม่นยำสูง

ข้อเสีย ของการแปรผันแบบสโลปคู่ คือ คาบเวลาที่ขยายออกไปที่ต้องการใช้ในการแปรผัน คอนเวอร์เตอร์แบบสโลปคู่ ต้องการคาบเวลาที่มากกว่า 100 ms ต่อการเปลี่ยนสัญญาณอินพุทที่มี แรงดันสูง ๆ ให้อยู่ในรูปสัญญาณดิจิตอล

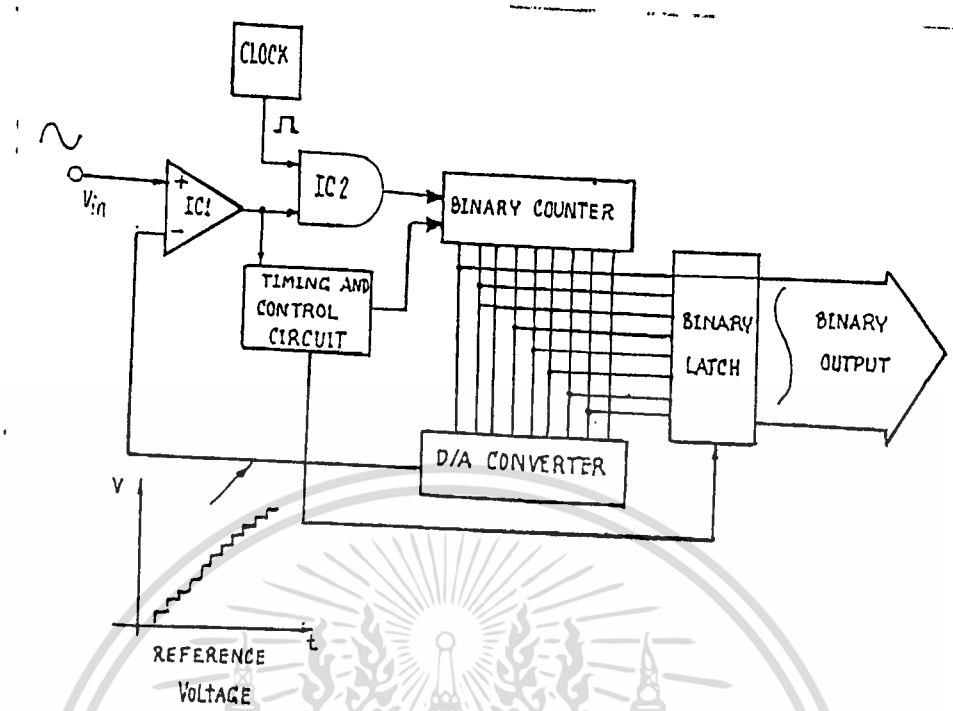
2.6.5 DAC แบบมีการป้อนกลับ (D/A feedback converters)

D/A คอนเวอร์ทเตอร์ ที่ใช้สัญญาณป้อนกลับ มาเป็น สัญญาณอ้างอิง ที่วงจร เปรียบเทียบมีสองชนิด คือ

1. วงจรนับแบบเดียว (Single counter)
2. วงจรนับแบบเทร็ดกิ้ง (Tracking counter)

วงจรของนับ ADC แบบวงจรนับเดียวได้มีการพัฒนาจนมีลักษณะคล้ายคลึงกันกับ ADC แบบสโลปเดียว ตลอดจนการทำงานของวงจรทั้งสองยังคล้ายกันอีกด้วย แต่ ADC แบบวงจรนับเดียว จะอ่านการนับสัญญาณนาฬิกาที่ได้จากวงจรนับเลขฐานสอง แล้วทำให้เป็นแรงดันป้อนกลับ ไปยังวงจร เปรียบเทียบ แทนวงจร อินทิเกทเตอร์ หรือแหล่งแรงดันแรมป์อื่น ๆ ดังรูปที่ 2.6.6

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

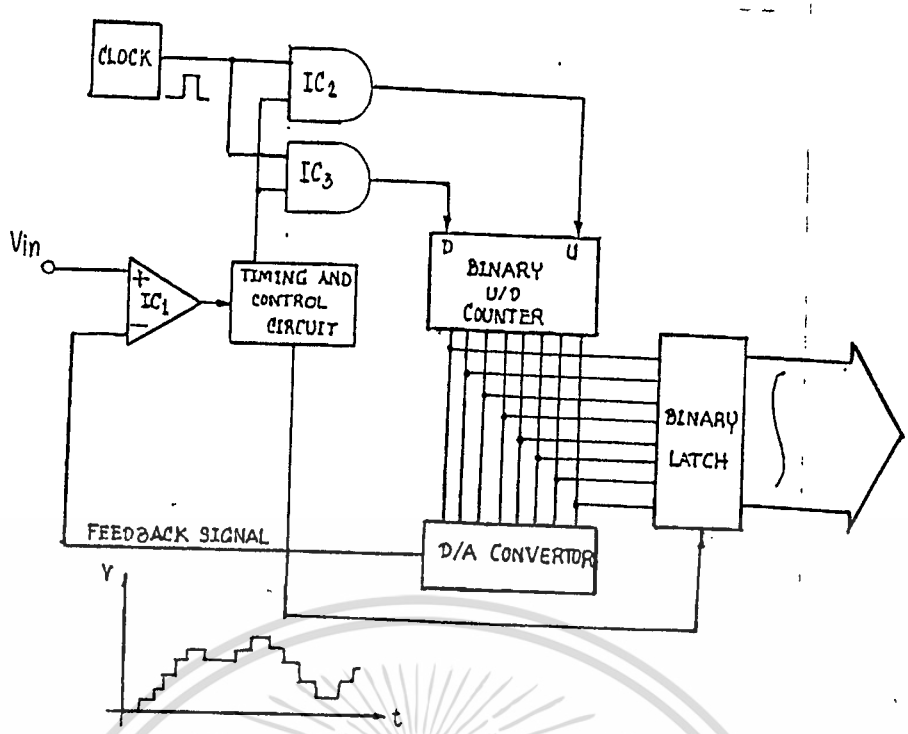


รูปที่ 2.6.6 D/A คอนเวอร์ทเตอร์ถูกใช้ในวงจร D/A คอนเวอร์ทเตอร์เพื่อที่จะเป็นตัวสร้างแรงดันแรมป์ไปควบคุมการทำงานของ A/D คอนเวอร์ทเตอร์

เมื่อสัญญาณ อะนาล็อก ทางอินพุท ถูกจ่ายเป็น วงจรเปรียบเทียบ เอ้าท์พุท ของมัน " High " ดังนั้นวงจรควบคุมจึงยอมให้สัญญาณนาฬิกาผ่านเข้าไปในวงจรนับ ๆ จะนับเลขฐานสองขึ้นไปเรื่อย ๆ วงจรควบคุมจะส่งสัญญาณไปยัง วงจรแลทช์เลขฐานสอง (Sinary latch) ให้ไปค้าง ค่าที่ได้จากวงจรนับที่เอ้าท์พุท ไว้หลังจากนั้น วงจรควบคุมจะ ทำการรีเซท วงจรนับสำหรับ วัฏจักรการแปรผันต่อไป

เทคนิคของวงจรนับแบบเทร็คกิ้ง สามารถทำการแปรผันได้เร็วกว่าแบบวงจรนับเดียว ซึ่งเป็นการแปรผัน ที่รวดเร็ว เพราะวงจรนับเทรดกิ้ง ใช้วงจรนับเลขฐานสองแบบขึ้นลงได้แทนวงจรนับขึ้นอย่างเดียวเหมือนกับตัวอย่างที่ผ่านมา ดังรูปที่ 2.6.7

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

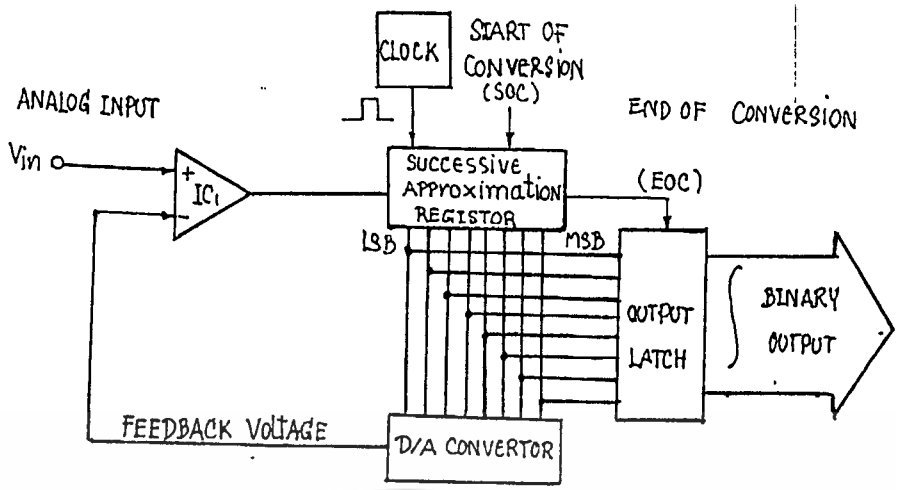


รูปที่ 2.6.7 วงจรนับเลขฐานสองแบบขึ้นลงทำให้ A/D คอนเวอร์ทเตอร์เปลี่ยนแปลงสัญญาณป้อนกลับตามสัญญาณอะนาล็อกทางอินพุทได้ซึ่งใช่ใน ADC แบบเทร็คกิ้ง

วงจรนับสามารถเพิ่มค่าขึ้นลงได้ขึ้นอยู่กับฐานะทางด้านเอ้าท์พุทของวงจรเปรียบเทียบ ซึ่งจะทำให้รหัสไบนารี่ที่ได้มีความเป็นจริงต่อสัญญาณอะนาล็อกมากขึ้น

การทำงานของ ADC เทร็คกิ้ง เริ่มต้นที่สัญญาณอะนาล็อกถูกป้อนมายังอินพุทของวงจรเปรียบเทียบการนับ บนวงจรนับเลขฐานสองขึ้นลง อาจจะเริ่มที่ค่าใด ๆ ก็ได้นั้นหมายถึงแรงดันป้อนกลับที่มาจากตัว DAC อาจจะมากกว่าหรือน้อยกว่าสัญญาณอะนาล็อกทางอินพุทก็ได้ ถ้าแรงดันป้อนกลับ มีค่ามากกว่า สัญญาณอะนาล็อก ทางอินพุท เอ้าท์พุท จะมีฐานะเป็น " low " และวงจรควบคุมจะส่งสัญญาณไปเปิดเกทให้พัลส์ของสัญญาณนาฬิกาผ่านไปยังวงจรนับ ดังนั้นจึงเป็นการลดค่าเลข ฐานสองซึ่งเป็นเอ้าท์พุทของวงจรนับและเป็นการลงแรงดันป้อนกลับที่วงจรเปรียบเทียบ เมื่อแรงดัน ป้อนกลับตกลงต่ำกว่า แรงดันอินพุท เอ้าท์พุทของวงจรแบบเปรียบเทียบจะฐานะเป็น " High " ทันทีและวงจรควบคุม จะส่งสัญญาณไปยังวงจรแลทช์ทางด้าน เอ้าท์พุท ให้ค้างเอ้าท์พุทไว้ วงจรเกทจะส่ง สัญญาณนาฬิกา ไปเพิ่มเอ้าท์พุท ของวงจรนับขึ้น เป็นเหตุให้วงจรนับค่าขึ้นอีกครั้ง

เทคนิคแบบ วงจรนับเทร็คกิ้ง นั้นมีความเร็วสูงกว่าเทคนิค แบบวงจรนับเดียว แต่มันยังมีข้อเสีย คือเทคนิคแบบวงจรนับเทร็คกิ้งเหมาะสำหรับการแปรสัญญาณอินพุทที่มีการเปลี่ยนอย่างรวดเร็ว ให้อยู่ในรูปสัญญาณดิจิตอล ได้ดี



รูปที่ 2.6.8 SAR เป็นตัวเร่งความเร็วของกระบวนการแปลงสัญญาณอะนาล็อกเป็นดิจิตอล

2.6.6 ADC แบบประมาณค่าหลาย ๆ ครั้ง (Successive-approximation ADC)

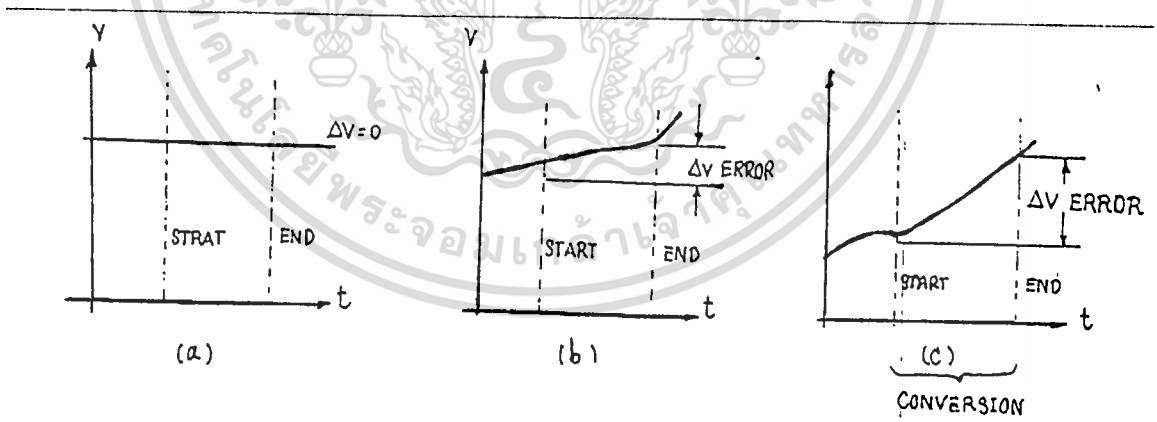
เทคนิคการประมาณค่าหลาย ๆ ครั้ง เป็นเทคนิคที่มีความสามารถสูงและใช้งานได้ดีซึ่งสามารถแปลง สัญญาณอะนาล็อก ให้เป็นสัญญาณดิจิตอลได้รวดเร็ว และมีประสิทธิภาพ เพราะไม่มีการ ออสซิลเลท หัวใจของ SA คอนเวอร์เตอร์คือ อุปกรณ์ที่เรียกว่า " Successive - approximation registor " (SAR) ซึ่งเป็นอุปกรณ์ที่มีจุดประสงค์ต่างจากวงจรนับทั่ว ๆ ไปอย่างมากดังรูป 2.6.8

การทำงานเริ่มต้น เมื่อสัญญาณอะนาล็อก ถูกป้อนให้กับคอนเวอร์ทเตอร์ และพัลส์การแปลงเริ่มต้น (Start conversion pulse "SOC")ถูกป้อนให้กับตัว SAR พัลส์สัญญาณนาฬิกาลูกแรกที่ป้อนให้กับตัวอินพุท ของอินเวอร์ทเตอร์ ตัว SAR จะ " on "เอ้าท์พุทของบิทนับสูงสุด ดังนั้นจะเป็นปรับเอ้าท์พุทของ DAC เป็น 50% ของแรงดันเอ้าท์พุทของอินเวอร์เตอร์ ตัว SAR จะมองไปยังเอ้าท์พุทของวงจรเปรียบเทียบว่า เอ้าท์พุทของ DAC มีค่ามากกว่าหรือน้อยกว่าสัญญาณอะนาล็อก ทางอินพุท ถ้าแรงดันของ DAC มีค่า มากกว่า วงจรเปรียบเทียบ จะยังคงอยู่ในสภาวะ "off" ดังนั้นตัว SAR จะ "off" บิทในนัยสูงสุดและให้ชื่อว่าสภาวะ "0" ถ้าแรงดันของ DAC มีค่าน้อยกว่าสัญญาณอะนาล็อกทางอินพุทของวงจรเปรียบเทียบจะยังคงทำงานอยู่ ดังนั้นตัว SAR ยังปล่อยให้บทนัยสูงสุด " on " อยู่ และเราเรียกว่าสภาวะหนึ่งนี้ว่า " 1 " ซึ่งสภาวะ " 1 " หรือ " 0 " นี้ จะทำภาย

ในพัลส์ของ สัญญาณนาฬิกาเพียงพัลส์เดียว บนสัญญาณนาฬิกาถัดไป ตัว SAR จะ " on " บิทนัยสูงสุด อันดับสอง จะเปรียบเทียบไปเรื่อย ๆ ตัว SAR จะส่งสัญญาณสิ้นสุดการ แปรผัน (End of conversion : EOC) ไปทำการค้างผลลัพท์ ไว้ที่เป็น เลขฐานทางเอ้าท์พุทไว้

2.6.7 Quantizing Error

Quantizing error ซึ่งมีความสามารถเนื่องมาจากการเปลี่ยนแปลงอะนาล็อกทางด้านอินพุท อยู่ระหว่างที่ทำการแปลงสัญญาณ อยู่การทำงานเริ่มต้นขึ้นที่ ADC ต้องการเวลาในช่วงหนึ่ง เพื่อที่จะสร้าง ดิจิตอลออกมาทางด้าน เอ้าท์พุท ถ้าแรงดันอินพุทเกิดการเปลี่ยนแปลงขึ้นในระหว่างการแปลงสัญญาณ ไบนารี่เอ้าท์พุทสุดท้ายจะแทนระดับแรงดันที่ท้ายสุดของวัฏจักร แทนที่จะเป็นช่วงเริ่มต้นเมื่อไม่การเปลี่ยนแปลงแรงดันอินพุทขึ้นเช่น ในกรณีนี้จะไม่เกิด Quantizing error ขึ้น ดังแสดงในรูป สัญญาณที่มีการเปลี่ยนแปลงอย่างรวดเร็ว หรือ ที่เราเรียกว่า " slew rate " นั้นจะก่อให้เกิด Quantizing error มากยิ่งขึ้นดังแสดงในรูป ตามลำดับ 2.6.9



รูปที่ 2.6.9 Quantizing error เกิดขึ้นเนื่องจากสัญญาณอะนาล็อกทางด้านอินพุทเกิดการเปลี่ยนแปลงขึ้นในระหว่างวัฏจักรการแปลงสัญญาณในรูป A จะไม่เกิด Quantizing error ขึ้นเนื่องจากแรงดันอินพุทไม่มีการเปลี่ยนแปลง ในรูป B เกิด Quantizing error เล็กน้อย ในรูป C เมื่อสัญญาณอินพุทความถี่สูงขึ้นระดับ Quantizing error จะมากขึ้นด้วย

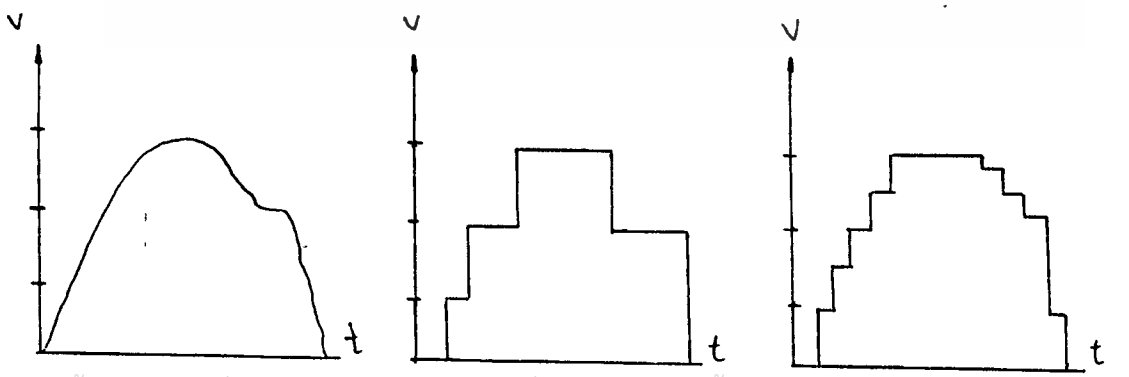
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.7 D/A คอนเวอร์เตอร์ (Digital to Analog Convertor

D/A คอนเวอร์เตอร์ หรือ เรียกย่อ ๆ ว่า ตัว DAC เป็นตัวแปลงรหัสเลขฐานสองรองจากคอมพิวเตอร์ หรือ จากวงจรดิจิตอล ใด ๆ ให้กลายเป็น ระดับอนาล็อก ที่มีความสัมพันธ์กับระบบเลขฐานสอง ตัว DAC สามารถนำไปใช้ ขับอุปกรณ์ที่เป็นอนาล็อกได้ เช่น มิเตอร์ ,มอเตอร์ อุปกรณ์ควบคุม หรือ วงจรที่เกี่ยวกับสัญญาณเสียง เช่น เครื่องเล่นคอมแพ็คดิสก์ ตัว DAC ในเครื่องเล่นคอมแพ็คนั้นถูกใช้สำหรับการเปลี่ยนข้อมูลที่บันทึกเป็นสัญญาณดิจิตอลบนแผ่น CD ให้กลายเป็นสัญญาณเสียงที่มีคุณภาพสูงออกมาให้เราได้ยิน

ต่อไปนี้เราจะพิจารณาแนวความคิดที่สำคัญของ D/A เริ่มจากความละเอียดของ DAC เราจะนิยมไว้เป็นระดับแรงดันในแต่ละขั้น ที่เอาท์พุทสามารถจะผลิตออกมาได้ ซึ่งมีความสัมพันธ์โดยตรงต่อจำนวนของบิททางด้านอินพุทที่อยู่ในรูปของรหัสไบนารี DAC ขนาด 4 บิท จะมีอินพุทบิทอยู่ 4 อินพุท ซึ่งมีความละเอียดเท่ากับ 4 จำนวนของระยะและความแตกต่างของระดับสัญญาณอะนาล็อก ทางด้านเอ้าท์พุท ที่ DAC ขนาด 4 บิทสามารถผลิตได้ด้วยระดับแรงดัน 16 ขั้นด้วยกัน

ที่นี้มาดู DAC ขนาด 8 บิท สามารถให้สัญญาณอะนาล็อกทางด้านเอ้าท์พุทที่เป็นระดับแรงดันได้ 2^8 หรือ 256 ระดับ DAC ขนาด 12 บิท สามารถในระดับแรงดันทางเอ้าท์พุทได้ 2^{12} หรือ 4096 ระดับ อย่างที่เราได้เห็นแล้วว่า DAC มีขนาดอินพุทบิทมากเท่าไรความละเอียด และ ความถูกต้องของระดับแรงดันอะนาล็อกทางเอ้าท์พุทที่ DAC สามารถผลิตได้จะมากขึ้นตาม ดังแสดงในรูปที่ 2.7.1



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น มีสงวนลิขสิทธิ์ตามกฎหมายสำหรับการนำไปใช้

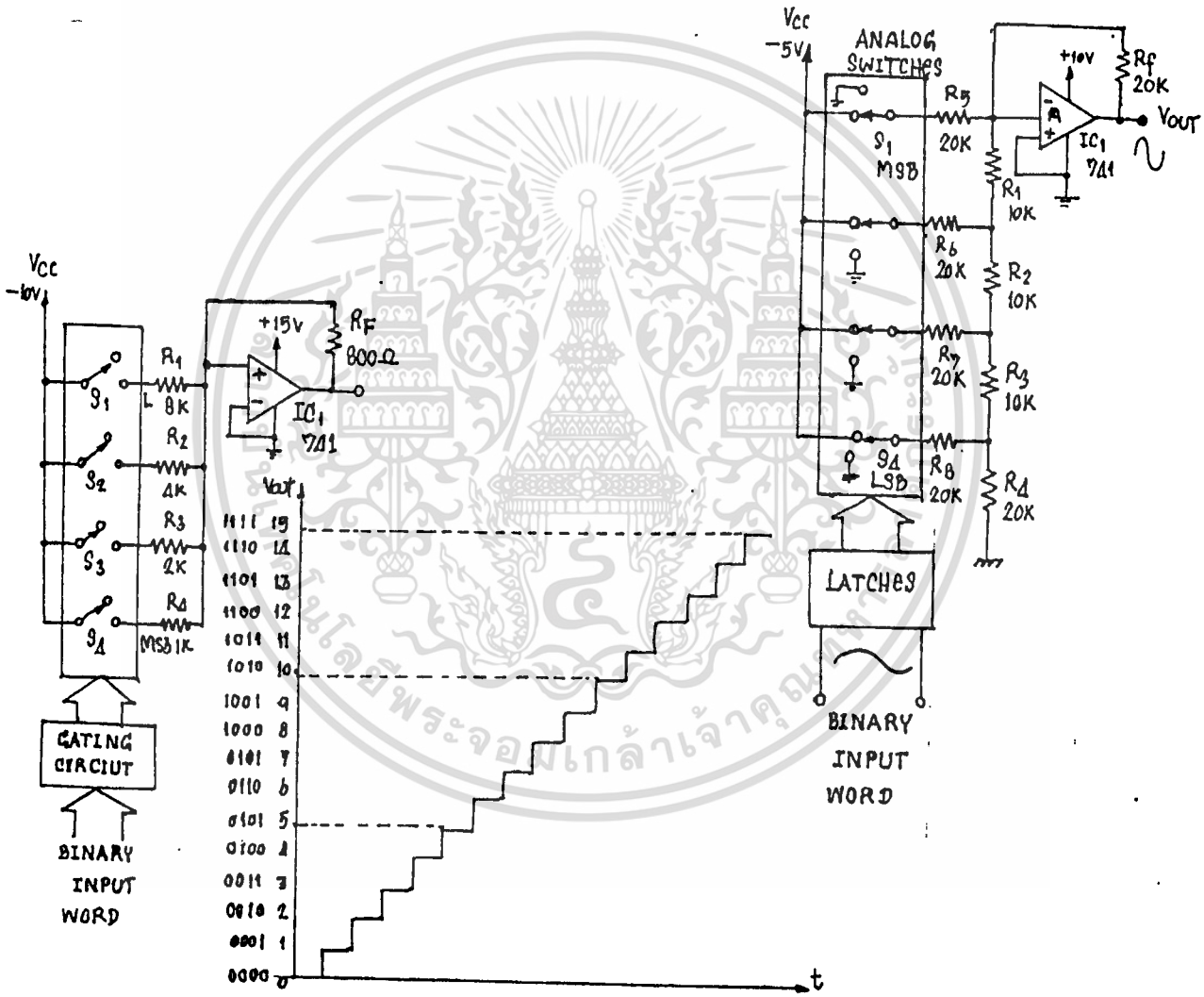
รูปที่ 2.7.1 แสดงความละเอียดของแรงดันทางเอ้าท์พุท ตัว DAC ยิ่งมีอินพุทมากเท่าไร ความละเอียดทางเอาท์พุทจะมากขึ้นตาม

ถัดมาจากความละเอียดของ DAC เราจะมาพิจารณาถึง เวลาเข้าสู่สภาวะของตัว (Settingtime) เวลาเข้าสู่สภาวะคงตัว เป็นค่าของเวลาที่ระดับแรงดันเอ้าท์พุทเข้าสู่สภาวะคงตัว เมื่อรหัสไบนารี่ ทางอินพุทเปลี่ยนแปลงไป โดยปกติ จะคิดที่สัญญาณ ทางเอ้าท์พุทคงที่ ในช่วง +- 1/2 ของ LSB (Least Singificant Bit) ของค่าที่คาดว่าจะเป็นหลังจากไบนารี่ทางด้าน อินพุทเปลี่ยนไปนั้นหมายความว่าในเงื่อนไขของการปฏิบัติงานจริง ๆ มีความสัมพันธ์กับค่า ซึ่งเป็นอยู่ในขณะนั้นต่อ LSB มีค่าเท่ากับ $10/2^8$ หรือ 0.039 โวลท์ ครึ่งหนึ่งของค่า $10/2^8$ เป็น 0.0195 โวลท์ ดังนั้น ค่าเวลาที่ระดับแรงดันเอ้าท์พุท เข้าสู่สภาวะคงที่ควรจะเป็นค่าเวลาที่เอ้าท์พุทเพิ่มขึ้นถึง 0.0195 โวลท์ ของค่าระดับที่คาดหมายไว้ ตามปกติค่าเวลาเข้าสู่สภาวะคงตัว มีค่าน้อยกว่า 10 μs ค่าความแม่นยำเป็นแฟกเตอร์ ที่สำคัญอีกตัวหนึ่งของ DAC ในเงื่อนปกติ ค่าความ แม่นยำของ DAC คือ +- ทุก ๆ ตำแหน่งจาก $^1/_2$ ถึง 2 ค่า LSB แรงดันเอ้าท์พุทสามารถเปลี่ยนแปลงไปในทาง + หรือ - ค่าของ 1 บิท ถ้า DAC มีแรงดันเอ้าท์พุทอยู่ในช่วง 0 ถึง 5 โวลท์ มีความละเอียดเท่ากับ 12 บิท LSB ควรจะเป็น $1/2^{12}$ หรือ 0.00122 โวลท์ สำหรับทุก ๆ ค่าของรหัสไบนารี่ ทางด้านเอ้าท์พุทแรงดันอาจจะสูง หรือ ต่ำกว่า ค่าที่คาดหมายไว้ 0.00122 โวลท์ ถ้า DAC ตัวเดียวกันมีค่าความแม่นยำเท่ากับ 1/2 ค่าความถูกต้อง LSB ค่าเอาท์พุทจะสามารถผิดพลาดได้ +- 0.00122 หรือ +- 0.00061 โวลท์ยิ่งค่าความละเอียดก็จะมากขึ้นตาม และ จะมีค่าใกล้เคียงกับค่าเอ้าท์พุทที่คาดไว้ หลายปีที่ผ่านมา ได้มีการค้นคิดพัฒนา วิธีการของการเปลี่ยน สัญญาณดิจิตอล ไปเป็นอะนาล็อก 2 วิธีด้วยกัน : Binary Weighted และ Binary Ladder D/A

2.7.1 Binary-Weighed Resistor D/A

เทคนิคจัดน้ำหนักของรหัสไบนารี่เป็นวิธีที่ง่ายที่สุด และเก่าที่สุดของการแปลงดิจิตอลบิทให้ กลายเป็น สัญญาณอะนาล็อก วงจรของ Binary Weighted Resistor D/A แสดงไว้ใน รูปที่ 2.7.2

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งไม่อนุญาตให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 2.7.2 แสดงวงจร Binary Weighted DAC และกราฟแสดงเอาท์พุท ของ DAC ต่อสัญญาณดิจิตอล

รหัสไบนารี่ จะถูกป้อนให้ขาของเกทอนาล็อกสวิทช์ เมื่อรหัสไบนารี่เป็น 0000 ถูกป้อนให้ที่ เกทอะนาล็อกสวิทช์ ทั้งหมดจะเปิดวงจร ดังนั้น จึงไม่มีแรงดันเอ้าท์พุทจ่ายไปให้ออปแอมป์ เอ้าท์พุทจากออปแอมป์จึงเป็นศูนย์ เมื่อรหัสไบนารี่เป็น 0001 สวิทช์ S_1 จะปิดลง และ แรงดัน 10 โวลท์จะจ่ายให้กับ R_1 เพราะว่าอินพุทของออปแอมป์จะมองได้ว่าเป็นกราวด์เสมือน (Virtoal Ground) เป็นผลให้แรงดัน 10 โวลท์ตกคร่อมตัวต้านทาน 8K (R_1) ซึ่งเป็นเหตุให้เกิดกระแส 1.25 mA (10 V/8000) ไหลผ่านความต้านป้อนกลับ (R_f) ค่า 800 แรงดันตกคร่อม R_f ควรจะมีค่า 800 x 1.25 mA หรือเท่ากับ 1 โวลท์

เมื่อรหัสไบนารี่เปลี่ยนไปเป็น 0010 S_1 จะเปิด และ 2.5 mA (10 V/4000) ไหลผ่าน R_2 แรงดันตกคร่อม R_f มีค่าเท่ากับ 800 x 2.5 mA หรือ 2 โวลท์ รหัสไบนารี่ 0100 จะให้แรงดันเอ้าท์พุทเท่ากับ 4 โวลท์และรหัสไบนารี่เป็น 1000 แรงดันเอ้าท์พุทจะมีค่าเป็น 8 โวลท์ จะสังเกตุได้ว่ารหัสทางด้านอินพุทและค่าของ R_f มีผลกระทบต่อระดับแรงดันทาง เอ้าท์พุท

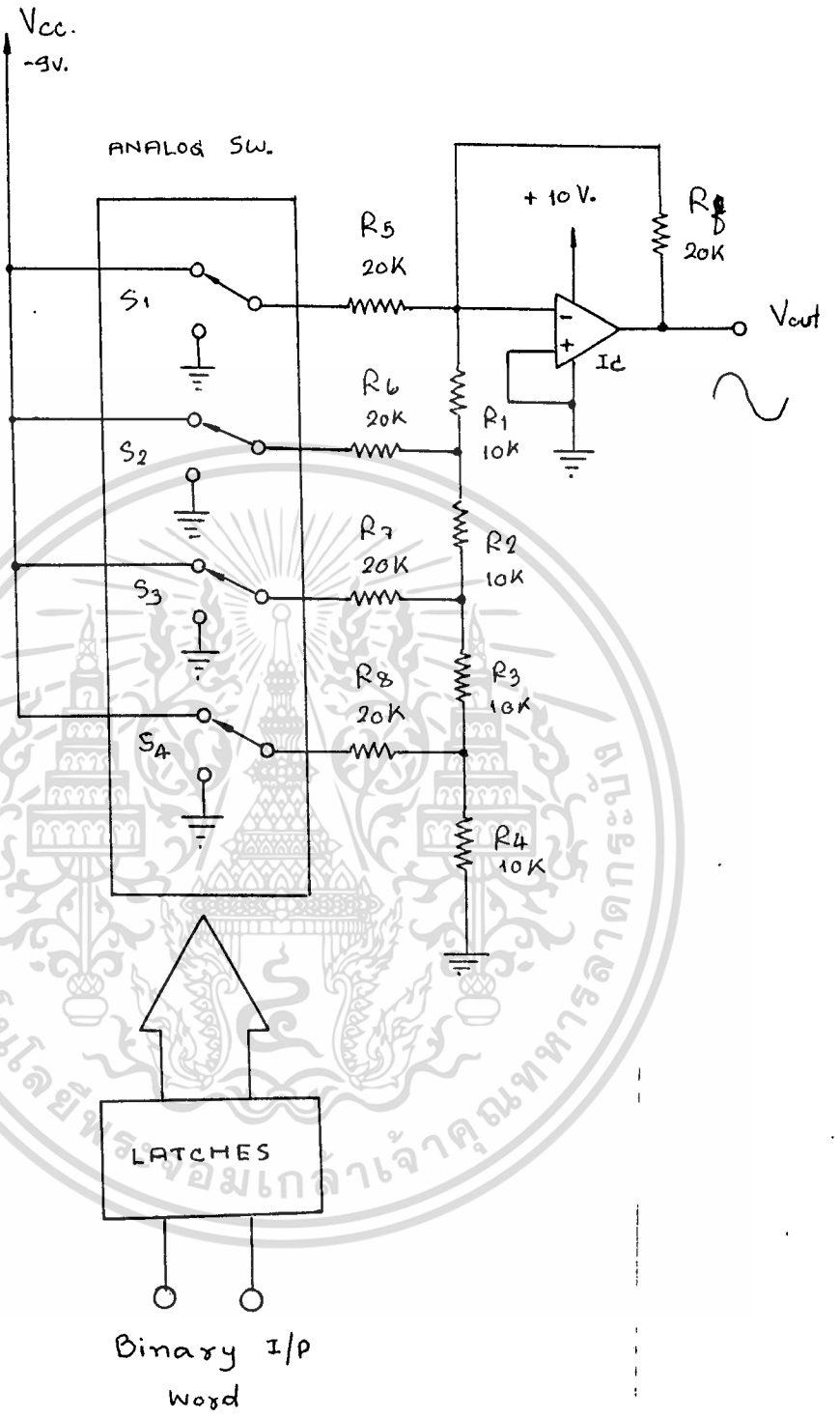
สวิทช์แต่ละตัว สามารถปิดวงจรพร้อมกันได้ เมื่อทำการร่วมกันเพื่อสร้างสัญญาณอะนาล็อกทางเอ้าท์พุท ที่มีค่าจาก 0 ถึง 15 โวลท์ (0000=0 โวลท์ , 0111=7 โวลท์, และ 1111 = 15 โวลท์)ในการเพิ่มขึ้น 1 โวลท์ต่อ 1 ขั้น

ถึงแม้ว่า Binary Weighted resistor DAC มีลักษณะวงจรง่าย ๆ ตรงไปตรงมา แต่ไม่สะดวกในการนำไปใช้งาน ถ้าต้องการความละเอียดของ DAC มากกว่า 4 บิท เพราะว่าค่าของตัวต้านทานที่ใช้ มากมายหลายค่าเกินไป ซึ่งต่างจาก Ladder network ที่ต้องการใช้ตัวต้านทาน เพียง 2 ค่าเท่านั้น

2.7.2 Ladder Network D/A

เทคนิค เลดเดอร์เน็ทเวิร์คสามารถสร้างกรงดันตามน้ำหนักของรหัสไบนารี่โดยอาศัยความต้านทานเพียง 2 ตัว เท่านั้น ที่จัดในลักษณะวงจรแบ่งแรงดัน หรือ ที่เรียกว่าไบนารี่เลดเดอร์ (Binary Ladder) ดังแสดงในรูปที่ 2.7.3

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.7.3 แสดงวงจรของวงจร Binary Ladder DAC

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ถึงแม้ว่าวงจร DAC แบบเลดเดอร์เน็ทเว็อร์คนั้นดูผ่าน ๆ แล้วค่อนข้างจะยาก แต่การทำงานของวงจร ก็ยังคล้ายกับการทำงานของวงจร Binary weighted DAC เกทที่ต่ออยู่ในลักษณะอนุกรม ถูกใช้สำหรับอะนาล็อกสวิทช์ทั้งหมดจะเปิดออก ดังนั้นแรงดันเอ้าท์พุทที่ได้จากออปแอมป์จึงมีค่าเป็น ศูนย์ สวิทช์ S_1 จะปิดเมื่อเกทได้รับรหัสไบนารี่เป็น 1000 เป็นผลให้เกิดแรงดันเอ้าท์พุท 5 โวลท์ ออกมาจากออปแอมป์ ถ้ารหัสไบนารี่เป็น 0010 สวิทช์ S_3 ก็จะปิดลง และ ทำให้เกิดแรงดัน 1.25 โวลท์ ที่เอาท์พุท และ สุดท้ายที่อินพุทเป็น 0001 สวิทช์ S_4 จะปิดลง ออปแอมป์จะให้แรงดันเอ้าท์พุทออกมา 0.0625 โวลท์ จะสังเกตุได้ว่า แต่ละแรงดันเอ้าท์พุทอยู่ในรูปอันดับของ ไบนารี่ คือ เอ้าท์พุทสามารถเปลี่ยนจาก 0 ถึง 10 โวลท์ เพิ่มขึ้นขั้นละ 0.625 โวลท์ (24 หรือ 16 ขั้น)

ข้อดี ของเลดเดอร์เน็ทเว็อร์ด DAC คือสามารถออกแบบได้ง่าย เนื่องจากใช้ความต้านทาน เพียง 2 ค่าเท่านั้น และ ในทุกวันนี้ บริษัทผู้ผลิต DAC เกือบทั้งหมดจะใช้เทคนิคแบบเลดเดอร์ เน็ท เว็อร์ค ในการผลิต DAC

วงจรเลดเดอร์ มักจะมีความถูกต้อง แม่นยำ มากกว่าวงจร binary weighted เพราะว่าเราจะหาค่าความต้านทานที่ถูกต้อง 2 ค่า(เช่น 10K หรือ 20K)ได้ง่ายกว่าความต้านทานหลาย ๆ ค่า ที่ใช้ในวงจร Binary Weighted DAC ไอซี DAC สำเร็จรูปที่นิยมใช้อยู่เบอร์ DAC-08

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 3

การใช้ PC ควบคุมวงจรเพาว์เวอร์ซัพพลาย วงจรชาร์จแบตเตอรี่ และ ดีซีโวลท์มิเตอร์

(PC CONTROL POWER SUPPLY CIRCUIT CHARGE BATTERY CIRCUIT AND DC.VOLTMETER)

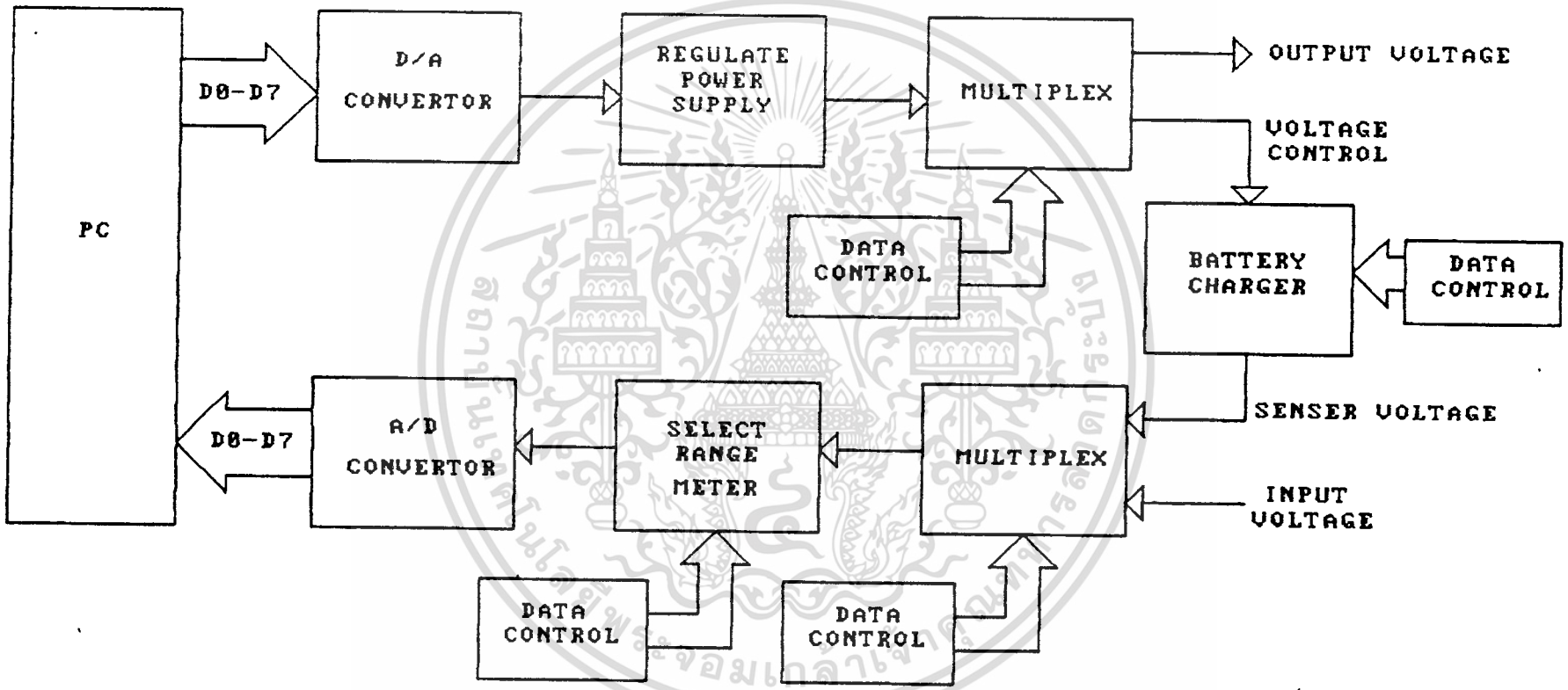
โครงการนี้ เป็นการใช้ ไมโครคอมพิวเตอร์ (PC) ประยุกต์ ใช้งานควบคุม Power supply ซึ่งสามารถเลือก Voltage ได้ 0-25 VDC จ่ายกระแสได้ 2 A. ความละเอียดขั้นละ 0.1 V. ควบคุมการชาร์จแบตเตอร์รี่ ในการเลือกกระแสที่ใช้ในแต่ละชนิด ทั้งหมด 4 ชนิด PP-3 , AAA , AA , C และ DC. Voltmeter ซึ่งสามารถ เลือกย่านการวัดได้ 4 Range คือ 200 mV. ,2V. ,20V. ,200V. ในการใช้งานเราจะใช้งานได้ 2 กรณีโดยเลือกจาก MENU บนจอภาพของ PC คือกรณีแรกเราจะใช้ Power Supply และ DC. Voltage กับงานทดลองหรืองานอื่น ๆ ภายนอก และ กรณีที่สอง เราจะใช้ Power Supply และ DC.Voltmeterในการชาร์จแบตเตอรี่

หลักการทำงาน

หลักการทำงานของวงจรซึ่งอาจจะกล่าวได้ จาก BLOCK DIAGRAM. ดังรูป 3.1 ซึ่งประกอบด้วย 3 ส่วนใหญ่ๆ คือ

- POWER SUPPLY.
- DC. VOLTMETER.
- BATTERY CHARGER.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



BLOCK DIAGRAM OF PC CONTROL POWER SUPPLY, CHARGE BATTERY CIRCUIT AND DC. VOLTMETER

โดยจะกล่าวที่ละหัวข้อดังต่อไปนี้

POWER SUPPLY

DATA D0-D7 จะเป็นตัวกำหนด Voltage ของ Power supply โดยจะเข้าวงจร D/A Convertor คือ แปลงสัญญาน DIGITAL ให้เป็นสัญญาน ANALOG แล้วนำสัญญาณอนาล็อกไปควบคุม Power Supply และ ขยายกระแสให้ได้ ตามต้องการ โดยจะมี DATA Control จาก PC เป็นตัวควบคุมในการเลือกการใช้งานของ Power Supply

DC. VOLTMETER

โดยจะมีการแปลงสัญญาณอนาล็อกที่รับเข้ามา ให้เป็นในรูปของสัญญาณ DIGITALโดยจะมี DATA Control จาก PC ที่ใช้ในการควบคุมอยู่ 2 ชุด คือ ชุดแรกจะเป็นการเลือก Range ของ DC. Voltmeter ที่ใช้ในการวัดและชุดที่สอง จะเป็น การเลือกการใช้งานของ DC. VoltMeter ว่าจะใช้ senser ในการชาร์จแบตเตอรี่ หรือ ใช้ในการทดลองหรือ งานอื่นๆภายนอก

BATTERY CHARGER

โดยจะนำส่วนของ Power Supply มาใช้ในการควบคุมกระแส ในการชาร์จแบตเตอร์รี่ของแต่ละชนิด โดยจะใช้วงจรแรงดัน ควบคุมกระแส และ ใช้ DC. Voltmeter มาตรวจจับในการชาร์จแบตเตอรี่ว่าเต็มแล้วหรือยัง ถ้าเต็มก็จะหยุดทำการชาร์จแบตเตอร์รี่ เพื่อง่ายต่อความเข้าใจ โดยจะกล่าว โดยละเอียดของวงจรซึ่งวงจรหลักๆ ประกอบด้วย

1. วงจร 8255 I/O Port
2. วงจร D/A Convertor และ Power Supply
3. วงจร A/D Convertor
4. วงจร Charge Battery

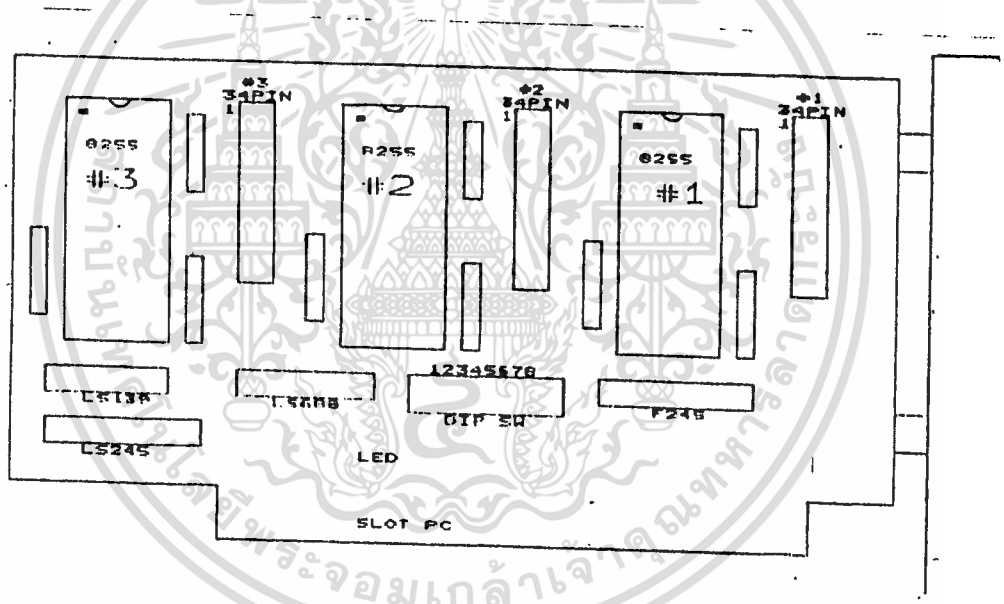
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.1 วงจร 8255 I/O Port

จากหลักการ และการทำงานของไอซี 8255 ในบทที่ 2 ที่ผ่านมา เราสามารถนำเอา ไอซี 8255 มาประยุกต์ใช้งานได้

3.1.1 ลักษณะของ 8255 I/O PORT CARD

วงจร 8255 I/O PORT จะเป็น card ต่อขยายระบบเครื่อง PC ให้มีส่วนของ input , output port ใช้งานมากขึ้นโดยจะมี portใช้งานเป็น input หรือ output จำนวน 9 port หรือ 72 bit I/O



รูปที่ 3.1.1 รูปลักษณะของ 8255 I/O CARD

3.1.2 การใช้งาน IC 8255

การใช้งานของ IC 8255 ได้อธิบายในหัวข้อ 2.1 ที่ผ่านมาแล้ว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

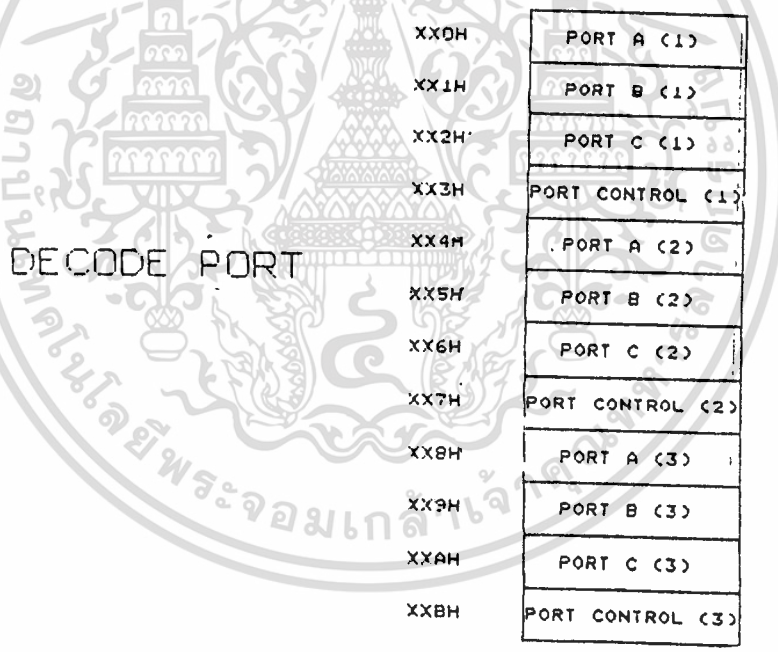
3.1.3 การทำงานของ 8255 I/O CARD

8255 I/O CARD จะประกอบไปด้วย 2 ส่วนใหญ่ ๆ คือส่วน IC 8255 ซึ่งเป็น IC ทำหน้าที่เป็น INPUT,OUTPUT PORT และส่วนของวงจร IC DECODE (เลือกตำแหน่งของ PORT 8255) คือ IC 74LS688,74LS139 และ DIP SW.

3.1.4 การ DECODE PORT

DECODE PORT 8255 บน CARD เราจะใช้ IC TTL 74LS688,IC TTL74LS139 และ DIP SW.8 PIN เป็นวงจร DECODE เพื่อให้สามารถปรับ SET DIP SW.ตั้งตำแหน่งเบอร์ PORT ของ CARD ได้ โดยในการปรับ DIP SW. นั้นจะต้องไม่ไปตรงกับตำแหน่ง PORT ของ เครื่องคอมพิวเตอร์ PC ด้วย ดังรูป 3.1.2

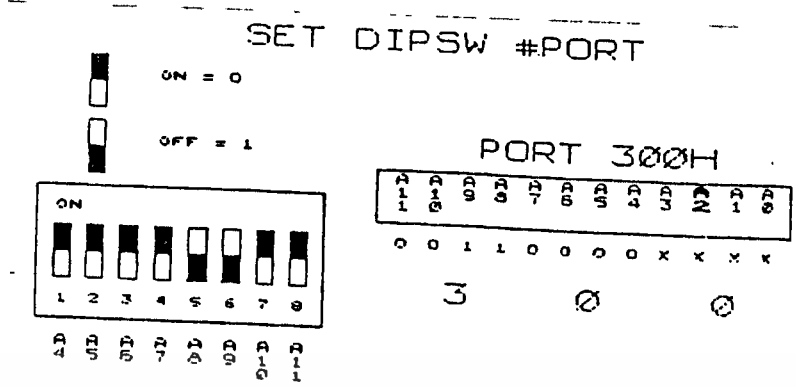
โดย CARD 8255 I/O PORT จะใช้ตำแหน่ง PORT 12 PORT ต่อ CARD



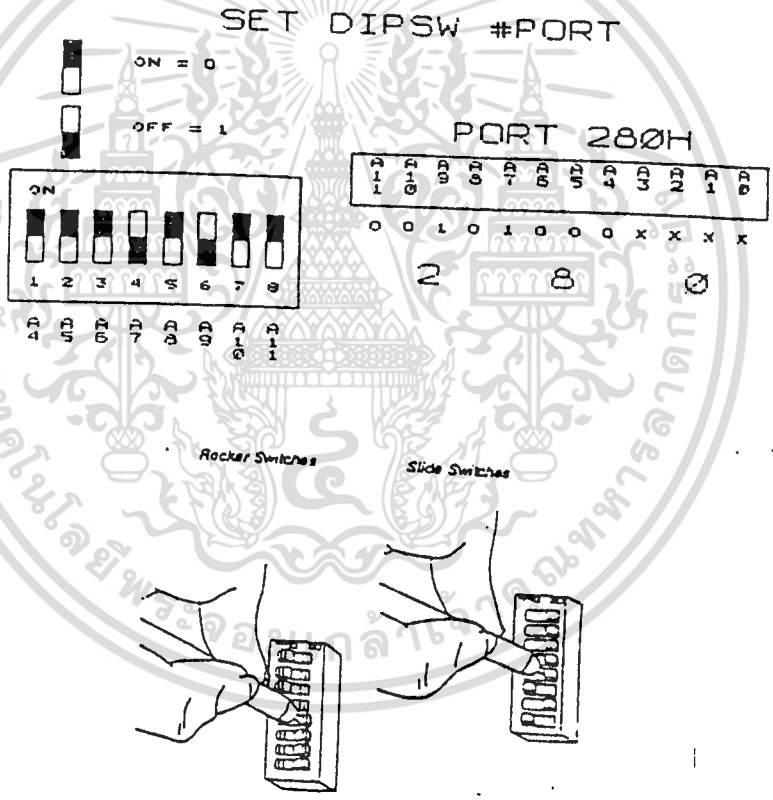
รูปที่ 3.1.2 แสดงตำแหน่ง PORT 12 ต่อ CARD 8255 I/O PORT

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เราตั้งเบอร์ DECODE PORT ได้โดยการปรับ DIPSW.ซึ่งก็มีค่าเท่ากับค่า Address นั้น ๆ เช่น เราตั้งตำแหน่ง 300H จะเซ็ท DIP SW. ดังนี้

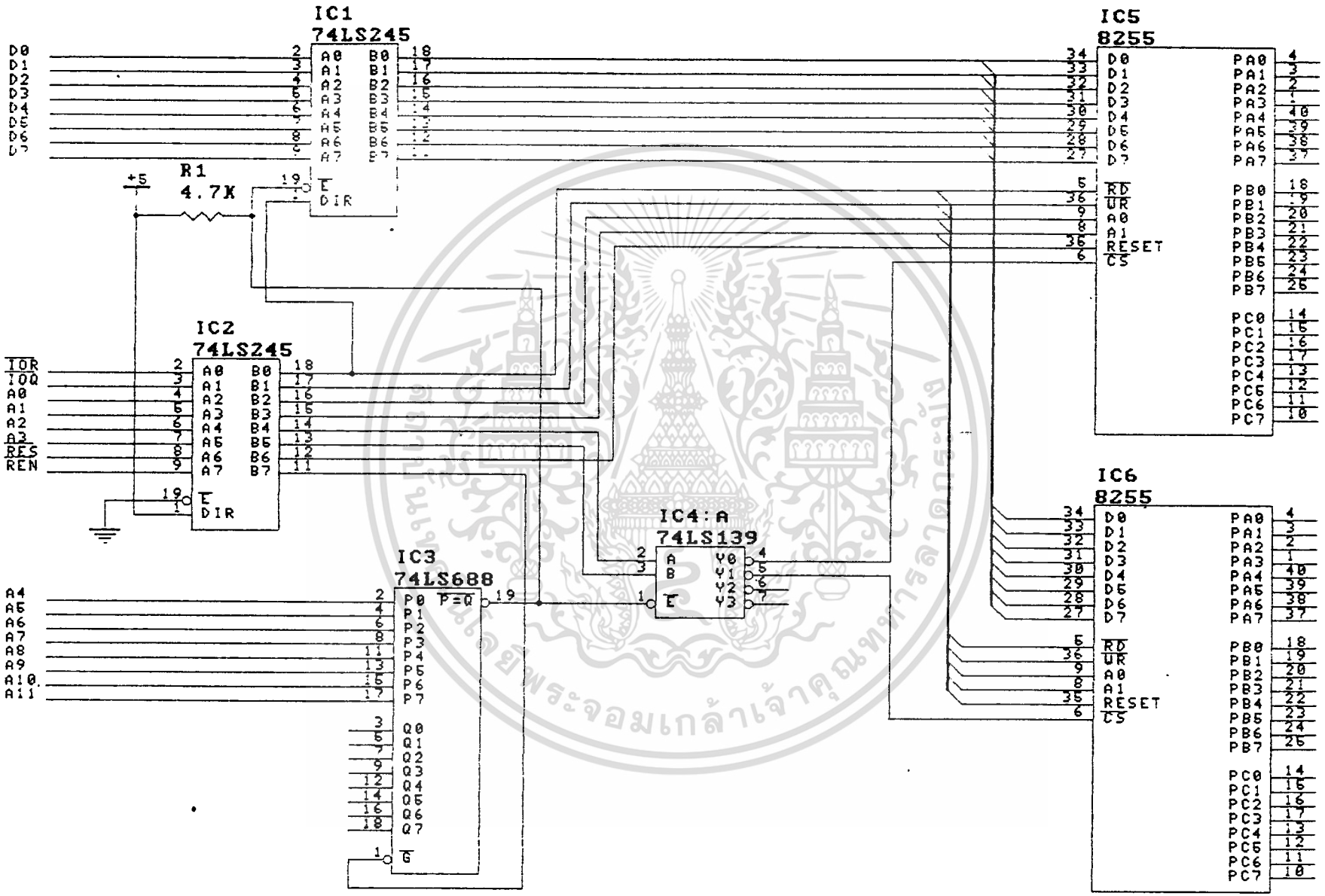


ถ้าเราต้องการตั้งตำแหน่ง 280H จะ Set DIP SW.ดังนี้



รูปที่ 3.1.3 แสดงการ Set DIP SW.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



INTERFACE 8255 I/O PORT CIRCUIT

3-2 วงจร D/A Convertor และ Power Supply

โครงสร้างวงจรนั้น เพื่อความง่าย แต่คุณภาพเชื่อถือได้ จึงใช้ IC Regurator # MC 1723 เป็นตัวที่ทำหน้าที่หลัก ในส่วนของพีซี จะส่งข้อมูล ดิจิตอลขนาด 8 บิต มาทาง 8255 I/O Port แล้วแปลงเป็นสัญญาน อะนาลอกด้วย ดีทูเอ ได้เป็นกระแสอ้างอิง ให้กับ MC1723 กระแสอ้างอิง ที่เปลี่ยนแปลงตาม ข้อมูล ดิจิตอล ก็จะเป็นตัวกำหนด แรงดัน Out Put ตามต้องการซึ่งที่ส่วน Output จะมี Transister กำลัง มาช่วยขยายกระแส เพื่อให้สามารถจ่ายกระแสได้มากขึ้น

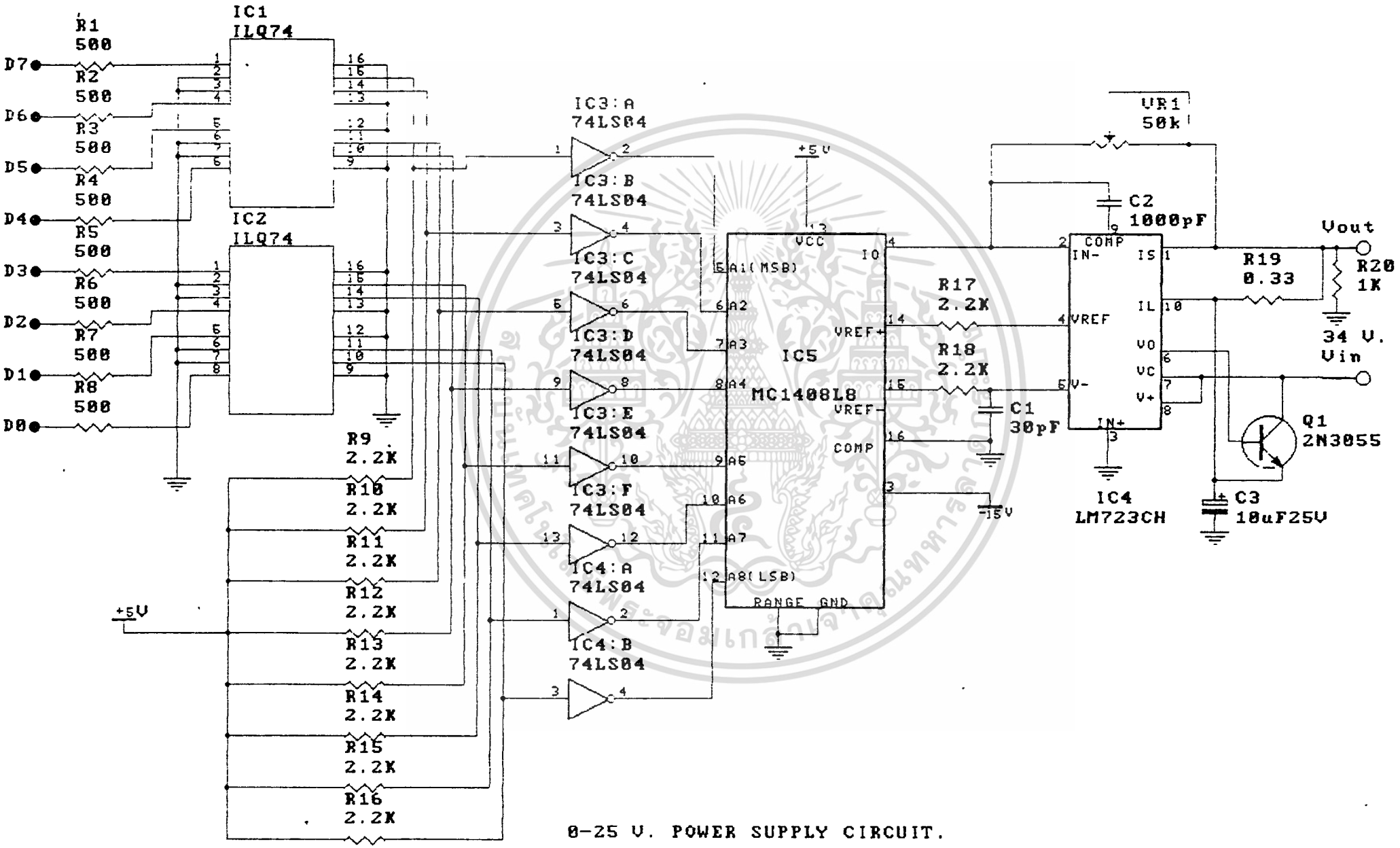
วงจรสมบูรณ์ของแหล่งจ่ายไฟ ควบคุมด้วย พีซีที่สมบูรณ์ เป็น IC แปลงสัญญาณดิจิตอล เป็น อนาลอก หรือ ดีทูเอ ที่ใช้คือ IC # MC1408L8 ซึ่งเป็นดีทูเอ ชนิด R-2R มีความละเอียด ขนาด 8 บิต หรือ 256 ระดับ ให้เอาท์พุตเป็นกระแส ซึ่งใช้งานที่พบเห็นทั่วไปจะต้องมี ออปแอมป์อีก 1 ตัว ทำการเปลี่ยนกระแส เป็นแรงดันแต่ในที่นี้ใช้ออปแอมป์ภายในตัว MC1723 ทำหน้าที่แทน

ข้อมูลดิจิตอลขนาด 8บิตได้จากเครื่องพีซีทาง 8255 I/O Portผ่านออปโต้คัปเปลอร์ เป็นตัวเชื่อมโยงสัญญาณโดยใช้แสง เพื่อแยกวงจรจากกันทางไฟฟ้ากับเครื่องพีซีลักษณะการต่อจะร่วมกัน กับสัญญาณจากออปโต้คัปเปลอร์ โดยมีตัวต้านทาน ยกระดับลอจิก ไว้ที่ "1" ลักษณะเช่นนี้เรียกว่า " wire OR " คือเป็นการต่อร่วมกันตรง ๆ แต่ทำตัวเสมือนเป็นออร์เกตตัวหนึ่ง

กระแสจากเอาท์พุตจาก IC MC 1408L8 ที่ขา 4 จะป้อนให้กับขาอินพุตกลับเฟสของออปแอมป์ภายในตัว IC MC1723 ที่ขา 2 และ มีกระแสอีกส่วนหนึ่งได้จากการป้อนกลับจากเอาต์พุตผ่าน VR ซึ่งเป็นตัวปรับแรงดันสูงสุด ส่วนขาอินพุตไม่กลับเฟสคือ ขา 3 ต่อลงกราวด์ แรงดันอ้างอิงสำหรับดีทูเอ IC MC1408L8 ด้านไฟบวก (V_{ref+}) ได้จากส่วนกำเนิดแรงดัน อ้างอิงใน IC MC1723 ด้านไฟลบ (V_{ref-}) ได้จากวงจรแหล่งไฟลบ (-15 v.)

แรงดันไฟตรง จากแหล่งจ่ายภายนอกจะต่อเข้าที่จุด Vin ผ่านทรานซิสเตอร์ผ่านกระแส TR 2N3055 มีตัวต้านทาน 0.33 OHM. เป็นตัวตรวจวัดกระแสเพื่อควบคุมกระแสที่จะผ่านไปยัง เอาท์พุตที่จุด V_{out} ทรานซิสเตอร์นี้ก็คือ ตัวที่ทำหน้าที่ขยายกระแสนั่น เอง ในที่นี้สามารถผ่ากระแสสูงสุดได้ 2 A. ซึ่งหากต้องการให้ได้กระแสมากกว่านี้ก็สามารถเปลี่ยนเป็นเบอร์อื่นที่ทนกำลังได้สูงสุดขึ้นก็ได้

ลิขสิทธิ์ของมหาวิทยาลัยเทคโนโลยีพระจอมเกล้าธนบุรี ห้ามมิให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น ไม่อนุญาตให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



0-25 V. POWER SUPPLY CIRCUIT.

วงจรนี้รับแรงดันอินพุตสูงสุดจำกัดอยู่ที่ MC 1723 ซึ่งทนได้เพียง 35 โวลท์เท่านั้น แรงดันอินพุตนี้ จะต้องสูงกว่าแรงดันเอาต์พุตสูงสุดอยู่ไม่น้อยกว่า 3 โวลท์ อันเป็นแรงดันตกคร่อมปกติ

ถ้าจ่ายแรงดันอินพุต ที่ 28.5 โวลต์ จะสามารถแรับแรงดันเอาต์พุตได้จาก 0-25 โวลท์ โดยมีความละเอียดขั้นละ 100 มิลลิโวลท์ ถ้าแรงดันอินพุตต่ำลง (โดยปรับ VR_1 ให้สอดคล้อง) ค่าแตกต่างแรงดันของแต่ละขั้นก็จะลดลงเป็นสัดส่วนกัน

ข้อดี ของแหล่งจ่ายไฟควบคุมด้วยพีซีวงจรนี้ก็ คือการใช้ออปโต้คัปเปลอร์เป็นตัวเชื่อมโยงสัญญาณ ทำให้กราวด์ของวงจรแยกจากกราวด์ของเครื่องพีซี ดังนั้นจึงสามารถใช้งานแหล่งจ่ายไฟเครื่องนี้ได้อย่าง อิสระ และ ไม่ต้องกลัวว่าจะมีผลกระทบ ทำความเสียหายต่อคอมพิวเตอร์ราคาแพง

3.2.1 การใช้งาน

ถ้าใช้งานลำพังโดยไม่ได้เชื่อมต่อกับคอมพิวเตอร์ก็ใช้ดิพสวิตช์ เป็นตัวกำหนดแรงดันเอาท์พุต ที่ต้องการค่าของตัวเลขดิจิตอล จะสัมพันธ์กับแรงดัน Output โดยเทียบกับค่าแตกต่างแรงดันของแต่ละขั้น ซึ่งค่าแตกต่างแรงดัน แต่ละขั้น ก็ได้จากแรงดันเอาท์พุต สูงสุดหารด้วย 256 โดยประมาณ เช่นถ้าแรงดันสูงสุด เป็น 25 โวลท์ (อินพุต 28.5) ค่าแรงดันแต่ละขั้น เท่ากับ 100 mV. (ประมาณ) หากต้องการปรับแรงดัน Output ไปที่ 15 V.ก็ต้องป้อนค่าตัวเลขดิจิตอล เท่ากับ 150 หรือเป็นเลขฐานสอง "10010110" ซึ่งการตั้งดิพสวิตช์นั้น สำหรับวงจรนี้ถ้าเลื่อนไปทาง ON มีค่าเป็น"0"และ OFF จะเป็น "1"

ส่วนการเชื่อมต่อกับเครื่องพีซีก็ใช้สายต่อไปทางคอนเนคเตอร์ ต้องการแรงดันเท่าไรก็ส่งค่าตัวเลขมาที่ พรินเตอร์ พอร์ต วิธีการคิดก็เช่นเดียวกันกับที่กล่าวไปแล้ว 8255 I/O Port หรือพอร์ตขนาน จะมีหมายเลขพอร์ตอยู่ที่ 300

3.3 วงจร A/D Converter

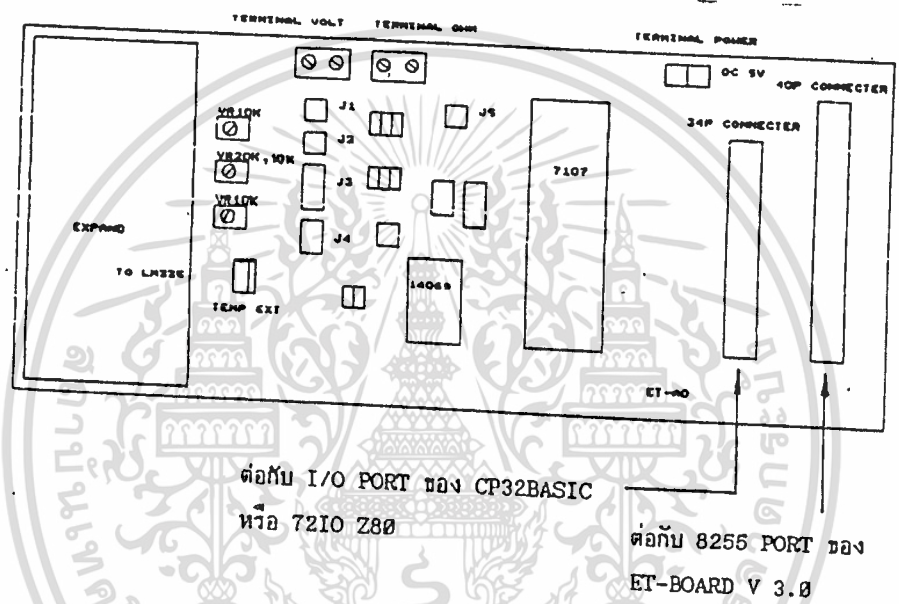
3.3.1 การใช้งานวงจร A/D CARD

ลักษณะของ A/D CARD

A/D CARD จัดเป็นการ์ด ประเภท เปลี่ยนแปลงระบบจาก อนาลอกเป็น ดิจิตอล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น สงวนลิขสิทธิ์ตามกฎหมาย ห้ามมิให้ทำซ้ำ ดัดแปลง เผยแพร่ หรือจำหน่ายเอกสารทุกครั้งที่มีการนำไปใช้

(ANALOG TO DIGITAL) หรือเรียกกันง่าย ๆ ว่า A TO D การ์ดนี้ใช้ IC 7107 เป็น A TO D ขนาด 3 1/2 DIGIT หรือ 0 - 1999 ระดับ คือเทียบเป็น A TO D ได้ขนาดประมาณ 10 BIT โดยที่ราคาจะถูกกว่า A TO D แบบ 10 BIT จริง ๆ อยู่มาก จากคุณสมบัติของ 7107 สามารถนำมาประยุกต์เป็นเครื่องมือวัดค่าต่าง ๆ ได้เช่นวัดค่าโอห์ม วัดค่าโวล์ท วัดอุณหภูมิ เป็นต้นเราได้ผลิตให้มีรูปแบบที่ใช้งานง่าย บนการ์ดเดียวสามารถวัดได้ 3 ค่า และ สามารถนำไปพัฒนาต่อได้โดยมีลาย PCB เอนกประสงค์เป็น EXPANDTION แสดงผลด้วย 7 SEGMENT โดยจะใช้คู่กับ ET BOARD,CP32ซึ่งเราได้ผลิตมาเพื่อSUPPORT อยู่แล้ว ดังจะกล่าวรายละเอียดต่อไป



รูปที่ 3.4.4 ส่วนประกอบของ ET-A/D CARD

3.3.2 การทำงานของวงจร

อุปกรณ์ที่สำคัญของวงจรคือ IC 7107 เป็นไอซีที่เปลี่ยนแปลง สัญญานอนาลอกเป็นดิจิตอล แสดงผลเป็นตัว เลข 7 SEGMENT โดยต่อออกจาก CONNECTOR ที่กำหนดไว้ดังรูปที่ 1 ซึ่งเป็น CONNECTOR 2 แบบ คือ แบบต่อใช้ร่วมกับ ET BOARD โดยออกจาก 8255 PORT เป็น CONNECTOR 40 p อีกแบบต่อร่วมกับ CP32 หรือ 72IOZ80 เป็น CONNECTOR 34 P การ์ดนี้เราได้ผลิตให้ใช้แรงดัน 5 V. จาก ET BOARD และ CP32 72IOZ80 โดยจะมีขั้วต่อไฟ 5V.ไว้ ให้ในกรณีใช้งานอเนกประสงค์ วงจรนี้จะใช้แรงดัน 5 V.ซึ่งต้องจ่ายกระแสได้ไม่น้อยกว่า 200 mA. แต่จะใช้กระแส 200 mA เพียงครั้งคราว

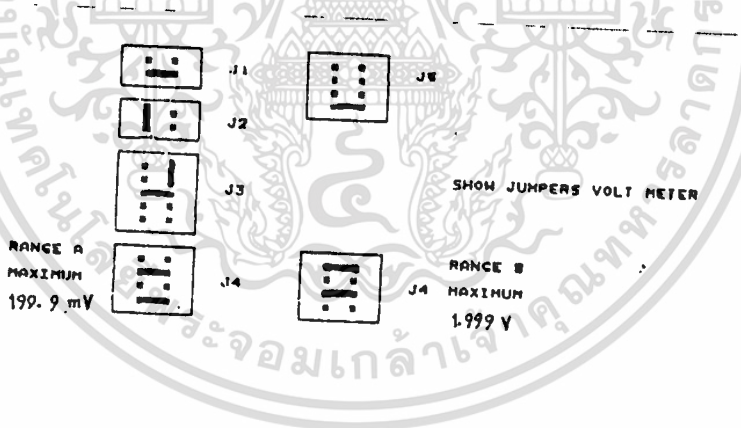
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเอกสารทุกครั้งที่มีการนำไปใช้

ปรกติจะใช้ 1 mA. โดยที่ขา 1 ของ IC 7107 เป็นไฟ +5 V. ขา 21 เป็นกราวด์ และขา 26 เป็นไฟ -5 V. จากวงจรจะใช้ไปลบเทียมมี IE 4069 , DIN4148 ใช้ 4069 ขยายสัญญาณจากขา 38 ของ IE 7170 ให้สูงมากขึ้น เพื่อเอาไปเข้าส่วนทวี แรงดันโดย 4148, 0.1 MF 47 MF จะได้ไฟลบเทียมประมาณ -2.3 V. ET-AD นำไปใช้งาน ร่วมกับ ET-BOARD, CP32 หรือ 72IO Z80. โดยจะให้แสดงผลที่ DISPLAY ของ ET-BOARD หรือ จะให้ผ่านทาง PORT ของ 72IO Z80 ก็ได้ และในกรณีนำไปใช้กับ CP32 ก็จะต่อออกทาง I/O PORT การ์ด ET-AD นี้กำหนดให้ใช้เป็นเครื่องวัด ได้ 3 อย่าง จะเลือกใช้เป็นเครื่องวัดแบบใด ๆ ได้โดยการเปลี่ยน ตำแหน่ง JUMPER และปรับค่าความต้านทาน VR1, VR2 และ VR3 แต่ในกรณีนี้เราจะใช้ในการวัดไฟฟ้ากระแสตรงเท่านั้น

3.3.3 เครื่องวัดแรงดัน (Volt Meter)

ขั้นตอนการปรับ A/D CARD ให้เป็น DC VOLT METER ดังนี้ คือ

1. เลือกต่อสาย JUMPER ในกรณีใช้งานร่วมกับ ET-BOARD,CP32 หรือ 72IO Z80
2. ปรับ JUMPER เป็น VOLT METER



รูปที่ 3.4.5 Show Jumpers Volt Meter

จากรูปการปรับ JUMPER จะเห็นว่า J4 นั้นมีให้เลือกปรับเป็น 2 RANGE การวัด คือ

- RANGE A จะวัดค่า VOLT ได้ MAXIMUM 199.9 mV.หรือ 0.199 v.
- RANGE B จะวัดค่า VOLT ได้ MAXIMUM 1.999 V.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น นอกจากนี้ยังห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

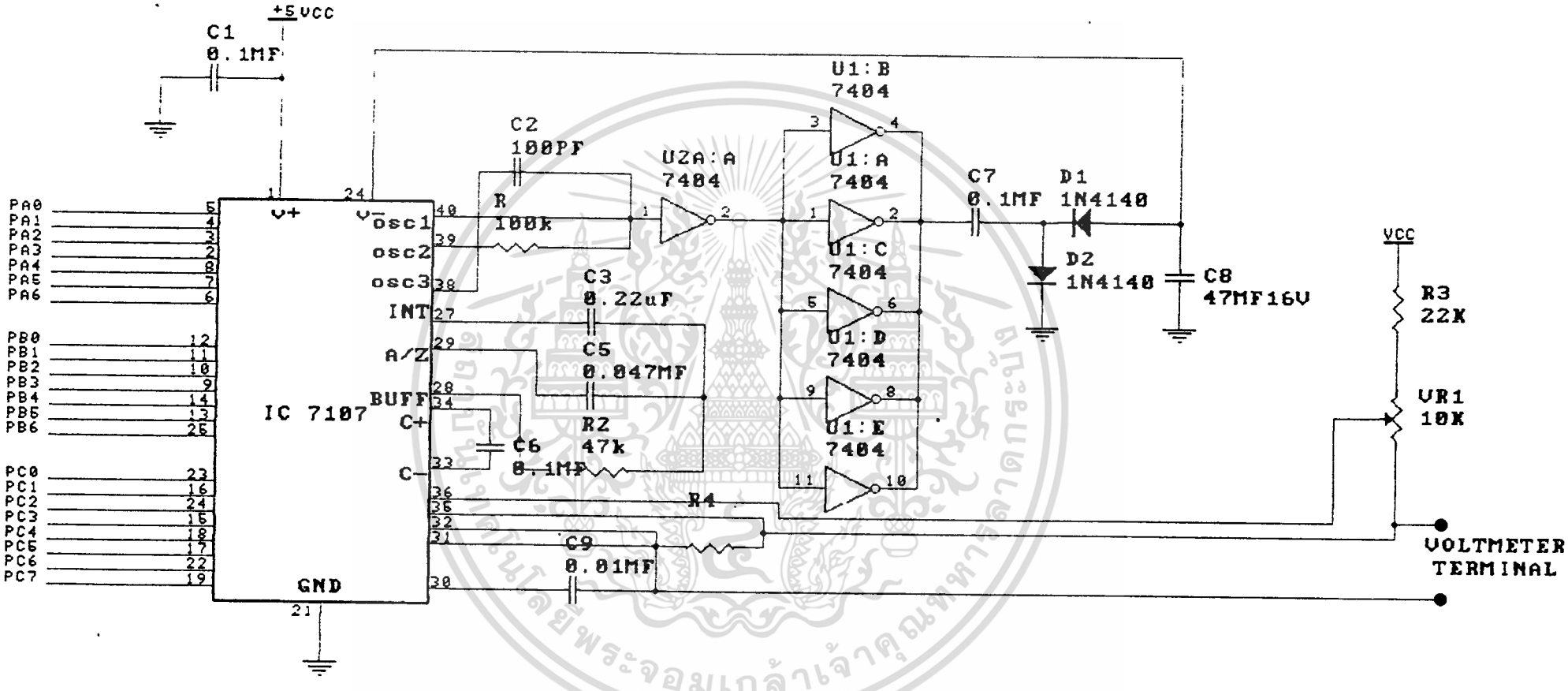
3. CALIBATE VOLT METER โดยต่อสายแรงดันไฟ DC ให้ตรงขั้วสายบวก ลบ ป้อนแรงดันค่าแรงดันระหว่าง 0-0.199 V. กรณี RANGE A และป้อนแรงดัน 0-1.999 V.ใน RANGE น่า VOLT METER ทั่วไป มาเทียบค่าให้อ่านได้ค่าเท่ากัน โดยปรับค่าความต้านทาน VR1 10K

4. CALIBATE ทั้งสอง RANGE ให้ได้ค่าเท่ากับ VOLT METER ที่นำมาเทียบและทดสอบอ่านแรงดันที่ค่าต่าง ๆ กัน เพื่อให้ได้ค่าความผิดพลาดน้อยที่สุด เมื่อ CALIBARE ได้ค่าที่ถูกต้องแล้ว ก็นำไใช้วัดเป็นโวล์ทมิเตอร์ได้เหมือนกับทั่วไป โดยมี 2 RANGE ให้เลือกใช้ซึ่งมีโปรแกรมให้ทดสอบด้านหลัง และสามารถนำไปพัฒนาใช้งานได้อีกมาก ซึ่งจะกล่าวไว้ในเรื่องการนำไปพัฒนา

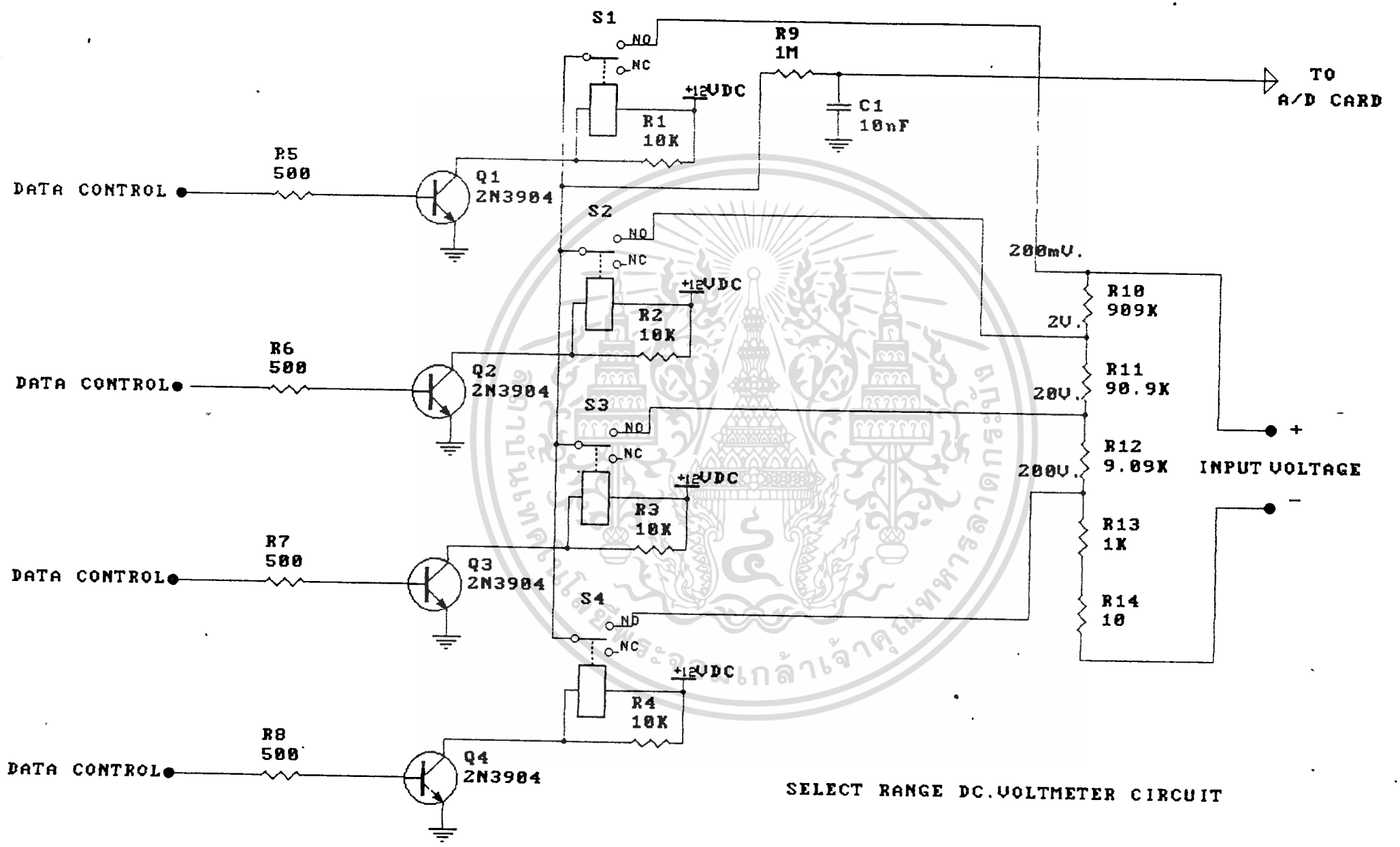
3.3.4 วัดแรงดันไฟฟ้า DC (DC VOLT METER)

จากเดิม วงจรสามารถวัดได้ 2 ย่านวัด คือ 199.9 mV.จากรูปเราใช้วงจร ATTENUTOR ซึ่งทำงานคล้าย ๆ กับ VOLTAGE DIVIDER ค่าความต้าน INPUT ใช้ค่า 1M และใช้ C 10 Nf Filter ความถี่ต่ำ ดังนั้นเราจะเพิ่มย่านการวัด ให้เพิ่มขึ้นเป็น 4 ย่าน จากความต้านทานที่ใช้เป็น ATTEMUTOR ดังนี้ จะได้ RANGE 0 - 200mV., 909 K - 2 V., 999.9 K - 20 V., 99.9 K - 200 V. และ ปรับ RANGE ของ A/D CARD ให้อยู่ที่ 199.9 mV. ต่อเข้ากับ TEMINAL VOLT METER ของ A/D CARD ได้ตามวงจร ดังรูป 3.4.6

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



A/D CONVERTOR CIRCUIT



SELECT RANGE DC.VOLTMETER CIRCUIT

3-4 วงจรชาร์จแบตเตอรี่ (Battery Charger)

ก่อนที่จะออกแบบชุดวงจร ชาร์จแบตเตอรี่ เราควรจะทราบ ถึงการชาร์จ และ ชนิดต่าง ๆ ของแบตเตอรี่ โดยต่อไปนี้ จะกล่าวถึงทฤษฎีของ แบตเตอรี่แบบ นิกเกิล-แคดเมียม

3.4.1 เซลแบบนิกเกิล-แคดเมียม (Nickel_Cadmium)

เซลแบบทุติยภูมิที่จะกล่าวก็คือ เซลแบบนิกเกิล-แคดเมียมเรียกกันย่อ ๆ ว่า นิ-แคด บางครั้ง เซลแบบนิ-แคดนี้ จะถูกเรียกว่า เซลแบบ DEAC ซึ่งจะเป็นชื่อย่อของบริษัทแรกที่ผลิตขึ้นมา คือ Deutsche Edison Akkulumulatoren Company ซึ่งอยู่ในเยอรมัน

ขั้วบวกของเซลแบบนิแคดนี้ทำจากนิกเกิลไฮเดรท (nickel hydrate) ส่วนขั้วลบนั้นทำ จากแคดเมี่ยมไฮดรอกไซด์ (cadmium hydroxide) อิเลคทรอไลท์ทำจากสารละลายโปรตัสเซียมไฮดรอกไซด์ (potassium hydroxide) ซึ่งก็เหมือนกับเซลแบบตะกั่วกรดคือ ปฏิกริยาเคมี ในการประจุและคายประจุ เป็นดังนี้

$$Cd+2NiOOh+2H_2O$$

$$Cd(OH)_2 +2Ni(OH)_2$$

โดยสถานะประจุเต็มที่ คือ ในเซลที่จะได้รับการประจุจนเต็มขั้วลบจะเป็น แคดเมี่ยมบริสุทธิ์ ซึ่งจะถูกออกซิไดซ์ (oxidised) ในระหว่างการคลายประจุ ในระหว่างการคลายประจุนอกจากปฏิกริยาหลักที่เกิดขึ้นที่ขั้วบวกแล้วนั้น จะมีปฏิกริยาข้างเคียงเกิดขึ้นด้วย ซึ่งจะก่อให้ เกิดก๊าซออกซิเจนสามารถเคลื่อนที่จากขั้วบวกไปรวมตัวกับขั้วลบ

ปฏิกริยาข้างเคียงที่เกิดขึ้นที่ขั้วลบจะผลิตก๊าซไฮโดรเจนขึ้น โดยจะเกิดขึ้นเมื่อขั้วลบอยู่ในสถานะประจุเต็มที่ โดยเราจะแน่ใจได้ว่า ก๊าซไฮโดรเจนที่เกิดขึ้นจะไม่รั่วไหลออกไป ถ้าทำให้ ขนาดของขั้วลบใหญ่กว่าขั้วบวก

เราจะเห็นได้ว่าเมื่อเซลถูกประจุจนเต็ม กระแสที่ไหลผ่านเซลทั้งหมด จะใช้ในการผลิตก๊าซออกซิเจนที่ขั้วบวกซึ่งจะผ่านไปรวมตัวกันที่ขั้วลบ ซึ่งโครงสร้างของเซลเป็นสิ่งที่จำเป็นมากที่จะต้องสร้างให้มีทางให้ ก๊าซออกซิเจน มารวมตัวได้ ถ้าไม่เช่นนั้นแล้ว ก๊าซออกซิเจนที่เกิดขึ้นจะทำให้ เกิดความเสียหายขึ้นได้ ซึ่งเราจะว่าเราสร้างด้วย แผ่นเพลทที่ผ่านการเผาเพื่อให้แผ่นเพลทมีรูพรุน มาก ๆ เพื่อช่วยให้ก๊าซออกซิเจนวิ่งจากขั้วบวกไปสู่ขั้วลบได้สะดวกขึ้น แผ่นเพลทที่ผ่านการเผานี้ จะเป็นประโยชน์โดยช่วยเพิ่มหน้าสัมผัสของแต่ละ

ขั้ว ถ้าเซลแบบกระดุมนี้ถูกประจุมากเกินไปตัวเซลนี้ก็จะถูกระเบิดขึ้นได้ดังนั้นจึงต้องระมัดระวังในการประจุไฟเข้าเซล ปัญหาที่เกิดขึ้นอีกข้อก็คือ เนื่องจากปฏิกริยารวมตัวของก๊าซออกซิเจน และ อิเลคโทรดขั้วลบ เป็นปฏิกริยาคายความร้อน ซึ่งจะทำให้ อุณหภูมิของเซลสูงขึ้น อาจทำความเสียหายแก่เซลได้

3.4.2 ปัญหาที่เกิดขึ้น

ปัญหาเกี่ยวกับการระเบิดของเซลแบบนี้ สามารถบรรเทาลงได้โดยการ ใช้เซลนิแคดแบบรูปทรงกระบอก ใช้แผ่นเพลทที่เผา มาทำเป็นขั้วบวก และขั้วลบอีก แต่เรานำมาม้วนให้เป็นรูปทรงกระบอก และ มีรูระบายติดตั้งอยู่ที่ฝาบนของเซล ซึ่งจะปล่อยก๊าซออกซิเจนออกสู่ภายนอกเมื่อความดันขึ้น สูงกว่า 90 ปอนด์/ตารางนิ้ว ดังนั้น ถ้าเซลถูกประจุมากเกินไปด้วยกระแสที่สูงเกิน ก๊าซออกซิเจน ที่เกิดขึ้นจะถูกระบายออกจากเซล ทำให้เซลไม่เกิดการระเบิดขึ้นอย่างไรก็ตาม ก๊าซออกซิเจนที่ระบายออกไป ก็ไม่สามารถหากลับมาทดแทนได้ เซลที่มีรูปร่างเป็นทรงกระบอกนี้สามารถที่จะเก็บพลังงานได้มากกว่าเซลแบบกระดุม เมื่อมีขนาดเท่า ๆ กันและสามารถจ่ายกระแสได้มากกว่าด้วย (เนื่องจากมีความต้านทานภายในต่ำ) กุญแจแห่งความสำเร็จของเซลแบบทรงกระบอกนี้คือการใช้ตาข่ายนิเกิลบริสุทธิ์ที่มีรูพรุนเล็ก ๆ มาทำเป็นตะแกรงเพื่อให้นิกเกิลไฮดอกไซด์และแคดเมียมไฮดรอกไซด์สามารถก่อตัวเป็นขั้วบวก และ ขั้วลบบนตะแกรงนั้น ได้อย่างรวดเร็วแผ่นนิกเกิลจะถูกเชื่อมกับอิเลคโทรด และ ต่อตัวกับถังด้านบนของตัวแบตเตอรี่ ประโยชน์อีกด้านหนึ่งของเซลที่มีโครงสร้างแบบทรงกระบอกก็ คือ สามารถทำให้เซลให้มีขนาด เท่ากับขนาดของเซลแบบปฐมภูมิที่มีใช้กันอยู่ได้ คือขนาด AAA ,AA ,C ,D และ ขนาด PP-3 และอื่น ๆอีกซึ่งหมายความว่าเราสามารถนำเซลแบบนิแคดนี้ มาใช้เป็นแหล่งจ่ายไฟในเครื่องใช้ต่าง ๆ ภายในบ้าน เช่น วิทยุ เทปคาสเซทท์ ไฟฉาย เป็นต้น แรงดันของเซล แบบนิแคดนี้ โดยปกติมี ค่าประมาณ 1.5 โวลท์ แล้วอาจจะทำให้คิดว่าเครื่องใช้ต่าง ๆ จะไม่ สามารถที่จะทำงานได้ถ้าเปลี่ยนจาก เซลแห้งธรรมดา ไปเป็น เซลแบบนิแคดขนาดเดียวกัน แต่ก็ไม่ เป็นความจริงเ นื่องจาก

1. แรงดันของเซลแห้ง ที่กล่าวมานั้นเป็นแรงดันตอนไม่มีโหลดอยู่ซึ่งแรงดันนี้จะตกลงเล็กน้อย เมื่อโหลดดึงกระแสไปใช้ ทั้งนี้เนื่องมาจากค่าความต้านทานภายในเซล ซึ่งเมื่อต่อเซลอนุกรมกัน หลาย ๆ เซลแล้ว แรงดันตอนใช้งานอาจจะเหลือเพียงเซลละ 1 โวลท์ (หรือน้อยกว่า) แต่ค่าความต้านทานภายในที่ต่ำมาก ๆ ของเซลแบบนิแคดนี้จะทำให้แรงดัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตอนใช้งานหลายอย่างดูเหมือนว่า แรงดันปกติที่ต่ำของเซลนิแคดนี้ จะให้คุณสมบัติที่ดีกว่าเซลธรรมดา

2. แรงดันของเซลแห้งนี้จะแปลเปลี่ยนไปตามกาลเวลา ถ้าเราเขียนกราฟระหว่างแรงดัน ของเซลแห้งกับเวลาและเปรียบเทียบกับกราฟของเซลแบบนิแคด จะเห็นได้ว่าแรงดันของเซลแห้ง จะสูงกว่าแรงดันของเซลแบบนิแคดในตอนแรก แต่เมื่อเซลคลายประจุออกไปแล้ว จะเห็นได้ว่าในที่สุดแรงดันของเซลแห้งนี้ จะเริ่มต่ำกว่าแรงดันของเซลแบบนิแคดในขณะที่แรงดันของเซลแบบนิแคดจะค่อนข้างคงที่ และ จุดหมดประจุ คือเวลาที่คิกว่าเซลคลายประจุหมด อย่างสมบูรณ์แล้วจะเกิดขึ้นอย่างรวดเร็วเมื่อใช้เซลแบบนิแคดนี้ในเครื่องใช้ไฟฟ้า เมื่อถึงจุดที่เซลหมดประจุเครื่องใช้นั้นก็จะหยุดทำงาน ทันที

3.4.3 เมื่อคิดถึงราคา

ชิ้นส่วนในการทำเซลแบบนิแคดนี้ได้เลือกสรรค์ และควบคุมอย่างเคร่งครัด ในการเลือกวัตถุมาใช้ ตลอดจนใช้เทคนิคในการผลิต ที่ประณีต จึงทำให้เซลแบบนิแคดนี้มีราคาที่แพงมาก โดยจะแพงกว่า เซลแบบอัลคาไลน์แมงกานีส ขนาดเดียวกันประมาณ 2.5 เท่า ในขณะที่ความจุพลังงานมีเพียง 1 ใน 4 ของแบบอัลคาไลน์แมงกานีส ถ้าเรามองผิวเผินแล้วอาจจะเห็นได้ว่าจะไม่คุ้มค่า แต่อย่าลืมว่าเซลแบบนิแคดนี้สามารถประจุไฟกับเข้าไปใหม่ได้ และ สามารถใช้ได้นานหลายครั้งกว่า ไม่เหมือนกับ เซลแบบปฐมภูมิ เช่น เซลแบบอัลคาไลน์ แมงกานีส เพื่อจะดูว่าเซลแบบนิแคดนี้มีราคา คุ้มค่ากว่าเซลแบบปฐมภูมิ เราต้องมาศึกษาถึงวิธีการประจุไฟก่อน ก่อนที่จะปะจุไฟให้กับเซลแบบ นิแคดไม่ให้เกิดความเสียหาย และ สามารถประจุได้เต็มที่นั้น เราจะต้องรู้ถึงค่าความประจุของเซล ก่อน ความจุของเซลแบบนิแคดนี้ คือ ปริมาณของพลังงานไฟฟ้าทั้งหมด ซึ่งเซลสามารถจ่ายออกมาได้เมื่อมันได้รับการประจุไฟจนเต็มที่ จะแสดงออกมาในรูปของตัวเลขเป็น แอมป์ชั่วโมง หรือมิลลิแอมป์-ชั่วโมง เซลขนาดใดก็ตามค่าตัวเลขจริง ๆ นี้ จะแปรเปลี่ยนไปโดยขึ้นอยู่กับกระแสที่จ่ายออกไป ดังนั้นมักจะกำหนดเงื่อนไขในการจ่ายกระแสเมื่อระบุถึงค่าความจุผู้ผลิตจะแจ้งถึงความจุโดยปกติของเซลขนาดต่าง ๆ ระบุไว้ในตารางต่าง ๆในเซลแบบทรงกระบอกนี้ค่าความจุโดยปกติ รู้จักกันในนามของ "ความจุ 5 ชั่วโมง" เนื่องจากเป็นจำนวนของพลังงานไฟฟ้าที่เซลจะจ่ายออกมาได้เมื่อใช้เวลาในการจ่ายกระแสใน 1 ชั่วโมงจะคายประจุออกจากเซลในเวลา 1 ชั่วโมง โดยให้สัญญลักษณ์ว่า "C" เช่นเดียวกับ อัตรการจ่ายใน 5 ชั่วโมง (C/5) จะหมายถึง กระแสที่สามารถจ่ายออกจากเซลในเวลา 5 ชั่วโมงเป็นต้น เซลจะถูกคิดว่าคายประจุหมดสิ้น เมื่อแรงดันของมัน 1 โวลท์ ค่าความจุของเซลจะ

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เพิ่มขึ้นเล็กน้อยถ้าอัตราการกระแสมีค่าลดต่ำลง และ ค่าความจุที่ลดลงจุเป็นผลมาจากอัตราการจ่ายกระแสสูงขึ้นความจุของเซลแบบกระดุม เช่น เดียวกับแบบทรงกระบอก จะแสดงอยู่ในรูปของ "ความจุ 10 ชั่วโมง" และด้วยอัตราการจ่าย กระแสเป็นเวลา 10 ชั่วโมง (c/10) จะทำให้แรงดันของลดลงเหลือ 1.1 โวลท์ ที่จุดนี้จะถูกพิจารณาว่า หมดประจุอย่างสมบูรณ์ เซลแบบนิแคดนี้จะไม่เหมือนกับ เซลแบบตะกั่ว-กรด (ถึงแม้จะเป็นเซลทุติยภูมิเหมือนกัน) ตรงที่เซลแบบนิแคดนี้จะต้องไม่ประจุไฟให้มันด้วย แหล่งจ่ายไฟที่มีแรงดัน คงที่เพราะว่าค่าความต้านทายภายใน ของมันมีค่าต่ำมากจะทำให้ กระแสที่ใช้ในการ ประจุมีค่ามากเกินไป ซึ่งจะทำความเสียหายให้แก่เซลได้ เราจะใช้แหล่งจ่ายไฟแบบ กระแสออกมาโดยกำหนดค่าได้ และ อยู่ในอัตราที่ปลอดภัยซึ่งแหล่งจ่ายไฟที่มีแรงดันคงที่จะจ่ายกระแสออกมาประจุเซลแบบนิแคด โดยผ่านตัวต้านทาน วิธีง่าย ๆในการสร้างวงจรประจุจะแสดงที่เพื่อใช้ในการประจุเซลนิแคด จำกัดกระแส ค่าของตัวต้านทานจะถูกเลือกให้ค่ากระแสที่ไปประจุเซลจะไม่เกินค่าปลอดภัย สำหรับเซลขนาดนั้นโดยวงจรนี้เราจะต้องใช้อย่างระมัดระวัง วิธีการประจุเซลโดยใช้วงจรนี้ สามารถประจุเซลได้ 3 วิธีใหญ่ ๆ คือ

1.การประจุทีละน้อย TRICKLE RECHARGE

ถ้วกระแสในวงจรถูกรักษาไว้ที่อัตราเท่ากับ C/10 (10% ของความจุ) แล้ว เซลที่จะหมดประจุอย่างสมบูรณ์ สามารถจะประจุได้ภายใน 10 ชั่วโมง แต่ความเป็นจริงจะใช้เวลามากกว่า 10 ชั่วโมง โดยเผื่อการสูญเสียไว้บ้าง จะใช้เวลาประจุ ทีละน้อยด้วยอัตราขนาดนี้สามารถประจุทิ้งไว้ค้างคืนได้ ประโยชน์อีกข้อหนึ่ง การที่แรงดันของ เซลนิแคดแปรเปลี่ยนไปตามเวลาเมื่อทำการประจุ (หลังจากการประจุอย่างสมบูรณ์แล้ว) ด้วยอัตรากระแส C/4 การประจุเซลด้วยอัตราขนาดนี้คือ ถึงแม้ว่าเซลจะถูกประจุเต็มแล้วก็ตาม ก็ไม่จำเป็นต้องนำเซลออก เนื่องจากถ้าเราประจุต่อไปก็จะไม่ ทำความเสียหายให้แก่เซลเนื่องจากก๊าซออกซิเจน ที่เกิดขึ้นทั้งหมดที่ขั้วบวกจะรวมตัวกับขั้วลบ การประจุเซลโดยวิธีนี้เป็นวิธีที่สามารถจะประจุโดยไม่มีข้อจำกัด ซึ่งจะไม่ทำความเสียหายแก่เซล ยกตัวอย่างเช่น เซลมีขนาดความจุ 500 มิลลิแอมป์-ชั่วโมง ถ้าประจุด้วยอัตราC/10 ก็เท่ากับ 10% ของความจุ คือ 50 มิลลิแอมป์

2. การประจุอย่างเร็ว (Fast Recharge)

เซลแบบนิแคดนี้สามารถประจุด้วยอัตราที่สูงกว่าได้ เช่นด้วยอัตรา C/3 (33% ของความจุ) ถึง C/5 (20% ของความจุ) โดยจะต้องเตรียมการตัดการประจุ เมื่อเซลได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น รวมทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รับการประจุเต็มที่แล้ว ซึ่งสามารถทำได้อย่างอัตโนมัติโดยใช้วงจรตรวจจับแรงดัน ซึ่งจะตัดกระแสที่ใช้ในการประจุออก เมื่อแรงดันของเซลเพิ่มขึ้น เกินกว่าค่าปัจจุบัน การแปรเปลี่ยนของแรงดันของเซล กับเวลาที่อัตราการประจุเท่ากับ C/4 (25% ของความจุ) จะเห็นได้ชัดว่าวิธีการนี้สามารถใช้ได้เฉพาะ ถ้าสามารถวัดค่าแรงดันได้อย่าง เที่ยงตรง และ ว่องไว สามารถตัดกระแสที่ใช้ประจุ ออกก่อนที่จะเกิดความเสียหายขึ้นปัญหาในการใช้การประจุแบบนี้ก็ คือ ถ้ากระแสที่ใช้ในการประจุค่าสูง ๆ นี้ไม่ได้ถูกตัดออกอย่างทันทีเมื่อเซลได้รับ ประจุเต็มที่แล้ว ก๊าซออกซิเจนที่เกิดขึ้นมากจากขั้วบวกนี้จะไม่สามารถไปรวมกับที่ขั้วลบ ในปริมาณที่เพียงพอ ความดันจะเพิ่มขึ้นอย่างรวดเร็ว และ เซลจะระบาย ก๊าซออกซิเจนออกไปโดยที่รูระบายที่ปิดไว้ จะเปิดออก และ ปล่อยก๊าซออกซิเจนกับอิเลคโทไลน์สูญเสีย ออกมาจากเซลแล้ว ก็ไม่สามารถเติมกลับเข้าไปใหม่ได้ ดังนั้น ความจุเซลจะลดลงอย่างถาวร ก็คือเซลนั้นจะมีความจุน้อยลง ตลอดไป

3.การประจุอย่างเร่งด่วน (Super-Fast Recharging)

มีบางกรณีที่ผู้ใช้ต้องการที่จะประจุเซลภายในเวลาเพียง2-3 นาที ยกตัวอย่างเช่นเครื่องบินเล็กที่ใช้แบตเตอรี่เป็นตัวจ่ายกำลังจะต้องการประจุเซลที่หมดประจุเพื่อที่จะนำเครื่องบินนี้บินขึ้นสู่อากาศอีกครั้งโดยเร็วที่สุด เท่าที่จะทำได้มันเป็นไปได้ที่จะประจุเซลอย่างเร่งด่วนด้วยอัตรา การประจุสูงถึง 4c (4เท่าของความจุ) หรือมากกว่านี้โดยวิธการต่อไปนี้คือวัดแรงกดดันของเซล และตัดกระแสที่ใช้ประจุออกเมื่อแรงดันของเซลขึ้นสูงถึงค่าที่ตั้งไว้ อย่างไรก็ตามมีวิธีการมแรกง่ายกว่า แล้วก็เที่ยงตรงด้วยโดยจาก หลักความจริง ที่ว่าเซลได้หมดประจุอย่างสมบูรณ์ ก่อนที่จะพยายาม ทำการประจุมันใหม่ให้ประจุไฟเข้า โดยกำหนดค่ากระแสประจุดงที่ไว้ใช้เวลาในการประจุ ตามที่ ต้องการเช่นหลังจากเซลหมดประจุแล้ว กระแสที่ใช้ในการประจุขนาด 3C (3เท่าของความจุ) จะถูกป้อนเป็นเวลา 20 นาที หรือ จะใช้กระแสในการประจุเป็น 5C (5เท่าของความจุ) ป้อนเข้าไปเป็นเวลา 12 นาที เป็นต้น แม้ว่าวิธีการนี้จะเป็น วิธีการที่ดี เช่น สำหรับนักเล่นเครื่องบินจำลอง ที่มีเพียงแหล่งจ่ายไฟ เป็นเพียงแบตเตอรี่รถยนต์ก็ตามก็เป็นสิ่งที่ควรระวังไว้ เนื่องจากการประจุมากเกินไปเพียง 2-3 วินาทีอาจจะทำให้เกิดการรั่วของเซลได้ กล่าวย่อ ๆ ก็คือเมื่อจะใช้วิธีการนี้เซลจะต้องหมดประจุอย่างเต็มที่และใช้กระแสในการประจุที่แน่นอน เป็นระยะเวลาที่ถูกต้อง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.4.4 เซลแบบกระดุม

เนื่องจากเซลแบบกระดุมนี้ รั่วไม่ได้จึงสำคัญมากที่จะต้อง ไม่ประจุมากเกินไปด้วยวิธีใดก็ตาม มิฉะนั้นเซลจะเกิดการระเบิดขึ้นได้ วิธีที่ดีก็คือ จำกัดค่าของกระแสในการประจุให้มีค่าต่ำไว้เช่นด้วยอัตรา C/30 (3.33% ของความจุ) และใช้วลาในการประจุประมาณ 1 วันครึ่ง แม้ว่าที่อัตรา การประจุขนาดนี้เราสามารถจะประจุได้เป็นเวลานานเท่าใดก็ตาม อย่างไรก็ ตามเซลแบบนิแคด นี้จะสามารถประจุไฟได้เป็นเวลานานแม้ว่าจะใช้งาน โดยการคลายประจุอย่างเร็วแล้วประจุด้วย กระแสจำนวนมากๆ มันก็สามารถทำงานได้อย่างมีประสิทิภาพโดยสามารถประจุได้เป็นร้อย ๆ ครั้ง แต่ถ้าใช้วิธีประจุทีละน้อยเราก็สามารถที่จะประจุได้เป็นพัน ๆ ครั้งทีเดียว เรามาลองเปรียบเทียบ ระหว่างเซล แบบนิแคดกับเซลปฐมภูมิเช่น อัลคาไลน์ แมงกานีสดู แต่เนื่องจากไม่สามารถจะเปรียบเทียบได้โดยตรงเนื่องจากขึ้นอยู่กับตัวแปรหลายอย่างเช่นจำนวนเซลที่ใช้งานราคาของ เครื่องประจุ (charge) ขนาด และ ราคาของเซลและการนำไปใช้งานอย่างไรก็ตามเราสามารถที่จะ เปรียบเทียบโดยไม่มีหลักเกณฑ์ที่แน่นอนได้เช่นกัน ยกตัวอย่างเช่น เรามีเครื่องใช้ไฟฟ้าชิ้นหนึ่งเป็นวิทยุเทปซึ่งใช้เซล ขนาด AAโดยใช้งานสัปดาห์ละ 10 ชั่วโมงที่กระแส 200 มิลลิแอมป์และแรงดันแหล่งจ่ายไฟประมาณ 2 ถึง 3 โวลท์ เราสามารถใช้เซลสองเซล ต่ออนุกรมกันและ จ่ายกระแสออกมา 200 มิลลิแอมป์ถ้าเราประมาณราคาของเครื่องประจุเซลแบบนิแคดเท่ากับ 300 บาท (อาจถูกกว่า) เซลแบบนิแคดขนาด AA ราคา 30บาท และเซลแบบอัลคาไลน์ แมงกานีส ขนาด AA ราคา 12 บาทเราสามารถที่จะเปรียบเทียบราคาค่าใช้จ่ายได้ดังนี้เซลแบบอัลคาไลน์ แมงกานีส 2 เซล (ความจุ 1,800 มิลลิแอมป์-ชั่วโมง ส่วนเซลแบบนิแคด 2 เซล (ความจุ 500 มิลลิแอมป์-ชั่งโมง) จะสามารถจ่ายกระแสขนาดนั้นได้เป็นเวลา 2.5 ชั่วโมง ดังนั้นจึงใช้เซลแบบอัลคาไลน์ แมงกานีส จำนวน 1.11 ชุดทุกสัปดาห์เป็นเงิน 27 บาท ต่อสัปดาห์ เช่นเดียวกับ เซลแบบนิแคดจะต้อง ประจุไฟใหม่ 4 ครั้ง (โดยใช้ค่าใช้จ่าย 7 สตางค์ต่อการประจุ 1 ครั้ง) เป็นเงิน 28 สตางค์ต่อสัปดาห์ เหตุนี้ขึ้นอยู่กับเงื่อนไขที่ว่าเครื่องวิทยุเทปนั้น ถูกใช้บ่อยแค่ไหนจุดคุ้มทุน ก็จะเกิดเร็วเท่านั้นถ้าจำนวนเซลที่ใช้มากยิ่งขึ้น และ ใช้เครื่องประจุเครื่องเดียวจุดคุ้มทุนก็จะเกิดเร็วยิ่งขึ้นซึ่งตามความเป็นจริง แล้วถ้าท่านใช้จำนวนเซล บ่อยครั้งเท่าไร ท่านก็ควรจะใช้เซลแบบนิแคดมากขึ้นเท่านั้น แม้ว่าการลงทุนในระยะแรกจะมีค่าสูงแต่จุดคุ้มทุน ก็จะเกิด ขึ้นเร็วเท่านั้นซึ่งนี่ก็เป็นเหตุผลที่ว่าทำไมผู้ผลิตเซลถึงไม่ค่อยส่งเสริมในผลิตภัณฑ์ เซลแบบนิแคดของตนเท่าไรเนื่องจากจะไปทำลายธุรกิจอันใหญ่โต และ ได้กำไรงามของเซลปฐมภูมิที่ตนผลิตอยู่เมื่อเราทราบทฤษฎี ของการชาร์จแบตเตอรี่ ที่ถูกต้องแล้ว ก็สามารถออกแบบวงจรให้ถูกต้อง ได้ดัง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.4.5 หลักการทำงานของวงจร

จากวงจร จะเป็นวงจรการใช้ Voltage มา Control กระแสที่ต้องการจะชาร์จ Batter ชนิดต่าง ๆ อย่างเหมาะสม โดย Voltage นี้จะมาจาก Power Supplyโดยจะเลือกค่า Voltage ที่เหมาะสมกับกระแสที่ต้องการและใช้วงจร Voltmeter มาตรวจสอบการชาร์จแบตเตอรี่ ว่าเต็มหรือยัง ถ้ามีการชาร์จเต็มตาม Voltage ที่ต้องการแล้วก็จะสั่งทาง ซอร์ฟแวร์ ให้ Power Supply จ่ายแรงดัน 0 V.มาที่ Voltage Control จะทำให้ Transister Q1 และ Q2 ซึ่งต่อแบบดาร์ลิงตันนั้น หยุดน่ากระแส จะไม่มีกระแสไหล IC จึงทำให้หยุดการชาร์จแบตเตอรี่ แบบอัตโนมัติ กระแสที่ใช้ในการชาร์จแบตเตอรี่นั้นจะใช้ตามชนิดของแบตเตอรี่ ดังนี้

PP3 -------> 10 mA.

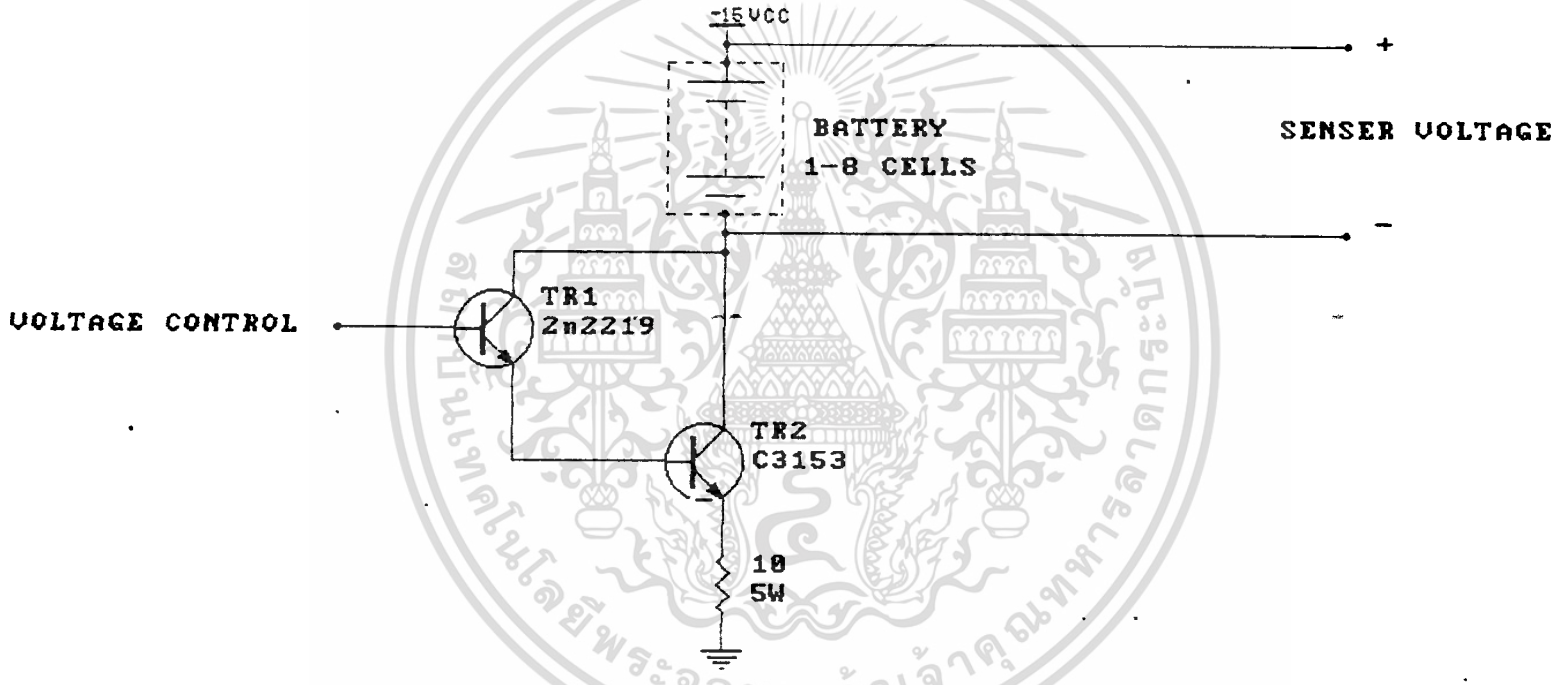
AAA -------> 30 mA.

AA -------> 50 mA.

C -------> 120 mA.

ในส่วนของจอภาพของ PC นั้นจะมีการแสดงสถานะของการชาร์จ คือ ชนิดของแบตเตอรี่ , กระแส ที่ใช้ในการชาร์จแบตเตอรี่ และ เวลาที่ใช้ในการชาร์จ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



CHARGE BATTERY CIRCUIT

บทที่ 4
ภาษาคอมพิวเตอร์

ก่อนที่จะศึกษา การเขียนโปรแกรม เราควรจะทำความเข้าใจกับภาษาคอมพิวเตอร์เสียก่อน ภาษาศอมพิวเตอร์ หมายถึงภาษาที่มนุษย์ใช้สั่งงานเพื่อให้เครื่อง คอมพิวเตอร์สามารถเข้าใจ และ ปฏิบัติงานให้ได้ แบ่งออกได้เป็น 3 ชนิดใหญ่ ๆ คือ

4.1 ภาษาเครื่อง (Machine Language)

4.2 ภาษาแอซเซมบลี (assembly Language)

4.3 ภาษาระดับสูง (Higher Level Language)

4.1 ภาษาเครื่อง

ในสมัยเริ่มแรก ที่ประดิษฐ์เครื่องคอมพิวเตอร์ได้นั้น ภาษาคอมพิวเตอร์ที่สามารถ สั่งให้เครื่องคอมพิวเตอร์สามารถนำไปตีความปฏิบัติงานได้โดยตรง ดังนั้น การเขียนโปรแกรมภาษาเครื่อง ผู้เขียนจะต้องเข้าใจ ระบบการทำงานของเครื่อง โดยชัดแจ้ง และ ต้องสั่งงานอย่างละเอียดทุกขั้นตอน ดังนั้น จึงเป็นงานที่ยุ่งยาก และ น่าเบื่อหน่าย ภาษาเครื่องนี้เครื่องคอมพิวเตอร์รับรู้ได้เลยไม่ต้องผ่านตัวแปล แต่เมื่อเขียนไปแล้ว เราจะทำการแก้ไข และ ตรวจสอบ ยุ่งยากมาก เพราะ ภาษาที่ใช้เขียนนั้น เป็นระบบเลขฐาน 2 ที่มีตัวเลขอยู่ 2 ตัวคือ 0 และ 1 รหัสเลขฐาน 2 ของคำสั่ง แต่ละตัวนี้จะต้องออกแบบ ไว้ก่อน และ จะแตกต่างกันไป ตามชนิดของคอมพิวเตอร์ การนำคำสั่งมาเรียงกันนั้น ขึ้นกับความสามารถในการทำงานของเครื่อง หากการทำงาน ชนิดนั้นๆ ในเครื่องไม่มีก็ไม่สามารถทำได้ ลักษณะการทำงานอีกอย่างหนึ่ง คือ จะคำนวณ ได้ ครั้งละ 2 ตัวเท่านั้น ตัวอย่างเช่น จะคำนวณ Y = A + B -(C + D) จะเขียน เป็นคำสั่งเดียวไม่ได้ ต้องเขียนคำสั่งให้ (C + D)ก่อน แล้วเอาค่าที่ได้ ลบ ออกจาก B จากนั้น จึงนำเอาค่าที่เหลือ มาบวกกับ A แต่ก่อนที่จะเขียนคำสั่งนี้ ต้องเขียนรหัสให้เครื่องทำการ บวก และ ลบ เข้าไปเก็บในเครื่องก่อน เมื่อจะทำงานนั้น เครื่องจะนำเอารหัสนี้ออกมา เพื่อ ทำการถอดรหัสโปรแกรมภาษาเครื่องอาจเป็นดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

คำสั่ง	รหัสภาษาเครื่อง	ความหมาย
1	001011010100	S = C + D
2	011001010101	R = B - S
3	001010101110	Y = A + R
4	010000000000	หยุดทำงาน

4.2 ภาษาแอซเซมบลี

ต่อมา เพื่อละความยุ่งยากของ ภาษาเครื่อง จึงได้มีการปรับปรุงและพัฒนาภาษาของเครื่องคอมพิวเตอร์ เพื่อให้สามารถเขียนโปรแกรม ได้สะดวกขึ้น ในขั้นแรกที่ง่ายที่สุด คือ การใช้ตัวอักษร เขียนแทนการทำงาน หรือ คำสั่งของภาษาเครื่อง แต่ลักษณะ ยังใกล้เคียงกับ ภาษาเครื่อง เราเรียกภาษานี้ว่า ภาษาแอซเซมบลี จากตัวอย่างเดิม

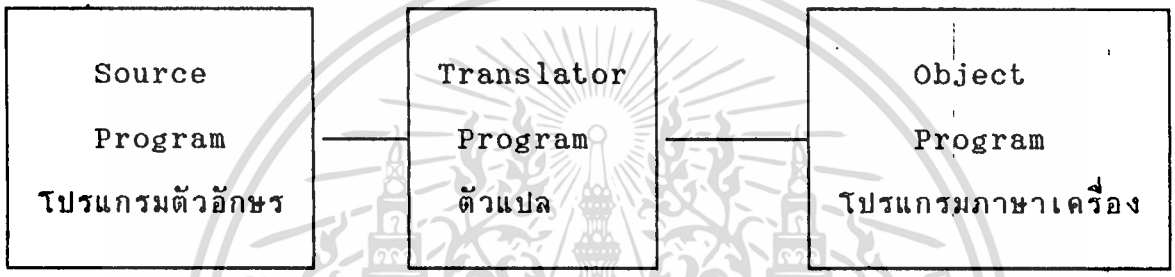
Y = A + B - (D + D)

เขียนเป็นภาษาแอซเซมบลีได้ดังนี้

คำสั่ง	คำสั่งแบบตัวอักษร	ความหมาย
1	S ADD C, D	S = C + D
2	R SUB B,S	R = B - S
3	Y ADD A, R	Y = A + R
4	HLT	หยุดทำงาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของลิขสิทธิ์ทุกครั้งที่มีการนำไปใช้

การเขียนโปรแกรมในรูปของตัวอักษรนี้ เมื่อป้อนเข้าเครื่องคอมพิวเตอร์แล้วจะมีการเปลี่ยนโปรแกรมจากภาษาตัวอักษร ให้เป็นภาษาเครื่องก่อน เพราะ ตัวคอมพิวเตอร์เองนั้นจะสามารถทำงานได้ ก็ต่อเมื่อ โปรแกรมนั้นอยู่ในรูปของเลขฐาน 2 เท่านั้น ปกติจะมีโปรแกรมสำหรับการเปลี่ยน ที่เรียกว่า ทรานสเลเตอร์โปรแกรม (Translator Program) ทำหน้าที่คล้ายล่ามแปลโปรแกรม ที่เขียนด้วยตัวอักษรที่ เรียกว่า ซอร์สโปรแกรม(Source Program) ให้เป็นโปรแกรมภาษาเครื่อง ซึ่งเรียกว่า ออบเจคท์โปรแกรม (Object Program) ดังแสดงในภาพ



4.3 ภาษาระดับสูง

เทคนิคการเขียนโปรแกรมได้ถูกปรับปรุงขึ้นเรื่อย ๆ เพื่อให้ผู้เขียนโปรแกรมสามารถเขียนโปรแกรมให้ใกล้เคียงกับ ภาษามนุษย์ และ ไม่ต้องรู้โครงสร้างของคอมพิวเตอร์เพียงแต่รู้กฎต่าง ๆ ของภาษา เท่านั้น ก็สามารถเขียนโปรแกรมได้ จึงได้มีการพัฒนาภาษาใหม่ขึ้นมา คือ ภาษาระดับสูง เพื่อให้มีความสะดวกต่อผู้ใช้ในชีวิตประจำวัน มากที่สุด การเขียนโปรแกรมโดยใช้ภาษานี้ ไม่ขึ้นกับ ลักษณะการทำงานภายในของเครื่อง ผู้ใช้จึงไม่ต้องเรียนรู้ถึงการทำงานของเครื่อง โปรแกรมในภาษาระดับสูง เวลาป้อนให้เครื่องคอมพิวเตอร์ จะต้องเปลี่ยนเป็นภาษาเครื่องก่อน คำสั่งที่เขียนโดยภาษานี้ หนึ่งคำสั่ง อาจได้รับการแปลให้เป็นภาษาเครื่องจักรมากกว่าหนึ่งคำสั่ง

โปรแกรมที่ทำหน้าที่แปลนี้ เรียกว่า **คอมไพเล่อร์ (Compiler)**

โปรแกรมภาษาระดับสูง ตัวแปล โปรแกรมภาษาเครื่อง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากตัวอย่างเดิม Y = A + B - (C + D) ถ้าใช้ภาษาระดับภาษาเครื่องสามารถเขียนสมการนี้เข้าไปในโปรแกรมได้เลยไม่ต้องแยกเป็นการคำนวณทีละ 2 ตัวแล้วคอมไพเลอร์ และ แปลคำสั่งนี้ออกเป็นชุดของคำสั่งภาษาเครื่องเอง จะเห็นว่า ภาษาระดับสูง เป็นภาษาที่ผู้ใช้สามารถใช้ได้โดยสะดวกไม่ต้องเรียนรู้ถึงเครื่องเพียงศึกษาถึง กฎเกณฑ์ที่กำหนดไว้ และ เขียนตามกฎ ก็จะสามารถ ได้โปรแกรมการทำงาน ที่ต้องการภาษาระดับสูง นี้ยังแบ่งออกเป็น 2 ประเภทคือ

4.3.1 Procedure Orlented Language

เป็นภาษาที่มีลักษณะใกล้เคียงกับงานที่จะทำเช่นงานทางด้านวิทยาศาสตร์ คณิตศาสตร์ ธุรกิจ เป็นต้น ภาษาประเภทนี้มีหลายภาษา การที่เราจะเขียนโปรแกรมด้วยภาษาใดนั้นย่อมขึ้นอยู่กับลักษณะงานที่จะทำ เพราะภาษาแต่ละภาษามีความเหมาะสมกับงานแตกต่าง ภาษาที่รู้จักกันดีได้แก่

1. ภาษาฟอร์แทรน (FORTRAN ย่อมาจาก FORmula TRANslation) เป็นภาษาที่เหมาะกับงานทางด้านวิทยาศาสตร์ วิศวกรรมศาสตร์ และคณิตศาสตร์ เป็นภาษาแรกที่มีผู้ยอมรับ และ ใช้กันแพร่หลาย ทั้งยังเป็นภาษาที่ง่ายต่อการศึกษาด้วย

2. ภาษาโคบอล(COBOL ย่อมาจาก Commom Business Oriented Language) เป็นภาษาที่สร้างขึ้นสำหรับงานทางด้านธุรกิจ

3. ภาษาอาร์พีจี (RPG ย่อมาจาก Report Programming Generator) เป็นภาษาที่เหมาะกับงานพิมพ์รายงานมาก ๆ กำลังเป็นที่นิยมใช้ทั่วไปในวงการธุรกิจแถบเอเชีย

4. ภาษาเบสิค (BASIC ย่อมาจาก Beginner's All purpose Symbolic Instruction Code) เป็นภาษา ที่สร้างขึ้นเพื่อ ให้มีลักษณะที่สามารถศึกษาได้ง่าย และใช้งานได้ง่าย

5. ภาษาแอลกอล (ALGOL ย่อมาจาก ALGOrithmic Language) เป็นภาษาที่เหมาะกับงานด้านวิทยาศาสตร์ที่มีตัวเลขค่าสูง ๆและเขียนง่ายกว่าภาษาฟอร์แทรนเป็นภาษาที่ใช้กันแพร่หลายในทวีปยุโรป

6. ภาษาพีแอลวัน (PL/I ย่อมาจาก Programming Language I) เป็นภาษาที่สร้างขึ้นเพื่อ ให้สามารถประยุกต์กับงานทางด้านวิทยาศาสตร์และธุรกิจ เป็นภาษาที่สร้างขึ้นมาทีหลัง มีประโยชน์มาก แต่ยังไม่เป็นที่ยอมรับกัน และยังยากต่อการศึกษาด้วย

ลิขสิทธิ์เป็นของจุฬาลงกรณ์มหาวิทยาลัย ห้ามนำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.3.2 Problem Oriented Language

เป็นภาษาที่สร้างขึ้นเพื่อใช้กับงานเฉพาะอย่าง ขอบข่ายของการประยุกต์อยู่ในวงที่แคบกว่า ตัวอย่างภาษานี้ ได้แก่ จีพีเอสเอส (GPS ย่อมาจาก General Purpose Simulation System) เป็นภาษาที่ใช้กับ งานด้านการจำลองแบบ การใช้ภาษา ชนิดนี้เสมือนกับการเขียนโปรแกรม ไว้ล่วงหน้า ผู้เขียนเพียงแต่กำหนดค่าข้อมูล ให้เครื่อง ก็จะสามารถทำงานให้ได้ การพัฒนาภาษาคอมพิวเตอร์ยังมีต่อไป ทั้งในด้านการคิดค้นภาษาใหม่ขึ้นมา และ การปรับปรุงภาษาที่มีอยู่แล้วให้มีประสิทธิภาพขึ้นจนในปัจจุบันมีภาษาระดับสูงมากมายหลายชนิด แต่สำหรับผู้ที่ เริ่มศึกษาคอมพิวเตอร์ ภาษาที่เหมาะสำหรับศึกษาในขั้นนี้ คือ ภาษาเบสิค

4.4 ภาษาเบสิค (BASIC)

คำว่า BASIC ย่อมาจาก Beginner's Allpurpose Symbolic Instruction Code โดยการนำเอาอักษรตัวแรกของแต่ละคำมาเขียนเรียงกัน ภาษาเบสิคเป็นภาษาที่พัฒนาขึ้นมาโดย ศาสตราจารย์เคเมนีย์ (Kemeny) และ ศาสตราจารย์ เคอทซ์ (Kuerz) แห่งมหาวิทยาลัยดาร์ทแมทซ์ (Dartmatz) แห่งประเทศสหรัฐอเมริกาใน กลางปี 1960 ภายหลังจากที่มีการพัฒนาภาษาเบสิคขึ้นมาแล้ว ได้มีการนำเอาไปใช้งาน ต่าง ๆ ในวงการคอมพิวเตอร์ จนกระทั่งในระยะตั้งแต่ปี 1975เป็นต้นมาได้มีการพัฒนาและสร้างคอมพิวเตอร์ขนาดเล็ก ที่เรียกว่า ไมโครคอมพิวเตอร์ (Microcomputer) ขึ้นมา ทำให้ราคา ของคอมพิวเตอร์ลดลง และ มีการนำไปใช้งานอย่างกว้างขวาง ภาษาหลักที่ใช้ ในไมโครคอมพิวเตอร์คือ ภาษาเบสิค ทำให้มีการใช้ภาษาเบสิคกันอย่างกว้างขวางยิ่งขึ้น

ปัจจุบันนี้ในประเทศผู้ผลิตคอมพิวเตอร์ เช่น สหรัฐอเมริกา หรือญี่ปุ่น ภาษาเบสิคเป็นภาษาที่มีผู้สนใจศึกษา ตั้งแต่นักเรียน ชั้นประถม ไปจนถึงผู้ชำนาญในสาขาต่าง ๆ แม้แต่ในประเทศไทยเองปัจจุบันนี้ไมโครคอมพิวเตอร์กำลังได้รับความนิยมแพร่หลายมากขึ้น เรื่อย ๆ ภาษาเบสิคได้รับการบรรจุเข้าในหลักสูตร เกี่ยวกับการบริหารธุรกิจ ของสถาบันการศึกษาหลายแห่ง และ มีการเปิดหลักสูตรอบรมภาษาเบสิคระยะสั้นกันหลายสถาบัน ตั้งแต่ได้มีการพัฒนาภาษาเบสิคขึ้นมาก็ได้มีผู้เชี่ยวชาญด้านคอมพิวเตอร์จำนวนมากได้พยายาม ศึกษา และพัฒนาเกี่ยวกับภาษาเบสิคเพิ่มขึ้น ทำให้ได้ภาษาเบสิคที่ต่างกันออกไปเป็นจำนวนมาก ภาษาเบสิคต่างชนิดกัน อาจมีการใช้คำ ที่แตกต่างกันบ้าง แต่หลักการใหญ่ ๆ แล้วไม่แตกต่างกัน ภาษาเบสิคที่จะกล่าวต่อไปนี้ เป็นภาษาเบสิคที่ใช้ขอบเขตของไมโครคอมพิวเตอร์เท่านั้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.4.1 ข้อได้เปรียบของภาษาเบสิค

ภาษาเบสิคเป็นภาษาที่มีลักษณะต่างไปจากภาษาอื่น เพราะ ลักษณะของภาษาเป็นแบบภาษาพูด ภาษาเบสิคที่เราใช้ติดต่อกับคอมพิวเตอร์ในลักษณะถามตอบทีละประโยค (ทีละบรรทัดของโปรแกรม) เมื่อมีการผิดพลาดก็จะแจ้งให้ทราบทันที ต่างจากภาษาฟอร์แทรนหรือภาษาอื่น ๆ ที่จะรวบรวมข้อผิดพลาดไว้บอกทีเดียวเมื่อจบโปรแกรม ถ้าจะเปรียบเทียบภาษาเบสิคเหมือนโทรศัพท์ที่สามารถโต้ตอบกัน ได้ครั้งละประโยคภาษาอื่น ๆ ก็เปรียบเทียบเหมือนการเขียนจดหมายที่ต้องการรอตอบกันทั้งฉบับ

สาเหตุที่เราสามารถใช้ภาษาเบสิคในการสนทนากับคอมพิวเตอร์ได้ก็เพราะว่า มีการแปลเป็นภาษาเครื่องที่คอมพิวเตอร์เข้าใจได้ในทันทีทันใด ซึ่งลักษณะเช่นนี้ เรียกว่าเป็นอินเตอร์พริทเตอร์ (Interpreter) หรือ ตัวแปล ทำให้สามารถตรวจหาข้อผิดพลาดได้ง่าย รวดเร็ว และ สะดวก ส่วนภาษาอื่น ๆ นั้นเครื่องจะตรวจ โปรแกรม นั้นจนหมดเสียก่อน ทำให้สามารถติดต่อในลักษณะ การสนทนากัน ลักษณะเช่นนี้ เรียกว่าเป็น คอมไพเลอร์ (Compiler) หรือ ตัวรวบรวม

4.4.2 คำสั่ง

คำสั่งในภาษาเบสิค แบ่งออกเป็น 2 ประเภทคือ

1. คำสั่งตรง (Direct Commands) เป็นคำสั่งที่สามารถปฏิบัติงานทันทีเมื่อมีข้อมูลพร้อมการ ใช้คำสั่งนี้ไม่ต้องมีหมายเลขบรรทัดกำกับ และใช้ได้ทีละคำสั่งเท่านั้น
2. คำสั่งตามขั้นตอน (Statements) เป็นคำสั่งที่ต้องการให้ทำงานต่อเนื่องกัน ต้องมีหมายเลขบรรทัดกำกับทุกครั้ง

4.4.2.1 คำสั่งตรง

1. คำสั่ง RUN

เป็นคำสั่งให้เครื่องทำการประมวลผลตามโปรแกรมที่อยู่ในหน่วยความจำ

รูปแบบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
RUN [หมายเลขบรรทัด]
```

2. คำสั่ง DELETE

เป็นคำสั่งที่ใช้ลบหรือตัดบรรทัดในโปรแกรมทิ้งไป

รูปแบบ

DELETE [หมายเลขบรรทัด 1] {,} [หมายเลขบรรทัด 2]

หมายเหตุ แต่ละกรณีแยกจากกัน

3. คำสั่ง NEW

เป็นคำสั่งลบโปรแกรมในหน่วยความจำออกให้หมด

รูปแบบ

NEW

4. คำสั่ง LIST

เป็นคำสั่งให้แสดงโปรแกรมที่เก็บอยู่ในหน่วยความจำออกมาดู

รูปแบบ

LIST [หมายเลขบรรทัด 1] [{,} [หมายเลขบรรทัด 2]]

5. คำสั่ง LIST "LPTO:"

เป็นคำสั่งให้แสดงโปรแกรมที่เก็บอยู่ในหน่วยความจำออกมาทางพริ้นเตอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปแบบ

```
LIST "LPTO:"[,[หมายเลขบรรทัด 1] [{,} [หมายเลขบรรทัด 2]]]
```

6. คำสั่ง HARDC

เป็นคำสั่งให้แสดงข้อมูลที่ปรากฏบนจอในขณะนั้นทั้งหมด ออกมาทางพริ้นเตอร์

รูปแบบ

```
HARDE [{0/1/2}]
```

7. คำสั่ง STOP

เป็นคำสั่งให้หยุดทำการประมวลผลชั่วคราว

รูปแบบ

```
STOP
```

8. คำสั่ง CONT

เป็นคำสั่งให้ทำการประมวลผลต่อจากที่หยุดไว้ด้วยคำสั่ง stop

รูปแบบ

```
CONT
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

9. คำสั่ง AUTO

เป็นคำสั่งที่ใช้กำหนดหมายเลขบรรทัดของแต่ละบรรทัด

รูปแบบ

```
AUTO  [[หมายเลขบรรทัดเริ่มต้น][,ค่าการเพิ่มขึ้น]]
```

10. คำสั่ง RESUM

เป็นคำสั่งเปลี่ยนแปลงหมายเลขบรรทัดในโปรแกรมใหม่

รูปแบบ

```
RENUM [[หมายเลขบรรทัดใหม่][,หมายเลขบรรทัดเก่า][,ค่าการเพิ่มขึ้น]]
```

11. คำสั่ง PRINT

เป็นคำสั่งให้แสดงค่าของข้อความ ให้ปรากฏบนจอ

รูปแบบ

```
PRINT [ข้อความ [{;} [ข้อความ]]----]
```

หมายเหตุ ข้อความในกรณีนี้ต้องประกอบด้วยค่าคงที่เท่านั้น

4.4.2.2 คำสั่งตามขั้นตอน

1. คำสั่ง LET

เป็นคำสั่งใช้กำหนดค่าของข้อความทางขวามือ ให้เป็นค่าของตัวแปรทางซ้ายมือ

เอกสารนี้สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น ขอสงวนห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปแบบ

```
[LET]  ตัวแปร = ข้อความ
```

2. คำสั่ง PRINT

เป็นคำสั่งให้แสดงค่าของข้อความ ให้ปรากฏบนจอ

รูปแบบ

```
PRINT [ข้อความ [{;} [ข้อความ]]===]
```

3. คำสั่ง INPUT

เป็นคำสั่งให้รับข้อมูลเข้ามาทางแป้นพิมพ์ แล้วกำหนดให้เป็นค่าของตัวแปรในคำสั่ง INPUT

รูปแบบ

```
INPUT ["ข้อความนำ" {;}]ตัวแปร [,ตัวแปร]...
  :
  :
  :
 RUN
[ข้อความนำ] ? ข้อมูล [,ข้อมูล]...
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4. คำสั่ง END

เป็นคำสั่งที่บอกให้ทราบว่าจบโปรแกรมแล้ว ให้หยุดทำการประมวลผล

รูปแบบ

```
END
```

5. คำสั่ง READ....DATA

เป็นคำสั่งให้อ่านค่าคงที่ที่กำหนดไว้ในคำสั่ง DATA แล้วกำหนดให้เป็นค่าของตัวแปร

รูปแบบ

```
READ  ตัวแปร [,ตัวแปร]...
  :
  :
  :
DATA  ค่าคงที่ [,ค่าคงที่]...
```

6. คำสั่ง RESTORE

เป็นคำสั่งให้นำค่าคงที่ในคำสั่ง DATA เก่ามาให้อีกครั้ง

รูปแบบ

```
RESTORE [หมายเลขบรรทัด]
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

7. คำสั่ง GO TO

เป็นคำสั่งให้ข้ามไปทำงานยังหมายเลขที่เขียนตามมาโดยไม่มีเงื่อนไข

รูปแบบ

```
GO TO [หมายเลขบรรทัด]
```

8. คำสั่ง ON....GO TO

เป็นคำสั่งให้ข้ามไปทำงานยังหมายเลขที่เขียนตามมาโดยมีเงื่อนไข

รูปแบบ

```
ON ข้อความ GO TO หมายเลขบรรทัด [,หมายเลขบรรทัด]...
```

9. คำสั่ง IF....THEN...ELSE

เป็นคำสั่งให้ทดสอบเงื่อนไข เพื่อเลือกคำสั่งจะให้ทำงานต่อ

รูปแบบ

```
IF ข้อความ THEN { คำสั่ง/หมายเลขบรรทัด }
           [ELSE { คำสั่ง/หมายเลขบรรทัด }]
```

10. คำสั่ง FOR....NEXT

เป็นคำสั่งให้ทำการประมวลผลซ้ำ ๆ กัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปแบบ

```
FOR ตัวแปร = ค่าเริ่มต้น  TO ค่าสุดท้าย
             [STEP ค่าการเพิ่มขึ้น]
:
:
:
NEXT  [ตัวแปร [,ตัวแปร]....]
```

11. คำสั่ง DIM

เป็นคำสั่งสำหรับจองเนื้อที่ให้กับตัวแปรที่มีตัวห้อย

รูปแบบ

```
DIM ตัวแปร (ค่าสูงสุดของตัวห้อย) [,ตัวแปร(ค่าสูงสุดของตัวห้อย]...
```

12. ฟังก์ชั่น TAB (.)

เป็นฟังก์ชั่นสำหรับการแสดงผลออกมา โดยเว้นระยะไปยังตำแหน่งที่เรากำหนดในวงเล็บ แล้วพิมพ์ในตำแหน่งถัดไป ฟังก์ชั่นนี้ใช้ในคำสั่ง PRINT

รูปแบบ

```
TAB (ตำแหน่งก่อนตำแหน่งที่จะให้พิมพ์)
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

13. คำสั่ง REM

เป็นคำสั่งบอกหมายเหตุสำหรับอธิบายแต่ละส่วนของโปรแกรม

รูปแบบ

```
REM ประโยคหมายเหตุ
```

14. คำสั่ง STOP

เป็นคำสั่งให้หยุดทำการประมวลผลชั่วคราว

รูปแบบ

```
STOP
```

15. คำสั่ง OPEN

เป็นคำสั่งให้ทำการเปิดไฟล์สำหรับแสดงผลทางพริ้นเตอร์

รูปแบบ

```
OPEN "O" .[#] หมายเลขไฟล์. "LPTO:(W)
```

16. คำสั่ง CLOSE

เป็นคำสั่งให้ทำการปิดไฟล์ที่เปิดอยู่

รูปแบบ

```
CLOSE [[#] หมายเลขไฟล์]
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
10  REM ****************************
11  REM          MAIN PROGRAM
20  N=0
21  OUT &H307,&H9B
22  OUT &H303,&H80
23  OUT &H300,00
24  OUT &H301,&H00
25  CLS
30  DEFINT A-N
31  DIM F$(5)
32  F$(1) ="BATTERY  CHARGER": F$(2)="POWER  SUPPLY    ":
    F$(3) ="DC.VOLTEMETER   ": F$(4)="OPERATE          ":
    F$(5) ="QUIT            "
34  NCHOICE =5
35  CHOICE =0
40  WHILE CHOICE < 5
50  GOSUB 200
60  LOCATE 10,1
70  ON CHOICE GOSUB 900,1500,2300,2900
80  WEND
90  CLS
100 END


200 REM *****************************
210 REM         MENU
230 SCREEN 0:A=3 : FOR B=1 TO 21 : A=A+1
240 LOCATE A,4:PRINT "CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC"
250 NEXT B:COLOR 7,0
260 C=0 : FOR D=1 TO 3 : C=C+1
270 LOCATE C,4:PRINT "IIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIII"
280 NEXT D:COLOR 10
290 LOCATE 8,29: PRINT" IMMMMMMMMMMMMMMMMM; "
300 LOCATE 9,29: PRINT" :  MENU  SELECT   : "
310 LOCATE 10,29:PRINT" LMMMMMMMMMMMMMMMMM9 "
320 LOCATE 11,29:PRINT" :                 : "
330 LOCATE 12,29:PRINT" :                 : "
340 LOCATE 13,29:PRINT" :                 : "
350 LOCATE 14,29:PRINT" :                 : "
360 LOCATE 15,29:PRINT" :                 : "
370 LOCATE 16,29:PRINT" HMMMMMMMMMMMMMMMMM< "
380 LOCATE 2,22 :COLOR 0,5:PRINT" THIS PROGRAM MADE BY KOH AND CHAT ":COLOR 7,0
430 REM ***********************************************
440 REM     GENERAL PURPOSE MENU CHOICE SUBROUTINE
460 G=10:FOR IC=1 TO NCHOICE
470 G=G+1
480 LOCATE G,31:PRINT F$(IC)
490 NEXT IC
500 LOCATE 22,8
510 COLOR 0,7
520 PRINT"USER ARROW KEYS TO CHOICE SELECTOR;USE<ENTER>KEY TO MAKE CHOICE
530 COLOR 7,0
540 CHOICE = 1
550 WHILE Y$<> CHR$(13)
560 LOCATE CHOICE+10,31
570 COLOR 0,7
580 PRINT F$(CHOICE)
590 COLOR 7,0
600 Y$ = INKEY$:IF Y$="" THEN 600
610 LOCATE CHOICE+10,31
620 PRINT F$(CHOICE)
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
630 IF Y$= CHR$(0) + CHR$(80) THEN GOSUB 690
640 IF Y$= CHR$(0) + CHR$(72) THEN GOSUB 740
650 IF Y$="" THEN GOSUB 650
660 WEND
670 Y$=""
680 RETURN
690 REM ************************
700 REM   DOWN ARROW KEY
710 CHOICE = CHOICE+1
720 IF CHOICE > NCHOICE THEN CHOICE =1
730 RETURN
740 REM ************************
750 REM    UP ARROW KEY
760 CHOICE = CHOICE-1
770 IF CHOICE < 1 THEN CHOICE = NCHOICE
780 RETURN


900 REM ********************************
910 REM    BATTERY CHARGER
920 DIM H(35)
930 GOSUB 1250
940 LOCATE 7,30:PRINT "   SIZE SELECTION  :  ";
950 LOCATE 10,30:PRINT"       1.    PP3       ";
960 LOCATE 12,30:PRINT"       2.    AAA       ";
970 LOCATE 14,30:PRINT"       3.     AA       ";
980 LOCATE 16,30:PRINT"       4.      C       ";
990 LOCATE 20,30:PRINT" (SELECT 1,2,3, OR 4) ";
1000 GOSUB 1210
1010 PRINT H$
1020 ON H GOTO 1030,1060,1090,1115
1030 GOSUB 1250
1040 VO=11
1050 GOTO 1119
1060 GOSUB 1250
1070 VO=12
1080 GOTO 1119
1090 GOSUB 1250
1091 VO=14
1100 GOTO 1119
1115 GOSUB 1250
1116 VO=21
1117 GOTO 1119
1118 GOSUB 1250
1119 LOCATE 12,24:INPUT " PLESE ENTER BATTERY COUNTS. (1-8) ",S
1120 IF S>8 THEN 1119
1121 IF S<1 THEN 1119
1122 OUT &H301,&H84
1123 OUT &H300,VO
1125 GOSUB 1250
1145 LOCATE 6,36:PRINT"  STATUS  "
1146 LOCATE 9,31:PRINT" BATTERY TYPE "
1147 IF H=1 THEN LOCATE 9,45:PRINT "  PP3  "
1149 IF H=2 THEN LOCATE 9,45:PRINT "  AAA  "
1150 IF H=3 THEN LOCATE 9,45:PRINT "   AA  "
1151 IF H=4 THEN LOCATE 9,45:PRINT "   C   "
1152 LOCATE 11,29 : PRINT " CURRENT CHARGE "
1153 IF H=1 THEN LOCATE 11,45:PRINT " 10  mA. "
1154 IF H=2 THEN LOCATE 11,45:PRINT " 30  mA. "
1155 IF H=3 THEN LOCATE 11,45:PRINT " 50  mA. "
1156 IF H=4 THEN LOCATE 11,45:PRINT " 120 mA. "
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
1170 LOCATE 15,29:PRINT"   CHARGE BATTERY NOW... "
1180 LOCATE 20,29:PRINT" (TO STOP,PRESS ANY KEY) "
1181 TIME$="00:00:00"
1182 LOCATE 17,34:PRINT "TIMES   "
1183 REM LOCATE 17,42:PRINT TIME$
1184 PCHOICE=3
1185 GOSUB 2763
1190 REM H$=INKEY$ :IF H$=""THEN 1183 ELSE GOTO 10
1200 REM GOTO 1190
1210 GOSUB 1240
1220 IF H$<>"1"AND H$<>"2"AND H$<>"3"AND H$<>"4" THEN GOTO 1210 ELSE H=VAL(H$):RETURN
1240 H$=INKEY$:IF H$=""THEN 1240 ELSE RETURN
1250 SCREEN 10
1260 COLOR 7,0
1270 WINDOW SCREEN (0,0)-(639,199)
1280 KEY OFF
1290 LINE(0,0)-(639,199),,B
1300 LOCATE 2,65:PRINT"VERSION 1.00"
1310 LINE (89,28)-(549,176),,BF
1320 LINE (85,26)-(554,176),,B
1330 RETURN


1500 REM ************************************
1510 REM          POWER SUPPLY
1520 REM LOCATE 20,65:PRINT INT(P*100+.5)/100
1530 COLOR 0,7
1540 LOCATE 20,12:INPUT " HOW MANY VOLTAGES TO SUPPLY DO YOU WANT (0-25.0)? ",P
1550 COLOR 7,0
1560 IF P<0 THEN GOSUB 1564
1561 IF P>25 THEN GOSUB 1564
1563 RETURN
1564 LOCATE 20,64:PRINT"      ":GOSUB 1530 :RETURN
1565 SCREEN 10
1568 WINDOW SCREEN(80,-10)-(240,130)
1570 LINE (145,30)-(230,65),1,B
1580 LINE (148,33)-(151,62),1,BF
1790 LINE (227,33)-(224,62),1,BF
1800 LINE (155,35)-(180,55),1,B
1810 LINE (155,50)-(180,55),1,BF
1820 LINE (195,35)-(220,55),1,B
1830 LINE (195,50)-(220,55),1,BF
1840 FOR I=1  TO 15
1850 LINE (144+I,30-I)-(230+I/2,30-I),1
1860 LINE-(230+I/2,66-I),1
1870 NEXT I
1880 CIRCLE(163,60),2,1
1890 CIRCLE (163,60),1,1
1900 CIRCLE (172,60),2,1
1910 CIRCLE (172,60),1,1
1920 CIRCLE (167,53),1,3
1930 CIRCLE (207,53),1,3
1940 CIRCLE (187,58),3,1
1950 LINE (202,57)-(213,63),1,B
1960 LINE (204,58)-(211,62),1,B
1970 LINE (207,58)-(211,62),1,BF
1980 LOCATE 13,39:PRINT"+"
1990 LOCATE 13,49:PRINT"-"
2000 LOCATE 13,68:PRINT"ON"
2010 LOCATE 10,51:PRINT INT(P*100+.5)/100
2020 LOCATE 11,52:PRINT"Volts"
2030 CIRCLE (207,54),12,3,49/7,61/7
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่อนุญาตให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
2040 CIRCLE (167,54),12,3,49/7,61/7
2050 FOR J=1 TO 1
2060 LINE (167+J,51-J)-(175+J/2,40-J),2
2070 NEXT J
2080 FOR K=1 TO 1
2090 LINE (207+K,51-K)-(215+K/2,40-K),2
2100 NEXT K
2101 PA1=&H20
2102 OUT &H301,PA1
2103 DP=P*10
2104 OUT &H300,DP
2105 RETURN


2300 REM ***********************************
2310 REM         DC.VOLTEMETER
2320 DIM M$(4)
2330 M$(1) = " 200 mV." : M$(2) = "  2  V. "
     M$(3) = "  20 V. " : M$(4) = " 200 V. "
2340 OCHOICE = 4
2350 PCHOICE = 0 :N=1
2360 WHILE CHOICE < 4
2380 PCHOICE =1
2390 WHILE Y$ <> CHR$(13)
2395 COLOR 0,7:LOCATE 20,24:PRINT" PLEASE SELECT RANGE OF METER ":COLOR 7,0
2400 LOCATE 18,35
2410 COLOR 0,7:PRINT M$(PCHOICE)
2420 COLOR 7,0
2430 Y$=INKEY$
2440 IF Y$="" THEN 2430
2450 LOCATE 18,35
2460 PRINT M$(PCHOICE)
2470 IF Y$=CHR$(0) + CHR$(80) THEN GOSUB  2550
2480 IF Y$=CHR$(0) + CHR$(72) THEN GOSUB  2580
2490 IF Y$="" THEN GOSUB 2490
2500 WEND
2510 Y$=""
2520 ON PCHOICE  GOSUB 2603,2601,2605,2607
2530 WEND
2540 RETURN
2550 PCHOICE = PCHOICE+1
2560 IF PCHOICE > OCHOICE THEN PCHOICE =1
2570 RETURN
2580 PCHOICE = PCHOICE-1
2590 IF PCHOICE<1 THEN PCHOICE = OCHOICE
2600 RETURN
2601 PA2=&H51:Z$="V.":GOTO 25
2602 RETURN
2603 PA2=&H52:Z$="mV.":GOTO 25
2604 RETURN
2605 PA2=&H54:Z$="V." :GOTO 25
2606 RETURN
2607 PA2=&H58:Z$="V.":GOTO 25
2608 RETURN
2610 SCREEN 10
2620 WINDOW SCREEN(80,-10)-(240,130)
2630 LINE(134,5)-(90,65),4,BF
2640 CIRCLE (113,43),10,0
2645 CIRCLE (129,56),2,0
2648 CIRCLE (129,56),1,0
2650 CIRCLE (97,56),2,0
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่อนุญาตให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
2660 CIRCLE (97,56),1,0
2670 LINE (131,8)-(93,23),0,BF
2680 LINE (111,54)-(115,31.5),0,BF
2690 LOCATE 7,14:PRINT M$(PCHOICE)
2700 LOCATE 11,8:PRINT" - "
2710 LOCATE 11,24:PRINT" + "
2711 LOCATE 5,23:PRINT Z$
2730 FOR L=1 TO 7
2740 LINE (89+L,6-L)-(134+L/2,6-L),1
2750 LINE-(134+L/2,67-L),1
2760 NEXT L
2761 PA=PA2+PA1
2762 OUT &H301,PA:PA1=&H00
2763 A=INP(&H304):LOCATE 1,10:REM PRINT A
2764 GOSUB 2779
2765 DG1=A
2766 A=INP(&H305):LOCATE 1,6:REM PRINT A
2767 GOSUB 2779
2768 IF A=I THEN  GOSUB 2797 ELSE GOSUB 2800
2769 A=INP(&H306):LOCATE 1,2:REM PRINT A
2770 GOSUB 2779
2771 IF A=I THEN  DG3=A*100 ELSE GOSUB 2785:DG3=(B+10)*100
2772 DG=DG1+DG2+DG3:IF N=0 THEN LOCATE 17,42:PRINT TIME$
2773 IF PCHOICE=2 THEN  DG=DG: LOCATE 5,15:PRINT DG/1000
2774 IF PCHOICE=1 THEN  DG=DG: LOCATE 5,15:PRINT DG/10
2775 IF PCHOICE=4 THEN  DG=DG: LOCATE 5,15:PRINT DG/10
2776 IF PCHOICE=3 THEN  DG=DG: LOCATE 5,15:PRINT DG/100
2777 S$=INKEY$:IF S$="" THEN 2763 ELSE 2778
2778 GOTO 10
2779 RESTORE:FOR I=9 TO 0 STEP -1
2780 READ D
2781 IF A=D THEN A=I  :GOTO 2784
2782 NEXT
2783 DATA 144,128,248,130,146,153,176,164,249,192
2784 RETURN
2785 IF A=192 THEN B=0
2786 IF A=64  THEN B=0
2787 IF A=121 THEN B=1
2788 IF A=36 THEN B=2
2789 IF A=48 THEN B=3
2790 IF A=25 THEN B=4
2791 IF A=18 THEN B=5
2792 IF A=2 THEN B=6
2793 IF A=120 THEN B=7
2794 IF A=0 THEN B=8
2795 IF A=16 THEN B=9
2796 RETURN
2797 DG2=A*10
2798 LOCATE 5,12:PRINT"+"
2799 RETURN
2800 GOSUB 2785
2801 DG2=B*10
2802 LOCATE 5,12
2803 PRINT "-"
2804 RETURN

2900 REM **********
2910 REM        OPERATE
2930 IF P<>0 THEN GOSUB 1565
2932 IF N=1 THEN GOSUB 2610
2940 CODE.K=0
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่อนุญาตให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
2950 DO UNTIL CODE.K<>0
2960 WHILE INSTAT
2970 CODE.K=ASC(INKEY$)
2980 WEND
2981 LOOP
3000 OUT &H300,00
3001 OUT &H301,&H00
3002 RETURN
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การใช้งานและวิเคราะห์โครงงาน

จากบทที่กล่าวมานั้น เป็นการพูดถึง ทฤษฎีเบื้องต้น และ การทำงานของการใช้ PC ควบคุมวงจร Power Supply ,วงจรชาร์จแบตเตอรี่ และ DC. Voltmeter โดยในบทนี้จะกล่าวถึงการใช้งานของการใช้ PC ควบคุมวงจร Power Supply,วงจรชาร์จแบตเตอรี่ และ DC. Voltmeter โดยเริ่มต้นนั้น เราจะเรียกใช้ File AS.exe แล้วจะปรากฏ MENU มาที่หน้าจอภาพ ของเครื่องไมโครคอมพิวเตอร์ ต่อไป เราจะต้องกำหนดจุดประสงค์ของเราว่าจะใช้ทำอะไร โดยแบ่งได้เป็น 2 กรณี คือ

1) การใช้ PC ไปควบคุม วงจร Power Supply และ DC. Voltmeter กับงานภายนอก โดยเราจะเลือกแถบสว่างโดยใช้แป้นลูกศร ขึ้นหรือลงมาที่ Power Supply แล้วกดแป้น Enter แล้วจิงใส่จำนวน Voltage ที่ต้องการ (0.0-25.0 Vdc.) แล้ว Enter เครื่อง PC ก็จะเก็บค่า Data ที่เราใส่เข้าไปในหน่วยความจำ และ ต่อไปเราก็เลือกแถบสว่างมาที่ DC. Voltmeter แล้ว Enter แล้วเราก็เลือก ย่านการวัดสูงสุด โดยการใช้แป้นพิมพ์ ลูกศร ขึ้น หรือ ลง แล้ว Enter เครื่อง PC ก็จะเก็บค่าตัวแปรของการวัดลงในหน่วยความจำต่อไปเราก็เลือกแถบสว่างมาที่ Operate แล้วกด Enter ชุดของ Control ก็จะทำงาน ตามที่เราใส่ค่าต่าง ๆ ลงไป แล้วจะมีการแสดงค่าต่าง ๆ ออกไปทาง จอภาพ แล้วถ้าเรา ต้องการจะหยุดการทำงาน หรือ กลับไป MENU เราก็ทำได้โดยกดแป้น Enter อีกที

2) การใช้ PC ไปควบคุม วงจรชาร์จแบตเตอรี่ โดยเราจะเลือก แถบสว่างมาก BATTERY CHARGER โดยใช้แป้นพิมพ์ลูกศร ขึ้น หรือ ลง แล้วกดแป้น ENTER ต่อไปโดยการกดแป้น 1 ,2 ,3 ,4 แล้วจิง ENTER ต่อไปเราก็เลือกชนิดของ BATTERY ให้ถูกต้องกับ BATTERY ที่เราจะเอามา CHARGE เพื่อเป็นประโยชน์ต่อการ เลือกกระแสที่เหมาะสมของแต่ละชนิด ในการชาร์จแล้วเราก็จะใส่จำนวนของ แบตเตอรี่ ที่จะนำมาชาร์จว่ามีกี่เซล (1-8 เซล) เพื่อใช้เป็น แรงดันอ้างอิงที่ใช้ในการหยุดชาร์จแล้ว ENTER เครื่อง PC ก็จะทำการควบคมชด CONTROL ต่าง ๆ ตามค่าตัวแปร ที่เราใส่ไป โดยที่ จอภาพจะแสดงชนิดของแบตเตอรี่ ,VOLTAGE ต่าง ๆ ขณะนั้น และ เวลาที่ใช้ในการชาร์จเครื่อง PC จะทำการควบคุม วงจรชาร์จแบตเตอรี่ ให้หยุดชาร์จได้เมื่อแบตเตอรี่ได้ทำการชาร์จเต็มแล้วและเมื่อมีการกดแป้น ENTER ที่จอภาพก็จะกลับไปที่ MENU เอง ในการใช้งานของชุดการใช้ PC ควบคุมวงจร Power Supply และ DC.Voltage นั้นอาจจะมีปัญหาในเรื่องของความเที่ยงตรง และ แม่นยำในส่วนนี้เรามีวิธีการป้องกันให้มีความแม่นยำ และ เที่ยงตรงได้โดย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

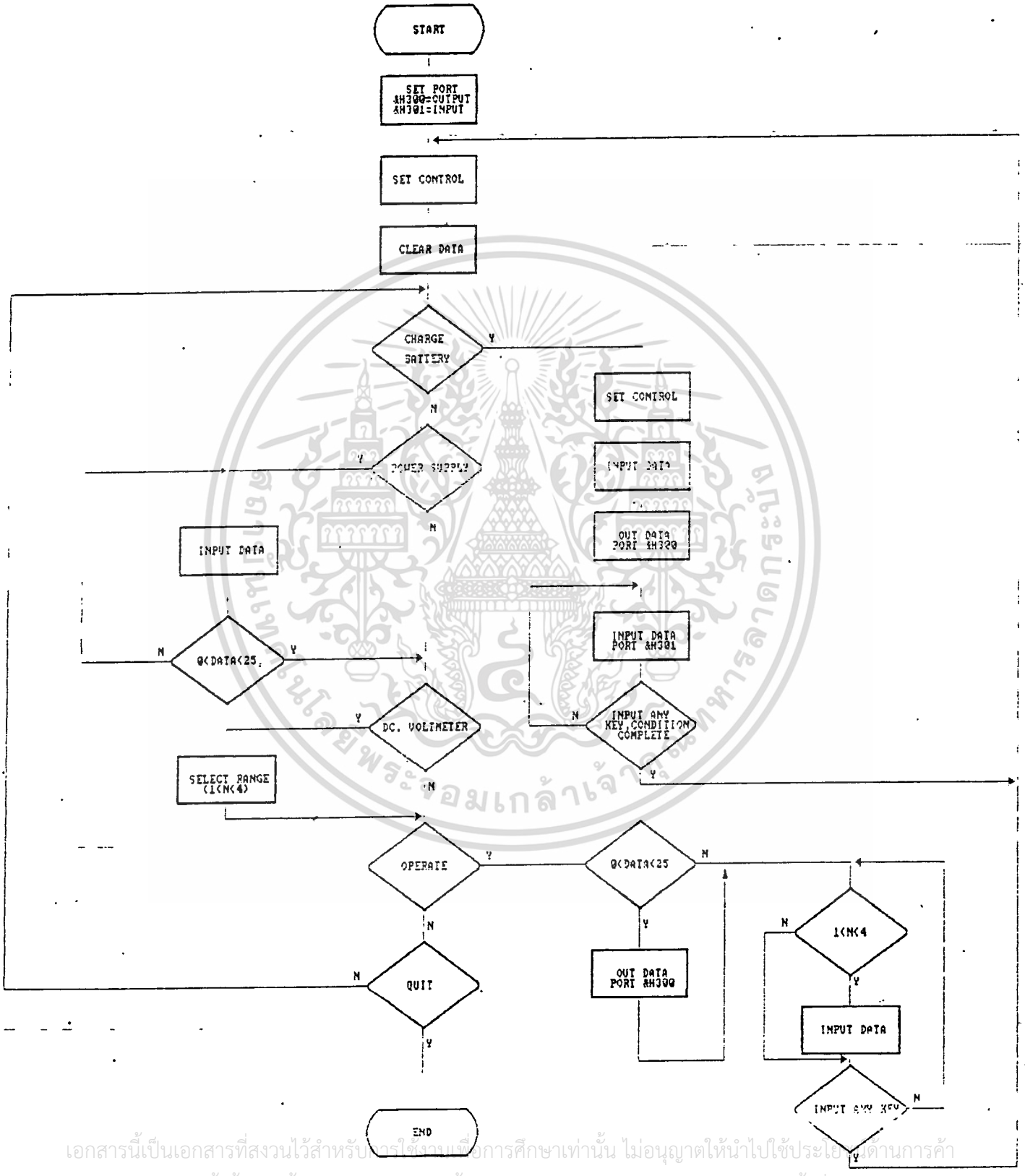
ในส่วนของ HARD WARE เราจะมี .VR เพื่อใช้ปรับค่า ให้มีความแม่นย่า และ เที่ยงตรง ได้ทั้งสองวงจร

สรุปและวิจารณ์ผลการทดลอง

ในส่วนของชุดการใช้ PC ควบคุมวงจร POWER SUPPLY ,วงจรชาร์จแบตเตอรี่ และ DC. Voltmeter นี้สิ่งที่สำคัญที่สุดก็ คือ ความเที่ยงตรง และแม่นย่านี้จึงจำเป็นอย่างยิ่งที่จะต้องพิจารณา วงจรที่จะนำมาใช้จะต้องมีเสถียรภาพที่ดี ซึ่งจะต้องพิจารณา หลายด้าน ไช่นความร้อน และ อื่น ๆ ซึ่งหากผู้อ่านมีวงจรที่ให้เสถียรภาพ ที่ดีกว่านี้ หรือ ข้อเสนอแนะต่าง ๆ ผู้จัดทำก็ต้องขออภัยไว้ ณ.ที่นี้ และ ยินดีที่จะรับ ข้อเสนอแนะต่าง ๆ เพื่อ เป็นประโยชน์ต่อผู้จัดทำ และ โครงงานนี้ให้มีประสิทธิภาพสูงยิ่ง ๆ ขึ้นไปอีก

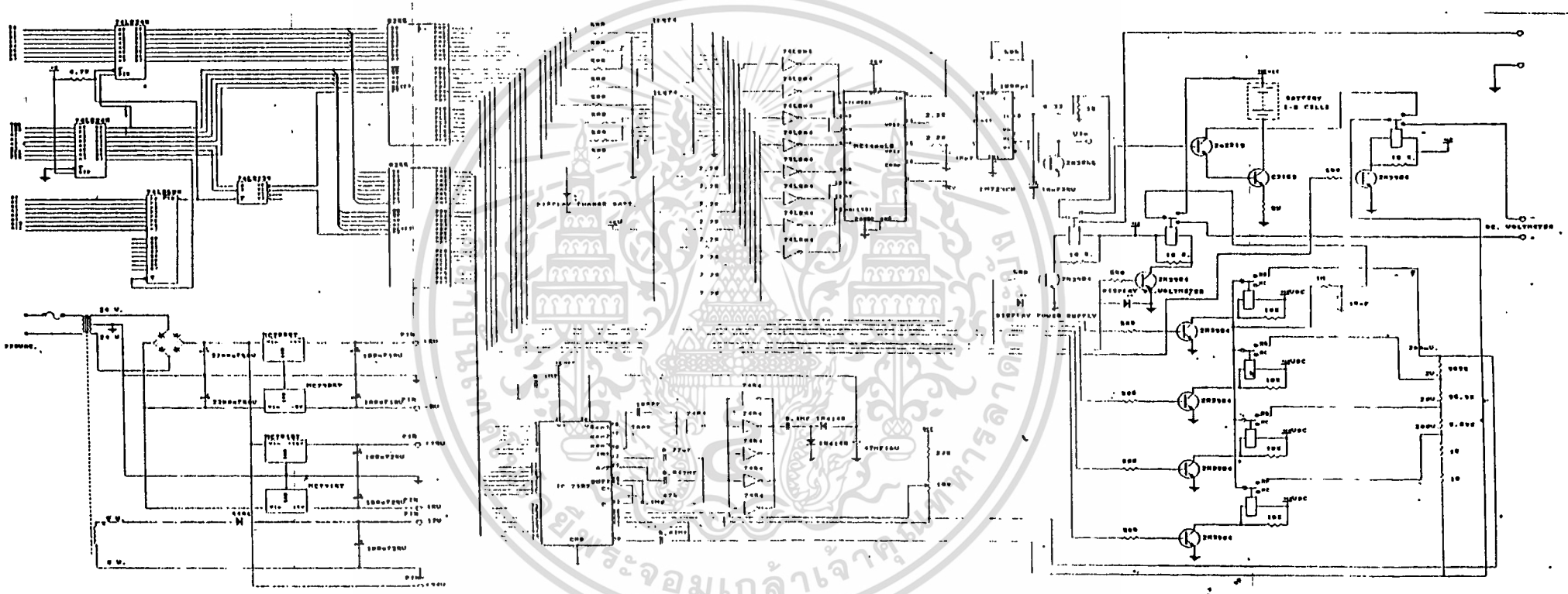
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

FLOW CHART แสดงการทำงานของเครื่อง



รูปวงจรรวม

POWER SUPPLY, BATTERY CHARGE, DC. VOLTMETER

ภาคผนวก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

National Semiconductor

LM723/LM723C Voltage Regulator

General Description

The LM723/LM723C is a voltage regulator designed primarily for series regulator applications. By itself, it will supply output currents up to 150 mA; but external transistors can be added to provide any desired load current. The circuit features extremely low standby current drain, and provision is made for either linear or foldback current limiting.

The LM723/LM723C is also useful in a wide range of other applications such as a shunt regulator, a current regulator or a temperature controller.

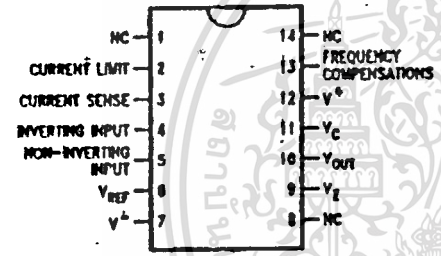
The LM723C is identical to the LM723 except that the LM723C has its performance guaranteed over a 0°C to +70°C temperature range, instead of −55°C to +125°C.

Features

- 150 mA output current without external pass transistor
- Output currents in excess of 10A possible by adding external transistors
- Input voltage 40V max
- Output voltage adjustable from 2V to 37V
- Can be used as either a linear or a switching regulator

Connection Diagrams

Dual-In-Line Package

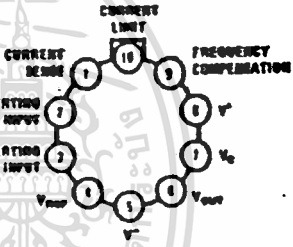


TL/H/8663–2

Top View

Order Number LM723J, LM723CJ, LM723CM or LM723CN
See NS Package J14A, M14A or N14A

Metal Can Package



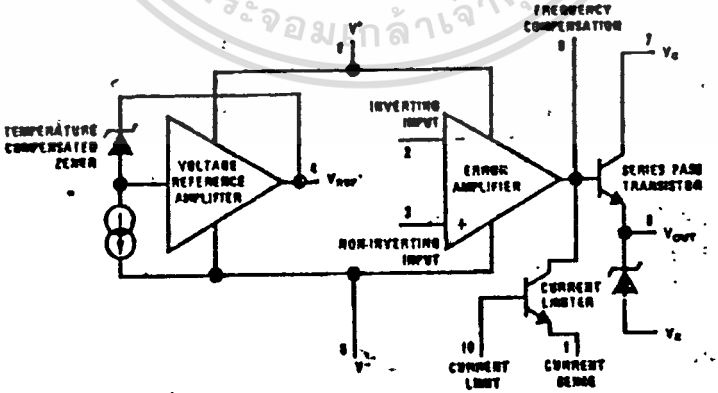
TL/H/8663–3

Note: Pin 5 connected to case.

Top View

Order Number LM723H or LM723CH
See NS Package H10C

Equivalent Circuit*



TL/H/8663–4

*Pin numbers refer to metal can package.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านก

UM7106R/07R

3½-Digit LCD A/D Converter

Features

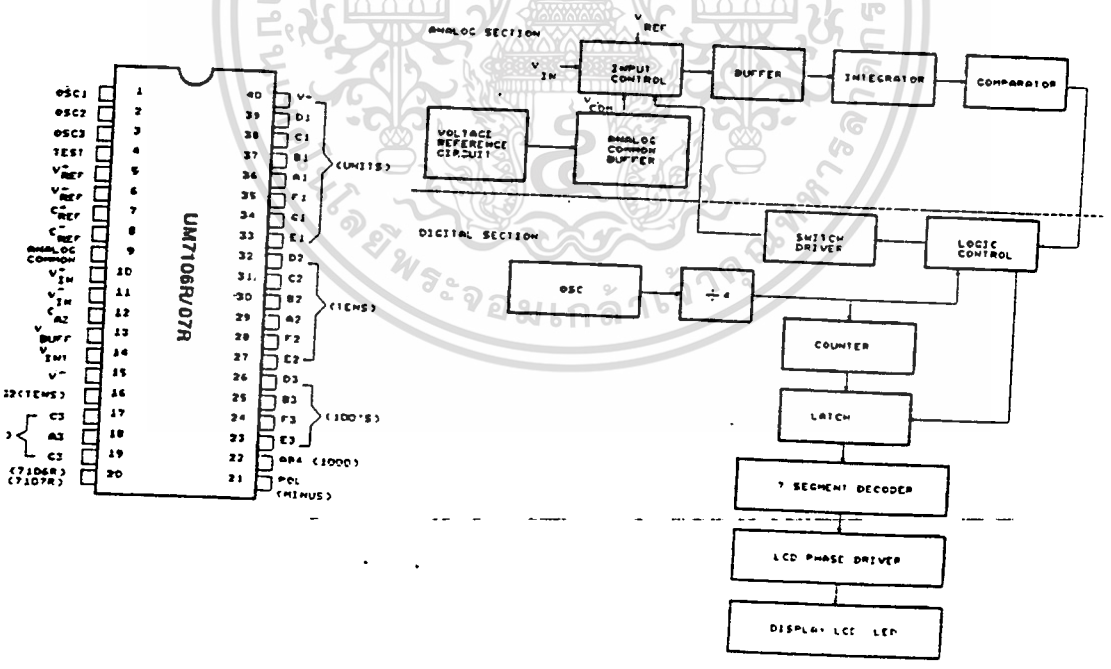
- Guaranteed zero reading with zero input
- True polarity indication for precision null detection
- Low noise
- Convenient 9V battery operation
- Low power operation-10mW
- High Impedance CMOS differential inputs-10 $10^{12}\Omega$
- True differential input and reference
- Direct display drive for UM7106R LCD, UM7107R LED
- No additional active components required
- Low linearity error guaranteed less than 1 count
- Internal reference with low temperature drift
- Applications: digital panel meters, digital multimeters, thermometers, capacitance meters, PH meters, photometers etc.

General Description

The UM7106R and UM7107R are high performance, low power 3½-digit A/D converters. All the necessary active devices are contained in a single CMOS IC, including seven-segment decoder, display driver, reference and lock. UM7106R is designed to interface with a liquid crystal display (LCD) and UM7107R will directly drive an instrument-sized light emitting diode (LED) display. The UM7106R and UM7107R bring the combination of high accuracy, versatility and economy. High accuracy, like auto-zero to less than 100μV for 2.00V full-scale measurement, or 10μV for 200.0 mV full-scale measurement, input bias current of 10pA max, and rollover of less than one count. The versatility of true differential input and reference is useful in all systems,such as strain gauges or bridge-type tranducers.

Pin Configuration

Block Diagram



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Absolute Maximum Ratings*

Supply Voltage (V+ to V-) 15V
Analog Input Voltage (either input) . . . . . V+ to V-
Reference Input Voltage (either input) . . . V+ to V-
Clock Input . . . . . Test to V+
Power Dissipation 800mW
Operating Temperature . . . . . 0°C to +70°C
Storage Temperature . . . . . -55°C to +150°C
Lead Temperature (Soldering, 60 sec) . . . . . 300°C

*Comments

Stresses above those listed under "Absolute Maximum Ratings" may cause permanent damage to the device. These are stress ratings only. Functional operation of this device at these or any other conditions above those indicated in the operational sections of this specification is not implied and exposure to absolute maximum rating conditions for extended periods may affect device reliability.

DC Electrical Characteristics

Unless otherwise noted, specifications apply to both the UM7106R, and UM7107R at V_{supply} (V+ to V-) = 9.0V

T_A = 25°C, f clock = 48KHz, UM7106R is tested in the CKT of Fig 7, UM7107R is tested in the CKT of Fig 8.

Parameter	Conditions	Min.	Typ.	Max.	Unit
Zero input Reading	V_{in} = 0.0V Full Scale = 200.0mV	-000.0	±000.0	+000.0	Digital Reading
Ratiometric Reading	V_{in} = V_{ref} V_{ref} = 100.0mV	999	999/1000	1000	Digital Reading
Linearity (Max. Deviation From Best Straight Line Fit)	Full Scale 200.0 mV or 2.000V	-1	±0.2	+1	Counts
Noise	V_{in} = 0V Full Scale 200.0mV	-	15	-	μV
Leakage Current Input	V_{in} = 0V	-	1	10	PA
Zero Reading Drift	V_{in} = 0 0° < T_A < 70°C	-	0.2	1	μV/°C
Scale Factor Temp Coeff	V_{in} = 199.0mV 0° < T_A < 70°C	-	1	5	ppm/°C
Analog Common Voltage (With Respect to Pos Supply)	25KΩ Between Common and Pos Supply	2.7	3.0	3.3	V
Temp Coeff of Analog Common	25KΩ Between Common and Pos Supply	-	20	50	ppm/°C

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

DC Electrical Characteristics (Continued)

Parameter	Conditions	Min.	Typ.	Max.	Unit
Supply Current (Doesn't Include LED and Com. Current)	V_{in} = 0		0.8	1.8	mA
UM7106R Only, Pk-Pk Segment Drive Voltage		4	5	6	V
UM7106R Only, Pk-Pk Back Plane Drive Voltage		4	5	6	V
UM7107R Only, Segment Sinking Current (Except Pin 22)	V_{supply} = 5.0V Segment Voltage = 3V	5	8.0		mA
UM7107R Only, Segment Sinking Current (Pin 22 Only)	V_{supply} = 5.0V Segment Voltage = 3V	10	16		mA
Rollover Error	$V_{in}^- = V_{in}^+$ = 200.0mV	-1	±0.2	+1	Counts
Common-mode Rejection Ratio	V_{cm} = ±1V_{in}, V_{in} = 0V Full-scale = 200.0mV		50	200	μV/V

Pin Description

Pin No.	Designation	Description
1	OSC1	Pins 1, 2, & 3 make up the oscillator section. For a 48KHz clock (3 readings per Second), connect pin 1 to the junction of a 100KΩ resistor and a 100 pF capacitor. The 100KΩ resistor is tied to pin 2 and the 100pF capacitor is tied to pin 3.
2	OSC2	See pin 1.
3	OSC3	See pin 1.
4	TEST	Lamp test. When pulled high (to V+), all segments will be turned on and the display should read-1888. It may also be used as a negative supply for externally generated decimal points.
5	V_{REF}^+	This analog input is required to generate a full-scale output (1,999 counts).Place 100 mV between pins 6 and 5 for 199.9 mV full-scale. Place 1.00 volt between pins 6 and 5 for 2 volts full-scale.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Pin Description (Continued)

Pin No.	Designation	Description
6	V_{REF}	See pin 5.
7	C^{+}_{REF}	A 0.1μF capacitor is used in most applications. If a large common-mode voltage exists (for example the V_{IN} pin is not at analog common), and a 200 mV scale is used, a 1.0μF is recommended and will hold the rollover error to 0.5 count.
8	C^{-}_{REF}	See pin 7.
9	Analog Common	This pin is primarily used to set the analog common-mode voltage for battery operation or in systems where the input signal is referenced to the power supply.
10	V^{+}_{IN}	The analog high input signal is connected to this pin.
11	V^{-}_{IN}	The analog low input is connected to this pin.
12	C_{AZ}	The size of the auto-zero capacitor influences the system noise. Use a 0.47μF capacitor for a 200mV full-scale, and a 0.047μF capacitor for a 2 volt full-scale.
13	V_{BUFF}	Integration resistor connection. Use a 47KΩ for a 200mV full-scale range and a 470KΩ for 2V full-scale range.
14	V_{INT}	Integrator output. Connection point for integration capacitor.
15	V-	Negative power supply voltage.
16	G2	Activates the G section of the tens display.
17	C3	Activates the C section of the hundreds display.
18	A3	Activates the A section of the hundreds display.
19	G3	Activates the G section of the hundreds display.
20	BP GND	LCD Backplane drive output (7106R). Digital Ground (7107R)
21	POL	Activates the negative polarity display.
22	AB4	Activates both halves of the 1 in the thousands display.
23	E3	Activates the E section of the hundreds display.
24	F3	Activates the F section of the hundreds display.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการ
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Pin Description (Continued)

Pin No.	Designation	Description
25	B3	Activates the B section of the hundreds display.
26	D3	Activates the D section of the hundreds display.
27	E2	Activates the E section of the tens display.
28	F2	Activates the F section of the tens display.
29	A2	Activates the A section of the tens display.
30	B2	Activates the B section of the tens display.
31	C2	Activates the C section of the tens display.
32	D2	Activates the D section of the units display.
33	E1	Activates the E section of the units display.
34	G1	Activates the G section of the units display.
35	F1	Activates the F section of the units display.
36	A1	Activates the A section of the units display.
37	B1	Activates the B section of the units display.
38	C1	Activates the C section of the units display.
39	D1	Activates the D section of the units display.
40	V+	Positive supply voltage.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการ
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Function Description

An input signal to be measured is applied to the integrating capacitance for a fixed time as determined by a clock counter. The accumulated charge will be proportional to the input signal, for a fixed clock rate and constant current. The resulting integral is returned to zero by integrating a reference signal of polarity opposite that of the input signal. The length of time required for the integrator to return to zero, as measured with the clock counter to display at output,is proportional to the average magnitude of the input signal over the integration period.

a. Analog Section

Fig. 1 shows the block diagram of the Analog Section for UM7106R and UM7107R. Each measurement cycle is divided into three parts. They are (1) Auto-zero [A-Z] (2) Signal integrated [INT] (3) Deintegrated [DE].

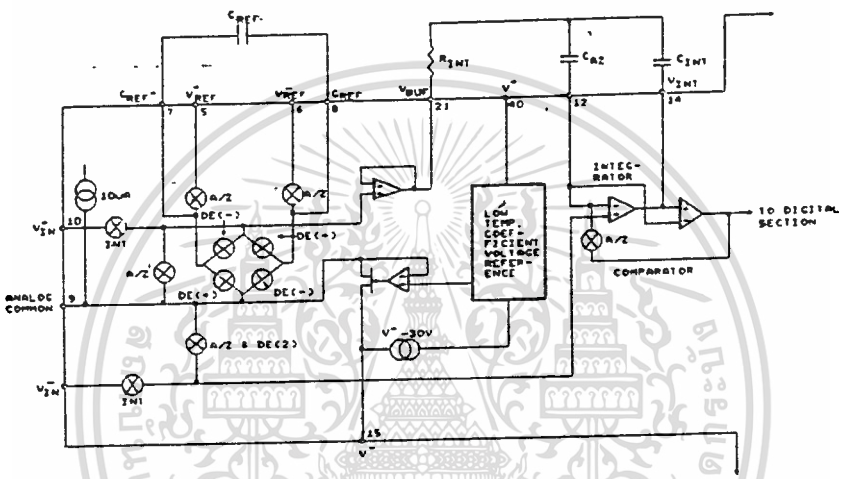


Figure 1. UM7106R/7107R Analog Section

1. Auto Zero Phase

During Auto-Zero, three things happen:

(1) input high and low are disconnected from PIN and shorted to analog COMMON.
(2) the reference capacitor is charged to the reference voltage.
(3) a feedback loop is closed around the system to charge the auto-zero capacitor C_{AZ} to compensate for offset voltages in the buffer amplifier, integrator and comparator.

2. Signal Integrate Phase

During signal integrate, the auto zero loop is opened ,the internal short is removed, and the internal inputs high and low are connected to the external pins. The converter then integrates the differential voltage between input high and input low for a fixed time (1000 counts). At the end of this phase, the polarity of the integrated signal is determined.

3. De-Integrated Phase

The final phase is de-integrate, or reference integrate. Input low is internally connected to Analog COMMON and input high is connected across the previously charged reference capacitor. Circuitry within the chip ensures that the capacitor will be connected with the correct polarity to cause the integrator output to return to zero. The time required to return to zero is proportional to the input signal. Specifically,the digital reading displayed is $1000\ (\frac{A_{in}}{V_{ref}})$.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการ
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

b. Digital Section

Fig. 2 and Fig. 3 show the digital section for the UM7106R and UM7107R, respectively. In the UM7106R, the internal digital ground is generated from a 6.2 volt Zener diode and a large p-channel follower. This supply is made stiff to absorb the relatively large capacitive current when the back plane (BP) is switched. The BP frequency is the clock frequency divided by 800. The segments are driven at the same frequency and amplitude and are in phase with BP when OFF, but out of phase when ON. In all cases, negligible DC voltage exists across the segments.

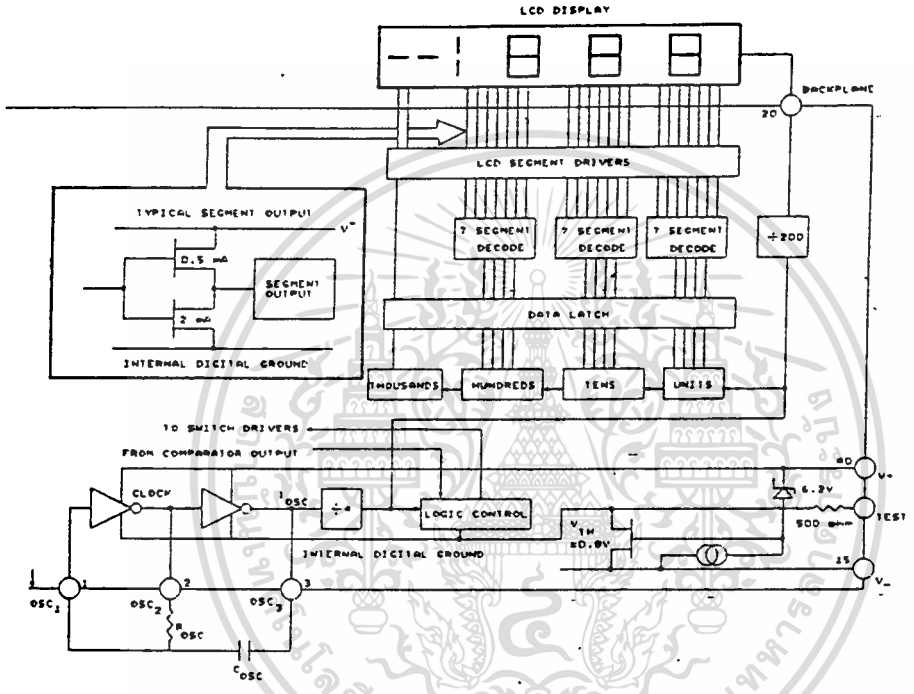


Figure 2. UM7106R Digital Section

Fig. 3 is the Digital Section of the UM7107R. It is identical except the regulated supply and back plane drive have been eliminated and the segment drive has been increased from 2 mA to 8 mA. PIN 22 must sink current twice the drive capability.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

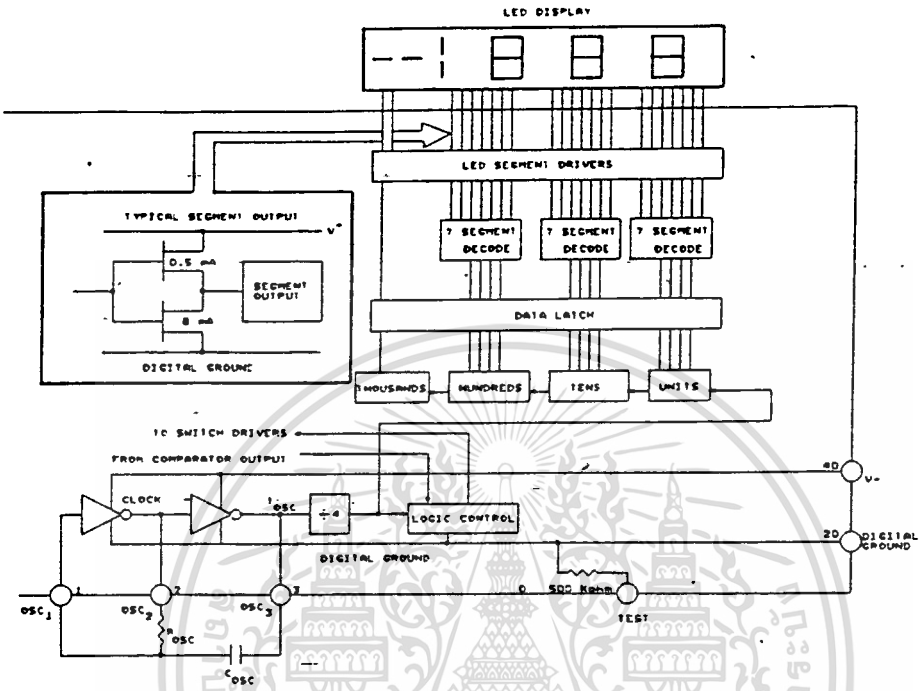
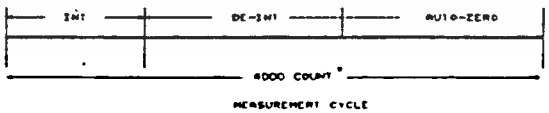


Figure 3. UM7107R Digital Section

System Timing

The oscillator frequency is divided by four before it clocks the decade counters. It is then further divided to form the three convert-cycle phase. These are signal integrated (1000 counts), reference de-integrate (0 to 2000 counts) and auto-zero (1000 to 3000 counts). For signals less than full scale, auto-zero gets the unused portion of reference deintegrate. This makes complete measure cycles of 4000 (16000 clock pulses) independent of input voltage.



* as a matter of fact, the total measurement cycle is 4001 counts for measured values less than 2000 counts. The measurement cycle becomes 4000 counts for overflow measurement.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Clock Circuit:

UM7106R/7107R may use the following three clocking methods:

1. An external oscillator connected to pin 1.
2. A crystal between pin 2 and pin 1.
3. An RC oscillator using all three pins (pin 3, 2 & 1).

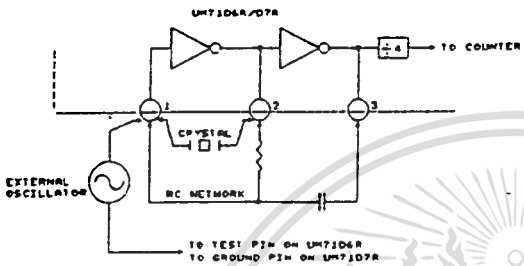


Figure 4. Clock Circuits

To achieve maximum rejection of 50/60 Hz pick up, the signal integrate cycle should be a multiple of 50/60 Hz. The following table describes the selection of oscillator frequencies for 50 or 60Hz respectively.

	Oscillator Frequencies
50Hz	..40KHz, 50KHz, 66$\frac{2}{3}$ KHz, 100KHz.....
60Hz	..33$\frac{1}{3}$ KHz, 40KHz, 48KHz, 60KHz, 80KHz, 100KHz....

Note that 40KHz (2.5 readings/second) will reject both 50 and 60Hz.

Component Value Selection:

Integrating Resistor (RINT):

The buffer amplifier and integrator are designed with class A output stages with 100μA of quiescent current each. They can supply 20μA drive current with negligible linearity errors. RINT should be large enough to remain in linear region but small enough to reduce the leakage current on the PC board.
For a 200mv full scale, RINT is 47KΩ; a 2 volt full scale needs a 470KΩ RINT.

Integrating Capacitor (CINT):

CINT should be chosen to give the maximum voltage swing without causing the saturation of integrator output swing. According to the superior temperature coefficient — 20ppm/°C of analog common, analog common will be normally used as the differential voltage reference. It is fine for a nominal ±2V full scale integrator output swing. For three readings/second (48KHz clock), a 0.22μF capacitor is suggested. If a different oscillator frequency is used, CINT must be changed in inverse proportion to maintain the nominal ±2V full scale integrator output swing.
An additional requirement of CINT is that CINT must have low dielectric absorption to minimize rollover error. Polypropylene capacitors give undetectable errors at reasonable cost.

Reference Voltage Capacitor (CREF):

The reference voltage used to ramp the integrator output voltage back to zero during the reference integrate cycle is stored on CREF. A 0.1μF capacitor gives good performance when V_{IN} is tied to analog common. If a large analog common voltage exists (V_{REF} unequal analog common)and a 200.0mV scale is used, a larger value is required to prevent rollover error. Generally 1.0μF will hold the rollover error to 0.5 count. In this case a mylar type dielectric capacitor is adequate.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการ
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Auto-Zero Capacitor (CAZ):

The CAZ value has some influence on system noise. The following combination is recommended:

Application	Adequate CAZ
200.0mV Full Scale	0.47μF
2.000V Full Scale	0.047μF

It is better to use a mylar type capacitor to implement CAZ

Oscillator Components (ROSC , COSC):

While using RC oscillator, the ROSC (between pin 2 and pin 1) should be 100KΩ and COSC is selected from the following equation:

$$Fosc = \frac{0.45}{Rosc. Cosc}$$ (Rosc in MΩ, Cosc in μF)

Reference Voltage Selection:

The analog input required to generate full scale output (2000 counts) is VIN = 2 VREF, thus

Required Full Scale Voltage	VREF
200.0mV	100.0mV
2.000V	1.000V

However, in many applications where A/D converter is connected to a transducer, there may exist a non-unity scale factor between the input voltage and the digital reading. For instance, a pressure transducer output is 400mV for 2000 lb/in^2, rather than dividing the input voltage by two, the VREF should be set to 200.0mV, then permit the transducer input to be used directly.

The differential Voltage reference can also be used when a digital zero reading is required when VIN is not zero. This case is common in temperature measuring instrumentation. A compensating offset Voltage can be applied between analog common and VIN and the transducer output is connected between VIN and analog common.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บรรณานุกรม

1. จิติ หนูแก้ว, **เทคนิคการเชื่อมต่อ IBM PC**, กรุงเทพฯ, บริษัทซีเอ็ด ยูเคชั่น จำกัด, 2535

2. ชัชวาล กุก่าจัด, **เชมิคอนดัคเตอร์อิเลคทรอนิกส์, เรกกูเลเตอร์ เพาเวอร์ซัพพลายควบคุมด้วยพีซี**, ฉบับที่ 112, 2534

3. ยืน ภู่วรวรรณ, **ทฤษฎี และ การประยุกต์ไมโครโปรเซสเซอร์ Z-80**, กรุงเทพฯ, บริษัท ซีเอ็ด ยูเคชั่น จำกัด, 2533

4. ยืน ภู่วรวรรณ, **เทคโนโลยีฮาร์ดแวร์ IBM PC**, บริษัท ซีเอ็ด ยูเคชั่น จำกัด, กรุงเทพ, บริษัท ซีเอ็ด ยูเคชั่น จำกัด, 2533

5. บริษัท ซีเอ็ด ยูเคชั่น , **คู่มือไมโครโปรเซสเซอร์ และ ไอซีที่เกี่ยวข้อง** , บริษัท ซีเอ็ด ยูเคชั่น จำกัด, 2536

6. อมร จงมหาวิญญู, **รวมโครงงาน เล่ม 6**, "เครื่องประจุถ่านนิแคด" กรุงเทพฯ, 2533

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้