การควบคุมระยะไกลโดยไมโครคอมพิวเตอร์

PC MICROCOMPUTER-BASED FOR REMOTE CONTROL

นาย สมบัติ แสงชาติ

นาย มังตรา ด่านสวัสดิ์

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาอุตสาหกรรมศาสตร์บัณฑิต

ภาควิชาเทคโนโลยีการวัดคุมทางอุตสาหกรรม

คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

พ.ศ. 2536

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

032623

PC MICROCOMPUTER-BASED FOR REMOTE CONTROL

Mr. Sombat Saengchati

Mr. Mungtra Darnswat

Project Report Submitted in Partial Fufillment of the Requi

For the Bachelor's Degree

Department of Industial Instrumentation Tenology

King Mongkut's Institute of Tecnology

1993

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

032623

หัวข้อปริญญานิพนธ์	การควบคุมระยะไกลโดยไมโครคอมพิวเตอร์
ชื่อ	นาย สมบัติ แสงชาติ
	นาย มังตรา ด่านสวัสดิ์
อาจารย์ที่ปรึกษา	ผศ. พิพัฒน์ เลาหสงคราม
ภาควิชา	เทคโนโลยีการวัดคุมทางอุตสาหกรรม
ปีการศึกษา	2536

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

บทคัดย่อ

ปริญญานิพนธ์เล่มนี้ เป็นการศึกษาการควบคุมอุณหภูมิและเซ็นเซอร์ระบบความปลอดภัยในระยะทางไกลๆโดยใช้ MICROCONTROLLER Z-80180 ส่งข้อมูลผ่าน RS-232 SERIAL INTERFACE จากนั้นค่าที่ได้จากการวัดจากระบบ จะมาปรากฏบน PC-MONITOR ในรูปของการฟิคสามมิติ และสามารถควบคุม การทำงานของระบบโดยการ ควบคุมที่ KEYBOARD ของ PERSONAL COMPUTER

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Project Report Title PC Microcomputer-Based for

Remote Control

Name Mr. Sombat Saengchati

Mr. Mungtra Darnswat

Project Advisor Asst. Prof. Phiphat Laohasongkram

Department Industrial Instrumentation Technol

Academic Year 1993

ABSTRACT

In this report we concern about in use of temperature and security sensors in case of long distance controlled by Z-180 Microprocessor and peripheral modules.The RD-232 serial interface method is brouth to share and connect to personal computer (com1 or com2)

Any values those measured must be appeared on personal computer moniter in there dimension graphic shapes and applied the personal computer keyboard to control

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กิตติกรรมประกาศ

ปริญญานิพนธ์นี้สำเร็จลุล่วงได้ด้วยดี โดยได้รับคำแนะนำและช่วยเหลือจากท่านอาจารย์ที่ปรึกษา ผศ.นิพนธ์ เลาหสงคราม ตลอดจนบรรดาเพื่อนน้องที่ได้ให้ความช่วยเหลือในด้านเครื่องมือ อุปกรณ์ ตลอดจนแนวทางแก้ไขปัญหาต่างๆ ที่ปรากฏขึ้นครั้งแล้วครั้งเล่า จนกระทั่งประสบผลสำเร็จดังที่ปรากฏ ทางคณะผู้จัดทำจึงขอขอบพระคุณทุกๆ ท่านเป็นอย่างสูงไว้ ณ. ที่นี้ด้วย

สมบัติ แสงชาติ

มังตรา ด่านสวัสดิ์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทนำ

ความมุ่งหมายของปริญญานิพนธ์

บทนำ

ในโลกปัจจุบันการใช้คอมพิวเตอร์ในองค์การธุรกิจ การศึกษา และวงการอุตสาหกรรมจะเป็นสิ่งที่ขาดแคลนมิได้เลย เพราะคอมพิวเตอร์เป็นเครื่องมือที่มีความสามารถสูง ทำให้ได้คำตอบ หรือผลลัพธ์ในทันทีโดยไม่ผิดพลาด

ดังนั้น การนำคอมพิวเตอร์มาประยุกต์ใช้กับงานอุตสาหกรรมเป็นแขนงหนึ่งของการใช้งานซึ่งกำลังเป็นที่นิยม เพราะคอมพิวเตอร์ที่มีความควบคุมระบบนั้น มีความเร็วในการทำงานสูงมาก มีขนาดเล็ก และเชื่อถือได้สูง

ในการนำเอาคอมพิวเตอร์ ไปควบคุมสัญญาณภายนอก หรือใช้คอมพิวเตอร์แสดงผลโดยได้รับสัญญาณจากภายนอก แล้วให้แสดงผลบน PERSONAL COMPUTER ในลักษณะต่างๆ เช่น แสดงผลในรูป GRAPHICS หรือ SYMBOL ต่างๆ USER เองสามารถเลือก หรือแก้ไข ข้อกำหนดในการใช้งานโดย แก้ไขที่ SOFT WARE ได้เลย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จุดประสงค์ของปริญญานิพนธ์

จุดประสงค์ของปริญญานิพนธ์ คือการนำเอาชิงเกิลบอร์ดของ Z-80180 มาประยุกต์กับงานควบคุมทางอุตสาหกรรม เช่น อุณหภูมิ เซ็นเซอร์ความปลอดภัย ระดับของของเหลว อัตราการไหล และอื่นๆ อีกมากขึ้นอยู่กับโปรแกรมและฮาร์ดแวร์ที่นำมาต่อร่วม ในที่นี้ได้นำเอา การวัดและการควบคุมอุณหภูมิ พร้อมด้วยเซ็นเซอร์ความปลอดภัย มาเป็นหัวข้อหลักในการศึกษา เพื่อที่จะศึกษาการทำงานของระบบและนำไปประยุกต์ใช้กับงานที่ควบคุมอื่นๆ อย่างเหมาะสม หรืองานที่มีลักษณะการควบคุมในระยะทางไกลๆ โดยใช้คอมพิวเตอร์เป็นตัวแสดงผล ซึ่งระบบเหล่านี้มีอยู่มากตามโรงงานอุตสาหกรรม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ

หน้า

บทที่ 1 PROCOMM PLUS

1.1 แนะนำให้รู้จัก PROCOMM PLUS 1

1.2 กล่าวทั่วไป 4

1.3 สิ่งที่ต้องการในการใช้ PROCOMM PLUS 4

1.4 การติดตั้ง PROCOMM PLUS 5

1.5 การติดตั้ง PROCOMM PLUS 7

1.6 การทำงานของ PCINSTAL 8

1.7 การกำหนดค่าต่างๆใน PROCOMM PLUS 9

1.8 การสตาร์ต PROCOMM PLUS 15

1.9 จอภาพโหมดเทอร์มินัล 18

1.10 การใช้คำสั่งต่างๆ ของ PROCOMM PLUS 19

1.11 การกำหนดพอร์ต COM 22

1.12 การออกจาก PROCOMM PLUS 25

1.13 ถ้ามีปัญหา 25

บทที่ 2 แนะนำให้รู้จักกับไมโครคอมพิวเตอร์ส่วนบุคคลของไอบีเอ็ม

2.1 แนะนำ 26

2.2 หน่วยหลัก 26

2.2.1 บอร์ดโปรเซสเซอร์ 28

2.2.2 แหล่งจ่ายกำลัง 28

2.2.3 ตัวขับฟล้อบบี้ดิส 28

2.2.4 ลำโพง 29

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.3 แป้นพิมพ์ของระบบ 29
2.4 เครื่องพิมพ์ของระบบ 30
2.4.1 โหมดของการพิมพ์ 30
2.4.2 ระหัสของเครื่องพมพ์ 30
2.5 จอแสดงผลแบบสองสี 31
2.6 คาร์ดอื่นๆ ที่แนะนำจากไอบีเอ็ม 31
2.6.1 อะแดปเตอร์ควบคุมฟล้อปปี้ดิส 31
2.6.2 คาร์ดพอร์ทเครื่องพิมพ์แบบขนาน 32
2.6.3 คาร์ดพอร์ทอนุกรม 32
2.6.4 คาร์ดควบคุมเกมส์ 32
2.6.5 คาร์ดแสดงผลแบบสี 32
2.6.6 คาร์ดแสดงผลแบบสองสีและพอร์ทเครื่องพิมพ์
แบบขนาน 33
2.6.7 คาร์ดโปรโต้ทาย 33
2.6.8 คาร์ด SDLC 34
2.6.9 คาร์ดขยายหน่วยความจำ 34
2.6.10 คาร์ดและอุปกรณ์อื่นๆ 34

บทที่ 3 การเชื่อมต่อสัญญาแอนาลอกกับคอมพิวเตอร์และการควบคุมทาง
อุตสาหกรม

3.1 แนะนำ 36
3.2 แนะนำคุณสมบัติเบื้องต้นของวงจรออปแอมพ์ 37
3.3 วงจรออปแอมพ์และการประยุกต์ใช้งาน 37
3.3.1 วงจรเปรียบเทียบที่ใช้ออปแอมมพ์ 37
3.3.2 วงจรขยายแอปแอมพ์แบบไม่กลับขั้ว 40
3.3.3 วงจรขยายแอปแอมพ์แบบกลับขั้ว 41

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.3.4 วงจรบวกที่ใช้ออปแอมพ์ 42
3.3.5 วงจรขยายความแตกต่างเบื้องต้น 44
3.3.6 วงจรขยายเครื่องมือวัด 45
3.3.7 วงจรอินทิเกรทที่ใช้ออปแอมพ์ 46
3.3.8 วงจรอนุพันธ์ที่ใช้ออปแอมพ์ 46
3.3.9 วงจรกรองแบบแอคทีฟที่ใช้ออปแอมพ์ 47
3.4 เซนเซอร์และทรานดิวเซอร์ 51
3.4.1 เซนเซอร์ทางแสง 51
3.4.2 เซนเซอร์อุณหภูมิ 53
3.4.2.1 เซนเซอร์อุณหภูมิแบบสารกึ่งตัวนำ 54
3.4.2.2 เทอร์โมคัปเปิล 55
3.5 ทรานสดิวเซอร์ความดันและแรง 58
3.5.1 เสตรนเกจน์และโหลดเซลล์ 58
3.5.2 หม้อแปลงความแตกต่างเชิงเส้น 61
3.5.3 เซนเซอร์ของไหล 63
3.6 เซนเซอร์ชนิดอื่นๆ 64

บทที่ 4 การแปลงดิตอลเป็นอนาลอก

4.1 แนะนำ 65
4.2 การแปลง D/A แบบใช้ตัวต้านทานแบ่งน้ำหนัก 66
4.3 การแปลง D/A แบบใช้ตัวต้านทานขั้นบันได 68
4.4 คุณสมบัติและข้อกำหนดของตัวแปลง D/A 71
4.4.1 ความละเอียด 71
4.4.2 ค่าเต็มพิกัดของแรงดันเอ้าพุท 72
4.4.3 ความเที่ยงตรง 73

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.4.4 ความเป็นเชิงเส้น 73
4.4.5 ค่าเวลาในการแปลง .....
4.5 การใช้งานและการเชื่อมต่อตัวแปลง D/A กับไมโครคอมพิวเตอร์..... 74

บทที่ 5 การแปลงอนาลอกเป็นดิจิตอล

5.1 แนะนำ 80
5.2 การแปลง A/D แบบอินทิเกรท 80
5.2.1 การแปลง A/D แบบสโลปเดี่ยว 81
5.2.2 การแปลง A/D แบบสโลปคู่ 83
5.2.3 การแปลง A/D แบบสามสโลป 84
5.2.4 การแปลง A/D แบบขนาน 86
5.3 การแปลง A/D แบบขนาน 87
5.4 การแปลง A/D แบบไบนารี่แรมพ์ 87
5.5 การแปลง A/D แบบประมาณค่าต่อเนื่อง 89
5.6 ระหัสจากการแปลง A/D 91
5.7 การเชื่อมต่อตัวแปลง A/D ชนิดต่างๆ เข้ากับไมโครคอมพิวเตอร์ 91
5.7.1 การเชื่อมต่อตัวแปลง A/D แบบตัวเปรียบเทียบขนาน..... 91
5.7.2 การเชื่อมต่อตัวแปลง A/D แบบสโลป 93
5.7.3 การเชื่อมต่อตัวแปลง A/D แบบประมาณค่าต่อเนื่อง 95

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 6 ระบบทางกลของบัสและคุณสมบัติทางกำลังของแหล่งจ่าย

6.1 แนะนำ 97

6.2 สล้อตเสียบคาร์ดของระบบบัส 97

6.3 ขนาดของคาร์ 97

6.4 ภาคจ่ายกำลังของระบบ 100

6.5 การชดเชยการขาดการจ่ายกำลัง 100

บทที่ 7 การใช้คำสั่งของ PROCOMM PLUS

7.1 กล่าวทั่วไป 102

7.2 Auto Answer (การตอบโดยอัตโนมัติ) ALT-Y ... 102

7.3 Break Key ALT-B 103

7.4 Chang Directory ALT-F7 103

7.5 Chat Mode ALT-O 104

7.6 Clear Screen ALT-C 105

7.7 Clipboard ALT-= 106

7.8 CR-CR/CF Toggle ALT-F3 108

7.9 Dialing Directory 110

7.10 Dos Gateway 110

7.12 Editor 111

7.13 Ealpsed Time 113

7.14 Exit 113

7.15 File Directory 114

7.16 Hang Up 115

7.18 Host Mode 116

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

7.19 Init Modem 116

7.20 Kermit Server Commands 117

7.21 Key Mapping 118

7.22 Line/Port Setyp 118

7.23 Log File On/Off 118

7.24 Log File Paise 119

7.25 Meta Key 120

บทที่ 8 เทคนิคการออกแบบวงจรดิจิตอล

8.1 การออกแบบวงจรส่วนที่สอง ส่วนแสดงผล 122

ผนวก

แนะนำให้รู้จักการติดต่อสื่อสาร

1. กล่าวทั่วไป 137
2. โมเด็ม 137
3. ASCII 139
4. บิตและไบต์ 139
5. การส่งข้อมูลแบบอนุกรมและแบบขนาน 140
6. บิตเริ่มต้นและบิตหยุด 141
7. Parity (การตรวจสอบ) 141
8. โพโตคอล 142
9. อัตราความเร็วในการรับส่งข้อมูล 144
10. Delux 144

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รายละเอียดทั่วไป 146

คุณสมบัติต่างๆ 147

อัตราค่าสูงสุดสมบูรณ์ (ABSOLUTE MAXIMUM RATING).. 148

รายละเอียดภายในส่วนของ ANALOG 151

หน้าที่ของ AUTO-ZERO (CAZ) 152

หน้าที่ของ SIGNAL INTEGRATE 153

หน้าที่ของ DE-INTEGRATE 153

ค่าความแตกต่างทางด้าน INPUT 154

ค่าความแตกต่างอ้างอิง 154

ANALOG COMMON 155

การทดสอบ 156

ภาพแสดงลักษณะ CP-180 162

ข้อกำหนดบน BOARD 163

รายละเอียดขา IC 169

รายละเอียดขา CONNECTOR 170

RS232 CONNECTION 174

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 1

PROCOMM PLUS

1.1 แนะนำให้รู้จัก PROCOMM PLUS

PROCOMM PLUS เป็นซอฟต์แวร์การติดต่อสื่อสารซึ่งมีทุกสิ่งทุกอย่างที่ต้องการ โดยมีคุณสมบัติย่อๆดังนี้

Fully-Automated Dialing Directory สามารถเก็บชื่อ หมายเลขและการติดตั้งคอมพิวเตอร์เพื่อการทำงานได้ถึง 200 เครื่อง ใน Dialing Directory (ไดเรกทอรีในการหมุนโทรศัพท์) และสามารถสร้างและใช้ไดเรกทอรีในการหมุนโทรศัพท์เป็นจำนานเท่าใดก็ได้ PROCOMM PLUS ช่วยในการใช้รหัสในการหมุนโทรศัพท์ (Dialing) ได้ถึง 10 รหัสในข้อมูลของเราดังนั้นจึงสามารถเรียกรหัสโทรศัพท์ทางไกลหรือหมุนโทรศัพท์โดยผ่านทางระบบ PBX ได้เหยง่าย PROCOMM PLUS สามารถหมุนโทรศัพท์ได้ทั้งหมายเลขเดียว หรือหลายหมายเลข ด้วยการหมุนใหม่โดยอัตโนมัติถ้าหากหมายเลขนั้นๆไม่วาง โปรแกรมนี้จะเก็บรักษาวันที่ท่านโทรศัพท์ครั้งล่าสุด และจำครั้งที่โทรศัพท์ไว้. เพื่อจะได้ใช้ตรวจสอบใบเก็บเงินค่าโทรศัพท์ได้โดยง่าย การจัดลำดับไดเรกทอรีทำได้ 5 วิธี และแม้แต่การใส่ข้อความเตือนความจำ และหมายเหตุไว้ในข้อมูลเพื่อไว้อ้างอิงในอนาคตก็ทำได้

Error-Corecting File Transfer Protocols (รหัสการย้ายไฟล์แก้ความผิด)PROCOMM PLUS ใช้กับรหัสดังกล่าวได้ถึง 15 รหัสมาตรฐาน และยอมให้กำหนดรหัสภายนอกได้อีกสามรหัส

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การข่าวสารสนเทศที่นิยมใช้กัน เรียกข้อมูลที่ต้องการเข้ามา แล้วพิมพ์เป็นรายงาน ทั้งหมดนี้ทำได้แม้แต่เข้านอนไปแล้ว และในขณะที่โทรศัพท์และอัตราค่าต่อโทรศัพท์ต่ำที่สุดก็ตาม นอกจากนี้ยังจัดไฟล์สคริปต์ตัวอย่างมาให้เพื่อติดต่อกับบริการข่าวสารสนเทศ Compuserve และบริการข่าวสารสนเทศอื่นๆได้อีกมาก

Keybaord Automation PROCOMM PLUS มี Meta keys แบบหลายฟังก์ชันคีย์ (Multifuction meta keys) สำหรับเก็บแถวตัวคาแรกเตอร์ที่ใช้สั่งบ่อยๆ (เช่น ชื่อ-ID ของผู้ใช้ และรหัสผ่าน) การรันสคริปต์โดยการกดคีย์เพียงคีย์เดียว การเอ็กซีคิวต์โปรแกรมภายนอก หรือการติดต่อกับโปรแกรมภายนอกที่เขียนขึ้นมาเอง นอกจากนี้ยังสามารถกำหนดคีย์ขึ้นใหม่ต่างหากได้ถึง 79 คีย์ ให้แก่แต่ละการจำลองแบบของสถานีปลายทางทั้ง 33 แบบได้อีกด้วย

Line Setting PROCOMM PLUS ทำงานที่ 300, 1200, 2400, 4800, 9600, 19200, 38400, 57600 และ 115200 baud สามารถทำการตรวจสอบความถูกต้อง (parity) อย่างใดก็ได้ จะเป็น 7 หรือ 8 บิตข้อมูล (data bits) และ 1 หรือ 2 บิตหยุด (stop bits) ก็ได้ PROCOMM PLUS สามาใช้กับพอร์ตอนุกรมที่ผู้ใช้กำหนดขึ้นได้ถึง 8 พอร์ต และสามารถกำหนดข้อมูลที่ป้อนเข้าแต่ละไดเรกทอรีในการหมุนโทรศัพท์ให้กับพอร์ตอนุกรมของตนเองได้

Host Mode โหมด Host ที่มีความสามารถสูงของ PROCOMM PLUS ยอมให้เรียกไปยังคอมพิวเตอร์ได้จากระยะไกล เพื่อการย้ายไฟล์โดยไม่ต้องมีผู้ใดไปคอยดูแล การส่งจดหมายทางคอมพิวเตอร์ การเรียก DOS การเอ็กซิคิวต์จากโปรแกรมระยะไกล และอื่นๆอีกมาก ผู้ควบคุมคอมพิวเตอร์ที่เป็นแม่ข่าย (Host) สามารถกำหนดชื่อผู้ใช้แต่ละคน (IDs) รหัสผ่านและระดับรักษาความปลอดภัยให้แก่ผู้ใช้ที่อยู่ทางด้านไกลได้ สามารถเพิ่มเมนูที่สร้างขึ้นเองและโปรแกรมภายนอกให้แก่คอมพิวเตอร์แม่ข่ายได้โดยง่าย

Chat Mode PROCOMM PLUS ช่วยให้แยกจอภาพได้เมื่อคุยกับอีก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Terminal Emulation (การจำลองแบบสถานีปลายทาง)PROCOMM PLUS สามารถจำลองแบบจอวีดีโอแสดงผล ของสถานีปลายทางที่นิยมกันถึง 33 ชนิด (รวมทั้ง ANSI ที่ใช้กันใน Bulletin board ด้วย) การจำลองแบบสถานีปลายทางมาสารถรันโปรแกรมใช้งานของมินิและเมนเฟรมคอมพิวเตอร์ได้ในแบบเต็มจอโดยใช้คอมพิวเตอร์ส่วนบุคคลเป็นสถานีปลายทางที่ควบคุมจากระยะไกล นอกจากนี้ PROCOMM PLUS ยังมีคุณลักษณะการทำงานได้ทั้งในแบบ 132 คอลัมน์จริง (ด้วยจอแสดงผลความละเอียดสูง และกราฟแสดงรูปภาพที่นิยมกันมากที่สุด) และในแบบ 132 คอลัมน์จำลองการ Scrolling (สำหรับจอแสดงผลและการ์ดแสดงผลอื่นๆทั้งหมดผสมกัน)

Context-Sensitive Help PROCOMM PLUS ให้ความช่วยเหลือโดยละเอียดแบบเต็มจอภาพ เมื่อกำลังอยู่ในโปรแกรมได้ โปรแกรมอำนวยความสะดวกในการให้ความช่วยเหลือตามข้อความในจอภาพ (Context sensitive) โดยสามารถอยู่เสมอว่ากำลังทำอะไรอยู่และจะแสดงความช่วยเหลือในงานที่กำลังทำอยู่ในขณะนั้น

A Powerful Script Language (ภาษา Script ที่มีความสามารถสูง)ลักษณะของภาษา Script ของ PROCOMM PLUS เป็นภาษาที่มีความเร็วและความสามารถสูงมาก เป็นภาษาในการแปล (Compiled) โปรแกรมโดยมีโครงสร้างคล้ายภาษา C ที่นิยมกันมาก สามารถใช้เงื่อนไข (IF/THEN/ELSE และ SWITCH/CASE) ได้ พร้อมด้วยการจัดการกับจอภาพที่ทันสมัยต่างๆ กับไฟล์ I/O ฟังก์ชันทางคณิตศาสตร์ การทำลูปวนตัวแปรของภาษานี้และตัวแปรที่ผู้ใช้กำหนดขึ้นเอง กรรมวิธีต่างๆ และอื่นๆอีกมากมาย โปรแกรมจะถูกคอมไพล์โดยอัตโนมัติ เพื่อให้ทำงานอย่างมีประสิทธิภาพด้วยความเร็ว และมีความปลอดภัยสูง สามารถใช้ไฟล์สคริปต์ เพื่อทำการติดต่อกับคอมพิวเตอร์ใหญ่ (Log-ons) ได้โดยอัตโนมัติ ทำการย้ายไฟล์ได้โดยไม่ต้องมีใครมาดูแล และงานอื่นๆอีกมากมาย หรือจะใช้ไฟล์สคริปต์เพื่อให้โปรแกรม PROCOMM PLUS กลับมาทำงานได้ตลอดเวลา ไม่ว่าจะเป็นกลางวันหรือกลางคืน เช่น PROCOMM PLUS จะเรียกบริ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

คนหนึ่งที่คอมพิวเตอร์ซึ่งอยู่ในระยะไกลได้

Multitasking Environment PROCOMM PLUS สามารถรันได้ในโปรแกรม Windows, DESQview, Topviewและซอฟต์แวร์ในการทำงานหลายๆอย่าง (Multitasking) ได้

1.2 กล่าวทั่วไป

ในบทนี้จะอธิบายในเรื่อง

- วิธีการติดตั้งโปรแกรม
- วิธีการติดตั้งไมเต็ม คอมพิวเตอร์ และโปรแกรม PROCOMM PLUS เพื่อเริ่มต้นทำงาน
- วิธีการเริ่มต้นและออกจาก PROCOMM PLUS
- วิธีป้อนคำสั่งในโหมดสถานีปลายทาง
- วิธีขอความช่วยเหลือเมื่อเกิดความต้องการ
- วิธีแก้ปัญหาที่เกิดขึ้น

1.3 สิ่งที่ต้องการในการใช้ PROCOMM PLUS

- คอมพิวเตอร์: IBM PC, XT, AT, PS/2 หรือที่คอมแพตทิเบิลกัน
- หน่วยความจำ: อย่างน้อยที่สุด 192K RAM
- โปรแกรมควบคุมระบบ: PC DOS หรือ MS-DOS เวอร์ชั่น 2.0 ขึ้นไป
- ดิสก์ไดรฟ์: อย่างน้อยที่สุดต้องมีหนึ่งดิสก์ไดรฟ์ หรือหนึ่งดิสก์ไดรฟ์กับหนึ่งฮาร์ดดิสก์
- การแสดงผล: จอโมโนโครมกับอะแดปเตอร์การแสดงผลจอคอมโพสิตพร้อมกับอะแดปเตอร์แสดงภาพสีจอ EGA กับอะแดปเตอร์แสดงผลแบบ EGA จอ VGA กับอะแดปเตอร์แสดงผลแบบ VGA หรือจอแสดงผลพร้อมกับอะแดปเตอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

แสดงผลที่คอมแพตทิเบิล กับการแสดงผลอย่างใดอย่างหนึ่งดังกล่าวแล้ว

- โมเด็ม: จะเป็นแบบโมเด็มใดๆก็ได้ทั้งที่เป็นแบบคอมแพตทิเบิลกับ Hayes หรือไม่ก็ได้ จะเป็นแบบติดตั้งภายนอกหรือภายในคอมพิวเตอร์มีความเร็วเท่าไรก็ได้ สำหรับคอมพิวเตอร์ที่ต่อกันโดยตรงด้วยสายแบบอนุกรมไม่ต้องการโมเด็ม แต่ต้องการใช้สายเคเบิลพิเศษ Null modem cable เพื่อต่อคอมพิวเตอร์สองเครื่องเข้าด้วยกัน รายละเอียดของ Null modem cable นี้จะอยู่ภาคผนวกท้ายเล่ม

- พอร์ตอนุกรม: ต้องการเมื่อใช้โมเด็มแบบติดตั้งภายนอก หรือใช้ต่อโดยตรงเข้ากับคอมพิวเตอร์อีกเครื่องหนึ่งเท่านั้น

1.4 การติดตั้ง PROCOMM PLUS

การติดตั้งซอฟต์ในการติดต่อสื่อสารอาจจะเป็นเรื่องยาก เช่น จะต้องใช้พอร์ต COM พอร์ตใด? โมเด็มของเราใช้กับความเร็ว 9600 baud ได้ไหม? จะใช้การจำลองแบบลดเนื่องสายทางแบบไหน?

คำถามเหล่านี้ที่เป็นเหตุให้โปรแกรม PCINSTAL เข้ามามีบทบาทการรันโปรแกรม ที่สะดวกต่อผู้ใช้โปรแกรมนี้นั้น ง่ายเหมือนกับการใช้โปรแกรม PROCOMM PLUS เองทีเดียวโปรแกรม PCINSTAL จะพิจารณาการติดตั้งที่ถูกต้องโดยอัตโนมัติตามที่ต้องการในการติดตั้ง หรือการปรับปรุงโปรแกรม PROCOMM PLUS ในคอมพิวเตอร์ให้

การติดตั้งไม่ใช่เพียงแต่ก๊อปปี้ไฟล์ต่างๆ ในแผ่นดิสก์ที่ให้มาเท่านั้น PROCOMM PLUS จะต้องติดตั้งด้วยโปรแกรม PCINSTAL ซึ่งจะต้องสร้างและจัดการไฟล์ข้อมูลทั้งหลายตามความจำเป็นให้ตามต้องการ

ถ้าท่านมี PROCOMM PLUS อยู่แล้ว PCINSTAL ก็สามารถปรับปรุง PROCOMM PLUS ให้เป็นเวอร์ชั่นใหม่ได้ (หรือถ้ากำหนดไดเรกทอรีอื่นให้กับไฟล์ใหม่ ก็จะต้องมีทั้งสองเวอร์ชั่นอยู่ในเครื่องเดียวกัน) ถ้ากำหนดไดเรกทอรีเดียวกันให้ PCINSTAL จะลบ และ/หรือ แปลงไฟล์ข้อมูลต่าง จากเวอร์ชั่น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เก่าในขณะที่ทำการปรับปรุงใหม่ให้

ก่อนรัน PCINTSAL จะต้องมั่นใจได้ว่ามีข้อมูลต่างๆ ดังต่อไปนี้

ชนิดของการต่อ PROCOMM PLUS ต่อเข้ากับคอมพิวเตอร์อื่นด้วยโมเด็ม (โดยใช้สายโทรศัพท์) หรือด้วยสายเคเบิลแบบอนุกรม?

พอร์ต COM ใช้พอร์ตการติดต่อสื่อสาร (หรือพอร์ต COM) ของคอมพิวเตอร์พอร์ตใด? ปกติจะเป็นตัวเลขจาก 1 ถึง 8 PCINSTAL จะพิจารณาพอร์ตที่มีอยู่ให้เองโดยอัตโนมัติ

หมายเลขประจำโปรแกรม หมายเลขประจำโปรแกรมของท่าน(จากแผ่นการ์ดที่ให้มากับโปรแกรม)

ชนิดของจอภาพ ถ้าใช้การ์ด VGA หรืออะแดปเตอร์แสดงกราฟิกแบบ Super EGA แล้ว PCINSTAL อาจจะถามว่าการแสดงผลทำในโหมด 132 ได้หรือไม่ ไม่ว่าจะใช้การแสดงผลของจอภาพแบบใด PCINSTAL จะพยายามหาแบบนั้นให้จนได้

เป็นการติดตั้งใหม่ หรือว่าเป็นการปรับปรุง PCINSTAL จะถามว่าเป็นการติดตั้ง PROCOMM PLUS ในครั้งแรก หรือว่าเป็นการปรับปรุงจากเวอร์ชั่นก่อน ถ้าเป็นการปรับปรุง PCINSTAL จะพยายามหาไดเรกทอรีที่เก็บก๊อปปี้ของโปรแกรม PROCOMM PLUS ไว้ในขณะนั้น ถ้าหาไดเรกทอรีนี้ไม่พบ PCINSTAL จะถามท่านว่าไดเรกทอรีนี้อยู่ที่ไหนต่อไป ถ้าใช้โมเด็มภายนอกให้ตรวจสอบดูว่าได้ต่อเข้ากับพอร์ตอนุกรมเรียบร้อยและเปิดสวิตช์ไว้แล้ว

ถ้าคอมพิวเตอร์มีเมาส์ ก็สามารถใช้เมาส์กับเมนูของ PCINSTALL

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ได้ เมื่อเลื่อนเมาส์ไป เคอร์เซอร์ที่เป็นแถบสว่างก็จะเลื่อนไปด้วย ปุ่มซ้ายของเมาส์จะทำหน้าที่เหมือนคีย์ ENTER ในขณะที่ปุ่มขวาทำหน้าที่เหมือนคีย์ ESCAPE (Esc)

พูดถึงคีย์ Esc เราสามารถจะกลับไปยังเมนูอันก่อนเมื่อใดก็ได้ใน PCINSTAL โดยกดคีย์ Esc การออกจาก PCINSTAL ให้กดคีย์ Esc ไปจนกระทั่ง PCINSTAL พากลับไปยังเครื่องหมายพร้อมรับคำสั่งของ DOS

ในการเริ่มต้น ให้ใส่ดิสก์แผ่นที่ 1 ในไดรฟ์ A แล้วพิมพ์ว่า

```
pcinstal
```

แล้วกด Enter PCINSTAL จะดำเนินการติดตั้งโปรแกรมให้ มีการทำรายงานให้ในระหว่างการติดตั้ง (ถ้าเป็นการปรับปรุง รายงานนี้จะมีขั้นตอนเพิ่มเติมที่จะต้องทำด้วยมือ) เมื่อการติดตั้ง PROCOMM PLUS เสร็จเรียบร้อยแล้ว PCINSTAL จะแสดงข้อความว่าเสร็จการติดตั้งแล้ว และให้ดูไฟล์ README ของเวอร์ชั่นนี้ได้

1.5 การติดตั้ง PROCOMM PLUS

จะต้องใช้โปรแกรม PCINSTAL ในแผ่นดิสก์ที่ให้มาเพื่อทำการติดตั้ง PROCOMM PLUS การก๊อปปี้ไฟล์ต่างๆ จากแผ่นดิสก์ที่ให้มาจะไม่เป็นการติดตั้ง PROCOMM PLUS ที่ถูกต้อง PCINSTAL ออกแบบมาให้ก๊อปปี้ไฟล์ที่เหมาะสมและจัด PROCOMM PLUS ให้คอมพิวเตอร์ทำงานได้ผลดีที่สุด เพื่อรันโปรแกรม PCINSTAL ให้ทำดังนี้

1. ใส่แผ่นดิสก์ของโปรแกรม PROCOMM PLUS เข้าไปในไดรฟ์ A
2. พิมพ์ว่า a:pcinstal แล้วกดคีย์ Enter
3. ทำตามคำแนะนำที่ปรากฏในจอภาพโปรแกรม PCINSTAL ซึ่งง่ายต่อผู้ใช้และเข้าใจง่าย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ถ้าจอแสดงผลโมโนโครมแบบผสมโดยมีการ์ดแสดงผลทางกราฟิกสี ให้ใช้สวิทช์ /b เพื่อบังคับ PCINSTAL ให้อยู่ในโหมดโมโนโครม (หรือสีขาว-ดำ) ในการรัน PCINSTAL ในโหมดโมโนโครม ให้พิมพ์ว่า

```
a:pcinstal /b
```

แล้วกดคีย์ Enter PCINSTAL จะปฏิบัติงานหลายอย่างตามความจำเป็นเพื่อการติดตั้งเป็นไปอย่างถูกต้อง (รวมทั้งการแปลงไฟล์ต่างๆ ตามความต้องการถ้าทำการปรับปรุงเวอร์ชั่นเก่าของโปรแกรม PROCOMM PLUS) นอกจากนี้ PCINSTAL ยังช่วยให้ปรับโปรแกรม PROCOMM PLUS ให้เหมาะสมกับฮาร์ดแวร์ใหม่ที่อาจจัดหามาให้ในอนาคตได้ (เช่น โมเด็มใหม่ การ์ดแสดงผลรูปภาพหรือจอภาพใหม่) โดยไม่ต้องทำการติดตั้งโปรแกรม PROCOMM PLUS ใหม่ทั้งหมด

1.6 การทำงานของ PCINSTAL

สำหรับการติดตั้งทุกชนิด PCINSTAL จะทำงานดังนี้

1. ก๊อปปี้ไฟล์ต่างๆ จากดิสก์ที่จ่ายมาให้ลงไว้ในฮาร์ดดิสก์ (หรือถ้ารัน PROCOMM PLUS ด้วยฟลอปปี้ดิสก์ไดรฟ์ก็จะก๊อปปี้ไฟล์ไว้ในแผ่นดิสก์ใช้งาน)

2. กำหนดข้อความควบคุมโมเด็มที่ถูกต้อง (ถ้ามีการแจ้งยี่ห้อและแบบของโมเด็มไว้ในขณะติดตั้ง) และการกำหนดค่าเริ่มต้นให้แก่โมเด็ม

3. จะปล่อยให้เลือกค่าที่กำหนดไว้ให้โดยปริยาย สำหรับทางเลือกต่างๆ ของ PROCOMM PLUS (เช่น path ที่กำหนดไว้ให้โดยปริยายสำหรับไฟล์ที่บรรจุลงในคอมพิวเตอร์รหัสในการย้ายไฟล์ที่กำหนดไว้ให้โดยปริยาย และการจำลองสถานีปลายทางที่กำหนดไว้ให้โดยปริยาย)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4. เพิ่มบรรทัดในการคอนฟิกมาเพื่อทำงานด้วย DOS ให้ไฟล์ AUTOEXEC.BAT และ CONFIG.SYS (ถ้ายังไม่มีไฟล์ทั้งสองนี้) ถ้ามีไฟล์ทั้งสองนี้อยู่แล้ว PCINSTAL จะขอให้ทำการแก้ไขไฟล์ทั้งสองนี้เสียใหม่

5. กำหนดโหมดขยายการแสดงผลวิดีโอ (สำหรับคอมพิวเตอร์ที่มีอะแดปเตอร์แสดงผลและจอภาพแสดงผลแบบ VGA) ยอมให้มีบรรทัดหรือคอลัมน์เพิ่มขึ้นในจอภาพ

6. ทำก๊อปปี้โปรแกรม PROCOMM PLUS เป็นสมบัติส่วนตัวของท่าน โดยใส่หมายเลขประจำโปรแกรมลงในจอภาพแสดงข้อมูลของโปรแกรม (เรียกได้จากโหมดเมอร์มินัล ด้วยการกดคีย์ Alt-I)

หากเป็นการปรับปรุงโปรแกรม PROCOMM PLUS จากเวอร์ชันเดิม PCINSTAL จะทำงานดังนี้

1. ทำงานทั้งหมดดังกล่าวมาแล้ว

2. แปลงไฟล์ Dialing Directory (.DIR) ไฟล์ Keyboard Map (.KBD) และไฟล์ Keyboard Macro (.KEY) ให้เป็นแบบฟอร์มที่ใช้ใน PROCOMM PLUS ของเวอร์ชันนี้ ในระหว่างการแปลงจะไม่มีการสูญหายของข้อมูลแต่อย่างใด

3. แปลงไฟล์ด้วยตัวแปรในการติดตั้ง (PCPLUS.PRM) ให้เป็นแบบฟอร์มที่ใช้ในเวอร์ชันนี้ของโปรแกรม

ทั้งหมดนี้เป็นงานที่ได้ทำไปเมื่อติดตั้งโปรแกรม PROCOMM PLUS ลงในคอมพิวเตอร์เสร็จเรียบร้อยแล้ว

1.7 การกำหนดค่าต่างๆใน PROCOMM PLUS

ถ้า PCINSTAL แจ้งว่าการดำเนินการติดตั้งโปรแกรมเรียบร้อยแล้ว ก็แสดงว่า PCINSTAL ได้ทำงานส่วนใหญ่ตามที่อธิบายไว้ในข้อนี้ไปแล้ว หากที่ดีควรจะข้ามไปอ่านหัวข้อ "การสตาร์ต PROCOMM PLUS" ได้เลย แต่ถ้าต้องการทำการติดตั้งเป็นขั้นตอนด้วยมือแล้ว ก็จะเห็นได้ว่ามีการทำงานต่างๆ ตามที่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

032623

อธิบายไว้ดังต่อไปนี้ เมื่อทำการขั้นตอนที่จำเป็นเสร็จเรียบร้อยแล้ว ให้ข้ามไปอ่านในหัวข้อ "การสาร์ต PROCOMM PLUS"

การ configure โมเด็ม

ก่อนที่จะเริ่มใช้ PROCOMM PLUS จะต้องตรวจดูให้แน่ใจว่าได้ติดตั้งโมเด็มไว้ถูกต้องแล้ว ถ้าระบุโมเด็มไว้ระหว่างการติดตั้ง PCINSTAL ก็จะเป็นการให้ PROCOMM PLUS ทำการกับโมเด็มได้ไว้เรียบร้อยแล้ว

การเตรียมโมเด็มขึ้นอยู่กับว่าโมเด็มที่มีอยู่เป็นโมเด็มภายใน (ติดตั้งอยู่ภายในแผ่นบอร์ดภายในคอมพิวเตอร์) หรือว่าเป็นโมเด็มภายนอก (แยกออกมานอกคอมพิวเตอร์)

การติดตั้งโมเด็ม

ถ้ายังไม่ได้ต่อโมเด็มเข้ากับคอมพิวเตอร์และกับสายโทรศัพท์ ก็ให้ทำเสียเดี๋ยวนี้เลย คู่มือโมเด็มจะบอกวิธีการต่อไว้ให้

โมเด็มส่วนใหญ่จะมีสวิตช์อยู่ชุดหนึ่ง ซึ่งใช้ตั้งคุณลักษณะของการปฏิบัติงาน ที่แน่นอนของโมเด็มไว้ มีความจำเป็นที่จะต้องตรวจสอบดูสวิตช์เหล่านี้สองสวิตช์ เพื่อให้ใช้คุณะสมบัติต่างๆ ของ PROCOMM PLUS ได้ทั้งหมด

ให้ตรวจสอบกับคู่มือของโมเด็มเกี่ยวกับสวิตช์ทั้งสองนี้ (ในคู่มือนั้นอาจเรียกสวิตช์ทั้งสองสวิตช์นี้ว่า "modem switches", "configuration switch" หรือ "DIP switches" ก็ได้) สวิตช์ที่จะตั้งตรวจสอบทั้งสองได้แก่

1. สวิตช์ CD (Carrier Detect) สวิตช์นี้เป็นสวิตช์ที่ 6 ของโมเด็มที่คอมแพตทิเบิลกับของ Hayes ส่วนมากให้ตั้งสวิตช์นี้ให้โมเด็มควบคุมสาย CD ไม่ได้)

2. สวิตช์ DTR (Data Terminal Ready) สวิตช์นี้เป็นสวิตช์ที่ 1 ของโมเด็มที่คอมแพตทิเบิลกับของ Hayes เป็นส่วนมาก ให้ตั้งสวิตช์นี้เพื่อให้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โมเด็มเฝ้าฟังสถานภาพของสาย DTR (ถ้าตั้งสวิตช์ไว้ที่อีกตำแหน่งหนึ่ง โมเด็มจะไม่รับรู้สถานภาพของสาย DTR) การติดตั้งสวิตช์นี้จะเร่งกรรมวิธีการหยุดทำงานให้เร็วขึ้น เมื่อเลือกคำสั่งมราเหมาะสมของ PROCOMM PLUS แล้ว ถ้าไม่ตั้งสวิตช์ไว้ในลักษณะที่กล่าวนี้ PROCOMM PLUS ก็จะหยุด การทำงานได้ แต่ว่าจะใช้เวลานานกว่า

หมายเหตุ ถ้าโมเด็มไม่สวิตช์เหล่านี้ ก็จะมีคำสั่งในซอฟต์แวร์ที่สามารถ ใช้เพื่อให้บรรจุวัตถุประสงค์เช่นเดียวกันได้ ให้อ่านคู่มือของโมเด็มดูเพื่อหาคำสั่งในการควบคุม CD และ DTR (โมเด็มส่วนใหญ่ใช้คำสั่ง &C1 และ &C2) จากนั้น ให้ใส่คำสั่งทั้งสองนี้ลงในคำสั่ง ในการกำหนดค่าเริ่มต้น (initialization) ของโมเด็ม ถ้าโมเด็มยอมรับคำสั่ง &W ให้พิมพ์ว่า AT &C1 &C2 &W (ในโหมดเทอร์มินัล) แล้วกดคีย์ Enter โมเด็มจะเก็บค่าที่ตั้งใหม่นี้ไว้ในหน่วยความจำที่ไม่ลบเลือน (non-volatile) คำสั่ง &C1 จะบอกให้โมเด็มติดตามสถานภาพที่แท้จริงของสายที่เกี่ยวกับ CD ส่วนคำสั่ง &C2 ก็จะทำอย่างเดียวกัน สำหรับ DTR และคำสั่ง &W จะเขียนข่าวสารสนเทศนี้ลงในหน่วยความจำไม่ลบเลือนของโมเด็ม (รายละเอียดการใช้คำสั่ง &W ให้ดูในคู่มือของโมเด็ม)

การตั้งค่าให้โมเด็มภายใน

ถ้ายังไม่ได้ติดตั้งโมเด็มเข้ากับคอมพิวเตอร์ และต่อกับสายโทรศัพท์ให้ทำเสียเดี๋ยวนี้เลย คู่มือของโมเด็มจะบอกวิธีการติดตั้งไว้ให้แล้ว

โมเด็มภายใน สามารถตั้งให้ทำงานโดยต่อกับพอร์ตการติดต่อสื่อสาร (COM ports) ใดๆ ก็ได้ โดยคิดว่าพอร์ตการติดต่อสื่อสารเป็น *การต่อ* ระหว่างโมเด็มกับคอมพิวเตอร์

พอร์ตการติดต่อสื่อสาร หรือเรียกอีกอย่างว่า *พอร์ตอนุกรม* โดยปกติแล้วไว้ใช้กับเครื่องพิมพ์ และอุปกรณ์การชี้ (เมาส์) (นอกเหนือจากโมเด็ม) ถ้าคอมพิวเตอร์ต่อกับอุปกรณ์ดังกล่าวไปแล้ว ก็อาจจะใช้พอร์ตสื่อสารไปแล้วพอร์ต

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หนึ่ง โมเด็มก็จะใช้ต่อเข้ากับพอร์ตดังกล่าวแล้วนี้ไม่ได้ ควรจะทราบว่าอุปกรณ์ดังกล่าว ต่อเข้ากับพอร์ตใด เพื่อจะได้ตั้งค่าสำหรับโมเด็มให้ทำงานกับพอร์ตสื่อสารอื่นๆ ที่เหลืออยู่ได้ถูกต้อง (ให้ดูเรื่องการตั้งค่าพอร์ต COM ของโมเด็มได้จากคู่มือของโมเด็ม)

PROCOMM PLUS สามารถทำงานกับพอร์ตการติดต่อสื่อสารพอร์ตหนึ่งพอร์ตใดใน 8 พอร์ตได้ คือ COM1 ถึง COM8

หมายเหตุ ถ้าโมเด็มมีสวิตช์ CD หรือ DTR ก็ให้ตั้งค่าสวิตช์ดังอธิบายมาแล้ว สำหรับโมเด็มภายนอก ถ้าโมเด็มไม่มีสวิตช์ดังกล่าว ให้ตรวจสอบกับคู่มือของโมเด็มว่ามีคำสั่งซอฟต์แวร์ที่สามารถใช้เพื่อการนี้อยู่หรือไม่ ถ้ามีให้ร่วมคำสั่งเหล่านั้นไว้ในคำสั่งการกำหนดค่าเริ่มต้นของโมเด็ม

การ configure คอมพิวเตอร์

ในการรัน PROCOMM PLUS ให้ได้ประสิทธิภาพสูงสุด จะต้องแน่ใจว่าไฟล์ CONFIG.SYS และไฟล์ AUTOEXEC.BAT (ถ้าคอมพิวเตอร์มีฮาร์ดดิสก์) จะต้องมีคำสั่งต่างๆ เป็นที่แน่นอนจำนวนหนึ่ง

ไฟล์ CONFIG.SYS และ AUTOEXEC.BAT ถ้ามี จะอยู่ในไดเรกทอรีรากของดิสก์ที่ใช้บูตเครื่อง ถ้ายังไม่คุ้นเคยกับไฟล์เหล่านี้ให้อ่านในคู่มือดอสหรือหนังสือเกี่ยวกับดอส เพื่อวิธีแก้ไขไฟล์เหล่านี้ ดังจะได้เสนอแนะต่อไป ถ้าให้ PCINSTAL แก้ไขไฟล์ทั้งสองนี้แล้ว ก็จะจัดการให้ฮาร์ดดิสก์ทำงานกับ PROCOMM PLUS ได้แล้ว

DOS Search Path สามารถสตาร์ต PROCOMM PLUS จากไดเรกทอรีใดก็ได้ในดิสก์โดยกำหนดไดเรกทอรีของ PCPLUS ลงไป (เป็นไดเรกทอรีที่ติดตั้ง PROCOMM PLUS) ในคำสั่ง PATH ของไฟล์ AUTOEXEC.BAT

PCPLUS Environment Variable เมื่อ PROCOMM PLUS จำเป็นต้องเรียก หรือสร้างไฟล์ต่างๆ ที่ใช้ในไดเรกทอรีของการหมุนโทรศัพท์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ในโปรแกรมอำนวยความสะดวก Setup ในโปรแกรมอำนวยความสะดวกในการกำหนดที่อยู่ของคีย์บอร์ดในคีย์ Meta และตารางแปลคีย์เหล่านี้

ปกติ PROCOMM PLUS จะดูแต่ไดเรกทอรีที่กำลังใช้งานอยู่เท่านั้น ถ้าต้องการให้เก็บไฟล์เหล่านี้ไว้ในไดเรกทอรีอื่น ให้กำหนดไดเรกทอรีนั้นลงในคำสั่ง SET PCPLUS ในไฟล์ AUTOEXEC.BAT แล้ว PROCOMM PLUS จะค้นหาไฟล์เหล่านี้ในไดเรกทอรีนั้น ถ้าค้นหาไม่พบในไดเรกทอรีที่กำลังใช้งานอยู่ในขณะนั้น

เช่น ถ้าต้องการเก็บไฟล์ต่างๆของ PROCOMM PLUS ไว้ในไดเรกทอรี \COM\PCPLUS ก็จะต้องในคำสั่งเหล่านี้ไว้ในไฟล์ AUTOEXEC.BAT ด้วยคือ

```
SET PCPLUS=C:\COM\PCPLUS
```

ถ้าได้กำหนดตัวแปรสภาพแวดล้อมนี้ลงไปแล้ว และต่อมาเกิดตัดสินใจว่า ต้องการที่จะเก็บไฟล์ต่างๆของ PROCOMM PLUS ไว้ให้ได้หลายๆชุด ก็สามารถลบตัวแปรนี้ออกไปได้โดยพิมพ์ว่า

```
set pcplus=
```

ที่เครื่องพร้อมรับคำสั่งของดอส แล้วกดคีย์ ENTER

คำสั่ง Files PROCOMM PLUS จำเป็นต้องเรียกหลายไฟล์ในขณะเดียวกันได้ (ไฟล์ต่างๆที่ PROCOMM PLUS ใช้อยู่ในภาคผนวกท้ายเล่ม) เพื่อเรียกไฟล์เหล่านี้ได้ จะต้องแน่ใจว่าคำสั่ง FILES ในไฟล์ CONFIG.SYS อย่างน้อยที่สุดต้องเท่ากับ 20

คำสั่ง BUFFERS คำสั่งนี้สามารถกำหนดให้เป็นจำนวนสูงเท่าไรก็ได้ตามแต่จะต้องการ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เพื่อให้เข้าถึงดิสก์ได้เร็วขึ้น ให้ใส่คำสั่ง BUFFERS ไว้ในไฟล์ CONFIG.SYS ดังนี้

BUFFER = 20 ถ้าคอมพิวเตอร์ใช้ไมโครโปรเซสเซอร์ 80286, 80386 หรือ 80486

BUFFER = 10 ถ้าคอมพิวเตอร์ใช้ไมโครโปรเซสเซอร์ 8088 หรือ 8086

การ Configure PROCOMM PLUS

แม้ว่า PROCOMM PLUS จะเปลี่ยนตัวแปรในการติดตั้งได้มากกว่า 100 ก็ตาม แต่ก็มีสิ่งเดียวเท่านั้นที่ต้องทำในการเริ่มสตาร์ตคือ การระบุพอร์ต COM ที่ RPOCOMM PLUS ใช้ (ซึ่งทำแล้วใน PCINSTAL) วิธีเปลี่ยนตัวแปรตัวอื่นๆ ของ Setup

PCINSTAL ได้ดัดแปลง PROCOMM PLUS ให้กับโมเด็มเรียบร้อยแล้ว (ถ้าเป็นหนึ่งในโมเด็มที่ปรากฏอยู่ในรายการ) ถ้าเปลี่ยนโมเด็มใหม่ PCINSTAL ก็สามารถดัดแปลง PROCOMM PLUS ให้กับโมเด็มใหม่ได้ โดยเพียงแต่สตาร์ต PCINSTAL อีกครั้งด้วยการใส่แผ่นโปรแกรมไว้ในไดรฟ์ A แล้วพิมพ์ว่า

```
a:pcinstal
```

ที่เครื่องหมายรับคำสั่งของดอส แล้วกด ENTER เพื่อเริ่มต้น ถ้าใช้โมเด็มที่ไม่ปรากฏในรายการที่มีให้เลือกในระหว่างการติดตั้ง ก็จะต้องให้ข้อมูลในวิธีการทำงานของโมเด็มแก่ PROCOMM PLUS อย่างพอเพียง โดยใช้โปรแกรมอำนวยความสะดวก Setup (รั้งจากที่สตาร์ต PROCOMM PLUS แล้ว) ให้ดูในคู่มือของโมเด็มว่า PROCOMM PLUS ต้องการทราบอะไรบ้าง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1.8 การสตาร์ต PROCOMM PLUS

การสตาร์ต PROCOMM PLUS ครั้งแรก

การสตาร์ต PROCOMM PLUS ในคอมพิวเตอร์ที่ใช้ฮาร์ดดิสก์ ต้องตรวจดูให้แน่ใจว่ามีเครื่องพร้อมรับคำสั่งของดอสปรากฏอยู่ในจอภาพ หากเรียบร้อยแล้วก็พิมพ์ลงไปว่า

```
cd\pcplus
```

หรือไดเรกทอรีที่ติดตั้ง PROCOMM PLUS ไว้ แล้วกดคีย์ ENTER หลังจากนั้นพิมพ์ลงไปว่า

```
pcplus
```

และกดคีย์ ENTER เครื่องหมาย PROCOMM PLUS จะปรากฏขึ้นในจอภาพ และ PROCOMM PLUS จะกำหนดค่าเริ่มต้นให้แก่โมเด็มโดยส่งแถวคำสั่งต่างๆ ไปให้แก่โมเด็ม

กดคีย์ใดๆ เพื่อป้อนโหมดเทอร์มินัลให้แก่ PROCOMM PLUS การป้อนโหมดเทอร์มินัลเข้าไปเพื่อจะได้ดำเนินการต่อไป (ให้ดูเรื่อง "จอภาพของโหมดเทอร์มินัล" ได้ในบทนี้)

การสตาร์ต PROCOMM PLUS ในคอมพิวเตอร์ที่ใช้ดิสก์ไดรฟ์ก่อนที่จะสตาร์ตโปรแกรม PROCOMM PLUS เป็นครั้งแรก ควรก๊อปปี้แผ่นดิสก์ใช้งานของแผ่นดิสก์ PROCOMM PLUS ไว้ (ให้ดูเรื่อง "การติดตั้ง PROCOMM

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

PLUS") หลังจากตรวจดูให้แน่ใจแล้วว่ามีเครื่องหมายพร้อมรับคำสั่งของดอส ปรากฏอยู่ในจอภาพ ให้ใส่แผ่นดิสก์ใช้งานของ PROCOMM PLUS (ซึ่งทำไว้ในระหว่างการติดตั้ง) เข้าไปในไดรฟ์ A เสร็จแล้วพิมพ์ลงไปว่า

```
pcplus
```

แล้วกดคีย์ ENTER เครื่องหมายของ PROCOMM PLUS จะปรากฏขึ้นในจอภาพ และ PROCOMM PLUS จะกำหนดค่าเริ่มต้นให้แก่โมเด็ม โดยส่งแถวคำสั่งต่างๆ ไปให้โมเด็ม

กดคีย์ใดๆ เพื่อป้อนโหมดเทอร์มินัลให้แก่ PROCOMM PLUS การป้อนโหมดเทอร์มินัลเข้าไปเพื่อจะได้ดำเนินการต่อไป (ให้ดูเรื่อง "จอภาพของโหมดเทอร์มินัล" ได้ในบทนี้)

หมายเหตุ โปรแกรม PROCOMM PLUS จำเป็นต้องเรียกไปยังแผ่นดิสก์ใช้งานของ PROCOMM PLUS ในไดรฟ์ A อยู่ตลอดเวลา จึงต้องปล่อยแผ่นดิสก์นี้ไว้ในไดรฟ์ A ในขณะที่กำลังรันโปรแกรม PROCOMM PLUS อยู่

การสตาร์ต PROCOMM PLUS ด้วยวิธีอื่น

สามารถเติมคำสั่งมูลฐานในการสตาร์ต PROCOMM PLUS ลงไปให้เหมาะกับการติดตั้งฮาร์ดแวร์ และเหมาะกับความพอใจของแต่ละบุคคลได้ (ถ้าเพิ่งเริ่มใช้ PROCOMM PLUS และต้องการใช้ on-line ให้เร็วที่สุดเท่าที่จะทำได้แล้ว ให้ข้ามตอนนี้ไปก่อน)

การสตาร์ต PROCOMM PLUS ในลักษณะพิเศษ ให้พิมพ์ *สวิตช์* ลงไปหลัง pcplus ที่เครื่องหมายพร้อมรับคำสั่งของดอส

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 1.1

พิมพ์ว่า	หมายความว่า
pcplus /s	สตาร์ต PROCOMM PLUS โดยไม่ต้องให้เกิดเสียงดังขึ้นเมื่อเปิด/ปิดวินโดว์ ถ้าต้องการใช้ PROCOMM PLUS โดยไม่ต้องให้มีเสียงดังอยู่เสมอเป็นการง่ายกว่าที่จะปิดเสียงเหล่านี้เสียโดยใช้โปรแกรม อำนายความสะดวก Setup
pcplus /b	สตาร์ต PROCOMM PLUS เพื่อให้แสดงผลเป็นสีขาวและดำ แม้ว่าจอภาพจะเป็นจอสีก็ตาม ควรใช้สวิตช์นี้ถ้าใช้จอโมโนโครมแบบผสมพร้อมด้วยบอร์ดแสดงผลกราฟิก อย่างไรก็ดีถ้าเปลี่ยนสีที่กำหนด ให้โดยปริยายเป็นสีดำและขาว โดยใช้โปรแกรมอำนายความสะดวก Setup หลังจากนั้นแล้วก็ไม่จำเป็นต้องใช้สวิตช์ /b อีกต่อไป
pcplus /fFILENAME	สตาร์ต PROCOMM PLUS แล้วเอ็กซีคิวต์ไฟล์สคริปต์ตามชื่อที่กำหนดให้ในทันที เช่น เมื่อป้อน pcplus /fstartup จะสตาร์ต PROCOMM PLUS แล้วเอ็กซิคิวต์ไฟล์สคริปต์ STARTUP.ASX (ไว้สำหรับแปลโปรแกรมต้นกำเนิด STARTUP.ASP ถ้าไม่พบ STARTUP.ASX) ให้สังเกตุว่าจะต้องไม่มีเว้นวรรคระหว่าง /f กับชื่อไฟล์สคริปต์)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

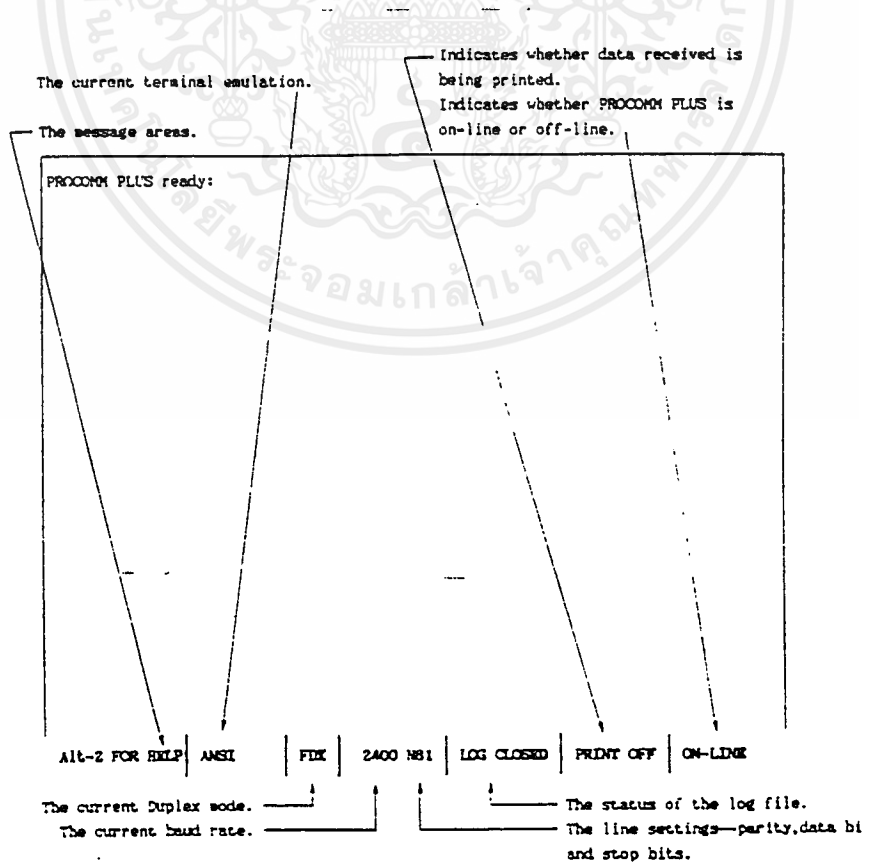
สามารถรวบรวมสวิตช์ต่างๆ เข้ามากกว่าหนึ่งสวิตช์ (โดยเรียงลำดับอย่างไรก็ได้) ไว้หลัง pcplus ได้ เช่น สามารถสตาร์ต PROCOMM PLUS ได้โดยป้อนว่า

```
pcplus /s /b /fstartup
```

ที่เครื่องหมายรับคำสั่งของดอส แล้วกดคีย์ ENTER

1.9 จอภาพโหมดเทอร์มินัล

เราสามารถป้อนคำสั่งต่างๆ ของ PROCOMM PLUS ได้ เมื่อ PROCOMM PLUS อยู่ในโหมดเทอร์มินัล



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บรรทัดล่างของจอภาพเรียกว่า *บรรทัดแสดงสถานะ (status line)* ส่วนจะมีอะไรอยู่ในบรรทัดแสดงสถานะภาพ นี้จะได้ทราบในภายหลัง

1.10 การใช้คำสั่งต่างๆของ PROCOMM PLUS

การแจ้งให้ PROCOMM PLUS ทำในสิ่งที่ต้องการได้โดยป้อนคำสั่งลงไป PROCOMM PLUS รับคำสั่งต่างๆ ได้สองวิธีคือ

1. ป้อนคำสั่งลงไปโดยตรงโดยใช้คีย์ ALT หรือ CTRL สามารป้อนคำสั่งของ PROCOMM PLUS ได้ทั้งหมดด้วยวิธีนี้ และเมื่อไรก็ได้

2. ป้อนคำสั่งได้จากบรรทัดในเมนูแบบดึงลงมา ซึ่งสามารถทำให้เมนูเกิดขึ้นที่ด้านบนของจอภาพได้ สามารถป้อนคำสั่งของ PROCOMM PLUS ด้วยวิธีนี้เกือบทั้งหมด แต่หลังจากที่เรียกบรรทัดของเมนูขึ้นมาแล้ว คำสั่งต่างๆ เลือกจากเมนูได้ด้วยคีย์เคอร์เซอร์ เมื่อกด ENTER จะเป็นการให้คำสั่งทำงาน ในขณะที่กดคีย์ ESC จะกลับไปยังเมนูที่แล้วมา (หรือโหมดเทอร์มินัล) สามารถเลือกคำสั่งในเมนูนี้ได้ด้วยการกดคีย์ตัวอักษรของชื่อคำสั่งได้เช่นกัน ถ้าเลือกเมนูย่อย (ซึ่งแสดงรายการคำสั่งต่างๆ อีกรายการหนึ่ง) ก็จำเป็นต้องกดคีย์ ALT ไว้ พร้อมกับกดคีย์อักษรตัวแรกของชื่อเมนูย่อย ถ้าใช้เม้าส์ก็สามารถใช้เม้าส์เรียกบรรทัดในเมนู และเลือกคำสั่งหรือเมนูย่อย ให้กดปุ่มเมาส์ที่โหมดเทอร์มินัลเพื่อแสดงเมนู แล้วเลื่อนแถบสว่างตรงคำสั่งที่เลือก และกดปุ่มซ้ายของเม้าส์เป็นการป้อนคำสั่งนั้น ถ้ากดปุ่มขวาของเม้าส์ในขณะที่บรรทัดในเมนูแบบดึงลงมากำลังทำงานอยู่ จะพากลับไปยังเมนูที่แล้วมา (หรือโหมดเทอร์มินัล)

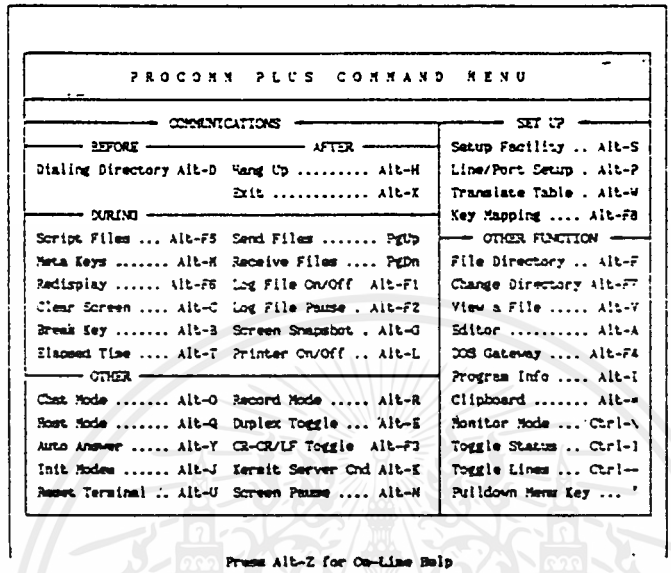
การใช้คีย์ ALT และคีย์ CTRL

ป้อนคำสั่งของ PROCOMM PLUS แบบใช้คีย์ ALT หรือคีย์ CTRL ได้โดยกดคีย์ ALT หรือคีย์ CTRL ไว้ ในขณะเดียวกันก็ให้กดคีย์ฟังก์ชัน หรือคีย์ตัวอักษรที่เหมาะสม เช่น วิธีป้อนคำสั่ง ALT-Z ซึ่งแสดงเมนูคำสั่ง (เป็นราย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การคำสั่งทั้งหมดที่กดคีย์ ALT) มีดังนี้

กดคีย์ ALT ไว้ แล้วกดคีย์ Z



รูปที่ 1.2

สามารถป้อนคำสั่งได้เกือบทั้งหมดที่เป็นรายการอยู่ในเมนูคำสั่งได้ เช่นเดียวกับที่ป้อนคำสั่ง ALT-Z โดยกดคีย์ ALT ไว้ในขณะที่กดคีย์อื่นด้วย ดังได้เห็นในเมนูแล้วว่า มีข้อยกเว้นสำหรับคีย์ PgUp หรือ PgDn (ซึ่งไม่จำเป็นต้องกดคีย์ ALT ด้วย) กับคีย์ CTRL-\ CTRL-I และ CTRL- - (ซึ่งต้องกดคีย์ CTRL ไว้)

หมายเหตุ ตลอดส่วนที่เหลือของหนังสือเล่มนี้ การกดคีย์ผสมของคีย์ ALT กับคีย์อื่น และคีย์ CRTL กับคีย์อื่น จะเขียนในลักษณะเดียวกันกับ ALT-Z

สามารถแสดงเมนูคำสั่งได้ เมื่อต้องการค้นหาคำสั่งที่จะใช้ อย่างไรก็ดีสามารถป้อนคำสั่ง โดยใช้คีย์ ALT หรือ CTRL เมื่อไรก็ได้ในโหมดเทอร์มินัล โดยไม่คำนึงว่าจะมีเมนูคำสั่งปรากฏอยู่ในจอภาพหรือไม่

หมายเหตุ เมนูคำสั่งเป็นส่วนหนึ่งของโปรแกรมอำนวยความสะดวก ในการให้ความช่วยเหลือของ PROCOMM PLUS การกดคีย์ ALT-Z จะแสดง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

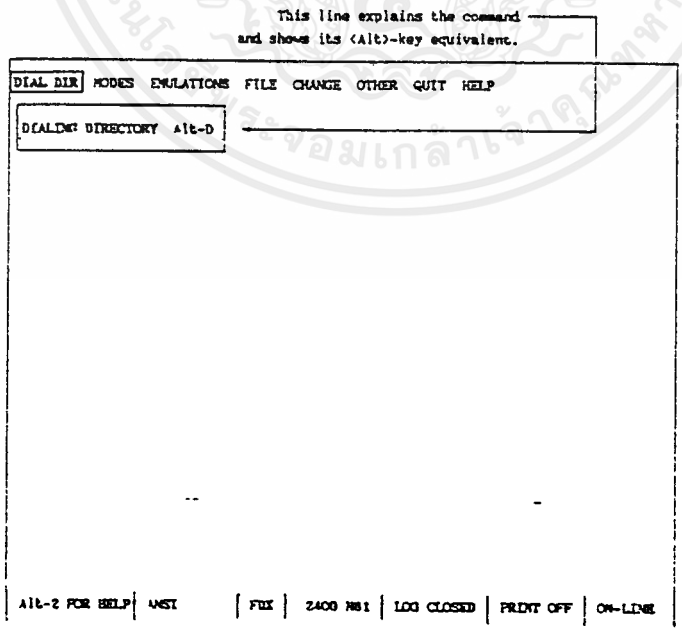
ข้อมูลเกี่ยวกับสิ่งที่กำลังทำอยู่ เช่น สมมติว่าในขณะนี้กำลังกดคีย์ ALT-Z อยู่ (โดยมีเมนูคำสั่งอยู่ในจอภาพ) แล้ว PROCOMM PLUS จะแสดงข้อมูลให้ความช่วยเหลือ เกี่ยวกับโหมดเทอร์มินัลโดยสามารถเลือกเรื่องใดๆ หรือค้นบรรทัดข้อความภายในโปรแกรมอำนวยความสะดวก ในการให้ความช่วยเหลือได้

การลบเมนูคำสั่งออกจากจอภาพ ทำได้โดยการกดคีย์ ESC

การใช้บรรทัดเมนูแบบดึงลงมา

ก่อนที่จะใช้บรรทัดเมนูแบบดึงลงมา เพื่อให้คำสั่งแก่ PROCOMM PLUS จะต้องให้บรรทัดในเมนูนั้นทำงาน ซึ่งทำได้โดยใช้คีย์บอร์ด หรือใช้เม้าส์ก็ได้

การให้บรรทัดในเมนูทำงานจากคีย์บอร์ด ให้กดคีย์ ในคีย์บอร์ดส่วนใหญ่ คีย์นี้คือคีย์ที่มีเครื่องหมาย tilde (~) อยู่ข้างบน



รูปที่ 1.3

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ในการทำให้บรรทัดในเมนูทำงาน ด้วยเม้าส์ให้กดปุ่มเม้าส์ปุ่มใดก็ได้

สามารถเลือกคำสั่งจากบรรทัดในเมนูได้สามวิธี คือ

1. โดยกดคีย์ ALT ไว้ แล้วกดคีย์ตัวอักษรตัวแรกของเมนูย่อยหรือของคำสั่ง ถ้าจะเลือกเมนูย่อย (อย่างเช่น FILE) ก็สามารถเลือกคำสั่ง ได้โดยกดคีย์อักษรตัวแรกของคำสั่ง และกดคีย์ ENTER

2. โดยทำแถบสว่างที่คำสั่ง (ด้วยคีย์ลูกศรขวาและซ้าย) และกดคีย์ ENTER ถ้าเลือกเมนูย่อยให้ใช้คีย์ลูกศรขึ้นและลง เพื่อทำให้แถบสว่างที่คำสั่งแล้วกดคีย์ ENTER

3. โดยการทำแถบสว่างที่คำสั่ง (ด้วยเม้าส์) แล้วกดปุ่มซ้ายของเม้าส์

แตกต่างไปจากบรรทัดในเมนูที่คล้ายๆ กันในโปรแกรมอื่นๆ

บรรทัดในเมนูของ PROCOMM PLUS จะหายไปในทันทีที่เลือกคำสั่ง สำหรับการไม่ให้บรรทัดในเมนูทำงานโดยไม่ต้องเลือกคำสั่ง ให้กดคีย์ ESC เราจะทราบวิธีใช้บรรทัดในเมนูทันทีที่กำหนดพอร์ต COM ที่โมเด็มใช้

ถ้าต้องการใช้คีย์อื่นนอกเหนือจากคีย์ เพื่อเรียกบรรทัดในเมนู ก็สามารถกำหนดคีย์นั้นได้โดยใช้โปรแกรมอำนวยความสะดวก Setup เช่นเดียวกับถ้าไม่ต้องการใช้บรรทัดในเมนูเลย ก็สามารถทำให้บรรทัดในเมนู ไม่ทำงานได้โดยใช้โปรแกรมอำนวยความสะดวก Setup

1.11 การกำหนดพอร์ต COM

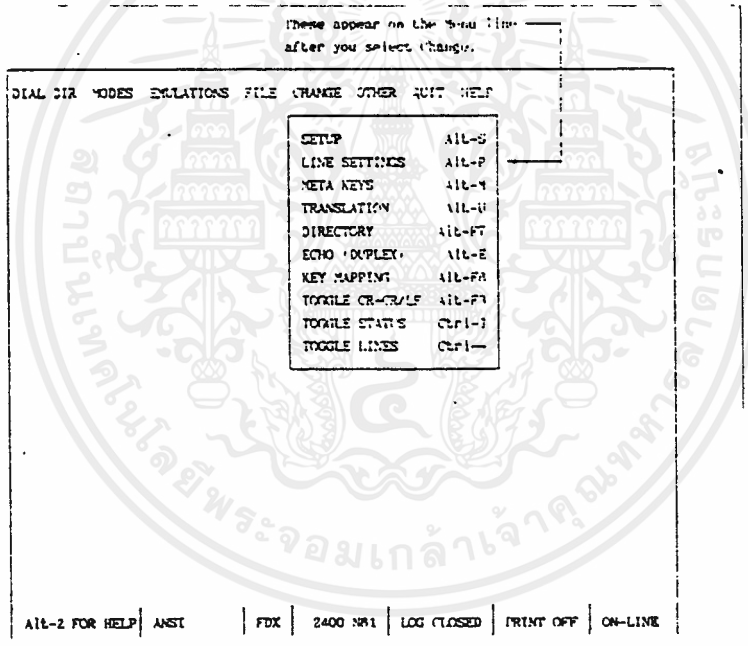
ก่อนที่จะใช้สายโทรศัพท์ (on-line) จำเป็นต้องบอก PROCOMM PLUS ว่าโมเด็มใช้พอร์ต COM ใดก่อน แม้ว่า PCINSTAL จะได้เลือกพอร์ต COM ให้กับ PROCOMM PLUS ไปแล้วในระหว่างติดตั้งแล้วก็ตาม พอร์ต COM เป็นตัวแปรการติดต่อสื่อสารที่ต้องเปลี่ยนอยู่เสมอ เพราะฉะนั้นเราจะใช้ตัวแปรนี้ เพื่อให้บรรทัดในเมนูแบบดึงลงมาทำงาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เราจะทำเช่นว่านี้ได้โดยกดคีย์ ALT-P แต่ในที่นี้เราจะใช้บรรทัด ในเมนูเพื่อเรียนรู้วิธีการทำงาน เริ่มต้นด้วยการให้บรรทัดในเมนูทำงานโดยการกดคีย์

ในการกำหนกพอร์ต COM จำเป็นต้องเลือก Change เมนูย่อยก่อน การเลือกคำสั่งนี้ให้กดคีย์ ALT-C หรือจะใช้เม้าส์โดยใช้คีย์เคอร์เซอร์ทำแถบสว่างที่คำสั่งก็ได้

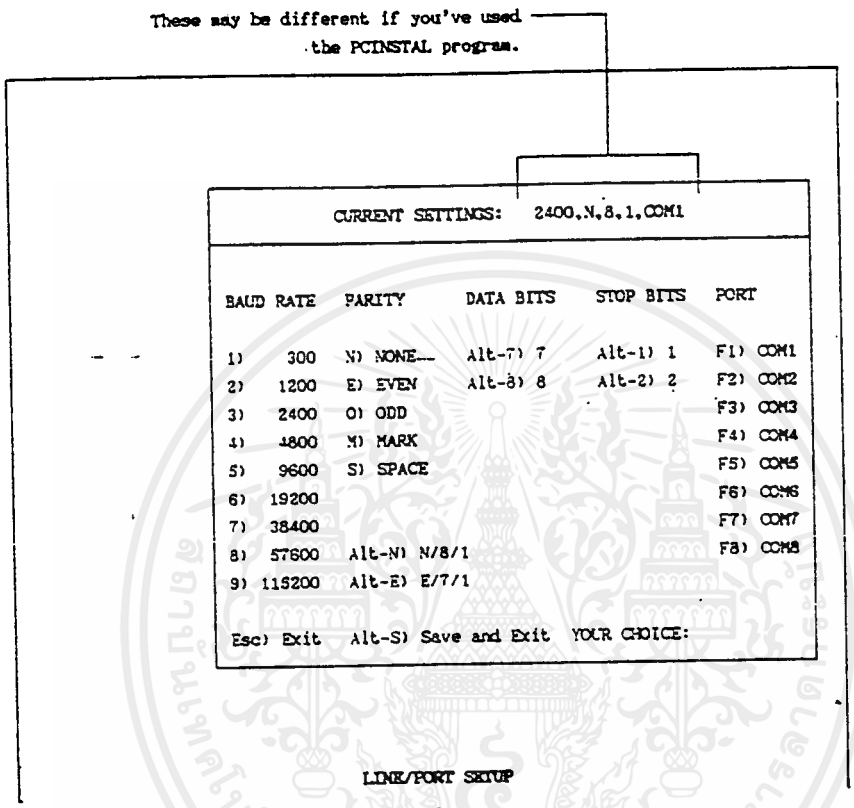
กดคีย์ลูกศรขวา 4 ครั้ง (หรือกดคีย์ ALT-C) เพื่อเลื่อนแถบสว่างไปที่คำสั่ง CHANGE



รูปที่ 1.4

ในบรรทัดเมนูนี้ให้เลือกคำสั่ง Line Settings โดกดคีย์ L คีย์ ENTER เพื่อเลือกคำสั่ง Line Settings PROCOMM PLUS จะแสดงจอภาพ ของคำสั่ง Line Setting ให้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 1.5

ถ้าพอร์ต COM ที่กำลังใช้อยู่เป็นคนละพอร์ตกับที่ปรากฏอยู่ที่ด้านบน ของจอภาพแล้ว ให้กดคีย์ฟังก์ชัน F1 ถึง F8 ที่ปรากฏอยู่ต่อจากพอร์ต COM ที่โทเด็มกำลังใช้อยู่ พอร์ต COM ที่เลือกไว้จะไปปรากฏที่ด้านบนของจอภาพ

การบันทึกการเปลี่ยนแปลงที่ทำไว้ และปิดจอภาพของ Line Settings ให้กดคีย์ ALT-S PROCOMM PLUS จะลบจอภาพออกไป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1.12 การออกจาก PROCOMM PLUS

การออกจาก PROCOMM PLUS และกลับไปยัง DOS ให้กดคีย์ ALT-X PROCOMM PLUS จะขอคำยืนยันว่าต้องการเลิก และกลับไปยัง DOS กดคีย์ Y เพื่อยืนยันว่าต้องการจะเลิก

ถ้า PROCOMM PLUS ถามว่าต้องการวางสาย (HANGUP LINE) หรือไม่ โมเด็มจะบอก PROCOMM PLUS ว่าโมเด็มต่ออยู่กับคอมพิวเตอร์อีกเครื่องหนึ่งแล้ว! เหตุอาจจะเป็นเพราะว่าตั้งสวิตช์ CD ของโมเด็มไว้ในระหว่างการติดตั้งไม่ถูกต้อง ให้ดูเรื่อง"การติดตั้งโมเด็ม" ในตอนท้ายของบทนี้ อีกเหตุผลหนึ่งก็อาจเป็นเพราะว่า ไปเลือกพอร์ตอนุกรมที่ไม่ได้ต่ออุปกรณ์อะไรไว้เลย

เมื่อติดตั้ง PROCOMM PLUS เสร็จเรียบร้อยแล้ว ก็สามารถใช้สายโทรศัพท์ (on-line) ได้

1.13 ถ้ามีปัญหา

1. ตรวจดูให้แน่ใจว่า คอมพิวเตอร์มีลักษณะตรงตามความต้องการ ดังคำอธิบายไว้ในข้อ "สิ่งที่ต้องการในการใช้ PROCOMM PLUS" ในบทนี้แล้ว

2. ตรวจดูให้แน่ใจว่าคอมพิวเตอร์ โมเด็มและสายเคเบิลต่างๆ ที่ต่อไว้ทำงานถูกต้อง

3. ถ้าไม่สามารถหาวิธีแก้ปัญหา ที่เกิดขึ้นด้วยการอ่านปริญญานิพนธ์เล่มนี้ ให้ลองอ่านภาคผนวกที่ท้ายเล่ม ในภาคผนวกนี้จะมีข้อมูลเพิ่มเติมที่สามารถช่วยแก้ปัญหาเฉพาะอย่างได้

4. ตรวจดูให้แน่ใจว่า ไม่ได้รันซอฟต์แวร์ที่เข้ากันไม่ได้กับ PROCOMM PLUS ในภาคผนวก หัวข้อ "การตรวจสอบซอฟต์แวร์ที่ไม่เข้ากัน" ซึ่งจะอธิบายวิธีพิจารณาว่าซอฟต์แวร์ที่ใช้จะมีปัญหาหรือไม่ไว้ให้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่2

แนะนำให้รู้จักกับระบบไมโครคอมพิวเตอร์ส่วนบุคคลของไอบีเอ็ม

(OVERVIEW OF THE IBM PERSONAL MICROCOMPUTER SYSTEM)

2.1 แนะนำ

ไมโครคอมพิวเตอร์ส่วนบุคคลของไอบีเอ็ม(IBM PC) ปัจจุบันนี้ได้รับความนิยมใช้กันแพร่หลาย และนิยมเรียกกันว่า พีซี (PC) เป็นระบบไมโครคอมพิวเตอร์ที่มีระบบที่ดี และสามารถนำไปใช้งานได้อย่างกว้างขวาง ตั้งแต่ใช้ในบ้าน ในที่ทำงาน ในห้องเรียน ในห้องทดลอง ฯลฯ เป็นระบบคอมพิวเตอร์ที่มีสถาปัตยกรรมภายในแบบ 16-บิต สามารถกำหนดแอดเดรสของหน่วยความจำได้จำนวนมาก และที่สำคัญก็คือบริษัทไอบีเอ็มได้เปิดเผยรายละเอียดในการออกแบบให้เป็นสาธารณะ ดังนั้นเครื่องของไอบีเอ็มจึงสามารถที่จะขยายระบบได้โดยง่าย ในบทที่ 1 ถึง 4 จะกล่าวถึงรายละเอียดต่าง ๆ ที่จะใช้ในการออกแบบเพื่อเชื่อมต่อกับระบบของไอบีเอ็ม พีซี สำหรับวิศวกร นักวิทยาศาสตร์ นักไมโครคอมพิวเตอร์ที่มีความสนใจ และสำหรับผู้สนใจทั่วไป ที่มีพื้นความรู้ทางด้านอิเลคทรอนิคส์ ซึ่งบุคคลเหล่านี้สามารถที่จะเชื่อมต่ออุปกรณ์ ที่เขาออกแบบเข้ากับระบบไมโครคอมพิวเตอร์ของไอบีเอ็ม พีซี ได้

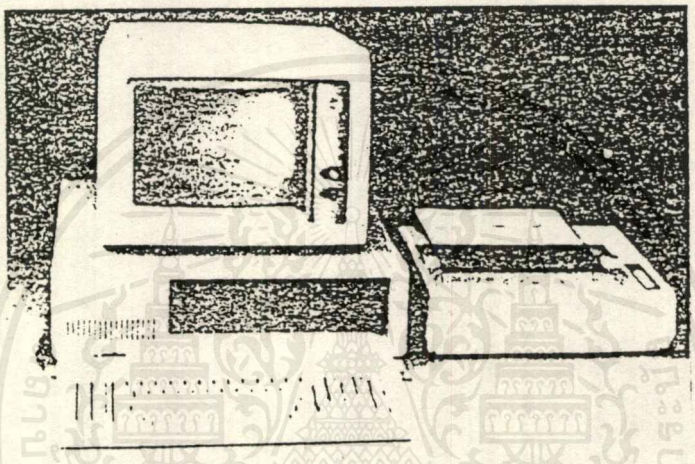
ส่วนประกอบพื้นฐานของไอบีเอ็มจะประกอบไปด้วย SYSTEM UNIT, แป้นพิมพ์,จอแสดงผล,และเครื่องพิมพ์ ภายใน SYSTEM UNIT จะประกอบไปด้วย ไมโครโปรเซสเซอร์ขนาด 16-บิต หน่วยความจำ และตัวขับฟล็อบปี้ดิส

2.2 หน่วยหลัก(SYSTEM UNIT)

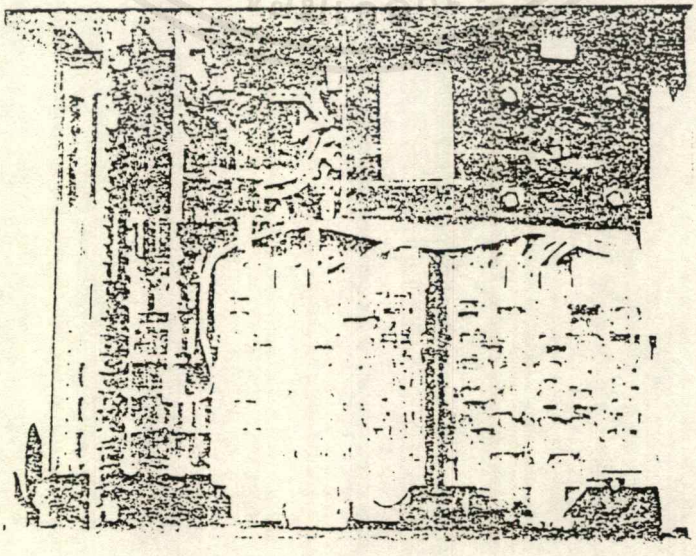
หน่วยหลัก จะประกอบไปด้วย ไมโครโปรเซสเซอร์ขนาด 16-บิต,

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หน่วยความจำชนิดอ่านได้อย่างเดียว(ROM-Read Only Memory),หน่วยความจำสำหรับผู้ใช้ชนิดอ่าน-เขียนได้(RAM-Random Access Memory),แหล่งจ่ายกำลัง, ลำโพงสำหรับการใช้งานทางด้านเสียงและดนตรี,พร้อมทั้งสล้อตขยายระบบ จำนวน 5 สล้อต ตัวขับฟล้อปปี้ดิสสามารถที่จะติดตั้งเข้าไปที่หน่วยหลักได้ 2 ตัว ดังรูปที่ 2.2 ซึ่งเป็นภาพถ่ายของหน่วยหลัก



รูปที่ 2.1 เครื่องไมโครคอมพิวเตอร์ส่วนบุคคลของไอบีเอ็ม



รูปที่ 2.2 ภาพถ่ายของหน่วยหลัก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.2.1 บอร์ดโปรเซสเซอร์(The Processor Board)

บอร์ดโปรเซสเซอร์ จะประกอบไปด้วยไมโครโปรเซสเซอร์ 8088 ซึ่งมีสถาปัตยกรรมภายในเป็นแบบ 16-บิต,หน่วยความจำรอมและแรม,แผ่นการ์ด I/O สำหรับแป้นพิมพ์และสำหรับจอแสดงผล,และการ์ดสำหรับควบคุมอื่นๆ ส่วนประกอบที่มีความสำคัญและน่าสนใจสำหรับการออกแบบ เพื่อเชื่อมต่อกับระบบ คือสล้อตขยายระบบ ซึ่งมีอยู่ 5 สล้อต(สำหรับเครื่องต้นกำเนิดจากไอบีเอ็ม) ซึ่งจะยอมให้เราสามารถที่จะเชื่อมต่ออุปกรณ์ที่เราออกแบบเข้ากับระบบได้ โดยผ่านทางสล้อตดังกล่าว

2.2.2 แหล่งจ่ายกำลัง(The Power Supply)

แหล่งจ่ายกำลังจะอยู่ภายในหน่วยหลัก ที่ด้านบนของ รูปที่ 2.2 ทำหน้าที่จ่ายกำลังไฟฟ้ากระแสตรงให้กับระบบ และจ่ายกำลังไฟฟ้ากระแสสลับให้กับจอแสดงผลด้วย แหล่งจ่ายกำลังจะมีแรงดันไฟตรงอยู่ 4 ชุด คือ +5V,-5V, +12V, และ -12V ซึ่งแหล่งจ่ายทั้ง 4 จะต่อให้เราสามารถใช้งานได้ที่สล้อตขยายระบบโดยตรง

2.2.3 ตัวขับฟล้อปปี้ดิส(The Diskette Drive)

ที่หน่วยหลักสามารถที่จะบรรจุ ตัวขับฟล้อปปี้ดิส ได้จำนวน 2 ตัว โดยใช้ดิสขนาด 5 1/4 นิ้ว มีความหนาแน่นที่แนะนำจากไอบีเอ็ม 2 ขนาด คือ หน้าเดียว-ความหนาแน่นเดี่ยว(Single Side-Single Density) ซึ่งมีความจุทั้งหมดเท่ากับ 180 กิโลไบต์ และแบบ สองหน้า-ความหนาแน่นคู่(Double Side -Double Density) ซึ่งมีความจุทั้งหมดเท่ากับ 360 กิโลไบต์

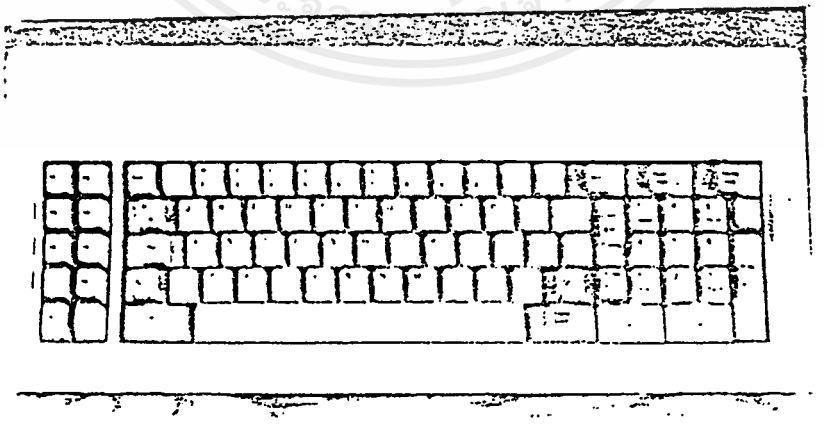
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.2.4 ลำโพง(Speaker)

จะติดตั้งที่ตำแหน่งด้านซ้ายของหน่วยหลัก จะเป็นลำโพงขนาดเล็ก เสียงของลำโพงจะควบคุมได้จากโปรแกรม โดยมากจะใช้เป็นสัญญาณเตือน (Alarm) หรือเป็นเสียงสำหรับการเล่นเกมส์

2.3 แป้นพิมพ์ของระบบ(THE SYSTEM KEYBOARD)

แป้นพิมพ์จากไอบีเอ็ม พีซี จะมีขนาด 83 คีย์ มีคีย์พิเศษต่างๆ เช่น ฟังค์ชั่นคีย์ ทางด้านซ้ายจะเป็นคีย์ทำหน้าที่ได้สองอย่าง คือคีย์ของตัวเลขและคีย์ที่ใช้ควบคุมตำแหน่งของเคอร์เซอร์(Cursor Control) แป้นพิมพ์จะเชื่อมต่อกับระบบโดยทางสายเคเบิลขนาด 4 เส้น คู่แรกเป็นแหล่งจ่ายไฟตรง อีกคู่เป็นสายสัญญาณ ตัวแป้นพิมพ์จะส่งรหัสที่ได้จากการกวาด(Scan Code) ขนาด 8-บิต ที่เกิดจากการกดคีย์,ปล่อยคีย์ ถ้ามีการกดคีย์ค้าง มันจะทำการส่งรหัสของคีย์นั้นแล้วหยุดชั่วขณะหนึ่งแล้วส่งรหัสดังกล่าวซ้ำในอัตราคงที่จนกว่าจะปล่อยคีย์



รูปที่ 2.3 แป้นพิมพ์แบบ 83 คีย์ ของไอบีเอ็ม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.4 เครื่องพิมพ์ของระบบ(THE SYSTEM PRINTER)

เครื่องพิมพ์ที่แนะนำจากไอบีเอ็ม มีความเร็วในการพิมพ์ 80 ตัวอักษรต่อนาที ชนิด ดอต-แมททริก(Dot-matrix) ทำการเชื่อมต่อกับระบบโดยผ่านทางพอร์ต I/O แบบขนาน ใช้มาตรฐานของ CENTORNIX สามารถทำการพิมพ์สองทิศทางได้(ทั้งตอนลากหัวพิมพ์ไปและตอนลากหัวพิมพ์กลับ) มีรูปแบบตัวอักษร(Font)ให้เลือก 12 รูปแบบ ในหนึ่งบรรทัดพิมพ์ได้สูงสุด 132 ตัวอักษร ระยะห่างระหว่างบรรทัดเพิ่มได้ทีละ 1/6 นิ้ว

2.4.1 โหมดของการพิมพ์(Printing Mode Supported)

เราสามารถเปลี่ยนโหมดและฟอนต์ ได้โดยการควบคุมจากโปรแกรม ในโหมดปรกติ จะเป็นการพิมพ์ 10 ตัวอักษรต่อนิ้ว 80 ตัวอักษรต่อบรรทัด ในโหมดตัวอักษรขยาย(Enlarged Mode) จะพิมพ์ 5 ตัวอักษรต่อนิ้ว 40 ตัวอักษรต่อบรรทัด ในโหมดตัวอักษรบีบ(Condensed Mode) จะพิมพ์ 16.5 ตัวอักษรต่อนิ้ว และ 132 ตัวอักษรต่อบรรทัด และมีโหมดพิเศษระหว่างตัวขยายกับตัวบีบ เช่น 8.25 ตัวอักษรต่อนิ้ว และ 66 ตัวอักษรต่อบรรทัด

2.4.2 รหัสของเครื่องพิมพ์(Printer Charecter Set)

เครื่องพิมพ์(ของไอบีเอ็ม)จะใช้รหัสแอสกี้(ASCII) 96 ตัว มีอักขระภายใน 9 ตัว ซึ่งเลือกได้จากการตั้งสวิชภายในเครื่องพิมพ์ รหัส 96 ตัวดังกล่าวจะเป็นส่วนหนึ่ง(Sub-Set)ของทั้งหมด 256 รหัสที่แสดงออกมาได้ที่จอแสดงผล จะเห็นได้ว่ามีตัวอักษรจำนวนมากที่แสดงออกมาที่จอแสดงผลได้ แต่จะทำการพิมพ์ออกที่เครื่องพิมพ์ไม่ได้

เครื่องพิมพ์ของไอบีเอ็มจะคล้ายกับเครื่องพิมพ์ของ EPSON MX-80 ซึ่งจะสามารถนำมาใช้แทนกันได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.5 จอแสดงผลแบบสองสี(THE MONOCHROME DISPLAY)

จอแสดงผลที่แนะนำจากไอบีเอ็มเป็นจอแบบสองสี หรือที่นิยมเรียกว่าจอแบบโมโนโครม มีความละเอียดสูง ซึ่งให้มาพร้อมกับอะแดปเตอร์แสดงผลที่ใช้กับเครื่องพิมพ์ได้ด้วย จะเสียบอยู่ที่สล้อตขยายระบบ ในระบบหลัก(STSTEM UNIT) สัญญาณที่เชื่อมต่อกับตัวแสดงผลจะทำการขับโดยตรงซึ่งจะมีการเชื่อมต่อทางแนวนอน,แนวตั้ง,สัญญาณภาพ(Video),และความเข้มของสัญญาณภาพ หน่วยแสดงผลจะเป็นจอสีเขียว ฉาบที่จอด้วยสารฟอสเฟอร์ มีความละเอียด 720x350 จุด ขนาดของจอตามแนวเส้นทะแยงมุมเท่ากับ 11.5 นิ้ว จอจะกวาดทางแนวตั้งที่ความถี่ 50 Hz ทางแนวนอนที่ความถี่ประมาณ 17 KHz สัญญาณภาพจะเลื่อนออกมา(Shift)ที่อัตรา 16.25 MHz ตัวอักษรในเท็กซ์โหมดมีขนาด 9x14 จุด

เราสามารถเปลี่ยนจอและอะแดปเตอร์แสดงผลเป็นแบบอื่น ๆ ได้โดยการนำมาเสียบเข้าไปที่สล้อตขยายระบบ

2.6 การ์ดอื่น ๆ ที่แนะนำจากไอบีเอ็ม(THE IMB PC FEATURE CARD)

ไอบีเอ็มได้แนะนำการ์ดอื่น ๆ นอกเหนือจากที่ได้กล่าวมาแล้ว เพื่อเพิ่มความสามารถของระบบ เช่น การขยายหน่วยความจำ,การต่ออุปกรณ์เพิ่ม, และการเชื่อมต่อมาตรฐานต่าง ๆ

2.6.1 อะแดปเตอร์ควบคุมฟล้อปปี้ดิส(Diskette Drive Attachment Adapter Card)

ภายในระบบหลักของไอบีเอ็ม จะติดตั้งตัวขับฟล้อปปี้ดิสได้สูงสุดสองตัว ซึ่งใช้แผ่นฟล้อปปี้ดิสขนาด 5 1/4 นิ้ว และจะติดตั้งเพิ่มภายนอกได้อีกสองตัว โดยใช้การ์ดเพียงการ์ดเดียว ซึ่งจะใช้ได้กับแผ่นฟล้อปปี้ดิสทั้งขนาดหน้าเดียวและสองหน้า

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.6.2 การ์ดพอร์ตของเครื่องพิมพ์แบบขนาน(Parallel Printer Port Attachment Card)

การ์ดนี้จะยอมให้มีการติดตั้งเครื่องพิมพ์ ที่เป็นมาตรฐานทางอุตสาหกรรม ที่ใช้มาตรฐานการเชื่อมต่อแบบขนานของ CENTRONIX

2.6.3 การ์ดพอร์ตอนุกรม(Serial Port Attachment Card)

การ์ดนี้จะเป็นอนุกรมแบบ ASYNC พร้อมกับการเชื่อมต่อทางไฟฟ้ากับ RS-232-C ซึ่งจะทำให้ต่อกับโมเด็มได้ เพื่อใช้ส่งข้อมูลผ่านทางคู่สายโทรศัพท์ มีอัตราการส่งผ่านข้อมูลจาก 50 ถึง 9600 Baud ขนาดเฟรมของข้อมูลสามารถที่จะโปรแกรมได้จาก 5,6,7 หรือ 8-บิต จำนวนของบิตสิ้นสุด (Stop Bit)จะโปรแกรมได้จาก 1 ถึง 1/2 ถึง 2 สามารถเลือกได้ว่าจะให้เป็นพาริตี้คู่ หรือคี่ หรือไม่มีพาริตี้ ภายในการ์ดจะมีจุดต่อ(Jumper)ให้เลือกสำหรับการเชื่อมต่อทางไฟฟ้าของ Current-loop ให้ด้วย

2.6.4 การ์ดควบคุมเกมส์(Game Control Adapter Card)

การ์ดนี้จะใช้สำหรับต่อกับ Joysticks หรือ Paddle เพื่อเล่นเกมส์ ภายในจะมี 4 อินพุต ซึ่งจะใช้ตรวจจับ(Sense)ค่าความต้านทานของตัวต้านทานของ Joysticks หรือ Paddle และอีก 4 อินพุตจะใช้สำหรับปุ่มกดของ Joysticks หรือ Paddle

2.6.5 การ์ดแสดงผลแบบสี(Color Graphics Display Adapter Card)

การ์ดนี้จะทำให้เราสามารถที่จะต่อไอบีเอ็ม พีซี เข้ากับเครื่องรับโทรทัศน์ได้ จะใช้ได้ทั้งกราฟฟิกโหมด และเท็กซ์โหมด ทั้งสีและขาวดำ ในเท็กซ์โหมดจะใช้ได้ทั้งแบบ 40 ตัวอักษรต่อบรรทัด 25 บรรทัดต่อ 1 จอ (40x25)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

และ 80x25 ในเท็กซ์โหมด ตัวอักษรจะมีขนาด 8x8 จุด จะเป็นสีขาวดำ หรือสีก็ได้ โดยจะมีสีพื้นหลังตัวอักษร(Background)ได้ 8 สี และสีของตัวอักษร (Foreground)ได้ 16 สี ตัวอักษรในเท็กซ์โหมดทำให้กระพริบได้ การแสดงผลจะทำแบบหลายหน้า(Multiple Page)ได้ เช่น มีหน้าขนาด 80x25 ได้ 4 หน้า และมี 40x25 ได้อีก 8 หน้า

ในกราฟฟิกโหมด จะมีอยู่สองโหมดคือ 320x200 จุด 4 สี และ 640x200 เป็นสีขาวดำ ในโหมด 320x200 แต่ละจุดจะโปรแกรมได้ 4 สี โดยมีสีให้เลือกสองกลุ่ม ในโหมด 640x200 เป็นได้เพียงสีขาวดำได้เท่านั้น ทั้งนี้เพราะว่า หน่วยความจำชนิดแรมขนาด 16 กิโลไบต์ภายในการ์ดจะใช้แทนการสว่าง หรือมืดได้เท่านั้น จะไม่มีหน่วยความจำเหลือพอที่จะแทนค่าของสีได้ และการ์ดนี้จะใช้ร่วมกับปากกาแสง(Light Pen)ได้

ภายในการ์ดจะมีหน่วยความจำชนิดรอม ซึ่งจะบรรจุรหัสการแสดงผลของเท็กซ์โหมด จำนวน 256 ตัว

2.6.6 การ์ดแสดงผลขาวดำและพอร์ตเครื่องพิมพ์แบบขนาน(Monocrhome Display Adapter and Parallel Printer Port Adapter Card)

การ์ดนี้จะใช้กับเครื่องพิมพ์และจอขาวดำได้ทั้งสองอย่าง ในส่วนของการแสดงผลจะมีเฉพาะโหมด 80x25 เท่านั้น กลุ่มของตัวอักษรจะเหมือนกับจอสี ตัวอักษรจะมีขนาด 9x14 จุด แต่ละตัวอักษรจะทำให้สว่างมาก (Highlighted)ได้ มีขีดเส้นได้ มีการกระพริบได้ และกลับขาวเป็นดำ (Revrese)ได้ การ์ดนี้จะใช้ร่วมกับปากกาแสงไม่ได้

2.6.7 การ์ดโปรโต้ทาย(Prototyping Card)

จะเป็นการ์ดเปล่า ใช้สำหรับให้ผู้ใช้สามารถที่จะต่ออุปกรณ์ต่าง ๆ ลงบนการ์ด เพื่อเชื่อมต่อกับระบบของไอบีเอ็ม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.6.8 การ์ด SDLC(SDLC Adapter Card)

การ์ดนี้จะใช้กับการสื่อสารชนิดใหม่ที่แนะนำโดยไอบีเอ็ม ซึ่งทั่ว ๆ ไปจะใช้กับกับการต่อเทอร์มินอล(Terminal) และคอมพิวเตอร์เข้าด้วยกันโดยผ่านทางคู่สายโทรศัพท์ SDLC ย่อมาจาก Synchronous Data Line Control การ์ดนี้จะต้องใช้กับโมเด็มชนิดพิเศษ ถ้าหากต้องการใช้กับคู่สายโทรศัพท์

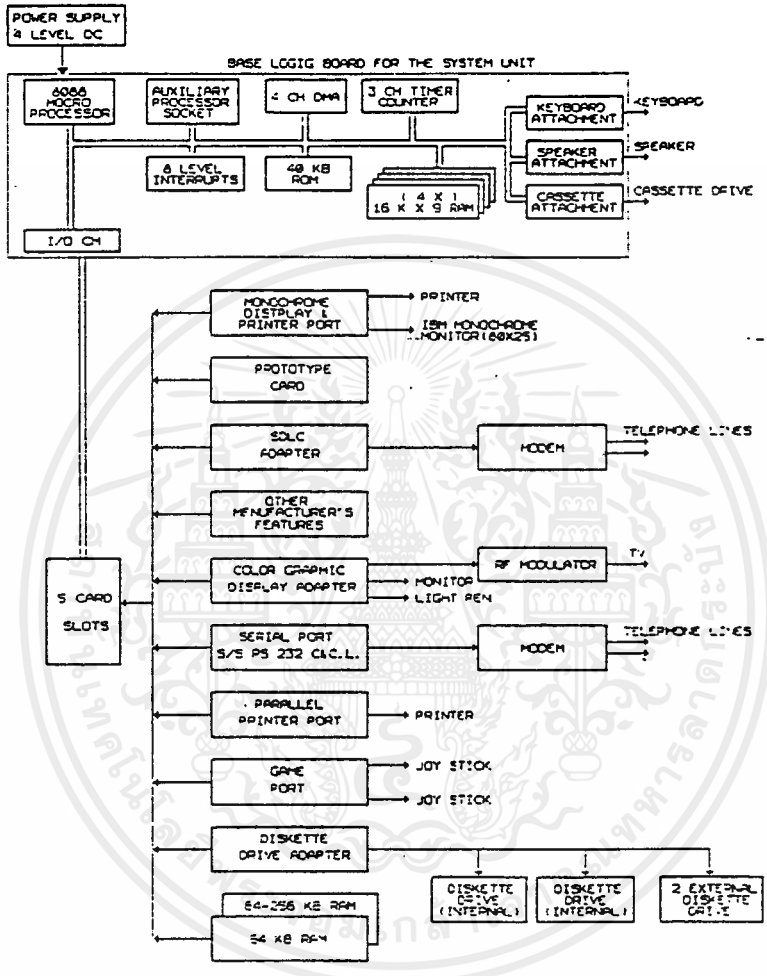
2.6.9 การ์ดขยายหน่วยความจำ(Memory Expansion Card)

ไอบีเอ็มแนะนำการขยายหน่วยความจำสองแบบคือ แบบ 64 กิโลไบต์ หรือ 64 ถึง 256 กิโลไบต์ ทั้งสองแบบจะเสียบลงไปที่สล้อตขยายระบบภายในการ์ดจะมี DIP สวิช สำหรับการตั้งแอดเดรสของหน่วยความจำ

2.6.10 การ์ดและอุปกรณ์อื่น ๆ(Other Adapters and Devices)

การ์ดต่าง ๆ ที่ใช้ได้กับไอบีเอ็ม พีซี ไม่ได้จำกัดว่าจะต้องเป็นของไอบีเอ็มเท่านั้น มีบริษัทจำนวนมากได้ผลิตการ์ดสำหรับใช้กับไอบีเอ็มออกมาจำหน่าย ซึ่งคุณจะทราบได้จากวารสารเกี่ยวกับคอมพิวเตอร์ทั้งหลายตามท้องตลาดจะมีการ์ดใหม่ ๆ ออกมามากมายให้คุณได้เลือกใช้ รูปที่ 2.4 เป็นการแสดงให้เห็นเฉพาะที่แนะนำมาจากไอบีเอ็มเท่านั้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.4 บล็อกไดอะแกรมของไอบีเอ็ม พีซี และการต่อระบบเพิ่มเติม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 3

การเชื่อมต่อสัญญาณอนาลอกกับคอมพิวเตอร์ และการควบคุมในทางอุตสาหกรรม (ANALOG INTERFACING AND INDUSTRIAL CONTROL)

3.1 แนะนำ

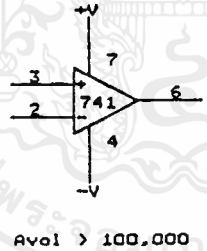
ในการที่จะใช้ไมโครคอมพิวเตอร์ควบคุมเครื่องจักรในโรงงานอุตสาหกรรม,เครื่องมือทางการแพทย์(medical instruments) หรือควบคุมยานยนต์ต่าง ๆ เราจำเป็นจะต้องรู้ค่าของ ตัวแปรต่าง ๆ เช่น ความดัน,อุณหภูมิ, อัตราการไหล(ของอากาศ ของเหลว ฯลฯ) ในการที่เราจะได้การแทนตัวแปรต่าง ๆ เหล่านี้ มาเป็นแรงดัน(หรือกระแส) จะกระทำได้หลายวิธี และเพื่อจะให้การควบคุมด้วยคอมพิวเตอร์ทำได้ เราจะต้องมีการแปลงสัญญาณ อนาลอกไปเป็นสัญญาณดิจิตอลซึ่งคอมพิวเตอร์สามารถที่จะเข้าใจได้

อันดับแรกที่สุดที่เราจะเกี่ยวข้องด้วยก็คือ เซนเซอร์ (Sensor) ซึ่งจะเปลี่ยนค่าทางกายภาพของความดัน(physical pressure),อุณหภูมิ หรือตัวแปรอื่น ๆ ไปเป็นอัตราส่วนของแรงดัน(หรือกระแส) และโดยส่วนมากสัญญาณทางไฟฟ้า จากเซนเซอร์เหล่านี้จะมีค่าต่ำมาก จึงอาจจะต้องมีการนำมาขยายให้โตขึ้นเสียก่อน และอาจจะจำเป็นที่จะต้องมีการกรอง(Filter) เอาเฉพาะสัญญาณที่เราต้องการเท่านั้น การขยายสัญญาณส่วนมากจะใช้วงจรออป-แอมพ์ และในลำดับสุดท้าย ก็คือการแปลงสัญญาณจากภายขยายเหล่านี้ ไปเป็นสัญญาณดิจิตอล โดยใช้ตัวแปลงอนาลอกไปเป็นดิจิตอล(A/D converter) ในบทนี้จะกล่าวถึงการแนะนำให้รู้จักกับบางวงจรที่ใช้ตัวแปลง A/D และกล่าวถึงการนำส่วนต่าง ๆ เหล่านี้มารวมเข้าด้วยกัน เพื่อใช้ไมโครคอมพิวเตอร์ควบคุมให้เป็นระบบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.2 แนะนำเกี่ยวกับคุณสมบัติของวงจร OP-AMP

รูป 3.1 แสดงวงจรและสัญญลักษณ์ของออปแอมพ์ ซึ่งสัญญลักษณ์ต่าง ๆ ที่จำเป็นเบื้องต้นที่เราจะต้องรู้คือ ขา +V และขา -V จะเป็นขาที่ต่อกับแหล่งจ่ายไฟโดยมากจะเป็น +15 และ -15 V หรืออาจจะเป็น +12 V และ -12V ออปแอมพ์ มีขาสัญญาณเข้า 2 ขา วงจรขยายสัญญาณของออปแอมพ์จะทำการขยายสัญญาณความแตกต่างระหว่างขาอินพุตทั้งสองนี้ โดยจะมีอัตราขยายประมาณ 100,000 เท่า(หรือมากกว่านี้) ขาอินพุต - เรียกว่า inverting อินพุต และขาอินพุต + เรียกว่า non-inverting อินพุต เครื่องหมาย + และ - ของทั้งสองอินพุตจะไม่เกี่ยวข้องกับแหล่งจ่ายไฟ +V และ -V แต่จะเกี่ยวข้องกับเฟสของสัญญาณที่จ่ายเข้ามาที่อินพุตทั้งสองแล้วได้ผลออกมาที่เอ้าต์พุต ตัวอย่างเช่น ขาอินพุต + มีค่าบวกมากกว่าขาอินพุต - แรงดันเอ้าต์พุต จะเป็นไปในทางบวกซึ่งเป็นการ in-phase(เฟสเดียวกันกับอินพุต)



รูป 3.1 OP-AMP SYMBOL

3.3 วงจร OP-AMP และการประยุกต์ใช้งาน

3.3.1 วงจรเปรียบเทียบที่ใช้ OP-AMP

จากที่กล่าวมาแล้วว่า ออปแอมพ์จะขยายสัญญาณความแตกต่างระหว่างขาอินพุตทั้งสองด้วยค่าอัตราขยายที่สูงมาก สมมุติว่าเรามีแหล่งจ่ายไฟให้ออปแอมพ์ ขนาด +15 V และ -15 V ถ้าเราต่อขา inverting อินพุต ลง

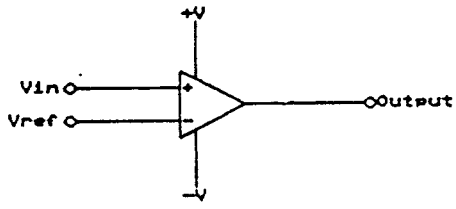
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กราวด์ และจ่ายแรงดัน +0.01 Vdc เข้าที่ขา non-inverting อินพุต ออปแอมพ์จะพยายามขยายสัญญาณสัญญาณนี้ด้วยอัตราการขยายที่สูงมาก (100,000 เท่าขึ้นไป) ซึ่งในสาเหตุดังกล่าวมาเราจะได้ ค่าอัตราการขยายคูณกับแรงดันความแตกต่างของอินพุตทั้งสองประมาณ 0.015 x 100,000 = 100 V แต่แหล่งจ่ายไฟของออปแอมพ์จะจ่ายแรงดันเอ้าต์พุต ได้อย่างมากที่สุดที่ต่ำกว่าแรงดันแหล่งจ่ายเล็กน้อย(ต่ำกว่าประมาณ 1-2 V) ดังนั้นในการนำเอาออปแอมพ์มาเป็นตัวเปรียบเทียบ เอ้าต์พุตที่ได้จะมีค่าสูงหรือต่ำขึ้นอยู่กับแรงดันอินพุต ที่จ่ายให้ขาอินพุททั้งสอง ถ้าหากแรงดันอินพุตมีค่ามากกว่าแรงดันอ้างอิง เป็นจำนวนเพียงไมโครโวลท์ ก็จะทำให้แรงดันเอ้าต์พุต เปลี่ยนสถานะได้ วงจรออปแอมพ์ที่ใช้ในลักษณะดังกล่าวเรียกว่า "ตัวเปรียบเทียบ" ดังรูป 3.2 แสดงวงจรของตัวเปรียบเทียบ ซึ่งจะเห็นว่าขา inverting อินพุต ไม่ได้ต่อลงกราวน์ ดังนั้นแรงดันที่เราต้องการนำมาเปรียบเทียบ จะเปรียบกับค่าแรงดันใด ๆ ก็ได้ที่ต่อกับขา Vref นี้

ตัวเปรียบเทียบจะมีการนำไปใช้อย่างกว้างขวาง เราอาจจะต่อตัวเปรียบเทียบกับตัวตรวจจับอุณหภูมิ(temperature sensors) ที่อยู่บน Boiler ในโรงงานอุตสาหกรรม เมื่อแรงดันจากตัวตรวจจับอุณหภูมิสูงกว่าแรงดันอ้างอิงที่เราตั้งไว้ เอ้าต์พุตของตัวเปรียบเทียบก็จะเปลี่ยนสถานะ และส่งสัญญาณไปขัดจังหวะ(Interept) ไมโครโปรเซสเซอร์ที่ทำการควบคุม Boiler ตัวอย่างของตัวเปรียบเทียบที่มีใช้ในปัจจุบัน เช่น เบอร์ LM319 ซึ่งมีแรงดันเอ้าต์พุตเข้ากันได้(Compatible) กับระดับแรงดันของทีทีแอล ซึ่งเราสามารถต่อเข้าโดยตรงกับพอร์ตของไมโครคอมพิวเตอร์ หรือต่อกับอินเตอร์รัพท์อินพุต ของระบบไมโครคอมพิวเตอร์

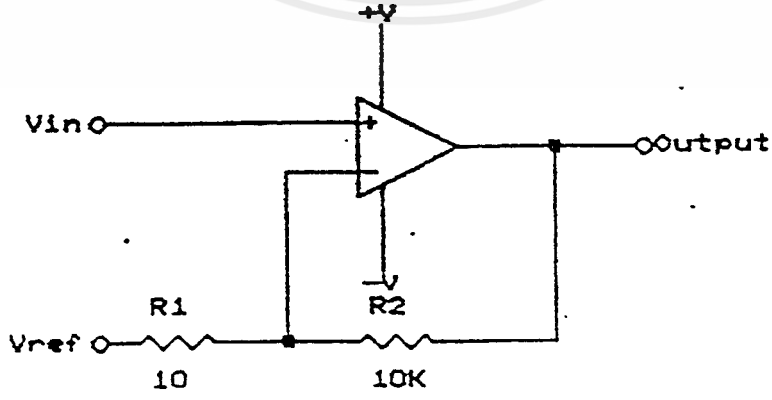
รูป 3.3 เป็นอีกวงจรหนึ่งของตัวเปรียบเทียบ โดยแรงดันอ้างอิงต่อกับขา non-inverting อินพุต และแรงดันอินพุต(ที่จะนำมาเปรียบเทียบ) จะต่อเข้ากับขา inverting ซึ่งการต่อลักษณะนี้จะเป็นการกลับกันกับสถานะเอ้าพุทของวงจรแรก(รูป 3.2) ในรูป 3.3 นี้จะมีตัวต้านทานป้อนกลับจากเอ้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูป 3.2 วงจรเปรียบเทียบเบื้องต้น

พุทมาที่อินพุต ซึ่งจะทำให้คุณสมบัติทางตัวเปรียบเทียบเปลี่ยนไป การต่อลักษณะนี้เราเรียกว่า Hysteresist ซึ่ง Hysteresist หมายความว่า เมื่อแรงดันอินพุตเปลี่ยนค่าไปทางบวก เอ้าต์พุตจะเปลี่ยนสถานะที่แรงดันหนึ่ง ซึ่งจะไม่เท่ากับระดับที่เปลี่ยนสถานะเมื่อแรงดันอินพุตเปลี่ยนไปในทางลบ ตัวอย่างเช่น thermostat ของเตารีดไฟฟ้า เมื่ออุณหภูมิของเตารีดเพิ่มถึงค่าหนึ่งสมมุติว่า 80 C thermostat จะตัดวงจรออกและเมื่ออุณหภูมิลดลงต่ำกว่า 80 C เล็กน้อย thermostat จะยังไม่ทำการต่อวงจรจนกว่าจะถึงค่าหนึ่งที่ต่ำกว่า 80 C เช่น 70 C เป็นต้น ซึ่งเป็นลักษณะเดียวกับการทำงานของ Hysterresist ถ้าหากไม่มี Hysteresist จะทำให้ thermostat ทำงานตัดต่อกลับไปมาอย่างรวดเร็วที่อุณหภูมิจุดเดียว



รูป 3.3 วงจรเปรียบเทียบที่มี Hysteresist

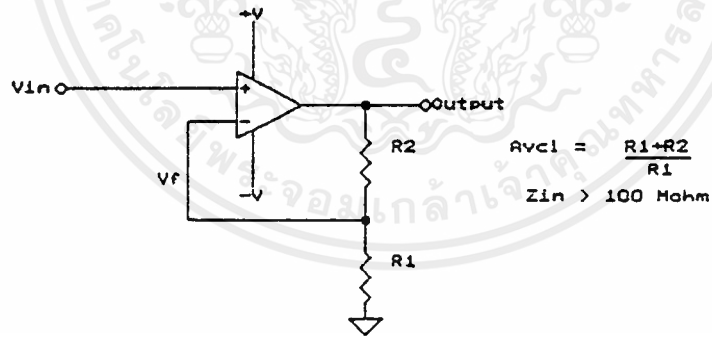
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ในการที่จะหาค่าของ Hysteresist ในวงจรดังเช่นรูป 11.1C สมมุติว่า Vref = 0 V และ Vout = 13 V เราสามารถใช้หลักการของการแบ่งแรงดันง่าย ๆ ดังที่แสดงในรูป

3.3.2 วงจรขยายออปแอมพ์แบบไม่กลับชั้ว

เมื่อเรานำเอาตัวต้านทานต่อป้อนกลับจาก เอ้าต์พุตมายังอินพุต เพื่อควบคุมอัตราการขยายของออปแอมพ์ รูป 3.4 แสดงการป้อนกลับแบบลบ ซึ่งจะทำให้วงจรมีเสถียรภาพดีขึ้น ตามรูป 3.4 แรงดันเอ้าต์พุตจะมีเฟสเดียวกันกับอินพุท ซึ่งมีการทำงานดังนี้

สมมุติว่าแรงดัน V_{in} = 0 V V_{out} = 0 V และแรงดันที่ขา + มีค่าเท่ากับ 0 V ต่อมาป้อนแรงดันที่ขา + = 0.01 V จะเกิดความแตกต่างระหว่างขา + และ − = 0.01 V ออปแอมพ์จะทำการขยายแรงดันนี้ด้วยค่า



รูป 3.4 วงจรขยายแบบไม่กลับชั้ว

อัตราขยาย = 100,000 เพื่อให้ได้เอ้าต์พุตเป็น 100 V ให้เร็วที่สุดเท่าที่ตัวมันจะทำได้ แต่เมื่อแรงดันเอ้าต์พุต เริ่มเพิ่มค่าเป็นบวกจะมีบางส่วนของแรงดันเอ้าพุทป้อนกลับมาที่ขา − ซึ่งจะทำให้แรงดันแตกต่างระหว่างขา + และ − มีค่าลดลง วงจรจะเข้าสู่สถานะที่เราสามารถคาดคะเนได้ค่าหนึ่งอย่างรวดเร็ว เมื่อแรงดัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ที่ขา - (V_f) มีค่าใกล้เคียงกับขา + มาก ๆ จะมีแรงดันแตกต่างระหว่างขา + และ ขา - มีค่าเป็นไมโครโวลท์ และถ้าเราสมมุติว่าแรงดันที่ขา + และ - มีค่าเท่ากัน เราก็จะหาแรงดันเอ้าต์พุตได้จากวงจรแบ่งแรงดันดังนี้

$V_{out} = V_{in}$ (R + R2)/R2 ถ้าหาก R2 = 99K, R1 = 1K จะได้

$V_{out} = V_{in} \times 100$

จะได้อัตราขยายลูปปิด (A_{CL}) = (R1 + R2)/R2

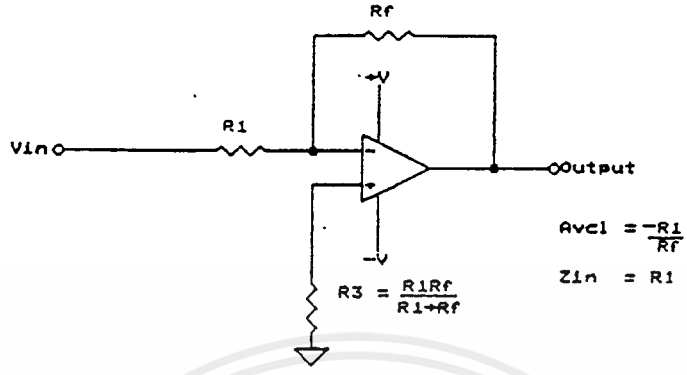
วงจรขยายออปแอมพ์แบบไม่กลับขั้ว สามารถจะนำไปทำเป็นบัฟเฟอร์ เพราะว่ามันมีคุณสมบัติที่มีความต้านทานทางขาเข้าสูงมาก โดย Bipolar ออปแอมพ์ทั่วๆ ไปจะมีอินพุตอิมพีแดนซ์มากกว่า 100 เมกกะโอห์ม แต่ถ้าเป็น FET ออปแอมพ์ เช่น LF356 ของ national Semicomduclor จะมีอินพุตอิมพีแดนซ์สูงถึง 10^{12} โอห์ม

3.3.3 วงจรขยายออปแอมพ์แบบกลับขั้ว

ดังรูป 3.5 โดยการต่อ ขา + ลงกราวน์ โดยผ่านตัวต้านทานตัวหนึ่ง และสัญญาณอินพุตที่เราต้องการขยายจะต่อเข้ากับขา - โดยผ่านตัวต้านทาน สัญญาณเอ้าต์พุตที่ได้จะมีเฟสแตกต่างกับอินพุตอยู่180 องศา(out of phase) ความต้านทาน R_f ใช้เป็นตัวป้อนกลับสัญญาณจากเอ้าต์พุตมาที่อินพุต ซึ่งจะทำให้ขาอินพุตทั้งสองมีค่าแรงดันเกือบเท่ากัน เนื่องจากขา non-inverting อินพุต ต่อลงกราวด์โดยผ่านตัวต้านทาน ในวงจรนี้ ขา - จะเป็นเสมือนกราวน์ อัตราขยายลูปปิดของวงจรที่มีความถี่ต่ำจะมีค่าเท่ากับอัตราส่วนของ -Rf/R1 เราสามารถที่จะหาสูตรได้โดยการคิดว่า ความต้านทานทั้งสองแบ่งแรงดันจาก V_{in} ที่ปลายด้านหนึ่งมีแรงดันที่จุดกลางเท่ากับศูนย์(เสมือนกราวน์) มี V_{out} อยู่ที่ปลายอีกด้านหนึ่งเครื่องหมายลบของอัตราขยายหมายความว่า แรงดันอินพุตและเอ้าต์พุต

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ต่างเฟสกัน 180 องศา อินพุตอิมพีแดนซ์ของวงจรนี้มีค่าโดยประมาณเท่ากับ R1 เพราะว่าขา - เป็นเสมือนกราวด์



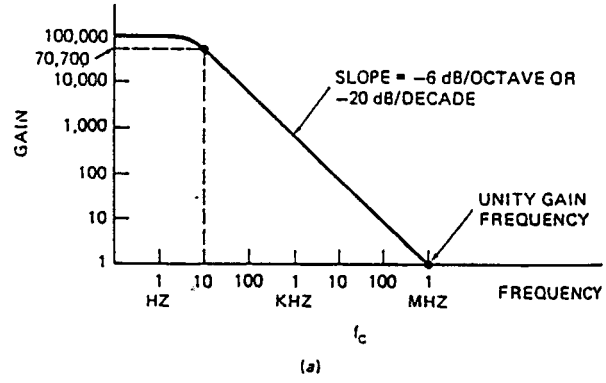
รูป 3.5 วงจรขยายแบบกลับขั้ว

คุณสมบัติอีกข้อหนึ่งของออปแอมพ์ ก่อนที่เราจะไปรู้ถึงวงจรอื่นๆของออปแอมพ์ คือ Gain-Bandwidth-Product เช่นจากอัตราขยายลูปเปิดมีค่าเป็น 100,000 ถ้าหากความถี่ที่ออปแอมพ์ทำการขยายมีค่าสูงขึ้นจะทำให้อัตราขยายนี้ลดลง รูป 3.6(a) แสดงอัตราขยายแรงดันลูปเปิดกับความถี่ สำหรับออปแอมพ์โดยทั่ว ๆ ไปเช่น 741 ที่จุดอัตราการขยายมีค่าเป็นหนึ่ง เรียกว่า Unity-gain Bandwidth หรือเรียกว่า gain-bandwidth product โดยปกติค่านี้มีความถี่ที่ 1 เมกกะเฮริต เช่นออปแอมพ์มี gain-bandwidth product เท่ากับ 1 เมกกะเฮริต ถ้าใช้ทำเป็นวงจรขยายมีอัตราขยายเท่ากับ 100 band-width ของวงจร(f_c) จะมีค่า = $1MH_Z/100 = 10KH_Z$ ดังรูป 3.6(b)

3.3.4 วงจรบวกที่ใช้ออปแอมพ์(Op-Amp Adder Circuit)

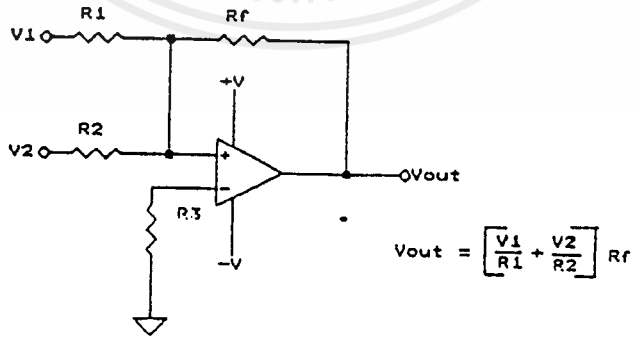
รูป 3.7 เป็นการวงจรขยายออปแอมพ์แบบกลับขั้ว ซึ่งจะทำการรวม(หรือบวก) หรือผสมสัญญาณอินพุตที่มีอินพุตเท่ากับ 2 หรือมากกว่าการทำงานเป็นดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูป 3.6 (a) อัตราขยายลูปเปิดกับความถี่ของออปแอมพ์ 741

(b) อัตราขยายลูปปิดกับความถี่ของออปแอมพ์ 741



รูป 3.7 วงจรบวก(หรือวงจรผสม)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากที่ผ่านมาขา - จะเป็นเสมือนกราวด์ แรงดัน V_1 จะทำให้เกิดกระแสไหลผ่าน R1 และ V_2 จะทำให้เกิดกระแสไหลผ่าน R2 กระแสทั้งสองจะรวมกัน(ผสมกัน) ที่จุดเสมือนกราวด์ ซึ่งโดยมากจะเรียกว่า summing point (จุดผสม) กระแสทั้งสองจะดึงผ่าน R_f โดยออปแอมพ์ เพื่อให้ขา - เป็นเสมือนกราวด์ แรงดันเอ้าต์พุตจะมีค่าเท่ากับ R_f คูณกับกระแสที่ไหลผ่าน R1 และ R2 (ซึ่งเท่ากับกระแสที่ไหลผ่าน R_f) คือ

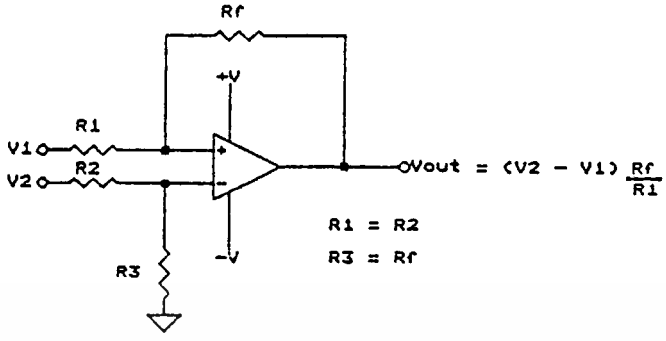
$$V_{out} = (V_1/R_1 + V_2/R_2) R_f \quad \ldots\ldots\ldots\ldots\ldots\ldots (3.1)$$

วงจรดังกล่าวนี้สามารถใช้ในการผสมสัญญาณเสียง และใช้เป็นตัวบวก "น้ำหนัก" ของกระแสในตัวแปลง D/A

3.3.5 วงจรขยายแตกต่างเบื้องต้น (Simple Differential-Input Amplifier Circuit)

ดังที่เราจะได้เห็นต่อไป เซนเซอร์ส่วนมากจะมีขาสัญญาณออกมาให้ใช้ 2 ขา โดยมีแรงดันไฟตรงปรากฏที่แต่ละปลายเรียกว่า Common-Mode Signal สัญญาณจริง ๆ ที่เราจำเป็นจะต้องนำมาขยายจากเซนเซอร์ จะเป็นแรงดันแตกต่างจำนวนน้อยมาก ๆ ระหว่างสายสัญญาณทั้งสองซึ่งโดยมากจะมีค่าตั้งแต่เป็น มิลลิโวลท์ จากรูป 3.8 ถ้าหากยังใช้วงจรขยายแบบธรรมดา เราจะได้แรงดันไฟตรงที่มีขนาดใหญ่ รูป 3.8 เป็นวงจรเบื้องต้น การวิเคราะห์วงจรนี้ไม่อยู่ในวิสัยที่จะทำได้ในที่นี้ แต่โดยพื้นฐานแล้วความต้านทานที่ขา + จะดึงให้แรงดันที่ขา + นี้มีค่าใกล้เคียงกับแรงดัน Common-Mode ถ้าหากความต้านทานมีค่าเท่ากันจริง ๆ ผลที่ได้ก็คือจะมีเฉพาะแรงดันแตกต่างระหว่าง V_1 และ V_2 เท่านั้นที่จะถูกขยาย แรงดันเอ้าต์พุตจะมีเพียงเส้นเดียว(ไม่รวมกราวด์) โดยเราพูดได้ว่าสัญญาณ Common-Mode จะถูกตัดทิ้งไป

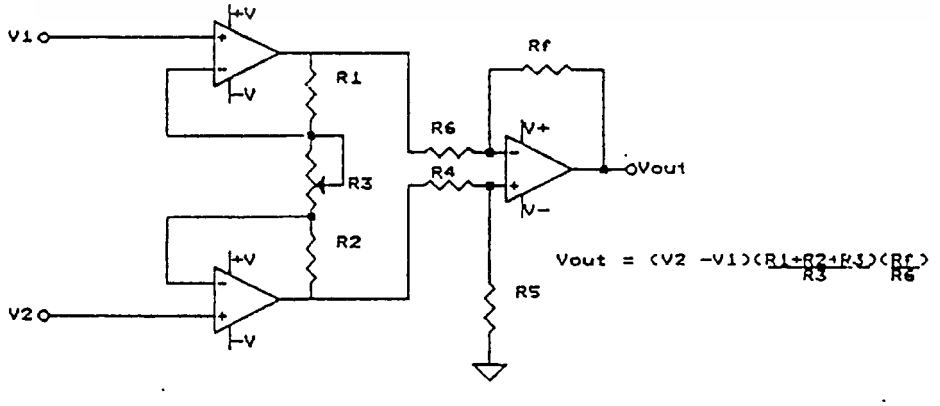
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูป 3.8 วงจรขยายความแตกต่าง

3.3.6 วงจรขยายเครื่องมือวัด(Instrumentation Amplifier Circuit)

วงจร 3.9 เป็นวงจรออปแอมพ์ ที่ใช้ประยุกต์ใช้งานที่จำเป็นจะต้องมีการกำจัดสัญญาณ Common-Mode แทนที่จะใช้วงจรขยายความแตกต่างธรรมดา ดังรูปที่ 3.8 โดยออปแอมพ์ 2 ตัวแรกจะเป็นตัวเอาแรงดัน Common-Mode ออกไปและออปแอมพ์ตัวสุดท้าย(ตัวที่ 3) จะทำการแปลงผลที่ได้ จากสัญญาณความแตกต่างไปเป็นสัญญาณที่อ้างอิงกับกราวด์ วงจรขยายเครื่องมือวัดดังกล่าวปัจจุบันมาในรูปของ IC เพียงตัวเดียว

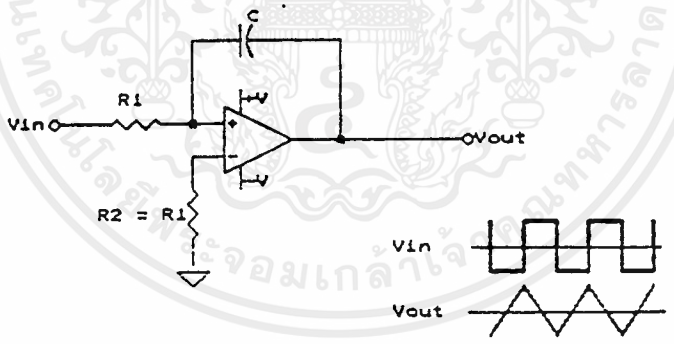


รูป 3.9 วงจรขยายเครื่องมือวัด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.3.7 วงจรอินทิเกรทที่ใช้ออปแอมพ์(An Op-Amp Intigrater Circuit)

รูป 3.10 เป็นวงจรออปแอมพ์ที่สามารถที่จะใช้ในการสร้างแรงดันที่เรียกว่า Linear Voltage Ramps แรงดันไฟตรงที่ป้อนให้กับอินพุตของวงจรนี้จะทำให้กระแสคงที่ของ V_{in}/R_1 ไหลผ่านจุดที่เป็นเสมือนกราวด์ ซึ่งกระแสนี้จะไหลไปที่แผ่นหนึ่งของตัวเก็บประจุ เพื่อที่จะให้ได้ขา - เป็นเสมือนกราวด์ เอ้าต์พุตของออปแอมพ์จะดึงกระแสที่มีขนาดเดียวกันนี้จากอีกแผ่นหนึ่งของตัวเก็บประจุซึ่งก็คือการประจุให้กับตัวเก็บประจุด้วยกระแสคงที่มีค่าเท่ากับ V_{in}/R_1 จากพื้นฐานทางฟิสิกก์ เราจะได้ว่า แรงดันที่คร่อมตัวเก็บประจุที่ประจุด้วยกระแสคงที่จะเป็น linear ramp และเพราะว่าเราต่อให้มีการขยายแบบกกลับขั้วแรงดัน linear ramp ที่เอ้าต์พุตจึงเป็นลบเมื่อแรงดันอินพุตเป็นบวก ที่เรียกว่าวงจรอินทิเกรเตอร์ เพราะว่ามันสร้างแรงดันเอ้าต์พุตเป็นสัดส่วนกับผลรวม(integrated) ของกระแสที่สร้างจากแรงดันอินพุต ภายใต้ 1 ระยะเวลา



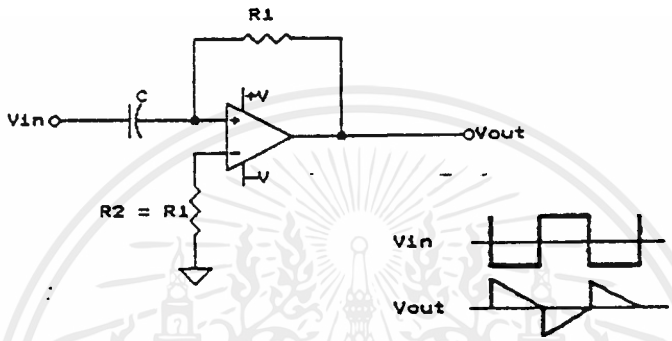
รูป 3.10 วงจรอินทิเกรท

3.3.8 วงจรอนุพันธ์ที่ใช้ออปแอมพ์(An Op-Amp Differentiater Circuit)

รูป 3.11 แสดงวงจรออปแอมพ์ ที่สร้างแรงดันเอ้าต์พุตเป็นสัดส่วนกับอัตราการเปลี่ยนแปลงของแรงดันอินพุต ถ้าหากแรงดันอินพุตคงที่(ที่ค่าใด ๆ) แรงดันเอ้าต์พุตจะมีค่าเป็นศูนย์ ถ้าหากมีแรงดัน "ใหม่" ป้อนเข้าไปที่อิน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

พุท แรงดันที่คร่อมตัวเก็บประจุ(ที่อินพุต)จะไม่สามารถเปลี่ยนแปลงทันที นั่นคือ แรงดันที่ขา Inverting จุดถูกยกระดับขึ้น(หรือลง) จากระดับแรงดันศูนย์โวลท์ ซึ่งจะทำให้เกิดการขับให้เอ้าต์พุตทำการประจุตัวเก็บประจุ ซึ่งจะเป็นการดึงระดับแรงดันที่ขา inverting กลับลงไปเป็นศูนย์โวลท์ดังเดิม ระยะเวลาที่ใช้ในการทำให้เอ้าต์พุตเป็นศูนย์โวลท์ หาได้จาก time canstant ของ R_1 และ C_1



รูป 3.11 วงจรอนุพันธ์

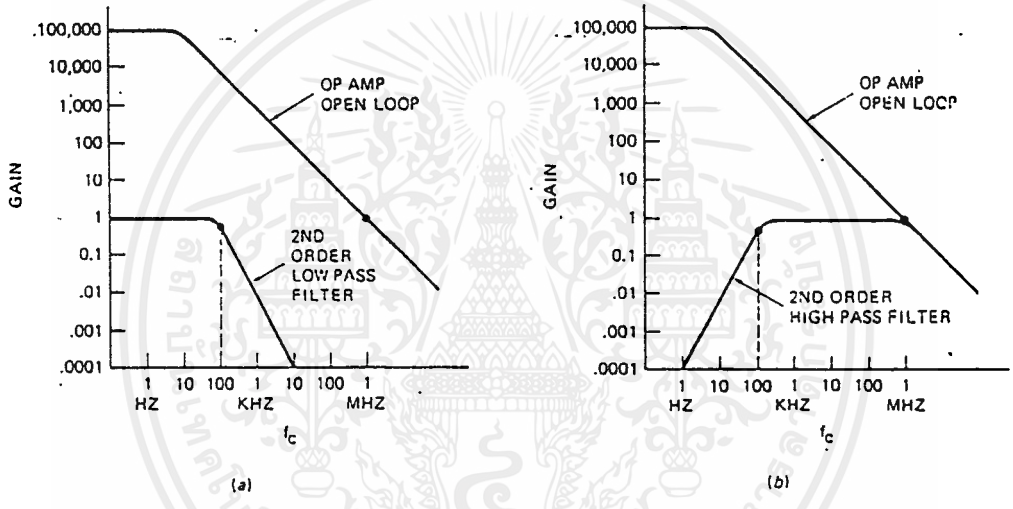
3.3.9 วงจรกรองแบบแอคทีฟที่ใช้ออปแอมพ์(Op-Amp Active-Filters)

ในการนำไปประยุกต์ใช้ในงานควบคุมบางอย่าง เราจำเป็นที่จะต้องกรองเอาสัญญาณที่เราไม่ต้องการออกไป ซึ่งอาจจะเป็นความถี่สูง หรือความถี่ต่ำ(เป็นสัญญาณรบกวนที่ได้จากตัวเซ็นเซอร์) ซึ่งเราทำได้โดยใช้วงจรกรอง RC แบบ passive แต่การใช้ Active filters จะให้คุณสมบัติในการควบคุมได้ดีกว่า การกรองโดยใช้ออปแอมพ์จะทำได้หลายรูปแบบ แต่พอจะพูดได้ว่ามีแบบความถี่ต่ำผ่าน(low-pass),ความถี่สูงผ่าน(high-pass) และความถี่ผ่านได้เป็นช่วง(band-pass)

วงจรกรองความถี่ต่ำผ่าน จุดที่ความถี่ที่ผ่านตกลงมา 0.707 ของค่าความถี่ต่ำ เราเรียกว่าความถี่วิกฤติ(Critical Frequency) รูป 3.12(a)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

แสดงกราฟและอัตราการขยายของวงจรกรองความถี่ต่ำผ่าน โดย f_c คือความถี่วิกฤติ ที่ความถี่สูงกว่า f_c อัตราการขยายจะตกลงมาอย่างรวดเร็ว ถ้าเป็น first-order เช่น ใช้ R และ C อย่างละตัว อัตราขยายจะตกลงมาด้วยค่า factor = 10 เมื่อความถี่เพิ่มขึ้น 10 เท่า (หรือเรียกว่าอัตราการขยายลดลง -20 DB/decade) ถ้าเป็น Second-order filter อัตราการขยายจะลดลงด้วยค่า factor = 100 เมื่อความถี่เพิ่มขึ้น 10 เท่า



รูป 3.12 อัตราขยายกับผลตอบสนองต่อความถี่ของวงจรกรอง

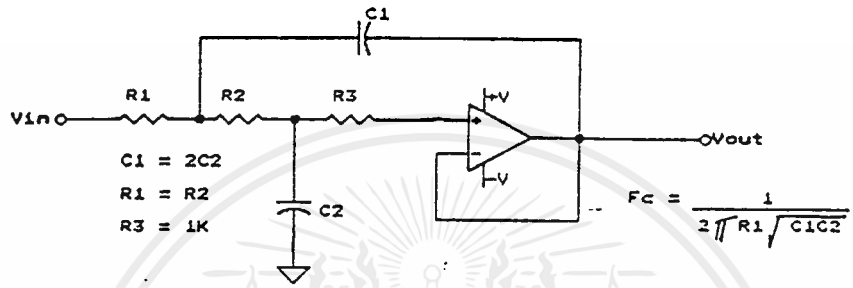
(a) แบบความถี่ต่ำผ่าน

(b) แบบความถี่สูงผ่าน

รูป 3.13 เป็นวงจรทั่ว ๆ ไป ของ scend-order แบบความถี่ต่ำผ่าน สิ่งที่น่าสังเกตก็คือ ทางเดินของแรงดันไฟตรงจะมาถึงขา non-inverting ได้ซึ่งหมายความว่า ออปแอมพ์จะขยายแรงดันไฟตรงได้ด้วย (พร้อมทั้งความถี่ต่ำ) ดังนั้นผลตอบสนองต่อความถี่ของมันจึงเป็นดังรูป 3.12(a)

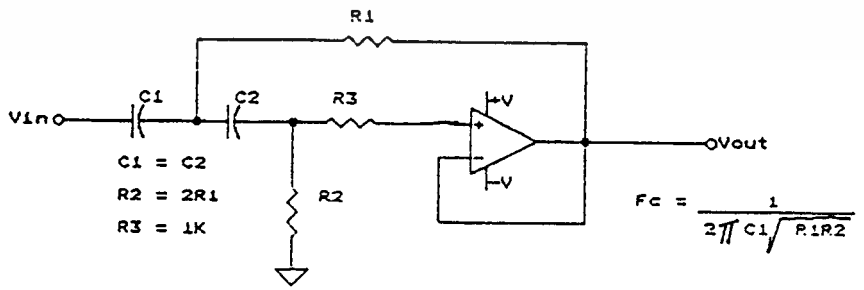
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ถ้าเป็นวงจรดังรูป 3.14 ซึ่งไม่มีทางเดินของแรงดันไฟตรงจากอินพุตมาถึงขา non-inverting ของออปแอมพ์(เพราะ C บล๊อคแรงดันไฟตรง) ดังนั้นวงจรนี้จะไม่ขยายแรงดันไฟตรง และไม่ขยายความถี่ต่ำด้วย จะได้กราฟผลตอบสนองต่อความถี่ดังรูป 3.12(b) เป็นลักษณะของวงจรของความถี่สูงผ่าน



รูป 3.13 วงจรกรองความถี่ต่ำผ่านชนิด Second Order

โดยที่ขอบเขตทางด้านความถี่สูงของวงจรจะถูกจำกัดอยู่ที่ค่า gain-bandwidth product ของออปแอมพ์ที่เราใช้



รูป 3.14 วงจรกรองความถี่สูงผ่านชนิด Scond Order

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สำหรับวงจรกรองความถี่ต่ำผ่านดังรูป 3.13 อัตราขยายในช่วงที่ราบเรียบของกราฟตอบสนองความถี่มีค่าเท่ากับ 1 หรือเรียกว่า unity ทั้งนี้เพราะว่าเอ้าต์พุตได้ถูกป้อนกลับโดยตรงมาที่ขา inverting (โดยไม่ผ่านตัวต้านทาน) ที่ความถี่ f_c อัตราขยายจะลดลง 0.707 และที่ความถี่สูงกว่า f_c อัตราขยายจะต้องตกลงอย่างรวดเร็ว สูตรที่ใช้สำหรับกำหนดความถี่ f_c คือ

$$f_c = \frac{1}{2\pi R1\ C1\ C2}$$:โดยมีเงื่อนไขว่า R1 = R2 และ : C1 มีค่าเป็น 2 เท่าของ C2

หน้าที่ของ R3 เป็น Damping-resistor(รอยหยักก่อนที่กราฟผลตอบสนองจะตกลงที่ความถี่ f_c) การป้อนกลับแบบบวกกระทำโดย C1 เป็นเหตุเดียวที่ทำให้อัตราการขยายของวงจรตกลง 0.707 ของค่า f_c

สำหรับวงจรกรองความถี่สูงผ่าน อัตราการขยายในช่วงที่ราบเรียบก็จะมีค่าเป็น 1 เช่นกัน โดยสมมุติว่า C1 = C2 (รูป 3.4) และค่า R2 มีค่าเป็น 2 เท่าของ R1 ค่าความถี่วิกฤติหาได้ดังนี้

$$f_c = \frac{1}{2\pi C1\ R1\ R2}$$

และหน้าที่ของ R3 ก็คือเป็น Damping-resistor

เมื่อเรานำวงจรกรองความถี่ต่ำผ่าน 0 และวงจรกรองความถี่สูงผ่าน มาต่ออนุกรมกัน ผลตอบสนองต่อความถี่ที่ได้ก็คือวงจรกรองความถี่ผ่านได้เป็นช่วง(band pass filter) แต่ที่มีวงจรเดี่ยว ๆ ที่จะทำให้ความถี่ผ่านเป็นช่วงได้อีกมากมาย ซึ่งไม่อยู่ในขอบเขตที่จะกล่าวถึงในที่นี้

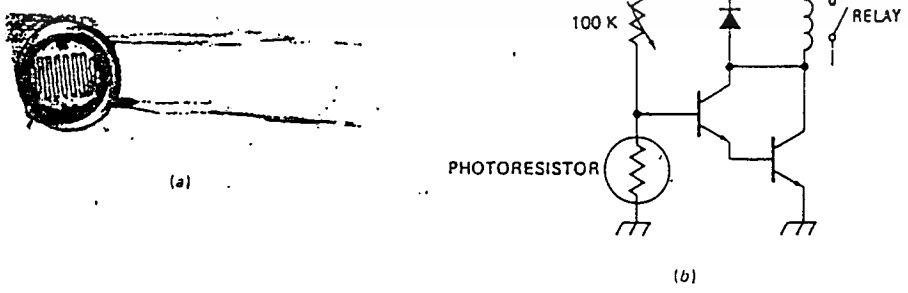
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.4 เซนเซอร์และทรานสดิวเซอร์(Sensors And Transducers)

ในที่นี้จะกล่าวถึงเซนเซอร์และทรานสดิวเซอร์ ในทางการแนะนำให้รู้จักเบื้องต้น เราสามารถที่จะหารายละเอียดการทำงานของเซนเซอร์ และทรานสดิวเซอร์ ได้จากคู่มือของผู้ผลิต

3.4.1 เซนเซอร์ทางแสง(Light Sensers)

ตัวอย่างง่าย ๆ ของเซนเซอร์ทางแสงก็คือ ตัวต้านทานที่มีค่าขึ้นอยู่กับแสงที่ตกกระทบตัวมัน เช่น โฟโต้ทรานซิสเตอร์ ดังรูป 3.15(a) ซึ่งจะมีช่องให้แสงลอดผ่านวัตถุโปร่งแสงเข้าไปกระทบ cadmiun sulfide ที่วางเป็นรูป ซิก–แซก ซึ่งสารดังกล่าวจะมีค่าความต้านทานขึ้นอยู่กับแสงที่ตกกระทบตัวมัน ตามตัวอย่างรูป 3.15(a) เป็นเบอร์ CL905 ที่มืดจะมีความต้านทานประมาณ 15 เมกกะโอห์ม และเมื่อแสงสว่างเต็มที่จะมีความต้านทานลดลงเหลือ 15 กิโลโอห์ม เนื่องจาก CL905 มีราคาถูกคุณสมบัติการตอบสนองต่อแสงจึงไม่ทันทีทันใดและไม่คงที่จึงเหมาะที่จะใช้เมื่อเราไม่ต้องการความละเอียด เช่นใช้ตรวจจับว่ามีแสงหรือไม่มีแสงดังวงจร 3.15(b)

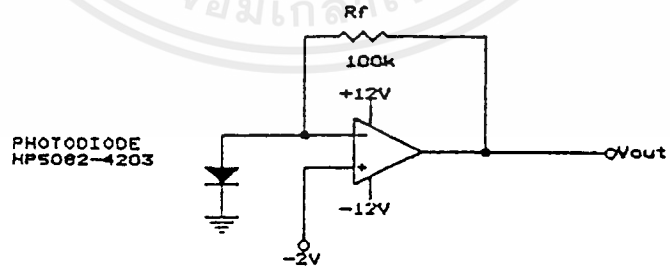


รูป 3.15 (a) โฟโตเซลล์ที่ใช้ แคดเมี่ยมซัลไฟน์

(b) วงจรใช้แสงควบคุมรีเลย์โดยใช้โฟโต้เซลล์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

อุปกรณ์อีกตัวหนึ่งที่ใช้ในการวัดจำนวนของแสงคือ โฟโต้ไดโอด (photodiode) โดยถ้าเราทำให้แสงกระทบไปที่รอยต่อของไดโอด ค่ากระแสรั่วไหลย้อนกลับจะมีค่าเพิ่มขึ้นเมื่อความสว่าง(จำนวน)ของแสงเพิ่มขึ้น ตัวอย่างของวงจรที่ใช้วัดจำนวนแสงดังรูป 3.16 ซึ่งเป็นวงจรทำหน้าที่แปลงแสรั่วไหลย้อนกลับของโฟโต้ไดโอดที่มีค่าน้อยมาก ๆ ไปเป็นแรงดันที่เป็นสัดส่วนกันกับกระแสรั่วไหลของโฟโต้ไดโอด โดยในวงจรดังกล่าวมีการป้อนแรงดันอ้างอิงที่ขา non-inverting ของออปแอมพ์ ดังนั้นที่ขา inverting ก็จะมีแรงดันใกล้เคียงกับขา non-inverting เพื่อใช้ในการทำเป็น reverse-bias ให้กับโฟโต้ไดโอด เมื่อแสงเปลี่ยนไปค่าแรงดันคร่อมโฟโต้ไดโอดก็จะเปลี่ยนไป ออปแอมพ์ก็จะทำหน้าที่ขยายแรงดันนี้ออกมาที่เอ้าต์พุตของวงจร ตัวอย่างในวงจรนี้ใช้โฟโต้ไดโอด เบอร์ HP 5082-4203 ซึ่งจะมีกระแสรั่วไหลย้อนกลับระหว่าง 0 uA ถึง 100uA เมื่อ R_f = 100 กิโลโอห์ม แรงดันเอ้าต์พุตของออปแอมพ์จะมีค่าอยู่ระหว่าง 0 V ถึง 10 V ถ้าเราไม่ต่อแรงดัน reverse-bias วงจรก็จะสามารถทำงานได้ แต่ผลตอบสนองต่อแสงจะไม่รวดเร็วเมื่อแสงเปลี่ยนไป การใช้ BIFET ออปแอมพ์ เบอร์ LM356 เพราะว่าออปแอมพ์ตัวนี้ต้องการกระแสอินพุต bias น้อยมาก (ถือว่าไม่มี)



รูป 3.16 วงจรวัดความเข้มของแสงอินฟราเรดที่ใช้โฟโต้ไดโอด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

วงจรดังที่กล่าวมาสามารถใช้ในการตรวจจับควันได้ โดยการใช้อินฟราเรด LED ให้แสงสอ่งมาที่โฟโต้ไดโอด เมื่อมีควันมาบังระหว่างอินฟราเรด LED กับโฟโต้ไดโอด จะทำให้จำนวนของแสงที่โดนรอยต่อเปลี่ยนไป การใช้อินฟราเรด LED เพราะโฟโต้ไดโอด มีความไวต่อแสงในช่วงนี้ดีที่สุด

อุปกรณ์ที่มีความไวต่อแสงอีกอย่างหนึ่งคือ โซลาร์เซลล์(Solarcell) โดยปรกติโซลาร์เซลล์จะมีขนาดใหญ่ และทำการ Dope อย่างหนักที่รอยต่อ P-N เมื่อแสงอาทิตย์ตกกระทบที่รอยต่อ P-N จะทำให้กระแสย้อนกลับไหลเหมือนกับโฟโต้ไดโอด แต่เพราะว่าโซลาร์เซลล์มีขนาดใหญ่ และ Dope อย่างหนักที่บริเวณรอยต่อ P-N ดังนั้น กระแสที่เกิดขึ้นจึงมีจำนวนมาก(เป็นมิลลิแอมแปร์) แทนที่จะเป็นไมโครแอมแปร์ ส่วนมากโซลาร์เซลล์จะใช้เป็นตัวให้กำลังแสงสว่าง ซึ่งทำให้ได้กำลังมากขึ้นโดยการนำมาต่อทั้งอนุกรมและขนาน เพื่อให้ได้แรงดันและกระแสตามต้องการได้

มิเตอร์วัดแสงในกล้องถ่ายรูป การขยายรูปภาพ และเครื่องจักรที่ใช้ทำแผ่นปริ้นจะใช้โซลาร์เซลล์ กระแสที่ได้จากโซลาร์เซลล์จะมีสัดส่วนเป็นเชิงเส้นกับจำนวนของแสงที่ตกกระทบตัวมัน ดังรูป 3.16 สามารถที่จะใช้ในการแปลงกระแสให้เป็นแรงดันได้ เพราะว่ากระแสที่เกิดจากแสงมีจำนวนมาก เราก็ไม่จำเป็นจะต้องต่อแรงดันอ้างอิงที่ขา non-inverting

3.4.2 เซนเซอร์อุณหภูมิ(Temperature Sensors)

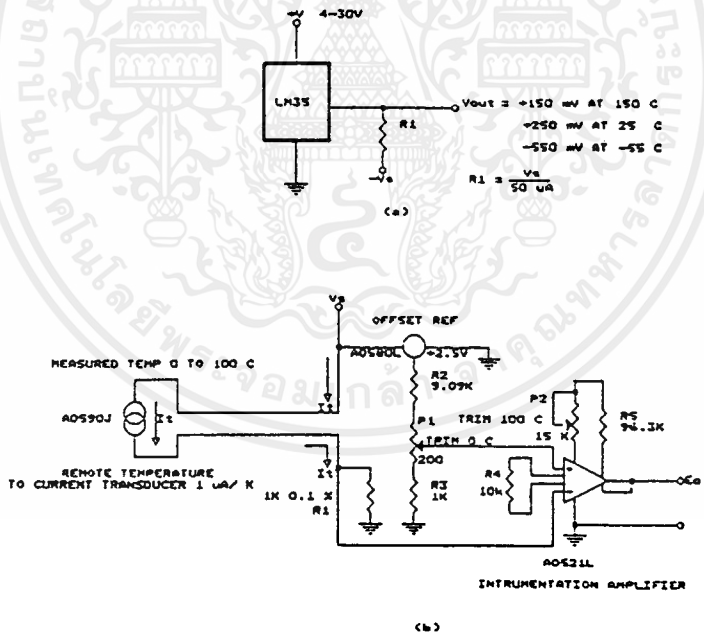
ตัวตรวจจับอุณหภูมิมีมากมายหลายชนิดในที่นี้จะกล่าวถึงเพียง 2 อย่าง คือ อุปกรณ์ที่เป็นสารกึ่งตัวนำ และเทอร์โมคับเปิล(thermocouple) ซึ่งสามารถที่จะใช้วัดอุณหภูมิที่สูงมาก ๆ ได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.4.2.1 เซนเซอร์อุณหภูมิที่เป็นสารกึ่งตัวนำ

(Semicondutor Temperature Sensors)

แบ่งได้เป็น 2 ชนิดใหญ่ ๆ คือ จำพวกที่ให้เอ้าต์พุตออกมาเป็นแรงดันและจ่ายออกมาเป็นกระแส ตัวอย่างของ temperature-Sensitive voltage source เช่น เบอร์ LM35 ดังวงจรรูป 3.17(a) แรงดันเอ้าต์พุตจะเพิ่มขึ้นทุก ๆ 10 mV ต่อการเพิ่มขึ้นของอุณหภูมิ 1 C โดยการต่อแรงดันอ้างอิง ลบ (V_S) ดังรูป LM35 จะวัดอุณหภูมิได้จาก -55 C ถึง +150 C โดยการปรับให้ได้เอ้าต์พุตมีค่าเท่ากับ 0 V ที่ 0 C เราสามารถที่จะนำแรงดันจาก LM35 มาขยายให้ได้ขนาดใหญ่ขึ้นตามต้องการได้ ตามความจำเป็นในการนำไปประยุกต์ใช้งาน



รูป 3.17 เซนเซอร์อุณหภูมิที่เป็นสารกึ่งตัวนำ

(a) ชนิดให้เอ้าต์พุตออกมาเป็นแรงดัน เบอร์ LM35

(b) ชนิดให้เอ้าต์พุตออกมาเป็นกระแส เบอร์ AD590

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตัวอย่างของ temperature-Sensitive current source เช่น เบอร์ AD590 ซึ่งจะให้กระแสออกมา 1 uA ต่อองศาเคลวิน(K) ดังรูป 3.17(b) เป็นตัวอย่างวงจรที่ใช้ในการแปลงกระแสให้เป็นแรงดันที่เป็นสัดส่วนกัน โดยกระแสจาก AD590 (I_t) จะไหลผ่านความต้านทานค่าประมาณ 1 กิโลโอห์มลงกราวด์ ซึ่งจะทำให้เกิดแรงดันคร่อมตัวต้านทานดังกล่าว = 1mV/ K ข้อดีของเซนเซอร์แบบให้กระแสออกมาก็คือ เนื่องจากการไหลของกระแสในสายตัวนำจะมีค่าเท่ากัน(ไหลเข้ามีค่าเท่ากับไหลออก) ดังนั้นการต่อสายยาว ๆ จากตัวเซนเซอร์แบบกระแสจึงไม่เกิดปัญหา

3.4.2.2 เทอร์โมคับเปิล(Themocouples)

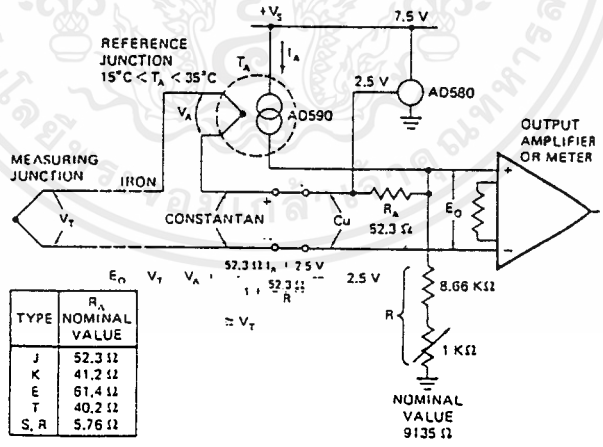
เมื่อเรานำโลหะต่างชนิด 2 ชนิดมาเชื่อมต่อกัน จะเกิดแรงดันคร่อมรอยต่อนั้น ๆ แรงดันที่เกิดขึ้นจะขึ้นอยู่กับชนิดของโลหะและอุณหภูมิรอบ ๆ รอยต่อ โดยแรงดันที่เกิดขึ้นจะมีค่าอยู่ระหว่าง 7 uV ถึง 75 uV(ขึ้นอยู่กับชนิดของโลหะ) เมื่ออุณหภูมิเพิ่มขึ้นแต่ละ C การนำโลหะหลาย ๆ ชนิดมาต่อรวมกันจะนำมาใช้ประโยชน์ตอนวัดพิสัยของอุณหภูมิที่แตกต่างกัน เทอร์โมคับเปิลที่นำเอาเหล็ก(Iron) และ constantan มาต่อกันจะเรียกว่าชนิด J (Type J thermocouple) จะใช้วัดอุณหภูมิตั้งแต่ -184 C ถึง +760 C รอยต่อของ platinum และ alloy-platinium และ 13% rhodium จะใช้วัดจาก 0 C ถึง 1600 C และเทอร์โมคับเปิลสามารถทำให้เล็กกระทัดรัดได้ แต่ปัญหาที่เกิดขึ้นมีอยู่ 3 อย่าง คือ

ปัญหาที่หนึ่ง จากความจริงที่ว่ามันให้แรงดันออกมาต่ำมาก ๆ จึงจำเป็นจะต้องขยายด้วยอัตราการขยายที่สูงมาก ๆ เช่น ถ้าจะนำไปขับตัวแปลง A/D

ปัญหาที่สอง เพื่อให้มีความถูกต้องแม่นยำที่สุด รอยต่อที่สองของโลหะชนิดเดียวกันจะต้องรวมเข้าไปด้วยคล้ายกับการอ้างอิงของวงจร การเพิ่มรอย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ต่อที่สองจะเรียกว่าการชดเชยรอยต่อเย็น(cold-junction compensation) รูป 3.18 แสดงถึงวงจรขยายสัญญาณจากเอ้าต์พุตของเทอร์โมคับเปิล โดยมีการชดเชยรอยต่อเย็นสำหรับเทอร์โมคับเปิลชนิด J สิ่งหนึ่งขอย้ำในที่นี้ก็คือในวงจรจะมีการต่อรอยต่ออ้างอิงที่มีทิศทางกลับขั้วกับรอยต่อที่ใช้ในการวัด โดยทำให้สายวัดเป็น contantan แล้วต่อสายดังกล่าวเข้ากับสายทองแดง เพื่อเข้าวงจรขยายสัญญาณ ผลที่ได้ก็คือได้เอ้าต์พุตที่เป็นแรงดันความแตกต่างระหว่าง เทอร์โมคับเปิล 2 ตัว ถ้าเราจะขยายสัญญาณที่เป็นผลรวมของทั้งสองรอยต่อ ปัญหาที่เป็นไปได้ที่จะเกิดขึ้นก็คือ เมื่ออุณหภูมิที่รอยต่อทั้งสองเปลี่ยนแปลงไป ซึ่งเราจะบอกไม่ได้ว่าอุณหภูมิที่รอยต่อไหนเปลี่ยนไป การแก้ปัญหาก็คือ เอารอยต่อที่เป็นรอยต่ออ้างอิงใส่ไว้ในถังน้ำแข็งหรือที่ ๆ มีอุณหภูมิคงที่ ซึ่งไม่ใช่เรื่องที่สะดวกนัก วงจรในรูปที่ 3.18 จะใช้การชดเชยทางอิเลคทรอนิคส์ เมื่ออุณหภูมิของรอยต่ออ้างอิงเปลี่ยนไป



TYPE	R_A NOMINAL VALUE
J	52.3 Ω
K	41.2 Ω
E	61.4 Ω
T	40.2 Ω
S, R	5.76 Ω

รูป 3.18 วงจรแสดงให้เห็นการนำเทอโมคับเปิลประยุกต์ใช้งานโดยมีการชดเชยรอยต่อเย็น

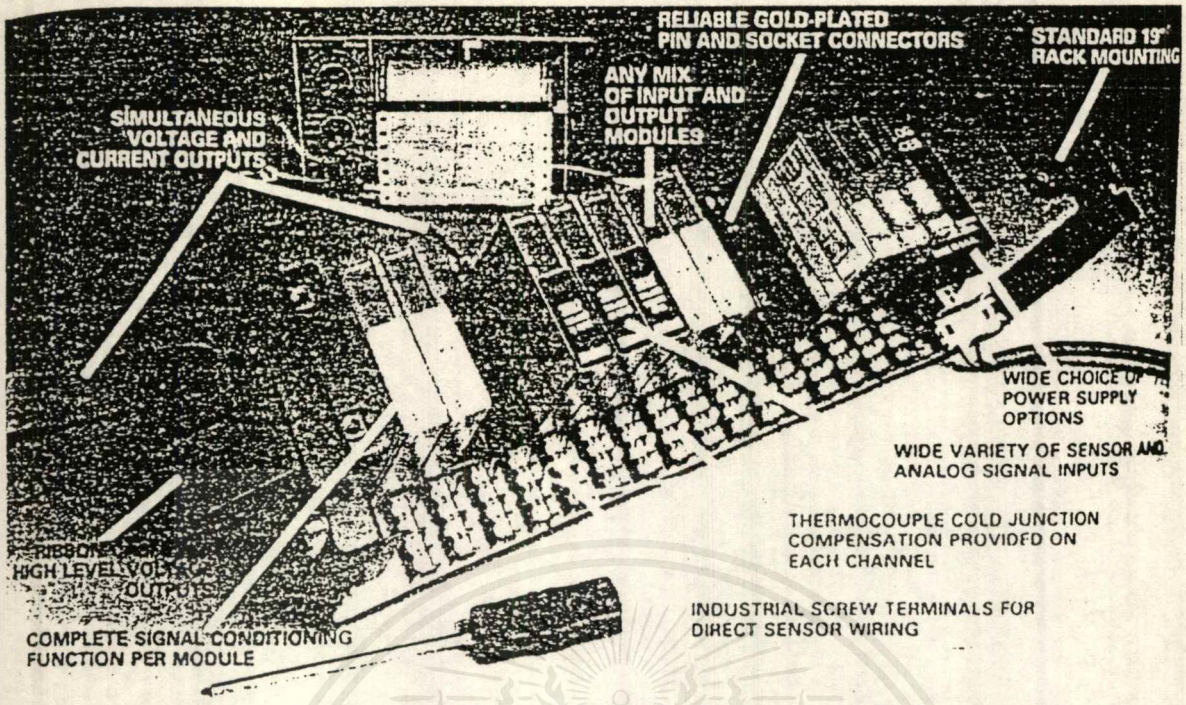
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากที่กล่าวมาแล้ว AD590 จะให้กระแสที่เป็นสัดส่วนกับอุณหภูมิจึงนำ AD590 มาต่อกับเทอร์โมคับเปิลที่เป็นตัวอ้างอิง โดยที่ AD590 ทั้งสองตัวจะอยู่ที่สภาวะแวดล้อมที่อุณหภูมิเดียวกัน กระแสจาก AD590 เมื่อไหลผ่านโครงข่ายความต้านทาน จะทำให้เกิดแรงดันเพื่อใช้ในการชดเชยการเปลี่ยนแปลงของรอยต่อที่เป็นรอยต่ออ้างอิงเมื่ออุณหภูมิเปลี่ยนไป วงจรขยายในที่นี้จะเป็นวงจรขยายความแตกต่าง หรือเป็นวงจรขยายเครื่องมือวัดก็ได้

ปัญหาที่สาม ก็คือ แรงดันเอ้าต์พุตที่ได้จะไม่เป็นเชิงเส้นกับการเปลี่ยนแปลงของอุณหภูมิ ซึ่งจะทำให้ถูกต้องได้โดยใช้วงจรทางอนาลอก โดยการเปลี่ยนอัตราการขยายของวงจรตามค่าของสัญญาณ แต่เมื่อนำเทอร์โมคับเปิลมาใช้กับไมโครคอมพิวเตอร์ การแก้ไขความไม่เป็นเชิงเส้นดังกล่าวก็จะทำได้ง่ายสะดวกและถูกต้องขึ้น โดยการสร้างตารางไว้ในรอม และใช้ตัวแปลง A/D แปลงเป็นดิจิตอล ค่าที่ได้จากการแปลง(ดิจิตอล) จะชี้ไปที่ตารางที่อยู่ในรอม ซึ่งในตารางจะบรรจุข้อมูลที่เป็นอุณหภูมิที่ถูกต้องไว้

ภายในสภาวะในโรงงานอุตสาหกรรมวงจรดังรูป 3.18 จะถูกบรรจุอยู่ใน package และติดตั้งอยู่บน Rack ดังรูป 3.19 ซึ่งเป็นอุปกรณ์ทางอนาลอก Siries 3B ใช้เป็นตัวปรับปรุงสัญญาณ(signal conditioning Module) โดยตัว 3B37 (ดังรูป) เป็นวงจรขยายสัญญาณที่มีการชดเชยรอยต่อเย็น(cold-junction componoation) Rack unit ดังกล่าวมีโครงสร้างที่สามารถที่จะต่อเข้ากับ Module ที่เราจะนำไปประยุกต์ใช้งาน เอ้าต์พุตของมันจะมีทั้งแรงดันและกระแส การส่งสัญญาณโดยใช้กระแสจะมีข้อดีตรงที่ความต้านทานของสายส่งสัญญาณจะไม่มีผลต่อสัญญาณที่เราส่ง โดยปกติการส่งกระแสสัญญาณดังกล่าวในระบบมาตรฐานทางเครื่องมือวัดทางอุตสาหกรรมจะมีค่ากระแสเป็น 4-20 mA โดยมีกระแสที่ 4mA จะแทนเอ้าต์พุตที่เป็นศูนย์ และ 20 mA จะแทนค่าเต็มเสกล (full scal value) เหตุผลที่ทำดังนี้เพราะว่ากระแสเท่ากับศูนย์เป็นการเปิดวงจร กระแสดังกล่าวจะแปลงกลับมาเป็นสัดส่วนกับแรงดันที่ปลายทางด้านรับโดยการให้กระแสดังกล่าวไหลผ่านตัวต้านทาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูป 3.19 Package ของวงจรปรับปรุงสัญญาณสำหรับใช้ในสภาพแวดล้อมโรงงานอุตสาหกรรม

3.15 ทรานสดิวเซอร์ความดันและแรง (Force And Transducer)

ในการที่จะแปลงแรง(Fource) หรือแรงดัน(แรงต่อพื้นที่) ให้เป็นสัดส่วนของสัญญาณทางไฟฟ้า วิธีโดยทั่ว ๆ จะใช้เสตรนเกจน์ และ linear variation differential transformems (LVDTs) ทั้งสองวิธีจะเกี่ยวข้องกับการเคลื่อนที่ของบางสิ่งบางอย่าง ซึ่งเป็นเหตุผลของการใช้คำว่าทรานสดิวเซอร์ แทนที่จะเป็นเซนเซอร์

3.5.1 สเตรนเกจน์และโหลดเซลล์(Strain Gages And Load cells)

สเตรนเกจน์ เป็นตัวต้านทานตัวเล็ก ๆ ค่าความต้านทานของมันจะเปลี่ยนแปลงเมื่อความยาวของมันเปลี่ยนไป อาจจะทำมาจากลวดบาง ๆ หรือ

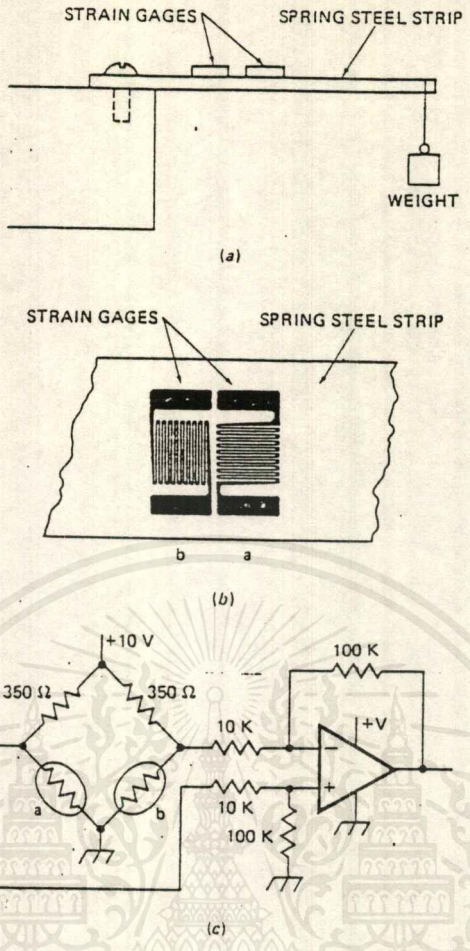
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

อาจจะเป็นวัสดุจำพวกสารกึ่งตัวนำ รูป 3.20(a) เป็นตัวอย่างการติดตั้งเสตรนเกจน์เพื่อใช้งานในการวัดแรงหรือวัดน้ำหนัก จากรูป 3.20(a) ปลายด้านหนึ่งของชิ้นส่วนที่เป็นสปริงจะติดอยู่กับพื้นผิวที่คงที่(ไม่เคลื่อนไหว) เสตรนเกจน์จะติดกาวที่ด้านบนของแท่งที่เคลื่อนที่ได้ แรงหรือน้ำหนักที่จะวัดจะอยู่ปลายอีกด้านหนึ่งของแท่ง(ที่เคลื่อนไหวได้) โดยแรงที่จะวัดจะทำให้แท่งงอ ค่าความต้านทานของเสตรนเกจน์จะเปลี่ยนไปในทางเพิ่มขึ้น ถ้างอมากความต้านทานจะมากเป็นสัดส่วนกับแรงที่มากระทำกับแท่งดังกล่าว ถ้าเราจ่ายกระแสให้ไหลผ่านเสตรนเกจน์ จะทำให้เกิดแรงดันคร่อมเสตรนเกจน์ และแรงดันจะมีค่าเปลี่ยนไปเมื่อความต้านทานของเสตรนเกจน์เปลี่ยนไป

แต่ข้อควรคำนึงก็คือ ค่าความต้านทานของเสตรนเกจน์ขึ้นอยู่กับอุณหภูมิด้วย เพื่อชดเชยปัญหาดังกล่าวจึงต้องมีการต่อเสตรนเกจน์เข้าด้วยกันดังรูป 12.20(b) ซึ่งนิยมใช้มากที่สุด โดยทั้งสองส่วนของเสตรนเกจน์จะเปลี่ยนแปลงค่าความต้านทานเมื่ออุณหภูมิเปลี่ยนไป แต่ชิ้นส่วน A จะเปลี่ยนแปลงความต้านทานต่อแรงที่กระทำ เมื่อนำทั้งสองชิ้นส่วนมาต่อแบบบริจด์สมดุลย์(balance-brigde) ดังรูป 11-20(c) การเปลี่ยนแปลงใด ๆ ของความต้านทานของแต่ละชิ้นส่วนที่เกิดจากอุณหภูมิจะไม่มีผลต่อเอ้าต์พุตของวงจรบริจด์ ตัวอย่างของแรงดันเอ้าต์พุต จากรูป 11.20(c) ประมาณ 2 ถึง 3 mV ต่อแรงดัน 1 voltที่จ่ายเข้าไป เช่นถ้าจ่ายเข้าไป 10 V (ที่บริจด์ ดังรูป) ค่าเอ้าต์พุต ที่ full-load จะมีแรงดัน 20 หรือ 30 mV ซึ่งเราอาจจะนำมาขยาย โดยใช้วงจรขยายความแตกต่าง หรือใช้วงจรขยายเครื่องมือวัด(Instrumentation Amplifier)

อุปกรณ์อนาลอก 3B16 จากรูป 3.19 จะให้แรงดันกระตุ้น = 10V และขยายสัญญาณที่เกิดขึ้นจากเสตรนเกจน์แบบบริจด์ การจะใช้ในรูปแบบอื่น ๆ เพื่อใช้วัดแรงหรือความดันชนิดอื่น ๆ ถ้าเราต่อเสตรนเกจน์แบบบริด์เข้ากับแท่งที่หักงอได้ดังรูป 3.20(a) เราจะเรียกว่า โหลดเซลล์ ซึ่งใช้ในการวัดน้ำหนัก รูป 3.21 จะแสดงการนำโหลดเซลล์จากรูป 3.20(b) ซึ่งอาจจะนำไปใช้เป็นเครื่องชั่งน้ำหนักโดยใช้ไมโครโปรเซสเซอร์ในการควบคุมการทำงาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

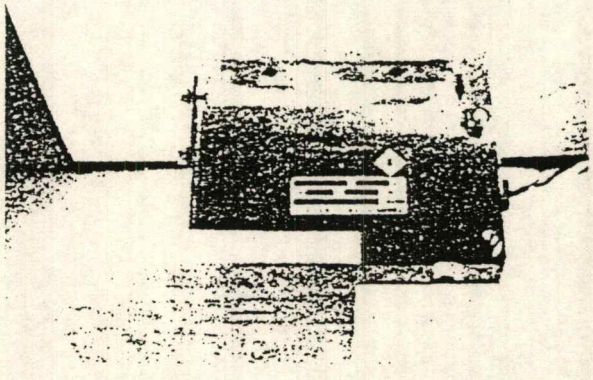


รูป 3.20 สเตรนเกจน์ที่ใช้ในการวัดแรง

(a) ด้านข้าง

(b) ด้านบน(ภาพขยาย)

(c) การต่อในวงจร



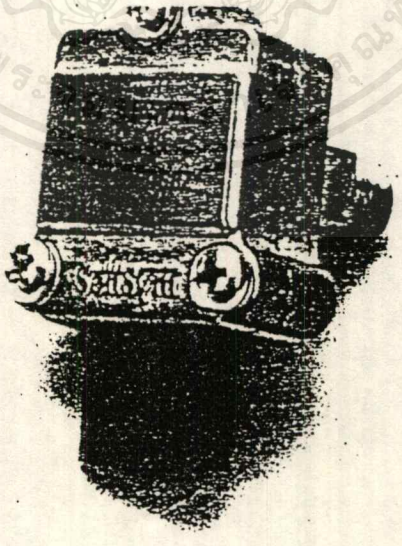
รูป 3.21 ภาพจริงของโหลดเซลล์ที่ใช้ในการวัดน้ำหนัก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ถ้าหากเราติดตั้งเสตรนเกจน์แบบบริดจ์กับแผ่นไดอะแฟรม(diaphram) ที่เคลื่อนที่ได้ในภาชนะที่เป็นเส้นด้าย(threaded housing) เอ้าต์พุตของบริจด์จะเป็นสัดส่วนกับความดันที่จ่ายเข้าไปที่ไดอะแฟรม ถ้าหากภายในเป็นสูญญากาศ ค่าที่อ่านได้จะเป็นค่าความดันสัมบูรณ์(absolute pressure) แต่ถ้าหากอีกด้านหนึ่งของไดอะแฟรมเปิดออก ค่าที่อ่านได้จะเป็นความดันบรรยากาศ(atmosphere pressure) และถ้าหากทั้งสองด้านของไดอะแฟรมต่อเข้ากับแหล่งจ่ายความดัน ค่าที่อ่านได้จะเป็นความแตกต่างของความดันทั้งสอง รูป 3.22 เป็นทรานสดิวเซอร์ความดัน วัดได้ 0 ถึง 15 lb ต่อตารางนิ้ว ทรานสดิวเซอร์ ดังรูป 3.22 อาจจะใช้ในการวัดความดันโลหิตในเครื่องมือวัดทางการแพทย์ โดยใช้ไมโครโปรเซสเซอร์ควบคุม

3.5.2 หม้อแปลงความแตกต่างเชิงเส้น (Linear Variable Differential Tranfermer)

ใช้อักษรย่อว่า LDVT ซึ่งเป็นชนิดหนึ่งของทรานสดิวเซอร์ที่นิยมใช้ในการวัดแรง,วัดความดันหรือวัดตำแหน่ง รูป 3.23 เป็นโครงสร้างพื้นฐานของ

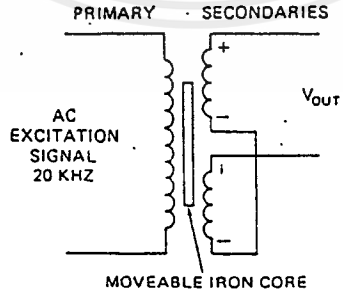


รูป 3.22 ภาพตัวอย่างทรานดิวเซอร์ที่ใช้ในการวัดความกดดัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

LDVT จะประกอบไปด้วยขดลวดจำนวน 3 ขดซึ่งเป็นแบบ wire-wound พันอยู่บนแกน(form)เดียวกัน โดยมีแกนเหล็กที่เคลื่อนที่ได้ สัญญาณกระตุ้นความถี่ 20 KH_Z จะจ่ายไปที่ขดปฐมภูมิ ทางด้านทุติยภูมิจะต่อวงจรแบบที่ให้เกิดแรงดันเหนี่ยวนำที่ตรงข้ามกับขดลวดอีกขดหนึ่ง ถ้าหากแกนของหม้อแปลงอยู่ที่ตรงกลางพอดี แรงดันที่เหนี่ยวนำของขดลวดทางด้านทุติยภูมิจะมีค่าเท่ากัน แรงดันเอ้าพุทจึงมีค่าเป็นศูนย์ ถ้าหากแกนเคลื่อนที่ออกจากจุดศูนย์กลางระหว่างขดลวดทั้งสอง ค่าแรงดันเหนี่ยวนำของขดลวดทั้งสองจะมีค่าไม่เท่ากัน จึงเกิดแรงดันเอ้าพุทปรากฏขึ้น ความสัมพันธ์ของเฟสระหว่างสัญญาณอินพุตและเอ้าต์พุต จะเป็นตัวชี้บอกทิศทางการเคลื่อนที่ของแกนเหล็กจากตำแหน่งศูนย์กลาง ขนาดของสัญญาณเอ้าต์พุตจะเป็นสัดส่วนกับระยะทางที่แกนเคลื่อนไปแบบเชิงเส้น (จากตำแหน่งศูนย์กลาง)

LDVT สามารถที่จะใช้โดยตรงในการวัดการเคลื่อนไป (displacemant) หรือวัดตำแหน่ง ถ้าหากเราเพิ่มสปริงเข้าไปที่แกน จะต้องมีแรงมากระทำจึงจะทำให้แกนเคลื่อนที่ได้ ซึ่งก็จะเป็นการใช้ LDVT ทำเป็นโหลดเซลล์สำหรับเครื่องชั่งน้ำหนักอิเลคทรอนิคส์



รูป 3.23 โครงสร้างของหม้อแปลงความแตกต่างเชิงเส้น

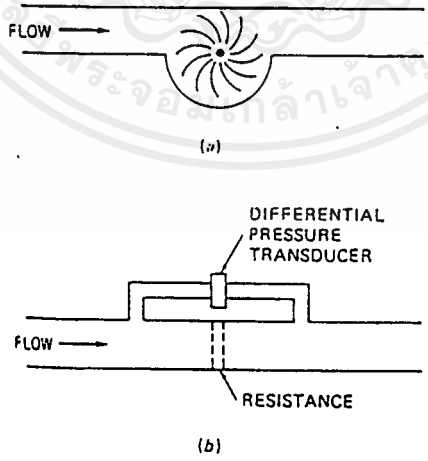
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.5.3 เซนเซอร์ของไหล(Flow Sensors)

ถ้าเราต้องการควบคุมอัตราการไหลของวัสดุสักอย่างหนึ่ง ในโรงงานอุตสาหกรรม(ของเรา) เราจำเป็นจะต้องวัดมันออกมาให้ได้ ซึ่งจะขึ้นอยู่กับวัสดุที่เราจะวัด

วิธีหนึ่งโดยการใช้วงล้อใบพัด(paddle wheel) ดังรูป 3.24(a) จำนวนรอบของการหมุนจะขึ้นอยู่กับอัตราการไหล(Flow Rate) ของของเหลวหรือก๊าซ เราอาจจะต่อตัวเข้ารหัสทางแสง(optical encoder) เข้าที่แกนของวงล้อใบพัด ซึ่งจะได้ข้อมูลทางดิจิตอลที่ขึ้นอยู่กับอัตราความเร็วในการหมุนของวงล้อใบพัด

วิธีที่สองที่นิยมใช้กันก็คือ ใช้ differantial-pressure ทรานสดิวเซอร์ ดังรูป 3.24(b) โดยการใส่ขดลวดเป็นตระแกรงไว้ในท่อเพื่อสร้างค่าความต้านทานขึ้นมาค่าหนึ่ง อัตราการไหลที่ผ่านตัวต้านทานจะทำให้เกิดความดันแตกต่างกันระหว่างสองด้านของตัวต้านทาน ตัวทรานสดิวเซอร์ความดันจะ



รูป 3.24 เซนเซอร์ของไหล

(a) แบบวงล้อใบพัด

(b) แบบความดันแตกต่าง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ให้เอ้าต์พุต ที่เป็นความแตกต่างระหว่างทั้งสองด้านของตัวต้านทาน ดังนั้นเอ้าพุทของทรานสดิวเซอร์ความดันจะเป็นสัดส่วนกับอัตราการไหลของของเหลว หรือก๊าซในท่อ

3.6 เซนเซอร์ชนิดอื่นๆ

ยังมีอีกมากมายหลายชนิดของเซนเซอร์หากไม่สามารถนำมากล่าวในที่นี้ได้หมด เช่น มีเซนเซอร์ที่ใช้วัด pH, วัดความหนาของวัสดุ หรือวัดทุกสิ่งทุกอย่างที่เราอยากจะวัด ซึ่งก็จะมีทรานสดิวเซอร์ใช้วัดสิ่งเหล่านั้นมากมาย เช่น ถ้าเราต้องการหาระดับของของเหลวในแท้งค์(tank) เราจะต้องติดตั้งทรานสดิวเซอร์ความดันที่ก้นของแท้งค์ ความดันของของเหลวจะเป็นสัดส่วนกับความสูงของของเหลวในแท้งค์เป็นต้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 4

การแปลงดิจิตอลเป็นอนาลอก

(DIGITAL TO ANALOG CONVERSION)

4.1 แนะนำ

การแปลงดิจิตอลเป็นอนาลอก จะใช้วงจรหรืออุปกรณ์ที่มีหน้าที่แปลงสัญญาณดิจิตอลซึ่งอาจจะเป็นแรงดันหรือกระแส ให้เป็นสัญญาณอนาลอกที่เปิดสัดส่วนกับสัญญาณดิจิตอลที่ป้อนเข้าไปที่อินพุตของวงจร เราสามารถเขียนสมการเอาต์พุตของการแปลง D/A ได้ดังนี้

$$X = K*A*B \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots (4.1)$$

โดยที่ X = เป็นแรงดันหรือกระแสทางด้านเอาต์พุต(อนาลอก)

A = ค่าอ้างอิงของอนาลอก(แรงดันหรือกระแส)

B = จำนวน(ค่า)ของตัวเลขไบนารี

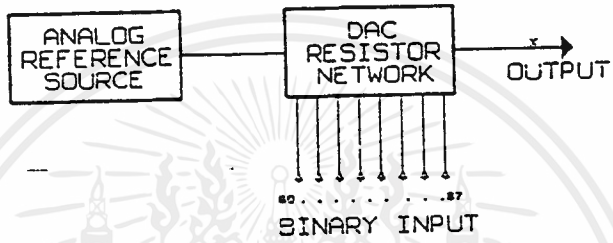
K = ค่าคงที่จะมีค่าเป็น 1 เสมอ

รูป 4.1 แสดงส่วนประกอบต่าง ๆ ของระบบการแปลงดิจิตอลเป็นอนาลอกโดยที่แหล่งจ่ายอ้างอิงของอนาลอกจะมีค่าคงที่ค่าหนึ่ง อาจจะเป็นแหล่งจ่ายแรงดันหรือกระแสที่ความเที่ยงตรงสูง จุดประสงค์เพื่อใช้ในการเปรียบเทียบกับอินพุตเพื่อจะสร้างแรงดันหรือกระแสที่เอาต์พุต

แหล่งจ่ายอ้างอิงของอนาลอกนี้ ถ้าอยู่ภายในตัวแปลง D/A จะเรียกว่า Non-multiplying-D/A ถ้าอยู่ภายนอกจะเรียกว่า Multiplying D/A ซึ่งถ้าอยู่ภายนอกจะต้องมีการออกแบบโดยการใช้แหล่งจ่ายอ้างอิงภายนอก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การแปลงดิจิตอลเป็นอนาลอก จะแบ่งตามการใช้ตัวต้านทาน ซึ่งจะมีการใช้ 2ตัวต้านทานโดยทั่วๆ ไปอยู่ 2 ลักษณะ คือแบบใช้ตัวต้านทานแบ่งน้ำหนักที่อินพุตของวงจร (Binary weighted resister ladder) และแบบใช้ตัวต้านทานขั้นบันได(R-2R ladder)



รูป 4.1 ส่วนประกอบของการแปลงดิจิตอลเป็นอนาลอก

4.2 การแปลง D/A แบบใช้ตัวต้านทานแบ่งน้ำหนัก(Binary Weighted Resis ter Ladder)

การใช้ตัวต้านทานแบบนี้ จะเป็นการแปลงข้อมูลดิจิตอลให้เป็นอนาลอกโดยตรง จะมีตัวต้านทานต่ออนุกรมอยู่กับแรงดันอ้างอิง(Vref) โดยจะมีอิเลคโทรนิคส์สวิชใช้ในการเปิด-ปิดสัญญาณ ตามสถานะของสัญญาณดิจิตอลซึ่งมีได้ 2 ระดับคือลอกจิค "0" กับ "1" การต่อลักษณะนี้ค่าของตัวต้านทานจะมีค่าในแต่ละน้ำหนักที่คูณด้วย 2 ตลอดคือจะมีตั้งแต่ R, 2R, 4R จนถึง nR หรือเขียนเป็นสูตรก็คือ $2^{(n-1)}R$ โดยที่ n คือจำนวนของบิตในการแปลง

เมื่อตัวต้านทานถูกต่อลงกราวด์ จะไม่มีการระแสไหลผ่านตัวต้านทานที่ต่ออยู่ แต่ ถ้าหากตัวต้านทานถูกต่อกับแรงดันอ้างอิง(แทนด้วย ระดับลอจิกที่ตรงกันข้ามกับ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การต่อตัวต้านทานลงกราวด์) จะมีกระแสไหลผ่านตัวต้านทาน ดังนั้นถ้าหากตัวต้านทานค่า R และ 2R ถูกต่อกับแรงดันอ้างอิงพร้อมกัน จะมีกระแสไหลผ่านตัวต้านทาน $R = V_{ref} / R$ และไหลผ่านตัวต้านทาน $2R = V_{ref}/ 2R$ ค่ากระแสทั้งสองค่า จะไหลมารวมกันที่จุดผสม(summing point) ดังรูป 9.2 เขียนเป็นสมการได้ดังนี้

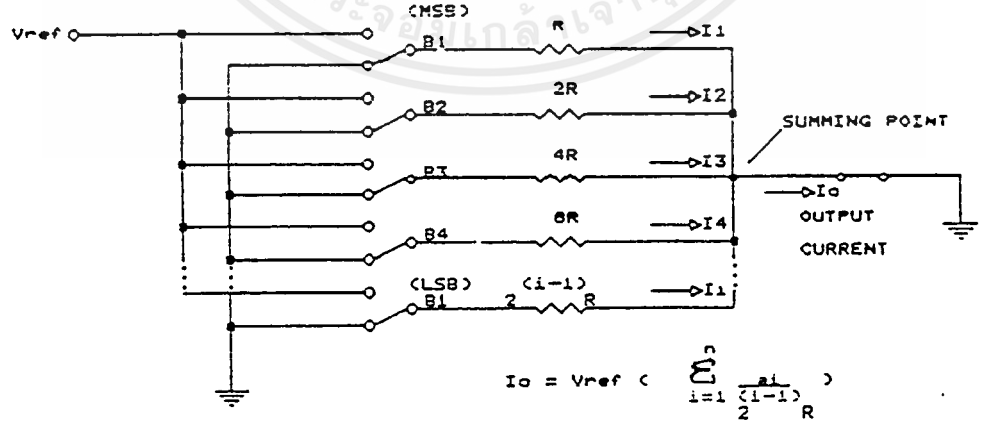
$$I_o = V_{ref} - 6\, {}^{ai}/_{2}(i-1).R \qquad \text{.....(4.2)}$$

โดยที่ I_o = กระแสเอาต์พุต(สัญญาณอนาลอก) มีหน่วยเป็น A

ai = ค่าตัวเลขไบนารีอินพุต("0" หรือ "1")

R = ตัวต้านทานตัวแรกที่มีค่าต่ำสุด(ของบิตนัยสำคัญสูงสุด-MSB-most Significant Bit)

จากรูป 4.2 จะเห็นว่าตัวต้านทาน ที่ทำให้กระแสไหลได้สูงสุดคือ R ดังนั้นตัวต้านทาน R จะเป็นตัวต้านทานของบิตนัยสำคัญสูงสุด(MSB-Mostsignificant Bit)



รูป 4.2 วงจรแปลง D/A แบบใช้ตัวต้านทานแบ่งน้ำหนัก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากรูป 4.2 จะเห็นว่าค่าของตัวต้านทานจะมีค่าเพิ่มขึ้นทีละมากๆ ถ้าหากมีการแปลงขนาด 8-บิต ถ้าหาก R มีค่าเท่ากับ 10K ตัวต้านตัวที่ 8 จะมีค่า 2^7R =128R = 128*10K = 1280K หรือ 1.28 M จะมีกระแสไหลผ่านตัวต้านทานค่าขนาดนี้น้อยมาก และจากธรรมชาติที่ว่า ตัวต้านทานค่ามากจะสร้างได้ยากและผลกระทบจากสิ่งแวดล้อมภายนอก คือความร้อนจะทำให้ค่าของความต้านทานเปลี่ยนไป(โดยเฉพาะตัวต้านทานที่มีค่ามากๆ) ทำให้ความละเอียดแม่นยำของการใช้ตัวต้านทานแบบนี้ลดลง เมื่อจำนวนบิตของการแปลงสูงขึ้น เราจะแก้ไขได้โดยการใช้ตัวต้านทานที่มีค่าอยู่ 2 ค่าเท่านั้นคือ แบบตัวต้านทานขั้นบันได(R-2R ladder) ดังจะได้กล่าวถึงต่อไป

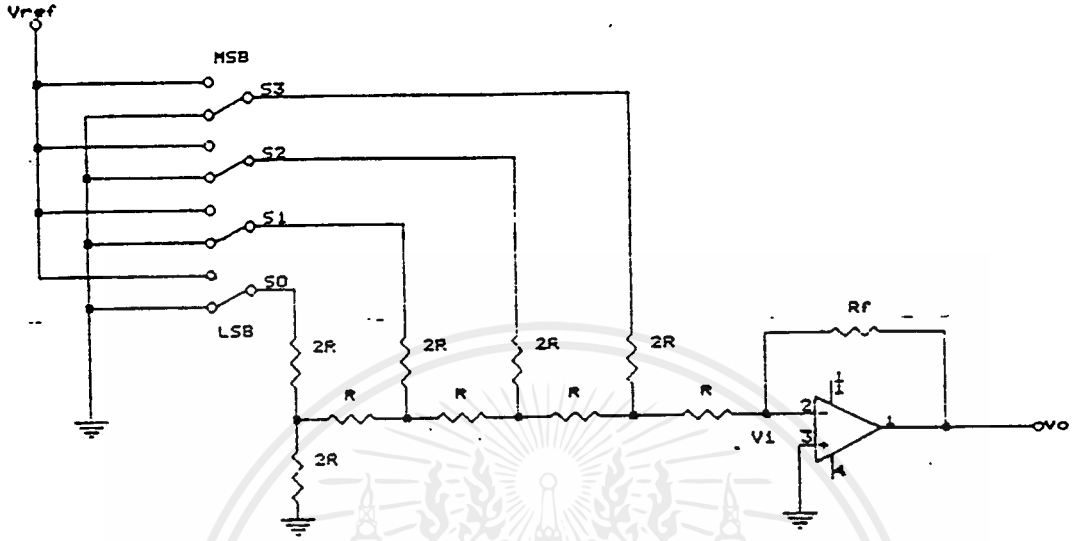
4.3 การแปลง D/A แบบใช้ตัวต้านทานขั้นบันได (R-2R Ladder D/A Converter)

วงจรเบื้องต้นแสดงดังรูป 4.3(a) จะเห็นว่ามีตัวต้านทานอยู่ 2 ค่าคือ R กับ 2R จากรูป 4.3(b) แสดงให้เห็นการต่อตัวต้านแบบขั้นบันไดให้ชัดเจนขึ้น การต่อตัวต้านแบบนี้ มีคุณสมบัติที่น่าสนใจก็คือ ถ้ามองเข้าที่ขั้วใดๆ ของ A, B, S0, S1, S2 หรือ S3 โดยขั้วที่ไม่มอง(ขั่วที่เหลือ) ต่อลงกราวด์จะมองเห็นค่าความต้านทานมีค่าเท่ากับ 3R เช่น มองที่ A โดยที่ S0,S1,S2,S3 และ B ต่อลงกราวด์ ค่าความต้านทานมองไปที่ A เทียบกับกราวด์จะมีค่า 3R

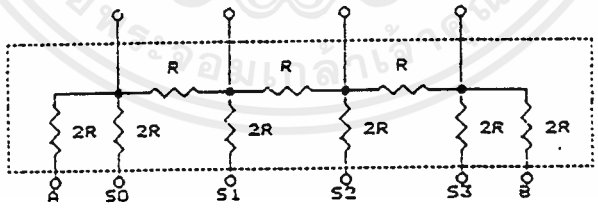
เพื่ออธิบายการทำงาน สมมุติว่ารูป 4.3(a) สวิชทุกตัวต่อลงกราวด์ยกเว้น S0 จะเขียนวงจรใหม่ได้ดังรูป 4.4(a)(เพื่อให้มองเห็นได้ชัดเจนขึ้น) โดยการใช้หลักการเทวินิน ดังรูป 4.4(b) ถึง d ในรูป 10.4(b) ที่ mode 3 เราจะแทนแรงดันเทวินินด้วยค่า Vr/2(Vr คร่อมตัวต้านทานสองตัวที่ค่าเท่ากันย่อมมีค่าเท่ากับครึ่งหนึ่ง) และความต้านทานเทวินินจะมีค่า = 2R//2R=R ที่ Node อื่นก็จะเป็นลักษณะเดียวกัน แรงดันเทวินินจะมีค่าลดลงเรื่อยๆ ตามการแบ่งแรงดันของ R รูป 4.4(d) จะได้แรงดันเทวินินเท่ากับ Vr/16 และความต้านทานมีค่าเท่ากับ 3R ต่อกับออปแอมพ์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ถ้าเราต่อ S1 เข้ากับ Vr โดยที่เหลือต่อลงกราวด์ จะได้แรงดันเทวินินมีค่า = Vr/8 โดยที่ค่าความต้านมีค่าเท่ากับ 3R เท่าเดิม



(a)



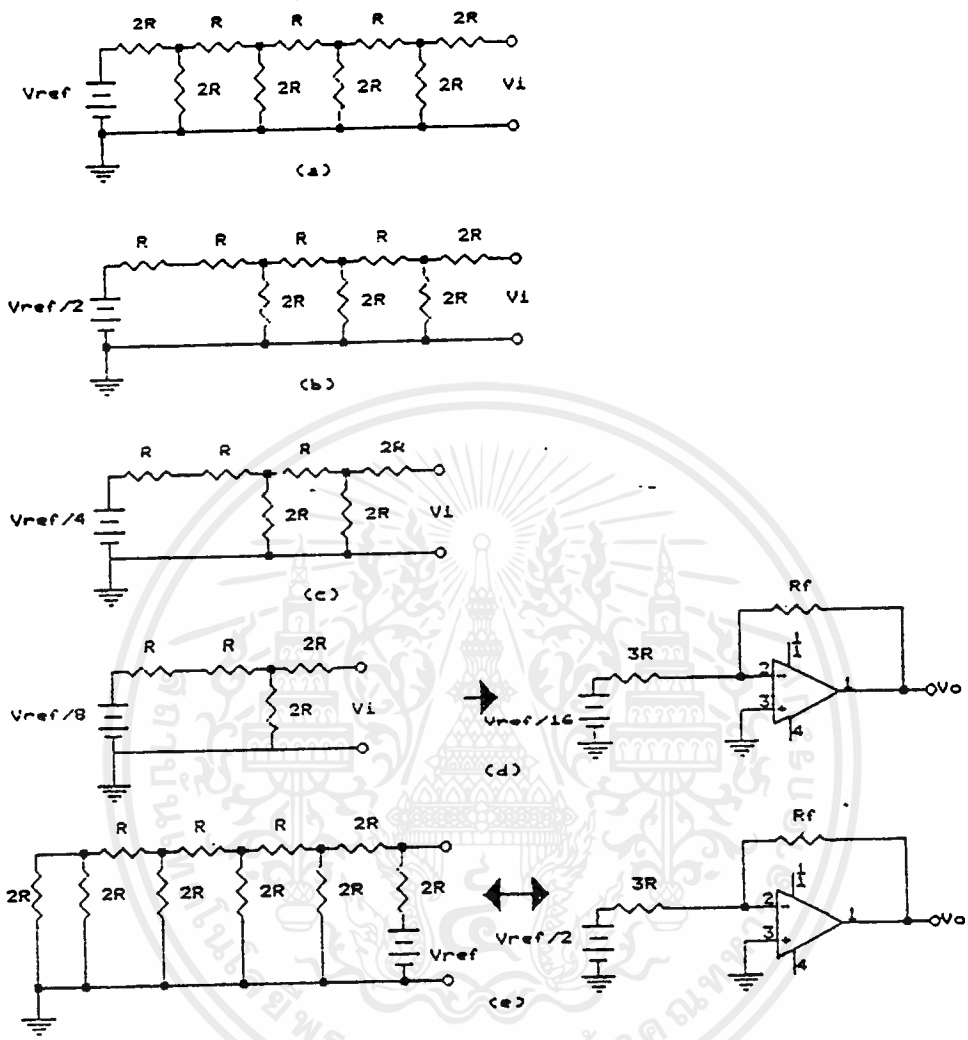
(b)

รูป 4.3 การแปลง D/A แบบใช้ตัวต้านทานขั้นบันได

(a) วงจร

(b) โครงข่ายตัวต้านทาน R-2R แลดเดอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



. รูป 4.4 การวิเคราะห์วงจรของตัวต้านทาน R-2R แลดเดอร์

(a)วงจรเมื่อ LSB = "1"

(b)ใช้ทฤษฎีเทวินินเปลี่ยนครั้งที่ 1

(c)เปลี่ยนครั้งที่ 2

(d)เปลียนครั้งที่ 3

(e)วงจรเมื่อ MSB = "1"

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

และถ้าต่อ S3 กับ Vr โดยที่เหลือต่อลงกราวด์ ดังรูป 4.4(c) จะได้แรงดันเทวินินมีค่า Vr/2 โดยที่ค่าความต้านทานมีค่าเท่ากับ 3R เท่าเดิม

จากที่ได้อธิบายมาเราจะเขียนเป็นสมการได้ดังนี้

$$V_o = -(Rf.Vr/3R).(S3/2^1 + S2/2^2 + S1/2^3 + S0/2^4) \quad .....(4.3a)$$

คูณด้วย 2^4 ทั้งสองข้างจะได้

$$2^4 \cdot V_o = -(Rf.Vr/3R).(2^3.S3 + 2^2.S2 + 2^1.S1 + 2^0.S0)$$

หารด้วย 2^4 ทั้งสองข้างจะได้

$$V_o = -(Rf.Vr/48R).(2^3 \cdot S3 + 2^2.S2 + 2^1.S1 + 2^0.S0) .....(4.3b)$$

โดยสมการที่ได้มาจากการใช้หลักการของ Supperposition โดยที่สวิช Si จะต่อกับ Vr เมื่อ Ai = "1" และ Si จะต่อลงกราวด์เมื่อ Ai = "o"

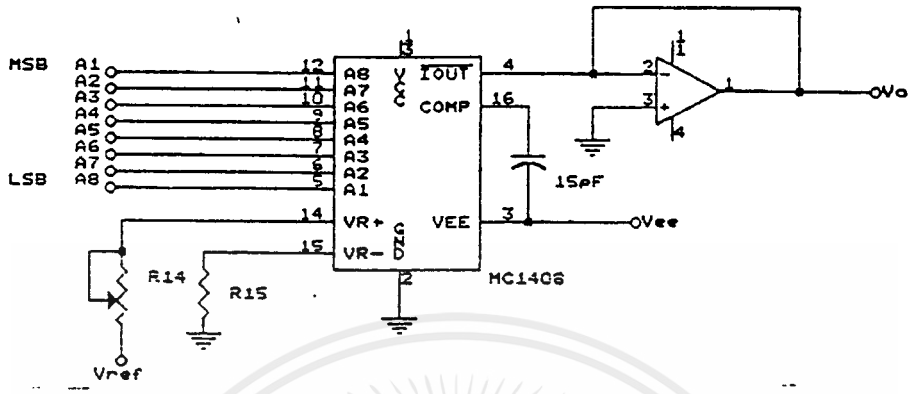
4.4 คุณสมบัติและข้อกำหนดของตัวแปลงดิจิตอลเป็นอนาลอก (D/A Characteristies And Specification)

รูป 4.5 แสดงไอซี ที่เป็นตัวแปลงสัญญาณดิจิตอลเป็นอนาลอก ที่ใช้ออปแอมป์เป็นตัวแปลงกระแสที่ได้จากเอาต์พุตของตัวแปลง D/A ไปเป็นแรงดันเราจะใช้วงจรนี้สำหรับการอธิบาย คุณสมบัติของตัวแปลงสัญญาณดิจิตอลเป็นอนาลอก

4.4.1 ความละเอียด (Resolution)

คุณสมบัติประการแรก ของตัวแปลงสัญญาณดิจิตอลเป็นอนาลอกที่คือความละเอียด(Resolution)ของการแปลง พิจารณาจากจำนวนของบิตของเลข

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูป 4.5 วงจรแปลงดิจิตอลเป็นอนาลอกใช้วงจรรวมหมายเลข MC1408

ฐานสองที่ป้อนเข้าที่อินพุตของตัวแปลง D/A ถ้าอินพุตเป็นเลขฐานสอง 8 ตัว ดังเช่นในรูป 4.5 มีระดับของเอาต์พุตที่เป็นไปได้เท่ากับ $2^8 = 256$ ระดับดังนั้น Resolution ของมันคือ 1 ใน 256 ถ้าตัวแปลง D/A มีสัญญาณอินพุตขนาด 12 บิต จะมี Resolution เท่ากับ 1 ใน 2^{12} หรือ 4096 บางครั้งจะคิดในรูปเปอร์เซ็นต์ เช่น Resolution ขนาด 8 บิต คือประมาณ 0.39 เปอร์เซ็นต์

4.4.2 ค่าเต็มพิกัดของแรงดันเอาต์พุต (Full Scale Output Voltage)

คุณสมบัติข้อต่อไปของดิจิตอลเป็นอนาลอกคือ ค่าเต็มพิกัดของแรงดันเอาต์พุต(full seale output voltage) สำหรับตัวแปลงสัญญาณในรูป 4.5 กระแสสำหรับสวิชทั้งหมดจ่ายโดย Vref ผ่าน R_{14} กระแสเอาต์พุตจากขา 4 ของตัวแปลง D/A ถูกดึงให้ไหลผ่าน Ro เพื่อสร้างแรงดันเอาต์พุต สูตรสำหรับ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

แรงดันเอาต์พุต ดังแสดงในรูป 4.5 ในสมการ A_1 แทนสภาวะของสวิชสำหรับบิตที่ 1 ถ้าสวิชปิดวงจรจะเป็นการยอมให้กระแสไหล (จะมีค่าเป็น "1" ในบิตนั้น ๆ ถ้าสวิชเปิดวงจรหมายถึงมีค่าเป็น "0" ในบิตนั้น ๆ) ในรูป 4.5 ถ้าสวิชทั้งหมดปิดวงจร เอาต์พุตจะมีค่า (10V)(255/256) = 9.961V แม้ว่าแรงดันเอาต์พุตไม่สามารถมีค่าถึง 10V ได้ แต่จะเป็นการอ้างอิงกับแรงดัน 10V ดังนั้นแรงดันเอาต์พุตสูงสุดของตัวแปลง D/A มีค่าน้อยกว่าค่าของแรงดันอ้างอิงอยู่เท่ากับค่าของ 1 บิตนัยสำคัญต่ำสุดเสมอ อีกตัวอย่างหนึ่งคือ สมมติตัวแปลงสัญญาณมี 12 บิต มี Vref = 10V ค่าบิตนัยสำคัญต่ำสุด(LSB) จะเป็น (10V)/(4096) = 2.44mV ดังนั้นค่าแรงดันสูงสุดของตัวแปลง D/A = 10 - 0.00244V หรือ 9.99756V

4.4.3 ความเที่ยงตรง(Accuracy)

ข้อกำหนดความเที่ยงตรง สำหรับตัวแปลงสัญญาณดิจิตอลเป็นอนาลอกคือการเปรียบเทียบระหว่าง เอาต์พุตที่เป็นจริงกับเอาต์พุตที่คาดหวังไว้ จะเป็นข้อกำหนดคล้ายกับข้อกำหนดเป็นเปอร์เซ็นต์ ของแรงดันเต็มพิกัดของเอาต์พุต(หรือกระแส) ถ้าตัวแปลง D/A มีค่า แรงดันเต็มพิกัดของเอาต์พุต 10V และค่าเที่ยงตรง 0.2% ดังนั้นค่าผิดพลาดสูงสุดสำหรับเอาต์พุตใด ๆ จะเป็น 0.002x10.00V เท่ากับ 20mv สำหรับค่าผิดพลาดสูงสุดของตัวแปลงสัญญาณดิจิตอลเป็นอนาลอกทางอุดมคติ ควรจะมีค่าไม่มากกว่า 1/2 ของบิตนัยสำคัญต่ำสุด (1/2 LSB)

4.4.4 ความเป็นเชิงเส้น(Linearity)

ข้อกำหนดอีกอันหนึ่งที่สำคัญ สำหรับตัวแปลงสัญญาณดิจิตอลเป็นอนาลอกคือ ความเป็นเชิงเส้น(Linearity) คือ การวัดความเบี่ยงเบนเท่าไรของเอ้าพุตที่เป็นพิสัยจากเส้นตรง ที่ตัวแปลงสัญญาณเป็นขั้นบันไดจากไม่มีสวิชทำงานเลยถึงสวิชทุกตัวทำงานหมด ค่าเบี่ยงเบนในทางอุดมคติของเอาต์พุตจากเส้นตรงควรมีค่าไม่มากไปกว่า +—1/2 ของค่าของ LSB อย่างไรก็ตามตัวแปลง D/A ส่วนมาก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ในท้องตลาดมีความเป็นเชิงเส้นมีค่าผิดพลาดมากกว่าค่านี้ ตัวอย่างเช่น National semiconducter DAC1020,DAC1021, ข้อกำหนดของความเป็นเชิงเส้นสำหรับ DAC1020 คือ 0.051% ซึ่งเหมาะสมสำหรับการแปลงขนาด 10 บิต DAC1021 มีข้อกำหนดความเป็นเชิงเส้นเป็น 0.10% ซึ่งอาจจะมีคำถามว่า จะดีหรือไม่ถ้ามีตัวแปลง D/A ขนาด 10 บิต ถ้าความเป็นเชิงเส้นมีค่าเทียบเท่ากับ 9บิต หรือ 8 บิต? คำตอบจะขึ้นอยู่กับความตอ้งการในการนำไปประยุกต์ใช้งาน จะมีอุปกรณ์บางอย่างที่ต้องการสัญญาณเอาต์พุตที่มีขนาดเล็ก ๆ ต้องการความละเอียดขนาด 10 บิต แต่จะไม่เป็นไรถ้าต้องการค่าเอาต์พุตที่ไม่เป็นเชิงเส้นสำหรับสัญญาณที่มีขนาดใหญ่ ๆ ราคาของตัวแปลงสัญญาณดิจิตอลเป็นอนาลอกไม่ขึ้นกับ resolution เท่านั้น แต่ยังเป็นขึ้นกับคุณสมบัติเฉพาะของความเป็นเชิงเส้นด้วย

4.4.5 ค่าเวลาในการเปลี่ยนแปลง(Setling Time)

ข้อกำหนดที่สำคัญอีกข้อหนึ่งคือค่าเวลาในการแปลง(setling time) เมื่อเกิดการเปลี่ยนแปลงค่าของฐานสองที่ป้อนให้กับอินพุตของตัวแปลงสัญญาณ เอ้าพุตจะเปลี่ยนแปลงเป็นค่าใหม่ได้ถูกต้อง จะต้องมีเวลาค่าหนึ่งขึ้นอยู่กับข้อกำหนดของเวลาในการแปลงของตัวแปลง D/A ค่าเวลาที่เอาต์พุตใช้ไปภายใน 1/2 LSB ของค่าสุดท้ายจะเรียกว่าค่าเวลา "Setting time" ดังเช่นตัวอย่าง DAC1020 ซึ่งเป็นตัวแปลง D/A ขนาด 10 บิต มีค่า "Setting time" อยู่ 500nS สำหรับค่าการเปลี่ยนแปลงเต็มพิกัดของเอาต์พุต คุณสมบัติข้อนี้สำคัญเพราะ 2ถ้าตัวแปลงสัญญาณทำงานที่ความถี่สูง ๆ มันอาจจะไม่มีเวลาจะตั้งค่าก่อนที่มันจะสวิชไปสู่อีกสถานะหนึ่ง

4.5 การใช้งานและการเชื่อมโยงตัวแปลงสัญญาณดิจิตอลเป็นอนาลอกกับคอมพิวเตอร์(D/A Applications and interfacing to computer)

ตัวแปลงสัญญาณดิจิตอลเป็นอนาลอก มีการใช้งานได้มากมายนอกเหนือการใช้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

งานร่วมกับไมโครคอมพิวเตอร์ ตัวอย่างในเครื่องเสียงที่เป็น compact-disk ตัวแปลงสัญญาณดิจิตอลเป็นอนาลอกขนาด 13-14 บิต ที่ใช้เปลี่ยนข้อมูลฐานสองที่อ่านได้จากแผ่น disk โดยเลเซอร์ไปเป็นสัญญาณเสียงที่เป็นอนาลอก IC จำพวกสังเคราะห์เสียงประกอบอยู่ในตัวแปลงสัญญาณดิจิตอลเป็นอนาลอก เพื่อเปลี่ยนแปลงข้อมูลเลขฐานสองที่ถูกเก็บไว้สำหรับคำพูดไปเป็นสัญญาณเสียง ที่เป็นอนาลอก อย่างไรก็ตามในที่นี้เราจะกล่าวถึงเบื้องต้น ในการใช้ตัวแปลงสัญญาณดิจิตอลเป็นอนาลอกกับไมโครคอมพิวเตอร์

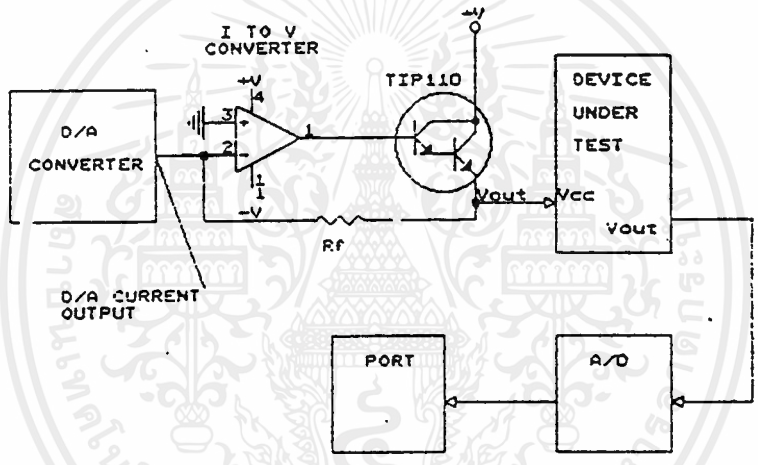
อินพุต A_1 ถึง A_8 ของตัวแปลงดิจิตอลเป็นอนาลอกในรูป 9.5 สามารถต่อโดยตรงกับเอาต์พุตพอร์ตของไมโครคอมพิวเตอร์ เราสามารถใช้โปรแกรม สร้างแรงดันที่ต้องการ โดยการส่งข้อมูลของแรงดันออกมาที่เอาต์พุตพอร์ตของตัวแปลงดิจิตอลเป็นอนาลอก เป็นความคิดริเริ่มที่ว่าควรจะใช้วงจรนี้ทำอะไรได้บ้าง?

ตัวอย่างแรก สมมติถ้าคุณต้องการที่จะสร้างตัวทดสอบ(Tester) ที่ควบคุมด้วยไมโครคอมพิวเตอร์ ซึ่งเราต้องการหาผลกระทบของการเปลี่ยนแปลงแรงดันของแหล่งจ่ายกำลัง กับแรงดันเอาต์พุตของแผงวงจรรวมของเครื่องขยาย (Amplifier)ถ้าเราต่อเอาต์พุตของตัวแปลงสัญญาณดิจิตอลเป็นอนาลอกเข้ากับแรงดันอ้างอิงอินพุตของแหล่งจ่ายกำลังชนิดที่โปรแกรมได้ หรือทำให้ง่ายเข้าด้วยการเพิ่มวงจรบัฟเฟอร์แบบกระแสสูง ดังแสดงในรูป 4.6 เข้ากับเอาต์พุตของตัวแปลงสัญญาณดิจิตอลเป็นอนาลอก คุณมีแหล่งจ่ายกำลังที่สามารถจะปรับค่าได้ภายใต้การควบคุมจากโปรแกรม เพื่อที่จะให้รู้ค่าแรงดันเอาต์พุตของIC ภายใต้การทดลองปรับค่าแรงดันของแหล่งจ่ายกำลัง เราก็เพียงต่ออินพุตของตัวแปลงสัญญาณอนาลอกเป็นดิจิตอลกับเอาต์พุตของเครื่องขยาย(Amplifier) และต่อเอาต์พุตของตัวแปลงสัญญาณอนาลอกเป็นดิจิตอลกับอินพุตพอร์ตของไมโครคอมพิวเตอร์ เราก็จะสามารถอ่านค่าแรงดันเอาต์พุตของเครื่องขยาย(Amplifier) ได้ดังรูป 10.6

การใช้งานอื่น ๆ เช่นอาจจะใช้ตัวแปลง D/A และบัฟเฟอร์กำลังสูง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สำหรับปรับค่าแหล่งจ่ายแรงดันให้กับ heater ที่เป็นความต้านทานที่มีค่าน้อย ๆ ภายใต้การควบคุมของโปรแกรม ความเร็วของดีซีมอเตอร์ขนาดเล็กจะเป็นสัดส่วนกับจำนวนของกระแสที่ผ่านตัวดีซีมอเตอร์ ดังนั้นเราสามารถต่อดีซีมอเตอร์ตัวเล็ก ๆ เข้ากับเอาต์พุตของบัฟเฟอร์กำลังสูง(power buffer) และควบคุมความเร็วของมอเตอร์ด้วยค่าเอาต์พุตของตัวแปลงสัญญาณดิจิตอลเป็นอนาลอก ข้อควรระวัง ถ้าไม่มีการควบคุมการป้อนกลับ(feed back) ความเร็วของมอเตอร์จะเปลี่ยนแปลงถ้าหากโหลดมีการเปลี่ยนแปลง



รูป 4.6 บัฟเฟอร์กำลังสูงสำหรับต่อกับเอาต์พุตจากตัวแปลง D/A

เราได้พูดถึงการใช้ตัวแปลง D/A ขนาด 8 บิต กับไมโครโปรเซสเซอร์ ในการเชื่อมโยงตัวแปลง D/A ขนาด 8 บิตทำได้ง่าย ๆ โดยการต่ออินพุตของตัวแปลง D/A กับเอาต์พุตพอร์ต(ของไมโครโปรเซสเซอร์) ถ้าไมโครโปรเซสเซอร์มีขนาด 8 บิต แต่ถ้าเราต้องการความละเอียดขนาด 12 บิต จะต้องแบ่งออกเป็น 8 บิตล่างกับพอร์ตหนึ่ง และอีก 4 บิตบนกับอีกพอร์ต แยกกัน เมื่อต้องการส่งข้อมูลออกมาเราจะต้องทำการเขียน 2 ครั้ง อย่างไรก็ตามเป็นปัญหาที่เกิดขึ้นจาก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

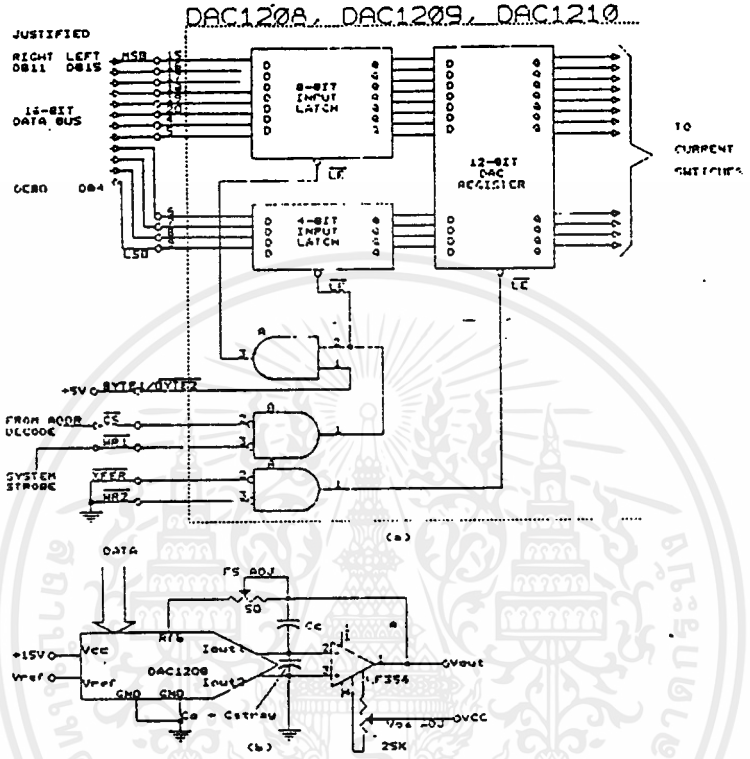
การเขียน 2 ครั้ง เช่นสมมติว่าถ้าคุณต้องการเปลี่ยนแปลงเอาต์พุตของตัวแปลงD/A ขนาด 12 บิต จากค่า 0000 1111 1111 เป็นค่า 0001 0000 0000 เมื่อเขียน 8 บิตล่าง เอาต์พุตจะเปลี่ยนจาก 0000 1111 1111 เป็น 0000 0000 0000 เมื่อทำการเขียน 4 บิตบน ค่าเอาต์พุต จะกลับไปสู่ค่าที่ต้องการคือ 0001 0000 0000 จุดนี้สำหรับเวลาระหว่างการเขียนเอาต์พุต 2 ครั้งกับค่าที่ไม่ต้องการ ในบางระบบอาจทำให้เกิดการผิดพลาดเสียหาย เราสามารถแก้ปัญหานี้ได้โดยใส่ แลทช์(Latches) ในด้านอินพุตแลทช์ สามารถที่จะโหลดข้อมูลแยกกันได้ และมีการสโตรปเข้าด้วยกันเพื่อการผ่านสัญญาณดิจิตอลทั้ง 12 บิตในเวลาเดียวกันได้

ตัวแปลงสัญญาณดิจิตอลเป็นอนาลอก ส่วนมากที่มีอยู่ในปัจจุบัน ประกอบด้วยแลทช์อยู่ภายใน(เพื่อทำให้ง่ายต่อการใช้งาน) รูป 4.7(a) แสดงบล็อกไดอะแกรมของ DAC1230 และ DAC1208 ขนาด 12 บิต แสดงแลทช์และวีจีสเตอร์ที่อยู่ภายใน DAC1230 มีส่วนที่เป็น 4 บิตบนต่อกับ 4 บิตล่าง ดังนั้นการเขียนขนาด 12 บิตสามารถเขียนได้ 2 ครั้ง จากพอร์ตหรือบัสข้อมูลขนาด 8 บิต เช่นไมโครโปรเซสเซอร์เบอร์ 8088 เป็นต้น สำหรับ DAC1208 แต่ละ 4 บิตอินพุตจะแยกต่างหากออกจากกัน ดังนั้นจึงสามารถต่อโดยตรงกับบัสในระบบที่มีขนาด 16 บิตได้ ดังแสดงในรูป 4.7(a) ตัวอย่างเช่น ถ้าต้องการต่อ DAC1208 กับบอร์ด SDK-86 สามารถต่ออินพุต DAC1208 เข้ากับบัสข้อมูลที่ 12 บิตล่างและต่ออินพุต CS กับตัวถอดระหัสของแอดเดรสของเอาต์พุต ต่ออินพุต WR1 กับ WR ของระบบ และต่อ WR2 และ XFER อินพุตลงกราวน์ อินพุต BYTE1/BYTE2 ต่อกับ VCC การส่งข้อมูลออกไปที่พอร์ตจะส่งไป 16 บิต ค่าพารามิเตอร์ของฝั่งเวลาสำหรับ DAC 1208 จะใช้ได้กับ 8086 ที่ความถี่นาฬิกา 5 เมกกะเฮิรต์(หรือน้อยกว่านี้) สำหรับความถี่ของนาฬิกาของ 8086 ที่มากกว่านี้ควรจะเพิ่มวงจรที่มีสถานะคอย(wait) ในแต่ละเวลาที่เขียน(write) ไปที่ตัวแปลงสัญญาณดิจิตอลเป็นอนาลอก

ตัวแปลงสัญญาณดิจิตอลเป็นอนาลอก ต้องการความเที่ยงตรงของ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

แรงดันอ้างอิงมาก วงจรในรูป 4.7(b) ใช้แรงดันอ้างอิง -10.00V ตัวแปลง D/A มีเอาต์พุตเป็นกระแส ดังนั้นจึงต้องใช้ออปแอมป์เปลี่ยนกระแสให้เป็น



รูป 4.7 (a) National DAC1208 12BIT D/A แสดงให้เห็นแลทช์ที่อยู่ภายใน

(b) แสดงการต่อกับส่วนที่เป็นอนาลอก

แรงดันดังรูป โดยใช้ FET ออปแอมพ์ เป็นตัวขยายสัญญาณ เพราะกระแสใบอัสอินพุต ของอินพุต(ของออปแอมพ์)แบบไบโพลาร์ จะมีผลต่อความเที่ยงตรงของเอาต์พุต DAC1208 และ DAC1230 มีความต้านทานป้อนกลับอยู่ภายในที่จะเหมาะสมกับคุณสมบัติต่ออุณหภูมิของความต้านแบ่งกระแสภายใน ดังนั้นต้องต่อความต้านทาน 50 ohm เพื่อตึงให้ถูกจุด(tweaking) แรงดันอ้างอิง -10.00V จะทำให้แรงดันเอ้าพุต มีค่าเท่ากับ (ค่าของดิจิตอล อินพุต / 4096) x (+10.00V)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตัวแปลงสัญญาณดิจิตอลเป็นอนาลอก จะมีกราวน์ทั้งทางอนาลอก กราวน์ และดิจิตอลกราวน์ เพื่อหลีกเลี่ยงการรับสัญญาณรบกวนมาที่ส่วนของดิจิตอล จากทางอนาลอก กราวน์ทั้งสองจะต่อเข้าด้วยกันที่แหล่งจ่ายกำลังเท่านั้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 5

การแปลงอนาลอกเป็นดิจิตอล

(ANALOG TO DIGITAL CONVERSION)

5.1 แนะนำ

ในการที่จะให้ไมโครคอมพิวเตอร์รับรู้สัญญาณอนาลอกได้ จะต้องมีการแทนสัญญาณอนาลอกระดับ(ค่า)หนึ่งๆ ด้วยค่าของตัวเลขไบนารี เราเรียกว่าการแปลงอนาลอกเป็นดิจิตอล(A/D Conversion) ในตัวแปลง A/D บางชนิดจะมีการใช้ตัวแปลงดิจิตอลเป็นอนาลอก(D/A Conversion)อยู่ภายในตัวแปลง A/D ด้วย วิธีการที่ใช้ในการแปลง A/D มีหลายวิธี ที่มีในปัจจุบันอาจแยกได้ 4 วิธีดังนี้

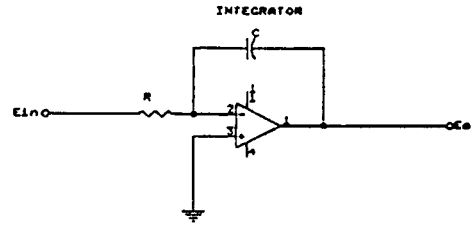
1. แบบอินทิเกรท (Integrating)
2. การแปลงอนาลอกเป็นดิจิตอลแบบขนาน (Parallel ADC)
3. แบบไบนารีแรมพ์ (Binary ramp)
4. แบบประมาณค่าอย่างต่อเนื่อง (Successive Approximation)

5.2 การแปลง A/D แบบอินทิเกรท(Integration ADC Circuit)

ตัวแปลง A/D แบบนี้จะใช้ออปแอมพ์เป็นตัวอินทิเกรท ซึ่งจะแบ่งได้อีก 3 ชนิด คือ สโลปเดี่ยว(Single-Slop),สโลปคู่(Dual-Slop) และแบบหลายๆ สโลป(Multiple-Slop)

รูป 5.1 เป็นวงจรอินทิเกรเตอร์ที่ใช้ออปแอมพ์จะมี R,C และออปแอมพ์ต่อกันดังรูป R จะต่ออนุกรมกับอินพุตและ C จะต่ออยู่ระหว่างอินพุตกับเอาต์พุตทำหน้าที่เป็นตัวป้อนกลับสัญญาณจากเอาต์พุตมาที่อินพุต

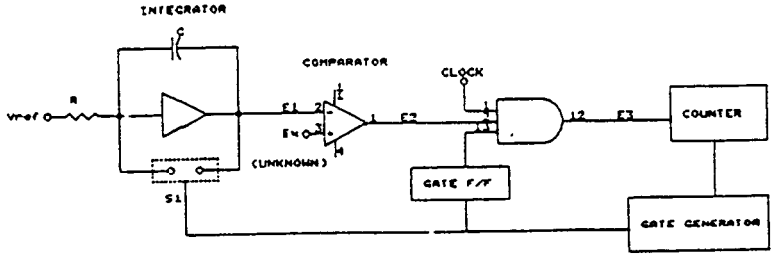
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูป 5.1 วงจรอินทิเกรทเตอร์

5.2.1 การแปลง A/D แบบสโลปเดี่ยว (Single-Slop Integrater ADC)

ดังรูป 5.2 วงจรจะประกอบด้วยออปแอมพ์ 2 ตัว ตัวแรกทำหน้าที่เป็นวงจรอินทิเกรท ตัวที่ 2 ทำหน้าที่เป็นตัวเปรียบเทียบ จะมีวงจรสร้างสัญญาณเปิด (Gate-generater) ซึ่งจะทำหน้าที่สั่งให้ เกต ฟลิปฟล้อป ,S1 และตัวนับ (Counter)ทำงาน

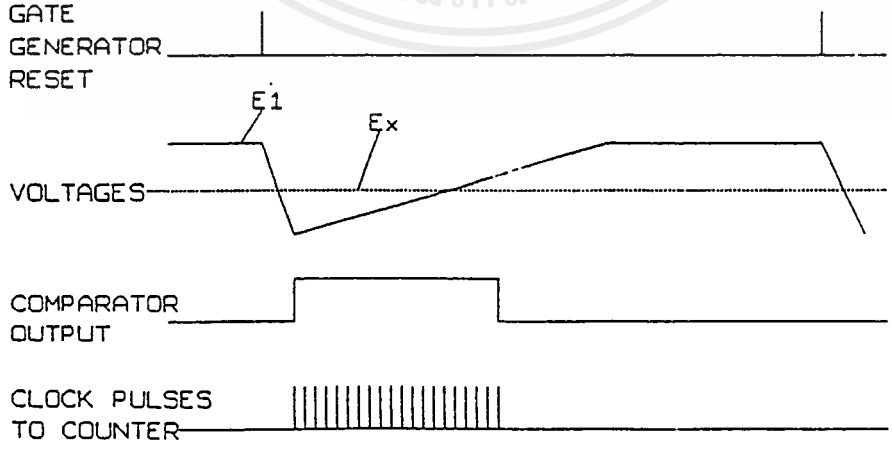


รูป 5.2 วงจรแปลง A/D แบบอินทิเกรทชนิดสโลปเดี่ยว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โดยปกติ S1 จะอยู่ในตำแหน่งเปิดวงจร ซึ่งจะสั่งจากตัวกำเนิดเกต (Gate-generater) จะเห็นได้ว่าเมื่อมีแรงดัน V_{ref} เข้ามา S1 จะเปิดวงจร แรงดัน E1 จะเป็นบวกเมื่อเทียบกับ Ex เนื่องจากตัวเปรียบเทียบเป็นแบบกลับขั้ว E2 จึงมีค่าเท่ากับ "0" ส่วนสัญญาณนาฬิกานั้นจะพร้อมอยู่เสมอที่จะผ่านออกไปที่ตัวนับ เมื่อ E2 มีค่าเป็น "1" และ เกตฟลิ๊ปฟล้อบ มีค่าเท่ากับ "1" E3 จะเท่ากับ "0" ไม่มีการนับเกิดขึ้น การทำงานจะเริ่มเมื่อ S1 ปิดวงจร (สั่งโดย gate-generater) ทำให้ C ลัดวงจร(เริ่มประจุ) หลังจากนั้น S1 จะเปิดวงจรทันที ตอนเริ่มต้นประจุแรงดัน E1 จะมีค่าเป็น 0 V.เอาต์พุตของตัวเปรียบเทียบ E2 จะมีค่าเป็น "1" (เพราะ E1 มีค่าน้อยกว่า Ex) ในขณะที่ E2 มีค่าเท่ากับ "1" เกตฟลิ๊ปฟล้อบจะอยู่ในสถานะเปิดเกต ทำให้สัญญาณนาฬิกาผ่านไปยังตัวนับได้ เมื่อ C ประจุจนแรงดันถึงค่า Ex ค่าเอาต์พุตของตัวเปรียบเทียบ E2 จะเปลี่ยนสถานะจาก "1" มาเป็น "0" เป็นการปิดเกตทำให้ไม่มีสัญญาณนาฬิกาส่งไปให้ตัวนับจึงเกิดการหยุดนับ ในวงรอบต่อไปจะเริ่มที่สัญญาณเริ่มต้นใหม่ คือ ทำให้ S1 ปิดวงจรอีกต่อไปเรื่อยๆ

ข้อดีของวงจรแบบสโลปเดี่ยวคือ มีราคาไม่แพง อาจจะพบเห็นใช้ในดิจิตอลมัลติมิเตอร์



รูป 5.3 ผังเวลาของการแปลง A/D แบบสโลปเดี่ยว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

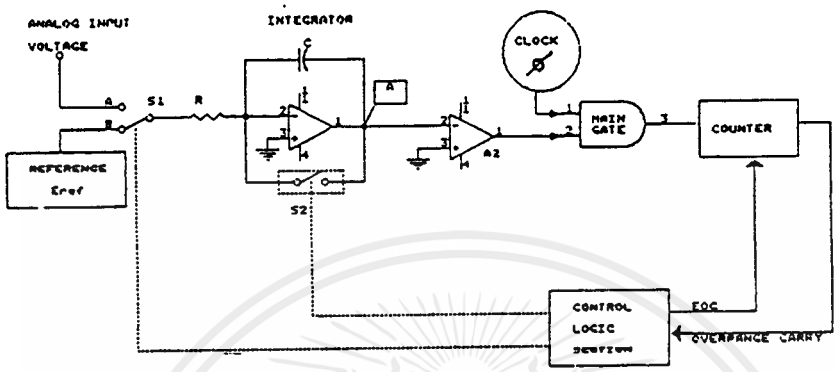
5.2.1 การแปลง A/D แบบสโลปคู่(Dual-Slop Integrater ADC)

ดังรูป 5.4 จะมีวงจรคล้ายกับแบบสโลปเดี่ยว จะแตกต่างกันอยู่ที่ วงจรแบบสโลปคู่จะมีการบอกได้ว่าเกิดการเกินพิสัย(Overange) ในกรณีที่ตัวนับไม่สามารถจะนับได้อีกโดยการส่งสัญญาณ Overange Carry เข้าไปบอกตัวควบคุมลอกจิก ตัวควบคุมลอจิกเมื่อได้รับสัญญาณ Overange Carry มันก็จะส่งสัญญาณจบการแปลง(Eoc-End Of Connersion) ออกมา

การทำงานเริ่มต้นด้วย Control-logic จะสั่งให้ S1 สวิชไปที่ตำแหน่ง A ซึ่งเป็นการต่ออินพุตของวงจรอิรทีเกรทเข้ากับสัญญาณอินพุต S2 จะถูกสั่งให้ปิดวงจรเป็นการสัตวงจร C จากนั้น S2 จะเปิดวงจรทันที ขณะที่ S2 เริ่มเปิดวงจร C จะเริ่มประจุ แรงดันที่จุด A จะเริ่มจากศูนย์ ทำให้ได้เอาต์พุตของตัวเปรียบเทียบ (คือแรงดัน A2) มีค่าเป็นลอจิก "1" เป็นการเปิดเกตให้สัญญาณนาฬิกาผ่านไปที่ตัวนับ เมื่อแรงดันเอาต์พุตของวงจรอินทิเกรท(A) มีค่าเพิ่มขึ้น ตัวนับก็จะนับต่อไปจนถึงจุดสูงสุดที่ไม่สามารถที่จะนับได้ ตัวนับก็จะส่งสัญญาณ Overange-Carry ออกไปบอก ตัวควบคุมเกต ซึ่งมันจะสั่งให้ S1 สวิชมาที่ตำแหน่ง B ซึ่งเป็นการต่อกับแรงดันอ้างอิง(E_{ref}) ตัวนับจะถูกรีเซ็ทให้เป็น "0" หมด ซึ่งช่วงเวลาจากเริ่มต้นจนถึงจุดนี้จะได้เวลาที่คงที่ค่าหนึ่ง(t0 ถึง t1 = คงที่) หลังจากที่ S1 สวิชมาที่ตำแหน่ง B จะทำให้ตัวนับเริ่มนับต่อจากศูนย์ จนแรงดันจากเอาต์พุตของวงจรอินทิเกรทผ่านศูนย์(Cross-Zero) จะทำให้เอาต์พุตของตัวเปรียบเทียบเปลี่ยนสถานะไปเป็น "0" ซึ่งเป็นการปิดเกตไม่ให้สัญญาณนาฬิกาผ่านไปที่ตัวนับ จะเห็นได้ว่าช่วงเวลาจาก t1 ถึง t2 จะขึ้นอยู่กับสัญญาณอินพุต(พูดได้ว่า หลังจากตัวนับถูกรีเซ็ท จนถึงเวลาที่เอาต์พุตของตัวเปรียบเทียบเปลี่ยนสถานะ จำนวนสัญญาณนาฬิกาที่นับได้จะขึ้นอยู่กับสัญญาณอินพุตที่ป้อนเข้ามาที่วงจรอินทิเกรท) ตามผังเวลาดังรูป 5.4

ข้อดีของแบบสโลปคู่ก็คือ มันจะให้ความละเอียดได้สูงในราคาที่พอสมควร(ไม่แพงจนเกินไป) ส่วนมากจะเป็นหัวใจสำคัญของดิจิตอลโวลท์มิเตอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูป 5.4 วงจรแปลง A/D แบบสโลปคู่

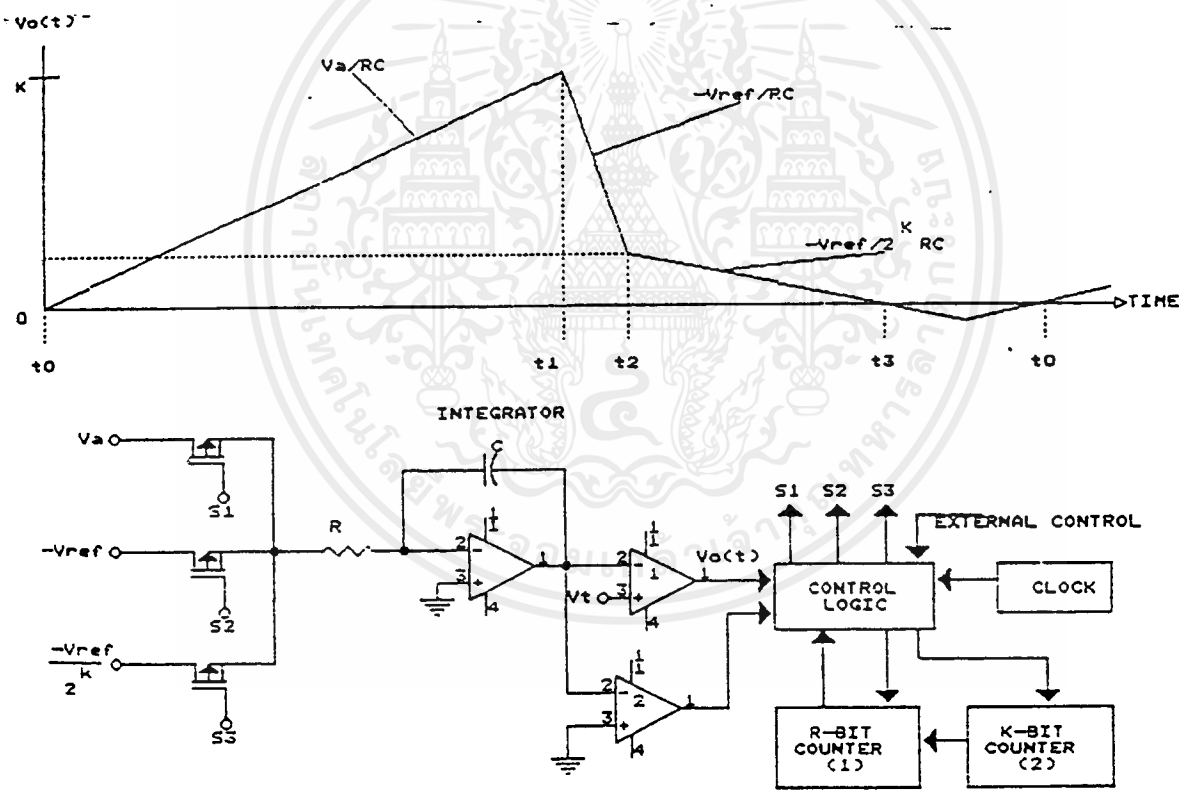
5.2.3 การแปลง A/D แบบสามสโลป(Triple-Slop Integration ADC)

ดังรูป 5.5 จากผังเวลาจะเห็นว่า S1, S2 และ S3 จะทำหน้าที่เป็นตัวเลือกอินพุตที่จะต่อเข้ากับวงจรอินทิเกรท

การทำงานจะเริ่มด้วย S1 ถูกสั่งให้ทำงาน เป็นการต่อเอาแรงดัน Va (แรงดันที่เราต้องการจะวัด) เข้ามาผ่านตัวเปรียบเทียบทำให้เกิดการนับ เมื่อตัวนับนับถึง R-Bit ตัว K-bit จะทำการเตือน R-Bit ให้ส่งสัญญาณลอจิกไปควบคุมให้ S2 ทำงาน เป็นการต่อเอาแรงดัน $-V_{ref}$ เข้ามาที่อินพุตของวงจรอินทิเกรท ช่วงนี้แรงดันเอาต์พุตของวงจรอินทิเกรทจะมีค่าเท่ากับ $-V_{ref}/RC$ จน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กระทั่งถึงค่า V_t ตัวเปรียบเทียบตัวที่ 1 จะทำให้เอาต์พุตของตัวควบคุมเกต (Control-logic) สั่งให้ S3 ทำงาน เป็นการต่อแรงดันอินพุตเข้ากับแรงดัน $-V_{ref}/2^K$ จะได้เอาต์พุตของวงจรอินทิเกรทมีค่าเท่ากับ $-V_{ref}/2^K RC$ ตัวเปรียบเทียบตัวที่ 2 จะทำงาน จนกระทั่งถึงช่วงเวลา t3 จะเห็นได้ว่ามีการทำงานทั้งหมด 3 สถาวะด้วยกัน

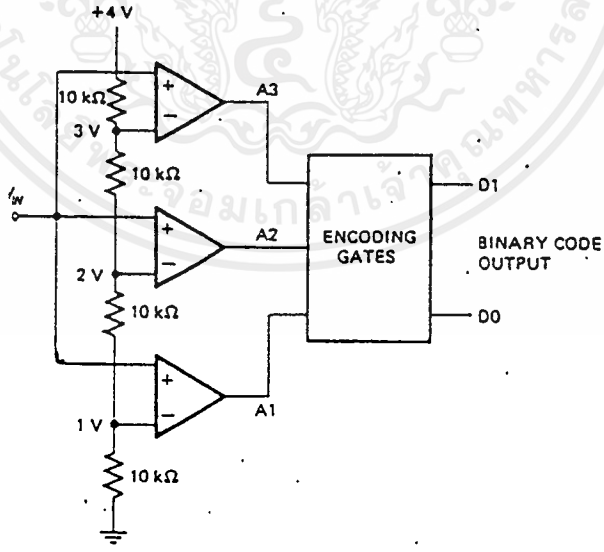


รูป 5.5 ตัวแปลง A/D แบบสามสโลป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5.3 การแปลง A/D แบบขนาน (Parallel ADC)

รูป 5.6 แสดงวงจรเบื้องต้นสำหรับตัวแปลง A/D ขนาด 2-บิตโดยการใช้ตัวเปรียบเทียบขนานกัน ความต้านทานแบ่งแรงดันใช้เพื่อการตั้งค่าอ้างอิงที่ขา - ของออปแอมพ์ แรงดันที่จุดบนสุดของการแบ่งแรงดัน จะแทนค่าเต็มพิกัดสำหรับตัวแปลง A/D นี้(3V) แรงดันอินพุต(อนาลอก) ที่จะแปลงจะจ่ายเข้ามาที่ขา + ของออปแอมพ์ ซึ่งเป็นการต่อเข้ามาแบบขนาน ถ้าหากแรงดันอินพุตมีค่ามากกว่าแรงดันอ้างอิงของตัวเปรียบเทียบ(ตัวนั้น) เอาต์พุตของตัวเปรียบเทียบ จะเปลี่ยนสถานะไปเป็นลอจิก "1" นั่นคือเอาต์พุตของตัวเปรียบเทียบ จะแทนระดับของสัญญาณอินพุตที่ป้อนเข้ามา ตัวอย่างเช่น ถ้าหากแรงดันอินพุตมีค่า 2.6 V แรงดันเอาต์พุตของตัวเปรียบ A1 และ A2 จะมีระดับลอจิกเป็น "1" ซึ่งเป็นการแทนแรงดัน 2.6V ตัวเข้าระหัสจะทำให้ได้ค่าดิจิตอลเอาต์พุตออกมาดังรูป 5.6



รูป 5.6 งวจรแปลง A/D แบบขนาน(หรือ Flash A/D)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ข้อดีของตัวแปลง A/D แบบขนานคือ ความเร็วในการแปลงจะสูงมากเราจะเรียกว่า Flash A/D เนื่องจากเอาต์พุตที่ได้จากตัวเปรียบเทียบไม่ใช่ระหัสไบนารี แต่เราสามารถเปลี่ยนให้เป็นระหัสไบนารีได้โดยใช้ตัวเข้าระหัส ข้อเสียของมันก็คือ จะต้องใช้ตัวเปรียบจำนวนมากเมื่อขนาดความละเอียดของการแปลงสูงขึ้น(จำนวนบิตของการแปลง) จากตัวอย่างรูป 5.6 เป็นการแปลงขนาด 2-บิต ต้องใช้ตัวเปรียบเทียบจำนวน 3 ตัว ถ้าหากต้องการความละเอียดขนาด N-บิต จะต้องใช้ตัวเปรียบเทียบถึง (2^n-1) ตัว เช่น ถ้าเป็นขนาด 8 บิตจะต้องใช้ตัวเปรียบเทียบ $(2^8-1) = 255$ ตัว ถ้าเป็น 10 บิตจะใช้ตัวเปรียบเทียบถึง 1023 ตัว ดังนั้นราคาของ Flash A/D จึงมีราคาแพงมาก ตัวอย่างที่มีขายในปัจจุบัน ขนาด 8 บิต จะมีความเร็วในการแปลง 20 nS

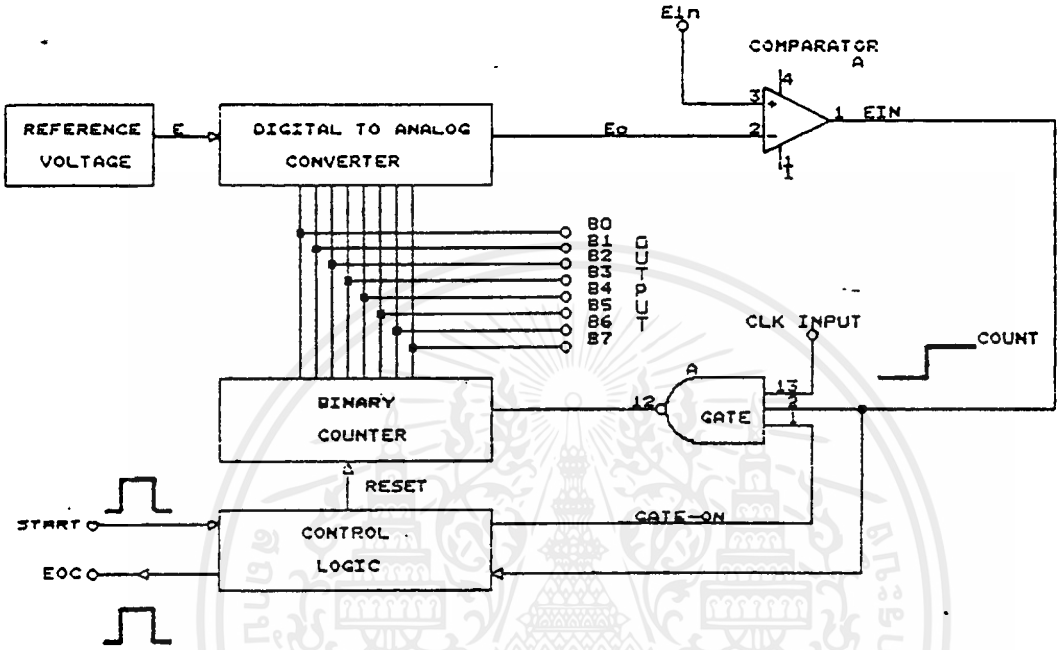
5.4 การแปลง A/D แบบไบนารี่แรมพ์(Binary Ramp ADC)

ไบนารี่แรมพ์ หรือ เรียกอีกอย่างหนึ่งว่าเซอร์โว(Servo) จะเป็นวงจรที่มีการป้อนกลับ(feed-back) เข้ามาเกี่ยวข้องด้วย ในส่วนหนึ่งของตัวแปลงแบบนี้จะมีตัวแปลง D/A ประกอบอยู่ด้วย โดยเอาต์พุตที่ได้จากตัวแปลง D/A จะป้อนกลับเป็นลูป(loop) ทำให้เกิดการเปรียบเทียบกับอินพุตดังจะได้กล่าวต่อไป

จากวงจรรูป 5.7 จะประกอบด้วยตัวเปรียบเทียบ, เกตหลัก, ตัวนับเลขไบนารี(Binary-Couter),แหล่งจ่ายอ้างอิง(Reference-Source) และส่วนควบคุมลอจิก(Control-logic-section)

ตัวแปลง D/A ที่ใช้ในตัวแปลง A/D ตามรูป 5.7 จะเป็นแบบใช้ความต้านทานขั้นบันได(R-2R Ladder) เอาต์พุตจากตัวแปลง D/A จะถูกป้อนกลับไปที่ตัวเปรียบเทียบ เพื่อทำการเปรียบเทียบกับอินพุต(ที่เป็นอนาลอก) ส่วนตัวเปรียบเทียบอื่นๆ จะต่ออยู่กับแหล่งจ่ายกระแสที่มีขนาดการแสจำนวนน้อยมาก เพื่อทำการกำจัดความผิดพลาดในวงจร

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูป 5.7 วงจรแปลง A/D แบบไบนารี่แรมพ์

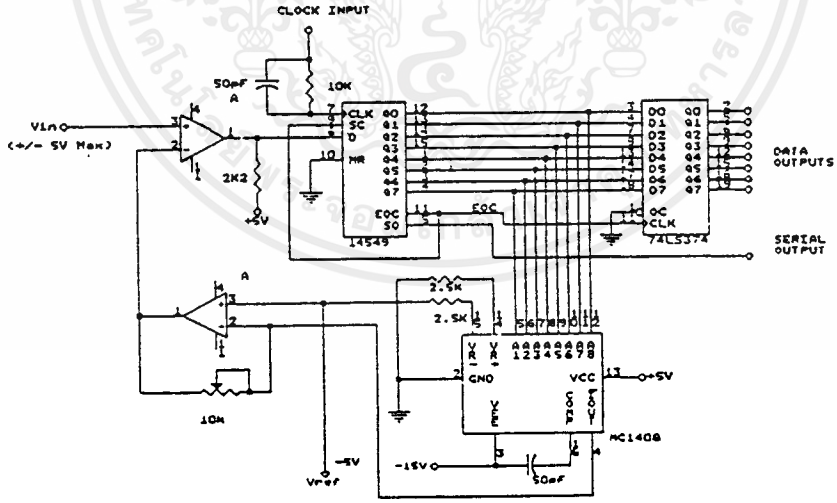
จากรูป 5.7 การทำงานจะเริ่มต้นด้วยตัวควบคุมลอจิกถูกสั่งให้ start (ควบคุมจากผู้ใช้ หรือ จากภายนอก) จะทำให้ตัวนับไบนารีถูกรีเซ็ต และ เกตหลัก จะเปิด เมื่อเราป้อนแรงดันที่จะวัด(Ein) เข้ามากที่ตัวเปรียบเทียบ จะทำให้เอาต์พุตของตัวเปรียบเทียบมีแรงดัน E_{IN} ซึ่งมีสถานะเป็น "1"เป็นการเปิดเกตให้สัญญาณนาฬิกาผ่านเข้าไปที่ตัวนับได้ เอาต์พุตของตัวนับไบนารีจะต่อกับอินพุตของตัวแปลง D/A ดังนั้นที่เอาต์พุตของตัวแปลง D/A จะมีแรงดันอนาลอกออกมาคือ Eo เมื่อ Ein(แรงดันที่เราต้องการวัด) สูงขึ้นจนมีค่าเท่ากับ Eo จะทำให้เอาต์พุตของตัวเปรียบเทียบ เปลี่ยนสถานะไปเป็นลอจิก "0" ทำให้ตัวนับไบนารีหยุดการ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

นับ หลังจากนั้นตัวควบคุมลอจิกก็จะส่งสัญญาณสิ้นสุดการแปลง(EOC)ออกมา เป็นการจบการแปลง 1 ครั้ง

5.5 การแปลง A/D แบบประมาณค่าอย่างต่อเนื่อง (Successive Approximation ADC)

ตัวแปลง A/D แบบประมาณค่าอย่างต่อเนื่อง(SA-ADC) ดังรูป 5.8 ซึ่งแสดงให้เห็นถึง วงจรซึ่งสามารถนำไปใช้งานจริงได้ จากวงจรรูป 5.8 จะมีความละเอียดในการแปลงขนาด 8 บิต หัวใจสำคัญของวงจรนี้คือ รีจิสเตอร์ประมาณค่าอย่างต่อเนื่อง(SAR-Successive Aproximation Register)หมายเลข MC14549 มีหน้าที่การทำงานดังจะได้กล่าวถึงต่อไป



รูป 5.8 วงจรแปลง A/D แบบประมาณอย่างต่อเนื่อง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เริ่มต้นด้วยการสั่งให้วงจรเริ่มทำขบวนการแปลง โดยป้อนสัญญาณนาฬิกาลูกแรกเข้าไปที่สัญญาณนาฬิกาอินพุต(ขา 7 ของ MC14549) ขาเอาต์พุตบิตนัยสำคัญสูงสุด(Q0) จะมีลอจิกเป็น "1" ซึ่งจะต่อกับอินพุตบิตนัยสำคัญสูงสุดของตัวแปลง D/A หมายเลข MC1408(ขา 5) MC1408 จะแปลงสัญญาณนี้ออกมาเป็นแรงดัน (ที่ออปแอมพ์ A2) ซึ่งแรงดันเอาต์พุตที่ได้จะป้อนให้กับตัวเปรียบเทียบ ถ้าหากแรงดันนี้ มีค่าสูงกว่าแรงดันอนาลอกที่เราจะวัด(Vin)เอาต์พุตของตัวเปรียบเทียบจะมีลอจิกเป็น "0" เป็นการบอก SAR ให้รู้ว่าแรงดันที่จะวัดมีค่าสูงกว่าแรงดันอ้างอิง (ที่สร้างมาเปรียบเทียบ) SAR จะทำการ "OFF" Q0(ทำให้ Q0 = "0") แต่ถ้าหากแรงดันที่ได้จากตัวแปลง D/A มีค่าต่ำกว่าแรงดัน Vinเอาต์พุตของตัวเปรียบเทียบจะมีลอจิกเป็น "1" เป็นการบอกให้ SAR ยังคงให้ขา Q0 มีลอจิก "1" (ON)อยู่ เมื่อมีสัญญาณนาฬิกาลูกต่อไปเข้ามา SAR จะทำให้บิทถัดไป "ON" ซึ่งจะทำให้ได้แรงดันที่อินพุตตัวเปรียบเทียบสูงขึ้น ถ้าผลของการเปรียบเทียบกับแรงดัน Vin ยังคงน้อยกว่าแรงดันเปรียบ SAR จะสั่งให้บิตถัดไป "ON" แล้วทำการเปรียบเทียบต่อไป จะเห็นได้ว่าเมื่อครบ 8 ลูกของสัญญาณนาฬิกา เอาต์พุตของ SAR ก็จะได้แรงดันที่มีค่าใกล้เคียงกับแรงดันอินพุต(Vin)ที่สุด SAR ก็จะส่งสัญญาณสิ้นสุดการแปลง(EOC) ออกมาที่ขา 5 ของ MC14549 จากวงจรรูป 5.8 สัญญาณ EOC จะใช้เป็นสัญญาณ Strobe ผลของการแปลงจากขา Q0 ถึง Q7 ของ MC14549 มาที่แลทช์หมายเลข 74LS374 ซึ่งที่เอาต์พุตของ74LS374 สามารถที่จะต่อเข้ากับพอร์ทอินพุต ของไมโครคอมพิวเตอร์เพื่อทำการอ่านเอาข้อมูลไปทำขบวนการต่อไป

รูป 5.8 จะมีการต่อสัญญาณ EOC เข้ากับ ขา SC(Start Conversion) ซึ่งจะเป็นการทำการแปลงอยู่ตลอดเวลา

มีตัวแปลง A/D สำเร็จรูปอื่นๆ อีกมากมายเช่น ADC0809 ของ National Semiconducter ภายในจะมีสวิชแบบมัลติเพลกซ์ สามารถต่อกับอินพุตได้ 8 ช่อง มีขนาดความละเอียด 8 บิต และ Setting time ประมาณ 100 ไมโครวินาที

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5.6 ระหัสจากการแปลงอนาลอกเป็นดิจิตอล(A/D Output Code)

ก่อนที่จะอธิบายการเชื่อมโยงการแปลงสัญญาณอนาลอกเป็นดิจิตอลต้องทำความเข้าใจเกี่ยวกับเอาต์พุตจากการแปลงอนาลอกเป็นดิจิตอลก่อน

เพื่อสะดวกในการใช้งานต่าง ๆ ของตัวแปลงสัญญาณอนาลอกเป็นดิจิตอลที่มีอยู่หลาย ๆ วิธี บางอย่างก็ยุ่งยาก วิธีที่ดีที่สุดเพื่อทำความเข้าใจกับระหัสต่าง ๆ เพื่อให้มองเห็นทั้งหมดเข้าด้วยกัน กับการแทนค่าดังในรูป 5.9 ค่าที่แสดงนี้ใช้กับตัวแปลงขนาด 8 บิต แต่สามารถขยายไปสู่จำนวนบิตใด ๆ ได้

สำหรับตัวแปลงสัญญาณอนาลอกเป็นดิจิตอล แบบสัญญาณอินพุตเป็นบวกอย่างเดียวหรือเรียกว่าแบบขั้วเดียว(unipolar) ส่วนมากจะเป็นระหัสฐานสองโดยตรงหรือสมนัย(Inverted)ของเลขฐานสอง ถ้าเอาต์พุตของตัวแปลงสัญญาณอนาลอกเป็นดิจิตอล ใช้ในการขับภาคแสดงผล จะสะดวกขึ้นที่จะมีระหัสเอาต์พุตเป็นระบบ BCD สำหรับการใช้งานที่มีอินพุต(อนาลอก)ทั้งบวกและลบ หรือที่เรียกว่าแบบสองขั้ว(Bipolar) ปกติใช้การเข้าระหัสแบบ Offset-binary Coding ดังที่เห็นในรูป 5.9 ค่าของ 0000 0000 ถึง 1111 1111 เป็นการเลื่อนค่าลงเพื่อที่จะแทน 0000 0000 ที่เป็นค่ามากที่สุดทางด้านลบและ 100000000 แทนค่าอินพุตที่มีค่าเป็นศูนย์ ตารางเข้าระหัสนี้ มีข้อดีที่การแทนค่าคอมพลีเมนต์ที่ 2 (2's complement) สามารถสร้างโดยการอินเวิร์ตบิตที่มีนัยสำคัญมากสุด

5.7 การเชื่อมโยงตัวแปลง A/D ชนิดต่างๆ เข้ากับไมโครคอมพิวเตอร์ (Interfacing Different Type of A/D Converters to Microcomputers

5.7.1 การเชื่อมโยงกับตัวแปลง A/D แบบตัวเปรียบเทียบขนาน (Interfacing To Parallel-Comparator A/D

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ในการใช้งานใด ๆ ที่ใช้ตัวแปลง A/D แบบตัวเปรียบเทียบชนาน

UNIPOLAR BINARY CODES

VALUE	10 VOLTS FULL SCALE	BINARY (BIN)	COMPLEMENTARY BINARY (CB)	INVERTED BINARY (IB)	INVERTED COMPLEMENTARY BINARY (ICB)
+FS −1 LSB	9.9609	1111 1111	0000 0000		
+½ FS	5.0000	1000 0000	0111 1111		
+¼FS −1 LSB	4.9609	0111 1111	1000 0000		
+1 LSB	0.0391	0000 0001	1111 1110		
ZERO	0.0000	0000 0000	1111 1111	0000 0000	1111 1111
−1 LSB	−0.0391			0000 0001	1111 1110
−½ FS + 1 LSB	−4.9609			0111 1111	1000 0000
−½ FS	−5.0000			1000 0000	0111 1111
−FS + 1 LSB	−9.9609			1111 1111	0000 0000

UNIPOLAR BINARY CODED DECIMAL CODES

VALUE	10 VOLTS FULL SCALE	BINARY CODED DECIMAL (BCD)	COMPLEMENTARY BINARY CODED DECIMAL (CBCD)	INVERTED BINARY CODED DECIMAL (IBCD)	INVERTED COMPLEMENTARY BINARY CODED DECIMAL (ICBCD)
+FS −1 LSB	9.9	1001 1001	0110 0110		
+½ FS	5.0	0101 0000	1010 1111		
+1 LSB	0.1	0000 0001	1111 1110		
ZERO	0.0	0000 0000	1111 1111	0000 0000	1111 1111
−1 LSB	−0.1			0000 0001	1111 1110
−½ FS	−5.0			0101 0000	1010 1111
−FS +1 LSB	−9.9			1001 1001	0110 0110

BIPOLAR BINARY CODES

VALUE	10 VOLTS FULL SCALE RANGE	OFFSET BINARY (OB)	COMPLEMENTARY OFFSET BINARY (COB)	TWO'S COMPLEMENT (TC)
+FS	5.0000			
+FS −1 LSB	4.9609	1111 1111	0000 0000	0111 1111
+1 LSB	0.0391	1000 0001	0111 1110	0000 0001
ZERO	0.0000	1000 0000	0111 1111	0000 0000
−1 LSB	−0.0391	0111 1111	1000 0000	1111 1111
−FS +1 LSB	−4.9609	0000 0001	1111 1110	1000 0001
−FS	−5.0000	0000 0000	1111 1111	1000 0000

รูป 5.9 ระหัสเอาต์พุตแบบต่างๆของตัวแปลง A/D

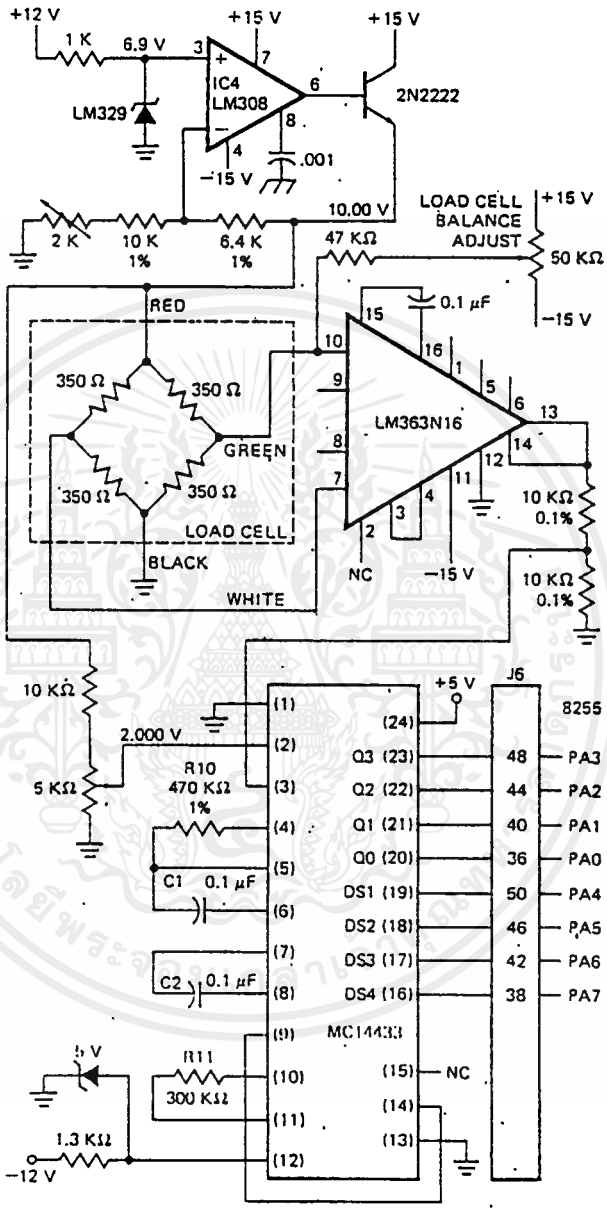
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ส่วนมากแล้วตัวแปลง A/D จะมีความเร็วมากกว่าตัวไมโครคอมพิวเตอร์ที่จะสามารถอ่านค่าได้ ดังนั้นการแยกวงจรที่ใช้โดยการส่งผ่านค่าดิจิตอลที่ได้จากการแปลงให้หน่วยความจำ(โดยตรง)ก่อน แล้วจึงให้ไมโครคอมพิวเตอร์มาจัดการกับข้อมูลจากหน่วยความจำภายหลัง การส่งผ่านลักษณะดังกล่าวเรียกว่า การเข้าถึงหน่วยความจำโดยตรง(DMA-Dirrect Memory Aceess) หลักการเบื้องต้นของ DMA คือการใช้ IC ควบคุมภายนอกบอกให้ไมโครโปรเซสเซอร์มีสถานะลอย(Float) กับบัสของมัน เพื่อให้ตัวควบคุม DMA เข้าไปจัดการกับบัสของระบบไมโครโพรเซสเซอร์แทน ตัวควบคุม DMA จะควบคุมบัสและทำการส่งถ่ายข้อมูลมาที่หน่วยความจำโดยตรง

5.7.2 การเชื่อมโยงกับตัวแปลง A/D แบบสโลป (Interfacing To Slope-Type A/D Converters)

ความสามารถที่มีอยู่ของตัวแปลง A/D แบบสโลป ในปัจจุบันจะถูกออกแบบเพื่อขับภาคแสดงผลแบบ 7 ส่วน(Seven-segment Display) ตัวอย่างเช่นในดิจิตอลโวลต์มิเตอร์ โดยปกติข้อมูลเอาต์พุตจะเป็นมิลติเพลกซ์ BCD หรือรูปแบบของตัวแสดงผล 7 ส่วน รูป 5.10 แสดงให้เห็นว่าสามารถต่อเอาต์พุตมัลติเพลกซ์ BCD ของตัวแปลง A/D ขนาด 3 1/2 หลักแบบสโลปที่ราคาไม่แพงหมายเลข MC14433 เข้ากับไมโครโพรเซสเซอร์พอร์ตได้อย่างไร รูป 5.10 เป็นการใช้ไมโครคอมพิวเตอร์ในการชั่งน้ำหนัก ข้อมูล BCD เป็นเอาต์พุตจากตัวแปลงสัญญาณ คือ Q0 ถึง Q3 และ DS1 ถึง DS4 จะเป็นสัญญาณสโตรบที่ลอจิกสูง (DS1 เป็นสโตรบของ Q0) การทำงานของ MC1443 คือ ครั้งแรกมันจะส่งเอาต์พุต BCD ของหลักนัยสำคัญสูงสุดออกมา(Q0) แล้วทำให้ DS1 มีลอจิกเป็น "1" หลังจากช่วงเวลาค่าหนึ่งจึงจะส่งเอาต์พุตของหลักนัยสำคัญรองลงมาจนครบ 4 หลัก ก็เป็นการจบใน 1 วงรอบของการแปลง มันก็จะเริ่มวงรอบการแปลงใหม่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูป 5.10 ตัวอย่างการใช้งานตัวแปลง A/D แบบสโลป

ที่มีเอาต์พุตเป็นระหัส BCD

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การอ่านค่าข้อมูลจากตัวแปลงสัญญาณนี้ใช้หลักการง่าย ๆ โดยการอ่านที่ละบิตที่สอดคล้องกับสัญญาณสโตรบจนกระทั่งพบว่าการมีลอจิกเป็น "1" แล้วอ่านค่าน้ำหนักนั้น และเก็บข้อมูลไว้ที่ตำแหน่งความจำสำรองสำหรับการอ้างอิงใช้งานต่อไป หลังจากได้อ่านระหัส BCD สำหรับหลักหนึ่งเสร็จ ก็ทำการหลักต่อไปจนครบ 4 หลัก เนื่องจากวงจรตามรูป 5.10 เป็นการแปลงอย่างต่อเนื่อง ดังนั้นจึงสามารถที่จะเรียกใช้โปรแกรมย่อยให้มาเอาข้อมูลไปได้ตลอดเวลา

เครื่องวัดความถี่,มิเตอร์วัดแรงดันและเครื่องวัดทดสอบอื่น ๆ ส่วนมากมี Multiplexed BCD เอาต์พุตอยู่ด้วยที่แผงด้านหลัง โดยการต่อและขั้นตอนที่อธิบายมา เราสามารถใช้เครื่องวัดเหล่านี้ เพื่อป้อนข้อมูลให้กับไมโครคอมพิวเตอร์ได้

5.7.3 การเชื่อมโยงตัวแปลง A/D แบบประมาณค่าต่อเนื่อง (Interfacing a Successive-Approximation A/D Converters)

ตัวแปลงสัญญาณแบบนี้โดยปกติมีเอาต์พุต สำหรับแต่ละบิต ระหัสเอ้าพุต ปกติเป็นไบนารี่โดยตรง(Straight Binary) หรือออฟเซ็ตไบนารี่ เราสามารถต่อเอาต์พุตแบบขนานของตัวแปลงสัญญาณ A/D เข้ากับอินพุตพอร์ทที่ต้องการและอ่านเอาต์พุตของตัวแปลงสัญญาณภายใต้โปรแกรมควบคุม ในการใช้ตัวแปลงสัญญาณอนาลอกเป็นดิจิตอลแบบ Successive-approximation มี 2 สัญญาณที่ต้องมีการเชื่อมโยงกับไมโครคอมพิวเตอร์ สำหรับการโอนถ่ายข้อมูล อันแรกคือสัญญาณสั่งให้เริ่มทำการแปลง(SC-Start Convert)ซึ่งเราจะต้องส่งออกมาที่เอ้าพุตพอร์ท(จากไมโครคอมพิวเตอร์) เพื่อบอกตัวแปลง A/D ว่าให้เริ่มกระทำการแปลงสัญญาณ สัญญาณที่สองคือ สัญญาณบอกสถานะของการแปลง จะเป็นเอาต์พุตจากตัวแปลงสัญญาณอนาลอกเป็นดิจิตอล เพื่อแสดงว่าการแปลงสัญญาณเสร็จเรียบร้อยและค่าระหัสเอาต์พุตสามารถที่จะนำไปใช้งานได้แล้ว ในบทต่อไปจะกล่าวถึงลำดับขั้นต่างๆ ของโปรแกรมที่เราจะใช้ในการที่จะทำให้ได้ข้อมูลมาใช้งาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ถ้าเราทำงานร่วมกับคอมพิวเตอร์ส่วนบุคคล เช่น IBM PC จะมีคาร์ดสำหรับตัวแปลงสัญญาณ A/D และ D/A ที่สามารถเสียบต่อโดยตรงกับตัวต่อบัสของเครื่องเหล่านี้ได้จำหน่ายในท้องตลาด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 6

ระบบทางกลของบัสและคุณสมบัติทางกำลังของแหล่งจ่าย

(SYSTEM-BUS MECHANICAL AND POWER CHARACTERISTICS)

6.1 แนะนำ

การออกแบบการเชื่อมโยง(Interface) จะต้องทำบนคาร์ดที่เสียบได้พอดีกับสล้อตเสียบคาร์ด ในบทนี้จะให้ข้อมูลเกี่ยวกับรูปร่างและขนาดของ สล้อต, ขนาดของคาร์ดที่จะพอดีในสล้อตเสียบคาร์ด และความสามารถในการจ่ายกำลังของระบบจะถูกกล่าวถึงในบทนี้ด้วย จะให้คำแนะนำเกี่ยวกับข้อมูลที่จำเป็น ของระดับต่างๆ ของแรงดันและกระแส และความคลาดเคลื่อนที่ยอมให้เกิดขึ้นได้

6.2 สล้อตเสียบคาร์ดของระบบบัส (System-Bus Card Slots)

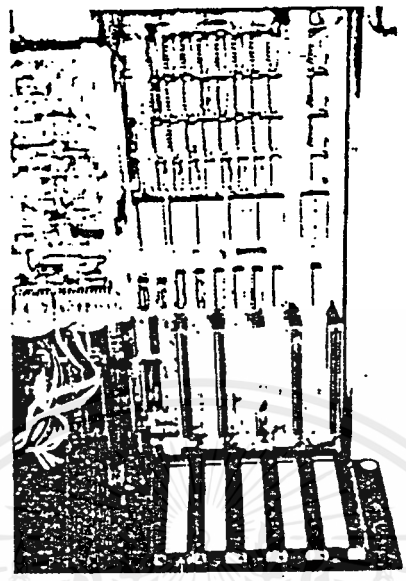
รูป 6.1 แสดงให้เห็นตำแหน่งที่ติดตั้งสล้อต ที่ใช้เสียบคาร์ดของระบบ สล้อตจะมีขนาดของขั้วต่อจำนวน 62 ขา(ข้างละ 31 ขา) ระยะห่างขาต่อขามีขนาด 100 มิล(หรือ 1/10 นิ้ว) พื้นที่สูงสุดของคาร์ดที่จะนำมาเสียบลงไปที่สล้อต 2มีขนาดประมาณ 4.2x13.0 นิ้ว สล้อตที่สองจะติดตั้งคาร์ดที่มีขนาดใหญ่เป็นพิเศษได้ ทั้งนี้เพราะที่บอร์ดโปรเซสเซอร์ไม่มีการวางอุปกรณ์ในแนวของสล้อตที่สอง ระ2ยะห่างของแต่ละคาร์ดเท่ากับ 1 นิ้ว การต่อสายเคเบิลทำได้ที่ด้านหลังของคาร์ด รูป 6.2 แสดงขาสัญญาณต่างๆ รูป 6.3 แสดงตำแหน่งของคาร์ด

6.3 ขนาดของคาร์ด (PC Card Size)

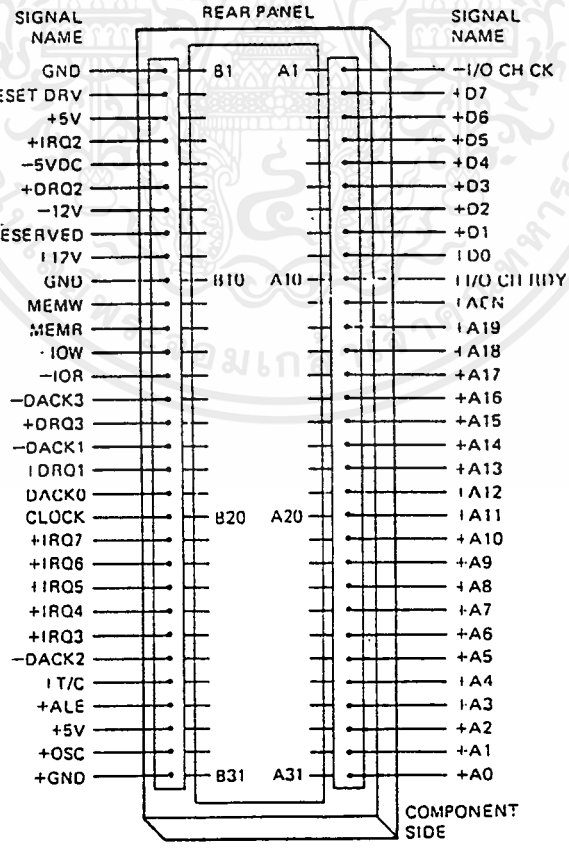
รูป 6.4 เป็นรูปวาดขนาดคาร์ดที่ใหญ่สุดที่จะใช้ได้กับสล้อตเสียบคาร์ดของ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ระบบ คาร์ดควรจะทำจากแผ่นปริ้นที่มีขนาดความหนาอย่างน้อยที่สุด 0.06 นิ้ว เพื่อที่จะทำให้แน่ใจได้ว่าคาร์ดที่นำมาเสียบฟิตพอดีกับสล้อต

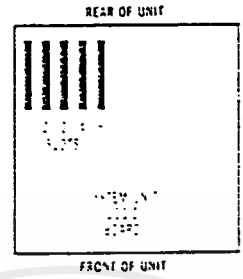


รูป 6.1 ตำแหน่งสล้อตภายในเครื่อง PC

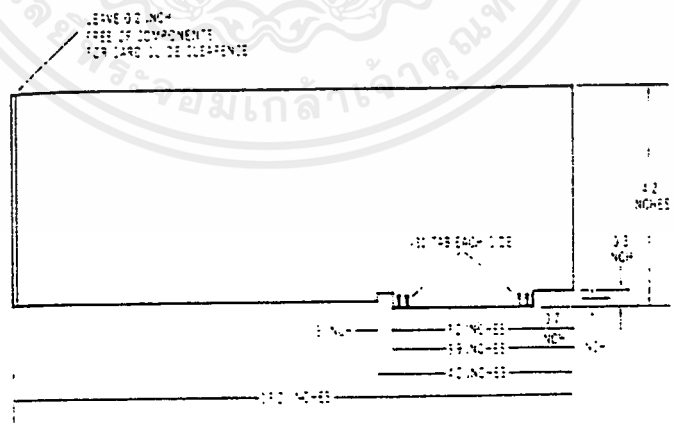


รูป 6.2 ตำแหน่งขาและสัญญาณต่างๆของสล้อต

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



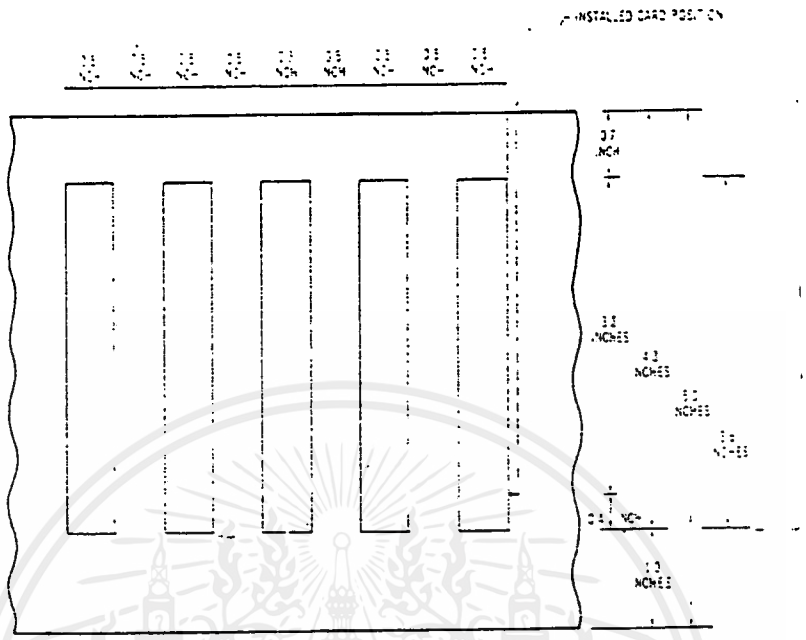
รูป 6.3 การเรียงลำดับและตำแหน่งของคาร์ดสล้อต



รูป 6.4 ขนาดใหญ่ที่สุดของคาร์ดที่จะใช้เสียบลงสล้อต

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูป 9.5 แสดงพื้นที่ว่างและขนาดสำหรับการเข้าสายเคเบิลที่อยู่ด้านหลังของ ของเครื่อง



รูป 6.5 ด้านหลังเครื่องที่จะต่อสายเคเบิลเข้าไปได้

มีผู้ผลิตหลายๆ ราย ทำคาร์ดเอนกประสงค์(Prototype card) ออกมาจำหน่าย ซึ่งเราสามารถที่จะต่อสายแบบ Wire-Wrap แล้วนำคาร์ดไปเสียบลงที่สล้อตได้เลย

6.4 ภาคจ่ายกำลังของระบบ (System Unit Power)

มีกำลังไฟตรงอยู่ 4 ระดับที่มีอยู่บนสล้อต ดังตาราง 6.1 ซึ่งจะแสดงให้เห็นว่าแต่ละคาร์ดจะใช้กำลังไฟได้สูงสุดเท่าใด และในการออกแบบควรจะแน่ใจว่ากระแสที่ใช้ในแต่ละคาร์ดจะเพียงพอที่แหล่งจ่ายกำลังจะจ่ายให้ได้หรือไม่

6.5 การชดเชยการขาดการจ่ายกำลัง (Power Decouoling)

ปัญหาที่เกิดได้ง่ายในการออกแบบวงจรอย่างหนึ่งคือ การจ่ายกำลังไม่เหมาะสมและขาดการชดเชย(De-coupling) เพราะว่าอุปาร์ณ์อิเลคทรอนิคส์ส่วนมากมี

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตาราง 6.1 ความสามารถที่จะจ่ายกำลังของภาคจ่ายกำลังไฟตรง

DC Power	Max (Vdc)	Min (Vdc)	Current (Amps)	Power (Watts)	Typical Current/Slot (Amperes)
+ 5 V DC	5.25	4.80	7.0	35.0	0.7
- 5 V DC	5.50	4.60	0.3	1.5	0.03
+12 V DC	12.6	11.52	2.0	24.0	0.10
-12 V DC	13.2	10.92	0.25	3.0	0.05

Note: The ac power outlet to the monochrome display is 120 V ac and it provides a maximum of 0.75 ampere.

ความต้องการกำลังต่างๆกัน ขึ้นอยู่กับการทำงานที่ปรากฏที่เวลาทันทีทันใด จึงต้องมีอุปกรณ์ที่นำมาชดเชย จุดประสงค์ก็คือ การจ่ายกำลังที่ต้องการมีขนาดเวลาช่วงเวลาสั้นๆ ของอุปกรณ์ ซึ่งแหล่งจ่ายกำลังจ่ายให้ทันทีทันใดไม่ได้ ทั้งนี้เพราะแหล่งจ่ายกำลังมีการเดินสายไฟมีขนาดยาวและใหญ่ ย่อมจะมีค่าความเหนี่ยวนำ และค่าของตัวเก็บประจุ ซึ่งจะทำให้การตอบสนองการจ่ายกำลังขนาดมากๆ ไม่ได้ในทันที การแก้ปัญหานี้ทำได้โดยการใช้ตัวเก็บประจุเป็นตัวจ่ายกระแสในช่วงเวลาสั้นๆ (Transient) ในตำแหน่งที่เกิดปัญหา ค่าความจุของตัวเก็บประจุที่ใช้ควรมีค่าประมาณ 8-20 uF ต่อคร่อมระหว่าง +5V และกราวด์ที่คาร์ด ตัวเก็บประจุควรจะเลือกใช้ชนิดที่ตอบสนองได้รวดเร็ว เช่น ชนิดแทนทาลัม เป็นต้น

สำหรับที่ความถี่สูงและค่ากำลังทรานเซียนท์ต่ำๆ ควรใช้ตัวเก็บประจุค่าประมาณ 0.1-0.01 uF ชนิดเซอรามิคส์ ต่อคร่อมระหว่างขาของแหล่งจ่ายไฟของอุปกรณ์กับกราวด์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่7

การใช้คำสั่งของ
PROCOMM PLUS

7.1 กล่าวทั่วไป

บทนี้จะอธิบายแต่ละคำสั่งในโหมดเทอร์มินัลของ PROCOMM PLUS เนื่องความสะดวกคำสั่งทั้งหมดจะมีอยู่ในโหมดเทอร์มินัล นอกจากนี้ยังสามารถป้อนคำสั่งของ PROCOMM PLUS ได้จากบรรทัดของเมนู รายละเอียดเกี่ยวกับป้อนคำสั่งให้ดูหัวข้อ "การใช้คำสั่งของ PROCOMM PLUS "

7.2 Auto Answer (การตอบโดยอัตโนมัติ) ALT-Y

PROCOMM PLUS สามารถตั้งโมเด็มให้ตอบการเรียกใช้โทรศัพท์เข้ามาโดยอัตโนมัติได้

การเปิดการตอบโดยอัตโนมัติ (Auto-Answer) ให้กดคีย์ ALT-Y หรือเรือกคำสั่ง MODS แล้วเลือก Answe จากบรรทัดของเมนู

หมายเหตุ เมื่อใช้คำสั่งนี้ PROCOMM PLUS จะส่งคำสั่ง auto-answer ที่กำหนดไว้ในโปรแกรมอำนวยความสะดวก setup ไปให้แก่โมเด็ม ถ้าauto-answer ม่ทำงานโมเด็มนั้นอาจจะต้องการคำสสั่งอื่น ให้ตรวจสอบดูข้อมูลของโมเด็มว่าตอ้งการคำสั่งอะไรแน่แล้วเปลี่ยนตำสั่งโดยใช้โปรแกรมอำนวยความสะดวก setup

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การปิดการ ตอบโดยอัตโนมัติ (Auto-Answer) ถ้าโทรศัพท์และโทรเด็มใช้สายโทรศัพท์เดียวกันและเปิด auto-answer ไว้โมเด็มจะตอบการเรียกโทรศัพท์ได้ทั้งจากคอมพิวเตอร์และจากอีกคนหนึ่ง ในการปิดการตอบโดยอัตโนมัติ ให้พิมพ์ว่า ATSO=0 แล้วกดคีย์ ENTER

คำสั่งนี้จะทำงานกับโมเด็มที่คอมแพตทิเบิลกับของ Hayes ถ้าโมเด็มไม่คอมแพตทิเบิลกับ Hayer ให้ตรวจสอบคู่มือของโมเด็มเพื่อหาวิธีการปิดการตอบโดยอัตโนมัติ

7.3 Break Key ALT-B

การ Break เป็นเวลาที่สัญญาณรอคอยให้ PROCOMM PLUS ส่งไปที่คอมพิวเตอร์เครื่องนั้นเกิดความสนใจ เช่น เราอาจจะส่งคำออกไป ซึ่งทราบในทันทีว่าเป็นคำสั่งที่ไม่ถูกต้อง เพื่อให้คอมพิวเตอร์เครื่องที่อยู่ไกลหยุดการประมวลผลคำสั่งนั้น ก็จะส่ง Break ไปเพื่อให้คอมพิวเตอร์เครื่องนั้นหยุดทำงานให้กดคีย์ ALT-B หรือเรียกคำสั่ง OTHER แล้วส่งสัญญาณ break โดยใช้โปรแกรมอำนวยความสะดวก setup

หมายเหตุ ถ้าคอมพิวเตอร์เครื่องที่อยู่ไกลต้องการคาเรกเตอร์เฉพาะเพื่อให้รับสัญญาณ breakได้

7.4 Change Directory ALT-F7

คำสั่งนี้ช่วยในการเปลี่ยนไดเรกทอรี และดิสก์ไดร์ที่กำลังใช้งานอยู่ การเปลี่ยนไดเรกทอรีที่กำลังใช้งานอยู่ให้กดคีย์หรือเลือกคำสั่ง CHANGE แล้วเลือกไดแรกทอรีจากบรรทัดเมนู พิมพ์ path แล้วกดคีย์ ENTER.

ตัวอย่างเช่น

-พิมพ์ A: เพื่อเปลี่ยนไดรฟ์ที่กำลังใช้อยู่เป็นไดฟร์ A

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

-พิมพ์ \word เพื่อเปลี่ยนไดเรกทอรี่ที่กำลังใช้งานอยู่เป็น \word

-พิมพ์ d:\telecomm เพื่ออเปลี่ยน PAHT ที่กำลังใช้งานอยู่ เป็น

D:\TELECOMM

คำสั่งนี้เป็นวิธีที่เหมาะ ในการเปลี่ยนไปยังไดเรกทอรี่ที่บรรจุไฟล์ที่ต้องการให้ข้อมูลไป (upload)

7.5 Chat Mode ALT-O

เมื่อกำลังสนทนาอยู่กับอีกคนหนึ่งแบบ on-line (โดยแต่ละคนพิมพ์ข้อความลงในคอมพิมพ์เตอร์ของตน) สิ่งที่พิมพ์ลงในคอมพิวเตอร์จะปรากฏในคอพิมพ์เตอร์อีกเครื่องหนึ่งในที่เดียวกับที่พิมพ์ลงไป ดังนั้นจึงอยากที่จะบอกถึงความแตกต่างระหว่างคอมพิวเตอร์ทั้งสองนั้นได้ ถ้าให้ Chat Mode ทำงานแล้ว PROCOMM PLUS จะแยกจอภาพออกเป็นสองส่วนตัวอักษรจากคอมพิวเตอร์ เครื่องที่อยู่ไกลจะปรากฏอยู่ในส่วนบน และสิ่งที่เราพิมพ์ลงไปจะปรากฏอยู่ในส่วนล่าง Chat Mode (โหมดการสนทนา) มีประโยชน์มากโดยเฉพาะถ้าใช้ CB Simulator ของ CompuServe

หมายเหตุ สามารถใช้ Meta keys ใน Chart Mode ได้ (ให้ดูหัวข้อ Meta Keys ในบทนี้)

การให้ Chat mode ทำงาน ให้กดคีย์ ALT-O หรือเลือกคำสั่ง MODES แล้วเลือกคำสั่ง Chat จากบรรทัดเมนู

ถ้าทำให้ชุลที่บันทึกไว้ในดิสก์ทำงาน (กดคีย์ ALT-F1) หรือพิมพ์ข้อมูล (กดคีย์ ALT-L) ก่อนที่จะให้โหมดการสนทนา (Chat Mode)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ทำงาน ข้อมูลเหล่านั้นจะทำงานต่อไปในโหมดการสนทนานี้ ในโหมดการสนทนานี้สามารถแสดงผลข้อมูลที่เลื่อนออกไปนอกจอภาพส่วนที่เป็นของคอมพิวเตอร์ในระยะไกล (REMOTE) หรือจอภาพส่วนของ คอทพิวน์เตอร์ของเรา (LOCAL) ได้อีก (โดยกดคีย์ ALT-F6)

การตรวจแก้ และส่งสิ่งที่พิมพ์ออกไป ไม่เหมือนกับโหมดเทอร์มินัลซึ่งส่งแต่ละคาแรกเตอร์ไปยังคอมพิวเตอร์เครื่องที่อยู่ไกลในทันทีที่พิมพ์ลงไปสิ่งที่พิมพ์ในโหมดการสนทนาจะยังไม่ส่งออกไปจนกว่าจะพิมพ์ไปจนจบบรรทัดโดยกดคีย์ ENTER อย่างนี้เรียกว่า block mode

ให้กดคีย BACKSPACE เพื่อตรวจแก้ในบรรทัดนั้น กดคีย์ ENTER เพื่อส่งข้อความในบรรทัดนั้นไปยังคอมพิวเตอร์ในระยะไกล

หมายเหตุ โหมดการสนทนาสามารถส่งตัวอักษรต่างๆออกไปได้เช่นเดียวกับโหมดเทอร์มินัล ให้ใช้ โปรแกรมอำนวยความสะดวก setup เพื่อเปลี่ยนค่าของ CHAT MODE จาก BLOCK ไปเป็นCHRACTER

การออกโหมดการสนทนา ให้กดคีย์ ESC PROCOMM PLUS จะกลับไปยังโหมดเทอร์มินัล

7.6 Clear Screen ALT-C

คำสั่งนี้จะทำการลบจอภาพ และกำหนดสีของจอภาพใหม่โดยกดคีย์ ALT-C หรือเลือกคำสั่งOTHER แล้วเลือก Clear Screen จากบรรทัดเมนู

หมายเหตุ ใช้คำสั่งนี้เมื่อติดต่อกับระบบที่เปลี่ยนสีกับจอภาพของเรา แต่ไม่กำหนดสีของจอภาพใหม่เมื่อเลิกติดต่อ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

7.7 Clipboard ALT-=

PROCOMM PLUS จะมีไฟล์ชื่อ Clipboardเพื่อช่วยเลือกและย้ายไฟล์จากคอมพิวเตอร์ระยะไกล มายังคอมพิวเตอร์ส่วนบุคคลของเราได้โดยอัตโนมัติชื่อไฟล์จะถูกตัดจากจอภาพลงไว้ใน Clipboard ต่อไปจะสามารถส่งชื่อไฟล์เดียว กลุ่มของชื่อไฟล์ หรือทั้งรายการไปยังคอมพิวเตอร์ระยะไกลเพื่อประมวลผลได้

ปกติแล้ว Clipboard ใช้เพื่อเลือกไฟล์ต่างๆ ในรายการไฟล์ของ bulletin boardเพื่อบรรจุข้อมูลลงไป ต่อมาเมื่อ BBSขอทราบไฟล์ที่ต้องการจะให้บรรจุข้อมูลแล้ว PROCOMM PLUSสามารถป้อนแต่ละชื่อไฟล์เรียงไปตามลำดับได้

ในขณะที่ทำการตัดอยู่นั้น PROCOMM PLUS จะค้นหาชื่อไฟล์ในจอภาพที่แสดงผลอยู่ชื่ออาจมีความยาวได้ถึง 8 ตัวอักษรตามด้วยจุดและนามสกุลไฟล์ได้อีก 3 ตัวอักษร

การเรียก Clipboard กดคีย์ ALT กับ = หรือเลือก file แล้วเลือก Clipboard จากบรรทัดเมนู PROCOMM PLUS จะหยุดจอภาพของเทอร์มินัลในขณะนั้นไว้ แล้วแสดงเมนูย่อยของคำสั่งให้

การเริ่มตัดชื่อไฟล์กดคีย์ C

การเลือกชื่อไฟล์ กดคีย์ลูกศรขึ้นหรือลง เพื่อเลื่อนชื่อไฟล์ขึ้นหรือลงแล้วกดคีย์ ENTER เพื่อเลือกชื่อไฟล์ในขณะนั้น

การจบการตัดชื่อไฟล์ให้กดคีย์ ESC

ถ้าต้องการแสดงผลอีกจอภาพหนึ่งของชื่อไฟล์จากคอมพิวเตอร์ในระยะไกลให้ออกจากเมนูย่อยของ Clipboard กลับไปยังโหมดเทอร์มินัลเป็นการชั่วคราวด้วยคีย์ ESC แล้วแสดงจอภาพใหม่ของข้อมูลและกลับไป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ยัง Clipboard ด้วยคำสั่งแบบกดคีย์ ALT-= นอกจากนี้ยังสามารถจับรายการไฟล์ยาวๆ ไว้ใน Redisplay buffer (ที่พักข้อมูลชั่วคราวในการแสดงผลใหม่) แล้วตัดไปไว้ใน Clipboard จากจอแสดงผลใหม่สามารถทำการตรวจแก้ชื่อไฟล์ (จากเมนูย่อยของ Clipboard) โดยการเพิ่มหรือแก้ไขชื่อไฟล์โดยตรงได้

การตรวจแก้ชื่อไฟล์ของ Clipboard

ให้กดคีย์ E สามารถใช้คีย์ในการตรวจแก้มาตรฐานของ PROCOMM PLUS เพื่อเพิ่ม แก้ไข หรือลบชื่อไฟล์ได้เมื่อตรวจแก้รายการของ Clip board เสร็จแล้วก็สามารถส่ง(หรือ ปะ)ชื่อไฟล์ (ต่างๆ) ไปยังคอมพิวเตอร์ในระยะไกลได้

หมายเหตุ ก่อนจะป้อนเมนูของ Clipboard ลงไป ให้แน่ใจว่าคอมพิวเตอร์ระยะไกลพร้อมที่จะรับรายการไฟล์ที่กำลังจะส่งไปได้

การปะชื่อไฟล์เดียวจาก Clipboard

ให้กดคีย์ s เป็นคำสั่ง Single Paste (ปะไฟล์เดียว) ที่ส่งไฟล์บนสุดไปยังคอมพิวเตอร์ในระยะไกล (พร้อมด้วยการปัดแคร่) และลบชื่อไฟล์นั้นออกจาก Clipboard ไปแล้วเลื่อนชื่อไฟล์ทั้งหมดขึ้นมา

หมายเหตุ รายการเลือกนี้มีประโยชน์โดยเฉพาะอย่างยิ่งทำการบรรจุรายการไฟล์ของ Clip board พร้อมด้วยโพรโทคอล (หลักเกณฑ์) อย่างเช่น XMODEM ลงไปเมื่อคอมพิวเตอร์ในระยะไกลขอทราบแต่ละไฟล์มาสามารถใช้ Single Paste ทำการปะชื่อไฟล์จาก Clipboard ให้และลบชื่อไฟล์นั้นออกไปโดยอัตโนมัติได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การปะไฟล์ที่ทำเครื่องหมายไว้จาก Clipboard

ให้กดคีย์ T เป็นคำสั่ง Tag Paste (ปะไฟล์ที่ทำเครื่องหมาย) เพื่อส่งชื่อไฟล์จำนวนหนึ่งไปยังคอมพิวเตอร์ในระยะไกล แต่ละไฟล์ที่ส่งไปนั้นจะแยกจากไฟล์ต่อไปด้วยคาแรกเตอร์ของการแยกจากกัน (separator) (ปกติจะเป็นเว้นวรรคเซมิโคลอน หรือการปัดแคร่ เราเลือกว่า PROCOMM PLUS จะใช้ตัวแยกจากกัน ตัวไหนได้จากโปรแกรม Setup) ไฟล์สุดท้ายในรายการจะตามหลังด้วยการปัดแคร่ (carriage return) ทำเครื่องหมายไฟล์ ที่จะปะได้โดยใช้คีย์ต่าง ๆ คล้ายกับใน dialing directory ชื่อไฟล์ที่ทำเครื่องหมายไว้นี้จะไม่ลบออกไปหลังจากที่ปะแล้ว

หมายเหตุ วิธีนี้ทำงานได้เป็นอย่างดีกับโพรโทคอลสำหรับหลายไฟล์ (multiple file proto-cols) อย่างเช่น ZMODEM หรือ YMODEM

การปะชื่อไฟล์ทั้งหมดจาก Clipboard

ให้กดคีย์ P เป็นคำสั่ง Paste All (ปะไฟล์ทั้งหมด) เพื่อส่งไฟล์ใน Clipboard ทั้งหมดไปที่คอมพิวเตอร์ในระยะไกล เช่นเดียวกับ Tag Paste ที่แต่ละไฟล์จะแยกจากไฟล์ต่อไปด้วยเว้นวรรคเซมิโคลอน หรือการปัดแคร่ ไฟล์ในลำดับสุดท้ายจะตามหลัง ด้วย การปัดแคร่ ชื่อไฟล์จะไม่ถูกลบออกไปหลังจากที่ปะแล้ว

การออกจาก Clipboard

ให้กดคีย์ ESC

7.8 CR-CR/LF TOGGLE ALT-F8

เพื่อให้คอมพิวเตอร์ของเราสามารถแสดงผลข้อความที่ส่งมา โดยคอม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

นิวเตอร์ระยะไกลได้อย่างถูกต้องแต่ละบรรทัดข้อความ จึงควรจะตามหลังด้วยทั้งการปัดแคร่ (CR-Carriage Return) และการขึ้นบรรทัดใหม่ (LF-Line Feed) การปัดแคร่จะเลื่อนเคอร์เซอร์ไปยังต้นบรรทัดนั้น แล้วการขึ้นบรรทัดใหม่จะนำเคอร์เซอร์ไปยังบรรทัดต่อไป คอมนิวเตอร์ระยะไกลส่วนใหญ่ จะจบบรรทัดข้อความที่ส่งมายังคอมนิวเตอร์ของเรา ด้วยการปัดแคร่และการขึ้นบรรทัดใหม่ผสมกัน (CR/LF)

ถ้าคอมนิวเตอร์จบบรรทัดข้อความด้วยการปัดแคร่ อย่างเดียวเท่านั้น เคอร์เซอร์ก็จะไม่ เลื่อนลงไปที่บรรทัดต่อไปเมื่อบรรทัดนั้นเต็มแล้ว หรือเมื่อกดคีย์ ENTER แล้ว (ข้อความที่ส่งเข้ามาจะพิมพ์ทับลงไปในบรรทัดสุดท้าย) ถ้าเกิดเหตุการณ์เช่นนี้ขึ้น ให้สับสวิตช์คำสั่งนี้ไปยัง CR/LFแล้ว PROCOMM PLUSจะเพิ่มการขึ้นบรรทัดใหม่ให้กับการปัดแคร่ที่ได้รับทุกตัว

ถ้าข้อความที่ปรากฏมีเว้นวรรคเป็นสองเท่า แสดงว่าคอมนิวเตอร์ทั้งสองเครื่องเพิ่มการขึ้นบรรทัดใหม่เข้าไป ให้ใช้คำสั่งนี้เพื่อเว้นการขึ้นบรรทัดใหม่เสีย

การสลับกันระหว่าง CR กับ CR/LF ที่ท้ายบรรทัดให้กดคีย์ ALT-F3 หรือเลือกคำสั่ง CHANGE แล้วเลือกสวิตช์ CR-CR/LF จากบรรทัดเมนูสามารถเปลี่ยนค่าที่กำหนดไว้โดยปริยาย สำหรับภาพนี้ได้ โดยใช้โปรแกรมอำนวยความสะดวกSETUP

หมายเหตุ คำสั่งนี้มีผลต่อบรรทัดข้อความที่ป้อนเข้าไป ถ้าต้องการเพิ่มการขึ้นบรรทัดใหม่ให้กับการปัดแคร่ใน บรรทัดข้อความที่ส่งไปในโหมดเทอร์มินัลเมื่อกดคีย์ ENTERให้ใช้คำสั่ง KEY MAPPING(หรือกดคีย์ ALT-F8) เพื่อเปลี่ยนรหัสของคีย์ ENTER จาก ^M (การปัดแคร่)ไปเป็น ^M^J (การปัดแคร่ และการขึ้นบรรทัดใหม่)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

7.9 DIALING DIRECTORY

ชื่อและหมายเลขของคอมพิวเตอร์ ที่เรียกจะเก็บไว้ใน dialing directory การทำให้ dialing directory ทำงาน ให้กดคีย์ ALT-D หรือเลือกคำสั่ง DIAL DIR จากบรรทัดของเมนู

7.10 DOS GATEWAY

PROCOMM PLUS สามารถรันคำสั่งของดอส หรือรันโปรแกรมอื่นๆได้ โดยไม่ต้องออกจาก PROCOMM PLUS

การเรียก DOS ให้กดคีย์ ALT-F4 หรือเลือกคำสั่ง OTHERแล้วเลือก DOS GATEWAY จากบรรทัด เมนู สำหรับการให้คำสั่ง DOS GATEWAY ทำงานนั้น จะต้องตั้งหลายๆอย่างให้ถูกต้องเสียก่อน ข้อแรก

ไฟล์ COMMAND.COM จะต้องอยู่ในไดเรกทอรีรากของฮาร์ดิสก์ หรือถ้าใช้ฟลอปปี้ดิสก์ ไฟล์นี้จะต้องอยู่ในไดเรกทอรีที่กำลังใช้งานอยู่ หรืออยู่ในไดเรกทอรีที่กำหนดไว้ใน DOS path อย่างใดอย่างหนึ่ง

ข้อสุดท้าย คอมพิวเตอร์จะต้องมีหน่วยความจำเลือพอรันโปรแกรมหรือคำสั่งของดอสได้ การกลับไปยัง PROCOMM PLUS ให้นิมน์ว่า EXIT ที่เครื่องหมายพร้อมรับคำสั่งของดอส แล้วกดคีย์ ENTER

หมายเหตุ ถ้ากำลังรันโปรแกรมการติดต่อสื่อสารอื่นๆ อยู่ในขณะที่อยู่ในดอส PROCOMM PLUS อาจจะทำงานผิดพลาดเมื่อกลับมายัง PROCOMM PLUS ถ้าเกิดเหตุดังกล่าวนี้ขึ้น ให้กไหนดสายใหม่ (line settings) โดยกดคีย์ ALT-? เพื่อทำการติดต่อใหม่

คำสั่งนี้สลับสวิตช์ของโมเด็มระหว่างการทำงานแบบ full duplex กับhalf duplex ให้กดคีย์ ALT-E หรือเลือกคำสั่ง CHANGE แล้วเลือกคำสั่ง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Echo จากบรรทัดเมนูโหมดของ duplex ที่กำลังใช้อยู่จะปรากฏอยู่ในบรรทัดแสดงสถานภาพ เช่น HDX (half duplex) Half duplex บางทีก็เรียกว่า local echo

ถ้าปรากฏตัวอักษรซ้อนกันสองตัว (lliikkee tthhiiss) ให้ตั้ง duplex เป็น full แต่ถ้าเมื่อ พิมพ์อะไรลงไปแล้วไม่ปรากฏในจอภาพให้ตั้ง duplex เป็น half

หมายเหตุ สามารถเปลี่ยนโหมดของ duplex ที่กำหนดไว้ให้โดยปริยายได้ (โหมดจะมีผลเมื่อเริ่มสตาร์ต PROCOMM PLUS) โดยใช้โปรแกรมอำนวยความสะดวก Setup

7.12 EDITOR

PROCOMM PLUS มีโปรแกรมตรวจแก้ข้อความ (text editor) อย่างรวดเร็วให้ด้วย (เรียกว่า PCEDIT) ซึ่งสามารถใช้จากภายในโปรแกรม PROCOMM PLUS ได้ โปรแกรม PCEDIT นี้มีประโยชน์มากโดยเฉพาะอย่างยิ่งในการเขียนและตรวจแก้สคริปต์ และไฟล์ข้อความของ ASCIT สามารถใช้โปรแกรมตรวจแก้ข้อความของเราเองในการสร้างและตรวจแก้ไฟล์ต่าง ๆ โดยไม่ ต้องออกจาก PROCOMM PLUS ได้ ถ้าเลือกใช้โปรแกรมตรวจแก้ข้อความของเราเองแทนที่จะใช้ PCEDIT ก็จำเป็นต้องกำหนด path ให้กับโปรแกรมตรวจแก้ข้อความนั้นด้วย (คือดิสก์และไดเรกทอรีที่เก็บโปรแกรมนั้นไว้) และให้กับคำสั่งที่สตาร์ตโปรแกรมนั้นโดยใช้โปรแกรมอำนวยความสะดวก Setup การเรียกใช้โปรแกรมตรวจแก้ข้อความ ให้กดคีย์ ALT-A หรือเลือกคำสั่ง FILE แล้วเลือก Editor จากบรรทัดเมนู PROCOMM PLUS จะแสดงวินโดว์ให้เพื่อกำหนดตัวแปรต่างๆ ของโปรแกรมตรวจแก้ข้อความ (editor) ตัวแปรเหล่านี้จะเป็นอะไรก็ได้ ซึ่งถ้าสตาร์ตโปรแกรมตรวจแก้ข้อความจากเครื่องหมายพร้อมรับ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

คำสั่งของดอส ปกติแล้วจะต้องนิมน์ตามหลังคำสั่งของโปรแกรมตรวจแก้ข้อความนั้น

ตัวอย่างเช่นโปรแกรมตรวจแก้ข้อความส่วนใหญ่ (รวมทั้งPCEDIT) จะให้เรากำหนดชื่อไฟล์ที่ ต้องการใช้ทำงานโดยนิมน์ชื่อไฟล์ไว้ข้างหลังคำสั่งที่สตาร์ตโปรแกรมตรวจแก้ข้อความ ถ้าโปรแกรมตรวจแก้ข้อความที่ใช้ยอมให้ทำเช่นนั้นได้ ในขั้นต่อไปต้องกำหนดชื่อไฟล์ที่ต้องการทำงานกับโปรแกรมตรวจแก้ข้อความลงไปด้วย

ให้นิมน์ตัวแปรต่าง ๆ ของโปรแกรมตรวจแก้ข้อความลงไป (ถ้ามี) แล้วกดคีย์ ENTER PROCOMM PLUS จะสตาร์ตโปรแกรมตรวจแก้ ข้อความนั้น เมื่อออกจากโปรแกรมตรวจแก้ข้อความ ก็จะกลับไปยัง PROCOMM PLUS โดยอัตโนมัติ (สามารถออกจาก PCEDIT ได้โดยกดคีย์ ALT-X)

หมายเหตุ เพื่อแสดงรายการคำสั่งของ PCEDIT ได้โดยกดคีย์ ALT-Z

หลังจากที่สตาร์ต PCEDIT แล้วสามารถรัน PECDIT จากนอกโปรแกรม PROCOMM PLUS (คือที่เครื่องหมายพร้อมรับคำสั่งของดอส) ได้โดยนิมน์ว่า PCEDIT แล้วกดคีย์ ENTER สามารถเริ่มการตรวจแก้ไฟล์ที่ต้องการได้โดยป้อนชื่อไฟล์นั้นลงไปก่อนกดคีย์ ENTER ถ้าใช้จอสี และต้องการรัน PCEDIT ในสีขาว - ดำ (เช่น เมื่อใช้จอภาพแบบ composite monochrome พร้อมด้วยการ์ด CGA) ให้นิมน์สวิตช์ /b ลงไปก่อนกด คีย์ ENTER เพื่อให้คำสั่ง Editor ทำงาน จะต้องตั้งอะไรหลาย ๆ อย่างให้ถูกต้องเสียก่อนดังนี้

- ไฟล์ COMMAND.COM จะต้องอยู่ในไดเรกทอรีรากในฮาร์ดดิสก์
- ถ้าใช้เครื่องที่มีแต่ปลอปปี้ดิสก์ ไฟล์นี้จะต้องอยู่ในดิสก์ที่อยู่ในไดรน์ A
- ไดรน์ และไดเรกทอรีที่เก็บไฟล์เอดิเตอร์ไว้(รวมทั้งคำสั่งที่สตาร์ดไฟล์) ต้องกำหนดไว้ในโปรแกรมอำนวยความสะดวก Setup
- คอมพิวเตอร์ต้องมีหน่วยความจำเหลือพอที่จะรันไฟล์เอดิเตอร์ได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

7.13 Ealpsed Time

PROCOMM PLUS จะต้องติดตามอยู่ตลอดเวลาว่าติดต่อกับคอมพิวเตอร์เครื่องอื่นเป็นเวลานานเท่าไร

การแสดงผลเวลาที่ใช้ ให้กดคีย์ ALT-T หรือเลือกคำสั่ง OTHER แล้วเลือก Elapse

จากบรรทัดเมนู PROCOMM PLUS จะแสดงผลวันที่และ เวลาในขณะนั้นรวมทั้งเวลาที่ใช้ในการเรียก (ถ้ากำลังติดต่ออยู่กับคอมพิวเตอร์เครื่องอื่น) หรือเวลาที่ติดต่อครั้งสุดท้าย

7.14 Exit

การออกจาก PROCOMM PLUS และกลับไปยัง DOS ให้กดคีย์ ALT-X หรือเลือกคำสั่ง QUIT แล้วเลือก Exit จากบรรทัด เมนู PROCOMM PLUS จะขอคำยืนยันว่าต้องการจะออกจากโปร แกรมหรือไม่ กดคีย์ Y ถ้าต้องการออก หรือกดคีย์ N เพื่อทำงานกับ PROCOMM PLUS ต่อไป ถ้ายังติดต่อกับคอมพิวเตอร์เครื่องอื่นอยู่ PROCOMM PLUS จะถามว่าต้องการ จะวางหูหรือไม่ ถ้า PROCOMM PLUS ถามว่า ต้องการจะวางหูหรือไม่โดยที่โมเด็มมิได้ติดต่อกับคอมพิวเตอร์อีกเครื่องหนึ่ง ดังนี้แสดงว่าตั้งสวิตช์ CD ของโมเด็ม ไว้ไม่ถูก

เมื่อออกจาก PROCOMM PLUS โดยไม่ได้วางหูโมเด็มก็ยังคงติดต่ออยู่กับคอมพิวเตอร์ในระยะไกล ดังนั้นจึงอาจจะต้องถูกเก็บเงินอยู่อีกหลังจากออกมาแล้วเวลาที่ยังคงติดต่ออยู่อีกขึ้นอยู่กับเครื่องคอมพิวเตอร์ ที่ติดต่อด้วย คอมพิวเตอร์บางเครื่องจะตัดการติดต่อหลังจากที่ไม่มีการทำงานเพียงไม่กี่นาที

หมายเหตุ ก่อนที่จะออกจากโปรแกรมให้บันทึกการเปลี่ยนแปลงที่ทำไว้ในโปรแกรม PROCOMM PLUS ไว้ (เช่น การเปลี่ยนแปลงในโปรแกรมอำนวย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ความสะดวก Setup) ซึ่งต้องการให้การเปลี่ยนแปลงนั้นเป็น ผลในการสตร์ต PROCOMM PLUS ครั้งต่อไป

7.15 File Directory

คำสั่งนี้จะให้รายการไฟล์ต่าง ๆ ในไดเรกทอรีใด ๆเป็นคำสั่งที่มีประโยชน์มากเมื่อจำชื่อไฟล์ ที่ต้องการส่งไปไม่ได้ หรือเมื่อต้องการดูไฟล์ที่เพิ่งได้รับมา

การดูไดเรกทอรีของไฟล์ ให้กดคีย์ ALT-F หรือเลือกคำสั่ง FILE แล้วเลือก Directory จากบรรทัดเมนู PROCOMM PLUS จะขอทราบว่าต้องการจะให้ทำรายการไฟล์ใดบ้าง พิมพ์ชื่อไฟล์ลงไป(สามารถใช้ widcards ของดอส ได้แก่ * และ ? เพื่อกำหนดชื่อไฟล์ที่ต้องการให้ทำรายการได้

เช่น:,

ถ้าต้องการให้ทำรายการไฟล์ทั้งหมดที่มีนามสกุลไฟล์ว่า .DOC ให้พิมพ์ว่า *.DOC)

สามารถเพิ่ม path ลงไปกับชื่อไฟล์ด้วยก็ได้ เช่น ในการทำรายการไฟล์ทั้งหมดที่มีนามสกุล ไฟล์ว่า .DOC ในไดเรกทอรีชื่อ C:\FILES ให้พิมพ์ชื่อไฟล์ลงไปว่า C:\FILES*.DOC ถ้าไม่ใส่ชื่อ path ลงไปด้วย PROCOMM PLUS จะให้รายการไฟล์ต่าง ๆ ในไดเรกทอรีที่กำลังใช้งานอยู่การทำ รายการไฟล์ทั้งหมดในไดเรกทอรีอื่นที่ไม่ใช่ไดเรกทอรีที่กำลังใช้งานอยู่ ให้พิมพ์ path แล้วตามด้วย backslash เช่น ในการทำรายการไฟล์ทั้งหมด ในไดเรกทอรีของ C:\FILES ให้พิมพ์ชื่อไฟล์ว่า C:\FILES\ แล้วกดคีย์ ENTER ถ้าที่ด้านล่างของวินโดว์แสดงรายการปรากฏคำว่า MORE (ยังมีอีก) ให้ กดคีย์ใดก็ได้เพื่อดู

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ไฟล์อื่น ๆ อีก ถ้าไม่ปรากฏคำว่า MORE ที่ด้านล่างของวินโดว์แสดงรายการ ให้กดคีย์ใด ๆ เพื่อกลับไปยังโหมดเทอร์มินัล

7.16 Hang Up

การตัดการติดต่อกับคอมพิวเตอร์ในระยะไกล ให้กดคีย์ ALT-H หรือเลือกคำสั่ง OTHER แล้วเลือก Hang Up จากบรรทัดเมนู PROCOMM PLUS จะวางหู แล้วแสดงคำว่า DISCONNECTING ชั่วเวลาสั้น ๆ ในบรรทัดแสดงสถานภาพ เพื่อให้ทราบว่าได้ตัดการติดต่อแล้ว เมื่อป้อนคำสั่งนี้ลงไปแล้ว PROCOMM PLUS จะพยายามวางหูก่อนโดยตัดสาย DTR (Data terminal Ready สัญญาณแจ้งโมเด็มว่าคอมพิวเตอร์พร้อมทำงาน) ดังนี้จะวางสายได้ถ้าตั้งสวิตช์ DTR ในโมเด็มไว้ถูกต้อง และถ้า Drop DRT to hangup ในโปรแกรมอำนวยความสะดวก Setup ตั้งไว้ที่ YES

ถ้าการตัด DTR ไม่ทำให้วางหูได้ PROCOMM PLUS จะส่งคำสั่งวางหูไปให้โมเด็ม ซึ่งคำสั่งนี้กำ หนดไว้ในโปรแกรมอำนวยความสะดวก Setup ดังนี้แล้วจะเป็นการวางหูโทรศัพท์ นอกเสียจากจะแน่ใจว่าคำสั่งในการวางหูโทรศัพท์ ที่กำหนดไว้ทำงานให้กับโมเด็มได้ ถ้าจำเป็นให้ตรวจสอบจากคู่มือของโมเด็ม เพื่อจะได้ทราบว่า คำสั่งในการวางหูโทรศัพท์ต้องการอะไรบ้าง

หมายเหตุ ถ้าคำสั่ง Hang Up ไม่ทำงาน ให้ดูรายละเอียดในภาคผนวกเกี่ยวกับ ปัญหาทั่ว ๆไปให้ใช้คำสั่งนี้เพื่อให้โปรแกรมอำนวยความสะดวก Help ทำงานถ้าป้อนคำสั่งนี้ในโหมดเทอร์มินัล PROCOMM PLUS จะแสดงเมนูคำสั่ง ซึ่งเป็นรายการคำสั่งที่ต้องกดคีย์ ALT

ถ้าป้อนคำสั่งนี้ด้วยเมนูคำสั่งในจอภาพ PROCOMM PLUS จะแสดงข้อมูลเกี่ยวกับโหมดเทอร์มินัล การขอความช่วยเหลือแบบ on-line บรรทัดแสดงสถานภาพและการเลียนแบบเทอร์มินัล ถ้าป้อนคำสั่งนี้นอกเหนือจากที่กล่าวแล้ว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

PROCOMM PLUS จะแสดงข้อมูลเกี่ยวกับสิ่งที่กำลังทำงานอยู่

การขอความช่วยเหลือ ให้กดคีย์ ALT-Z หรือเลือกคำสั่ง HELP จากบรรทัดเมนูสามารถทำให้โปรแกรมอำนวยความสะดวก Help ทำงานจากบรรทัดเมนูได้ก็ต่อเมื่อ PROCOMM PLUS อยู่ในโหมด เทอร์มินัลเท่านั้น เช่น ไม่สามารถเรียกโปรแกรม Help จากบรรทัดเมนูได้ในเมื่อกำลังใช้โปรแกรมอำนวยความสะดวก Setup หรือ dialing directory อยู่เนื่องแสดงผลจอภาพต่อไปหรือจอภาพที่แล้วของข้อมูลในการให้ความช่วยเหลือให้กดคีย์ PgDn หรือ PgUp ถ้าใช้เมาส์ไปตามทิศทางที่ต้องการ

การเปลี่ยนเรื่องของการให้ความช่วยเหลือให้กดคีย์ T จากโปรแกรมอำนวยความสะดวก Help หรือกดปุ่มซ้ายของเมาส์ที่โปรแกรมอำนวยความสะดวก Help

การปิดเมนูคำสั่งหรือจอภาพให้ความช่วยเหลือให้กดคีย์ ESC หรือกดปุ่มขวาของเมาส์

7.18 Host Mode

PROCOMM PLUS มีโหมด Host ที่มีความสามารถสูงมากในการเปลี่ยนคอมพิวเตอร์ส่วนบุคคลไปเป็น BBS โดยสมบูรณ์ Host จะช่วยให้ผู้ที่ใช้อยู่ในระยะไกล (รวมทั้งตัวเราเองเมื่อไม่อยู่) เรียกคอมพิวเตอร์ขอองเราได้ คุณสมบัติของโหมด Host ได้แก่การปกปิดรหัสผ่าน การย้ายไฟล์ operator page การเรียก DOS-SHELL และ ELECTการสตาร์ตโหมด Hos ให้กดคีย์ ALT-Q หรือเลือกคำสั่ง MODES แล้วเลือก Host จากบรรทัดเมนู

7.19 Init Modem

คำสั่งนี้ส่งแถวข้อความการตั้งค่าเริ่มต้นของโมเด็มที่กำหนด ไว้ในการ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Setup ไปยังพอร์ตการติดต่อ (com port) ที่กำลังทำงานอยู่ในขณะนั้น

การตั้งค่าเริ่มต้นของโมเด็ม ให้กดคีย์ ALT-J หรือเลือกคำสั่ง OTHER แล้วเลือก Init Modem จากบรรทัดเมนู

หมายเหตุ การตั้งค่าเริ่มต้นของโมเด็ม อาจมีความจำเป็นต้องทำถ้ารันอีกโปรแกรมหนึ่งที่เรียก โดยใช้พอร์ตเดียวกันกับที่ PROCOMM PLUS ใช้อยู่

7.20 Kermit Server Commands

kermit เป็นโพรโทคอลในการย้ายไฟล์ แบบรวมกลุ่มบิต (packet -oriented) ที่ใช้ใน การติดต่อสื่อสารระหว่างไมโครคอมพิวเตอร์กับคอมพิวเตอร์เมนเฟรม และมีใช้อยู่ใน bulletin board บ่อยเหมือนกันคำสั่งต่างๆของ kermit server ปกติจะทำให้คอมพิวเตอร์เครื่องอื่น (ที่รันในโหมด kermit server) กลายเป็นเครื่องที่ทำตาม (slave) เครื่องคอมพิวเตอร์ของเรา เครื่องคอมพิวเตอร์ของเราจะส่งคำสั่งต่าง ๆ ที่จะทำการเอ็กซิคิวต์ในเครื่องที่อยู่ในระยะไกล การใช้คำสั่งเหล่านี้ สามารถประหยัดการกดคีย์ได้หลายครั้งทีเดียวควรจะ ใช้คำสั่งนี้ก็ต่อเมื่อหลังจากให้คอมพิวเตอร์ในระยะไกลอยู่ในโหมด server แล้วเท่านั้น ถ้ายังไม่ได้ทำดังกล่าว ก็จำเป็ฯต้องย้ายไฟล์โดยใช้คีย์ PuUp และ PuDn (ดังอธิบายในหัวข้อ "การส่งไฟล์ และการรับไฟล์" แล้ว)

การใช้คำสั่ง kermit server ให้ป้อนคำสั่งที่คอมพิวเตอร์ระยะไกลต้องการเพื่อเป็ฯการตั้ง ไว้ในโหมด kermit server กดคีย์ ALT-Kหรือเลือกคำสั่ง FILE แล้วเลือก kermit sev Cmd จากบรรทัดเมนูกดคีย์ 1 เพื่อรับไฟล์ หรือกดคีย์ 2 เพื่อส่งไฟล์ พิมพ์ชื่อไฟล์ที่ต้องการรับหรือส่ง แล้วกดคีย์ ENTER เมื่อให้ server รับไฟล์ ชื่อไฟล์ต้องเป็ฯไปตามข้อตกลงของคอมพิวเตอร์ในระยะไกล กดคีย์ 3 (FINISH - จบแล้ว) เพื่อบอกคอมพิวเตอร์ในระยะไกลว่าทำงานกับคำสั่ง kermit จบแล้ว และต้องการกลับไปยังโปรแกรมปกติของคอมพิวเตอร์ กดคีย์ 4 (LOGOUT-เลิกการติดต่อ) เพื่อจบคำสั่ง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

kermit และเลิกการติดต่อกับคอมพิวเตอร์ในระยะไกล

7.21 Key Mapping

PROCOMM PLUS ช่วยให้กำหนดที่อยู่ใหม่ (remap) ของคีย์ใดคีย์หนึ่งใน 79 คีย์ที่อยู่ในแต่ละแบบจำลองของเทอร์มินัลได้ การกำหนดที่อยู่ใหม่ของคีย์นี้จะเปลี่ยนหน้าที่ของคีย์นั้น (คือการปฏิบัติ หรือ การปรากฏตัวอักษรจากการกดคีย์นั้น ๆ) สามารถกำหนดหน้าที่ของคีย์ชุดใหม่สำหรับแต่ละการจำลองแบบเทอร์มินัลได้

หมายเหตุ ก่อนที่จะเปลี่ยนการกำหนดใดๆ ให้ทำการก๊อปปี้สำรองของไฟล์ PCPLUS.KBD ไว้ก่อน การเปลี่ยนที่อยู่ของคีย์บอร์ด ให้กดคีย์ ALT-F8 หรือเลือกคำสั่ง change แล้วเลือก key mapping จากบรรทัดเมนู

7.22 Line/Port Setup

คำสั่งนี้ช่วยในการกำหนดการตั้งค่าต่าง ๆ ในสาย (ได้แก่ baud rate, parity,จำนวนบิตข้อมูล และจำนวนบิตหยุด) และการกำหนดฟอร์ตการติดต่อสื่อสารที่จะใช้ในการทำงานนี้ การเปลี่ยนการตั้งค่าต่าง ๆ หรือการกำหนดพอร์ตการติดต่อสื่อสาร ให้กดคีย์ ALT-P หรือเลือกคำสั่ง CHANGE แล้วเลือก Line Settings จากบรรทัดเมนู

7.23 Log File On/Off

คำสั่งนี้ช่วยในการลงบันทึกทุกสิ่งที่ปรากฏ ในจอแสดงผลเมื่อตติดต่อกับคอมพิวเตอร์อีกเครื่องหนึ่ง เราสามารถลดเวลาที่ใช้ในการติดต่อลง ซึ่งจะลดค่าโทรศัพท์ลงได้ โดยใช้ไฟล์ลงบันทึก แล้วอ่านรายละเอียดในไฟล์นั้นภายหลังเมื่อเลิกการติดต่อแล้ว(แทนที่สจะอ่านข้อมูลเมื่อข้อมูลเข้ามาอยู่ในจอภาพ)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การเปิดไฟล์สำหรับลงบันทึก ให้กดคีย์ ALT-F1 หรือเลือกคำสั่ง FILE แล้วเลือก Toggle Log จากบรรทัดเมนู PROCOMM PLUS จะขอทราบชื่อของไฟล์สำหรับลงบันทึกนี้ให้นิมพ์ชื่อของไฟล์สำหรับลงบันทึกนี้ลงไป สามารถข้ามขั้นตอนนี้ไปได้ถ้าต้องการใช้ชื่อไฟล์ที่กำหนด ไว้ให้โดยปริยายในโปรแกรมอำนวยความสะดวก Setup แล้วนอกจากว่าจะเปลี่ยนชื่อนั้น ชื่อไฟล์ที่ตั้งไว้ให้โดยปริยายคือ PCPLUS.LOG ถ้าป้อนชื่อไฟล์ที่แตกต่างจากนี้ลงไป สามารถเพิ่ม path โดยสมบูรณ์ให้กับชื่อไฟล์นี้ (รวมทั้งไดรฟ์และไดเรกทอรี) ถ้าไม่ได้กำหนด path ลงไป จะบันทึกไฟล์นี้ไว้ในไดเรกทอรีที่กำลังใช้อยู่ ถ้านิมพ์ชื่อ ไฟล์ที่มีอยู่แล้วลงไป ก็จะบวกข้อมูลใหม่ต่อท้ายไฟล์ที่มีอยู่แล้วนั้น เมื่อกดคีย์ ENTER PROCOMM PLUS จะ แสดงผลว่า LOGOPEN ในบรรทัดแสดงสถานภาพเพื่อแสดงว่าข้อมูลนั้นได้ลงบันทึกไว้แล้ว การเปิดไฟล์สำหรับลงบันทึกให้กดคีย์ ALT-F1 หรือเลือกคำสั่ง FILE แล้วเลือก Toggle Logo จากบรรทัดเมนู PROCOMM PLUS จะแสดงผลว่า LOG CLOSEDในบรรทัดแสดงสถานภาพเพื่อแสดงว่าเลิกลงบันทึกข้อมูลนั้นแล้ว

หมายเหตุ ถ้าเพียงต้องการหยุดพักชั่งคราวเทนที่จะปิดการลงบันทึกไฟล์ ให้ใช้คำสั่ง ALT-F2

7.24 Log File Pause

PROCOMM PLUS จะหยุดการลงบันทึกไฟล์ไว้เป็นการชั่วคราวเพื่อข้ามข้อมูลบางส่วนที่ปรากฏอยู่ในจอภาพไป การหยุดการลงบันทึกไฟล์ชั่วคราว ให้กดคีย์ ALT-F2 หรือเลือกคำสั่ง FILE แล้วเลือก Hold Log จากบรรทัดเมนู PROCOMM PLUS จะแสดงคำว่า LOG ON HOLD (หยุดพักการลงบันทึกไว้) ในบรรทัดแสดงสถานภาพเพื่อแสดงว่ามีการพักการลงบันทึกไว้เป็นการชั่วคราว

การกลับไปลงบันทึกไฟล์ ให้กดคีย์ ALT-F2 หรือเลือกคำสั่ง FILE แล้วเลือก Hold Log จากบรรทัดเมนู PROCOMM PLUS จะแสดงคำว่า LOG

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

OPEN (เปิดการลงบันทึก)ในบรรทัดแสดงสถานภาพเพื่อแสดงว่ากลับมาลงบันทึกข้อมูลอีกแล้ว

7.25 Meta Keys

Meta keys เป็นคำสั่งเอนกประสงค์ด้วยการกดคีย์ที่สามารถเก็บและสัญญาณ การกดคีย์ รันโปรแกรมภายนอกหรือรันสคริปต์ของ ASPECT ได้ จะสร้าง Meta key ได้โดยเลือก แบบก่อน แล้วเลือกลำดับของข้อความสคริปต์ของ ASPECT โปรแกรมภายนอกที่เป็นเอกเทศ หรือ โปรแกรมภายนอกที่เกี่ยวข้องซึ่งอ่านตัวแปรในการปฏิบัติงานจาก PROCOMM PLUS เมื่อกำหนดฟังก์ชันของ Meta key ไปแล้ว ก็สามารถป้อนค่าต่าง ๆ ดังต่อไปนี้ลงในฟิลด์ CONTENTS ได้

1. การกดคีย์สำหรับตัวหนังสือ (กดคีย์ได้ถึง 50 คีย์ต่อหนึ่ง Meta)
2. สคริปต์ที่จะรันในแบบของ ASPECT (สคริปต์สามารถจะอยู่ในแบบฟอร์มของ .ASX ที่แปล แล้ว หรือมิฉะนั้น PROCOMM PLUS จะพยายามแปลโปรแกรมต้นกำเนิด .ASP โดยอัตโนมัติ)
3. โปรแกรม .COM หรือ .EXE จากภายนอก ซึ่งจะเข้าไปเกี่ยวข้องกับค่าต่าง ๆ ของตัวแปรที่ PROCOMM PLUS ใช้อยู่

แต่ละไฟล์ของ Meta key ที่สร้างขึ้นมาสามารถบรรจุได้ถึง10 คีย์ (คือ ALT-1 ถึง ALT-0)

ไฟล์ Meta key ที่กำหนดไว้ให้โดยปริยายมีชื่อว่า PCPLUS.KEY ไฟล์นี้จะทำงานเมื่ออยู่ใน PROCOMM PLUS จนกว่าจะเปลี่ยนชื่อไฟล์เป็นอย่างอื่น เป็นไฟล์หนึ่งของรายการใน dialing directory เมื่อ โหลดไฟล์ Meta key ใหม่ หรือเมื่อใช้คำสั่ง ASPECT MLOAD

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หมายเหตุ สามารถใช้คำสั่ง ALT-M เพื่อเปลี่ยน Meta keys ใด ๆ หรือจะโหลดอีกชุดหนึ่งก็ได้ การแสดงผลของ Meta key ชุดที่ใช้อยู่ในปัจจุบัน ให้กดคีย์ ALT-M หรือเลือกคำสั่ง CHANGE แล้วเลือก Meta key จากบรรทัดเมนู การเพิ่มหรือสร้าง Meta key ให้กดคีย์ลูกศรขึ้นหรือลงเพื่อเลื่อนไปยังคีย์ที่ต้องการแก้ไข กด คีย์ R กดคีย์ spacebar เพื่อสลับไปมาระหว่าง TYPES (แบบ) ของ Meta key ทั้งสี่ กดคีย์ ENTER เมื่อทำเสร็จแล้วพิมพ์รายละเอียดที่ต้องการของคีย์นั้นลงไป ถ้าต้องการตรวจแก้สิ่งที่พิมพ์ลงไป ก็สามารถใช้คีย์ใน การตรวจแก้ตามรายการในภาคผนวกท้ายเล่ม กดคีย์ ENTER เมื่อทำเสร็จแล้ว

การบันทึกชุดของ Meta key หลังจากที่เพิ่มหรือเปลี่ยน Meta key แล้ว ก็ควรที่จะบันทึก Meta key ชุดนี้ไว้ถ้าต้องการใช้ชุดของ Meta key นี้ในการสตาร์ต PROCOMM PLUS ครั้งต่อไป ถ้าไม่บันทึก ชุดของ Meta key ไว้ คำสั่งการกดคีย์ใหม่จะมีอยู่ต่อเมื่อทำงานอยู่เท่านั้น

กดคีย์ S PROCOMM PLUS จะบันทึกการแก้ไขเปลี่ยนแปลงใน Meta key ชุดที่ใช้อยู่ในขณะนั้น กดคีย์ ENTER PROCOMM PLUS จะบันทึกไฟล์ Meta key ไว้ในไดเรกทอรีที่กำลังใช้อยู่ (หรือไดเรกทอรีที่มีชื่อว่า PCPLUS= เป็นตัวแปร)

การลบชุด Meta key ถ้าต้องการสร้างไฟล์ Meta key ขึ้นใหม่นั่นคือไฟล์ที่แตกต่างจาก Meta key ชุดที่ใช้อยู่ในปัจจุบันโดยสิ้นเชิง ให้กดคีย์ C เพื่อลบการกำหนดคีย์ทั้งสิบออกไป

หมายเหตุ คำสั่งนี้ลบ Meta key ชุดที่ใช้อยู่ในปัจจุบันออกจากหน่วยความจำแต่จะไม่ลบไฟล์ที่ ตรงกับที่เก็บไว้ในดิสก์

การโหลด Meta key ชุดอื่น ให้กดคีย์ L พิมพ์ชื่อไฟล์ของไฟล์ Meta key ลงไปเพื่อโหลด แล้วกดคีย์ ENTER

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 8

เทคนิคการออกแบบวงจรดิตอล

ในวงจรที่ออกแบบนี้ สามารถแบ่งการทำงานออกเป็นสองส่วนใหญ่ๆ คือ ส่วนเข้ารหัส จากการกดคีย์ทั้ง 16 คีย์ แล้วแปลงให้เป็นสัญญาณออกมา 4 เส้นแบบเลขฐานสองและส่วนเปลี่ยนสัญญาณที่เป็นเลขฐานสองทั้ง 4 เส้น ให้เป็นรหัสที่ใช้แสดงผล ออกแบบส่วนแรกจาก 16 คีย์ให้มี 4 เอาต์พุต

เมื่อรู้การทำงานของแต่ละส่วนแล้ว ก็ต้องเข้าถึงรายละเอียดในแต่ละส่วนโดยกำหนดประเภทไอซี ที่ใช้ ในที่นี้คือ ไอซีตระกูลทีทีแอล (TTL) ซึ่งเป็นตระกูลที่เหมาะสำหรับการทดลองเกี่ยวกับวงจรดิจิตอลคอมพิวเตอร์ทั่วไป จากนั้นก็เปิดดูที่คู่มือไอซีทีทีแอล เมื่อกล่าวถึงคู่มือแล้วขอแนะสักนิดว่า คู่มือเป็นสิ่งที่สำคัญ สำหรับนักออกแบบที่ควรมีไว้ และเปิดดูเป็นประจำเมื่อมีเวลาว่าง ว่าแต่ละวงจรทำหน้าที่อะไร ทำงานอย่างไร โดยดูในส่วนวงจรภายในจะทำให้เข้าใจยิ่งขึ้นเวลาออกแบบก็สามารถมองออกได้เลยว่า คุณสมบัติของวงจรที่ออกแบบนี้ควรใช้ ไอซีที่มีการทำงานในลักษณะใดประกอบ ในตอนนี้ใช้ไอซีที่มีการทำงานลักษณะการเข้ารหัส (encode) ซึ่งใช้เข้ารหัสจากทั้ง 16 คีย์ขอยกตัวอย่างสองเบอร์เพื่อใช้พิจารณา คือ เบอร์ 74147 มีสัญญาณเข้า 10 เส้น และสัญญาณออก 4 เส้นแบบ BCD ส่วนเบอร์ 74148 มีสัญญาณเข้า 8 เส้นและสัญญาณออก 3 เส้น แบบเลขฐานสอง แต่ที่ใช้งานคือสัญญาณเข้า 16 เส้นและออก 4 เส้น แบบเลขฐานสอง เนื่องจากไม่มีในคู่ มือไอซี ดังนั้นจึงเลือกเบอร์ 74148 ซึ่งมีสัญญาณเข้า 8 เส้น โดยใช้ไอซีสองตัวมาประกอบกัน แต่ก็มีปัญหาที่ว่าจะต่ออย่างไรมันจึงจะทำงานได้ ถึงตรงนี้ก็ต้องมาดูรายระเอียดในตารางการทำงาน ของ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่8

เทคนิคการออกแบบวงจรดิตอล

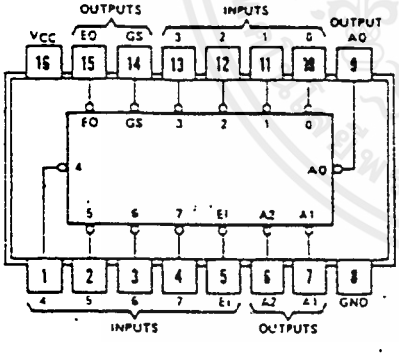
ในวงจรที่ออกแบบนี้ สามารถแบ่งการทำงานออกเป็นสองส่วนใหญ่ๆ คือ ส่วนเข้ารหัส จากการกดคีย์ทั้ง 16 คีย์ แล้วแปลงให้เป็นสัญญาณออกมา 4 เส้นแบบเลขฐานสองและส่วนเปลี่ยนสัญญาณที่เป็นเลขฐานสองทั้ง 4 เส้น ให้เป็นรหัสที่ใช้แสดงผล ออกแบบส่วนแรกจาก 16 คีย์ให้มี 4 เอาต์พุต

เมื่อรู้การทำงานของแต่ละส่วนแล้ว ก็ต้องเข้าถึงรายละเอียดในแต่ละส่วนโดยกำหนดประเภทไอซี ที่ใช้ ในที่นี้คือ ไอซีตระกูลทีทีแอล (TTL) ซึ่งเป็นตระกูลที่เหมาะสำหรับการทดลองเกี่ยวกับวงจรดิจิตอลคอมพิวเตอร์ทั่วไป จากนั้นก็เปิดดูที่คู่มือไอซีทีทีแอล เมื่อกล่าวถึงคู่มือแล้วขอแนะสักนิดว่า คู่มือเป็นสิ่งที่สำคัญ สำหรับนักออกแบบที่ควรมีไว้ และเปิดดูเป็นประจำเมื่อมีเวลาว่าง ว่าแต่ละวงจรทำหน้าที่อะไร ทำงานอย่างไร โดยดูในส่วนวงจรภายในจะทำให้เข้าใจยิ่งขึ้นเวลาออกแบบก็สามารถมองออกได้เลยว่า คุณสมบัติของวงจรที่ออกแบบนี้ควรใช้ ไอซีที่มีการทำงานในลักษณะใดประกอบ ในตอนนี้ใช้ไอซีที่มีการทำงานลักษณะการเข้ารหัส (encode) ซึ่งใช้เข้ารหัสจากทั้ง 16 คีย์ขอยกตัวอย่างสองเบอร์เพื่อใช้พิจารณา คือ เบอร์ 74147 มีสัญญาณเข้า 10 เส้น และสัญญาณออก 4 เส้นแบบ BCD ส่วนเบอร์ 74148 มีสัญญาณเข้า 8 เส้นและสัญญาณออก 3 เส้น แบบเลขฐานสอง แต่ที่ใช้งานคือสัญญาณเข้า 16 เส้นและออก 4 เส้น แบบเลขฐานสอง เนื่องจากไม่มีในคู่ มือไอซี ดังนั้นจึงเลือกเบอร์ 74148 ซึ่งมีสัญญาณเข้า 8 เส้น โดยใช้ไอซีสองตัวมาประกอบกัน แต่ก็มีปัญหาที่ว่าจะต่ออย่างไรมันจึงจะทำงานได้ ถึงตรงนี้ก็ต้องมาดูรายระเอียดในตารางการทำงาน ของ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ไอซีเบอร์ 74148 นี้ ดังตารางที่ 1 จากตารางที่ 1 ดูที่อินพุต EI (enable input) จะเห็นว่าถ้าให้เป็น "L" แล้วอินพุท 0 ถึง 7 มีการทำงานเป็นแบบลำดับความสำคัญ (priority) คือ หากอินพุต 7 เป็น "L" มันจะไม่สนใจ อินพุตอื่นๆเลยโดยค่าเอาต์พุท A0 ถึง A2 เปลี่ยนแปลงตามอินพุทที่เข้ามา สามารถนำอินพุท 0 ถึง 7 นี้ไปต่อกับคีย์ได้ ส่วนเอาต์พุท EO (enable output) มีสภาวะ "L" เมื่ออินพุท 0 ถึง 7 เป็น "H" หมด ซึ่งแทนสภาวะนี้ในขณะไม่มีการกดคีย์ หากมีการกดคีย์แล้วที่เอาต์พุต EO ก็จะกลายเป็น สภาวะ "H" ดังนั้นหากใช้ทั้งหมด 16 คีย์ต้องใช้ไอซีสองตัวและใช้เอาต์พุท EO ตัวใดตัวหนึ่งเป็นตัวไปควบคุมสวิตช์ เพื่อเลือกตัวที่มีการกดคีย์ได้ถูกต้อง โดยใช้ EO นี้เป็นเอาต์พุตที่สี่ (ไอซี 74148 มี 3 เอาต์พุต)

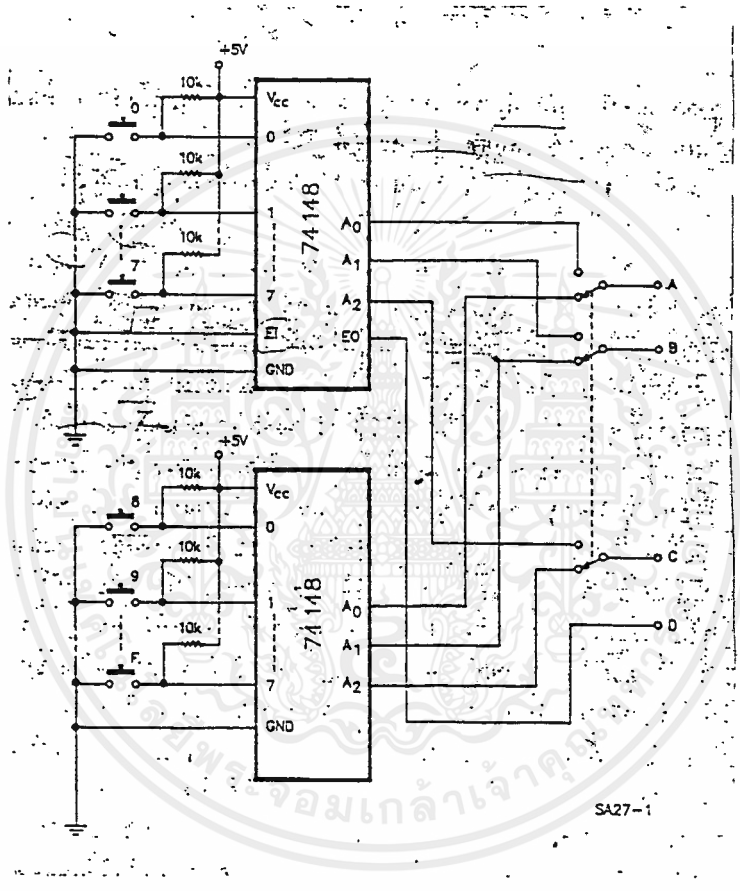
ตารางที่ 1 แสดงวงจรภายในและตารางการทำงานของไอซี 74148



อินพุต									เอาต์พุต				
EI	0	1	2	3	4	5	6	7	A2	A1	A0	GS	EO
H	X	X	X	X	X	X	X	X	H	H	H	H	H
L	H	H	H	H	H	H	H	H	H	H	H	H	L
L	X	X	X	X	X	X	X	L	L	L	L	L	H
L	X	X	X	X	X	X	L	H	L	L	H	L	H
L	X	X	X	X	X	L	H	H	L	H	L	L	H
L	X	X	X	X	L	H	H	H	L	H	H	L	H
L	X	X	X	L	H	H	H	H	H	L	L	L	H
L	X	X	L	H	H	H	H	H	H	L	H	L	H
L	X	L	H	H	H	H	H	H	H	H	L	L	H
L	L	H	H	H	H	H	H	H	H	H	H	L	H

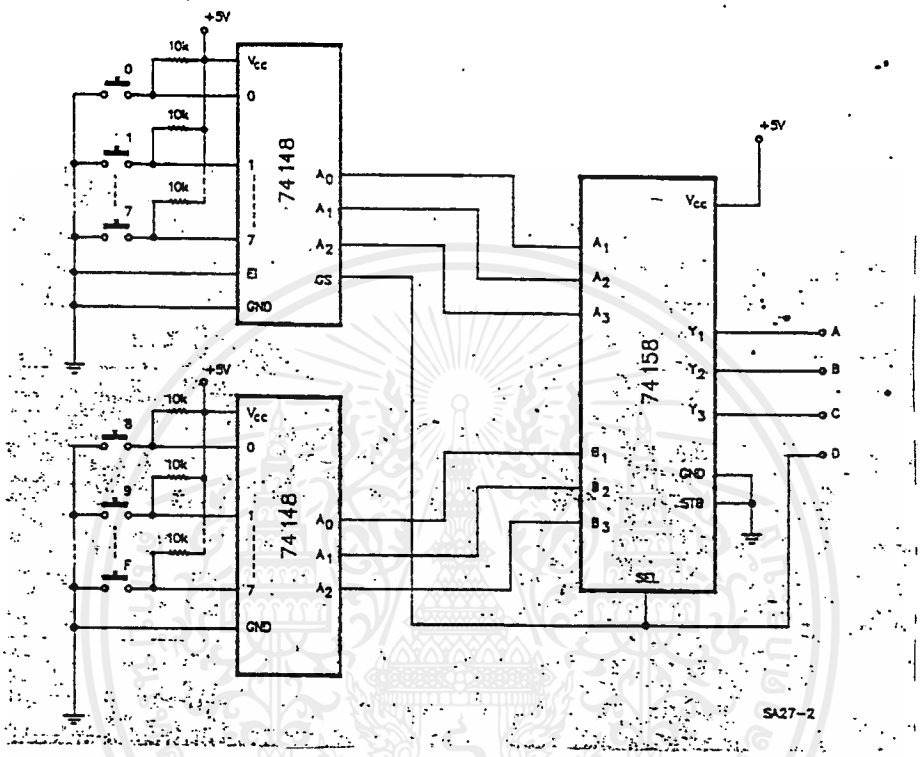
สิ่งที่ต้องเพิ่มให้กับวงจรในตอนนี้ก็คือสวิตช์ 3 ตัว ซึ่งสามารถควบคุมได้ด้วยสัญญาณลอจิก ดังแสดงในรูปที่ 8.1

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 8.1 แสดงการต่อไอซี 74148 เข้ากับสวิตช์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



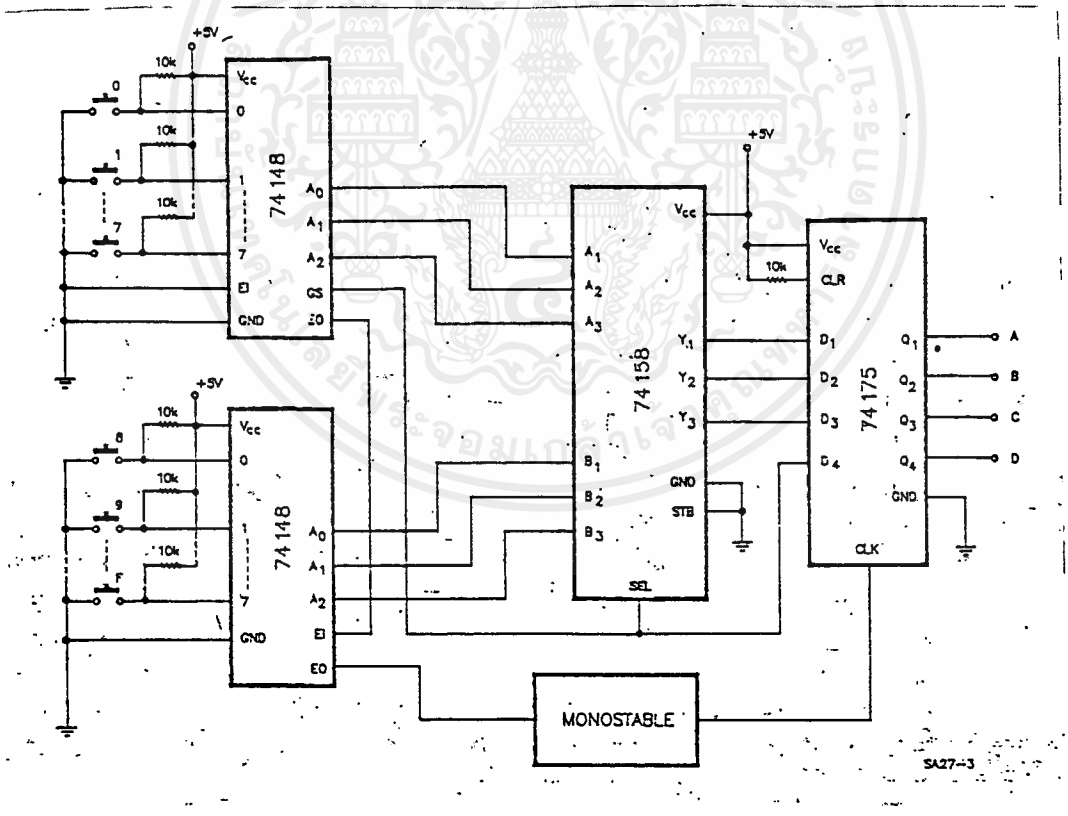
รูปที่ 8.2 แสดงการใช้ไอซี 74158 มาต่อแทนสวิตช์

ตารางที่ 2 แสดงการทำงานของไอซี 74158.

อินพุต				เอาต์พุต
STROBE	SELECT	A	B	Y
H	X	X	X	H
L	L	L	X	H
L	L	H	X	L
L	H	X	L	H
L	H	X	H	L

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โดยใช้ไอซีทำงานแทนสวิตช์ก็เปิดหาในคู่มือเช่นเคย คือ ดูกลุ่มที่ทำงานประเภทตัวเลือกข้อมูล (data selectors) มีอยู่ 4 เบอร์ ที่ตรงกับความต้องการคือ เบอร์ 74157, 74158, 74257, และ 74258 ในที่นี้เลือกเบอร์ 74158 เนื่องจากมีเอาต์พุตกลับลอจิก (ไอซี 74158 ที่เอาต์พุตจะกลับลอจิก ดังนั้นจึงต้องมีการกลับลอจิก อีกครั้งหนึ่งเพื่อจะได้เอาต์พุตปกติ) และมีฟังก์ชันการทำงานเพียงพอในการเลือกเบอร์ไอซีนี้ เลือกตัวที่มีฟังก์ชันการทำงานน้อยที่สุดและเพียงพอต่อการใช้งานเพื่อเป็นการลดต้นทุนการผลิต รวมไปถึงการประหยัดไฟ กล่าวคือไอซีที่มีฟังก์ชันการทำงานมากก็จะกินไฟมากนั่นเอง ไอซีเบอร์ 74158 นี้มีวงจรสวิตช์ภายในอยู่ 4 ชุด สามารถควบคุมได้ด้วยขา SELECT ซึ่งแสดง



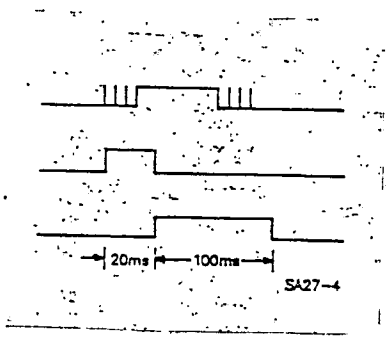
รูปที่ 8.3 แสดงวงจรเมื่อต่อไอซี 74175 เข้าไป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางการทำงานได้ดังตารางที่ 2 จากตารางที่ 2 เมื่อให้ STROBE เป็น "L" และให้ SELECT เป็น "L" แล้วที่เอาต์พุท Y กลับลอจิกกับอินพุท A แต่ถ้าให้ SELECT และ "H" เอาต์พุท Y กลับลอจิกกับอินพุต B โดยนำ SELECT นี้ไปต่อกับขา EO ของไอซีเบอร์ 74148 แต่เนื่องจากถ้าใช้ขา EO นี้ไปต่อกับ SELECT แล้วที่เอาต์พุตจะกลับคือเมื่อกดคีย์ของไอซีตัวบน (คีย์ 0 ถึง 7) ที่เอาต์พุทแสดงค่า 8 ถึง F และถ้ากดคีย์ ของไอซีตัวล่าง (คีย์ 8 ถึง F) ที่เอาต์พุตจะเป็นค่า 8 ถึง F ดังนั้นใช้ขา GS แทนขา EO (มีลอจิกกลับกับขา EO) แสดงได้ดังรูปที่ 8.2 วงจรที่ออกแบบนี้ยังไม่สมบูรณ์คือเมื่อกดคีย์แล้วปล่อย ค่าที่เอาต์พุทจะหายไปด้วย ดังนั้นการเก็บค่าที่เอาต์พุตไว้เมื่อคีย์ ต้องหาไอที่มีคุณสมบัติที่เก็บค่าได้ ซึ่งก็เป็นคุณสมบัติของ D ฟลิปฟลอป

โดยเปิดคู่มือหาไอซีที่มี D ฟลิปฟลอปภายในอยู่ 4 ชุด ได้เบอร์ 74175 โดยปกติ D ฟลิปฟลอป จะเก็บค่าก็ต่อเมื่อให้สัญญาณกระตุ้นที่ขาคล็อก ดังนั้นวงจรนี้จึงต้องหาสัญญาณมาเข้าที่ขาคล็อก ซึ่งก็คือสัญญาณที่บอกให้ทราบว่ามีการกดคีย์ ในที่นี้ใช้สัญญาณจากขา EO ของไอซี 74148 ตัวล่าง โดยต้องต่อขา EO ของตัวบนมาเข้ากับ EI ตัวล่าง ซึ่งเป็นการรวมสัญญาณ EO ทั้งสองแล้ว นำออกมาตัวล่างตัวเดียว แต่ก็ไม่สามารถต่อ EO เข้ากับคล็อกของไอซี 74175 ได้โดยตรง เนื่องจากที่ขาคล็อกของ 7415 ทำงานที่สัญญาณนาฬิกาขาขึ้น เมื่อต่อ EO เข้ากับคล็อกโดยตรงแล้วหากกดคีย์สัญญาณ EO จะออกมาพร้อมกับที่ส่งค่าคีย์ที่กดออกไปซึ่งอาจทำให้ข้อมูลผิดพลาดได้ จึงต้องมีการหน่วงเวลาช่วงหนึ่งก่อนและในการกดคีย์นี้จะเกิดการกระทบของหน้าสัมผัสเป็นผลทำให้เกิดสัญญาณเบ๊าซ์ (bounce) หรือพัลส์เล็กๆ เกิดขึ้น ซึ่งสิ่งนี้ทำให้ข้อมูลที่เอาต์พุทมีการผิดพลาดได้มาก วิธีแก้ก็คือต้องสร้างวงจรที่ทำให้เหลือพัลส์เพียงพัลส์เดียว ที่จะนำไปต่อกับคล็อกของไอซี 74175 วงจรนี้เราเรียกว่า วงจรโมโนสเตเบิ้ล (monostable) ก่อนอื่นขอให้ดูวงจรเมื่อเราต่อไอซี 74175 เข้าไปก่อนดังรูปที่ 8.3 จากรูปที่ 8.3 วงจรส่วนที่เหลือคือส่วนโมโนสเตเบิ้ล ซึ่งต้องทำการออกแบบก่อนอื่นต้องกำหนดฟังก์ชัน การทำงานของวงจรส่วนนี้ก่อนว่าให้มันทำงานอย่างไร โดยในที่นี้กำหนดให้เป็นไปตามรูปที่ 8.4

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 8.4 แสดงรูปสัญญาณจากการกดคีย์ (EO) และสัญญาณคล็อกที่ต้องการ

จากรูปที่ 8.4 EO เป็นสัญญาณที่ได้จากการกดคีย์โดยมีพัลส์เล็กๆ ออกมาด้วย และส่วนของเอาต์พุตของวงจรที่ต้องสร้างขึ้น คือสัญญาณรูปล่างทั้งสองโดยรูปกลางเป็นการหน่วงเวลาเพื่อเลื่อนช่วงกระตุ้นขาคล็อกจากช่วงที่มีพัลส์เล็กๆ ไปเป็นช่วงเวลาสัญญาณคงที่ ซึ่งช่วงเวลานี้ประมาณ 20 มิลลิวินาที หลังจากหน่วงเวลาแล้วก็จะกระตุ้นขาคล็อก ให้ทำงานและต้องหน่วงเวลาอีกระยะหนึ่งเพื่อให้ปล่อยคีย์ก่อน ซึ่งช่วงเวลานี้ใช้เวลาประมาณ 100 มิลลิวินาที จากนั้นจึงรอรับคีย์ใหม่อีกครั้งได้ จากการกำหนดสัญญาณที่เอาต์พุต ของวงจรโมโนสเตเบิ้ลในรูปที่ 4 นั้นสังเกตุว่า การทำงานแบ่งเป็นสองช่วงด้วยกัน ดังนั้นต้องสร้างวงจรโมโนสเตเบิ้ลสองวงจร โดยวงจรแรกทำงานที่สัญญาณนาฬิกาขาขึ้น มีช่วงเวลาหน่วงประมาณ 20 มิลลิวินาที ส่วนวงจรที่สองทำงานที่สัญญาณนาฬิกา ขาลงมีช่วงเวลาหน่วงประมาณ 100 มิลลิวินาที เมื่อทราบรายละเอียดทุกอย่างของวงจรแล้ว ต่อไปก็ทำการออกแบบ โดยหาไอซีที่ทำหน้าที่นี้ ซึ่งก็มีอยู่หลายตัวคือ เบอร์ 74121,74122, 74221, และ 74123 ซึ่งทั้งหมดแบ่งได้สองประเภทคือ กระตุ้นได้เพียงเดียวและแบบกระตุ้นซ้ำได้แบบกระตุ้นได้เพียงครั้งเดียวนี้เมื่อมีสัญญาณมากระตุ้นมันจะทำงานทันที และไม่สนใจกับสัญญาณที่เข้าอีกจนกว่าช่วงเวลาที่หน่วงสิ้นสุด ส่วนแบบกระตุ้นซ้ำได้เมื่อมีสัญญาณมากระตุ้นมันจะทำงานทันที และหากมีสัญญาณกระตุ้นเข้ามามันจะเริ่มหน่วงเวลาใหม่ ดังนั้นช่วงหน่วงเวลาอาจนานกว่าที่กำหนดไว้ เมื่อมาพิจารณาจากวงจรแล้วควรใช้แบบกระตุ้นซ้ำได้ ซึ่งดีกว่าแบบกระตุ้นเพียงครั้งเดียว เนื่อง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากช่วงการเกิดพลัส์เล็กๆ ของแต่ละหน้าสัมผัสอาจไม่เท่ากัน ข้อพิจารณาในการเลือกเบอร์ไอซีต่อไปก็คือในวงจรของเราต้องใช้ถึงสองวงจรภายในตัวเดียวกันจะดีมาก คุณสมบัติเหล่านี้ทำให้เราเลือกเบอร์ 74123

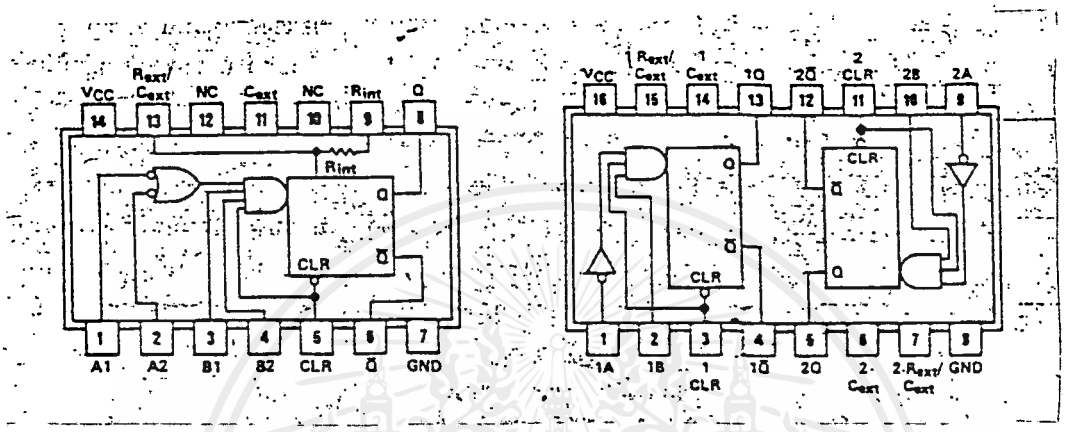
ตารางที่ 3 แสดงการทำงานของไอซี 74122, 74123

อินพุต					เอาต์พุต	
CLEAR	A1	A2	B1	B2	Q	$\bar{Q}$
L	X	X	X	X	L	H
X	H	H	X	X	L*	H*
X	X	X	L	X	L*	H*
X	X	X	X	L	L*	H*
H	L	X	↓	H	⎍	⎎
H	L	X	H	↑	⎍	⎎
H	X	L	↓	H	⎍	⎎
H	X	L	H	↑	⎍	⎎
H	H	↓	H	H	⎍	⎎
H	↓	↓	H	H	⎍	⎎
H	↓	H	H	H	⎍	⎎
↑	L	X	H	H	⎍	⎎
↑	X	L	H	H	⎍	⎎

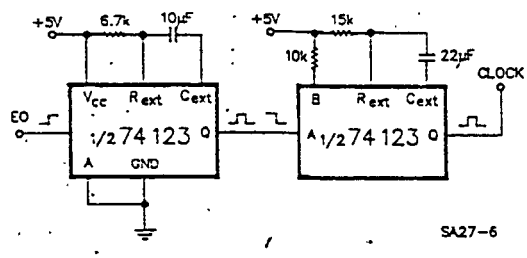
อินพุต			เอาต์พุต	
CLEAR	A	B	Q	$\bar{Q}$
L	X	X	L	H
X	H	X	L*	H*
X	X	L	L*	H*
H	↓	↑	⎍	⎎
H	↓	H	⎍	⎎
↑	L	H	⎍	⎎

จากตารางที่ 3 เป็นการแสดงการทำงานของไอซีเบอร์ 74123 ซึ่งอินพุท A ใช้กระตุ้นด้วยสัญญาณนาฬิกาขาลง ส่วนอินพุต B ใช้กระตุ้นด้วยสัญญาณนาฬิกาขาขึ้น ในวงจรแรกใช้กระตุ้นด้วยสัญญาณนาฬิกาขาขึ้น จึงต้องต่อกับอินพุต B สำหรับอินพุท A ต่อลงกราวด์ ส่วนวงจรที่สองกระตุ้นด้วย สัญญาณนาฬิกาขาลง ดังนั้นการต่อเข้าอินพุต A ส่วนอินพุต B ต้องต่อที่ "H" ซึ่งแสดงการต่อได้ดังรูปที่ 8.6

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 8.5 แสดงการวางขาของไอซี 74122, 74123



รูปที่ 8.6 แสดงการต่อวงจรโมโนสเตเบิ้ลให้ได้สัญญาณตามรูปที่ 8.4

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากรูปที่ 8.6 ตัวต้านทาน 10 กิโลโอห์ม ต่อไว้เพื่อให้ขา B เป็น "H" ตลอด ซึ่งไม่ต่อกับ +5 โวลต์ โดยตรงเนื่องจากทำให้กระแสไหลเข้ามากเกินไป ค่าของตัวต้านทานและตัวเก็บประจุที่ยังไม่รู้ค่า สามารถคำนวณได้จากอ่านเวลาหน่วงที่เราต้องการ โดยมีวิธีคำนวณดังนี้

$$\text{ค่าเวลาหน่วง } (T_{ow}) = 0.3\ RC\ (1 + 0.7/R)$$

$$T_{ow} / 0.3\ C = R + 0.7$$

$$R = T_{ow} / 0.3\ C + 0.7$$

คำนวณวงจรแรก ค่าหน่วงเวลา 20 มิลลิวินาที (ms) โดยกำหนดให้ค่า C_1 = 10 ไมโครฟารัด (μF)

ดังนั้น $R_1 = 20\ ms / (0.3 \times 10\ \mu F) + 0.7$

$= 6.6$ กิโลโอห์ม

แทนได้ด้วยค่า 6.7 กิโลโอห์ม

คำนวณวงจรที่สอง ค่าหน่วงเวลา 100 มิลลิวินาที โดยกำหนดให้ค่า C_2 = 22 ไมโครฟารัด

ดังนั้น $R_2 = 100\ ms / (0.3 \times 22\ \mu F) - 0.7$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

= 15 กิโลโอห์ม

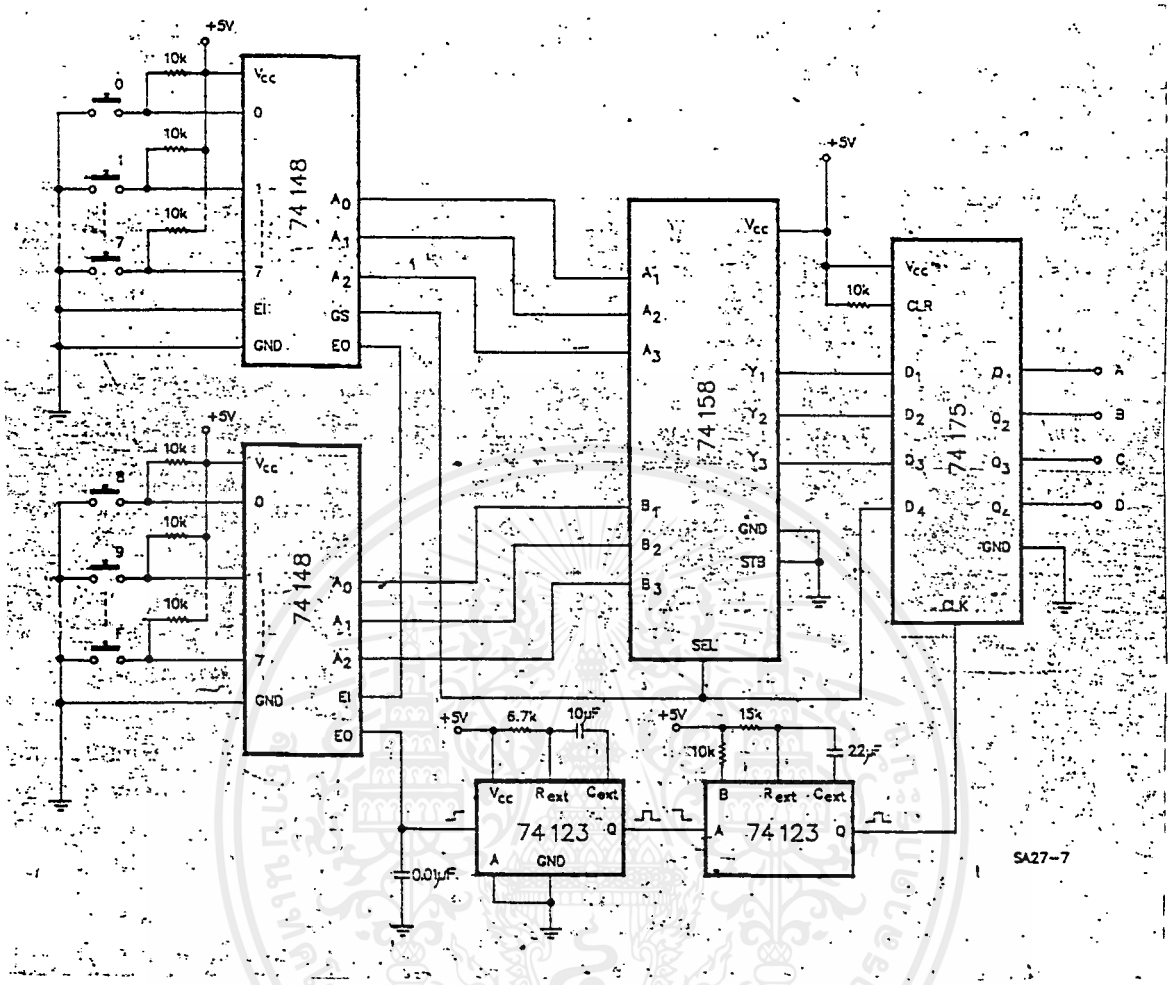
เมื่อได้ค่าต่างๆ แล้ววงจรโมโนสเตเบิ้ลก็เรียบร้อย และเมื่อนำไปประกอบกับวงจรที่ได้ออกแบบไว้ในรูปที่ 8.3 แล้วจะสามารถแสดงได้ดังรูปที่ 8.7 (ค่าตัวเก็บประจุ 0.01 ไมโครฟารัด ที่ก่อนเข้าสายไอซี 74123 มีไว้เพื่อลดสัญญาณรบกวนที่อาจเกิดขึ้นในระหว่างการทำงาน)

8.1 การออกแบบวงจรส่วนที่สอง ส่วนแสดงผล

เมื่อได้ส่วนรับคีย์แล้ว ต่อไปนำค่าคีย์ที่กดไปเปลี่ยนเป็นรหัสที่ใช้แสดงผลกับ LED 7 ส่วนได้ ในการออกแบบวงจรเปลี่ยนรหัสนี้สามารถใช้ตาราง คาร์นอร์-เมป (Karnaugh map) ได้ แต่ถ้า หากวงจรมีหลายๆ เอาต์พุต และมีไอซีเกตมาประกอบเป็นจำนวนมากจะทำให้สิ้นเปลืองและยุ่งยากใน การประกอบวงจรนี้ก็เช่นกัน พบว่ามีเอาต์พุตมากถึง 7 เอาต์พุตด้วยกัน การออกแบบให้วงจรมีการประกอบกันของของเกตให้มีจำนวนน้อยนั้น เป็นสิ่งที่ทำได้ยากมาก ดังนั้นในวงจรนี้ไม่ใช้การประกอบกันของเกต แต่นำเอาหน่วยความจำที่เราเรียกกันว่า อีพรอม (EPROM) มาใช้แทนโดยการโปรแกรมข้อมูลในตำแหน่งที่ต้องการเข้าไปก่อน จากนั้นก็เพียงใช้ขาตำแหน่งหน่วยความจำเป็นอินพุต และให้ข้อมูลที่ออกไปเป็นเอาต์พุต อีพรอมที่มีขายในปัจจุบันนี้ที่นิยมใช้มากที่สุดคือ เบอร์ 2716 ซึ่งมีขนาดความจำ 2048X8 หรือ 2 กิโลไบต์ ซึ่งนำมาใช้ในวงจรนี้ วิธีการต่อก็ทำเหมือนกับการต่อแบบหน่วยความจำ โดยไอซีเบอร์นี้

มีขาเลือกตำแหน่งทั้งหมด 11 ขา แต่ใช้เป็นขาอินพุตเพียง 4 ขา ดังนั้นขาที่เหลือต้องต่อลงกราว์ ให้หมด ส่วนขาข้อมูลมีทั้งหมด 8 ขา แต่ใช้เพียง 7 ขา เพื่อนำขาไปต่อกับ LED 7 ส่วน ในการต่อกับ LED 7 ส่วนนี้จะต่อโดยตรงไม่ได้ เพราะจะทำให้ไอซีต้องจ่ายกระแสมากเกินไป ดังนั้นจึง ต้องมีไอซีบัฟเฟอร์ (buffer) มาเป็นตัวช่วยจ่ายกระแสให้กับ LED 7 ส่วน ส่วนขาที่เลือกให้หน่วยความจำทำงาน (ขา OE และขา CE) ต้องต่อลงกราว์เพื่อให้มันทำงานได้ตลอด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 8.7 แสดงวงจรที่สมบูรณ์ของวงจรรับคีย์ 16 คีย์ให้มี 4 เอาต์พุต

เวลา ในส่วนของบัฟเฟอร์ใช้ไอซี ทีทีแอล เบอร์ 74244 มีเกตภายใน 8 ชุด ไอซีเบอร์นี้มีขาควบคุมที่สามารถทำให้เอาต์พุตเป็นสภาวะลอย (ความต้านทานสูง) ได้ดังนั้นต้องต่อขานี้ลงกราว์ตลอดวงจรที่ได้แสดงในรูปที่ 8.8 ไอซี 2716 นี้ต้องมีโปรแกรมเข้าไปก่อนที่จะนำไปใช้งานได้ วิธีการกำหนดข้อมูลเพื่อนำเข้าเก็บนี้ ทำได้โดยการกระจายค่าในแต่ละไบต์ ออกเป็นบิตในแต่ละส่วนของ LED 7 ส่วน ซึ่งสามารถเขียนได้ดังตารางที่ 4

จากตารางที่ 4 คอลัมน์แรกเป็นตัวเลขที่ต้องการหาว่า มีข้อมูลเป็นอะไร ส่วนคอลัมน์กลางคือ D7-D0 เป็นการกำหนดให้ในแต่ละส่วนของ LED 7

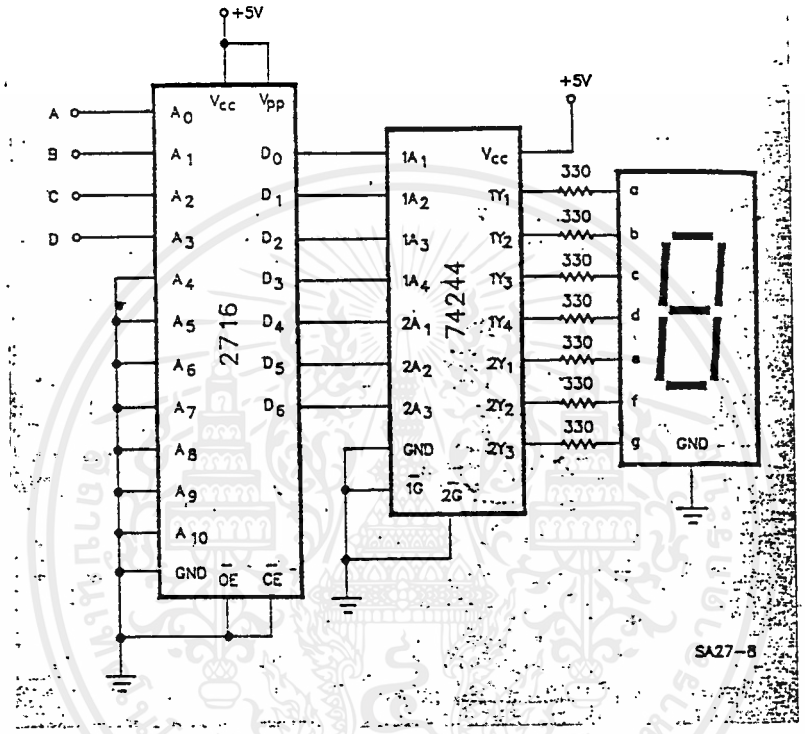
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 4 แสดงการหาค่าข้อมูลที่จะนำไปเก็บในไอซี 2716

ตัวเลข	D7	D6 (g)	D5 (f)	D4 (e)	D3 (d)	D2 (c)	D1 (b)	D0 (a)	ข้อมูล
0	0	0	1	1	1	1	1	1	3F
1	0	0	0	0	0	1	1	0	06
2	0	1	0	1	1	0	1	1	5B
3	0	1	0	0	1	1	1	1	4F
4	0	1	1	0	0	1	1	0	66
5	0	1	1	0	1	1	0	1	6D
6	0	1	1	1	1	1	0	1	7D
7	0	0	0	0	0	1	1	1	07
8	0	1	1	1	1	1	1	1	7F
9	0	1	1	0	1	1	1	1	6F
A	0	1	1	1	0	1	1	1	77
B	0	1	1	1	1	1	0	0	7C
C	0	0	1	1	1	0	0	1	39
D	0	1	0	1	1	1	1	0	5E
E	0	1	1	1	1	0	0	1	79
F	0	1	1	1	0	0	0	1	71

ส่วนแทนแต่ละบิต คือส่วนที่เป็น 0 ของตัวแสดงผลจะถูกแทนด้วยบิตที่ 0 โดยกำหนดว่าถ้าหลอดติดให้บิตนั้นมีค่าเป็น 1 และถ้าหลอดดับให้หลอดนั้นมีค่าเป็น 0 ส่วนคอลัมน์สุดท้ายเป็นค่าข้อมูลที่ได้รวมเอาแต่ละบิตเข้ามาเป็น 1 ไบต์ ซึ่งนำค่าเหล่านี้ไปโปรแกรมเข้าไปในตัวไอซี 2716 โดยเรียง ตามลำดับการเก็บตั้งแต่ไบต์ที่ 0 จนถึงไบต์ที่ F เมื่อเรานำเอาวงจรรับคีย์ และวงจรแสดงผลมาประกอบกันเป็นวงจรที่สมบูรณ์แล้ว สามารถแสดงได้ดังรูปที่ 8.9

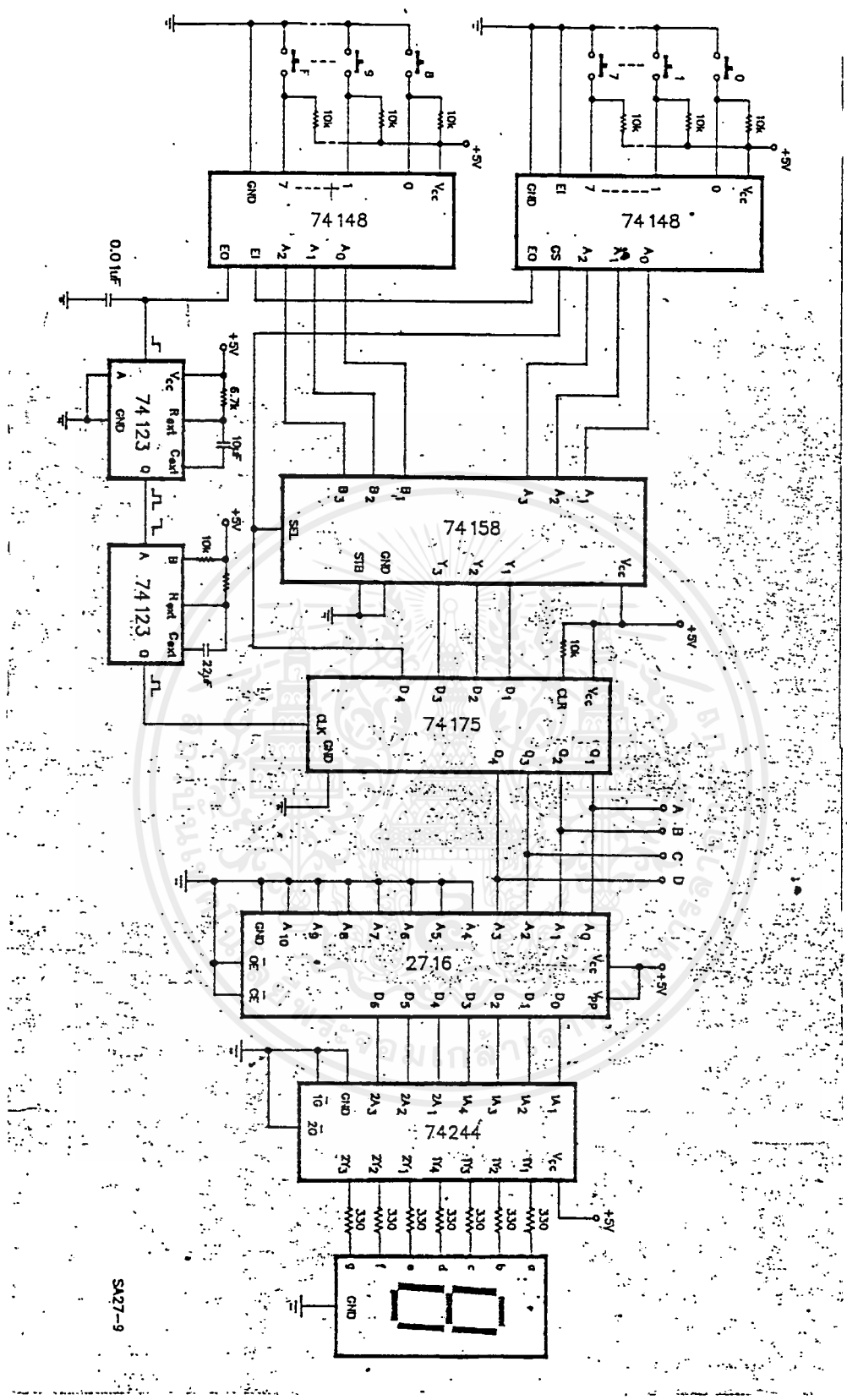
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 8.8 แสดงวงจรของส่วนแสดงผล

ไอซี ทีทีแอลที่ได้อธิบายไว้นี้เป็นชนิดธรรดา เราสามารถใช้ทดลองได้ ถ้านำไปใช้งานจริงแล้ว ขอแนะนำว่าควรใช้ชนิดที่มีเขียน LS ไว้ที่หลังตัวเลข 74 เช่น 74LS148, 74LS158 เป็นต้น ซึ่งไอซี ชนิดนี้จะกินไฟน้อยกว่าชนิดธรรมดา และเป็นที่นิยมใช้กันมาก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 8.9 แสดงวงจรสมบูรณ์ของทั้งสองส่วน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ผนวก

แนะนำให้รู้จักการติดต่อสื่อสาร

1. กล่าวทั่วไป

ถ้ามีความคุ้นเคยกับศัพท์เทคนิคที่ใช้อธิบายสิ่งที่คอมพิวเตอร์ และโมเด็มปฏิบัติดีแล้ว ก็ข้ามผนวกนี้ไปได้เลย อย่างไรก็ดี ถ้ายังไม่เข้าใจศัพท์เหล่านี้ดีนัก คำนิยามต่อไปนี้อาจจะช่วยให้เข้าใจคำอธิบายและคำสั่งในปริญญานิพนธ์เล่มนี้ได้ดีขึ้น ปริญญานิพนธ์เล่มนี้พยายามเรียบเรียงเนื้อหาให้เข้าใจได้ง่ายที่สุด เท่าที่จะทำได้อยู่แล้ว และพยายามที่จะอธิบายแต่ละขั้นตอนให้ชัดเจนด้วยภาษาง่ายๆ แต่ในบางแห่งก็จำเป็นต้องใช้ศัพท์เทคนิคอยู่บ้าง การติดต่อสื่อสารนี้เป็นเรื่องที่มีเทคนิคสูง แต่แม้กระนั้นผู้ใช้ส่วนมากก็ทราบเทคโนโลยีนี้น้อยมาก ในการอธิบายวิธีที่คอมพิวเตอร์สองเครื่องเริ่ม และรักษาการติดต่อสื่อสารไว้นั้น แทบจะหลีกไม่พ้นที่จะต้องใช้ศัพท์เทคนิค

2. โมเด็ม

ในการเรียกคอมพิวเตอร์อีกเครื่องหนึ่ง จะต้องไม่มีเพียงสายโทรศัพท์เท่านั้น แต่จะต้องมีวิธีการบางอย่างเพื่อแปลงสัญญาณไฟฟ้าของคอมพิวเตอร์ให้เป็นสัญญาณที่ส่งไปในสายโทรศัพท์ได้ โดยสรุป (และง่ายด้วย) คอมพิวเตอร์จะสร้างสัญญาณไฟฟ้าแบบฐานสอง (คือมีประจุไฟฟ้าบวกและลบ) ขึ้น ซึ่งนับว่าเป็นระบบที่ดี

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

และทำงานเร็วอย่างไม่น่าเชื่อ แต่กลับทำงานไม่ได้ดีในระบบโทรศัพท์ เพราะว่าโทรศัพท์ใช้สัญญาณที่เรียกว่า *สัญญาณความถี่เสียง* หรือ *สัญญาณอะนาลอก* ในการส่งเสียงออกไป หากลองเปรียบเทียบสัญญาณแบบฐานสองดูแล้วจะเหมือนกับมีสวิตช์เปิด/ปิดโดยต่อเนื่องนั่นเอง ในขณะที่สัญญาณความถี่เสียงนั้นเปรียบเทียบเหมือนคลื่นที่เคลื่อนที่ขึ้นๆ ลงๆ หรือมีการเพิ่มและลดระดับเสียงได้

ด้วยเหตุผลทางเทคนิคจึงเป็นไปไม่ได้ที่คอมพิวเตอร์จะส่งสัญญาณไฟฟ้าแบบฐานสองผ่านไปยังอุปกรณ์โทรศัพท์ที่ออกแบบมาสำหรับสัญญาณความถี่เสียงปัญหานี้แก้ได้โดยโมเด็มซึ่งทั้งสองสัญญาณที่ "MODulates" และที่ "DEMODulates" (MO-DEM) จะถูกส่งไปมาระหว่างคอมพิวเตอร์กับโทรศัพท์ โดยโมเด็มจะ *ปรับ* สัญญาณที่ออกจากคอมพิวเตอร์โดยแปลงจากสัญญาณตัวเลข (ดิจิตอล) ให้เป็นสัญญาณต่อเนื่อง (อะนาลอก) และที่ปลายทางจะ *ปรับกลับ* สัญญาณต่อเนื่อง (อะนาลอก) ให้เป็นสัญญาณตัวเลข (ดิจิตอล) รายละเอียดวิธีใช้การทำงาน และคุณลักษณะทางเทคนิคของโมเด็มให้ดูในคู่มือของโมเด็มนั้นๆ ในขณะนี้สิ่งที่จำเป็นต้องทราบจริงๆ เพียงอย่างเดียวเท่านั้นคือ คำว่า "การคอมแพตทิเบิลกับ Hayes" หมายความว่าอย่างไร

ตามที่ใช้กันทั่วไปนั่นคำว่า "การคอมแพตทิเบิลกับ Hayes" หมายความว่าโมเด็มนั้นใช้ชุดคำสั่งที่พัฒนาขึ้นโดยผู้ผลิตยี่ห้อ Hayes คำสั่งเหล่านั้นทั้งหมดใช้คำว่า "AT"(ย่อมาจาก ATtention)เป็นคำนำหน้า (prefix) ตามด้วยเครื่องหมายเฉพาะสำหรับการปฏิบัติต่างๆ เช่น ATDT ใช้เพื่อสั่งโมเด็มให้หมุนหมายเลขซึ่งใช้โหมด "tone"ในขณะที่ ATDP หมายถึงการใช้โหมด pluse ATH หมายถึงวางหู (Hang-up) และอื่นๆเพราะเหตุที่โมเด็มส่วนมากในท้องตลาดปัจจุบันนี้เป็นโมเด็มที่คอมแพตทิเบิลกับ Hayes ค่าต่างๆ ที่ PROCOMM PLUS กำหนดไว้โดยปริยายจึงคอมแพตทิเบิลกับ Hayes ด้วยแม้ว่าจะได้เปลี่ยนค่าต่างๆของ PROCOMM PLUS ให้เหมาะกับความต้องการของโมเด็มไปบ้างแล้วก็ตาม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3. ASCII

ASCII ย่อมาจาก "American Standard Code for Information Interchange"(รหัสมาตรฐานในการแลกเปลี่ยนข้อมูลของอเมริกา) รหัส ASCIIเป็นค่าตัวเลขหรับการอ่าน(หรือส่ง/รับ) แต่ละคาแรกเตอร์โดยคอมพิวเตอร์แต่คาแรกเตอร์จะกำหนดค่าเป็นจำนวนเลขไว้ให้เช่นตัวอักษรพิมพ์ใหญ่ A มีค่าเป็นตัวเลขฐานสิบเท่ากับ 65ในขณะที่ตัวอักษรพิมพ์เล็ก a มีค่าเป็น 97 ตัวอักษรพิมพ์เล็กทุกตัวจะมีค่ามากกว่าตัวอักษรพิมพ์ใหญ่ตัวเดียวกันอยู่ 32 ตัวคาแรกเตอร์ที่พิมพ์ได้ทุกตัว ได้แก่ ตัวเลข เครื่องหมายวรรคตอนเป็นต้นจะมีค่าตัวเลขกำหนดให้แต่ละตัว นอกจากนี้ ยังมีค่าตัวเลขสำหรับคาแรกเตอร์พิเศษซึ่งปกติจะเห็นได้ในจอภาพของคอมพิวเตอร์เหมือนกัน เช่น คาแรกเตอร์ควบคุม เป็นต้น

ปกติชุดคาแรกเตอร์ของ ASCII มีอยู่ 128 ตัวโดยมีค่าตั้งแต่ 0 ถึง 127โดยที่ยังมีคาแรกเตอร์ชุดอื่นๆให้ใช้อีกหลายชุดแต่ก็ไม่เป็นที่นิยมจนกระทั่งมีคอมพิวเตอร์ IBM PC เกิดขึ้น เมื่อคอมพิวเตอร์ส่วนบุคคลของ IBM ออกสู่ตลาดแล้ว IBM ได้กำหนดคาแรกเตอร์ ASCII ชุดพิเศษ (extended) ประกอบด้วยคาแรกเตอร์ต่างๆ 256 ตัว มีค่าตั้งแต่ 0 ถึง 255 คาแรกเตอร์ 128 ตัวที่มีค่าน้อย (0 ถึง 127)เหมือนกับชุดคาแรกเตอร์ ASCII มาตรฐานโดยที่อีก 128 ตัวที่มีค่าสูง (128 ถึง 255) ประกอบด้วยคาแรกเตอร์ของการเขียนเส้นตรง คาแรกเตอร์ทางคณิตศาสตร์ และคาแรกเตอร์ภาษาต่างประเทศ

4. บิตและไบต์

"บิต" คือหน่วยที่เล็กที่สุดที่ใช้เก็บข้อมูล ไว้ในหน่วยความจำของคอมพิวเตอร์ (หรือในดิสก์) อาจจะเปรียบเทียบบิตเดียวเหมือนสวิตช์ปิดหรือเปิดก็ได้ โดยปกติแล้ว บิตแทนค่าด้วยเลข 1 หรือ 0 ของการคำนวณเลขฐานสอง (การคำนวณแบบที่คอมพิวเตอร์ใช้อยู่) แต่ละคาแรกเตอร์ ตัวเลข คาแรกเตอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ควบคุม และข้อมูลพิเศษชิ้นเล็กๆ จะถูกเก็บไว้ในคอมพิวเตอร์เป็น "ไบต์" ซึ่งเป็นหน่วยประกอบด้วย 8 บิต บิตหนึ่งแทนค่าด้วย 1 หรือ 0 และไบต์หนึ่งจะเป็นแถวของเลข 1 หรือ 0 จำนวนแปดตัว (เช่น 10010010)

5. การส่งข้อมูลแบบอนุกรมและแบบขนาน

โดยอุดมคติแล้วจะทำการแลกเปลี่ยนข้อมูลกัน ระหว่างคอมพิวเตอร์กับคอมพิวเตอร์ หรือระหว่างคอมพิวเตอร์กับอุปกรณ์โดยรอบ (อย่างเช่น เครื่องพิมพ์) ในลักษณะที่เป็นแปดบิต หรือหนึ่งไบต์ต่อครั้ง การแลกเปลี่ยนข้อมูลดังกล่าวเรียกว่า *การส่งข้อมูลแบบขนาน* เนื่องจากส่งทั้งแปดบิตไปในขณะเดียวกันผ่านสายที่ขนานกันทั้งแปดสาย อย่างไรก็ดี ในสายโทรศัพท์ทำการส่งข้อมูลแบบขนานไม่ได้ เพราะสายโทรศัพท์มีเพียงสายเดียวในการส่งข้อมูลไปในทิศทางหนึ่ง และอีกสายหนึ่งในการส่งข้อมูลไปในอีกทิศทางหนึ่ง ดังนั้น ในการส่งข้อมูลของคอมพิวเตอร์ไปทางสายโทรศัพท์ (ซึ่งเป็นวิธีที่ใช้ติดต่อกับกับคอมพิวเตอร์เครื่องอื่นเป็นส่วนใหญ่) จึงต้องมีการปรับปรุงบ้างเล็กน้อย แต่ละไบต์จะต้องแตกออกเป็นบิตย่อยๆ และส่งบิตเหล่านั้นไปครั้งละหนึ่งบิต วิธีการส่งข้อมูลแบบนี้เรียกว่า *การส่งข้อมูลแบบอนุกรม* หรือ *การย้ายข้อมูลแบบอนุกรม*

ความยากในการส่งข้อมูลทีละบิต ที่อาจจะมีอยู่นั้นมองเห็นได้ชัดเจน เพราะในการส่งข้อมูลแบบขนาน ไบต์ใดเป็นไบต์เริ่มต้นและไบต์ใดเป็นไบต์จบนั้นเห็นได้ชัดเจน (ทั้งหมดมีหน่วยเป็นไบต์) แต่การส่งข้อมูลแบบอนุกรมมักจะเสี่ยงต่อการที่คอมพิวเตอร์รับข้อมูลไม่ทราบว่า ไบต์หนึ่งจบที่ไหนและอีกไบต์เริ่มต้นที่ไหน และหากมีการสูญหายไปแม้แต่บิตเดียวเนื่องจากความบกพร่องของสายโทรศัพท์แล้วจะเป็นผลให้ข้อมูลที่ได้รับหลังจากนั้นไปกลายเป็นขยะที่อ่านไม่รู้เรื่องได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

6. บิตเริ่มต้นและบิตหยุด

แม้ว่าปัญหานี้ส่วนใหญ่จะแก้ไขโดยใช้ บิตเริ่มต้นและบิตหยุด ก็ตามโมเด็มจึงต้องส่งบิตเริ่มต้น ซึ่งอยู่ข้างหน้าแต่ละไบต์เต็มๆ ออกไปเป็นสัญญาณไฟแรงต่ำที่จะบอกคอมพิวเตอร์รับข้อมูล (โดยผ่านโมเด็มของคอมพิวเตอร์นั้น) ว่าสิ่งที่ส่งตามนั้นเป็นไบต์ข้อมูล สำหรับที่ท้ายของไบต์นั้น โมเด็มที่ส่งข้อมูลจะมีสัญญาณไฟแรงต่ำอีกสัญญาณหนึ่ง ซึ่งคราวนี้เป็นบิตหยุด รวมอยู่ด้วยเพื่อแสดงให้ทราบว่าจบไบต์นั้นแล้ว

7. Parity (การตรวจสอบ)

เนื่องจากอาจจะมีข้อมูลที่เป็นขยะเกิดขึ้นได้ โดยเฉพาะในสายโทรศัพท์ที่มักจะเกิดการสูญหายของบิตไปแม้เพียงบิตเดียวก็ตาม จึงต้องมีวิธีการบางอย่าง ที่อย่างน้อยพอเป็นเครื่องหมายบอกความผิดพลาดที่น่าจะเป็นไปได้ เพื่อให้คอมพิวเตอร์ส่งข้อมูลทราบได้ว่าการรับข้อมูลนั้นไม่ถูกต้อง ปัญหาดังกล่าวนี้แก้ได้ง่ายที่สุดด้วยสิ่งที่เรียกว่า *parity (การตรวจสอบ)* การตรวจสอบนี้เป็นวิธีการตรวจสอบอย่างหยาบๆเพื่อประกันว่าการส่งข้อมูล แต่ละไบต์ถูกต้อง คาแรกเตอร์ของ ASCII ในชุดธรรมดา (128 ตัวแรก) แต่ละคาแรกเตอร์สามารถค่าได้ด้วย 7 บิตเท่านั้น ส่วนบิตที่ 8 สามารถ ละทิ้งเสียได้ หรือจะใช้เป็นบิตตรวจสอบ (parity bit) ก็ได้ถ้าใช้ดังกล่าวนี้บิตตรวจสอบจะแสดงให้ทราบคุณลักษณะของไบต์นั้น ถ้าคุณลักษณะและไบต์นั้นไม่ตรงกับด้านรับข้อมูล คอมพิวเตอร์รับข้อมูลจะตรวจพบว่าเกิดการส่งข้อมูลผิดพลาดขึ้นแล้ว โดยสรุป บิตตรวจสอบจะใช้เพื่อแสดงให้เห็น ว่าไบต์ที่ส่งไปนั้นมีเลข 1 เป็นเลขจำนวนคู่หรือคี่ เช่น ถ้าไบต์นั้นเป็น 1000001 (เลขฐานสองแทน A) ก็จะมีเลข 1 เพียงสองตัว แต่ถ้าคาแรกเตอร์ที่ส่งไปเป็น s แล้ว ไบต์นั้นจะเป็น 110001 ซึ่งเป็นไบต์ที่มีเลข 1 อยู่สามตัว ดังนั้นบิตตรวจสอบจึงใช้เพื่อแสดงว่าไบต์ที่ส่งไปมีเลข 1 เป็นจำนวนคู่หรือคี่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การตรวจสอบกำหนดได้หลายวิธีถ้ากำหนดการตรวจสอบเป็น "Even" (คู่) แล้ว บิตตรวจสอบจะกำหนดให้เป็นเลข 0 เพื่อให้แสดงว่ามีเลข 1 เป็นจำนวนคู่ และกำหนดให้เป็น 1 เพื่อแสดงว่ามีเลข 1 เป็นจำนวนคี่ ถ้ากำหนดการตรวจสอบเป็น "Odd" (คี่) แล้ว บิตตรวจสอบเป็น 0 จะแสดงให้เห็นว่ามีเลข 1 เป็นจำนวนคี่ และบิตตรวจสอบเป็น 1 จะแสดงว่ามีเลข 1 เป็นจำนวนคู่ เช่น เมื่อใช้ even parity ตัวอักษรนิมน์ใหญ่ A จะส่งไปว่า 01000001 ในขณะเดียวกันไบต์ที่ สอง a จะส่งไปว่า 11100001 จะใช้การตรวจสอบ (parity) นี้ต่อเมื่อส่งข้อมูลในแบบ 7 บิตเท่านั้น

ดังนี้แล้วจะก่อให้เกิดจุดที่มีความสำคัญขึ้น ในระบบปฏิบัติการ CP/M (ซึ่งการควบคุมคอมพิวเตอร์ส่วนบุคคลในสมัยแรกๆ เป็นส่วนมาก) จึงใช้เนียง 7 บิตเท่านั้นสำหรับแทนตัวคาแรกเตอร์ของ ASCII ที่มีอยู่ 128 ตัว บิตที่แปดมักจะไม่นำมาพิจารณาหรือใช้ในการตรวจสอบ อย่างไรก็ดีในปัจจุบันนี้ โดยที่ระบบปฏิบัติการเป็นดอส และมีชุดคาแรกเตอร์ของ ASCII เพิ่มขึ้นจึงต้องการทั้งหมด 8 บิต เพื่อแทนคาแรกเตอร์ของ ASCII ทั้ง 256 ตัว ดังนั้นการใช้ "no parity" กับข้อมูล 8 บิต จึงหมายความว่ามีการใช้ข้อมูล 8 บิตจริงๆ ซอฟต์แวร์เก่าๆ บางอย่างมักจะไม่พิจารณาบิตที่แปด ดังนั้นเมื่อใช้ no parity ในคอมพิวเตอร์ในขณะที่คอมพิวเตอร์ในระยะไกลส่งข้อมูลแบบ no parity จริงมา จึงไปด้วยกันได้ อย่างไรก็ดี PROCOMM PLUS ใช้ 8 บิต เต็มที ดังนั้นจึงอาจเกิดแถวตัวหนังสือรูปร่างแปลกๆ ในจอภาพก็ได้ ถ้าการทำงานด้วยข้อมูล 8 บิต และ no parity แล้วเกิดขยะดังกล่าวขึ้นในจอภาพ ให้ลองเปลี่ยนเป็นข้อมูลแบบ 7 บิต และ even parity

8. โพรโทคอล

มีวิธีการตรวจสอบความถูกต้องของการส่งข้อมูลที่เชื่อถือได้ดีกว่าอีกวิธีหนึ่ง เรียกว่า *โพรโทคอล* วิธีนี้ยุ่งยากกว่าการตรวจสอบด้วย parity

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ที่ง่ายกว่ามาก แต่ก็มีความเชื่อถือได้สูงกว่ามากเช่นกัน ดังนั้นวิธีจึงมีประโยชน์มาก โดยเฉพาะในการส่งโปรแกรมแทนที่จะส่งเพียงแค่ข้อความเท่านั้น ความผิดพลาดที่ปรากฏในข้อความจะตรวจหาและแก้ไขให้ถูกต้องได้ง่าย ถ้าพบคำว่า "dnd" ก็สามารถเดาได้ง่ายว่าที่ถูกแล้วควรจะเป็นอะไร และข้อความที่ปรากฏมี dnd อยู่ด้วย ก็จะทำให้ทราบได้ว่าคำที่ถูกต้องเป็นอะไรได้ชัดยิ่งขึ้น แต่ตรงกันข้าม ถ้ามีไบต์หนึ่งเกิดผิดพลาด เป็นขยะอยู่ในโปรแกรมที่แปรแล้ว (compiled) (เช่น เกม หรือโปรแกรมประมวลผลคำ) จะไม่สามารถหาที่ผิดพลาดได้ (อย่างน้อนที่สุดก็ไม่พร้อมที่จะหา) และจะเป็นเหตุให้รันโปรแกรมนั้นไม่ได้ หรือทำให้เกิดการผิดพลาดอย่างรุ่นแรงขึ้นได้

เพื่อหลีกเลี่ยงปัญหานี้ จึงมีการค้นคิดพัฒนาโพรโทคอลในการติดต่อสื่อสารขึ้นมาหลายอย่าง PROCOMM PLUS ก็สามารถใช้กับโพรโทคอลที่นิยมเหล่านั้นได้ โพรโทคอลทำงานโดยทำเครื่องหมายเริ่มต้นและจบให้กับกลุ่มข้อมูลขนาดใหญ่ๆ พร้อมทั้งคาแรกเตอร์นิเศษต่างๆ ที่สำคัญคือคอมพิวเตอร์รับข้อมูลจะส่งข้อความ ไปยังคอมพิวเตอร์ส่งข้อมูลว่า "เริ่มส่งข้อมูลได้แล้วตั้งแต่บัดนี้" เมื่อได้รับข้อความนี้แล้วคอมพิวเตอร์ส่งข้อความไปว่า "รับข้อมูลได้ทั้งหมดไหม?" ถ้าคอมพิวเตอร์รับข้อมูลตอบว่า "รับได้แล้ว" คอมพิวเตอร์ส่งขอ้มูลจะเตรียมข้อมูลกลุ่มต่อไปไว้และส่งข้อมูลนั้นด้วยวิธีเดียวกัน การทำงานของโพรโทคอลที่จริงแล้วจะไม่ใช่ที่กล่าวมานี้แต่เปรียบเทียบแล้วจะคล้ายๆ กัน อย่างน้อยที่สุดเพื่อแสดงให้เห็นว่าโพรโทคอลทำงานอย่างไร

มีข้อแตกต่างที่สำคัญอีกอย่างหนึ่ง ระหว่างการตรวจสอบด้วย parity กับการตรวจสอบด้วยโพรโทคอลคือ ในขณะที่ใช้การตรวจสอบด้วย parity นั้นไม่ต้องห่วงกังวนว่า จะเกิดอะไรขึ้นกับคอมพิวเตอร์อีกเครื่องหนึ่ง แต่ถ้าตรวจสอบด้วยโพรโทคอลแล้ว คอมพิวเตอร์รับและส่งข้อมูลจะต้องมีโพรโทคอล เดียวกันทั้งสองเครื่องจึงจะทำงานได้ ดังนั้นถ้าใช้โพรโทคอล Kermit ในการรับส่งข้อมูลแล้วทั้งคอมพิวเตอร์ของเราและคอมพิวเตอร์ที่เราติดต่อด้วย (มักเรียกว่า "คอมพิวเตอร์ในระยะไกล") จะต้องทำงานด้วยโพรโทคอล Kermit ไปพร้อมๆ กัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ในที่นี้ไม่จำเป็นต้องอธิบายรายละเอียดว่า โพรโทคอลทำงานอย่างไร Bulletin boad หรือคอมพิวเตอร์ระยะไกลส่วนมากแล้วจะกำหนดโพรโทคอลให้ใช้ และสามารถกำหนดให้ PROCOMM PLUS ทำตามไปได้ ถ้าเรายังมีความสนใจในทางเทคนิคของโพรโทคอล ก็มีหนังสือดีๆ มากมายในท้องตลาดที่หาซื้อมาศึกษาดูได้ แม้กระนั้นก็จำเป็นต้องทราบว่าโพรโทคอลใดบ้างที่ใช้กับ PROCOMM PLUS ได้ เพื่อที่อย่างน้อยจะได้ทราบข้อมูลเบื้องต้นที่จำเป็นต้องใช้

9. อัตราเร็วในการรับส่งข้อมูล

Baud rate คือความเร็วในการส่งบิตข้อมูล เป็นบิตข้อมูลต่อวินาที (ย่อว่า bps) หาได้โดยการหารจำนวนบิตเดียวของข้อมูลที่ส่งด้วยระยะเวลาที่ใช้ในการส่งบิตข้อมูลนั้น ดังนั้น baud rate 300 จึงหมายความว่าสามารถส่งบิตเดียวได้ภายใน 0.0033 วินาที หรือส่งข้อมูลไป 300 บิตในหนึ่งวินาที ในทำนองเดียวกัน baud rate 1200 แสดงว่าส่งได้ 1200 bps เนื่องจากในการส่งข้อมูลคาแรกเตอร์หนึ่งตัวใช้ 10 บิต (8 บิตข้อมูล 1 บิตเริ่มต้น และอีก 1 บิตหยุด) ดังนั้น 1200 bps จึงคำนวณออกมาได้เป็น 120 คาแรกเตอร์ต่อวินาที (cps) และ 300 bps ได้ 30 cps (มีข้อยกเว้นในกฎข้อนี้ที่ว่า baud rate กับ bps เหมือนกันนั้นไม่จริง แต่ในขณะนี้ไม่จำเป็นต้องกังวนเรื่องนี้แต่อย่างใด)

10. Duplex

คำว่า "full duplex" และ "half duplex" นี้ ผู้เชี่ยวชาญคอมพิวเตอร์กล่าวถึงเหมือนกับว่าทุกคนทราบความหมายอย่างท่องแท้แล้ว แต่ที่จริงแล้วอธิบายได้ง่ายมาก

half duplex แสดงว่าโมเด็มส่งหรือรับข้อมูลได้ แต่ได้อย่างเดียวเท่านั้นในเวลาที่กำหนดให้ ส่วน full duplex นั้นเหมือนโทรศัพท์ที่คนสองคน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สามารถพูดพร้อมๆ กันได้ หรืออย่างน้อย (เพื่อให้สุภาพขึ้น) คนหนึ่งจะเริ่มพูดได้เมื่ออีกคนหนึ่งพูดจบ โมเด็มแบบ full duplex จึงทำงานเหมือนโทรศัพท์ โมเด็มแบบนี้ยอมรับว่าเมื่อคอมพิวเตอร์เครื่องหนึ่งหยุดส่งข้อมูล อีกเครื่องหนึ่งจึงเริ่มส่งข้อมูลบ้าง

ในการติดต่อสื่อสาร duplex มักใช้ในการควบคุม หรือแสดงให้ทราบว่าใครควบคุมความรับผิดชอบในการว่งข้อมูลย้อนกลับเมื่อทำงานในแบบ full duplex คาแรกเตอร์ที่พิมพ์ลงไปจะถูกส่งไปยังเครื่องในระยะไกล ซึ่งจะส่งคาแรกเตอร์นั้นย้อนกลับมาให้เราตรงที่แสดงผลคาแรกเตอร์นั้นอยู่ เมื่อใช้ half duplex คาแรกเตอร์ที่พิมพ์ลงไปจะย้อนกลับอยู่ภายในซอฟต์แวร์ที่กำลังใช้อยู่ และคาแรกเตอร์ที่ส่งไปยังคอมพิวเตอร์ในระยะไกลจะไม่ย้อนกลับ

การใช้การตั้งโพรโทคอลผิดอาจทำให้เกิดปัญหาบางประการได้ ถ้าเห็นคาแรกเตอร์ซ้อนกัน สองตัวในจอภาพ (อย่างเช่น ddoouubblleecchhaarraacctteerrss) แล้วจะต้องตั้ง duplex เป็น full หรือหากว่าควรจะมีคาแรกเตอร์อยู่ในจอภาพแต่ปรากฏว่าไม่มีเลย ให้ตั้ง duplex เป็น half

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ICL7106 / 7107
3 1/2 – DIGIT / LED
SINGLE – CHIP ADC

รายละเอียดทั่วไป

IC 7106 และ IC 7107 เป็นไอซีที่ตอบสนองความต้องการได้สูง ใช้กำลังงานต่ำ ขนาด 3 1/2 DIGIT ซึ่งประกอบไปด้วย A/D CONVERTER, 7 SEGMENT DECODER, DISPLAY DIVER, ค่าแรงดันอ้างอิง และรูปคลื่นนาฬิกา IC 7106 ออกแบบไว้สำหรับการ INTERFACE กับ LIQUID CRITAL DISPLAY หรือ LCD โดยประกอบไปด้วย วงจรขับภายใน สำหรับ IC 7107 จะออกแบบไว้สำหรับการต่อร่วมกับเครื่องวัดต่างๆ หรือตัวกำเนิดแสง เช่น LED DISPLAY IC 7106 และ IC 7107 IC ทั้งสองนี้ จะให้ค่าความละเอียดในการวัดสูง ความกระทัดรัดในการใช้งาน และราคาถูกคุณสมบัติของ AUTO-ZERO มีค่าน้อยกว่า 10 μV, ZERO DRIFT มีค่าน้อยกว่า 1 μV/C ค่ากระแส BIAS ทางด้าน I/P มีค่า 10 PAmax ค่าความผิดพลาดทั่วไปมีค่าน้อยกว่า 1 ครั้งต่อการนับค่า DIFFERENTIAL I/P รวมทั้งค่าแรงดันและกระแสอ้างอิง ส่วนใหญ่จะเชื่อมโยงมาจากระบบการใช้งาน แต่ถ้าหาก ใช้งานร่วมกับ LOAD CELLS, STRAIN GAUGES หรือว่า TRANSDUCERS แบบอื่นๆ จะไม่นิยมใช้แหล่งจ่ายไฟเดียวกันกับระบบที่ใช้วัด ท้ายสุดจะให้คุณสมบัติทางด้านการประหยัด พลังงานในขณะที่ทำงาน (IC 7106) สำหรับอุปกรณ์ที่จะนำไปต่อร่วมด้วย (ในการวัด) เช่น หน้าปัดมิเตอร์ จะต้องต่อ PASSIVE ได้สูงสุด 10 ตัวเท่านั้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

แล้วนำผลที่ได้ไปแสดงค่า

คุณสมบัติต่างๆ

- ให้ค่าความเที่ยงตรงสูง เช่น ให้ค่าในการอ่านเท่า 0 องศา เมื่อค่า I/P เป็น 0 โวล์ท ทุกๆ SCALE
- ให้ค่าความเที่ยงตรงในการวัดสูง
- กระแส I/P มีค่าเท่ากับ 1 PA
- ให้ค่าความแตกต่างของ I/P รวมทั้ง ค่าแรงดันและค่ากระแสได้ถูกต้อง
- อุปกรณ์ภายใน BOARD ประกอบด้วย (ไม่รวมกับอุปกรณ์เพิ่มเติมภายนอก)

  LCD IC L7106

  LED IC L7107

- สัญญาณรบกวนต่ำ มีค่าน้อยกว่า 15 μV P-P วงจรสัญญาณนาฬิกาและค่าแรงดันอ้างอิงถูกบรรจุอยู่ภายใน
- ใช้กำลังงานต่ำ โดยมีค่าต่ำกว่า 10 mW
- ไม่จำเป็นต้องมีวงจรภายนอกเพิ่มเติม
- เป็นผลิตภัณฑ์ใหม่ที่มีจำหน่ายทั่วไปในตลาด
- ชุด KID มีจำหน่ายทั่วไปเช่นกัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Characteristics	Test 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} = 100mV	999	999/1000	1000	Digital Reading
Rollover Error (Difference in reading for equal positive and negative inputs near Full Scale)	$-V_{IN} = +V_{IN} \cong$ 200.0mV	−1	±.2	+1	Counts
Linearity (Max. deviation from best straight line fit)	Full scale = 200.0mV or full scale = 2.000V (Note 6)	−1	±.2	+1	Counts
Common Mode Rejection Ratio (Note 4)	V_{CM} = ±1V, V_{IN} = 0V Full Scale = 200.0mV		50		μV/V
Noise (Pk-Pk value not exceeded 95% of time)	V_{IN} = 0V Full Scale = 200.0mV		15		μV
Leakage Current Input	V_{IN} = 0 (Note 6)		1	10	pA
Zero Reading Drift	V_{IN} = 0 0° < T_A < 70°C (Note 6)		0.2	1	μV/°C
Scale Factor Temperature Coefficient	V_{IN} = 199.0mV 0° < T_A < 70°C (Ext. Ref. Oppm/°C) (Note 6)		1	5	ppm/°C
V^+ Supply Current (Does not include LED current for 7107)	V_{IN} = 0		0.8	1.8	mA
V^- Supply Current (7107 only)			0.6	1.8	mA
Analog Common Voltage (With respect to Pos. Supply)	25kΩ between Common & Pos. Supply	2.4	2.8	3.2	V
Temp. Coeff. of Analog Common (With respect to Pos. Supply)	25kΩ between Common & Pos. Supply		80		ppm/°C

อัตราค่าสูงสุดสมบูรณ์ (ABSOLUTE MAXIMUM RATINGS)

Supply.Voltage

IC L7106, V + to V - ..... 15 V

IC L7107, V + to GND ..... + 6 V

IC L7107, V - to GND ..... - 9 V

Analog Input Voltage (either input) (Note 1) ... ..

.....v + to v -

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Reference Input Voltage (either input) .....

.....V + to V -

Clock Input

IC L7106 ..... TEST to V +

IC L7107 ..... GND to V +

Power Dissipation (Note 2)

Ceramic Package ..... 1000 mW

Plastic Package ..... 800 mW

Operating Temperature ..... 0 C to + 70 C

Storage Temperature ..... - 65 C to + 150 C

Lead Temperature (Soldering, 10 sec) ..... 300 C

หมายเหตุ จากข้อมูลต่างๆ ของอัตราสูงสุดสมบูรณ์ ดังได้กล่าวแล้วถ้าหากใช้ความถี่ต่ำหรือสูงกว่าที่กำหนดไว้จะทำให้อุปกรณ์เสียหายได้ จากข้อมูลดังกล่าวเป็นการบอกถึงคุณประโยชน์สูงสุดในการทำงานของอุปกรณ์เท่านั้น ส่วนรายละเอียดเพิ่มเติมอื่น ได้จะกล่าวถึงอีกต่อไปซึ่งความเสียหาย เกิดขึ้นกับอุปกรณ์ จะมีผลมาจากการใช้งานเกินกว่าอัตราที่ได้กำหนดไว้ ตามอัตราสูงสุดสมบูรณ์ ซึ่งทำให้อุปกรณ์ขาดความเชื่อถือ

หมายเหตุ 1 ค่าแรงดันด้าน I/P จะจ่ายค่าแรงดันและ ให้กระแสออกมาซึ่งค่ากระแสนี้จะถูกจำกัดไว้ที่ +/- 100 μA

หมายเหตุ 2 อัตราการสูญเสียกำลังงานอาจเกิดขึ้นได้ ถ้าหากต่ออุปกรณ์เข้ากับ Board หรือรอยบัดกรีไม่ดี

หมายเหตุ 3 รายละเอียดดังกล่าวสามารถใช้ได้ทั้ง IC L7106 และIC L7107

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Characteristics	Test Conditions	Min	Typ	Max	Unit
7106 ONLY Pk-Pk Segment Drive Voltage Pk-Pk Backplane Drive Voltage (Note 5)	V^+ to V^- = 9V	4	5	6	V
7107 ONLY Segment Sinking Current (Except Pin 19 & 20)	V^+ = 5.0V Segment voltage = 3V	5	8.0		mA
(Pin 19 only) (Pin 20 only)		10 4	16 7		mA

รูปแสดง คุณสมบัติทางไฟฟ้า

ภายใต้อุณหภูมิ 25 C ความถี่ของสัญญาณนาฬิกา 48 KH_z สำหรับการทดสอบ IC L7106 และ IC L7107 ดูได้จากวงจรทดสอบทั้ง 2

หมายเหตุ 4 กล่าวถึง Differanctial I/P

หมายเหตุ 5 วงจรขับภายในจะมี Phase ตรงกันกับ วงจรขับ Segment เมื่อ Segment ดับอยู่และต่าง Phase 180 องศา เมื่อ Segment ทำงาน และอัตราการเปลี่ยนแปลงของความถี่ 20 ครั้งต่อวินาที ส่วนค่าแรงดันไฟตรงสำหรับอุปกรณ์บน Board ใช้น้อยกว่า 50 mV

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หมายเหตุ 6 ไม่มีการทดสอบ เพราะอุปกรณ์ทุกอย่างได้ตามมาตรฐานกำหนด

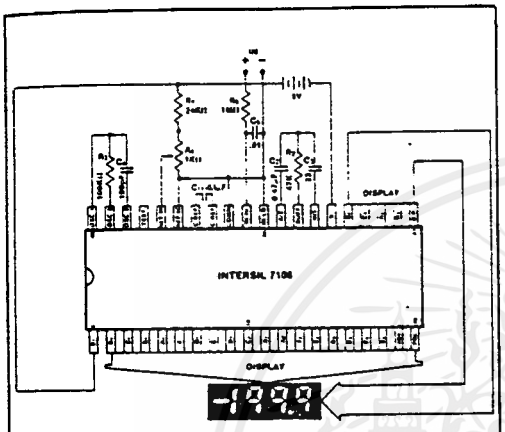


Figure 2: ICL7106 Test Circuit and Typical Application With Liquid Crystal Display

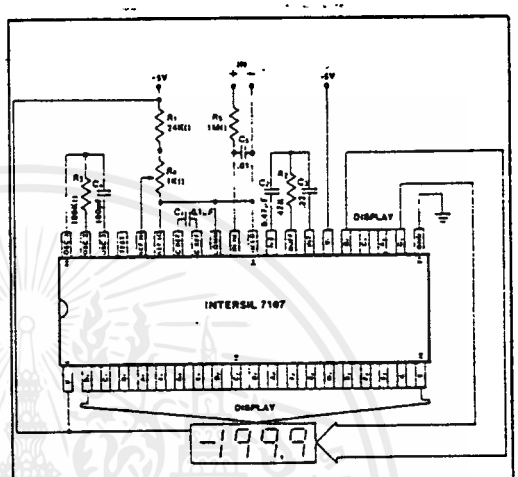


Figure 3: ICL7107 Test Circuit and Typical Application With LED Display

วงจรทดสอบ

รายละเอียดภายใน

ส่วนของ ANALOG

ในรูปที่ 4 แสดงถึงส่วนที่เป็น Analog ของ ICL7106 และ 7107 ในการวัดค่าในแต่ละครั้งจะถูกแบ่งออกเป็น 3 ส่วน ดังนี้คือ

1. Auto-zero(A/Z)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

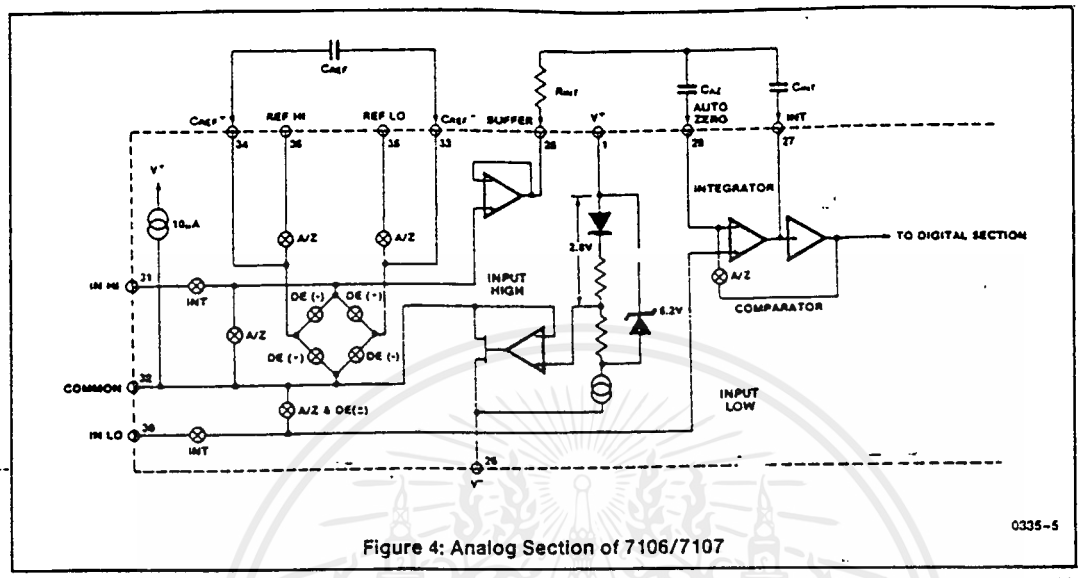


Figure 4: Analog Section of 7106/7107

รูปแสดง วงจรส่วนของ Analog

2. Signal integrate (INT)

3. de-integrate (DE)

หน้าที่ของ Auto-Zero (CAZ)

ในระหว่างที่ทำกระบวนการของ Auto-zero จะปรากฏเหตุการณ์ 3 อย่างขึ้นดังนี้คือ ;อันดับแรก input light และ LOW จะไม่ต่ออยู่กับขาของ IC เหมือนสภาวะปกติ แต่จะไปต่อเข้ากับ Analog Common, อันดับที่ 2 ; Capacitor อ้างอิงจะจ่ายประจุให้แก่ Voltage อ้างอิง และขั้นที่ 3 ชุดของ feedback จะถูกต่อเข้ากับระบบ เพื่อจะ Charge ประจุให้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กับตัวเก็บประจุของ Auto-zero (CAZ) เพื่อชดเชยค่าแรงดัน offset ของ buffer Amplifier ,integrator และ comparator; ตราบใดที่วงจร Lomparator ยังคงต่ออยู่กับระบบแล้วก็มีเพียงสาเหตุนี้เท่านั้น ที่จะทำให้วงจรมีความแม่นยำในการวัดลดลงไปบ้าง เนื่องจากสัญญานรบกวนของวงจรนี้ ในทำนองอื่นแล้ว ค่าของ offset หมายถึงค่าของ input นั่นเอง ซึ่งโดยทั่วไปแล้วมีค่าน้อยกว่า 10 μV

หน้าที่ของ Signal Integrate

ในขณะที่ Signal integrate หรือ Auto-zeroloop ถูกเปิดไว้ค่าภายในจะเปลี่ยนแปลงโดยการเคลื่อนย้ายที่อยู่ใหม่และค่า input ภายในซึ่งเป็นค่า high และ low กว้างได้ ซึ่งเริ่มจาก 1 volt จนกระทั่งไปเป็นค่าอื่นๆ ของแหล่งจ่ายไฟหรือหากสัญญาณทางด้าน input ไม่มีการจ่ายไฟกลับมายังส่วนของ converter power supply แล้ว IN LO จะทำหน้าที่ต่อตัวเองเข้ามายัง Analog Common ให้อยู่ในสภาวะที่ถูกต้องและเหมาะสม.

หน้าที่ของ De-Integrate

หน้าที่ปราการสุดท้ายของระบบคือชุดหรือค่าอ้างอิง Integrate ค่าของ input LOW จะต่ออยู่ภายในกับ Analog common และ Input high จะต่ออยู่กับตัวเก็บประจุอ้างอิงการต่อหรือประกอบค่าตัวเก็บประจุบน Chip ต้องแน่ใจมาก่อนแล้วว่าเป็นการใส่ขั้วที่ถูกต้องในกรณีที่การให้ค่าที่เป็น ศูนย์ออกมาในกรณีต้องเปลี่ยนค่าเป็นศูนย์การแสดงผล ในการอ่านค่าเท่ากับ 1000 (Vin/VREF)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ค่าความแตกต่างทางด้าน INPUT

ทางด้าน INPUT ยอมให้ ค่าความแตกต่างของ VOLTAGE จากแหล่งจ่ายไฟไหนๆก็ได้ ภายในย่านการขยายของ INPUT AMPLIFIER หรือจากรายละเอียดได้กำหนดค่าไว้ที่ 0.5 VOLTE ซึ่งมีค่าต่ำกว่าค่าของแหล่งจ่ายไฟบวกถึง 1 VOLTE เหมือนค่าของแหล่งจ่ายไฟลบ ในย่านนี้ระบบ CMRR ถึง 86 dB ถึงอย่างไรก็ตาม ส่วนที่น่าจะให้ความสนใจก็เห็นจะได้แก่ การที่ทำให้ผลที่ได้จาก OUTPUT ไม่ให้มีการอิ่มตัว และการที่ระบบอาจเกิดผลเสียหายได้อีกประการหนึ่งคือ การใช้ค่า VOLTAGE ที่มีขนาดมากเกินไป เช่น มีค่าเป็นบวกและเป็นลบมากเกินไป สัญญาณทางด้าน INPUT ที่มีค่าเป็นลบจะเป็นตัวขับผลรวมของค่าบวก ถ้าหากเกิดมีปัญหาในแง่ของการแกว่งของสัญญาณ สำหรับการใช้งานในสภาวะวิกฤติ ผลของ OUTPUT เกิดการแกว่งของสัญญาณ สามารถลดการแกว่งลงได้เต็มที่ 2 VOLTES ซึ่งทำให้ค่าความเที่ยงตรงในการวัดมีผลกระทบน้อยมาก

ค่าความแตกต่างอ้างอิง

ค่า VOLTAGE อ้างอิงได้จากทุกๆที่ ภายในแหล่งจ่ายไฟของ CONVERTER แหล่งจ่ายไฟหลักของระบบ ที่เช็คค่าความผิดพลาดเห็นจะได้แก่ ROLL-OVER VOLTAGE ซึ่งสืบเนื่องมาจาก ตัวเก็บประจุอ้างอิง หรือผลของตัวเก็บประจุแบบ STRAY ถ้าหากในระบบเกิดมีขนาดของ VOLTAGE มากเกิน ไปตัวเก็ยประจุอ้างอิงจะทำการเก็บประจุ แต่ถ้าหากระบบนี้ต้องการทำงานในส่วนของ DE-INTEGRATE ตัวเก็บประจุอ้างอิงจะทำการจ่ายประจุให้กับระบบด้วย ค่าคว่ามแตกต่างของการอ้างอิงสำหรับ ค่าอินพุตที่ที่มีทั้งไฟบวกและลบนี้ จะเป็นการเช็คค่าความผิดผลาดแบบ ROLL-OVER อย่างไรก็ดีควรนิจารณาเลือกตัวเก็บประอ้างอิง ที่มีค่าการเก็บประจุมากกว่า อย่างไรก็ดี ควรนิจารณาเลือกตัวเก็บ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

STAY ซึ่งจะทำให้รักษาสภาวะของการวัดค่าความผิดพลาดไม่น้อยกว่าของการวัดค่าความผิดพลาดได้ไม่น้อยกว่า 0.5 ครั้ง ในสภาวะผิดปกติ

ANALOG COMMON

ขาของ ANALOG COMMON ประกอบไปด้วยชุดจ่ายแรงดันให้กับแบตเตอรี่ (7106) หรือระบบอื่นๆ โดยทั่วไป ขาคอมมอนนี้จะใช้ค่าแรงดันมีค่าโดยประมาณเท่ากับ 2.8 โวลท์ ซึ่งให้ค่าลบมากกว่าค่าบวก ANALOG COMMON จะทำงานเมื่อมีสภาวะคล้ายกับ Auto-zero และ De-integrate ถ้า IN LO มีค่าแตกต่างจาก Analog common ค่าแรงดันของ COMMON MODE ที่อยู่ภายใน system จะทำหน้าที่รักษาสภาพแรงดันให้คงที่ โดยใช้ CMRR ของ converter แต่ในบางกรณี IN LO ก็จะถูกกำหนดไว้ที่ค่าคงที่ตายตัว แต่ในที่นี้ Analog Common จะถูกต่อ ณ.จุดเดียวกัน และใช้แรงดันจาก converter

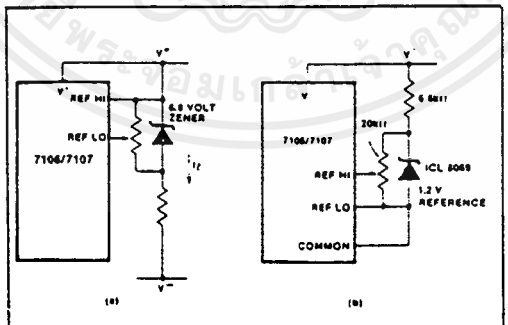


Figure 5: Using an External Reference

ภายใน IC ส่วนของ Analog Common ต่ออยู่กับ N channel

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ของ FETใช้ค่ากระแสประมาณ 30 mA และสภาพแรงดันมีค่าเท่ากับ 2.8 volt โดยจะมีค่าต่ำกว่าค่าของแหล่งจ่ายไฟบวกอย่างไรก็ดี กระแสที่ใช้จากแหล่งจ่ายกระแสใช้เพียง 20 μ A เท่านั้น

การทดสอบ

ขา TEST จะถูกใช้งานได้ใน 2 ส่วน สำหรับ ICL7106 แล้วจะใช้เพื่อเป็นตัวสร้างแหล่งจ่ายไฟแบบ DIGITAL โดยผ่านมาทาง R 500 OHMS หรือในอีกหรือก็จะใช้เพื่อเป็น แหล่งจ่ายไฟลบ เพื่อจ่ายให้กับอุปกรณ์ ภายนอก เช่น seven-segment หรือการแสดงผลแบบ LCD.

ในรูปที่ 6 และ 7 จะแสดงถึงการใช้งานในกรณีต่างๆกัน โดยที่ load ต้องใช้กระแสไม่ควรเกิน1 mA

สำหรับการใช้งานในลักษณะที่ 2 คือ การทดสอบโดยการแสดงผลทางหลอดไฟ XX (LAMP TEST)เมื่อการ TEST มีค่าแรงดันเป็นบวก (pull

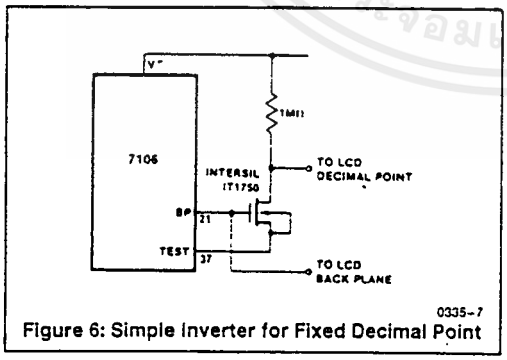


Figure 6: Simple Inverter for Fixed Decimal Point

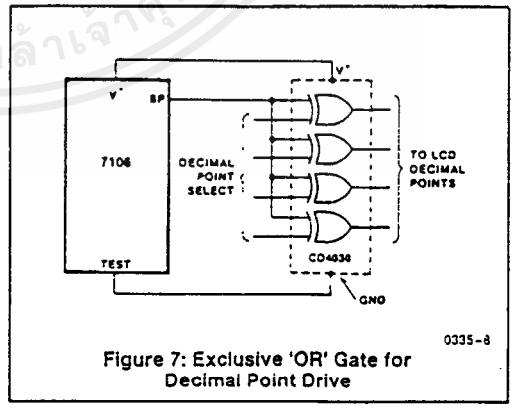


Figure 7: Exclusive 'OR' Gate for Decimal Point Drive

รูปแสดง วงจรทดสอบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

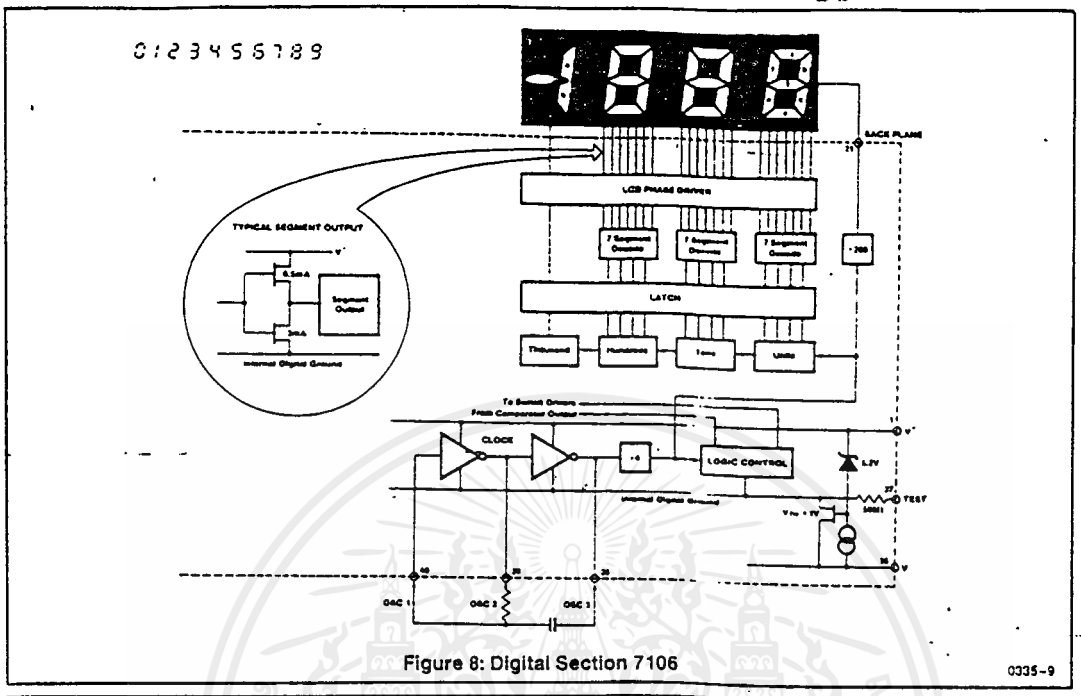


Figure 8: Digital Section 7106

0335-9

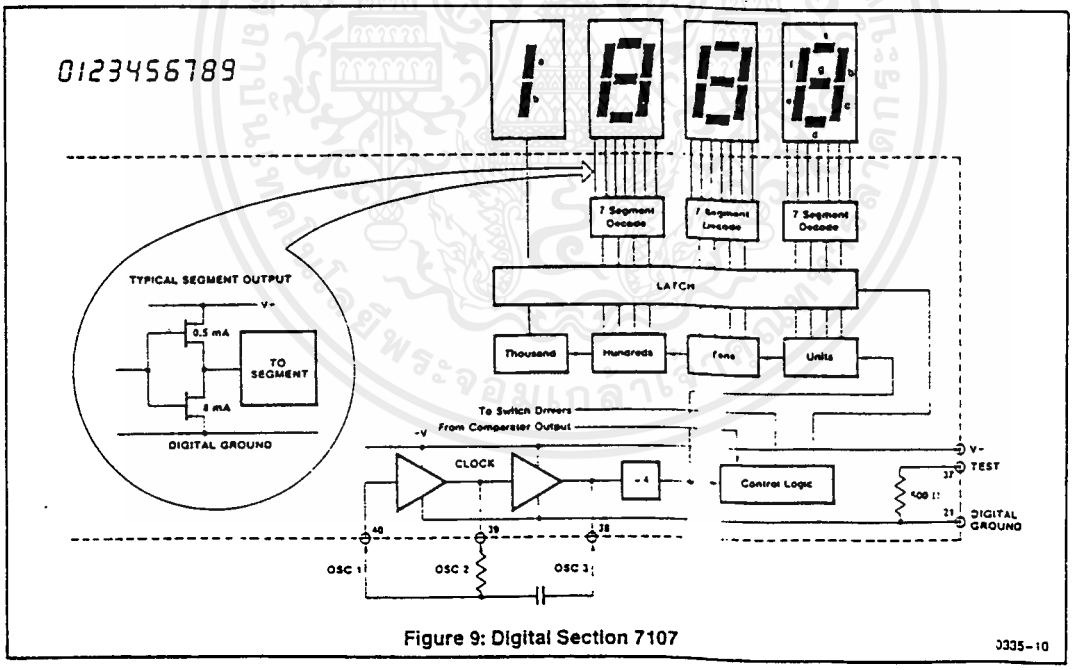


Figure 9: Digital Section 7107

0335-10

รูปแสดง วงจรแสดงผลของ 7106

และ 7107

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

high) Segment ทุก ๆ Segment จะทำงานและให้ค่า -1888 โดยขา TEST จะใช้กระแส 10 mA เมื่ออยู่ในสภาวะนี้

คำเตือน สำหรับ ICL7106 ใน Mode ของ LAMP TEST, Segment จะมีค่า Dc-volt (ซึ่งไม่ใช่ค่าพัลซ์) และจะทำให้ LCD ไหม้เสียหายได้

ระบบคาบเวลา

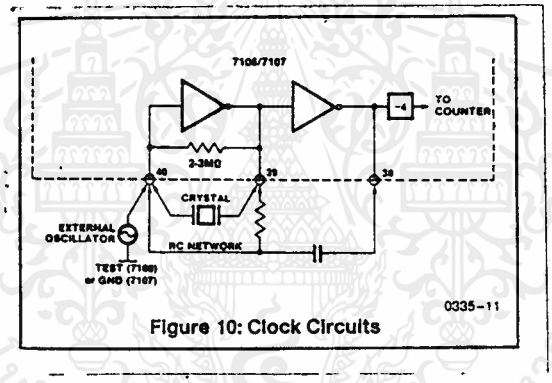


Figure 10: Clock Circuits

รูปแสดง วงจรสร้างสัญญาณนาฬิกา

รูปที่ 10 จะแสดงถึงวงจรกำเนิดสัญญาณนาฬิกาที่ใช้ในICL7106 และ 7107 โดยตั้งอยู่บนบรรทัดฐาน 3 ประการดังนี้

1. ค่าของสัญญาณนาฬิกาภายนอกจะต่ออยู่ที่ขา 40
2. Crystal จะต่ออยู่ระหว่างขา 39 และ 40
3. ค่า R-C oscillator จะต่ออยู่ทั้ง 3 ขาคือ 38,39 และ 40

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การเลือกค่าของอุปกรณ์ต่างๆ

Integrating Regisistor

ทั้ง Buffer และ Integrator จะใช้ขยายแบบ Class A. โดยใช้กระแส output 100 μA

ตัวเก็บประจุ Integrating

ตัวเก็บประจุ Integrating จะเลือกมาเพื่อใช้งานสำหรับให้ค่า การเปลี่ยนแปลงของแรงดันสูงสุดโดยค่าเปลี่ยนแปลงของแรงดันมีค่าประมาณ 0.3 Volt จากแหล่งจ่ายไฟทั่วไปส่วนเพิ่มเติมตัวเก็บประจุแบบ Integrating จะต้องมีการดูดกลืนของ Dielectric มีค่าต่ำเพื่อป้องกันค่าความผิดพลาดในการวัด

ตัวเก็บประจุแบบ Auto-Zero

ขนาดของตัวเก็บประจุ Auto-Zero ขนาดของตัวเก็บประจุแบบ Auto-Zero นี้เป็นตัวกำจัดสัญญาณรบกวนระบบ โดยมีขนาดเท่ากับ 200 mV Full Scale มีค่าเท่ากับ 0.47 μF ถ้าใช้ขนาด 2 Vlote จะมีค่า0.047 μF ซึ่งเป็นตัวลดขนาดสัญญาณรบกวนของระบบ

ตัวเก็บประจุอ้างอิง

จะให้ผลดีเมื่อเลือกใช้ที่ค่า 0.1 μF และ 200 mV

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ส่วนประกอบของ Oscillator

ทุกๆ ย่านความถี่ต้องใช้ค่าความต้านทานขนาด 100 กิโลโอห์ม และหาค่า C ได้จากสมการดังนี้

$$F = 0.45 / RC$$

เมื่อ $F = 48\ KH_z$ (การอ่านค่า 3 ครั้งต่อวินาที)

$$C = 100\ PF$$

POWER SUPPLY ของ ICL7107

IC เบอร์ 7107 ออกแบบไว้ใช้งานในย่านแรงดัน +/- 5 Vlote อย่างไรก็ตาม ถ้าหากไม่มี Voltage ค่าลบพอที่จะจ่ายให้ได้ สามารถกำเนิดค่า Voltage จาก Clock Output โดยใช้ DIODE 2 ตัว และ Capacitor 2 ตัว ดังรูป

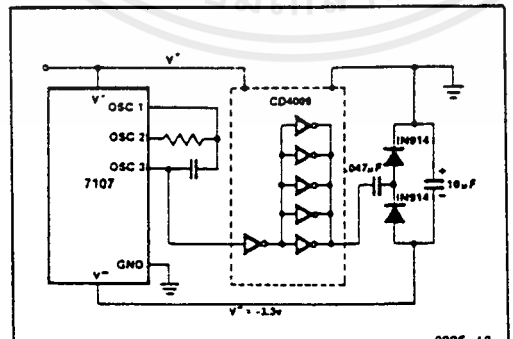


Figure 11: Generating Negative Supply from +5V

รูปแสดง วงจรให้กำเนิดสัญญาณไฟลบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ในความเป็นจริงแล้ว ไม่ควรเลือกใช้ค่า Voltage Supply ที่มีค่าเป็นลบ และเงื่อนไขต่างๆ ของการใช้ + 5 Volte ของระบบ ดังนี้

1. Input Signal สามารถอ้างอิงจากย่าน Common Mode ของ Converter

2. ค่าของสัญญาณน้อยกว่า +/- 1.5 Volte

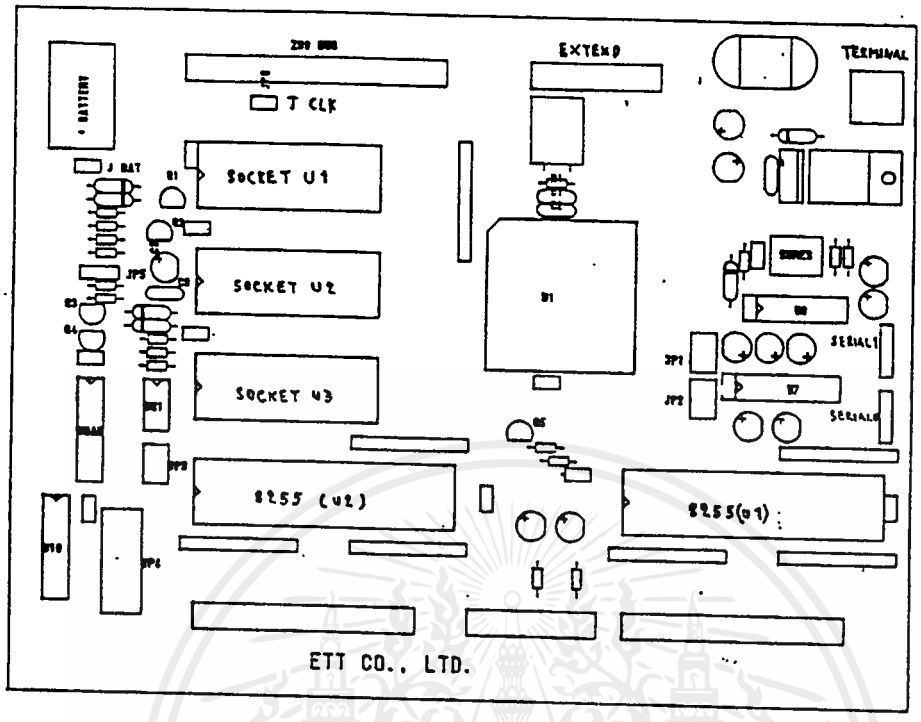
3. ค่าแรงดันอ้างอิง จากภายนอกมีความจำเป็นต้องใช้ด้วย

การใช้งานแบบต่างๆ

IC เบอร์ 7106 และ 7107 สามารถนำไปประยุกต์ใช้งานได้กว้างขวาง ดังเช่น วงจรต่างๆดังต่อไปนี้ เป็นเพียงตัวอย่างการนำไปใช้งานของ A/D Converter หรือ ขึ้นอยู่กับการประยุกต์ของผู้ใช้เอง

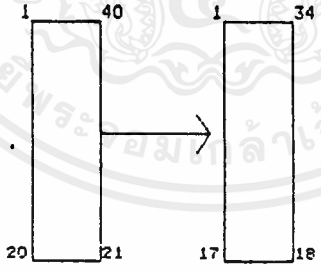
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาพแสดงลักษณะ CP-180



จุดที่ SCREEN ผิดบน BOARD

1) I/O COUNECTOR 1 และ 2 ต้องเป็น 34P
และที่ SCREEN ใน PRINT CONNECTOR I/O 1,2 จะสลับกัน



2) LED ของ 8255 จะสลับกับความเป็นจริงคือ ที่ SCREEN PC71 เป็น PC72 และ PC72 เป็น PC71

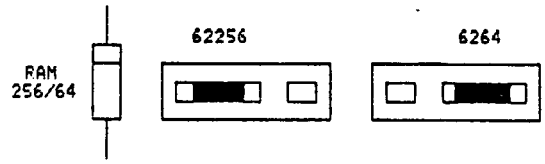
-2-

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

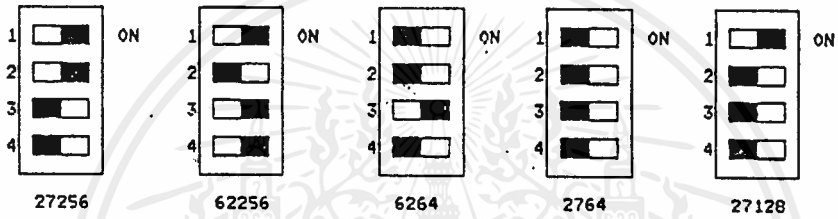
ข้อกำหนดบน BOARD

SOCKET 1 ใช้กับ ROM หรือ EPROM 27256 อย่างเดียว

SOCKET 2 ใช้กับ RAM 6264 (8K) หรือ 62256 (32K) โดยการเลือก JUMPER (JP5)



SOCKET 3 สามารถใส่ IC ได้หลายเบอร์ โดยการ SET DIP SW 4 P ดังรูป



JP SP อยู่ใกล้กับ Q5 และ LED PC72 ซึ่งใช้ต่อกับลำโพง

JP RES ใกล้กับ SWITCH RESET สำหรับต่อ SWITCH RESET ไปที่อื่น

JP PAGE (JP 3) ใกล้ DIP SW 4 PIN ใช้สำหรับเลือก PAGE ของ PORT ว่าให้อยู่ใน DECODE แบบ 256 PORT หรือ 64 K PORT โดยมีรูปแบบดังนี้ :-

8255 ทั้ง 2 ตัวบน BOARD จะอ้างที่ 0C0XXH

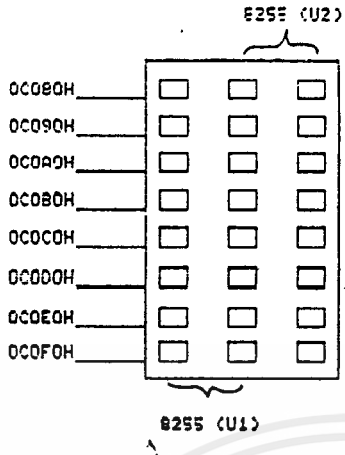
8255 ทั้ง 2 ตัวบน BOARD จะอ้างที่ XCH - XFH

ซึ่งค่า X นั้นยังขึ้นอยู่กับการเลือก JUMPER PORT อีกครั้งหนึ่ง

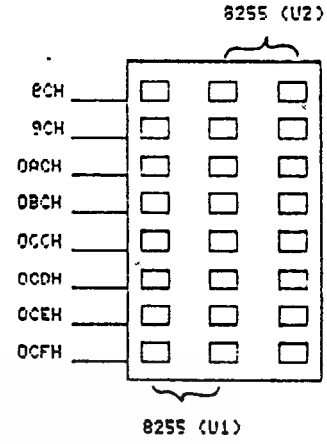
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

JP PORT (JP 4) ใช้เลือก NUMBER PORT โดยสามารถเลือกได้ดังรูป

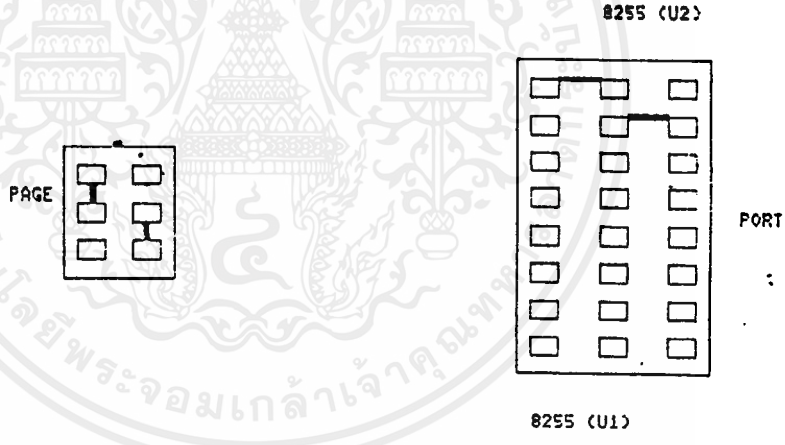
เมื่อเลือก PAGE เป็น 0C0XXH



เมื่อเลือก PAGE เป็น XCH - XFH

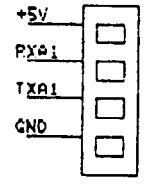


ซึ่ง BOARD สำเร็จที่ทำขึ้น จะถูกกำหนดเป็น 0C080H ของ 8255 (U1) และ 0C090H ของ 8255 (2) ซึ่งเป็นของ PRINTER PORT ด้วย ซึ่งผู้ใช้สามารถเปลี่ยนแปลงได้ แต่เมื่อใช้กับ SOFT WARE DEBUG ต้องกำหนดตามนี้

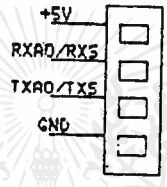


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

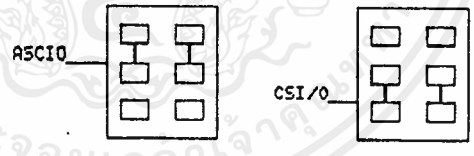
SERIAL 1 เป็น CONNECTOR SERIAL PORT ASCI CHANEL 1 โดยขา สัญญาณจะเป็นดังรูป :-



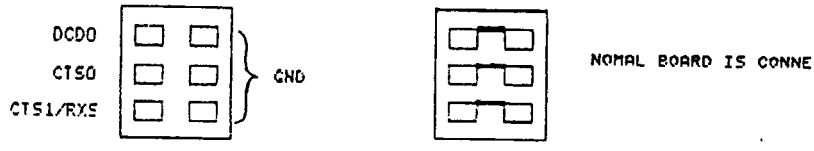
SERIAL 0 เป็น CONNECTOR SERIAL PORT ของ ASCI CHANEL 0 และยังเลือกได้ว่าให้เป็นขั้วต่อของ CLOCK SERIAL I/O ได้ โดยการ SET JUMPER ที่ ASCI/CSIO



JP ASCI/CSIO (JP1) ใช้สำหรับเลือกว่าต้องการใช้ SERIAL PORT CHANEL 0 หรือ CLOCK SERIAL I/O โดย SET ดังรูป และเมื่อใช้ CSI/O ต้องนำ JUMPER CTS1/RXS ใน JP MODEM ออกด้วย



JP MODEM (JP2) ใช้สำหรับเลือกว่าต้องการนำสัญญาณ MODEM ไปใช้ หรือไม่สำหรับการต่อ SERIAL PORT ใช้เองในกรณีสัญญาณ ครบปกติบน BOARD จะถูก JUMP ไว้ให้



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

JP BAT อยู่ใกล้กับ BATTERY (J BAT) โดยเมื่อใส่ BATTERY แล้ว JUMP ที่ JUMPER นี้ จะเป็นการ BACK UP RAM ของ SOCKET U2

JP CLK เป็น JUMPER เลือก CLOCK ว่าให้ CLOCK จาก CPU ออกไปที่ BUS Z80 40 PIN. ประโยชน์ เมื่อต้องการนำ BOARD ไปแทนระบบ Z80 เดิม เพื่อไม่ให้ CLOCK ชนกัน

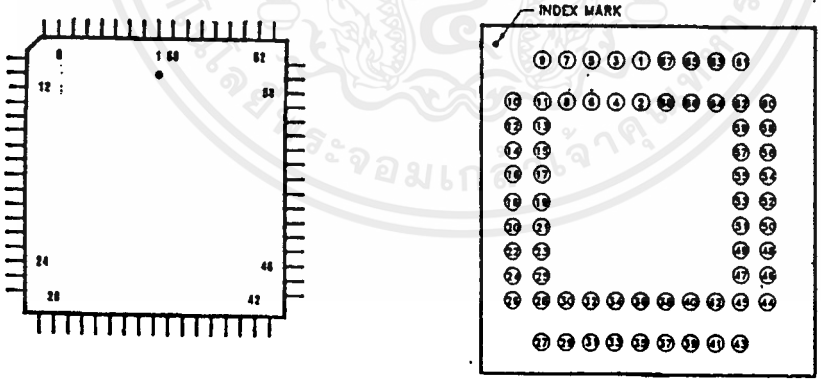
TERMINAL 2 P สำหรับต่อ SUPPLY DC 5 V ให้กับ BOARD โดยตรง

CONNECTOR Z80 BUS สำหรับ INTERFACE กับอุปกรณ์ภายนอก

CONNECTOR EXTEND เป็นส่วนขาสัญญาณที่เพิ่มจาก Z80 สำหรับ INTERFACE กับอุปกรณ์ภายนอก

<u>คำแนะนำในการประกอบ</u>

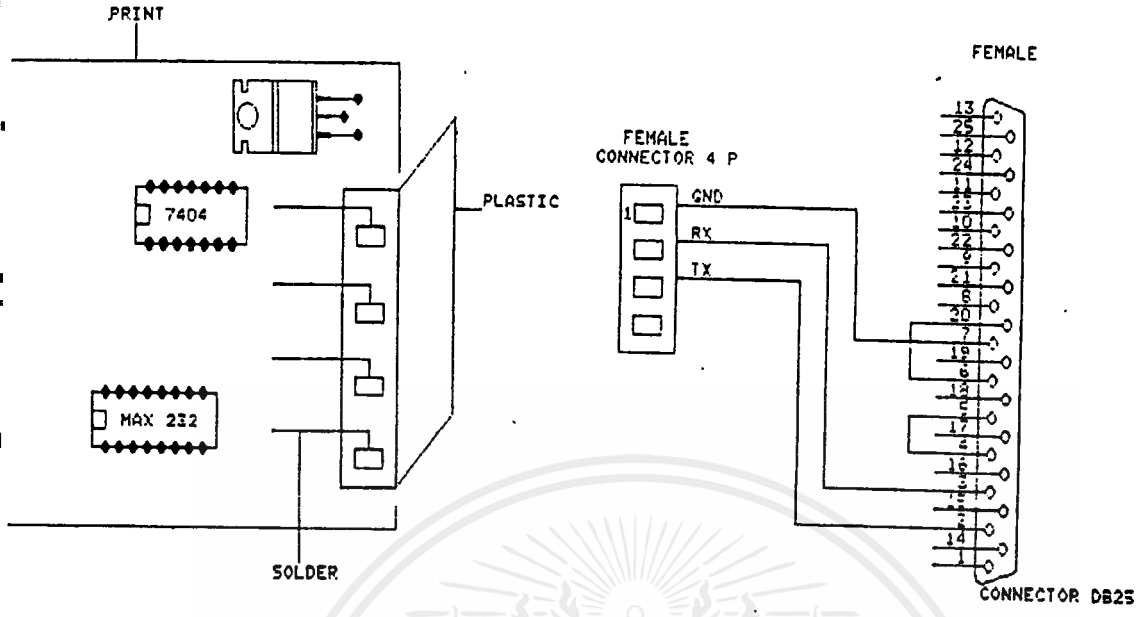
1) ควรใส่อุปกรณ์ตามลำดับ คือ ตัวต้านทาน , ไดโอด , ตัวเก็บประจุ , SOCKET
2) ตัวเก็บประจุและไดโอดต้องใส่ให้ถูกขั้วรวมทั้ง TRANSISTOR ด้วย
3) LED ขายาว คือ A ขาสั้น คือ K
4) การใส่ SOCKET 68 PIN PLCC ให้ดูรอยบากริมซ้ายจะเป็นตัวแสดงว่าเป็นด้านบนที่จะใส่ IC ด้านขา 1 และเมื่อจะใส่ IC Z80180 ก็ให้ดูรอยบากเช่นเดียวกัน และใส่ให้ตรงกับ SOCKET โดยตัว IC จะมีจุดวงกลมตรงกลางแสดงถึงตำแหน่งขา 1 ของ IC ดังรูป :-



การนับขา SOCKET 68P PLCC (TOP VIEW)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5) การใส่ขั้ว CONNECTOR 4 PIN ของ SERIAL PORT ทั้ง 2 CHANEL ต้องหันด้านที่มี พลาสติก ให้อยู่ชิดแผ่น PRINT และด้านที่เป็น ขั้วทองแดง ให้หันเข้ามาด้านใน และการต่อสาย SERIAL PORT กับ PC เป็นดังรูป

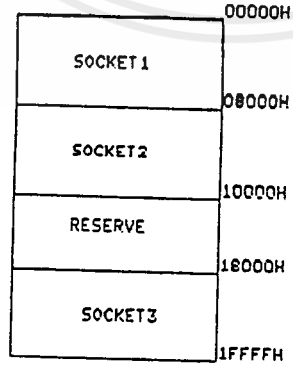


5) การบัดกรีให้กระทำเฉพาะด้านล่างเพียงด้านเดียวก็พอ เพราะ PRINT เป็นแบบ PLATE THROUGH HOLES

6) การติดตั้งและบัดกรี REGULATOR IC ให้ดูดังรูป :-

7) เมื่อทุกอย่างเรียบร้อยแล้วจึงค่อยใส่ IC ลงใน SOCKET ระวังอย่าให้ผิดเบอร์โดยเฉพาะ Z80180 ให้ดูขาในการใส่ให้ดี

รายละเอียดการจัดหน่วยความจำ



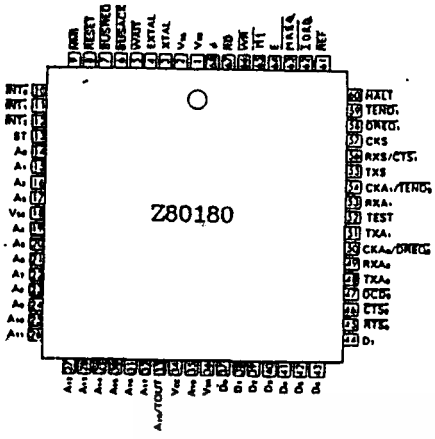
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

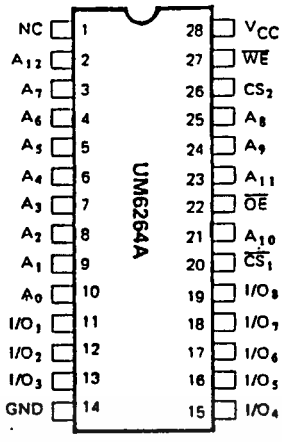
Specification

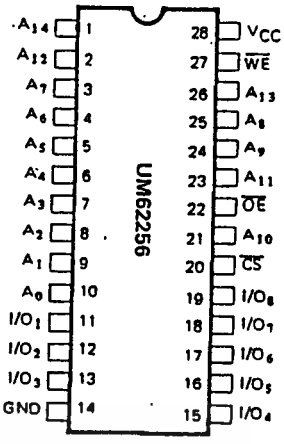
CPU	Z80180
Memory	27256 Rom (32 Kbyte at 00000H) 6264/62256 (8 Kbyte on board,expand 32 kbyte) & Back up 27256/27128/2764/6264/62256 Socket expand (Max 32 K byte)
Port	8255 I/O Port 48 Bit
Printer Port	1 Port
Serial Port	2 Chanel RS 232 & 1 Chanel can select clock serial I/O
Clock Rate	6.144 Mhz
Power supply	Comsumption 5V DC & Terminal 5V DC Main Input 9-12 V DC
Connector	1 40-Pin Expansion Header-Strip (Z80 Pin) 1 20-Pin Extend (Z80180 Pin) Header-Strip for (Modem,DMA,Expand memory 1Mbyte) 2 34-Pin Peripheral Header-Strip (8255 Port) 1 24-Pin Seclect Decode port 1 6-Pin Select Page of Port 1 6-Pin Select Modem control 1 2-Pin for battery back up 1 2-Pin Clock Select for connector old Z80 system 1 2-Pin Jumper SW reset 1 2-Pin Speaker
LED	1 Power Red LED 2 PC7 Port 8255*2 Orange LED 1 Halt Green LED
PCB Size	13 * 17.5 cm

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

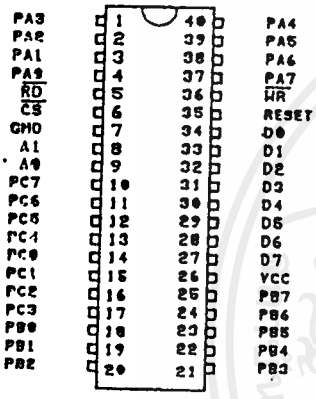
รายละเอียดขา IC

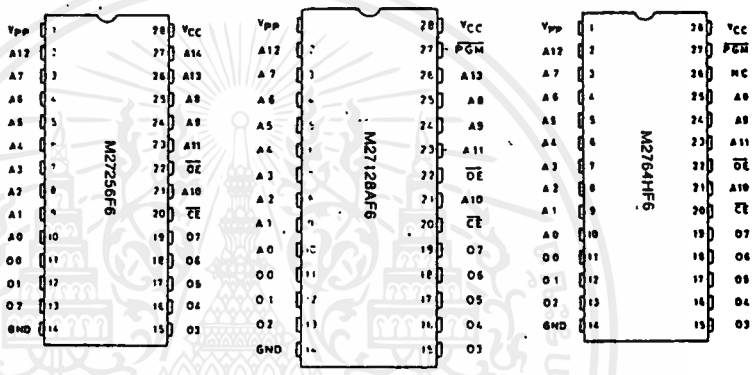


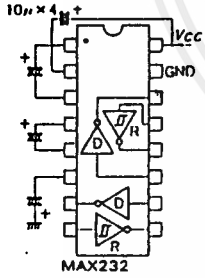


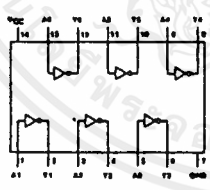


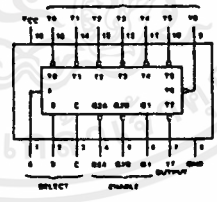
8255 PPI

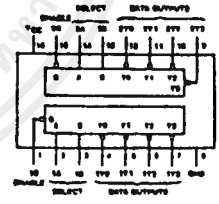








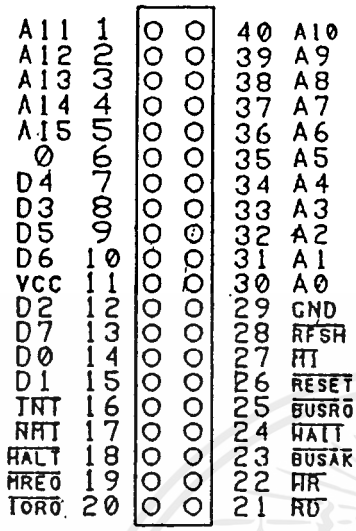




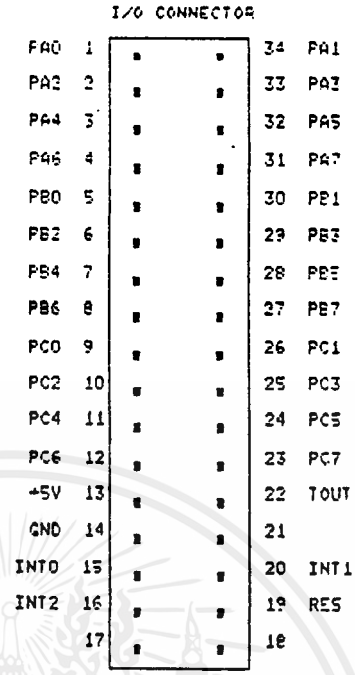
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รายละเอียดขา CONNECTOR

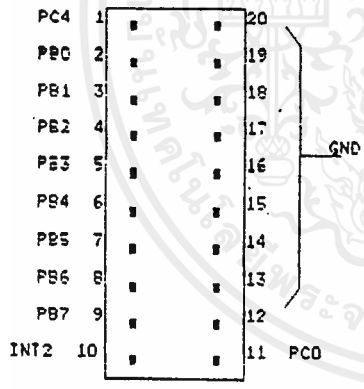
Z80 CPU CONNECTOR



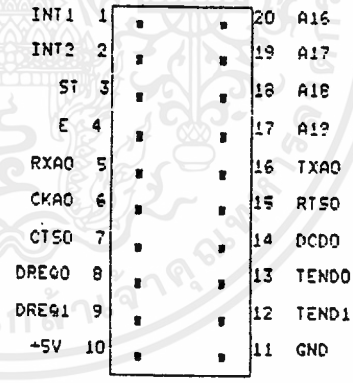
I/O CONNECTOR



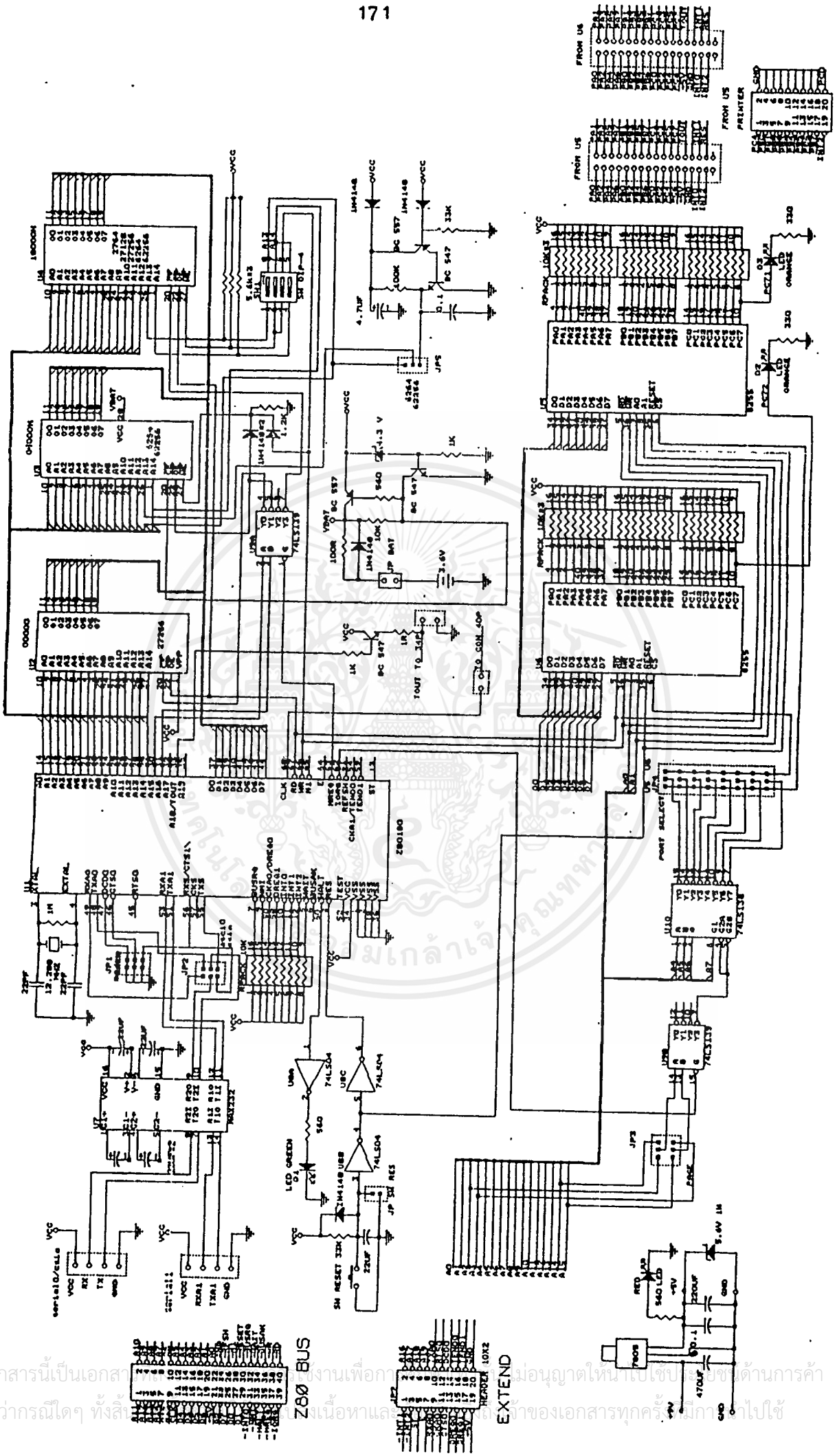
PRINTER



EXTEND



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งไม่อนุญาตให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รายการอุปกรณ์

ชุด IC

1	ตัว	Z80180 CPU	2	ตัว	8255 I/O PORT
1	ตัว	6264 RAM	1	ตัว	74LS04
1	ตัว	74LS138	1	ตัว	MAX232
1	ตัว	74LS139			

ชุด SOCKET

1	ตัว	68 PIN (PLCC)	3	ตัว	16 PIN
3	ตัว	28 PIN	1	ตัว	14 PIN
2	ตัว	40 PIN			

ชุด REGISTOR

7	ตัว	10K RPACK	2	ตัว	1K
3	ตัว	5.6K	1	ตัว	1.2K
1	ตัว	1	1	ตัว	100K
2	ตัว	33K	3	ตัว	560
2	ตัว	330	1	ตัว	10K
1	ตัว	100	1	ตัว	1M

ชุด CONDENSER

1	ตัว	470 UF/16V	5	ตัว	22 UF/16V
1	ตัว	220/16V	1	ตัว	4.7/16V แทนทาลั่ม
12	ตัว	0.1 แทนทาลั่ม	2	ตัว	22 PF

ชุด HEADERSTRIP

1	ตัว	40 PIN	2	ตัว	2 PIN แถวเดี่ยว
2	ตัว	34 PIN	1	ตัว	3 PIN แถวเดี่ยว
2	ตัว	20 PIN	2	ตัว	6 PIN
1	ตัว	16 PIN			
1	ตัว	16 PIN แถวเดี่ยว			

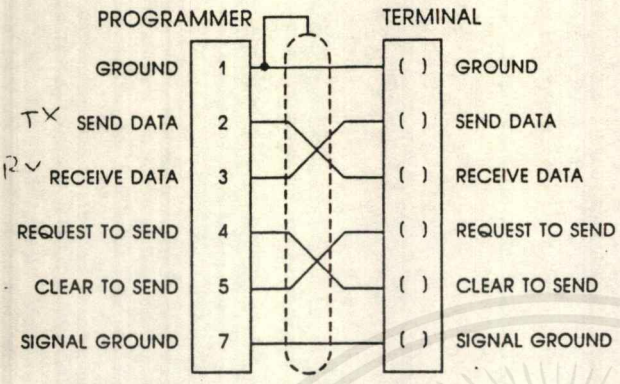
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ชุดเบ็ดเตล็ด

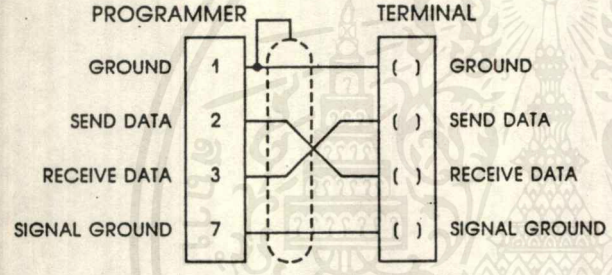
6	ตัว	IN4148	1	ตัว	7805
1	ตัว	ZENER 5.6 V 1 W	1	ตัว	HEAT SINK
1	ตัว	ZENER 4.3 V	1	ตัว	JACK ADAPTER
2	ตัว	BC 557	1	ตัว	TERMINAL 2 P
3	ตัว	BC 547	1	ตัว	DIPSWITCH 4 P
1	ตัว	X'TAL 12.288 MHz	11	ตัว	JUMPER 2 P
2	ตัว	CONNECTOR 2 P	2	ตัว	CONNECTOR 4 P
1	ตัว	MINISWITCH	4	ตัว	LED (แดง 1 , เขียว 1 , ส้ม 2)
1	แผ่น	PRINT CP-180	1	เล่ม	SHEET การประกอบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

RS232 Connection



Half/Full Duplex, With Handshake

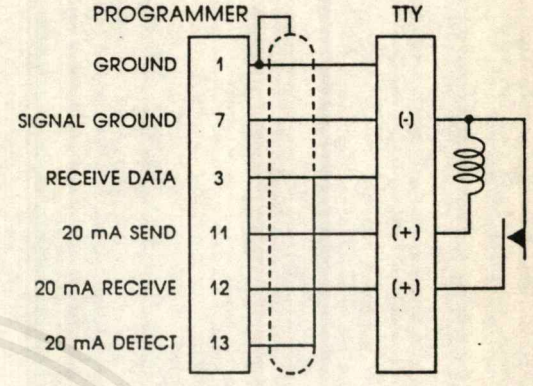


Half/Full Duplex, Without Handshake

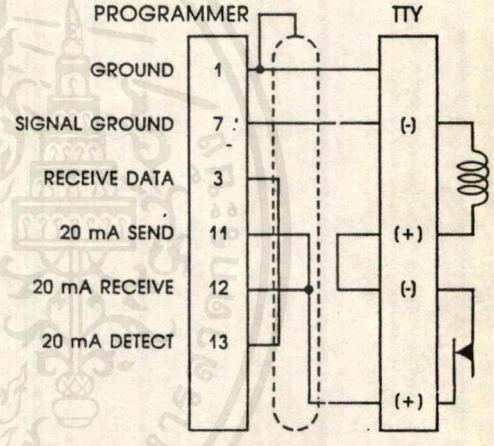
NOTES

1. *All signals are named with respect to the originating unit.*
2. *All undesignated pins are to be left open.*
3. *For applications that do not require handshaking, the programmer's clear to send line is pulled up internally.*
4. *Host system's pin numbers may differ.*

20 mA Current Loop Connection



Full Duplex



Half Duplex

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ส่วนของ โปรแกรม

SOFTWARE PART

- ASSEMBLY
- TURBU C++

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
R
                ;*****************************************;
                ;*              MAIN PROGRAM             *;
                ;*****************************************;


        .ORG    8000H
        .EQU    TX,82H          ;DSP DATA ON SCN
        .EQU    RX,81H          ;RECV DATA FRM KEBD
        .EQU    CTRL_PORT,9BH   ;PORT A,B,C = INPUT
        .EQU    PORTU,92H      ;U2 A,B = I/P C =  O/P
        .EQU    PORTA,08CH
        .EQU    ESC,01BH        ;QUIT PROGRAM
        .EQU    PORTB,08DH
        .EQU    PORTC2,09EH     ;PORT C U2
        .EQU    PORTC,08EH
        .EQU    CTRL,08FH       ;CONTROL PORT U1
        .EQU    CTRL2,9FH       ;CONTROL PORT U2
        .EQU    PORTA2,9CH      ;PORT A (U2)
        .EQU    TEMP,54H        ;54=T ALPHABET ASII.
        .EQU    SWITCH,53H      ;53=SWITCH CODE
        .EQU    LTEMP,74H       ;74=t
        .EQU    LSWITCH,73H     ;73=s
        .EQU    DECIMAL,0FFEBH
        .EQU    HEXIMAL,0FFE8H
        .EQU    C_PORT,02H      ;COUNTER PORT B,C

         LD     A,CTRL_PORT     ;LOAD PORT CONTROL
         OUT    (CTRL),A        ;SET A B C = INPUT PORT
         LD     A,PORTU         ;LOAD PORT CONTROL a,b = input ,c = o/p
         OUT    (CTRL2),A       ;
         LD     A,00H
         OUT    (PORTC2),A
         LD     HL,LED
         LD     (HL),A
         LD     A,02H
         LD     (DIGIT),A
         LD     A,PORTB
         LD     (PORTIN),A

LOOP1:

         LD     DE,0000H        ;USE BY LOOP
         LD     HL,LED         ;ADDR LED FLASHING
         LD     A,(HL)
         BIT    7,A             ;TEST LAMP
         JP     Z,ON            ;IF LED IS OFF
         RES    7,A             ;SET LED OFF
         JP     ON1
ON:      SET    7,A             ;SET LED ON
ON1:     LD     (HL),A
         OUT    (PORTC2),A

    ;***********************;
    ;         END           ;
    ;***********************;

         LD A,RX        ;LD RX COMMAND INTO A
         RST 10H        ;RX SUNBROUTINE
         LD A,C         ;Instore data frm C into A
         CP TEMP        ;Compare alphabet ASCII T.
         JP Z,LOOP2     ;IF COMMAND=TEMP THEN JUMP TO LOOP2
```

เอกสารนี้เป็นเอกสารสงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        CP LTEMP
        JP Z,LOOP2
        CP SWITCH       ;Compare switch code
        JP Z,LOOP3
        CP LSWITCH
        JP Z,LOOP3
DELAY1: JP LOOP1        ;WAIT COMMAND

LOOP2:
        CALL  ON_COM        ;ON RS-232
        CALL  Z,MAIN        ;If key=T then escape.
        CALL  WAITING
        JP    LOOP1         ;Back to loop 1 if uncommands.

LOOP3:
        CALL ON_COM
        CALL SEND_SWITCH
        CALL WAITING
        JP LOOP1


ON_COM: LD    HL,LED
        LD    A,(HL)
        SET   0,A             ;ON COMMUNICATION
        LD    (HL),A
        OUT   (PORTC2),A
        RET

WAITING: LD   DE,0260H
LP:     DEC   DE
        LD    A,D
        OR    A
        JP    NZ,LP
        LD    HL,LED
        LD    A,(HL)
        RES   0,A
        LD    (HL),A
        RET


MAIN:   CALL  BUFFER          ;CLEAR ADDRESS BUFFER
        LD    IX,CH_DEC       ;LOAD CHANGE DECIMAL TO REG.IX
        LD    A,C_PORT
        LD    (DIGIT),A
INPUT:  LD    A,(PORTIN)      ;LOAD PORTIN TO REG.A
        LD    C,A             ;LOAD REG.A TO REG.C
        IN    A,(C)           ;LOAD TEN DIGITS TO REG.A
        CALL  DECCONV         ;CONVERT TO DECIMAL
        LD    A,B             ;TRANSFER DECIMAL VALUE TO REG.A
        LD    (IX+0),A        ;SAVE DECIMAL VALUE TO ADDRESS FFE8H
        INC   IX              ;INCREMENT ADDRESS IX FOR NEXT DECIMAL VALUE
        LD    A,(DIGIT)       ;LOAD COUNTER PORT TO REG.A
        DEC   A               ;DECREMENT COUNTER FOR COUNT PORT
        LD    (DIGIT),A       ;SAVE COUNTER PORT VALUE
        LD    A,(PORTIN)      ;LOAD PORT NUMBER TO REG.A
        INC   A               ;INCREMENT NEXT PORT
        LD    (PORTIN),A      ;SAVE PORT NUMBER
        INC   HL              ;INCREMENT DECIMAL BUFFER
        LD    A,(DIGIT)       ;CHECK COUNTER
        OR    A               ;IF END OF PORT?
        JP    Z,INPUT1        ;IF END OF DATA CONVERT TO HEXCIMAL
        JP    INPUT           ;JUMP TO NEXT PORT
INPUT1: LD    IX,CH_DEC       ;RESERVE TEMPORARY DECIMAL
        LD    A,(IX+1)        ;LOAD HIGH BYTE TO REG.A
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        LD      B,(IX+0)        ;LOAD LOW BYTE TO REG.B
        SLA     A               ;SHIFT LOW BYTE TO HIGH BYTE WITH ONE DIGIT
        SLA     A               ;SHIFT LOW BYTE TO HIGH BYTE WITH ONE DIGIT
        SLA     A               ;SHIFT LOW BYTE TO HIGH BYTE WITH ONE DIGIT
        SLA     A               ;SHIFT LOW BYTE TO HIGH BYTE WITH ONE DIGIT
        OR      B               ;ADD LOW + HIGH BYTE FOR SEND HEXIMAL DATA
        LD      (DECIMAL),A     ;KEEP THE DECIMAL DATA
        LD      A,PORTB         ;LOAD PORTB TO REG.A
        LD      (PORTIN),A      ;SET PORTB FOR BEGINING
        IN      A,(PORTA)       ;
        CALL    DECCONV         ;CONVERT VALUE IN REG.A TO DECIMAL
        LD      A,B             ;LOAD DECIMAL VALUE IN REG.B TO REG.A
        SLA     A               ;SHIFT LOW BYTE TO HIGH BYTE WITH ONE DIGIT
        SLA     A               ;SHIFT LOW BYTE TO HIGH BYTE WITH ONE DIGIT
        SLA     A               ;SHIFT LOW BYTE TO HIGH BYTE WITH ONE DIGIT
        SLA     A               ;SHIFT LOW BYTE TO HIGH BYTE WITH ONE DIGIT
        LD      IX,SH_VALUE     ;LOAD POINTER (SHOW VALUE) TO REG.IX
        LD      (IX+1),A        ;LOAD DECIMAL OF 10'DIGIT TO REG.(IX+1)
        LD      A,(DECIMAL)     ;LOAD DECIMAL OF 100'DIGIT TO REG.A
        LD      (IX+0),A        ;LOAD DECIMAL OF 100'DIGIT IN REG.A TO REG.(IX
        CALL    HEXCONV         ;CONVERT TO HEXCIMAL
        LD      IX,SH_VALUE     ;LOAD POINTER DECIMAL
        LD      A,(IX+0)        ;LOAD DECIMAL VALUE (SECOND DIGIT) TO REG.A
        LD      C,A             ;LOAD DECIMAL VALUE IN REG.A TO REG.C
        CALL    SEND_HEX        ;CALL SUBROUTINE SEND HEXCIMAL DATA
        LD      A,02EH          ;LOAD SYMBOL '.' TO REG.A
        LD      C,A             ;LOAD SYMBOL '.' IN REG.A TO REG.C
        CALL    SEND_ASCII      ;CAL SUBROUTINE SEND ASCII DATA
        LD      IX,SH_VALUE     ;LOAD POINTER DECIMAL
        LD      A,(IX+1)        ;LOAD DECIMAL VALUE (FIRST DIGIT) TO REG.A
        LD      C,A             ;LOAD DECIMAL VALUE IN REG.A TO REG.C
        CALL    SEND_HEX        ;CALL SUBROUTINE SEND HEXCIMAL DATA
        RET

        ;*****************************************;
        ;*    CLEAR ADDRESS BUFFER FFE8H-FFEBH   *;
        ;*****************************************;

BUFFER: LD      B,06H           ;SET COUNTER FOR CLEAR 8 ADDRESS
        LD      HL,HEXIMAL      ;LOAD ADDRESS TO CLEAR
        LD      A,00H           ;LOAD A=00H
  LOOP: LD      (HL),A          ;LOAD ADDRESS HL = 00H
        INC     HL              ;INCREMENT HL=HL+1
        DEC     B               ;DECREMENT B=B+1
        JP      NZ,LOOP         ;JUMP TO LOOP IF B<>0
        RET                     ;RETURN TO CALL


        ;*****************************************;
        ;*      CONVERT DECIMAL TO HEXCIMAL      *;
        ;*****************************************;

HEXCONV: LD     A,8DH           ;SUBROUTINE  DTOH
         RST    10H             ;END SUBROUTINE
         RET                    ;RETURN TO CALL


        ;*****************************************;
        ;* CONVERT SEGMENT TO DECIMAL ( 3 DIGITS )*;
        ;*****************************************;

DECCONV: XOR    0FFH        ;CONVERT FROM 0 TO 1
         AND    7FH         ;SET BIT 8 = 0
         LD     C,A         ;LOAD REG.A TO REG.C
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        LD    B,00H       ;SET COUNTER
        LD    HL,TABLE    ;LOAD DATA TABLE FOR REFFERENT

COUNT:  LD    A,C         ;LOAD REG.C TO REG.A
        CP    (HL)        ;COMPARE REG.A WHIT DATA TABLE
        JP    Z,NUM       ;JUMP TO 'NUM'
        INC   HL          ;INCREMENT HL=HL+1
        INC   B           ;INCREMENT B=B+1
        LD    A,B         ;LOAD REG.B TO REG.A
        CP    09H         ;IF NON INPUT VALUE (A)
        JP    NC,ERROR     ;THEN JUMP TO 'ERROR'
        JP    COUNT       ;JUMP TO COUNT FOR OTHER DIGIT

NUM:    RET

        ;*****************************************;
        ;*       SEND ASCII DATA ON SCREEN       *;
        ;*****************************************;

SEND_ASCII:
        LD    A,082H      ;SUBROUTINE TXBYTE
        RST   10H         ;END SUBROUTINE
        RET               ;RETURN TO CALL

        ;*****************************************;
        ;*     SEND HEXCIMAL DATA ON SCREEN      *;
        ;*****************************************;

SEND_HEX:
        LD    A,083H      ;SUBROUTINE RXBYTE
        RST   10H         ;END SUBROUTINE
        RET               ;RETURN TO CALL
END:     RST   18H        ;QUIT PROGRAM

        ;*****************************************;
        ;*              SEND SWITCH              *;
        ;*****************************************;

SEND_SWITCH:
        IN    A,(PORTA2)  ;RECEIVE THE SWITCH PORT
        AND   07H         ;CLEAR BIT 2-7 TO 00H
        LD    C,A         ;LOAD A TO C REGISTER
        CALL  SEND_HEX    ;SEND SWITCH CODE TO SERIES PORT
        RET               ;RETURN TO MAIN

      ;*************************************;
      ;*  IF TEMP OVER 99 DEC. FIX IT TO 99 *;
      ;*************************************;

ERROR:  LD    B,09H           ;FIX  IT TO 9
        RET

TABLE:  .DB   3FH   ; 3F (SEGMENT) = 0 (DEC NUMBER)
        .DB   06H   ; 06 (SEGMENT) = 1 (DEC NUMBER)
        .DB   5BH   ; 5B (SEGMENT) = 2 (DEC NUMBER)
        .DB   4FH   ; 4F (SEGMENT) = 3 (DEC NUMBER)
        .DB   66H   ; 66 (SEGMENT) = 4 (DEC NUMBER)
        .DB   6DH   ; 6D (SEGMENT) = 5 (DEC NUMBER)
        .DB   7DH   ; 7D (SEGMENT) = 6 (DEC NUMBER)
        .DB   07H   ; 07 (SEGMENT) = 7 (DEC NUMBER)
        .DB   7FH   ; 7F (SEGMENT) = 8 (DEC NUMBER)
        .DB   6FH   ; 6F (SEGMENT) = 9 (DEC NUMBER)
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                .ORG 0EFFEH
          .DB    96H             ;BUARD RATE
          .DB    0A3H            ;AUTO RUN
                .ORG 0F000H

DIGIT:    .DB    00H             ;RESERVE FOR COUNTER PORT
LED:      .DB    00H             ;STATUS LED
PORTIN:   .DB    00              ;SET START AT PORT 'B'
CH_DEC:   .DB    00H,00H         ;RESERVE FOR DECIMAL DATA
SH_VALUE: .DB    00H,00H        ;RESERVE FOR SHOW VALUE ON SCREEN
          .END
```

B:\>

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                ;*****************************************;
                ;*                MAIN PROGRAM             *;
                ;*****************************************;


        .ORG    8000H
        .EQU    TX,82H          ;DSP DATA ON SCN
        .EQU    RX,81H          ;RECV DATA FRM KEBD
        .EQU    CTRL_PORT,9BH   ;PORT A,B,C = INPUT
        .EQU    PORTU,92H      ;U2 A,B = I/P C =  O/P
        .EQU    PORTC2,09EH     ;PORT C U2
        .EQU    CTRL2,9FH       ;CONTROL PORT U2
        .EQU    PORTA2,9CH      ;PORT A (U2)
        .EQU    SWITCH2,55H     ;SWITCH BD.2 "U"

        LD      A,PORTU         ;LOAD PORT CONTROL a,b = input ,c = o/p
        OUT     (CTRL2),A       ;
        LD      A,00H
        OUT     (PORTC2),A
        LD      HL,LED
        LD      (HL),A

LOOP1:

        LD    DE,0000H          ;USE BY LOOP
        LD    HL,LED           ;ADDR LED FLASHING
        LD    A,(HL)
        BIT   7,A              ;TEST LAMP
        JP    Z,ON             ;IF LED IS OFF
        RES   7,A              ;SET LED OFF
        JP    ON1
ON:     SET   7,A              ;SET LED ON
ON1:    LD    (HL),A
        OUT   (PORTC2),A

    ;***********************;
    ;         END           ;
    ;***********************;

        LD A,RX         ;LD RX COMMAND INTO A
        RST 10H         ;RX SUNBROUTINE
        LD A,C          ;Instore data frm C into A
        CP SWITCH2       ;Compare switch bd.2 code
        JP Z,LOOP3
DELAY1: JP LOOP1        ;WAIT COMMAND


LOOP3:
        CALL ON_COM
        CALL SEND_SWITCH
        CALL WAITING
        JP LOOP1


ON_COM: LD    HL,LED
        LD    A,(HL)
        SET   0,A               ;ON COMMUNICATION
        LD    (HL),A
        OUT   (PORTC2),A
        RET
WAITING: LD    DE,0270H
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
LP:       DEC   DE
          LD    A,D
          OR    A
          JP    NZ,LP
          LD    HL,LED
          LD    A,(HL)
          RES   0,A
          LD    (HL),A
          RET


          ;*****************************************;
          ;*       SEND HEXCIMAL DATA ON SCREEN    *;
          ;*****************************************;

SEND_HEX:
          LD    A,083H    ;SUBROUTINE RXBYTE
          RST   10H       ;END SUBROUTINE
          RET             ;RETURN TO CALL
END:       RST   18H       ;QUIT PROGRAM

          ;*****************************************;
          ;*              SEND SWITCH              *;
          ;*****************************************;

SEND_SWITCH:
          IN    A,(PORTA2)  ;RECEIVE THE SWITCH PORT
          AND   07H         ;CLEAR BIT 2-7 TO 00H
          LD    C,A         ;LOAD A TO C REGISTER
          CALL  SEND_HEX    ;SEND SWITCH CODE TO SERIES PORT
          RET               ;RETURN TO MAIN

               .ORG 0EFFEH
            .DB   96H           ;BUARD RATE
            .DB   0A3H          ;AUTO RUN
                 .ORG 0F000H

DIGIT:      .DB   00H           ;RESERVE FOR COUNTER PORT
LED:        .DB   00H           ;STATUS LED
PORTIN:     .DB   00            ;SET START AT PORT 'B'
CH_DEC:     .DB   00H,00H       ;RESERVE FOR DECIMAL DATA
SH_VALUE:   .DB   00H,00H      ;RESERVE FOR SHOW VALUE ON SCREEN
            .END


B:\>
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
#include <bios.h>
#include <graphics.h>
#include <stdlib.h>
#include <stdio.h>
#include <conio.h>
#include <stdarg.h>
#include <math.h>
#include <dos.h>
#define  ESC       0x1b
#define  FALSE      0
#define  TRUE       1
#define F1         0x3B
#define F2         0x3C
#define F3         0x3D
#define SWITCH2    'U'
#define CHECK      'R'
#define TEMP       'T'
#define SWITCH     'S'
#define COM2        1   /* com2 */
#define DATA_READY 0x100
#define SETTINGS ( 0xe0 | 0x03 | 0x00 | 0x00)
 /* baurd rate =9600,parity = non,stop_bit = 1,data = 8 */
int gprintf(int *xloc, int *yloc, char *fmt, ...);
int CODE,BDCODE;
void changetextstyle(int font, int direction, int charsize);

char *Fonts[] = {
  "DefaultFont",   "TriplexFont",   "SmallFont",
  "SansSerifFont", "GothicFont"
};

char *LineStyles[] = {
  "SolidLn",  "DottedLn",  "CenterLn",  "DashedLn",  "UserBitLn"
};

char *FillStyles[] = {
  "EmptyFill",  "SolidFill",      "LineFill",      "LtSlashFill",
  "SlashFill",  "BkSlashFill",    "LtBkSlashFill", "HatchFill",
  "XHatchFill", "InterleaveFill", "WideDotFill",   "CloseDotFill"
};

char *TextDirect[] = {
  "HorizDir",  "VertDir"
};

char *HorizJust[] = {
  "LeftText",   "CenterText",   "RightText"
};

char *VertJust[] = {
  "BottomText",  "CenterText",  "TopText"
};

struct PTS {
  int x, y;
};      /* Structure to hold vertex points      */

            /*********************/
            /*  main program     */
            /*********************/

main()
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
{
  /* request auto detection */
  int gdriver = EGA, gmode=EGAHI, errorcode;
/* initialize graphics, local variables */
   initgraph(&gdriver, &gmode, "");

  /* read result of initialization */
   errorcode = graphresult();
  if (errorcode != grOk)  /* an error occurred */
  {
     printf("Graphics error: %s\n", grapherrormsg(errorcode));
     printf("Press any key to halt:");
     getch();
     exit(1); /* terminate with error code */
  }
      clrscr();
      bioscom(0,SETTINGS,COM2);
      CODE = 3;
            /*****************************/
            /* program loop non contion */
            /*****************************/

    while (CODE != 4)
    {
       switch (CODE)
         {
            case 3:
             {
                CODE = all_chanel();
              break;
             }
            case 1:
             {
               CODE = chanel_one();
               break;
             }
            case 2:
             {
                CODE = chanel_two();
                break;
             }
       }

      }
     closegraph();
     clrscr();
     return 0;

   }


            /************************/
            /*  program all chanel */
            /************************/
   all_chanel()
   {
       int count,new,disp[10],tp,sw,h,page=1,setpage=0;
       int xp1,yp1,midx,midy,MaxX,MaxY,ystep,scal,tsw,sw1,sw2,sw3,sw4;
       int swbd2,sensor2,in,DONE = FALSE,CODE;
       char temp,sensor;
       char buffer[11];
       char *TEXT[]={
                    "Alarm","Normal"};
    h = 3 * textheight( "H" );
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น และห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
  MaxX = getmaxx();
  MaxY = getmaxy();
  cleardevice();   /* clear graphic mode */
    while(!DONE)  /* loop no condition */
    {
     struct viewporttype vp;
     scal = 0;
         midx = (getmaxx() / 2)-100;
          midy = (getmaxy() / 2)+130;
    count = 5;           /******************/
     new = 1;            /*  command temp  */
      temp = TEMP;       /******************/
      sends(temp,count,new,&disp);
      count = 8;         /******************/
     new = 5;            /*  command switch */
    sensor = SWITCH;     /******************/
     sends(sensor,count,new,&disp);
      sensor2 = SWITCH2;
        count = 10;
         new = 8;
          sends(sensor2,count,new,&disp);
            tp = ((disp[1]+2)-50)*10 | ((disp[2]+2)-50);
            sw = ((disp[7]+2)-50);
            swbd2 = ((disp[9]+2)-50);
            setpage = page;
       setactivepage(page); /* set graphic mode */
         cleardevice();
          disp_graph(tp,sw,swbd2); /*  draw graphic */

/*  PRINT STRING TO CRT IN ANY FONT */
              setbkcolor(0);
              setcolor(14);
              getviewsettings( &vp );
            settextjustify( CENTER_TEXT, TOP_TEXT );
          changetextstyle( TRIPLEX_FONT, HORIZ_DIR, 4 );
        outtextxy( MaxX/2, 6, "MC. Base for Remote Control All Chanel"
           changetextstyle( DEFAULT_FONT, HORIZ_DIR,1);
              setcolor(14); /* yello */
              /***********  Y-scale are 0-100 **************/
          for (ystep = 0;ystep <220 ;ystep = ystep+20)
              {
               itoa(scal,buffer,10);
               outtextxy((midx-70),(midy-30)-ystep,buffer);  /* write s
               scal = scal+10;
              }
            settextjustify( BOTTOM_TEXT, LEFT_TEXT );
        xp1=200;
       yp1=295;
  setcolor (14);
  gprintf( &xp1, &yp1, "Temp    Zone_1    Zone_2     Zone_3     Zone4")
 xp1 = 200;
 yp1 = 310;
setcolor(14);
gprintf( &xp1, &yp1, "%c%c%c%c%c C",disp[1],disp[2],disp[3],disp[4],dis
 setcolor (15);
 xp1 = 100;
 yp1 = 330;
  gprintf(&xp1,&yp1, "F1 Chanel One    F2 Chanel Two    F3 All Chanel
  setcolor (14);
      tsw = sw;
      if (!(tsw & 0x1))          /* if switch1 was open */
         sw1 = 0;                /* sw =1 */
      else
         sw1 = 1;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
          tsw = sw;
              if (!(tsw & 0x2))      /* if switch2 was open */
                 sw2 = 0;             /* sw =1 */
                  else
            sw2 = 1;

           tsw = swbd2;
          if (!(tsw & 1))      /* if switch3 was open */
        sw3 = 0;             /* sw =1 */
    else
      sw3 = 1;
  tsw = swbd2;
if(!(tsw & 2))
   sw4 = 0;
 else
   sw4 = 1;
         settextjustify( BOTTOM_TEXT, LEFT_TEXT );
          xp1=275;
            yp1=310;
             gprintf( &xp1,&yp1, "%s",TEXT[sw1]);
             xp1 = xp1+80;
             yp1 = yp1-10;
             gprintf( &xp1,&yp1, "%s",TEXT[sw2]);
             xp1 = xp1+85;
             yp1 = yp1-10;
             gprintf( &xp1,&yp1, "%s",TEXT[sw3]);
             xp1 = xp1+90;
             yp1 = yp1-10;
             gprintf( &xp1,&yp1, "%s",TEXT[sw4]);


  /******** END WRITE TEXT IN ANY FONT *************/

             page = setpage^1;
              setvisualpage(setpage);
     delay (100);
   page = page;
     if (kbhit())
       {
         if ((in = getch()) == F1)
           {
            CODE = 1;      /* Code of board one */
            DONE = TRUE;
            nosound();}
         if (in == F2)
           {
           CODE = 2;      /* Code of board two */
            DONE = TRUE;
            nosound();
           }
         if (in == ESC)
           {
             CODE = 4;     /* exit */
             DONE = TRUE;
             nosound();
           }
      }
 }
return CODE;
}
                /****************************/
                /*     CHANEL    ONE        */
                /****************************/
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
chanel_one()
{
    int count,new,disp[9],tp,sw,h,page=1,setpage=0;
    int xp1,yp1,midx,midy,MaxX,MaxY,ystep,scal,tsw,sw1,sw2;
    int sensor2,in,DONE = FALSE,CODE;
    char temp,sensor;
    char buffer[11];
    char *TEXT[]={
              "Alarm","Normal"};
    h = 3 * textheight( "H" );
    MaxX = getmaxx();
    cleardevice();   /* clear graphic mode */
      while(!DONE)  /* loop no condition */
      {
       struct viewporttype vp;
       scal = 0;
       midx = (getmaxx() / 2)-100;
       midy = (getmaxy() / 2)+130;
      count = 5;              /*******************/
        new = 1;              /*  command temp   */
         temp = TEMP;         /*******************/
          sends(temp,count,new,&disp);
         count = 8;           /*******************/
        new = 5;              /*  command switch */
      sensor = SWITCH;        /*******************/
       sends(sensor,count,new,&disp);
              tp = ((disp[1]+2)-50)*10 | ((disp[2]+2)-50);
              sw = ((disp[7]+2)-50);
              setpage = page;
        setactivepage(page); /* set graphic mode */
          cleardevice();
           disp_graph1(tp,sw); /*  draw graphic */

/*  PRINT STRING TO CRT IN ANY FONT */
              setbkcolor(0);
              setcolor(14);
              getviewsettings( &vp );
            settextjustify( CENTER_TEXT, TOP_TEXT );
          changetextstyle( TRIPLEX_FONT, HORIZ_DIR, 4 );
         outtextxy( MaxX/2, 6, "MC. Base for Remote Control Chanel One"
           changetextstyle( DEFAULT_FONT, HORIZ_DIR,1);
              /***********  Y-scale are 0-100 **************/
            for (ystep = 0;ystep <220 ;ystep = ystep+20)
              {
               itoa(scal,buffer,10);
               outtextxy((midx-70),(midy-30)-ystep,buffer);  /* write s
               scal = scal+10;
              }
            settextjustify( BOTTOM_TEXT, LEFT_TEXT );
        xp1=200;
       yp1=295;
   setcolor (14);
   gprintf( &xp1, &yp1, "Temp    Zone_1    Zone_2");
  xp1 = 200;
  yp1 = 310;
 setcolor(14);
 gprintf( &xp1, &yp1, "%c%c%c%c%c C",disp[1],disp[2],disp[3],disp[4],dis
  setcolor (15);
  xp1 = 100;
  yp1 = 330;
   gprintf(&xp1,&yp1, "F2 Chanel Two       F3 All Chanel      ESC Exit")
    setcolor (14);
     tsw = sw;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    if (!(tsw & 0x1))      /* if switch1 was open */
       sw1 = 0;             /* sw =1 */
    else
       sw1 = 1;

         tsw = sw;
             if (!(tsw & 0x2))      /* if switch2 was open */
                sw2 = 0;             /* sw =1 */
                 else
           sw2 = 1;

  settextjustify( BOTTOM_TEXT, LEFT_TEXT );
           xp1=275;
             yp1=310;
              gprintf( &xp1,&yp1, "%s",TEXT[sw1]);
              xp1 = xp1+80;
              yp1 = yp1-10;
              gprintf( &xp1,&yp1, "%s",TEXT[sw2]);
              xp1 = xp1+85;
              yp1 = yp1-10;


  /******** END WRITE TEXT IN ANY FONT *************/

              page = setpage^1;
               setvisualpage(setpage);
     delay (100);
   page = page;
     if (kbhit())
       {
         if ((in = getch()) == F3)
          {
           CODE = 3;      /* Code of chenel two */
           DONE = TRUE;
           nosound();
          }
         if (in == F2)
          {
           CODE = 2;      /* all chanel */
           DONE = TRUE;
           nosound();
          }
         if (in == ESC)
           {
            CODE = 4;     /* exit */
            DONE = TRUE;
            nosound();
           }
       }
   }
  return CODE ;
 }

              /*****************************/
              /*      CHANEL  TWO          */
              /*****************************/

chanel_two()
{
    int count,new,disp[3],tp,swbd2,h,page=1,setpage=0;
    int xp1,yp1,midx,midy,MaxX,MaxY,ystep,scal,tsw,sw3,sw4;
    int sensor2,in,DONE = FALSE,CODE;
    char temp,sensor;
    char buffer[11];
    char *TEXT[]={
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่อนุญาตให้นำไปดัดแปลง เผยแพร่ หรือจำหน่าย ไม่ว่ากรณีใดๆ ทั้งสิ้น ไม่อนุญาตให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
            "Alarm","Normal"};
   h = 3 * textheight( "H" );
   MaxX = getmaxx();
   MaxY = getmaxy();
   cleardevice();   /* clear graphic mode */
     while(!DONE)  /* loop no condition */
     {
      struct viewporttype vp;

     scal  = 0;
        sensor2 = SWITCH2;
          count = 3;
           new = 1;
            sends(sensor2,count,new,&disp);
              swbd2 = ((disp[2]+2)-50);
              setpage = page;
        setactivepage(page); /* set graphic mode */
          cleardevice();
           disp_graph2(swbd2); /*  draw graphic */

 /*  PRINT STRING TO CRT IN ANY FONT */
                setbkcolor(0);
                setcolor(14);
                getviewsettings( &vp );
              settextjustify( CENTER_TEXT, TOP_TEXT );
            changetextstyle( TRIPLEX_FONT, HORIZ_DIR, 4 );
          outtextxy( MaxX/2, 6, "MC. Base for Remote Control Chanel Two"
             changetextstyle( DEFAULT_FONT, HORIZ_DIR,1);
    settextjustify( BOTTOM_TEXT, LEFT_TEXT );
         xp1=200;
        yp1=295;
    setcolor (14);
    gprintf( &xp1, &yp1, "Zone_3    Zone_4");
   setcolor (15);
   xp1 = 100;
   yp1 = 330;
    gprintf(&xp1,&yp1, "F1 Chanel One       F3 All Chanel      ESC Exit")
    setcolor (14);
        tsw = swbd2;
        if (!(tsw & 0x1))     /* if switch1 was open */
           sw3 = 0;           /* sw =1 */
        else
           sw3 = 1;

            tsw = swbd2;
                if (!(tsw & 0x2))     /* if switch2 was open */
                   sw4 = 0;           /* sw =1 */
                    else
              sw4 = 1;

     settextjustify( BOTTOM_TEXT, LEFT_TEXT );
               xp1=200;
                 yp1=310;
                  gprintf( &xp1,&yp1, "%s",TEXT[sw3]);
                  xp1 = xp1+80;
                  yp1 = yp1-10;
                  gprintf( &xp1,&yp1, "%s",TEXT[sw4]);
                  xp1 = xp1+85;
                  yp1 = yp1-10;


     /******** END WRITE TEXT IN ANY FONT ************/
                  page = setpage^1;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                    setvisualpage(setpage);
            delay (100);
          page = page;
            if (kbhit())
              {
                if ((in = getch()) == F1)
                  {
                   CODE = 1;      /* Code of chenel two */
                   DONE = TRUE;
                   nosound();
                  }
                if (in == F3)
                  {
                   CODE = 3;      /* all chanel */
                   DONE = TRUE;
                   nosound();
                  }
                if (in == ESC)
                  {
                    CODE = 4;     /* exit */
                    DONE = TRUE;
                    nosound();
                  }
              }
      }
     return CODE ;
    }
                /*************************************/
                /*   subrutine send and recives data */
                /*************************************/
  sends(in,loop,n,out)
   int loop,n,*out;
  char in;
    {
        int start,end,count,status,count1 = 0 ;
         bioscom(1, in, COM2);  /* sent command to control bd. */

     while (n<loop)
       {
       if (count1 > 10 )       /* wait for bord send data */
         {
           bioscom(1,in,COM2);   /* if not data send command again */
           count1 = 0;
          }
           start = clock();
           status = bioscom(3, 0, COM2);
            if (status & DATA_READY)
             if ((out[n] = bioscom(2, 0, COM2) & 0x7F) != 0)  /* if it have
                 n = n++;   /* counter wait for data of temp or switch */
                 end = clock();
                 count = end-start;
                 count1 = count+count1;
       }
      return 0;
     }
                /****************************/
                /*  display graphic         */
                /****************************/
  disp_graph(int temp,int sw,int swbd2)
{
  int midx, midy,scaley,scalex;
  int sw1,sw2,sw3,sw4,tsw;
  int beep1 =900,beep2 = 1000,beep3 = 2000,beep4 = 800,beep5 = 2500;
```

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
midx = (getmaxx() / 2)-100;
midy = (getmaxy() / 2)+100;

 for (scaley=0;scaley<220;scaley=scaley+20)  /* draw scal temp total ten */
  {
    line(midx-50,(midy+3)-scaley,midx-55,(midy+3)-scaley);
  }

{
   /* draw the 3-d bar */
  line(midx-50,midy+3,midx+330,midy+3);   /* horizontal line */
     line(midx-50,midy+3,midx-50,midy-200);  /* vertical line */
        if (temp > 70)
         {
            setfillstyle(CLOSE_DOT_FILL,12);
            bar3d(midx-20, (midy+3)-(temp*2), midx+20, midy+2,10,8 );
            beep(beep4);
          }
       else
          {
            setfillstyle(CLOSE_DOT_FILL,10);
            bar3d(midx-20, (midy+3)-(temp*2), midx+20, midy+2,10,8 );
          }
         tsw = sw;
         if (!(tsw & 1))      /* if switch1 was open */
            {
              sw1 = -90;              /* sw =1 */
              beep(beep1);
             }
         else
            sw1 = 110;

             tsw = sw;
                if (!(tsw & 2))      /* if switch2 was open */
                  {
                    sw2 = -90;              /* sw =1 */
                    beep(beep2);
                  }
                else
                   sw2 = 110;

              tsw = swbd2;
             if (!(tsw & 1))      /* if switch3 was open */
              {
                sw3 = -90;              /* sw =1 */
                beep(beep3);
              }
            else
                sw3 = 110;
      tsw = swbd2;
       if (!(tsw & 2))
        {
          sw4 = -90;
          beep(beep5);
        }
    else
        sw4 = 110;

         setfillstyle(WIDE_DOT_FILL,13);
         bar3d(midx+50 , (midy-90)-sw1, midx+90 , midy+2,10,8 );
         setfillstyle(CLOSE_DOT_FILL,14);
         bar3d(midx+120, (midy-90)-sw2, midx+160, midy+2,10,8 );
         setfillstyle(INTERLEAVE_FILL,10);
         bar3d(midx+190, (midy-90)-sw3, midx+230, midy+2,10,8 );
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
            setfillstyle(INTERLEAVE_FILL,9);
            bar3d(midx+260, (midy-90)-sw4, midx+300, midy+2,10,8 );
  }
  return 0 ;
  }
                /*******************************/
                /*  display graphic chanel one */
                /*******************************/
  disp_graph1(int temp,int sw)
{
   int midx, midy,scaley,scalex;
   int sw1,sw2,tsw;
   int beep1 =900,beep2 = 1000,beep4 = 800;
   midx = (getmaxx() / 2)-100;
   midy = (getmaxy() / 2)+100;

   for (scaley=0;scaley<220;scaley=scaley+20)  /* draw scal temp total ten */
    {
      line(midx-50,(midy+3)-scaley,midx-55,(midy+3)-scaley);
    }

   {
    /* draw the 3-d bar */
    line(midx-50,midy+3,midx+330,midy+3);   /* horizontal line */
       line(midx-50,midy+3,midx-50,midy-200);  /* vertical line */
          if (temp > 70)
           {
              setfillstyle(CLOSE_DOT_FILL,12);
              bar3d(midx-20, (midy+3)-(temp*2), midx+20, midy+2,10,8 );
              beep(beep4);
           }
         else
           {
              setfillstyle(CLOSE_DOT_FILL,10);
              bar3d(midx-20, (midy+3)-(temp*2), midx+20, midy+2,10,8 );
           }

          tsw = sw;
          if (!(tsw & 1))      /* if switch1 was open */
             {
               sw1 = -90;              /* sw =1 */
               beep(beep1);
              }
          else
             sw1 = 110;

              tsw = sw;
                  if (!(tsw & 2))      /* if switch2 was open */
                     {
                       sw2 = -90;              /* sw =1 */
                       beep(beep2);
                     }
                  else
                      sw2 = 110;
            setfillstyle(WIDE_DOT_FILL,13);
            bar3d(midx+50 , (midy-90)-sw1, midx+90 , midy+2,10,8 );
            setfillstyle(CLOSE_DOT_FILL,14);
            bar3d(midx+120, (midy-90)-sw2, midx+160, midy+2,10,8 );
}
  return 0 ;
  }
                /*******************************/
                /*  display graphic chanel two */
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                /*******************************/

  disp_graph2(int swbd2)
{
   int midx, midy,scaley,scalex;
   int sw3,sw4,tsw;
   int beep3 =2000,beep5 = 2500;
   midx = (getmaxx() / 2)-100;
   midy = (getmaxy() / 2)+100;


   {
    /* draw the 3-d bar */
    line(midx-50,midy+3,midx+150,midy+3);   /* horizontal line */
       line(midx-50,midy+3,midx-50,midy-200);  /* vertical line */
           tsw = swbd2;
           if (!(tsw & 1))      /* if switch1 was open */
              {
                sw3 = -90;              /* sw =1 */
                beep(beep3);
              }
           else
              sw3 = 110;

               tsw = swbd2;
                   if (!(tsw & 2))      /* if switch2 was open */
                      {
                        sw4 = -90;              /* sw =1 */
                        beep(beep5);
                      }
                   else
                        sw4 = 110;
           setfillstyle(WIDE_DOT_FILL,10);
           bar3d(midx-20 , (midy-90)-sw3, midx+20 , midy+2,10,8 );
           setfillstyle(CLOSE_DOT_FILL,9);
           bar3d(midx+50, (midy-90)-sw4, midx+90, midy+2,10,8 );
}
  return 0 ;
  } /*                                                              */
/*      GPRINTF: Used like PRINTF except the output is sent to the  */
/*      screen in graphics mode at the specified co-ordinate.       */
/*                                                                  */

int gprintf( int *xloc, int *yloc, char *fmt, ... )
{
  va_list  argptr;                      /* Argument list pointer      */
  char str[240];                        /* Buffer to build sting into */
  int cnt;                              /* Result of SPRINTF for return */

  va_start( argptr, fmt );              /* Initialize va_ functions   */

  cnt = vsprintf( str, fmt, argptr );   /* prints string to buffer    */
  outtextxy( *xloc, *yloc, str );       /* Send string in graphics mode */
  *yloc += textheight( "H" ) + 2;       /* Advance to next line       */

  va_end( argptr );                     /* Close va_ functions        */

  return( cnt );                        /* Return the conversion count */

}

/*                                                                  */
/*      CHANGETEXTSTYLE: similar to settextstyle, but checks for    */
/*      errors that might occur whil loading the font file.         */
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
/*                                                                    */

void changetextstyle(int font, int direction, int charsize)

{
  int ErrorCode;

  graphresult();                          /* clear error code          */
  settextstyle(font, direction, charsize);
  ErrorCode = graphresult();              /* check result              */
  if( ErrorCode != grOk ){                /* if error occured          */
    closegraph();
    printf(" Graphics System Error: %s\n", grapherrormsg( ErrorCode ) );
    exit( 1 );
  }
}
                /**********************/
                /*     Sound          */
                /**********************/

 beep(f)
   int f; /* frequency */
   {
    int count = 0;
    while( count < 2)
    {
      sound (f);
      delay (100);
      nosound();
      delay (40);
      count = count++;
    }
    return(0);
   }
B:\>
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เอกสารอ้างอิง

1. Douglas V.Hell "MICROCOMPUTER AND INTERFACING: Programming and Harware" McGRAW-HELL'Inc. ,1986
2. Seteven Holzner "C WITH ASSEMPLY LANGAUGE" Simon & Schster,Inc. ,1989
3. Lewis C. Eggebrecht "INTERFACING TO IBMPERSONAL COMPUTER" Howard W.sams & Co.,Inc. ,1983
4. Yuen Bhoorawan "SEMICONDUCTOR ELECTRONIC JOUNAL: File microprocessor CH . UCATION. , 1993

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้