สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เครื่องตรวจกระดาษคำตอบแบบอัตโนมัติ

CHECKER AUTOMACHINE

โดย

นาย นฤพร หินสมุทร

นาย วัชรินทร์ อรรคเศรษฐัง

นาย สมชาย ทิพย์ประเสริฐสุข

นาย สมาน พุทธรังษี

นาย สุรรัตน์ บัวสมบูรณ์

นาย พูลศักดิ์ ธรรมนิยมกุล

วัน เดือน ปี 29 ก.ค. 2541

เลขทะเบียน 038033

เลขเรียกหนังสือ T 39053 ส325ค

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตร์บัณฑิต

สาขาวิชา เทคโนโลยีการวัดคุมทางอุตสาหกรรม

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2539

ปริญญานิพนธ์ปีการศึกษา 2539

ภาควิชา เทคโนโลยีการวัดคุมทางอุตสาหกรรม

สาขาวิศวกรรมการวัดคุมทางอุตสาหกรรม

คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เรื่อง

เครื่องตรวจกระดาษคำตอบแบบอัตโนมัติ

CHECKER AUTOMACHINE

ผู้จัดทำ

1. นาย นฤพร	หินสมุทร	37012097
2. นาย วัชรินทร์	อรรคเศรษฐัง	37012108
3. นาย สมชาย	ทิพย์ประเสริฐสุข	37012115
4. นาย สมาน	พุทธรังษี	37012120
5. นาย สุรรัตน์	บัวสมบูรณ์	37012123
6. นาย พูลศักดิ์	ธรรมนิยมกุล	37013436

.....อาจารย์ที่ปรึกษา

(อาจารย์วิริยะ กองรัตน์)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เครื่องตรวจกระดาษคำตอบแบบอัตโนมัติ

นาย นฤพร หินสมุทร
นาย วัชรินทร์ อรรคเศรษฐัง
นาย สมชาย ทิพย์ประเสริฐสุข
นาย สมาน พุทธรังษี
นาย สุรรัตน์ บัวสมบูรณ์
นาย พูลศักดิ์ ธรรมนิยมกุล
อาจารย์วิริยะ กองรัตน์ อาจารย์ที่ปรึกษา
ปีการศึกษา 2539

บทคัดย่อ

เครื่องตรวจกระดาษคำตอบเป็นเครื่องมือสำหรับตรวจเช็คคำตอบของข้อสอบบนกระดาษคำตอบที่ได้จัดทำขึ้นมาเป็นพิเศษซึ่งกระดาษคำตอบนี้จะอาศัยการตรวจจับการดูดกลืนแสงอินฟาเรด (Infrared)ของสารคาร์บอน(Carbon) ที่ระบายบนกระดาษคำตอบ จะให้ความรวดเร็วและถูกต้องแม่นยำโดยไม่ต้องอาศัยคนตรวจการตรวจจับจะใช้โฟโตไดโอดเป็นตัวรับสัญญาณแสงที่ส่งมาจากตัวกำเนิดแสงอินฟาเรดส่งผ่านสารคาร์บอน สัญญาณที่ได้จากโฟโตไดโอดจะเป็นสัญญาณอนาลอกดังนั้นก่อนการนำไปทำการประมวลผลข้อมูลจะต้องแปลงให้เป็นสัญญาณดิจิตอลก่อน หลังจากนั้นข้อมูลที่เป็นดิจิตอลจะถูกส่งเข้าไปประมวลผลในตัวไมโครคอนโทรเลอร์ (MCS51) แล้วข้อมูลจะถูกส่งไปยังเครื่องคอมพิวเตอร์ที่มีโปรแกรมซึ่งได้ทำการเขียนขึ้นเพื่อรองรับข้อมูลจากตัวไมโครคอนโทรลเลอร์ โดยโปรแกรมนี้จะใช้เก็บข้อมูลที่ได้ให้เป็นระเบียบและแสดงผลออกมาทางหน้าจอคอมพิวเตอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

CHECKER AUTOMACHINE

MR. NARUEPORN HINSAMOOTH
MR. WATCHARIN AKKASETTHANG
MR. SOMCHAI TIPPRASEARTSUK
MR. SUMAN PUTHARANGSEE
MR. SURARAT BUASOMBOON
MR. POOLSAK TUMMANIYOMKUL
MR. VIRIYA KONGRAT ADVISER

1996

Abstract

This projet is the machine for check answer the question which on the special answer sheet. It check is very quickly and don't have to people check. It will autometic check by sensing the carbon on the answer sheet with infared LED and detecting by photo diode and applying square wave signal follow to carbon on the answer sheet. Signal that can receive from photo diode will be analog signal. So we must convert this analog signal to digital signal. After that, digital data will transfer to process into Microcontroller (MCS-51). And data will sent to computer that have a suitable program to support this data. This program to used for maintence data and display result by monitor.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กิตติกรรมประกาศ

ขอขอบคุณ อาจารย์วิริยะ กองรัตน์ เป็นอย่างสูงซึ่งเป็นอาจารย์ที่ปรึกษาปริญญานิพนธ์ฉบับนี้ที่ให้ความช่วยเหลือในด้านต่าง ๆ และให้คำปรึกษาเกี่ยวกับการทำปริญญานิพนธ์

ขอขอบคุณเพื่อน ๆ ที่ช่วยเหลือในด้านต่าง ๆ

ท้ายสุดของกิตติกรรมประกาศนี้ ขอขอบพระคุณ คุณพ่อ คุณแม่ ซึ่งเป็นผู้ที่ให้ความช่วยเหลืออย่างมากทั้งด้านกำลังใจ และทุนในการทำปริญญานิพนธ์

คณะผู้จัดทำ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ

บทที่ 1 บทนำ 1

บทที่ 2 ทฤษฎีและหลักการอินฟาเรด 3

2.1 ทฤษฎีของแสงและตัวกำเนิดแสงอินฟาเรด 3

2.2 หลักการของตัวรับแสงอินฟาเรด 10

2.3 ผลตอบสนองของคลื่นแสงอินฟาเรดเมื่อวิ่งผ่านแถบขาว-ดำ 15

บทที่ 3 ไมโครคอนโทรลเลอร์ 20

3.1 โครงสร้างและคุณสมบัติของไมโครคอนโทรลเลอร์ (MCS-51) 20

3.2 ANT-32 EBMEDDED CONTROLLER 29

3.3 การใช้งาน 8255 กับ 8051 33

บทที่ 4 การสื่อสารข้อมูลผ่านทางพอร์ตขนาน (พอร์ตพริ้นท์เตอร์) 40

4.1 พอร์ตอินพุต/เอาต์พุต 40

4.2 พอร์ตรีจีสเตอร์ 43

4.4 การติดต่อระหว่างโฮสและพริ้นท์เตอร์ 46

4.5 สายเคเบิล 48

4.6 การโปรแกรมผ่านทางพอร์ตขนาน 50

บทที่ 5 การประยุกต์ใช้งานโปรแกรม Visual Basic for Windows 56

5.1 Visual Basic for Windows 56

5.2 การสร้างและการใช้งาน Dynamic Link Library 57

บทที่ 6 การออกแบบและการทำงาน 64

6.1 การออกแบบวงจรที่ใช้กับเครื่องตรวจกระดาษคำตอบ 64

6.2 โครงสร้างของเครื่องตรวจกระดาษคำตอบ 70

6.3 การทำงานของเครื่องตรวจกระดาษคำตอบ 75

6.4 การทำงานของเครื่องตรวจข้อสอบแสดงเป็นบล็อกไดอะแกรม 76

บทที่ 7 รายละเอียดการทำงานของโปรแกรมเครื่องตรวจข้อสอบ 80

7.1 โปรแกรมบนเครื่องคอมพิวเตอร์ 80

7.2 โปรแกรมบนเครื่องคอมพิวเตอร์ 92

7.3 รายละเอียดโปรแกรม 95

7.4 วิธีการใช้งานเครื่องตรวจข้อสอบ 96

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญตาราง

ตารางที่ 2.1 การแผ่ของพลังงานแม่เหล็กไฟฟ้า 3
ตารางที่ 2.2 ตารางเปรียบเทียบพลังงานและความยาวคลื่นจากตัวกำเนิด 4
แสงที่สร้างจากสารต่างชนิดกัน
ตารางที่ 2.3 เปรียบเทียบพลังงานมที่ได้จากตัวกำเนิดแสงที่สร้างที่สร้าง 5
จาก GaAlAs และ GaAs
ตารางที่ 3.1 แสดงบิตควบคุมที่อยู่ในรีจีสเตอร์ TMOD 27
ตารางที่ 3.2 แสดงบิตควบคุมที่อยู่ในรีจีสเตอร์ SCON 28
ตารางที่ 3.3 ตารางแสดงลักษณะของ GROUP 35
ตารางที่ 3.4 ตารางแสดงหน้าที่การทำงานของขาสัญญาณไอซี 8255 35
ตารางที่ 3.5 แสดงรายชื่อของรีจีสเตอร์ที่ตำแหน่งแอดเดรสต่างๆของ 8255 37
ตารางที่ 3.6 แสดงความหมายของค่าแอดเดรสเมื่อใช้งานร่มกันกับระดับลอจิก 37
ของขาสัญญาณ RD\ และ WR\
ตารางที่ 4.1 แสดงการกำหนดตำแหงอินพุต/เอาต์พุตแอดเดรสสำหรับพอร์ตขนาน 40
ตารางที่ 4.2 แสดงค่าเบสแอดเดรสของพอร์ตขนาน 40
ตารางที่ 4.3 แสดงขาของโฮสและพริ้นท์เตอร์ที่ต่อกัน 48
ตารางที่ 7.1 ตารางรายละเอียดนักศึกษา, รายชื่อนักศึกษา, ... 81
ตารางที่ 7.2 ตารางบันทึกคำตอบของข้อสอบ 81
ตารางที่ 7.3 รายละเอียดโปรแกรมทั้งหมดที่ได้รับการพัฒนาบนเครื่องตรวจข้อสอบ 95

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูปภาพ

รูปที่ 2.1 กราฟแสดงการเปรียบเทียบความยาวคลื่นที่ได้จากตัวกำเนิดแสงอินฟาเรดจากสารต่างชนิดกัน 5
รูปที่ 2.2 ชั้นของการโดปสารตัวกำเนิดแสงชนิด GaAlAs 6
รูปที่ 2.3 การแพร่กระจายของแสงจากเลนส์แบบต่างๆ 7
รูปที่ 2.4 การแพร่กระจายของแสงจากการใช้เลนส์ต่างชนิดกัน 8
รูปที่ 2.5 ลักษณะการแพร่กระจายของแสงในตัวกำเนิดแสง 8
รูปที่ 2.6 การใช้เลนส์ช่วยในการปรับปรุงคุณภาพแสง 9
รูปที่ 2.7 การกระจายของสนามแม่เหล็กไฟฟ้าในสารกึ่งตัวนำ 10
รูปที่ 2.8 ลักษณะโครงสร้างโดยทั่วไปของโฟโตไดโอด 11
รูปที่ 2.9 วงจรจำลองของโฟโตไดโอดและการขยายบริเวณดีพลีทชั่นให้มากขึ้น 12
รูปที่ 2.10 วงจรจำลองของโฟโตไดโอดและการขยายบริเวณดีพลีทชั่นให้มากขึ้น(2) 13
รูปที่ 2.11 วงจรโฟโตทรานสซีสเตอร์ 14
รูปที่ 2.12 กราฟแสดงคุณสมบัติของโฟโตทรานซีสเตอร์ 14
รูปที่ 2.13 การต่อใช้งานโฟโตทรานซีสเตอร์ในลักษณะต่างๆ 15
รูปที่ 2.14 การสะท้อนแสงที่ตกกระทบแถบขาว-ดำ 16
รูปที่ 2.15 ลักษณะของสัญญาณที่ได้จากการสะท้อนแสงเมื่อผ่านแถบสี 17
รูปที่ 2.16 ผลลการตอบสนองที่ตำแหน่งต่างๆ 17
รูปที่ 2.17 ความสัมพันธ์ของแสงที่ไหลผ่านกับแสงที่จ่ายออกของโฟโต 18
รูปที่ 2.18 ความสัมพันธ์ของแสงกับกระแสที่ไหลผ่านโฟโต 19
รูปที่ 2.19 เปรียบเทียบขนาดของตัวกำเนิดแสงกับตัวสะท้อนแสงที่มีผลต่อสัญญาณช่างกัน 19
รูปที่ 3.1 การเชื่อมต่อกับหน่วยความจำโปรแกรม 20
รูปที่ 3.2 การเชื่อมต่อกับหน่วยความจำข้อมูล .....
รูปที่ 3.3 การจัดพื้นที่หน่วยความจำโปรแกรมภายในและภายนอก 21
รูปที่ 3.4 การจัดพื้นที่หน่วยความจำข้อมูล 21
รูปที่ 3.5 พื้นที่การกำหนดตำแหน่งบิตของหน่วยความจำขอ้มูลภายในจากรูปที่ 3.4 23
รูปที่ 3.6 แสดงวงจรการทำงานของตัวจับเวลา/ตัวนับที่ 1 ในโหมด 0 และโหมด 1 25
รูปที่ 3.7 แสดงวงจรการทำงานของตัวจับเวลา/ตัวนับที่ 1 ในโหมด 2 25
รูปที่ 3.8 วงจรการทำงานของตัวจับเวลาตัวนับที่ 0 ในโหมดที่ 3 28

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 3.9 วงจร ANT-32 V2.0 31
รูปที่ 3.10 การจัดการหน่วยความจำของบอร์ด ANT-32 V2.0 33
รูปที่ 3.11 แผนภาพแบบบล็อกภายในและสัญญาณของไอซีเบอร์ 8255 34
รูปที่ 3.12 ความหมายของบิตภายในไบต์ข้อมูลควบคุมของ 8255 36
รูปที่ 3.13 แผนภาพแสดงการสร้างสัญญาณเลือกอุปกรณ์ (CS\) ให้กับ 8255 38
โดยการถอดรหัสจากบัสแอดเดรส A2-A7
รูปที่ 3.14 แผนภาพวงจรแสดงการเชื่อมต่อระหว่าง 8255 กับ 8051 39
รูปที่ 4.1 ไดอะแกรมของอะแดปเตอร์ต่อกับเครื่องพิมพ์ 41
รูปที่ 4.2 แสดงวงจรพริ้นท์เตอร์บนเครื่อง IBM PC 42
รูปที่ 4.3 แสดงการเชื่อมโยงระหว่างคอมพิวเตอร์กับเครื่องพิมพ์ 43
รูปที่ 4.4 โครงสร้างของ Parallel Printer Register ตัวที่ 1 44
รูปที่ 4.5 โครงสร้างของ Parallel Printer Register ตัวที่ 2 45
รูปที่ 4.6 โครงสร้างของ Parallel Printer Register ตัวที่ 3 46
รูปที่ 4.7 หัวต่อเครื่องพิมพ์ 25 ขา 49
รูปที่ 4.8 แสดงการต่อขาในแต่ละปลายของสายเคเบิลที่เป็น Parallel Tranfer Cable 50
รูปที่ 4.9 The Communication Protocal at the Hardware Oriented Data 51
รูปที่ 6.1 แสดงวงจรภาคส่งแสงอินฟาเรด 64
รูปที่ 6.2 แสดงวงจรภาครับแสงอินฟาเรด 64
รูปที่ 6.3 แสดงวงจรรวมภาคส่งแสงอินฟาเรด 65
รูปที่ 6.4 แสดงวงจรรวมภาครับสัญญาณอินฟาเรด 66
รูปที่ 6.5 แสดงวงจรนับ 68
รูปที่ 6.6 วงจรควบคุม DC Motor 69
รูปที่ 6.7 ลักษณะภายนอกของเครื่องตรวจกระดาษคำตอบ 70
รูปที่ 6.8 แสดงที่ใส่กระดาษก่อนผ่านเข้าเครื่องตรวจ 71
รูปที่ 6.9 แสดงการติดตั้งสปริง ลูกยาง และมอเตอร์ชุดฟีดกระดาษเข้า 71
รูปที่ 6.10 แสดงการติดตั้งชุดฟีดกระดาษและตัว Sensor 72
รูปที่ 6.11 แสดงการติดตั้งตัว Sensor เมื่อมองจากด้านข้าง 72
รูปที่ 6.12 แสดงที่รองรับกระดาษที่ผ่านการตรวจแล้ว 73
รูปที่ 6.13 แสดงการติดตั้งอุปกรณ์ชุดฟีดกระดาษเข้าและชุดขับให้กระดาษ 74
เคลื่อนที่เมื่อมองจากด้านบน
รูปที่ 6.14 แสดงภาพภายในของเครื่องตรวจกระดาษคำตอบ 75
รูปที่ 6.15 บล็อกไดอะแกรมแสดงการทำงานของเครื่องตรวจข้อสอบ 78

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 6.16 แสดงวงจรศูนย์กลางการเชื่อมต่อระหว่างอุปกรณ์ต่างๆ 79
รูปที่ 7.1 แสดงผังการงานระบบเครื่องตรวจข้อสอบ.....72
รูปที่ 7.2 ผังการทำงานของการส่งรหัสคำสั่ง.....85
รูปที่ 7.3 ผังการทำงานของการรับข้อมูลที่ได้จากการตรวจจับ 86
รูปที่ 7.4 แสดงลักษณะตำแหน่งต่างๆ ของข้อมูลบนกระดาษคำตอบที่ทำการถอดรหัส 87
รูปที่ 7.5 ผังการทำงานของโปรแกรมถอดรหัสนักศึกษา.....89
รูปที่ 7.7 ผังการถอดรหัสคำตอบด้านหน้า.....90
รูปที่ 7.8 ผังการถอดรหัสคำตอบด้านหลัง 91
รูปที่ 7.9 ผังการทำงานของโปรแกรมบนเครื่องตรวจข้อสอบ.....93
รูปที่ 7.10 ผังการทำงานของโปรแกรมอ่านข้อมูลจากชุดตรวจจับพร้อมส่งข้อมูล 94
รูปที่ 7.11 แสดงรายละเอียดเกี่ยวกับโปรแกรมเครื่องตรวจข้อสอบ 96
รูปที่ 7.12 แสดงสถานะพร้อมทำงานของโปรแกรมเครื่องตรวจข้อสอบ.....97
รูปที่ 7.13 แสดงการเปิดตารางข้อมูลเพื่อใช้งาน.....97
รูปที่ 7.14 รูปแสดงรายละเอียดทั้งหมดของนักศึกษา.....98
รูปที่ 7.15 แสดงกราฟผลของการตรวจคำตอบ.....99
รูปที่ 7.16 ตารางสรุปรายละเอียดของการตราจคำตอบ 100
รูปที่ 7.17 แสดงการเลือกอุปกรณ์ของเครื่องตรวจข้อสอบเพื่อทดสอบ.....100
รูปที่ 7.18 หน้าจอโปรแกรมทดสอบการทำงานของมอเตอร์ 101
รูปที่ 7.19 หน้าจอโปรแกรมทดสอบการทำงานของชุดตรวจจับ.....101
รูปที่ 7.20 หน้าจอโปรแกรมทดสอบการทำงานของชุดนับกระดาษคำตอบ.....102
รูปที่ 7.21 หน้าจอปรับแต่งข้อมูลเบื้องต้นของระบบ.....103
รูปที่ 7.22 หน้าจอโปรแกรมจัดการคำตอบของระบบ 103
รูปที่ 7.23 หน้าจอแสดงคำตอบของระบบ.....104
รูปที่ 7.24 หน้าจอโปรแกรมอ่านคำตอบมาตรฐานเข้าสู่ระบบ 104
รูปที่ 7.25 แสดงข้อความไม่สามารถติดต่อกับเครื่องตรวจข้อสอบได้.....105
รูปที่ 7.26 คำเตือนแสดงการเปิดฐานข้อมูลที่มีโครงสร้างไม่ตรงกับระบบ.....105
รูปที่ 7.27 แสดงข้อความว่าขณะนี้กระดาษที่ถาดของเครื่องตรวจคำตอบหมด 105
รูปที่ 7.28 หน้าจอแสดงโปรแกรมอ่านคำตอบจากกระดาษคำตอบของนักศึกษา.....106
รูปที่ 7.29 แสดงข้อความเตือนว่าไม่พบรหัสนักศึกษาในฐานข้อมูล.....107
รูปที่ 7.30 แสดงข้อความเตือนว่ารหัสนักศึกษาที่ทำการตรวจมีความผิดพลาด 107

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 1
บทนำ

การทำข้อสอบแบบกากบาทสมัยก่อนเป็นที่นิยมทำกันของผู้ออกข้อสอบเพราะตรวจง่ายและให้ความรวดเร็วพอสมควรโดยผู้ตรวจไม่ต้องเสียเวลาในการคิดตามคำตอบที่ผู้สอบตอบมา แต่อย่างไรก็ตามข้อสอบแบบนี้ก็ยังจำเป็นต้องใช้คนในการตรวจสอบอยู่ดีและมีโอกาสผิดพลาดได้มากยิ่งถ้าข้อสอบมีจำนวนมากๆจะทำให้เสียเวลาในการตรวจเป็นอย่างมาก ต่อมาจึงมีการคิดหาวิธีการทำข้อสอบแบบใหม่โดยไม่จำเป็นต้องใช้คนในการตรวจข้อสอบแต่จะใช้เครื่องมืออิเลคทรอนิคส์เข้ามาช่วยในการตรวจสอบซึ่งจะให้ผลในการตรวจที่แม่นยำไม่ลำเอียงกับผู้เข้าสอบสามารถตรวจได้รวดเร็วและจำนวนมากแม้ต้องตรวจติดต่อกันเป็นระยะเวลานานๆก็ไม่มีปัญหา ดังนั้นจึงทำให้การตรวจข้อสอบเป็นไปได้โดยง่ายและรวดเร็ว

การทำงานของเครื่องตรวจกระดาษคำตอบนั้นจะอาศัยคุณสมบัติของการดูดกลืนคลื่นแสงอินฟาเรดของสารคาร์บอนมาประยุกต์ใช้งาน นั่นคือเนื่องจากสารคาร์บอนดูดกลืนคลื่นแสงอินฟาเรดได้ดี ดังนั้นจะใช้ตัวตรวจจับแสงอินฟาเรดตรวจจับปริมาณของพลังงานแสงที่สามารถผ่านการดูดกลืนของสารคาร์บอนเข้ามาตกกระทบตัวมันแล้วทำให้ตรวจจับเกิดการเปลี่ยนแปลงทางฟิสิกส์ภายในตัวตรวจจับซึ่งมีผลทำให้เกิดการเปลี่ยนแปลงของพลังงานขึ้นภายในตัวตรวจจับพอที่จะนำสัญญาณนั้นมาประยุกต์ใช้งานได้ การทำงานของเครื่องตรวจข้อสอบจะมีความสลับซับซ้อนมากน้อยเพียงใดก็ขึ้นอยู่กับความต้องการของผู้ใช้ว่าต้องการให้ความสะดวกในการใช้งานของเครื่องตรวจกระดาษคำตอบมากน้อยเพียงใด ถ้าต้องการความสะดวกในการใช้งานมากขึ้นความสลับซับซ้อนของอุปกรณ์ทางอิเล็กทรอนิกส์ที่จะนำมาประกอบเพื่อสร้างความเป็นเครื่องดังกล่าวก็ย่อมจะสลับซับซ้อนเป็นเงารวมตัวรวมทั้งราคาสร้างด้วย แต่โดยหลักการแล้วก็ยังคงมีลักษณะที่คล้ายกันมาก ประสิทธิภาพ ความเที่ยงตรงก็ยังขึ้นอยู่กับอุปกรณ์และการสร้างของบริษัทนั้นๆด้วย

ในปัจจุบันจะเห็นว่าเครื่องตรวจกระดาษคำตอบไม่มีการประกอบขึ้นใช้เองในประเทศไทยแต่จะเป็นการสั่งเข้ามามากกว่าซึ่งในการสั่งเข้ามาจะเสียค่าใช้จ่ายจำนวนมากและมีจำนวนเครื่องที่ใช้งานอยู่จำนวนจำกัดทั้งที่ความต้องการในการใช้งานมีจำนวนหลายแห่งตามสถานที่ต่างๆของรัฐโดยเฉพาะสถานศึกษาซึ่งนับว่ามีความจำเป็นต้องใช้เครื่องตรวจกระดาษคำตอบมีหลักการทำงานเบื้องต้นทั่วๆไปนั้นไม่ยากสำหรับการศึกษา

การศึกษาการทำงานเบื้องต้นของข้อมูลของอุปกรณ์อิเล็กทรอนิกส์เป็นสิ่งจำเป็นอย่างมากสำหรับผู้ต้องการค้นคว้าทางด้านนี้เพราะการทำงานทุกอย่างจะอาศัยพื้นฐานของงานนั้นๆเข้ามาช่วยด้วยการศึกษางานทางด้านอิเล็กทรอนิกส์ก็เช่นกันอย่างไรก็ตามเครื่องตรวจกระดาษคำตอบนี้ก็ยังคงเป็นเครื่องต้นแบบเพื่อศึกษาแต่ก็ใช้งานจริงได้ ในปัจจุบันเทคโนโลยีทางอิเล็กทรอนิกส์มี

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ความก้าวหน้าไปมากสามารถนำอุปกรณ์ย่อยๆทางอิเล็กทรอนิกส์เข้ามารวมอยู่ในชิบตัวเดียวกันได้จำนวนมากๆทำให้ประหยัดทั้งอุปกรณ์และเงินลงทุน ดังที่กล่าวมาแล้วว่าการจัดทำโครงงานนี้มุ่งเพื่อที่จะเป็นผู้นำแนวทางเบื้องต้นให้กับผู้ต้องการศึกษาการทำงานเบื้องต้นของเครื่องตรวจกระดาษคำตอบเพื่อให้สามารถนำไปประยุกต์ให้วงจรมีความซับซ้อนมากขึ้นไป นอกจากนี้ยังจะสามารถนำไปใช้ในงานจริงได้อีกด้วย หลักการทำงานโดยทั่วไปของเครื่องตรวจกระดาษคำตอบแบ่งได้ดังนี้

ส่วนวงจรภาคตรวจจับข้อมูล เป็นตัวหลักของเครื่องเพราะถ้าภาคนี้ทำงานผิดพลาดไปแล้วก็ยากที่จะนำไปประมวลผลจะถูกต้องได้ เนื่องจากโครงงานนี้อาศัยหลักการของการดูดกลืนคลื่นแสงอินฟาเรดดังนั้นอุปกรณ์ที่ใช้ในการตรวจจับก็ย่อมหนีไม่พ้นตัวกำเนิดแสงอินฟาเรด เมื่อมีตัวกำเนิดแสงอินฟาเรดซึ่งอุปกรณ์ทั้งสองจะต้องทำงานสัมพันธ์กันและสารคาร์บอนที่จะใช้ในการตรวจจับก็เป็นสิ่งจำเป็น

ส่วนควบคุมการตรวจจับและประมวลผล จะใช้ตัวไมโครคอนโทรลเลอร์(MCS-51)เป็นตัวควบคุมและประมวลผลข้อมูลที่เป็นสัญญาณดิจิตอลซึ่งส่งมาจากวงจรตรวจจับ โดยการประมวลผลจะสามารถทำได้อย่างรวดเร็วและถูกต้องแม่นยำ

ส่วนแสดงผล เมื่อตัวไมโครคอนโทรลเลอร์ประมวลผลแล้วข้อมูลที่จะถูกส่งออกไปแสดงผลทางคอมพิวเตอร์ ซึ่งในตัวคอมพิวเตอร์จะมีโปรแกรมที่ได้ออกแบบไว้เพื่อใช้แสดงผลข้อมูลที่ส่งมาจากตัวไมโครคอนโทรลเลอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 2

ทฤษฎีและหลักการอินฟาเรด

2.1 ทฤษฎีของแสงและตัวกำเนิดแสงอินฟาเรด

พลังงานแสงเป็นพลังงานแม่เหล็กไฟฟ้าแบบหนึ่งที่มีความถี่ ความยาวคลื่นและความเร็วเป็นไปตามสมการด้านล่าง ซึ่งแสงแต่ละชนิดมีความยาวคลื่นแตกต่างกันออกไปตามความถี่ของแสงนั้นๆ

$$\lambda = v/f$$

เมื่อ

λ : ความยาวคลื่นมีหน่วยเป็นเมตร

f : ความถี่มีหน่วยเป็นเฮิทซ์

v : ความเร็วของแสงซึ่งมีค่าเท่ากับ 3×10^{8} เมตรต่อวินาที

ทฤษฎีควอนตัม(quantum)ของแพลงค์กล่าวว่าแหล่งกำเนิดพลังงานจะไม่แผ่หรือปล่อยพลังงานออกมาอย่างต่อเนื่องแต่จะปล่อยออกมาเป็นกลุ่มๆ และมีหน่วยขนาดของกลุ่มเรียกว่า "ควอนตัม" สำหรับการแผ่พลังงานแม่เหล็กเรียกว่า "โฟตอน(Photon)" ซึ่งอาจถือว่าเป็นอนุภาคพลังงานแผ่(Radiation)ที่เกิดจากสสารที่มีคุณสมบัติซึ่งจะเคลื่อนที่ผ่านสูญญากาศด้วยความเร็วแสง โดยค่าพลังงานหนึ่งโฟตอน จะเป็นไปตามสูตรดังนี้

$$1 \text{ Photon} = hf$$

เมื่อ

f : ค่าความถี่ของพลังงานแผ่

h : ค่าคงตัวของแพลงค์เท่ากับ 6065×10^{-24} จูลน์-วินาที

ความถี่ของการแผ่พลังงานแม่เหล็กไฟฟ้าของคลื่นบางชนิดเป็นไปตามตารางข้างล่าง

ชนิด	ความยาวคลื่น(m)	ความถี่(Hz)	Quantum energy(ev)
วิทยุ	10^{4} - 10^{3}	$3x10^{4}$ $3x10^{11}$	$1.2x10^{-10}$ - $1.2x10^{-3}$
อินฟาเรด	10^{-3} - 10^{-7}	$3x10^{11}$ - $4x10^{14}$	$1.2x10^{-3}$ - 1.7
visible	$7x10^{-7}$ - $4x10^{7}$	$4x10^{14}$ - $7x10^{14}$	1.7 - 3
ultra	$4x10^{7}$ - 10^{9}	$7x10^{14}$ - $3x10^{17}$	3 - $1.2x10^{3}$
x-ray	10^{9} - 10^{11}	$3x10^{17}$ - $3x10^{19}$	$1.2x10^{3}$ - $1.2x10^{6}$

ตารางที่ 2.1 การแผ่ของพลังงานแม่เหล็กไฟฟ้า

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ขบวนการแผ่มี 2 ลักษณะ

1.เกิดจากอิเลคตรอนและโฮลที่อยู่บริเวณรอยต่อPNซึ่งจะทำให้พลังงานโฟตอนมีค่าประมาณเท่ากับช่วงแถบพลังงาน (Band gap Energy) ซึ่งมีค่าสูงมาก

2.เกิดจากอิเลคตรอนและโฮลเคลื่อนที่มารวมกันบริเวณแทปเลเวล (Trap Level Photon) ซึ่งจะมีพลังงานเท่ากับผลต่างของแถบพลังงาน (Energy Gap) กับค่าบินดิ้ง (Binding)

จากกระบวนการทั้งสองจะได้โฟตอนออกมามีค่าความยาวคลื่นเป็นไปตามสมการดังนี้

$$\lambda = 1240/\Delta E$$

เมื่อ

ΔE : ค่าความแตกต่างของพลังงานมีหน่วยเป็นอิเลคตรอนโวลท์

ดังนั้นถ้าใช้สารต่างชนิดกันค่าของ ΔE จะมีค่าแตกต่างกันด้วย

สารกึ่งตัวนำ	Band Gap Energy	Emission(nm)
Ge	0.66	1800
Si	1.09	1200
GaAs	1.43	880
GaP	2.24	550
GaAs60P40	1.91	650
AlSb	1.60	775
InSb	0.18	6900
SiC	2.2	563

ตารางที่ 2.2 ตารางเปรียบเทียบพลังงานและความยาวคลื่นจากตัวกำเนิดแสงที่สร้างจากสารต่างชนิดกัน

ตารางที่ 2.2 แสดงถึงพลังงานและความยาวคลื่นที่ได้จากการสร้างตัวกำเนิดแสงด้วยสารที่ต่างชนิดกัน จะเห็นได้ว่าตัวกำเนิดแสงที่สร้างจาก GaAs จะให้ความยาวคลื่นอยู่ในช่วงของแสงอินฟาเรดคือ 882 นาโนเมตร

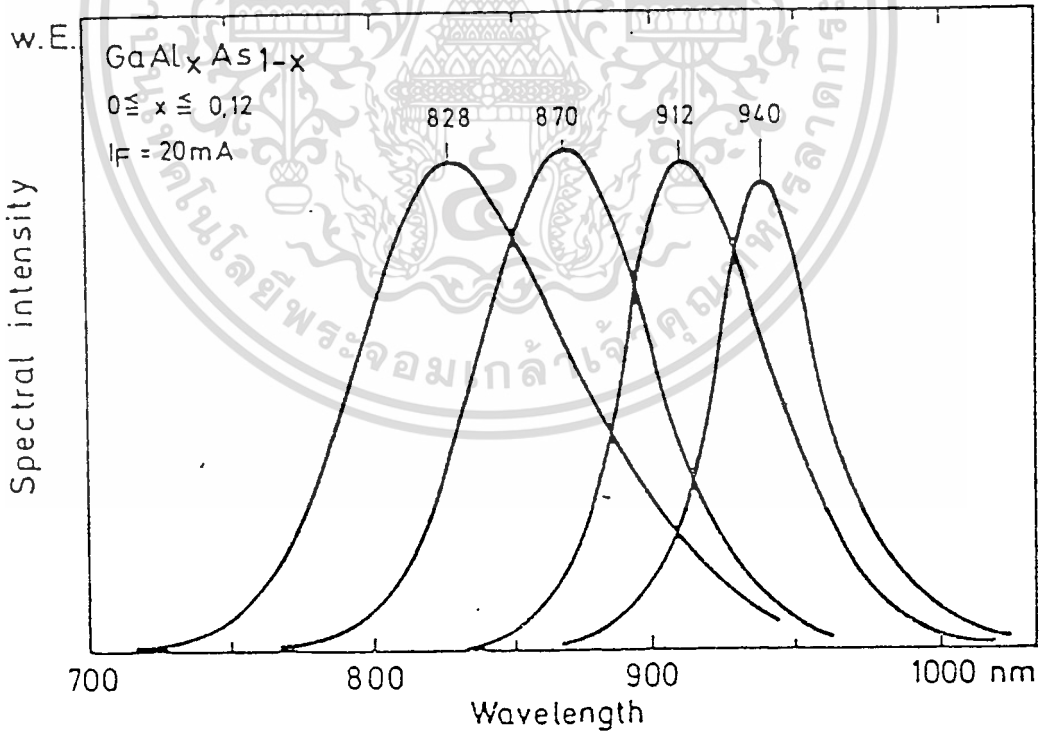
จากรูปที่ 2.1 จะเห็นว่าตัวกำเนิดแสงที่ทำจากสารต่างชนิดกันจะตอบสนองต่อคลื่นต่างกัน สาร GaAs จะให้ค่าความยาวคลื่นดีที่สุดที่ 940 นาโนเมตร แต่ถ้าเพิ่มสาร Al เข้าไปที่สาร GaAs จะทำให้ได้ความยาวคลื่นใหม่ที่ 828 นาโนเมตร ซึ่งโดยทั่วไปช่วงของแสงอินฟาเรดที่น่าสนใจจะ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

อยู่ในช่วง 870-879 นาโนเมตร ดังนั้นค่าความยาวคลื่น 828 นาโนเมตร จึงเป็นความยาวคลื่นที่ใกล้เคียงช่วงของแสงอินฟาเรดซึ่งได้จากสาร GaAlAs ข้อมูลเปรียบเทียบพลังงานระหว่าง GaAs และ GaAlAs แสดงไว้ในตารางที่ 2.3

Parameter	GaAs:Si	GaAlAs:Si	Unit
Maximum mean forward current	150	100	mA
output power at If = 100 mA	12-18	16-26	mW
output power at If = 1.5 mA Pulsed	110-160	210-350	mW
Peak wavelenght	940	880	nm
forward voltage at If = 100 mA	1.4	1.4	V
forward voltage at If = 1.5 mA Pulsed	1.9	3	V

ตารางที่ 2.3 เปรียบเทียบพลังงานที่ได้จากตัวกำเนิดแสงที่สร้างจาก GaAlAs และ GaAs

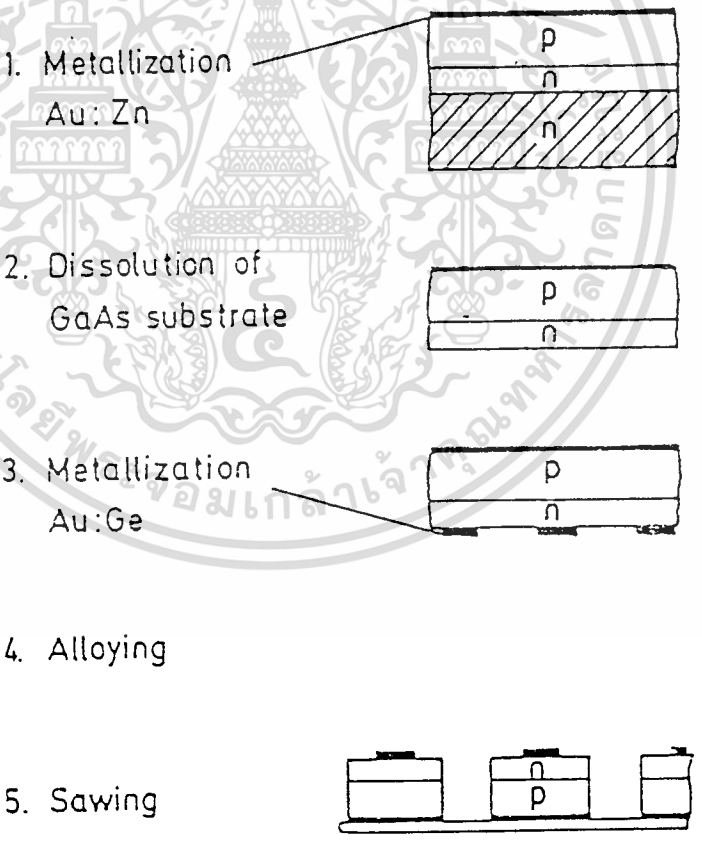


รูปที่ 2.1 กราฟแสดงการเปรียบเทียบความยาวคลื่นที่ได้จากตัวกำเนิดแสงอินฟาเรดจากสารต่างชนิดกัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ประสิทธิภาพของตัวกำเนิดแสงอินฟาเรดที่ทำจากGaAlAs จะมีค่ามากกว่า GaAs ประมาณ 40% ลักษณะชั้นของการโคป (dope) สารที่ใช้ทำตัวกำเนิดแสงอินฟาเรดชนิด GaAlAs แสดงดังรูปที่ 2.2

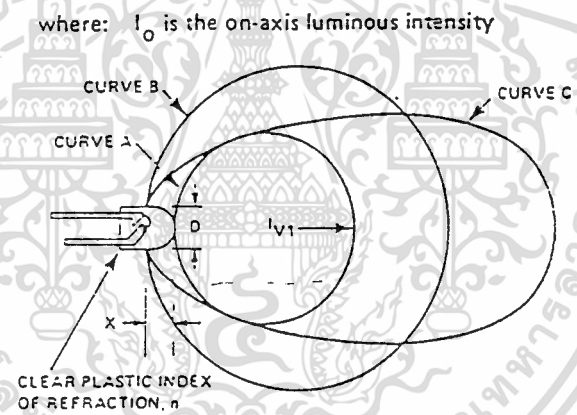
จากตารางที่ 2.3 จะสังเกตุได้ว่า GaAlAs จะมีค่ากระแสทางด้านอินพุตมีค่าน้อยกว่าตัวกำเนิดแสงอินฟาเรดแบบ GaAs ดังนั้นจะช่วยประหยัดพลังงานได้มากกว่าและการจ่ายกำลังและความต่อเนื่องของคลื่นทางเอาต์พตทจะมีผลโดยตรงกับความร้อนที่จะเกิดขึ้นบริเวณรอยต่อPN ซึ่งประสิทธิภาพของตัวจ่ายแสงอินฟาเรดจะลดลงเมื่ออุณหภูมิสูงขึ้นการเปลี่ยนแปลงจะเป็นไปตามอัตราส่วน -1%/°C ในช่วงทำงานเต็มที่ที่จุดอิ่มตัวจะมีความร้อนเกิดขึ้นอย่างมาก



รูปที่ 2.2 ชั้นของการโคปสารตัวกำเนิดแสงชนิด GaAlAs

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ข้อจำกัดในการเพิ่มของสัญญาณทางเอาต์พุตโดยไม่เกี่ยวกับการเปลี่ยนแปลงกระแสอินพุตจะเป็นผลจากความร้อน การแก้ไขทำได้โดยเพิ่มผิวหน้าของตัวสารที่ใช้ทำให้มากขึ้นสาเหตุเพราะมุมตกกระทบรวมมีค่าต่ำมากนั่นเอง ข้อจำกัดสองค่าที่สำคัญของตัวจ่ายแสงอินฟาเรดนั้นคือความร้อนและการอิ่มตัวของสารซึ่งจะแก้ไขได้โดยเพิ่มพื้นที่ของรอยต่อ PN ซึ่งจะเป็นผลให้ความร้อนลดลง การแพร่กระจายของแสงในตัวกำเนิดแสงอินฟาเรดดังรูปที่ 2.3 จะเห็นว่าขึ้นอยู่กับการบังคับเลนส์ยิ่งมีเลนส์กว้างมากก็จะทำให้การแพร่กระจายของแสงได้มาก



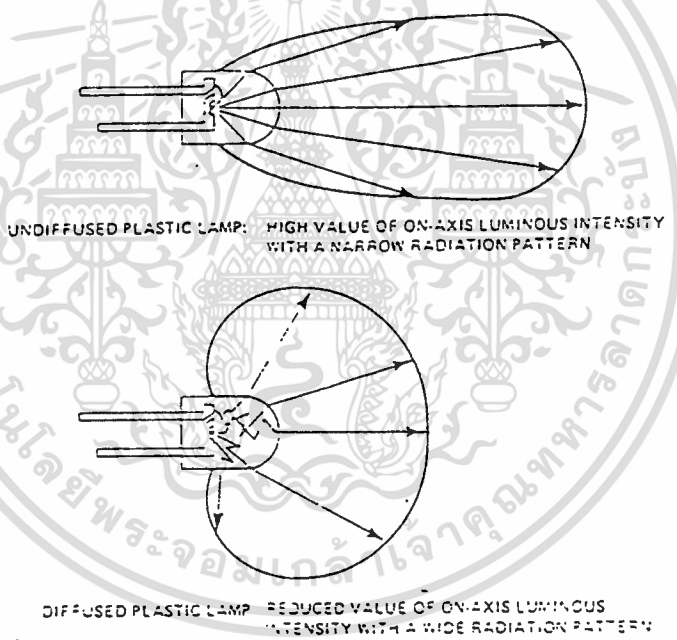
รูปที่ 2.3 การแพร่กระจายของแสงจากเลนส์แบบต่างๆ

การกระจายของแสงออกจากตัวกำเนิดแสงอินฟาเรดจะขึ้นอยู่กับลักษณะโครงสร้างของเลนส์ซึ่งแตกต่างกัน ในเลนส์แบบทรงกลม (Spherical Dome Lens) นั้นจะให้ย่านการทำงาน (Pattern) ที่เหมาะสมกับการส่งแบบจุดต่อจุด (Point to Point) ซึ่งจุดทั้งสองจะตรึงติดอยู่กับท์แน่นอนและจะส่งไปได้ไกลกว่าแบบอื่นๆ ส่วนแบบวงรี (Aspheric Dome Lens) เหมาะกับงานครอบคลุมพื้นที่ แต่ส่งไปได้ไม่ไกลถึงแม้ตัวกำเนิดแสงจะสร้างจากสารชนิดเดียวกันแต่ถ้าเลนส์ต่างกันก็จะได้กราฟการทำงานต่างกันด้วยดังรูปที่ 2.4

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

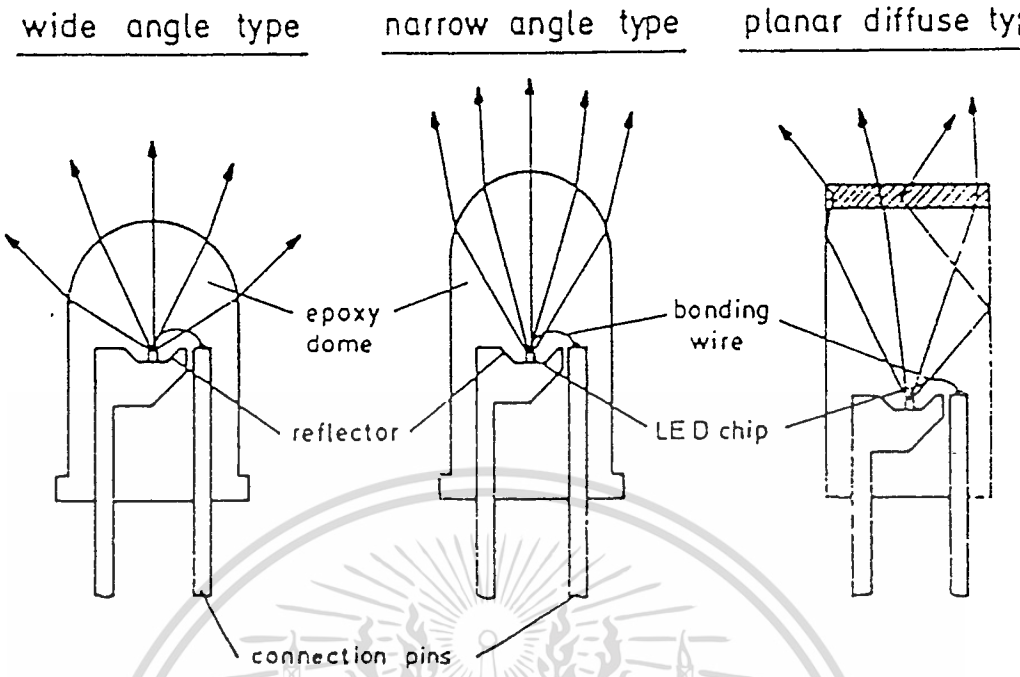
นอกจากนี้ชนิดของตัวกำเนิดแสงอินฟาเรดแต่ละชนิดก็จะให้การกระจายของแสงออกไปตามชนิดที่ใช้ทำตัวกำเนิดแสงเองซึ่งจะมีผลเช่นเดียวกับการจัดเลนส์ของตัวกำเนิดแสงดังรูปที่ 2.5

อย่างไรก็ตามเราสามารถที่จะทำการติดตั้งเพื่อกำหนดให้การกระจายของแสงให้มากที่สุดได้โดยการจัดเลนส์ภายนอกของตัวกำเนิดแสงดังรูปที่ 2.6 ซึ่งเลนส์จะช่วยกระจายแสงออกไปได้มากขึ้น

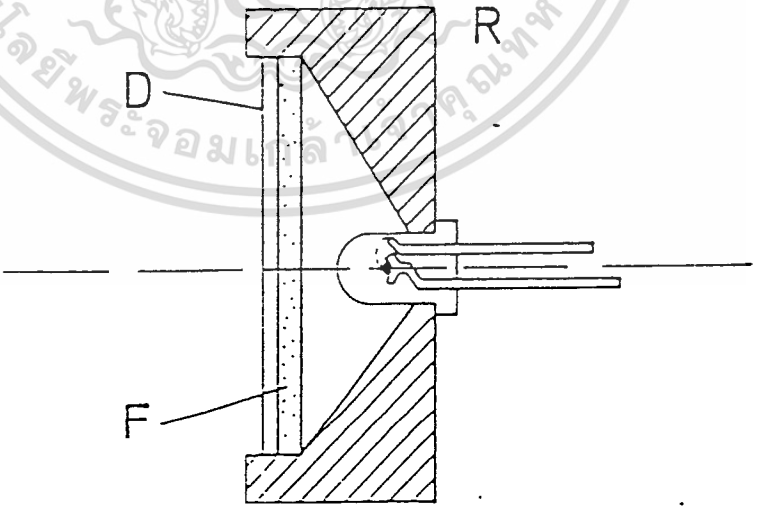


รูปที่ 2.4 การแพร่กระจายของแสงจากการใช้เลนส์ต่างชนิดกัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.5 ลักษณะการแพร่กระจายของแสงในตัวกำเนิดแสง



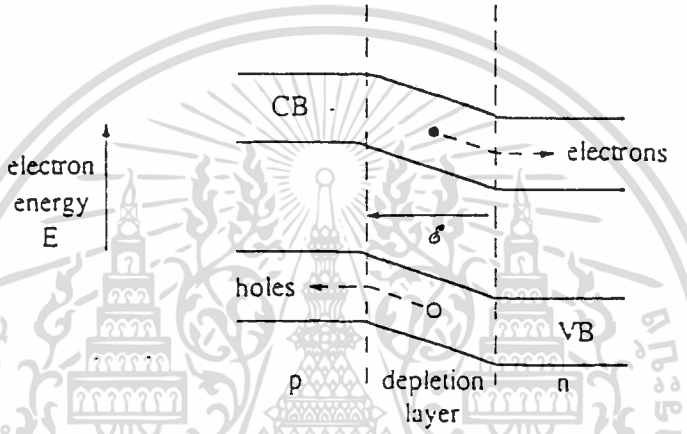
รูปที่ 2.6 การใช้เลนส์ช่วยในการปรับปรุงคุณภาพของแสง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.2 หลักการของตัวรับแสงอินฟาเรด

2.2.1 โฟโตไดโอด

เมื่อสารกึ่งตัวนำได้รับโฟตอนจะทำให้เกิดคู่อิเล็กตรอน-โฮลซึ่งจะทำให้เกิดการเคลื่อนที่ของอิเล็กตรอนทำให้เกิดกระแสไหลขึ้นนั่นคือเมื่อเกิดการแตกตัวของคู่อิเล็กตรอน-โฮล การแตกตัวจะเกิดบริเวณที่มีสนามแม่เหล็กไฟฟ้าหรือบริเวณที่แสงตกกระทบมากๆดังรูปที่ 2.7-2.9



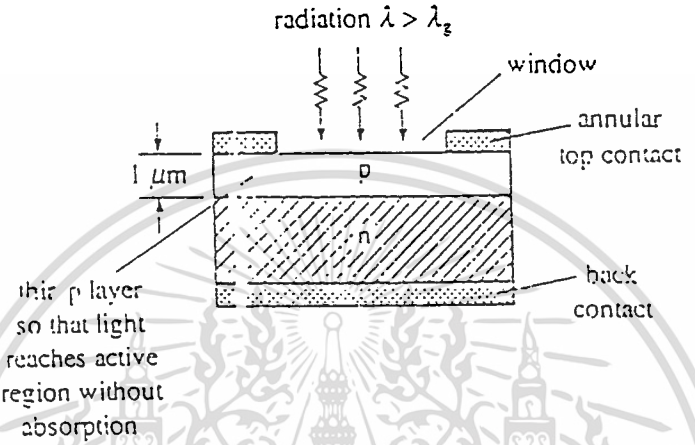
รูปที่ 2.7 การกระจายของสนามแม่เหล็กไฟฟ้าในสารกึ่งตัวนำ

จากรูปที่ 2.7 จะเห็นว่าการกระจายของแสงหรือสนามแม่เหล็กไฟฟ้าในสารกึ่งตัวนำไดโอดมีลักษณะไม่สม่ำเสมอในบริเวณรอยต่อของสาร PN ที่บริเวณดีพลิทชั่นโดยทั่วไปเพื่อเพิ่มประสิทธิภาพการทำงานต่อตัวโฟโตไดโอดควรจะจัดให้โฟตอนส่วนใหญ่ถูกดูดซับในบริเวณรอยต่อหรือดีพลิทชั่นให้มากที่สุด ตรงความลึกที่โฟตอนจะผ่านไปได้ก่อนจะถูกดูดซับพลังงานแสงที่บริเวณผิวหน้าไป

ส่วนความยาวคลื่นที่ยาวขึ้นสามารถที่จะเจาะผ่านผลึกได้ลึกกว่าดังนั้นเพื่อให้โฟโตไดโอดสามารถตอบสนองต่อช่วงความยาวคลื่นที่กว้างจึงควรจะมีชั้น P ที่บางเพื่อให้ความยาวคลื่นสั้นเจาะผ่านเข้ามาได้และมีบริเวณดีพลิทชั่นที่หนามากๆ เพื่อให้รับกระแสของโฟตอนสูงสุดจากโฟตอนที่มีความยาวคลื่นยาวผ่าน ความหนาของบริเวณดีพลิทชั่นขึ้นอยู่กับไบอัสกลับและค่าความต้านทานของบริเวณที่มีการแยกตัวของอิเล็กตรอน-โฮล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โดยปกติแม้ไม่มีการไบอัสกลับก็สามารถเกิดพื้นที่ดีพลีทชั่นได้ ซึ่งสนามที่เกิดขึ้นจากการแพร่พาหะของระหว่างรอยต่อ PN การไบอัสกลับจะช่วยเพิ่มสนามและขยายบริเวณดีพลีทชั่นให้มากขึ้นไปอีกดังรูปที่ 2.9



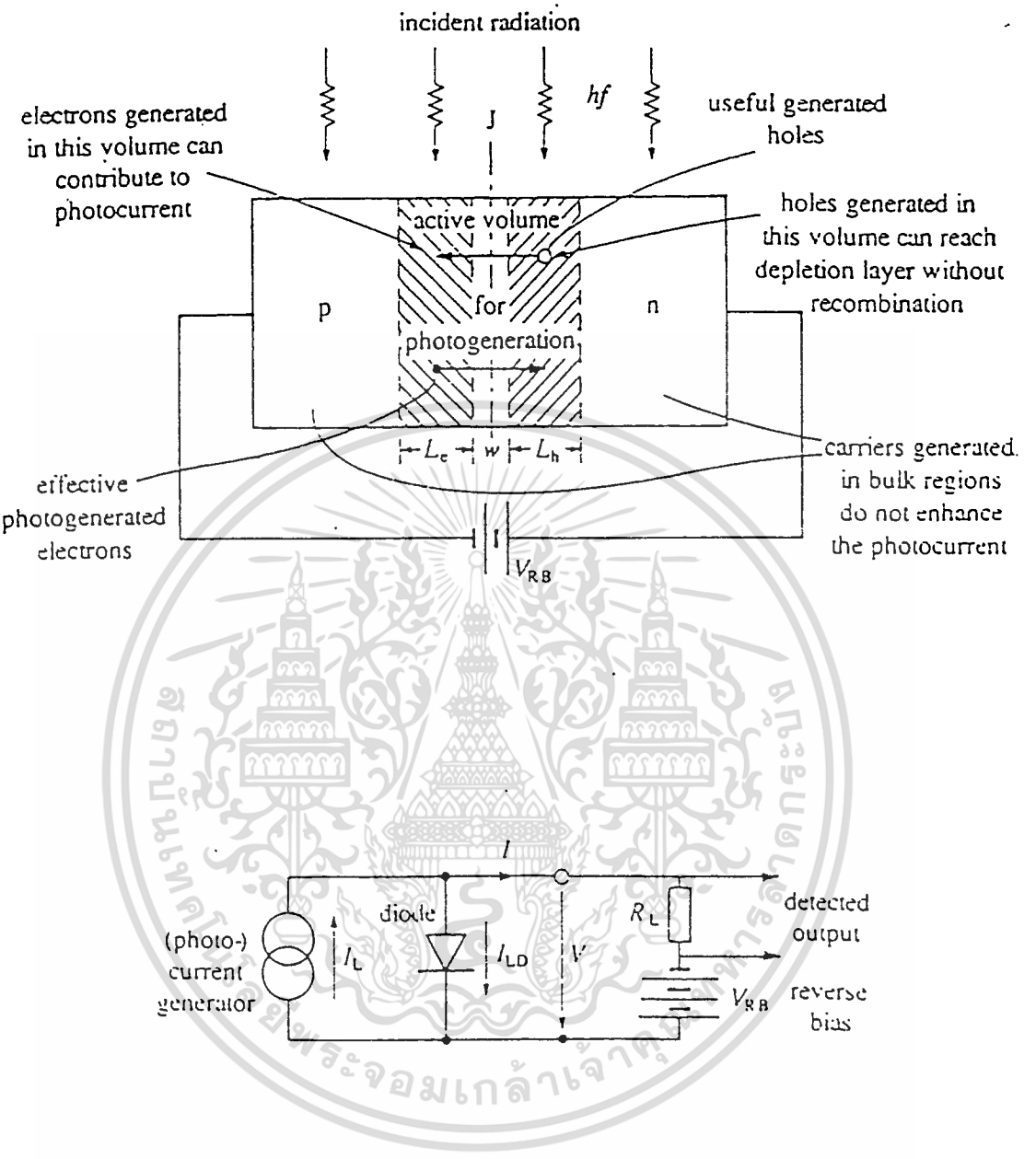
รูปที่ 2.8 ลักษณะโครงสร้างโดยทั่วไปของโฟโตไดโอด

2.2.2 แบบจำลองของวงจรโฟโตไดโอด (equipvalente circuit)

แบบจำลองของวงจรโฟโตไอโอดสามารถแสดงได้ดังรูปที่ 2.9 ค่ากระแส I_L จะขึ้นอยู่กับปริมาณแสงที่ตัวไดโอดได้รับ ในกรณีไม่มีการให้ไบอัส (Zero Bias) กับโฟโตไดโอดกระแส I_L จะทำให้ขั้วอโนด (anode) เป็นบวกเมื่อเทียบกับคาโธด (Cathode) กระแส I_L ส่วนหนึ่งจะไหลกลับผ่านตัวไดโอดและอีกส่วนจะไหลผ่านตัวต้านทานถ้าโหลด (Load) มีค่าสูงมากกระแสส่วนมากจะไหลผ่านตัวไดโอด

การทำงานโดยไม่ให้การไบอัสเรียกว่าโหมดโฟโตโวลตาอิก (Photo Voltaic mode) การทำงานในโหมดโฟโตโวลตาอิกสามารถทำงานได้ทั้งแบบเชิงเส้นและแบบลอกาลิทึมซึ่งขึ้นอยู่กับค่าโหลดตัวต้านทาน การทำงานของโหมดลอกาลิทึมจะเกิดขึ้นเมื่อโหลดมีค่าสูงมากๆ($>10^{11}$) ส่วนการทำงานแบบเชิงเส้นจะเกิดเมื่อโหลดมีค่าความต้านทานต่ำเมื่อเทียบกับความต้านทานไดนามิกส์ของโฟโตไดโอด ค่าศักดาสูงสุดในการทำงานแบบไม่ให้ไบอัสคือ V_L ประมาณ 100 มิลลิโวลท์ ขึ้นอยู่กับความละเอียดที่ต้องการ ถ้าค่า R_L สูงกว่าจะทำให้มีความไวมากกว่าแต่ช่วงการตอบสนองเชิงเส้นจะลดลง ค่า R_L สูงสุดจะอยู่ในช่วง 5 เมกกะโอห์ม ถึง 550 เมกกะโอห์ม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.9 วงจรจำลองของโฟโตไดโอดและการขยายบริเวณดีพลีทชั่นให้มากขึ้น

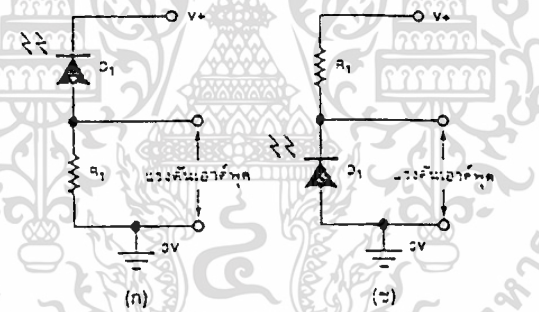
การทำงานแบบให้ไบอัสกลับเรียกว่าโหมดกระแสโฟโต (Photocurrent Mode) ซึ่งมีข้อดีกว่าแบบโฟโตโวลท์ตาอิก คือ

1. มีความเร็วสูง
2. เสถียรภาพดี
3. ช่วงไดนามิกส์สูงกว่า
4. สัมประสิทธ์ทางอุณหภูมิสูงกว่า

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5. การตอบสนองต่อแสงดีกว่า

จากลักษณะของโฟโตไดโอดที่กล่าวมาแล้วนั้นจะเห็นว่าเมื่อใดที่รอยต่อ PN ได้รับการไบอัสกลับจะเกิดกระแสรั่วไหลย้อนกลับ และอิมพีแดนซ์ของรอยต่อ PN นี้มีความไวต่อแสงมากเป็นพิเศษ คือจะมีอิมพีแดนซ์สูงเมื่ออยู่ในที่มืดและมีอิมพีแดนซ์ต่ำเมื่ออยู่ในที่สว่าง ไดโอดทั่วไปนั้นจะถูกหุ้มรอยต่อนี้ไว้ด้วยวัสดุทึบแสงเพื่อป้องกันไม่ให้เกิดปรากฏการณ์นี้ขึ้น แต่สำหรับโฟโตไดโอดเป็นไดโอดซึ่งถูกผลิตขึ้นมาเพื่อให้เกิดปรากฏการณ์นี้โดยเฉพาะ ดังนั้นรอยต่อจึงต้องหุ้มด้วยวัสดุที่แสงสามารถผ่านได้ดีที่สุดไดโอดชนิดนี้มีสองแบบคือ ชนิดที่ตอบสนองต่อแสงผ่านอินฟาเรด ในการนำไปใช้งานโฟโตไดโอดจะถูกต่อในลักษณะได้รับการไบอัสกลับสำหรับแรงดันเอาต์พุต เป็นแรงดันที่ตกคร่อมตัวต้านทานโหลดที่ต่ออนุกรมกับโฟโตไดโอดและกราวด์ ดังรูปที่ 2.10 โฟโตไดโอดจะถูกนำมาประยุกต์ใช้งานที่เกี่ยวกับสัญญาณไฟสลับที่มีการเปลี่ยนแปลงเร็ว ๆ สำหรับการประยุกต์ใช้โฟโตไอโอดชนิดที่ตอบสนองต่อแสงอินฟาเรด เช่นการใช้ในวงจรรีโมทคอนโทรลวงจรสัญญาณเตือนต่างๆ ที่ใช้แสงอินฟาเรดในการควบคุม



รูปที่ 2.10 วงจรจำลองของโฟโตไดโอดและการขยายบริเวณดีพลีทชั่นให้มากขึ้น

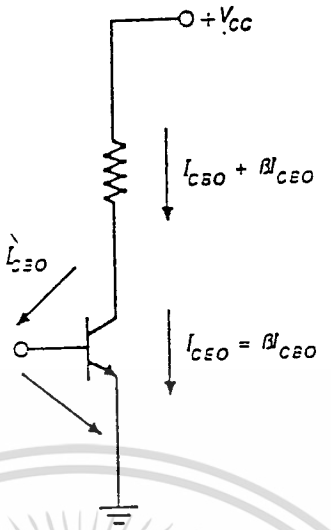
2.2.3 โฟโตทรานซิสเตอร์

หลักการทำงานเบื้องต้นจะเหมือนกับทรานซิสเตอร์ทั่วไปแต่โฟโตทรานสซีสเตอร์จะไม่มีขาเบส ซึ่งกระแสเบสจะได้รับจากการเปลี่ยนแปลงของแสงที่ตกกระทบบริเวณรอยต่อของสาร PN ตามที่กล่าวมาแล้ว วงจรการทำงานของโฟโตทรานสซีสเตอร์โดยทั่วไปแสดงในรูปที่ 2.11

สมมุติกำหนดให้ I_λ เป็นกระแสเบสที่เกิดจากการได้รับพลังงานแสงผลลัพธ์ของกระแสอิเลคตรอนจะเป็นดังนี้

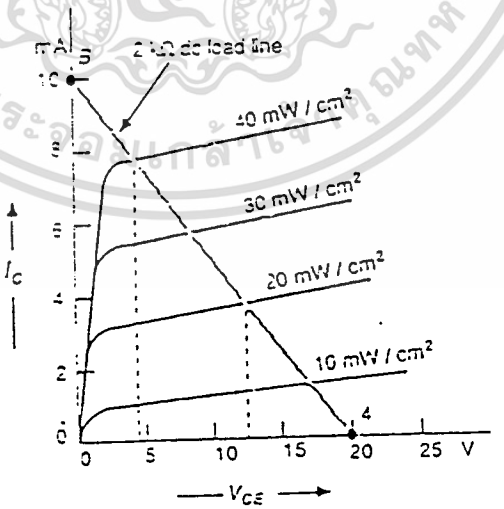
$$I_c = hfe/I_\lambda$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.11 วงจรของโฟโตทรานซิสเตอร์

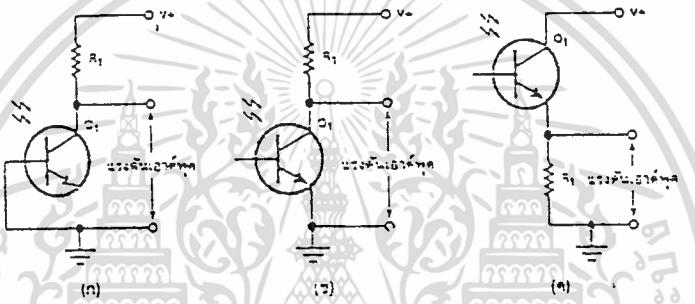
กราฟคุณสมบัติของโฟโตทรานซิสเตอร์จะแสดงได้ดังรูปที่ 2.12 สำหรับความสัมพันธ์ระหว่างความเข้มของแสงและกระแส คือถ้าเพิ่มความเข้มของแสงมากขึ้นกระแสคอลเลคเตอร์จะมีค่าเพิ่มขึ้นนั่นคือกระแส I_c จะเปลี่ยนไปเป็นสัดส่วนโดยตรงกับความเข้มของแสง



รูปที่ 2.12 กราฟแสดงคุณสมบัติของโฟโตทรานซิสเตอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การใช้งานโฟโตทรานซิสเตอร์มีอยู่ด้วยกันดังนี้ การต่อใช้งานพื้นฐานแสดงในรูปที่ 2.13 โดยรูปที่ 2.13ก ขาเบสของทรานซิสเตอร์ถูกต่อลงกราวด์ ดังนั้นทรานซิสเตอร์จะทำงานเหมือนกับโฟโตไดโอดทุกประการ ส่วนรูปที่ 2.13ข และ 2.13ค ขาเบสของทรานซิสเตอร์ถูกปล่อยลอยไว้และเมื่อใดที่ทรานซิสเตอร์ได้รับแสงก็จะมีกระแสไหลผ่านรอยต่อเบส-คอลเลคเตอร์ของทรานซิสเตอร์ ซึ่งจะทำให้กระแสที่ผ่านจากคอลเลคเตอร์มายังอิมิตเตอร์ของทรานซิสเตอร์มีปริมาณเพิ่มขึ้นอย่างมาก กระแสนี้จะทำให้ได้แรงดันเอาต์พุตที่ตกคร่อมตัวต้านทาน R_1 ที่ต่ออนุกรมอยู่มีค่าเพิ่มขึ้น



รูปที่ 2.13 การต่อใช้งานโฟโตทรานซิสเตอร์ในลักษณะต่างๆ

เมื่อเปรียบเทียบกับโฟโตไดโอดโฟโตทรานซิสเตอร์จะมีความไวต่อแสงมากกว่าประมาณ 100 เท่าแต่ในด้านความถี่ใช้งานสูงสุดสำหรับโฟโตทรานซิสเตอร์จะใช้งานได้ที่ความถี่ต่ำกว่าโฟโตทรานไดโอดหลายเท่า ในการใช้งานอาจจะต่อตัวต้านทานปรับค่าได้ระหว่างขาเบสและขาอิมิตเตอร์ของโฟโตทรานซิสเตอร์ด้วย เพื่อให้สามารถเลือกได้ว่าต้องการใช้กับงานที่มีความไวต่อแสงมากๆหรือต้องการใช้ที่มีความถี่สูงๆ โดยเมื่อความต้านทานปรับค่าได้ลัดวงจรก็จะทำหน้าที่เป็นโฟโตไดโอดไป

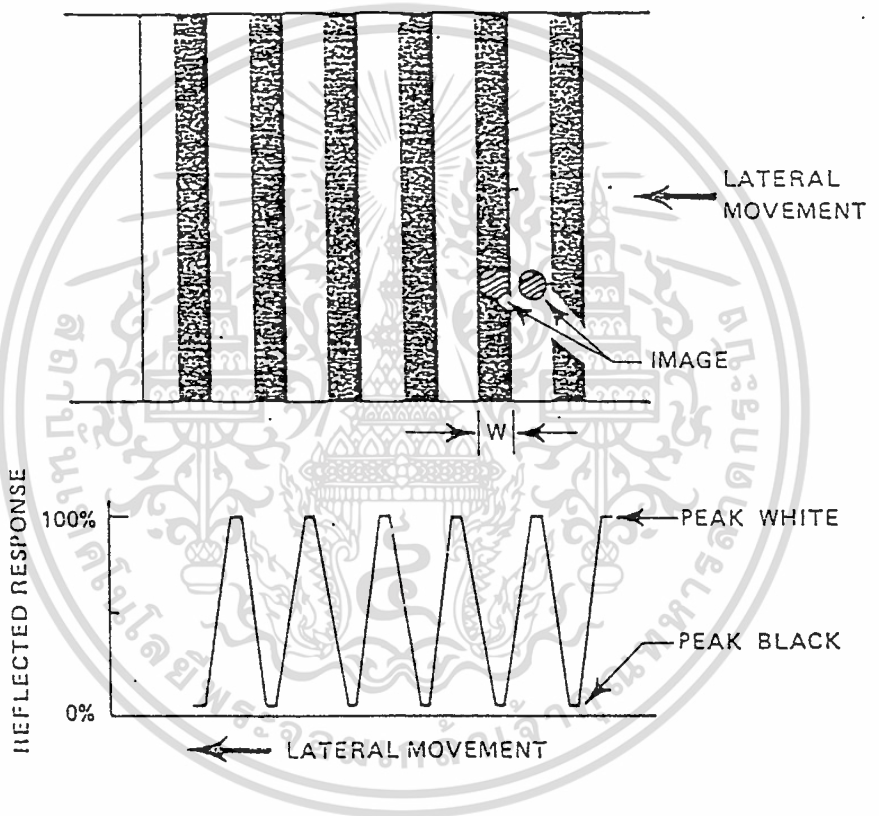
2.3 ผลตอบสนองของคลื่นแสงอินฟาเรดเมื่อวิ่งผ่านแถบสีขาว-ดำ

พิจารณาการรับส่งการสะท้อนของแสงอินฟาเรดเมื่อวิ่งผ่านสีดำดังแสดงในรูปที่ 2.14 จะเห็นว่าผลตอบสนองของแสงระหว่างช่องว่างสีขาวและสีดำซึ่งจะให้สัญญาณออกมาเป็นอนาลอก

จากรูปที่ 2.14 จะเห็นว่าช่องว่างขาวและดำเท่ากันดังนั้นจะมีผลตอบสนองของแสงดังกราฟด้านล่างซึ่งเราใช้ตัวกำเนิดแสงเป็นแบบวงกลมที่มีขนาดเส้นผ่าศูนย์กลาง d มีขนาดน้อยกว่า

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

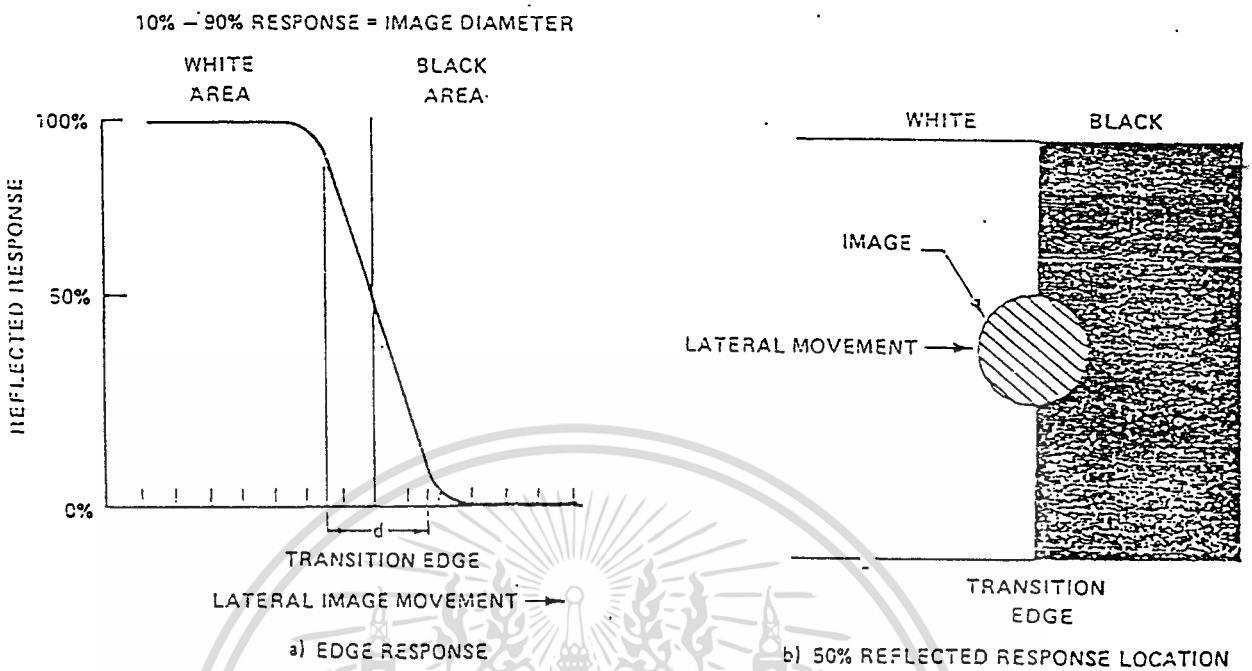
ขนาดของช่องว่าง เมื่อแถบสีวิ่งผ่านตามทิศทางตามลูกศรจะทำให้สัญญาณตอบสนองของสัญญาณที่ตำแหน่งต่างๆคิดเป็นเปอร์เซ็นท์ต่างกันดังแสดงในรูป 2.15 ความแตกต่างถ้าค่าสูงสุดและต่ำสุดคือค่าที่เราต้องการนำไปใช้ในการควบคุม



รูปที่ 2.14 การสะท้อนแสงที่ตกกระทบแถบขาว-ดำ

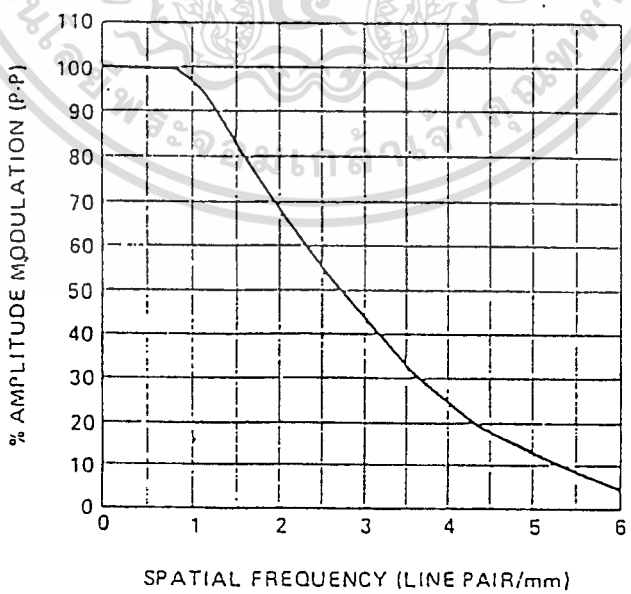
เมื่อตัวกำเนิดแสงวิ่งผ่านมาถึงตำแหน่งที่เป็นขาวและดำ ความชันของสัญญาณก็จะเปลี่ยนแปลงไปเป็นต่างๆ เกิดเป็นความชันดังรูป 2.16 นั่นคือค่าที่เปลี่ยนแปลงจากสูงและต่ำจะมีค่าเวลามากซึ่งไม่เป็นที่ต้องการ ความต้องการคือการเปลี่ยนระดับของสัญญาณจากต่ำสุดไปสูงสุดต้องใช้เวลาน้อยที่สุดเท่าที่จะเป็นไปได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



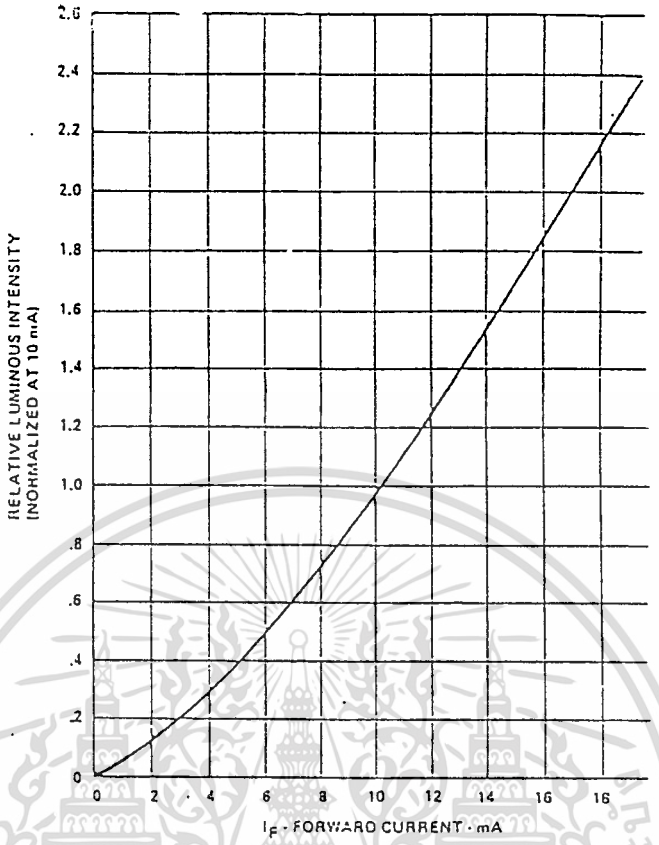
รูปที่ 2.15 ลักษณะของสัญญาณที่ได้จากการสะท้อนของแสงเมื่อผ่านแถบสี

เมื่อคำนวณผลตอบสนองของสัญญาณที่ตำแหน่งต่างๆก้นจะได้กราฟดังรูปที่ 2.16



รูปที่ 2.16 ผลการตอบสนองที่ตำแหน่งต่างๆ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.17 ความสัมพันธ์ของกระแสที่ไหลผ่านกับแสงที่จ่ายออกของโฟโต

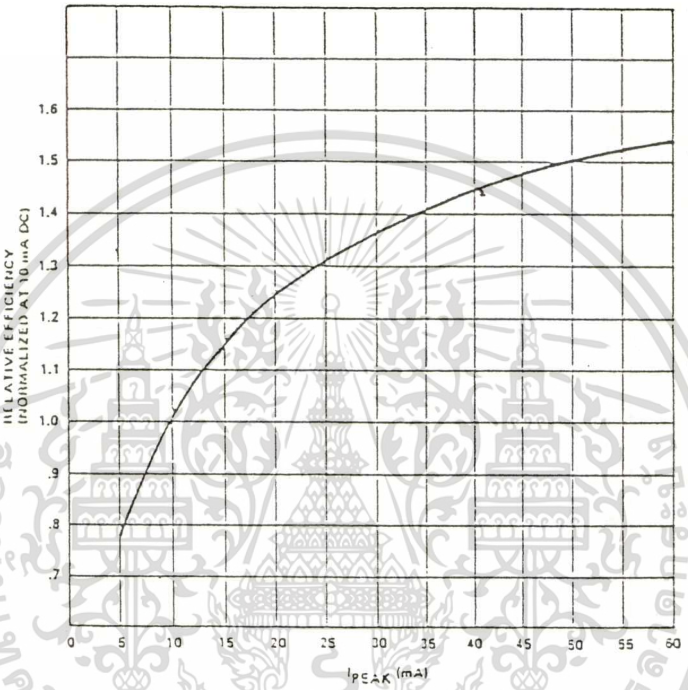
ตัวกำเนิดแสงอินฟาเรดมีการเปลี่ยนแปลงค่าความต้านทานทางพลศาสตร์(dynamic) เมื่อขณะทำงานน้อยมากโดยปกติจะใช้กระแสในการขับโดยต่ออนุกรมเข้าไปโดยตรงการต่อความต้านทานอนุกรมเพื่อจำกัดกระแสเป็นสิ่งจำเป็น ทั้งนี้จะขึ้นอยู่กับว่า $V_{bat} >> V_f$; $R >> R_s$ มากๆก็ควรจะต่อความต้านทานอนุกรมเพื่อจำกัดกระแสไว้ ความเข้มของแสงจะขึ้นอยู่กับการเปลี่ยนแปลงของกระแสที่ให้แก่ตัวกำเนิดแสงดังแสดงในกราฟรูป 2.17

เนื่องจากการเลื่อนของกระดาษคำตอบจะทำให้ได้ลักษณะรูปคลื่นดังแสดงในรูปที่กล่าวมาแล้วนั้นจะเห็นว่า รูปคลื่นนั้นเป็นลักษณะคล้ายรูปคลื่นไซน์(sine wave)แต่สัญญาณที่จะนำไปใช้งานจริงนั้นต้องการสัญญาณเพียง 0 กับ 1 เท่านั้นจำจะต้องแก้ไขดังกล่าวมาแล้ว เมื่อใช้ตัวตรวจจับแสงแล้ววัดสัญญาณที่ตรวจจับได้เป็นสัญญาณไฟฟ้า เปรียบเทียบกับเวลาจะได้ดังรูปที่ 2.19

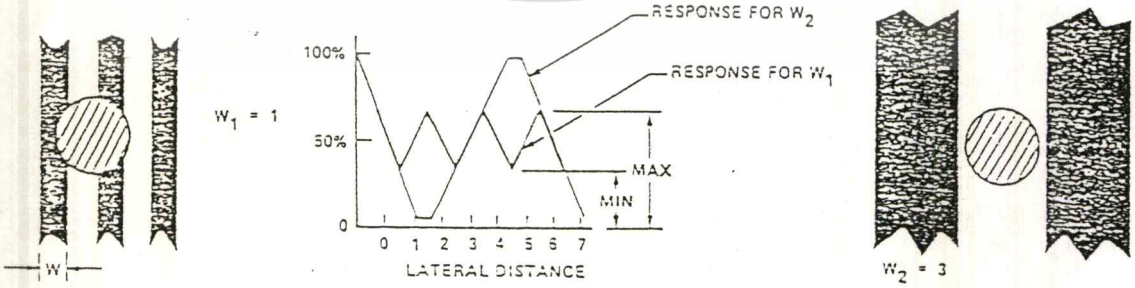
ตามที่กล่าวมาแล้วว่าการนำสัญญาณที่ตรวจจับไปใช้งานจริงสัญญาณที่ได้จะเป็นสัญญาณแบบดิจิตอลที่เป็น 1 กับ 0 เท่านั้น จากนั้นจึงจะใช้เป็นสัญญาณป้อนให้กับวงจรภาคอื่นต่อไปแต่เนื่องจากสัญญาณที่ได้จากตัวตรวจจับไม่ได้เป็นสัญญาณดิจิตอลตามต้องการจำเป็นที่จะต้องมีวงจรมาทำให้สัญญาณที่ได้เป็นสัญญาณดิจิตอลจากวงจรเปรียบเทียบโดยใช้ไอซีออปแอมมาช่วยแปลงสัญญาณที่ได้ให้เป็นสัญญาณดิจิตอล ซึ่งจะเห็นได้ว่าสัญญาณอินพุตจะมีลักษณะคล้าย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กับสัญญาณดิจิตอล ในวงจรเปรียบเทียบจะมีสัญญาณเอาต์พุตที่เป็นทั้งบวกและลบแต่วงจรที่ต้องการใช้เพียง 0 กับ 1 เท่านั้นดังนั้นเพื่อให้ได้สัญญาณเอาต์พุตตามต้องการจะจ่ายแรงดันให้กับออปแอมเป็น +Vcc และ 0 V เท่านั้นก็จะได้สัญญาณเอาต์พุตตามต้องการ



รูปที่ 2.18 ความสัมพันธ์ของแสงกับกระแสที่ไหลผ่านโฟโต



รูปที่ 2.19 เปรียบเทียบขนาดของตัวกำเนิดแสงกับตัวสะท้อนแสงที่มีผลต่อสัญญาณต่างกัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้
038033

บทที่ 3
ไมโครคอนโทรลเลอร์

3.1 โครงสร้างและคุณสมบัติของไมโครคอนโทรลเลอร์ (MCS-51)

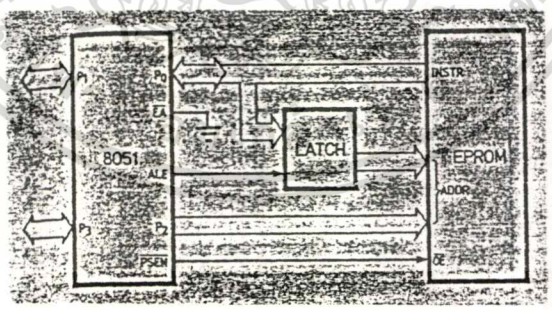
3.1.1 การจัดหน่วยความจำ

ในระบบของ MCS-51 จะมีการแบ่งหน่วยความจำเหมือนกับซีพียูทั่วไปคือ จะแบ่งเป็น 2 ลักษณะตามชนิดของข้อมูลที่เก็บดังนี้

-หน่วยความจำข้อมูล (data memory)

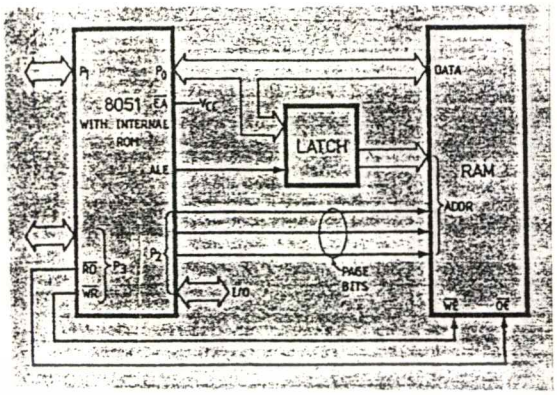
-หน่วยความจำโปรแกรม (program memory)

ถ้าจะพูดกันง่ายๆ หน่วยความจำข้อมูลหมายถึงหน่วยความจำส่วนที่เป็นแรม (RAM) ซึ่งเราสามารถอ่านและเขียนข้อมูลเปลี่ยนแปลงได้ตลอดเวลา แต่ไม่สามารถรันโปรแกรมบนหน่วยความจำส่วนนี้ได้ ส่วนหน่วยความจำโปรแกรมจะหมายถึงหน่วยความจำที่อ่านได้อย่างเดียว (ROM) โดยหน่วยความจำทั้ง 2 ประเภทนี้จะถูกแยกออกจากกันด้วยคำสั่งทางซอฟต์แวร์และลักษณะการติดต่อทางฮาร์ดแวร์ด้วย ดังรูปที่ 3.1 และ 3.2



รูปที่ 3.1 การเชื่อมต่อกับหน่วยความจำโปรแกรม

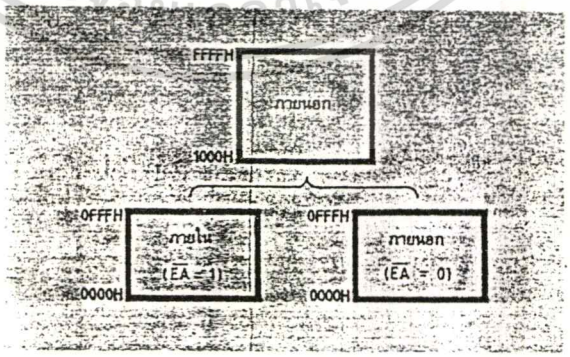
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.2 การเชื่อมต่อกับหน่วยความจำข้อมูล

1. หน่วยความจำโปรแกรม

ใน MCS-51 จะแบ่งหน่วยความจำออกเป็น 2 ส่วนคือ หน่วยความจำโปรแกรมภายในและหน่วยความจำโปรแกรมภายนอก หน่วยความจำโปรแกรมภายในก็คือหน่วยความจำประเภท ROM หรือ EPROM ที่อยู่ภายในตัว MCS-51 เบอร์ 8051 หรือ 8751 ตามลำดับ ซึ่งมีขนาด 4 กิโลไบต์ ส่วนหน่วยความภายนอก MCS-51 โดยอาจจะเป็นการต่อเพื่อขยายหน่วยความจำเพิ่มเนื่องจากหน่วยความจำภายในไม่พอ โดย MCS-51 สามารถเลือกให้รันโปรแกรมบนหน่วยความจำภายในหรือภายนอกก็ได้ โดยควบคุมที่ขา EA (ขา 31)แต่สำหรับเบอร์ 8031 จะต้องต่อขา EA ลงกราวด์เสมอ แสดงในรูปที่ 3.3

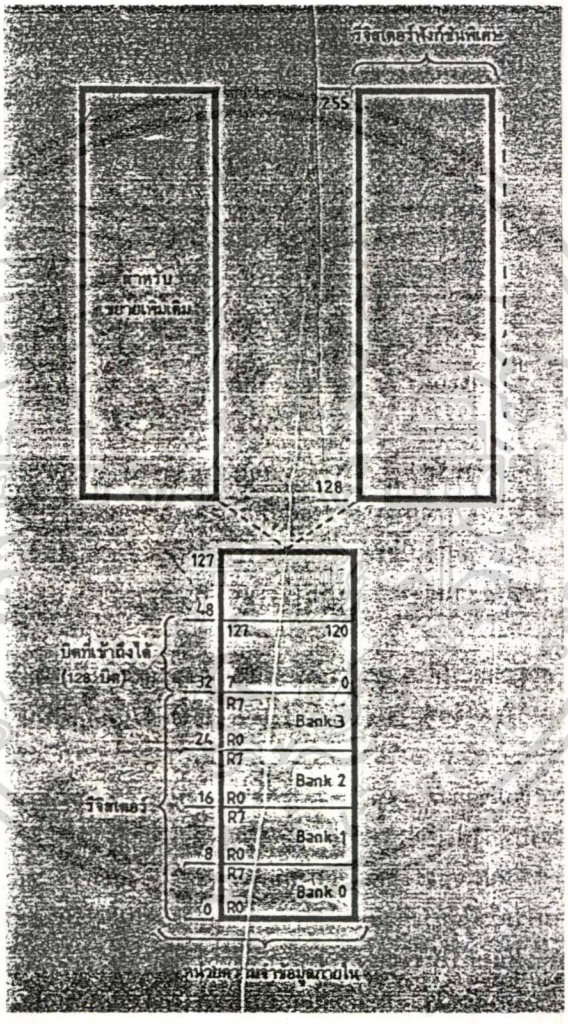


รูปที่ 3.3 การจัดพื้นที่หน่วยความจำโปรแกรมภายในและภายนอก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2. หน่วยความจำข้อมูล

หน่วยความจำข้อมูลของ MCS-51 ก็ถูกแบ่งเป็น 2 ส่วนเหมือนกันคือ หน่วยความจำข้อมูลภายในและภายนอก โดยหน่วยความจำภายนอกจะเข้าถึงได้ด้วยคำสั่ง MOVX เท่านั้น สำหรับหน่วยความจำภายในของ 8031/8051 และ 8751 จะมีทั้งหมด 256 ไบต์ โดยแบ่งเป็น 128 ไบต์ส่วนบน ซึ่งใช้เป็นที่อยู่ของรีจิสเตอร์ฟังก์ชันพิเศษ และอีก 128 ไบต์ส่วนล่าง ซึ่งจะถูกใช้งานทั่วไป ดังรูปที่ 3.4 และรูปที่ 3.5

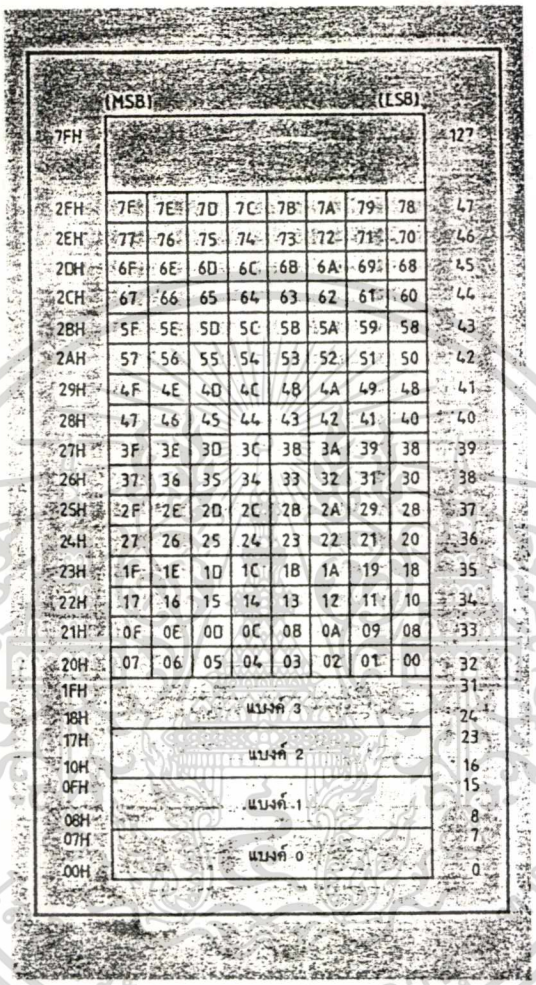


รูปที่ 3.4 การจัดพื้นที่หน่วยความจำข้อมูล

จากรูปที่ 3.5 จะเห็นว่าพื้นที่หน่วยความจำภายใน 128 ไบต์ส่วนล่าง (00H-7FH) จะถูกแบ่งเป็น 3 ส่วนคือ ส่วนของรีจิสเตอร์แบ่งค์ (00H-1FH),ส่วนพิเศษที่สามารถเข้าถึงตำแหน่งบิตได้โดยตรง (20H-2FH) และส่วนที่ใช้งานทั่วไป (30H-7FH)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ส่วนรีจีสเตอร์แบ่งค์มีทั้งหมด 4 แบ่งค์ แต่เราสามารถใช้ได้ครั้งละแบ่งค์เท่านั้น การเลือกใช้แบ่งค์ไหนขึ้นอยู่ที่เรากำหนดค่าในรีจีสเตอร์ PSW



รูปที่ 3.5 พื้นที่การกำหนดตำแหน่งบิตของหน่วยความจำข้อมูลภายในจากรูปที่ 3.4

คำสั่งที่ใช้ในการติดต่อกับหน่วยความจำข้อมูล ซึ่งจะแบ่งลักษณะการกำหนดเลขที่อยู่ของหน่วยความจำข้อมูลได้ 4 โหมดด้วยกันคือ

-**โหมดการกำหนดเลขที่อยู่รีจีสเตอร์** จะเป็นการติดต่อกับข้อมูลที่อยู่ในรีจีสเตอร์โดยตรง ตัวอย่างเช่น

```
MOV A,R0 ; ย้ายค่าใน R0 ไปไว้ที่ ACC
ADD A,R3 ;บวกค่าใน ACC กับค่าใน R3 แล้วนำผลลัพธ์ไปไว้ใน ACC
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

-โหมดกำหนดเลขที่อยู่โดยตรง จะเป็นการติดต่อกับข้อมูลที่ตำแหน่งแอดเดรสที่กำหนดโดยตรง ซึ่งจะต้องบริเวณตำแหน่งของแรมภายใน 256 ไบต์ ตัวอย่างเช่น

MOV A,@ R0 ; ย้ายค่าข้อมูลในหน่วยความจำที่ถูก R0 ชี้ไว้ใน ACC

ADD A, @ R1 ; บวกค่าข้อมูลใน ACC กับค่าข้อมูลในหน่วยความจำที่ถูก R1 ชี้

-โหมดการกำหนดเลขที่อยู่ข้อมูลโดยทันที จะเป็นการติดต่อกับข้อมูลคงที่ ตัวอย่างเช่น

MOV A,40H ; ย้ายข้อมูลที่ตำแหน่ง 40H ไปไว้ที่ ACC

ADD A,41H ; บวกค่าข้อมูลใน ACC กับข้อมูลในตำแหน่ง 41H

-โหมดการกำหนดเลขที่อยู่โดยอ้อม จะเป็นการติดต่อกับข้อมูลที่ที่ใช้ค่าในตัวรีจิสเตอร์ R0 หรือ R1 เป็นตัวชี้ตำแหน่ง ตัวอย่างเช่น

MOV A,# 22H ; นำค่า 22H ไปไว้ใน ACC

ADD A,# 41H ; นำค่า 41H บวกเข้ากับค่า ACC

3.1.2 ตัวจับเวลา / ตัวนับ (TIMER / COUNTER)

ในเบอร์ 8031/8051 และ 8751 จะมีตัวจับเวลา / ตัวนับขนาด 16 บิต จำนวน 2 ตัว คือ ไทเมอร์ / เคาน์เตอร์ 0 และไทเมอร์ / เคาน์เตอร์ 1 ส่วนเบอร์ 8032/8052 จะมีเพิ่มอีก 1 ตัว โดยแต่ละตัวสามารถที่จะกำหนดให้ทำงานเป็นตัวจับเวลาหรือตัวนับได้โดยการเซตหรือเครียร์บิต C/T ที่ตัวรีจิสเตอร์ควบคุม TMOD ซึ่งอยู่ในกลุ่มรีจิสเตอร์ฟังก์ชันพิเศษ

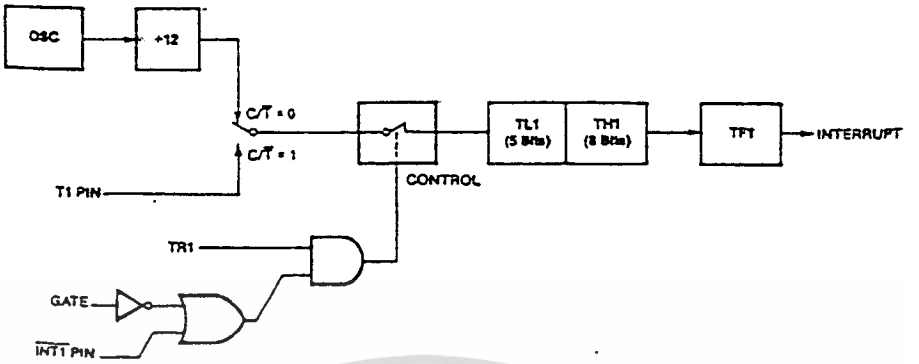
ในการกำหนดให้ทำงานเป็นตัวจับเวลา ตัวรีจิสเตอร์ TH1 และ TL1 ซึ่งทำหน้าที่เป็นตัวเก็บค่าจำนวนพัลส์ที่เข้ามาจะเพิ่มค่าทุกๆ แมชีนไซเคิล โดยแต่ละแมชีนไซเคิลจะประกอบด้วย 12 คาบออสซิเลเตอร์ ดังนั้นอัตราการนับแต่ละครั้งจะใช้เวลาเท่ากับ 1/12 ของความถี่ออสซิเลเตอร์ ซึ่งส่วนใหญ่จะใช้ในงานอินเตอร์รัพต์ RTC (Real Time Clock)

และถ้าให้ทำงานเป็นตัวนับ รีจิสเตอร์ตัวนับจะเพิ่มค่าทุกครั้งที่มีการเปลี่ยนแปลงสถานะจาก "1" เป็น "0" ที่ขา T0 หรือขา T1 โดยอัตราความถี่สูงสุดที่สามารถนับได้ต้องไม่เกิน 1/24 ของความถี่ออสซิเลเตอร์

ตัวจับเวลา / ตัวนับสามารถโปรแกรมให้มันทำงานได้ต่างกันถึง 4 โหมด โดยการตั้งค่าในรีจิสเตอร์ TMOD ซึ่งจะกล่าวถึงการทำงานของแต่ละโหมดดังนี้

โหมด 0 รีจิสเตอร์ตัวนับจะถูกกำหนดให้มี 13 บิต ประกอบด้วยรีจิสเตอร์ TH1 8 บิต และ TL1 อีก 5 บิต อันดับต่ำ ซึ่งสามารถกำหนดให้เป็นตัวจับเวลาหรือตัวนับได้โดยการเซตหรือเครียร์ที่บิต C/T ในตัวรีจิสเตอร์ TMOD การทำงานของรีจิสเตอร์ตัวนับจะนับขึ้นครั้งละ 1 เมื่อมีสัญญาณเข้ามา 1 ลูก และเมื่อนับจนเป็น "1" หมดทุกบิต ก็จะกลับมาเป็น "0" หมดทุกบิตใหม่ ซึ่งจะเป็นการเกิดโอเวอร์โฟลว์ (Overflow) ไปทดแฟลกอินเตอร์รัพต์ TF1 ให้เป็น "1" ลักษณะวงจรเป็นดังรูปที่ 3.6

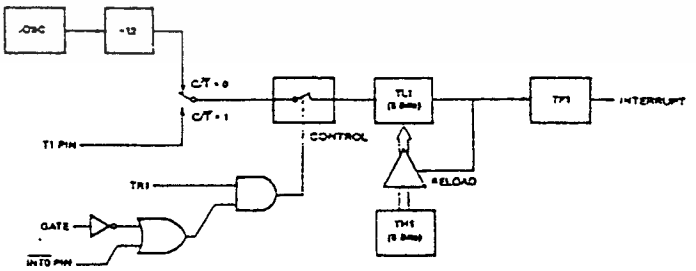
เอกสารนี้สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.6 แสดงวงจรการทำงานของตัวจับเวลา/ตัวนับที่ 1 ในโหมด 0 และโหมด 1

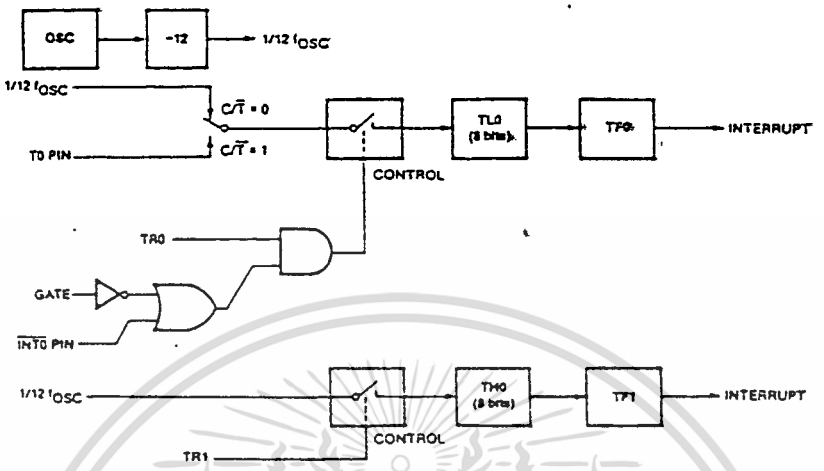
โหมด 1 การทำงานจะเหมือนกับโหมด 0 ทุกอย่างยกเว้นรีจิสเตอร์ตัวนับจะเป็นขนาด 16 บิต

โหมด 2 จะใช้รีจิสเตอร์ TL1 เป็นตัวนับเพียงตัวเดียวและเมื่อ TL1 นับเป็น "1" หมดทุกบิต ก็จะเป็นการโหลดค่าจากรีจิสเตอร์ TH1 เข้าไปไว้ใน TL1 โดยอัตโนมัติ และทำการทดแฟลกอินเตอร์รัพต์ TF1 ให้เป็น "1" ค่าใน TH1 นี้เราสามารถตั้งค่าได้ด้วยซอฟแวร์ ลักษณะวงจรดังรูปที่ 3.7



รูปที่ 3.7 แสดงวงจรการทำงานของตัวจับเวลา/ตัวนับที่ 1 ในโหมด 2

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.8 วงจรการทำงานของตัวจับเวลา/ตัวนับที่ 0 ในโหมด 3

โหมด 3 เป็นการเพิ่มตัวจับเวลาขึ้นอีก 1 ตัว แต่จะเป็นขนาด 8 บิตทั้งคู่ ซึ่งลักษณะการทำงานอื่นๆจะเหมือนกับโหมด 0 การทำงานแสดงดังรูปที่ 3.8

สำหรับรายละเอียดบิตควบคุมในรีจีสเตอร์ TMOD แสดงได้ดังต่อไปนี้

TIMER/COUNT 1				TIMER/COUNT 0			
GATE	C/T	M1	M0	GATE	C/T	M1	M0

GATE : เซตเป็น "1" จะเป็นการอีนาเบิลตัวจับเวลา / ตัวนับให้ถูกควบคุมด้วยขา INTx ต้องมีสถานะ สูงและบิต TRX ใน TCON ต้องเซตเป็น "1" จึงจะเริ่มทำงาน แต่ถ้า GATE เป็น "0" ตัวจับเวลา /ตัวนับจะถูกควบคุมให้เริ่มทำงานด้วยบิต TRX เท่านั้น

C/T : เป็นบิตควบคุมในการเลือกทำงานเป็นตัวจับเวลา / ตัวนับถ้าเป็น "0" จะทำงานเป็นตัวจับเวลา ถ้าเป็น "1"ทำงานเป็นตัวนับ

M1,M0 : เป็นตัวเลือกโหมดการทำงานดังต่อไปนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

M1	M0	โหมดการทำงาน
0	0	โหมด 0
0	1	โหมด 1
1	0	โหมด 2
1	1	โหมด 3

ตารางที่ 3.1 แสดงบิตควบคุมที่อยู่ไรีจีตเตอร์ TMOD

รายละเอียดบิตควมคุมในรีจีสเตอร์ TCON แสดงได้ดังต่อไปนี้

TF_1	TR_1	TF_0	TR_0	IE_1	IT_1	IE_0	IT_0

TFx : เป็นแฟลกอินเตอร์รัพต์จะถูกเซตด้วยฮาร์ดแวร์เมื่อเกิดโอเวอร์โฟลว์ของตัวจับเวลา / ตัวนับและเครียร์ตัวเองโดยอัตโนมัติ เมื่อทำงานอินเตอร์รัพต์นั้นเสร็จเรียบร้อยแล้ว

TRx : เป็นบิตควบคุมให้ตัวจับเวลา /ตัวนับ เริ่มทำงานโดยการเซตให้เป็น "1" และให้หยุดทำงานโดยการเครียร์ให้เป็น "0" โดยซอฟต์แวร์

IEx : เป็นแฟลกอินเตอร์รัพต์จากสัญญาณภายนอก เซตด้วยฮาร์ดแวร์เมื่อมีสัญญาณของการอิน เตอร์รัพต์ปรากฏที่ขา INTx และจะเครียร์ตัวเองโดยอัตโนมัติ เมื่อกระโดดไปทำงานบริการอินเตอร์รัพต์ที่ขอมาเรียบร้อยแล้ว

ITx : เป็นบิตควบคุมรูปแบบสัญญาณอินเตอร์รัพต์ภายนอกจะเซต /เครียร์ด้วยซอฟต์แวร์ โดยถ้าเซตเป็น "1" จะถูกอินเตอร์รัพต์ด้วยสัญญาณขอบขาลง และเครียร์เป็น "0" จะถูกอินเตอร์รัพตด้วยระดับสัญญาณแรงดันต่ำ

หมายเหตุ x หมายถึงเลข 1 หรือเลข 0

3.1.3 พอร์ตอนุกรม

MCS-51 จะมีพอร์ตอนุกรมเป็นแบบฟูลดูเพล็กซ์ (fullduplex) สามารถที่จะสงและรับข้อมูลได้พร้อมกันเพราะมีบัฟเฟอร์ 2 ตัว ใช้ในการรับตัวหนึ่งและสงตัวหนึ่ง โดยโครงสร้างรีจีสเตอร์บัฟเฟอร์ทั้ง 2 ตัวนี้จะแยกแต่การติดต่อจะใช้ชื่อเดียวกันคือ SBUF พอร์ตอรุกรมของ MCS-51 สามารถที่จะโปรแกรมให้การทำงานได้แตกต่างกัน 4 โหมด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โหมด 0 ข้อจะเข้าและออกทางขา RXD โดยการเลื่อนสัญญาณนาฬิกาออกที่ขา TXD ข้อมูลจะเป็น 8 บิตโดยจะส่งบิตนัยสำคัญต่ำ (LSB) ก่อน อัตราบ๊อดจะคงที่ที่ 1/12 ของความถี่ออสซิเลเตอร์

โหมด 1 เป็นรับ / ส่งข้อมูลขนาด 10 บิต โดยการส่งออกทางขา TXD และรับเข้าทางขา RXD รูปแบบบิตจะประกอบด้วย 1 บิตสตาร์ทเป็น "0" , 8 บิตข้อมูล และ 1สต๊อปบิตเป็น "1" อัตราบ๊อด (baud rate) แปรผันได้ตามการตั้งตัวจับเวลาตัวที่ 1 โดยสูตรดังนี้

$$\text{อัตราบ๊อด} = \frac{2^{SMOD}}{32} \times \frac{\text{ความถี่ออสซิเลเตอร์}}{12 \times (256 - TH1)}$$

โหมด 2 เป็นการรับส่งข้อมูลขนาด 11 บิต เข้าทางขา RXD และส่งออกทางขา TXD ประกอบด้วย 1 บิตสตาร์ตมีค่า "0" , 9 บิตข้อมูล และ 1 บิต สต๊อป โดยการรับข้อมูลบิตที่ 9 จะถูกนำมาเก็บที่บิต RB8 ใน SCON ส่วนการส่งจะต้องใส่บิตที่ 9 ไว้ใน TB8 ของ SCON ก่อน อัตราบ๊อดสามารถเลือกได้ 2 อัตรา คือ 1/32 หรือ 1/64 ของความถี่ออสซิเลเตอร์ขึ้นอยู่กับการเซตบิต SMOD ใน รีจีสเตอร์ PCON ซึ่งรายละเอียดของรีจีสเตอร์ SCON แสดงในรูปที่ 11

โหมด 3 จะเหมือนกับโหมด 2 ทุกอย่าง ยกเว้นอัตราบ๊อดจะแปรผันตามการตั้งตัวจับเวลา 1 ซึ่งจะใช้โหมดเดียวกับโหมด 1

SM0	SM1	SM2	REN	TB8	RB8	TI	RI

SM0,SM1 : เป็นตัวกำหนดโหมดการใช้งานของพอร์ตอนุกรมดังนี้

SM0	SM1	โหมด
0	0	0
0	1	1
1	0	2
1	1	3

ตารางที่ 3.2 แสดงบิตควบคุมที่อยู่ในรีจีสเตอร์ SCON

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

SM2 : ควบคุมอินาเบิลการใช้โปรเซสเซอร์หลายตัวในการสื่อสารซึ่งกันและกันในโหมด 2 และ 3

REN : ตัวอินาเบิลอนุกรมการรุบการรับเมื่อเซตเป็น "1" และถ้าเป็น "0" เป็นการดิสเอเบิล

TB8 : เป็นตัวเก็บข้อมูลบิตที่ 9 ที่จะส่งจะในโหมด 2 และ 3

RB8 : เป็นตัวรับข้อมูลบิตที่ 9 ในโหมด 2 และ 3 ส่วนโหมด 1 จะเป็นสต๊อปบิต

TI : เป็นแฟลกอินเตอร์รัพต์การเซตด้วยฮาร์ดแวร์ที่ปลายช่วงบิตที่ 8 ในโหมด 0 หรือที่จุดเริ่มต้นของบิตสต๊อปในโหมดอื่น ในการส่งแบบอนุกรมของทุกโหมดจะต้องบิตนี้ด้วยโปรแกรมหลังการส่ง

RI : เป็นแฟลกอินเตอร์รัพต์การรับ เซตด้วยฮาร์ดแวร์ของปลายช่วงบิตที่ 8 ในโหมด 0 หรือครึ่งของช่วงบิตสต๊อปในโหมดอื่น ในการรับแบบอนุกรมจะเครียร์บิตนี้ด้วยโปรแกรมหลังการรับทุกครั้ง

3.2 ANT-32 EMBEDDED CONTROLLER

คำว่า "EMBEDDED CONTROLER" มักจะใช้เรียกบอร์ดที่ถูกออกแบบมาเพื่องานควบคุมโดยเฉพาะคือ เป็นบอร์ดที่ถูกติดตั้งอยู่ในเครื่องใช้ต่างๆปัจจุบันบอร์ดประเภทนี้นอกจากการใช้งานควบคุมแล้วมักจะมีการออกแบบเพื่อให้ทำการพัฒนาได้ง่ายขึ้นด้วย รวมทั้งมีการออกแบบให้เป็นลักษณะ "อเนกประสงค์" มากขึ้น คือ สามารถนำไปประยุกต์ใช้กับงานต่างๆได้ทันที โดยไม่ต้องเสียเวลามาออกแบบใหม่ทุกครั้ง ANT-32 คือ บอร์ดไมโครคอนโทรลเลอร์ที่ถูกออกแบบและสร้างโดยคนไทย ซึ่งมีการพัฒนามานานพอสมควรจนปัจจุบันเป็นเวอร์ชั่นที่ 2 แล้ว ANT-32 เป็นบอร์ดที่สามารถนำไปประยุกต์ใช้กับงานควบคุมทั่วไป มีขนาดเล็ก พร้อมทั้งวงจรพื้นฐานต่างๆที่มีให้ครบถ้วน และที่สำคัญคือ ANT-32 มีโปรแกรมรองรับในการพัฒนาได้อย่างสมบูรณ์ โดยมี REM31 และ BASIC32 ให้เลือกใช้ได้ สามารถพัฒนาด้วยภาษาแอสเซมบลีหรือภาษาเบสิกได้

3.2.1 คุณสมบัติของบอร์ด

- ใช้กับไมโครคอนโทรลเลอร์ในตระกูล MCS-51 (8031/8032)
- ความถี่สัญญาณนาฬิกา 11.0592 เมกะเฮิรต์
- หน่วยความจำอีพรอม 8 ถึง 32 กิโลไบต์ สหรับโปรแกรม
- หน่วยความจำแรม 8 ถึง 32 กิโลไบต์ สำหรับเก็บข้อมูลสามารถแบคอัพข้อมูลได้โดยใช้แบตเตอรี่ลิเธี่ยม
- ใช้กับหน่วยความจำชนิด EEPROM ขนาด 8 ถึง 32 กิโลไบต์ได้
- มีพอร์ตเบอร์ 8255 จำนวน 2 พอร์ต สำหรับการต่อไปใช้งานภายนอก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- มีวงจรพอร์ตอนุกรม (RS-232) สำหรับการต่อเข้ากับเครื่องไมโครคอมพิวเตอร์
- มีวงจรวอตช์ดอกและเทอร์นอัพ/คาวน์รีเซต ด้วยชิพเบอร์ MCS-691
- มีวงจร RTC (Real Time Clock) ใช้ชิพเบอร์ DS1202
- มีขั้วต่อสำหรับพอร์ต 1 ของไมโครคอนโทรลเลอร์โดยเฉพาะ
- มีขั้วต่อ SYSTEM BUS ทำให้ขยายระบบได้ง่าย และสามารถใช้กับบอร์ดขยายต่างๆที่จะมีขึ้นในอนาคต
- สามารถเลือกเบอร์หน่วยความจำหรือกำหนดคุณสมบัติต่าง ๆของเครื่องได้ด้วยจัมเปอร์
- ขนาดของบอร์ด 10.2 ซม. คูณ 14.2 ซม.

3.2.2 การทำงาน

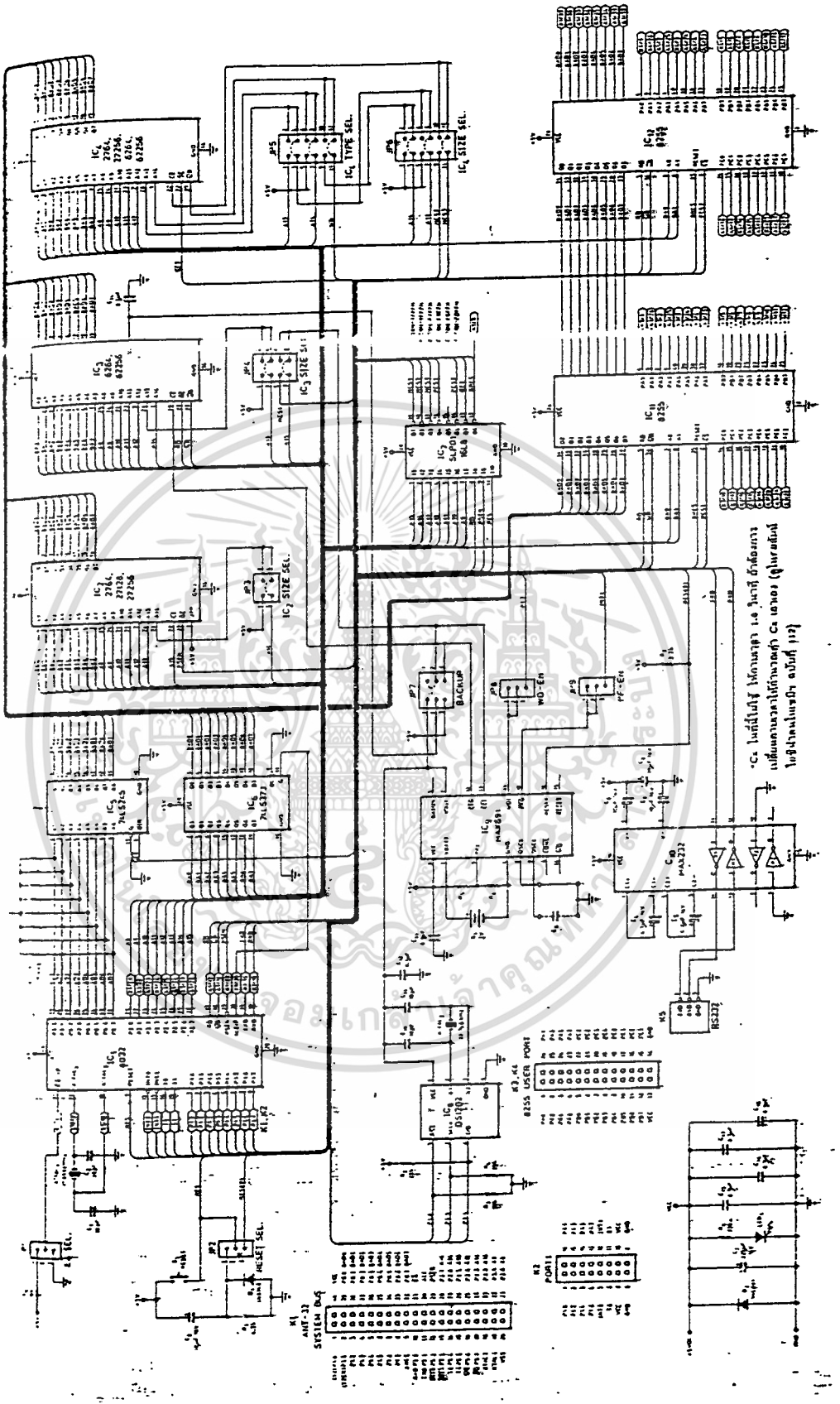
วงจรสมบูรณ์ของบอร์ด ANT V2.0 แสดงดังรูปที่ 3.9 ประกอบด้วยไมโครคอนโทรลเลอร์ 8032, หน่วยความจำ (ROM/RAM), พอร์ตอินพุต/เอาต์พุต, ส่วนถอดรหัสแอดเดรสหน่วยความจำและพอร์ต, ส่วนวงจรเวลาจริง (RTC), วงจรส่วนของวอตช์ด็อกไทร์เมอร์ (Watch Dog Timer) และแบตเตอรี่สำรองสำหรับ RTC และ RAM, วงจรแปลงแรงดันสำหรับพอร์ตอนุกรม

IC_1 เป็นชิพไมโครคอนโทรลเลอร์ในตระกูล MCS-51 ของ INTEL เบอร์ 8032 ทำงานที่ความถี่ 11.0592 เมกะเฮิรตร์ ทำการติดต่อกับหน่วยความจำ 3 ส่วนคือ ICญ$_2$เป็นหน่วยความจำภายนอกที่ใช้เก็บโปรแกรม (external program memory) ใช้กับอีพรอมขนาด 8-32 กิโลไบต์ เบอร์ 2764, 27128 หรือ 27256 โดยใช้จัมเปอร์ JP_3 ส่วนที่ 2 คือ IC_3 เป็นหน่วยความจำที่ใช้เก็บข้อมูล (data memory) ใช้กับแรม 8 ถึง 32 กิโลไบต์ เบอร์ 6264 หรือ 62256 เลือกขนาดด้วยจัมเปอร์ JP_4 และ IC_3 นี้มีแบตเตอรี่แบคอัพสำหรับเก็บข้อมูลเมื่อไฟดับด้วย และส่วนที่ 3 คือ IC_4 เป็นหน่วยความจำที่ใช้เก็บโปรแกรมและเก็บข้อมูล (program and data memmory) ใช้กับอีพรอมหรือ EEPROM และแรมขนาด 8 ถึง 32 กิโลไบต์ โดยใช้อีพรอมเบอร์ 2764, 27256 หรือ EEPROM เบอร์ 2864(A), 28256(A) และแรมเบอร์ 6264, 62256 ซึ่งเลือกชนิดและขนาดของ IC_4 ด้วยจัมเปอร์ JP_5 และ JP_6 ในกรณีต้องการใช้หน่วยความจำภายในที่ใช้เก็บโปรแกรม (internal program memory) ภายใน IC_1 สำหรับชิพไมโครคอนโทรลเลอร์เบอร์ 8051, 8052, 8751 และ 8752 สามารถได้โดยจัมเปอร์ JP_1

ส่วนของพอร์ตนอกจากพอร์ตภายใน (internal port) ของซีพียูคือ PORT 1 รวมทั้งขา INT0, INT1, T0 และ T1 แล้ว ยังมีพอร์ตภายนอกคือ IC_{11} และ IC_{12} เบอร์ 8255 ให้ใช้งานเพิ่มเติมอีกถึง 48 บิต

ส่วนถอดรหัสแอดเดรสของหน่วยความจำ สำหรับ IC_2 ใช้สายสัญญาณแอดเดรส A_{15} เพื่อกำหนดให้ IC_2 อยู่ที่แอดเดรส 0000H-7FFFH ส่วน IC_3 และ IC_4 ใช้สายสัญญาณแอดเดรส

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.9 วงจร ANT-32 V2.0

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

A_{13}- A_{15} ร่วมกับ IC_7 จัดให้ IC_3 อยู่ที่แอดเดรส 0000H-7FFFH และ IC_4 อยู่ที่แอดเดรส 8000H-F7FFH ส่วนพอร์ต IC_{11} จัดให้อยู่ที่แอดเดรส F800H-F9FFH และ IC_{12} อยู่ที่แอดเดรส FC00H-FDFFH IC_7 นอกจากใช้ถอดรหัสแอดเดรสแล้วยังใช้สร้างสัญญาณอ่านข้อมูลสำหรับหน่วยความจำ IC_4 และเป็นบัฟเฟอร์สัญญาณรีเซตซีพียูอีกด้วย

วงจรนาฬิกาเวลาจริง (real time clock) คือ IC_8 ใช้ชิพเบอร์ DS1202 (serial timekeeper chip) อินเตอร์เฟสกับซีพียูแบบอนุกรม โดยใช้สายสัญญาณเพียง 3 เส้นเท่านั้นคือ RST (reset), SCLK (serial clock) และ I/O (data input/output) IC_9 เป็นส่วนของวงจรวอตช์ด็อกไทร์เมอร์และวงจรแบตเตอรี่สำรองข้อมูล ใช้ไอซีเบอร์ MAX691 (Microprocessor Supervisery Circuits) วงจรส่วนนี้จะคอยกระตุ้น (รีเซต) ให้ซีพียูเริ่มทำงานใหม่ เมื่อระบบแฮงก์ (hang) เพื่อเพิ่มเสถียรภาพการทำงานของระบบให้ดีขึ้น ซึ่งสามารถกำหนด time-out period ได้โดยค่า Cx และวงจรวอตช์ด็อกไทร์เมอร์นี้ยังสามารถกำหนดให้สามารถอีนาเบิลและดิสเอเบิลได้ด้วยจัมเปอร์ JP_8 สำหรับวงจรแบตเตอรี่สำรองข้อมูลใช้สำหรับเก็บค่าเวลาของ IC_8 และเก็บข้อมูลของแรม IC_3 ในขณะไฟดับ (power fail) โดยใช้แบตเตอรี่ลิเธียม สำหรับข้อดีของแบตเตอรี่ชนิดนี้คือ จัดวงจรโดยไม่ต้องมีวงจรชาร์จ และมีอายุการใช้งานนานกว่าแบตเตอรี่ชนิดนิเกิลแคดเมี่ยม วงจรนี้สามารถสำรองได้ทั้ง RTC และ ได้นานนับปี การกำหนดสถานภาพการแบคอัพข้อมูลใน IC_2 กระทำได้ใช้จัมพ์เปอร์ JP_7 นอกจากนั้นแล้ว ยังมีวงจรตรวจจับแรงดัน V_{cc} เมื่อแรงดันตก วงจรในส่วนนี้จะถูกใช้สำหรับที่ต้องการเก็บค่าสถานะข้อมูลบางอย่างของระบบลงแรมก่อนไฟดับ แรงดันที่ตรวจจับกำหนดได้จากค่า R_5 และ R_6 และกำหนดให้อีนาเบิล หรือดิสเอเบิลได้ด้วยจัมพ์เปอร์ JP_9 ส่วนสัญญาณรีเซตซีพียูสามารถเลือกได้ว่าจะใช้วงจร R, C รีเซตหรือจาก IC โดยจัมเปอร์ JP_2

IC_{10} ทำหน้าที่ปรับระดับแรงดันสำหรับสัญญาณข้อมูลอนุกรมจากขา TXD, RDX ของ IC_1 ให้เป็นไปตามระดับแรงดันตามมาตรฐาน RS-232 ซึ่งจะใช้กับการสื่อสารรับส่งข้อมูลระหว่างบอร์ด ANT-32 กับเครื่องไมโครคอมพิวเตอร์

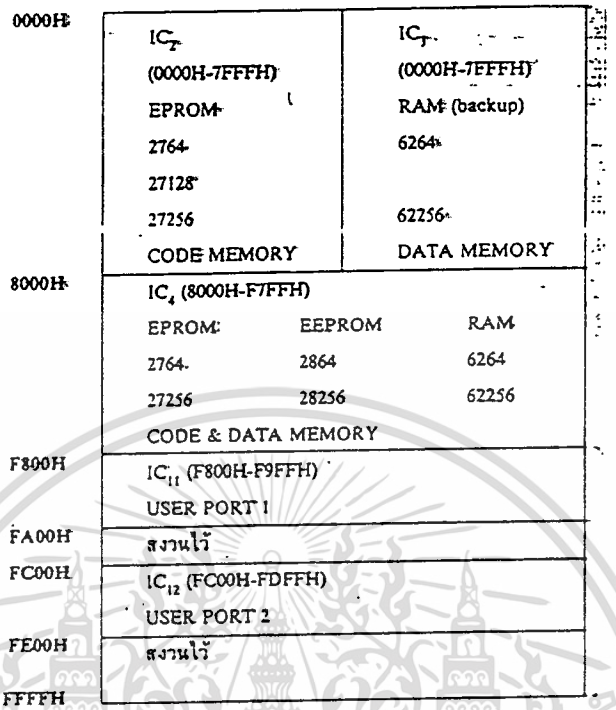
3.2.4 การจัดการหน่วยความจำ

รูปที่ 3.10 แสดงถึงการจัดการหน่วยความจำของบอร์ด ANT-32 V2.0 ซึ่ง IC_2 เป็นส่วนของรอมมอนิเตอร์และ IC_3 และ IC_3 เป็นแรมแบคอัพ ทั้ง IC_2 และ IC_3 ถูกถอดรหัสที่แอดเดรสเดียวกันคือ 0000H-7FFFH สำหรับ IC_4 สามารถใช้ได้ทั้งอีพรอม, EEPROM และแรมอยู่ในตำแหน่งแอดเดรส 8000H-F7FFH และแอดเดรสช่วง F800H-FFFFH ถูกใช้สำหรับเป็นแอดเดรสของพอร์ต โดย IC_{11} อยู่ที่แอดเดรส F800H-F9FFH และ IC_{12} ที่แอดเดรส FC00H-FDFFH

3.2.5 การนำไปใช้งาน

ลำพังเพียงบอร์ด ANT-32 อย่างเดียวยังไม่สามารถทำให้งานที่ต้องการจนสำเร็จได้ จำเป็นต้องอาศัยโปรแกรมที่เขียนขึ้นโดยเฉพาะ โดยมีโปรแกรมให้เลือก 2 ลักษณะคู่กันคือ REM31

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.10 การจัดการหน่วยความจำของบอร์ด ANT-32 V2.0

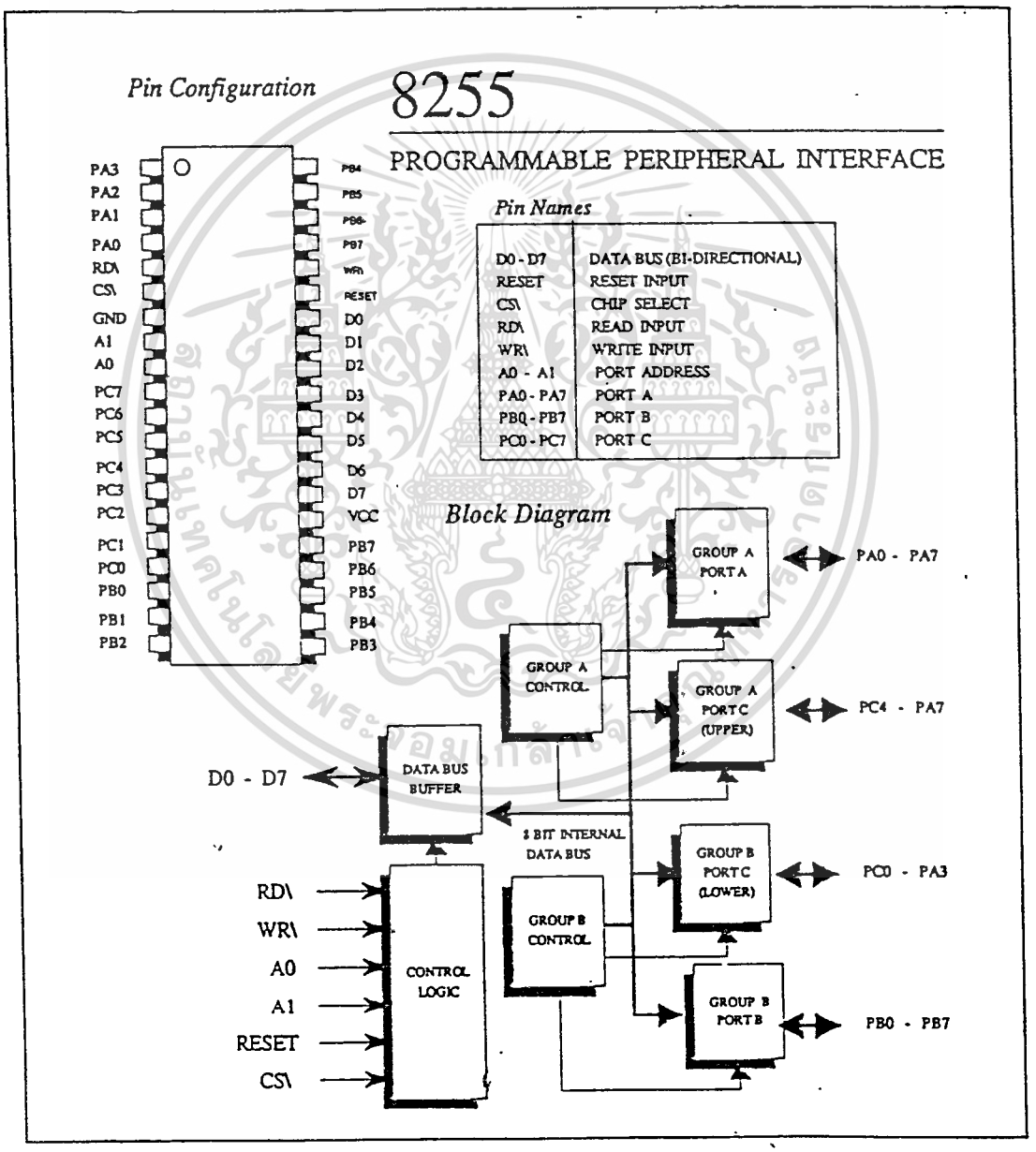
และ BASIC32 ถ้าผู้ใช้มีบอร์ดในลักษณะที่อีพรอมอีมูเลเตอร์ก็สามารถนำมาใช้เป็นตัวพัฒนาได้เหมือนกัน การงานก็เพียงแต่นำอีพรอมที่บรรจุโปรแกรมนี้ (REM31 หรือ BASIC32) ไปเสียบลงบนบอร์ด ANT-32 แล้วต่อสายอินเตอร์เฟสระหว่างบอร์ดกับเครื่องไมโครคอมพิวเตอร์ จากนั้นใช้โปรแกรมสำหรับการสื่อสารทางอนุกรมให้ไมโครคอมพิวเตอร์ติดต่อกับบอร์ดได้ ลักษณะของโปรแกรมที่ใช้ ถ้าเป็น REM31 ก็จะเป็นชุดคำสั่งที่คล้ายคลึงกับคำสั่ง DEBUG ของ DOS แต่ถ้าเป็น BASIC32 ก็จะสามารถเขียนเป็นภาษาเบสิกได้เลย

3.3 การใช้งาน 8255 กับ 8051

ไอซีเบอร์ 8255 ได้รับการออกแบบมาเพื่อทำหน้าที่เป็นพอร์ต สำหรับการรับส่งข้อมูลแบบขนานระหว่างอุปกรณ์ภายนอกกับไมโครคอนโทรลเลอร์ จากแผนภาพในรูปที่ 3.11 จะเห็นว่า 8255 ประกอบด้วยบล็อกของหน่วยการทำงานหลายส่วนอยู่ภายในบล็อกทางด้านขวามือจำนวน 4 บล็อก เป็นส่วนที่เชื่อมต่อกับอุปกรณ์ภายนอกโดยตรงผ่านทางเส้นสัญญาณที่ระบุชื่อว่า PA0-PA7,PB0-PB7 และ PC0-PC7 กลุ่มของสัญญาณเหล่านี้จำแนกออกเป็น 3 กลุ่ม คือ พอร์ต A (PA)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

พอร์ต B (PB) และ พอร์ต C (PC) สำหรับบล็อกถัดเข้ามาบริเวณส่วนกลางที่มีชื่อว่า GROUP A CONTROL และ GROUP B CONTROL ทำหน้าที่กำหนดการทำงานของพอร์ตทั้งสาม บล็อกทั้งสองนี้เชื่อมต่อกับบล็อกอื่นๆผ่านทางบัสข้อมูลภายใน 8255 เอง สำหรับบล็อกการทำงานทางด้านซ้ายที่มีชื่อว่า Data bus buffer และ read/write control logic ทำหน้าที่เชื่อมต่อระหว่างระบบบัสของไมโครคอนโทรลเลอร์กับ 8255 เพื่อรับหรือส่งข้อมูลระหว่างกันตามระดับลอจิกของขาสัญญาณ RD\ และ WR\ ตามลำดับ



รูปที่ 3.11 แผนภาพแบบบล็อกภายในและขาสัญญาณของไอซีเบอร์ 8255

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.2.1 การจำแนกกลุ่มของพอร์ต 8255

ในบรรดาพอร์ตทั้งสามของ 8255 พอร์ต A พอร์ต B และพอร์ต C โดยพื้นฐานนั้น ล้วนเป็นพอร์ตแบบขนานที่ประกอบด้วยสัญญาณ 8 เส้น ซึ่งแต่ละเส้นจะแทนบิตของข้อมูลพอร์ต ซึ่งอาจจะกล่าวในอีกลักษณะว่าเป็นพอร์ตแบบ 8 บิต นอกจากนี้ยังสามารถอ้างถึงแต่ละบิตของเส้นสัญญาณพอร์ตโดยอิสระ อย่างไรก็ตาม 8255 ได้จัดกลุ่มของพอร์ตเหล่านี้ออกเป็นสองกลุ่ม (Group) คือ Group A และ Group B เพื่อประโยชน์ในการกำหนดรูปแบบการทำงานของพอร์ต ดังตารางต่อไปนี้

ตารางที่ 3.3 ตารางแสดงลักษณะของ Group

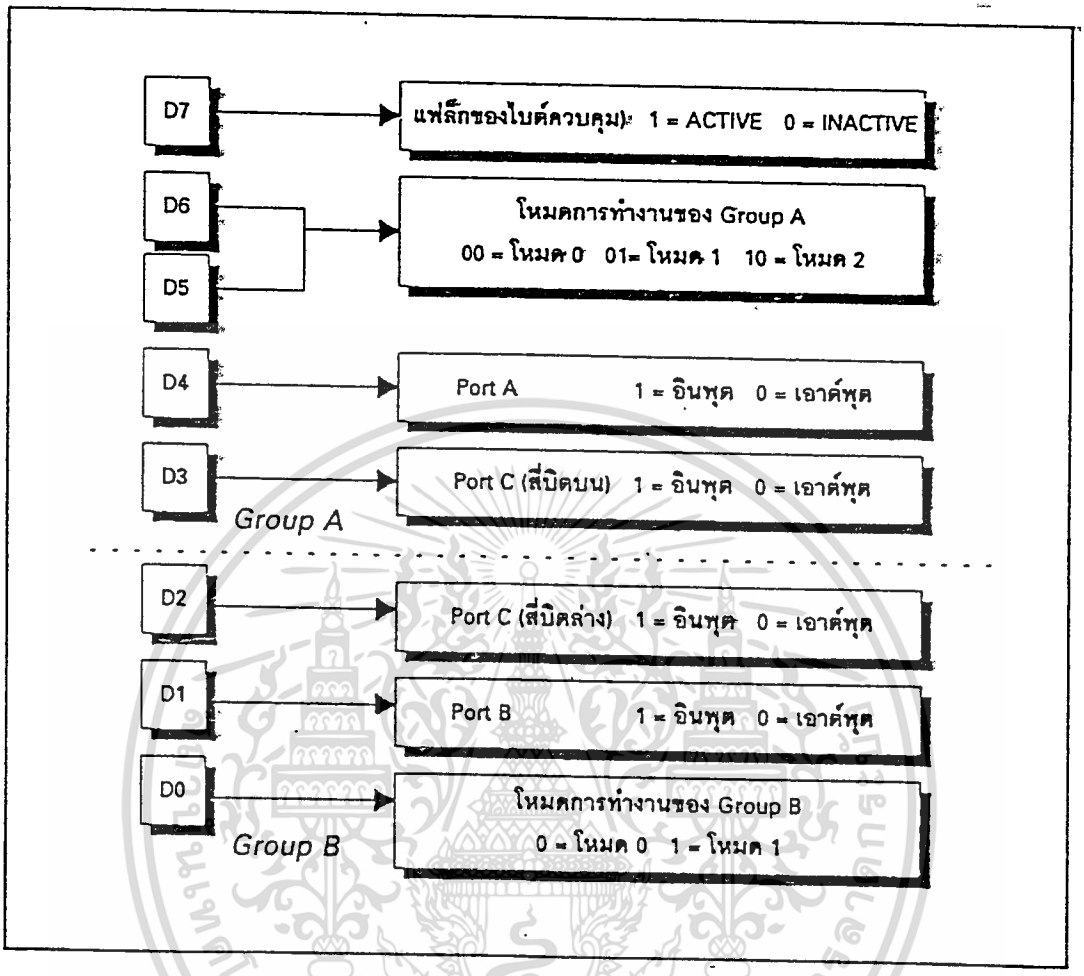
ชื่อกลุ่ม	ลักษณะ
Group A	พอร์ต A จำนวน 8 บิต (ทุกบิตของพอร์ต) พอร์ต C จำนวน 4 บิต (เฉพาะ 4 บิตบนของพอร์ต)
Group B	พอร์ต B จำนวน 8 บิต (ทุกบิตของพอร์ต) พอร์ต C จำนวน 4 บิต (เฉพาะ 4 บิตล่างของพอร์ต)

จากตารางข้างต้นจะเห็นว่าจำนวนเส้นสัญญาณของพอร์ตC(PC0-PC7)ได้ถูกแยกออกเป็นกลุ่มคือกลุ่ม 4 บิตล่าง (Lower nibble)จาก PC0-PC3 และกลุ่ม 4 บิตบน Upper nibble) จาก PC4-PC7 ดังนั้น Group A และ Group B ของ 8255 จึงมีจำนวนบิตในแต่ละกลุ่มเป็นจำนวน12 บิต

ตารางที่ 3.4 ตารางแสดงหน้าที่การทำงานของขาสัญญาณไอซี 8255

สัญญาณ	ความหมาย
D0-D7	กลุ่มของเส้นสัญญาณข้อมูลของ 8255 เมื่อมีการเขียนหรืออ่าน
CS\	สัญญาณเลือกอุปกรณ์ เมื่อขาสัญญาณ นี้เป็นระดับลอจิกต่ำซีพียูก็สามารถเขียนหรืออ่านข้อมูลจาก 8255 ได้
RD\	สัญญาณบอกสถานะต้องการอ่านข้อมูลจากรีจีสเตอร์ของ 8255
WR\	สัญญาณบอกสถานะต้องการเขียนข้อมูลให้กับรีจีสเตอร์ของ 8255
A0-A1	สัญญาณระบุตำแหน่งรีจีสเตอร์ภายใน 8255 ที่ต้องการ
RESET	สัญญาณการรีเซตวงจรทำงานภายใน 8255 เพื่อเริ่มต้นใหม่
PA0-PA7	กลุ่มของสัญญาณ 8 เส้น เมื่อทำการติดต่อกับพอร์ต A ของ 8255
PB0-PB7	กลุ่มของสัญญาณ 8 เส้น เมื่อทำการติดต่อกับพอร์ต B ของ 8255
PC0-PC7	กลุ่มของสัญญาณ 8 เส้น เมื่อทำการติดต่อกับพอร์ต C ของ 8255

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.12 ความหมายของบิตภายในไบต์ข้อมูลควบคุมของ 8255

3.2.2 รูปแบบคำสั่งเพื่อกำหนดการทำงานของ 8255

การกำหนดให้พอร์ตทั้งสามของ 8255 ทำงานในลักษณะต่างๆ กันหรือที่เรียกว่า โหมดการทำงาน (Mode) จะเริ่มด้วยการสั่งค่าข้อมูลไบต์หนึ่งให้กับรีจิสเตอร์ควบคุมการทำงานภายใน 8255 ข้อมูลนี้จะเรียกว่า ไบต์ข้อมูลควบคุม (Control word) โดยแต่ละบิตของข้อมูลนี้จะมีความหมายที่ระบุถึงความต้องการต่างๆ ไปดังแสดงในรูปที่ 3.12 การส่งข้อมูลไบต์นี้จะต้องเริ่มต้นเป็นอันดับแรกก่อนที่จะได้มีการดำเนินการใดกับ 8255 ทั้งสิ้น

ตามความหมายของบิตภายในตารางของรูปที่ 3.12 จะเห็นว่าการเลือกให้พอร์ตใดทำหน้าที่เป็นพอร์ตอินพุตก็เพียงแต่กำหนดค่าข้อมูล 1 ให้กับบิตที่เกี่ยวข้องกับพอร์ตนั้น หรือกรณีตรง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ข้ามสำหรับพอร์ตเอาต์พุตก็เพียงกำหนดค่าข้อมูล 0 เท่านั้น อย่างไรก็ตามการกำหนดให้ไบต์ข้อมูลควบคุมนี้มีผลอย่างถูกต้อง ก็จะต้องทำการกำหนดให้บิต D7 มีค่าเป็น 1 เสมอ

3.3.3 การเชื่อมต่อ 8255 กับ 8051

เมื่อพิจารณาแผนภาพของ 8255 จะเห็นว่ามีขาสัญญาณแอดเดรสจำนวน 2 เส้น คือ A0 และ A1 ทำให้ตำแหน่งของแอดเดรสที่จะอ้างถึงได้มีค่าเป็น 2^2 หรือเท่ากับ 4 ตำแหน่ง ซึ่งแต่ละตำแหน่งจะมีความหมายถึงการระบุรีจีสเตอร์หรือพอร์ตภายใน 8255 ดังตารางต่อไปนี้

ตารางที่ 3.5 แสดงชื่อของรีจีสเตอร์ที่ตำแหน่งแอดเดรสต่างๆของ 8255

A1	A0	ชื่อของรีจีสเตอร์
0	0	พอร์ต A
0	1	พอร์ต B
1	0	พอร์ต C
1	1	รีจีสเตอร์

เมื่อพิจารณาค่าของแอดเดรสเหล่านี้ร่วมกับระดับลอจิกของขาสัญญาณ RD\ และ WR\ จะเป็นการอ่านหรือเขียนข้อมูลทางขาสัญญาณ D0-D7 ให้กับรีจีสเตอร์นั้นตามลำดับดังตารางต่อไปนี้

ตารางที่ 3.6 แสดงความหมายของค่าแอดเดรสเมื่อใช้งานร่วมกับระดับลอจิกของขาสัญญาณ RD\ และ WR\

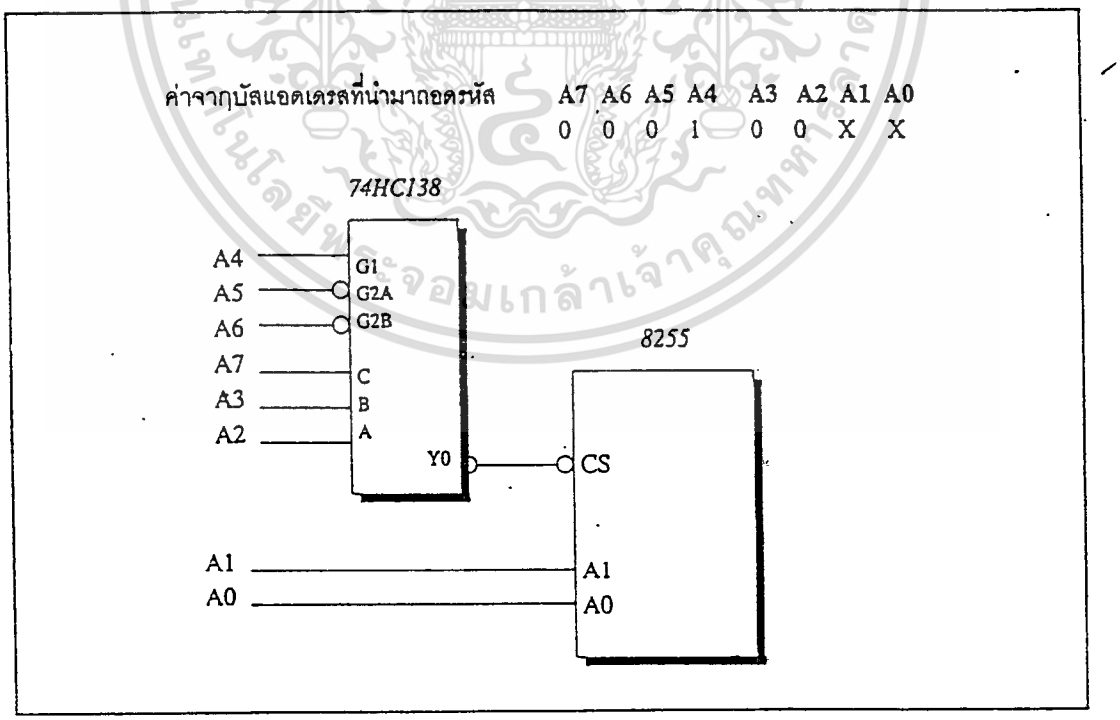
	WR\	A1	A0	ความหมาย
0	1	0	0	ส่ง(หรือเขียน)ข้อมูลให้กับพอร์ต A
1	0	0	0	รับ(หรืออ่าน)ข้อมูลจากพอร์ต A
0	1	0	1	ส่ง(หรือเขียน)ข้อมูลให้กับพอร์ต B
1	0	0	1	รับ(หรืออ่าน)ข้อมูลจากพอร์ต B
0	1	1	0	ส่ง(หรือเขียน)ข้อมูลให้กับพอร์ต C
1	0	1	0	รับ(หรืออ่าน)ข้อมูลจากพอร์ต C
0	1	1	1	ส่ง(หรือเขียน)ข้อมูลให้กับรีจีสเตอร์ควบคุม
1	0	1	1	เป็นสภาวะที่ไม่ถูกต้อง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ดังนั้นโดยทั่วไปจึงมักจะกำหนดให้แอดเดรสของ 8255 ทั้งสี่ตำแหน่งนี้ อยู่ในแอดเดรสช่วงใดช่วงหนึ่งของระบบ เช่น 10h, 11h, 12h, และ 13h โดยขาสัญญาณแอดเดรสที่นอกเหนือไปจาก A0 และ A1 นำเข้ามายังตัวถอดระหัสแอดเดรส เพื่อสร้างสัญญาณเลือกอุปกรณ์ (CS\) ในช่วงแอดเดรสที่ต้องการ ให้ดูตัวอย่างในรูปที่ 3.13 สัญญาณ CS\ นี้จะเป็นสภาวะลอจิกต่ำก็ต่อเมื่อค่าในบัสแอดเดรส A2-A7 มีค่าเท่ากับ 0000100xx ใช้เพื่อระบุถึงรีจีสเตอร์ภายใน 8255 เพื่อทำการอ่านหรือเขียนข้อมูล ดังนั้นจากวงจรนี้แอดเดรสของรีจีสเตอร์ภายใน 8255 จะมีค่าตามตารางนี้

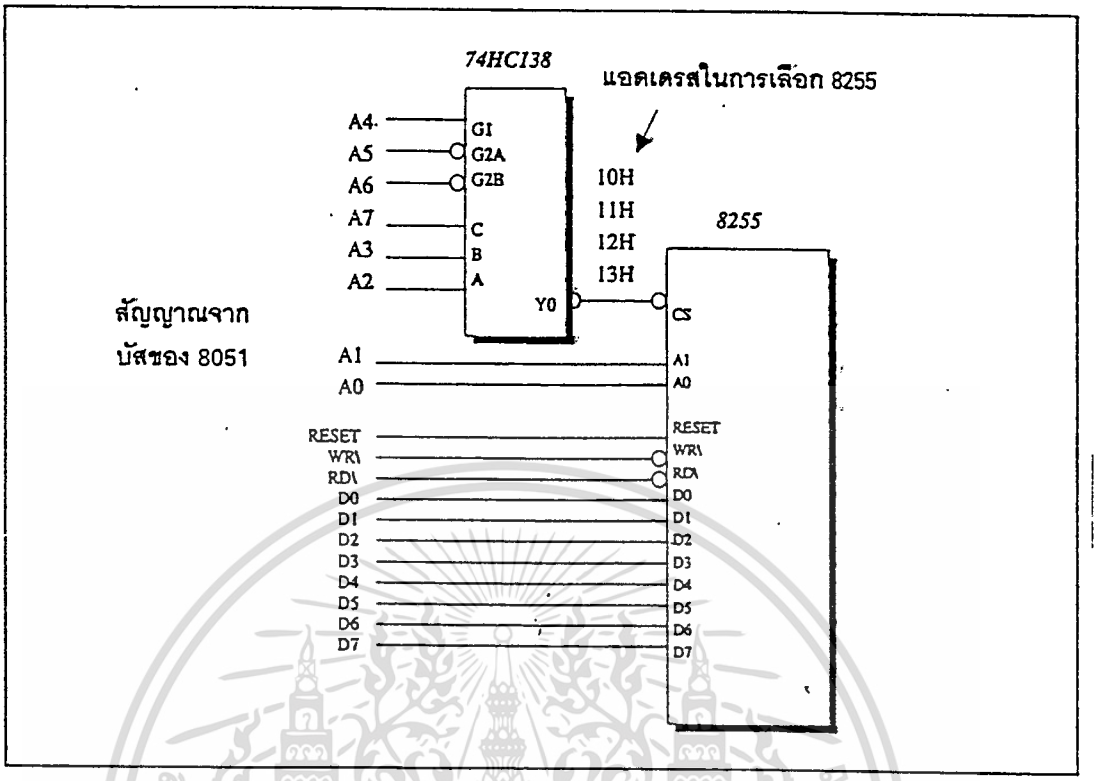
ตารางที่ 3.7 แสดงความหมายของค่าตำแหน่งแอดเดรสของรีจีสเตอร์ภายใน 8255

ตำแหน่งแอดเดรส	ความหมาย
10h	พอร์ต A
11h	พอร์ต B
12h	พอร์ต C
13h	รีจีสเตอร์ควบคุม



รูปที่ 3.13 แผนภาพแสดงการสร้างสัญญาณเลือกอุปกรณ์(CS\)ให้กับ 8255 โดยการถอดระหัสจากบัสแอดเดรส A2-A7

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.14 แผนภาพวงจรแสดงการเชื่อมต่อระหว่าง 8255 กับ 8051

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 4

การสื่อสารข้อมูลผ่านทางพอร์ตขนาน (พอร์ตพริ้นท์เตอร์)

พอร์ตขนานของคอมพิวเตอร์

การเชื่อมโยงกับเครื่องพิมพ์จะผ่านทางพอร์ตขนาน 8 บิต โดยบริษัทไอบีเอ็มได้ออกแบบให้สามารถต่อได้ตามมาตรฐานพอร์ตเซนโตรนิกส์ ซึ่งมีขนาด 25 ขา พอร์ตนี้เป็นได้ทั้งอินพุตและเอาต์พุต โดยมีกำลังขับได้ถึง 12 ทีทีแอลเอาต์พุต

4.1 พอร์ตอินพุต/เอาต์พุต

โดยทั่วไปพอร์ตนี้ปกติจะใช้เป็นเอาต์พุต เพื่อส่งข้อมูลให้กับเครื่องพิมพ์ แต่ก็ยังสามารถรับข้อมูลเป็นอินพุตพอร์ตได้เช่นกัน พอร์ตของเครื่องพิมพ์จะมีสัญญาณต่างๆ เพื่อใช้ในการตรวจสอบสัญญาณซึ่งกันและกันด้วย พอร์ตขนานสามารถติดตั้งบนคอมพิวเตอร์ได้มากกว่า 3 พอร์ต การกำหนดตำแหน่งอินพุต/เอาต์พุตแอดเดรสสำหรับ Parallel Interface แสดงได้ในตาราง

Port	Interface
3BCH - 3BFH	Parallel interface on MDA card
378H - 37FH	Parallel interface 1
278H - 27FH	Parallel interface 2

ตารางที่ 4.1 แสดงการกำหนดตำแหน่งอินพุต/เอาต์พุตแอดเดรสสำหรับพอร์ตขนาน

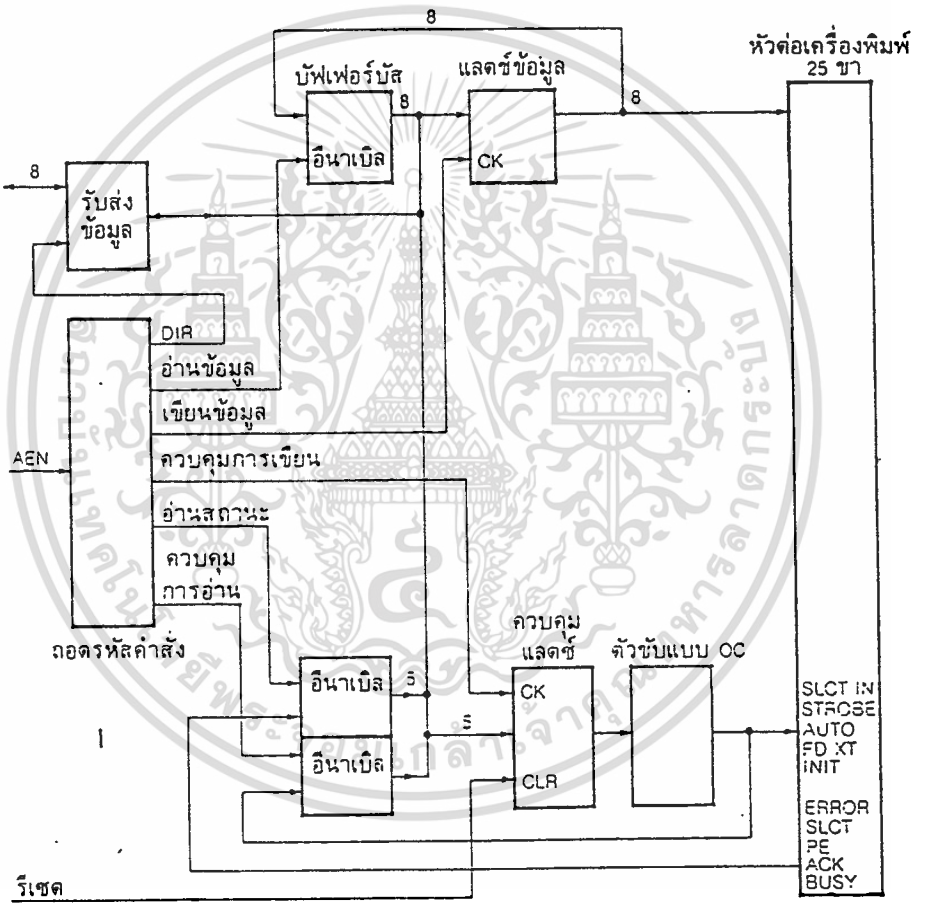
ไบออสจะกำหนดชื่อ LPT1 , LPT2 และ LPT3 ของพอร์ตขนานโดยผ่านทางเบสแอดเดรส(Base Address) ซึ่งเป็นการเปลี่ยนค่าใน BIOS Vaiable Segment โดย Vaiable Segment สามารถใช้กับ Parallel Port ได้ 4 Port แต่ไบออสจะมองเห็นเพียง 3 Port เท่านั้น ไบออสจะสามารถใช้กับ Variable Port ตัวที่ 4 ได้นั้นต้องมีการป้อนค่าเบสแอดเดรสที่ค่าออฟเซทแอดเดรส 000 EH เข้าไปใน BIOS Variable Segment ซึ่งแสดงในตารางต่อไปนี้

Address	Content
0040:0008H	Base address of LPT1
0040:000AH	Base address of LPT2
0040:000CH	Base address of LPT3
0040:000EH	Base address of LPT4

ตารางที่ 4.2 แสดงค่าเบสแอดเดรสของพอร์ตขนาน

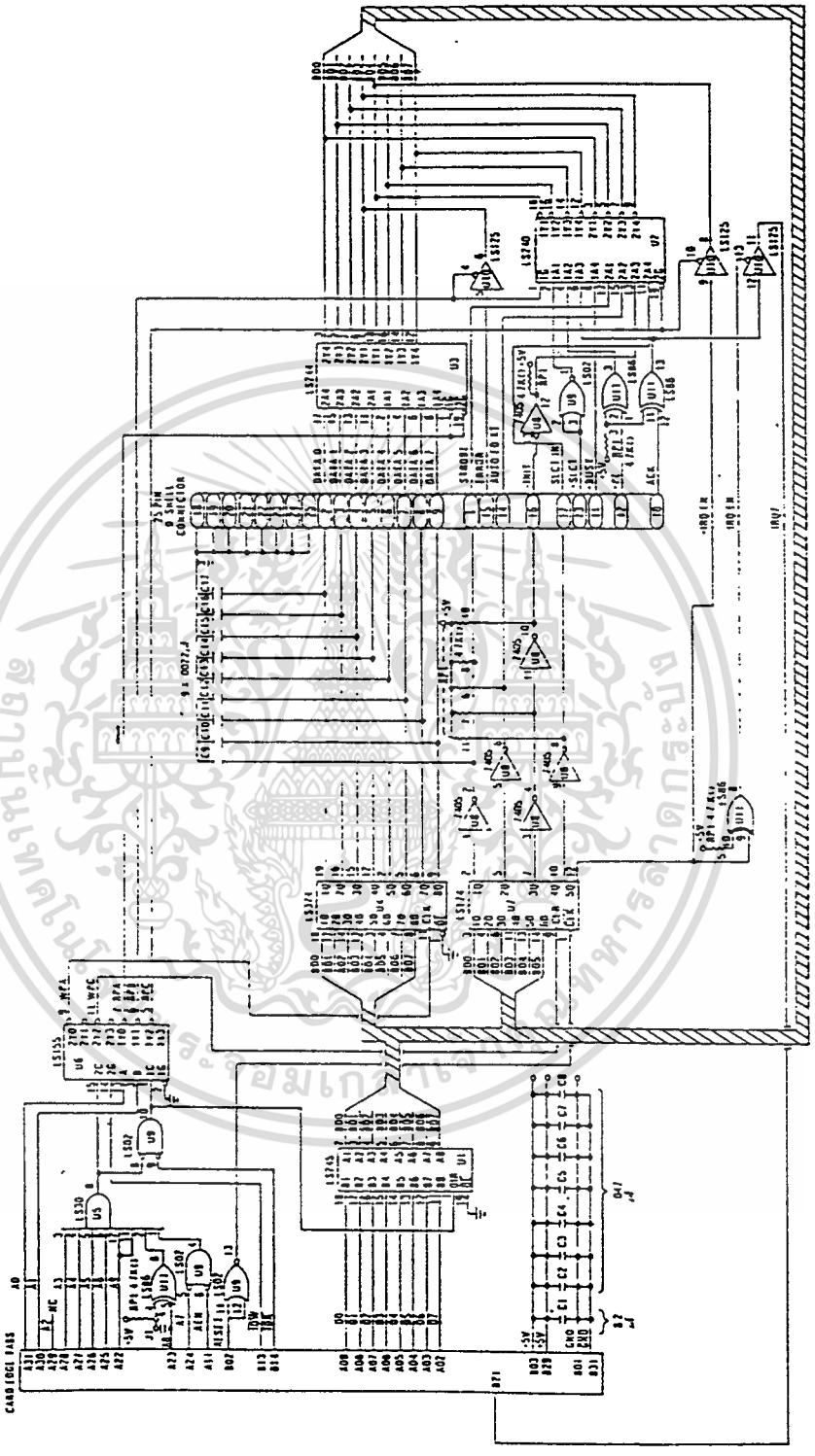
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ปกติบริษัทไอบีเอ็มได้ออกแบบไบออสไว้ให้ต่อกับเครื่องพิมพ์ได้ถึง 4 ตัวพร้อมกัน ดังนั้นจึงต้องเซตหมายเลขพอร์ตอินพุตเอาต์พุตไว้แตกต่างกัน เช่นในการใช้งานพอร์ตเครื่องพิมพ์ทั่วไปอาจอยู่บนอะแดปเตอร์เครื่องพิมพ์ต่างหากจะมีหมายเลขพอร์ตเพื่อกำหนดเป็นพอร์ต LPT1 โดยมีบล็อกไดอะแกรมดังรูปที่ 4.1



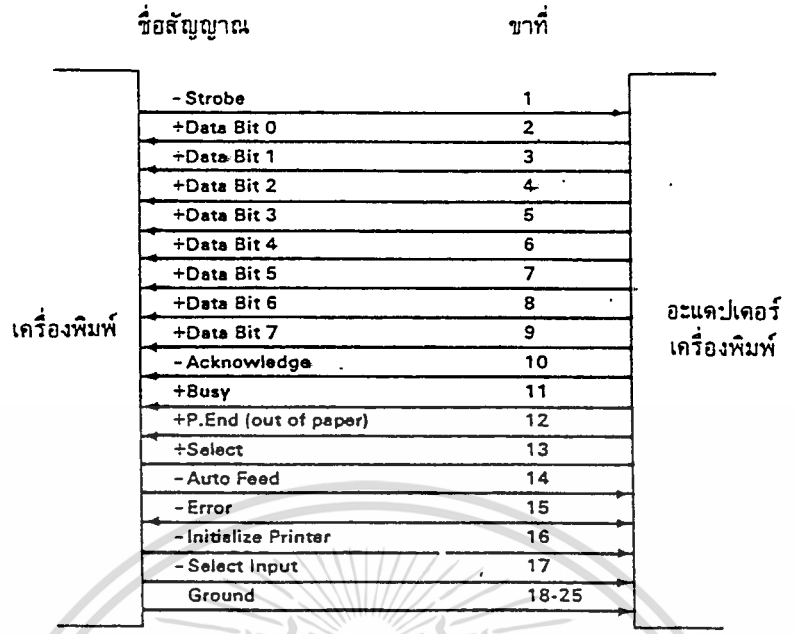
รูปที่ 4.1 ไดอะแกรมของอะแดปเตอร์ต่อกับเครื่องพิมพ์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.2 แสดงวงจรพริ้นท์เตอร์พอร์ตบนเครื่อง IBM PC

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.3 แสดงการเชื่อมโยงระหว่างคอมพิวเตอร์กับเครื่องพิมพ์

4.2 พอร์ตรีจิสเตอร์ (Port Register)

เมื่อพิจารณาตำแหน่งในแอดเดรสเมมโมรี (address memory) Parallel Port ทั้งหมดใช้รีจิสเตอร์เหมือนกันในการอินเตอร์เฟส ซึ่งประกอบขึ้นด้วย พอร์ต 3 พอร์ตๆ เหล่านี้จะใช้พอร์ตแอดเดรสตัวแรกของทั้ง 3 พอร์ต ของการ์ด เช่น 378H , 379H , 374H สำหรับเป็น Variable Port ตัวแรก

ความหมายของแต่ละบิตในพอร์ตรีจิสเตอร์จะแสดงในหัวข้อถัดไป เมื่อเปรียบเทียบหน้าที่ของแต่ละบิตในตารางกับคำสั่งของ Parallel Cable เราจะเห็นว่ามีค่าต่างกันการเชื่อมต่อโดยตรงระหว่างบิตของพอร์ตรีจิสเตอร์กับสาย Parallel Cable เมื่อบิตหนึ่งในรีจิสเตอร์ถูกเซตเป็น 1 สัญญานจะถูกส่งไปในสายที่ตรงกันกับบิตนั้น ถ้าบิตถูกเซตค่าเป็น 0 กระแสในสายจะมีสถานะเป็น LOW โดยกระแสในสายจะสะท้อนสถานะของรีจิสเตอร์บิตซึ่งจะถูกจัดการโดยซอฟแวร์

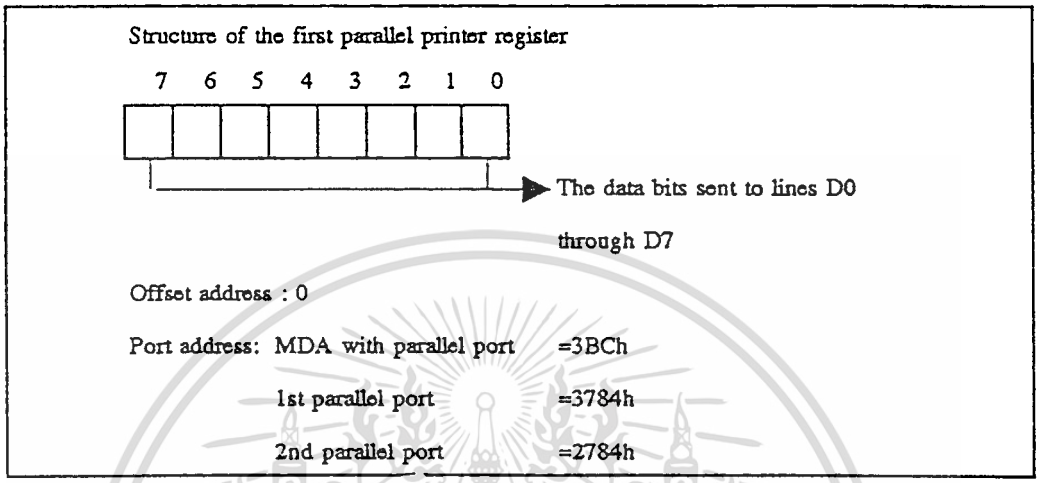
4.2.1 พอร์ตส่งข้อมูลออกเครื่องพิมพ์

บิตทั้ง 8 ของ Parallel Port ตัวแรกจะใช้ลอจิกบวก รีจิสเตอร์จัดเก็บข้อมูล 8 บิตที่จะส่งออกทางสายข้อมูล (Data Line) D0-D7 และส่งไปยังตัวรับ โดยรีจิสเตอร์นี้จะใช้เพียงส่งข้อมูลในเอาท์พุตรีจิสเตอร์ไปบน Parallel Port เท่านั้น ไม่ใช้เพื่อรับค่าข้อมูลเข้ามา

-พอร์ตหมายเลข 378H สำหรับบอร์ดอะแดปเตอร์เครื่องพิมพ์

-พอร์ตหมายเลข 3BCH สำหรับบอร์ดโมโนโครม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.4 โครงสร้างของ Parallel Printer Register ตัวที่ 1

4.2.2 พอร์ตรับค่าจากพริ้นท์เตอร์เพื่อใช้ควบคุมสถานะของเครื่องพิมพ์

รีจิสเตอร์ตัวที่ 2 ใช้ควบคุมสถานะของพริ้นเตอร์ โดยรีจิสเตอร์นี้จะสามารถอ่านได้เพียงอย่างเดียวไม่สามารถเขียนได้ รีจิสเตอร์จะสะท้อนคุณสมบัติของ Status Line ซึ่งมาจากพริ้นเตอร์

-พอร์ตหมายเลข 378H สำหรับการอ่านข้อมูลจากพอร์ตขนาน กรณีบอร์ดอะแดปเตอร์เครื่องพิมพ์

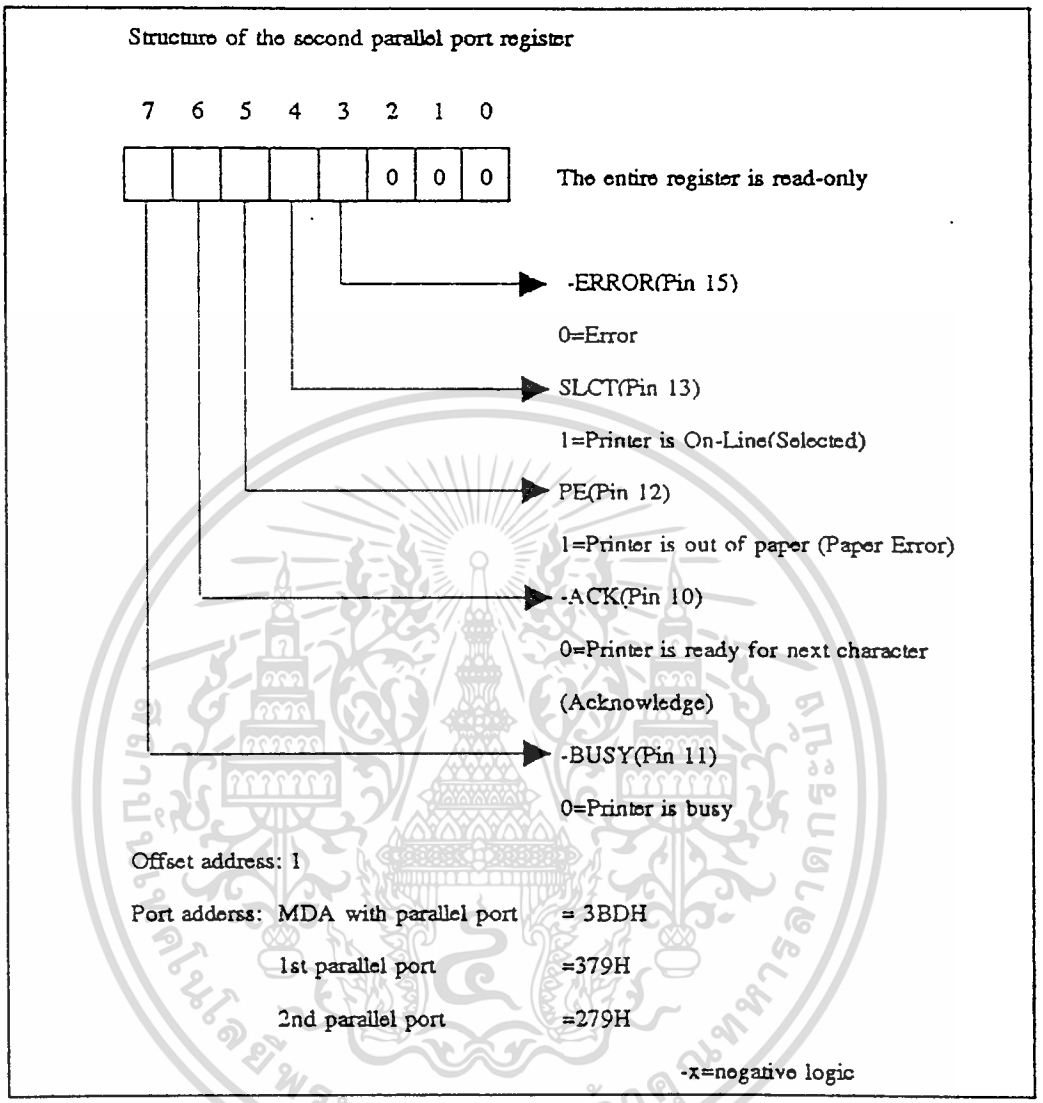
-พอร์ตหมายเลข 3BCH สำหรับอ่านข้อมูลจากพอร์ตขนาน กรณีบอร์ดโมโนโครม

-พอร์ตหมายเลข 379H สำหรับการอ่านสถานะจากพอร์ตอินพุต ในกรณีบอร์ดอะแดปเตอร์เครื่องพิมพ์

-พอร์ตหมายเลข 3BDH สำหรับการอ่านสถานะจากพอร์ตอินพุต ในกรณีบอร์ดโมโนโครม

-พอร์ตหมายเลข 3BEH สำหรับการอ่านสถานะ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.5 โครงสร้างของ Parallel Port Register ตัวที่ 2

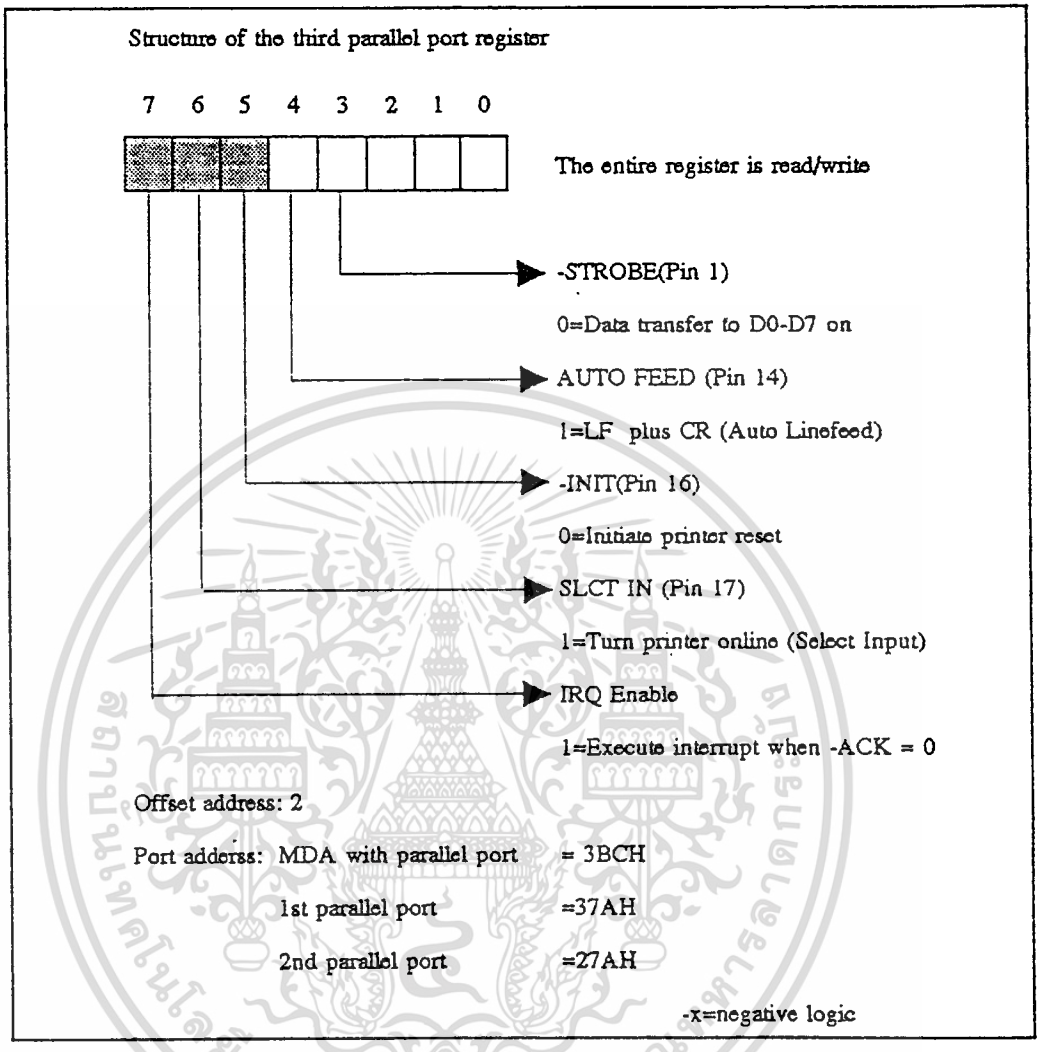
4.2.3 พอร์ตส่งสัญญาณออกเพื่อควบคุมเครื่องพิมพ์สำหรับตอบโต้สัญญาณ

รีจิสเตอร์ตัวที่ 3 ใช้ควบคุมพริ้นเตอร์และฮาร์ดแวร์ นอกจากนี้ยังมีหน้าที่สำคัญในการถ่ายโอน charecter ยกเว้น 4 บิต ซึ่งจะต่อกับ Parallel Port บิตที่ซ้อนอยู่ในรีจิสเตอร์สามารถเป็นฮาร์ดแวร์อินเตอร์รัพท์ (Hardware Interrupt) ได้โดยกำหนดให้ค่าสัญญาน ACK เป็น LOW เพื่อแสดงพริ้นเตอร์ได้รับค่า character สุดท้ายแล้วเราสามารถจำกัดขอบเขตที่อินเตอร์รัพท์จะปฏิบัติโดย DIP สวิทช์บนพอร์ต เราสามารถเลือกระหว่าง IRQ7 และ 5 ซึ่งตรงกับอินเตอร์รัพท์ 0FH และ 0OD

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

-พอร์ตหมายเลข 37AH สำหรับบอร์ดอะแดปเตอร์เครื่องพิมพ์

-พอร์ตหมายเลข 3BEH สำหรับบอร์ดโมโนโครม



รูปที่ 4.6 โครงสร้างของ Parallel Port ตัวที่ 3

4.3 การติดต่อระหว่างพริ้นเตอร์และโฮส

หน้าที่ของแต่ละขาในพอร์ตและความหมายของรีจิสเตอร์พอร์ต มีดังต่อไปนี้

- STROBE LINE

สายสโตรปมีความสำคัญสำหรับการเก็บแทรคของข้อมูล เมื่อโฮสเซตบิตนี้เป็น 0 พริ้นเตอร์จะรู้ว่า character ได้ถูกส่งมาบนสายแล้ว โฮสจะต้องไม่ส่งสัญญาน STROBE อย่างรวดเร็วเกินไป เพราะจะทำเพราะจะทำให้พริ้นเตอร์อ่านค่า charactor ซ้ำ 2 ครั้ง ทั้งนี้เนื่องจากพริ้นเตอร์ต้องการเวลาเพียงไมโครเซค ในการอ่านค่า characte จากสายข้อมูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

BUSY LINE

สาย Busy จะสร้างขึ้นหลังจากได้รับสัญญาน Strobe ซอฟแวร์หรือ ROM BIOS จะคอยจนกระทั่งพริ้นเตอร์ส่งสัญญาน Busy กลับก่อนที่จะส่ง character ถัดไป สาย Busy เป็นเพียงขาใน Parallel Port ที่มีการแปลงสัญญานเมื่อมันเป็นตัวรับ โดยค่าที่ส่งจากโฮสไปตัวรับจะมีค่าเป็น 0 ส่วนพริ้นเตอร์จะส่งค่าที่เป็น 1 ไปบนสาย Busy

-ACKnowledge Line

พริ้นเตอร์จะส่งสัญญาน ACK เป็น 0 ไปบนสาย ACKnowledge เพราะสายนี้เป็นลอจิกลบ โฮสจะรับค่าที่เป็น 1 ซึ่งแสดงว่าพริ้นเตอร์ได้รับค่า character ที่ส่งไปแล้ว

เวลาที่สัญญานต้องการใช้ในการส่งค่า character 1 ค่าจะใช้เวลา 10 μs คือจะมีอัตราการส่งผ่านข้อมูลเท่ากับ 100000 character/Sec อย่างไรก็ตามในความเป็นจริง โปรเซสเซอร์ต้องมีการบวกค่าของ Exturn time เข้าไป อัตราการส่งจริงจะมีค่าประมาณ 1/100 (1000 character/Sec)

4.4 ผลตอบสนองของพริ้นเตอร์ (Printer Responds)

การติดต่อระหว่างโฮสและพริ้นเตอร์จะเป็นการส่งผ่านข้อมูลเพียงด้านเดียวที่พริ้นเตอร์ส่งกลับมายังโฮส พริ้นเตอร์ใช้ขา 3 ขาในการส่งสัญญานกลับมายังโฮส ซึ่งได้แก่ ขา ERROR , SLCT และ PE ทั้ง 3 ขาจะต่อกับบิตที่ตรงกันใน Parallel Port ตัวแรก

- SLCT จะแสดงการเลือกซึ่งจะตรงกับสวิทช์ ONLINE ที่อยู่ทางด้านหน้าของพริ้นเตอร์ ถ้าเราปิดสวิทช์ ONLINE ของพริ้นเตอร์ ๆ จะส่งสัญญานไปยังโฮสโดยใช้สาย SLCT

- PE จะแสดง Paper Error ซึ่งจะบอกให้โฮสทราบว่าไม่มีกระดาษในพริ้นเตอร์หรือมีการติดขัดของกระดาษ ค่าผิดพลาดประเภทนี้ จะแยกจากการผิดพลาดเนื่องจากการส่งผ่านข้อมูลปกติ ที่ส่งมาบนสาย ERROR เป็นเพราะว่าค่า Paper Error สามารถแก้ไขได้โดยผู้ใช้แต่การผิดพลาดเนื่องจากการส่งผ่านข้อมูลเป็นปัญหาที่แก้ไขได้ยาก

4.4.1 Host Control

โฮสมีการควบคุมสัญญานที่ใช้ในการสั่งงานพริ้นเตอร์ โดยสัญญานที่ใช้คือ AUTO , FEED , INST และ SLCT IN บิตที่ใช้รับสัญญานเหล่านี้จะอยู่ใน Parallel Port Register ตัวที่ 3 ซึ่งค่าเหล่านี้สามารถอ่านและถ่ายเทโดยซอฟแวร์

4.4.2 Auto Feed

จะบอกพริ้นเตอร์ว่ามีการเพิ่ม Linefeed ในทุก CR (Carriage Return) character (ASCII code 13) ซึ่งจะรับสัญญานนี้และเซตเป็น 1 สายนี้จะรวมกันเพราะพริ้นเตอร์ทั้งหมดไม่สามารถรับของกลับในทิศทางเดียวกันได้ เมื่อรับค่า CR (Carriage return) character พริ้นเตอร์ส่วนจะกลับไปเริ่มต้นทำงาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ใหม่เมื่อเรามีการเพิ่ม Linefeed ซึ่งจะมีการเคลื่อนย้ายหรือพิมพ์ไปยังไลน์ถัดไป ดังนั้น LF (Linefeed) character จะต้องเพิ่มเข้าไปต่างหาก

โฮสสามารถใช้สาย SLCT IN ในการ OFFLINE ของพริ้นเตอร์โดยส่งสัญญาน 1 ไป ปกติสายเส้นนี้จะเซตเป็น LOW ซึ่งโดยปกติพริ้นเตอร์จะ ONLINE

โฮสใช้สาย INIT ในการรีเซตพริ้นเตอร์ การรีเซตจะเซตบิตที่ตรงกันเป็น 0 ในเวลาสั้นๆ และเซตกลับเป็น 1 พริ้นเตอร์จะทำการรีเซตตัวเองซ้ำกัน

4.5 สาย Cable

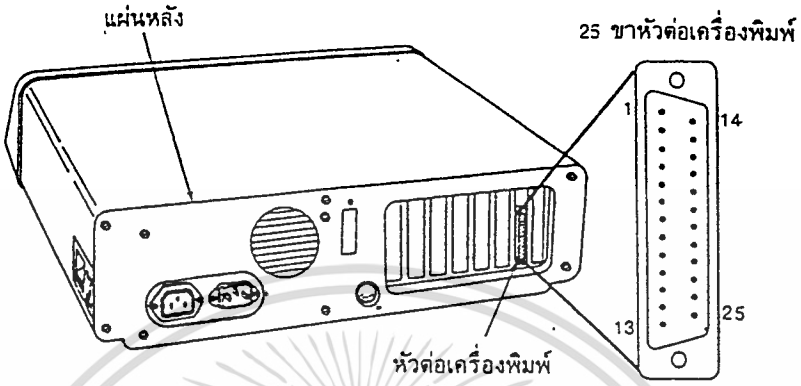
การถ่ายเทข้อมูลที่สมบูรณ์ระหว่าง Host กับ Printer จะทำได้ถ้ามีการตรวจสอบขาที่พอร์ตทั้งสองที่เชื่อมต่อกันโดย Proper Cable ซึ่งสัญญานที่ส่งออกโดยขาและเส้นทางที่ขานั้นเชื่อมต่อกันต้องได้มาตราฐาน Centronic จะบอกถึงหน้าที่ของขาในแต่ละพอร์ต และแต่ละไลน์ใน Cable

ตารางข้างล่างนี้จะแสดงขาของโฮส และพริ้นเตอร์พอร์ตที่ต่อกัน

Computer pin	Printer pin	Signal name	Meaning
1 →	1	-STROBE	Indicates data transfer
2 →	2	D0	Data line - bit 0
3 →	2	D1	Data line - bit 1
4 →	2	D2	Data line - bit 2 2
5 →	2	D3	Data line - bit 3
6 →	2	D4	Data line - bit 4
7 →	2	D5	Data line - bit 5
8 →	2	D6	Data line - bit 6
9 →	2	D7	Data line - bit 7
10 ←	10	-ACK	Last character received
11 ←	11	-BUSY	Printer busy
12 ←	12	PE	Printer has no paper
13 ←	13	SLCT	Printer is online
14 →	14	-AUTO FEED	Automatic CR after LF
15 ←	32	-ERROR	Data transfer error
16 →	31	-INIT	Reset printer
17 →	36	SLCT IN	Turn printer online
18-25 ← →	19-30	GND	Ground

ตารางที่ 4.3 แสดงขาของโฮสและพริ้นท์เตอร์ที่ต่อกัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.7 หัวต่อเครื่องพิมพ์ 25 ขา

ปัญหาเมื่อต้องการนำ Parallel Port มาใช้สำหรับการส่งผ่านข้อมูลระหว่างคอมพิวเตอร์ 2 เครื่องนั้นคือ Parallel Port บนคอมพิวเตอร์ทั้งคู่มีคอนเนคเตอร์ตัวเมียทั้งคู่ ดังนั้นปลายข้างหนึ่งของ Plug ไม่สามารถต่อกับคอมพิวเตอร์อีกเครื่องได้ อีกปัญหาหนึ่งคือ ปกติการสื่อสารแบบขนานจะทำได้เพียงด้านเดียว คอมพิวเตอร์ตัวหนึ่งสามารถใช้สายข้อมูล (Data Line) D0-D7 ในการส่งข้อมูล แต่ไม่สามารถรับข้อมูลกลับมาบนสายและคอมพิวเตอร์อีกตัวก็ไม่สามารถส่งข้อมูลมาได้ด้วย

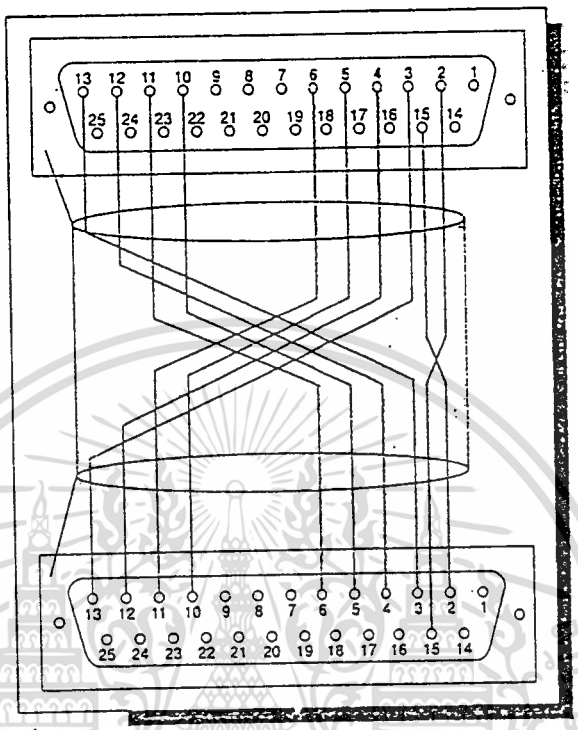
การส่งผ่านข้อมูลระหว่างคอมพิวเตอร์ 2 เครื่องต้องการๆ เชื่อมต่อแบบ 2 ทิศทาง ตัวรับตัวจะต้องส่งกลับค่า Checksum ของข้อมูลที่รับเข้ามา เพื่อให้ตัวส่งรู้ว่าค่าข้อมูลที่รับไปนั้นเกิดความผิดพลาดขึ้นหรือไม่

STATUS LINE จะใช้โดยพริ้นเตอร์ เพื่อส่งกลับค่าสถานะของการติดต่อของคอมพิวเตอร์ ซึ่งได้แก่ สาย ERROR , SCLT , PE , ACK และ BUSY ซึ่งสัมพันธ์กับ Parallel Port ตัวที่ 2 สายเส้นนี้จะต่อกับสายข้อมูล D0-D4 ซึ่งหมายถึงตัวรับค่าอ่านค่าเอาท์พุตจากตัวส่งตลอดสถานะของสายก่อน สายข้อมูล D0-D4 จากตัวรับต่อกับ Status Line ของตัวส่งที่สามารถส่งสายแบบ 2 ทาง

การต่อสายข้อมูล D0-D4 จะต่อกับสายสถานะ ERROR , SLCT , PE , ACK , และ BUSY ต่อไปจะเป็นการประยุกต์สำหรับทั้งตัวรับและตัวส่ง โดยข้อมูลจะถูกส่งออกผ่านทาง 5 บิตแรกของ Parallel Port ตัวแรก ซึ่งการรับข้อมูลจะใช้บิต 3 ถึงบิต 7 ในรีจิสเตอร์ตัวที่ 2 ของเครื่องที่จะติดต่อด้วย

รูปต่อไปเป็นการแสดงการต่อขาในแต่ละปลายของสายเคเบิ้ลที่ทำเป็น Parallel Transfer Cable

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



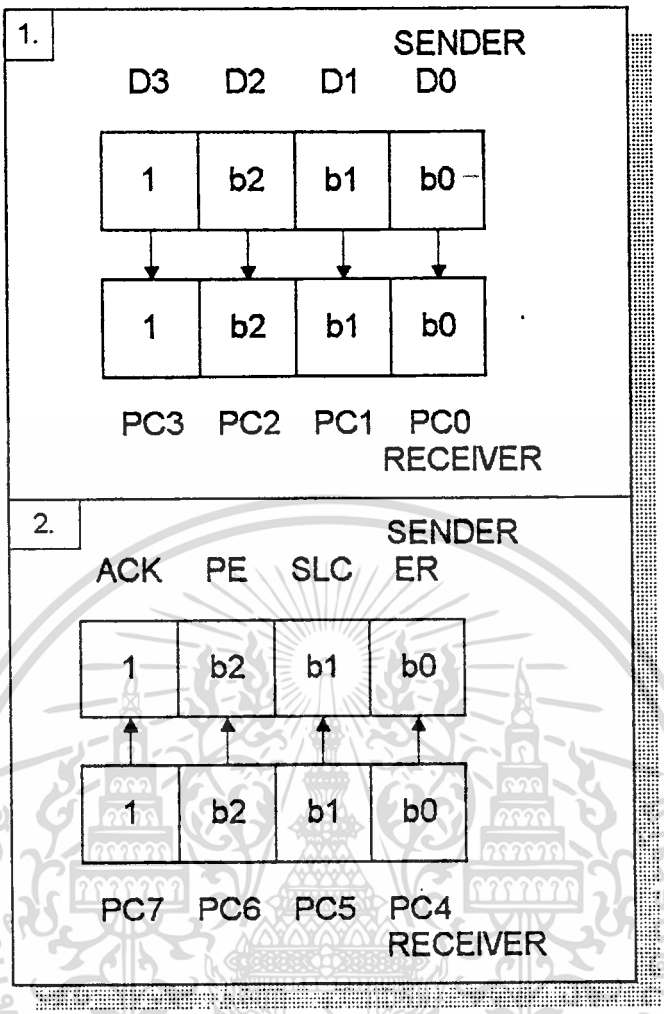
รูปที่ 4.8 แสดงการต่อขาในแต่ละปลายของสายเคเบิลที่เป็น Parallel Tranfer Cable

4.6 การโปรแกรมผ่านทางพอร์ตขนาน

4.6.1 การถ่ายโอนข้อมูล

สิ่งแรกที่ต้องคำนึงถึงคือตัวส่งข้อมูล ซึ่งจะต้องพิจารณาว่าการส่งข้อมูลในแต่ละครั้งการที่จะประสบความสำเร็จได้อาจจะต้องใช้การส่งข้อมูลซ้ำๆเพื่อให้ข้อมูลถูกต้อง ข้อมูลที่จะส่งผ่านทางสายสัญญาณ จะมีเพียง 4 bits โดยจะพิจารณาขั้นตอนการส่งแต่ละส่วนดังรายละเอียดต่อไปนี้ ขั้นตอนแรก เมื่อส่งข้อมูลจำนวน 3 bits ทาง parallel port ณขณะนี้ ข้อมูลจะถูกส่งไปที่สาย D0-D2 และเมื่อทำการ เปลี่ยนค่าใน bit 3 จะทำให้ตัวรับรู้ว่าขณะนี้ได้มีการส่งข้อมูลมาแล้ว ให้รับข้อมูลที่ส่งมา แล้วส่งข้อมูลที่ได้รับเพื่อตรวจสอบกับตัวส่งว่าข้อมูลที่ส่งมาถูกต้องหรือไม่ ถ้าข้อมูลถูกต้องก็จะส่งข้อมูลชุดถัดไปให้ สำหรับลำดับขั้นตอนสามารถศึกษาได้จากรูปที่ 4.9

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.9 The comunication protocal at the hardware oriented data

4.6.2 Time out problems

โดยปกติฟังก์ชันที่ใช้ในการติดต่อสื่อสารจะไม่มีปัญหาการเกิดขึ้น แต่ปัญหาที่จะเกิดขึ้นอยู่ที่สัญญาณทางไฟฟ้า ซึ่งจะเกิดปัญหาขึ้นเสมอ โดยจะเกิดขึ้นทั้งตัวรับและตัวส่ง ปัญหาที่เกิดขึ้นก็จะเป็นในส่วนของการรอทั้งทางด้านตัวรับและตัวส่ง เพื่อหลีกเลี่ยงปัญหาที่จะเกิดขึ้นอันได้แก่ การรอจนไม่รู้จักจบสิ้น คำตอบของปัญหานี้คือการใช้เวลาเป็นตัวกำหนด โดยค่าของเวลาจะถูกกำหนดโดยระบบเพื่อรอคำตอบของสัญญาณจากคู่ตรงข้ามก่อนที่จะจบการทำงาน
ต่อไปนี้เป็นตัวอย่างของการใช้ เวลาในการควบคุม จุดประสงค์คือการกำหนดค่าเพื่อควบคุมการส่งและการรับให้สามารถทำงานได้อย่างถูกต้อง โดยลักษณะจะเป็นดังนี้

pseudocode:

TimeOutCount = MAXVALUE

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
WHILE ( Bit4 = 1 ) AND ( TimeOutCount > 0 ) DO
 BEGIN
 END
IF TimeOutCount = 0 THEN
 error
ELSE
 o.k.
END
```

4.6.3 การซิงค์โครไนซ์เซชั่น

เมื่อโปรโตคอลในการติดต่อสื่อสารถูกกระตุ้น โปรโตคอลจะทำงานตามขั้นตอนโดยที่ปราศจากปัญหาใดๆ แม้ว่าในบางครั้งอาจเกิดความผิดพลาด หรือมีปัญหาบางอย่างขณะที่เริ่มต้นทำงาน โดยก่อนที่จะเริ่มการติดต่อสื่อสารนั้น ในบิท BUSY ของผู้ส่งและรับสัญญาณจะต้องมีค่าเป็น 0 ซึ่งถ้าไม่เป็นไปตามขั้นตอนดังกล่าวแล้ว ผู้รับสัญญาณจะถือว่ามีการส่งสัญญาณออกมาและจะทำการอ่านข้อมูล ถึงแม้ว่าผู้ส่งสัญญาณจะไม่ได้ส่งสัญญาณต่างๆ มาเลยก็ตาม ก่อนที่จะมีการติดต่อสื่อสารเกิดขึ้นนั้น ผู้ส่งและผู้รับสัญญาณจะต้องมีการทำงานแบบซิค์โครไนซ์ เพราะในการที่จะกำหนดให้ผู้ส่งหรือผู้รับสัญญาณฝ่ายใดฝ่ายหนึ่งเริ่มต้นการทำงานก่อนนั้น จะก่อให้เกิดความสับสน สำหรับโปรแกรมสาธิตที่จะแสดงดังกล่าวนี้ ผู้รับสัญญาณจะเปรียบเสมือนจุดเริ่มต้นของเรา

ขั้นตอนแรก ผู้รับสัญญาณจะต้องรอเพื่อให้ผู้ส่งสัญญาณทำการเซ็ตค่าของบิท BUSY เป็น 0 โดยวิธีการซิงค์โครไนซ์เซชั่นที่สมบูรณ์นั้นจะต้องมีค่าของบิท BUSY ทั้งผู้รับและผู้ส่งสัญญาณเป็น 0 ในโปรแกรมนี้จะมีการทำงานในเส้นทางของ Port Unit

โดยโปรซีเดอร์เริ่มต้นจะใช้ลิมิตของ time out โดยขั้นตอนการทำงานดังกล่าวได้กล่าวได้แสดงไว้ก่อนหน้านี้แล้ว โดยสิ่งที่เพิ่มเข้ามาในโปรแกรมนี้ก็คือ ผู้ใช้สามารถจะออกจากตัวโปรแกรมได้ตลอดเวลา โดยการกดคีย์ <Esc> ซึ่งจะต้องรอเวลาสำหรับช่วงเวลาการ time out ที่จะมาถึงด้วย

4.6.4 การหยุดโปรแกรม

ทั้งโปรแกรมจะประกอบไปด้วยตัวควบคุมการอินเตอร์รัพท์ของคีย์บอร์ด ซึ่งจะถูกกระตุ้นโดยการกดคีย์ <Esc> เช่นเดียวกัน เมื่อมีการควบคุมการอินเตอร์รัพท์ของไทม์เมอร์ โปรแกรมจะใช้ตัวแปรเพื่อทำการติดต่อสื่อสารกับควบคุมการอินเตอร์รัพท์ของคีย์บอร์ด ในกรณีนี้จะเป็นตัวแปรชนิด BOOL ซึ่งจะทำการเซ็ตค่าเป็น TRUE โดยการกดคีย์ <Esc>

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เราสามารถหลีกเลี่ยงขั้นตอนของการอ่านตัวแปรที่ไม่มีความต่อเนื่องภายในลูปการอ่าน โดยทำการดัปปลิ้งลูปดังกล่าวร่วมกับตัวแปร time out ซึ่งการดำเนินการในขั้นตอนนี้จะเซ็ตตัวแปรที่เป็นเท็จให้เป็น TRUE และเซ็ตค่าตัวแปร time out เป็น 0 โดยการกดคีย์ <Esc>

ในโปรแกรมนี้จะเป็นการตอบสนองอย่างง่าย ในการพิจารณาถึงช่วงเวลา time out ที่จะมาถึง เมื่อมีการตรวจสอบ ตัวแปรที่เป็นเท็จอย่างรวดเร็ว จะทำให้สามารถได้ข้อสรุปถึงสาเหตุของการอินเตอร์รัพท์ได้เป็นอย่างดี

4.6.5 โปรโตคอลในระดับของบล็อค

ระดับของบล็อคจะเป็นระดับที่สูงกว่าระดับของไบท์ โดยระดับของบล็อคจะใช้สำหรับการถ่ายโอนบล็อคของข้อมูลทั้งหมดจากผู้ส่งสัญญาณไปยังผู้รับสัญญาณ โดยขั้นตอนนี้จะถูกควบคุมโดยโปรโตคอลของซอฟท์แวร์ ซึ่งไม่เกี่ยวข้องกับส่วนของฮาร์ดแวร์ เพราะการทำงานไม่ได้ตั้งอยู่บนเส้นทางของการรับและการส่งที่เกิดจากระดับของไบท์ ในโปรแกรมนี้จะใช้เส้นทางของผู้รับและผู้ส่งสัญญาณเป็น SendABlock และ ReceiveABlock ตามลำดับ

บล็อคจะบรรจุไปด้วยข้อมูลดังต่อไปนี้ ส่วนของ token ซึ่งมีความสำคัญมากของบล็อค และเป็นการอธิบายถึง ขนาด ความยาวของบล็อค รวมทั้งตัวของบล็อคเอง โดย token จะถูกใช้จนกระทั่งตัวรับสัญญาณยอมรับสิ่งที่ส่งมา โดยไม่ได้มีการอ่านในบล็อคข้อมูล

ในรูปแบบนี้เส้นทางของ SendABlock จะผ่าน token จำนวนของไบท์ภายในบล็อค และตัวชี้ไปยังตัวของบล็อคข้อมูลเอง โดยทั้ง token และจำนวนของไบท์จะถูกกำหนดโดยชนิดของเฮดเดอร์และมีการดูแลรักษาที่แยกออกมาจากบล็อคข้อมูลที่เป็นอยู่จริงๆ ตัวรับสัญญาณจะได้รับส่วนของเฮดเดอร์เป็นส่วนแรก หลังจากนั้นจึงจะได้รับส่วนของไบท์แรกของข้อมูลที่ส่งมา ลองคิดดูซิว่า จะมีอะไรเกิดขึ้น ถ้าผู้ส่งสัญญาณทำการส่งบล็อคข้อมูลขนาด 120 ไบท์ แต่ผู้รับสัญญาณจะทำการรับบล็อคข้อมูลไป 200 ไบท์ ตัวผู้รับจะทำการนับไบท์ถัดไปอีก 80 ไบท์ หลังจากที่ได้รับส่วนแรกของบล็อคข้อมูล ซึ่งจะก่อให้เกิดความวุ่นวายเป็นอย่างมากในการติดต่อสื่อสาร

สิ่งที่ควรจำก็คือ ที่ระดับของไบท์ ผู้รับสัญญาณจะทำการส่งข้อมูลในทุกๆ ไบท์ไปยังผู้ส่งสัญญาณ ดังนั้นโปรโตคอลในระดับของบล็อคจะรู้ในทันทีว่าเฮดเดอร์ส่วนใดเหมาะสมที่จะทำการถ่ายโอน แต่มีข้อเสียที่ว่า โปรโตคอลจะไม่ทราบถึงตัวผู้รับสัญญาณ ถึงแม้ว่าจะได้รับส่วนของเฮดเดอร์ที่ถูกต้องก็ตาม ดังนั้นผู้ส่งสัญญาณจะต้องแสดงให้ผู้รับสัญญาณได้ทราบ โดยการส่งคาแร็กเตอร์มาตรฐาน ดังนั้นผู้รับสัญญาณจะไม่ก็จะไม่มีการอ้างถึงส่วนของสัญญาณที่ไม่ถูกต้องนั่นคือจะไม่เกิดความผิดพลาด หรือ ERROR ในส่วนของเฮดเดอร์ขึ้นนั่นเอง

ในการส่งสัญญาณกลับคืนมานั้น เมื่อการถ่ายโอนข้อมูลประสบความสำเร็จ ผู้ส่งสัญญาณจะส่งคาแร็กเตอร์ ACK (Acknowledge) หรือส่งคาแร็กเตอร์ NAK (Non-Acknowledge) เมื่อเกิดความผิดพลาด ซึ่งคาแร็กเตอร์ ACK จะไม่ต่อเข้ากับชื่อของส่วนพินพอร์ท ถึงแม้ว่าจะมีฟังก์ชั่น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

พื้นฐานที่เหมือนกัน โดยในโปรแกรมนี้ คาแร็กเตอร์ ACK และ NAK จะถูกแทนที่ด้วยรหัส 00H และ FFH ตามลำดับ

แต่อย่างไรก็ตามก็สามารถเปลี่ยนไปใช้รหัสอื่นๆ ได้ตามต้องการ คาแร็กเตอร์เหล่านี้จะแสดงในส่วนโปรโตคอลการสื่อสาร ดังนั้นจึงจะมีการตรวจสอบที่ระดับของไบท์ เพื่อให้การถ่ายโอนข้อมูลประสบความสำเร็จ

เมื่อผู้รับสัญญาณได้รับส่วนของเฮดเดอร์และคาแร็กเตอร์ ACK ที่ตามมา โดยไม่เกิดความผิดพลาด ผู้ส่งสัญญาณจะสามารถเริ่มต้นการถ่ายโอนบล็อคข้อมูลที่แท้จริง เมื่อมีปัญหาเกิดขึ้น ผู้ส่งจะทำการส่งส่วนเฮดเดอร์ซ้ำอีกครั้งหนึ่ง เพราะว่าได้รับคาแร็กเตอร์ NAK จากการพยายามส่งข้อมูลครั้งแรก เมื่อผู้รับสัญญาณได้รับส่วนของเฮดเดอร์และคาแร็กเตอร์ ACK มาครั้งหนึ่งแล้ว จะทำการรวบรวมข้อมูลบนบล็อคข้อมูลที่รับได้ ซึ่งจะเห็นได้ว่า เราจะใช้รูปแบบของบิทที่แตกต่างกันมากสำหรับสองคาแร็กเตอร์ โดยคาแร็กเตอร์ ACK จะเปลี่ยนไปเป็นคาแร็กเตอร์ NAK เมื่อการถ่ายโอนข้อมูลเกิดความผิดพลาด

ถ้ามีเกิดความผิดพลาดขึ้นอย่างต่อเนื่อง ผู้ส่งสัญญาณจะไม่พยายามที่จะทำการส่งส่วนเฮดเดอร์อีกต่อไป โดยจะทำการเซ็ตค่า MAXTRY ให้คงที่ เพื่อเป็นการบอกให้ผู้ส่งสัญญาณรู้ว่ามีจำนวนของความผิดพลาดเกิดขึ้นกี่ครั้ง ก่อนที่การส่งบล็อคข้อมูลจะเกิดความล้มเหลว ส่วนของคาแร็กเตอร์ ACK และ NAK จะใช้เพื่อยืนยันการรับบล็อคข้อมูล คาแร็กเตอร์ที่ส่งกลับจะถูกส่งไปยังตัวรับสัญญาณเท่านั้น หลังจากที่บล็อคข้อมูลถูกส่งไปหมดแล้ว

ในวิธีการเหล่านี้ จะสามารถค้นหาความผิดพลาดที่เกิดขึ้นจากการติดต่อสื่อสารได้ทุกๆ ชนิด โปรโตคอลจะทำลายการเกิด checksum ซึ่งเป็นวิธีของการตรวจสอบความผิดพลาดที่เกิดจากการถ่ายโอนข้อมูลในซอฟแวร์การสื่อสารชนิดอื่น

4.6.5 การอ่านสถานะ <Esc>

เมื่อมีการถ่ายโอนข้อมูลครั้งหนึ่งแล้ว การส่งไบท์สุดท้ายจะเป็นการแสดงการเสร็จสิ้นขั้นตอนการทำงาน แต่ในไบท์นี้จะส่งจากผู้รับไปยังผู้ส่งสัญญาณ นั่นก็คือ ผู้รับจะทำการส่งสัญญาณ ESCAPE ไปยังผู้ส่งสัญญาณ ถึงแม้ว่าสิ่งเล่านี้อาจจะไม่จำเป็นมากนัก เพราะตัวผู้รับสัญญาณสามารถออกจากซอฟท์แวร์การสื่อสารได้ตลอดเวลา โดยการใช้ ESCAPE และจำปล่อยให้ผู้ส่งสัญญาณรอรับความผิดพลาดที่เกิดจาก time out

โดยเหตุที่ว่าวิธีนี้ไม่ใช่วิธีการแก้ไขปัญหาที่ดีที่สุด โปรโตคอลจะให้ผู้ส่งทำการรอไบท์ ESCAPE จากผู้รับหลังจากที่มีการถ่ายโอนบล็อคข้อมูลอย่างสมบูรณ์ ถ้าผู้ส่งสัญญาณได้รับค่า TRUE มันจะทำการออกจากโปรแกรมรวมเข้าด้วยกับข้อความที่เหมาะสม หรือถ้าไม่เป็นเช่นนั้น ผู้ส่งสัญญาณก็จะทำงานด้วยโปรแกรมดังกล่าวต่อไป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สำหรับข้อความชนิดนี้ จะต้องเตรียมทิศทางการติดต่อสื่อสารไว้ทั้งสองทิศทาง ตั้งแต่ ผู้ส่งสัญญาณสามารถตัดสินใจที่จุยุติการติดต่อสื่อสารได้ตลอดเวลา ซึ่งโปรซีเดอร์นี้จะแตกต่างไปจากผู้รับสัญญาณ เมื่อเริ่มต้นการติดต่อสื่อสาร เส้นทาง SendABlock จะเป็นตัวกำหนดว่าควรทำการกดคีย์ <Esc> หรือไม่ ถ้ามีการกดคีย์ดังกล่าว มันจะทำการส่งตัว token เท็จไปแทนที่ส่วนเฮดเดอร์ของบล็อคข้อมูลที่แท้จริง โดย token เท็จนี้จะรู้ได้ที่ตัวรับสัญญาณ เมื่อตัวรับสัญญาณยอมรับ token นี้แล้ว มันจะตัดสินใจว่าเป็นสัญญาณที่ให้ออกจากตัวโปรแกรม

โดยการสร้างกลไก ESCAPE เข้าไปในโปรโตคอลของบล็อค เราสามารถที่จะหลีกเลี่ยงการเกิดการคอยคิวของ ESCAPE อย่างถาวรที่ระดับสูงกว่าได้ ในโปรแกรมนี้จะเกี่ยวข้องกับระดับของไฟล์ที่เหนือกว่าระดับของบล็อค ในระดับของไฟล์ จะใช้โปรโตคอลในระดับบล็อค เพื่อถ่ายโอนส่วนของไบท์ทั้งหมดแยกไปเป็นส่วนๆ เราไม่ต้องเข้าไปลึกถึงรายละเอียดเกี่ยวกับระดับของไฟล์ ในเมื่อรายการของโปรแกรมดังกล่าวนี้เป็นเนื้อหาที่สมบูรณ์อยู่แล้ว

จำไว้ว่าเส้นทางที่ระดับของไบท์และบล็อค สามารถใช้ในการถ่ายโอนได้ทุกๆ ข้อมูลระหว่างผู้รับและผู้ส่งสัญญาณ ดังนั้นทั้งผู้รับและผู้ส่งสัญญาณมีโอกาสที่จะเกิดความล้มเหลวในการติดต่อสื่อสารได้ทุกๆ จุด ซึ่งเส้นทางเหล่านี้จะสามารถเก็บรักษาไว้เหมือนกับเป็นฐานข้อมูลของโปรแกรมการถ่ายโอนข้อมูลของเรา ซึ่งสามารถที่จะแข่งขันกับรูปแบบการถ่ายโอนข้อมูลแบบแพคเก็ตได้ เช่นเดียวกับ LapLinK® เพราะพวกเขายังมีส่วนที่ยังไม่สมบูรณ์อยู่ นั่นก็คือ ความเร็ว เนื่องจากสิ่งเหล่านี้ยังต้องการเส้นทางในระดับไบท์ทั้งหมดที่เขียนด้วยภาษาแอสแซมบลีอย่างสมบูรณ์แบบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 5

การประยุกต์ใช้งานโปรแกรม Visual Basic for Windows

5.1 Visual Basic for Windows

Visual Basic for Window หรือ VB พัฒนาขึ้นโดยบริษัทไมโครซอฟต์ มีจุดประสงค์หลักก็คือ การสร้างและพัฒนาแอพพลิเคชั่นบนวินโดวส์โดยเฉพาะ VB ใช้ประโยชน์จากคุณสมบัติและสภาพแวดล้อมของวินโดวส์ในการออกแบบแอพพลิเคชั่น ทำให้เขียนโปรแกรมด้วย VB สั้น และกระทัดรัด เมื่อเทียบกับการเขียนโปรแกรมด้วยภาษาอื่น เช่น Turbo Pascal, C++ หรือ Small Talk ซึ่งต้องเขียนโปรแกรมยาวกว่า เพื่อให้ได้รูปลักษณ์ของหน้าต่าง (window) แบบเดียวกัน

VB ที่นิยมใช้ในปัจจุบันคือ Visual Basic 3.0 (VB 3.0) ซึ่งพัฒนามาจาก Visual Basic 2.0 (VB 2.0) VB 3.0 มีคุณลักษณะเด่นเพิ่มเติมจาก VB 2.0 หลายประการ ดังจะสรุปคุณลักษณะทั่วๆ ไป และที่น่าสนใจของ VB 3.0 และ 2.0 ดังนี้

- ขนาดของโปรแกรมไม่ใหญ่จนเกินไป กล่าวคือ VB 2.0 ใช้เนื้อที่ฮาร์ดดิสก์ประมาณ 4.8 MB ซึ่งจะรวมตัวอย่างของแอพพลิเคชั่นและตัวอย่างไอคอนแบบต่างๆ ส่วน VB 3.0 ใช้เนื้อที่ฮาร์ดดิสก์ประมาณ 24 MB เพราะเพิ่มชุดคำสั่งเกี่ยวกับการจัดการฐานข้อมูล, Window API และตัวอย่างไฟล์รูปแบบ bitmap ซึ่งนำมาใช้ในฟอร์มของ VB
- ใช้กับไมโครซอฟต์วินโดวส์ 3.0 ขึ้นไป
- ออกแบบมาเพื่อใช้สร้างแอพพลิเคชั่นบนวินโดวส์ และใช้ประโยชน์จากสภาพแวดล้อมในรูปแบบกราฟฟิกที่เรียกว่า Graphic User Interface - GUI โปรแกรม VB จึงสั้น เข้าใจง่าย และไม่ต้องใช้งานร่วมกับซอฟต์แวร์ทูลที่ใช้ในการพัฒนาแอพพลิเคชั่นบนวินโดวส์ เช่น Software Development Kit - SDK ของ Microsoft C++ หรือ Turbo Vision
- มีระบบ Online help และตัวอย่างโปรแกรมแบบ Card file เพื่อสอนวิธีการใช้โปรแกรม ซึ่งจะเริ่มจากตัวอย่างโปรแกรมของแอพพลิเคชั่นแบบง่ายๆ จนถึงตัวอย่างโปรแกรมของแอพพลิเคชั่นที่ซับซ้อน
- สามารถเรียกใช้ Window API (Applocation Program Interface) ซึ่งเป็นหัวใจการทำงานของวินโดวส์ และนำไอคอนจาก Icon Library ของ VB มาใช้ในแอพพลิเคชั่น ซึ่งมีให้เลือกใช้มากมาย
- ตัวแปลเป็นแบบคอมไพเลอร์ ทำให้สามารถเรียกใช้โปรแกรมย่อยร่วมกับแอพพลิเคชั่นอื่นๆ ขณะทำงานได้ และในการคอมไพเลอร์จะไม่รวมโปรแกรมย่อยเข้าไปใน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โปรแกรมหลัก แต่แยกคอมไพล์เก็บไว้ใน Dynamic Link Library - DDL ทำให้ไม่สามารถใช้เนื้อที่บนหน่วยความจำได้มาก

- คำสั่งและฟังก์ชั่นการใช้งานคล้ายคลึงกับ Quick Basic แต่ออกแบบให้ใช้งานได้ง่ายกว่า และเพิ่มคำสั่งเพื่อใช้งานบนวินโดวส์
- โปรแกรมในส่วนที่ติดต่อกับผู้ใช้งาน (User Interface) สามารถวาดได้โดยตรงโดยใช้ Toolbox วาดลงบนหน้าต่างที่เรียกว่า "form" และกำหนดคุณสมบัติต่างๆ ของฟอร์มจากหน้าต่างคุณสมบัติ (Properties Window) แทนที่ต้องเขียนโปรแกรมสร้างหน้าต่างให้ยุ่งยาก นอกจากนี้ยังสามารถโหลดไฟล์รูปภาพของแอพพลิเคชั่นอื่นมาบรรจุในฟอร์มได้ด้วย
- วิธีการเขียนโปรแกรมเป็นแบบ Event-driven Programming คือเราจะเขียนโปรแกรมให้ทำงานต่างๆ ตามลักษณะเหตุการณ์ที่เกิดขึ้นจากการกระทำของผู้ใช้งานเป็นขั้นตอนไปเรื่อยๆ ซึ่งเป็นลักษณะของ GUI เช่น การคลิกเมาส์เลือกคำสั่ง การกดคีย์บอร์ด หรือการเลือกเมนูคำนั่ง
- โปรแกรมที่เขียนด้วย VB สามารถใช้ระบบแลกเปลี่ยนหรือเชื่อมโยงข้อมูลกับแอพพลิเคชั่นต่างๆ บนวินโดวส์ ในลักษณะที่เรียกว่า Dynamic Data Exchange - DDE และ Object Linking and Embedding - OLE
- สามารถรันโปรแกรมให้ทำงานทีละส่วน หรือกำหนดจุดหยุดในโปรแกรม เพื่อตรวจหาและแก้ไขจุดผิดพลาดในโปรแกรม รวมทั้งมี Debugging Tools ช่วยอำนวยความสะดวกในการทำงานดังกล่าว
- มีซอฟต์แวร์เพิ่มขีดความสามารถของ VB เพิ่มเติมหลายตัว ได้แก่ Control Development Kit, Button Tool, Custom Control Factory และ Quick Pak Professional เป็นต้น

5.2 การสร้างและการใช้งาน Dynamic Link Library

ในส่วนนี้จะกล่าวถึงรายละเอียดโดยทั่วไปของ Dynamic Link Library ได้แก่ ความหมายของ DLL การเขียนโปรแกรมสร้าง DLL และการนำ DLL ไปใช้งานในโปรแกรม Visual Basic

5.2.1 ความหมายของ DLL

Dynamic Link Library หรือ DLL คือกลุ่มของไฟล์หรือไบนารีที่เก็บชุดคำสั่งโปรแกรมย่อยหรือข้อแตกต่างหรือข้อกำหนดต่างๆ ซึ่งแอพพลิเคชั่นหรือโปรแกรมสามารถนำไปใช้งานได้ในระบบปฏิบัติการวินโดวส์ DLL จะจัดเก็บเป็นไฟล์อยู่ในไดเรกทอรีต่างๆ ทั้งไดเรกทอรีของวินโดวส์เองและภายในไดเรกทอรีของแอพพลิเคชั่นอื่นบนวินโดวส์ สำหรับ DLL ที่อยู่ในไดเรกทอรีหลักและซิสเต็มไดเรกทอรีของวินโดวส์จะถือว่าเป็นหัวใจสำคัญในการกำหนดพารามิเตอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

และฟังก์ชั่นการทำงานต่างๆ ในสภาพแวดล้อมที่เป็น GUI ของวินโดวส์ ตัวอย่างไฟล์ DLL ของวินโดวส์ได้แก่ ไฟล์ DLL ของระบบ เช่น KRNL x86.EXE, GUI.EXE หรือ USER.EXE, ไฟล์ DLL ของไดรเวอร์ต่างๆ เช่น KEYBOARD.DRV หรือ SYSTEM.ERV, ไฟล์ DLL ของฟอนต์ของตัวอักษร ได้แก่ ไฟล์นามสกุล.FON ต่างๆ และไฟล์ DLL มาตรฐานทั่วไป ซึ่งเป็นไฟล์นามสกุล.DLL เป็นต้น โดยปกติเมื่อทำการรันวินโดวส์ ไฟล์นามสกุล .DLL เหล่านี้จะโหลดมายังหน่วยความจำอัตโนมัติ

ในตอนเริ่มแรกที่ทำการติดตั้งวินโดวส์ในระบบไฟล์ INSTALL.EXE ของวินโดวส์จะสร้างไฟล์ DLL ต่างๆ มากมาย ไฟล์ DLL เหล่านี้จะเป็นไฟล์ DLL ของวินโดวส์โดยเฉพาะ อย่างไรก็ตามสำหรับแอพพลิเคชั่นต่างๆ บนวินโดวส์จะมีไฟล์ DLL เฉพาะที่ใช้งานภายในแอพพลิเคชั่นเช่นกัน ไฟล์ DLL ของแอพพลิเคชั่นจะเกิดขึ้น เมื่อทำการติดตั้งแอพพลิเคชั่นในระบบ หรือเมื่อทำการเขียนโปรแกรมสร้างไฟล์ DLL ที่เรียกว่า Custom DLL ไว้ใช้งานโดยเฉพาะภายในแอพพลิเคชั่นก็ได้ สำหรับในภาษาโปรแกรม Visual Basic เราสามารถนำ Custom DLL มาใช้งานภายในโปรแกรมที่เขียนขึ้นมาเพื่อเพิ่มความสามารถของการสร้างแอพพลิเคชั่นบนวินโดวส์เช่นเดียวกับการเรียกใช้ API ของวินโดวส์บางตัว

5.2.2 รูปแบบของ DLL

ไฟล์ DLL ที่นำมาใช้งานในแอพพลิเคชั่นหรือโปรแกรมโดยปกติจะมีอยู่ 2 รูปแบบคือ

- ไฟล์ DLL ที่เป็นชุดคำสั่งหรือโปรแกรมย่อย
- ไฟล์ DLL ที่เป็นข้อกำหนดต่างๆ เช่น ข้อกำหนดในการจัดสรรทรัพยากร การกำหนดพารามิเตอร์ของระบบ การกำหนดอุปกรณ์ฮาร์ดแวร์ ข้อกำหนดของตัวแปรและฟังก์ชั่นการทำงาน เป็นต้น โดยในส่วนนี้จะกล่าวถึงเฉพาะไฟล์ DLL ที่เป็นชุดคำสั่งหรือโปรแกรมย่อยเท่านั้น

5.2.3 ข้อดีของการใช้งาน DLL

ข้อดีของการใช้งาน DLL เมื่อเทียบกับโปรแกรมย่อยทั่วๆ ไปที่เขียนขึ้นมาเพื่อแยกการทำงานออกจากโปรแกรมหลักในการเขียนโปรแกรมโดยทั่วไป พอสรุปได้ดังนี้

- แอพพลิเคชั่นหรือโปรแกรมต่างๆ ภายในแอพพลิเคชั่นสามารถใช้งาน DLL ร่วมกันได้
- การทำงานของแอพพลิเคชั่นหรือโปรแกรมที่เรียกใช้ DLL โดยทั่วไปจะเร็วกว่าโปรแกรมที่เรียกใช้โปรแกรมย่อยที่เขียนขึ้นภายในโปรแกรมโดยตรง เพราะว่าการคอมไพล์และโหลด DLL จะไม่ซ้ำซ้อนกัน
- ประหยัดเนื้อที่ในหน่วยความจำ
- การแก้ไขหรือเปลี่ยนแปลง DLL เพื่อปรับปรุงและพัฒนาแอพพลิเคชั่นหรือโปรแกรมทำได้สะดวกและรวดเร็ว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- เราสามารถสร้าง Custom DLL โดยใช้ภาษาโปรแกรมต่างๆ ได้หลายภาษา เช่น Basic, Pascal, C หรือ C++ แต่โดยทั่วไปเรานิยมใช้ภาษา C หรือ C++ เพราะเป็นภาษาที่มีประสิทธิภาพสูงในการเขียนโปรแกรมใช้งานบนวินโดวส์

5.2.4 การเขียนโปรแกรมสร้าง DLL

ในหัวข้อนี้จะกล่าวถึงวิธีการเขียนโปรแกรมสร้าง Custom DLL ของชุดคำสั่งหรือโปรแกรย่อย เพื่อกำหนดฟังก์ชั่นการทำงานที่ต้องการโดยใช้ภาษาโปรแกรม C หรือ C++ โดยในการเขียนนั้นไม่ต้องกังวลเกี่ยวกับไวยากรณ์ของคำสั่งหรือถ้อยแถลงต่างๆ ของภาษา C หรือ C++ ที่ปรากฏภายในตัวอย่างโปรแกรมเกี่ยวกับ DLL ภายในหัวข้อนี้ เนื่องจากเป็นเพียงการแสดงรูปแบบหรือโครงสร้างมาตรฐานของ Custom DLL เพื่อเป็นแนวทางสำหรับการเขียนโปรแกรม DLL ที่ใช้งานทั่วไปๆ (ผู้เขียนโปรแกรมสร้าง Custom DLL ควรมีความรู้ ความชำนาญทางด้านโปรแกรมภาษา C หรือ C++ และระบบคอมพิวเตอร์ดีพอสมควร เพราะการเขียนโปรแกรมสร้าง DLL เป็นงานโปรแกรมที่ซับซ้อนและต้องใช้ชุดคำสั่งที่เกี่ยวข้องกับระบบปฏิบัติการโดยตรง ดังนั้นอาจจะเกิด error ที่มีผลต่อระบบคอมพิวเตอร์ได้) โดยสามารถที่จะทำความเข้าใจในการทำงานของตัวอย่างโปรแกรมภาษา C หรือ C++ อย่างคร่าวๆ โดยการอ่านคำอธิบาย ซึ่งเขียนอยู่ภายในเครื่องหมาย /*, */ หรือ // ซึ่งมีรูปแบบดังตัวอย่างต่อไปนี้

```
/* This is a comment for C or C++ programming language. */
// This is a comment for C++ programming langualge only.
```

สำหรับ DLL source code ที่ถูกคอมไพล์เป็น object code ด้วยคอมไพล์ของ C หรือ C++ ซึ่งสามารถรันบนวินโดวส์ได้ เช่น Microsoft C++, Quick C for Windows, Borland C++ หรือ Turbo C++ for windows เป็นต้น

5.2.5 โครงสร้างของ Custom DLL

โดยทั่วไปภายใน Custom DLL ที่เป็นชุดคำสั่งหรือโปรแกรมย่อย (procedure) จะประกอบไปด้วยส่วนของโปรแกรมที่สำคัญ 3 ส่วนหลักๆ ดังนี้

- **ส่วนของ Header** เป็นส่วนของ Custom DLL ที่เรียกใช้ไฟล์ header ต่างๆ (ไฟล์นามสกุล .h) เพื่อนำมาใช้งานใน Custom DLL ได้แก่ไฟล์ header ของวินโดวส์ (window.h), ไฟล์ header ของดอส (dos.h) เป็นต้น ในการเรียกไฟล์ header ดังกล่าว จะใช้ถ้อยแถลง include ดังแสดงได้ดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
# include <windows.h>
# include <dos.h>
```

นอกจากนี้ภายในส่วนของ header ยังมีส่วนของการกำหนดฟังก์ชั่นต่างๆ ของ prototype เพื่อกำหนดให้แอพพลิเคชั่นหรือโปรแกรมอื่นสามารถเรียกใช้ฟังก์ชั่นจาก Custom DLL นี้ เราจะกำหนดฟังก์ชั่นของ prototype โดยใช้ถ้อยแถลง exterm ดังตัวอย่างต่อไปนี้

```
extern "C" long FAR PASCAL -export SumOf (long x, long y)
```

(ในตัวอย่างเป็นการกำหนดให้แอพพลิเคชั่นหรือโปรแกรมอื่นสามารถเรียกใช้ฟังก์ชั่น Sumof ( ) ที่กำหนดใน Custom DLL ได้)

- ส่วนของโปรแกรมหลัก ซึ่งประกอบด้วยโปรแกรมหลัก และอาจมีโปรแกรมย่อยสำหรับกำหนดฟังก์ชั่น prototype ที่กำหนดในส่วนของ header ก็ได้ โดยปกติโปรแกรมหลักจะเขียนอยู่ภายในฟังก์ชั่น "LibMain ( )" เมื่อ Custom DLL ถูกโหลดขึ้นมา วินโดวส์จะประมวลผลถ้อยแถลงต่างๆ ในโปรแกรมหลักโดยอัตโนมัติ สำหรับในส่วนของโปรแกรมย่อยกำหนดฟังก์ชั่น prototype เราจะเขียนอยู่ภายในฟังก์ชั่น "-export" ตัวอย่างต่อไปนี้แสดงส่วนของโปรแกรมหลักที่เขียนอยู่ภายในฟังก์ชั่น LibMain ( ) และ -export

ตัวอย่างโปรแกรมหลัก

```
int FAR PASCAL LibMain (HINSTANCE hInstance, WORD
    wDataSegment,WORD wHeapSine,tPSTR 1 ps = (mdLine)
{
    if (wHeapSine ! = 0)
        UnlockData (0);
    return 1 ;
}
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตัวอย่างของโปรแกรมย่อยที่อยู่ในส่วนของโปรแกรมหลัก

```
double FAR PASCAL_export Pmt(double prin,double rate,int periods)
{
    /*Returns the periodic payment on a loan.
    prin = principal (amount borrowed)
    rate = Interest rate per period
    periods = number of periods
    /*
    double Result ;
    Result = (prin * rate) / (1.0 - pow (rate + 1.0,
        (-periods)));
    return (Result);
}
```

- ส่วนที่เป็น **Exit Procedure** (สำหรับวินโดวส์จะเรียกว่า Windows Exit Procedure - WEP) จะเป็นโปรแกรมย่อยที่ใช้ระบุการทำงานหลังสิ้นสุดการใช้งาน DLL สำหรับใน Custom DLL ที่รันบนวินโดวส์ การใช้งาน WEP มีรูปแบบดังนี้

```
Int FAR PASCAL WEP (int bSystemExit)
{
return 1 ;
}
```

5.2.6 ไฟล์ Module Definition

ในบางครั้งจำเป็นต้องมีการกำหนดรายละเอียดของการคอมไพล์ Custom DLL รายละเอียกเหล่านี้โดยปกติจะเขียนเป็นเท็กซ์ไฟล์แยกต่างหากจาก Custom DLL ที่เรียกว่า Module Defmition ดังแสดงในตัวอย่างต่อไปนี้

```
LIBRARY        Filename
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

DESCRIPTION	'Financial functions DLL'
EXETYPE	WINDOWS
CODE	PRELOAD MOVEABLE DISCARDABLE
DATA	PRELOAD MOVEABLE SIGNAL
HEAPSIZE	1024

ตัวอย่างต่อไปนี้จะแสดงโครงสร้างของ Custom DLL ที่เขียนด้วยภาษาโปรแกรม C หรือ ภาษา C++

```
#include <windows.h>
/*Other needed #include statements go here. */
/*Function prototype go here (if they are not in their own include file). */
/*Actual DLL functions go either here or below. */
int FAR PASCAL LibMain (HINSTANCE hInstance, WORD wDataSegment,
WORD wHeapsize, LPSTR lpszCmdLine)
{
    if  (wHeapSize ! + 0)
        UnlockData (0);
   return 1 ;
}
/*You may not need to include the WEP function in the source code
   because some compilers add it to the DLL automatically.
*/
int FAR PASCAL WEP (int bSystemExit)
{
    return 1 ;
}
/*DLL functions can go here. */
```

สังเกตว่าชื่อที่กำหนดให้กับ "LIBRARY" จะเป็นชื่อที่อ้างอิงถึงชื่อไฟล์ Custom DLL ที่สร้างขึ้น ในการนำไฟล์ module definition ไปใช้งาน เราจะ "Add" ไฟล์ module definition ดัง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กล่าวในโปรเจ็กต์ของ C หรือ C++ ที่ใช้สร้าง Custom DLL ในลักษณะเดียวกับการ "Add" ไฟล์โมดูลของ VB (ไฟล์นามสกุล .BAS) ในโปรเจ็กต์ที่ใช้สร้างแอพพลิเคชั่นบนวินโดวส์

5.2.7 การนำ Custom DLL ไปใช้งานภายในโปรแกรม Visual Basic

เช่นเดียวกับการเรียกใช้โปรแกรมย่อยแบบ API ของวินโดวส์ โดยการเรียกใช้หรือการนำ Custom DLL ของวินโดวส์หรือแอพพลิเคชั่นต่างๆ มาใช้งานภายในโปรแกรม VB จะใช้ถ้อยแถลง ซึ่งมีรูปแบบดังนี้

```
Declare Function FunctionName Lib DLLName [([ArgumentList])] [As Type]
```

โดยที่ *FunctionName* เป็นชื่อของฟังก์ชั่น prototype ที่กำหนดใน Custom DLL หลังถ้อยแถลง -export ที่ต้องการ

DLLName เป็นชื่อพาทกำหนดตำแหน่งของไฟล์ Custom DLL รวมทั้งชื่อไฟล์ Custom DLL (ไฟล์นามสกุล .DLL) ที่ใช้งาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 6

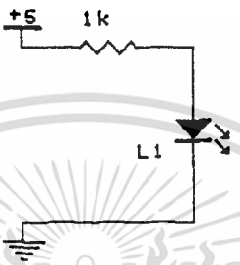
การออกแบบและการทำงาน

6.1 การออกแบบวงจรที่ใช้กับเครื่องตรวจกระดาษคำตอบ

6.1.1 วงจรชุดตรวจจับแสงอินฟราเรด

ชุดตรวจจับแสงอินฟราเรดนั้นสามารถที่จะแบ่งออกเป็น 2 ส่วน คือ ภาคส่ง และภาครับ

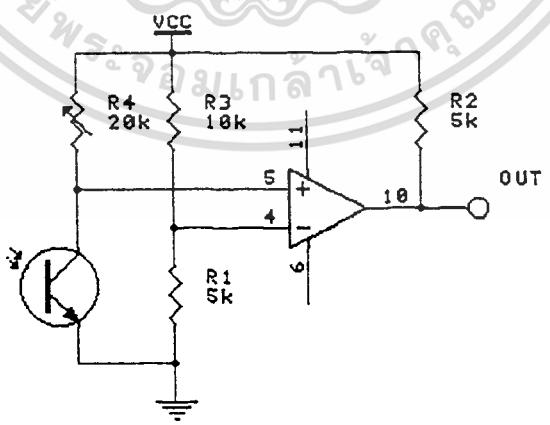
1. วงจรภาคส่ง



รูปที่ 6.1 แสดงวงจรภาคส่งแสงอินฟราเรด

ภาคส่งแสงอินฟราเรดนั้นใช้หลอด LED เป็นตัวส่งสัญญาณแสงให้กับภาครับที่เป็น PHOTOTRANSISTOR โดยหลอด LED นั้นจะต้องต่อกับตัวความต้านทานด้วยเพื่อเป็นการลดแรงดันตกคร่อมหลอด LED ให้เหมาะสม

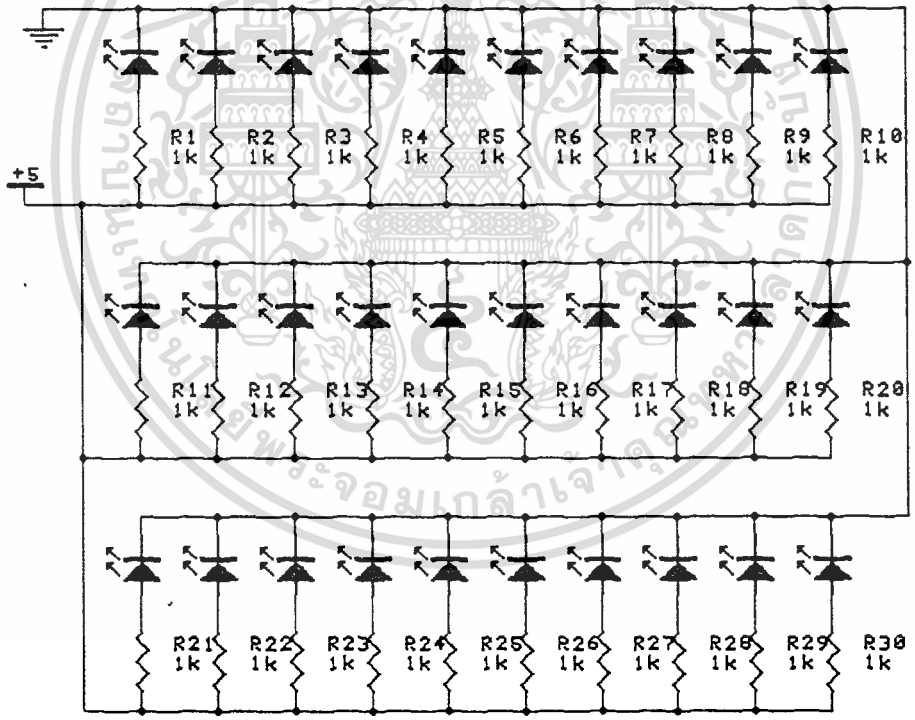
2. วงจรภาครับ



รูปที่ 6.2 แสดงวงจรภาครับแสงอินฟราเรด

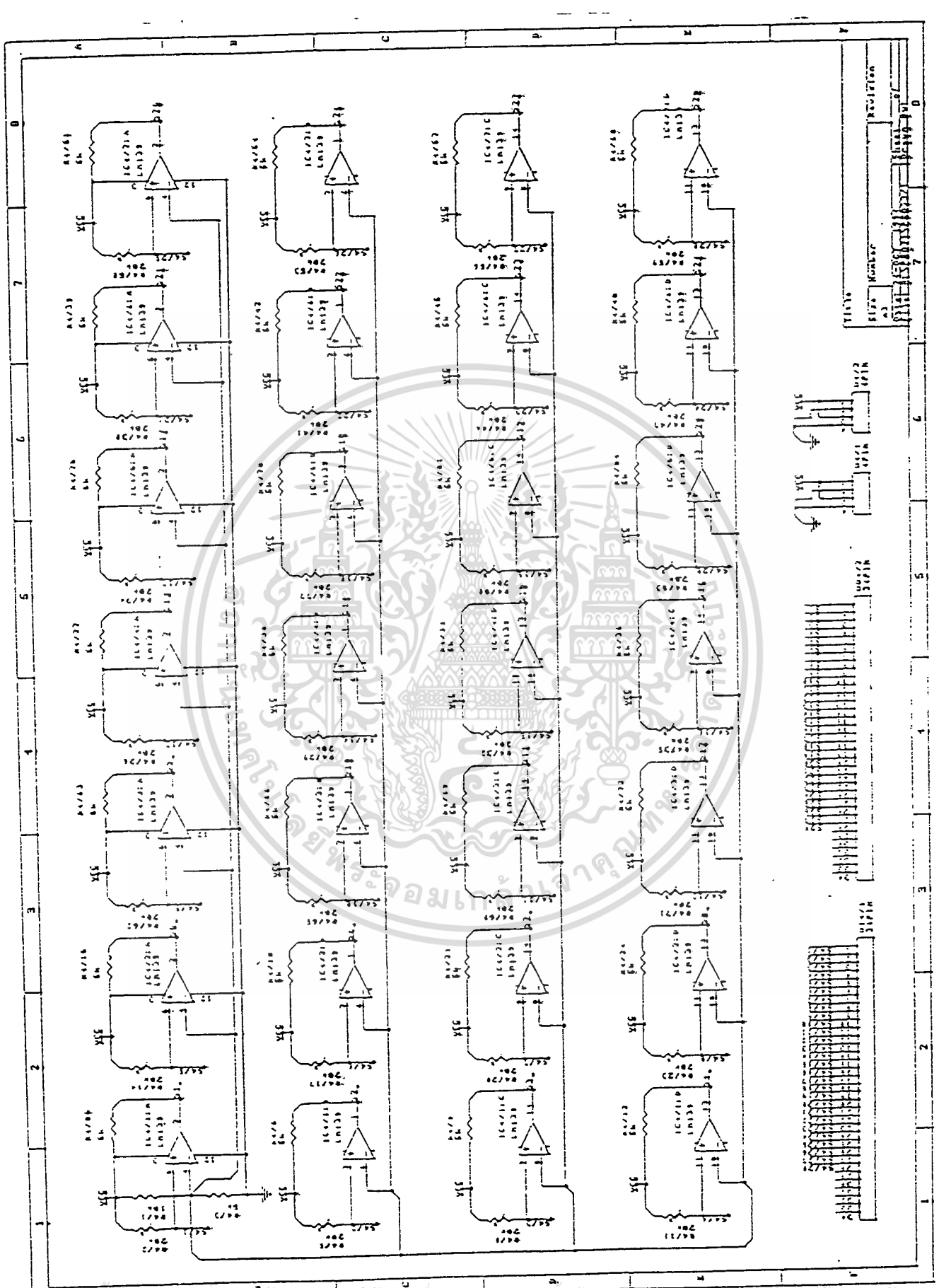
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาครับแสงอินฟราเรดใช้ตัว PHOTOTRANSISTOR เป็นตัวรับแสงอินฟราเรดจากภาคส่ง โดยการทำงานของ PHOTOTRANSISTOR นั้นเมื่อมีแสงมาตกกระทบตัวPHOTOTRANSISTOR จะทำให้ทรานซิสเตอร์นำกระแสทำให้แรงดันที่ตกคร่อมตัวออปแอมป์ทางด้านขาบวกมีค่าเปลี่ยนแปลงไป โดยเมื่อ PHOTOTRANSISTOR ทำงานจะทำให้แรงดันที่ตกคร่อมมีการเปลี่ยนแปลงไปตามปริมาณแสงที่ได้รับ คือเมื่อมีปริมาณแสงมากกระแสที่ไหลผ่านตัว PHOTOTRANSISTOR ก็จะมีกระแสเพิ่มขึ้น ซึ่งเมื่อ PHOTOTRANSISTOR นำกระแสก็จะทำให้แรงดันที่เข้าทางขาบวกของออปแอมป์มีค่าน้อยกว่าทางด้านขาลบทำให้เอาต์พุทของออปแอมป์มีค่าเป็น “0” R_4 ที่ต่อจากวงจรจะเป็นตัวปรับค่าความไวของวงจร



รูปที่ 6.3 แสดงวงจรรวมภาคส่งแสงอินฟราเรด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 6.4 แสดงวงจรรวมภาครับสัญญาณอินฟราเรด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

6.1.2 วงจรการนับจำนวนของข้อสอบที่ผ่านเข้าเครื่องตรวจ

การทำงานของวงจรนับแบ่งออกเป็น 3 ภาคใหญ่ๆ คือ ภาคนับความถี่แบบ BCD ภาคถอดรหัส BCD เป็นรหัส LED 7 ส่วน และภาคแสดงผล

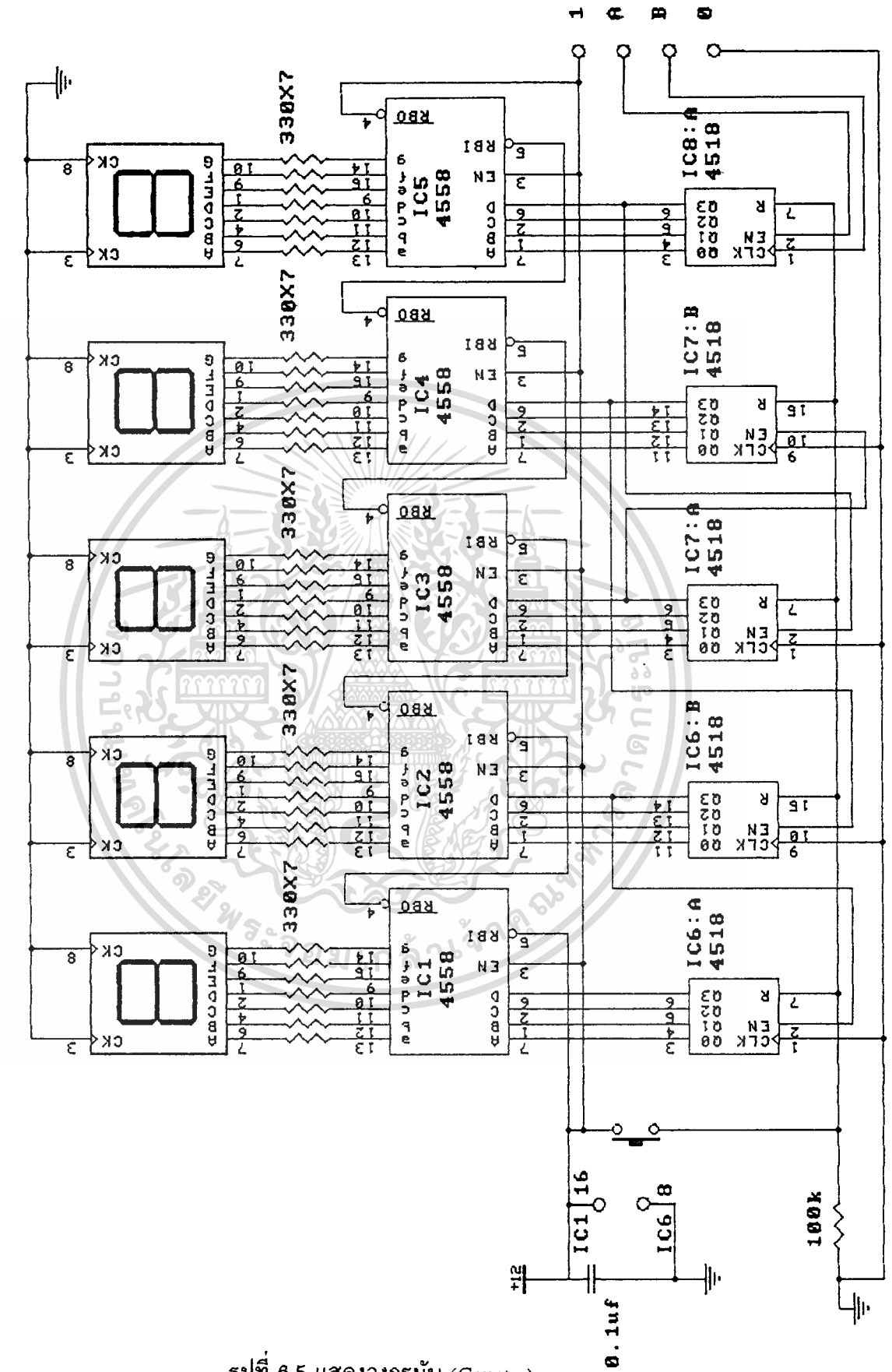
ในภาคนับความถี่แบบ BCD ใช้ไอซีเบอร์ MC14518 ซึ่งภายในมีวงจรนับแบบ BCD 4 บิต อยู่ 2 ชุด ดังนั้นวงจรนี้จึงใช้ไอซี MC14518 เพียง 2 ตัว สำหรับตัวเลข 4 หลัก จากวงจรตามรูป $IC_{6/2}$ จะนับสัญญาณดิจิตอลที่เข้ามาจนถึงค่าสูงสุด ก็จะส่งสัญญาณทดไปยังหลักหน้าถัดไป และจะกลับไปเริ่มต้นนับใหม่

สำหรับในภาคถอดรหัส BCD เป็นรหัส LED 7 ส่วน ใช้ไอซีเบอร์ MC14558 5 ตัว ตามจำนวนหลักที่ใช้ ซึ่งจะรับรหัส BCD 4 บิตจาก IC_5 และ IC_6 แปลงเป็นเลขฐานสิบ (0 ถึง 9) ส่วนภาคแสดงผลนั้นเพียงแต่รับสัญญาณเข้าไปที่ส่วนไหนส่วนนั้นก็จะสว่าง

6.1.3 วงจรควบคุม DC.MOTOR

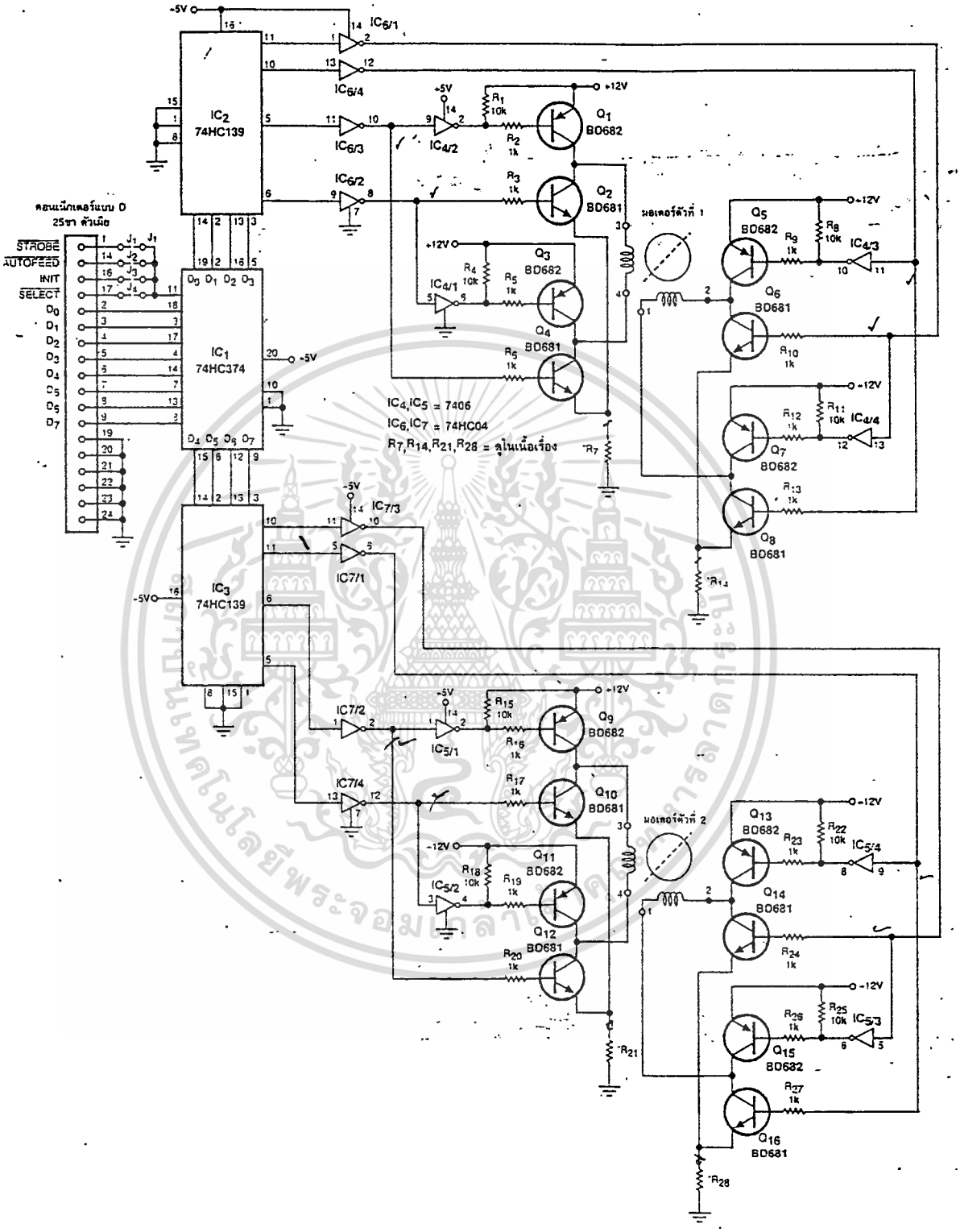
เป็นวงจรที่ควบคุมให้ DC.MOTOR สามารถทำงานและหยุดทำงานได้ โดยวงจรนี้เป็นวงจรที่สามารถควบคุม DC MOTOR ได้จากเครื่องคอมพิวเตอร์ หลักการทำงานใช้ข้อมูล 8 บิตที่ป้อนผ่นซอฟต์แวร์จากคอมพิวเตอร์ โดยข้อมูลนี้จะส่งผ่านมาทางตัวไมโครคอนโทรลเลอร์มายังบอร์ดควบคุมเพื่อสั่งให้ DC MOTOR ทำงาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 6.5 แสดงวงจรนับ (Counter)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



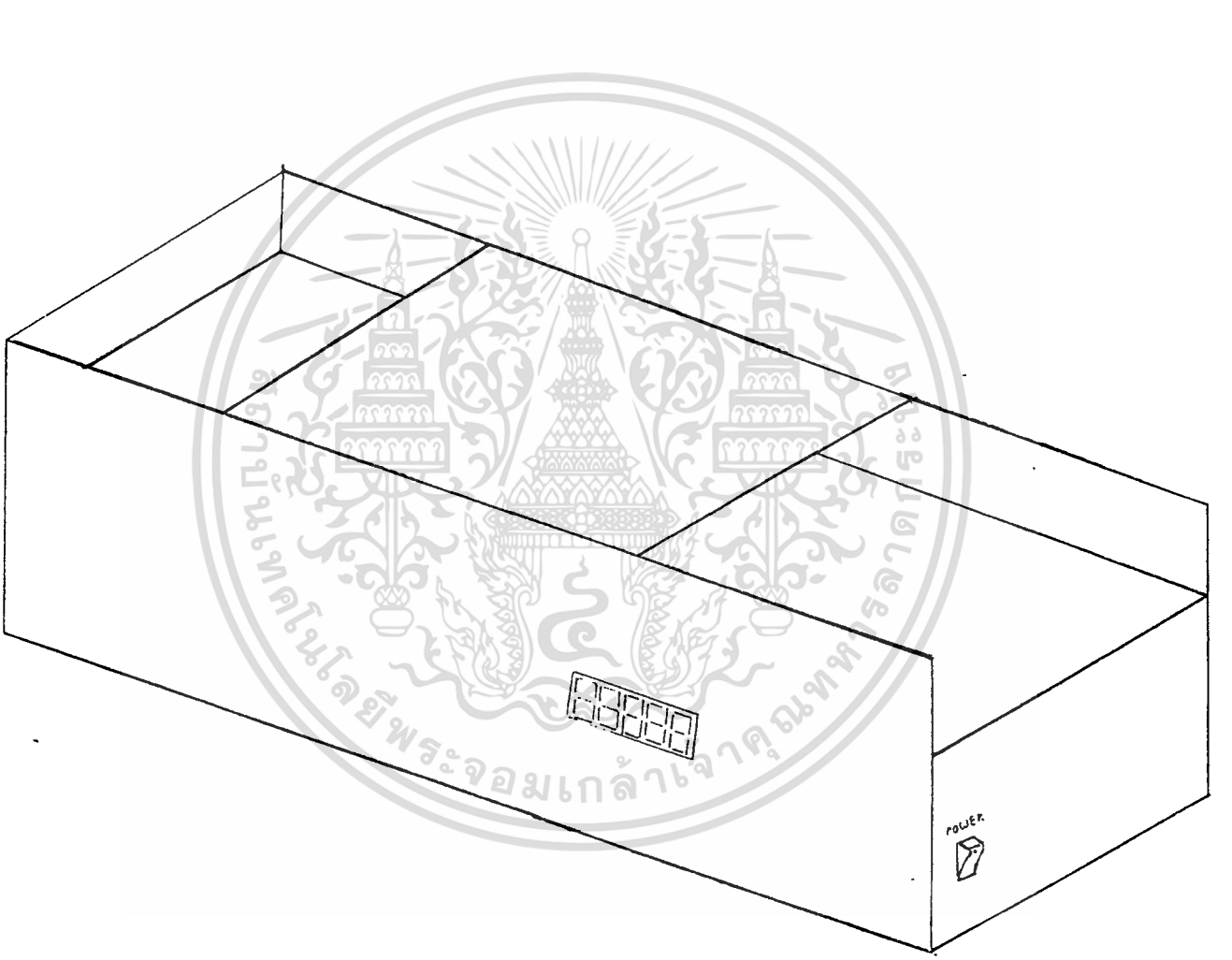
รูปที่ 6.6 วงจรควบคุม DC MOTOR

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

6.2 โครงสร้างของเครื่องตรวจกระดาษคำตอบ

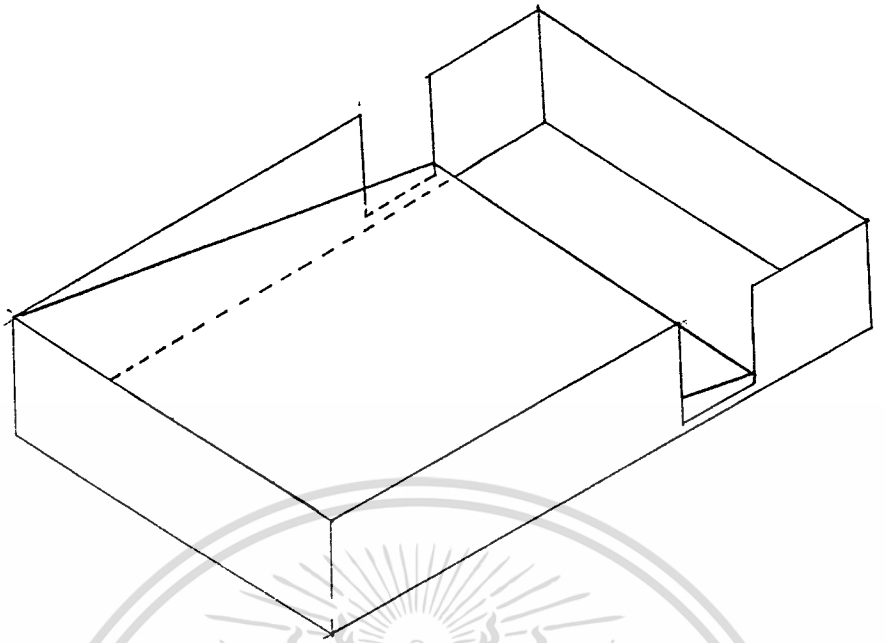
วัสดุที่ใช้ประกอบขึ้นเป็นเครื่องตรวจกระดาษคำตอบนี้จะมทำจากแผ่นพลาสติก ซึ่งเป็นวัสดุที่หาได้ง่าย ราคาค่อนข้างถูก การตัดแต่งขึ้นรูปทำได้ง่าย เหตุผลที่ใช้แผ่นพลาสติกเป็นวัสดุในการทำโครงสร้างเนื่องจากโครงงานเป็นการทำขึ้นมาเพื่อเป็นต้นแบบในการผลิตเพื่อใช้งานจริงจึงต้องใช้วัสดุที่หาได้ง่ายและดัดแปลงตกแต่งได้ง่ายแต่สามารถนำไปใช้งานได้จริง

ลักษณะโครงสร้างในส่วนต่างๆของเครื่องตรวจกระดาษคำตอบแสดงดังรูปต่อไปนี้

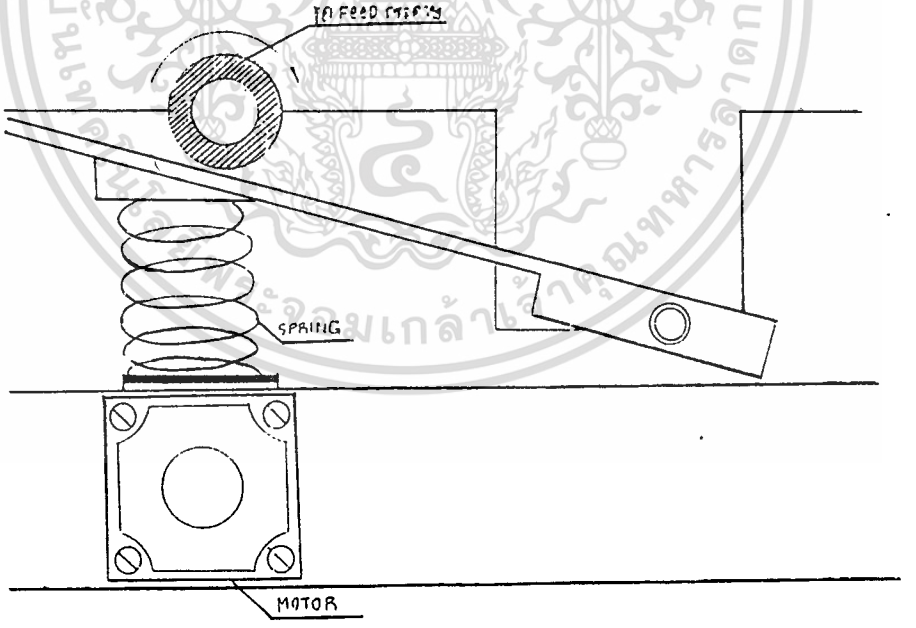


รูปที่ 6.7 ลักษณะภายนอกของเครื่องตรวจกระดาษคำตอบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

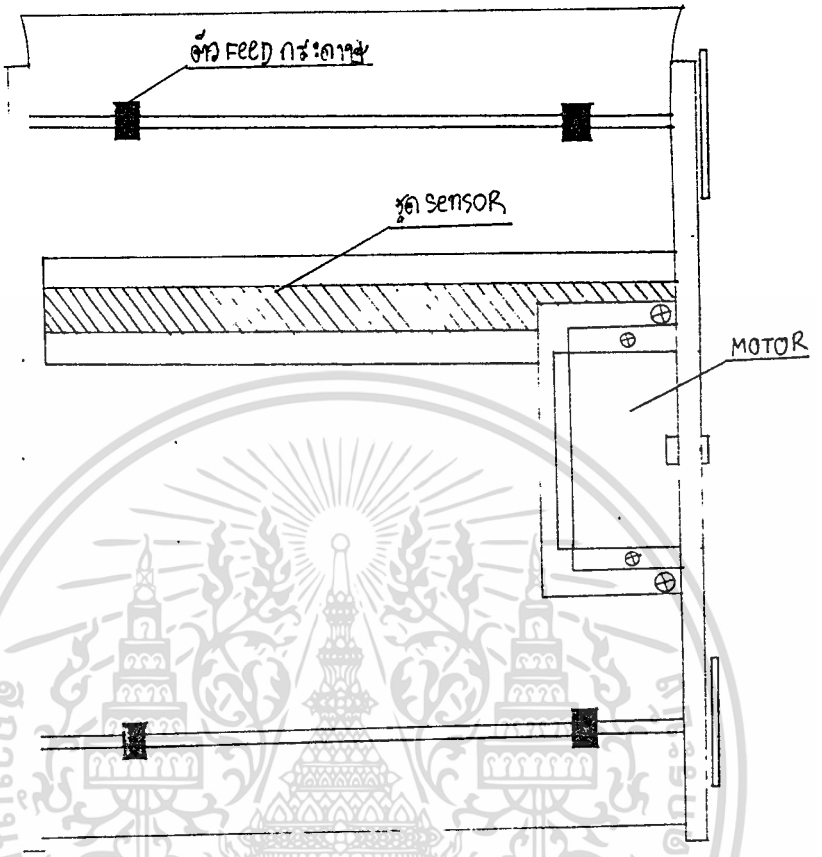


รูปที่ 6.8 แสดงที่ใส่กระดาษก่อนผ่านเข้าเครื่องตรวจ

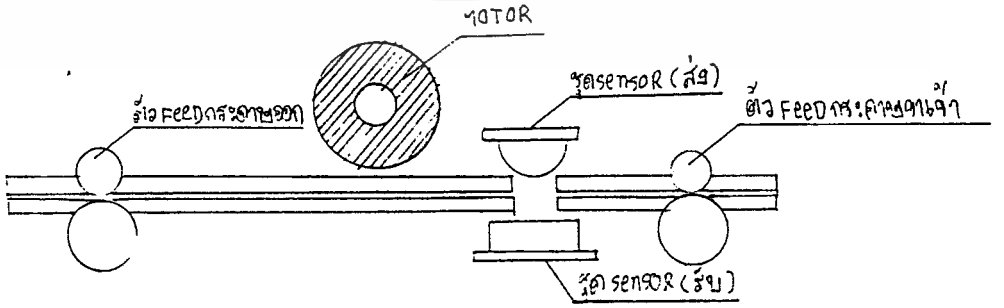


รูปที่ 6.9 แสดงการติดตั้งสปริง ลูกยาง และมอเตอร์ชุดฟีดกระดาษเข้า

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

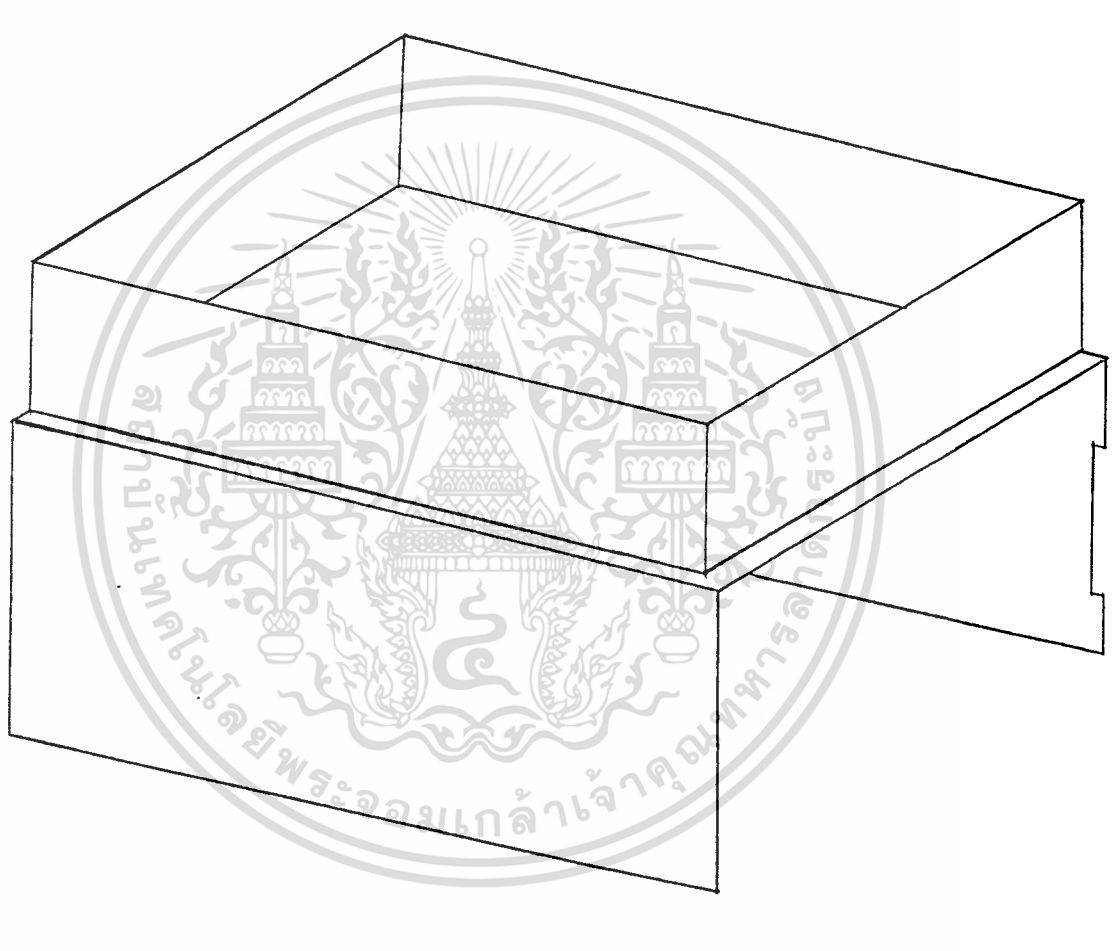


รูปที่ 6.10 แสดงการติดตั้งชุดฟีดกระดาษและตัว Sensor เมื่อมองจากด้านบน



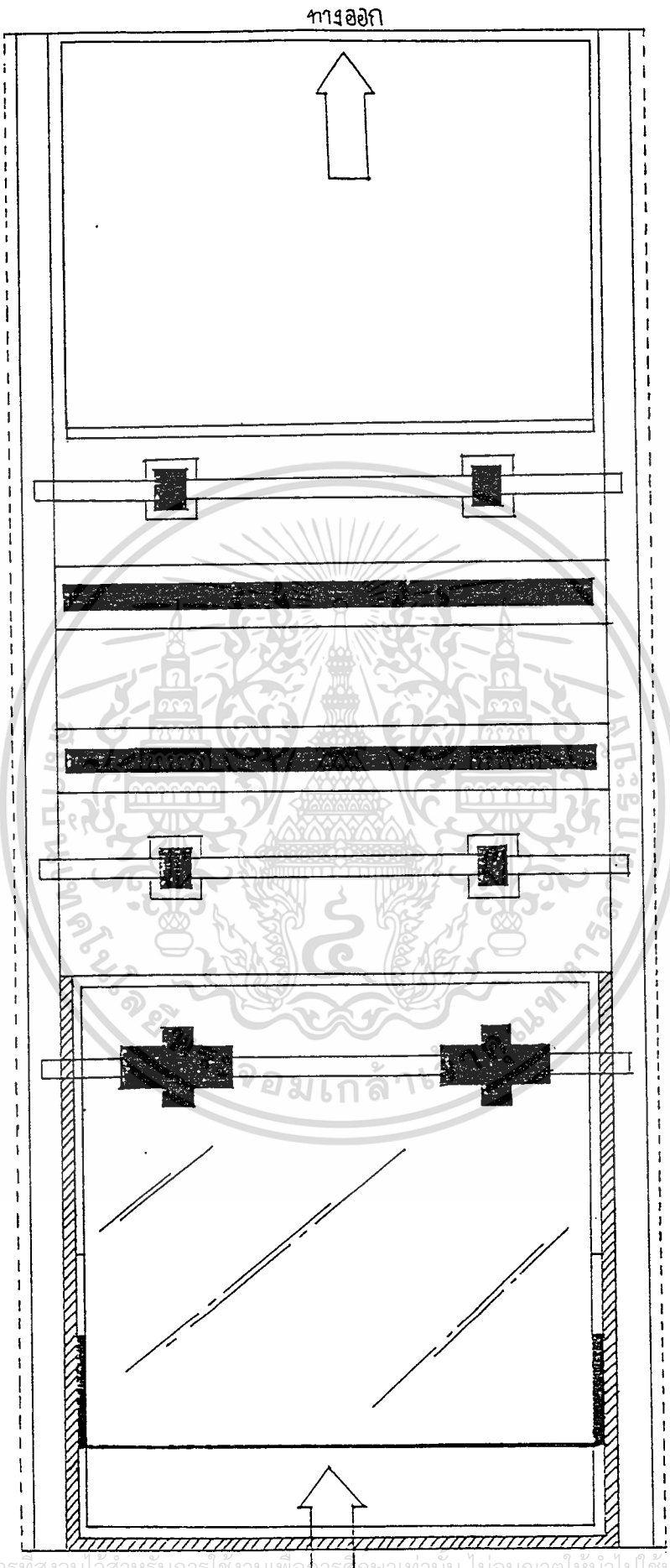
รูปที่ 6.11 แสดงการติดตั้งตัว Sensor เมื่อมองจากด้านข้าง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 6.12 แสดงที่รองรับกระดาษที่ผ่านการตรวจแล้ว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 6.13 แสดงการติดตั้งอุปกรณ์ชุดฟีดการะดามเข้าและชุดขับ
ให้กระดาษเคลื่อนที่เมื่อมองจากด้านบน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

6.3 การทำงานของเครื่องตรวจกระดาษคำตอบ

ลักษณะการทำงานของเครื่องตรวจกระดาษคำตอบจะแบ่งเป็นส่วนหลักๆ ได้ 3 ส่วน คือ

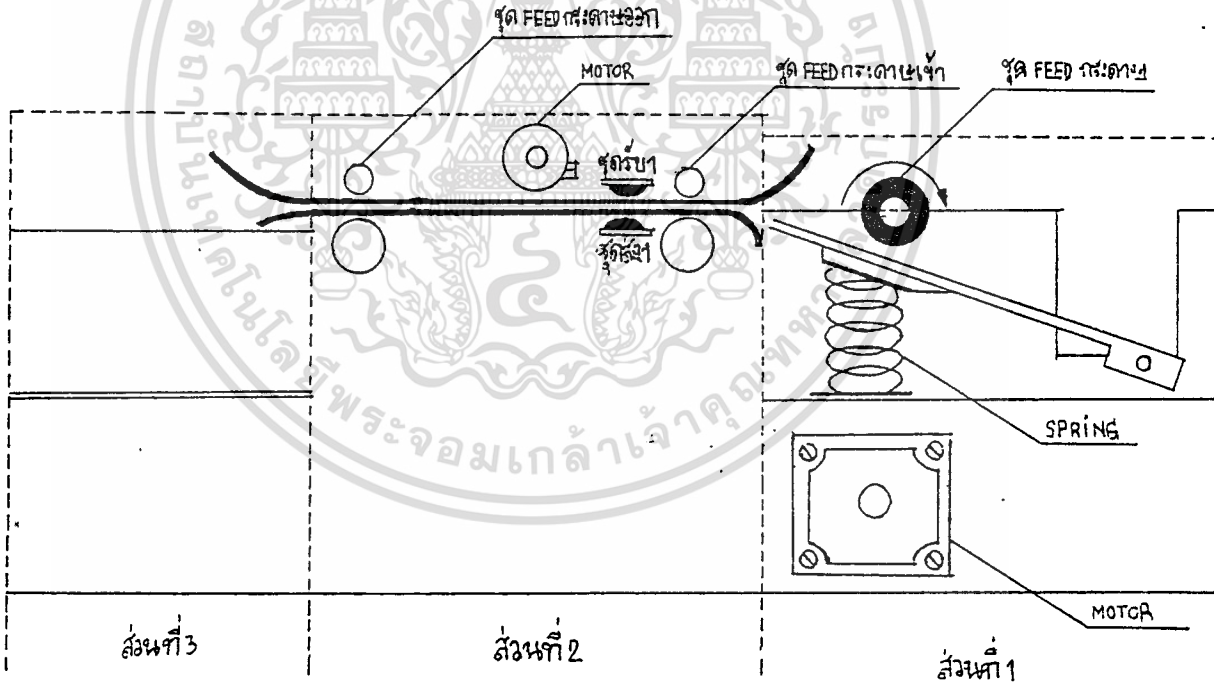
- ส่วนของชุด Feed กระดาษคำตอบ
- ส่วนของ sensor
- ส่วนของมอเตอร์ฟีดกระดาษคำตอบ

6.3.1 ส่วนของชุด Feed กระดาษคำตอบของเครื่องตรวจกระดาษคำตอบ

ชุด Feed กระดาษคำตอบ จะประกอบไปด้วยส่วนประกอบหลักๆ 3 ส่วนคือ

- ส่วนของชุดใส่กระดาษคำตอบ

โดยในส่วนนี้เราจะนำกระดาษคำตอบที่ต้องการตรวจมาใส่ ซึ่งในส่วนนี้จะมีชุด feed 1 ชุด สำหรับ feed กระดาษคำตอบเข้าสู่ชุด sensor ต่อไป โดยในการ feed จะอาศัย motor กับสายพานเป็นตัวขับกระดาษคำตอบ และในการส่งกระดาษคำตอบขึ้นไปสู่ชุด feed เราจะอาศัยแรงส่งจากสปริง ดังรูปส่วนที่ 1 ของรูปที่ 6.14



รูปที่ 6.14 แสดงภาพภายในของเครื่องตรวจกระดาษคำตอบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- ส่วนของชุด Feed กระดาษคำตอบ

เมื่อกระดาษคำตอบถูกส่งมาจากชุดใส่กระดาษคำตอบ กระดาษคำตอบจะเข้าสู่ชุด feed โดยการ feed จะอาศัย motor และสายพาน เป็นตัวขับชุด feed ทางเข้า และชุด feed ทางออกของชุด sensor โดยเมื่อผ่านการ feed ของชุด feed ทางเข้าแล้ว กระดาษคำตอบถึงจะเคลื่อนที่เข้าสู่การ sensor โดยอาศัยตัวรับ-ส่งอินฟราเรดเป็นตัว sensor โดยจะเคลื่อนที่เข้าสู่ชุด sensor และกระดาษคำตอบจะเคลื่อนที่ออกไปเข้าสู่ชุด feed ทางออก ชุด feed ทางออกจะทำหน้าที่ขับกระดาษคำตอบให้เคลื่อนที่ออกไปยังส่วนรับกระดาษคำตอบที่ผ่านการตรวจเสร็จแล้ว ดังรูปส่วนที่ 2 ของรูปที่ 6.14

- ส่วนชุดรับกระดาษคำตอบ

ชุดรับกระดาษคำตอบจะรับกระดาษคำตอบที่ผ่านการตรวจแล้วลงสู่ชุดรับกระดาษคำตอบ ดังรูปส่วนที่ 3 ของรูปที่ 6.14

6.3.2 ส่วนของชุด Sensor

เมื่อกระดาษผ่านเข้าชุด Sensor ตัวตรวจจับแสงอินฟราเรดจะทำการเชคค่าความเข้มของกระดาษในตำแหน่งที่มีการฝนคำตอบ จากนั้นจะส่งค่าที่อ่านได้ไปยังไมโครคอนโทรลเลอร์(MCS-51) ตัวไมโครคอนโทรลเลอร์จะเก็บค่าคำตอบไว้และส่งผ่านต่อไปยังเครื่องคอมพิวเตอรเพื่อทำการประมวลผลต่อไป

6.3.3 ส่วนควบคุม DC Motor ของชุดฟีดกระดาษคำตอบ

DC Motor ที่ใช้กับชุดฟีดจะมีด้วยกัน 2 ตัว โดยมอเตอร์ตัวที่ 1 ใช้เป็นตัวขับชุดฟีดกระดาษผ่านตัว Sensor ส่วนมอเตอร์ตัวที่ 2 ใช้สำหรับขับชุดฟีดกระดาษจากที่วางเข้าไปยังชุด Sensor การควบคุมมอเตอรใช้วงจรในรูปที่ ซึ่งเป็นวงจรที่สามารถควบคุมมอเตอร์ได้ถึง 2 ตัวในเวลาเดียวกัน การควบคุมด้วยวงจรนี้เป็นการควบคุมโดยใช้คอมพิวเตอร์สั่งงานให้มอเตอร์หมุนหรือหยุดหมุนได้ตามต้องการ

มอเตอร์ตัวที่ 1 จะถูกควบคุมให้หมุนอยู่ตลอดเวลา แต่สำหรับมอเตอร์ตัวที่ 2 นั้นจะถูกควบคุมให้ทำงานเป็นจังหวะตามโปรแกรมที่เขียนขึ้น โดยการหมุนแต่ละครั้งชุดฟีดจะทำการฟีดกระดาษเข้า 1 แผ่นแล้วหยุดชั่วขณะแล้วจึงทำงานฟีดกระดาษแผ่นต่อไปเข้าไปใหม่เช่นนี้ไปเรื่อยจนกว่ากระดาษจะหมด

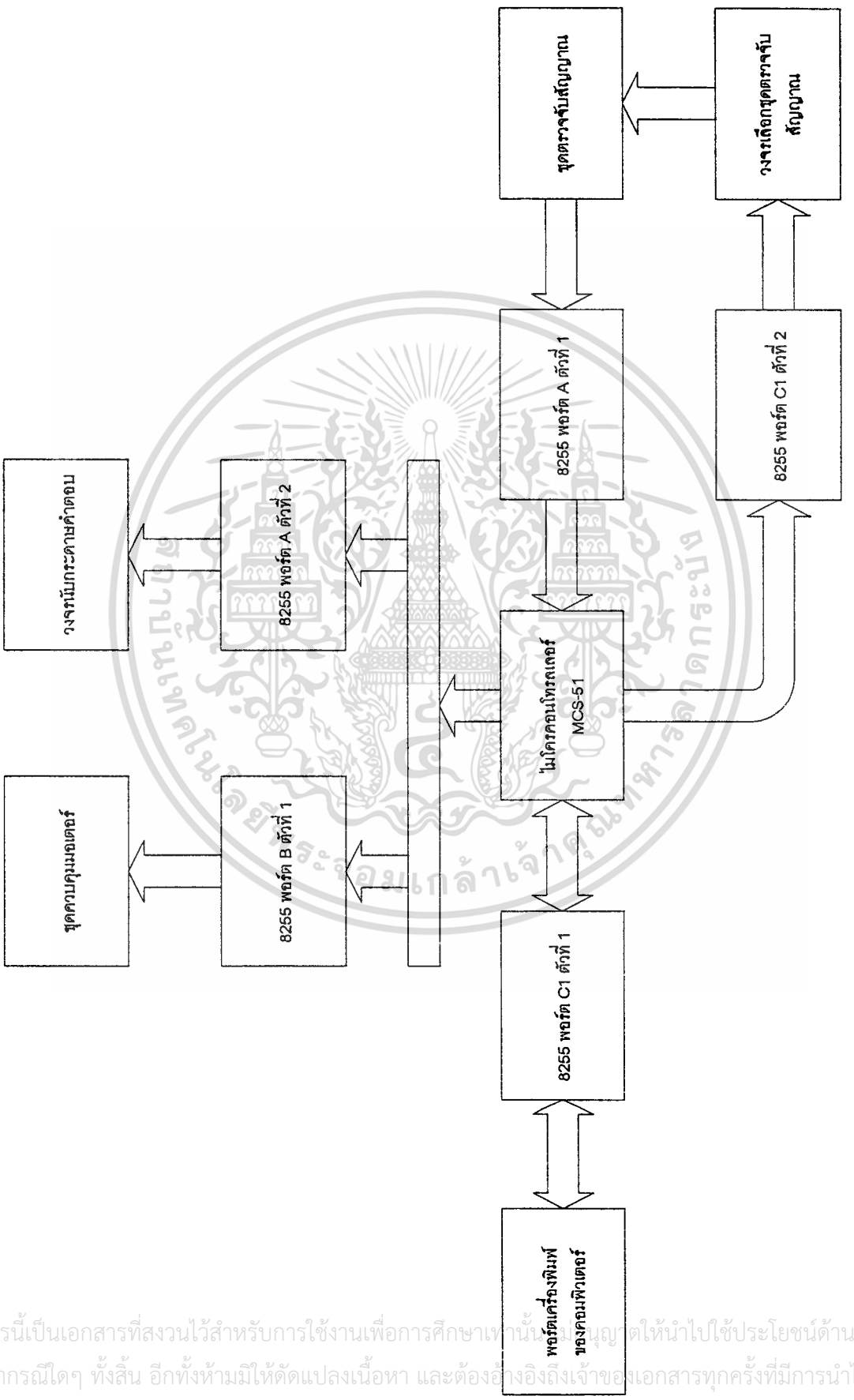
6.4 การทำงานของเครื่องตรวจข้อสอบแสดงเป็นบล็อกไดอะแกรม

จากรูปแสดงบล็อกไดอะแกรมการทำงานของเครื่องตรวจข้อสอบโดยที่จะมี วงจรควบคุมที่เป็นศูนย์กลางของระบบทั้งหมด โดยใช้ ไมโครคอนโทรลเลอร์เบอร์ MCS-51 ซึ่งจะมีการติดต่อกับอุปกรณ์รอบข้างและติดต่อกับเครื่องคอมพิวเตอร์โดยผ่านทาง พอร์ตขนานของเครื่องพิมพ์ เพื่อให้มีความสามารถให้การอ่านข้อมูลและเขียนข้อมูลได้เร็วขึ้น การอ่านข้อมูลจากการตรวจจับจะ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

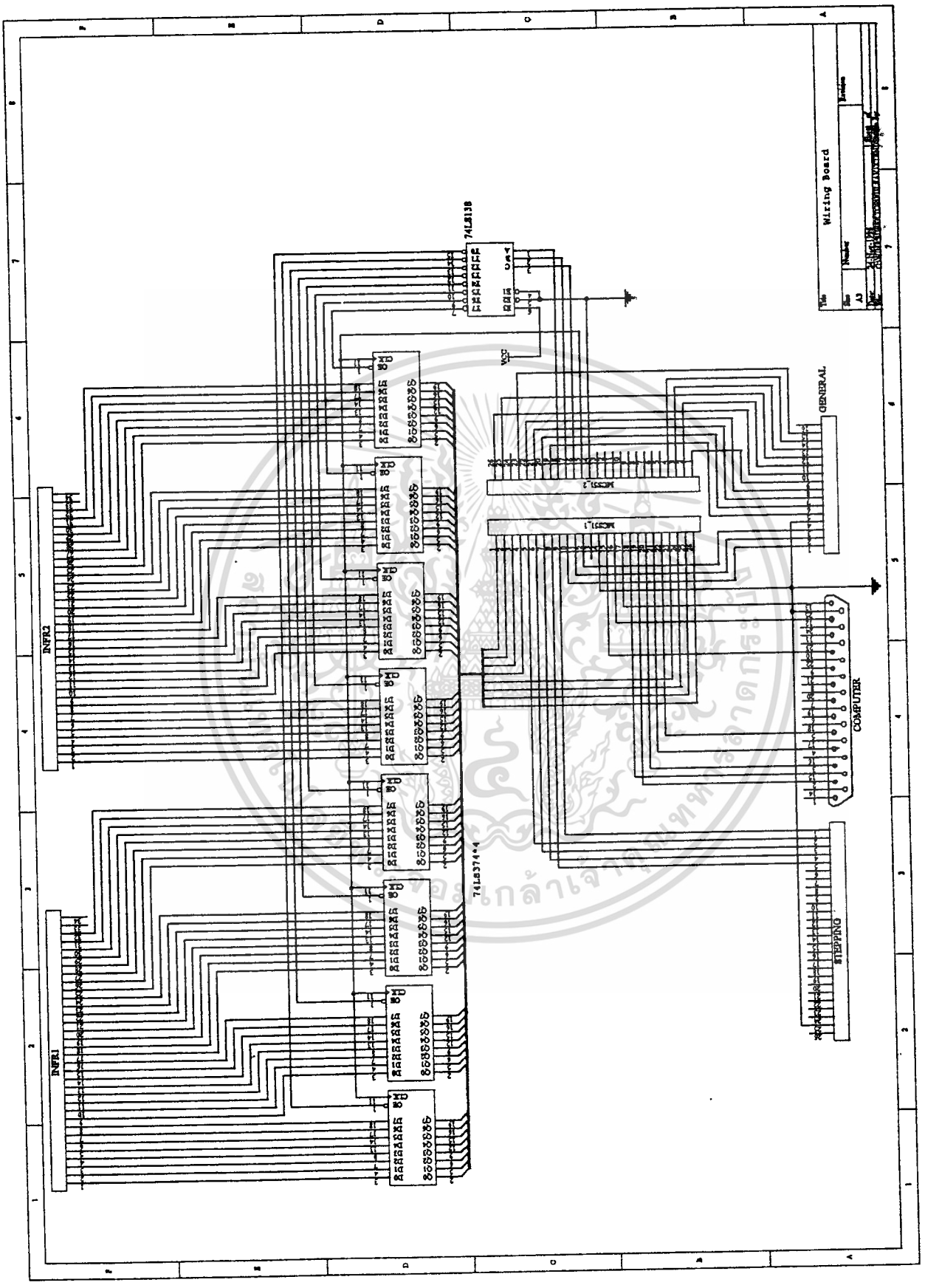
ต้องควบคุมวงจรเลือกชุดตรวจจับจากรูป เนื่องจาก สัญญาณที่ไมโครคอนโทรลเลอร์สามารถที่จะอ่านได้ มีเพียง 8 บิตแต่ข้อมูลที่ได้จากการตรวจจับ มีทั้งสิ้น 28 บิต จึงต้องใช้วงจรเลือกชุดตรวจจับในการอ่านข้อมูลในแต่ละครั้ง และอีกส่วนที่ตัว ไมโครคอนโทรลเลอร์จะต้องจัดการคือการส่งสัญญาณเพื่อควบคุมมอเตอร์ในการนำกระดาษเข้าเครื่องตรวจข้อสอบ และการนับจำนวนกระดาษคำตอบที่ทำการตรวจโดยส่งข้อมูลให้กับชุดวงจรนับกระดาษคำตอบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 6.15 บล็อกไดอะแกรมแสดงการทำงานของเครื่องตรวจข้อสอบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 6.16 แสดงวงจรศูนย์กลางการเชื่อมต่อระหว่างอุปกรณ์ต่างๆ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 7

รายละเอียดการทำงานของโปรแกรมเครื่องตรวจข้อสอบ

โปรแกรมเครื่องตรวจข้อสอบแบ่งออกเป็น 2 ส่วนด้วยคือ :

- โปรแกรมทำงานบนเครื่องคอมพิวเตอร์ ใช้ภาษาวิชวลเบสิก
- โปรแกรมทำงานบนเครื่องตรวจข้อสอบใช้ ภาษาแอสแซมบลีของหน่วยประมวลผลเอ็มซีเอส 51

7.1 โปรแกรมบนเครื่องคอมพิวเตอร์

7.1.1 หลักการเบื้องต้น

การพัฒนาโปรแกรมบนภาษา วิชวลเบสิกช่วยให้การพัฒนาโปรแกรมสามารถพัฒนาโปรแกรมได้อย่างรวดเร็ว สำหรับโปรแกรมเครื่องตรวจข้อสอบที่ได้พัฒนาขึ้นในส่วนที่อยู่บนเครื่องคอมพิวเตอร์มีหน้าที่ดังต่อไปนี้

- ทำหน้าที่สื่อสารระหว่างผู้ใช้โปรแกรมกับเครื่องตรวจข้อสอบ
- ทำหน้าที่สื่อสารระหว่างเครื่องคอมพิวเตอร์กับเครื่องตรวจข้อสอบ
- ทำหน้าที่สั่งให้เครื่องตรวจข้อสอบทำงานตามต้องการ
- ทำการประมวลผลคะแนนสอบที่ได้จากเครื่องตรวจข้อสอบ
- ทำการตรวจข้อสอบซ้ำเพื่อความมั่นใจ ได้ทั้งหมด 3 ครั้งด้วยกัน
- แสดงรายละเอียดคะแนนเป็นกราฟ และผลสรุปของคะแนนที่ทำการตรวจ
- สั่งพิมพ์รายละเอียดคะแนนที่ได้ออกทางเครื่องพิมพ์ พร้อมทั้งผลลัพธ์อื่นๆ
- สามารถนำข้อมูลที่ทำการตรวจถ่ายโอนไปแต่ละระบบเพื่อทำการตัดคะแนน หรือทำรายการอื่นๆ ตามต้องการ

7.1.2 โครงสร้างพื้นฐานของซอฟท์แวร์ ประกอบด้วย :

- โปรแกรมที่ใช้งาน
- ฐานของข้อมูลของโปรแกรม เพื่อใช้ในการเก็บและบันทึกผลของคะแนนที่ทำการตรวจสอบได้

สำหรับโปรแกรมที่จะอธิบายในตอนต่อไป แต่ในส่วนของฐานข้อมูลมีรายละเอียดดังนี้

ฐานข้อมูลของระบบจะมีตารางใช้งานทั้งหมด 2 ตารางคือ

1. ตารางรายละเอียดรหัสนักศึกษา ,รายชื่อนักศึกษา และ คะแนนที่ทำได้
2. ตารางบันทึกคำตอบของข้อสอบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางรายละเอียดนักศึกษา, รายชื่อนักศึกษา และคะแนนที่ทำได้ มีโครงสร้างดังนี้

โครงสร้างของตารางข้อมูล : SCORE.DBF						
จำนวนของรายการข้อมูล : 91						
ปรับปรุงล่าสุด : 10/22/96						
ฟิลด์ที่	ชื่อฟิลด์	รายละเอียด	ชนิด	ความกว้าง	ทศนิยม	ลำดับ
1	STD_ID	รหัสนักศึกษา	อักษร	15	0	↑
2	STD_NAME	ชื่อนักศึกษา	อักษร	50	0	
3	SCORE1	ช่องเก็บคะแนน 1	ตัวเลข	3	0	
4	SCORE2	ช่องเก็บคะแนน 2	ตัวเลข	3	0	
5	SCORE3	ช่องเก็บคะแนน 3	ตัวเลข	3	0	
** รวม **				75		

ตารางที่ 7.1 ตารางรายละเอียดนักศึกษา, รายชื่อนักศึกษา, ...

ตารางบันทึกคำตอบของข้อสอบ มีโครงสร้างดังนี้

โครงสร้างของตารางข้อมูล : ANSWER.DBF						
จำนวนของรายการข้อมูล . 150						
ปรับปรุงล่าสุด : 10/22/96						
ฟิลด์ที่	ชื่อฟิลด์	รายละเอียด	ชนิด	ความกว้าง	ทศนิยม	ลำดับ
1	ITEM	ข้อย่อยของข้อสอบ (1-150)	ตัวเลข	3	0	↑
2	CHOICE	คำตอบของข้อสอบแต่ละข้อ (1-5)	ตัวเลข	1	0	
** รวม **				5		

ตารางที่ 7.2 ตารางบันทึกคำตอบของข้อสอบ

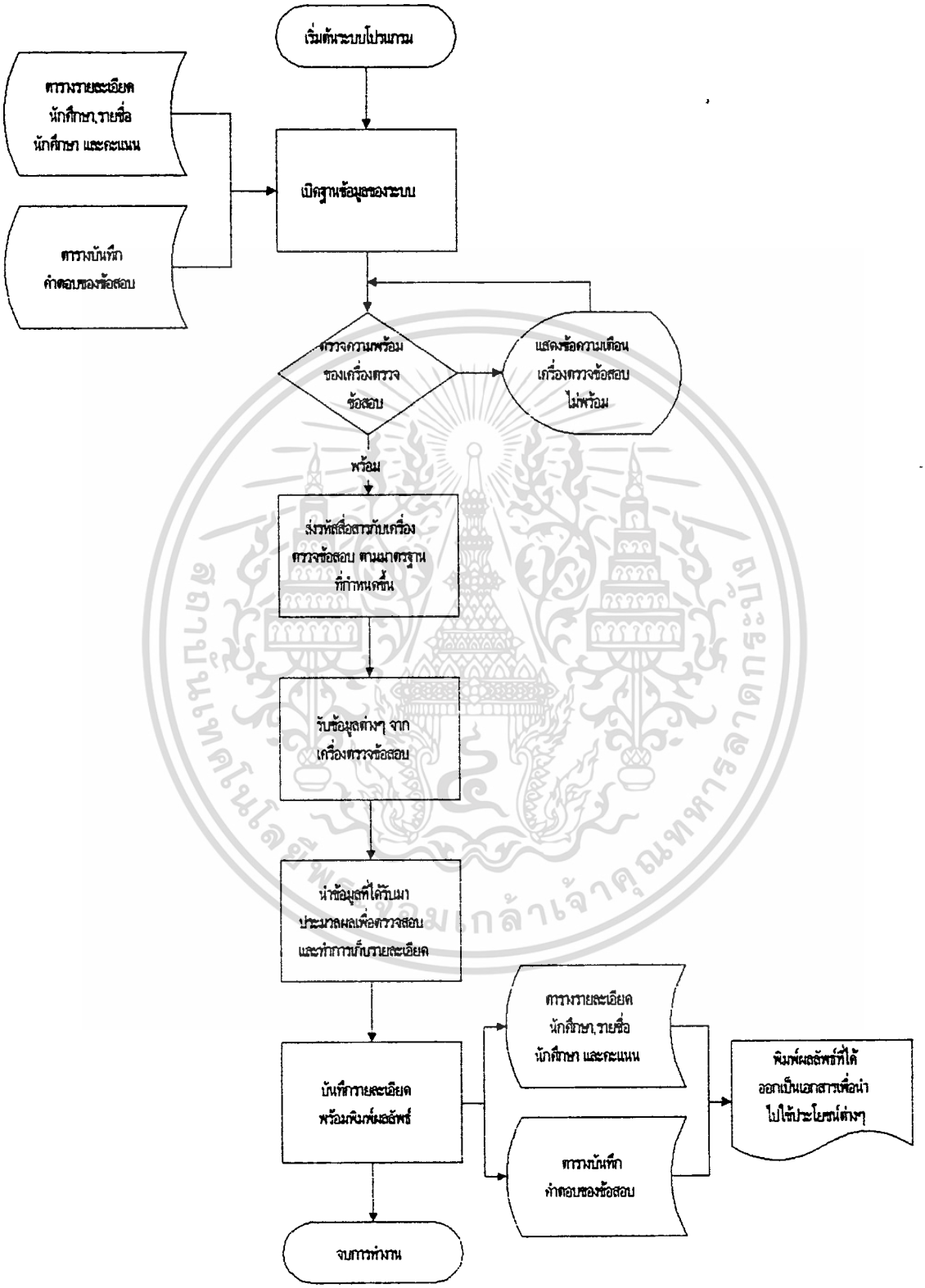
7.1.3 ผังการทำงานระบบของโปรแกรมเครื่องตรวจข้อสอบ

คือ เอกสารที่แสดงรายละเอียดการเชื่อมต่อกันของการทำงาน หรืออุปกรณ์ต่างๆ กับระบบงาน สำหรับสัญลักษณ์โดยทั่วไปจะเป็นสัญลักษณ์ที่ใช้ในการเขียน ผังงาน ตัวอย่างที่เชื่อมต่อกับระบบงานมีดังต่อไปนี้ ฮาร์ดดิสก์, ตารางข้อมูล, อุปกรณ์สื่อสารภายนอก เป็นต้น

ผังการทำงานของระบบ แบ่งส่วนประกอบได้เป็น 3 ส่วนด้วยกันคือ ทางซ้ายสำหรับ สัญลักษณ์นำเข้าของข้อมูล ทางขวาสำหรับ สัญลักษณ์ทางออก และสำหรับตรงกลางสำหรับสัญลักษณ์การประมวลผล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รายละเอียดผังการทำงานระบบของเครื่องตรวจข้อสอบมีดังต่อไปนี้



รูปที่ 7.1 แสดงผังการทำงานระบบเครื่องตรวจข้อสอบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

7.1.4 รายละเอียดการทำงานของโปรแกรมเครื่องตรวจข้อสอบบนคอมพิวเตอร์

หลักการทำงาน - รายละเอียดหลักถูกแบ่งออกเป็น 3 ส่วนด้วยกันคือ

1. ส่วนติดต่อกับผู้ใช้
2. ส่วนติดต่อกับเครื่องตรวจข้อสอบ
3. ส่วนประมวลผล (การถอนรหัสข้อมูล)

ส่วนติดต่อกับผู้ใช้

ประกอบด้วยเมนูและฟอร์มที่ใช้ในการติดต่อดังมีรายละเอียดดังนี้

เมนู

File

- Open - ใช้เปิดฐานข้อมูลเพื่อบันทึกคะแนน
- Close - ปิดฐานข้อมูลที่เปิด
- Save - บันทึกผลของคะแนนลงฐานข้อมูล
- Print - พิมพ์รายละเอียดออกทางเครื่องพิมพ์
- Exit - จบการทำงาน

View

- Show all data - แสดงข้อมูลนักศึกษารวมทั้งคะแนน
- View Graph Information - นำข้อมูลนักศึกษาแสดงเป็นการตรวจสอบความผิดพลาด
- Show Summary and Statistic - แสดงผลสรุปที่ได้จากการตรวจคำตอบ

Run

- Start - เริ่มทำการตรวจคำตอบ
- Stop - หยุดตรวจคำตอบ
- Respond self-test - ทดสอบอุปกรณ์รอบเครื่องตรวจข้อสอบ

Options

- Setup Program - ติดตั้งค่าเริ่มต้นให้โปรแกรม
- Setup Answer - ติดตั้งคำตอบที่ใช้งาน

Help

- Help on checker auto-machine - ข้อมูลช่วยเหลือเบื้องต้น
- About checker auto-machine - เกี่ยวกับโปรแกรม

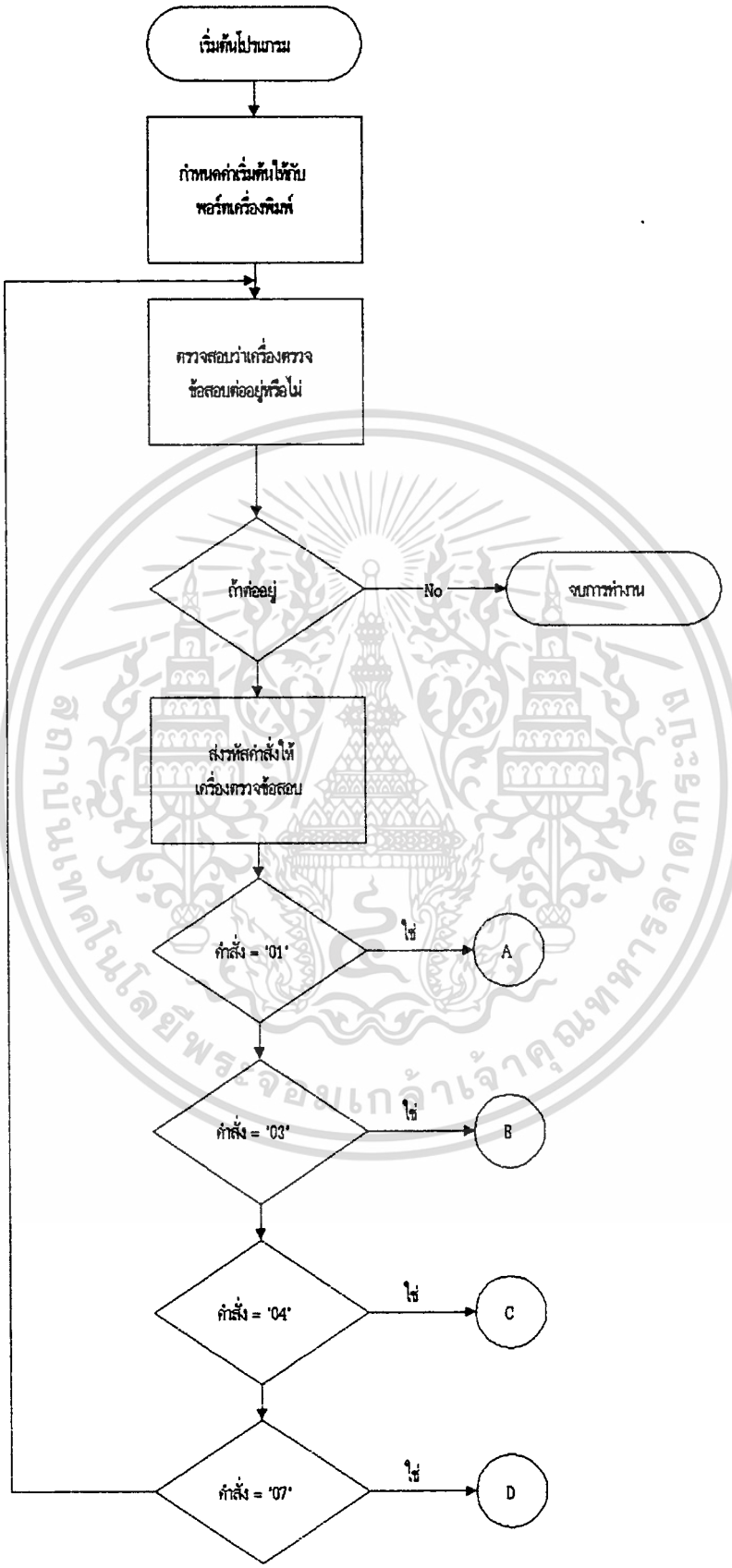
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

7.1.5 ส่วนติดต่อกับเครื่องตรวจข้อสอบ

การทำงานของโปรแกรมเครื่องตรวจข้อสอบบน เครื่องคอมพิวเตอร์โดยส่วนใหญ่จะเป็นการเขียนโปรแกรมให้มีความสามารถในการใช้งานได้ง่ายขึ้น สำหรับส่วนที่สำคัญที่จะขออธิบายคือส่วนที่จะทำการติดต่อกับเครื่องตรวจข้อสอบ ซึ่งเป็นส่วนที่จะทำให้การใช้งาน โปรแกรมเครื่องตรวจข้อสอบสามารถทำงานได้อย่างมีประสิทธิภาพ รายละเอียดของขั้นตอนการติดต่อกับเครื่องตรวจข้อสอบมีดังนี้

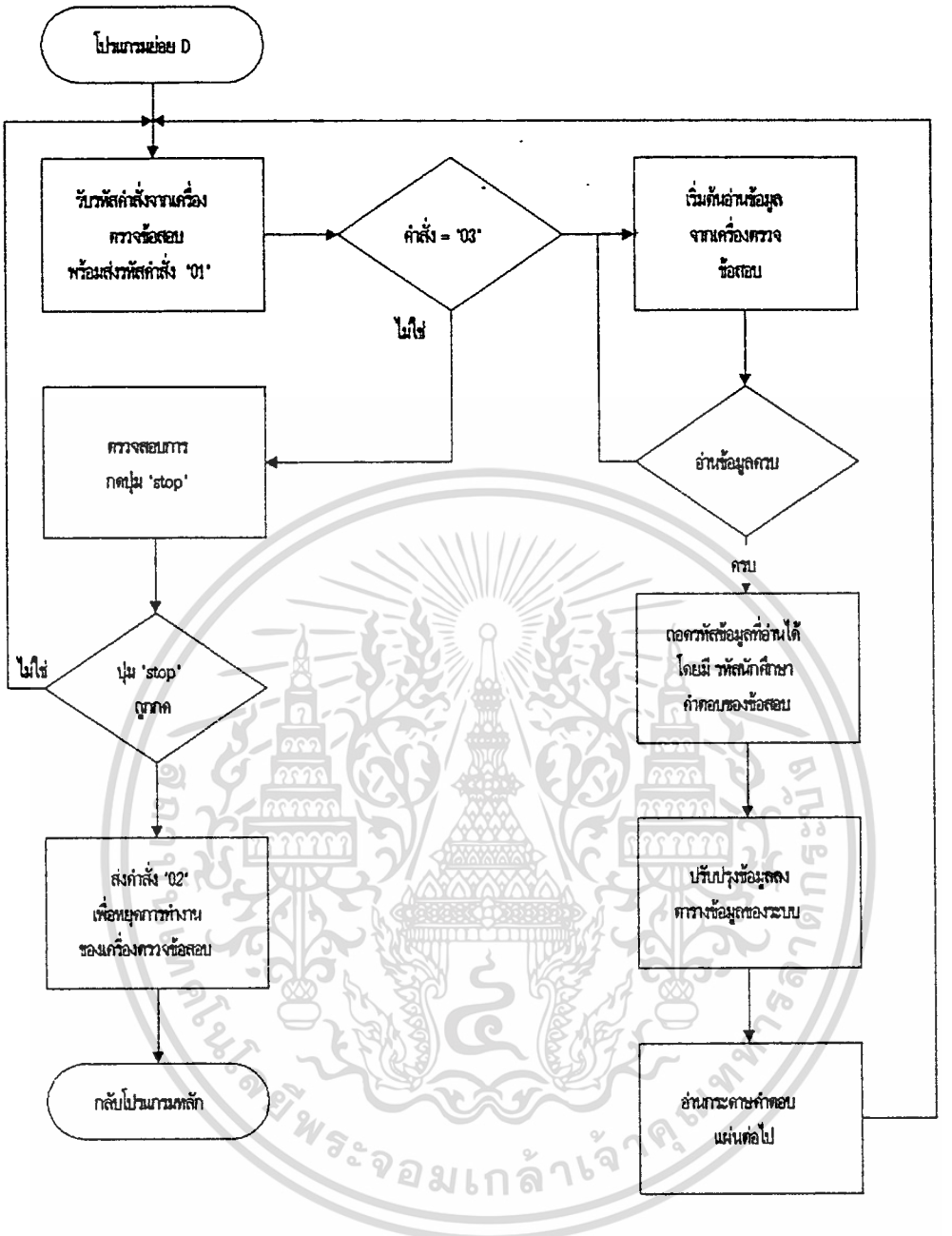
1. กำหนดพอร์ตเครื่องพิมพ์ให้สามารถใช้งานในการติดต่อกับเครื่องตรวจข้อสอบได้ โดยรายละเอียดของการกำหนดอยู่ในเนื้อหา การเขียนโปรแกรมติดต่อสื่อสารระหว่างคอมพิวเตอร์ผ่านทางพอร์ตเครื่องพิมพ์
2. ตรวจสอบว่ามีเครื่องตรวจข้อสอบติดอยู่หรือไม่ ถ้ามีให้ส่งรหัสที่จะใช้งานเครื่องตรวจข้อสอบ
3. รหัสที่ส่งมีดังนี้ (รูปที่ 7.2 ผังการทำงานของการส่งรหัสคำสั่ง)
    - รหัสคำสั่งทดสอบมอเตอร์ โปรแกรมย่อย A
    - รหัสคำสั่งทดสอบการทำงานของชุดตรวจจับ โปรแกรมย่อย B
    - รหัสคำสั่งทดสอบการทำงานของชุดนับกระดาษคำตอบ โปรแกรมย่อย C
    - รหัสคำสั่งให้เครื่องตรวจข้อสอบอ่านคำตอบจากชุดตรวจจับ โปรแกรมย่อย D
4. ถ้ามีการสั่งให้เครื่องตรวจข้อสอบอ่านคำตอบจะมีหลักการทำงานดังนี้ (รูปที่ 7.3 ผังการทำงานของการรับข้อมูลของการตรวจจับ)
    - โปรแกรมจะส่งข้อมูลสื่อสารกับเครื่องตรวจข้อสอบตลอดเวลา
    - ถ้าโปรแกรมถูกสั่งให้หยุดการทำงานจะส่งรหัสคำสั่ง “02” ให้กับเครื่องตรวจข้อสอบเพื่อบอกว่าขณะนี้เป็นการหยุดการทำงาน
    - การอ่านคำตอบโปรแกรมจะรู้ว่าเครื่องตรวจข้อสอบส่งรายละเอียดของคำตอบมาให้ก็ต่อเมื่อได้รับรหัสคำสั่ง “03” ซึ่งเป็นรหัสแสดงว่าเครื่องตรวจข้อสอบจะเริ่มทำการส่งข้อมูลที่ได้จากการอ่านจากชุดตรวจจับสัญญาณมาให้
    - เมื่ออ่านข้อมูลครบโปรแกรมตรวจข้อสอบจึงจะสั่งให้เครื่องตรวจข้อสอบอ่านกระดาษคำตอบแผ่นต่อไป โดยส่งรหัสเริ่มต้นการอ่านอีกครั้งนั้นคือรหัสคำสั่ง “1”

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 7.2 ผังการทำงานของการส่งรหัสคำสั่ง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 7.3 ผังการทำงานของการรับข้อมูลที่ได้จากการตรวจจับ

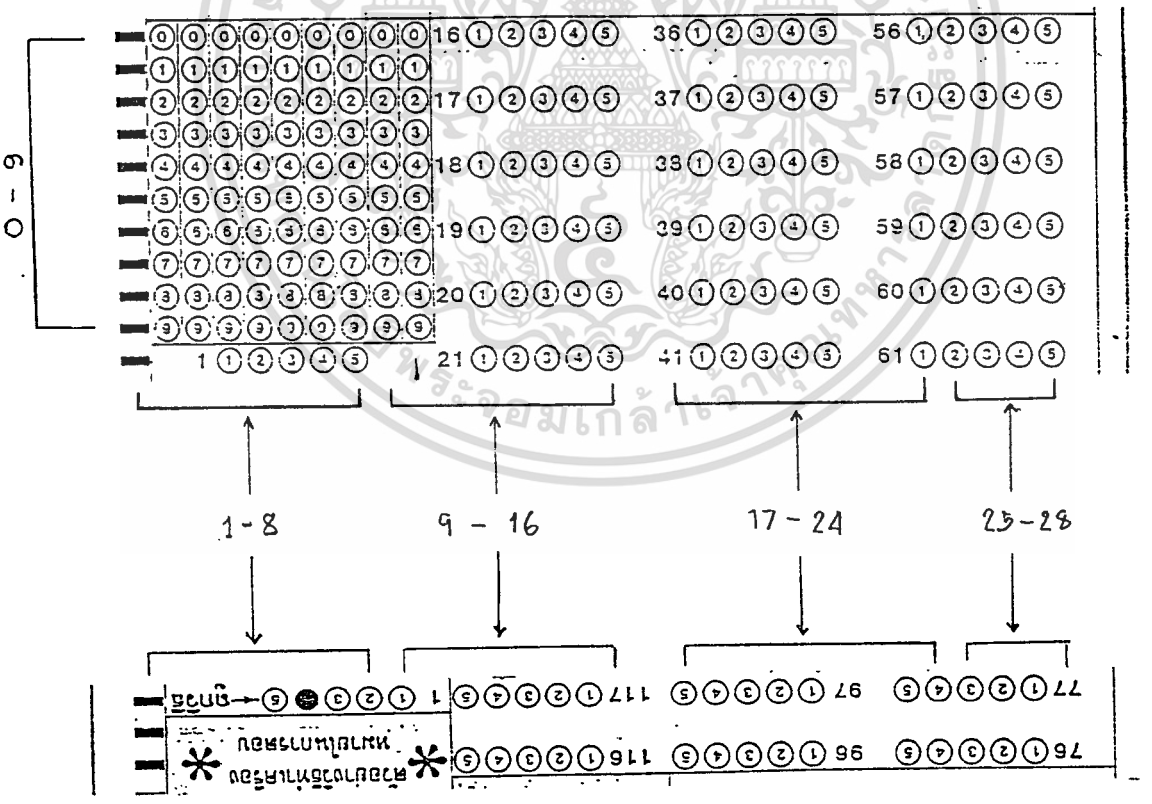
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

7.1.6 ส่วนประมวลผล (หลักการถอดรหัสจากข้อมูลที่อ่านได้)

การทำงานของเครื่องตรวจข้อสอบ เมื่อได้มีการตรวจจับ ข้อมูลบนแผ่นคำตอบ จะมีการส่งข้อมูลที่ทำการตรวจจับได้ มาที่โปรแกรมบนเครื่องคอมพิวเตอร์ ขั้นตอนต่อมาเครื่องคอมพิวเตอร์จะทำการถอดรหัสข้อมูลที่ได้รับ เพื่อให้ข้อมูลที่ได้รับสามารถที่จะนำไปใช้งานได้ สำหรับรูปแบบของข้อมูลจะมีอยู่ทั้งหมด 3 ส่วนด้วยกันคือ

- ตำแหน่งของรหัสนักศึกษา
- ตำแหน่งของคำตอบด้านหน้า
- ตำแหน่งของคำตอบด้านหลัง

จากรูปที่ 7.4 จะเป็นลักษณะการแสดงตำแหน่งต่างๆ ของข้อมูลที่จะทำการถอดรหัส บนกระดาษคำตอบ ตำแหน่ง 1 จะเป็นส่วนของตำแหน่งรหัสนักศึกษา สำหรับอีก 8 ตำแหน่งที่เหลือ จะเป็นตำแหน่งของคำตอบ บนกระดาษคำตอบ โดยแบ่งออกเป็นส่วนๆ ด้วยกันดังนี้คือ 4 กลุ่มคอลัมน์กระดาษคำตอบด้านหน้า และอีก 4 กลุ่มคอลัมน์กระดาษคำตอบด้านหลัง พิจารณาหลักการถอดรหัสแต่ละส่วนดังนี้



รูปที่ 7.4 แสดงลักษณะตำแหน่งต่างๆ ของข้อมูลบนกระดาษคำตอบที่ทำการถอดรหัส

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

7.1.7 การถอดรหัสนักศึกษา

โดยปกติข้อมูลของคำตอบที่อ่านได้จากชุดตรวจจับสัญญาณจะถูกเก็บไว้ในตัวแปร BUFFER(x,4) สำหรับ x คือตำแหน่งของกลุ่มคอลัมน์ที่ถูกจับแบ่งไว้เป็น 4 ส่วน ในส่วนของรหัสนักศึกษา จะใช่กลุ่มของ คอลัมน์ที่ 1 และ 2 โดยดูได้จากรูปที่ 7.4 (ก) เนื่องจากการแบ่งรหัสนักศึกษาในแต่ละหลักถูกจัดให้อยู่ในแต่ละคอลัมน์ย่อยของ ซึ่งรวมอยู่ใน 10 แถวของตำแหน่งที่อยู่ของรหัสนักศึกษา การถอดรหัสจึงใช้วิธีการถอดรหัสทีละคอลัมน์ย่อย เพื่อให้สามารถตรวจสอบความถูกต้อง ลักษณะการทำงานของโปรแกรมจะมีดังนี้ (รูปที่ 7.5)

- อ่านข้อมูลทีละคอลัมน์ย่อย โดยในแต่ละคอลัมน์ย่อยจะอ่านข้อมูลทั้งหมด 10 แถว
- ตรวจสอบว่าข้อมูลที่อ่านได้เป็น 1 หรือไม่ ถ้าใช่ให้เก็บแถวที่ อ่านได้
- อ่านแถวต่อไปพร้อมกับบวกค่าของการอ่านในแต่ละแถว
- ถ้าค่าของการอ่านมากกว่า 1 แสดงว่ามีการระบายรหัสนักศึกษาผิดพลาด ให้คืนรหัส "ERR" เพื่อบอกว่ารหัสนักศึกษาที่อ่านพบเป็นข้อมูลที่ผิดพลาด
- แต่ถ้าไม่มีข้อผิดพลาด ให้อ่านคอลัมน์ย่อยถัดไปจนครบ 9 คอลัมน์ย่อย

7.1.8 การถอดรหัสคำตอบด้านหน้า

สำหรับคำตอบด้านหน้าจะเป็นคำตอบที่มีการเรียงข้อมูลของรหัสจากน้อยไปหามาก ฉนั้นวิธีการถอดรหัสจึงสามารถที่จะกระทำได้ง่าย โดยการระบุข้อที่ต้องการรหัส ก็จะทำให้รู้ว่าข้อมูลของข้อที่ต้องการอยู่ที่แถวหรือคอลัมน์ใด วิธีการต่อมาคือเมื่อทำการถอดรหัสได้ จะต้องตรวจสอบว่าข้อมูลที่ทำการถอดรหัสผิดหรือไม่ สำหรับขั้นตอนการถอดรหัสจะมีดังนี้ (รูปที่ 7.6)

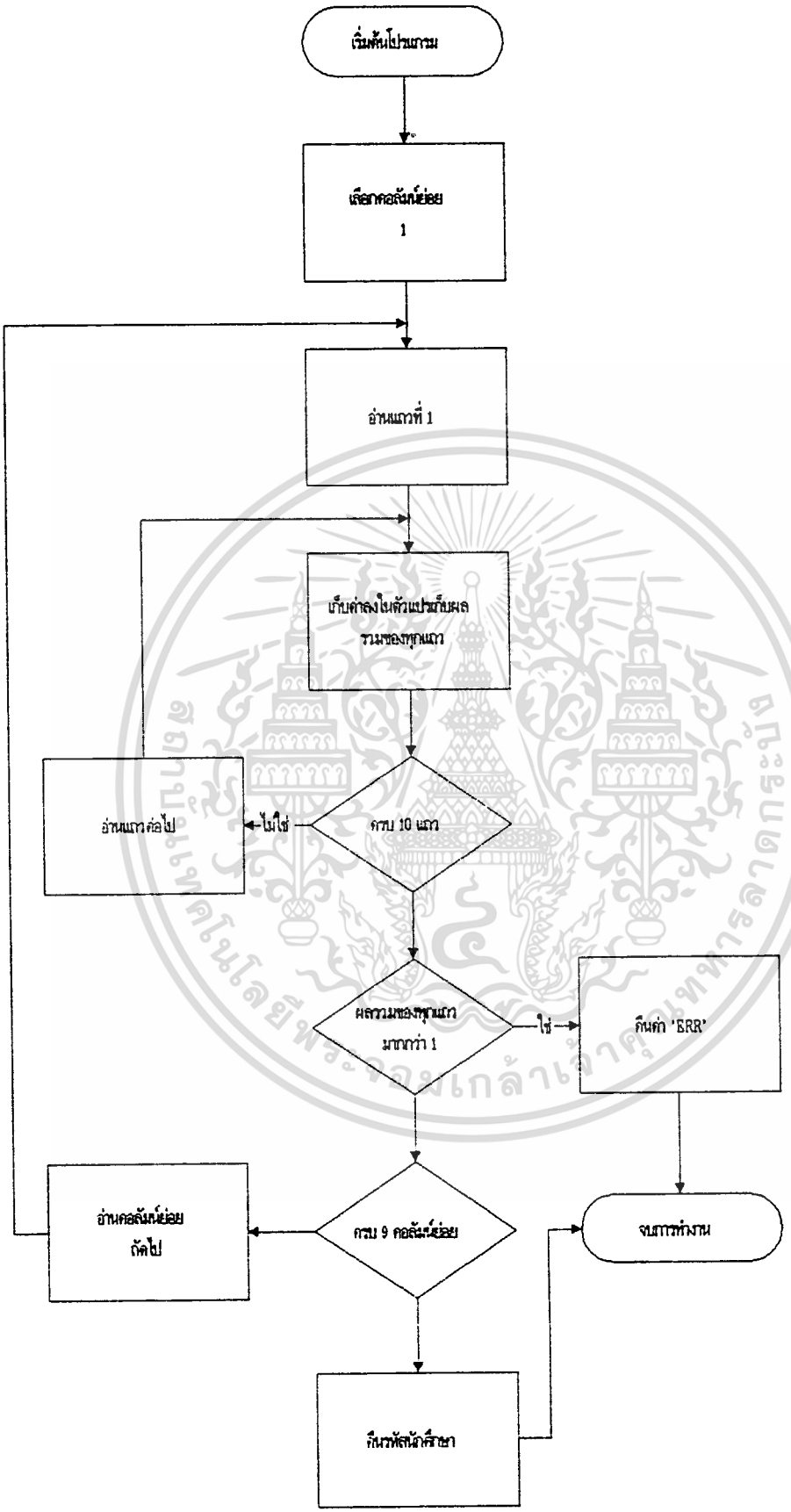
- ระบุแถวและคอลัมน์ที่ต้องการถอดรหัสคำตอบ
- อ่านคำตอบจากข้อมูลที่ได้รับ ซึ่งจะเป็นรหัสของเลขฐาน 16 ฉนั้นจะต้องเปลี่ยนเลขฐาน 16 ให้เป็นเลขข้อของคำตอบก่อน ซึ่งมีทั้งหมดด้วยกัน 5 ข้อ
- ตรวจสอบว่าข้อมูลของคำตอบถูกต้องหรือไม่ ถ้าไม่ถูกต้องให้กำหนดคำตอบเป็น 0 แต่ถ้าถูกต้องให้กำหนดเป็นเลข 1-5
- ตัวอย่างของข้อมูล ถ้าอ่านได้ 1 คำตอบคือ 1

2	:	2
4	:	3
8	:	4
16	:	5

7.1.9 การถอดรหัสคำตอบด้านหลัง

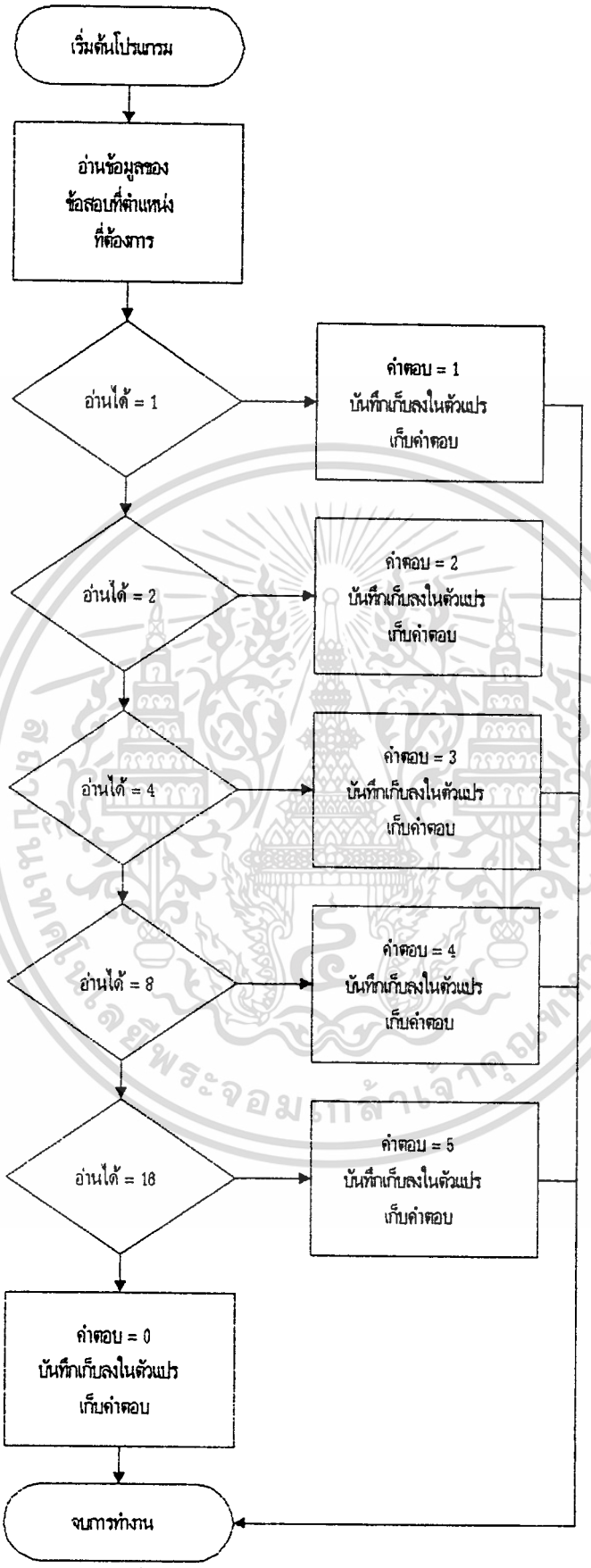
สำหรับวิธีการทำงานจะเหมือนกับการถอดรหัสด้านหน้า เพียงแต่มีการกลับตัวเลขจากหน้า กลับมาเป็นหลังก่อน (รูปที่ 7.7)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



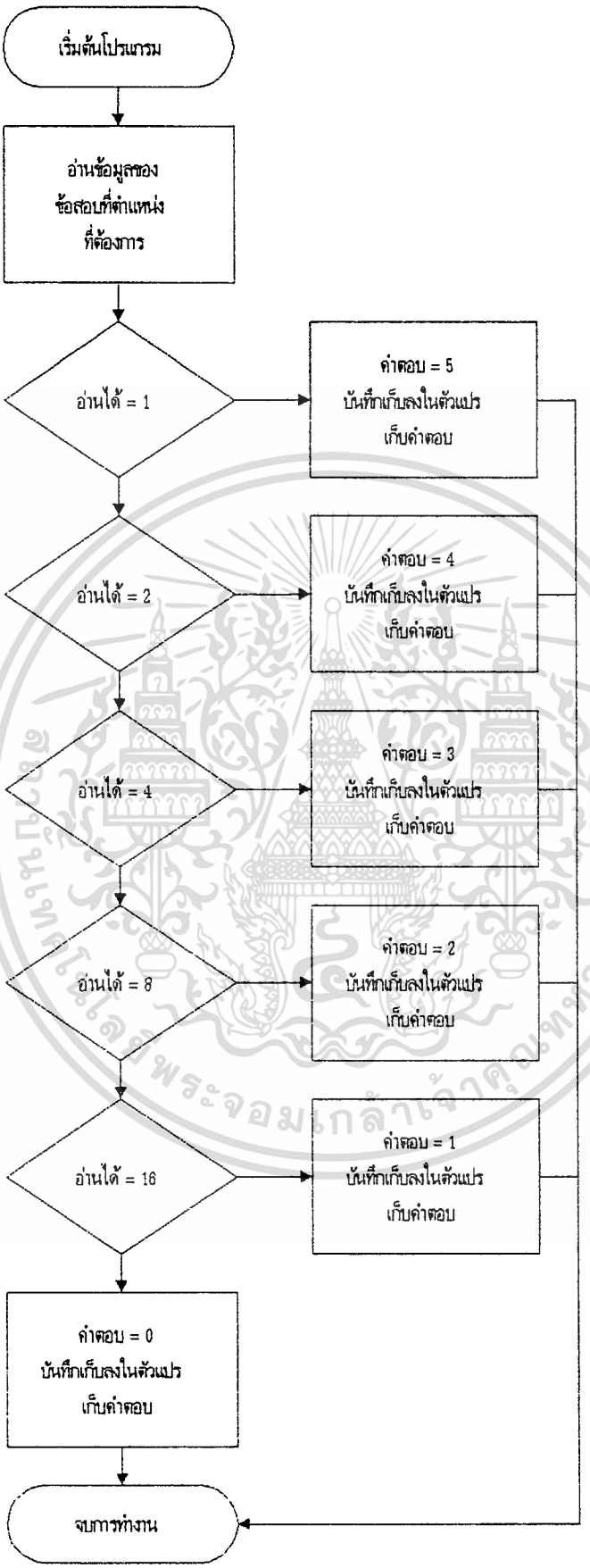
รูปที่ 7.5 ผังการทำงานของโปรแกรมถอดรหัสนักศึกษา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 7.7 ผังการถอดรหัสคำตอบด้านหน้า

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 7.8 ผังการถอดรหัสคำตอบด้านหลัง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

7.2 โปรแกรมบนเครื่องตรวจข้อสอบ

7.2.1 หลักการเบื้องต้น

โปรแกรมบนเครื่องตรวจข้อสอบถูกพัฒนาขึ้นเพื่อให้มีความสามารถในการทำงานเฉพาะจุด โดยไม่มีการกำหนดรายละเอียดที่ใช้ในการตรวจสอบคะแนน เพื่อให้โปรแกรมมีความยืดหยุ่นในการทำงาน โปรแกรมชุดนี้จึงมีหน้าที่เพียงส่งข้อมูลให้กับคอมพิวเตอร์เพื่อทำการประมวลผล แยกต่างหากจากเครื่องตรวจข้อสอบ การทำงานของโปรแกรมจะมีหน้าที่ดังนี้

- ติดต่อสื่อสารระหว่างเครื่องคอมพิวเตอร์กับเครื่องตรวจข้อสอบ
- สั่งให้มอเตอร์หมุนเพื่อนำกระดาษคำตอบเข้าให้ชุดตรวจจับสัญญาณทำงาน
- อ่านข้อมูลที่ได้จากชุดตรวจจับสัญญาณ
- แสดงจำนวนกระดาษคำตอบที่ทำการตรวจในแต่ละครั้ง

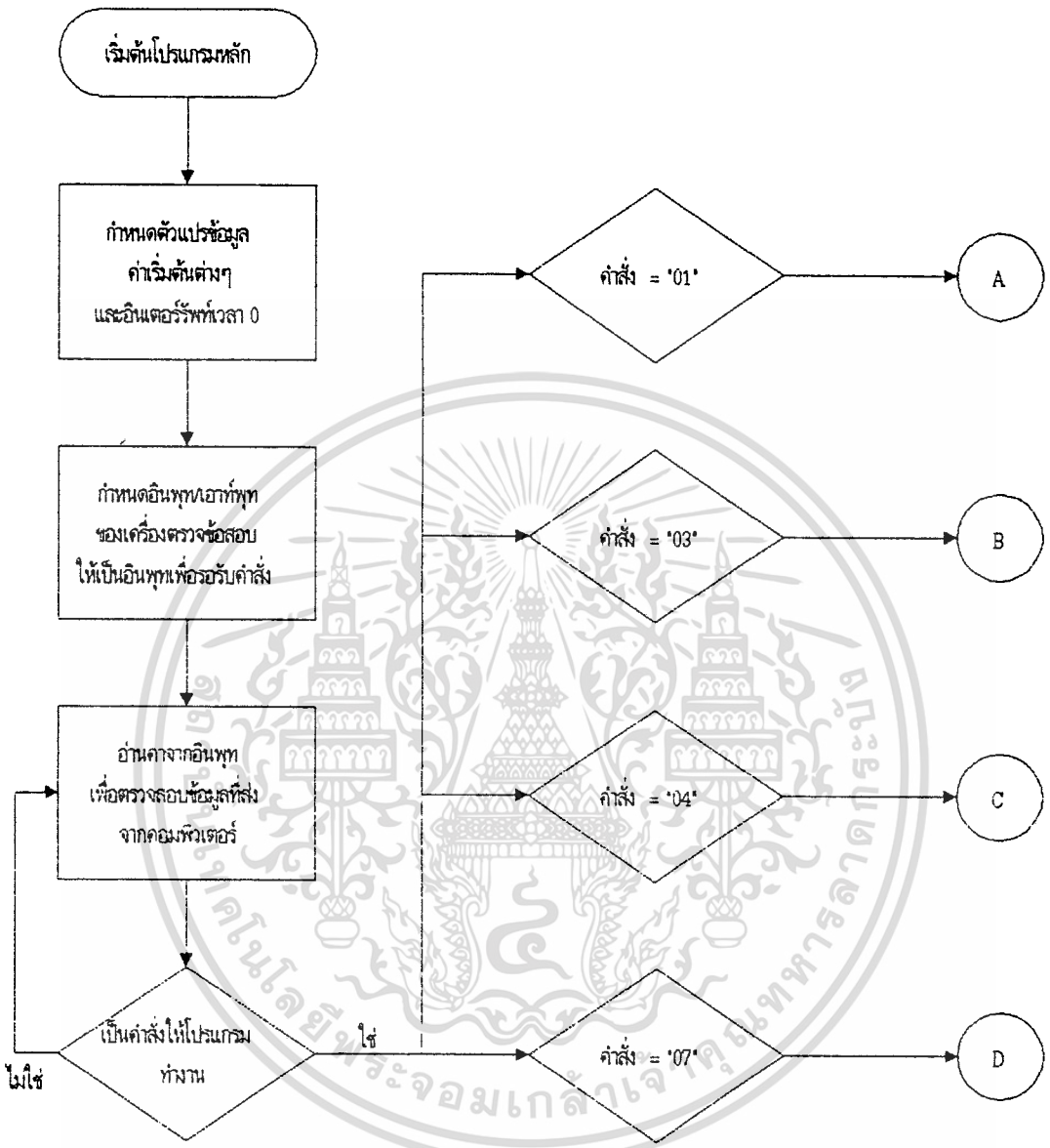
7.2.2 โครงสร้างของโปรแกรมประกอบด้วย :

โปรแกรมใช้งาน ได้ถูกพัฒนาให้มีขั้นตอนในการทำงานดังรายละเอียดต่อไปนี้

1. กำหนดค่าเริ่มต้นของระบบ ซึ่งได้แก่การกำหนดพอร์ตใช้งานต่างๆ และการกำหนด อินเตอร์รัพท์เวลา 0 เพื่อใช้ในการตั้งเวลาในการสื่อสาร
2. รอรับคำสั่งจากเครื่องคอมพิวเตอร์เพื่อทำงาน โดยเมื่อรับคำสั่งแล้วจะไปทำตามโปรแกรมย่อยต่างๆ ตามลักษณะการทำงานของโปรแกรมย่อยนั้นๆ แต่หลังจากทำงานที่โปรแกรมย่อยเสร็จก็จะกลับมารอรับคำสั่งจากเครื่องคอมพิวเตอร์ต่อ จนกว่าจะปิดเครื่อง
3. โปรแกรมย่อยการทำงานของโปรแกรมบนเครื่องตรวจข้อสอบประกอบด้วย
    - โปรแกรมทดสอบการทำงานของมอเตอร์
    - โปรแกรมทดสอบการทำงานของชุดตรวจจับ
    - โปรแกรมทดสอบการทำงานของวงจรนับกระดาษคำตอบ
    - โปรแกรมอ่านผลลัพธ์ที่ได้จากการตรวจจับของชุดตรวจจับ
4. นอกจากโปรแกรมที่กล่าวมาแล้วในระบบของโปรแกรมเครื่องตรวจข้อสอบยังได้มีการพัฒนาโปรแกรมย่อยที่จะถูกเรียกใช้อย่างสม่ำเสมอในโปรแกรมย่อยหลัก เพื่อความสะดวกในการใช้งาน และการพัฒนาโปรแกรม

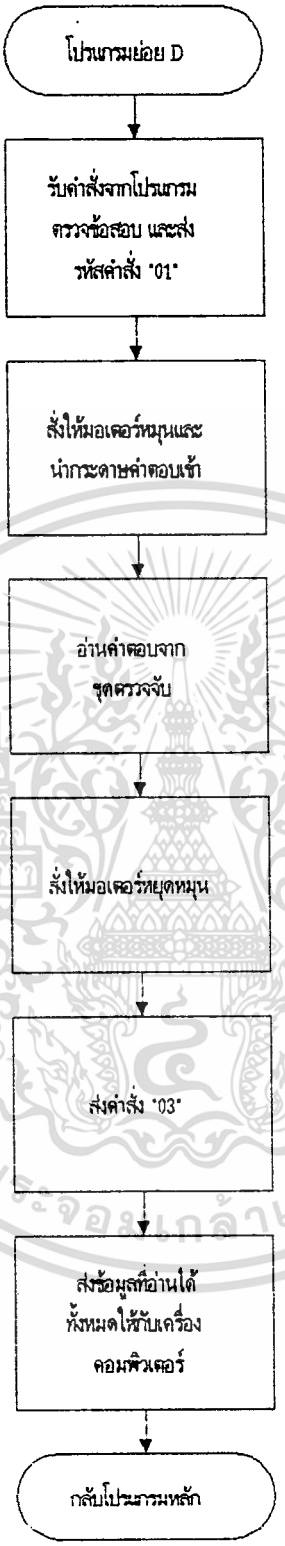
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

7.2.3 ผังการทำงานโปรแกรมบนเครื่องตรวจข้อสอบ



รูปที่ 7.9 ผังการทำงานของโปรแกรมบนเครื่องตรวจข้อสอบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 7.10 ผังการทำงานของโปรแกรมอ่านข้อมูลจากชุดตรวจจับพร้อมส่งข้อมูลให้เครื่องคอมพิวเตอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

7.3 รายละเอียดโปรแกรม

โปรแกรม	รายละเอียด
about.frm	แสดงโปรแกรม
answer.frm	โปรแกรมแสดงคำตอบของระบบ
anwermo.frm	โปรแกรมหลักตัวเลือกรายละเอียดคำตอบ
cnt_test.frm	โปรแกรมทดสอบการทำงานของวงจรนับ
comm.frm	โปรแกรมหลักอ่านข้อสอบจากเครื่องตรวจข้อสอบและหาคะแนน
gphview.frm	โปรแกรมพิมพ์กราฟผลการตรวจคำตอบลงบนหน้าจอ
inf_test.frm	โปรแกรมทดสอบการทำงานของชุดตรวจจับ
look_up.frm	โปรแกรมแสดงคะแนนของนักศึกษา
options.frm	โปรแกรมหลักตัวเลือกปรับค่าของโปรแกรม
rdanswer.frm	โปรแกรมอ่านคำตอบเข้าระบบ
selftest.frm	โปรแกรมหลักทดสอบการทำงานของอุปกรณ์รอบๆเครื่องตรวจ
static.frm	โปรแกรมแสดงสถิติของคำตอบที่ตรวจได้
stmotor.frm	โปรแกรมทดสอบการทำงานของมอเตอร์
tester.frm	โปรแกรมหลักเครื่องตรวจข้อสอบ
checksub.bas	โปรแกรมย่อยถอดรหัสคำตอบและตรวจคำตอบ
comm.bas	โปรแกรมย่อยติดต่อสื่อสาร
extfunc.bas	โปรแกรมย่อยประกาศชื่อโปรแกรมย่อยของวินโดว์
global.bas	โปรแกรมย่อยประกาศตัวแปรทั่วไป
utils.bas	โปรแกรมย่อยรวมฟังก์ชันและซับรูทีนทั่วไป
xcontrol.asm	โปรแกรมควบคุมเครื่องตรวจข้อสอบ

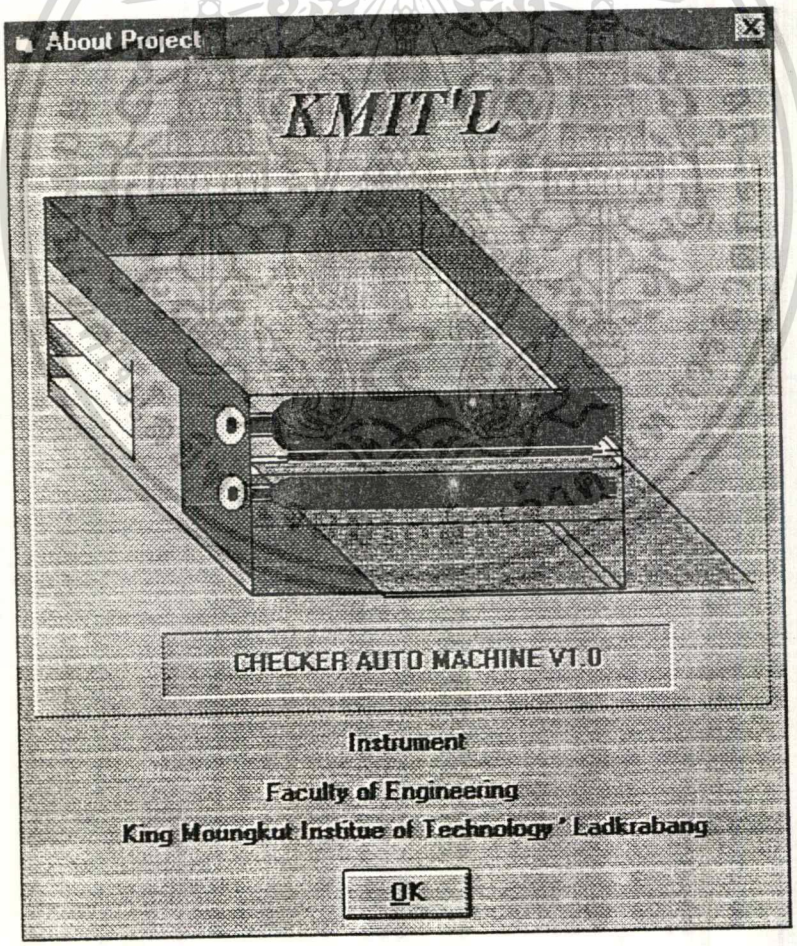
ตารางที่ 7.3 รายละเอียดโปรแกรมทั้งหมดที่ได้รับการพัฒนาบนเครื่องตรวจข้อสอบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

7.4 วิธีการใช้งานเครื่องตรวจข้อสอบ

การใช้งานเครื่องตรวจข้อสอบจำเป็นที่จะต้องเข้าใจหลักการทำงานในส่วนที่เกี่ยวข้อง เนื่องโปรแกรมเครื่องตรวจข้อสอบจะมีการใช้งานข้อมูล ร่วมกับโปรแกรมอื่นๆ เช่นโปรแกรมตัดเกรดนักศึกษา วิธีการใช้งานโปรแกรมให้ทำตามขั้นตอนดังนี้

1. ต่อสายสื่อสารระหว่างเครื่องคอมพิวเตอร์และเครื่องตรวจข้อสอบ โดยสายสื่อสารที่ใช้เป็นสายสื่อสารทางพอร์ตขนาน
2. เปิดเครื่องตรวจข้อสอบเพื่อให้มีสถานะพร้อมในการทำงาน และนำกระดาษคำตอบใส่ในช่องใส่กระดาษคำตอบโดยสามารถใส่ได้ในปริมาณ 15-20 แผ่นต่อการตรวจในแต่ละครั้ง
3. สั่งให้โปรแกรมเครื่องตรวจข้อสอบบนเครื่องคอมพิวเตอร์ทำงาน หลังจากโปรแกรมทำงานตัวโปรแกรม จะแสดงรายละเอียดเกี่ยวกับโปรแกรมดังรูปที่ 7.11 ให้เลือกปุ่ม “OK”



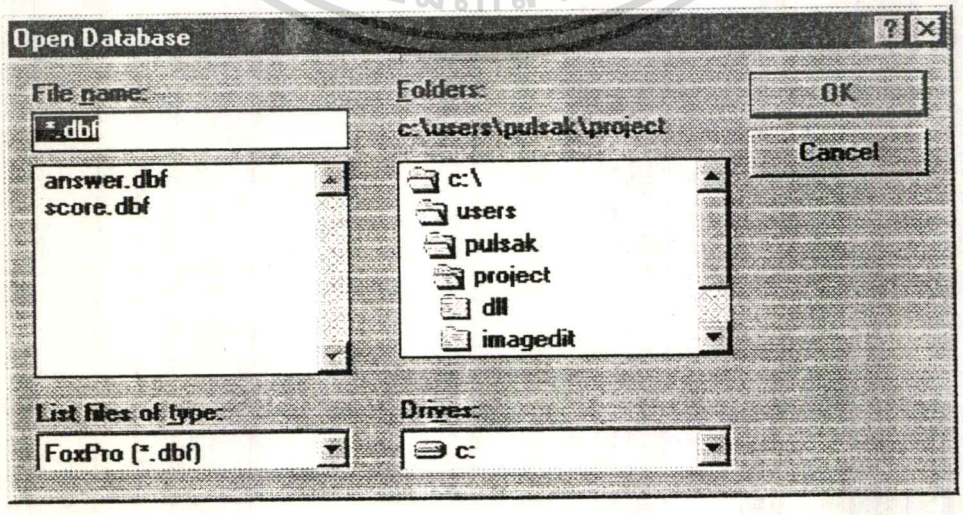
รูปที่ 7.11 แสดงรายละเอียดเกี่ยวกับโปรแกรมเครื่องตรวจข้อสอบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4. หลังจากเลือกปุ่ม "OK" รายละเอียดของโปรแกรมจะหายไป ต่อจากนั้นโปรแกรมจะอยู่ในสถานะที่สามารถใช้งานได้ดังรูปที่ 7.12

รูปที่ 7.12 แสดงสถานะพร้อมทำงานของโปรแกรมเครื่องตรวจข้อสอบ

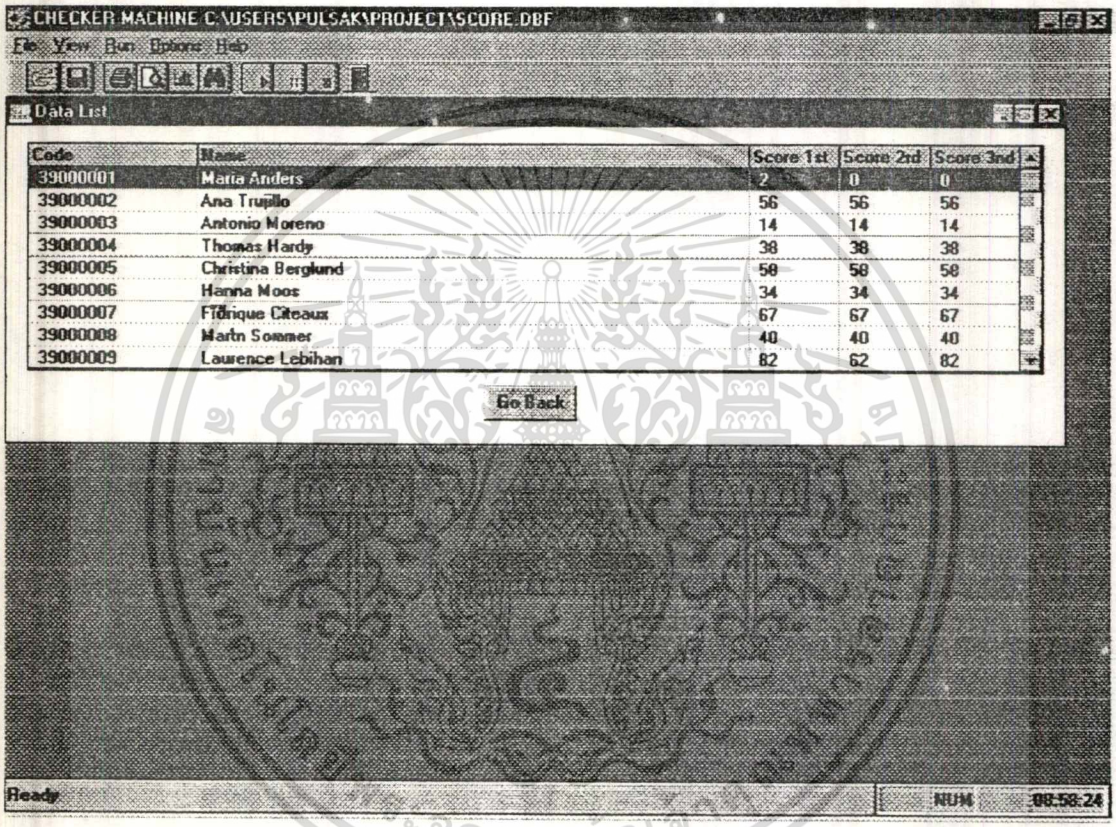
5. การตรวจคำตอบจะต้องมีการเก็บคะแนนที่ทำการตรวจได้ลงในฐานข้อมูล ซึ่งจะมีตารางการทำงานดังรายละเอียดที่กล่าวไปแล้ว ฉนั้นวิธีการเปิดฐานข้อมูลเพื่อเก็บรายละเอียดให้เลือกเมนู "file", "open" หลังจากที่เลือกแล้วจะปรากฏดังรูปที่ 7.13 ให้เปิดตารางข้อมูล "score.dbf" ซึ่งเป็นตารางเก็บรายละเอียดนักศึกษา



รูปที่ 7.13 แสดงการเปิดตารางข้อมูลเพื่อใช้งาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

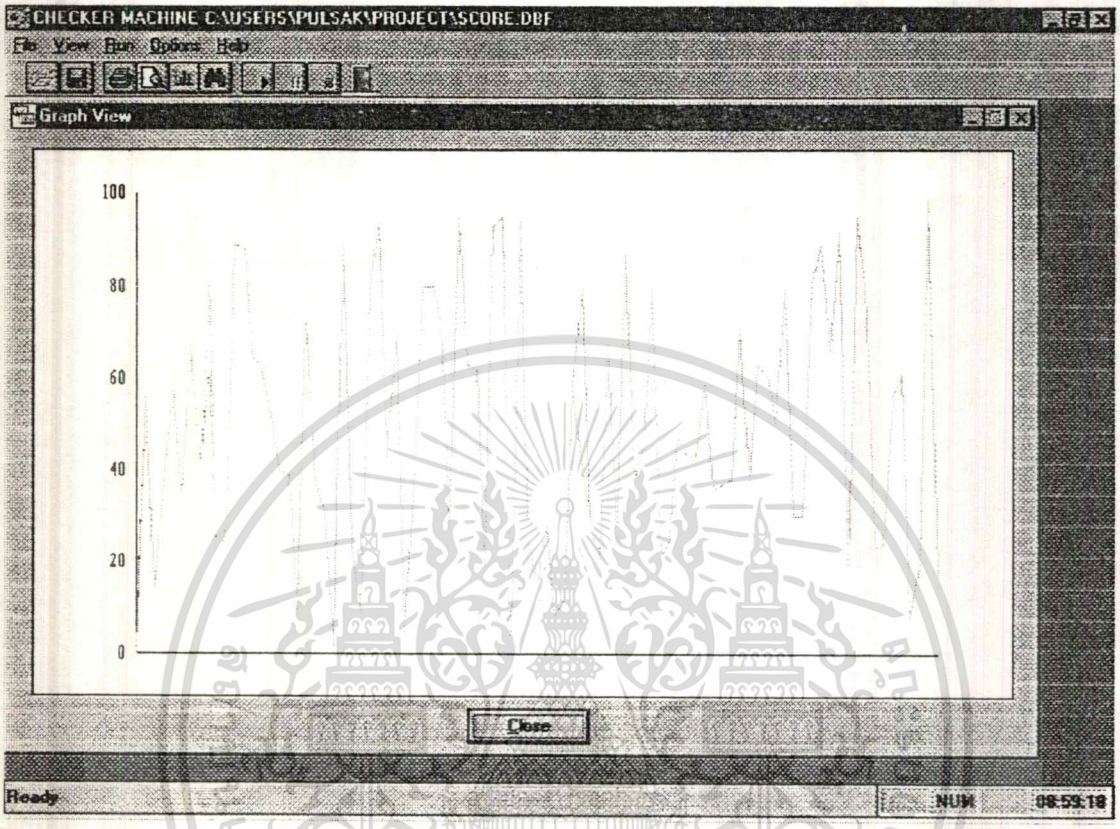
6. เมื่อเปิดฐานข้อมูลของระบบ หากต้องการดูข้อมูลภายในว่ามีนักศึกษาหรือคะแนนของนักศึกษาแต่ละคนสามารถที่จะดูได้โดยเลือกเมนู "View", "Show all data" หลังจากเลือกเมนูที่กล่าวมาข้างต้นจะปรากฏดังรูปที่ 7.14 ซึ่งเป็นการแสดงรายละเอียดทั้งหมดของนักศึกษา ได้แก่ รหัสนักศึกษา, ชื่อนักศึกษา, คะแนนที่ได้จากการตรวจคำตอบทั้ง 3 ครั้ง หลังจากดูข้อมูลนักศึกษาแล้ว หากต้องการออกจากการดูรายละเอียดนักศึกษาให้ใช้เมาส์ชี้ไปที่ "go back" เพื่อกลับเข้าสู่ระบบควบคุมหลัก



รูปที่ 7.14 รูปแสดงรายละเอียดทั้งหมดของนักศึกษา

7. สำหรับผลการตรวจคำตอบซึ่งเก็บอยู่ในตาราง "score.dbf" สามารถที่จะพิมพ์ออกมาเป็นกราฟเพื่อช่วยในการตรวจสอบข้อผิดพลาด ณ ตำแหน่งต่างๆโดยกราฟที่พิมพ์ออกทางหน้าจอจะมีทั้งหมด 3 เส้นด้วยกัน คือผลที่ได้จากการตรวจคำตอบทั้ง 3 ครั้งเพื่อนำมาเปรียบเทียบความแตกต่างของคะแนนที่เกิดขึ้นสำหรับลักษณะของกราฟแสดงดังรูปที่ 7.15 สำหรับวิธีการเรียกกราฟขึ้นมาแสดงให้เลือกเมนู "View", "View graph information"

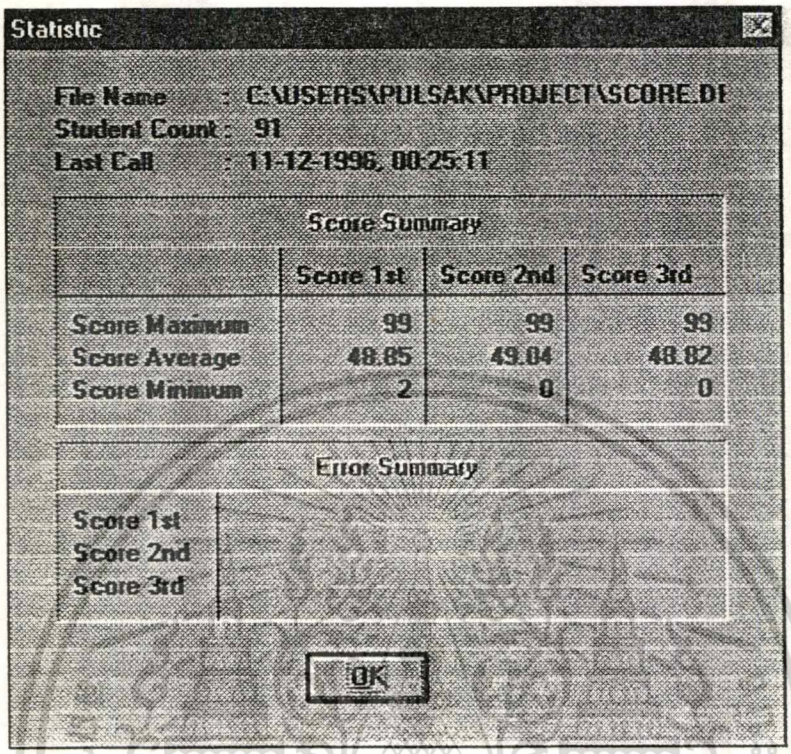
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 7.15 แสดงกราฟผลของการตรวจคำตอบ

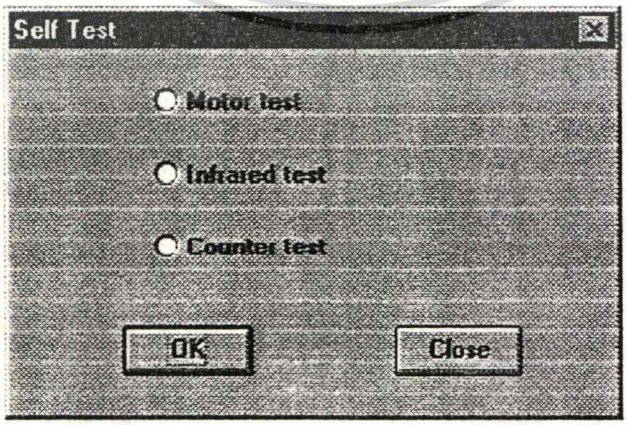
8. นอกจากจะมีกราฟเพื่อใช้ในการตรวจข้อค่าผิดพลาดที่ได้จากการตรวจคำตอบแล้ว ในโปรแกรมเครื่องตรวจข้อสอบยังสามารถทำการหาค่าเฉลี่ยของคะแนน, ค่าสูงสุด และค่าต่ำสุด โดยทำการพิมพ์เป็นตารางสรุปรายละเอียด พร้อมกันนี้ยังแสดงข้อผิดพลาดที่เกิดจากการตรวจคำตอบไว้ให้ดูเพื่อใช้ในการพิจารณา แสดงดังรูปที่ 7.16 สำหรับวิธีการเรียกใช้งานให้เลือกเมนู "View". "Show Summary and Statistic"

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 7.16 ตารางสรุปรายละเอียดของการตรวจคำตอบ

9. อุปกรณ์ของเครื่องตรวจข้อสอบสามารถที่จะใช้โปรแกรมเครื่องตรวจข้อสอบสั่งให้ทำงานเพื่อตรวจสอบ การทำงานในแต่ละจุด สำหรับการเลือกอุปกรณ์ของเครื่องจะแสดงดังรูปที่ 7.17 โดยทำการเลือกเมนู "run", "Respond self-test" โดยอุปกรณ์ของเครื่องตรวจข้อสอบที่สามารถตรวจสอบได้มีด้วยกันดังนี้

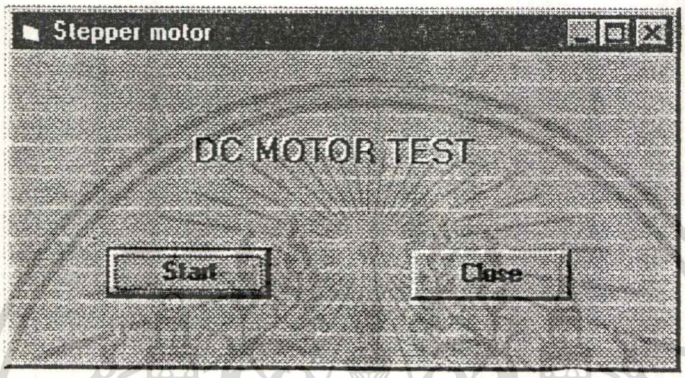


รูปที่ 7.17 แสดงการเลือกอุปกรณ์ของเครื่องตรวจข้อสอบเพื่อทดสอบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

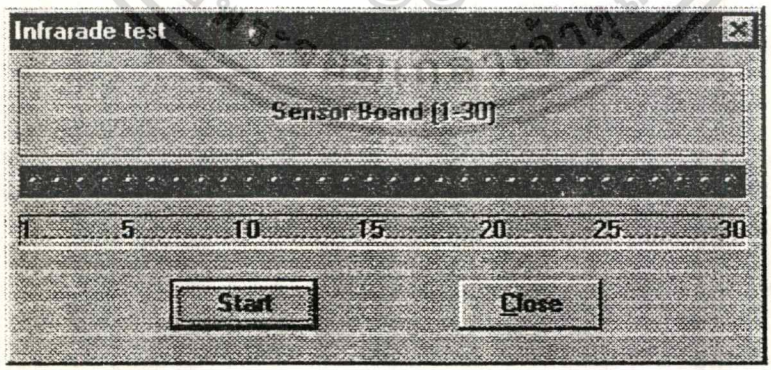
- การทดสอบมอเตอร์นำกระเข้าเครื่องตรวจข้อสอบ
- การทดสอบชุดตรวจจับคำตอบ
- การทดสอบชุดนับกระดาษคำตอบ

10. จากรูปที่ 7.18 เป็นหน้าจอโปรแกรมทดสอบการทำงานของมอเตอร์นำกระดาษเข้าเครื่องตรวจข้อสอบ ให้เลือก "Start" เพื่อสั่งให้มอเตอร์ทำงาน หากมอเตอร์หมุนทั้ง 2 ตัวแสดงว่าขณะนี้มอเตอร์มีการทำงานปกติ



รูปที่ 7.18 หน้าจอโปรแกรมทดสอบการทำงานของมอเตอร์

11. สำหรับชุดตรวจจับกระดาษคำตอบซึ่งมีความสำคัญมาในการทำงาน เพราะความผิดพลาดที่เกิดขึ้นมักจะมาจากจุดนี้สามารถจะทำการตรวจสอบตำแหน่งต่างๆ ของชุดตรวจจับได้ ดังรูปที่ 7.19 จากรูปจะทำให้สามารถทราบได้ว่าปัจจุบันที่ชุดตรวจจับตัวใดบ้างที่มี ปัญหาเกิดการปิดบังสัญญาณหรือ ชุดตรวจจับมีปัญหาต่างๆ ห

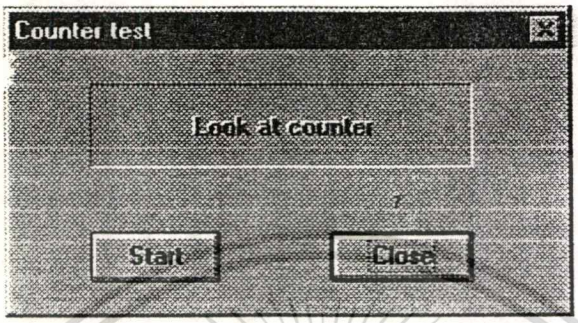


รูปที่ 7.19 หน้าจอโปรแกรมทดสอบการทำงานของชุดตรวจจับ

12. อุปกรณ์ของเครื่องตรวจข้อสอบที่เหลือคือชุดนับกระดาษคำตอบ เพื่อตรวจสอบการทำงานของวงจรนับ ว่ามีการทำงานปกติหรือไม่โดยจะทำการนับเป็นตัวเลขเปลี่ยนแปลงไปตลอด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เวลาดังรูปที่ 7.20 เป็นหน้าจอแสดงโปรแกรมทดสอบการทำงานของชุดนับกระดาษคำตอบบนเครื่อง ตรวจข้อสอบ

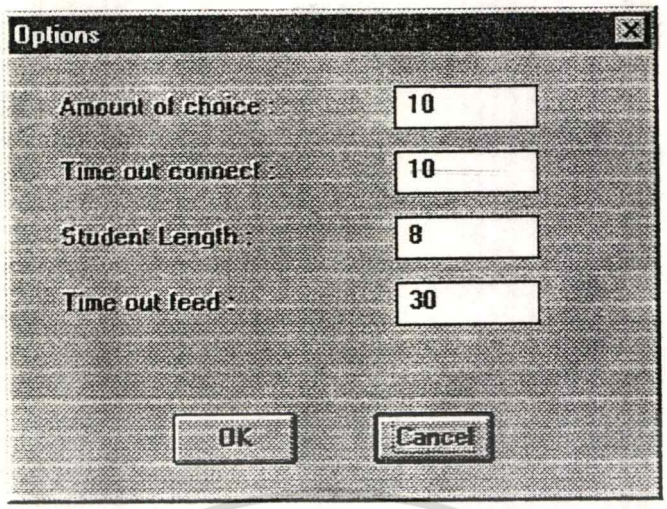


รูปที่ 7.20 หน้าจอโปรแกรมทดสอบการทำงานของชุดนับกระดาษคำตอบ

13. โดยปกติการออกแบบข้อสอบมักจะไม่มีการกำหนดจำนวนข้อที่ตายตัว และสำหรับรหัสนักศึกษาซึ่งแต่ละที่มักจะมีการกำหนดที่ไม่เหมือนกัน ในโปรแกรมเครื่องตรวจข้อสอบบนเครื่องคอมพิวเตอร์สามารถที่จะให้ ผู้ใช้เปลี่ยนแปลงรายละเอียดของข้อมูลเบื้องต้น เพื่อให้โปรแกรม สามารถที่จะนำไปใช้งานได้ในทุกๆ สถานะการณ์ ดังรูปที่ 7.21 เป็นหน้าจอปรับแต่งข้อมูลเบื้องต้นสำหรับโปรแกรม สามารถเลือกได้จากเมนู "options", "Setup Program" รายละเอียดที่ปรากฏบนหน้าจอมีดังนี้

- Amount of choice - จำนวนของข้อสอบที่จะทำการตรวจ ซึ่งมีผลต่อคะแนนที่จะได้รับ
- Time out connect - เป็นเวลาที่คอมพิวเตอร์ใช้ในการติดต่อกับเครื่องตรวจข้อสอบหากเกินเวลาที่กำหนดโปรแกรมเครื่องตรวจข้อสอบจะแจ้งรายละเอียดให้ทราบ ดังรูปแสดงข้อมูลผิดพลาดที่เกิดขึ้นในระบบซึ่งจะกล่าวถึงต่อไป
- Student Length - เป็นความยาวของรหัสนักศึกษา
- Time out feed - เวลาที่ใช้ในการนำกระดาษเข้าจากถาดสู่ชุดตรวจจับ

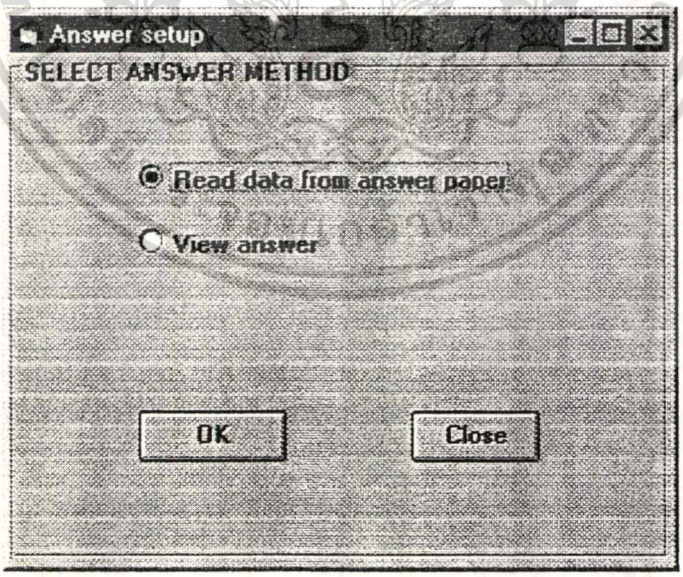
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 7.21 หน้าจอปรับแต่งข้อมูลเบื้องต้นของระบบ

14. การที่โปรแกรมจะตรวจข้อสอบได้จำเป็นที่จะต้องมีคำตอบที่เป็นมาตรฐานสำหรับเครื่อง โดยทำการอ่านคำตอบของระบบจาก กระดาษคำตอบที่ถูกจัดทำขึ้นเฉพาะ การจัดการคำตอบของระบบสามารถทำได้โดยแสดงดังรูปที่ 7.22 โดยเลือกเมนู "options", "Setup Answer" รายละเอียดที่ปรากฏบนหน้าจอมีดังนี้

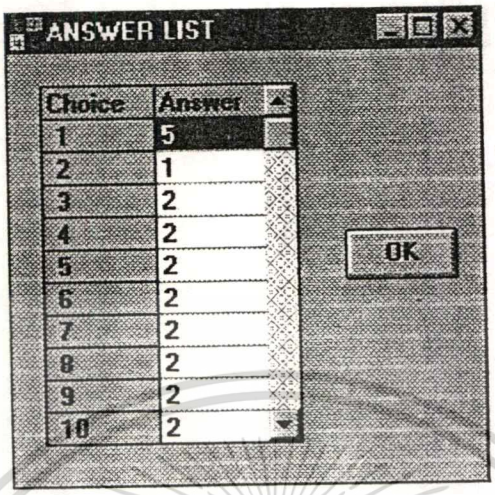
- Read data from answer paper - คือการเรียกโปรแกรมอ่านคำตอบจากกระดาษคำตอบมาตรฐาน
- View answer - แสดงรายละเอียดคำตอบของระบบ



รูปที่ 7.22 หน้าจอโปรแกรมจัดการคำตอบของระบบ

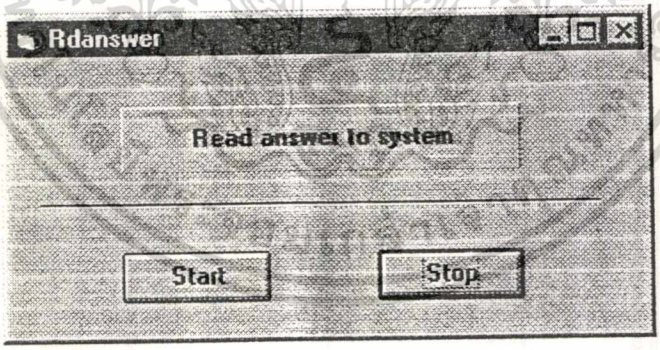
15. หากเลือก "View answer" จะเป็นการแสดงคำตอบของระบบว่าปัจจุบันข้อใดมีคำตอบเป็นอย่างไรดังรูปที่ 7.23

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 7.23 หน้าจอแสดงคำตอบของระบบ

16. หากเลือก "Read data from answer paper" จะปรากฏหน้าจอดังรูปที่ 7.24 ซึ่ง ณ ที่นี้จะต้องวางกระดาษคำตอบไว้ในถาดเพื่อให้เครื่องสามารถที่จะดึงกระดาษคำตอบมาตรฐาน โดยจะต้องเลือก "Start" เพื่อสั่งให้เครื่องตรวจข้อสอบทำงาน หลังจากนั้นข้อมูลของคำตอบจะเข้าสู่ระบบ โดยสามารถตรวจสอบความถูกต้องของคำตอบได้จากหน้าจอในรูปที่ 7.23



รูปที่ 7.24 หน้าจอโปรแกรมอ่านคำตอบมาตรฐานเข้าสู่ระบบ

17. รายละเอียดต่อไปเป็นรายละเอียดของหน้าจอแสดงความผิดพลาดที่เกิดขึ้นของโปรแกรม ดังนี้

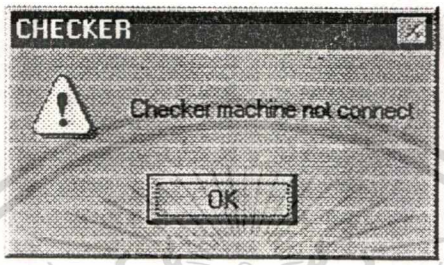
รูปที่ 7.25 - แสดงข้อความว่าไม่สามารถที่จะติดต่อกับเครื่องตรวจข้อสอบได้ ให้ตรวจสอบว่าเปิดเครื่องตรวจข้อสอบหรือยัง และตรวจสอบสายสัญญาณการสื่อสารว่ามีการต่ออยู่หรือไม่

รูปที่ 7.26 - แสดงข้อความผิดพลาดเมื่อทำการเลือกฐานข้อมูลโดยมีโครงสร้างที่โปรแกรมไม่รูปจัก โดยโปรแกรมจะทำการตรวจสอบฐานข้อมูลที่จะทำการเปิดก่อนว่าตรงกับ

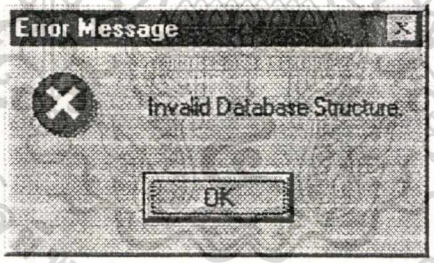
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ฐานข้อมูลของระบบหรือไม่ ถ้าไม่ตรงจะแจ้งข้อความเตือนพร้อมกับหยุดการทำงานของระบบไว้ ณ ที่เดิม

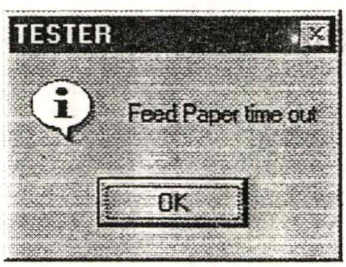
รูปที่ 7.27 - แสดงข้อความว่าขณะนี้กระดาษที่ถาดหมดให้ใส่กระดาษคำตอบลงในถาดให้เรียบร้อยก่อนที่ระบบจะเริ่มทำการตรวจคำตอบต่อ



รูปที่ 7.25 แสดงข้อความไม่สามารถติดต่อกับเครื่องตรวจข้อสอบได้



รูปที่ 7.26 คำเตือนแสดงการเปิดฐานข้อมูลที่มีโครงสร้างไม่ตรงกับระบบ

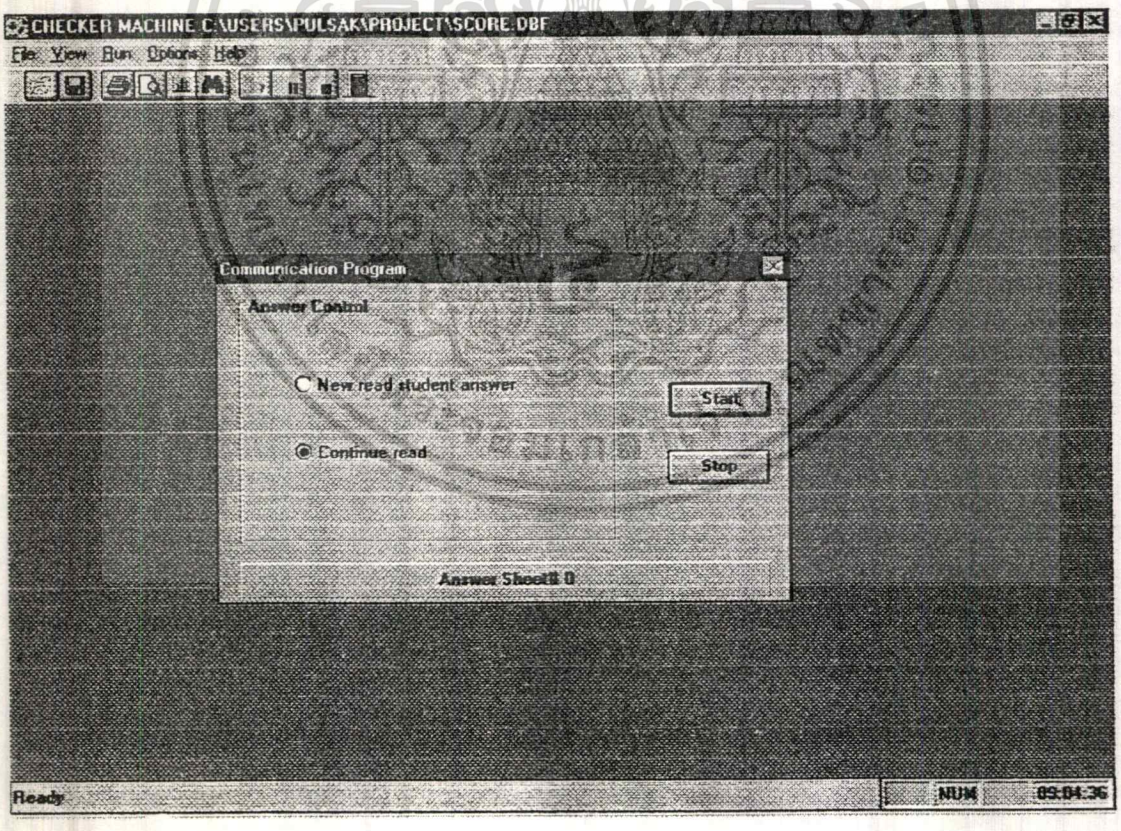


รูปที่ 7.27 แสดงข้อความว่าขณะนี้กระดาษที่ถาดของเครื่องตรวจคำตอบหมด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

18. ลำดับต่อไปจะเป็นการทำงานในส่วนของการอ่านคำตอบจากกระดาษคำตอบของนักศึกษา จากรูปที่ 7.28 เป็นหน้าจอแสดงการอ่านคำตอบจากกระดาษคำตอบของนักศึกษาเพื่อทำการตรวจให้คะแนน โดยเลือกจากเมนู "run", "start" รายละเอียดในรูปที่ 7.28 ประกอบด้วย

- Answer Sheet# - คือจำนวนกระดาษคำตอบที่ทำการตรวจของระบบ ณ เวลานั้น
- New read student answer - เป็นตัวเลือกเพื่อบอกโปรแกรมว่าต้องการอ่านคำตอบของระบบใหม่ การเลือกตัวเลือกนี้จะมีผลต่อ จำนวนกระดาษคำตอบที่ทำการอ่านซึ่งจะถูกเริ่มนับจาก ศูนย์ใหม่
- Continue read - เป็นตัวเลือกที่ช่วยให้การนับกระดาษคำตอบที่ทำการตรวจมีการนับอย่างต่อเนื่อง เหตุผลที่มีตัวเลือกนี้ เนื่องจากอาจเกิดปัญหา กระดาษติดขัด ทำให้จำเป็นที่จะต้องหยุดการทำงานของโปรแกรมชั่วคราวโดยการสั่ง "stop" หลังจากที่แก้ปัญหาเรียบร้อยเมื่อต้องการนับต่อจะสามารถทำได้อย่างสะดวกขึ้น



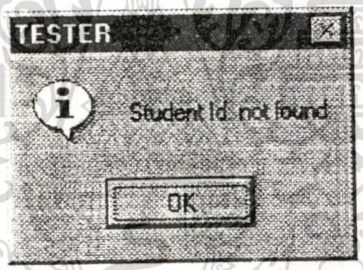
รูปที่ 7.28 หน้าจอแสดงโปรแกรมอ่านคำตอบจากกระดาษคำตอบของนักศึกษา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

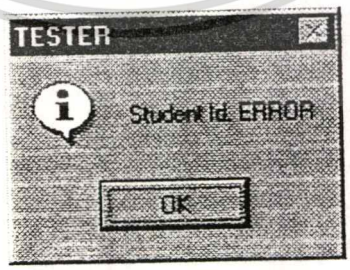
19. โดยปกติการตรวจคำตอบหากไม่มีข้อผิดพลาดใดๆ เกิดขึ้นระบบโปรแกรมจะดำเนินการตรวจคำตอบไปเรื่อยๆ แต่หากมีข้อผิดพลาดในการตรวจคำตอบตัวโปรแกรมจะแจ้งรายละเอียดให้ทราบ ซึ่งจะมีข้อผิดพลาด 2 ส่วนด้วยกัน คือ

- รูปที่ 7.29 - ไม่สามารถพบรหัสนักศึกษาในฐานข้อมูล
- รูปที่ 7.30 - รหัสนักศึกษาที่อ่านไม่สามารถตรวจสอบได้ แสดงว่ามีการกำหนดรหัสผิด

ข้อผิดพลาดทั้ง 2 ระบบจะไม่มีการบันทึกคำตอบ แต่จะแจ้งให้ทราบทางหน้าจอเพื่อทำการคัดกระดาษคำตอบออก เพื่อตรวจสอบอีกครั้ง



รูปที่ 7.29 แสดงข้อความเตือนว่าไม่พบรหัสนักศึกษาในฐานข้อมูล



รูปที่ 7.30 แสดงข้อความเตือนว่ารหัสนักศึกษาที่ทำการตรวจมีความผิดพลาด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

มหาวิทยาลัยธรรมดำแหง

*** คำแนะนำ ***

* ใช้ดินสอดำ 2B หรือดำกว่า ระบายในวงกลมที่ต้องการ

* เมื่อต้องการแก้ไขวงที่ระบายไปแล้ว ใช้ยางลบลบให้สะอาดก่อน แล้วจึงระบายวงใหม่

แถว_____

ที่นั่ง_____

ชื่อ-สกุล _____

สถานที่ _____

กระบวนวิชา_____ วันที่_____ เวลา_____

รหัสประจำตัว

1	2	3	4	5	6	7	8	9
0	0	0	0	0	0	0	0	0
1	1	1	1	1	1	1	1	1
2	2	2	2	2	2	2	2	2
3	3	3	3	3	3	3	3	3
4	4	4	4	4	4	4	4	4
5	5	5	5	5	5	5	5	5
6	6	6	6	6	6	6	6	6
7	7	7	7	7	7	7	7	7
8	8	8	8	8	8	8	8	8
9	9	9	9	9	9	9	9	9

ตอบได้เฉพาะนักศึกษาที่มีรหัส 2 ตัวแรกเป็นเลขตั้งแต่ 27 ขึ้นไป

1 ① ② ③ ④ ⑤	16 ① ② ③ ④ ⑤	36 ① ② ③ ④ ⑤	56 ① ② ③ ④ ⑤
2 ① ② ③ ④ ⑤	17 ① ② ③ ④ ⑤	37 ① ② ③ ④ ⑤	57 ① ② ③ ④ ⑤
3 ① ② ③ ④ ⑤	18 ① ② ③ ④ ⑤	38 ① ② ③ ④ ⑤	58 ① ② ③ ④ ⑤
4 ① ② ③ ④ ⑤	19 ① ② ③ ④ ⑤	39 ① ② ③ ④ ⑤	59 ① ② ③ ④ ⑤
5 ① ② ③ ④ ⑤	20 ① ② ③ ④ ⑤	40 ① ② ③ ④ ⑤	60 ① ② ③ ④ ⑤
6 ① ② ③ ④ ⑤	21 ① ② ③ ④ ⑤	41 ① ② ③ ④ ⑤	61 ① ② ③ ④ ⑤
7 ① ② ③ ④ ⑤	22 ① ② ③ ④ ⑤	42 ① ② ③ ④ ⑤	62 ① ② ③ ④ ⑤
8 ① ② ③ ④ ⑤	23 ① ② ③ ④ ⑤	43 ① ② ③ ④ ⑤	63 ① ② ③ ④ ⑤
9 ① ② ③ ④ ⑤	24 ① ② ③ ④ ⑤	44 ① ② ③ ④ ⑤	64 ① ② ③ ④ ⑤
10 ① ② ③ ④ ⑤	25 ① ② ③ ④ ⑤	45 ① ② ③ ④ ⑤	65 ① ② ③ ④ ⑤
11 ① ② ③ ④ ⑤	26 ① ② ③ ④ ⑤	46 ① ② ③ ④ ⑤	66 ① ② ③ ④ ⑤
12 ① ② ③ ④ ⑤	27 ① ② ③ ④ ⑤	47 ① ② ③ ④ ⑤	67 ① ② ③ ④ ⑤
13 ① ② ③ ④ ⑤	28 ① ② ③ ④ ⑤	48 ① ② ③ ④ ⑤	68 ① ② ③ ④ ⑤
14 ① ② ③ ④ ⑤	29 ① ② ③ ④ ⑤	49 ① ② ③ ④ ⑤	69 ① ② ③ ④ ⑤
15 ① ② ③ ④ ⑤	30 ① ② ③ ④ ⑤	50 ① ② ③ ④ ⑤	70 ① ② ③ ④ ⑤
	31 ① ② ③ ④ ⑤	51 ① ② ③ ④ ⑤	71 ① ② ③ ④ ⑤
	32 ① ② ③ ④ ⑤	52 ① ② ③ ④ ⑤	72 ① ② ③ ④ ⑤
	33 ① ② ③ ④ ⑤	53 ① ② ③ ④ ⑤	73 ① ② ③ ④ ⑤
	34 ① ② ③ ④ ⑤	54 ① ② ③ ④ ⑤	74 ① ② ③ ④ ⑤
	35 ① ② ③ ④ ⑤	55 ① ② ③ ④ ⑤	75 ① ② ③ ④ ⑤

DATA PRODUCTS TOPPAN MOORE LTD. 319-1484-5.

กระดาษคำตอบหน้า 1

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กระดาษคำตอบ

	Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ Ⓕ Ⓖ Ⓗ Ⓘ Ⓙ Ⓚ Ⓛ Ⓜ Ⓝ Ⓞ Ⓟ Ⓠ Ⓡ Ⓢ Ⓣ Ⓤ Ⓥ Ⓦ Ⓧ Ⓨ Ⓩ
	Ⓐ Ⓑ Ⓒ Ⓓ Ⓔ Ⓕ Ⓖ Ⓗ Ⓘ Ⓙ Ⓚ Ⓛ Ⓜ Ⓝ Ⓞ Ⓟ Ⓠ Ⓡ Ⓢ Ⓣ Ⓤ Ⓥ Ⓦ Ⓧ Ⓨ Ⓩ
	⓪ ① ② ③ ④ ⑤ ⑥ ⑦ ⑧ ⑨
	⓪ ① ② ③ ④ ⑤ ⑥ ⑦ ⑧ ⑨
	⓪ ① ② ③ ④ ⑤ ⑥ ⑦ ⑧ ⑨

ห้ามขีดเขียนหรือกรอกข้อความอื่นใด นอกเหนือจากที่ระบุไว้

* เพื่อประโยชน์ของนักศึกษาเอง จงอ่านและปฏิบัติตามคำแนะนำ

* ตัวอย่างวิธีทำเครื่องหมายในการตอบ *

1 ① ② ③ ● ⑤ ← ถูกวิธี

2 ① ② ③ ⊗ ⑤ ← ผิดวิธี

3 ① ② ③ ◔ ⑤ ← ผิดวิธี

4 ① ② ③ • ⑤ ← ผิดวิธี

76 ① ② ③ ④ ⑤	96 ① ② ③ ④ ⑤	116 ① ② ③ ④ ⑤	
77 ① ② ③ ④ ⑤	97 ① ② ③ ④ ⑤	117 ① ② ③ ④ ⑤	
78 ① ② ③ ④ ⑤	98 ① ② ③ ④ ⑤	118 ① ② ③ ④ ⑤	
79 ① ② ③ ④ ⑤	99 ① ② ③ ④ ⑤	119 ① ② ③ ④ ⑤	
80 ① ② ③ ④ ⑤	100 ① ② ③ ④ ⑤	120 ① ② ③ ④ ⑤	
81 ① ② ③ ④ ⑤	101 ① ② ③ ④ ⑤	121 ① ② ③ ④ ⑤	136 ① ② ③ ④ ⑤
82 ① ② ③ ④ ⑤	102 ① ② ③ ④ ⑤	122 ① ② ③ ④ ⑤	137 ① ② ③ ④ ⑤
83 ① ② ③ ④ ⑤	103 ① ② ③ ④ ⑤	123 ① ② ③ ④ ⑤	138 ① ② ③ ④ ⑤
84 ① ② ③ ④ ⑤	104 ① ② ③ ④ ⑤	124 ① ② ③ ④ ⑤	139 ① ② ③ ④ ⑤
85 ① ② ③ ④ ⑤	105 ① ② ③ ④ ⑤	125 ① ② ③ ④ ⑤	140 ① ② ③ ④ ⑤
86 ① ② ③ ④ ⑤	106 ① ② ③ ④ ⑤	126 ① ② ③ ④ ⑤	141 ① ② ③ ④ ⑤
87 ① ② ③ ④ ⑤	107 ① ② ③ ④ ⑤	127 ① ② ③ ④ ⑤	142 ① ② ③ ④ ⑤
88 ① ② ③ ④ ⑤	108 ① ② ③ ④ ⑤	128 ① ② ③ ④ ⑤	143 ① ② ③ ④ ⑤
89 ① ② ③ ④ ⑤	109 ① ② ③ ④ ⑤	129 ① ② ③ ④ ⑤	144 ① ② ③ ④ ⑤
90 ① ② ③ ④ ⑤	110 ① ② ③ ④ ⑤	130 ① ② ③ ④ ⑤	145 ① ② ③ ④ ⑤
91 ① ② ③ ④ ⑤	111 ① ② ③ ④ ⑤	131 ① ② ③ ④ ⑤	146 ① ② ③ ④ ⑤
92 ① ② ③ ④ ⑤	112 ① ② ③ ④ ⑤	132 ① ② ③ ④ ⑤	147 ① ② ③ ④ ⑤
93 ① ② ③ ④ ⑤	113 ① ② ③ ④ ⑤	133 ① ② ③ ④ ⑤	148 ① ② ③ ④ ⑤
94 ① ② ③ ④ ⑤	114 ① ② ③ ④ ⑤	134 ① ② ③ ④ ⑤	149 ① ② ③ ④ ⑤
95 ① ② ③ ④ ⑤	115 ① ② ③ ④ ⑤	135 ① ② ③ ④ ⑤	150 ① ② ③ ④ ⑤

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โปรแกรม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โปรแกรม xcontrol.asm

```
;
;*********************************************************;
; CHECKER AUTO MACHINE CONTROL PROGRAM USING THE ANT-32     ;
;     FOR COMPATIBLE PC BY PARALLEL PRINTER PORTS           ;
;                                                           ;
;             INSTUMENT ENGINEERING KMIT'L                  ;
;                       1996                                ;
;*********************************************************;
;
$Title "ANT-32 CHECKER AUTOMACHINE CONTROL"
;
; Internal Register/RAM Usage:
;
HLDST    EQU     08h              ; Memory hold step data
STEP     EQU     0Ah              ; STEP COUNT 2 BYTE
SNS1     EQU     0Ch              ; Sensor board 1 count
SNS2     EQU     0Dh              ; Sensor board 2 count
TO_VAL   EQU     11h              ; Time out value
BUF1PTR  EQU     13h              ; Buffer 1 pointer
BUFFER   EQU     20h              ; Buffer read data from sensor reserve 8 byte
                                  ; 20h - 27h


;
; Register address and constant value help for write program.
;
; Register :
REG_A    EQU     0E0h             ; Reg. A addr.
REG_B    EQU     0F0h             ; Reg. B addr.
REG_R0   EQU     000h             ; Reg. R0 addr.
REG_R1   EQU     001h             ; Reg. R1 addr.
REG_R2   EQU     002h             ; Reg. R2 addr.
REG_R3   EQU     003h             ; Reg. R3 addr.
REG_R4   EQU     003h             ; Reg. R4 addr.
REG_R5   EQU     003h             ; Reg. R5 addr.
REG_R6   EQU     003h             ; Reg. R6 addr.
REG_R7   EQU     003h             ; Reg. R7 addr.
REG_DPH  EQU     083h             ; Reg. DPH addr.
REG_DPL  EQU     082h             ; Reg. DPL addr.
REG_TL0  EQU     08Ah             ; Reg. TL0 addr.
REG_TH0  EQU     08Ch             ; Reg. TH0 addr.
;
; Constants :
TRUE     EQU     01h
FALSE    EQU     00h
ON       EQU     01h
OFF      EQU     00h
;
; Initialize port address
; 8255 port 1
PORTA_1  EQU     0F800h           ; PORT A
PORTB_1  EQU     0F801h           ; PORT B
PORTC_1  EQU     0F802h           ; PORT C
CONTP_1  EQU     0F803h           ; MODE PORT
CONWD_1  EQU     00091h           ; CONTROL WORD

; 8255 port 2
PORTA_2  EQU     0FC00h           ; PORT A
PORTB_2  EQU     0FC01h           ; PORT B
PORTC_2  EQU     0FC02h           ; PORT C
CONTP_2  EQU     0FC03h           ; MODE PORT
CONWD_2  EQU     0008Ah           ; CONTROL WORD


; Miscellaneous Equates:
;
TIME     EQU     -1000            ; Value for 1000 timer clocks = 1 millisecond.
TIMEHI   EQU     HIGH TIME        ; High byte of timer value.
TIMELO   EQU     LOW TIME         ; Low byte of timer value.
BUFF1    EQU     0500h            ; Buffer address for infrared 1
BUFF2    EQU     0600h            ; Buffer address for infrared 2
DMAXHI   EQU     02h
DMAXLO   EQU     33h
;
; Constants :
TO_DEFAULT       EQU     5000     ; Time out default value ( 5 second )
;
;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
;*****************************************************************************
;
; Reset and Interrupt Jump Table
;
        ORG     00h             ; Power-on reset.
        LJMP    START
;
        ORG     0Bh             ; Timer 0.
        LJMP    TIMEOUT         ; Set time out flag.
;
;*****************************************************************************
;
; Space area for store data
;
        ORG     100h

;
;*****************************************************************************
;
; Power up reset routine:
; Set up timeout timer, enable timer interrupt.
;
START:
        MOV     SP,#40h         ; Initialize stack pointer.

; Initialize interupt timer mode
;
        MOV     IE,#00000111b   ; Timer0, INT0, and INT1 enabled.
        MOV     IP,#00000010b   ; Timer0 high priority
        MOV     TL0,#TIMELO
        MOV     TH0,#TIMEHI
        MOV     TMOD,#00000001b ; Operate Timer0 in mode 1.
        MOV     TCON,#00010101b ; Timer0 run, I0 and I1 = edge.

; Initialize Port 1, 2 for communicate with computer and sensor infrared
;


;
; Initialize Port
; set control word table
;  #  Port  Action      Desciption                    Position
; ---------------------------------------------------------------------
;  1    A       i       Sensor data input             PA0-PA7
;       B       o       5 bit Motor Stepping          PB0-PB3 Data
;                                                     PB4     Latch
;                       3 bit LED Display             PB5-PB7
;       C       i       Link to computer              PC0-PC3
;               o                                     PC4-PC7
; ---------------------------------------------------------------------
;  2    A       o       Output drive motor DC         PA0-PA4
;                       Output drive Counter          PA6-PA7
;       B       i       Not used
;       C       o       Sensor selection              PC0-PC3
;               i       general keyboard              PC4-PC7
;
; wait for 8255 until ready (about 1 second)
        MOV     R2,#01h
        LCALL   DTSEC
; initial port when start
; select port#1
        MOV     A,#CONWD_1
        MOV     DPTR,#CONTP_1
        MOVX    @DPTR,A         ; setup the 8255 no 1

; select port#2
        MOV     A,#CONWD_2
        MOV     DPTR,#CONTP_2
        MOVX    @DPTR,A         ; setup the 8255 no 2


; off motor
        MOV     R2,#0
        LCALL   MOTOR
; reset counter
        LCALL   RCNT
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
;*****************************************************************************
;
; Main Routine:
; Executes while for calling from Computer
;


MAIN:
        MOV     R2,#FALSE
        LCALL   PORTINIT
        MOV     A,R3
        CJNE    A,#TRUE,MAIN


        LCALL   RCVD

CMD1:   CJNE    A,#1,CMD3               ; motor feed test
        MOV     A,#3                    ; select motor 1 and 2
        MOV     R2,#1                   ; motor on
        LCALL   MOTOR                   ; call sub motor

                                        ; Wait until stop press
CMD1L:
        LCALL   RCVD
        CJNE    A,#2,CMD1L

        MOV     A,#3                    ; select motor 1 and 2
        MOV     R2,#0                   ; motor off
        LCALL   MOTOR                   ; call sub motor

        AJMP    MAINDONE

CMD3:   CJNE    A,#3,CMD4
        ; infrared test
        LCALL   SINF
        LCALL   SNDRD
        AJMP    MAINDONE
CMD4:   CJNE    A,#4,CMD7
        ; counter test
CMD4L:
        LCALL   PGEN                    ; send pulse (1 second) to counter
        LCALL   RCVD                    ; receive command from computer
        CJNE    A,#2,CMD4L              ; if command = 2, it mean stop count

        LCALL   RCNT                    ; reset counter

        AJMP    MAINDONE
CMD7:   CJNE    A,#7,MAINDONE
        ; start checker
        LCALL   XSINF

MAINDONE:

        LJMP    MAIN
;
;*****************************************************************************
;
; Time out (Timer0) Interrupt Service:
;
;
TIMEOUT:
        PUSH    PSW
        PUSH    ACC
        CLR     EA                      ; Disable timer
        MOV     TL0,#TIMELO             ; Reload time register
        MOV     TH0,#TIMEHI

        MOV     A,TO_VAL+1
        JZ      TIMEOUT1
        DEC     TO_VAL+1
        SJMP    TIMEOUTEXIT
TIMEOUT1:
        MOV     A,TO_VAL
        JZ      TIMEOUTEXIT
        DEC     TO_VAL
        DEC     TO_VAL+1
TIMEOUTEXIT:
        SETB    EA                      ; Enable timer
        POP     ACC
        POP     PSW
        RETI
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
;
;*****************************************************************************
;
; Pulse generate : Send clock to counter circuit
;
PGEN:
        ORL     A,#01000000b    ; Send pulse high ( Bit A6 )
        MOV     DPTR,#PORTA_2
        MOVX    @DPTR,A

        MOV     R2,#1
        LCALL   DTSEC           ; build pulse width

        ANL     A,#10111111b    ; Send pulse low ( Bit A6 )
        MOV     DPTR,#PORTA_2
        MOVX    @DPTR,A

        MOV     R2,#1
        LCALL   DTSEC           ; build pulse width
        RET

;********* RCNT ***********
; Reset counter
;
RCNT:
        ORL     A,#10000000b    ; Send pulse high ( Bit A7 )
        MOV     DPTR,#PORTA_2
        MOVX    @DPTR,A

        MOV     R2,#1
        LCALL   DTSEC           ; build pulse width

        ANL     A,#01111111b    ; Send pulse low ( Bit A7 )
        MOV     DPTR,#PORTA_2
        MOVX    @DPTR,A

        MOV     R2,#1
        LCALL   DTSEC           ; build pulse width
        RET


;
;*****************************************************************************
; Function and Subroutine for communicate
;
; ------------------------------------------------------------
; GetData       : Get data from Port C1
; Input         : None.
; Output        : A = Data
; Reg. Effect   : DPTR
; ------------------------------------------------------------
; FUNCTION
GetData:
        ; section receive data
        MOV     DPTR,#PORTC_1
        MOVX    A,@DPTR
        ; clear bit 4-7 of read data
        ANL     A,#0FH
        RET

; ------------------------------------------------------------
; PutData       : Put data to Port C1
; Input         : A = Data to be put
; Output        : None.
; Reg. Effect   : DPTR
; ------------------------------------------------------------
; PROCEDURE
PutData:
        MOV     DPTR,#PORTC_1
        ; rolate bit lower to upper
        RL      A
        RL      A
        RL      A
        RL      A
        MOVX    @DPTR,A
        RET
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
; ------------------------------------------------------------
; PortInit      : Initialize the register needed for transfer
; Input         : R2 = Sender = #TRUE if sender, #FALSE if receiver
; Output        : R3 = TRUE if register initializes successfully
; Reg. Effect   :
; ------------------------------------------------------------
PortInit:
        MOV     TO_VAL,#HIGH TO_DEFAULT ; Start time out counter
        MOV     TO_VAL+1,#LOW TO_DEFAULT
        SETB    EA
        CJNE    R2,#TRUE,PIRcv

PISnd:
        MOV     A,#08h                  ; Send  : 00001000b
        LCALL   Putdata

PISndL1:
        LCALL   Getdata                 ; Wait for 00001000b

        CJNE    A,#8,PISndL1T
        MOV     R3,#TRUE
        AJMP    PIExit

; Port initial send loop 1 time out
PISndL1T:
        MOV     R0,TO_VAL+1             ; Time out check
        CJNE    R0,#0,PISndL1
        MOV     R0,TO_VAL
        CJNE    R0,#0,PISndL1
        MOV     R3,#FALSE

        AJMP    PIExit


PIRcv:
        LCALL   Getdata                 ; Wait for 00001000b

        CJNE    A,#8,PIRcvL1T
        AJMP    PIRcvL1

; Port initial receive loop 1 time out
PIRcvL1T:
        MOV     R0,TO_VAL+1             ; Time out check
        CJNE    R0,#0,PIRcv
        MOV     R0,TO_VAL
        CJNE    R0,#0,PIRcv
        MOV     R3,#FALSE

        AJMP    PIExit
PIRcvL1:
        MOV     A,#08h                  ; Send  : 00001000b
        LCALL   Putdata
        MOV     R3,#TRUE

PIExit:
        CLR     EA                      ; Disable interupt timer 0

        RET


; ------------------------------------------------------------
; SendData      : Send data to computer and check the result
; Input         : R2 = Data to be sent
; Output        : A = Transfer successful? (0=error, 01 = O.K.)
; Reg. Effect   : B
; ------------------------------------------------------------
; FUNCTION
SndD:

; --- Send and re-read data ----------------------------------
        ANL     REG_R2,#07h             ; clear upper bit
        MOV     A,R2                    ; Sending set bit 3
        LCALL   PutData

        ; Wait for message or time out
        MOV     TO_VAL,#HIGH TO_DEFAULT ; Start time out counter
        MOV     TO_VAL+1,#LOW TO_DEFAULT
        SETB    EA
SndDW:
        LCALL   GetData
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        JNB     ACC.3,SndDEW            ; wait until bit 3 set
        MOV     R0,TO_VAL+1             ; Time out check
        CJNE    R0,#0,SndDW
        MOV     R0,TO_VAL
        CJNE    R0,#0,SndDW

SndDEW:                                 ; Send data exit wait
        CLR     EA
        LCALL   GetData

        MOV     TO_VAL,#HIGH TO_DEFAULT ; Start time out counter
        MOV     TO_VAL+1,#LOW TO_DEFAULT
        SETB    EA

PRS:                                    ; RESET PORT SEND
        MOV     A,#08h                  ; Send  : 00001000b
        LCALL   Putdata

PRSL1:
        LCALL   Getdata                 ; Wait for 00001000b

        CJNE    A,#8,PRSL1T
        AJMP    PRSE

; Port initial send loop 1 time out
PRSL1T:
        MOV     R0,TO_VAL+1             ; Time out check
        CJNE    R0,#0,PRSL1
        MOV     R0,TO_VAL
        CJNE    R0,#0,PRSL1
PRSE:                                   ; reset port exit
        CLR     EA

        RET


; ---------------------------------------------------------------
; RecvData      : Receives data from remote computer and
;                 send returned parts for testing
;
; Input         : None
; Output        : A = Received data
; Reg. Effect   : B
; ---------------------------------------------------------------
; FUNCTION
RcvD:

; --- Receive and re-send data ------------------------------
        ; Wait for message or time out
        MOV     TO_VAL,#HIGH TO_DEFAULT ; Start time out counter
        MOV     TO_VAL+1,#LOW TO_DEFAULT
        SETB    EA
RcvDW:                                  ; receive data wait
        LCALL   GetData
        JNB     ACC.3,RcvSnd            ; wait until bit 3 set
        MOV     R0,TO_VAL+1             ; Time out check
        CJNE    R0,#0,RcvDW
        MOV     R0,TO_VAL
        CJNE    R0,#0,RcvDW

        CLR     EA
RcvSnd:
        LCALL   GetData
        MOV     R3,A

        LCALL   PutData                 ; send data read again

        MOV     TO_VAL,#HIGH TO_DEFAULT ; Start time out counter
        MOV     TO_VAL+1,#LOW TO_DEFAULT
        SETB    EA

PR:                                     ; Reset port
        LCALL   Getdata                 ; Wait for 00001000b

        CJNE    A,#8,PRL1T
        AJMP    PRL1
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
; Port initial receive loop 1 time out
PRL1T:
        MOV     R0,TO_VAL+1             ; Time out check
        CJNE    R0,#0,PR
        MOV     R0,TO_VAL
        CJNE    R0,#0,PR
PRL1:
        MOV     A,#08h                   ; Send  : 00001000b
        LCALL   Putdata

        CLR     EA                      ; Disable interupt timer 0
        MOV     A,R3

        RET
;
;*****************************************************************************
; Sub Routine:
; For calling and use general purpose.
;
;******* DMSEC ********
;DELAY SUBROUTIME
;IN  = R2
;REG = R2,R3
DMSEC:  MOV     R3,#0E6H ;DELAY 1/1000 SEC
DMSEC1: NOP
        NOP
        DJNZ    R3,DMSEC1
        DJNZ    R2,DMSEC
        RET

;******** DTSEC **********
;DELAY SUBROUTIME
;IN  = R2
;REG = R2,R3,R4
DTSEC:  MOV     R3,#0B3H ;DELAY 1/10 SEC
DTSEC1: MOV     R4,#00H
        DJNZ    R4,$
        NOP
        NOP
        DJNZ    R3,DTSEC1
        DJNZ    R2,DTSEC
        RET

;******** DXSEC **********
; Delay time for control stepping motor
; IN  = R2 time period
; REG = R2,R3
;
DXSEC:  MOV     R3,03Ah
DXSEC1: NOP
        NOP
        DJNZ    R3,DXSEC1
        DJNZ    R2,DXSEC
        RET

;*************************************************************
; CombBit       : Combind bit from register
; Input         : R0 = 2 bit upper combind
;                 R1 = 3 bit middle combind
;                 R2 = 3 bit lower combind
; Output        : A = result of combind
; Reg. Effect   : B
; Bit map       :
;
;               R0.1 R0.0 R1.2 R1.1 R1.0 R0.2 R0.1 R0.0
;   Reg. A      7    6    5    4    3    2    1    0
;*************************************************************
CombBit:
        MOV     A,R0            ; Upper bit ( 2 bit )
        SWAP    A               ; Swap to upper
        RL      A               ; Rolate bit to high bit
        RL      A               ;
        MOV     B,A             ; Store to temp variable
        MOV     A,R1            ; Call next bit to combind
        RL      A               ; Rolate bit to middle bit
        RL      A               ;
        RL      A               ;
        ORL     A,B             ; Execute or bit for combind bit
        ORL     A,R0            ; Result
        RET
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
;
; Increase word in internal memory
; Input : R0 high byte
;         R1 low byte
;
INCW:   INC     @R1
        CJNE    @R1,#0,INCW_EXIT
        INC     @R0
INCW_EXIT:
        RET


;
; Decrease word in internal memory
; Input : R0 high byte
;         R1 low byte
;
DECW:   DEC     @R1
        CJNE    @R1,#0,DECW_EXIT
        CJNE    @R0,#0,DEC1
        AJMP    DECW_EXIT
DEC1:   DEC     @R0
DECW_EXIT:
        RET

;
;-------------------------------------------------------------------------
;  S E C T I O N    R E A D   D A T A
;-------------------------------------------------------------------------
;

;
;*************************************************************************
;
; Scan loop from sensor infrared
; Step :
;  1. turn on motor
;  2. wait until motor feed paper or time out (10 second)
;       receive value #1 from computer and send value #1 to computer
;     2.1 if receive value #2 - command stop by computer
;     2.2 if send value #2    - command stop by machine
;         because nopaper load
;  3. read data from sensor and computer
;     3.1 if computer send #1 - computer ready
;     3.2                 #2 - computer stop
;     3.3 if machine  send #1 - machine ready
;     3.4                 #2 - paper error
;     3.5                 #3 - machine start send data sensor
;
XSINF:
        LCALL   RCVD                    ; wait until computer read
        CJNE    A,#1,XSINFC0            ; Reg A = 1, ready
        SJMP    XSINF0                  ;       = 2, stop
                                        ;
XSINFC0:
        CJNE    A,#2,XSINF
        AJMP    XSINFEXIT

XSINF0:
        MOV     A,#3                    ; select motor 1 and 2
        MOV     R2,#1                   ; turn on motor
        LCALL   MOTOR

XSINF1:                                 ; loop again
        MOV     R2,#1                   ; send start data
        LCALL   SNDD

        LCALL   RCVD
        CJNE    A,#2,WAITPP1
        AJMP    XSINFEXIT
WAITPP1:
        LCALL   SINF                    ; read data
        MOV     A,BUFFER
        JNB     ACC.0,XSINF1            ; jump if position ready

RDSS:                                   ; Start read data from sensor
        MOV     B,#48                   ; 49 positions
        MOV     BUF1PTR,#HIGH BUFF1     ; initial buffer 1 pointer
        MOV     BUF1PTR+1,#LOW BUFF1
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
RDSS1:
        MOV     R2,#4                   ; program ready
        LCALL   SNDD

        LCALL   RCVD
        CJNE    A,#2,RDSS2
        AJMP    XSINFEXIT
RDSS2:

        LCALL   SINF                    ; read data
        MOV     A,BUFFER
        JNB     ACC.0,RDSS1             ; jump if position ready

RDSS3:                                  ; move data from internal memory to data
memory
        LCALL   MOV2DM                  ;

        MOV     R0,B
        CJNE    R0,#31,RDSS4
        MOV     A,#1
        MOV     R2,0
        LCALL   MOTOR
RDSS4:
        DJNZ    B,RDSS1
SDSS:                                   ; send data to computer

        MOV     R2,#3                   ; send start data
        LCALL   SNDD
        LCALL   RCVD                    ; ignore receive data

        LCALL   SNDXD                   ; call subroutine send data

        LJMP    XSINF

XSINFEXIT:
        MOV     A,#3                    ; select motor 1 and 2
        MOV     R2,#0                   ; turn off motor
        LCALL   MOTOR

        RET

;
;*************************************************************************
;
; Move data from internal memory to data memory
;
MOV2DM:

        PUSH    B
        MOV     R0,#BUFFER
        MOV     B,#4h
MOV2DM1:
        MOV     A,@R0
        MOV     DPH,BUF1PTR
        MOV     DPL,BUF1PTR+1
        MOVX    @DPTR,A

        INC     BUF1PTR+1
        INC     R0
        DJNZ    B,MOV2DM1
        POP     B

        RET


;
;*************************************************************************
;
; Scan loop from sensor infrared
;
SINF:

        MOV     R0,#BUFFER
        MOV     B,#0h
SINF1:
        MOV     A,B
        MOV     DPTR,#PORTC_2   ; Select port C2 for select D-FF
        MOVX    @DPTR,A
        SETB    ACC.3           ; Send clock to signal clock in 74LS374
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        MOVX    @DPTR,A
        MOV     R2,#00
        DJNZ    R2,$

        CLR     ACC.3           ; Clear clock
        MOVX    @DPTR,A


        MOV     DPTR,#PORTA_1   ; Read data to buffer
        MOVX    A,@DPTR
        MOV     @R0,A

        INC     R0
        INC     B
        MOV     R1,B
        CJNE    R1,#4,SINF1

        RET


;
;*****************************************************************
; Send data to computer from internal memory
;
SNDRD:
        MOV     R0,#BUFFER
        MOV     B,#08h

SNDRDL1:

        MOV     R1,#03h
        MOV     A,@R0
SNDRDL2:
        PUSH    ACC
        CJNE    R1,#01,CLR3BIT
        ANL     A,#03
        SJMP    SNDRDL2E
CLR3BIT:
        ANL     A,#07

SNDRDL2E:                       ; LOOP CHANGE CONDITION
        PUSH    REG_R1
        PUSH    REG_R0
        PUSH    B

        MOV     R2,A
        LCALL   SNDD

        POP     B
        POP     REG_R0
        POP     REG_R1
        POP     ACC
        RR      A
        RR      A
        RR      A

        DJNZ    REG_R1,SNDRDL2
        INC     R0
        DJNZ    B,SNDRDL1

        RET

;
;*****************************************************************
; Send data from data memory to computer
; Address data memory - 500h
;

SNDXD:
        MOV     DPTR,#BUFF1
        MOV     B,#47
SNDXDL0:
        MOV     R0,#4
SNDXDL1:
        PUSH    DPH
        PUSH    DPL

        MOV     R1,#03h
        MOVX    A,@DPTR
SNDXDL2:
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        PUSH    ACC
        CJNE    R1,#01,XCLR3BIT
        ANL     A,#03
        SJMP    SNDXDL2E
XCLR3BIT:
        ANL     A,#07
                                        ; LOOP CHANGE CONDITION
SNDXDL2E:
        PUSH    REG_R0
        PUSH    REG_R1
        PUSH    B

        MOV     R2,A
        LCALL   SNDD

        POP     B
        POP     REG_R1
        POP     REG_R0
        POP     ACC
        RR      A
        RR      A
        RR      A

        DJNZ    REG_R1,SNDXDL2
        POP     DPL
        POP     DPH
        INC     DPTR
        DJNZ    R0,SNDXDL1
        DJNZ    B,SNDXDL0

        RET
;
;*************************************************************************
;
; Motor 1 ON/OFF
; Input : Reg R2 = 0 motor off, 1 motor on
;         A      = 1 motor 1
;                  2 motor 2
;                  3 motor 1,2
;
MOTOR:
        PUSH    DPH
        PUSH    DPL
        MOV     DPTR,#PORTB_1           ; select port output
        CJNE    R2,#0,MOTORON           ; select motor mode (on or off)
                                        ; motor mode off
MOTOROFF:
        CJNE    A,#1,MOTOR2OFF          ; check motor 1 off
        MOV     A,#8
        SJMP    MOTORSND
MOTOR2OFF:
        CJNE    A,#2,MOTOR12OFF         ; check motor 2 off
        MOV     A,#4
        SJMP    MOTORSND
MOTOR12OFF:
        CLR     A                       ; check motor all off
        SJMP    MOTORSND
                                        ; motor mode on
MOTORON:
        CJNE    A,#1,MOTOR2ON           ; check motor 1 on
        MOV     A,#04
        SJMP    MOTORSND
MOTOR2ON:
        CJNE    A,#2,MOTOR12ON          ; check motor 2 on
        MOV     A,#08
        SJMP    MOTORSND
MOTOR12ON:
        MOV     A,#12                   ; check motor 1 and 2 on
                                        ; send data to D-FF
MOTORSND:
        MOVX    @DPTR,A
; send clock to D-FlipFlop
        SETB    ACC.4                   ; set bit A reg
        MOVX    @DPTR,A
        MOV     R2,#0
        DJNZ    R2,$
        CLR     ACC.4                   ; clear bit in A reg
        MOVX    @DPTR,A
        POP     DPL
        POP     DPH
        RET
        END
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โปรแกรม about.frm

```
VERSION 2.00
Begin Form About
   BackColor       =   &H00C0C0C0&
   BorderStyle     =   1  'Fixed Single
   Caption         =   "About Project"
   ClientHeight    =   6480
   ClientLeft      =   1695
   ClientTop       =   1005
   ClientWidth     =   5760
   Height          =   6885
   Left            =   1635
   LinkTopic       =   "Form2"
   MaxButton       =   0   'False
   MinButton       =   0   'False
   ScaleHeight     =   6480
   ScaleWidth      =   5760
   Top             =   660
   Width           =   5880
   Begin SSCommand Command3D1
      Caption         =   "&OK"
      Font3D          =   3  'Inset w/light shading
      Height          =   375
      Left            =   2400
      TabIndex        =   7
      Top             =   6000
      Width           =   975
   End
   Begin SSPanel Panel3D5
      BackColor       =   &H00C0C0C0&
      BevelOuter      =   0  'None
      Caption         =   "Instrument"
      Font3D          =   1  'Raised w/light shading
      ForeColor       =   &H00000000&
      Height          =   255
      Left            =   720
      TabIndex        =   2
      Top             =   4920
      Width           =   4335
   End
   Begin SSPanel Panel3D4
      BackColor       =   &H00C0C0C0&
      BevelOuter      =   0  'None
      Caption         =   "King Moungkut Institue of Technology '
Ladkrabang"
      Font3D          =   1  'Raised w/light shading
      ForeColor       =   &H00000000&
      Height          =   375
      Left            =   720
      TabIndex        =   4
      Top             =   5520
      Width           =   4455
   End
   Begin SSPanel Panel3D3
      BackColor       =   &H00C0C0C0&
      BevelOuter      =   0  'None
      Caption         =   "Faculty of Engineering"
      Font3D          =   1  'Raised w/light shading
      ForeColor       =   &H00000000&
      Height          =   255
      Left            =   600
      TabIndex        =   5
      Top             =   5280
      Width           =   4335
   End
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
  Begin SSPanel Panel3D6
     BackColor       =   &H00C0C0C0&
     BevelOuter      =   0  'None
     Caption         =   "KMIT'L"
     Font3D          =   1  'Raised w/light shading
     FontBold        =   -1  'True
     FontItalic      =   -1  'True
     FontName        =   "Times New Roman"
     FontSize        =   24
     FontStrikethru  =   0   'False
     FontUnderline   =   0   'False
     ForeColor       =   &H00000000&
     Height          =   615
     Left            =   720
     TabIndex        =   1
     Top             =   120
     Width           =   4335
  End
  Begin SSPanel Panel3D1
     BackColor       =   &H00C0C0C0&
     Font3D          =   1  'Raised w/light shading
     ForeColor       =   &H00000000&
     Height          =   3975
     Left            =   120
     TabIndex        =   0
     Top             =   855
     Width           =   5535
     Begin PictureBox Picture1
        BorderStyle      =   0  'None
        Height           =   3015
        Left             =   120
        Picture          =   ABOUT.FRX:0000
        ScaleHeight      =   3015
        ScaleWidth       =   5295
        TabIndex         =   6
        Top              =   120
        Width            =   5295
     End
     Begin SSPanel Panel3D2
        BackColor       =   &H00C0C0C0&
        BevelOuter      =   1  'Inset
        Caption         =   "CHECKER AUTO MACHINE V1.0"
        DragIcon        =   ABOUT.FRX:11E22
        Font3D          =   3  'Inset w/light shading
        ForeColor       =   &H00000000&
        Height          =   495
        Left            =   960
        TabIndex        =   3
        Top             =   3360
        Width           =   3855
     End
   End
End
Sub Command1_Click ()
    Unload about
End Sub

Sub Command3D1_Click ()
    Unload Me
End Sub

Sub Form_Load ()
    Left = (Screen.Width - Width) / 2
    Top = (Screen.Height - Height) / 2
End Sub
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โปรแกรม answer.frm

```
VERSION 2.00
Begin Form answer
   BackColor       =   &H00C0C0C0&
   BorderStyle     =   1  'Fixed Single
   Caption         =   "ANSWER LIST"
   ClientHeight    =   3195
   ClientLeft      =   1305
   ClientTop       =   3210
   ClientWidth     =   3570
   Height          =   3600
   Icon            =   ANSWER.FRX:0000
   Left            =   1245
   LinkTopic       =   "Form1"
   ScaleHeight     =   3195
   ScaleWidth      =   3570
   Top             =   2865
   Width           =   3690
   Begin SSCommand Command3D1
      Caption         =   "OK"
      Font3D          =   3  'Inset w/light shading
      Height          =   375
      Left            =   2520
      TabIndex        =   1
      Top             =   1320
      Width           =   855
   End
   Begin Grid Grid1
      BackColor       =   &H00FFFFFF&
      ForeColor       =   &H00000000&
      Height          =   2655
      Left            =   240
      Rows            =   10
      ScrollBars      =   2  'Vertical
      TabIndex        =   0
      Top             =   240
      Width           =   1935
   End
End

Sub Command3D1_Click ()
    Unload answer
End Sub

Sub Form_Load ()
    grid1.Row = 0
    grid1.Col = 0
    grid1.Text = "Choice"
    grid1.ColWidth(0) = 7 * Twips2ChrHor
    grid1.Col = 1
    grid1.Text = "Answer"
    grid1.ColWidth(1) = 7 * Twips2ChrHor

    grid1.Rows = answertbl.RecordCount

    mouseglass

    '
    ' Set flag for special handling of first row
    '
    firstRow = True
    colText = " "
    grid1.Rows = 1
    grid1.Cols = 2

    answertbl.MoveFirst
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    Do While answertbl.EOF = False
        rowText = " "
        For i = 0 To grid1.Cols - 1
            rowText = rowText + Format$(answertbl.Fields(i).Value, "0")

            If i < grid1.Cols - 1 Then
                rowText = rowText + Chr$(9)
             End If
        Next
        '
        ' Add new row to grid, the first non-fixed row has to
        ' be handled as a special case
        '
        grid1.AddItem rowText

        '
        ' Move to the next row
        '
        answertbl.MoveNext              ' same skip in DBase
    Loop
    grid1.FixedRows = 1
    mousenormal

End Sub
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โปรแกรม answermo.frm

```
VERSION 2.00
Begin Form answermode
   BackColor       =   &H00C0C0C0&
   Caption         =   "Answer setup"
   ClientHeight    =   3780
   ClientLeft      =   3570
   ClientTop       =   1755
   ClientWidth     =   4950
   Height          =   4185
   Left            =   3510
   LinkTopic       =   "Form1"
   ScaleHeight     =   3780
   ScaleWidth      =   4950
   Top             =   1410
   Width           =   5070
   Begin SSFrame Frame3D1
      Caption         =   "SELECT ANSWER METHOD"
      Font3D          =   3  'Inset w/light shading
      Height          =   3735
      Left            =   0
      TabIndex        =   0
      Top             =   0
      Width           =   4935
      Begin SSOption Option3D2
         Caption         =   "View answer"
         Font3D          =   1  'Raised w/light shading
         Height          =   375
         Left            =   960
         TabIndex        =   4
         Top             =   1200
         Width           =   3255
      End
      Begin SSCommand Command3D2
         Caption         =   "Close"
         Font3D          =   3  'Inset w/light shading
         Height          =   375
         Left            =   3000
         TabIndex        =   3
         Top             =   2640
         Width           =   975
      End
      Begin SSCommand Command3D1
         Caption         =   "OK"
         Font3D          =   3  'Inset w/light shading
         Height          =   375
         Left            =   960
         TabIndex        =   2
         Top             =   2640
         Width           =   1095
      End
      Begin SSOption Option3D1
         Caption         =   "Read data from answer paper"
         Font3D          =   1  'Raised w/light shading
         Height          =   375
         Left            =   960
         TabIndex        =   1
         Top             =   720
         Width           =   3015
      End
   End
End
Sub Command3D1_Click ()
    If option3d1.Value = True Then
        rdanswer.Show
    End If
    If option3d2.Value = True Then
        answer.Show modal
    End If
End Sub

Sub Command3D2_Click ()
    Unload answermode
End Sub
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โปรแกรม checksub.bas

```
Function decode_id () As String
Const start_row = 7
Dim temp As String
Dim tempdg As Integer
For i = 1 To IDLEN
    tempdg = -1
    x = 0
    For j = 0 To 9
        If i < 8 Then
            x = x + logicand(shiftbitr(buffer(start_row + j, 1), i), 1)
        Else
            x = x + logicand(shiftbitr(buffer(start_row + j, 2), i - 8), 1)
        End If
        If tempdg = -1 And x = 1 Then
            tempdg = j
        End If
    Next
    temp$ = temp$ + Format$(tempdg, "0")
Next
If x = 1 Then
    decode_id = temp$
Else
    decode_id = "ERR"
End If
End Function
Sub decodeanswer1 ()
    For i = 1 To 75
        Select Case answerbuf(i)
        Case 1
            answerbuf(i) = 1
        Case 2
            answerbuf(i) = 2
        Case 4
            answerbuf(i) = 3
        Case 8
            answerbuf(i) = 4
        Case 16
            answerbuf(i) = 5
        Case Else
            answerbuf(i) = 0
        End Select
    Next
End Sub
Sub decodeanswer2 ()
    For i = 76 To 150
        Select Case answerbuf(i)
        Case 1
            answerbuf(i) = 5
        Case 2
            answerbuf(i) = 4
        Case 4
            answerbuf(i) = 3
        Case 8
            answerbuf(i) = 2
        Case 16
            answerbuf(i) = 1
        Case Else
            answerbuf(i) = 0
        End Select
    Next
End Sub
Function result () As Integer
Dim x As Integer
    x = 0       ' store result
    answertbl.MoveFirst
    For i = 1 To CHOICES
        If answerbuf(i) = answertbl.Fields("choice").Value Then
            x = x + 1
        End If
        answertbl.MoveNext
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    Next
    result = x
End Function
' save answer to answer buffer
' startrow = row of choice from 1 - 47
' startcol = column of group choices 1 - 4
' chnummin = choice number min of column
' chnummax = choice number max of column
' bitpos   = pit position to shift
Sub savans2buf (ByVal startrow As Integer, ByVal startcol As Integer, ByVal
chnummin As Integer, ByVal chnummax As Integer, ByVal bitpos As Integer)
Dim xchoice
For i = 0 To chnummax - chnummin
    answerbuf(i + chnummin) = shiftbitr(buffer(startrow + i * 2, startcol),
bitpos)
Next
End Sub
Sub savans2buf4 ()
Dim xchoice, chnummax, chnummin
chnummax = 75
chnummin = 56
startrow = 7
For i = 0 To chnummax - chnummin
    answerbuf(i + chnummin) = shiftbitr(buffer(startrow + i * 2, 3), 7)
    answerbuf(i + chnummin) = answerbuf(i + chnummin) Or
shiftbitl(logicand(buffer(startrow + i * 2, 4), &HF), 1)
Next
End Sub
Sub savans2buf5 ()
Dim xchoice, chnummax, chnummin
chnummax = 95
chnummin = 76
startrow = 8
For i = 0 To chnummax - chnummin
    answerbuf(i + chnummin) = shiftbitr(buffer(startrow + i * 2, 3), 6)
    answerbuf(i + chnummin) = answerbuf(i + chnummin) Or
shiftbitl(logicand(buffer(startrow + i * 2, 4), &HF), 2)
Next
End Sub
Function savedecode () As String
    savans2buf 17, 1, 1, 15, 3
    savans2buf 7, 2, 16, 35, 3
    savans2buf 7, 3, 36, 55, 1
    savans2buf4
    savans2buf5
    savans2buf 8, 3, 96, 115, 0
    savans2buf 8, 2, 116, 135, 2
    savans2buf 18, 1, 136, 150, 3
    decodeanswer1
    decodeanswer2
    savedecode = decode_id()
End Function
Sub xrdanswer ()
    savans2buf 17, 1, 1, 15, 3
    savans2buf 7, 2, 16, 35, 3
    savans2buf 7, 3, 36, 55, 1
    savans2buf4
    savans2buf5
    savans2buf 8, 3, 96, 115, 0
    savans2buf 8, 2, 116, 135, 2
    savans2buf 18, 1, 136, 150, 3
    decodeanswer1
    decodeanswer2

    answertbl.MoveFirst
    For i = 1 To CHOICES
        answertbl.Edit
        answertbl.Fields("choice") = answerbuf(i)
        answertbl.Update
        answertbl.MoveNext
    Next
End Sub
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โปรแกรม cnt_test.frm

```
VERSION 2.00
Begin Form cnt_test
   BackColor       =   &H00C0C0C0&
   BorderStyle     =   3  'Fixed Double
   Caption         =   "Counter test"
   ClientHeight    =   1905
   ClientLeft      =   1095
   ClientTop       =   1485
   ClientWidth     =   4200
   Height          =   2310
   Left            =   1035
   LinkTopic       =   "Form1"
   ScaleHeight     =   1905
   ScaleWidth      =   4200
   Top             =   1140
   Width           =   4320
   Begin SSPanel Panel3D1
      BackColor       =   &H00C0C0C0&
      BevelInner      =   1  'Inset
      BevelOuter      =   0  'None
      BorderWidth     =   0
      Caption         =   "Look at counter"
      Font3D          =   1  'Raised w/light shading
      ForeColor       =   &H00FF0000&
      Height          =   615
      Left            =   600
      TabIndex        =   2
      Top             =   240
      Width           =   2895
   End
   Begin SSCommand Start
      Caption         =   "Start"
      Font3D          =   3  'Inset w/light shading
      Height          =   375
      Left            =   600
      TabIndex        =   1
      Top             =   1320
      Width           =   975
   End
   Begin SSCommand Closecmd
      Caption         =   "Close"
      Font3D          =   3  'Inset w/light shading
      Height          =   375
      Left            =   2400
      TabIndex        =   0
      Top             =   1320
      Width           =   1095
   End
End
Sub Start_Click ()
If start.Caption = "Start" Then
    closecmd.Enabled = False
    start.Caption = "Stop"
    qflg = False
    If port_init(True) = False Then
        Beep
        MsgBox "Checker machine not connect", 48, APPNAME$
        Exit Sub
    End If
    senddata (4)
Else
    qflg = True
    start.Caption = "Start"
    closecmd.Enabled = True
End If
Do While qflg <> True
    DoEvents
Loop
senddata (2)
End Sub
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โปรแกรม comm.frm

```
VERSION 2.00
Begin Form comm
   BackColor       =   &H00C0C0C0&
   BorderStyle     =   1  'Fixed Single
   Caption         =   "Communication Program"
   ClientHeight    =   3420
   ClientLeft      =   2400
   ClientTop       =   2730
   ClientWidth     =   6225
   ControlBox      =   0   'False
   Height          =   3825
   Left            =   2340
   LinkTopic       =   "Form1"
   MaxButton       =   0   'False
   MDIChild        =   -1  'True
   MinButton       =   0   'False
   ScaleHeight     =   3420
   ScaleWidth      =   6225
   Top             =   2385
   Width           =   6345
   Begin SSPanel sheetcnt
      BackColor       =   &H00C0C0C0&
      BorderWidth     =   0
      Caption         =   "Answer Sheet #"
      Font3D          =   0  'None
      ForeColor       =   &H00000000&
      Height          =   375
      Left            =   240
      TabIndex        =   5
      Top             =   3000
      Width           =   5775
   End
   Begin SSFrame Frame3D1
      Caption         =   "Answer Control"
      Font3D          =   0  'None
      Height          =   2655
      Left            =   240
      TabIndex        =   2
      Top             =   120
      Width           =   4095
      Begin SSOption contoption
         Caption         =   "Continue read "
         Font3D          =   3  'Inset w/light shading
         ForeColor       =   &H00C00000&
         Height          =   255
         Left            =   600
         TabIndex        =   4
         Top             =   1560
         Width           =   2775
      End
      Begin SSOption newoption
         Caption         =   "New read student answer"
         Font3D          =   3  'Inset w/light shading
         ForeColor       =   &H00C00000&
         Height          =   255
         Left            =   600
         TabIndex        =   3
         Top             =   840
         Width           =   3015
      End
   End
   Begin SSCommand Stopcmd
      Caption         =   "Stop"
      Font3D          =   3  'Inset w/light shading
      Height          =   375
      Left            =   4905
      TabIndex        =   1
      Top             =   1800
      Width           =   1110
   End
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
   Begin SSCommand start
      Caption         =   "Start"
      Font3D          =   3  'Inset w/light shading
      Height          =   375
      Left            =   4920
      TabIndex        =   0
      Top             =   1080
      Width           =   1110
   End
End
Dim qflg As Integer, xsheetcnt%

Sub Build_Click ()
    If newoption.value = True Then
    End If
    If contoption.value = True Then
    End If
End Sub

Sub Form_Load ()
    qflg = False
    xsheetcnt = 0
    ref_option
    ref_cnt
End Sub

Sub ref_cnt ()
    sheetcnt.Caption = "Answer Sheet#" + Str(xsheetcnt)
End Sub

Sub ref_option ()
    If Dir$(app.Path + "\xxx.txt") = "" Then
        newoption.value = True
        contoption.value = False
        contoption.Enabled = False
    Else
        newoption.value = False
        contoption.value = True
        contoption.Enabled = True
    End If
End Sub

Sub start_Click ()
Dim SAVDATA, xdata, tinstant
    If port_init(True) = False Then
        Beep
        MsgBox "Checker machine not connect", 48, APPNAME$
        Exit Sub
    End If
    senddata (7)
    stopcmd.Enabled = True
    start.Enabled = False
    qflg = False

    If newoption.value = True Then
        xsheetcnt = 0          ' sheet count
        ErrorId = 0            ' error count
    End If

    xtimeout = False           ' flag time out feed
    tinstant = Now             ' initial feed time start
    senddata (1)               ' start code
    Do While qflg = False
        xrecvdata = recvdata()
        senddata (1)
        Select Case xrecvdata
        Case 1
        Case 2
            Beep
            MsgBox "Error Stop", 48, APPNAME$
            qflg = True
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        Case 3
            For i = 1 To 47
                For j = 0 To 3
                    x = 0
                    For k = 0 To 2
                        x = x + recvdata() * (2 ^ (3 * k))
                    Next
                    buffer(i, j + 1) = x
                Next
            Next
            xsheetcnt = xsheetcnt + 1
            ref_cnt
            ' decode student id and score
            x = savedecode()
            If x <> "ERR" Then
                Mytbl.Seek "=", x
                If Mytbl.NoMatch Then
                    ErrorId = ErrorId + 1
                    Beep
                    MsgBox "Student Id. not found", MB_ICONINFORMATION
                Else
                    Mytbl.Edit
                    Mytbl.Fields("score1") = result()
                    Mytbl.Update
                End If
            Else
                ErrorId = ErrorId + 1
                Beep
                MsgBox "Student Id. ERROR", MB_ICONINFORMATION
            End If
            tinstant = Now  ' reset time out feed
            sendcode (1)
        Case 4              ' machine start read
        End Select
        If Difftime(tinstant) > TIMEOUTFD Then
            xtimeout = True
            qflg = True
        End If
        DoEvents
    Loop
    xrecvdata = recvdata()
    senddata (2)
    Beep
    MsgBox "Feed Paper time out", MB_ICONINFORMATION

    stopcmd.Enabled = False
    start.Enabled = True

End Sub

Sub stopcmd_Click ()
    qflg = True
End Sub

Sub xView_click ()
    xxshow.Show
End Sub
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โปรแกรม gphview.frm

```
VERSION 2.00
Begin Form GphView
   BackColor          =   &H00C0C0C0&
   Caption            =   "Graph View"
   ClientHeight       =   6660
   ClientLeft         =   525
   ClientTop          =   2010
   ClientWidth        =   11070
   Height             =   7065
   Icon               =   GPHVIEW.FRX:0000
   Left               =   465
   LinkTopic          =   "Form1"
   MaxButton          =   0   'False
   MDIChild           =   -1  'True
   ScaleHeight        =   6660
   ScaleWidth         =   11070
   Top                =   1665
   Width              =   11190
   Begin CommandButton CmdClose
      Caption            =   "&Close"
      Height             =   375
      Left               =   4920
      TabIndex           =   1
      Top                =   6120
      Width              =   1335
   End
   Begin SSPanel Panel3D1
      AutoSize           =   3  'AutoSize Child To Panel
      BackColor          =   &H00C0C0C0&
      BevelOuter         =   1  'Inset
      BorderWidth        =   4
      Font3D             =   1  'Raised w/light shading
      ForeColor          =   &H00FF0000&
      Height             =   5835
      Left               =   225
      TabIndex           =   0
      Top                =   180
      Width              =   10635
      Begin GRAPH Graph1
         GraphType          =   6  'Line
         Height             =   5805
         Labels             =   3  'Y Labels only
         Left               =   15
         NumSets            =   3
         TabIndex           =   2
         ThickLines         =   0  'Off
         Ticks              =   0  'Off
         Top                =   15
         Width              =   10605
      End
   End
End
Sub CmdClose_Click ()
    Unload GphView
End Sub

Sub Form_Load ()
    ' Initial form size and position
    GphView.Height = 6975
    GphView.Width = 11190
    GphView.Top = 0
    GphView.Left = 0

    '
    ' This could take a while, display hourglass
    '
    Screen.MousePointer = 11
    ' Create data graph grid
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    Graph1.NumSets = 3
    Graph1.NumPoints = MyTbl.RecordCount

    ' Round and create Score 1
    Graph1.ThisSet = 1
    Graph1.ThisPoint = 1
    MyTbl.MoveFirst          ' go top of Table
    Do While MyTbl.EOF = False
        ' Add data to graph
        Graph1.GraphData = MyTbl.Fields("score1").Value
        '
        ' Move to the next row
        '
        MyTbl.MoveNext           ' same skip in DBase
    Loop

    ' Round and create Score 2
    Graph1.ThisSet = 2
    Graph1.ThisPoint = 1
    MyTbl.MoveFirst          ' go top of Table
    Do While MyTbl.EOF = False
        ' Add data to graph
        Graph1.GraphData = MyTbl.Fields("score2").Value
        '
        ' Move to the next row
        '
        MyTbl.MoveNext           ' same skip in DBase
    Loop

    ' Round and create Score 3
    Graph1.ThisSet = 3
    Graph1.ThisPoint = 1
    MyTbl.MoveFirst          ' go top of Table
    Do While MyTbl.EOF = False
        ' Add data to graph
        Graph1.GraphData = MyTbl.Fields("score3").Value
        '
        ' Move to the next row
        '
        MyTbl.MoveNext           ' same skip in DBase
    Loop

    '
    ' Return the mouse pointer to normal
    '
    Screen.MousePointer = 0

End Sub
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โปรแกรม inf_test.frm

```
VERSION 2.00
Begin Form inf_test
   BackColor       =   &H00C0C0C0&
   BorderStyle     =   3  'Fixed Double
   Caption         =   "Infrarade test"
   ClientHeight    =   2280
   ClientLeft      =   1590
   ClientTop       =   1350
   ClientWidth     =   5625
   Height          =   2685
   Left            =   1530
   LinkTopic       =   "Form4"
   ScaleHeight     =   2280
   ScaleWidth      =   5625
   Top             =   1005
   Width           =   5745
   Begin SSPanel Panel3D3
      BackColor       =   &H00C0C0C0&
      BevelOuter      =   1  'Inset
      BorderWidth     =   0
      Caption         =
"1...........5...........10...........15...........20...........25.....
.....30"
      Font3D          =   3  'Inset w/light shading
      ForeColor       =   &H00FF0000&
      Height          =   255
      Left            =   60
      TabIndex        =   4
      Top             =   1200
      Width           =   5475
   End
   Begin CommandButton closecmd
      Caption         =   "&Close"
      Height          =   375
      Left            =   3360
      TabIndex        =   3
      Top             =   1680
      Width           =   1095
   End
   Begin SSPanel Panel3D2
      BackColor       =   &H00C0C0C0&
      BevelInner      =   2  'Raised
      BevelOuter      =   1  'Inset
      BorderWidth     =   0
      Caption         =   "Sensor Board (1-30)"
      Font3D          =   3  'Inset w/light shading
      ForeColor       =   &H00FF0000&
      Height          =   675
      Index           =   0
      Left            =   60
      TabIndex        =   2
      Top             =   120
      Width           =   5475
   End
   Begin CommandButton StartStop
      Caption         =   "Start"
      Height          =   375
      Left            =   1200
      TabIndex        =   1
      Top             =   1680
      Width           =   1095
   End
   Begin SSPanel Panel3D1
      BackColor       =   &H00000000&
      BevelOuter      =   1  'Inset
      BorderWidth     =   0
      Font3D          =   3  'Inset w/light shading
      ForeColor       =   &H00FF0000&
      Height          =   255
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
Left            =   60
TabIndex        =   0
Top             =   840
Width           =   5475
Begin Image inf
   Height           =   120
   Index            =   58
   Left             =   5280
   Picture          =   INF_TEST.FRX:0000
   Top              =   60
   Width            =   120
End
Begin Image inf
   Height           =   120
   Index            =   56
   Left             =   5100
   Picture          =   INF_TEST.FRX:009A
   Top              =   60
   Width            =   120
End
Begin Image inf
   Height           =   120
   Index            =   54
   Left             =   4920
   Picture          =   INF_TEST.FRX:0134
   Top              =   60
   Width            =   120
End
Begin Image inf
   Height           =   120
   Index            =   52
   Left             =   4740
   Picture          =   INF_TEST.FRX:01CE
   Top              =   60
   Width            =   120
End
Begin Image inf
   Height           =   120
   Index            =   50
   Left             =   4560
   Picture          =   INF_TEST.FRX:0268
   Top              =   60
   Width            =   120
End
Begin Image inf
   Height           =   120
   Index            =   48
   Left             =   4380
   Picture          =   INF_TEST.FRX:0302
   Top              =   60
   Width            =   120
End
Begin Image inf
   Height           =   120
   Index            =   46
   Left             =   4200
   Picture          =   INF_TEST.FRX:039C
   Top              =   60
   Width            =   120
End
Begin Image inf
   Height           =   120
   Index            =   44
   Left             =   4020
   Picture          =   INF_TEST.FRX:0436
   Top              =   60
   Width            =   120
End
Begin Image inf
   Height           =   120
   Index            =   42
   Left             =   3840
```

เอกสารนี้เป็นเอกสารสงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
   Picture          =   INF_TEST.FRX:04D0
   Top              =   60
   Width            =   120
End
Begin Image inf
   Height           =   120
   Index            =   40
   Left             =   3660
   Picture          =   INF_TEST.FRX:056A
   Top              =   60
   Width            =   120
End
Begin Image inf
   Height           =   120
   Index            =   38
   Left             =   3480
   Picture          =   INF_TEST.FRX:0604
   Top              =   60
   Width            =   120
End
Begin Image inf
   Height           =   120
   Index            =   36
   Left             =   3300
   Picture          =   INF_TEST.FRX:069E
   Top              =   60
   Width            =   120
End
Begin Image inf
   Height           =   120
   Index            =   34
   Left             =   3120
   Picture          =   INF_TEST.FRX:0738
   Top              =   60
   Width            =   120
End
Begin Image inf
   Height           =   120
   Index            =   32
   Left             =   2940
   Picture          =   INF_TEST.FRX:07D2
   Top              =   60
   Width            =   120
End
Begin Image inf
   Height           =   120
   Index            =   30
   Left             =   2760
   Picture          =   INF_TEST.FRX:086C
   Top              =   60
   Width            =   120
End
Begin Image inf
   Height           =   120
   Index            =   28
   Left             =   2580
   Picture          =   INF_TEST.FRX:0906
   Top              =   60
   Width            =   120
End
Begin Image inf
   Height           =   120
   Index            =   26
   Left             =   2400
   Picture          =   INF_TEST.FRX:09A0
   Top              =   60
   Width            =   120
End
Begin Image inf
   Height           =   120
   Index            =   24
   Left             =   2220
```

เอกสารนี้เป็นเอกสารสงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
   Picture          =   INF_TEST.FRX:0A3A
   Top              =   60
   Width            =   120
End
Begin Image inf
   Height           =   120
   Index            =   22
   Left             =   2040
   Picture          =   INF_TEST.FRX:0AD4
   Top              =   60
   Width            =   120
End
Begin Image inf
   Height           =   120
   Index            =   20
   Left             =   1860
   Picture          =   INF_TEST.FRX:0B6E
   Top              =   60
   Width            =   120
End
Begin Image inf
   Height           =   120
   Index            =   18
   Left             =   1680
   Picture          =   INF_TEST.FRX:0C08
   Top              =   60
   Width            =   120
End
Begin Image inf
   Height           =   120
   Index            =   16
   Left             =   1500
   Picture          =   INF_TEST.FRX:0CA2
   Top              =   60
   Width            =   120
End
Begin Image inf
   Height           =   120
   Index            =   14
   Left             =   1320
   Picture          =   INF_TEST.FRX:0D3C
   Top              =   60
   Width            =   120
End
Begin Image inf
   Height           =   120
   Index            =   12
   Left             =   1140
   Picture          =   INF_TEST.FRX:0DD6
   Top              =   60
   Width            =   120
End
Begin Image inf
   Height           =   120
   Index            =   10
   Left             =   960
   Picture          =   INF_TEST.FRX:0E70
   Top              =   60
   Width            =   120
End
Begin Image inf
   Height           =   120
   Index            =   6
   Left             =   600
   Picture          =   INF_TEST.FRX:0F0A
   Top              =   60
   Width            =   120
End
Begin Image inf
   Height           =   120
   Index            =   59
   Left             =   5280
```

เอกสารนี้เป็นเอกสารสงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
   Picture          =   INF_TEST.FRX:0FA4
   Top              =   60
   Width            =   120
End
Begin Image inf
   Height           =   120
   Index            =   57
   Left             =   5100
   Picture          =   INF_TEST.FRX:103E
   Top              =   60
   Width            =   120
End
Begin Image inf
   Height           =   120
   Index            =   55
   Left             =   4920
   Picture          =   INF_TEST.FRX:10D8
   Top              =   60
   Width            =   120
End
Begin Image inf
   Height           =   120
   Index            =   53
   Left             =   4740
   Picture          =   INF_TEST.FRX:1172
   Top              =   60
   Width            =   120
End
Begin Image inf
   Height           =   120
   Index            =   51
   Left             =   4560
   Picture          =   INF_TEST.FRX:120C
   Top              =   60
   Width            =   120
End
Begin Image inf
   Height           =   120
   Index            =   49
   Left             =   4380
   Picture          =   INF_TEST.FRX:12A6
   Top              =   60
   Width            =   120
End
Begin Image inf
   Height           =   120
   Index            =   47
   Left             =   4200
   Picture          =   INF_TEST.FRX:1340
   Top              =   60
   Width            =   120
End
Begin Image inf
   Height           =   120
   Index            =   45
   Left             =   4020
   Picture          =   INF_TEST.FRX:13DA
   Top              =   60
   Width            =   120
End
Begin Image inf
   Height           =   120
   Index            =   43
   Left             =   3840
   Picture          =   INF_TEST.FRX:1474
   Top              =   60
   Width            =   120
End
Begin Image inf
   Height           =   120
   Index            =   41
   Left             =   3660
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
   Picture          =   INF_TEST.FRX:150E
   Top              =   60
   Width            =   120
End
Begin Image inf
   Height           =   120
   Index            =   39
   Left             =   3480
   Picture          =   INF_TEST.FRX:15A8
   Top              =   60
   Width            =   120
End
Begin Image inf
   Height           =   120
   Index            =   37
   Left             =   3300
   Picture          =   INF_TEST.FRX:1642
   Top              =   60
   Width            =   120
End
Begin Image inf
   Height           =   120
   Index            =   35
   Left             =   3120
   Picture          =   INF_TEST.FRX:16DC
   Top              =   60
   Width            =   120
End
Begin Image inf
   Height           =   120
   Index            =   33
   Left             =   2940
   Picture          =   INF_TEST.FRX:1776
   Top              =   60
   Width            =   120
End
Begin Image inf
   Height           =   120
   Index            =   31
   Left             =   2760
   Picture          =   INF_TEST.FRX:1810
   Top              =   60
   Width            =   120
End
Begin Image inf
   Height           =   120
   Index            =   29
   Left             =   2580
   Picture          =   INF_TEST.FRX:18AA
   Top              =   60
   Width            =   120
End
Begin Image inf
   Height           =   120
   Index            =   27
   Left             =   2400
   Picture          =   INF_TEST.FRX:1944
   Top              =   60
   Width            =   120
End
Begin Image inf
   Height           =   120
   Index            =   25
   Left             =   2220
   Picture          =   INF_TEST.FRX:19DE
   Top              =   60
   Width            =   120
End
Begin Image inf
   Height           =   120
   Index            =   23
   Left             =   2040
```

เอกสารนี้เป็นเอกสารสงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
   Picture          =   INF_TEST.FRX:1A78
   Top              =   60
   Width            =   120
End
Begin Image inf
   Height           =   120
   Index            =   21
   Left             =   1860
   Picture          =   INF_TEST.FRX:1B12
   Top              =   60
   Width            =   120
End
Begin Image inf
   Height           =   120
   Index            =   19
   Left             =   1680
   Picture          =   INF_TEST.FRX:1BAC
   Top              =   60
   Width            =   120
End
Begin Image inf
   Height           =   120
   Index            =   17
   Left             =   1500
   Picture          =   INF_TEST.FRX:1C46
   Top              =   60
   Width            =   120
End
Begin Image inf
   Height           =   120
   Index            =   15
   Left             =   1320
   Picture          =   INF_TEST.FRX:1CE0
   Top              =   60
   Width            =   120
End
Begin Image inf
   Height           =   120
   Index            =   13
   Left             =   1140
   Picture          =   INF_TEST.FRX:1D7A
   Top              =   60
   Width            =   120
End
Begin Image inf
   Height           =   120
   Index            =   11
   Left             =   960
   Picture          =   INF_TEST.FRX:1E14
   Top              =   60
   Width            =   120
End
Begin Image inf
   Height           =   120
   Index            =   7
   Left             =   600
   Picture          =   INF_TEST.FRX:1EAE
   Top              =   60
   Width            =   120
End
Begin Image inf
   Height           =   120
   Index            =   0
   Left             =   60
   Picture          =   INF_TEST.FRX:1F48
   Top              =   60
   Width            =   120
End
Begin Image inf
   Height           =   120
   Index            =   1
   Left             =   60
```

เอกสารนี้เป็นเอกสารสงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
      Picture         =   INF_TEST.FRX:1FE2
      Top             =   60
      Width           =   120
   End
   Begin Image inf
      Height          =   120
      Index           =   2
      Left            =   240
      Picture         =   INF_TEST.FRX:207C
      Top             =   60
      Width           =   120
   End
   Begin Image inf
      Height          =   120
      Index           =   3
      Left            =   240
      Picture         =   INF_TEST.FRX:2116
      Top             =   60
      Width           =   120
   End
   Begin Image inf
      Height          =   120
      Index           =   8
      Left            =   780
      Picture         =   INF_TEST.FRX:21B0
      Top             =   60
      Width           =   120
   End
   Begin Image inf
      Height          =   120
      Index           =   9
      Left            =   780
      Picture         =   INF_TEST.FRX:224A
      Top             =   60
      Width           =   120
   End
   Begin Image inf
      Height          =   120
      Index           =   4
      Left            =   420
      Picture         =   INF_TEST.FRX:22E4
      Top             =   60
      Width           =   120
   End
   Begin Image inf
      Height          =   120
      Index           =   5
      Left            =   420
      Picture         =   INF_TEST.FRX:237E
      Top             =   60
      Width           =   120
   End
 End
End
Dim qflg

Sub close_Click ()
    Unload inf_test
End Sub

Sub closecmd_Click ()
    Unload Me
End Sub

Sub Command1_Click ()
End Sub

Sub Form_Load ()
    qflg = False
End Sub
Sub inf1dsp (xValue As Double)
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    Dim i As Integer, OnFlg As Double, twodbl As Double
    twodbl = 2
    For i = 0 To 29
        OnFlg = xValue And twodbl ^ i
        If OnFlg = 2 ^ i Then
            inf(i * 2).ZOrder 1
            inf(i * 2 + 1).ZOrder
        Else
            inf(i * 2).ZOrder
            inf(i * 2 + 1).ZOrder 1
        End If
    Next
End Sub

Sub StartStop_Click ()
Dim x As Double, tmp As Integer
Dim xbuffer1() As Integer
    ReDim xbuffer1(0 To 3)
    If startstop.Caption = "Stop" Then
        qflg = True
        closecmd.Enabled = True
        Exit Sub
    End If
    startstop.Caption = "Stop"
    closecmd.Enabled = False
    Do Until qflg
        For i = 0 To 3
            xbuffer1(i) = 0
        Next

        If port_init(True) = False Then
            Beep
            MsgBox "Checker machine not connect", 48, APPNAME$
            Exit Sub
        End If
        senddata (3)
        For i = 0 To 3
            x = 0
            For j = 0 To 2
                If i = 3 And j = 2 Then
                    ' ignore 2 bit for overflow
                    tmp = recvdata()
                    Exit For
                End If
                x = x + recvdata() * (2 ^ (3 * j))
            Next
            xbuffer1(i) = x
        Next
        x1 = 0
        For i = 0 To 3
            x1 = x1 + xbuffer1(i) * (2 ^ (8 * i))
        Next

        inf1dsp (x1)
        DoEvents
    Loop

End Sub
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โปรแกรม lookup.frm

```
VERSION 2.00
Begin Form LookUp
   BorderStyle     =   1  'Fixed Single
   Caption         =   "Data List"
   ClientHeight    =   3360
   ClientLeft      =   315
   ClientTop       =   3930
   ClientWidth     =   11475
   Height          =   3765
   Icon            =   LOOKUP.FRX:0000
   Left            =   255
   LinkTopic       =   "Form1"
   MaxButton       =   0   'False
   MDIChild        =   -1  'True
   ScaleHeight     =   3360
   ScaleWidth      =   11475
   Top             =   3585
   Width           =   11595
   Begin CommandButton Exit
      Caption         =   "Go Back"
      Height          =   375
      Left            =   5160
      TabIndex        =   1
      Top             =   2745
      Width           =   1005
   End
   Begin Grid List
      BackColor       =   &H00FFFFFF&
      FixedCols       =   0
      ForeColor       =   &H00000000&
      Height          =   2430
      Left            =   225
      Rows            =   10
      TabIndex        =   0
      Top             =   180
      Width           =   11020
   End
End
Sub Exit_Click ()
    Unload Lookup
End Sub

Sub Form_Load ()
    Dim i As Integer
    Dim fieldname   As String
    Dim rowText As String
    Dim firstRow As Integer

    ' Initial top and left position
    Lookup.Top = 0
    Lookup.Left = 0

    '
    ' This could take a while, display hourglass
    '
    Screen.MousePointer = 11

    MyTbl.MoveFirst          ' go top of Table

    List.Rows = 2
    List.Cols = MyTbl.Fields.Count
    List.Row = 0
    For i = 0 To List.Cols - 1
        List.Col = i
        fieldname = MyTbl.Fields(i).Name

        Select Case i
            Case 0
                List.Text = "Code"
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
            Case 1
                List.Text = "Name"
            Case 2
                List.Text = "Score 1st"
            Case 3
                List.Text = "Score 2rd"
            Case 4
                List.Text = "Score 3nd"
        End Select
        Set MyFld = MyTbl(fieldname)
        List.ColWidth(i) = MyFld.Size * Twips2ChrHor
    Next

    '
    ' Set flag for special handling of first row
    '
    firstRow = True
    colText = " "
    List.Rows = 2

    Do While MyTbl.EOF = False
        rowText = " "
        For i = 0 To List.Cols - 1
            Select Case MyTbl.Fields(i).Type
                Case DB_TEXT
                    rowText = rowText + MyTbl.Fields(i).Value
                Case DB_DOUBLE
                    rowText = rowText + Str$(MyTbl.Fields(i).Value)
            End Select

            If i < List.Cols - 1 Then
                rowText = rowText + Chr$(9)
            End If
        Next
        '
        ' Add new row to grid, the first non-fixed row has to
        ' be handled as a special case
        '
        If (firstRow = True) Then
            List.SelStartRow = 1
            List.SelEndRow = 1
            List.SelStartCol = 0
            List.SelEndCol = List.Cols - 1
            List.Clip = rowText
            firstRow = False
        Else
            List.AddItem rowText
        End If
        '
        ' Move to the next row
        '
        MyTbl.MoveNext              ' same skip in DBase
    Loop

    '
    ' Return the mouse pointer to normal
    '
    Screen.MousePointer = 0

End Sub
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โปรแกรม options.frm

```
VERSION 2.00
Begin Form Options
   BackColor       =   &H00C0C0C0&
   BorderStyle     =   3  'Fixed Double
   Caption         =   "Options"
   ClientHeight    =   3255
   ClientLeft      =   1320
   ClientTop       =   2190
   ClientWidth     =   4830
   Height          =   3660
   Left            =   1260
   LinkTopic       =   "Form2"
   MaxButton       =   0   'False
   MinButton       =   0   'False
   ScaleHeight     =   3255
   ScaleWidth      =   4830
   Top             =   1845
   Width           =   4950
   Begin TextBox Text4
      Height          =   315
      Left            =   2880
      TabIndex        =   9
      Top             =   1680
      Width           =   1095
   End
   Begin SSPanel Panel3D4
      Alignment       =   1  'Left Justify - MIDDLE
      BackColor       =   &H00C0C0C0&
      BevelOuter      =   0  'None
      BorderWidth     =   0
      Caption         =   "Time out feed :"
      Font3D          =   1  'Raised w/light shading
      ForeColor       =   &H00FF0000&
      Height          =   255
      Left            =   360
      TabIndex        =   8
      Top             =   1680
      Width           =   2055
   End
   Begin TextBox Text3
      Height          =   315
      Left            =   2880
      TabIndex        =   7
      Top             =   1200
      Width           =   1095
   End
   Begin SSPanel Panel3D3
      Alignment       =   1  'Left Justify - MIDDLE
      BackColor       =   &H00C0C0C0&
      BevelOuter      =   0  'None
      BorderWidth     =   0
      Caption         =   "Student Length :"
      Font3D          =   1  'Raised w/light shading
      ForeColor       =   &H00FF0000&
      Height          =   255
      Left            =   360
      TabIndex        =   6
      Top             =   1200
      Width           =   2055
   End
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
Begin TextBox Text2
   Height          =   315
   Left            =   2880
   TabIndex        =   5
   Top             =   720
   Width           =   1095
End
Begin TextBox Text1
   Height          =   315
   Left            =   2880
   TabIndex        =   4
   Top             =   240
   Width           =   1095
End
Begin SSPanel Panel3D2
   Alignment       =   1  'Left Justify - MIDDLE
   BackColor       =   &H00C0C0C0&
   BevelOuter      =   0  'None
   BorderWidth     =   0
   Caption         =   "Amount of choice :"
   Font3D          =   1  'Raised w/light shading
   ForeColor       =   &H00FF0000&
   Height          =   255
   Left            =   360
   TabIndex        =   3
   Top             =   240
   Width           =   2055
End
Begin SSPanel Panel3D1
   Alignment       =   1  'Left Justify - MIDDLE
   BackColor       =   &H00C0C0C0&
   BevelOuter      =   0  'None
   BorderWidth     =   0
   Caption         =   "Time out connect :"
   Font3D          =   1  'Raised w/light shading
   ForeColor       =   &H00FF0000&
   Height          =   255
   Left            =   360
   TabIndex        =   2
   Top             =   720
   Width           =   2055
End
Begin SSCommand Command3D1
   Caption         =   "OK"
   Font3D          =   3  'Inset w/light shading
   Height          =   375
   Left            =   1200
   TabIndex        =   1
   Top             =   2640
   Width           =   915
End
Begin SSCommand Command3D2
   Caption         =   "Cancel"
   Font3D          =   3  'Inset w/light shading
   Height          =   375
   Left            =   2700
   TabIndex        =   0
   Top             =   2640
   Width           =   915
End
End
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
Sub Command3D1_Click ()
    CHOICES = Val(TEXT1.TEXT)
    TIMEOUT = Val(TEXT2.TEXT)
    IDLEN = Val(TEXT3.TEXT)
    TIMEOUTFD = Val(TEXT4.TEXT)
    XCHOICE$ = TEXT1.TEXT
    XTIMEOUT$ = TEXT2.TEXT
    XIDLEN$ = TEXT3.TEXT
    XTIMEOUTFD$ = TEXT4.TEXT
    i = WriteProfileString(APPNAME$, "CHOICES", XCHOICE$)
    i = WriteProfileString(APPNAME$, "TIMEOUT", XTIMEOUT$)
    i = WriteProfileString(APPNAME$, "IDLEN", XIDLEN$)
    i = WriteProfileString(APPNAME$, "TIMEOUTFD", XTIMEOUTFD$)
Unload options
End Sub

Sub Command3D2_Click ()
    Unload options
End Sub

Sub Form_Load ()
    TEXT1.TEXT = Str(CHOICES)          ' Amount of choices
    TEXT2.TEXT = Str(TIMEOUT)          ' Time out connect
    TEXT3.TEXT = Str(IDLEN)            ' Student id length
    TEXT4.TEXT = Str(TIMEOUTFD)        ' Time out feed
End Sub
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โปรแกรม rdanswer.frm

```
VERSION 2.00
Begin Form rdanswer
   BackColor       =   &H00C0C0C0&
   Caption         =   "Rdanswer"
   ClientHeight    =   2055
   ClientLeft      =   810
   ClientTop       =   1470
   ClientWidth     =   4785
   Height          =   2460
   Left            =   750
   LinkTopic       =   "Form1"
   ScaleHeight     =   2055
   ScaleWidth      =   4785
   Top             =   1125
   Width           =   4905
   Begin SSPanel Panel3D1
      BackColor       =   &H00C0C0C0&
      BorderWidth     =   0
      Caption         =   "Read answer to system"
      Font3D          =   0  'None
      ForeColor       =   &H00000000&
      Height          =   495
      Left            =   840
      TabIndex        =   2
      Top             =   360
      Width           =   3015
   End
   Begin SSCommand start
      Caption         =   "Start"
      Font3D          =   3  'Inset w/light shading
      Height          =   375
      Left            =   840
      TabIndex        =   1
      Top             =   1440
      Width           =   1110
   End
   Begin SSCommand Stopcmd
      Caption         =   "Stop"
      Font3D          =   3  'Inset w/light shading
      Height          =   375
      Left            =   2760
      TabIndex        =   0
      Top             =   1440
      Width           =   1110
   End
   Begin Line Line2
      BorderColor     =   &H00FFFFFF&
      X1              =   240
      X2              =   4440
      Y1              =   1090
      Y2              =   1090
   End
   Begin Line Line1
      X1              =   240
      X2              =   4440
      Y1              =   1080
      Y2              =   1080
   End
End
Dim qflg As Integer

Sub Form_Load ()
    qflg = False
End Sub
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
Sub start_Click ()
Dim SAVDATA, xdata, tinstant
    If port_init(True) = False Then
        Beep
        MsgBox "Checker machine not connect", 48, APPNAME$
        Exit Sub
    End If
    senddata (7)
    stopcmd.Enabled = True
    start.Enabled = False
    qflg = False

    xtimeout = False          ' flag time out feed
    tinstant = Now            ' initial feed time start
    senddata (1)              ' start code
    Do While qflg = False
        xrecvdata = recvdata()
        senddata (1)
        Select Case xrecvdata
        Case 1
        Case 2
            Beep
            MsgBox "Error Stop", 48, APPNAME$
            qflg = True
        Case 3
            For i = 1 To 47
                For j = 0 To 3
                    x = 0
                    For k = 0 To 2
                        x = x + recvdata() * (2 ^ (3 * k))
                    Next
                    buffer(i, j + 1) = x
                Next
            Next
            ' decode student id and score
            xrdanswer         ' read answer to table
            tinstant = Now    ' reset time out feed
            sendcode (1)
            qflg = True
        Case 4                ' machine start read
        End Select
        If Difftime(tinstant) > TIMEOUTFD Then
            xtimeout = True
            qflg = True
        End If
        DoEvents
    Loop
    xrecvdata = recvdata()
    senddata (2)
    If xtimeout = True Then
        Beep
        MsgBox "Feed Paper time out", MB_ICONINFORMATION
    End If

    stopcmd.Enabled = False
    start.Enabled = True

End Sub

Sub Stopcmd_Click ()
    qflg = True
End Sub
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โปรแกรม selftest.frm

```
VERSION 2.00
Begin Form selftest
   BackColor       =   &H00C0C0C0&
   BorderStyle     =   3  'Fixed Double
   Caption         =   "Self Test"
   ClientHeight    =   2700
   ClientLeft      =   4200
   ClientTop       =   4065
   ClientWidth     =   4560
   Height          =   3105
   Left            =   4140
   LinkTopic       =   "Form3"
   ScaleHeight     =   2700
   ScaleWidth      =   4560
   Top             =   3720
   Width           =   4680
   Begin SSOption Option3D4
      Caption          =   "Counter test"
      Font3D           =   0  'None
      Height           =   255
      Left             =   1080
      TabIndex         =   4
      Top              =   1320
      Width            =   2775
   End
   Begin SSOption Option3D1
      Caption          =   "Motor test"
      Font3D           =   0  'None
      Height           =   255
      Left             =   1080
      TabIndex         =   3
      Top              =   240
      Width            =   2895
   End
   Begin SSOption Option3D3
      Caption          =   "Infrared test"
      Font3D           =   0  'None
      Height           =   255
      Left             =   1080
      TabIndex         =   2
      TabStop          =   0   'False
      Top              =   780
      Width            =   2895
   End
   Begin SSCommand Command3D1
      Caption          =   "OK"
      Font3D           =   3  'Inset w/light shading
      Height           =   375
      Left             =   840
      TabIndex         =   0
      Top              =   2040
      Width            =   975
   End
   Begin SSCommand Command3D2
      Caption          =   "Close"
      Font3D           =   3  'Inset w/light shading
      Height           =   375
      Left             =   2880
      TabIndex         =   1
      Top              =   2040
      Width            =   975
   End
End
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
Sub Command3D1_Click ()
    If option3d1.Value = True Then
        stmotor.Show modal
    End If
    If option3d3.Value = True Then
        inf_test.Show modal
    End If
    If option3d4.Value = True Then
        cnt_test.Show modal
    End If
End Sub

Sub Command3D2_Click ()
    Unload selftest
End Sub
```


.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โปรแกรม static.frm

```
VERSION 2.00
Begin Form Statistic
   BackColor       =   &H00C0C0C0&
   BorderStyle     =   1  'Fixed Single
   Caption         =   "Statistic"
   ClientHeight    =   5040
   ClientLeft      =   1425
   ClientTop       =   1740
   ClientWidth     =   5700
   ControlBox      =   0   'False
   Height          =   5445
   Left            =   1365
   LinkTopic       =   "Form1"
   MaxButton       =   0   'False
   MinButton       =   0   'False
   ScaleHeight     =   5040
   ScaleWidth      =   5700
   Top             =   1395
   Width           =   5820
   Begin SSCommand CmdOk
      Caption         =   "&OK"
      Font3D          =   3  'Inset w/light shading
      Height          =   375
      Left            =   2220
      TabIndex        =   28
      Top             =   4440
      Width           =   915
   End
   Begin SSPanel Panel3D1
      BackColor       =   &H00C0C0C0&
      BevelOuter      =   1  'Inset
      BorderWidth     =   0
      Font3D          =   1  'Raised w/light shading
      ForeColor       =   &H00FF0000&
      Height          =   1335
      Left            =   360
      TabIndex        =   0
      Top             =   2880
      Width           =   5055
      Begin SSPanel Panel3D2
         BackColor       =   &H00C0C0C0&
         BevelOuter      =   1  'Inset
         BorderWidth     =   0
         Caption         =   "Error Summary"
         Font3D          =   1  'Raised w/light shading
         ForeColor       =   &H00FF0000&
         Height          =   375
         Left            =   0
         TabIndex        =   1
         Top             =   0
         Width           =   5055
      End
      Begin Line Line1
         X1              =   1200
         X2              =   1200
         Y1              =   360
         Y2              =   1320
      End
      Begin Label Label6
         BackColor       =   &H00C0C0C0&
         Caption         =   "Score 3rd"
         Height          =   255
         Index           =   2
         Left            =   120
         TabIndex        =   2
         Top             =   960
         Width           =   855
      End
      Begin Label Label6
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
      BackColor         =   &H00C0C0C0&
      Caption           =   "Score 2nd"
      Height            =   255
      Index             =   1
      Left              =   120
      TabIndex          =   3
      Top               =   720
      Width             =   975
   End
   Begin Label Label6
      BackColor         =   &H00C0C0C0&
      Caption           =   "Score 1st"
      Height            =   255
      Index             =   0
      Left              =   120
      TabIndex          =   4
      Top               =   480
      Width             =   855
   End
End
Begin SSPanel Panel3D4
   BackColor         =   &H00C0C0C0&
   BevelOuter        =   1  'Inset
   BorderWidth       =   0
   Caption           =   "Score Summary"
   Font3D            =   1  'Raised w/light shading
   ForeColor         =   &H00FF0000&
   Height            =   375
   Left              =   360
   TabIndex          =   5
   Top               =   1080
   Width             =   5055
End
Begin SSPanel Panel3D3
   BackColor         =   &H00C0C0C0&
   BevelOuter        =   1  'Inset
   BorderWidth       =   0
   Font3D            =   1  'Raised w/light shading
   ForeColor         =   &H00FF0000&
   Height            =   1335
   Left              =   360
   TabIndex          =   13
   Top               =   1440
   Width             =   5055
   Begin Label strscore3min
      Alignment         =   1  'Right Justify
      BackColor         =   &H00C0C0C0&
      Caption           =   "0"
      Height            =   255
      Left              =   4080
      TabIndex          =   22
      Top               =   960
      Width             =   855
   End
   Begin Label strscore2min
      Alignment         =   1  'Right Justify
      BackColor         =   &H00C0C0C0&
      Caption           =   "0"
      Height            =   255
      Left              =   2880
      TabIndex          =   27
      Top               =   960
      Width             =   855
   End
   Begin Label strscore1min
      Alignment         =   1  'Right Justify
      BackColor         =   &H00C0C0C0&
      Caption           =   "0"
      Height            =   255
      Left              =   1800
      TabIndex          =   26
      Top               =   960
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
   Width           =   855
End
Begin Label strscore3avg
   Alignment       =   1  'Right Justify
   BackColor       =   &H00C0C0C0&
   Caption         =   "0"
   Height          =   255
   Left            =   4080
   TabIndex        =   25
   Top             =   720
   Width           =   855
End
Begin Label strscore2avg
   Alignment       =   1  'Right Justify
   BackColor       =   &H00C0C0C0&
   Caption         =   "0"
   Height          =   255
   Left            =   2880
   TabIndex        =   24
   Top             =   720
   Width           =   855
End
Begin Label strscore1avg
   Alignment       =   1  'Right Justify
   BackColor       =   &H00C0C0C0&
   Caption         =   "0"
   Height          =   255
   Left            =   1800
   TabIndex        =   23
   Top             =   720
   Width           =   855
End
Begin Label strscore3max
   Alignment       =   1  'Right Justify
   BackColor       =   &H00C0C0C0&
   Caption         =   "0"
   Height          =   255
   Left            =   4080
   TabIndex        =   21
   Top             =   480
   Width           =   855
End
Begin Label strscore2max
   Alignment       =   1  'Right Justify
   BackColor       =   &H00C0C0C0&
   Caption         =   "0"
   Height          =   255
   Left            =   2880
   TabIndex        =   20
   Top             =   480
   Width           =   855
End
Begin Label strscore1max
   Alignment       =   1  'Right Justify
   BackColor       =   &H00C0C0C0&
   Caption         =   "0"
   Height          =   255
   Left            =   1800
   TabIndex        =   19
   Top             =   480
   Width           =   855
End
Begin Line Line8
   X1              =   3840
   X2              =   3840
   Y1              =   0
   Y2              =   1320
End
Begin Line Line7
   X1              =   2760
   X2              =   2760
   Y1              =   0
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
   Y2              =   1320
End
Begin Line Line6
   X1              =   1680
   X2              =   1680
   Y1              =   0
   Y2              =   1320
End
Begin Line Line5
   X1              =   0
   X2              =   5040
   Y1              =   360
   Y2              =   360
End
Begin Label Label4
   BackColor       =   &H00C0C0C0&
   Caption         =   "Score Minimum"
   Height          =   255
   Index           =   2
   Left            =   120
   TabIndex        =   9
   Top             =   960
   Width           =   1575
End
Begin Label Label4
   BackColor       =   &H00C0C0C0&
   Caption         =   "Score Average"
   Height          =   255
   Index           =   1
   Left            =   120
   TabIndex        =   10
   Top             =   720
   Width           =   1575
End
Begin Label Label5
   BackColor       =   &H00C0C0C0&
   Caption         =   "Score 3rd"
   Height          =   255
   Index           =   2
   Left            =   3960
   TabIndex        =   11
   Top             =   120
   Width           =   975
End
Begin Label Label5
   BackColor       =   &H00C0C0C0&
   Caption         =   "Score 2nd"
   Height          =   255
   Index           =   1
   Left            =   2880
   TabIndex        =   12
   Top             =   120
   Width           =   975
End
Begin Label Label5
   BackColor       =   &H00C0C0C0&
   Caption         =   "Score 1st"
   Height          =   255
   Index           =   0
   Left            =   1800
   TabIndex        =   15
   Top             =   120
   Width           =   975
End
Begin Label Label4
   BackColor       =   &H00C0C0C0&
   Caption         =   "Score Maximum "
   Height          =   255
   Index           =   0
   Left            =   120
   TabIndex        =   14
   Top             =   480
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
      Width           =   1575
   End
End
Begin Label LastTest
   BackColor       =   &H00C0C0C0&
   Height          =   255
   Left            =   1800
   TabIndex        =   18
   Top             =   720
   Width           =   3855
End
Begin Label StudentCnt
   BackColor       =   &H00C0C0C0&
   Height          =   255
   Left            =   1800
   TabIndex        =   17
   Top             =   480
   Width           =   3615
End
Begin Label FileName
   BackColor       =   &H00C0C0C0&
   Height          =   255
   Left            =   1800
   TabIndex        =   16
   Top             =   240
   Width           =   3615
End
Begin Label Label3
   BackColor       =   &H00C0C0C0&
   Caption         =   "Last Call          :"
   Height          =   255
   Left            =   360
   TabIndex        =   8
   Top             =   720
   Width           =   1335
End
Begin Label Label2
   BackColor       =   &H00C0C0C0&
   Caption         =   "Student Count :"
   Height          =   255
   Left            =   360
   TabIndex        =   7
   Top             =   480
   Width           =   1455
End
Begin Label Label1
   BackColor       =   &H00C0C0C0&
   Caption         =   "File Name       :"
   Height          =   255
   Left            =   360
   TabIndex        =   6
   Top             =   240
   Width           =   1455
End
End
Sub CmdOk_Click ()
    Unload statistic
End Sub

Sub Form_Load ()
Dim score1max, score2max, score3max
Dim score1min, score2min, score3min
Dim score1sum, score2sum, score3sum
Dim temp As String

    temp$ = String(16, 0)
    statistic.Height = 5445
    statistic.Width = 5856
    statistic.Top = Int((Screen.Height - statistic.Height) / 2)
    statistic.Left = Int((Screen.Width - statistic.Width) / 2)
    ' initial value
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    FileName.Caption = filePrj
    StudentCnt.Caption = Str(mytbl.RecordCount)
    i = GetProfileString(APPNAME$, "LASTDATE", "none", temp$, 16)
    LastTest.Caption = temp$
    i = GetProfileString(APPNAME$, KEYSTATISTIC1$, "", temp$, 16)
    LastTest.Caption = LastTest.Caption + ", " + temp$

    ' initial value
    score1sum = 0
    score2sum = 0
    score3sum = 0
    score1min = 999
    score2min = 999
    score3min = 999
    score1max = 0
    score2max = 0
    score3max = 0
    mytbl.MoveFirst
    '
    ' This could take a while, display hourglass
    '
    Screen.MousePointer = 11

    Do Until mytbl.EOF
        score1sum = score1sum + mytbl.Fields("score1").Value
        score2sum = score2sum + mytbl.Fields("score2").Value
        score3sum = score3sum + mytbl.Fields("score3").Value
        If mytbl.Fields("score1").Value < score1min Then
            score1min = mytbl.Fields("score1").Value
        End If
        If mytbl.Fields("score2").Value < score2min Then
            score2min = mytbl.Fields("score2").Value
        End If
        If mytbl.Fields("score3").Value < score3min Then
            score3min = mytbl.Fields("score3").Value
        End If

        If mytbl.Fields("score1").Value > score1max Then
            score1max = mytbl.Fields("score1").Value
        End If
        If mytbl.Fields("score2").Value > score2max Then
            score2max = mytbl.Fields("score2").Value
        End If
        If mytbl.Fields("score3").Value > score3max Then
            score3max = mytbl.Fields("score3").Value
        End If
        mytbl.MoveNext
    Loop
    '
    ' Return the mouse pointer to normal
    '
    Screen.MousePointer = 0
    strscore1min = Str$(score1min)
    strscore2min = Str$(score2min)
    strscore3min = Str$(score3min)
    strscore1max = Str$(score1max)
    strscore2max = Str$(score2max)
    strscore3max = Str$(score3max)
    strscore1avg = Format$(score1sum / mytbl.RecordCount, "0.00")
    strscore2avg = Format$(score2sum / mytbl.RecordCount, "0.00")
    strscore3avg = Format$(score3sum / mytbl.RecordCount, "0.00")

End Sub
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โปรแกรม stmotor.frm

```
VERSION 2.00
Begin Form stmotor
   BackColor       =   &H00C0C0C0&
   Caption         =   "Stepper motor"
   ClientHeight    =   2280
   ClientLeft      =   1095
   ClientTop       =   1485
   ClientWidth     =   4965
   Height          =   2685
   Left            =   1035
   LinkTopic       =   "Form1"
   ScaleHeight     =   2280
   ScaleWidth      =   4965
   Top             =   1140
   Width           =   5085
   Begin CommandButton Command2
      Caption          =   "Close"
      Height           =   375
      Left             =   3000
      TabIndex         =   2
      Top              =   1440
      Width            =   1215
   End
   Begin SSPanel Panel3D1
      BackColor        =   &H00C0C0C0&
      BevelOuter       =   0  'None
      BorderWidth      =   0
      Caption          =   "DC MOTOR TEST"
      Font3D           =   1  'Raised w/light shading
      FontBold         =   -1  'True
      FontItalic       =   0  'False
      FontName         =   "MS Sans Serif"
      FontSize         =   12
      FontStrikethru   =   0  'False
      FontUnderline    =   0  'False
      ForeColor        =   &H00FF0000&
      Height           =   735
      Left             =   600
      TabIndex         =   1
      Top              =   360
      Width            =   3615
   End
   Begin CommandButton Command1
      Caption          =   "Start"
      Height           =   375
      Left             =   720
      TabIndex         =   0
      Top              =   1440
      Width            =   1215
   End
End
Dim qflg

Sub Command1_Click ()
If command1.Caption = "Start" Then
    command2.Enabled = False
    command1.Caption = "Stop"
    qflg = False
    If port_init(True) = False Then
        Beep
        MsgBox "Checker machine not connect", 48, APPNAME$
        Exit Sub
    End If
    senddata (1)
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
Else
    qflg = True
    command1.Caption = "Start"
    command2.Enabled = True
End If
Do While qflg <> True
    senddata (1)
    DoEvents
Loop
senddata (2)

End Sub
Sub Command2_Click ()
    Unload Me
End Sub
```


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โปรแกรม tester.frm

```
VERSION 2.00
Begin MDIForm tester
   Caption          =   "CHECKER MACHINE"
   ClientHeight     =   5445
   ClientLeft       =   1125
   ClientTop        =   2160
   ClientWidth      =   7755
   Height           =   6135
   Icon             =   TESTER.FRX:0000
   Left             =   1065
   LinkMode         =   1  'Source
   LinkTopic        =   "MDIForm1"
   Top              =   1530
   Width            =   7875
   WindowState      =   2  'Maximized
   Begin PictureBox Picture1
      Align            =   1  'Align Top
      Height           =   600
      Left             =   0
      ScaleHeight      =   570
      ScaleWidth       =   7725
      TabIndex         =   11
      Top              =   405
      Visible          =   0   'False
      Width            =   7755
      Begin CommonDialog OpenDb
         DefaultExt       =   "*.dbf"
         DialogTitle      =   "Open Database"
         Filter           =   "FoxPro (*.dbf) | *.dbf"
         Left             =   45
         Top              =   45
      End
   End
   Begin PictureBox StatusBar
      Align            =   2  'Align Bottom
      BackColor        =   &H00C0C0C0&
      ForeColor        =   &H00C0C0C0&
      Height           =   360
      Left             =   0
      ScaleHeight      =   330
      ScaleWidth       =   7725
      TabIndex         =   1
      Top              =   5085
      Width            =   7755
      Begin Timer Timer1
         Interval         =   1000
         Left             =   8325
         Top              =   -45
      End
      Begin MhState KeyStat1
         Autosize         =   -1  'True
         Height           =   420
         Index            =   2
         Left             =   7875
         Style            =   2  'Insert State
         TabIndex         =   10
         Tag              =   "INS"
         TimerInterval    =   250
         Top              =   -45
         Value            =   0   'False
         Visible          =   0   'False
         Width            =   420
      End
      Begin MhState KeyStat1
         Autosize         =   -1  'True
         Height           =   420
         Index            =   1
         Left             =   7875
         Style            =   1  'Num Lock
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
   TabIndex          =   9
   Tag               =   "NUM"
   TimerInterval     =   250
   Top               =   -45
   Value             =   0   'False
   Visible           =   0   'False
   Width             =   420
End
Begin MhState KeyStat1
   Autosize          =   -1  'True
   Height            =   420
   Index             =   0
   Left              =   7875
   TabIndex          =   8
   Tag               =   "CAP"
   TimerInterval     =   250
   Top               =   -45
   Value             =   0   'False
   Visible           =   0   'False
   Width             =   420
End
Begin SSPanel StatusMessagePanel
   Alignment         =   1  'Left Justify - MIDDLE
   BackColor         =   &H00C0C0C0&
   BevelInner        =   2  'Raised
   BevelOuter        =   0  'None
   BorderWidth       =   0
   Caption           =   "Ready"
   Font3D            =   0  'None
   ForeColor         =   &H00000000&
   Height            =   345
   Left              =   0
   RoundedCorners    =   0   'False
   TabIndex          =   7
   Top               =   0
   Width             =   5205
End
Begin SSPanel KeyStatus
   BackColor         =   &H00C0C0C0&
   BevelInner        =   2  'Raised
   BevelOuter        =   0  'None
   BorderWidth       =   0
   Font3D            =   0  'None
   ForeColor         =   &H00000000&
   Height            =   350
   Left              =   5220
   RoundedCorners    =   0   'False
   TabIndex          =   2
   Top               =   0
   Width             =   2535
   Begin SSPanel TimePanel
      BackColor      =   &H00C0C0C0&
      BevelInner     =   1  'Inset
      BevelOuter     =   0  'None
      BorderWidth    =   0
      Caption        =   "99:99:99"
      Font3D         =   0  'None
      ForeColor      =   &H00000000&
      Height         =   285
      Left           =   1665
      RoundedCorners =   0   'False
      TabIndex       =   6
      Top            =   30
      Width          =   825
   End
   Begin SSPanel KeyStatusPanels
      BackColor      =   &H00C0C0C0&
      BevelInner     =   1  'Inset
      BevelOuter     =   0  'None
      BorderWidth    =   0
      Caption        =   "INS"
      Font3D         =   0  'None
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
      ForeColor       =   &H00000000&
      Height          =   285
      Index           =   2
      Left            =   1125
      RoundedCorners  =   0   'False
      TabIndex        =   5
      Top             =   30
      Width           =   495
   End
   Begin SSPanel KeyStatusPanels
      BackColor       =   &H00C0C0C0&
      BevelInner      =   1  'Inset
      BevelOuter      =   0  'None
      BorderWidth     =   0
      Caption         =   "NUM"
      Font3D          =   0  'None
      ForeColor       =   &H00000000&
      Height          =   285
      Index           =   1
      Left            =   585
      RoundedCorners  =   0   'False
      TabIndex        =   4
      Top             =   30
      Width           =   495
   End
   Begin SSPanel KeyStatusPanels
      BackColor       =   &H00C0C0C0&
      BevelInner      =   1  'Inset
      BevelOuter      =   0  'None
      BorderWidth     =   0
      Caption         =   "CAP"
      Font3D          =   0  'None
      ForeColor       =   &H00000000&
      Height          =   285
      Index           =   0
      Left            =   45
      RoundedCorners  =   0   'False
      TabIndex        =   3
      Top             =   30
      Width           =   495
   End
End
Begin Line LineStatus
   BorderColor     =   &H00000000&
   BorderWidth     =   2
   X1              =   5220
   X2              =   5220
   Y1              =   0
   Y2              =   405
End
End
Begin SSPanel ToolsBarPanel
   Align           =   1  'Align Top
   BackColor       =   &H00C0C0C0&
   BorderWidth     =   0
   Font3D          =   0  'None
   ForeColor       =   &H00000000&
   Height          =   405
   Left            =   0
   TabIndex        =   0
   Top             =   0
   Width           =   7755
   Begin SSRibbon ToolsBar
      BackColor       =   &H00C0C0C0&
      BevelWidth      =   0
      Height          =   330
      Index           =   5
      Left            =   2075
      Outline         =   0   'False
      PictureDisabled =   TESTER.FRX:0302
      PictureDn       =   TESTER.FRX:0484
      PictureDnChange =   0  'Use 'PictureUp' Bitmap Unchanged
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
   PictureUp         =   TESTER.FRX:0606
   RoundedCorners    =   0   'False
   Top               =   30
   Width             =   360
End
Begin SSRibbon ToolsBar
   BackColor         =   &H00C0C0C0&
   BevelWidth        =   0
   Height            =   330
   Index             =   4
   Left              =   1745
   Outline           =   0   'False
   PictureDisabled   =   TESTER.FRX:0788
   PictureDn         =   TESTER.FRX:090A
   PictureDnChange   =   0  'Use 'PictureUp' Bitmap Unchanged
   PictureUp         =   TESTER.FRX:0A8C
   RoundedCorners    =   0   'False
   Top               =   30
   Width             =   360
End
Begin SSRibbon ToolsBar
   BackColor         =   &H00C0C0C0&
   BevelWidth        =   0
   Height            =   330
   Index             =   3
   Left              =   1400
   Outline           =   0   'False
   PictureDisabled   =   TESTER.FRX:0C0E
   PictureDn         =   TESTER.FRX:0D90
   PictureDnChange   =   0  'Use 'PictureUp' Bitmap Unchanged
   PictureUp         =   TESTER.FRX:0F12
   RoundedCorners    =   0   'False
   Top               =   30
   Width             =   360
End
Begin SSRibbon ToolsBar
   BackColor         =   &H00C0C0C0&
   BevelWidth        =   0
   Height            =   330
   Index             =   9
   Left              =   3645
   Outline           =   0   'False
   PictureDisabled   =   TESTER.FRX:1094
   PictureDn         =   TESTER.FRX:1216
   PictureDnChange   =   0  'Use 'PictureUp' Bitmap Unchanged
   PictureUp         =   TESTER.FRX:1398
   Top               =   45
   Width             =   360
End
Begin SSRibbon ToolsBar
   BackColor         =   &H00C0C0C0&
   BevelWidth        =   0
   Height            =   330
   Index             =   8
   Left              =   3240
   Outline           =   0   'False
   PictureDisabled   =   TESTER.FRX:151A
   PictureDn         =   TESTER.FRX:169C
   PictureDnChange   =   0  'Use 'PictureUp' Bitmap Unchanged
   PictureUp         =   TESTER.FRX:181E
   Top               =   30
   Width             =   360
End
Begin SSRibbon ToolsBar
   BackColor         =   &H00C0C0C0&
   BevelWidth        =   0
   Height            =   330
   Index             =   7
   Left              =   2880
   Outline           =   0   'False
   PictureDisabled   =   TESTER.FRX:19A0
   PictureDn         =   TESTER.FRX:1B22
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    PictureDnChange =   0  'Use 'PictureUp' Bitmap Unchanged
    PictureUp       =   TESTER.FRX:1CA4
    Top             =   30
    Width           =   360
 End
 Begin SSRibbon ToolsBar
    BackColor       =   &H00C0C0C0&
    BevelWidth      =   0
    Height          =   330
    Index           =   6
    Left            =   2535
    Outline         =   0   'False
    PictureDisabled =   TESTER.FRX:1E26
    PictureDn       =   TESTER.FRX:1FA8
    PictureDnChange =   0  'Use 'PictureUp' Bitmap Unchanged
    PictureUp       =   TESTER.FRX:212A
    Top             =   30
    Width           =   360
 End
 Begin SSRibbon ToolsBar
    BackColor       =   &H00C0C0C0&
    BevelWidth      =   0
    Height          =   330
    Index           =   2
    Left            =   1050
    Outline         =   0   'False
    PictureDisabled =   TESTER.FRX:22AC
    PictureDn       =   TESTER.FRX:242E
    PictureDnChange =   0  'Use 'PictureUp' Bitmap Unchanged
    PictureUp       =   TESTER.FRX:25B0
    Top             =   30
    Width           =   360
 End
 Begin SSRibbon ToolsBar
    BackColor       =   &H00C0C0C0&
    BevelWidth      =   0
    Height          =   330
    Index           =   1
    Left            =   590
    Outline         =   0   'False
    PictureDisabled =   TESTER.FRX:2732
    PictureDn       =   TESTER.FRX:28B4
    PictureDnChange =   0  'Use 'PictureUp' Bitmap Unchanged
    PictureUp       =   TESTER.FRX:2A36
    Top             =   30
    Width           =   360
 End
 Begin SSRibbon ToolsBar
    AutoSize        =   1  'Adjust Picture Size To Button
    BackColor       =   &H00C0C0C0&
    BevelWidth      =   0
    Height          =   330
    Index           =   0
    Left            =   225
    Outline         =   0   'False
    PictureDisabled =   TESTER.FRX:2BB8
    PictureDn       =   TESTER.FRX:2D3A
    PictureDnChange =   0  'Use 'PictureUp' Bitmap Unchanged
    PictureUp       =   TESTER.FRX:2EBC
    RoundedCorners  =   0   'False
    Top             =   30
    Width           =   360
 End
End
Begin Menu m_file
   Caption         =   "&File"
   Begin Menu m_file_open
      Caption      =   "&Open..."
      Shortcut     =   ^O
   End
   Begin Menu m_file_close
      Caption      =   "&Close"
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
      Enabled          =   0   'False
   End
   Begin Menu m_file_save
      Caption          =   "&Save"
      Shortcut         =   ^S
   End
   Begin Menu m_file_separate0
      Caption          =   "-"
   End
   Begin Menu m_file_print
      Caption          =   "&Print..."
      Shortcut         =   ^P
   End
   Begin Menu m_file_separate2
      Caption          =   "-"
   End
   Begin Menu m_file_exit
      Caption          =   "E&xit"
   End
End
Begin Menu m_view
   Caption          =   "&View"
   Begin Menu m_view_showall
      Caption          =   "&Show All Data"
   End
   Begin Menu m_view_graph
      Caption          =   "&View Graph Information"
   End
   Begin Menu m_view_summ
      Caption          =   "Sh&ow Summary and Statistic"
   End
   Begin Menu m_view_find
      Caption          =   "&Find..."
      Shortcut         =   ^F
   End
End
Begin Menu m_run
   Caption          =   "&Run"
   Begin Menu m_run_start
      Caption          =   "&Start"
   End
   Begin Menu m_run_stop
      Caption          =   "St&op"
   End
   Begin Menu m_run_separate0
      Caption          =   "-"
   End
   Begin Menu m_run_self
      Caption          =   "&Respond self-test"
   End
End
Begin Menu m_options
   Caption          =   "&Options"
   Begin Menu m_options_setp
      Caption          =   "Setup Program"
   End
   Begin Menu m_options_seta
      Caption          =   "Setup answer"
   End
End
Begin Menu m_help
   Caption          =   "&Help"
   Begin Menu m_help_on
      Caption          =   "Help on checker auto-machine"
   End
   Begin Menu m_help_separate
      Caption          =   "-"
   End
   Begin Menu m_help_about
      Caption          =   "About checker auto-machine"
   End
End
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
End
Option Explicit

Sub CloseProject ()
    i = WriteProfileString(APPNAME$, KEYSTATISTIC$, Date$)
    i = WriteProfileString(APPNAME$, KEYSTATISTIC1$, Time$)
    mytbl.Close
    answertbl.Close
    MyDb.Close
    Unload GphView
    Unload Lookup
    Unload Statistic
    ActStr = "NEW"
    ObjEnaDis
End Sub

Sub KeyStat1_change (Index As Integer)
    '
    ' Some key has changed state, fix the text displayed for it
    '
    If (KeyStat1(Index).Value = True) Then
        KeyStatusPanels(Index).Caption = KeyStat1(Index).Tag
    Else
        KeyStatusPanels(Index).Caption = ""
    End If
End Sub

Sub m_file_close_Click ()
    SelectMenu "CLOSE"
End Sub

Sub m_file_exit_Click ()
    SelectMenu "EXIT"
End Sub

Sub m_file_open_Click ()
    SelectMenu "Open"
End Sub

Sub m_help_about_Click ()
    about.Show modal
End Sub

Sub m_options_seta_Click ()
    answermode.Show modal
End Sub

Sub m_options_setp_Click ()
    options.Show modal
End Sub

Sub m_run_self_Click ()
    If port_init(True) = False Then
        Beep
        MsgBox "Checker machine not connect", 48, APPNAME$
        Exit Sub
    End If
    selftest.Show modal
End Sub

Sub m_run_start_Click ()
    SelectMenu "Run"
End Sub

Sub m_run_stop_Click ()
    SelectMenu "Stop"
End Sub

Sub m_view_find_Click ()
    SelectMenu "VIEW FIND"
End Sub
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
Sub m_view_graph_Click ()
    SelectMenu "VIEW GRAPH"
End Sub

Sub m_view_showall_Click ()
    SelectMenu "View All"
End Sub

Sub m_view_summ_Click ()
    SelectMenu "View Summ"
End Sub

Sub MDIForm_Load ()

    ' get profile from win.ini
    CHOICES = GetProfileInt(APPNAME$, "CHOICES", 100)
    TIMEOUT = GetProfileInt(APPNAME$, "TIMEOUT", 10)
    IDLEN = GetProfileInt(APPNAME$, "IDLEN", 8)
    TIMEOUTFD = GetProfileInt(APPNAME$, "TIMEOUTFD", 10)

    SetDataAccessOption 1, ".\tester.ini"
    '
    ' Display the current state of the keys
    '
    For i = 0 To 2
        KeyStat1_change (i)
    Next
    ' Reset flag when form load
    ActStr = "New"      ' Because form load is first start

    ObjEnaDis

    about.Show modal
    Exit Sub

End Sub

Sub MDIForm_Resize ()
    KeyStatus.Left = Tester.Width - KeyStatus.Width - 150
    If KeyStatus.Left / Twips2ChrHor < 40 Then
        KeyStatus.Left = 41 * Twips2ChrHor
    End If

    LineStatus.X1 = KeyStatus.Left
    LineStatus.X2 = LineStatus.X1
    StatusMessagePanel.Width = KeyStatus.Left - 20
End Sub

Sub mnu_edval_Click ()
End Sub

Sub mnu_test_Click ()
    test.Show
End Sub

' Object Enable and Disable
' Because some item enable and some item must disable
Sub ObjEnaDis ()
    Dim i As Integer
    Select Case UCase(ActStr)
        Case "NEW"
            ' Set Window Caption
            Tester.Caption = testerfrm + " (Open Database to Start)"
            ' Enable Tools Bar
            ToolsBar(0).Enabled = True  ' Open
            ToolsBar(8).Enabled = True  ' Exit
            ' Disable Tools Bar
            For i = 1 To 8
                ToolsBar(i).Enabled = False
            Next
            ' Enable Menu
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        m_file_open = True
        m_file_exit = True
        ' Disable Menu
        m_run_start.Enabled = False
        m_run_stop.Enabled = False

        m_file_close = False
        m_file_save.Enabled = False
        m_file_print.Enabled = False

        m_view_showall = False
        m_view_graph = False
        m_view_summ = False
        m_view_find = False
        m_view_decode = False
        m_options_seta = False

    Case "OPEN"
        ' Set Window Caption
        Tester.Caption = testerfrm + " " + FilePrj
        ' Enable Tools Bar
        ToolsBar(1).Enabled = True
        ToolsBar(2).Enabled = True
        ToolsBar(3).Enabled = True
        ToolsBar(4).Enabled = True
        ToolsBar(5).Enabled = True
        ToolsBar(6).Enabled = True
        ' Disable Tools Bar
        ToolsBar(0).Enabled = False
        ' Enable Menu
        m_file_close = True
        m_file_save = True
        m_file_print = True
        m_run_start.Enabled = True
        m_view_showall = True
        m_view_graph = True
        m_view_summ = True
        m_view_find = True
        m_view_decode = True
        m_options_seta = True
        ' Disable Menu
        m_file_open = False
     Case "RUN"
        ' Enable Tools Bar
        ToolsBar(7).Enabled = True
        ToolsBar(8).Enabled = True
        ' Disable Tools Bar
        ToolsBar(6).Enabled = False
        ' Enable Menu
        m_run_stop.Enabled = True
        ' Disable Menu
        m_run_start.Enabled = False
        m_run_self.Enabled = False
     Case "STOP"
        ' Enable Tools Bar
        ToolsBar(6).Enabled = True
        ' Disable Tools Bar
        ToolsBar(7).Enabled = False
        ToolsBar(8).Enabled = False
        ' Enable Menu
        m_run_start.Enabled = True
        m_run_self.Enabled = True
        ' Disable Menu
        m_run_stop.Enabled = False

    End Select

End Sub

Sub OpenProject ()
    Dim i As Integer, fldmax
    Dim DbfField() As afield
```

เอกสารนี้สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    fldmax = 5
    ReDim DbfField(fldmax)
    DbfField(0).fieldname = "STD_ID"
    DbfField(0).fieldtype = DB_TEXT
    DbfField(0).fieldlen = 15
    DbfField(0).fielddec = 0
    DbfField(1).fieldname = "STD_NAME"
    DbfField(1).fieldtype = DB_TEXT
    DbfField(1).fieldlen = 50
    DbfField(1).fielddec = 0
    DbfField(2).fieldname = "SCORE1"
    DbfField(2).fieldtype = DB_DOUBLE
    DbfField(2).fieldlen = 8
    DbfField(2).fielddec = 0
    DbfField(3).fieldname = "SCORE2"
    DbfField(3).fieldtype = DB_DOUBLE
    DbfField(3).fieldlen = 8
    DbfField(3).fielddec = 0
    DbfField(4).fieldname = "SCORE3"
    DbfField(4).fieldtype = DB_DOUBLE
    DbfField(4).fieldlen = 8
    DbfField(4).fielddec = 0


    OpenDb.Flags = OFN_HIDEREADONLY
    OpenDb.Action = 1

    FilePrj = OpenDb.Filename
    If Alltrim(FilePrj) = "" Then
        Exit Sub
    End If

    Set MyDb = OpenDatabase(justpath(FilePrj), True, False, "FoxPro 2.5;")
    Set mytbl = MyDb.OpenTable(juststem(FilePrj))
    Set answertbl = MyDb.OpenTable("answer")

    ' check structure of table
    If mytbl.Fields.Count <> fldmax Or mytbl.EOF = True Then
        GoTo InvalidStru
    End If
    For i = 0 To mytbl.Fields.Count - 1
        If   mytbl.Fields(i).Name   <>   DbfField(i).fieldname   Or
mytbl.Fields(i).Size  <>  DbfField(i).fieldlen  Or  mytbl.Fields(i).Type  <>
DbfField(i).fieldtype Then
            GoTo InvalidStru
        End If
    Next
    mytbl.Index = "std_id"

    ActStr = "Open"

    ObjEnaDis

    GoTo Finished


InvalidStru:
    Beep
    MsgBox "   Invalid Database Structure.", MB_ICONSTOP, "Error Message"
    mytbl.Close
    answertbl.Close
    MyDb.Close

Finished:
    Exit Sub
End Sub

Sub RunProcess ()
    ActStr = "RUN"
    ObjEnaDis
    comm.Show
End Sub
```

เอกสารนี้จัดทำขึ้นเพื่อการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
Sub SelectMenu (menu As String)
    Select Case UCase(menu)
        Case "OPEN"
            OpenProject
        Case "CLOSE"
            CloseProject
        Case "EXIT"
            End
        Case "RUN"
            RunProcess
        Case "VIEW ALL"
            Unload Lookup
            Load Lookup
        Case "VIEW GRAPH"
            Unload GphView
            Load GphView
        Case "VIEW SUMM"
            Statistic.Show modal
        Case "VIEW FIND"
            findcode.Show
        Case "STOP"
            stopprocess
    End Select

End Sub

Sub stopprocess ()
    ActStr = "Stop"
    Unload comm
    ObjEnaDis
End Sub

Sub Timer1_Timer ()
    '
    ' Another second gone ... update the display
    '
    TimePanel.Caption = Time$
End Sub

Sub ToolsBar_Click (Index As Integer, Value As Integer)
    If Value = True Then
        ToolsBar(Index).Value = False
        Exit Sub
    End If
    Select Case Index
        Case 0      ' Open file
            SelectMenu "Open"
        Case 1      ' Save file
        Case 2      ' Print Data
        Case 3      ' View Data
            SelectMenu "View All"
        Case 4      ' View Graph
            SelectMenu "VIEW GRAPH"
        Case 5
            SelectMenu "VIEW FIND"
        Case 6      ' Play Program
            SelectMenu "Run"
        Case 7      ' Pause Program
        Case 8      ' Stop Program
            SelectMenu "Stop"
        Case 9      ' Exit Door
        SelectMenu "Exit"
    End Select
End Sub
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โปรแกรม comm.bas

```
'Get data from controller
Function get_data () As Integer
    get_data = logicand(ShiftBitR(GetB(&H379), 3), &HF)
End Function

'Port_Init      : Initializes the registers needed for transfer
'Input          : SENDER = TRUE if sender, FALSE if receiver
'Output         : TRUE if register initializes successfully
Function port_init (ByVal Sender As Integer) As Integer
Dim Tinstant
    Tinstant = Now                          ' Initial current time
    If Sender = True Then                   ' Device = Sender ?
        put_data &H8                        ' Send :   00001000b
        Do Until get_data() = &H8 Or DiffTime(Tinstant) > TIMEOUT' wait for
00000000b
        Loop
    Else
        Do Until get_data() = &H8 Or DiffTime(Tinstant) > TIMEOUT' wait for
00000000b
        Loop
        put_data &H8                        ' Send :   00001000b
    End If
    If DiffTime(Tinstant) - TIMEOUT > 0 Then
        port_init = False
    Else
        port_init = True
    End If
End Function

'Put data to controller
Sub put_data (ByVal value As Integer)
    PutB OutPort, value
End Sub

'RecvData       : Receive data from controller and send returned parts for
testing
'Input          : None
'Output         : Receive data
Function recvdata () As Integer
Dim Tinstant                    ' Time instant
Dim xGetData                    ' Received data

' --- Receive and re-send data ----------------------------------------

    Tinstant = Now                  ' Time instant
    Do Until logicand(get_data(), &H8) = 0 Or DiffTime(Tinstant) > TIMEOUT'
Wait for message
    Loop
    xGetData = get_data()
    put_data logicand(xGetData, &H7)    ' re - send data

    ' reset port
    Tinstant = Now
    Do Until get_data() = &H8 Or DiffTime(Tinstant) > TIMEOUT' wait for
00000000b
    Loop
    put_data &H8                    ' Send :   00001000b

    recvdata = xGetData

End Function
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
'SendData        : Send data to controller and check the result
'Input           : DSend = Data to be sent
'Output          : Transfer successful? (0=error, -1=O.K.)
Sub senddata (ByVal Dsend As Integer)
Dim RcvData As Integer        ' Received data
Dim xGetData As Integer       ' extend data get from controller
Dim Tinstant                  ' Time instant

' --- Send and re-read data ----------------------------------------

    Tinstant = Now                        ' Initialize time out counter
    put_data logicand(Dsend, &H7)         ' send data
    Do Until logicand(get_data(), &H8) = 0 Or DiffTime(Tinstant) > TIMEOUT'
Wait for message
    Loop
    xGetData = get_data()

    ' reset port
    Tinstant = Now                        ' Initialize time out counter
    put_data &H8                          ' Send :   00001000b
    Do Until get_data() = &H8 Or DiffTime(Tinstant) > TIMEOUT' wait for
00000000b
    Loop

End Sub
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โปรแกรม extfunc.bas

```
' Function DLL in Windows
Declare Function GetTextExtent Lib "GDI" (ByVal hDC As Integer, ByVal
lpString As String, ByVal nCount As Integer) As Long
Declare Function GetDC Lib "User" (ByVal hWnd As Integer) As Integer
Declare Function ReleaseDC Lib "User" (ByVal hWnd As Integer, ByVal hDC As
Integer) As Integer
Declare Function OpenComm Lib "User" (ByVal lpComName As String, ByVal
wInQueue As Integer, ByVal wOutQueue As Integer) As Integer
Declare Function Closecomm Lib "User" (ByVal nCid As Integer) As Integer

Declare Function GetB Lib "IO_PORT.DLL" (ByVal PortId As Integer) As
Integer
Declare Sub PutB Lib "IO_PORT.DLL" (ByVal PortId As Integer, ByVal Value As
Integer)
Declare Function ShiftBitR Lib "IO_PORT.DLL" (ByVal Value As Integer, ByVal
BitPos As Integer) As Integer
Declare Function ShiftBitL Lib "IO_PORT.DLL" (ByVal Value As Integer, ByVal
BitPos As Integer) As Integer
Declare Function LogicAnd Lib "IO_PORT.DLL" (ByVal Value As Integer, ByVal
AndValue As Integer) As Integer

' Profile function
Declare Function GetProfileInt Lib "Kernel" (ByVal lpAppName$, ByVal
lpKeyName$, ByVal nDefault%) As Integer
Declare Function WriteProfileString Lib "Kernel" (ByVal lpApplicationName$,
ByVal lpKeyName$, ByVal lpString$) As Integer
Declare Function GetProfileString Lib "Kernel" (ByVal lpAppName$, ByVal
lpKeyName$, ByVal lpDefault$, ByVal lpReturnedString$, ByVal nSize%) As
Integer
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โปรแกรม global.bas

```
' Application
Global Const APPNAME$ = "CHECKER"
Global Const KEYSTATISTIC$ = "LASTDATE"
Global Const KEYSTATISTIC1$ = "LASTTIME"
Global Const KEYSTATISTIC2$ = "IDLEN"

' Global key status
Global Const key_cap = False
Global Const key_ins = False
Global Const key_num = True

' Global mouse press
Global Const mouse_left_press = 1
Global Const mouse_right_press = 2


' Global Action String
Global ActStr As String


' Global Const Convert Variable
Global Const Twips2ChrHor = 120
Global Const Twips2ChrVer = 240

' Global Const Caption
Global Const testerfrm = "CHECKER MACHINE"
' Global file action
Global FilePrj As String                ' Project file name to open
Global MyDb As database                 ' Directory store table
Global MyTbl As table                   ' Table to Open
Global MyFld As Field                   ' Field in table
Global AnswerTbl As table                 ' Table store answer
' Global const structure database
Global Const DB_TEXT = 10
Global Const DB_DOUBLE = 7
' Global variable Communication
Global PaperCnt As Integer              ' Paper Count from machine
Global ErrorFlg As Integer              ' Error Flag Status
' Define Type field structure
Type afield
    fieldname As String                 ' Field Name
    fieldtype As Integer                ' Field Type
    fieldlen As Integer                 ' Field Lenght
    fielddec As Integer                 ' Field Decimal
End Type
' Global for options value
Global CHOICES As Integer               ' Amount of choice
Global TIMEOUT As Integer               ' Time out define
Global IDLEN As Integer                 ' Student id length
Global TIMEOUTFD As Integer             ' Time out feed define
' Global variable and constants for transfer protocol
Global OK, i, TOKEN, TRNFLEN, CMDCODE
Global Const ACK = 0                    'Acknowledge
Global Const NAK = &HFF                 'Non-Acknowledge
Global Const MAX_TRY = 10               'Maximum number of tries
' Tokens for communication between sender and receiver
Global Const TOK_DATSTART = 7           'Start of data send
Global Const TOK_DATNEXT = 1            'Next data block
Global Const TOK_DATEND = 2             'End file data transfer
Global Const TOK_ENDIT = 3              'End program
Global Const TOK_ESCAPE = 4             '<Stop> on controller board
Global Const InpPort = &H379            'Input port address
Global Const OutPort = &H378            'Output port address
' Array variable store result
Global buffptr As Integer
Global buffer(47, 4) As Integer
Global answerbuf(150) As Integer
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โปรแกรม utils.bas

```
Function Alltrim (ByVal msg As String) As String
    Alltrim = LTrim(RTrim(msg))
End Function

Function At (ByVal find_str As String, mainstr As String) As Integer
    For i = 1 To Len(mainstr)
        If Mid$(mainstr, i, Len(find_str)) = find_str Then
            At = i
            Exit For
        End If
    Next
End Function

Function DiffTime (Instant) As Integer
Dim HourDiff, MinuteDiff, SecondDiff   ' Declare variables
Dim TotalMinDiff, TotalSecDiff
Dim TimeNow

    TimeNow = Now
    HourDiff = Hour(TimeNow) - Hour(Instant)  ' Get differences.
    MinuteDiff = Minute(TimeNow) - Minute(Instant)
    SecondDiff = Second(TimeNow) - Second(Instant) + 1
    If SecondDiff = 60 Then
        MinuteDiff = MinuteDiff + 1 ' Add 1 to minute.
        SecondDiff = 0  ' Zero seconds.
    End If
    If MinuteDiff = 60 Then
        HourDiff = HourDiff + 1 ' Add 1 to hour.
        MinuteDiff = 0  ' Zero minutes.
    End If
    TotalMinDiff = (HourDiff * 60) + MinuteDiff ' Get totals.
    TotalSecDiff = (TotalMinDiff * 60) + SecondDiff


    DiffTime = TotalSecDiff

End Function

Function justext (ByVal filname As String) As String
    filname = justfname(filname)    ' prevents problems with ..\ paths
    ext = ""
    If At(".", filname) > 0 Then
        ext = Mid$(filname, At(".", filname) + 1, 3)
    End If
    justext = UCase(ext)
End Function

Function justfname (ByVal filname As String) As String
    If RAT("\", filname) > 0 Then
       filname = Mid$(filname, RAT("\", filname) + 1, 255)
    End If
    If At(":", filname) > 0 Then
       filname = Mid$(filname, At(":", filname) + 1, 255)
    End If
    justfname = Alltrim(UCase(filname))
End Function
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
Function justpath (ByVal filname As String) As String
    filname = Format$(LTrim(RTrim(filname)), ">")
    If InStr(filname, "\") > 0 Then
       filname = Mid$(filname, 1, RAT("\", filname))
       If Right(filname, 1) = "\" And Len(filname) > 1 And Mid$(filname,
       Len(filname) - 1, 1) <> ":" Then
          filname = Mid$(filname, 1, Len(filname) - 1)
       End If
       justpath = filname
    Else
       justpath = ""
    End If
End Function

Function juststem (ByVal filname As String) As String
    If RAT("\", filname) > 0 Then
        filname = Mid$(filname, RAT("\", filname) + 1, 255)
    End If
    If At(":", filname) > 0 Then
        filname = Mid$(filname, At(":", filname) + 1, 255)
    End If
    If At(".", filname) > 0 Then
        filname = Mid$(filname, 1, At(".", filname) - 1)
    End If
    juststem = Alltrim(UCase(filname))
End Function

Sub mouseglass ()
    '
    ' This could take a while, display hourglass
    '
    Screen.MousePointer = 11
End Sub

Sub mousenormal ()
    '
    ' Return the mouse pointer to normal
    '
    Screen.MousePointer = 0
End Sub

Function RAT (ByVal find_str As String, mainstr As String) As Integer
    For i = Len(mainstr) To 1 Step -1
        If Mid$(mainstr, i, Len(find_str)) = find_str Then
            RAT = i
            Exit For
        End If
    Next
End Function

'Separate bit
Function SepBit (ByVal Number As Integer, ByVal bitpos As Integer) As
Integer
    Select Case bitpos
    Case 0
        Number = Number & &H7
    Case 1
        Number = Number & &H38
        Number = ShiftBitR(Number, 3)
    Case 2
        Number = Number & &HC0
        Number = ShiftBitR(Number, 6)
    End Select
    SepBit = Number
End Function
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

DATA SHEET

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

MOTOROLA
SEMICONDUCTOR
TECHNICAL DATA

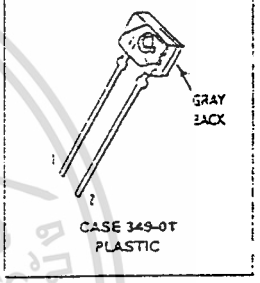
Photo Detector
Darlington Output

MRD711

PHOTO DETECTOR
DARLINGTON OUTPUT
NPN SILICON
50 VOLTS

. . . designed for a wide variety of industrial processing and control applications requiring a sensitive detector. The MRD711 is in an identical package and is designed for use with the MLED71 infrared emitter.

- Miniature, Low Profile, Clear Plastic Package
- Designed for Automatic Handling and Accurate Positioning
- Side Looking, with Molded Lens
- High Volume, Economical



CASE 349-07
PLASTIC

MAXIMUM RATINGS (T_A = 25°C unless otherwise noted)

Rating	Symbol	Value	Unit
Collector-Emitter Voltage	V_{CEO}	50	Volts
Total Device Dissipation @ T_A = 25°C Derate above 25°C (Note 1)	P_D	150 2	mW mW/°C
Operating and Storage Junction Temperature Range	T_J, T_{stg}	−40 to +100	°C
Lead Soldering Temperature (5 sec. max, 1/16" from case) (Note 2)	—	260	°C

ELECTRICAL CHARACTERISTICS (T_A = 25°C unless otherwise noted)

Characteristic	Symbol	Min	Typ	Max	Unit
Collector Dark Current (V_{CE} = 10 V, H = 0)	I_D	—	—	100	nA
Collector-Emitter Breakdown Voltage (I_C = 1 mA, H = 0)	$V_{(BR)CEO}$	50	—	—	Volts
Capacitance (V_{CC} = 5 V, f = 1 MHz)	C_{ce}	—	3.9	—	pF

OPTICAL CHARACTERISTICS (T_A = 25°C unless otherwise noted)

Collector Light Current (V_{CE} = 5 V, H = 500 μW/cm², λ = 940 nm)		I_L	5	25	—	mA
Turn-On Time	H = 500 μW/cm², V_{CC} = 5 V R_L = 100 Ω	t_{on}	—	125	—	μs
Turn-Off Time		t_{off}	—	150	—	
Saturation Voltage (H = 500 μW/cm², λ = 940 nm, I_C = 2 mA, V_{CC} = 5 V)		$V_{CE(sat)}$	—	0.75	1	Volts
Wavelength of Maximum Sensitivity		λ_s	—	0.8	—	μm

Notes: 1. Measured with device soldered into a typical printed circuit board.
2. Heat sink should be applied to leads during soldering to prevent case temperature from exceeding 100°C.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

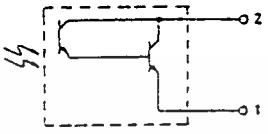


Figure 1. Typical Operating Circuit

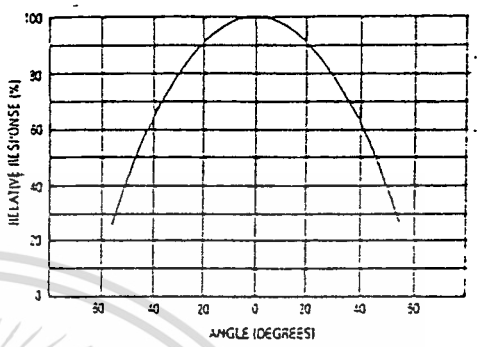


Figure 2. Angular Response

TYPICAL COUPLED CHARACTERISTICS USING MLED71
EMITTER AND MRD711 PHOTODARLINGTON DETECTOR

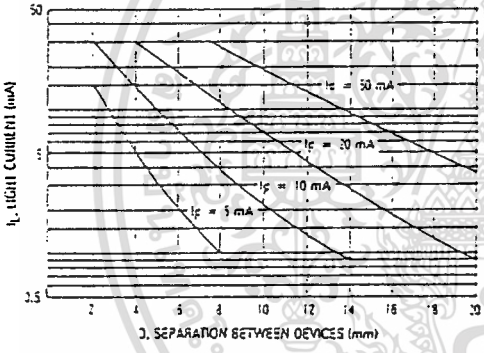


Figure 3. Continuous MRD711 Collector Light Current versus Distance from MLED71

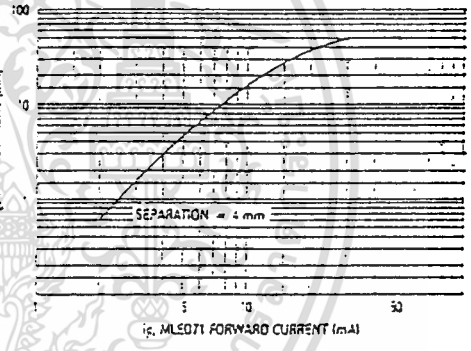
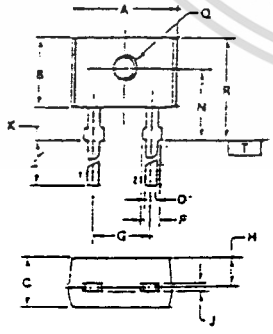


Figure 4. Instantaneous MRD711 Collector Light Current versus MLED71 Forward Current

OUTLINE DIMENSIONS

CASE 349-01
PLASTIC



NOTES:
1. DIMENSIONS A, B AND C ARE DATUMS.
2. POSITIONAL TOLERANCE FOR D DIMENSION:
⌖ ⌀0.25 (⌀0.010) M A M B M C
3. POSITIONAL TOLERANCE FOR Q DIAMETER:
⌖ ⌀0.25 (⌀0.010) M A M B M
4. ⏊ IS SEATING PLANE.
5. DIMENSIONING AND TOLERANCING PER ANSI Y14.5, 1973.

STYLE 2:
PIN 1. EMITTER
2. COLLECTOR

DIM	MILLIMETERS MIN	MILLIMETERS MAX	INCHES MIN	INCHES MAX
A	5.43	5.60	0.215	0.185
B	2.79	3.30	0.110	0.130
C	4.83	5.18	0.080	0.175
D	0.43	0.50	0.017	0.024
F	1.14	1.40	0.045	0.055
G	2.54 BSC		0.100 BSC	
H	7.62 BSC		0.060 BSC	
J	0.33	0.56	0.009	0.022
K	12.83	19.05	0.505	0.750
N	2.05	3.30	0.120	0.130
Q	0.76	1.52	0.030	0.060
R	3.81	4.80	0.150	0.195

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

MOTOROLA
SEMICONDUCTOR
TECHNICAL DATA

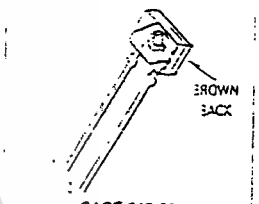
Infrared LED

MLED71

INFRARED
LED
940 nm

This device is designed for a wide variety of infrared applications, including keyboards, end-of-tape sensors, coin or paper handlers, and other general sensing applications. The MLED71 can be used in conjunction with any MRD700 series detector. It features high power output, using gallium arsenide technology.

- Low Cost
- Popular Case 349 Package, with Molded Lens
- Uses Stable Long-Life LED Technology
- Clear Epoxy Package



CASE 349-01
PLASTIC

4

MAXIMUM RATINGS

Rating	Symbol	Value	Unit
Reverse Voltage	V_R	5	Volts
Forward Current — Continuous	I_F	50	mA
Forward Current — Peak Pulse	I_F	1	A
Total Power Dissipation @ T_A = 25°C (Note 1) Derate above 55°C	P_D	80 2	mW mW/°C
Ambient Operating Temperature Range	T_A	−40 to +100	°C
Storage Temperature	T_{stg}	−40 to +100	°C
Lead Soldering Temperature (Note 2)	—	260	°C

ELECTRICAL CHARACTERISTICS (T_A = 25°C unless otherwise noted)

Characteristic	Symbol	Min	Typ	Max	Unit
Reverse Leakage Current (V_R = 5 V)	I_R	—	0.05	100	μA
Forward Voltage (I_F = 50 mA)	V_F	—	1.3	1.5	V
Temperature Coefficient of Forward Voltage	ΔV_F	—	−1.5	—	mV/K
Capacitance (V = 0 V, f = 1 MHz)	C	—	18	—	pF

OPTICAL CHARACTERISTICS (T_A = 25°C unless otherwise noted)

Peak Wavelength (I_F = 50 mA)	λ_P	—	940	—	nm
Spectral Half-Power Bandwidth	$\Delta\lambda$	—	48	—	nm
Continuous Power Output (I_F = 50 mA) (Note 3)	P_O	2	2.5	—	mW
Instantaneous Power Output (I_F = 100 mA)	P_O	—	5	—	mW
Instantaneous Axial Intensity (I_F = 100 mA) (Note 4)	I_O	—	2.5	—	mW/sr
Power Half-Angle	θ	—	±10	—	°
Optical Turn-On and Turn-Off Times	t_{on}, t_{off}	—	1	—	μs

Notes: 1. Measured with device soldered into a typical printed circuit board.
2. 5 seconds max; 1/16 inch from case. Heat sink should be applied during soldering, to prevent case temperature from exceeding 100°C.
3. Measured using a Photodyne 88xLA with a 4π integrating sphere.
4. On 0° axis, with cone angle of ±7°.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

TYPICAL CHARACTERISTICS

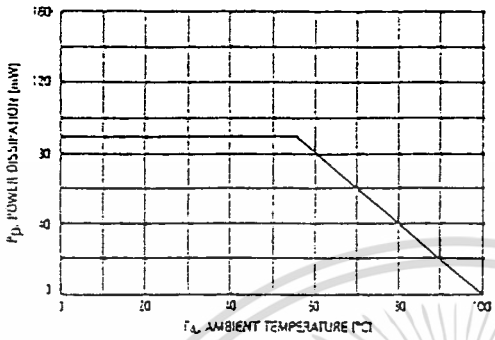


Figure 1. Power Dissipation

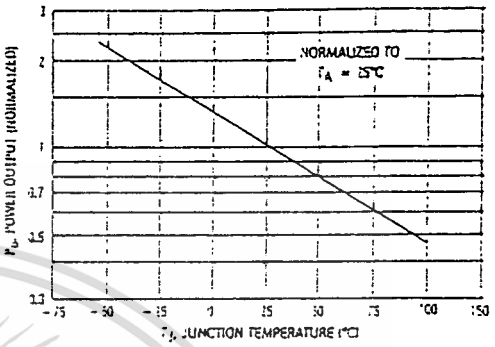


Figure 2. Instantaneous Power Output versus Ambient Temperature

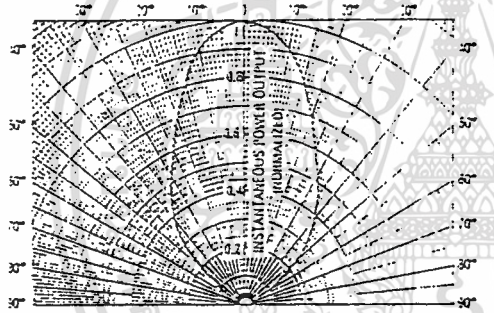


Figure 3. Spatial Radiation Pattern

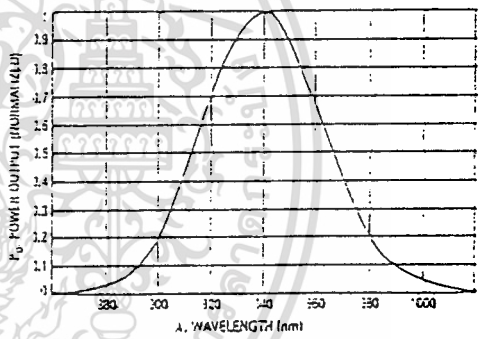


Figure 4. Relative Spectral Output

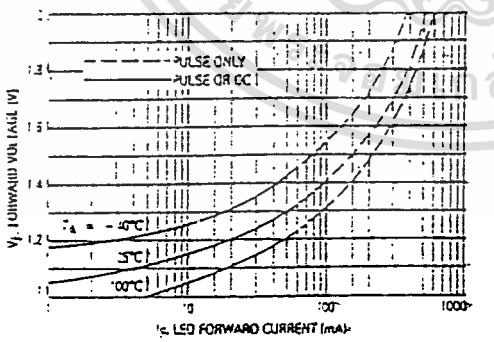


Figure 5. Forward Voltage versus Forward Current

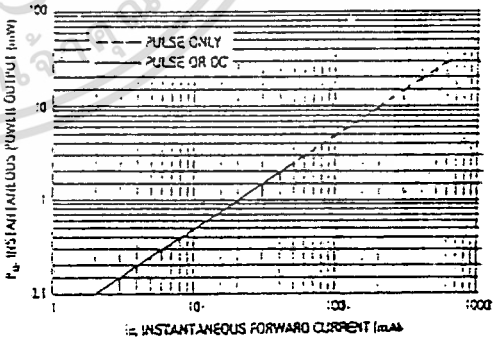
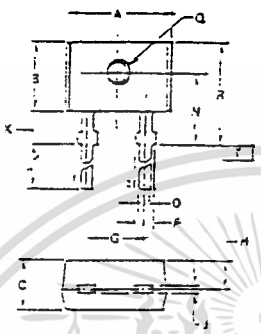


Figure 6. Instantaneous Power Output versus Forward Current

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

OUTLINE DIMENSIONS

CASE 349-01
PLASTIC



NOTES:
1. DIMENSIONS A, B AND C ARE DATUMS.
2. POSITIONAL TOLERANCE FOR Q DIMENSION:
   ⌀ 0.25 (0.010) M | A M | B M | C M
3. POSITIONAL TOLERANCE FOR Q DIAMETER:
   ⌀ 0.25 (0.010) M | A M | B M
4. ... IS SEATING PLANE.
5. DIMENSIONING AND TOLERANCING PER ANSI Y14.5, 1973.

STYLE 1:
PIN 1. CATHODE
2. ANODE

DIM	MILLIMETERS MIN	MILLIMETERS MAX	INCHES MIN	INCHES MAX
A	1.43	1.60	0.125	1.185
B	1.79	1.20	1.110	0.120
C	1.03	1.18	1.080	1.125
D	1.43	1.50	1.017	1.024
F	1.14	1.40	1.045	1.055
G	2.54 BSC		0.100 BSC	
H	1.52 BSC		0.060 BSC	
J	1.23	1.56	0.009	0.022
K	12.83	13.05	0.505	0.750
N	1.05	1.20	0.170	0.170
Q	3.76	3.92	0.250	0.060
R	1.31	1.60	0.150	0.185

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

National Semiconductor

LM139/LM239/LM339/LM139A/LM239A/LM339A/ LM2901/LM3302 Low Power Low Offset Voltage Quad Comparators

General Description

The LM139 series consists of four independent precision voltage comparators with an offset voltage specification as low as 2 mV max for all four comparators. These were designed specifically to operate from a single power supply over a wide range of voltages. Operation from split power supplies is also possible and the low power supply current drain is independent of the magnitude of the power supply voltage. These comparators also have a unique characteristic in that the input common-mode voltage range includes ground, even though operated from a single power supply voltage.

Application areas include limit comparators, simple analog to digital converters; pulse, squarewave and time delay generators; wide range VCO; MOS clock timers; multivibrators and high voltage digital logic gates. The LM139 series was designed to directly interface with TTL and CMOS. When operated from both plus and minus power supplies, they will directly interface with MOS logic— where the low power drain of the LM339 is a distinct advantage over standard comparators.

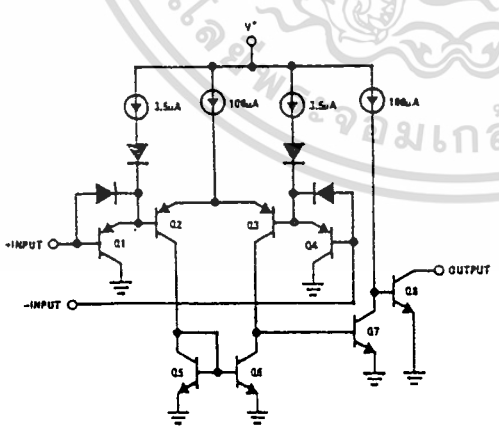
Advantages

- High precision comparators
- Reduced V_{OS} drift over temperature
- Eliminates need for dual supplies
- Allows sensing near GND
- Compatible with all forms of logic
- Power drain suitable for battery operation

Features

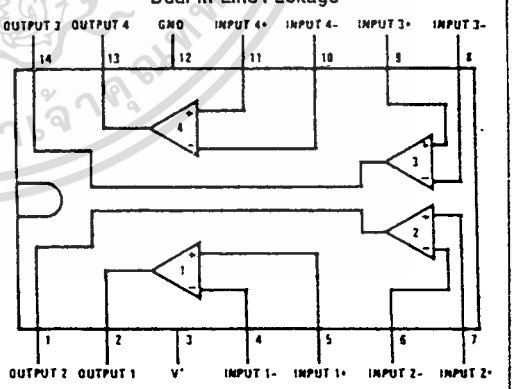
- Wide single supply voltage range of dual supplies
  - LM139 series, 2 V_{DC} to 36 V_{DC} or
  - LM139A series, LM2901 ±1 V_{DC} to ±18 V_{DC}
  - LM3302 2 V_{DC} to 28 V_{DC} or ±1 V_{DC} to ±14 V_{DC}
- Very low supply current drain (0.8 mA) — independent of supply voltage
- Low input biasing current 25 nA
- Low input offset current ±5 nA
- and offset voltage ±3 mV
- Input common-mode voltage range includes GND
- Differential input voltage range equal to the power supply voltage
- Low output saturation voltage 250 mV at 4 mA
- Output voltage compatible with TTL, DTL, ECL, MOS and CMOS logic systems

Schematic and Connection Diagrams



TL/H/5706–1

Dual-In-Line Package



TOP VIEW TL/H/5706–2

Order Number LM139J, LM139AJ, LM239J, LM239AJ, LM339J, LM339AJ, LM2901J or LM3302J
See NS Package Number J14A
Order Number LM339AM, LM339M or LM2901M
See NS Package Number M14A
Order Number LM339N, LM339AN, LM2901N or LM3302N
See NS Package Number N14A

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Absolute Maximum Ratings

If Military/Aerospace specified devices are required, please contact the National Semiconductor Sales Office/Distributors for availability and specifications. (Note 10)

	LM139/LM239/LM339 LM139A/LM239A/LM339A LM2901	LM3302
Supply Voltage, V^+	36 V_{DC} or ±18 V_{DC}	28 V_{DC} or ±14 V_{DC}
Differential Input Voltage (Note 8)	36 V_{DC}	28 V_{DC}
Input Voltage	−0.3 V_{DC} to +36 V_{DC}	−0.3 V_{DC} to +28 V_{DC}
Input Current ($V_{IN} < -0.3$ V_{DC}), (Note 3)	50 mA	50 mA
Power Dissipation (Note 1)		
Molded DIP	1050 mW	1050 mW
Cavity DIP	1190 mW	
Small Outline Package	760 mW	
Output Short-Circuit to GND, (Note 2)	Continuous	Continuous
Storage Temperature Range	−65°C to +150°C	−65°C to +150°C
Lead Temperature (Soldering, 10 seconds)	260°C	260°C

	LM139/LM239/LM339 LM139A/LM239A/LM339A LM2901	LM3302
Operating Temperature Range		−40°C to +85°C
LM339/LM339A	0°C to +70°C	
LM239/LM239A	−25°C to +85°C	
LM2901	−40°C to +85°C	
LM139/LM139A	−55°C to +125°C	
Soldering Information		
Dual-In-Line Package		
Soldering (10 seconds)	260°C	260°C
Small Outline Package		
Vapor Phase (60 seconds)	215°C	215°C
Infrared (15 seconds)	220°C	220°C
See AN-450 "Surface Mounting Methods and Their Effect on Product Reliability" for other methods of soldering surface mount devices.		
ESD rating (1.5 kΩ in series with 100 pF)	600V	600V

Electrical Characteristics ($V^+ = 5$ V_{DC}, $T_A = 25°C$, unless otherwise stated)

Parameter	Conditions	LM139A			LM239A, LM339A			LM139			LM239, LM339			LM2901			LM3302			Units
		Min	Typ	Max	Min	Typ	Max	Min	Typ	Max	Min	Typ	Max	Min	Typ	Max	Min	Typ	Max	
Input Offset Voltage	(Note 9)		±1.0	±2.0		±1.0	±2.0		±2.0	±5.0		±2.0	±5.0		±2.0	±7.0		±3	±20	mV_{DC}
Input Bias Current	$I_{IN(+)}$ or $I_{IN(-)}$ with Output in Linear Range, (Note 5), $V_{CM} = 0V$		25	100		25	250		25	100		25	250		25	250		25	500	nA_{DC}
Input Offset Current	$I_{IN(+)} - I_{IN(-)}$, $V_{CM} = 0V$		±3.0	±25		±5.0	±50		±3.0	±25		±5.0	±50		±5	±50		±3	±100	nA_{DC}
Input Common-Mode Voltage Range	$V^+ = 30$ V_{DC} (LM3302, $V^+ = 28$ V_{DC}) (Note 6)	0		$V^+ - 1.5$	0		$V^+ - 1.5$	0		$V^+ - 1.5$	0		$V^+ - 1.5$	0		$V^+ - 1.5$	0		$V^+ - 1.5$	V_{DC}
Supply Current	$R_L = \infty$ on all Comparators,		0.8	2.0		0.8	2.0		0.8	2.0		0.8	2.0		0.8	2.0		0.8	2.0	mA_{DC}
	$R_L = \infty$, $V^+ = 36V$, (LM3302, $V^+ = 28$ V_{DC})					1.0	2.5		1.0	2.5		1.0	2.5		1.0	2.5		1.0	2.5	mA_{DC}
Voltage Gain	$R_L \geq 15$ kΩ, $V^+ = 15$ V_{DC} $V_o = 1$ V_{DC} to 11 V_{DC}	50	200		50	200		50	200		50	200		25	100		2	30		V/mV
Large Signal Response Time	V_{IN} = TTL Logic Swing, V_{REF} = 1.4 V_{DC}, $V_{RL} = 5$ V_{DC}, $R_L = 5.1$ kΩ,		300			300			300			300			300			300		ns
Response Time	$V_{RL} = 5$ V_{DC}, $R_L = 5.1$ kΩ, (Note 7)		1.3			1.3			1.3			1.3			1.3			1.3		μs
Output Sink Current	$V_{IN(-)} = 1$ V_{DC}, $V_{IN(+)} = 0$, $V_O \leq 1.5$ V_{DC}	6.0	16		6.0	16		6.0	16		6.0	16		6.0	16		6.0	16		mA_{DC}

LM139/LM239/LM339/LM139A/LM239A/LM339A/LM2901/LM3302

Electrical Characteristics ($V^+ = 5\ V_{DC}$, $T_A = 25°C$, unless otherwise stated) (Continued)

Parameter	Conditions	LM139A			LM239A, LM339A			LM139			LM239, LM339			LM2901			LM3302			Units
		Min	Typ	Max	Min	Typ	Max	Min	Typ	Max	Min	Typ	Max	Min	Typ	Max	Min	Typ	Max	
Saturation Voltage	$V_{IN(-)} \geq 1\ V_{DC}$, $V_{IN(+)} = 0$, $I_{SINK} \leq 4\ mA$		250	400		250	400		250	400		250	400		250	400		250	500	mV_{DC}
Output Leakage Current	$V_{IN(+)} = 1\ V_{DC}$, $V_{IN(-)} = 0$, $V_O = 5\ V_{DC}$		0.1			0.1			0.1			0.1			0.1			0.1		nA_{DC}

Electrical Characteristics ($V^+ = 5.0\ V_{DC}$, Note 4)

Parameter	Conditions	LM139A			LM239A, LM339A			LM139			LM239, LM339			LM2901			LM3302			Units
		Min	Typ	Max	Min	Typ	Max	Min	Typ	Max	Min	Typ	Max	Min	Typ	Max	Min	Typ	Max	
Input Offset Voltage	(Note 9)			±4.0			±4.0			±9.0			±9.0		±9	±15			±40	mV_{DC}
Input Offset Current	$I_{IN(+)} - I_{IN(-)}$, $V_{CM} = 0V$			±100			±150			±100			±150		±50	±200			±300	nA_{DC}
Input Bias Current	$I_{IN(+)}$ or $I_{IN(-)}$ with Output in Linear Range, $V_{CM} = 0V$ (Note 5)			300			400			300			400		200	500			1000	nA_{DC}
Input Common-Mode Voltage Range	$V^+ = 30\ V_{DC}$ (LM3302, $V^+ = 28\ V_{DC}$) (Note 6)	0		$V^+ - 2.0$	0		$V^+ - 2.0$	0		$V^+ - 2.0$			$V^+ - 2.0$	0		$V^+ - 2.0$	0		$V^+ - 2.0$	V_{DC}
Saturation Voltage	$V_{IN(-)} \geq 1\ V_{DC}$, $V_{IN(+)} = 0$, $I_{SINK} \leq 4\ mA$			700			700			700			700		400	700			700	mV_{DC}
Output Leakage Current	$V_{IN(+)} \geq 1\ V_{DC}$, $V_{IN(-)} = 0$, $V_O = 30\ V_{DC}$, (LM3302, $V_O = 28\ V_{DC}$)			1.0			1.0			1.0			1.0			1.0			1.0	μA_{DC}
Differential Input Voltage	Keep all V_{IN}'s $\geq 0\ V_{DC}$ (or V^-, if used), (Note 8)			36			36			36			36			36			28	V_{DC}

Note 1: For operating at high temperatures, the LM339/LM339A, LM2901, LM3302 must be derated based on a 125°C maximum junction temperature and a thermal resistance of 95°C/W which applies for the device soldered in a printed circuit board, operating in a still air ambient. The LM239 and LM139 must be derated based on a 150°C maximum junction temperature. The low bias dissipation and the "ON-OFF" characteristic of the outputs keeps the chip dissipation very small ($P_D \leq 100$ mW), provided the output transistors are allowed to saturate.

Note 2: Short circuits from the output to V^+ can cause excessive heating and eventual destruction. When considering short circuits to ground, the maximum output current is approximately 20 mA independent of the magnitude of V^+.

Note 3: This input current will only exist when the voltage at any of the input leads is driven negative. It is due to the collector-base junction of the input PNP transistors becoming forward biased and thereby acting as input diode clamps. In addition to this diode action, there is also lateral NPN parasitic transistor action on the IC chip. This transistor action can cause the output voltages of the comparators to go to the V^+ voltage level (or to ground for a large overdrive) for the time duration that an input is driven negative. This is not destructive and normal output states will re-establish when the input voltage, which was negative, again returns to a value greater than $-0.3\ V_{DC}$ (at 25°C).

Note 4: These specifications are limited to $-55°C \leq T_A \leq +125°C$, for the LM139/LM139A. With the LM239/LM239A, all temperature specifications are limited to $-25°C \leq T_A \leq +85°C$, the LM339/LM339A temperature specifications are limited to $0°C \leq T_A \leq +70°C$, and the LM2901, LM3302 temperature range is $-40°C \leq T_A \leq +85°C$.

Note 5: The direction of the input current is out of the IC due to the PNP input stage. This current is essentially constant, independent of the state of the output so no loading change exists on the reference or input lines.

Note 6: The input common-mode voltage or either input signal voltage should not be allowed to go negative by more than 0.3V. The upper end of the common-mode voltage range is $V^+ - 1.5V$ at 25°C, but either or both inputs can go to $+30\ V_{DC}$ without damage (25V for LM3302), independent of the magnitude of V^+.

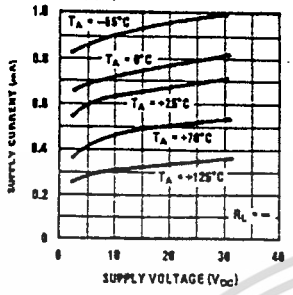
Note 7: The response time specified is a 100 mV input step with 5 mV overdrive. For larger overdrive signals 300 ns can be obtained, see typical performance characteristics section.

Note 8: Positive excursions of input voltage may exceed the power supply level. As long as the other voltage remains within the common mode range, the comparator will provide a proper output state. The low input voltage state must not be less than $-0.3\ V_{DC}$ (or 0.3 V_{DC} below the magnitude of the negative power supply, if used) (at 25°C).

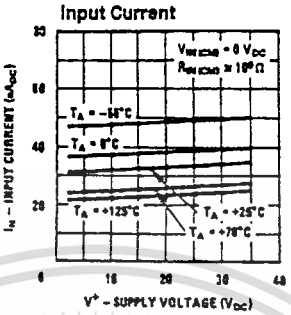
Note 9: [illegible] V^+ from 5 V_{DC} to 30 V_{DC}, and over the full input common-mode range (0 V_{DC} to $V^+ - 1.5\ V_{DC}$) at 25°C. For LM3302, V^+ from 5 V_{DC} to 28 V_{DC}.

Typical Performance Characteristics LM139/LM239/LM339, LM139A/LM239A/LM339A, LM3302

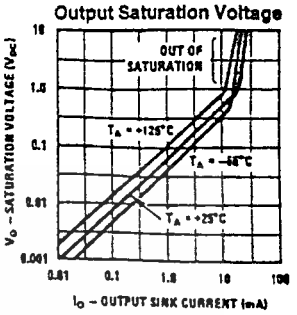
Supply Current



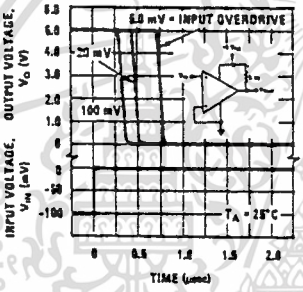
Input Current



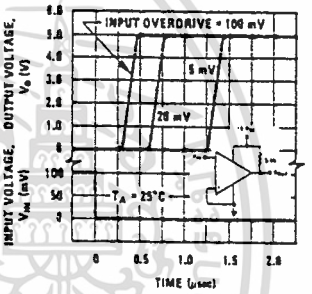
Output Saturation Voltage



Response Time for Various Input Overdrives—Negative Transition



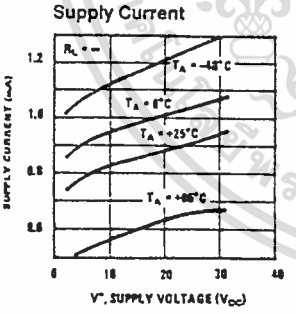
Response Time for Various Input Overdrives—Positive Transition



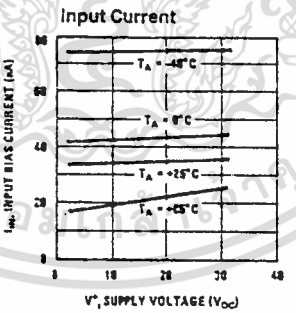
TL/H/5706–6

Typical Performance Characteristics LM2901

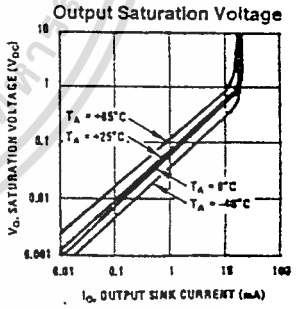
Supply Current



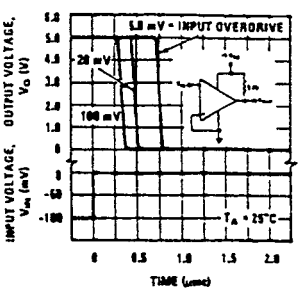
Input Current



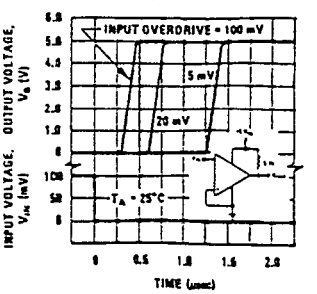
Output Saturation Voltage



Response Time for Various Input Overdrives-Negative Transition



Response Time for Various Input Overdrives-Positive Transition



TL/H/5706–7

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

MOTOROLA

SN54/74LS138

1-OF-8-DECODER/ DEMULTIPLEXER

LOW POWER SCHOTTKY

DESCRIPTION — The LSTTL/MSI SN54LS/74LS138 is a high speed 1-of-8 Decoder/Demultiplexer. This device is ideally suited for high-speed bipolar memory chip select address decoding. The multiple input enables allow parallel expansion to a 1-of-24 decoder using just three LS138 devices or to a 1-of-32 decoder using four LS138s and one inverter. The LS138 is fabricated with the Schottky barrier diode process for high speed and is completely compatible with all Motorola TTL families.

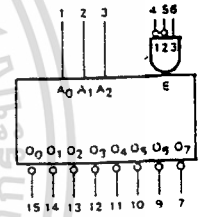
- DEMULTIPLEXING CAPABILITY
- MULTIPLE INPUT ENABLE FOR EASY EXPANSION
- TYPICAL POWER DISSIPATION OF 32 mW
- ACTIVE LOW MUTUALLY EXCLUSIVE OUTPUTS
- INPUT CLAMP DIODES LIMIT HIGH SPEED TERMINATION EFFECTS

PIN NAMES

		LOADING (Note a) HIGH	LOW
$A_0 - A_2$	Address Inputs	0.5 U.L.	0.25 U.L.
$\overline{E}_1, \overline{E}_2$	Enable (Active LOW) Inputs	0.5 U.L.	0.25 U.L.
E_3	Enable (Active HIGH) Input	0.5 U.L.	0.25 U.L.
$\overline{O}_0 - \overline{O}_7$	Active LOW Outputs (Note b)	10 U.L.	5(2.5) U.L.

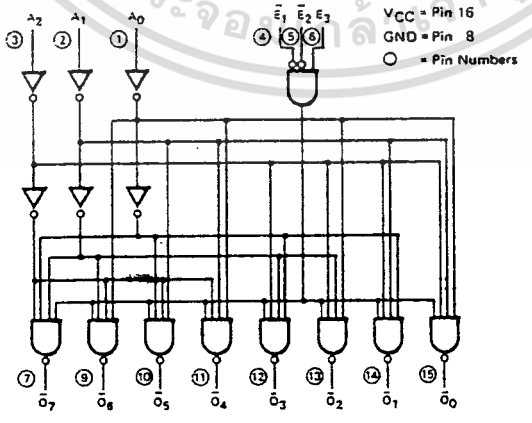
NOTES:
a. 1 TTL Unit Load (U.L.) = 40 μA HIGH/1.6 mA LOW.
b. The Output LOW drive factor is 2.5 U.L. for Military (54) and 5 U.L. for Commercial (74) Temperature Ranges.

LOGIC SYMBOL

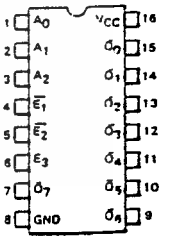


V_{CC} = Pin 16
GND = Pin 8

LOGIC DIAGRAM



CONNECTION DIAGRAM DIP (TOP VIEW)



J Suffix — Case 620-09 (Ceramic)
N Suffix — Case 648-08 (Plastic)

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

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

FUNCTIONAL DESCRIPTION — The LS138 is a high speed 1-of-8 Decoder/Demultiplexer fabricated with the low power Schottky barrier diode process. The decoder accepts three binary weighted inputs (A_0, A_1, A_2) and when enabled provides eight mutually exclusive active LOW outputs ($\overline{O}_0$-$\overline{O}_7$). The LS138 features three Enable inputs, two active LOW ($\overline{E}_1$, $\overline{E}_2$) and one active HIGH (E_3). All outputs will be HIGH unless $\overline{E}_1$ and $\overline{E}_2$ are LOW and E_3 is HIGH. This multiple enable function allows easy parallel expansion of the device to a 1-of-32 (5 lines to 32 lines) decoder with just four LS138s and one inverter. (See Figure a.)

The LS138 can be used as an 8-output demultiplexer by using one of the active LOW Enable inputs as the data input and the other Enable inputs as strobes. The Enable inputs which are not used must be permanently tied to their appropriate active HIGH or active LOW state.

TRUTH TABLE

INPUTS						OUTPUTS							
$\overline{E}_1$	$\overline{E}_2$	E_3	A_0	A_1	A_2	$\overline{O}_0$	$\overline{O}_1$	$\overline{O}_2$	$\overline{O}_3$	$\overline{O}_4$	$\overline{O}_5$	$\overline{O}_6$	$\overline{O}_7$
H	X	X	X	X	X	H	H	H	H	H	H	H	H
X	H	X	X	X	X	H	H	H	H	H	H	H	H
X	X	L	X	X	X	H	H	H	H	H	H	H	H
L	L	H	L	L	L	L	H	H	H	H	H	H	H
L	L	H	H	L	L	H	L	H	H	H	H	H	H
L	L	H	L	H	L	H	H	L	H	H	H	H	H
L	L	H	H	H	L	H	H	H	L	H	H	H	H
L	L	H	L	L	H	H	H	H	H	L	H	H	H
L	L	H	H	L	H	H	H	H	H	H	L	H	H
L	L	H	L	H	H	H	H	H	H	H	H	L	H
L	L	H	H	H	H	H	H	H	H	H	H	H	L

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

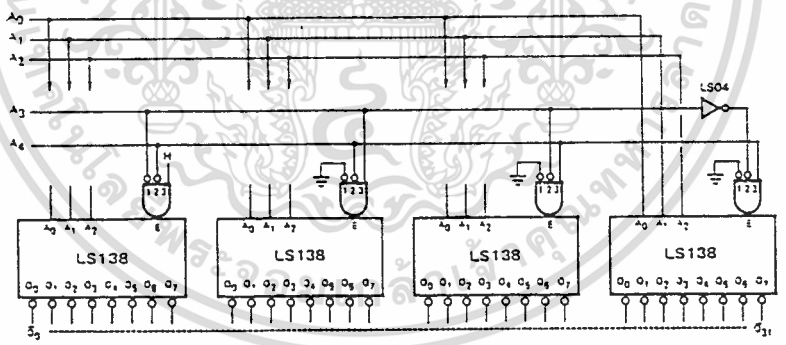


Fig. a.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

GUARANTEED OPERATING RANGES

SYMBOL	PARAMETER		MIN	TYP	MAX	UNIT
V_{CC}	Supply Voltage	54 74	4.5 4.75	5.0 5.0	5.5 5.25	V
T_A	Operating Ambient Temperature Range	54 74	−55 0	25 25	125 70	°C
I_{OH}	Output Current — High	54, 74			−0.4	mA
I_{OL}	Output Current — Low	54 74			4.0 8.0	mA

DC CHARACTERISTICS OVER OPERATING TEMPERATURE RANGE (unless otherwise specified)

SYMBOL	PARAMETER		LIMITS MIN	LIMITS TYP	LIMITS MAX	UNITS	TEST CONDITIONS	
V_{IH}	Input HIGH Voltage		2.0			V	Guaranteed Input HIGH Voltage for All Inputs	
V_{IL}	Input LOW Voltage	54			0.7	V	Guaranteed Input LOW Voltage for All Inputs	
		74			0.8			
V_{IK}	Input Clamp Diode Voltage			−0.65	−1.5	V	V_{CC} = MIN, I_{IN} = −18 mA	
V_{OH}	Output HIGH Voltage	54	2.5	3.5		V	V_{CC} = MIN, I_{OH} = MAX, V_{IN} = V_{IH} or V_{IL} per Truth Table	
		74	2.7	3.5		V		
V_{OL}	Output LOW Voltage	54,74		0.25	0.4	V	I_{OL} = 4.0 mA	V_{CC} = V_{CC} MIN, V_{IN} = V_{IL} or V_{IH} per Truth Table
		74		0.35	0.5	V	I_{OL} = 8.0 mA	
I_{IH}	Input HIGH Current				20	μA	V_{CC} = MAX, V_{IN} = 2.7 V	
					0.1	mA	V_{CC} = MAX, V_{IN} = 7.0 V	
I_{IL}	Input LOW Current				−0.4	mA	V_{CC} = MAX, V_{IN} = 0.4 V	
I_{OS}	Short Circuit Current		−20		−100	mA	V_{CC} = MAX	
I_{CC}	Power Supply Current				10	mA	V_{CC} = MAX	

AC CHARACTERISTICS: T_A = 25°C

SYMBOL	PARAMETER	LEVEL OF DELAY	LIMITS MIN	LIMITS TYP	LIMITS MAX	UNITS	TEST CONDITIONS
t_{PLH} t_{PHL}	Propagation Delay Address to Output	2 2		13 27	20 41	ns	V_{CC} = 5.0 V C_L = 15 pF
t_{PLH} t_{PHL}	Propagation Delay Address to Output	3 3		18 26	27 39	ns	
t_{PLH} t_{PHL}	Propagation Delay $\overline{E}_1$ or $\overline{E}_2$ Enable to Output	2 2		12 21	18 32	ns	
t_{PLH} t_{PHL}	Propagation Delay E_3 Enable to Output	3 3		17 25	26 38	ns	

AC WAVEFORMS

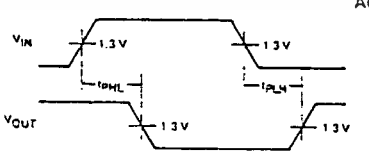


Fig. 1

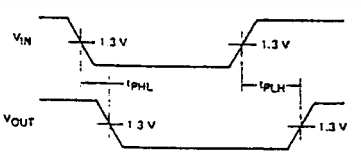


Fig. 2

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

MOTOROLA

SN54/74LS139

DUAL 1-OF-4-DECODER/ DEMULTIPLEXER

LOW POWER SCHOTTKY

DESCRIPTION — The LSTTL/MSI SN54LS/74LS139 is a high speed Dual 1-of-4 Decoder/Demultiplexer. The device has two independent decoders, each accepting two inputs and providing four mutually exclusive active LOW Outputs. Each decoder has an active LOW Enable input which can be used as a data input for a 4-output demultiplexer. Each half of the LS139 can be used as a function generator providing all four minterms of two variables. The LS139 is fabricated with the Schottky barrier diode process for high speed and is completely compatible with all Motorola TTL families.

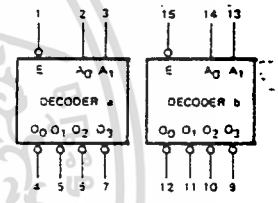
- SCHOTTKY PROCESS FOR HIGH SPEED
- MULTIFUNCTION CAPABILITY
- TWO COMPLETELY INDEPENDENT 1-OF-4 DECODERS
- ACTIVE LOW MUTUALLY EXCLUSIVE OUTPUTS
- INPUT CLAMP DIODES LIMIT HIGH SPEED TERMINATION EFFECTS

PIN NAMES

		LOADING (Note a) HIGH	LOW
A_0, A_1	Address Inputs	0.5 U.L.	0.25 U.L.
$\overline{E}$	Enable (Active LOW) Inout	0.5 U.L.	0.25 U.L.
$\overline{O}_0 - \overline{O}_3$	Active LOW Outputs (Note b)	10 U.L.	5 (2.5) U.L.

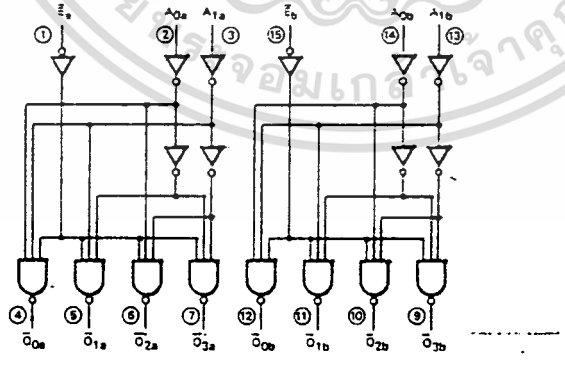
NOTES:
a. 1 TTL Unit Load (U.L.) = 40 µA HIGH/1.6 mA LOW.
b. The Output LOW drive factor is 2.5 U.L. for Military (54) and 5 U.L. for Commercial (74) Temperature Ranges.

LOGIC SYMBOL



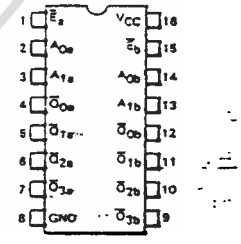
V_{CC} = Pin 16
GND = Pin 8

LOGIC DIAGRAM



V_{CC} = Pin 16
GND = Pin 8
○ = Pin Numbers

CONNECTION DIAGRAM
DIP (TOP VIEW)



J Suffix — Case 620-09 (Ceramic)
N Suffix — Case 648-08 (Plastic)

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

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

FUNCTIONAL DESCRIPTION — The LS139 is a high speed dual 1-of-4 decoder/demultiplexer fabricated with the Schottky barrier diode process. The device has two independent decoders, each of which accept two binary weighted inputs (A_0, A_1) and provide four mutually exclusive active LOW outputs ($\overline{O}_0$-$\overline{O}_3$). Each decoder has an active LOW Enable ($\overline{E}$). When $\overline{E}$ is HIGH all outputs are forced HIGH. The enable can be used as the data input for a 4-output demultiplexer application.

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

TRUTH TABLE

INPUTS			OUTPUTS			
$\overline{E}$	A_0	A_1	$\overline{O}_0$	$\overline{O}_1$	$\overline{O}_2$	$\overline{O}_3$
H	X	X	H	H	H	H
L	L	L	L	H	H	H
L	H	L	H	L	H	H
L	L	H	H	H	L	H
L	H	H	H	H	H	L

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

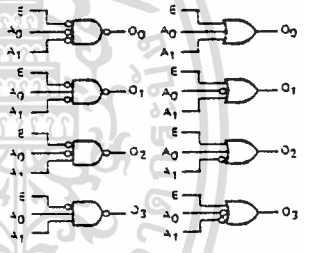


Fig. a

GUARANTEED OPERATING RANGES

SYMBOL	PARAMETER		MIN	TYP	MAX	UNIT
V_{CC}	Supply Voltage	54 74	4.5 4.75	5.0 5.0	5.5 5.25	V
T_A	Operating Ambient Temperature Range	54 74	−55 0	25 25	125 70	°C
I_{OH}	Output Current — High	54, 74			−0.4	mA
I_{OL}	Output Current — Low	54 74			4.0 8.0	mA

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

DC CHARACTERISTICS OVER OPERATING TEMPERATURE RANGE (unless otherwise specified)

SYMBOL	PARAMETER		LIMITS MIN	LIMITS TYP	LIMITS MAX	UNITS	TEST CONDITIONS	
V_{IH}	Input HIGH Voltage		2.0			V	Guaranteed Input HIGH Voltage for All Inputs	
V_{IL}	Input LOW Voltage	54			0.7	V	Guaranteed Input LOW Voltage for All Inputs	
		74			0.8			
V_{IK}	Input Clamp Diode Voltage			−0.65	−1.5	V	V_{CC} = MIN, I_{IN} = −18 mA	
V_{OH}	Output HIGH Voltage	54	2.5	3.5		V	V_{CC} = MIN, I_{OH} = MAX, V_{IN} = V_{IH} or V_{IL} per Truth Table	
		74	2.7	3.5		V		
V_{OL}	Output LOW Voltage	54,74		0.25	0.4	V	I_{OL} = 4.0 mA	V_{CC} = V_{CC} MIN, V_{IN} = V_{IL} or V_{IH} per Truth Table
		74		0.35	0.5	V	I_{OL} = 8.0 mA	
I_{IH}	Input HIGH Current				20	μA	V_{CC} = MAX, V_{IN} = 2.7 V	
					0.1	mA	V_{CC} = MAX, V_{IN} = 7.0 V	
I_{IL}	Input LOW Current				−0.4	mA	V_{CC} = MAX, V_{IN} = 0.4 V	
I_{OS}	Short Circuit Current		−20		−100	mA	V_{CC} = MAX	
I_{CC}	Power Supply Current				11	mA	V_{CC} = MAX	

AC CHARACTERISTICS: T_A = 25°C

SYMBOL	PARAMETER	LEVEL OF DELAY	LIMITS MIN	LIMITS TYP	LIMITS MAX	UNITS	TEST CONDITIONS
t_{PLH}	Propagation Delay	2		13	20	ns	V_{CC} = 5.0 V, C_L = 15 pF
t_{PHL}	Address to Output	2		22	33		
t_{PLH}	Propagation Delay	3		18	29	ns	
t_{PHL}	Address to Output	3		25	38		
t_{PLH}	Propagation Delay	2		16	24	ns	
t_{PHL}	Enable to Output	2		21	32		

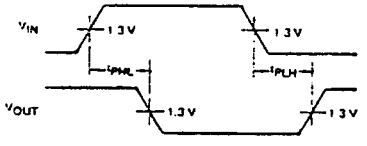


Fig. 1

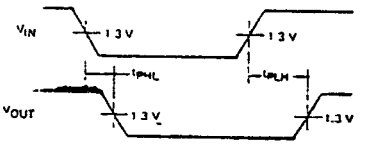


Fig. 2

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

MOTOROLA

SN54/74LS373
SN54/74LS374

OCTAL TRANSPARENT LATCH WITH 3-STATE OUTPUTS; OCTAL D-TYPE FLIP-FLOP WITH 3-STATE OUTPUT

LOW POWER SCHOTTKY

DESCRIPTION — The SN54LS/74LS373 consists of eight latches with 3-state outputs for bus organized system applications. The flip-flops appear transparent to the data (data changes asynchronously) when Latch Enable (LE) is HIGH. When LE is LOW, the data that meets the setup times is latched. Data appears on the bus when the Output Enable ($\overline{OE}$) is LOW. When $\overline{OE}$ is HIGH the bus output is in the high impedence state.

The SN54LS/74LS374 is a high-speed, low-power Octal D-type Flip-Flop featuring separate D-type inputs for each flip-flop and 3-state outputs for bus oriented applications. A buffered Clock (CP) and Output Enable (OE) is common to all flip-flops. The SN54LS/74LS374 is manufactured using advanced Low Power Schottky technology and is compatible with all Motorola TTL families.

- EIGHT LATCHES IN A SINGLE PACKAGE
- 3-STATE OUTPUTS FOR BUS INTERFACING
- HYSTERESIS ON LATCH ENABLE
- EDGE-TRIGGERED D-TYPE INPUTS
- BUFFERED POSITIVE EDGE-TRIGGERED CLOCK
- HYSTERESIS ON CLOCK INPUT TO IMPROVE NOISE MARGIN
- INPUT CLAMP DIODES LIMIT HIGH SPEED TERMINATION EFFECTS

PIN NAMES

		LOADING (Note a) HIGH	LOW
D_0-D_7	Data Inputs	0.5 U.L.	0.25 U.L.
LE	Latch Enable (Active HIGH) Input	0.5 U.L.	0.25 U.L.
CP	Clock (Active HIGH going edge) Input	0.5 U.L.	0.25 U.L.
$\overline{OE}$	Output Enable (Active LOW) Input	0.5 U.L.	0.25 U.L.
O_0—O_7	Outputs (Note b)	65(25) U.L.	15(7.5) U.L.

NOTES:
a. 1 TTL Unit Load (U.L.) = 40 μA HIGH/1.6 mA LOW.
b. The Output LOW drive factor is 7.5 U.L. for Military and 25 U.L. for Commercial (74) Temperature Ranges. The Output HIGH drive factor is 25 U.L. for Military (54) and 65 U.L. for Commercial (74) Temperature Ranges.

TRUTH TABLE

LS373

Dn	LE	$\overline{OE}$	On
H	H	L	H
L	H	L	L
X	X	H	Z*

LS374

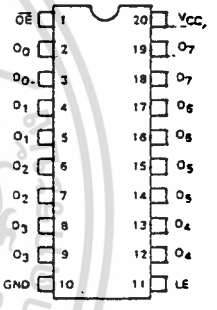
Dn	CP	$\overline{OE}$	On
H	↑	L	H
L	↑	L	L
X	X	H	Z*

H = HIGH Voltage Level
L = LOW Voltage Level
X = Immaterial
Z = High Impedence

*Note: Contents of flip-flops unaffected by the state of the Output Enable input ($\overline{OE}$)

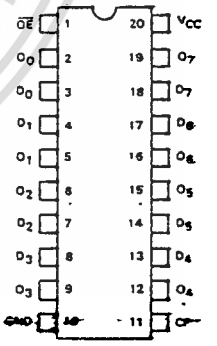
CONNECTION DIAGRAM DIP (TOP VIEW)

SN54LS/74LS373



CONNECTION DIAGRAM DIP (TOP VIEW)

SN54LS/74LS374



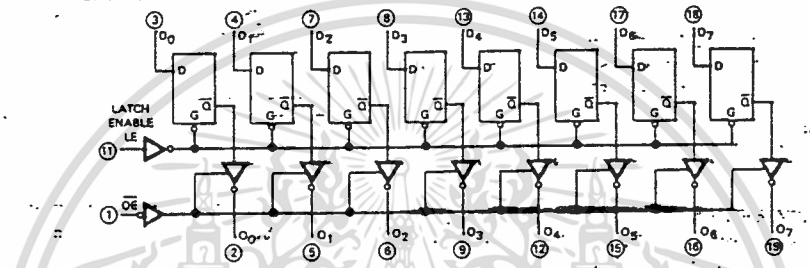
J Suffix — Case 732-03 (Ceramic)
N Suffix — Case 738-03 (Plastic)

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

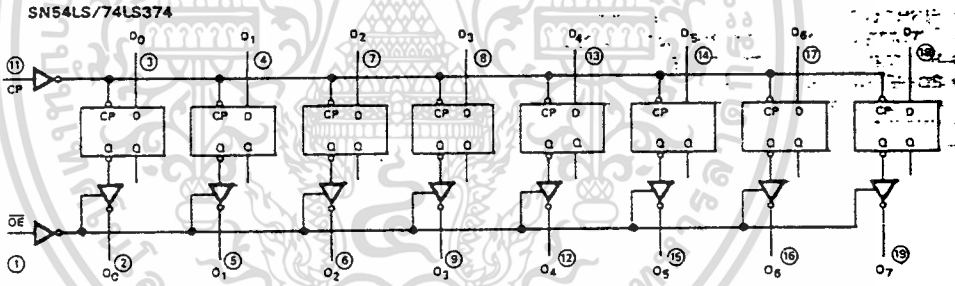
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

LOGIC DIAGRAMS

SN54LS/74LS373



SN54LS/74LS374



V_{CC} = Pin 20
GND = Pin 10
○ = Pin Numbers

GUARANTEED OPERATING RANGES

SYMBOL	PARAMETER		MIN	TYP	MAX	UNIT
V_{CC}	Supply Voltage	54 74	4.5 4.75	5.0 5.0	5.5 5.25	V
T_A	Operating Ambient Temperature Range	54 74	−55 0	25 25	125 70	°C
I_{OH}	Output Current — High	54 74			−1.0 −2.6	mA
I_{OL}	Output Current — Low	54 74			12 24	mA

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

SCL4518B
SCL4520B

CMOS DUAL UP COUNTERS

FEATURES

- Two Independent 4-Bit Counters
- Internally Synchronous for High Speed
- Dual BCD (SCL4518B) and Dual Binary (SCL4520B) Configurations
- Direct Reset
- Logic Edge-Clocked Design
- Trigger from either Edge of Clock Signal
- Static Operation— DC to 5MHz @ 10Vdc

DESCRIPTION

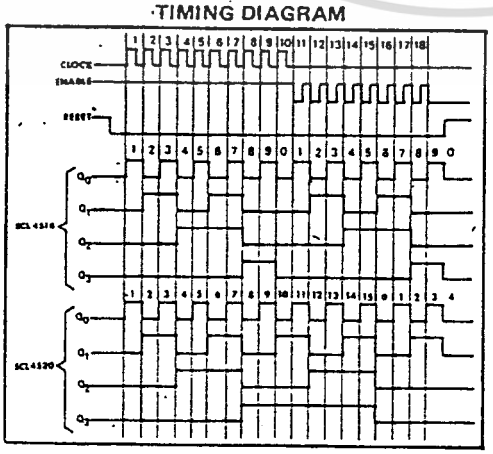
The SCL4518B Dual BCD Counter and the SCL4520B Dual Binary Counter are constructed with MOS P-channel and N-channel enhancement-mode devices in a single monolithic structure. Each consists of two identical, independent, internally synchronous 4-stage counters. The counter stages are type-D flip-flops, with interchangeable Clock and Enable lines for incrementing on either the positive-going or negative-going transition as required when cascading multiple stages. Each counter can be cleared by applying a high level on the Reset line. In addition, the SCL4518B will count out of all undefined states within two clock periods. These complementary MOS up counters find primary use in multi-stage synchronous or ripple counting applications requiring low power dissipation and/or high noise immunity.

TRUTH TABLE

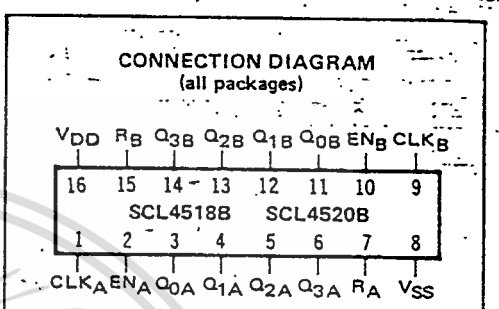
CLOCK	ENABLE	RESET	ACTION
⎍	1	0	Increment Counter
0	⎍	0	Increment Counter
⎍	X	0	No Change
X	⎍	0	No Change
⎍	0	0	No Change
1	⎍	0	No Change
X	X	1	Q0 thru Q3 = 0

X = Don't Care

TIMING DIAGRAM



CONNECTION DIAGRAM
(all packages)



Add suffix for package:

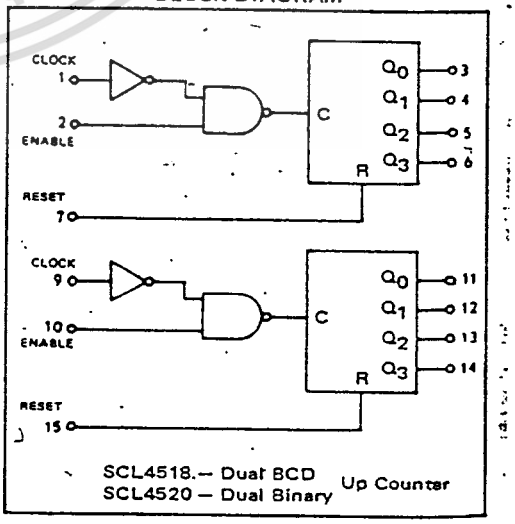
- C 16-pin Cerdip
- D 16-pin Ceramic
- E 16-pin Epoxy
- F 16-pin Flat
- H Chip

RECOMMENDED OPERATING CONDITIONS

For maximum reliability:

DC Supply Voltage	$V_{DD} - V_{SS}$	3 to 15	Vdc
Operating Temperature	T_A		
C, D, F, H Device		-55 to +125	°C
E Device		-40 to +85	°C

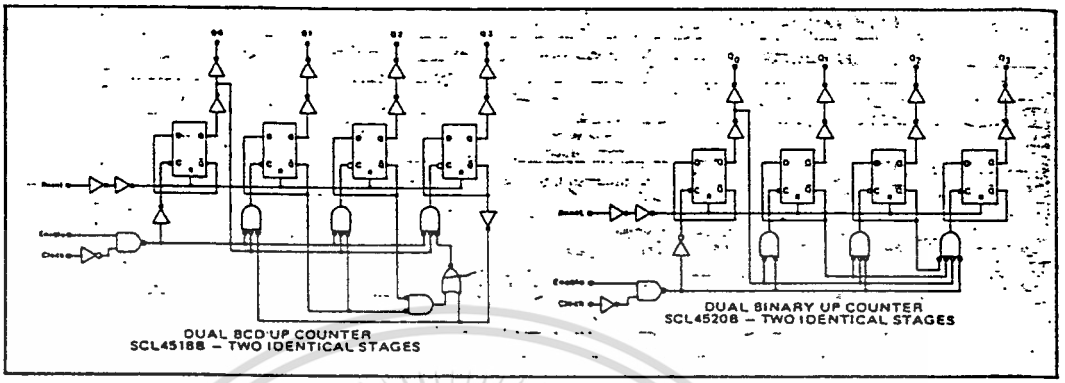
BLOCK DIAGRAM



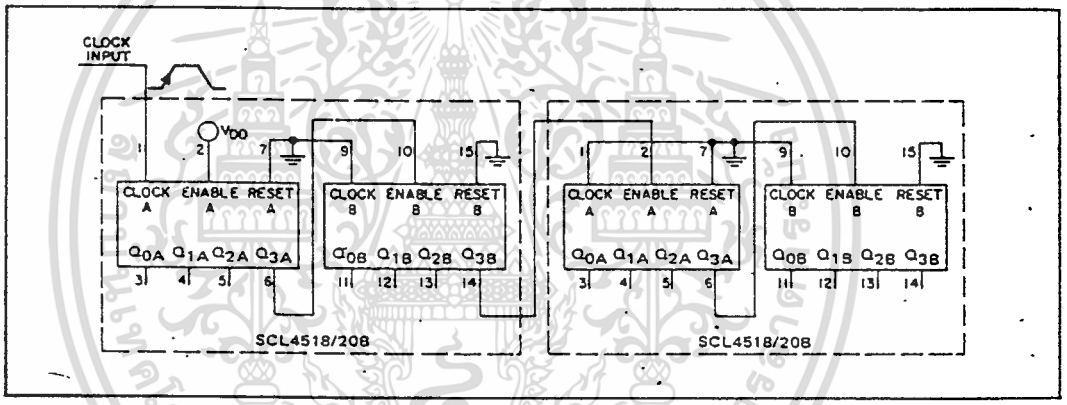
SCL4518 — Dual BCD
SCL4520 — Dual Binary
Up Counter

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

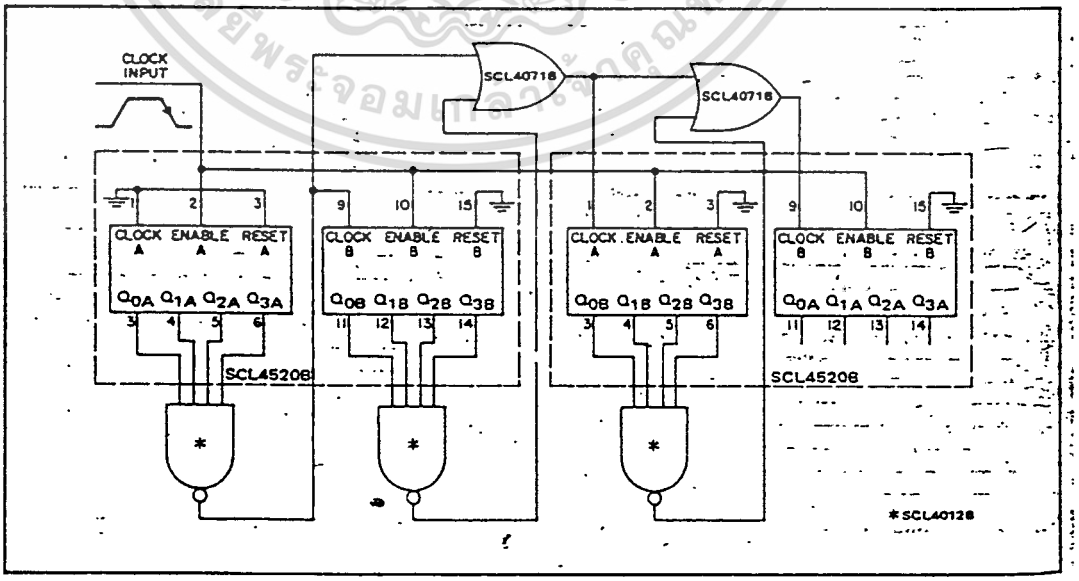
LOGIC DIAGRAMS



APPLICATIONS INFORMATION



Ripple cascading of four counters with positive-edge triggering.



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ELECTRICAL CHARACTERISTICS

STATIC CHARACTERISTICS[1]

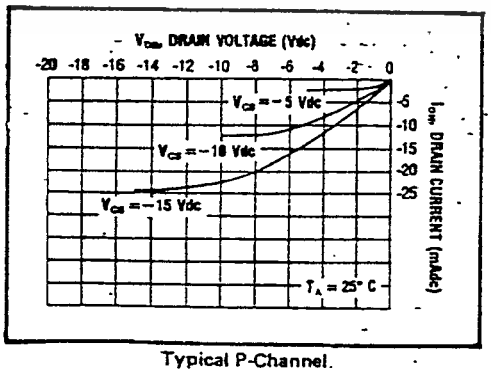
PARAMETER		V_{DD} (Vdc)	CONDITIONS	T_{LOW}[3] Min.	T_{LOW}[3] Max.	+25°C Min.	+25°C Typ.	+25°C Max.	T_{HIGH}[3] Min.	T_{HIGH}[3] Max.	Units
QUIESCENT DEVICE CURRENT	I_{DD}	5	$V_{IN} = V_{SS}$ or V_{DD}	–	5	–	0.05	5	–	150	μAdc
		10	All valid input	–	10	–	0.1	10	–	300	
		15	combinations	–	20	–	0.2	20	–	600	

NOTES: [1] Remaining Static Electrical Characteristics are listed under "SCL40008 Series Family Specifications".
[2] T_{LOW} = -55°C for C, D, F, H device.
= -40°C for E device.
T_{HIGH} = +125°C for C, D, F, H device.
= + 85°C for E device.

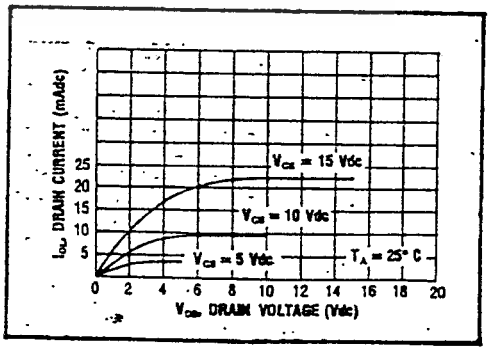
DYNAMIC CHARACTERISTICS (C_L = 50pF, T_A = 25°C)

PARAMETER		V_{DD} (Vdc)	Min.	Typ.	Max.	Units
CLOCKED OPERATION						
PROPAGATION DELAY TIME From Clock or Clock Enable	t_{PLH}, t_{PHL}	5	–	225	450	ns
		10	–	100	200	
		15	–	80	160	
OUTPUT TRANSITION TIME	t_{TLH}, t_{THL}	5	–	130	260	ns
		10	–	65	130	
		15	–	50	100	
MINIMUM CLOCK PULSE WIDTH	PW_{CL}	5	–	120	240	ns
		10	–	50	100	
		15	–	40	80	
MINIMUM CLOCK ENABLE PULSE WIDTH	PW_{CE}	5	–	200	400	ns
		10	–	90	180	
		15	–	75	150	
MAXIMUM CLOCK FREQUENCY	f_{CL}	5	1.0	2.0	–	MHz
		10	2.5	5.0	–	
		15	3.0	6.0	–	
MAXIMUM CLOCK OR CLOCK ENABLE RISE & FALL TIME[1]	t_{rCL}, t_{fCL}	5	15	–	–	μs
		10	5	–	–	
		15	3	–	–	
RESET OPERATION						
PROPAGATION DELAY TIME	t_{PHL}	5	–	225	450	ns
		10	–	100	200	
		15	–	80	160	
MINIMUM RESET PULSE WIDTH	PW_R	5	–	120	240	ns
		10	–	50	100	
		15	–	40	80	
RESET REMOVAL TIME	t_{rem}	5	–	100	200	ns
		10	–	50	100	
		15	–	40	80	

[1] When units are cascaded, the maximum rise and fall times of the clock input should be equal to or less than the transition times of the data outputs driving data inputs, plus the propagation delay of the output driving stage for the output capacitive load.



Typical P-Channel
Source Current Characteristics



Typical N-Channel
Sink Current Characteristics

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1. Application Engineering Staff "OP.TO ELECTRONICS/FIBER-OPTIC APPLICATION MANUAL" , Hewlet pachard company, 1977.
2. Klaus Gillessen, Werner Schairer "LIGHT EMITTING DIODES AND INTRODUCTION", Prentice - hall international. Engle cliffs, Newjersey, 1986.
3. บริษัท สกายบุ๊ก จำกัด "Visual Basic for windows 3.0" , สกายบุ๊กส์ จำกัด, 2538
4. จิติ หนูแก้ว "เทคนิคการเชื่อมต่อ IBM PC กับอุปกรณ์ภายนอก" , ซีเอ็ดยูเคชั่น จำกัด, 2534
5. ผศ. พิพัฒน์ เลาหสงคราม "ไมโครคอนโทรลเลอร์" , ภาควิชาเทคโนโลยีการวัดคุมทางอุตสาหกรรม คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีเจ้าคุณทหารลาดกระบัง, 2537
6. Michael Thscher "PC INTERN System programming" , Abacus ,1992
7. ยืน ภู่วรวรรณ "เทคโนโลยี ไมโครคอมพิวเตอร์ 16 บิต", ซีเอ็ดยูเคชั่น จำกัด, 2531

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้