ระบบควบคุมการเข้าออกโดยใช้บัตรแม่เหล็ก

Access Control System Using Magnetic Card

โดย

นายจุฬชาติ จงอยู่สุข 37014066

นายเริงรุจ รุจนะไกรกานต์ 37014364

อาจารย์ที่ปรึกษา

อาจารย์เกียรติวรรณ ทรงสัตย์

วัน เดือน ปี 11 ธค. 2541

เลขทะเบียน 038873

เลขเรียกหนังสือ T 10112 ว148ร

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต

สาขาวิศวกรรมระบบควบคุม

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2540

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

038873

ปริญญานิพนธ์ปีการศึกษา 2540

ภาควิชาวิศวกรรมระบบควบคุม

คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เรื่อง ระบบควบคุมการเข้าออกโดยใช้บัตรแม่เหล็ก

Access Control System Using Magnetic Card

ผู้จัดทำ

1. นายจุฬชาติ จงอยู่สุข
2. นายเริงรุจ รุจนะไกรกานต์

_____ อาจารย์ที่ปรึกษา

(อาจารย์เกียรติวรรณ ทรงสัตย์)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ระบบควบคุมการเข้าออกโดยใช้บัตรแม่เหล็ก

Access Control System Using Magnetic Card

โดย

นายจุฬชาติ จงอยู่สุข

รหัส 37014066

นายเริงรุจ รุจนะไกรกานต์

รหัส 37014364

อาจารย์ที่ปรึกษา

อาจารย์เกียรติวรรณ ทรงสัตย์

บทคัดย่อ

ปริญญานิพนธ์ฉบับนี้เป็นการออกแบบ สร้างและพัฒนา ระบบควบคุมการเข้าออกโดยใช้บัตรแม่เหล็ก โครงสร้างของระบบประกอบไปด้วยหน่วยประมวลผล 2 หน่วย ได้แก่ เซิร์ฟเวอร์และเทอร์มินอล เชื่อมต่อกันเป็นระบบโครงข่ายแบบบัสทั้งระบบ เซิร์ฟเวอร์ทำหน้าที่ควบคุมการผ่านเข้าออกแบบเบ็ดเสร็จผ่านโครงข่าย RS-485 ซึ่งสามารถเชื่อมต่อกับเทอร์มินอลได้ถึง 32 หน่วย เทอร์มินอลแต่ละหน่วยทำหน้าที่อ่านบัตรแม่เหล็กเพื่อนำข้อมูลและเวลาผ่านเข้าออกของผู้ใช้ส่งไปยังเซิร์ฟเวอร์เพื่อทำการตรวจสอบความปลอดภัยแบบหลายระดับและบันทึกลงในหน่วยความจำของเซิร์ฟเวอร์ โดยโปรโตคอลที่ใช้ในการสื่อสารข้อมูลได้ยึดรูปแบบของ HDLC (High Level Data Link Control) เป็นแนวทาง ในส่วนของหน่วยประมวลผลได้ใช้ไมโครคอนโทรลเลอร์ในตระกูล MCS-51 เป็นหน่วยประมวลผลกลาง

ABSTRACT

This thesis presents a design and development of the Access Control System. The system comprises of two units, server module and terminal module connected in bus-wired system. The server is a centralized host that hooks maximum up to 32 terminals via RS-485. The terminal is the interface part of the system that starts the operation of multilevel security when a magnetic card is slid the Personal Identity Number (PIN) is sent to server. If authorization is correct, PIN number and access time are registered. The protocal used as the rule of data communication in this specific system is based on the HDLC (High-level Data Link Control) Protocal. And the microcontroller MCS-51 family is selected to be the processor units of both server and terminal modules by the feature of an enhanced serial port for multi-processor communications.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กิตติกรรมประกาศ

ขอบพระคุณอาจารย์เกียรติวรรณ ทรงสัตย์ เป็นอย่างสูงที่เป็นอาจารย์ที่ปรึกษา และได้ให้ความรู้ คำแนะนำ ความช่วยเหลือต่างๆ จนทำให้ปริญญานิพนธ์ฉบับนี้สำเร็จลุล่วงไปได้ด้วยดี

ขอบพระคุณคณาจารย์ทุกๆ ท่าน ที่ได้ประสิทธิประสาทวิชาความรู้แก่ผู้จัดทำมาโดยตลอด

ขอบพระคุณคุณพ่อ คุณแม่ ที่ท่านได้ให้การเลี้ยงดู อบรม สั่งสอน และให้กำลังใจมาโดยตลอด

ขอบคุณพี่ๆ และน้องๆ ที่ให้กำลังใจ แนะนำในสิ่งที่ดีในทุกๆ ด้าน

ขอบคุณเพื่อนๆ ร่วมสถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง และโดยเฉพาะอย่างยิ่งเพื่อนๆ ภาควิชาวิศวกรรมระบบควบคุม รุ่น 33 ทุกๆ คน ที่ให้ความช่วยเหลือ ด้านเครื่องมือเครื่องใช้ คำแนะนำ กำลังใจ และความเป็นเพื่อนที่ดีตลอดมา

และสุดท้ายขอขอบคุณภาควิชาวิศวกรรมระบบควบคุม คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง ที่ให้โอกาสผู้จัดทำได้เข้ามาศึกษาเล่าเรียน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ

บทคัดย่อ I

กิตติกรรมประกาศ II

สารบัญ III

สารบัญตาราง V

สารบัญภาพ VI

บทที่ 1 บทนำ 1

กล่าวนำ 1

วัตถุประสงค์ 1

หลักการที่ใช้ 2

บทที่ 2 หลักการและทฤษฎี 7

2.1 บัตรแม่เหล็ก 7

2.1.1 คุณสมบัติของบัตรแม่เหล็กตามมาตรฐาน ISO 7

2.1.2 ชุดรหัสข้อมูลในแทร็คที่สอง 8

2.1.3 รูปแบบของข้อมูลที่บันทึกในแทร็คที่สองบนบัตรแม่เหล็ก 10

2.1.4 การอ่านข้อมูลจากบัตรแม่เหล็ก 10

2.2 การสื่อสารข้อมูลแบบอนุกรมโดยใช้หลักการของ Multiprocessor Communications 18

2.3 HDLC Protocol 19

2.3.1 HDLC Protocol (High-level Data Link Control Protocol) 19

2.3.2 รูปแบบของเฟรม (Frame format) 22

2.3.3 ลักษณะการส่งเฟรมโต้ตอบของโปรโตคอล 25

2.4 มาตรฐานการสื่อสารข้อมูลอนุกรม RS-232 31

2.5 มาตรฐานการสื่อสารข้อมูลอนุกรม RS-485 32

บทที่ 3 โครงสร้างของระบบ 34

3.1 การออกแบบสร้างและพัฒนาส่วนประมวลผลของระบบ 34

3.1.1 เทอร์มินอลโมดูล (Terminal Module) 34

3.1.2 เซิร์ฟเวอร์โมดูล (Server Module) 36

3.1.3 วงจรไฟเลี้ยง (Power Supply) 38

3.2 การพัฒนาโปรแกรมจัดการและควบคุมระบบ 43

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ (ต่อ)

โปรแกรมสนับสนุนการติดต่อกับคอมพิวเตอร์และการติดต่อกับผู้ใช้ของส่วนเซิร์ฟเวอร์ 60

บทที่ 4 บทสรุปและวิจารณ์ 71

บทที่ 5 แนวทางการพัฒนา 73

เอกสารอ้างอิง 74

ภาคผนวก ก. แผ่นพิมพ์ลายวงจรและภาพถ่ายวงจร 75

ภาคผนวก ข. ข้อมูลทางเทคนิคของอุปกรณ์ที่สำคัญบนบอร์ด 79

รายละเอียดของ MAX485 80

รายละเอียดของ 75176 91

รายละเอียดของ DS1232 หรือ MAX1232 101

ภาคผนวก ค. โปรแกรมการจัดการและควบคุมระบบควบคุมการเข้าออกโดยใช้บัตรแม่เหล็ก 108

โปรแกรมจัดการและควบคุมระบบส่วนเทอร์มินอลโมดูล 109

โปรแกรมจัดการและควบคุมระบบส่วนเซิร์ฟเวอร์โมดูล 119

โปรแกรมจัดการและควบคุมระบบส่วนไมโครคอมพิวเตอร์โดยใช้ภาษาปาสคาล 136

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญตาราง

ตารางที่	หน้า
2.1 แสดงรหัสข้อมูลตัวเลขสำหรับแทร็คที่สอง	9
2.2 แสดงตัวอย่างข้อมูลจากบัตรแม่เหล็กของธนาคารกรุงเทพ	14
2.3 แสดงตัวอย่างข้อมูลจากบัตรแม่เหล็กของธนาคารไทยพาณิชย์	16
2.4 แสดง Control Field	28

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญภาพ

รูปที่	หน้า
1.1 แสดงส่วนประกอบทางด้านฮาร์ดแวร์ (Hardware)	2
1.2 แสดงการเชื่อมต่อโครงข่ายแบบ BUS Topology	3
1.3 แสดงการเชื่อมต่อโครงข่ายแบบ RING Topology	4
1.4 แสดงการเชื่อมต่อโครงข่ายแบบ STAR Topology	4
1.5 แสดงการเชื่อมต่อโครงข่ายแบบสมบูรณ์ (Fully Connected Topology)	4
1.6 แสดงการเชื่อมต่อโครงข่ายแบบผสม (Combined Topology)	5
1.7 แสดง OSI Layered Protocol Model	6
2.1 แสดงตำแหน่งแทร็คที่สองของบัตรแม่เหล็กตามมาตรฐาน ISO	7
2.2 แสดงรูปแบบของข้อมูลที่บันทึกในแทร็คที่สองบนบัตรแม่เหล็ก	10
2.3 แสดงตัวอ่านบัตรแม่เหล็ก	10
2.4 แสดงแผนภาพของวงจรอ่านบัตรแม่เหล็กในตัวอ่านบัตรแม่เหล็ก	11
2.5 แสดงรูปคลื่นสัญญาณของ ISO Pulse ที่ถูกดีมอดดูเลท	11
2.6 แสดงวงจรอ่านบัตรแม่เหล็กโดยใช้ชิพ MC 54910 P	12
2.7 แสดงสัญญาณเอาท์พุทของ MC 54910 P	13
2.8 แสดงระบบอสมดุลแบบ Point-to-point	21
2.9 แสดงระบบอสมดุลแบบ Multi-point	21
2.10 แสดงระบบสมดุล (Point-to-point)	21
2.11 แสดงแบบเฟรมของ HDLC Protocol	22
2.12 รูป (ก) เขตควบคุม 8 บิท และ (ข) เขตควบคุม 16 บิท	23
2.13 แสดงรหัสชุดคำสั่งและคำตอบรับ	24
2.14 แสดงรูปแบบโปรโตคอลที่ได้ออกแบบไว้	27
2.15 แสดงจังหวะการรับส่งข้อมูลระหว่างเซิร์ฟเวอร์กับเทอร์มินอล	30
2.16 แสดง RS-232 Connector	31
2.17 แสดงการส่งและการรับสัญญาณระหว่าง DTE และ DCE ในมาตรฐาน RS-232	32
2.18 แสดงการเชื่อมต่อของ RS-422-A	33
2.19 แสดงการเชื่อมต่อของ RS-485	33
3.1 แสดงระบบควบคุมการเข้าออกโดยใช้บัตรแม่เหล็ก	34
3.2 แสดงรูปวงจรของส่วนรับส่งข้อมูลอนุกรม RS-485	35

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญภาพ (ต่อ)

รูปที่	หน้า
3.3 แสดงวงจรขับรีเลย์เพื่อสั่งเปิดหรือปิด	36
3.4 แสดงวงจรของเทอร์มินอล	37
3.5 แสดงการเชื่อมต่อสวนแสดงผลของเซิร์ฟเวอร์	38
3.6 แสดงวงจรของเซิร์ฟเวอร์	39
3.7 แสดงวงจรของเซิร์ฟเวอร์โดยใช้บอร์ด ANT-C51	40
3.8 แสดงการเชื่อมต่อ CPU เข้ากับโครงข่าย RS-485	41
3.9 แสดงวงจรไฟเลี้ยง (Power Supply) และรีเลย์ (Relay)	42
3.10 แสดงโฟล์วชาร์ตของโปรแกรมหลักของเทอร์มินอลโมดูล	44
3.11 แสดงโฟล์วชาร์ตของโปรแกรมบริการการรูดบัตร (โปรแกรมย่อย MAG_RD)	45
3.12 แสดงโฟล์วชาร์ตโปรแกรมย่อย CARDCL	46
3.13 แสดงโฟล์วชาร์ตโปรแกรมย่อย CARDRD	47
3.14 แสดงโพล์วชาร์ตโปรแกรมย่อย CARDAJ	48
3.15 แสดงโฟล์วชาร์ตโปรแกรมย่อย CARDCK	49
3.16 แสดงโฟล์วชาร์ตโปรแกรมย่อย CARDAX	50
3.17 แสดงโฟล์วชาร์ตโปรแกรมย่อย CARDRE	51
3.18 แสดงโฟล์วชาร์ตโปรแกรมบริการการขอใช้ NETWORK (LINK)	52
3.19 แสดงโฟล์วชาร์ตโปรแกรมการแปลความหมาย (EDIT)	53
3.20 แสดงโฟล์วชาร์ตโปรแกรมหลักของเซิร์ฟเวอร์โมดูล	54
3.21 แสดงโฟล์วชาร์ตโปรแกรมบริการการเพิ่ม USER และโปรแกรมการสร้างฐานข้อมูล	55
3.22 แสดงโฟล์วชาร์ตโปรแกรมบริการ USER	56
3.23 แสดงโฟล์วชาร์ตโปรแกรมบริการการค้นหาข้อมูล	57
3.24 แสดง MEMORY MAP ของหน่วยความจำภายนอกขนาด 16 KBYTE ของเซิร์ฟเวอร์	58
3.25 แสดง MEMORY MAP ของหน่วยความจำภายนอกขนาด 4 BYTE บน	59
3.26 แสดงแผนภาพการเชื่อมต่อ (Interface) ระหว่างผู้ใช้ (User) กับไมโครคอมพิวเตอร์และเซิร์	60
3.27 แสดงเมนูของโปรแกรมที่ใช้ในการเชื่อมต่อกับเซิร์ฟเวอร์โดยไมโครคอมพิวเตอร์	61
3.28 แสดงหน้าจอของเมนู Add New User	62
3.29 แสดงหน้าจอของการใส่ข้อมูลของผู้ใช้คนใหม่	63
3.30 แสดงหน้าจอเตรียมพร้อมเพื่อการรูดบัตรและบันทึกข้อมูล	64

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญภาพ (ต่อ)

รูปที่	หน้า
3.31 แสดงหน้าจอของเมนู DOWNLOAD DATA	65
3.32 แสดงหน้าจอการดาวน์โหลดข้อมูลจากไฟล์	66
3.33 แสดงภาพการแสดงผลและติดตามการส่งข้อมูล	67
3.34 แสดงหน้าจอของเมนู Broadcast	67
3.35 แสดงหน้าจอของเมนูการลบข้อมูลผู้ใช้ออกจากฐานข้อมูล	68
3.36 แสดงรายละเอียดของโปรแกรม (Version)	69
3.37 แสดงหน้าจอการหน่วงเวลา	69
3.38 แสดงหน้าจอของเมนู Upload Byte	70
ก.1 แสดงลายวงจรด้านหน้าของเทอร์มินอลโมดูล	76
ก.2 แสดงลายวงจรด้านหลังของเทอร์มินอลโมดูล	76
ก.3 แสดงรูปถ่ายของเทอร์มินอลโมดูล	77
ก.4 แสดงรูปถ่ายของเซิร์ฟเวอร์โมดูล	77
ก.5 แสดงรูปถ่ายของตัวอ่านบัตรแม่เหล็ก	78
ก.6 แสดงรูปถ่ายของชุดวงจรไฟเลี้ยง (Power Supply) และรีเลย์ (Relay)	78

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 1
บทนำ

กล่าวนำ

ปัจจุบันคอมพิวเตอร์มีบทบาทสำคัญในการทำงานของทุก ๆ หน่วยงานในด้านต่าง ๆ ซึ่งมีประโยชน์ในด้านการตรวจสอบข้อมูล,การประมวลผล,การตรวจสอบความปลอดภัย,การทำงานร่วมกันภายใต้ทรัพยากรเดียวกัน ฯลฯ ซึ่งจะทำให้เกิดความสะดวกและรวดเร็วมากขึ้นในการทำงาน

ในการทำวิทยานิพนธ์ฉบับนี้ นำเสนอระบบควบคุมการเข้าออกโดยใช้บัตรแม่เหล็ก (Access Control System Using Magnetic Card) ซึ่งจะมีประโยชน์ในด้านการตรวจสอบข้อมูลการเข้าออกของระบบโดยใช้คอมพิวเตอร์ และความรู้ทางด้านไมโครคอนโทรลเลอร์มาประยุกต์ใช้งาน สามารถนำมาใช้กับระบบที่ออกแบบไว้โดยใช้กับการสื่อสารข้อมูลแบบอนุกรมเป็นระบบโครงข่าย พร้อมทั้งนำบัตรแม่เหล็ก (Magnetic Card) มาประยุกต์ใช้งานด้วย โดยอาศัยข้อมูลที่บันทึกอยู่ภายในบัตรแม่เหล็กของแต่ละบุคคล (PIN : Personal Identity Number) เป็นข้อมูลประจำตัวที่จะใช้ในการตรวจสอบข้อมูลของระบบ ซึ่งจะต้องมีความยืดหยุ่นและสามารถพัฒนากับระบบควบคุมการเข้าออกที่เราออกแบบเอาไว้

วัตถุประสงค์

การทำวิทยานิพนธ์เรื่องระบบควบคุมการเข้าออกโดยใช้บัตรแม่เหล็ก (Access Control System Using Magnetic Card) นี้ได้กำหนดวัตถุประสงค์ไว้ดังนี้คือ

1. ศึกษาและทำการออกแบบ สร้าง ระบบควบคุมการเข้าออก โดยใช้การสื่อสารข้อมูลแบบอนุกรมของ MCS-51 ไมโครคอนโทรลเลอร์
2. พัฒนาระบบโครงข่าย RS-485 แบบ Half-Duplex ซึ่งรับส่งข้อมูลด้วยคู่สายสัญญาณรับส่งเพียงสองเส้น เพื่อใช้ในการสื่อสารข้อมูลอนุกรมภายในระบบ
3. สามารถนำระบบการสื่อสารแบบโครงข่ายมาประยุกต์ใช้งานภายในระบบได้อย่างเหมาะสมภายใต้ทรัพยากรอันจำกัด
4. พัฒนาระบบโดยนำบัตรแม่เหล็ก (Magnetic Card) มาใช้ในระบบควบคุม เพื่ออาศัยข้อมูลที่บันทึกอยู่ภายในบัตรแม่เหล็กมาทำการตรวจสอบการเข้าออก
5. ออกแบบและพัฒนาโปรแกรมสำหรับใช้ในระบบควบคุมซึ่งอาศัยหลักการทำงานของ MCS-51 ไมโครคอนโทรลเลอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หลักการที่ใช้

สำหรับระบบควบคุมการเข้าออกโดยใช้บัตรแม่เหล็กนี้จะใช้หลักการของ

1. บัตรแม่เหล็ก , การอ่านข้อมูลจากบัตรแม่เหล็กไปใช้งาน
2. การสื่อสารข้อมูลแบบอนุกรมโดยใช้หลักการของ Multiprocessor Communication
3. HDLC Protocol (High - level Data Link Protocol)
4. มาตรฐานการสื่อสารข้อมูลอนุกรม RS-232
5. มาตรฐานการสื่อสารข้อมูลอนุกรม RS-485

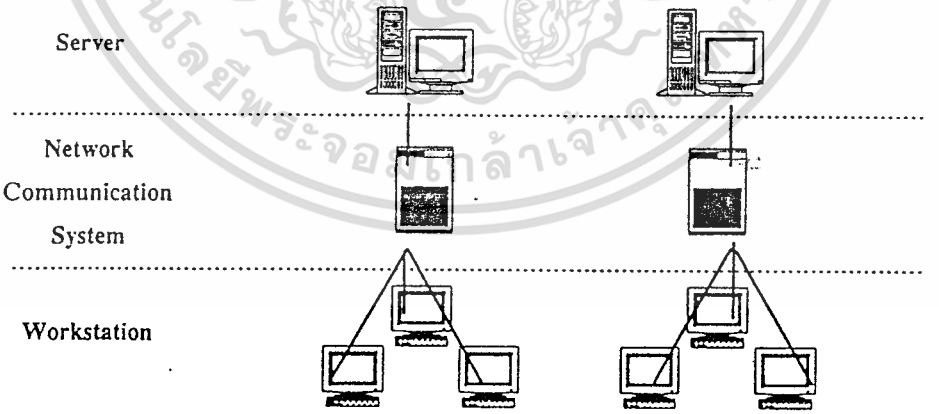
ส่วนประกอบและประโยชน์ของโครงข่าย (Network)

ในระบบโครงข่ายทั่วไปสามารถแบ่งเป็นส่วนประกอบหลักๆ ได้ดังนี้

- Server
- Network Communication System
- Workstation
- Network Operating System

ซึ่งจะแยกอธิบายเป็น 2 ส่วนคือ ส่วนฮาร์ดแวร์ (Hardware) และส่วนซอร์ฟแวร์ (Software)

1. <u>ส่วนฮาร์ดแวร์</u> (<u>Hardware</u>)



รูปที่ 1.1 แสดงส่วนประกอบทางด้านฮาร์ดแวร์ (Hardware)

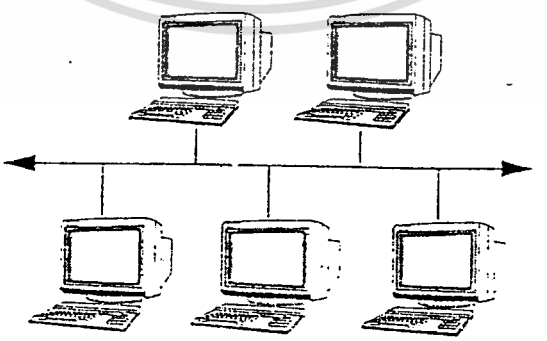
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เซิร์ฟเวอร์ (SERVER) มีหลายชนิด แต่ถ้านำเอาเครื่องไมโครคอมพิวเตอร์ทั่วๆ ไปมาใช้ เราจะเรียกว่า "File Server" File Server จะเป็นเครื่องคอมพิวเตอร์ที่ใช้เก็บข้อมูลและโปรแกรมที่ใช้งานในระบบ Network รวมทั้งควบคุมการทำงานของระบบให้เป็นไปอย่างถูกต้อง File Server มักนิยมใช้เครื่องไมโครคอมพิวเตอร์ที่อยู่ในระดับตั้งแต่ 80486 ขึ้นไป โดยมีหน่วยความจำ (RAM) ไม่น้อยกว่า 8 เมกกะไบท์ (MB) และมีฮาร์ดดิสค์ (Harddisk) ที่มีความจุสูง อย่างไรก็ดี ความเร็วการทำงานของหน่วยประมวลผลกลาง (Central Processing Unit : CPU) จะวัดเป็นเมกกะเฮิร์ต (MHz) และความเร็วของฮาร์ดดิสค์ (Harddisk) เราเรียกว่า "Access time" ยิ่งมีมากเท่าใดก็มีผลต่อประสิทธิภาพของโครงข่ายมากเท่านั้น

เวิร์คสเตชั่น (Workstation) ในที่นี้หมายถึงส่วนเทอร์มินอล (Terminal) ต่างๆ ของโครงข่าย ซึ่งอาจเป็นเครื่องไมโครคอมพิวเตอร์ (PC) หรือ ชุดการทำงานต่างๆ เช่น เครื่องตอกเวลา เครื่องรูดบัตรแม่เหล็ก เป็นต้น ถ้าเป็น PC ทั่วๆ ไป เราเรียกว่า Normal Workstation หรือถ้าเป็น Diskless Workstation ซึ่งหมายถึง PC หรืออุปกรณ์ติดตั้งอื่นๆ ที่ไม่มีหน่วยความจำสำรองทำให้ไม่สามารถเริ่มระบบได้ด้วยตนเอง จึงจำเป็นต้องมีอุปกรณ์ที่เรียกว่า Boot ROM ซึ่งทำหน้าที่จัดรูปแบบของเวิร์ค-สเตชั่นตามทรัพยากรที่ถูกกำหนดให้ทำงานสัมพันธ์กับระบบ

Network Communication System เป็นหัวใจของระบบโครงข่าย หมายถึง ระบบการสื่อสารภายในของโครงข่าย หรือลักษณะการส่งข้อมูลตามสายนั่นเอง ลักษณะการเชื่อมต่อมีหลายแบบ (Network Topology) คือ

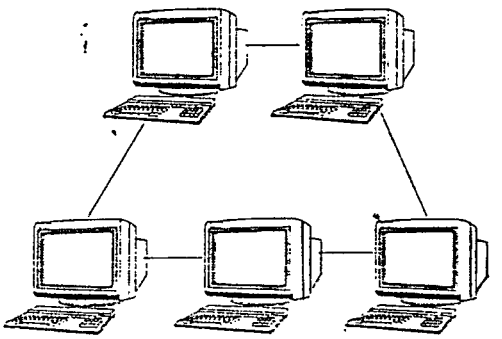
แบบ BUS Topology



รูปที่ 1.2 แสดงการเชื่อมต่อโครงข่ายแบบ BUS Topology

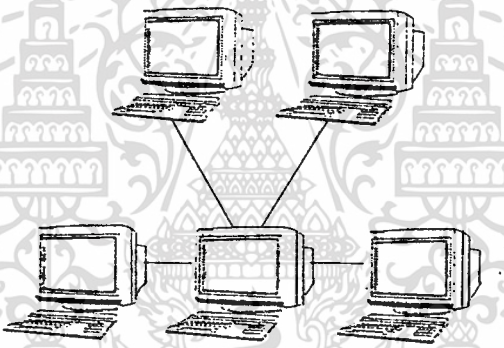
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

แบบ RING Topology



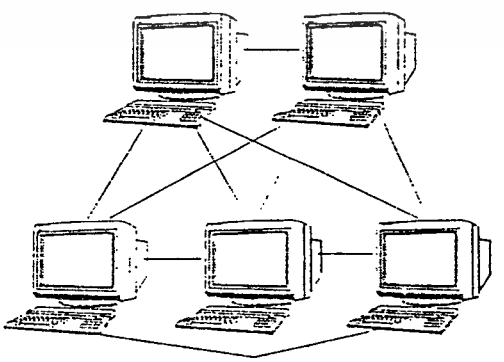
รูปที่ 1.3 แสดงการเชื่อมต่อโครงข่ายแบบ RING Topology

แบบ STAR Topology



รูปที่ 1.4 แสดงการเชื่อมต่อโครงข่ายแบบ STAR Topology

แบบ สมบูรณ์ (Fully Connected Topology)

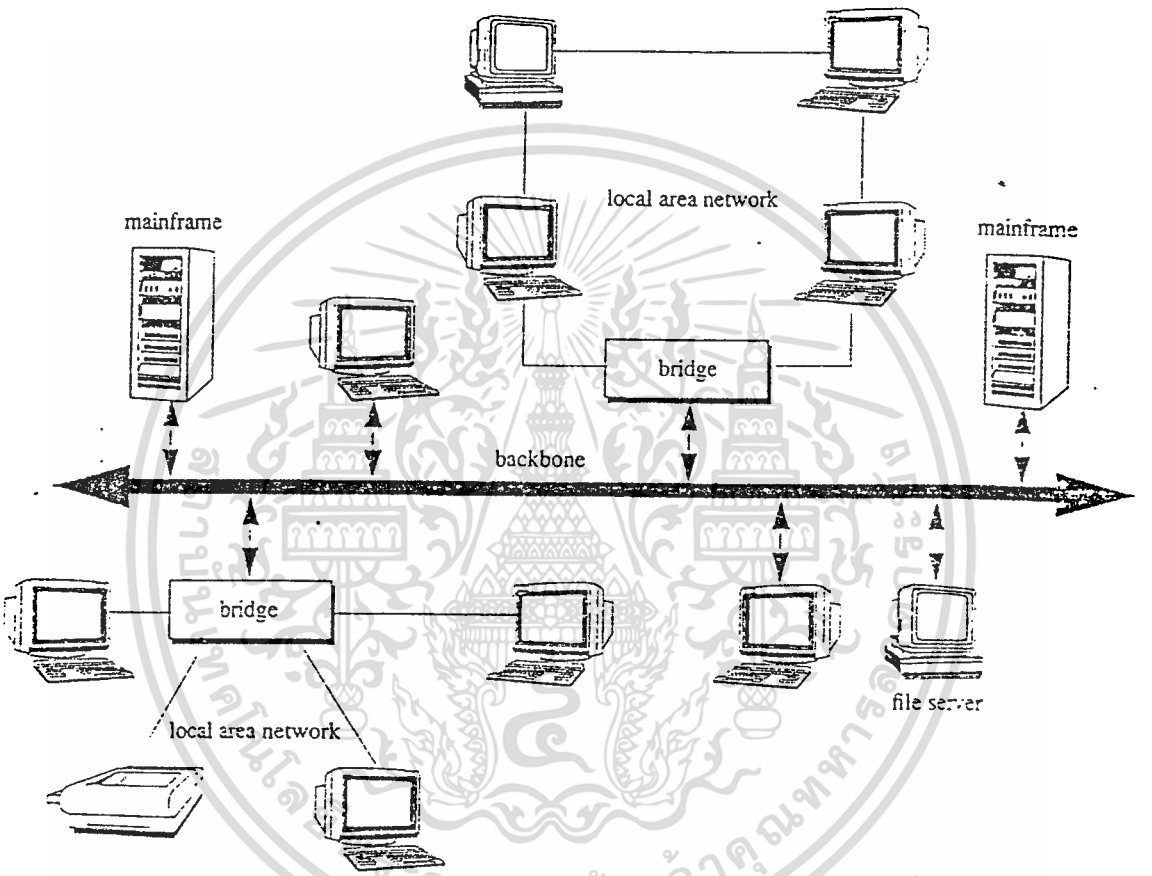


รูปที่ 1.5 แสดงการเชื่อมต่อโครงข่ายแบบสมบูรณ์ (Fully Connected Topology)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โครงข่ายแบบสมบูรณ์ถือเป็นเป้าหมายของการบริหารโครงข่ายในอุดมคติเพราะเป็นลักษณะของการติดต่อที่ไม่มีการชนกันของข้อมูล ไม่มีการหน่วงเวลา แต่ราคาสูงและไม่คุ้มค่าทางเศรษฐกิจเมื่อบางอุปกรณ์ติดต่อกันน้อยมาก แก้ไขได้โดยการส่งทางอ้อม

แบบผสม (Combined Topology)



รูปที่ 1.6 แสดงการเชื่อมต่อโครงข่ายแบบผสม (Combined Topology)

หลายๆ โครงข่ายคอมพิวเตอร์ใช้ลักษณะโครงข่ายแบบผสม ซึ่งต้องประกอบไปด้วย Common Bus หรือ BACK BONE ทำหน้าที่เหมือนสายบัสในรูปแบบแรก (รูปที่ 1.2) แต่ให้อัตราการส่งผ่านข้อมูลที่เร็วมากและมีประสิทธิภาพมากกว่า เหมาะสำหรับการติดต่อขอเข้าใช้ข้อมูลใน File Server

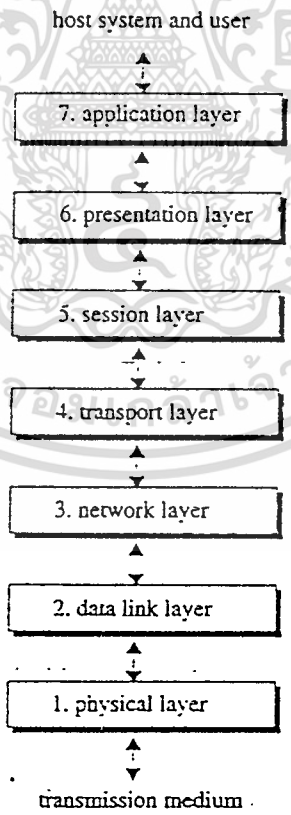
อย่างไรก็ดีแต่ละรูปแบบก็มีข้อจำกัดต่างกัน เช่น จำกัดในเรื่องของความยาวของสาย Cable ทั้งระบบรวมกัน ความเร็วในการทำงานที่แตกต่างกัน รวมทั้งอุปกรณ์ที่ใช้งานก็แตกต่างกันด้วย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โครงข่ายที่นิยมใช้ในองค์การมักเป็นวงเล็กๆ เรียกว่า ระบบ LAN (Local Area Network) และถ้ามีการขยายระบบจะต้องมีการเชื่อมต่อกันโดยอาศัย Bridge หรือ Router เพื่อให้สามารถใช้ข้อมูลร่วมกันได้อย่างทั่วถึง เราจะเรียกระบบนี้ว่า WAN (Wide Area Network) ลักษณะของ Bridge ดังกล่าวก็คืออุปกรณ์ที่เชื่อมต่อ LAN 2 วงที่มีแบบ (Topology) เดียวกันเข้าด้วยกัน ในขณะที่ Router ก็คืออุปกรณีที่เชื่อมต่อ LAN 2 วงที่มีแบบ ต่างกันเข้าด้วยกัน

2. **ส่วนซอฟท์แวร์** (Software)

ส่วนของซอฟท์แวร์คือส่วน Operating System ของระบบซึ่งทำหน้าที่ในการควบคุมการทำงานทั้งหมดของระบบ ไม่ว่าจะเป็นความปลอดภัยของระบบ สิทธิในการใช้งานของแต่ละผู้ใช้ การใช้ทรัพยากรร่วมกัน รวมทั้งการเชื่อมต่อระบบ LAN เข้ากับระบบคอมพิวเตอร์ สิ่งหนึ่งที่เราควรรู้สำหรับการบริหารโครงข่ายหรือการพัฒนาโปรแกรมสำหรับโครงข่ายก็คือ **ระบบเปิด** (Open System) ซึ่งจะหมายถึง กลุ่มของโปรโตคอลที่ทำให้ระบบ 2 ระบบเชื่อมต่อกันได้โดยไม่สนใจรูปแบบของโปรโตคอลเลย ISO ได้ออกแบบจำลองของการเชื่อมต่ออุปกรณ์จำนวนมากเพื่อเข้าสู่โครงข่าย เรียกว่า OSI (Open Systems Interconnect) ประกอบด้วย 7 เลเยอร์ดังรูปที่ 1.7



รูปที่ 1.7 แสดง OSI Layered Protocol Model

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

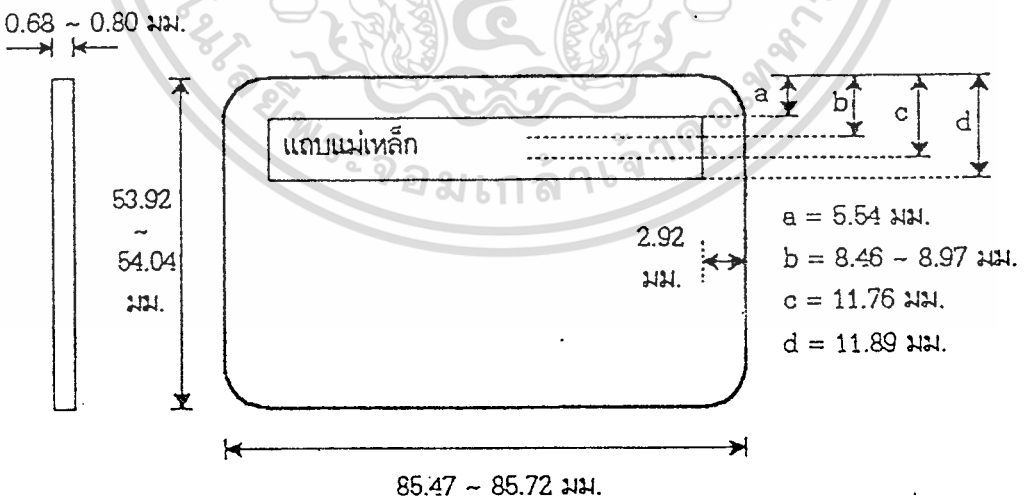
บทที่ 2
หลักการและทฤษฎี

2.1 บัตรแม่เหล็ก

สำหรับการออกแบบระบบควบคุมการเข้าออกโดยใช้บัตรแม่เหล็ก โดยใช้ข้อมูลบนบัตรแม่เหล็กเป็นข้อมูลประจำตัวของเจ้าของบัตรที่จะใช้ในการตรวจสอบข้อมูลนั้น ในขั้นต้นนี้เราได้ใช้บัตร ATM ของธนาคารพาณิชย์และบัตรแม่เหล็กของบางห้างร้านมาประยุกต์ใช้ในระบบควบคุมของเรา โดยที่บัตร ATM นี้มีรูปแบบของข้อมูลที่บันทึกอยู่บนแถบแม่เหล็กและคุณลักษณะต่าง ๆ เป็นไปตามมาตรฐานสากล หรือ International Organization for Standardization (ISO)

2.1.1 คุณสมบัติของบัตรแม่เหล็กตามมาตรฐาน ISO

สำหรับบัตรแม่เหล็กที่เป็นบัตร ATM หรือ บัตรเครดิตของธนาคารพาณิชย์ต่าง ๆ นั้น จะมีแถบแม่เหล็กที่มีแทร็ค (Track) การบันทึกข้อมูลจำนวนทั้งสิ้น 3 แทร็ค โดย**แทร็คที่สอง**ของแถบแม่เหล็กนี้จะบันทึกตัวเลขที่อ้างอิงกับหมายเลขบัญชีของเจ้าของบัตรเอาไว้ ซึ่งในการนำมาประยุกต์ใช้กับระบบควบคุมการเข้าออกโดยใช้บัตรแม่เหล็กนี้เราจะใช้ข้อมูลในแทร็คที่สองนี้มาใช้งานเป็นข้อมูลประจำตัวของเจ้าของบัตรในการที่จะนำมาตรวจสอบต่อไป



รูปที่ 2.1 แสดงตำแหน่งแทร็คที่สองของบัตรแม่เหล็กตามมาตรฐาน ISO

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากรูปที่ 2.1 บริเวณของแทร็คที่สองของบัตรแม่เหล็กจะอยู่ระหว่าง
เส้นขนาน (b) ค่า 8. 97 มิลลิเมตร (0.353 นิ้ว) (ค่าสูงสุด)
หรือค่า 8.46 มิลลิเมตร (0.333 นิ้ว) มิลลิเมตร กับ
เส้นขนาน (c) ค่า 11.76 มิลลิเมตร (0.463 นิ้ว)

2.1.2 **ชุดรหัสข้อมูลในแทร็คที่สอง**

ข้อมูลที่บันทึกอยู่ในแทร็คที่สองของบัตรแม่เหล็กจะเป็นตัวเลขอย่างเดียว โดยที่ตัวเลข 1 ตัว จะประกอบด้วยบิทข้อมูลแบบ BCD 4 บิท และบิทพาริตี้ 1 บิท ซึ่งใช้ในการตรวจสอบข้อมูลของแต่ละตัวเลข โดยตรวจสอบแบบพาริตี้คี่ (Odd Parity) ซึ่ง ISO ได้ระบุจำนวนข้อมูลสูงสุดที่สามารถบันทึกในแทร็คที่สองไว้ไม่เกิน 40 ตัว (รวมสัญลักษณ์เริ่มต้นและสิ้นสุดข้อมูล) ส่วนชุดรหัสข้อมูลตัวเลขของแต่ละตัวเลขสำหรับแทร็คที่สอง แสดงไว้ดังตารางที่ 2.1 ดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 2.1 แสดงรหัสข้อมูลตัวเลขสำหรับแทร็คที่สอง

p บิทพาริตี้	b4	b3	b2	b1	รหัส
1	0	0	0	0	0
0	0	0	0	1	1
0	0	0	1	0	2
1	0	0	1	1	3
0	0	1	0	0	4
1	0	1	0	1	5
1	0	1	1	0	6
0	0	1	1	1	7
0	1	0	0	0	8
1	1	0	0	1	9
1	1	0	1	0	A
0	1	0	1	1	B1
1	1	1	0	0	A
0	1	1	0	1	B2
0	1	1	1	0	A
1	1	1	1	1	B3

จากตารางที่ 2.1

A เป็นตำแหน่งของสัญลักษณ์ที่ใช้เฉพาะในระบบควบคุมทางฮาร์ดแวร์

B1 เป็นสัญลักษณ์การเริ่มต้นของข้อมูล (start sentinel)

B2 เป็นสัญลักษณ์ตัวแยกข้อมูล (separator)

B3 เป็นสัญลักษณ์การสิ้นสุดของข้อมูล (stop sentinel)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.1.3 รูปแบบของข้อมูลที่บันทึกในแทร็คที่สองบนบัตรแม่เหล็ก

ข้อมูลที่บันทึกในแทร็คที่สองบนบัตรแม่เหล็ก ATM ของธนาคารพาณิชย์ทั่ว ๆ ไปมีรูปแบบดังรูปที่ 2.2

SYN	B1	ข้อมูล	B2	ข้อมูล	B2	ข้อมูล	B3	LRC	SYN

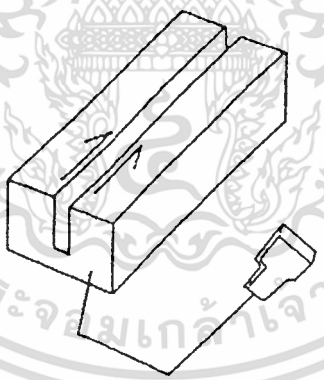
SYN : Synchronization characters

LRC : Longitudinal redundancy check

รูปที่ 2.2 แสดงรูปแบบของข้อมูลที่บันทึกในแทร็คที่สองบนบัตรแม่เหล็ก

2.1.4 การอ่านข้อมูลจากบัตรแม่เหล็ก

การอ่านข้อมูลจากบัตรแม่เหล็ก สามารถทำได้โดยให้แถบแม่เหล็กสัมผัสกับหัวอ่านของตัวอ่านบัตรแม่เหล็ก และทำการเลื่อนบัตรแม่เหล็กให้แถบแม่เหล็กเคลื่อนที่ด้วยความเร็วคงที่ ซึ่งจะต้องมีโปรแกรมรองรับการอ่านข้อมูลจากบัตรแม่เหล็กอยู่แล้ว

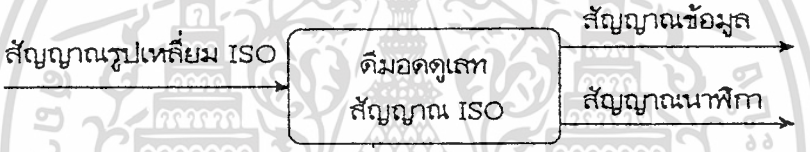


รูปที่ 2.3 แสดงตัวอ่านบัตรแม่เหล็ก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

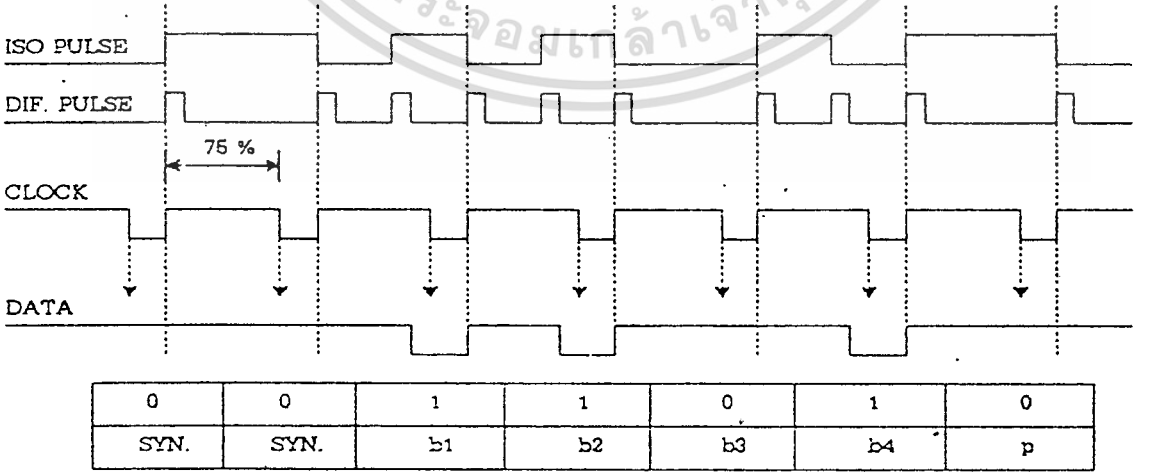
Pin	Symbol	Description	I/O	สาย
1	TD2	TRACK DATA	OUT	น้ำตาล
2	TC2	TRACK CLOCK	OUT	แดง
3	-	-	-	-
4	-	-	-	-
5	CP	CARD PRECENT	OUT	ส้ม
6	VCC	5V SUPPLY	IN	เหลือง
7	GND	SIGNAL GROUND	-	เขียว
8	-	-	-	-
9	-	-	-	-

ตัวอ่านบัตรแม่เหล็กจะมีการดีมอดดูเลทสัญญาณรูปเหลี่ยม ISO เพื่อให้ได้สัญญาณนาฬิกาสำหรับการอ่าน และสัญญาณข้อมูลดังรูปที่ 2.4



รูปที่ 2.4 แสดงแผนภาพของวงจรอ่านบัตรแม่เหล็กในตัวอ่านบัตรแม่เหล็ก

การแยกสัญญาณข้อมูลออกจากสัญญาณนาฬิกาโดยใช้หลักการดีมอดดูเลทสัญญาณ ISO สามารถพิจารณาได้จากรูปคลื่นสัญญาณ ดังแสดงในรูปที่ 2.5



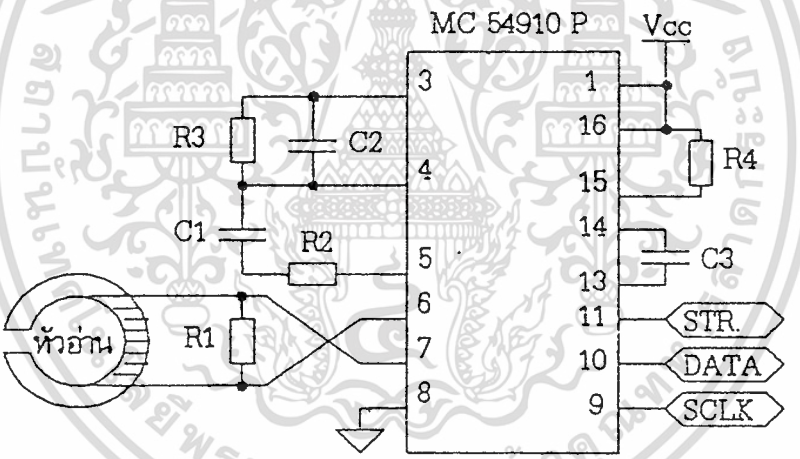
รูปที่ 2.5 แสดงรูปคลื่นสัญญาณของ ISO Pulse ที่ถูกดีมอดดูเลท

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากรูปที่ 2.5 การแยกสัญญาณข้อมูลออกจากสัญญาณรูปเหลี่ยม (ISO) ได้นั้นจะต้องพิจารณาการกลับขั้วของสัญญาณรูปเหลี่ยม ISO ในช่วง Standard Time หรือไม่ ถ้ามีการกลับขั้วของสัญญาณรูปเหลี่ยม ISO แสดงว่าข้อมูลมีค่าทางลอจิกเป็น "1" และถ้าไม่มีการกลับขั้วของสัญญาณรูปเหลี่ยม ISO แสดงว่าข้อมูลมีค่าทางลอจิกเป็น "0" โดยที่ค่า Standard Time มีค่าประมาณ 75% คงที่ ของคาบเวลา 1 บิท เพราะเป็นค่าที่เหมาะสมกับตัวอ่านบัตรแม่เหล็กแบบที่ใช้การรูดผ่านหัวอ่านบัตรด้วยความเร็วคงที่ ส่วนสัญญาณที่ได้จากการดีมอดดูเลทจะมีสัญญาณข้อมูล และสัญญาณนาฬิกา สำหรับการอ่านข้อมูล โดยที่ทุก ๆ ขอบขาลงของสัญญาณนาฬิกา บิทข้อมูลจริงจะกลับลอจิกกับสัญญาณข้อมูลที่อ่านได้

2.1.5 วงจรอ่านข้อมูลบัตรแม่เหล็กโดยใช้ชิพ MC 54910 P

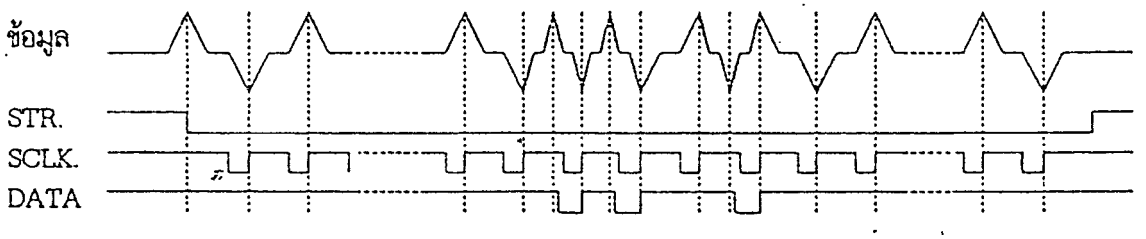
MC 54910 P เป็นชิพไอซีที่ใช้ในการอ่านบัตรแม่เหล็กแบบรูดบัตรด้วยมือโดยเฉพาะซึ่งอยู่ภายในตัวอ่านบัตรแม่เหล็ก



รูปที่ 2.6 แสดงวงจรอ่านบัตรแม่เหล็กโดยใช้ชิพ MC 54910 P

สัญญาณเอาท์พุทมี 3 สัญญาณ ได้แก่ สัญญาณแสดงการเริ่มต้นของการรูดบัตร (STR) , สัญญาณข้อมูล (DATA) และสัญญาณนาฬิกา (CLK) หรือสัญญาณ Standard Time

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.7 แสดงสัญญาณเอาท์พุทของ MC 54910 P

รูปแบบของข้อมูลบนบัตรแม่เหล็กนั้น สามารถดูและตรวจสอบได้ ซึ่งตัวเลขที่เป็นข้อมูล 1 ตัว จะประกอบด้วยบิทข้อมูลแบบ BCD จำนวน 4 บิท และบิทพาริตี้อีก 1 บิท โดยตรวจสอบแบบพาริตี้คี่นั้น ผลการอ่านข้อมูลออกมา โดยนำตัวอย่างจากบัตรแม่เหล็กที่เป็นบัตร ATM ของธนาคารพาณิชย์ 2 ธนาคารดังต่อไปนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เปิดตาราง

[30][42] ----> [0B] ----> [00001011] = B1

ASCII เลขฐาน 16 (Hex) เลขฐานสอง (Binary) **รหัสข้อมูลตัวเลขในแทร็คที่สอง**

ดู 5 บิทล่าง

ตารางที่ 2.2 แสดงตัวอย่างข้อมูลจากบัตรแม่เหล็กของธนาคารกรุงเทพ

ลำดับ	ASCII	Hex	Binary	**รหัสข้อมูลตัวเลขในแทร็คที่สอง**
1	[30][42]	[0B]	[00001011]	B1 (เริ่ม)
2	[31][30]	[10]	[00010000]	0
3	[31][30]	[10]	[00010000]	0
4	[31][30]	[10]	[00010000]	0
5	[31][30]	[10]	[00010000]	0
6	[31][30]	[10]	[00010000]	0
7	[30][32]	[02]	[00000010]	2
8	[30][31]	[01]	[00000001]	1
9	[31][36]	[16]	[00010110]	6
10	[30][31]	[01]	[00000001]	1
11	[30][31]	[01]	[00000001]	1
12	[31][36]	[16]	[00010110]	6
13	[31][33]	[13]	[00010011]	3
14	[30][32]	[02]	[00000010]	2
15	[31][35]	[15]	[00010101]	5
16	[30][31]	[01]	[00000001]	1
17	[30][34]	[04]	[00000100]	4
18	[30][31]	[01]	[00000001]	1
19	[30][31]	[01]	[00000001]	1
20	[30][32]	[02]	[00000010]	2
21	[30][44]	[0D]	[00001101]	B2 (ตัวแยก)
22	[30][31]	[01]	[00000001]	1
23	[30][32]	[02]	[00000010]	2
24	[31][39]	[19]	[00011001]	9
25	[31][39]	[19]	[00011001]	9

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 2.2 แสดงตัวอย่างข้อมูลจากบัตรแม่เหล็กของธนาคารกรุงเทพ (ต่อ)

ลำดับ	ASCII	Hex	Binary	รหัสข้อมูลตัวเลขในแทร็คที่สอง
26	[31][30]	[10]	[00010000]	0
27	[31][30]	[10]	[00010000]	0
28	[31][30]	[10]	[00010000]	0
29	[30][31]	[01]	[00000001]	1
30	[31][33]	[13]	[00010011]	3
31	[31][33]	[13]	[00010011]	3
32	[31][33]	[13]	[00010011]	3
33	[31][46]	[1F]	[00011111]	B3 (จบ)
34	[30][38]	[08]	[00001000]	8
35	[31][30]	[10]	[00010000]	0
36	[31][30]	[10]	[00010000]	0
37	[31][30]	[10]	[00010000]	0
38	[31][30]	[10]	[00010000]	0
39	[31][30]	[10]	[00010000]	0
40	[31][30]	[10]	[00010000]	0

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 2.3 แสดงตัวอย่างข้อมูลจากบัตรแม่เหล็กของธนาคารไทยพาณิชย์

ลำดับ	ASCII	Hex	Binary	รหัสข้อมูลตัวเลขในแทร็คที่สอง
1	[30][42]	[0B]	[00001011]	B1 (เริ่ม)
2	[31][30]	[10]	[00010000]	0
3	[31][30]	[10]	[00010000]	0
4	[30][31]	[01]	[00000001]	1
5	[30][34]	[04]	[00000100]	4
6	[30][38]	[08]	[00001000]	8
7	[30][32]	[02]	[00000010]	2
8	[31][30]	[10]	[00010000]	0
9	[31][30]	[10]	[00010000]	0
10	[30][32]	[02]	[00000010]	2
11	[31][36]	[16]	[00010110]	6
12	[31][35]	[15]	[00010101]	5
13	[30][32]	[02]	[00000010]	2
14	[31][39]	[19]	[00011001]	9
15	[30][37]	[07]	[00000111]	7
16	[30][38]	[08]	[00001000]	8
17	[31][39]	[19]	[00011001]	9
18	[30][44]	[0D]	[00001101]	B2 (ตัวแยก)
19	[30][31]	[01]	[00000001]	1
20	[30][32]	[02]	[00000010]	2
21	[31][39]	[19]	[00011001]	9
22	[31][39]	[19]	[00011001]	9
23	[30][44]	[0D]	[00001101]	B2 (ตัวแยก)
24	[30][32]	[02]	[00000010]	2
25	[30][37]	[07]	[00000111]	7
26	[30][34]	[04]	[00000100]	4
27	[31][39]	[19]	[00011001]	9
28	[31][39]	[19]	[00011001]	9
29	[31][30]	[10]	[00010000]	0

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 2.3 แสดงตัวอย่างข้อมูลจากบัตรแม่เหล็กของธนาคารไทยพาณิชย์ (ต่อ)

ลำดับ	ASCII	Hex	Binary	รหัสข้อมูลตัวเลขในแทร็คที่สอง
30	[31][30]	[10]	[00010000]	0
31	[30][38]	[08]	[00001000]	8
32	[30][38]	[08]	[00001000]	8
33	[30][44]	[0D]	[00001101]	B2 (ตัวแยก)
34	[30][31]	[01]	[00000001]	1
35	[31][30]	[10]	[00010000]	0
36	[31][30]	[10]	[00010000]	0
37	[31][30]	[10]	[00010000]	0
38	[31][30]	[10]	[00010000]	0
39	[31][46]	[1F]	[00011111]	B3 (จบ)
40	[31][39]	[19]	[00011001]	9

เมื่อเปรียบเทียบข้อมูลในแทร็คที่สองของบัตรแม่เหล็ก ของธนาคารทั้ง 2 แล้วพบว่า ข้อมูลในแต่ละลำดับ (1 ถึง 40) **ไม่มี** รูปแบบที่แน่นอนตายตัว ดังนั้นจึงจำเป็นต้องนำข้อมูลทั้ง 40 ข้อมูลมาใช้ในการบันทึกในฐานข้อมูล , การรับ - ส่งข้อมูล , การค้นหาข้อมูล , การเปรียบเทียบข้อมูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.2 การสื่อสารข้อมูลแบบอนุกรมโดยใช้หลักการของ Multiprocessor Communications

ไมโครคอนโทรลเลอร์ MCS-51 มี Serial Port ซึ่งสามารถรับและส่งข้อมูลแบบอนุกรมได้โดยผู้ใช้ไม่ต้องต่อไอซีเพิ่มเติมเข้าไป ทำให้มีความสะดวกในการนำไปประยุกต์ใช้งานที่ต้องมีการติดต่อข้อมูลแบบอนุกรม ซึ่ง Serial Port ที่มีใน MCS-51 สามารถทำงานได้ทั้งในแบบ Half Duplex และ Full Duplex ในแบบ Full Duplex หมายความว่า มันสามารถรับและส่งข้อมูลได้พร้อม ๆ กัน โดยมีการ Buffer ในการรับข้อมูลให้ด้วย

Serial Port ใน MCS-51 สามารถทำงานในโหมดต่าง ๆ กันได้ 4 แบบด้วยกัน เพื่อความเหมาะสมกับงานแต่ละงานดังนี้

Mode 0 : ข้อมูลแบบอนุกรมถูกรับเข้าและส่งออกผ่านทางขา RXD ส่วนขา TXD จะให้ Shift Clock ในโหมดนี้จะทำการรับส่งข้อมูลแบบ 8 บิท (8 Data Bits)โดยรับและส่งบิทต่ำก่อน (LSB First) ส่วนอัตราการส่งข้อมูล (Baud Rate) ถูกกำหนดไว้แน่นอนที่ 1/12 ของความถี่ออสซิลเลเตอร์ที่ใช้ในระบบ

Mode 1 : ข้อมูลจำนวน 10 บิท ถูกส่ง (ผ่านทางขา TXD) หรือถูกรับ (ผ่านทางขา RXD) โดยมี 1 start bit ซึ่งมีค่าเป็น 0 , 8 data bits (รับและส่งบิทต่ำก่อน) และ 1 stop bit ซึ่งมีค่าเป็น 1 ขณะทำการรับ stop bit จะไปอยู่ในบิท RB8 ของ SFR (Special Function Register) SCON ส่วนค่า Baud Rate สามารถเปลี่ยนแปลงได้

Mode 2 : ข้อมูลจำนวน 11 บิทถูกส่ง (ผ่านขา TXD) หรือถูกรับ (ผ่านขา RXD) มี 1 start bit , 8 data bits , บิทที่ 9 เป็นบิทที่สามารถโปรแกรมได้ (Programmable 9th Data bit) และ 1 stop bit โดยในขณะส่งข้อมูล บิทข้อมูลบิทที่ 9 (the 9th data bit) ซึ่งเป็นบิท TB8 ในรีจีสเตอร์ SCON สามารถถูกกำหนดให้เป็น 0 หรือ 1 ส่วนในขณะรับข้อมูล บิทข้อมูลบิทที่ 9 (the 9th data bit) จะไปอยู่ในบิท RB8 ของรีจีส-เตอร์ SCON โดยไม่สนใจ stop bit ค่า Baud Rate สามารถตั้งให้เป็น 1/32 หรือ 1/64 ของความถี่ออสซิลเลเตอร์ที่ใช้

Mode 3 : ข้อมูลจำนวน 11 บิทถูกส่ง (ผ่านขา TXD) หรือถูกรับ (ผ่านขา RXD) โดยมี 1 start bit , 8 data bits , 1 Programmable 9th bit และ 1 stop bit ซึ่งจะเห็นได้ว่าในโหมด 3 จะเหมือนกับโหมด 2 ทุกอย่าง เว้นเสียแต่ในโหมด 3 นี้สามารถเปลี่ยนแปลงค่า Baud Rate ได้เหมือนในโหมด 1

Multiprocessor Communications

การทำงานของ Serial Port ในโหมด 2 และ 3 มีรูปแบบการใช้งานพิเศษนอกเหนือจากการรับส่งข้อมูลธรรมดาตามที่ได้กล่าวมาแล้ว การใช้งานพิเศษที่ว่านี้คือ การติดต่อสื่อสารระหว่าง CPU ด้วยกันเอง (Multiprocessor Communications) โดย CPU ที่ใช้ในระบบมีจำนวนหลายตัว และสามารถ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เลือกการติดต่อกับตัวใดก็ได้ โดยการทำงานใน Mode 2 บิทที่ 9 ที่รับเข้ามาจะถูกนำไปไว้ในบิท RB8 จากนั้น stop bit ก็จะถูกรับตามเข้ามา ซึ่ง Serial Port สามารถโปรแกรมเพื่อที่เมื่อ stop bit ถูกรับเข้ามาแล้ว จะมีผลไปกระตุ้นให้ส่วน Serial Port Interrupt ทำการอินเทอร์รัพท์ CPU ก็ต่อเมื่อบิท RB8 = 1 ซึ่งการทำเช่นนี้ได้ก็โดยการ set บิท SM2 ในรีจีสเตอร์ SCON ซึ่งรายละเอียดและแนวทางในการใช้คุณสมบัติของ Multiprocessor Communication มีดังนี้คือ

เมื่อหน่วยประมวลผลหลัก (Master Processor) ต้องการส่งข้อมูลจำนวนหนึ่งไปยังหน่วยประมวลผลย่อย (Slave Processor) ตัวหนึ่งจากที่มีอยู่หลายตัวในระบบ ในขั้นแรก หน่วยประมวลผลหลักจะต้องส่งค่า Address ขนาด 1 ไบท์ ซึ่งจะเป็นค่าที่บอกตำแหน่งของหน่วยประมวลผลเป้าหมายที่ต้องการติดต่อด้วย Address byte จะมีข้อแตกต่างจาก Data byte ทั่วไปตรงที่ บิทที่ 9 จะเป็น "1" เมื่อข้อมูลนั้นเป็น Address และจะเป็น "0" เมื่อข้อมูลนั้นเป็น Data หากในหน่วยประมวลผลย่อยมีการ set บิท SM2 = 1 แล้ว จะไม่มีหน่วยประมวลผลย่อยตัวใดถูกอินเทอร์รัพท์หากข้อมูลที่เข้ามาเป็น Data แต่ถ้าข้อมูลที่ได้รับเข้ามาเป็น Address (บิทที่ 9 เป็น 1) หน่วยประมวลผลย่อยทุกตัวจะถูกอินเทอร์รัพท์ เพื่อว่าแต่ละตัวจะสามารถตรวจสอบได้ว่าข้อมูลขนาด 1 ไบท์ที่ได้รับเข้ามามีค่าตรงกับแอดเดรสของตัวมันเองหรือไม่ หน่วยประมวลผลย่อยที่แอดเดรสตรงกับข้อมูลที่ได้รับเข้ามา จะทำการเคลียร์บิท SM2 และเตรียมรับข้อมูลที่เป็น Data ซึ่งจะตามเข้ามาหลังจากรับแอดเดรสแล้ว ส่วนหน่วยประมวลผลย่อยตัวอื่นที่ตรวจสอบข้อมูลดูแล้วปรากฏว่าไม่ตรงกับแอดเดรสของตัวมันเองจะยังคงปล่อยให้บิท SM2 ของมันถูก set ต่อไปและกลับไปทำงานที่ค้างอยู่ก่อนได้รับการอินเทอร์รัพท์ต่อ โดยไม่สนใจข้อมูลที่เป็น Data ซึ่งตามเข้ามาหลัง Address เช่นกัน

สำหรับระบบควบคุมการเข้าออกโดยใช้บัตรแม่เหล็กที่ได้ออกแบบไว้นั้น เรากำหนดให้การทำงานของ Serial Port ของ MCS-51 ในระบบทำงานใน Mode 3 เพราะสามารถทำการเปลี่ยนแปลงค่า Baud Rate ในการรับส่งข้อมูลได้

2.3 HDLC Protocol

2.3.1 HDLC Protocol (High-level Data Link Control Protocol)

เป็นโปรโตคอล ที่ถูกกำหนดขึ้นโดย ISO โดยมีรหัสคือ "ISO 4335" สามารถประยุกต์เข้ากับการเชื่อมต่อทางกายภาพหลายรูปแบบ และมีหน้าที่การทำงานดังนี้

- ใช้ได้กับการเชื่อมต่อแบบจุดต่อจุดและหลายจุด
- ใช้ได้กับการส่งผ่านข้อมูลระยะใกล้และไกล
- ใช้ได้กับการส่งผ่านข้อมูลแบบฮาล์ฟดูเพล็กซ์ (Half Duplex) และฟูลดูเพล็กซ์ (Full Duplex)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้
038873

- เป็นโปรโตคอลที่มีการจัดการในระดับบิท (Bit-Oriented Control) เหมาะกับ MCS-51 ที่มีลักษณะการทำงานในระดับบิท (Bit Addressability)

- ใช้ได้กับระบบที่มีการกำหนดประเภทของสถานีทั้ง 3 ประเภท ได้แก่

1. **สถานีปฐมภูมิ** (Primary Station)

บางครั้งเรียกว่า Host station หรือ Control station สถานีปฐมภูมิจะเป็นตัวจัดการการไหลของข้อมูลทั้งหมด โดยส่งคำสั่งให้กับสถานีอื่นๆ และตัดสินใจกับผลตอบสนองของแต่ละสถานี

2. **สถานีทุติยภูมิ** (Secondary Station)

บางครั้งเรียกว่า Target station หรือ Guest station สถานีทุติยภูมิจะตอบสนองต่อคำสั่งที่สถานีปฐมภูมิส่งมาและไม่สามารถส่งคำสั่งใดๆ ได้

3. **สถานีผสม** (Combined Station)

เป็นสถานีที่สามารถส่งคำสั่งและตอบสนองต่อคำสั่งของสถานีต่างๆ ได้

โหมดที่สามารถใช้ HDLC protocal ได้มี 3 โหมด

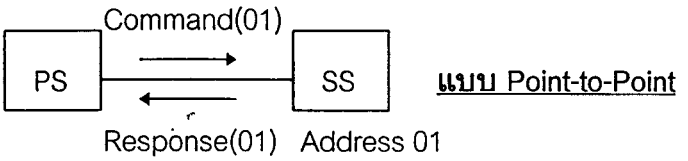
1. Normal Response Mode (NRM) เป็นระบบที่สถานีปฐมภูมิเป็นตัวควบคุมการรับและส่งข้อมูลของสถานีทุติยภูมิ นั่นคือสถานีทุติยภูมิจะสามารถรับและส่งข้อมูลได้เมื่อได้รับคำสั่งหรือได้รับอนุญาตจากสถานีปฐมภูมิเท่านั้น ลักษณะการเชื่อมต่อในโหมดนี้ทีทั้งแบบ Point-to-point Link และ Multipoint Link

2. Asynchronous Response Mode (ARM) เป็นระบบที่เชื่อมต่อได้ในลักษณะเดียวกับ NRM แต่การทำงานของสถานีทุติยภูมิจะเป็นอิสระกว่า คือสามารถส่งข้อมูลให้กับสถานีปฐมภูมิได้โดยไม่ต้องรอการอนุญาต อย่างไรก็ดีสถานีทุติยภูมิก็ยังไม่สามารถส่งคำสั่งไปยังสถานีปฐมภูมิได้ นิยมใช้กับระบบ Point-to-point Link

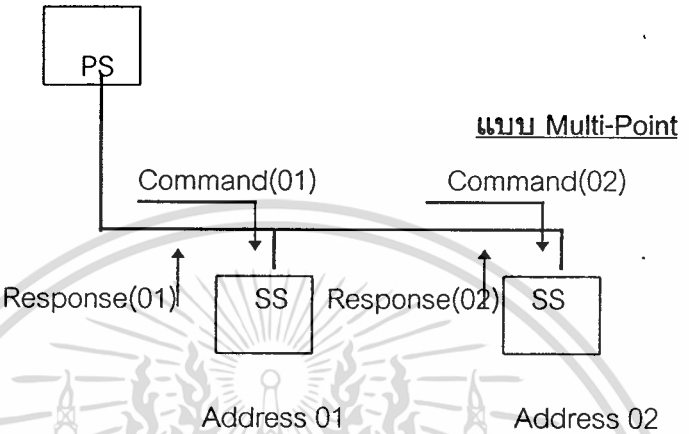
3. Asynchronous Balanced Mode (ABM) ใช้กับระบบที่มีการต่อแบบสถานีผสม แต่ละสถานีสามารถส่งข้อมูลและคำสั่งได้อย่างอิสระ มักใช้ในรูปแบบการเชื่อมต่อระหว่างคอมพิวเตอร์ 2 เครื่อง

ระบบที่มีการกำหนดสถานีปฐมภูมิ และ สถานีทุติยภูมิ เรียกว่า ระบบอสมดุล (Unbalanced System) ซึ่งการเชื่อมต่อเป็นไปได้ทั้งจุดต่อจุดและหลายจุด สำหรับระบบที่ประกอบด้วยสถานีผสมทั้งหมดเรียกว่า ระบบสมดุล (Balance System) ซึ่งการเชื่อมต่อเป็นได้เฉพาะจุดต่อจุดเท่านั้น ดังรูปที่ 2.8

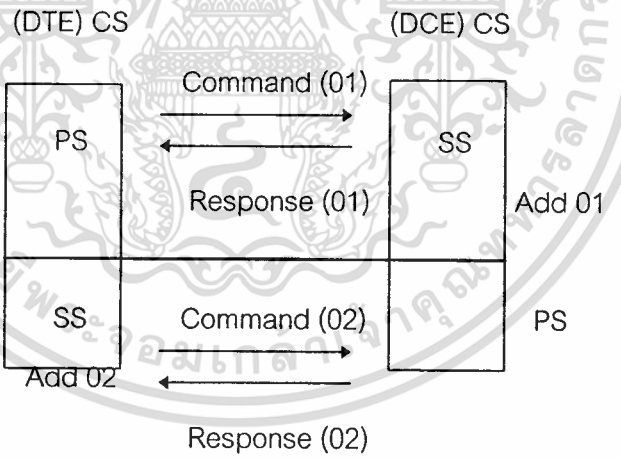
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.8 แสดงระบบอสมดุลแบบ Point-to-point



รูปที่ 2.9 แสดงระบบอสมดุลแบบ Multi-point



รูปที่ 2.10 แสดงระบบสมดุล (Point-to-point)

PS = Primary Station (Master Processor)

SS = Secondary Station (Slave Processor)

CS = Combine Station

DTE = Data Terminal Equipment (เช่น PC เป็นต้น)

DCE = Data Circuit-terminating Equipment (เช่น Modem เป็นต้น)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

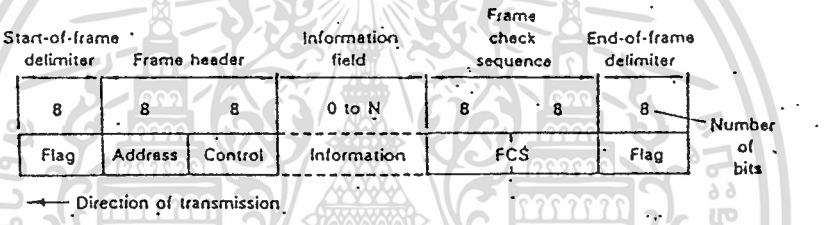
รูปแบบของการส่งผ่านข้อมูลตามข้อกำหนดของโปรโตคอล HDLC แบ่งออกเป็น 3 ลักษณะคือ

1. NRM (Normal Response Mode) ใช้กับระบบอสมดุลสถานีปฐมภูมิต้องการจะติดต่อสถานีทุติยภูมิในกรณีใดก็ได้ แต่สถานีทุติยภูมิจะมีสิทธิ์ติดต่อกับสถานีปฐมภูมิได้ก็ต่อเมื่อสถานีปฐมภูมิอนุญาตเท่านั้น

2. ARM (Asynchronous Response Mode) ใช้กับระบบอสมดุลสถานีทุติยภูมิสามารถจะติดต่อกับสถานีปฐมภูมิได้โดยไม่จำเป็นต้องให้สถานีปฐมภูมิอนุญาตก่อน

3. ABM (Asynchronous Balanced Mode) ใช้กับระบบสมดุลแต่ละสถานีสามารถส่งผ่านข้อมูลถึงกันได้โดยไม่จำเป็นต้องได้รับอนุญาตจากสถานีที่ต้องการจะติดต่อด้วยเสียก่อน

2.3.2 **รูปแบบของเฟรม** (Frame format)



รูปที่ 2.11 แสดงแบบเฟรมของ HDLC Protocol

รูปแบบเฟรมของโปรโตคอล HDLC ประกอบด้วย 5 เขต คือ เขตแฟล็ก (flag field) , เขตแอดเดรส (address field) , เขตควบคุม (control field) , เขตข้อมูล (information field) และ เขตตรวจสอบเฟรม (Frame Check Sequence field : FCS)

เขตแฟล็ก

เขตแฟล็กมีขนาด 8 บิท เป็นเขตที่ใช้บอกถึงจุดเริ่มต้นและจุดสิ้นสุดของเฟรม ซึ่งเรียกว่า แฟล็กเปิด (opening flag) และแฟล็กปิด (closing flag) มีรหัสคงที่คือ 01111110B (7EH)

เขตแอดเดรส

เขตแอดเดรสเป็นเขตที่ใช้บอกถึงหมายเลขของสถานีซึ่งขึ้นอยู่กับว่าเป็นระบบอสมดุลหรือสมดุล แต่ในที่นี้จะกล่าวเพียงระบบอสมดุลเท่านั้น คือ เมื่อสถานีปฐมภูมิส่งเฟรมไปยังสถานีทุติยภูมิ แอดเดรสนี้จะเป็นหมายเลขของสถานีทุติยภูมิ และถ้าเฟรมนี้ถูกส่งจากสถานีทุติยภูมิ แอดเดรสนี้จะเป็นหมายเลขของสถานีทุติยภูมินั้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เขตควบคุม

เขตควบคุมเป็นเขตที่ใช้บอกถึงประเภทของเฟรมซึ่งแบ่งออกเป็น 3 ประเภท คือ เฟรมข้อมูล (Information frame) ซึ่งถูกกำหนดโดยบิทที่มีนัยสำคัญน้อยบิทแรกของขอบเขตควบคุมจะต้องมีค่าเป็น "0" เป็นเฟรมที่ใช้ในการรับส่งข้อมูล ประเภทที่สองคือ เฟรมควบคุมและดูแล (Supervisory frame) ซึ่งถูกกำหนดโดยบิทที่มีนัยสำคัญน้อยสองบิทแรกจะต้องมีค่าเป็น "10" เป็นเฟรมที่ใช้ในการสอบถามและแสดงสถานภาพการรับข้อมูลของแต่ละสถานี และประเภทสุดท้ายคือเฟรม Unnumbered ซึ่งถูกกำหนดโดยบิทที่มีนัยสำคัญน้อยสองบิทแรกจะต้องมีค่าเป็น "11" เป็นเฟรมที่ใช้ในการส่งผ่านคำสั่ง หรือคำตอบรับในการควบคุมระบบ

	CONTROL FIELD BITS							
BIT ORDER	1	2	3	4	5	6	7	8
I frame format	0	N(S)			P/F	N(R)		
S frame format	1	0	S	S	P/F	N(R)		
U frame format	1	1	M	M	P/F	M	M	M

(ก)

	EXTENDED CONTROL FIELD BITS															
	1st Octet									2nd Octet						
BIT ORDER	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
I frame format	0	N(S)							P	N(R)						
S frame format	1	0	S	S	0	0	0	0	/	N(R)						
U frame format	1	1	M	M	*	M	M	M	F	0	0	0	0	0	0	0

* The value of this bit is undefined

LEGEND

N(S) = Send Sequence Number
N(R) = Receive Sequence Number
P/F = Poll/Final Bit
S = Supervisory Bits
M = Modifier Bits

(ข)

รูปที่ 2.12
รูป (ก) เขตควบคุม 8 บิท และ (ข) เขตควบคุม 16 บิท

***หมายเหตุ** เขตควบคุมขนาด 8 bit เป็นลักษณะมาตรฐาน (Standard Format) ของ HDLC ส่วนเขตควบคุมขนาด 16 bit เป็นการขยายการทำงานของ HDLC กับระบบที่มีขนาดใหญ่ขึ้นเรียกว่า Extended Format

ส่วนต่าง ๆ ที่อยู่ภายในเขตควบคุมมีดังนี้

ก. N(S) (send sequence number) ใช้ในการควบคุมการไหลของข้อมูลโดยใช้เป็นค่าที่บอกว่าเป็นเฟรมที่เท่าไร

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ข. N(R) (receive sequence number) ใช้ในการควบคุมการไหลของข้อมูล โดยฝ่ายรับใช้เป็นค่าที่บอกว่าคาดว่าจะได้รับเฟรมต่อไปเป็นเฟรมที่เท่าไร หรือ เป็นค่าที่บอกว่าเฟรมที่ไม่สามารถรับได้เป็นเฟรมที่เท่าไร สำหรับคำตอบรับ หรือ คำตอบปฏิเสธตามลำดับ

ค. P/F (Poll/Final) การ poll หมายถึงการที่สถานีใดสถานีหนึ่งส่งคำสั่งไปให้อีกสถานีหนึ่งเพื่อต้องการการตอบกลับ การ final หมายถึงการที่สถานีที่ถูกทำการตอบรับต่อการ poll นั้น ๆ จะเห็นได้การ poll จะเกิดกับเฟรมที่เป็นคำสั่งเสมอ และจะต้องกำหนดบิท P/F ให้มีค่าเป็น "1" แต่ถ้าเป็น "0" จะไม่เกิดการ poll สำหรับการ final จะเกิดกับเฟรมที่เป็นคำตอบรับเสมอ และจะต้องกำหนดบิท P/F ให้มีค่าเป็น "1" แต่ถ้าเป็น "0" จะไม่เกิดการ final

ง. S (Supervisory code) เป็นรหัสที่กำหนดชุดคำสั่ง และคำตอบรับ

S FRAMES	BITS	
COMMANDS/RESPONSES	3	4
RR — Receive Ready	0	0
REJ — Reject	0	1
RNR — Receive Non Ready	1	0
SREJ — Selective Reject	1	1

U FRAMES		CONTROL FIELD BITS				
COMMANDS	RESPONSES	3	4	6	7	8
SNRM		0	0	0	0	1
SNRME		1	1	0	1	1
SARM	DM	1	1	0	0	0
SARME		1	1	0	1	0
SABM		1	1	1	0	0
SABME		1	1	1	1	0
DISC	RD	0	0	0	1	0
	UA	0	0	1	1	0
SIM	RIM	1	0	0	0	0
TEST	TEST	0	0	1	1	1
XID	XID	1	1	1	0	1
UI	UI	0	0	0	0	0
	FRMR	1	0	0	0	1

รูปที่ 2.13 แสดงรหัสชุดคำสั่งและคำตอบรับ

จ. M (Unnumbered code) เป็นรหัสที่กำหนดชุดคำสั่งและคำตอบรับ ดังรูป

เขตควบคุมจะมีขนาด 8 บิท หรือ 16 บิท ก็ได้ขึ้นอยู่กับการกำหนดของระบบนั้นๆ ในกรณีแบบ 8 บิท จะทำให้สถานีสามารถรับส่งข้อมูลได้สูงสุด 7 เฟรมติดต่อกัน โดยไม่ต้องรอการตอบกลับจากสถานีตรงข้าม และในกรณีแบบ 16 บิท จะทำให้สถานีสามารถรับส่งข้อมูลกันได้สูงสุด 127 เฟรมติดต่อกันโดยไม่ต้องรอการตอบกลับจากสถานีตรงกันข้าม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เขตข้อมูล

เขตข้อมูลมีขนาดเป็นจำนวนเท่าของ 8 บิท ใช้สำหรับบรรจุข้อมูล

เขตตรวจสอบเฟรม

มีขนาด 16 บิท เป็นเขตที่ใช้ในการตรวจสอบความถูกต้องของเฟรมที่ได้รับที่ปลายทางโดยใช้หลักการของ CRC (Cyclic Redundancy Check) สำหรับโปรโตคอล HDLC ใช้ $X^{16} + X^{12} + X^5 + 1$ เป็น generator polynomial

2.3.3 ลักษณะการส่งเฟรมโต้ตอบของโปรโตคอล HDLC

U-Frame (Unnumbered frame) ใช้สำหรับ Link Control ได้แก่

1. SABM (Set Asynchronous Balance Mode) เป็น Command ที่ใช้สำหรับ Set up หรือ Initial เพื่อเปลี่ยนจาก Disconnect Mode มาเป็น Operational Mode ที่ใช้ในการรับส่งข้อมูลแบบมาตรฐาน (Standard Format) สำหรับระบบ SABM นอกจากนี้ยังทำให้ Sending และ Receiving Counter ถูกรีเซตเป็น "0" ด้วยเพื่อเริ่มต้นกันใหม่

2. UA (Unnumbered Acknowledge) เป็น Response ที่ตอบรับ SABM เพื่อเข้าสู่ Operational Mode ดังกล่าว และใช้สำหรับตอบรับ command ต่างๆ

3. Disc (Disconnect) เป็น Command ที่ทำให้เริ่มต้นการทำงานจาก Operational Mode มาเป็น Disconnect Mode ซึ่งไม่สามารถรับส่งข้อมูลกันได้ คำสั่งนี้จะสมบูรณ์ ได้ก็ต่อเมื่อสถานีที่ถูก Disconnect ส่งสัญญาณ UA ตอบรับมาก่อน

4. DM (Disconnect Mode) เป็น Response มีหน้าที่ 2 อย่างคือ

4.1 ขอให้สถานีปฐมภูมิส่ง Command SABM มาเพื่อเปลี่ยนเข้าสู่ Operational Mode

4.2 เป็น Response ที่บอกว่าไม่สามารถเข้าสู่ Operational Mode ได้

5. FRMR (Frame Reject) เป็น Response อยู่ใน Operational Mode เพื่อเป็น Reject frame ที่ผิดปกติในลักษณะ

5.1 Control field ที่ผิดปกติ

5.2 มี Information field ที่ยาวเกินไป

5.3 Frame ที่ไม่ควรมี Information field แต่กลับมีเช่น SABM เป็นต้น

5.4 Frame ที่มี Flow control (counter) ผิดปกติ

Information frame (I-Frame) เป็น Frame ที่มีข้อมูลที่จะรับส่งบรรจุอยู่ และมี Counter ในการตอบรับคือ N(R) และในการส่งคือ N(S) เพื่อใช้ใน Flow and Error Control

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Supervisory frame (S-Frame) ใช้สำหรับ Flow and Error Control

บิท Poll / Final

บิท Poll ใช้ใน Command ที่ต้องการได้รับ Response ตอบกลับมา

บิท Final ใช้ใน Response ที่ตอบรับ Command ที่ส่งบิท Poll มา

ซึ่งการใช้บิท Poll / Final นี้จะใช้ใน 2 ลักษณะคือ

1. การ Set Mode

2. การ Re-Transmission

การออกแบบโปรโตคอลบนพื้นฐานของ HDLC Protocol

จากหลักการของ HDLC Protocol ในหัวข้อที่ 2.3 พบว่าลักษณะเด่นของ HDLC Protocol ประกอบไปด้วย

1. จัดการกับข้อมูลในระดับบิท : การทำงานในระดับบิทจะช่วยให้การสื่อสารทางพอร์ทอนุกรมมีประสิทธิภาพมากยิ่งขึ้น เพราะถ้าการแปลความหมายของข้อมูลที่ได้รับเข้ามามีความรวดเร็วมากเท่าใด ความเร็วสูงสุดในการส่งข้อมูลก็จะเพิ่มขึ้น หรือถ้าจำนวนข้อมูลที่จะทำให้อุปกรณ์ตัวรับเข้าใจความหมายมีจำนวนน้อยมากเท่าใด ประสิทธิภาพของการส่งก็จะเพิ่มขึ้นด้วยเช่นกัน

2. มีการแบ่งหน้าที่ของข้อมูลในหนึ่งเฟรมเป็นหลายหน้าที่ เหมาะกับการประยุกต์ใช้กับระบบทั่วๆ ไป HDLC Protocol ได้จัดให้ 1 เฟรมมี 7 ไบท์ ดังที่ได้อธิบายไว้ข้างต้น ยกตัวอย่างเช่น Control Field ทำหน้าที่ออกคำสั่งหรือตอบรับคำสั่งในขณะที่ Information Field ระบุข้อมูลที่อุปกรณ์ตัวรับต้องการ

3. มีการตรวจสอบเฟรมข้อมูลโดยวิธี Cyclic Redundancy Check ซึ่งให้ผลการเข้ารหัสที่มีประสิทธิภาพสูง นั่นคือ ข้อมูลแต่ละข้อมูลจะให้รหัสตรวจสอบที่แตกต่างกัน แสดงให้เห็นถึงประสิทธิภาพการตรวจสอบข้อมูล

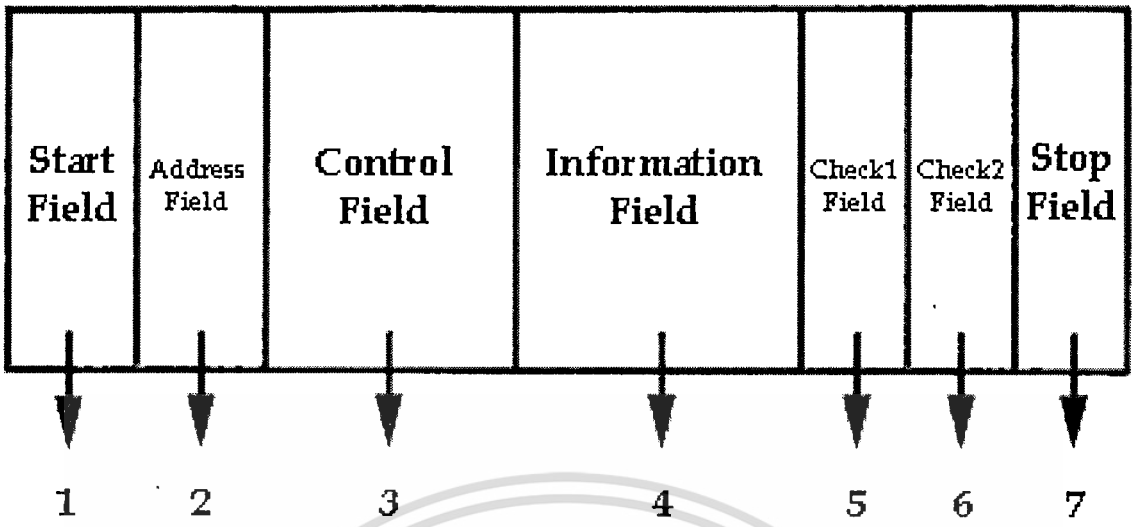
จากลักษณะทั้ง3ประการ เราสามารถนำมาประยุกต์สร้างโปรโตคอลด้วยไมโครคอนโทรลเลอร์ตระกูล MCS-51 ซึ่งเป็นโปรเซสเซอร์ของระบบที่ได้ทำการออกแบบและ**เป็นการสนับสนุนการทำงานในระดับบิท**

หลักการออกแบบโปรโตคอลสำหรับระบบควบคุมการเข้าออกโดยใช้บัตรแม่เหล็ก

ระบบที่ถูกจำลองเพื่อใช้กับระบบควบคุมการเข้าออกโดยใช้บัตรแม่เหล็ก จะเป็นห้องที่ประกอบไปด้วยอุปกรณ์ที่ต้องการควบคุมผ่านโครงข่าย หรือในสำนักงานที่ประกอบด้วยเครื่องคอมพิวเตอร์จำนวนมากที่ต้องการจำกัดการทำงานสำหรับผู้ใช้ที่ถูกระบุไว้เท่านั้นเป็นต้น สิ่งที่สำคัญในการออกแบบโปรโตคอลก็คือ โปรโตคอลที่ใช้จะต้องสั้น และให้ข้อมูลได้หลายลักษณะเพื่อนำไปประยุกต์ได้หลายงาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ลักษณะของโปรโตคอลที่ได้ออกแบบไว้



รูปที่ 2.14 แสดงรูปแบบโปรโตคอลที่ได้ออกแบบไว้

จากรูปที่ 2.14 โปรโตคอลได้ถูกแบ่งออกเป็น 7 ส่วน ดังต่อไปนี้

1. START FIELD : เป็นไบท์ข้อมูลขนาด 8 บิท มีค่าเป็น 01111110B เป็นสัญญาณเริ่มต้นของข้อมูลที่รับเข้ามาโดยอิงพื้นฐานของ HDLC Protocol

2. ADDRESS FIELD : เป็นไบท์ข้อมูลแสดงแอดเดรสของอุปกรณ์ที่ต่ออยู่บนโครงข่าย มีหลักการดังต่อไปนี้

- เมื่อมีการส่งข้อมูลจากเทอร์มินอลไปเซิร์ฟเวอร์ : Address Field จะบรรจุแอดเดรสของเทอร์มินอลต้นทาง เพื่อให้เซิร์ฟเวอร์ตรวจสอบได้ว่าเทอร์มินอลตัวใดติดต่อมา
- เมื่อมีการส่งข้อมูลจากเซิร์ฟเวอร์ไปเทอร์มินอล : Address Field จะบรรจุแอดเดรสของเทอร์มินอลปลายทาง เพื่อกระตุ้นเทอร์มินอลตัวที่ต้องการติดต่อให้ทำงานตาม Control Field ที่จะกล่าวถึงต่อไป

การกำหนดชื่อของเซิร์ฟเวอร์และเทอร์มินอลในโปรแกรม S485.ASM (ไฟล์สำหรับเซิร์ฟเวอร์)และ T485.ASM (ไฟล์สำหรับเทอร์มินอล) ซึ่งเป็นโปรแกรมที่ถูกบรรจุใน EPROM ของไมโครคอนโทรลเลอร์นั้นมีดังต่อไปนี้

เซิร์ฟเวอร์	: 10000001B
ไมโครคอมพิวเตอร์	: 11111111B
เทอร์มินอล 1	: 00000001B
เทอร์มินอล 2	: 00000010B

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เทอร์มินอล 3 : 00000011B

.

.

.

เทอร์มินอล 32 : 00111111B

3. CONTROL FIELD : เป็นไบท์ข้อมูลคำสั่งหรือสัญญาณตอบรับซึ่งใช้ในการจัดการและควบคุมการสื่อสารในระบบ ประกอบด้วย

ตารางที่ 2.4 แสดง Control Field

สัญลักษณ์	รหัสข้อมูล	ชื่อเต็ม	หน้าที่
DM	00001111B	Disconnecting Mode	เทอร์มินอลส่งให้เซิร์ฟเวอร์เพื่อเริ่มต้นการเชื่อมต่อเข้าสู่โครงข่าย
UA	01100011B	Unnumbered Acknowledge	เทอร์มินอลและเซิร์ฟเวอร์ใช้ตอบรับคำสั่งที่เข้ามาก่อนหน้านี้เพื่อส่งสัญญาณให้ตัวส่งเข้าสู่ขั้นตอนต่อไป
BZ	11100011B	Busy	เซิร์ฟเวอร์ส่งให้เทอร์มินอลทุกๆ ตัวทราบว่า เซิร์ฟเวอร์กำลังทำการรับส่งข้อมูลอยู่ เทอร์มินอลจะไม่สามารถส่งข้อมูลใดๆ ได้
MTU	01001111B	Multi-user	เซิร์ฟเวอร์ส่งให้เทอร์มินอลเพื่อบอกว่า เทอร์มินอลมีการรูดการ์ดของผู้ใช้มากกว่า 1 คน โดยที่คนแรกยังไม่ได้ทำการ Logout
SIM	10000011B	Simultaneous Use	เซิร์ฟเวอร์ส่งให้เทอร์มินอลเพื่อบอกว่าผู้ใช้คนเดียวกันมีการรูดบัตรมากกว่า 1 เทอร์มินอล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สัญลักษณ์	รหัสข้อมูล	ชื่อเต็ม	หน้าที่
FREE	00011111B	Free to Communicate	เซิร์ฟเวอร์ส่งให้เทอร์มินอล เพื่อบอกว่าขณะนี้สายว่าง แล้วสามารถรับส่งข้อมูลกับเทอร์มินอลอื่นๆ ได้
PCBC	00011011B	PC Broadcasting	เซิร์ฟเวอร์ส่งให้เทอร์มินอล เพื่อบอกว่าไมโครคอมพิวเตอร์ต้องการกระจายข่าว
SABM	00111111B	Set Asynchronous Balance Mode	เซิร์ฟเวอร์ส่งให้เทอร์มินอล เข้าสู่ระบบสามารถส่งคำสั่งและตอบรับได้
NRGT	10111111B	Unauthorized User	เซิร์ฟเวอร์ส่งให้เทอร์มินอล เพื่อระบุว่าผู้ใช้ที่รูดบัตรที่เทอร์มินอลนั้นไม่มีชื่ออยู่ในฐานข้อมูล
DISC	01000011B	Disconnect Signal	เซิร์ฟเวอร์ส่งให้เทอร์มินอล เพื่อสั่งให้เทอร์มินอลที่ขอ Logout ถูกตัดออกจากระบบ
D_BYTE	00000000B	Data Byte	เทอร์มินอลส่งให้เซิร์ฟเวอร์ เพื่อระบุว่าสิ่งที่อยู่ใน Information Field เป็นข้อมูล

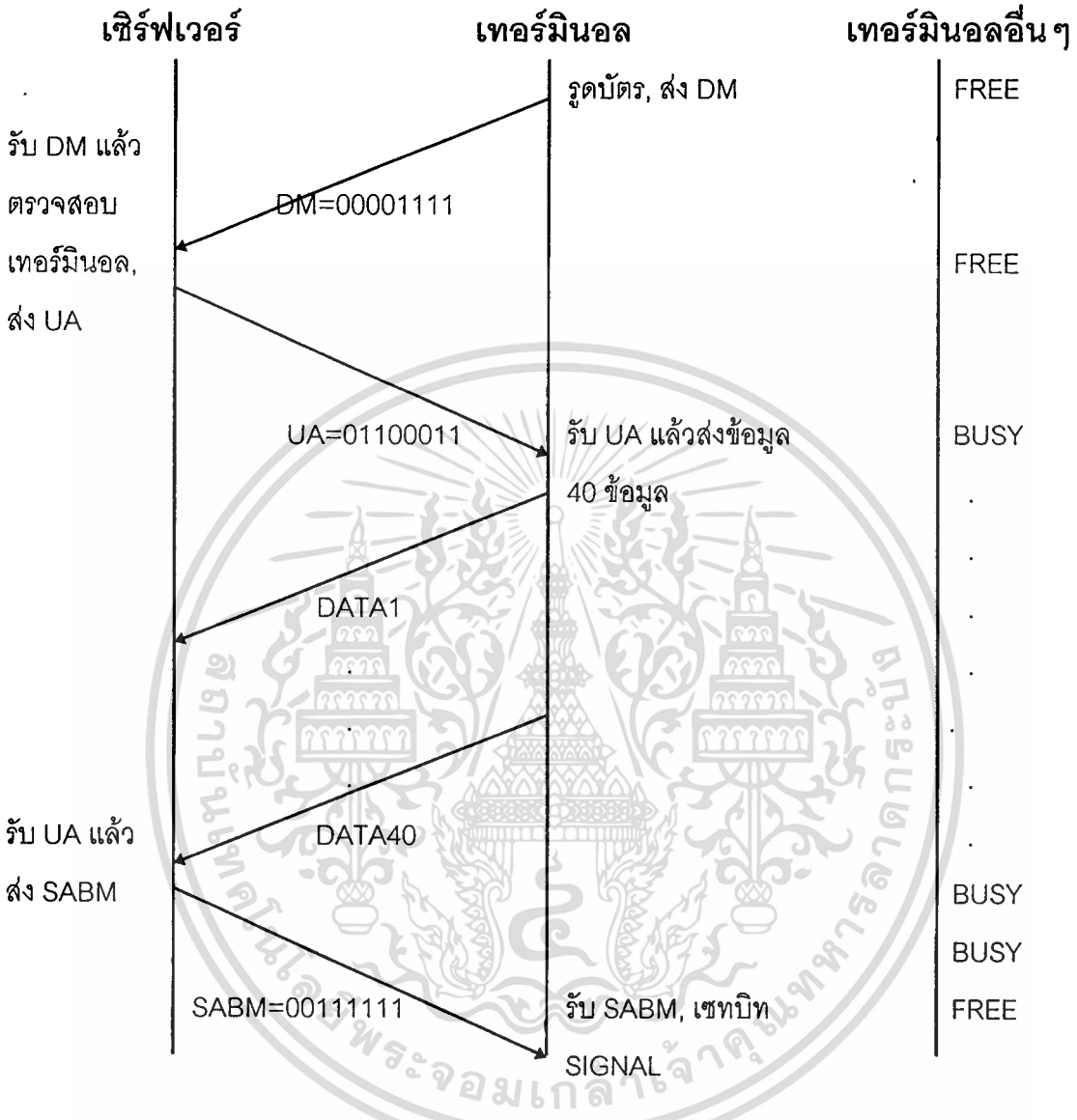
4. INFORMATION FIELD : เป็นส่วนที่สำคัญในการสื่อสาร เพื่อเป็นการสื่อให้ผู้ที่ต้องการติดต่อทราบว่าเราต้องการอะไร ในโปรโตคอลที่ออกแบบนี้ Information Field จะมีความหมายก็ต่อเมื่อ Control Field ถูกระบุเป็น D_BYTE โดยปกติถ้าเฟรมใดๆ ที่ไม่ได้ระบุเป็น D_BYTE ส่วนของ Infomation Field จะเป็นศูนย์

5. CHECK-1 FIELD : สำหรับการตรวจเช็คความถูกต้องของข้อมูล มีค่าเดียวกับ Information Field

6. CHECK-2 FIELD : สำหรับการตรวจเช็คความถูกต้องของข้อมูล มีค่าเดียวกับ Information Field

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

7. STOP FIELD : เป็นไบท์ข้อมูลขนาด 8 บิท มีค่าเป็น 01111110B เป็นสัญญาณสิ้นสุดการรับส่งข้อมูลโดยอิงพื้นฐานของ HDLC Protocol

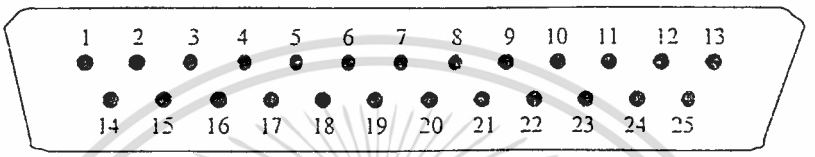


รูปที่ 2.15 แสดงจังหวะการรับส่งข้อมูลระหว่างเซิร์ฟเวอร์กับเทอร์มินอล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.4 มาตรฐานการสื่อสารข้อมูลอนุกรม RS-232

มาตรฐานการสื่อสารข้อมูลอนุกรม RS-232 นั้นเป็นที่รู้จักกันโดยทั่วไป ถูกพัฒนาขึ้นโดย The Electronics Industries Association (EIA) และแบบที่เป็นที่นิยมใช้ทั่วไปคือ RS-232-C มาตรฐานที่เห็นได้เด่นชัดมากที่สุดคือจำนวนสายที่มี 25 เส้น (RS-232 Standard) ระหว่าง DTE และ DCE (บางทีเรียกว่า DB-25 Cable) ซึ่งสายแต่ละเส้นจะมีหน้าที่ในการสื่อสารแตกต่างกันไป ดังรูปที่ 2.16



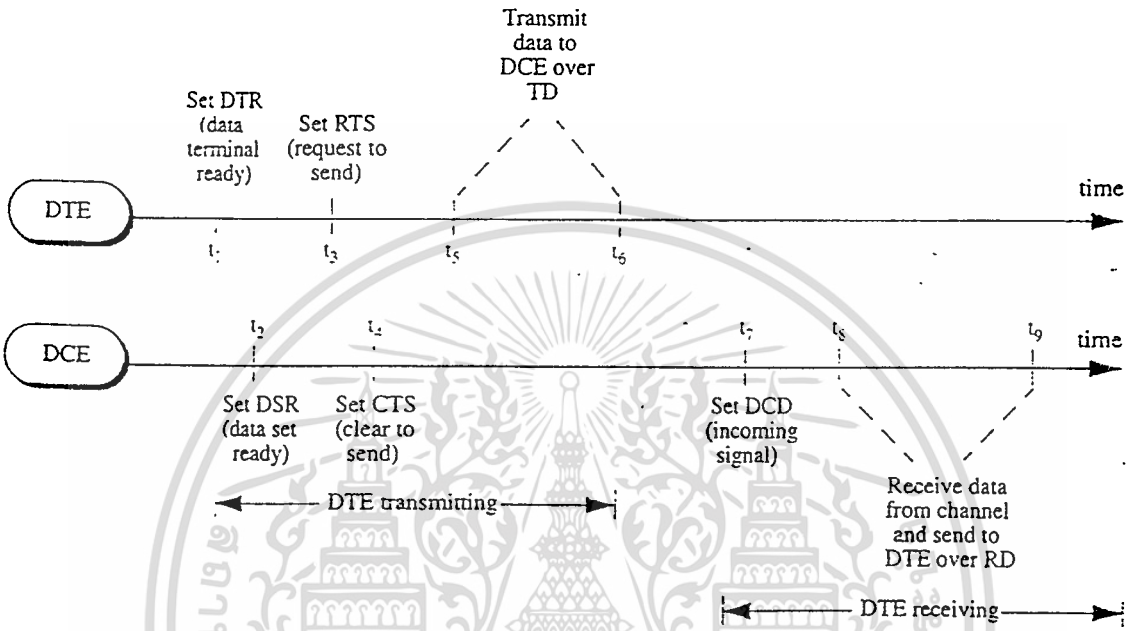
รูปที่ 2.16 แสดง RS-232 Connector

ตัวอย่างเช่น DTE คือ เครื่องไมโครคอมพิวเตอร์ (PC) และ DCE คือโมเดม ซึ่งอุปกรณ์ทั้งสองจะไม่ส่งข้อมูลถ้าอุปกรณ์อีกเครื่องไม่ต้องการรับข้อมูลนั้น เมื่อ DCE เชื่อมต่อเข้ากับโครงข่าย(Network) ซึ่งมี DTE เชื่อมต่ออยู่ด้วย DCE ต้องรู้ว่าเมื่อไรที่ DTE พร้อมในการส่งข้อมูล (ดูรูปที่ 2.17 ประกอบ) เมื่อ DTE ส่งสัญญาณ DTR (Data Terminal Ready ที่เวลา t1) ต่อมา DCE จะตอบสนองต่อสัญญาณโดยเชื่อมตัวเองเข้ากับโครงข่าย ต่อมา DCE จะส่งสัญญาณ DSR (Data Set Ready) เพื่อตอบกลับไปว่าพร้อมสำหรับรับข้อมูล (ที่เวลา t2) ตอนนี้อุปกรณ์ทั้งสองก็พร้อมในการติดต่อสื่อสารกันแล้ว

หาก DTE ขออนุญาตในการส่งข้อมูลให้แก่ DCE มันก็จะส่งสัญญาณ RTS (Request To Send) ช่วงนี้ระบบจะมีทิศทางการสื่อสารเป็นแบบ Half-duplex ในการตอบรับสัญญาณ RTS นั้น DCE จะเข้าสู่โหมด Transmit หมายความว่ามันพร้อมที่จะส่งข้อมูลออกไปสู่โครงข่ายแล้ว และส่งสัญญาณ CTS (Clear To Send) และสุดท้ายแล้ว DTE ก็จะส่งข้อมูลออกไปตามสายสัญญาณ TD (Transmit Data)

เมื่อ DCE ตรวจพบสัญญาณเข้ามาจากโครงข่าย มันจะส่งสัญญาณ DCD (Incoming Signal) DCE จะส่งสัญญาณเหล่านี้ไปยัง DTE โดยใช้สายสัญญาณ RD (Receive Data)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.17 แสดงการส่งและการรับสัญญาณระหว่าง DTE และ DCE ในมาตรฐาน RS-232

โดยทั่วๆ ไปตัวเชื่อมต่อ (Connector) กับพอร์ต RS-232 ที่ใช้กันจะมีสายต่อไม่ถึง 25 เส้น เราเรียกว่าเป็นสับเซตของ RS-232 (RS-232 Subsets) ส่วนมากจะมีสาย 8 หรือ 9 เส้น และมีผู้นิยมใช้กันอย่างแพร่หลาย เพราะผู้ใช้ไม่ต้องการใช้ความสามารถของมาตรฐาน RS-232 อย่างเต็มที่ ซึ่งพวกเขาต้องการที่จะสื่อสารเพียงเพื่อติดต่อและรับส่งข้อมูลเท่านั้น

มาตรฐาน RS-232 สามารถส่งข้อมูลได้ 20,000 บิทต่อวินาที (bps) โดยระยะทางหรือความยาวของสายประมาณ 50 ฟุต

2.5 มาตรฐานการสื่อสารข้อมูลอนุกรม RS-485

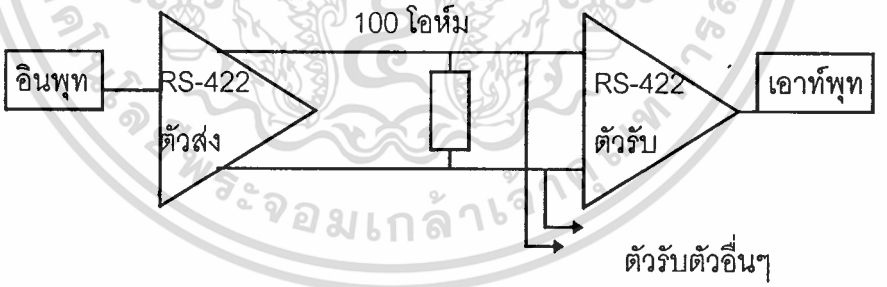
การสื่อสารข้อมูลอนุกรมระหว่างคอมพิวเตอร์ด้วยกัน หรือ คอมพิวเตอร์กับอุปกรณ์รอบข้างด้วยตัวรับและตัวส่งแบบไม่สมดุลย์ เช่น RS-232-C จะประสบปัญหาเกิดข้อมูลผิดพลาดได้ง่าย เมื่อมี

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

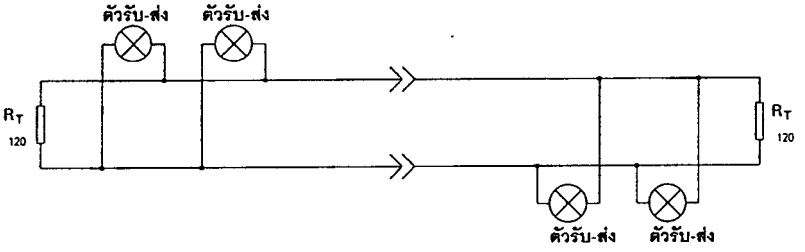
การรับส่งข้อมูลด้วยอัตราการรับส่งข้อมูลสูง,การใช้คู่สายสัญญาณรับส่งข้อมูลยาวมากๆ ในการรับส่งข้อมูลที่มีระยะทางที่ไกล รวมทั้งการวางคู่สายสัญญาณรับส่งผ่านบริเวณที่มีสัญญาณรบกวนสูง The Electronics Industries Association (EIA) จึงได้พัฒนามาตรฐานการสื่อสารข้อมูลอนุกรมแบบสมดุลย์ (ตัวรับและตัวส่งไม่ใช้สายสัญญาณกราวน์ร่วมกัน) ขึ้นเพื่อแก้ไขปัญหาดังกล่าว โดยได้พัฒนาขึ้น 2 มาตรฐาน ได้แก่ RS-422-A และ RS-485 ซึ่งมีการประยุกต์ใช้งานแตกต่างกัน ขึ้นอยู่กับความเหมาะสมของระบบงาน

จากระบบที่ได้ออกแบบไว้ เราเลือก RS-485 มาใช้งาน ซึ่งเป็นมาตรฐานการสื่อสารข้อมูลเพื่อให้ตัวรับและตัวส่งจำนวนมากคู่ สามารถใช้คู่สายสัญญาณรับส่งร่วมกันได้ (multipoint multiple drivers and receivers) และมีสายสัญญาณเพียง 1 คู่ ที่สำคัญคือไม่จำกัดรูปแบบของโปรโตคอลที่จะนำมาใช้งานกับระบบ โดยขึ้นอยู่กับผู้พัฒนาว่าจะเลือกใช้โปรโตคอลแบบไหนมาใช้งาน และยังสามารถใช้กับตัวรับ 32 ชุด และตัวส่ง 32 ชุด RS-485 ยังมีความไวของตัวรับส่งสูง สามารถใช้งานซึ่งต้องอาศัยสายสัญญาณที่ยาวมากๆ ได้และได้ยาวถึง 4000 ฟุต นอกจากนี้ยังมีข้อดีอยู่มากมาย รวมทั้งตัวรับและตัวส่งมีราคาไม่สูง ทำให้ RS-485 ถูกนำมาใช้งานในระบบการสื่อสารข้อมูลอนุกรมแบบโครงข่ายอย่างแพร่หลาย

ส่วน RS-422-A นั้นในหนึ่งคู่สายสัญญาณรับส่งจะมีตัวส่งเพียงชุดเดียว ส่วนตัวรับมีได้สูงสุดไม่เกิน 10 ชุดเท่านั้น การสื่อสารข้อมูลเป็นแบบทิศทางเดียว ทำให้เกิดปัญหาในการสื่อสารข้อมูลจากตัวรับกลับมาที่ตัวส่ง เพราะจะต้องเพิ่มคู่สายสัญญาณรับส่งอีกหนึ่งคู่สาย เพื่อทำให้สามารถสื่อสารข้อมูลแบบสองทิศทางได้ ดังรูปที่ 2.18



รูปที่ 2.18 แสดงการเชื่อมต่อของ RS-422-A

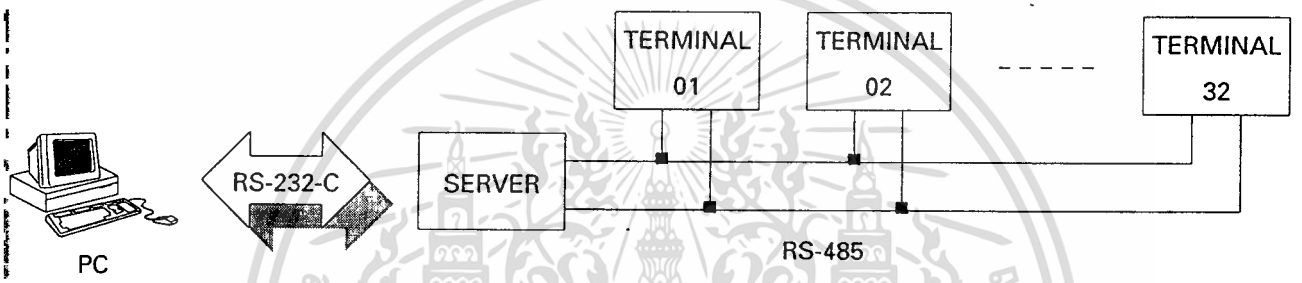


รูปที่ 2.19 แสดงการเชื่อมต่อของ RS-485

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 3
โครงสร้างของระบบ

โครงสร้างของระบบควบคุมการเข้าออกโดยใช้บัตรแม่เหล็ก ประกอบด้วยส่วนประมวล 2 หน่วย ได้แก่ เซิร์ฟเวอร์ และ เทอร์มินอล ทำงานเป็นอิสระต่อกัน และสามารถติดต่อสื่อสารข้อมูลระหว่างกันได้โดยผ่านระบบโครงข่ายที่เชื่อมต่อกันตามมาตรฐาน RS-485 ดังแสดงในรูปที่ 3.1



รูปที่ 3.1 แสดงระบบควบคุมการเข้าออกโดยใช้บัตรแม่เหล็ก

จากรูปที่ 3.1 การสื่อสารข้อมูอนุกรมแบบโครงข่าย RS-485 สำหรับระบบควบคุมการเข้าออกโดยใช้บัตรแม่เหล็ก เป็นการสื่อสารข้อมูลระหว่าง เซิร์ฟเวอร์ 1 โมดูล กับ เทอร์มินอลจำนวน 32 โมดูล (สูงสุด) ส่วนไมโครคอมพิวเตอร์นั้นสื่อสารข้อมูลกับเซิร์ฟเวอร์ผ่านทาง RS-232-C

3.1 การออกแบบสร้างและพัฒนาส่วนประมวลผลของระบบ

3.1.1 เทอร์มินอลโมดูล (Terminal Module)

3.1.1.1 หน่วยประมวลผลกลาง (CPU)

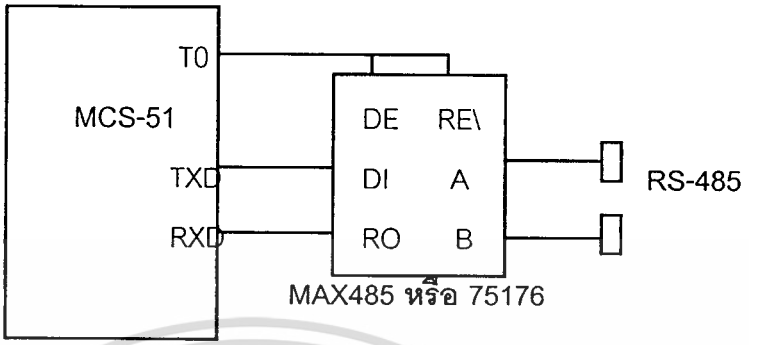
เทอร์มินอลใช้ชิพหน่วยประมวลผลกลางเบอร์ 89C2051 ซึ่งอยู่ในตระกูล MCS-51 ไมโครคอนโทรลเลอร์ เป็นหน่วยประมวลผลกลาง

3.1.1.2 ส่วนรับส่งข้อมูลอนุกรม RS-485

เทอร์มินอลติดต่อกับเซิร์ฟเวอร์ผ่าน MAX485 หรือ 75176 RS-485 transceiver ซึ่งเชื่อมต่อกับขา TXD และ RXD ของ MCS-51 และใช้ขาสัญญาณ T0 เป็นสัญญาณเลือกการทำงาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

แบบรับ หรือส่งข้อมูล การรับส่งข้อมูลเป็น Half Duplex โดยใช้สายสัญญาณรับส่งข้อมูลเพียงสองเส้น สำหรับรับส่งข้อมูลกับเซิร์ฟเวอร์ดังรูปที่ 3.2



รูปที่ 3.2 แสดงรูปวงจรของส่วนรับส่งข้อมูลอนุกรม RS-485

3.1.1.3 **ส่วนอินพุท-เอาท์พุทของเทอร์มินอล**

1) อินพุท เป็นส่วนเชื่อมต่อกับชุดรูดบัตร โดยสัญญาณจากชุดรูดบัตรมี 3 เส้น ได้แก่

- สัญญาณข้อมูล (DATA) เชื่อมต่อเข้ากับ P1.0 ของ MCS-51
- สัญญาณนาฬิกา (CLOCK) เชื่อมต่อเข้ากับ P1.1 ของ MCS-51
- สัญญาณแสดงการเริ่มต้นของการรูดบัตร (STR หรือ PRECENT) เชื่อมต่อเข้ากับ P1.2 ของ MCS-51

2) เอาท์พุท เป็นส่วนเชื่อมต่อกับเอาท์พุทรีเลย์ โดยจะทำงานเมื่อเทอร์มินอลได้รับคำสั่งจากเซิร์ฟเวอร์ว่า อนุญาตให้ผู้ถือบัตรแม่เหล็กทำการใช้เทอร์มินอลตัวนั้นๆ ได้ ซึ่งผู้ถือบัตรต้องมีข้อมูลบัตรอยู่ในฐานข้อมูลแล้ว

3.1.1.4 **ส่วนแสดงผล**

1) LED แสดงผล ต่างๆได้แก่

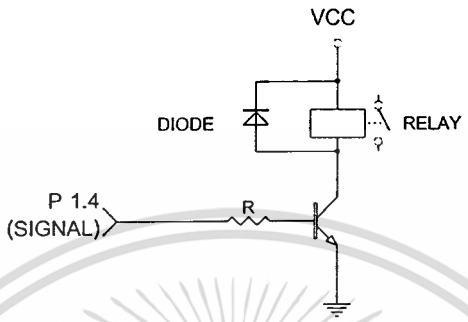
- LED ที่เชื่อมต่อกับพอร์ต 1 (P1) ของหน่วยประมวลผลกลางเพื่อใช้ดูสถานะต่างๆ ของเทอร์มินอล
- LED แสดงไฟเลี้ยงของเทอร์มินอล
- LED แสดงข้อมูลเข้าหน่วยประมวลผลจากการรูดบัตร
- LED แสดงการส่งสัญญาณเอาท์พุทรีเลย์ (SIGNAL)

2) BUZZER ส่งสัญญาณเสียงเพื่อแสดงสถานะต่างๆ ของเทอร์มินอล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.1.1.5 **ส่วนวงจรสั่งการอุปกรณ์ภายนอก**

เป็นวงจรขับรีเลย์เพื่อให้เปิด (ON) หรือปิด (OFF) โดยได้สัญญาณมาจาก P1.4 (SIGNAL) ของหน่วยประมวลผลกลางดังรูปที่ 3.3



รูปที่ 3.3 แสดงวงจรขับรีเลย์เพื่อสั่งเปิดหรือปิด

3.1.2 **เซิร์ฟเวอร์โมดูล** (Server Module)

3.1.2.1 **หน่วยประมวลผลกลาง** (CPU)

เซิร์ฟเวอร์ใช้ชิพหน่วยประมวลผลกลางเบอร์ 89C51 หรือ 89C52 ซึ่งอยู่ในตระกูล MCS-51 ไมโครคอนโทรลเลอร์ เป็นหน่วยประมวลผลกลาง

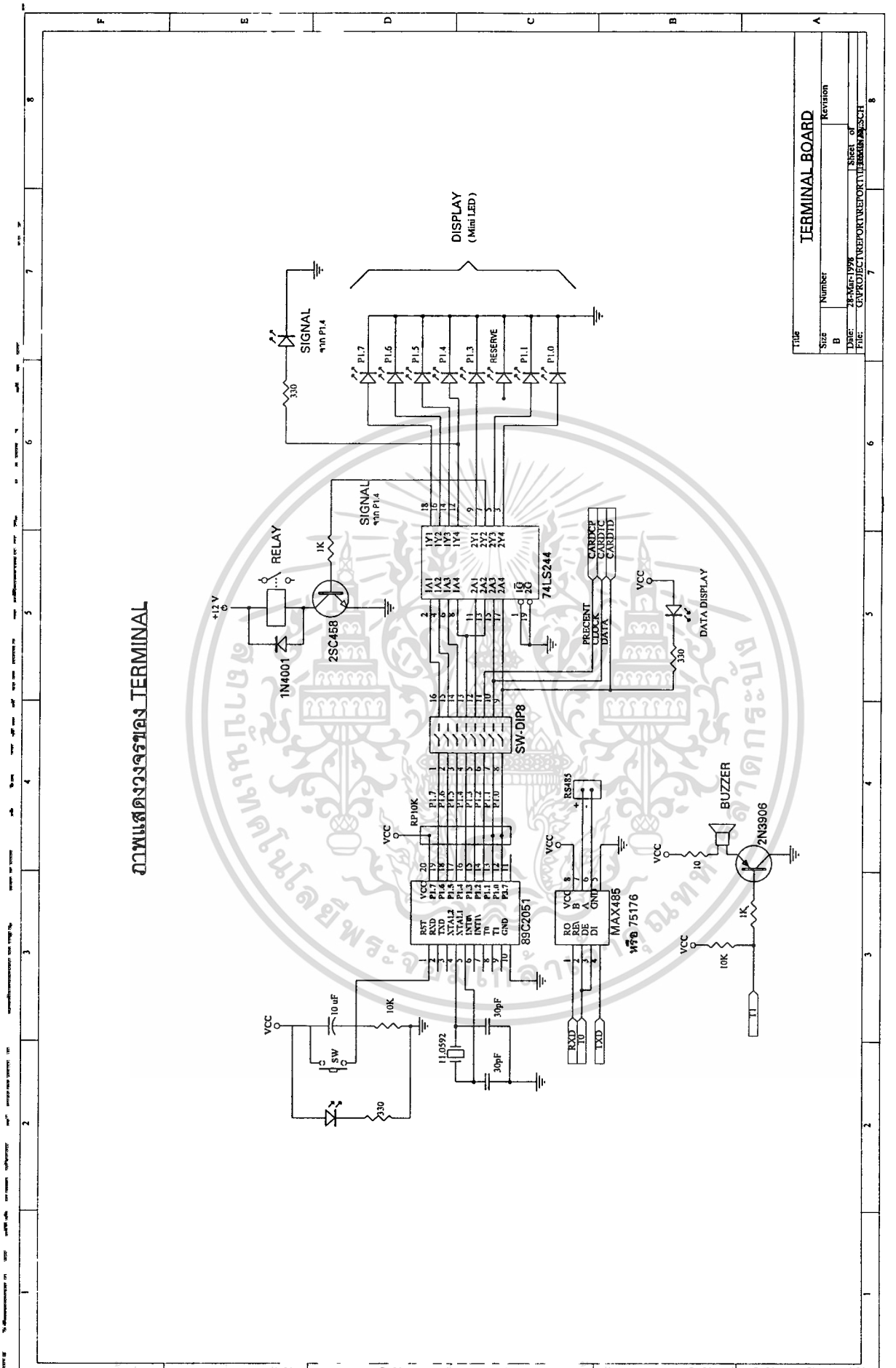
3.1.2.2 **หน่วยความจำ** (Memory)

หน่วยความจำภายนอกของเซิร์ฟเวอร์ ประกอบด้วย SRAM เบอร์ 6264 ซึ่งเป็นหน่วยความจำขนาด 8 KBYTE (8K x 8) จำนวน 2 ตัว รวมแล้วเป็นหน่วยความจำขนาด 16 KBYTE เป็นส่วนเก็บข้อมูลที่เทอร์มินอลส่งมาให้ และข้อมูลที่ได้รับจากไมโครคอมพิวเตอร์

3.1.2.3 **ส่วนวงจรรีเซท**

วงจรรีเซท บนเซิร์ฟเวอร์ จะใช้ชิพเบอร์ MAX1232 โดยจะใช้สำหรับการรีเซทระบบ ซึ่งการรีเซทแบบนี้จะให้ผลดีกว่าการรีเซทด้วยวงจร RC ทั่ว ๆ ไป โดยจะทำการรีเซททั้งช่วง POWER UP และ DOWN จึงแน่ใจได้ว่าระบบจะไม่เกิดสภาพรวนจากการรีเซทอย่างแน่นอน การรีเซทจาก MAX1232 นี้จะตั้งระดับไว้ที่ 10 % ของ VCC คือ ระบบจะทำงานทันทีเมื่อไฟ VCC ต่ำกว่า 4.5 V

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.4 แสดงวงจรของเทอร์มินอล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.1.2.4 **ส่วนรับส่งข้อมูลอนุกรม** RS-485

เซิร์ฟเวอร์ติดต่อกับเทอร์มินอลต่าง ๆ ผ่าน MAX485 หรือ 75176 RS-485 transceiver ซึ่งการเชื่อมต่อเหมือนกันกับเทอร์มินอล ดังรูปที่ 3.2

3.1.2.5 **ส่วนรับส่งข้อมูลอนุกรม** RS-232-C

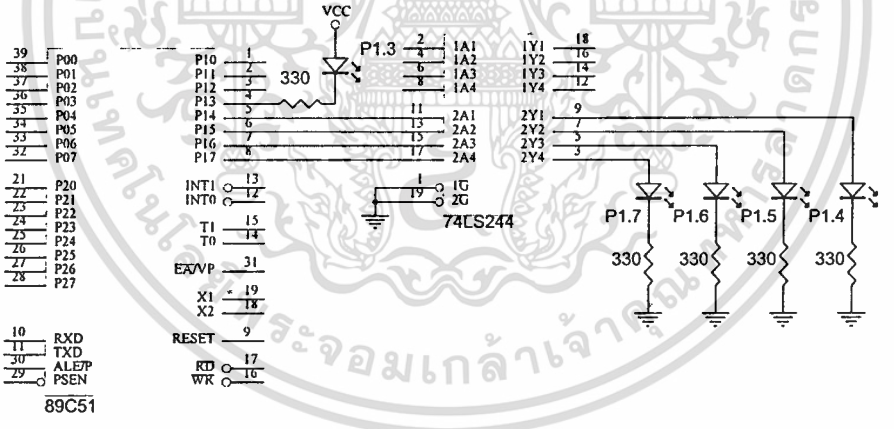
เซิร์ฟเวอร์ติดต่อกับไมโครคอมพิวเตอร์ผ่าน MAX232 ตามมาตรฐานการรับส่งข้อมูลอนุกรมแบบ RS-232-C

3.1.2.6 **ส่วนอินพุทของเซิร์ฟเวอร์**

เป็นส่วนเชื่อมต่อกับชุดรูดบัตร โดยใช้สัญญาณอินพุทจำนวน 3 สัญญาณเหมือนกับการเชื่อมต่อในเทอร์มินอล

3.1.2.7 **ส่วนแสดงผล**

ใช้ LED เป็นตัวแสดงผลโดยต่อเข้ากับ P1.4, P1.5, P1.6 และ P1.7 เพื่อแสดงสถานะของเซิร์ฟเวอร์ในขณะต่าง ๆ และ P1.3 ดังรูปที่ 3.5

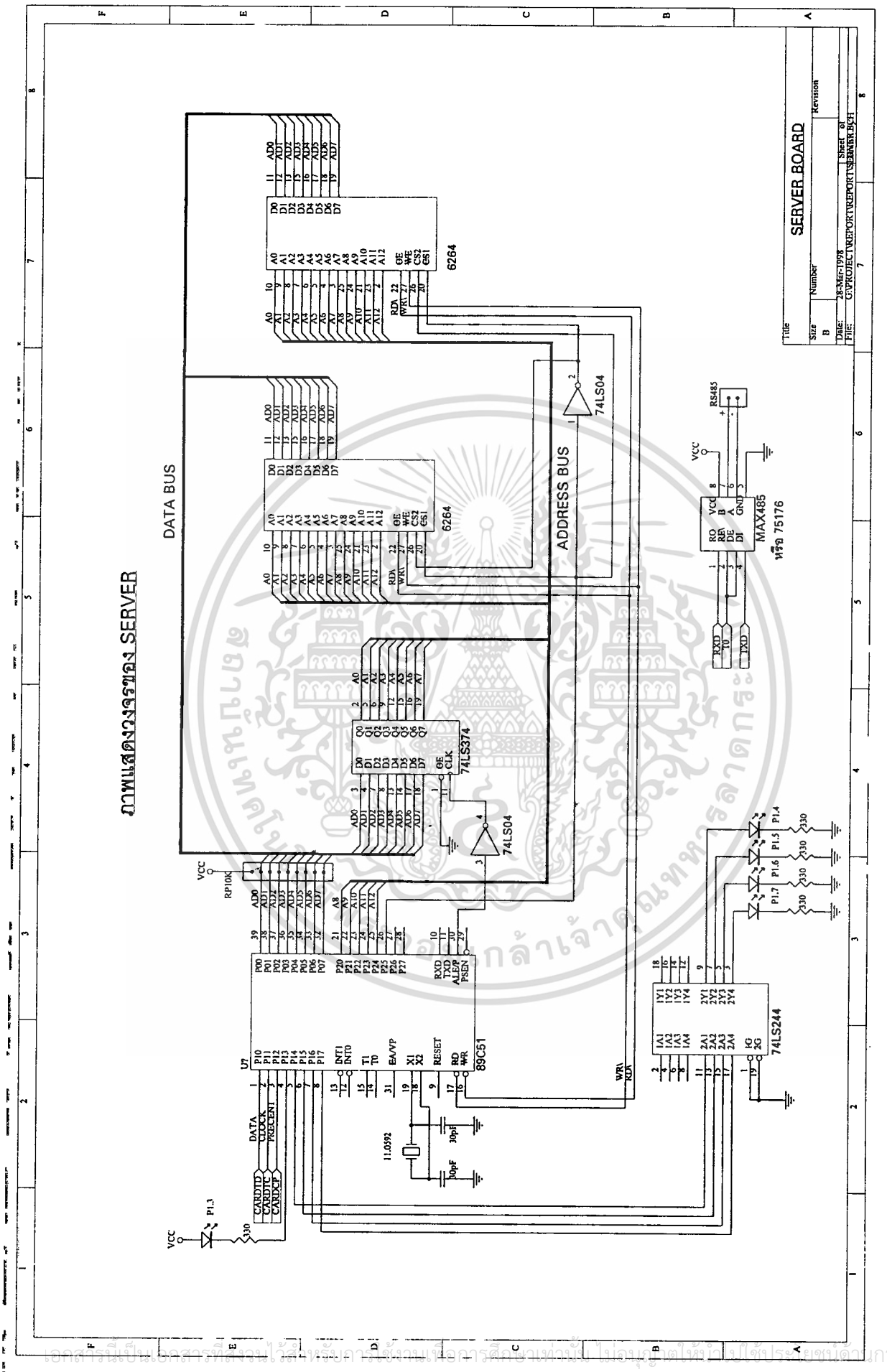


รูปที่ 3.5 แสดงการเชื่อมต่อส่วนแสดงผลของเซิร์ฟเวอร์

3.1.3 **<u>วงจรไฟเลี้ยง</u>** (Power Supply)

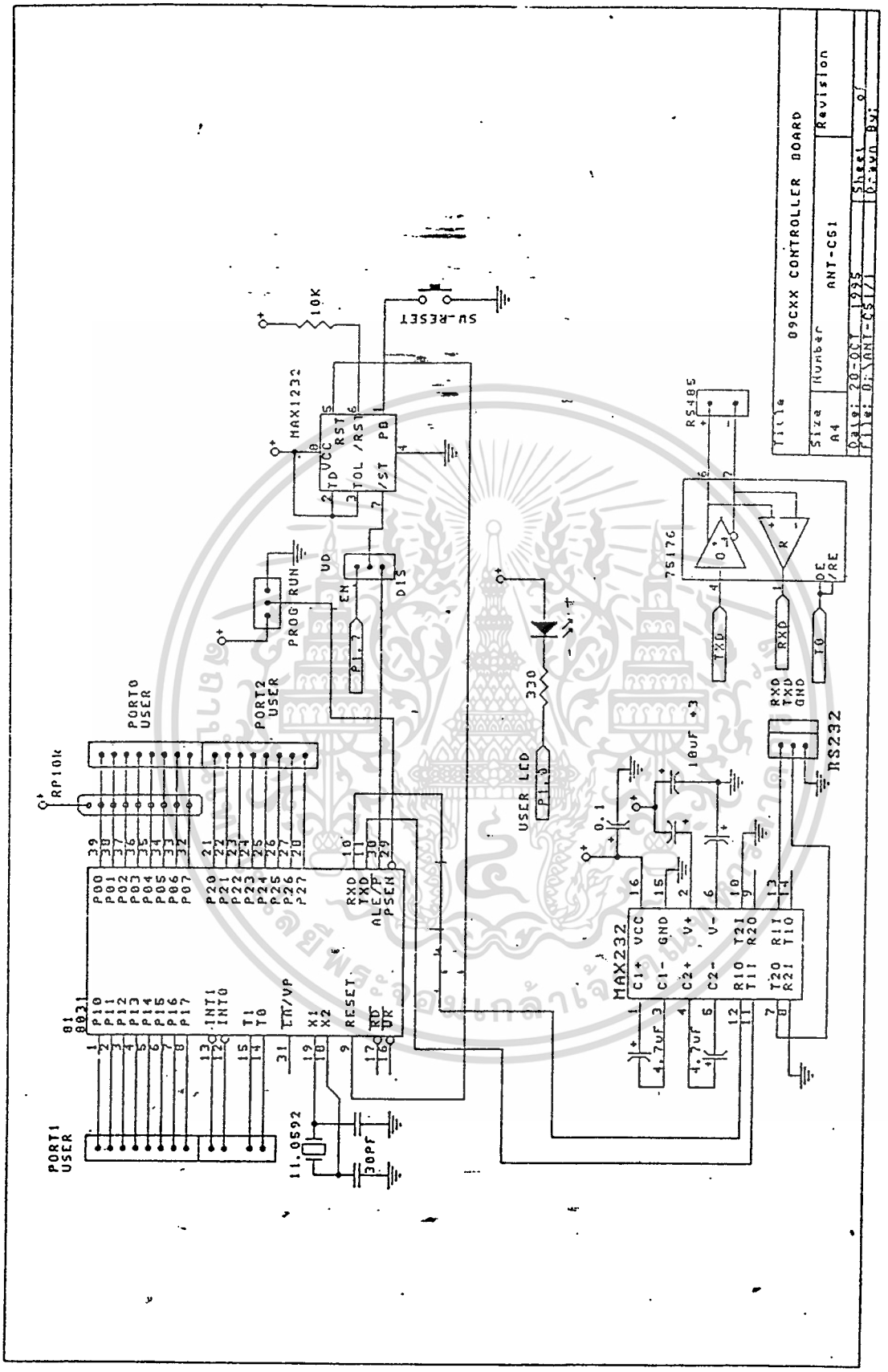
วงจรไฟเลี้ยงมีการจ่ายไฟเลี้ยง 2 ค่า คือ ขนาด 12 โวลต์ สำหรับจ่ายให้แก่รีเลย์และขนาด 5 โวลต์ สำหรับจ่ายให้แก่วงจรของเทอร์มินอล ดังรูปที่ 3.9

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



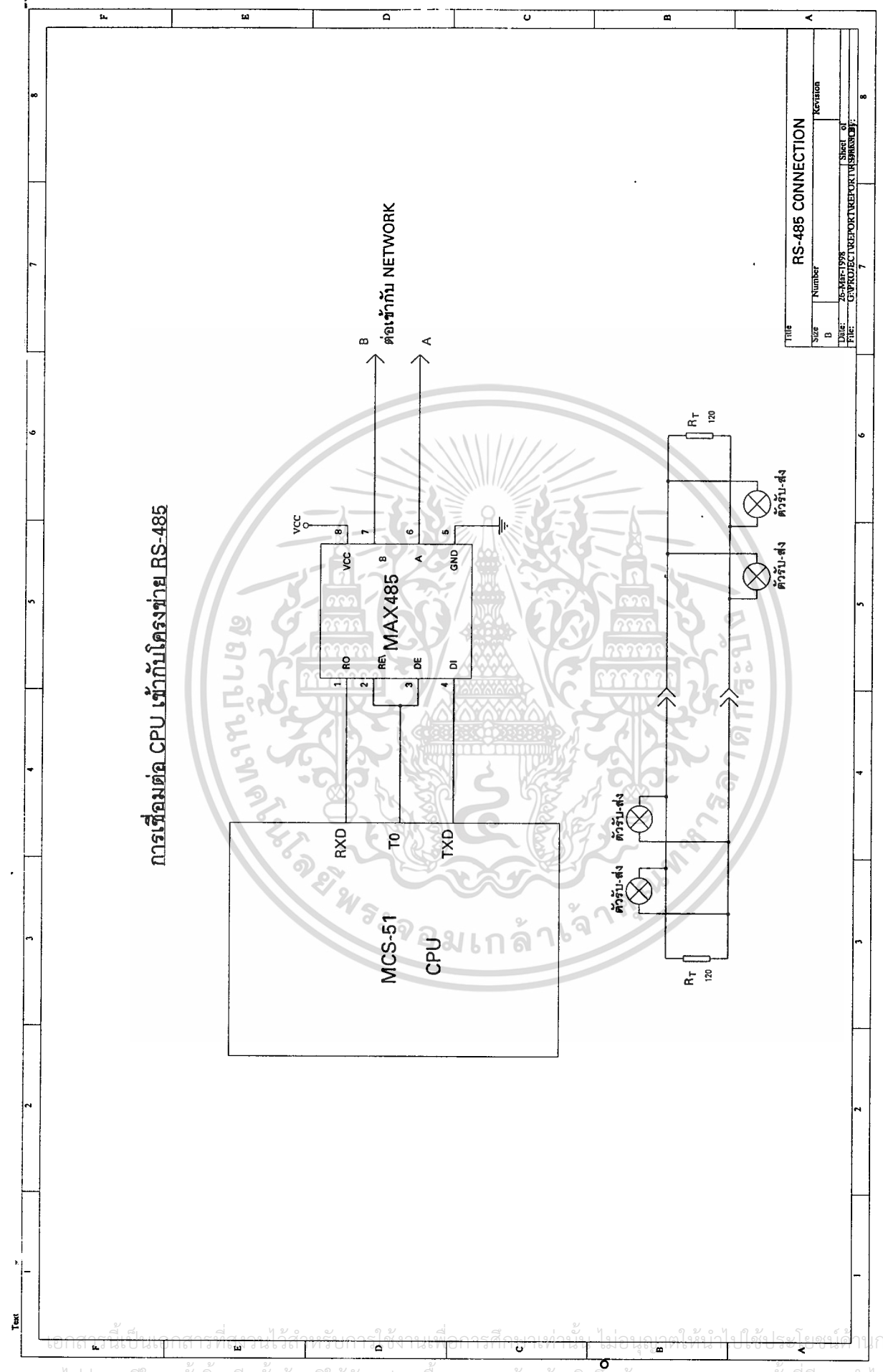
รูปที่ 3.6 แสดงวงจรของเซิร์ฟเวอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



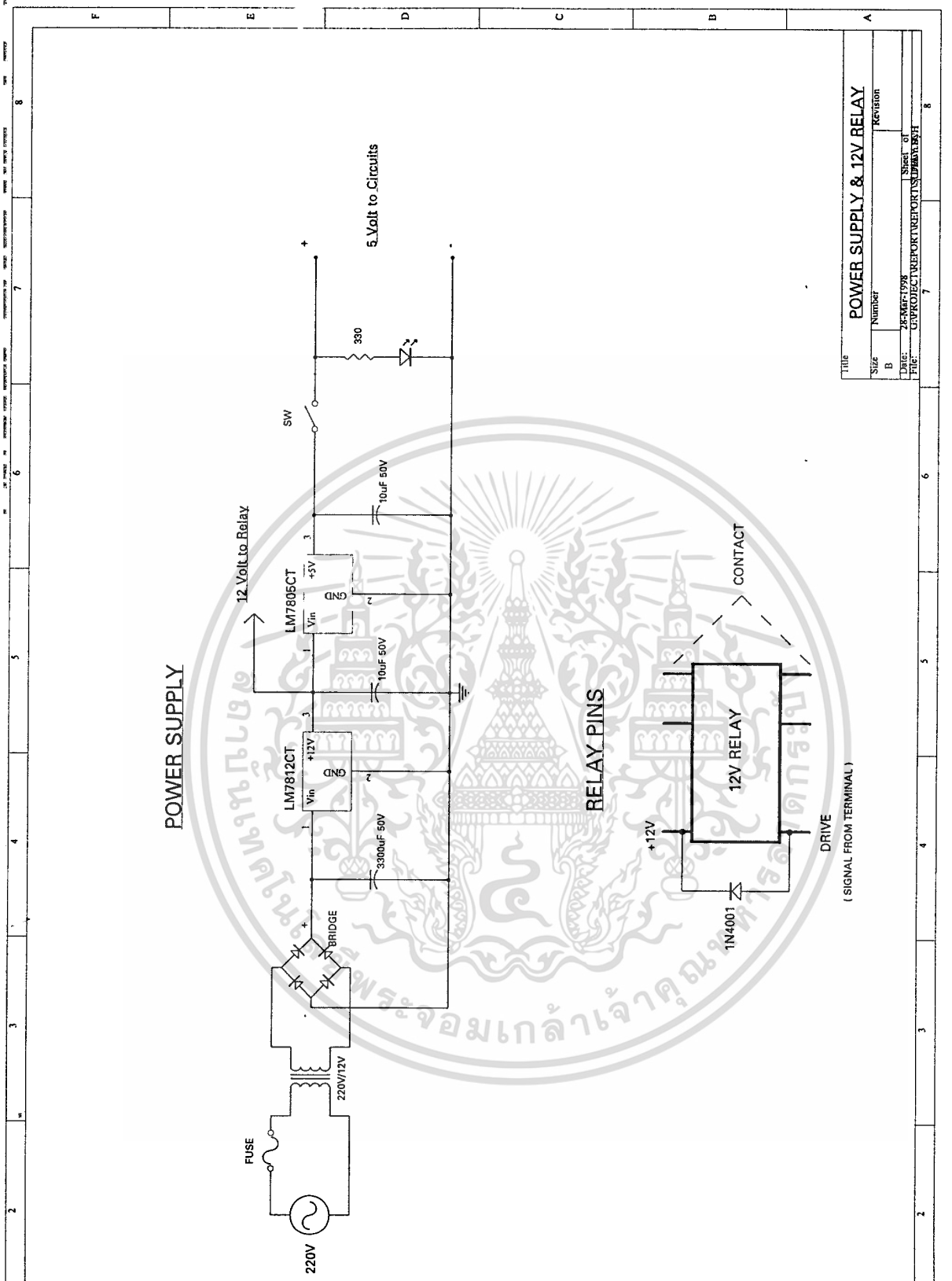
รูปที่ 3.7 แสดงวงจรของเซิร์ฟเวอร์โดยใช้บอร์ด ANT-C51

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่3.8 แสดงการเชื่อมต่อ CPU เข้ากับโครงข่าย RS-485

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.9 แสดงวงจรไฟเลี้ยง (Power Supply) และรีเลย์ (Relay)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.2 การพัฒนาโปรแกรมจัดการและควบคุมระบบ

เนื่องจากระบบควบคุมการเข้าออกโดยใช้บัตรแม่เหล็กประกอบด้วยส่วนประมวลผล 2 หน่วย ทำงานอิสระต่อกัน และสามารถรับส่งข้อมูลแบอนุกรมระหว่างหน่วยประมวลผลได้ โดยผ่านมาตรฐาน RS-485 ระหว่างเซิร์ฟเวอร์กับเทอร์มินอล และ มาตรฐาน RS-232-C ระหว่างไมโครคอมพิวเตอร์กับเซิร์ฟเวอร์ ทำให้ต้องแบ่งส่วนการพัฒนาโปรแกรมควบคุมระบบออกเป็น 3 ส่วนดังนี้คือ

3.2.1 <u>โปรแกรมจัดการและควบคุมระบบส่วนเทอร์มินอลโมดูล</u>

โฟล์วชาร์ตการทำงาน แสดงในรูปที่ 3.10 ถึง 3.19

3.2.2 <u>โปรแกรมจัดการและควบคุมระบบส่วนเซิร์ฟเวอร์โมดูล</u>

โฟล์วชาร์ตการทำงาน แสดงในรูปที่ 3.20 ถึง 3.23

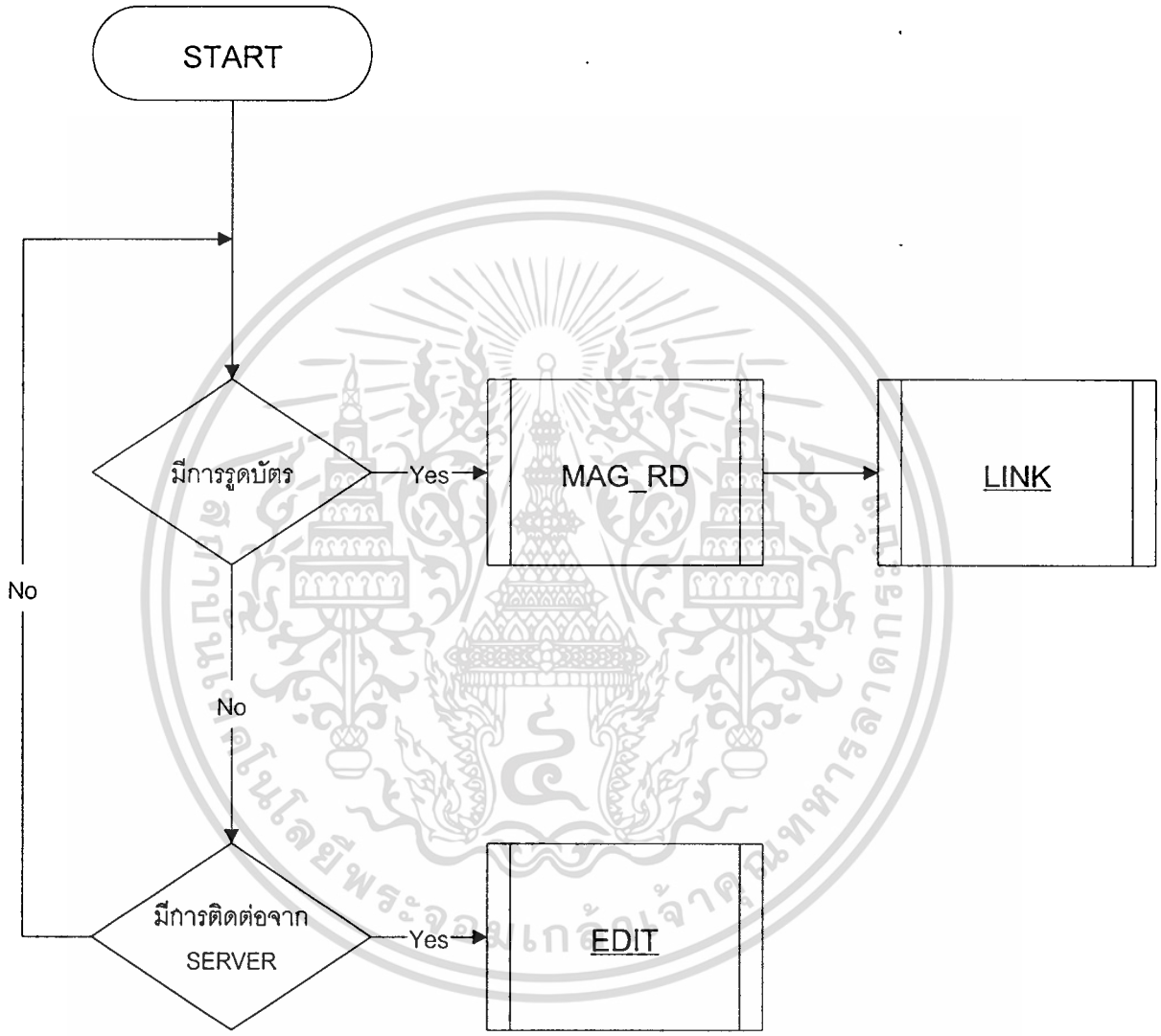
รูปที่ 3.24 และ 3.25 แสดง MEMORY MAP ของหน่วยความจำภายนอก

3.2.3 <u>โปรแกรมจัดการและควบคุมระบบส่วนไมโครคอมพิวเตอร์</u>

รูปแสดงการทำงาน แสดงในรูปที่ 3.26 ถึง 3.37

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

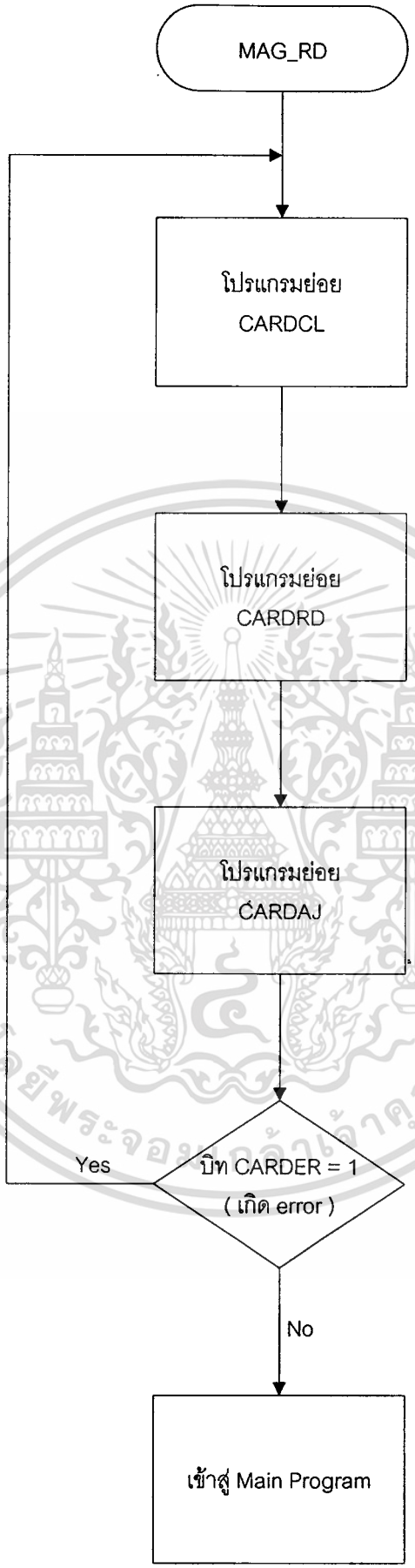
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



TERMINAL MAIN PROGRAM

รูปที่ 3.10 **แสดงโฟล์วชาร์ตของโปรแกรมหลักของเทอร์มินอลโมดูล**

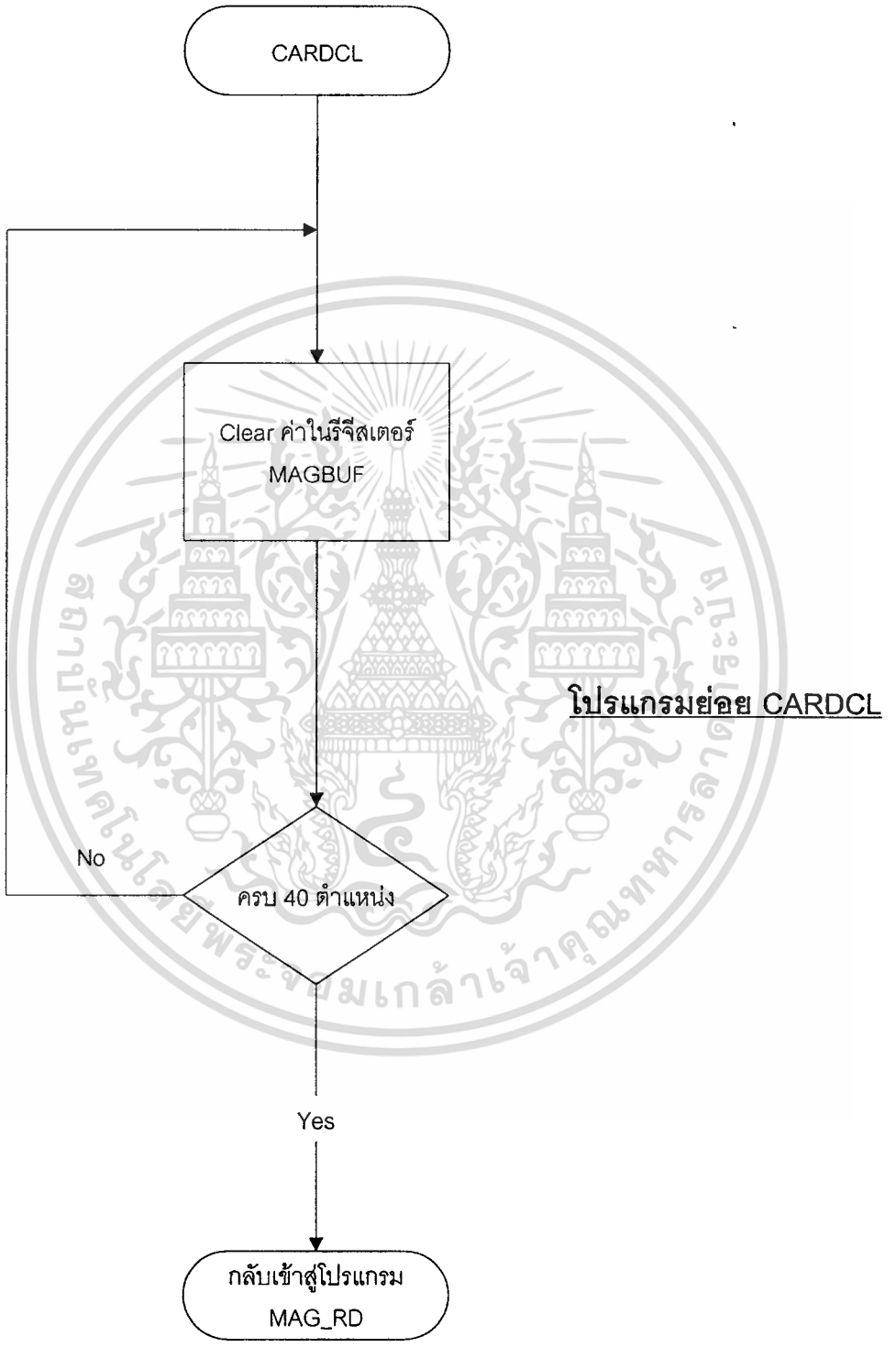
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



โปรแกรม MAG_RD

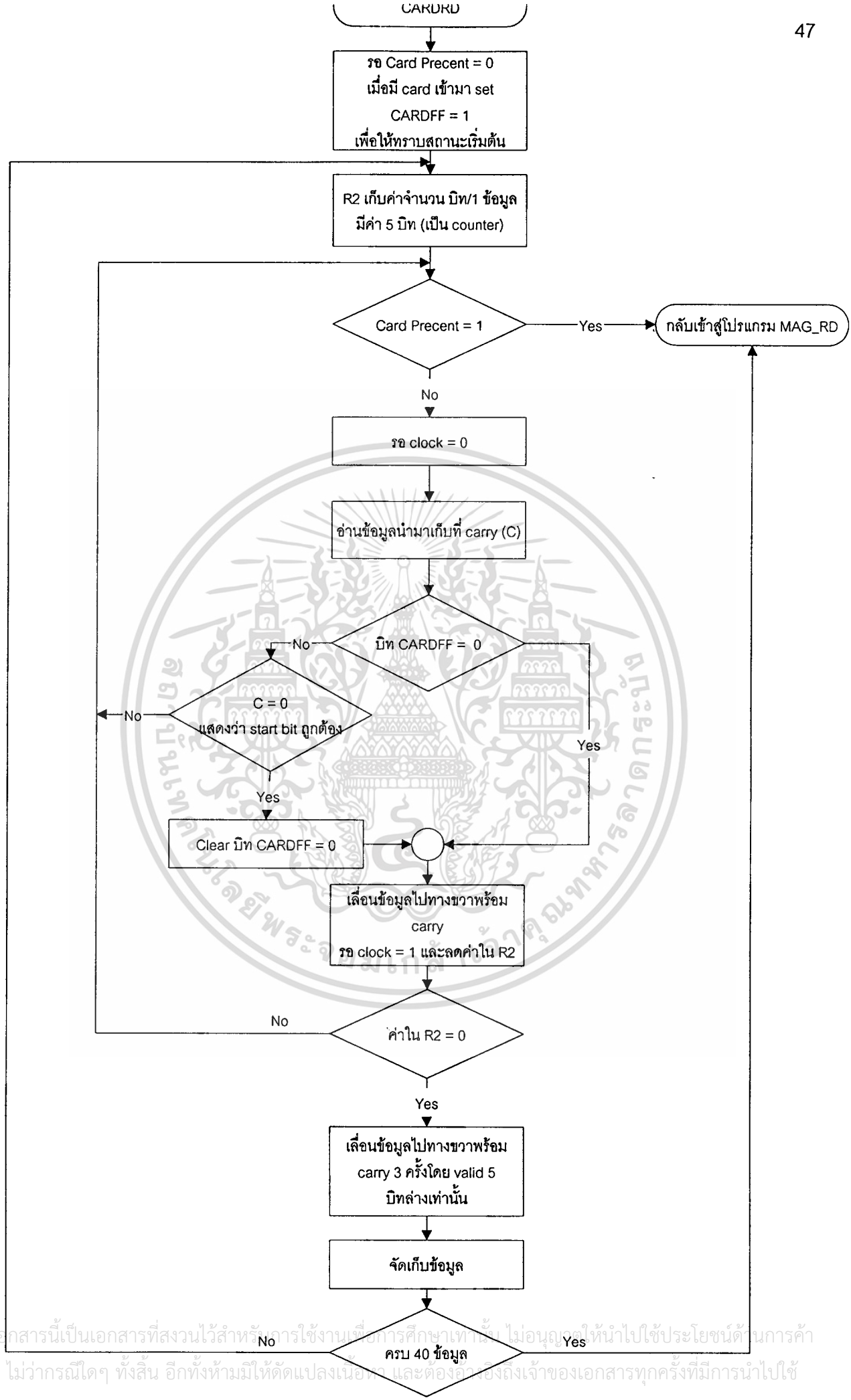
รูปที่ 3.11 แสดงโฟล์วชาร์ตของโปรแกรมบริการการรูดบัตร (โปรแกรมย่อย MAG_RD)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



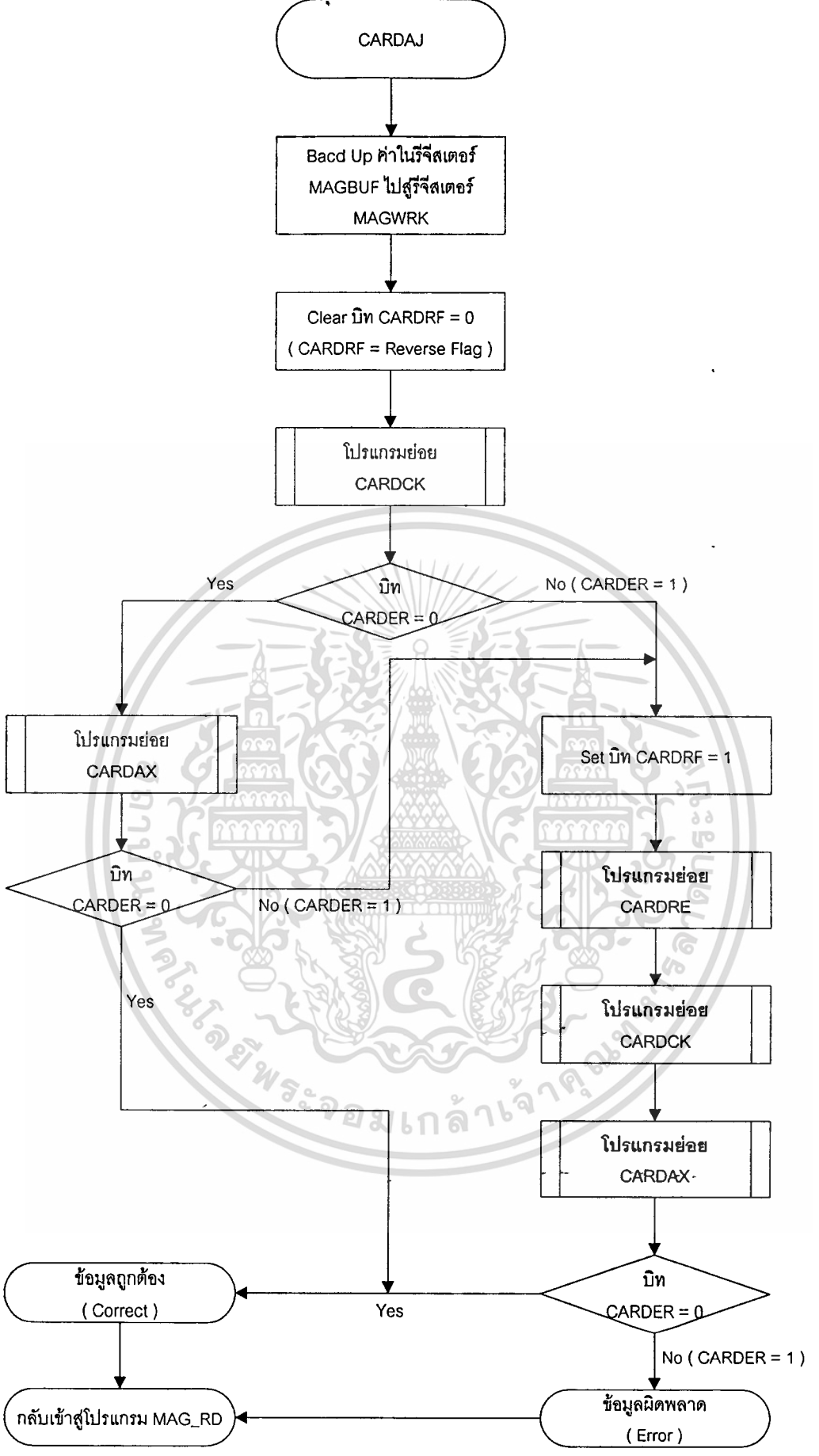
รูปที่ 3.12 **แสดงโฟล์วชาร์ตโปรแกรมย่อย** CARDCL

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

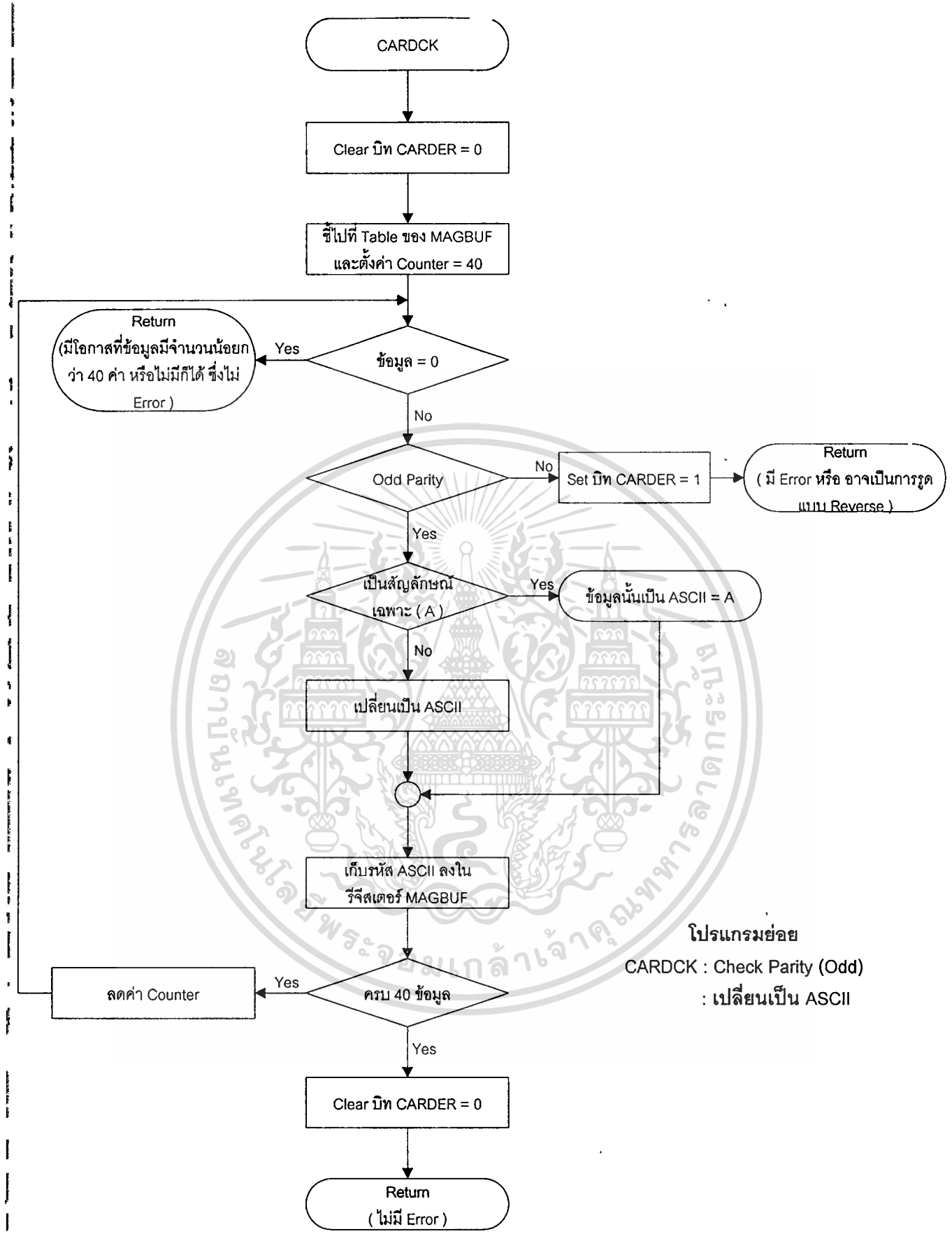
รูปที่ 3.13 แสดงโฟล์วชาร์ตโปรแกรมย่อย CARDRD



โปรแกรมย่อย CARDAJ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น

รูปที่ 3.14 แสดงโฟล์วชาร์ตโปรแกรมย่อย CARDAJ



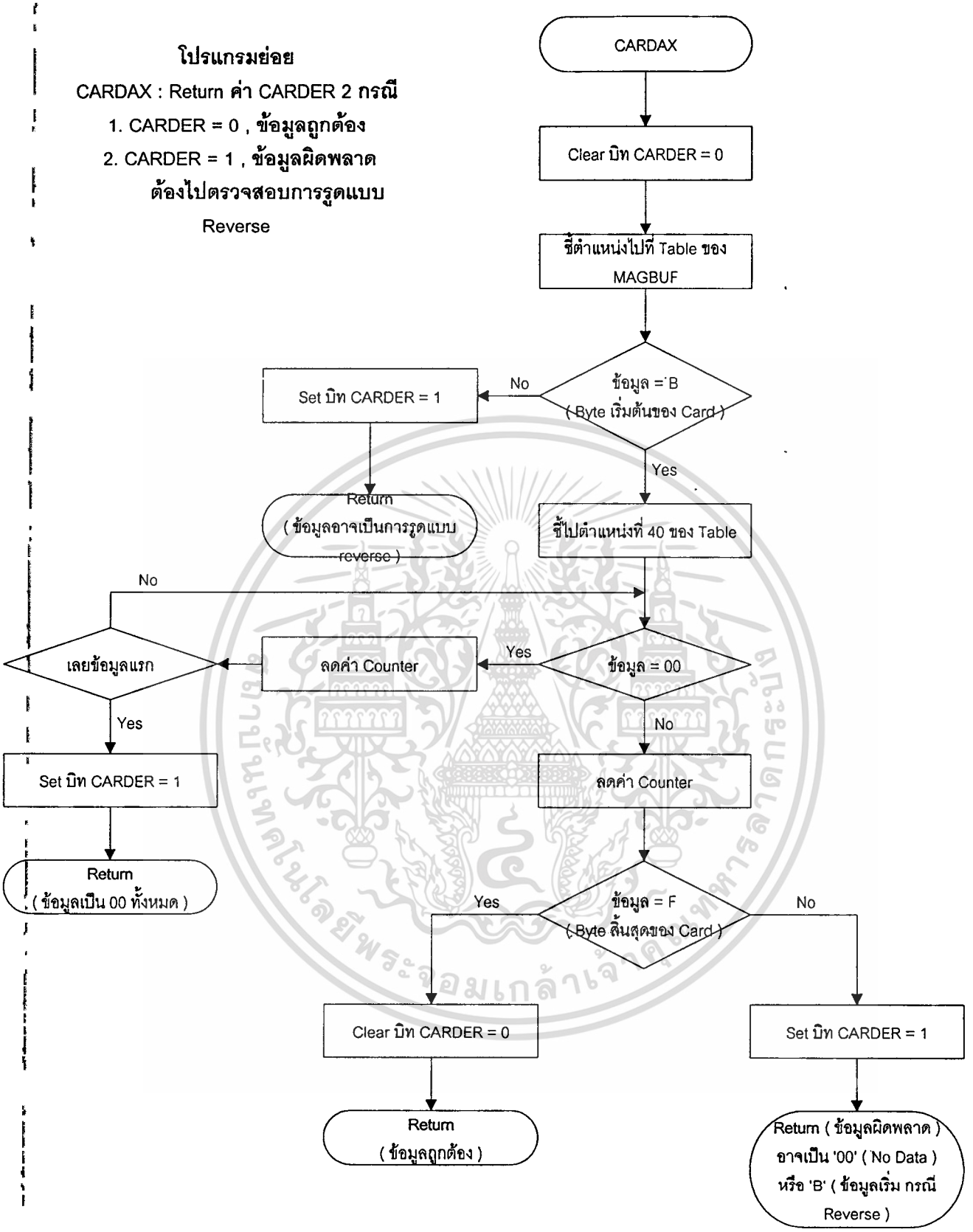
รูปที่ 3.15 แสดงโฟล์วชาร์ตโปรแกรมย่อย CARDCK

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โปรแกรมย่อย

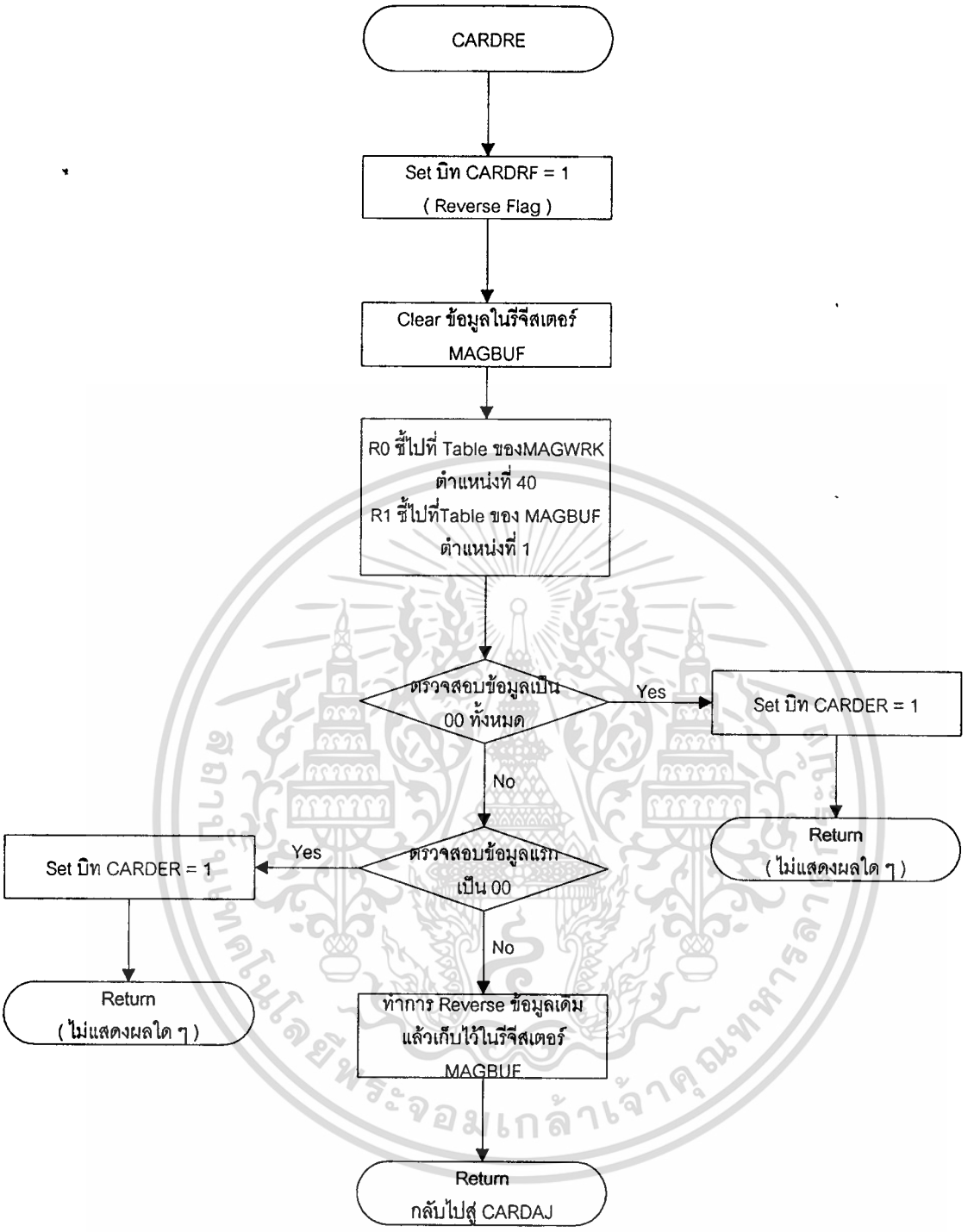
CARDAX : Return ค่า CARDER 2 กรณี

1. CARDER = 0 , ข้อมูลถูกต้อง
2. CARDER = 1 , ข้อมูลผิดพลาด ต้องไปตรวจสอบการรูดแบบ Reverse



รูปที่ 3.16 แสดงโฟล์วชาร์ตโปรแกรมย่อย CARDAX

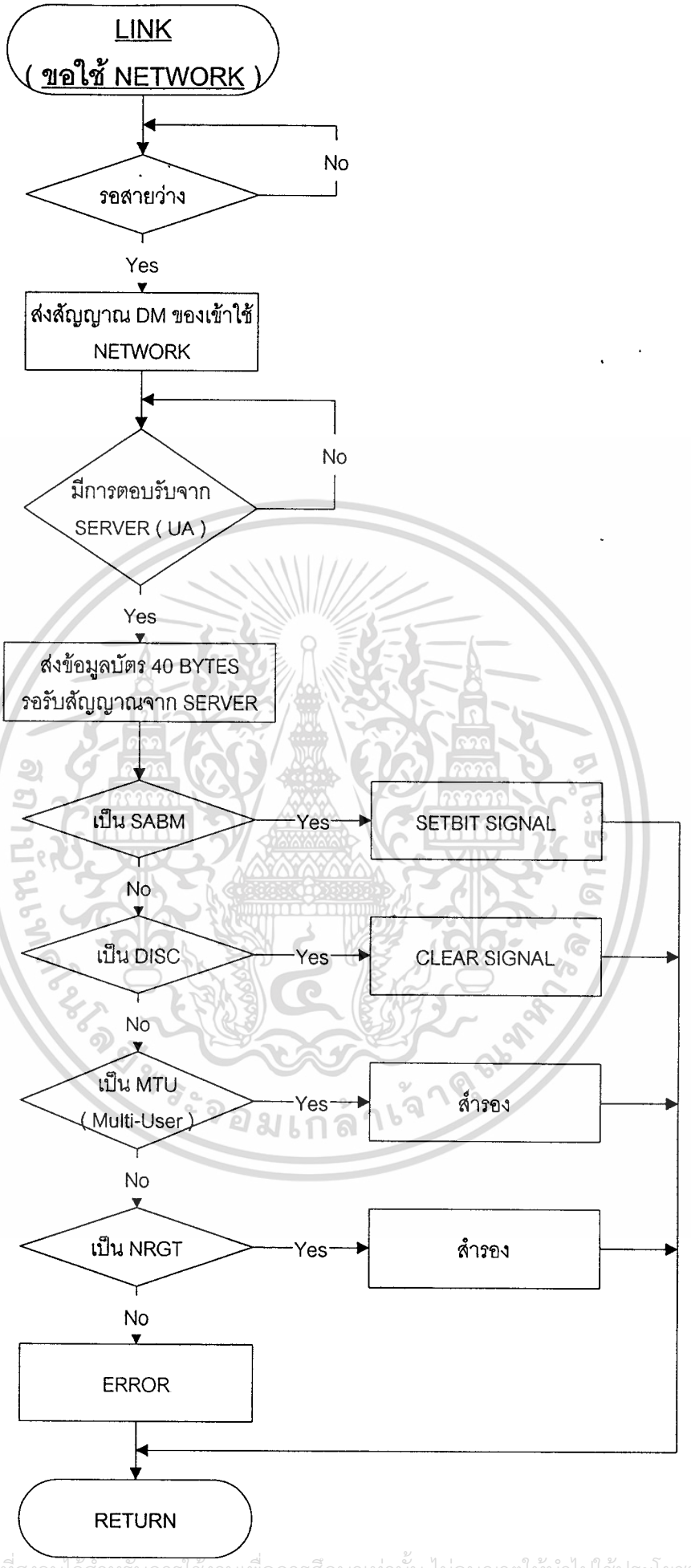
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



โปรแกรมย่อย CARDRE
ทำการ Reverse ข้อมูล

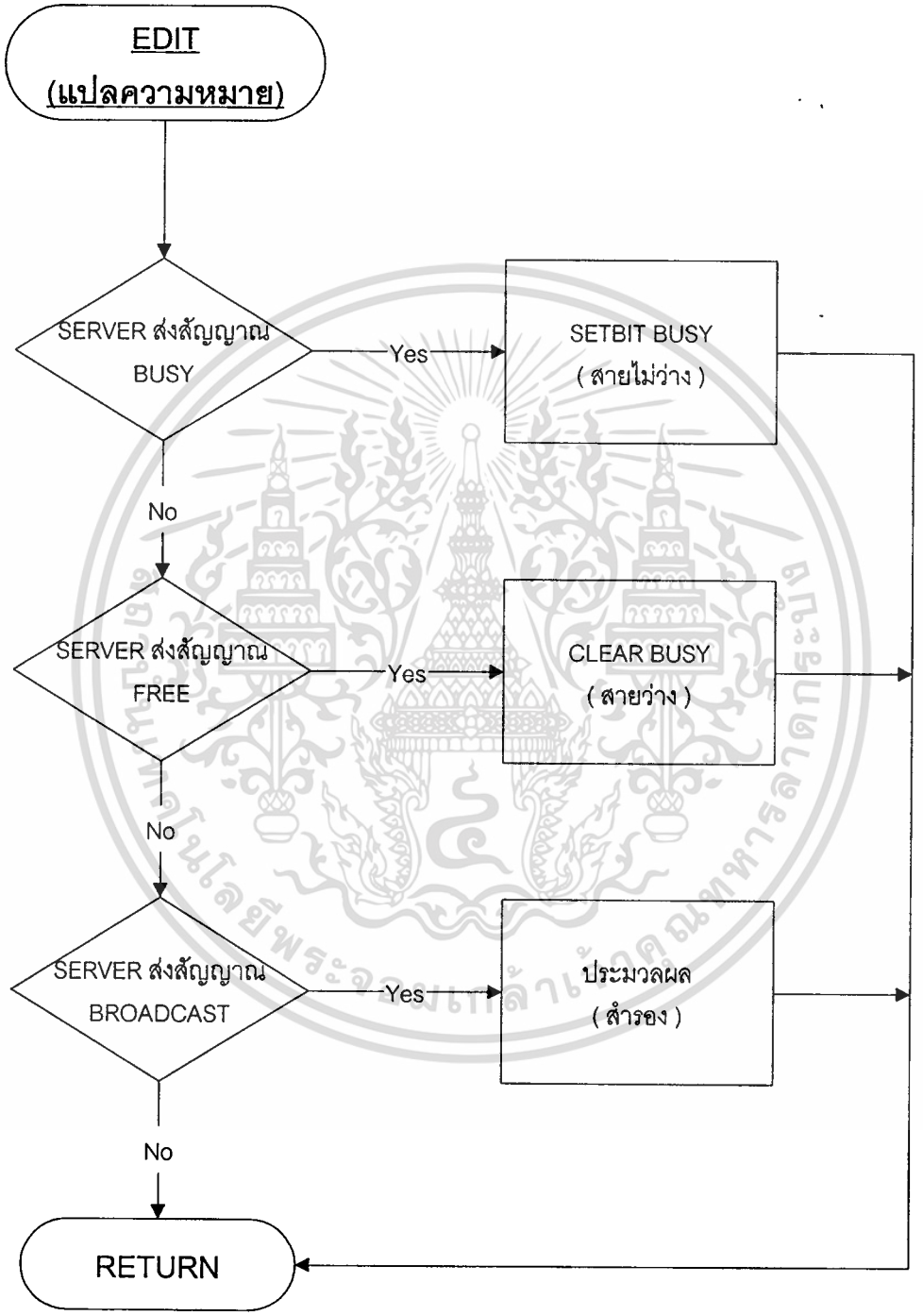
รูปที่ 3.17 แสดงโฟล์วชาร์ตโปรแกรมย่อย CARDRE

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



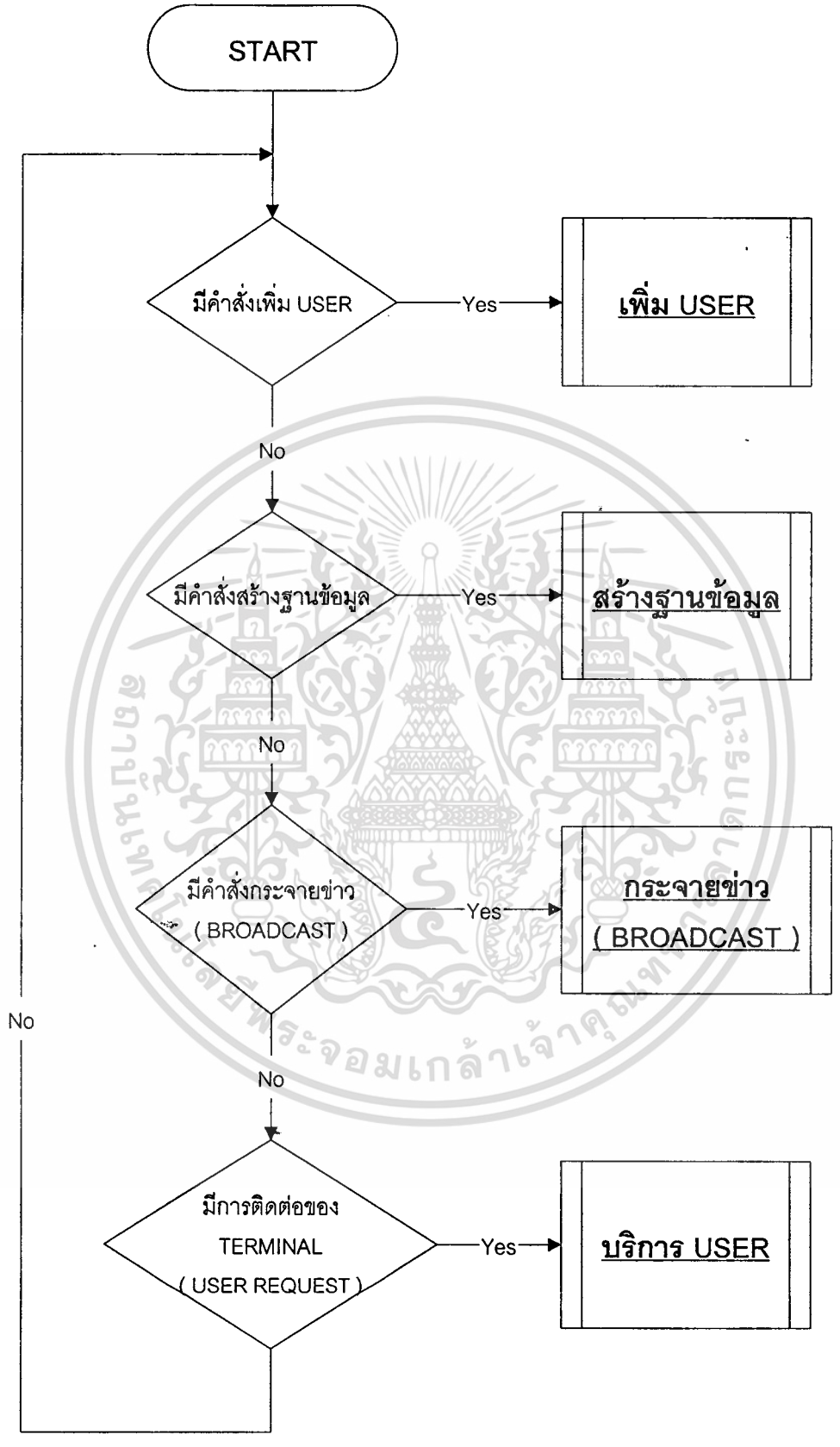
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น

รูปที่ 3.18 แสดงโฟล์วชาร์ตโปรแกรมบริการการขอใช้ NETWORK (LINK)



รูปที่ 3.19 แสดงโฟล์วชาร์ตโปรแกรมการแปลความหมาย (EDIT)

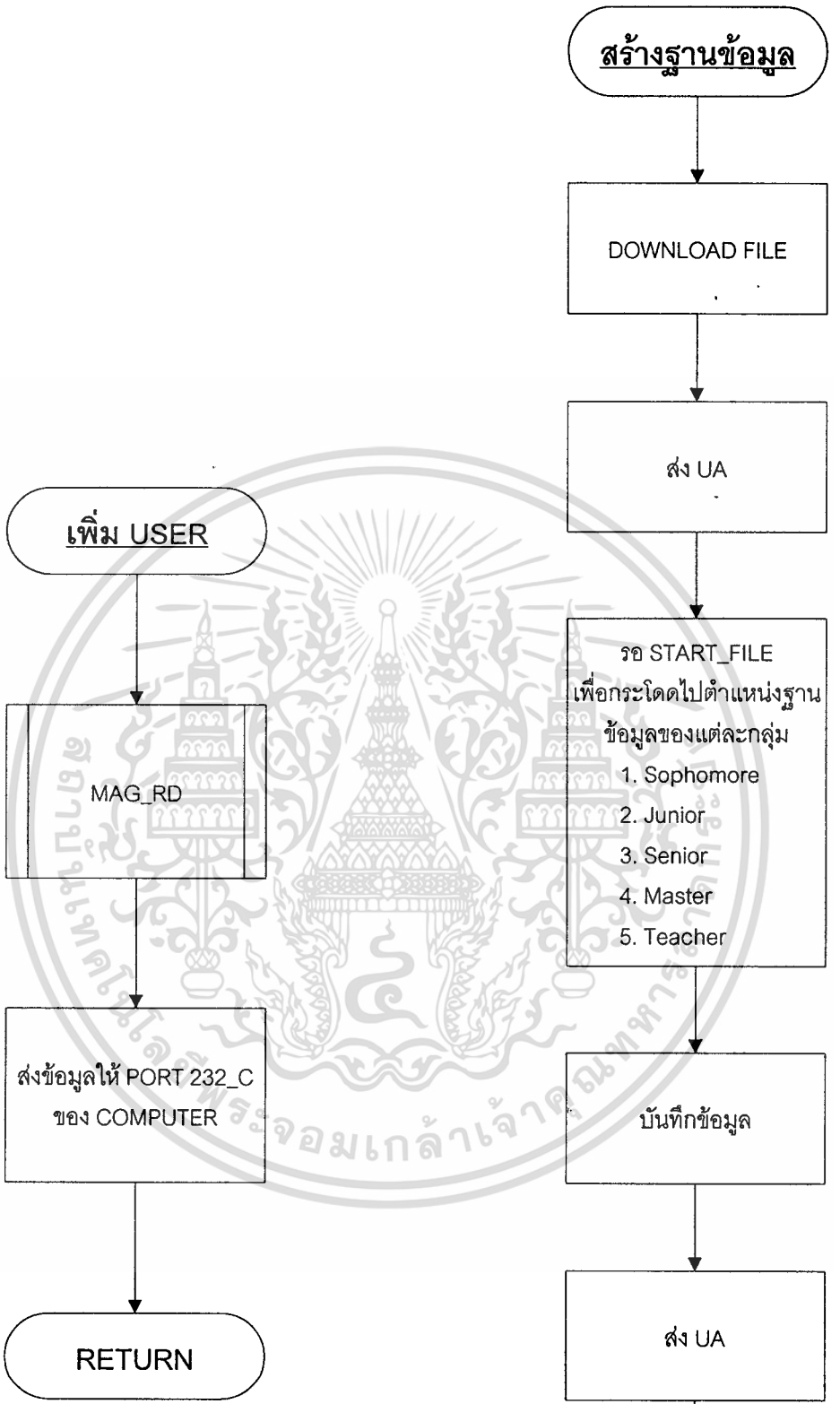
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



SERVER MAIN PROGRAM

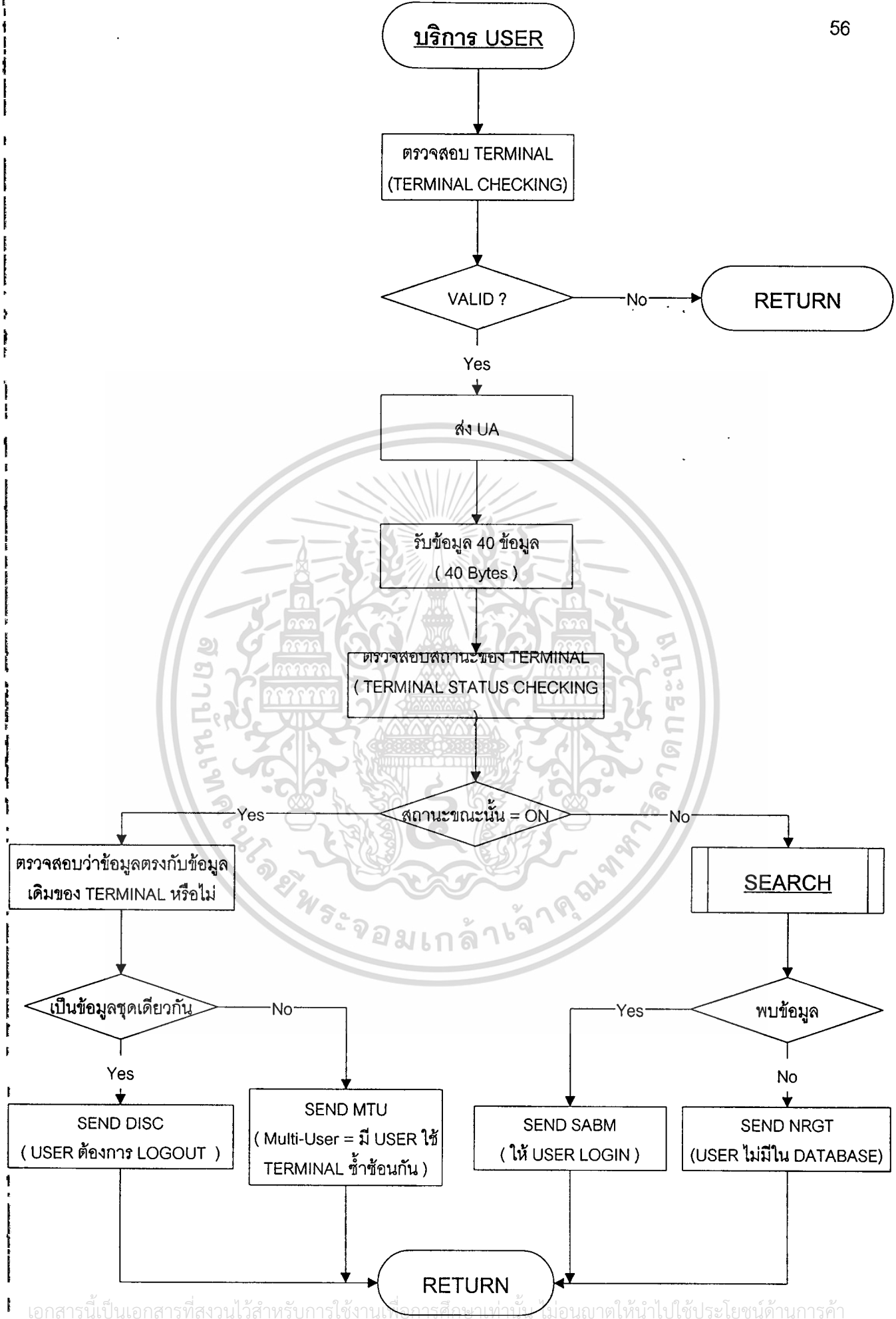
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งไม่อนุญาตให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 3.20 แสดงโฟล์วชาร์ตโปรแกรมหลักของเซิร์ฟเวอร์โมดูล



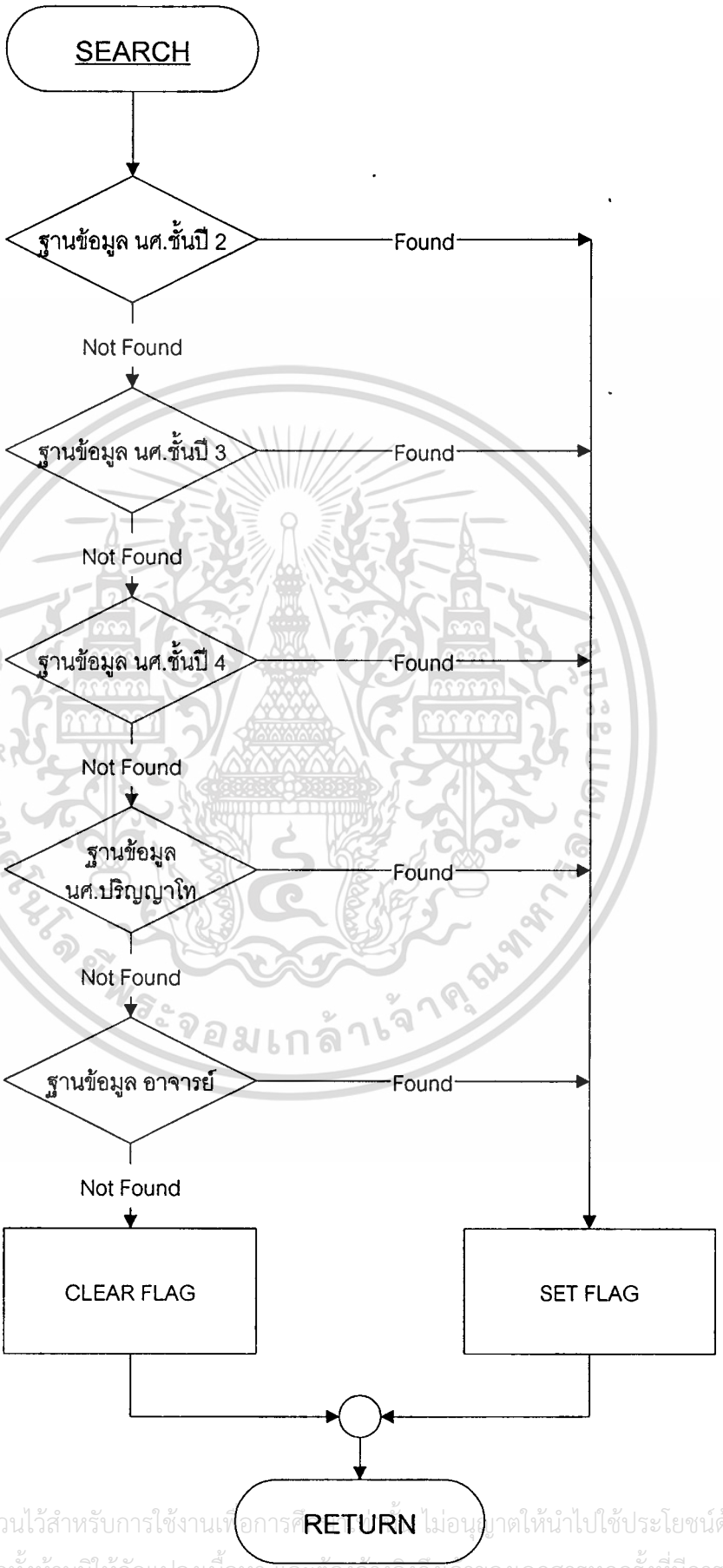
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของผลงานทุกครั้งที่มีการนำไปใช้

รูปที่ 3.21 แสดงโฟล์วชาร์ตโปรแกรมบริการการเพิ่ม USER และโปรแกรมการสร้างฐานข้อมูล



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

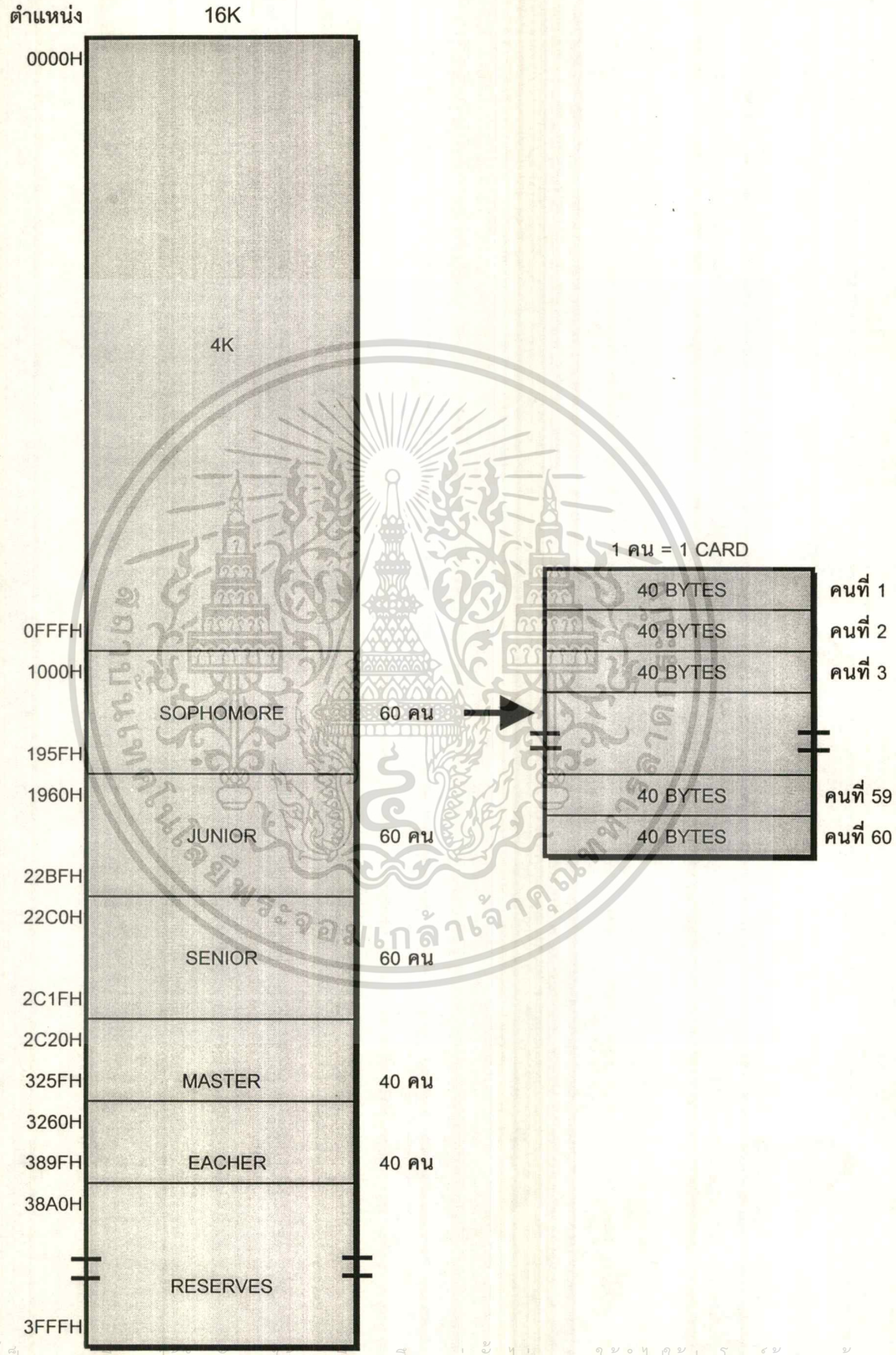
รูปที่ 3.22 แสดงโฟล์วชาร์ตโปรแกรมบริการ USER



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 3.23 แสดงโฟล์วชาร์ตโปรแกรมบริการการค้นหาข้อมูล

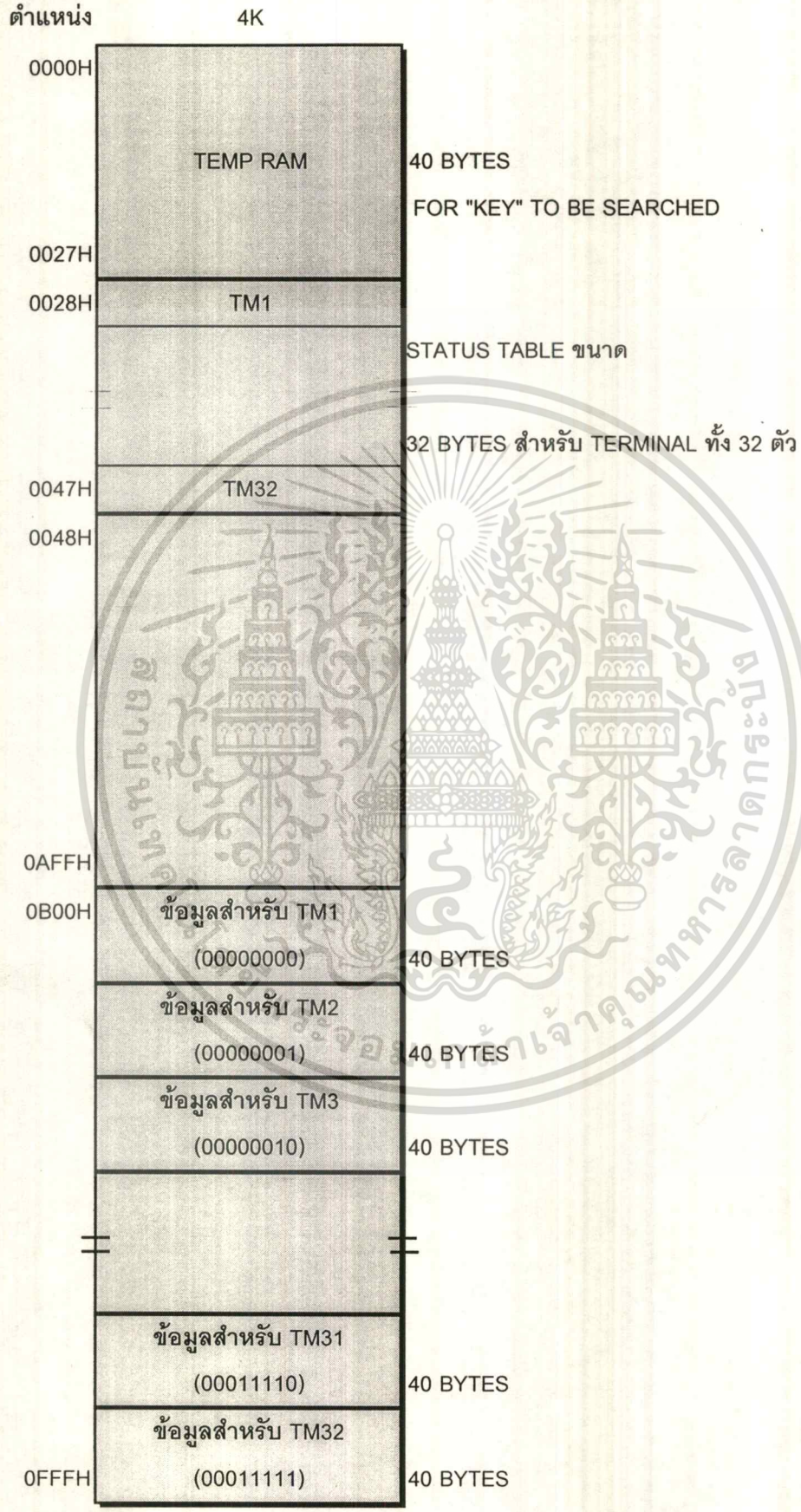
MEMORY MAP ของ EXTERNAL RAM ขนาด 16 K ที่ SERVER



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 3.24 แสดง MEMORY MAP ของหน่วยความจำภายนอกขนาด 16 KBYTE ของเซิร์ฟเวอร์

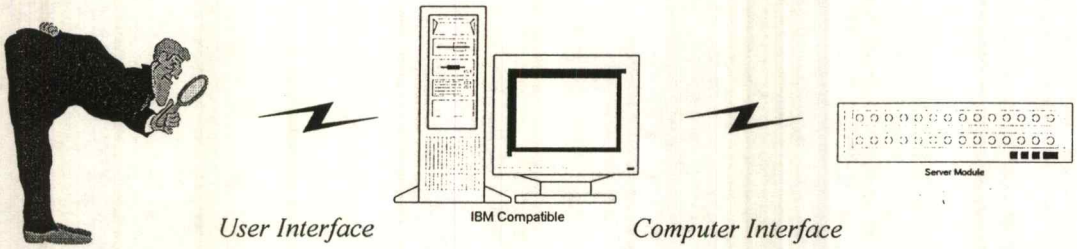
4K บนของ EXTERNAL RAM ใช้สำหรับ TEMPORARY STATION DATA



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 3.25 แสดง MEMORY MAP ของหน่วยความจำภายนอกขนาด 4 KBYTE บน

โปรแกรมสนับสนุนการติดต่อกับคอมพิวเตอร์และการติดต่อกับผู้ใช้ของส่วนเซิร์ฟเวอร์



รูปที่ 3.26 แสดงแผนภาพการเชื่อมต่อ (Interface) ระหว่างผู้ใช้ (User) กับไมโครคอมพิวเตอร์และเซิร์ฟเวอร์

ในส่วนของการติดต่อ (Interfacing) กับเซิร์ฟเวอร์โมดูลโดยคอมพิวเตอร์ พบว่าจำเป็นจะต้องมีส่วนของโปรแกรมซึ่งสนับสนุนการทำงานดังต่อไปนี้

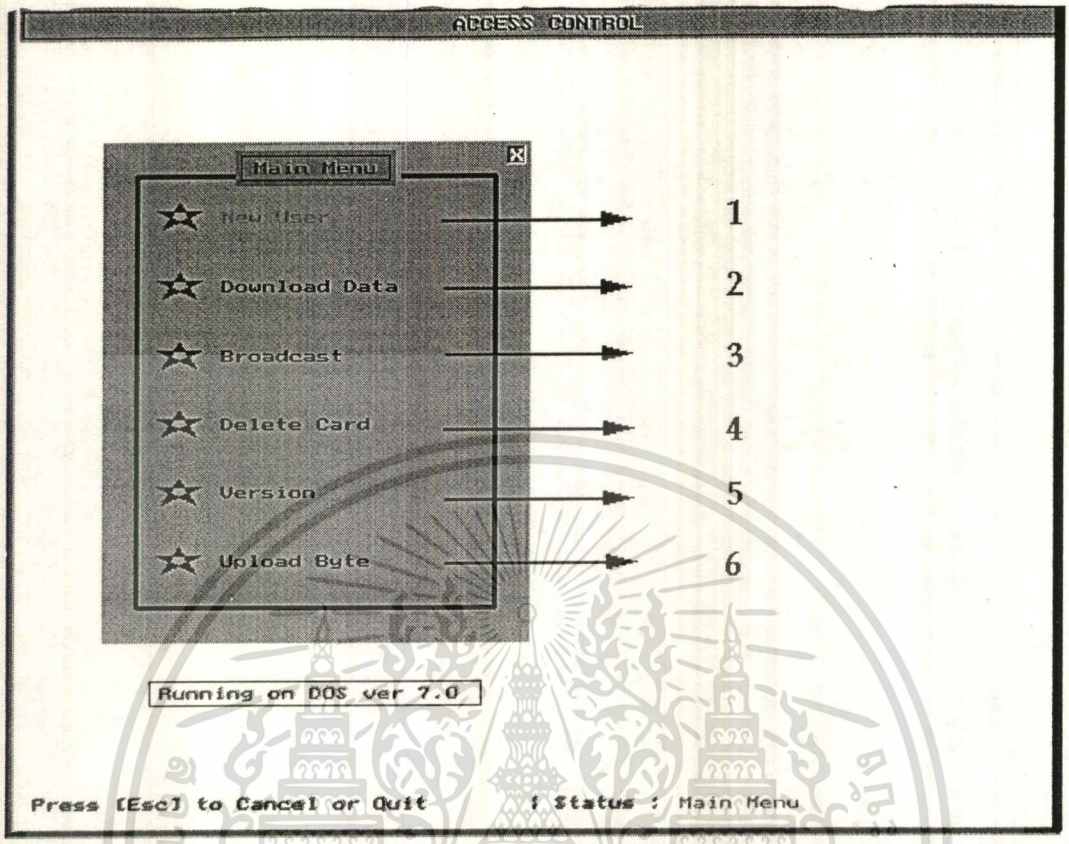
1. โปรแกรมต้องสามารถติดต่อผ่านทางพอร์ตอนุกรมของคอมพิวเตอร์ โดยสามารถเปลี่ยน Baud Rate ได้ เช่น 9600 , 2400 , 1200 เป็นต้น

2. โปรแกรมต้องสามารถจำลองการทำงานแบบมัลติโปรเซสเซอร์ของไมโครคอนโทรลเลอร์ตระกูล MCS-51 ได้ นั่นคือ โปรแกรมต้องสามารถส่งข้อมูลอนุกรมแบบ 9 Bit-UART ได้ และสามารถกำหนดบิทที่ 9 ได้จากโปรแกรม

3. เนื่องจากการทำงานบนคอมพิวเตอร์มีความรวดเร็วกว่าในส่วนของเซิร์ฟเวอร์ประมาณ 10 เท่า ทำให้การส่งข้อมูลในบางจังหวะไม่สัมพันธ์กัน ดังนั้นการทำงานระหว่างคอมพิวเตอร์และเซิร์ฟเวอร์จึงต้องมีลักษณะเป็นแบบ Hand Shaking นั่นคือ จะมีการส่งสัญญาณรับรู้ (Acknowledge) ทุกๆ ครั้งที่ประมวลผลการทำงานส่วนของตัวมันเองเสร็จสิ้น

โปรแกรม ACCESS CONTROL (Access.exe) คอมไพล์มาจากไฟล์ของปาสคาล "Access.pas" ได้สนับสนุนในส่วนของคอมพิวเตอร์อินเตอร์เฟสดังต่อไปนี้ (ดูรูปประกอบ)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.27 แสดงเมนูของโปรแกรมที่ใช้ในการเชื่อมต่อกับเซิร์ฟเวอร์
โดยไมโครคอมพิวเตอร์

1. เพิ่มข้อมูลของผู้ใช้คนใหม่ในฐานข้อมูล : ADD NEW USER

   โปรแกรมส่วนนี้มีหน้าที่บันทึกข้อมูลบัตร ซึ่งได้บันทึกโดยการรูดบัตรแม่เหล็กบนเซิร์ฟเวอร์ลงในฐานข้อมูลที่ผู้ใช้ได้เลือกไว้

2. สร้างฐานข้อมูลบนเซิร์ฟเวอร์ : DOWNLOAD DATA

   โปรแกรมส่วนนี้จะทำการส่งข้อมูลบนฐานข้อมูลที่ผู้ใช้เลือกไว้ลงไปบันทึกในหน่วยความจำภายนอกของเซิร์ฟเวอร์โมดูล มีความจำเป็นในการทำงานทุกครั้ง เพราะจะช่วยให้เซิร์ฟเวอร์ตรวจสอบข้อมูลของบัตรที่รูดบนเทอร์มินอลได้ (มักใช้ช่วงเริ่มการทำงานเท่านั้น ---startup)

3. กระจายข่าว : BROADCAST

   โปรแกรมส่วนนี้จะช่วยให้เทอร์มินอลทุกๆ ตัวสามารถรับรู้คำสั่งที่ถูกส่งออกไปได้ ซึ่งมีประโยชน์ในด้านความรวดเร็วในการส่งข้อมูลให้ทุกๆ เทอร์มินอลพร้อมๆ กัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4. ลบข้อมูลผู้ใช้ออกจากฐานข้อมูล : DELETE CARD

โปรแกรมส่วนนี้ใช้ลบข้อมูลผู้ใช้ออกจากฐานข้อมูล ซึ่งควรเป็นหน้าที่ของผู้จัดการระบบ จะทำให้การรูดบัตรของผู้ใช้ที่ถูกลบข้อมูลไม่ประสบความสำเร็จ

5. รายละเอียดของโปรแกรม : VERSION

เป็นส่วนที่ให้รายละเอียดเกี่ยวกับโปรแกรม เช่น ชื่อของโปรแกรม การเปลี่ยนแปลงครั้งสุดท้าย ผู้เขียน อาจารย์ที่ปรึกษา เป็นต้น

6. อ่านข้อมูลที่รับเข้ามาทางพอร์ตอนุกรม : UPLOAD BYTE

โปรแกรมส่วนนี้มีประโยชน์ในการช่วยพิจารณาความผิดพลาดต่างๆ ที่เกิดขึ้นในระบบ ใช้ติดตามการทำงานในแต่ละขั้นตอนว่าเป็นไปตามปกติหรือไม่ สามารถรับข้อมูลได้ 2 แบบ คือ ข้อมูลไร้รูปแบบ (Non-Format Data) และข้อมูลแบบ HDLC Protocol

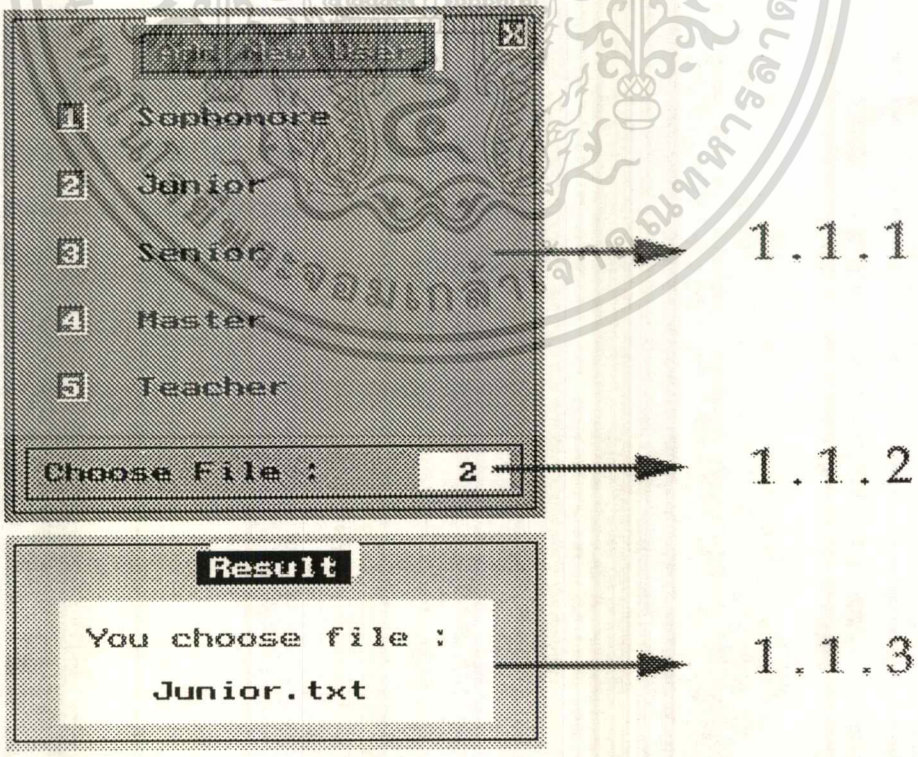
การทำงานในแต่ละส่วนของโปรแกรม ACCESS CONTROL

ในแต่ละเมนูย่อยของโปรแกรม ACCESS CONTROL จะประกอบไปด้วยเมนูที่ย่อยลงไปอีก เพื่อสร้างลำดับการทำงานที่สมบูรณ์ ดังที่จะอธิบายต่อไปนี้

1. **เพิ่มข้อมูลของผู้ใช้คนใหม่ในฐานข้อมูล** : ADD NEW USER

ในเมนูนี้ประกอบด้วยลำดับการเข้าถึงฐานข้อมูล 3 ลำดับ คือ

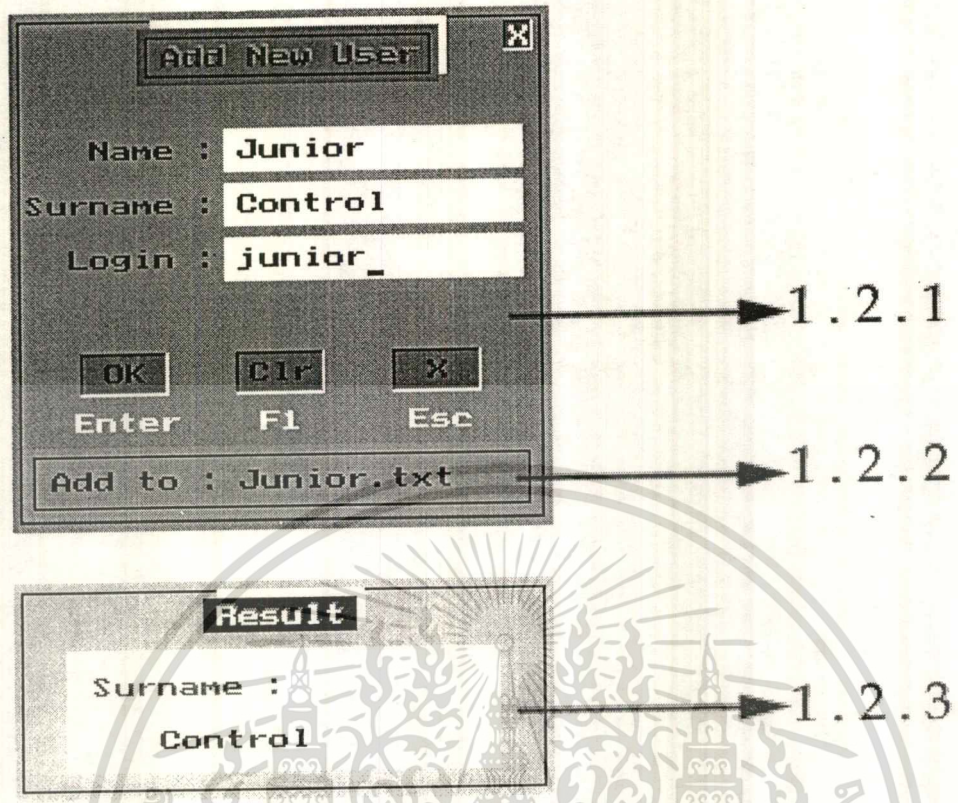
ลำดับที่ 1 เลือกฐานข้อมูล เมื่อผู้ใช้กดคีย์บอร์ดหรือคลิกส์ที่ปุ่มตัวเลขจะปรากฏผลดัง ต่อไปนี้



รูปที่ 3.28 แสดงหน้าจอของเมนู Add New User

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ลำดับที่ 2 ใส่ข้อมูลของผู้ใช้



รูปที่ 3.29 แสดงหน้าจอของการใส่ข้อมูลของผู้ใช้คนใหม่

1.2.1 เมนูข้อมูลของผู้ใช้

เป็นรายละเอียดที่จำเป็นต่อการสร้างฐานข้อมูล คือ ต้องประกอบด้วย ชื่อ นามสกุล และ Login Name ของผู้ใช้ และมีปุ่มจัดการในเมนูนี้ 3 ปุ่ม คือ OK สำหรับตกลงการใส่ข้อมูล ถ้ามีช่องใดช่องหนึ่งขาดหายไปจะต้องใสใหม่ทั้งหมด , ปุ่ม CLR ใช้สำหรับเขียนข้อมูลใหม่ทั้งหมดเมื่อเกิดการผิดพลาดมากๆ และปุ่ม ESC สำหรับยกเลิกการใส่ข้อมูล

1.2.2 ส่วนแสดงชื่อฐานข้อมูลที่จะถูกบรรจุ

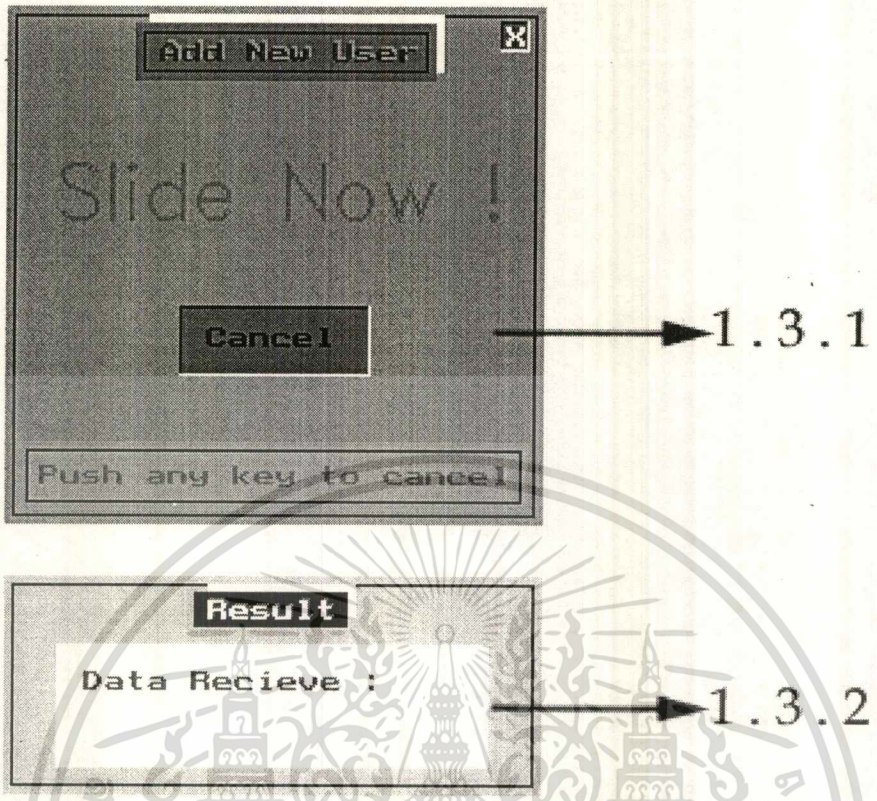
ต่อเนื่องมาจากลำดับที่ 1

1.2.3 ส่วนแสดงรายละเอียดของผู้ใช้ที่ถูกบรรจุไปแล้ว

สำหรับการตรวจเช็ค ชื่อ นามสกุล หรือ Login Name ที่บันทึกลงไป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ลำดับที่ 3 รูดบัตรเพื่อบันทึกข้อมูล



รูปที่ 3.30 แสดงหน้าจอเตรียมพร้อมเพื่อการรูดบัตรและบันทึกข้อมูล

1.3.1 เมนูรอรับการรูดบัตรหรือยกเลิก

เมนูนี้เป็นสัญญาณบอกผู้ใช้ว่าพร้อมจะรับข้อมูลที่จะได้จากการรูดบัตรที่เซิร์ฟเวอร์ หรือเมื่อต้องการยกเลิกการเพิ่มข้อมูลก็สามารถทำได้โดยการกดคีย์ใดคีย์หนึ่งบนคีย์บอร์ด

1.3.2 ส่วนแสดงผลข้อมูล

เป็นส่วนที่ใช้แสดงผลข้อมูลที่รับเข้ามาทางพอร์ตอนุกรม (40 ข้อมูลต่อ 1 บัตร) เมื่อรับครบ 40 ข้อมูลจะพบข้อความว่า

[Login Name] has been registered.

หรือถ้ามีการยกเลิกการบันทึกข้อมูลระหว่างการบันทึกข้อมูลจะปรากฏข้อความว่า

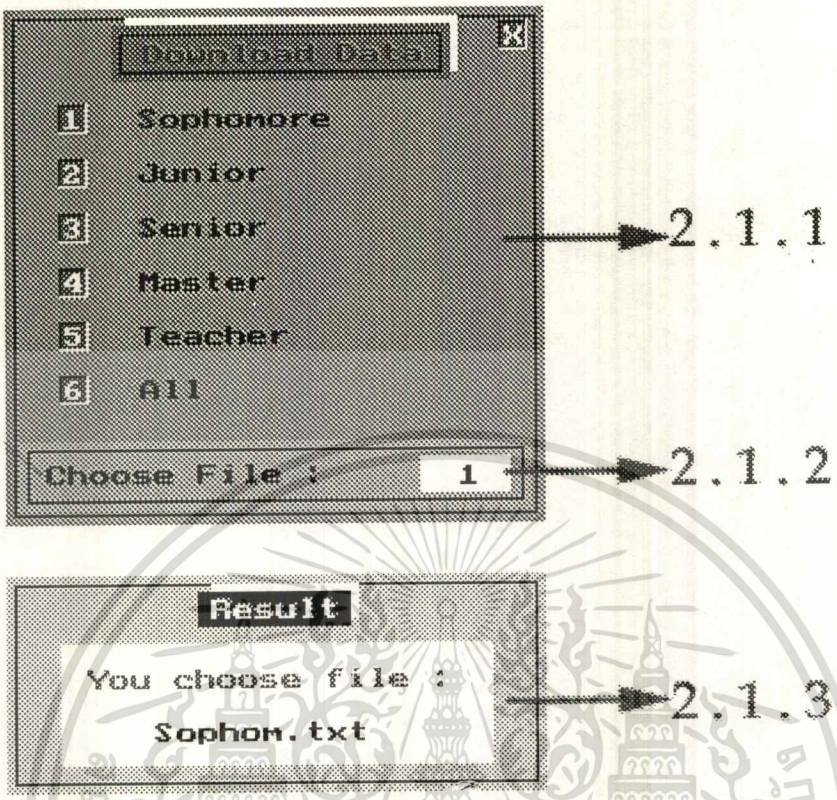
Data Error. Please try again.

2. **สร้างฐานข้อมูลบนเซิร์ฟเวอร์** : DOWNLOAD DATA

ในเมนูนี้ประกอบด้วยการเข้าถึงข้อมูล 3 ลำดับ คือ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ลำดับที่ 1 เลือกข้อมูลที่ต้องการสร้างเป็นฐานข้อมูลบนเซิร์ฟเวอร์



รูปที่ 3.31 แสดงหน้าจอของเมนู DOWNLOAD DATA

2.1.1 แสดงเมนูตัวเลือกฐานข้อมูลทั้ง 5 แบบ รวมทั้งการสร้างฐานข้อมูลทั้งหมด (ข้อมูลทั้ง 5 ฐานจะถูกดาวน์โหลดลงบนเซิร์ฟเวอร์)

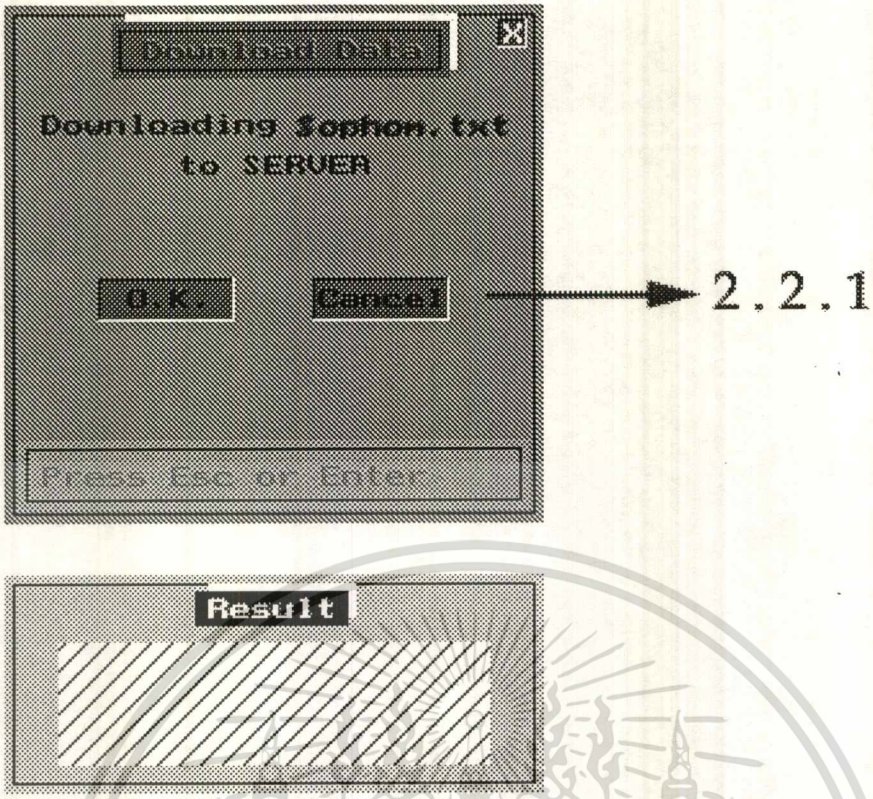
2.1.2 แสดงเลขนำหน้าชื่อฐานข้อมูลที่ถูกเลือก

2.1.3 แสดงชื่อไฟล์ที่จะถูกดาวน์โหลดลงบนเซิร์ฟเวอร์

ลำดับที่ 2 ยืนยันการดาวน์โหลดข้อมูล

2.2.1 เนื่องจากการดาวน์โหลดข้อมูลเสียเวลานาน ถ้ามีการผิดพลาดจะไม่สามารถเปลี่ยนแปลงได้จนกว่าจะดาวน์โหลดเสร็จ ปุ่ม OK สำหรับการตกลง และปุ่ม ESC สำหรับยกเลิกการส่งข้อมูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.32 แสดงหน้าจอการดาวน์โหลดข้อมูลจากไฟล์

ลำดับที่ 3 แสดงผลและติดตามการส่งข้อมูล

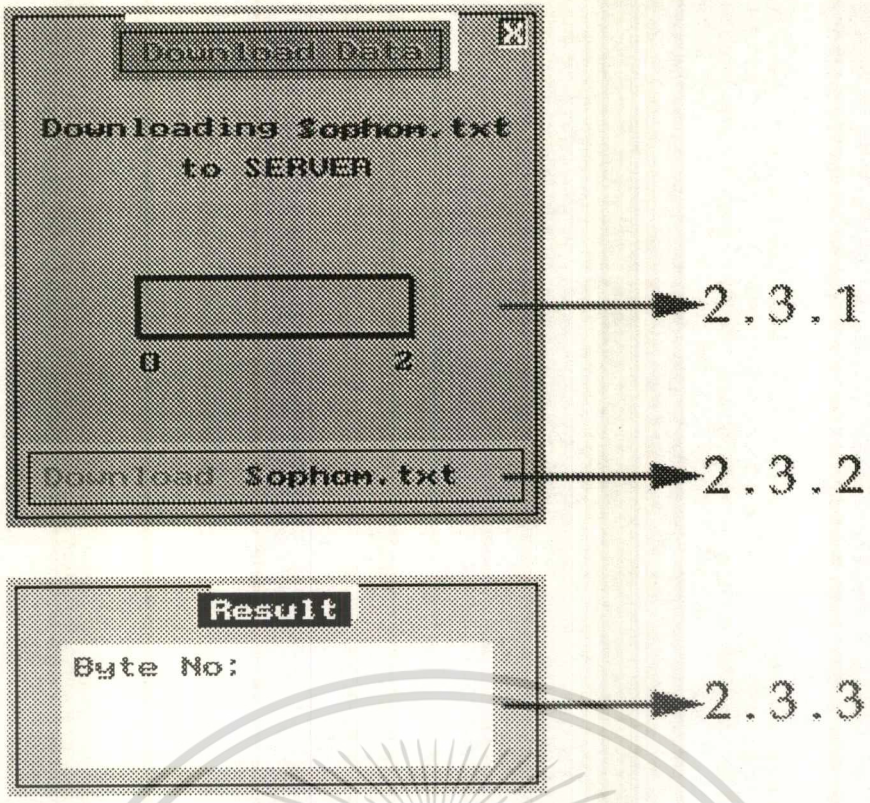
ในเมนูนี้จะมีบาร์แสดงการทำงานของการดาวน์โหลดข้อมูล ซึ่งจำลองมาจากการส่งจริง พร้อมทั้งข้อมูลแต่ละไบท์ที่ดาวน์โหลดลงไปด้วย (ดูรูปที่ 3.33)

2.3.1 แสดง Progress Bar ตัวเลขที่มุมขวาด้านล่างของบาร์ หมายถึง จำนวนบัตรที่มีอยู่ในฐานข้อมูล เมื่อบาร์เป็นสีเขียวเต็มช่อง เป็นสัญญาณบอกว่าข้อมูลในฐานนั้นได้ถูกส่งเสร็จแล้ว

2.3.2 แสดงชื่อไฟล์ที่ถูกดาวน์โหลด เป็นผลต่อเนื่องจากขั้นตอนที่ 1 และ 2

2.3.3 ส่วนแสดงข้อมูลที่ถูกดาวน์โหลด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



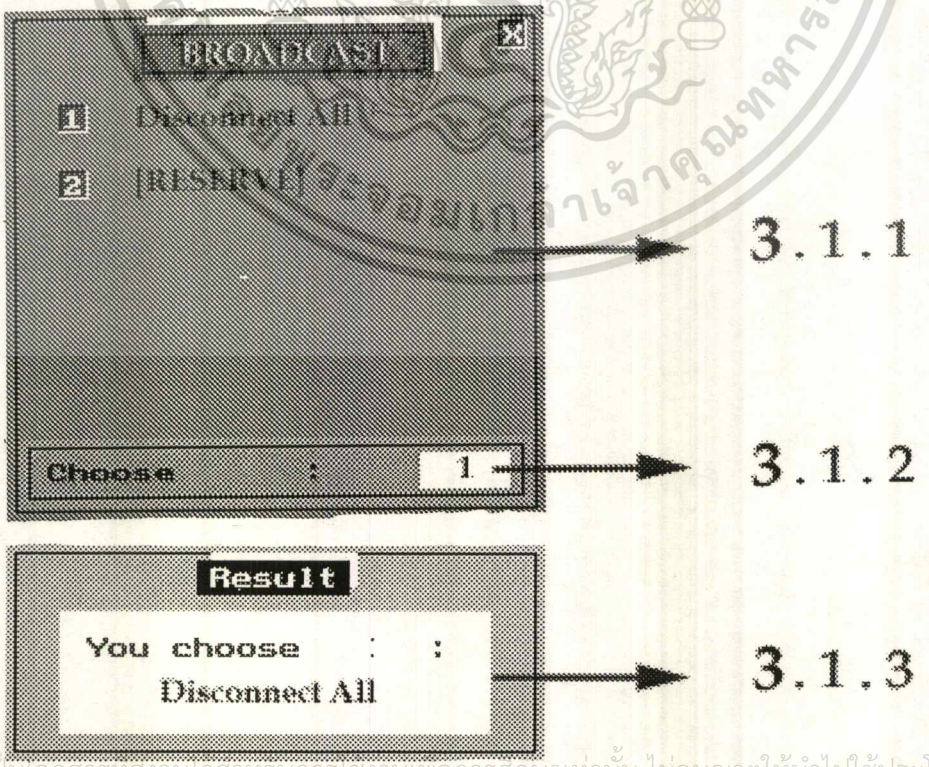
รูปที่ 3.33 แสดงภาพการแสดงผลและติดตามการส่งข้อมูล

3. **กระจายข่าว** : BROADCAST

การกระจายข่าว เป็นการส่งข้อมูลแบบเข้าถึงทุกๆ เทอร์มินอล เป็นการสั่งให้เทอร์มินอลทำงานได้ตามต้องการพร้อมๆ กัน ประกอบด้วยการทำงาน 2 ลำดับดังต่อไปนี้

ลำดับที่ 1 เลือกคำสั่ง เมนูตัวเลือกสามารถเพิ่มได้เองโดยผู้ควบคุมโครงข่าย

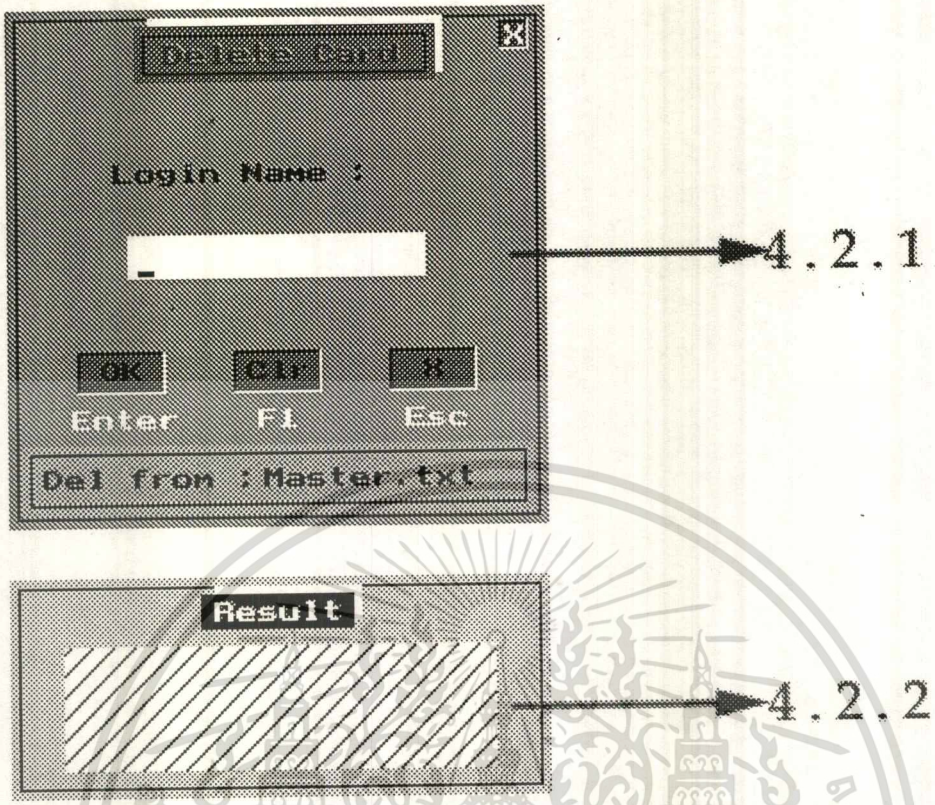
ลำดับที่ 2 ปฏิบัติคำสั่ง



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 3.34 แสดงหน้าจอของเมนู Broadcast

4. **ลบข้อมูลผู้ใช้ออกจากฐานข้อมูล** : DELETE CARD



รูปที่ 3.35 แสดงหน้าจอของเมนูการลบข้อมูลผู้ใช้ออกจากฐานข้อมูล

ในเมนูนี้ประกอบไปด้วยลำดับการทำงาน 2 ขั้นตอน คือ

ลำดับที่ 1 เลือกฐานข้อมูลจากเมนูตัวเลือก : หน้าต่างนี้เหมือนกับหน้าต่างแบบที่ 1.1.1 มีประโยชน์คือ ช่วยให้สามารถลดเวลาในการค้นหาได้

ลำดับที่ 2 ใส่ข้อมูลเป็น Login Name ที่ต้องการค้นหา แล้วทำการลบ

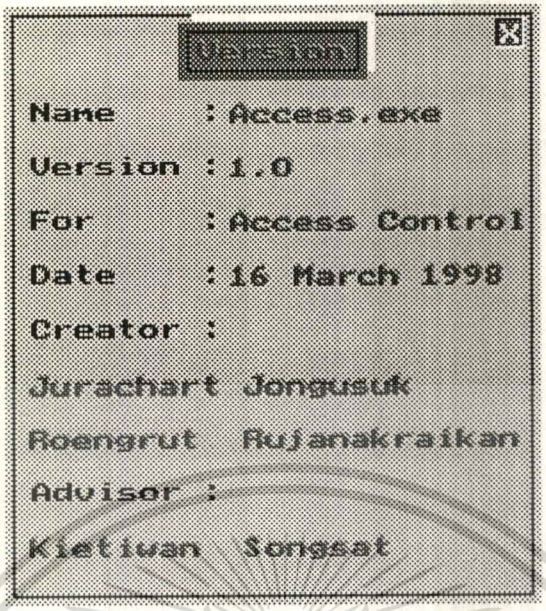
4.2.1 ช่องสำหรับใส่ Login Name ที่ต้องการค้นหา : เมื่อใส่ข้อมูลเสร็จจะต้องเลือกปุ่ม ใดปุ่มหนึ่งดังต่อไปนี้ คือ ปุ่ม OK สำหรับค้นหา , ปุ่ม CLR สำหรับค้นหาข้อมูลใหม่ และปุ่ม ESC สำหรับยกเลิกการค้นหา

4.2.2 แสดงผลการค้นหาข้อมูล มี 2 กรณี คือ FOUND และ NOT FOUND เมื่อตอบตกลงการลบข้อมูล จะปรากฏข้อความว่า

"[Login Name] has been deleted."

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5. **รายละเอียดของโปรแกรม** : VERSION

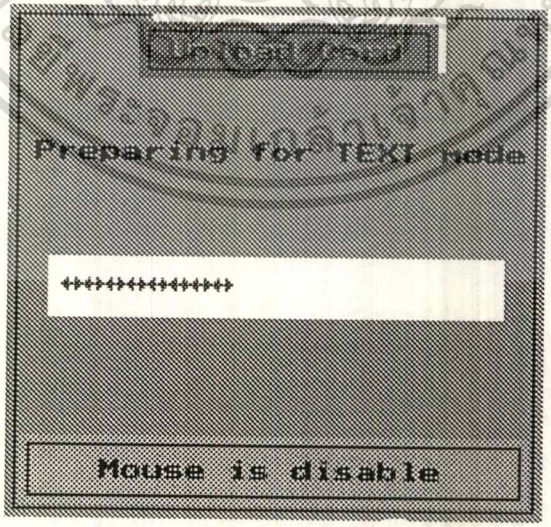


รูปที่ 3.36 แสดงรายละเอียดของโปรแกรม (Version)

6. **อ่านข้อมูลที่รับเข้ามาทางพอร์ตอนุกรม** : UPLOAD BYTE

ประกอบไปด้วยการทำงาน 2 ลำดับดังต่อไปนี้

ลำดับที่ 1 เป็นหน้าจอหน่วงเวลาเพื่อให้ผู้ใช้สามารถยกเลิกการอัพโหลดข้อมูลถ้าลูกศรสีเขียวยังไม่เต็มช่อง เมื่อผู้ใช้กดปุ่มใดๆ ก็ตามบนคีย์บอร์ด จะหมายถึง การยกเลิกการอัพโหลดข้อมูล และเข้าสู่โปรแกรมหลัก (Main Program) ทันที



รูปที่ 3.37 แสดงหน้าจอการหน่วงเวลา

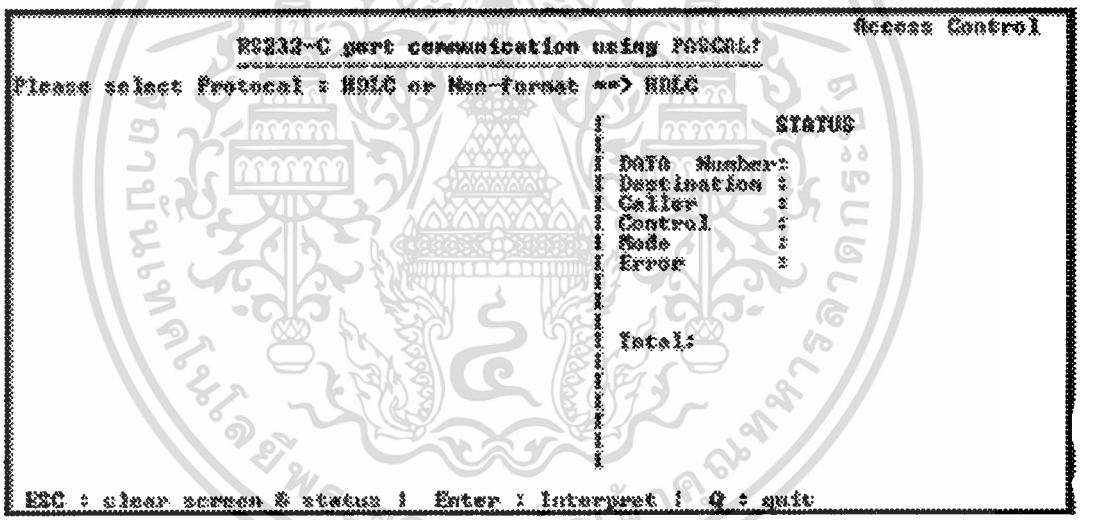
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ลำดับที่ 2 เข้าสู่ Text Mode และเลือกชนิดโปรโตคอลที่จะรับเข้ามาทางพอร์ตอนุกรม โดยโปรแกรมจะรอการกดคีย์ และปรากฏข้อความว่า

"Please select Protocol : HDLC or Non-format"

- ถ้าเลือก N จะได้หน้าจอว่าง เพื่อรอข้อมูลและนำมาแสดงผลบนหน้าจอ ลักษณะนี้ข้อมูลที่รับจะเป็นแบบไร้รูปแบบ
- ถ้าเลือก H จะได้หน้าจอดังรูปที่ 3.38

เมนูนี้มีประโยชน์ในการแปลความหมาย HDLC โปรโตคอล เช่น สามารถระบุได้ว่าข้อมูล 1 ไบท์ที่รับได้นั้น เป็นคำสั่งหรือข้อมูล และส่งมาจากไหน แล้ต้องการติดต่อกับใคร และถ้ามีข้อผิดพลาดจะสามารถตรวจสอบได้ในระดับหนึ่ง สำหรับ Hot Keys ได้แสดงไว้ด้านล่างของจอ (ไม่สามารถใช้เมาส์ได้)



รูปที่ 3.38 แสดงหน้าจอของเมนู Upload Byte

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 4
บทสรุปและวิจารณ์

บทสรุป

ระบบควบคุมการเข้าออกโดยใช้บัตรแม่เหล็ก (Access Control System Using Magnetic Card) ที่ได้ออกแบบ และพัฒนาขึ้นมานี้มีการทำงานของหน่วยประมวลผลของเทอร์มินอลโมดูล และเซิร์ฟเวอร์โมดูล ภายใต้ระบบการสื่อสารข้อมูลแบบโครงข่าย RS-485 ได้ผลของการทำงานและมีประสิทธิภาพของระบบเป็นไปตามวัตถุประสงค์และจุดมุ่งหมายที่ได้วางไว้สำหรับในการทำวิทยานิพนธ์นี้ และได้ออกแบบและพัฒนาโปรโตคอล และโปรแกรมให้สามารถทำงานในการติดต่อสื่อสารข้อมูลอนุกรมในระบบโครงข่าย RS-485 ได้ รวมทั้งมีความยืดหยุ่นต่อโครงสร้างของระบบ (1 server x 32 terminal) คือสามารถมีจำนวนเทอร์มินอลได้สูงสุด 32 เทอร์มินอล (สูงสุด) หรือน้อยกว่าได้ และทำให้สะดวกในการจัดการระบบและการตรวจสอบข้อมูลของผู้ถือบัตรแม่เหล็ก (User) ที่ใช้ในการผ่านเข้าออกระบบ จากการมีการติดต่อสื่อสารอนุกรมมาตรฐาน RS-232-C นั้นทำให้สะดวกในการจัดการและการตรวจสอบฐานข้อมูลของระบบอีกด้วย

บทวิจารณ์

การพิจารณาและวิจารณ์ระบบควบคุมการเข้าออกโดยใช้บัตรแม่เหล็ก ที่ได้ทำการออกแบบสร้างและพัฒนาขึ้นมานี้ จะพิจารณาเป็นหัวข้อดังต่อไปนี้

1. หน่วยประมวลผลของเทอร์มินอลโมดูล

หน่วยประมวลผลกลางใช้ชิพเบอร์ 89C2051 ในตระกูลไมโครคอนโทรลเลอร์ MCS-51 ซึ่งมีขนาดเล็ก (20 ขา) เหมาะกับขนาดของเทอร์มินอลโมดูลที่มีขนาดเล็ก และมีหน้าที่การทำงานไม่มากนักนั่นคือ ทำหน้าที่คอยติดต่อกับเซิร์ฟเวอร์โมดูล อ่านข้อมูลจากบัตรแม่เหล็กของผู้ถือบัตร (User) และส่งข้อมูลที่อ่านได้จากบัตรแม่เหล็กไปยังเซิร์ฟเวอร์ รวมทั้งมีการนำชิพ MAX485 หรือ 75176 เป็นตัวจัดการรับส่งข้อมูลในมาตรฐาน RS-485 ซึ่งตัวถังมีขนาด 8 ขา มีขนาดเล็กมาใช้งาน และมีวงจรทั้งหมดบนเทอร์มินอลโมดูลไม่ซับซ้อนมากนัก ทำให้ขนาดไม่ใหญ่มาก อีกทั้งการออกแบบให้ทำงานเป็นอิสระต่อกัน สามารถติดต่อรับส่งข้อมูลโดยผ่านมาตรฐาน RS-485 ได้ ซึ่งไม่ต้องเชื่อมต่อสายสัญญาณกราวน์ (GROUND) ร่วมกันทั้งระบบ เพื่อป้องกันการลัดวงจรของทั้งระบบได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2. หน่วยประมวลผลของเซิร์ฟเวอร์โมดูล

หน่วยประมวลผลกลางใช้ชิพเบอร์ 89C51 หรือ 89C52 ในตระกูลไมโครคอนโทรลเลอร์ MCS-51 มีขนาดใหญ่ขึ้นมา คือขนาด 40 ขา ซึ่งจะต้องมีหน้าที่การทำงานที่ซับซ้อนมากกว่าเทอร์มินอลโมดูลมาก รวมถึงต้องมีการใช้งานหน่วยความจำภายนอกเพิ่มขึ้นอีก 16 KBYTE โดยใช้ชิพเบอร์ 6264 (8K x 8) จำนวน 2 ตัว เพื่อใช้ในการเก็บฐานข้อมูลต่างๆ ที่จำเป็นในการค้นหาและตรวจสอบข้อมูล

ในการติดต่อสื่อสารข้อมูลอนุกรมใช้มาตรฐาน 2 แบบคือ

- มาตรฐาน RS-485 ใช้ในการติดต่อสื่อสาร รับส่งข้อมูลและคำสั่งกับเทอร์มินอลต่างๆ ผ่านสายส่ง 2 เส้น โดยมีชิพ MAX485 หรือ 75176 เป็นตัวจัดการรับส่งข้อมูลดังกล่าว โดยไม่ต้องเชื่อมต่อสายสัญญาณกราวน์ (GROUND) ร่วมกันทั้งระบบ

- มาตรฐาน RS-232-C ใช้ในการติดต่อสื่อสารรับส่งข้อมูลกับเครื่องไมโครคอมพิวเตอร์ สำหรับจัดการฐานข้อมูลทั้งหมด

3. **ระบบโครงข่าย** RS-485

จากการออกแบบและเลือกใช้ MAX485 หรือ 75176 เป็นตัวจัดการรับส่งข้อมูลอนุกรมในมาตรฐาน RS-485 ซึ่งรับส่งข้อมูลแบบ Half-Duplex มาใช้งาน ทำให้ระบบใช้สายสัญญาณเพียงสองเส้นในการรับส่งข้อมูล และการรับส่งข้อมูลได้ผลถูกต้องสูง เนื่องจากเป็นการรับส่งข้อมูลแบบสมดุลย์

4. **ความคล่องตัวและความยืดหยุ่นของระบบ**

ระบบที่ออกแบบเซิร์ฟเวอร์สามารถติดต่อกับเทอร์มินอลในการติดต่อสื่อสารมาตรฐาน RS-485 ได้จำนวนมากสุดถึง 32 โมดูล ซึ่งถือว่ามีความคล่องตัวและความยืดหยุ่นมากสำหรับระบบควบคุมการเข้าออกขนาดเล็ก โดยพัฒนาโปรแกรมที่ใช้กับเทอร์มินอลและเซิร์ฟเวอร์อย่างเหมาะสม และได้พัฒนาการติดต่อสื่อสารกับไมโครคอมพิวเตอร์ เพื่อจัดการเกี่ยวกับฐานข้อมูลในการจัดการระบบ โดยผ่านมาตรฐาน RS-232-C และพัฒนาโปรแกรมที่ใช้ในการจัดการและติดต่อที่เหมาะสม ทำให้สะดวกในการจัดการระบบรวมมากขึ้นและง่ายขึ้นด้วย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 5
แนวทางการพัฒนา

ระบบควบคุมการเข้าออกโดยใช้บัตรแม่เหล็ก (Access Control System Using Magnetic Card) สามารถจะนำมาประยุกต์ใช้กับกการตรวจสอบการเข้าออกในอาคาร สถานที่ทำงาน ฯลฯ โดย วิทยานิพนธ์ฉบับนี้ เป็นเพียงการออกแบบและพัฒนาในเบื้องต้นเท่านั้น ซึ่งทางภาควิชาจะนำเอาผลงานนี้ไปใช้จริงกับการตรวจสอบการเข้าออกห้องคอมพิวเตอร์ของภาควิชา ดังนั้นระบบควบคุมที่ได้ออกแบบในเบื้องต้นนี้จึงจำเป็นต้องมีการพัฒนาต่อไปอีก ซึ่งมีข้อเสนอแนะและแนวทางในการพัฒนาระบบดังนี้คือ

- ระบบในแต่ละโมดูล ควรจะมีส่วนของระบบฐานเวลาจริง (Real Time Clock) ที่สามารถระบุเวลาเป็น วันที่ เดือน ปี ชั่วโมง นาที วินาที ได้ ซึ่งจะเป็นฟังก์ชันที่จำเป็นและมีประโยชน์อย่างมากในการระบุ ตรวจสอบ บันทึกข้อมูลการเข้าออกระบบของการรูดบัตรแม่เหล็กต่างๆ ที่ใช้รูดในแต่ละช่วงเวลาได้อย่างถูกต้อง และจะเป็นข้อมูลที่เป็นประโยชน์ต่อไปในภายหน้า

- มีการตั้งเวลาในการอนุญาตให้ผู้ใช้ (User) สามารถใช้เครื่องคอมพิวเตอร์ได้ในช่วงเวลาที่ผู้จัดการระบบ (Administrator) เป็นคนจำกัดหรือไม่จำกัดเวลาในการใช้ (Time Out)

- มีการตั้งช่วงเวลาผ่านได้ในแต่ละวัน (Time Zone) รวมทั้งมีการตรวจสอบและมีการอนุญาตหรือไม่อนุญาตให้ผู้ใช้ (User) สามารถใช้เครื่องคอมพิวเตอร์ได้ในวันหยุดสุดสัปดาห์ (Holiday) หรือวันหยุดพิเศษของแต่ละเดือน (Special Day)

- ระบบสามารถทำการตรวจสอบผู้ใช้ (User) ว่าเมื่อผู้ใช้รายไหนทำการรูดบัตรแล้วจะอนุญาตให้เข้าออกระบบได้ (Enable) หรือไม่ได้ (Disable) ในช่วงวัน เวลาใดบ้าง ซึ่งส่วนนี้จะเป็นหน้าที่ของผู้จัดการระบบ

- เซิร์ฟเวอร์โมดูลสามารถ Upload ข้อมูลของการเข้าออกทั้งหมดในแต่ละช่วงเวลา หรือในแต่ละวันให้แก่เครื่องไมโครคอมพิวเตอร์ที่เชื่อมต่อกับเซิร์ฟเวอร์โมดูลของระบบ และช่วยให้ผู้จัดการระบบทำหน้าที่ในการตรวจสอบระบบได้ รวมทั้งควรจะทำการบันทึกและแสดงผลข้อมูลการเข้าออกทั้งหมดในแต่ละช่วงเวลาได้เพื่อประโยชน์ในการตรวจสอบต่อไป

- ในกรณีที่ระบบขาดไฟเลี้ยง หรือไฟตก ซึ่งจะเป็นสาเหตุให้ข้อมูลในหน่วยความจำบนเซิร์ฟเวอร์โมดูลหายไป และมีผลเสียหายต่อการตรวจสอบและบันทึกข้อมูลของระบบเป็นอย่างมาก จึงควรมีไฟสำรองให้แก่เซิร์ฟเวอร์โมดูลเพื่อป้องกันเหตุดังกล่าว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เอกสารอ้างอิง

[1] กิตติ ภักดีวัฒนะกุล, "สร้างและพัฒนาระบบ LAN", พิมพ์ครั้งที่ 2, หจก.ไทยเจริญการพิมพ์, กรุงเทพฯ , 2540.

[2] นุกูล กระจาย, "กราฟิกส์และเกมคอมพิวเตอร์ด้วยเทอร์โบปาสคาล", หจก. เอซ-เอน การพิมพ์, กรุงเทพฯ, 2536.

[3] สุเจตน์ จันทรังษี, "ไมโครคอนโทรลเลอร์ซีพเดี่ยว 8051", โครงการตำราวิชาการวิทยาลัยมหานคร, กรุงเทพฯ, 2535.

[4] "โพรโทคอล X.25/HDLC", ปริญญานิพนธ์ภาควิชาเทคโนโลยีอิเล็กทรอนิกส์, สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง ,กรุงเทพฯ, 2536

[5] "PRO-100 VERSION 1.0 89CXX PROGRAMMER BOARD", บริษัทศิลารีเสิร์ชจำกัด, กรุงเทพฯ.

[6] "DATA BOOK2", บริษัทศิลารีเสิร์ชจำกัด, กรุงเทพฯ.

[7] "MICROPROCESSOR DATA BOOK MCS-51 MICROCONTROLLERS", ETT CO., LTD.

[8] William A. Shay, "UNDERSTANDING DATA COMMUNICATIONS AND NETWORKS", PWS Publishing Company, 1995.

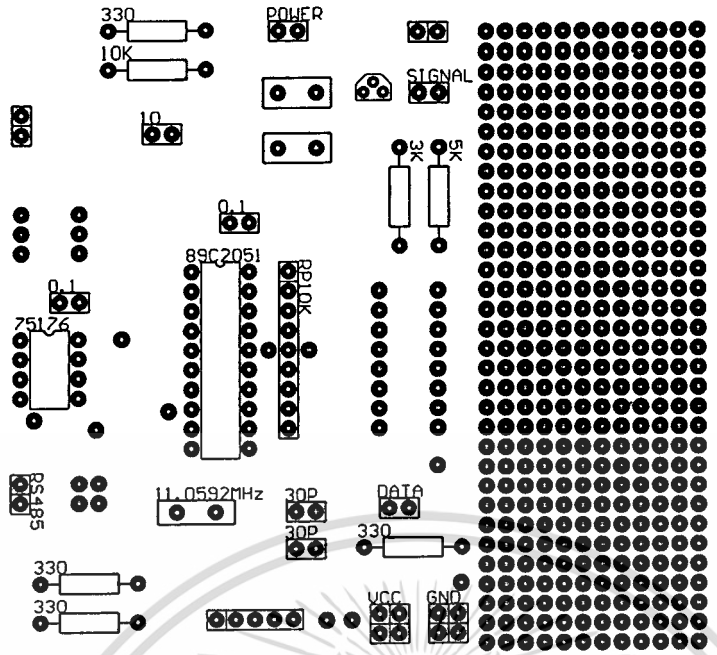
[9] Peter Abel, "IBM PC ASSEMBLY LANGUAGE AND PROGRAMMING", Second Edition, Prentice-Hall, 1991.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

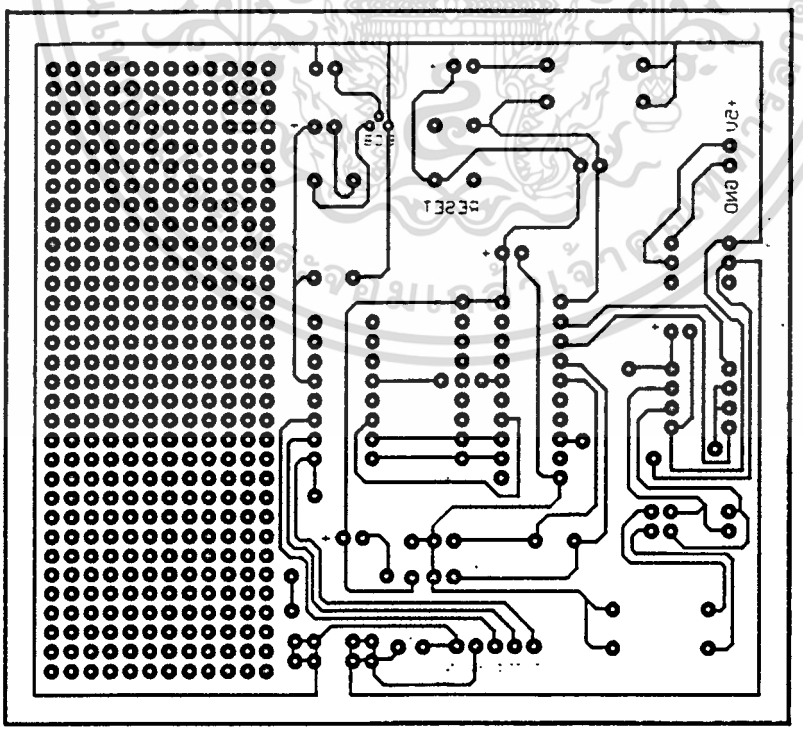
ภาคผนวก ก.

แผ่นพิมพ์ลายวงจร และภาพถ่ายวงจร

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

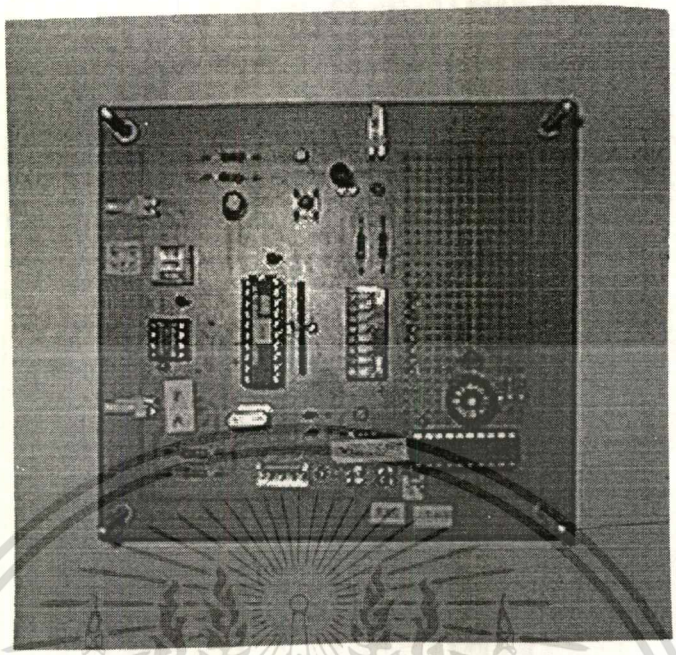


รูปที่ ก.1 แสดงลายวงจรด้านหน้าของเทอร์มินอลโมดูล

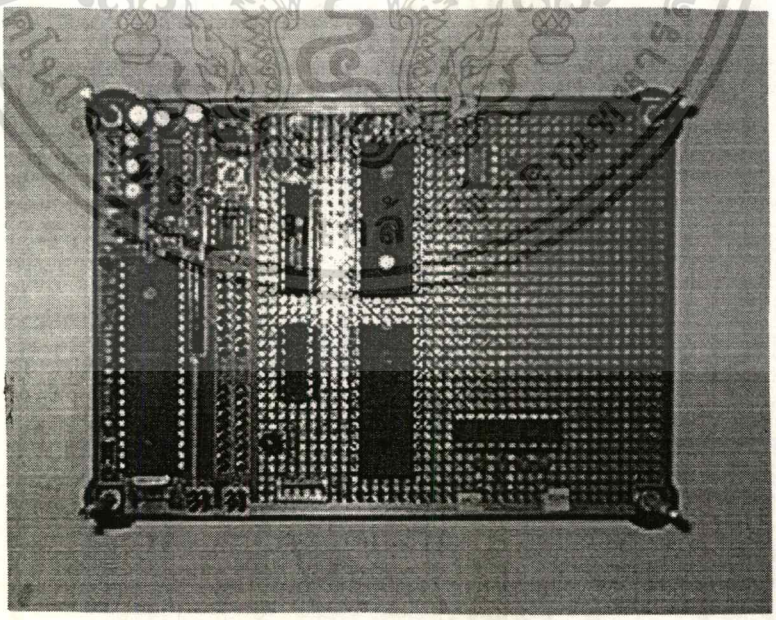


รูปที่ ก.2 แสดงลายวงจรด้านหลังของเทอร์มินอลโมดูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

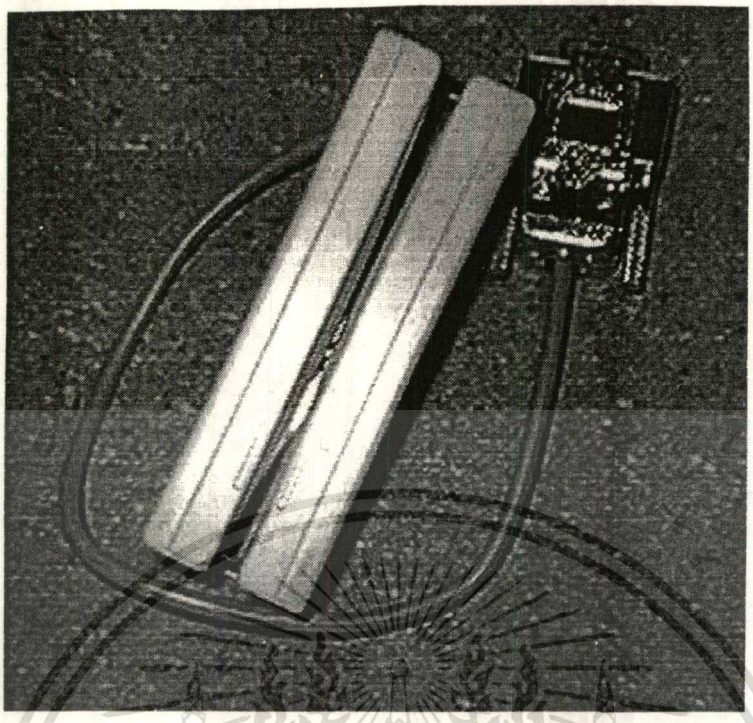


รูปที่ ก.3 แสดงรูปถ่ายของเทอร์มินอลโมดูล

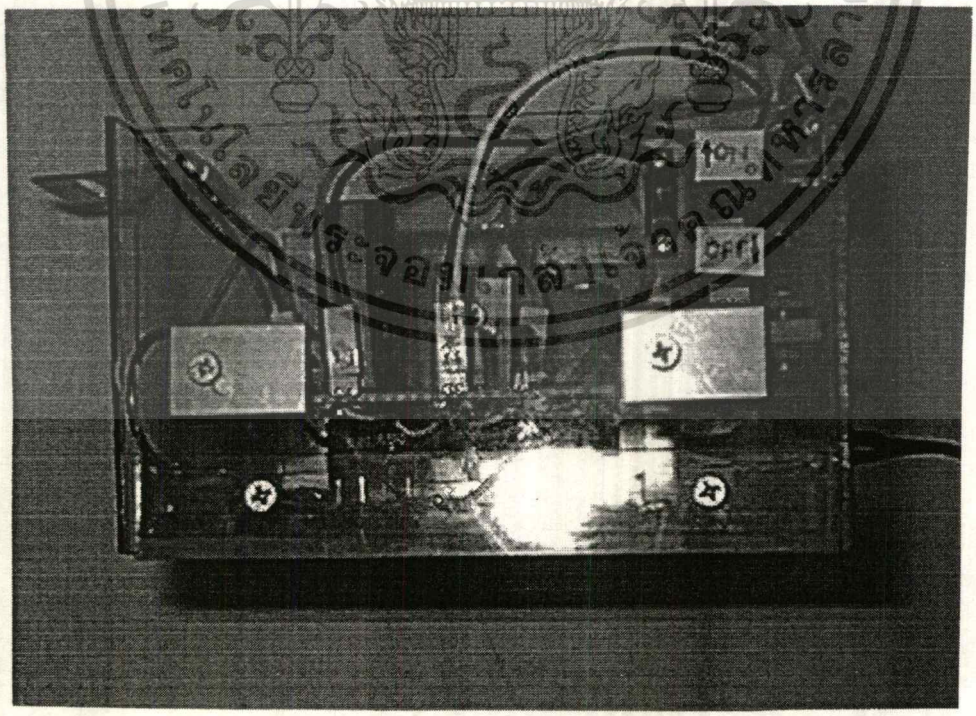


รูปที่ ก.4 แสดงรูปถ่ายของเซิร์ฟเวอร์โมดูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ ก.5 แสดงรูปถ่ายของตัวอ่านบัตรแม่หล็ก



รูปที่ ก.6 แสดงรูปถ่ายของชุดวงจรไฟเลี้ยง (Power Supply) และรีเลย์ (Relay)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ข.

ข้อมูลทางเทคนิคของอุปกรณ์ที่สำคัญบนบอร์ด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รายละเอียดของ MAX485

19-0122; Rev 5; 2/96

MAXIM

Low-Power, Slew-Rate-Limited RS-485/RS-422 Transceivers

General Description

The MAX481, MAX483, MAX485, MAX487–MAX491, and MAX1487 are low-power transceivers for RS-485 and RS-422 communication. Each part contains one driver and one receiver. The MAX483, MAX487, MAX488, and MAX489 feature reduced slew-rate drivers that minimize EMI and reduce reflections caused by improperly terminated cables, thus allowing error-free data transmission up to 250kbps. The driver slew rates of the MAX481, MAX485, MAX490, MAX491, and MAX1487 are not limited, allowing them to transmit up to 2.5Mbps.

These transceivers draw between 120µA and 500µA of supply current when unloaded or fully loaded with disabled drivers. Additionally, the MAX481, MAX483, and MAX487 have a low-current shutdown mode in which they consume only 0.1µA. All parts operate from a single 5V supply.

Drivers are short-circuit current limited and are protected against excessive power dissipation by thermal shutdown circuitry that places the driver outputs into a high-impedance state. The receiver input has a fail-safe feature that guarantees a logic-high output if the input is open circuit.

The MAX487 and MAX1487 feature quarter-unit-load receiver input impedance, allowing up to 128 MAX487/MAX1487 transceivers on the bus. Full-duplex communications are obtained using the MAX488–MAX491, while the MAX481, MAX483, MAX485, MAX487, and MAX1487 are designed for half-duplex applications.

Applications

Low-Power RS-485 Transceivers
Low-Power RS-422 Transceivers
Level Translators
Transceivers for EMI-Sensitive Applications
Industrial-Control Local Area Networks

Features

- ♦ In µMAX Package: Smallest 8-Pin SO
- ♦ Slew-Rate Limited for Error-Free Data Transmission (MAX483/487/488/489)
- ♦ 0.1µA Low-Current Shutdown Mode (MAX481/483/487)
- ♦ Low Quiescent Current:
  120µA (MAX483/487/488/489)
  230µA (MAX1487)
  300µA (MAX481/485/490/491)
- ♦ -7V to +12V Common-Mode Input Voltage Range
- ♦ Three-State Outputs
- ♦ 30ns Propagation Delays, 5ns Skew (MAX481/485/490/491/1487)
- ♦ Full-Duplex and Half-Duplex Versions Available
- ♦ Operate from a Single 5V Supply
- ♦ Allows up to 128 Transceivers on the Bus (MAX487/MAX1487)
- ♦ Current-Limiting and Thermal Shutdown for Driver Overload Protection

Ordering Information

PART	TEMP. RANGE	PIN-PACKAGE
MAX481CPA	0°C to +70°C	8 Plastic DIP
MAX481CSA	0°C to +70°C	8 SO
MAX481CUA	0°C to +70°C	8 µMAX
MAX481C/D	0°C to +70°C	Dice*

Ordering Information continued at end of data sheet.
** Contact factory for dice specifications.*

Selection Table

PART NUMBER	HALF/FULL DUPLEX	DATA RATE (Mbps)	SLEW-RATE LIMITED	LOW-POWER SHUTDOWN	RECEIVER/ DRIVER ENABLE	QUIESCENT CURRENT (µA)	NUMBER OF TRANSMITTERS ON BUS	PIN COUNT
MAX481	Half	2.5	No	Yes	Yes	300	32	8
MAX483	Half	0.25	Yes	Yes	Yes	120	32	8
MAX485	Half	2.5	No	No	Yes	300	32	8
MAX487	Half	0.25	Yes	Yes	Yes	120	128	8
MAX488	Full	0.25	Yes	No	No	120	32	8
MAX489	Full	0.25	Yes	No	Yes	120	32	14
MAX490	Full	2.5	No	No	No	300	32	8
MAX491	Full	2.5	No	No	Yes	300	32	14
MAX1487	Half	2.5	No	No	Yes	230	128	8

For free samples & the latest literature: http://www.maxim-ic.com, or phone 1-800-998-8800

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Low-Power, Slew-Rate-Limited RS-485/RS-422 Transceivers

ABSOLUTE MAXIMUM RATINGS

Supply Voltage (V_{CC}) 12V
Control Input Voltage ($\overline{RE}$, DE) ..... -0.5V to (V_{CC} + 0.5V)
Driver Input Voltage (DI) ..... -0.5V to (V_{CC} + 0.5V)
Driver Output Voltage (A, B) ..... -8V to +12.5V
Receiver Input Voltage (A, B) ..... -8V to +12.5V
Receiver Output Voltage (RO) ..... -0.5V to (V_{CC} +0.5V)
Continuous Power Dissipation (T_A = +70°C)
8-Pin Plastic DIP (derate 9.09mW/°C above +70°C) 727mW
14-Pin Plastic DIP (derate 10.00mW/°C above +70°C) ..800mW
8-Pin SO (derate 5.88mW/°C above +70°C) 471mW
14-Pin SO (derate 8.33mW/°C above +70°C) 667mW
8-Pin µMAX (derate 4.1mW/°C above +70°C) 830mW
8-Pin CERDIP (derate 8.00mW/°C above +70°C) 640mW
14-Pin CERDIP (derate 9.09mW/°C above +70°C) 727mW
Operating Temperature Ranges
MAX4_ _C_ _/MAX1487C_ A ..... 0°C to +70°C
MAX4_ _E_ _/MAX1487E_ A ..... -40°C to +85°C
MAX4_ _MJ_/MAX1487MJA ..... -55°C to +125°C
Storage Temperature Range ..... -65°C to +160°C
Lead Temperature (soldering, 10sec) ..... +300°C

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

DC ELECTRICAL CHARACTERISTICS

(V_{CC} = 5V ±5%, T_A = T_{MIN} to T_{MAX}, unless otherwise noted.) (Notes 1, 2)

PARAMETER	SYMBOL	CONDITIONS		MIN	TYP	MAX	UNITS
Differential Driver Output (no load)	V_{OD1}					5	V
Differential Driver Output (with load)	V_{OD2}	R = 50 Ω (RS-422)		2			V
		R = 27 Ω (RS-485), Figure 4		1.5		5	
Change in Magnitude of Driver Differential Output Voltage for Complementary Output States	ΔV_{OD}	R = 27 Ω or 50 Ω, Figure 4				0.2	V
Driver Common-Mode Output Voltage	V_{OC}	R = 27 Ω or 50 Ω, Figure 4				3	V
Change in Magnitude of Driver Common-Mode Output Voltage for Complementary Output States	ΔV_{OD}	R = 27 Ω or 50 Ω, Figure 4				0.2	V
Input High Voltage	V_{IH}	DE, DI, $\overline{RE}$		2.0			V
Input Low Voltage	V_{IL}	DE, DI, $\overline{RE}$				0.8	V
Input Current	I_{IN1}	DE, DI, $\overline{RE}$				±2	µA
Input Current (A, B)	I_{IN2}	DE = 0V; V_{CC} = 0V or 5.25V, all devices except MAX487/MAX1487	V_{IN} = 12V			1.0	mA
			V_{IN} = -7V			-0.8	
		MAX487/MAX1487, DE = 0V, V_{CC} = 0V or 5.25V	V_{IN} = 12V			0.25	mA
			V_{IN} = -7V			-0.2	
Receiver Differential Threshold Voltage	V_{TH}	-7V ≤ V_{CM} ≤ 12V		-0.2		0.2	V
Receiver Input Hysteresis	ΔV_{TH}	V_{CM} = 0V			70		mV
Receiver Output High Voltage	V_{OH}	I_O = -4mA, V_D = 200mV		3.5			V
Receiver Output Low Voltage	V_{OL}	I_O = 4mA, V_D = -200mV				0.4	V
Three-State (high impedance) Output Current at Receiver	I_{OZR}	0.4V ≤ V_O ≤ 2.4V				±1	µA
Receiver Input Resistance	R_{IN}	-7V ≤ V_{CM} ≤ 12V, all devices except MAX487/MAX1487		12			kΩ
		-7V ≤ V_{CM} ≤ 12V, MAX487/MAX1487		48			kΩ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Low-Power, Slew-Rate-Limited RS-485/RS-422 Transceivers

DC ELECTRICAL CHARACTERISTICS (continued)

(V_{CC} = 5V ±5%, T_A = T_{MIN} to T_{MAX}, unless otherwise noted.) (Notes 1, 2)

PARAMETER	SYMBOL	CONDITIONS			MIN	TYP	MAX	UNITS
No-Load Supply Current (Note 3)	I_{CC}	MAX488/MAX489, DE, DI, $\overline{RE}$ = 0V or V_{CC}				120	250	µA
		MAX490/MAX491, DE, DI, $\overline{RE}$ = 0V or V_{CC}				300	500	
		MAX481/MAX485, $\overline{RE}$ = 0V or V_{CC}	DE = V_{CC}			500	900	
			DE = 0V			300	500	
		MAX1487, $\overline{RE}$ = 0V or V_{CC}	DE = V_{CC}			300	500	
			DE = 0V			230	400	
		MAX483/MAX487, $\overline{RE}$ = 0V or V_{CC}	DE = 5V	MAX483		350	650	
				MAX487		250	400	
			DE = 0V			120	250	
Supply Current in Shutdown	I_{SHDN}	MAX481/483/487, DE = 0V, $\overline{RE}$ = V_{CC}				0.1	10	µA
Driver Short-Circuit Current, V_O = High	I_{OSD1}	-7V ≤ V_O ≤ 12V (Note 4)			35		250	mA
Driver Short-Circuit Current, V_O = Low	I_{OSD2}	-7V ≤ V_O ≤ 12V (Note 4)			35		250	mA
Receiver Short-Circuit Current	I_{OSR}	0V ≤ V_O ≤ V_{CC}			7		95	mA

SWITCHING CHARACTERISTICS—MAX481/MAX485, MAX490/MAX491, MAX1487

(V_{CC} = 5V ±5%, T_A = T_{MIN} to T_{MAX}, unless otherwise noted.) (Notes 1, 2)

PARAMETER	SYMBOL	CONDITIONS		MIN	TYP	MAX	UNITS
Driver Input to Output	t_{PLH}	Figures 6 and 8, R_{DIFF} = 54Ω, C_{L1} = C_{L2} = 100pF		10	30	60	ns
	t_{PHL}			10	30	60	
Driver Output Skew to Output	t_{SKEW}	Figures 6 and 8, R_{DIFF} = 54Ω, C_{L1} = C_{L2} = 100pF			5	10	ns
Driver Rise or Fall Time	t_R, t_F	Figures 6 and 8, R_{DIFF} = 54Ω, C_{L1} = C_{L2} = 100pF	MAX481, MAX485, MAX1487	3	15	40	ns
			MAX490C/E, MAX491C/E	5	15	25	
			MAX490M, MAX491M	3	15	40	
Driver Enable to Output High	t_{ZH}	Figures 7 and 9, C_L = 100pF, S2 closed			40	70	ns
Driver Enable to Output Low	t_{ZL}	Figures 7 and 9, C_L = 100pF, S1 closed			40	70	ns
Driver Disable Time from Low	t_{LZ}	Figures 7 and 9, C_L = 15pF, S1 closed			40	70	ns
Driver Disable Time from High	t_{HZ}	Figures 7 and 9, C_L = 15pF, S2 closed			40	70	ns
Receiver Input to Output	t_{PLH}, t_{PHL}	Figures 6 and 10, R_{DIFF} = 54Ω, C_{L1} = C_{L2} = 100pF	MAX481, MAX485, MAX1487	20	90	200	ns
			MAX490C/E, MAX491C/E	20	90	150	
			MAX490M, MAX491M	20	90	200	
$\lvert t_{PLH} - t_{PHL} \rvert$ Differential Receiver Skew	t_{SKD}	Figures 6 and 10, R_{DIFF} = 54Ω, C_{L1} = C_{L2} = 100pF			13		ns
Receiver Enable to Output Low	t_{ZL}	Figures 5 and 11, C_{RL} = 15pF, S1 closed			20	50	ns
Receiver Enable to Output High	t_{ZH}	Figures 5 and 11, C_{RL} = 15pF, S2 closed			20	50	ns
Receiver Disable Time from Low	t_{LZ}	Figures 5 and 11, C_{RL} = 15pF, S1 closed			20	50	ns
Receiver Disable Time from High	t_{HZ}	Figures 5 and 11, C_{RL} = 15pF, S2 closed			20	50	ns
Maximum Data Rate	f_{MAX}			2.5			Mbps
Time to Shutdown	t_{SHDN}	MAX481 (Note 5)		50	200	600	ns

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Low-Power, Slew-Rate-Limited RS-485/RS-422 Transceivers

SWITCHING CHARACTERISTICS—MAX481/MAX485, MAX490/MAX491, MAX1487 (continued)

(V_{CC} = 5V ±5%, T_A = T_{MIN} to T_{MAX}, unless otherwise noted.) (Notes 1, 2)

PARAMETER	SYMBOL	CONDITIONS	MIN	TYP	MAX	UNITS
Driver Enable from Shutdown to Output High (MAX481)	$t_{ZH(SHDN)}$	Figures 7 and 9, C_L = 100pF, S2 closed		40	100	ns
Driver Enable from Shutdown to Output Low (MAX481)	$t_{ZL(SHDN)}$	Figures 7 and 9, C_L = 100pF, S1 closed		40	100	ns
Receiver Enable from Shutdown to Output High (MAX481)	$t_{ZH(SHDN)}$	Figures 5 and 11, C_L = 15pF, S2 closed, A - B = 2V		300	1000	ns
Receiver Enable from Shutdown to Output Low (MAX481)	$t_{ZL(SHDN)}$	Figures 5 and 11, C_L = 15pF, S1 closed, B - A = 2V		300	1000	ns

SWITCHING CHARACTERISTICS—MAX483, MAX487/MAX488/MAX489

(V_{CC} = 5V ±5%, T_A = T_{MIN} to T_{MAX}, unless otherwise noted.) (Notes 1, 2)

PARAMETER	SYMBOL	CONDITIONS	MIN	TYP	MAX	UNITS
Driver Input to Output	t_{PLH}	Figures 6 and 8, R_{DIFF} = 54Ω, C_{L1} = C_{L2} = 100pF	250	800	2000	ns
	t_{PHL}		250	800	2000	
Driver Output Skew to Output	t_{SKEW}	Figures 6 and 8, R_{DIFF} = 54Ω, C_{L1} = C_{L2} = 100pF		100	800	ns
Driver Rise or Fall Time	t_R, t_F	Figures 6 and 8, R_{DIFF} = 54Ω, C_{L1} = C_{L2} = 100pF	250		2000	ns
Driver Enable to Output High	t_{ZH}	Figures 7 and 9, C_L = 100pF, S2 closed	250		2000	ns
Driver Enable to Output Low	t_{ZL}	Figures 7 and 9, C_L = 100pF, S1 closed	250		2000	ns
Driver Disable Time from Low	t_{LZ}	Figures 7 and 9, C_L = 15pF, S1 closed	300		3000	ns
Driver Disable Time from High	t_{HZ}	Figures 7 and 9, C_L = 15pF, S2 closed	300		3000	ns
Receiver Input to Output	t_{PLH}	Figures 6 and 10, R_{DIFF} = 54Ω, C_{L1} = C_{L2} = 100pF	250		2000	ns
	t_{PHL}		250		2000	
$\lvert t_{PLH} - t_{PHL} \rvert$ Differential Receiver Skew	t_{SKD}	Figures 6 and 10, R_{DIFF} = 54Ω, C_{L1} = C_{L2} = 100pF		100		ns
Receiver Enable to Output Low	t_{ZL}	Figures 5 and 11, C_{RL} = 15pF, S1 closed		20	50	ns
Receiver Enable to Output High	t_{ZH}	Figures 5 and 11, C_{RL} = 15pF, S2 closed		20	50	ns
Receiver Disable Time from Low	t_{LZ}	Figures 5 and 11, C_{RL} = 15pF, S1 closed		20	50	ns
Receiver Disable Time from High	t_{HZ}	Figures 5 and 11, C_{RL} = 15pF, S2 closed		20	50	ns
Maximum Data Rate	f_{MAX}	t_{PLH}, t_{PHL} < 50% of data period	250			kbps
Time to Shutdown	t_{SHDN}	MAX483/MAX487 (Note 5)	50	200	600	ns
Driver Enable from Shutdown to Output High	$t_{ZH(SHDN)}$	MAX483/MAX487, Figures 7 and 9, C_L = 100pF, S2 closed			2000	ns
Driver Enable from Shutdown to Output Low	$t_{ZL(SHDN)}$	MAX483/MAX487, Figures 7 and 9, C_L = 100pF, S1 closed			2000	ns
Receiver Enable from Shutdown to Output High	$t_{ZH(SHDN)}$	MAX483/MAX487, Figures 5 and 11, C_L = 15pF, S2 closed			2500	ns
Receiver Enable from Shutdown to Output Low	$t_{ZL(SHDN)}$	MAX483/MAX487, Figures 5 and 11, C_L = 15pF, S1 closed			2500	ns

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

NOTES FOR ELECTRICAL/SWITCHING CHARACTERISTICS

Note 1: All currents into device pins are positive; all currents out of device pins are negative. All voltages are referenced to device ground unless otherwise specified.

Note 2: All typical specifications are given for V_{CC} = 5V and T_A = +25°C.

Note 3: Supply current specification is valid for loaded transmitters when DE = 0V.

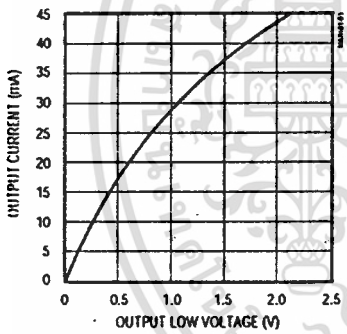
Note 4: Applies to peak current. See *Typical Operating Characteristics*.

Note 5: The MAX481/MAX483/MAX487 are put into shutdown by bringing RE high and DE low. If the inputs are in this state for less than 50ns, the parts are guaranteed not to enter shutdown. If the inputs are in this state for at least 600ns, the parts are guaranteed to have entered shutdown. See *Low-Power Shutdown Mode* section.

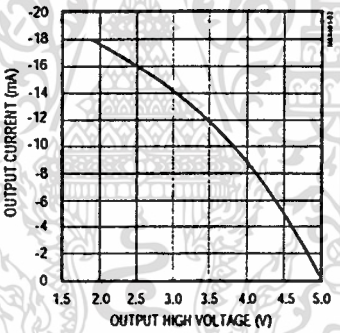
Typical Operating Characteristics

(V_{CC} = 5V, T_A = +25°C, unless otherwise noted.)

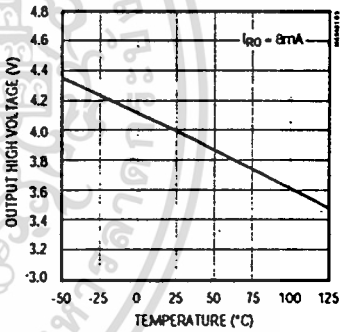
OUTPUT CURRENT vs. RECEIVER OUTPUT LOW VOLTAGE



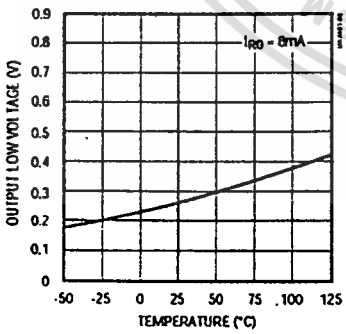
OUTPUT CURRENT vs. RECEIVER OUTPUT HIGH VOLTAGE



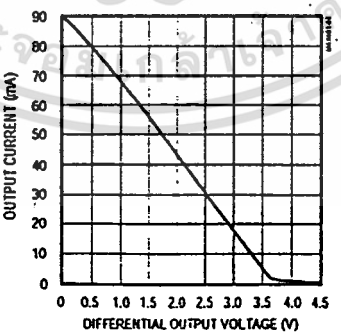
RECEIVER OUTPUT HIGH VOLTAGE vs. TEMPERATURE



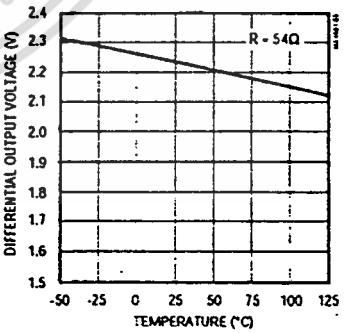
RECEIVER OUTPUT LOW VOLTAGE vs. TEMPERATURE



DRIVER OUTPUT CURRENT vs. DIFFERENTIAL OUTPUT VOLTAGE



DRIVER DIFFERENTIAL OUTPUT VOLTAGE vs. TEMPERATURE



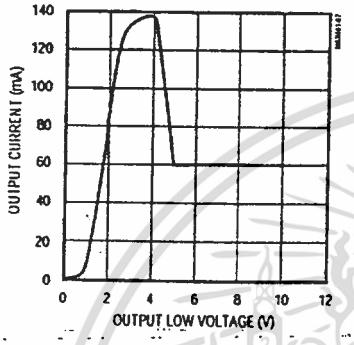
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Low-Power, Slew-Rate-Limited RS-485/RS-422 Transceivers

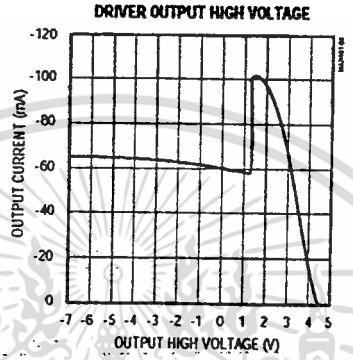
Typical Operating Characteristics (continued)

(VCC = 5V, T_A = +25°C, unless otherwise noted.)

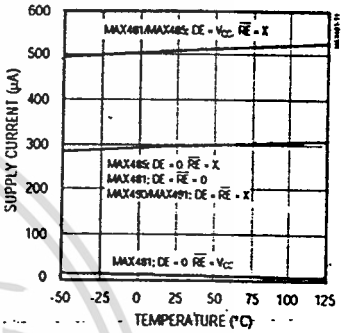
OUTPUT CURRENT vs. DRIVER OUTPUT LOW VOLTAGE



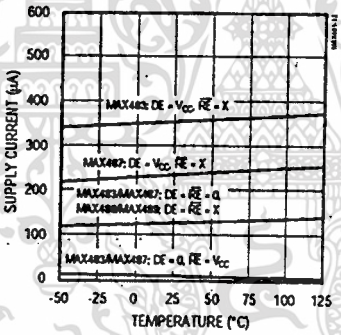
OUTPUT CURRENT vs. DRIVER OUTPUT HIGH VOLTAGE



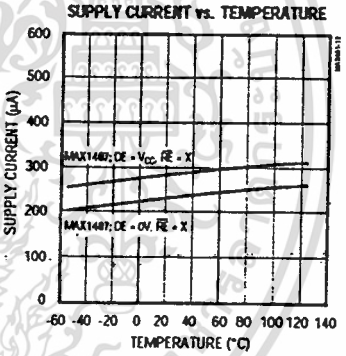
MAX481/MAX485/MAX490/MAX491 SUPPLY CURRENT vs. TEMPERATURE



MAX483/MAX487–MAX489 SUPPLY CURRENT vs. TEMPERATURE



MAX1487 SUPPLY CURRENT vs. TEMPERATURE



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Pin Description

PIN					NAME	FUNCTION
MAX481/MAX483/ MAX485/MAX487/ MAX1487		MAX488/ MAX490		MAX489/ MAX491		
DIP/SO	µMAX	DIP/SO	µMAX	DIP/SO		
1	3	2	4	2	RO	Receiver Output: If A > B by 200mV, RO will be high; If A < B by 200mV, RO will be low.
2	4	—	—	3	$\overline{RE}$	Receiver Output Enable. RO is enabled when $\overline{RE}$ is low; RO is high impedance when $\overline{RE}$ is high.
3	5	—	—	4	DE	Driver Output Enable. The driver outputs, Y and Z, are enabled by bringing DE high. They are high impedance when DE is low. If the driver outputs are enabled, the parts function as line drivers. While they are high impedance, they function as line receivers if $\overline{RE}$ is low.
4	6	3	5	5	DI	Driver Input. A low on DI forces output Y low and output Z high. Similarly, a high on DI forces output Y high and output Z low.
5	7	4	6	6, 7	GND	Ground
—	—	5	7	9	Y	Noninverting Driver Output
—	—	6	8	10	Z	Inverting Driver Output
6	8	—	—	—	A	Noninverting Receiver Input and Noninverting Driver Output
—	—	8	2	12	A	Noninverting Receiver Input
7	1	—	—	—	B	Inverting Receiver Input and Inverting Driver Output
—	—	7	1	11	B	Inverting Receiver Input
8	2	1	3	14	V_{CC}	Positive Supply: 4.75V ≤ V_{CC} ≤ 5.25V
—	—	—	—	1, 8, 13	N.C.	No Connect—not internally connected

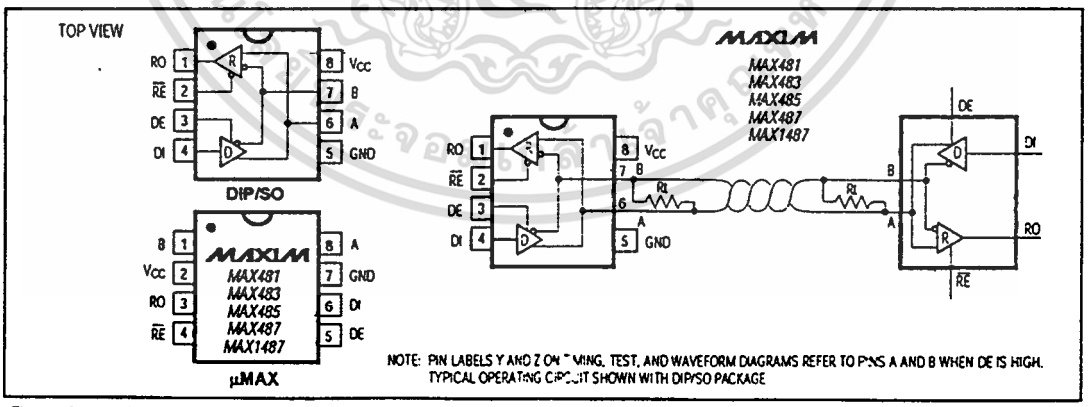


Figure 1. MAX481/MAX483/MAX485/MAX487/MAX1487 Pin Configuration and Typical Operating Circuit

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Low-Power, Slew-Rate-Limited RS-485/RS-422 Transceivers

Switching Waveforms

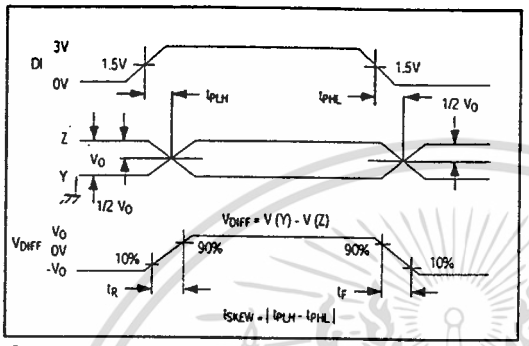


Figure 8. Driver Propagation Delays

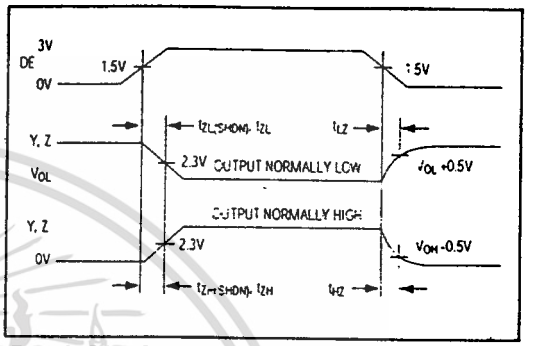


Figure 9. Driver Enable and Disable Times (except MAX488 and MAX490)

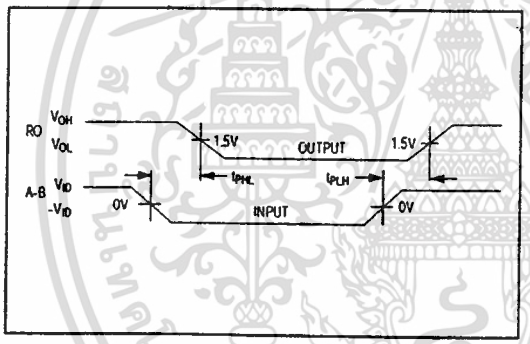


Figure 10. Receiver Propagation Delays

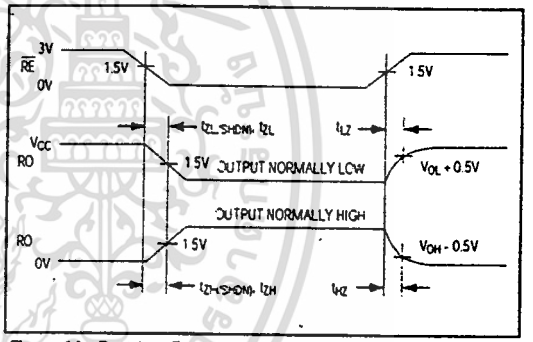


Figure 11. Receiver Enable and Disable Times (except MAX488 and MAX490)

Function Tables (MAX481/MAX483/MAX485/MAX487/MAX1487)

Table 1. Transmitting

INPUTS			OUTPUTS	
$\overline{RE}$	DE	DI	Z	Y
X	1	1	0	1
X	1	0	1	0
0	0	X	High-Z	High-Z
1	0	X	High-Z*	High-Z*

X = Don't care
High-Z = High impedance
* Shutdown mode for MAX481/MAX483/MAX487

Table 2. Receiving

INPUTS			OUTPUT
$\overline{RE}$	DE	A-B	RO
0	0	≥ +0.2V	1
0	0	≤ -0.2V	0
0	0	Inputs open	1
1	0	X	High-Z*

X = Don't care
High-Z = High impedance
* Shutdown mode for MAX481/MAX483/MAX487

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

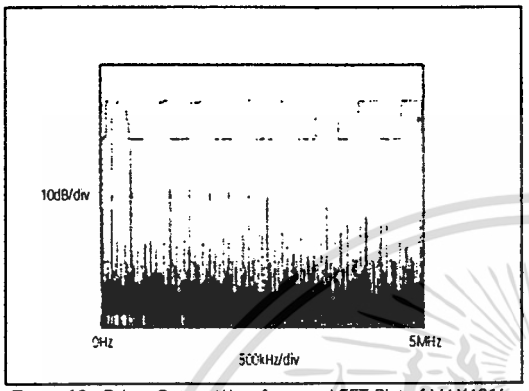


Figure 12. Driver Output Waveform and FFT Plot of MAX481/MAX485/MAX490/MAX491/MAX1487 Transmitting a 150kHz Signal

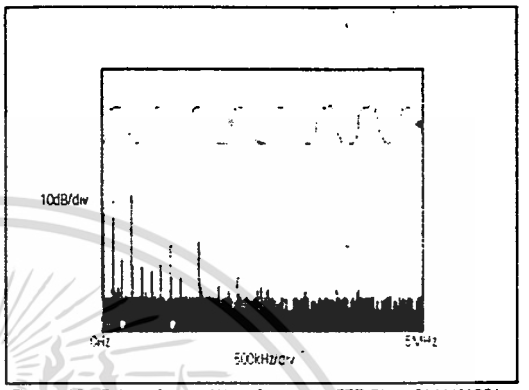


Figure 13. Driver Output Waveform and FFT Plot of MAX483/MAX487–MAX489 Transmitting a 150kHz Signal

Low-Power Shutdown Mode (MAX481/MAX483/MAX487)

A low-power shutdown mode is initiated by bringing both $\overline{RE}$ high and DE low. The devices will not shut down unless both the driver and receiver are disabled. In shutdown, the devices typically draw only 0.1µA of supply current.

$\overline{RE}$ and DE may be driven simultaneously; the parts are guaranteed not to enter shutdown if $\overline{RE}$ is high and DE is low for less than 50ns. If the inputs are in this state for at least 600ns, the parts are guaranteed to enter shutdown.

For the MAX481, MAX483, and MAX487, the t_{ZH} and t_{ZL} enable times assume the part was not in the low-power shutdown state (the MAX485/MAX488–MAX491 and MAX1487 can not be shut down). The $t_{ZH(SHDN)}$ and $t_{ZL(SHDN)}$ enable times assume the parts were shut down (see *Electrical Characteristics*).

It takes the drivers and receivers longer to become enabled from the low-power shutdown state ($t_{ZH(SHDN)}$, $t_{ZL(SHDN)}$) than from the operating mode (t_{ZH}, t_{ZL}). (The parts are in operating mode if the $\overline{RE}$, DE inputs equal a logical 0,1 or 1,1 or 0, 0.)

Driver Output Protection

Excessive output current and power dissipation caused by faults or by bus contention are prevented by two mechanisms. A foldback current limit on the output stage provides immediate protection against short circuits over the whole common-mode voltage range (see *Typical Operating Characteristics*). In addition, a thermal shutdown circuit forces the driver outputs into a high-impedance state if the die temperature rises excessively.

Propagation Delay

Many digital encoding schemes depend on the difference between the driver and receiver propagation delay times. Typical propagation delays are shown in Figures 15–18 using Figure 14's test circuit.

The difference in receiver delay times, $|t_{PLH} - t_{PHL}|$, is typically under 13ns for the MAX481, MAX485, MAX490, MAX491, and MAX1487 and is typically less than 100ns for the MAX483 and MAX487–MAX489.

The driver skew times are typically 5ns (10ns max) for the MAX481, MAX485, MAX490, MAX491, and MAX1487, and are typically 100ns (800ns max) for the MAX483 and MAX487–MAX489.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

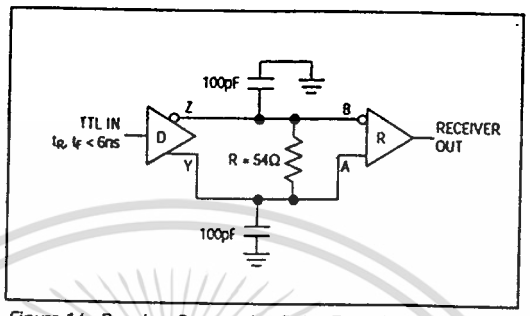


Figure 14. Receiver Propagation Delay Test Circuit

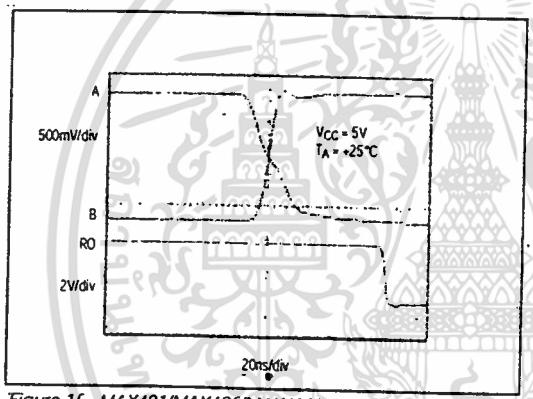


Figure 15. MAX481/MAX485/MAX490/MAX491/MAX1487 Receiver t_{PHL}

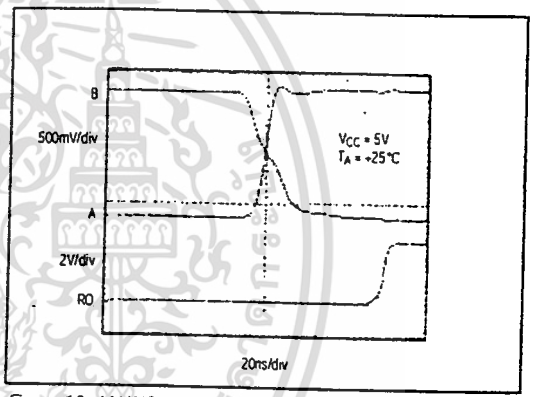


Figure 16. MAX481/MAX485/MAX490/MAX491/MAX1487 Receiver t_{PLH}

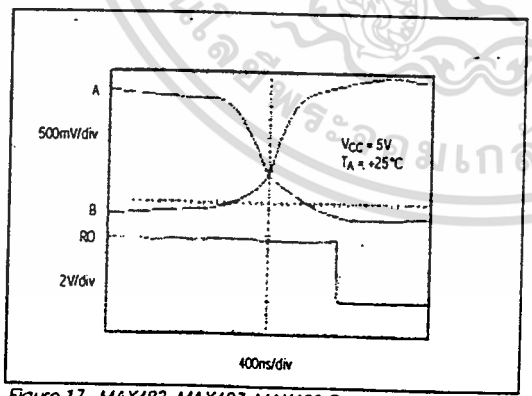


Figure 17. MAX483, MAX487–MAX489 Receiver t_{PHL}

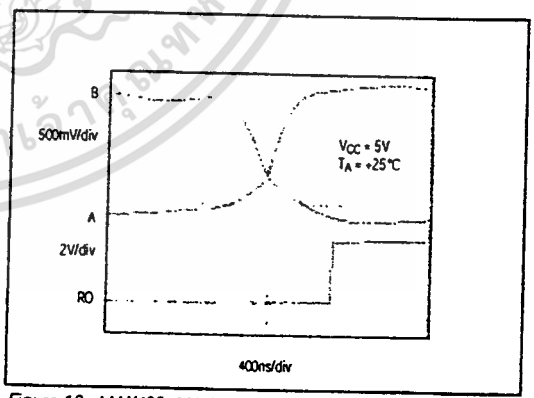


Figure 18. MAX483, MAX487–MAX489 Receiver t_{PLH}

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Low-Power, Slew-Rate-Limited RS-485/RS-422 Transceivers

Line Length vs. Data Rate

The RS-485/RS-422 standard covers line lengths up to 4000 feet. For line lengths greater than 4000 feet, see Figure 23.

Figures 19 and 20 show the system differential voltage for the parts driving 4000 feet of 26AWG twisted-pair wire at 110kHz into 120 Ω loads.

Typical Applications

The MAX481, MAX483, MAX485, MAX487–MAX491, and MAX1487 transceivers are designed for bidirectional data communications on multipoint bus transmission lines.

Figures 21 and 22 show typical network applications circuits. These parts can also be used as line repeaters, with cable lengths longer than 4000 feet, as shown in Figure 23.

To minimize reflections, the line should be terminated at both ends in its characteristic impedance, and stub lengths off the main line should be kept as short as possible. The slew-rate-limited MAX483 and MAX487–MAX489 are more tolerant of imperfect termination.

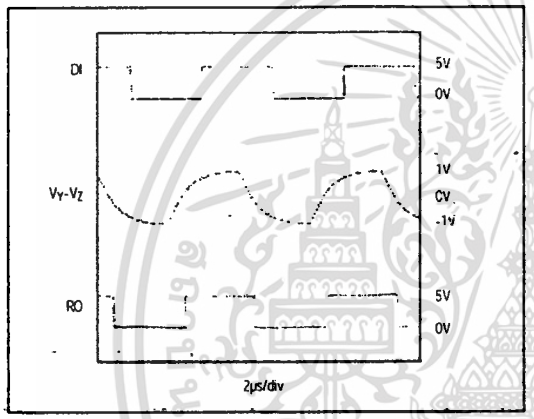


Figure 19. MAX481/MAX485/MAX490/MAX491/MAX1487 System Differential Voltage at 110kHz Driving 4000ft of Cable

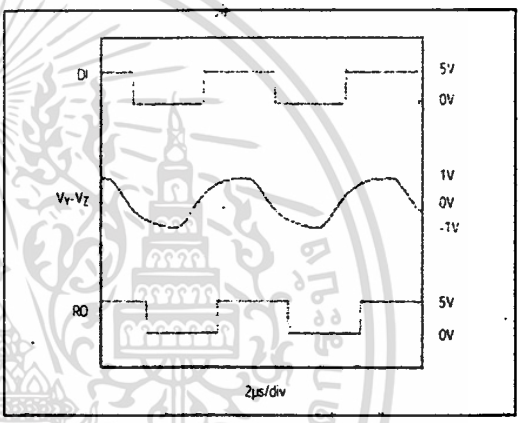


Figure 20. MAX483, MAX487–MAX489 System Differential Voltage at 110kHz Driving 4000ft of Cable

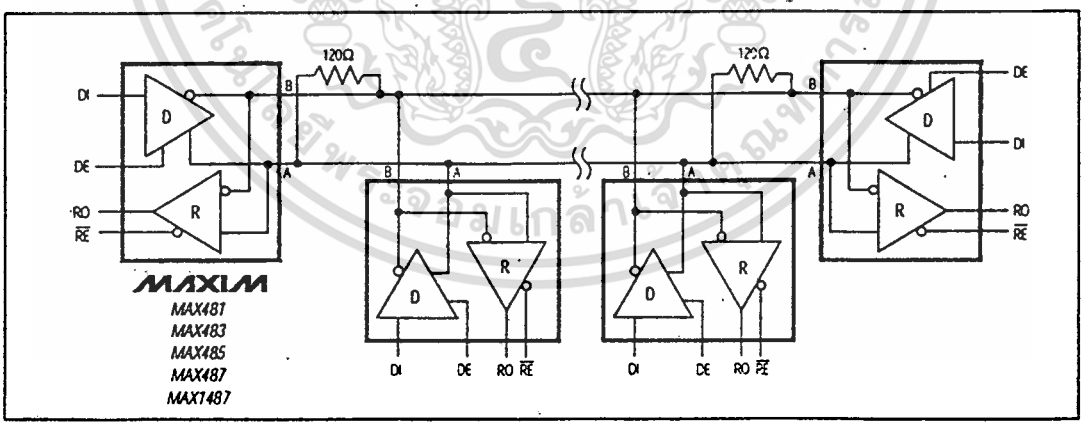


Figure 21. MAX481/MAX483/MAX485/MAX487/MAX1487 Typical Half-Duplex RS-485 Network

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

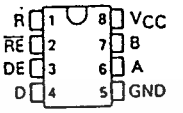
รายละเอียดของ 75176

SN75176A
DIFFERENTIAL BUS TRANSCEIVER

D2619, JUNE 1984—REVISED AUGUST 1989

- Bidirectional Transceiver
- Meets EIA Standards RS-422A and CCITT Recommendations V.11 and X.27
- Designed for Multipoint Transmission on Long Bus Lines in Noisy Environments
- 3-State Driver and Receiver Outputs
- Individual Driver and Receiver Enables
- Wide Positive and Negative Input/Output Bus Voltage Ranges
- Driver Output Capability. . . ±60 mA Max
- Thermal Shutdown Protection
- Driver Positive and Negative Current Limiting
- Receiver Input Impedance . . . 12 kΩ Min
- Receiver Input Sensitivity . . . ±200 mV
- Receiver Input Hysteresis . . . 50 mV Typ
- Operates from Single 5-V Supply
- Low Power Requirements

D OR P
DUAL-IN-LINE PACKAGE
(TOP VIEW)



FUNCTION TABLE (DRIVER)

INPUT	ENABLE	OUTPUTS	
D	DE	A	B
H	H	H	L
L	H	L	H
X	L	Z	Z

FUNCTION TABLE (RECEIVER)

DIFFERENTIAL INPUTS	ENABLE	OUTPUT
A − B	$\overline{RE}$	R
$V_{ID} > 0.2$ V	L	H
$-0.2\text{ V} < V_{ID} < 0.2$ V	L	?
$V_{ID} < -0.2$ V	L	L
X	H	Z

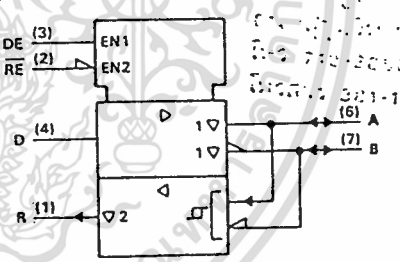
H = high level, L = low level, ? = indeterminate,
X = irrelevant, Z = high impedance (off)

description

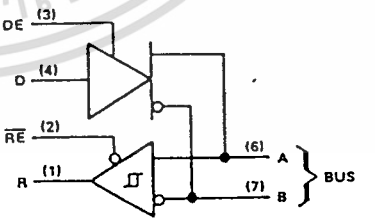
The SN75176A differential bus transceiver is a monolithic integrated circuit designed for bidirectional data communication on multipoint bus transmission lines. It is designed for balanced transmission lines and meets EIA Standard RS-422A and CCITT Recommendations V.11 and X.27.

The SN75176A combines a 3-state differential line driver and a differential-input line receiver both of which operate from a single 5-V power supply. The driver and receiver have active-high and active-low enables, respectively, that can be externally connected together to function as direction control. The driver differential outputs and the receiver differential inputs are connected internally to form differential input/output (I/O) bus ports that are designed to offer minimum loading to the bus whenever the driver is disabled or V_{CC} = 0. These ports feature wide positive and negative common-mode voltage ranges making the device suitable for party-line applications.

logic symbol



logic diagram (positive logic)



PRODUCTION DATA documents contain information current as of publication date. Products conform to specifications per the terms of Texas Instruments standard warranty. Production processing does not necessarily include testing of all parameters.

Copyright © 1989, Texas Instruments Incorporated

TEXAS INSTRUMENTS
POST OFFICE BOX 655303 • DALLAS, TEXAS 75265

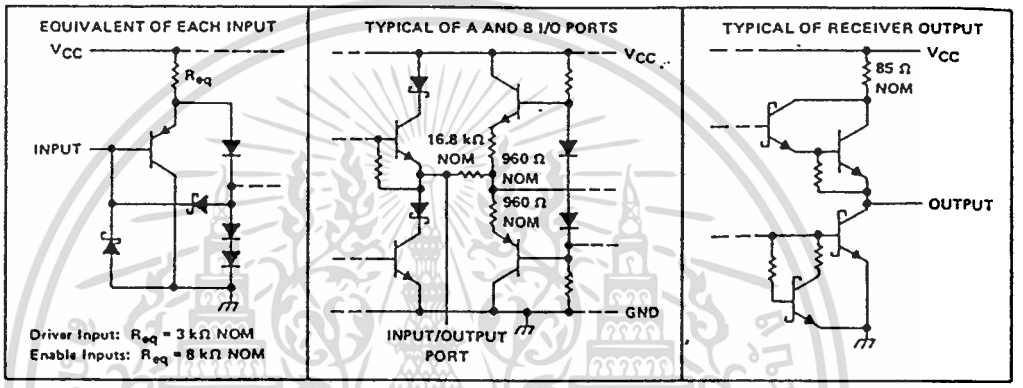
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

description (continued)

The driver is designed to handle loads up to 60 mA of sink or source current. The driver features positive- and negative-current limiting and thermal shutdown for protection from line fault conditions. Thermal shutdown is designed to occur at a junction temperature of approximately 150°C. The receiver features a minimum input impedance of 12 kΩ, input sensitivity of ±200 mV, and a typical input hysteresis of 50 mV.

The SN75176A can be used in transmission line applications employing the SN75172 and SN75174 quadruple differential line drivers and the SN75173 and SN75175 quadruple differential line receivers.

schematics of inputs and outputs



TEXAS INSTRUMENTS
POST OFFICE BOX 655303 • DALLAS, TEXAS 75265

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

absolute maximum ratings over operating free-air temperature range (unless otherwise noted)

Supply voltage, V_{CC} (see Note 1) . . . . . 7 V
Voltage at any bus terminal . . . . . −10 V to 15 V
Enable input voltage . . . . . 5.5 V
Continuous total dissipation at (or below) 25°C free-air temperature (see Note 2):
D package . . . . . 725 mW
P package . . . . . 1000 mW
Operating free-air temperature range . . . . . 0°C to 70°C
Storage temperature range . . . . . −65°C to 150°C
Lead temperature 1,6 mm (1/16 inch) from case for 10 seconds . . . . . 260°C

NOTES: 1. All voltage values, except differential input/output bus voltage, are with respect to network ground terminal.
2. For operation above 25°C free-air temperature, derate the D package to 464 mW at 70°C at the rate of 5.8 mW/°C and derate the P package to 640 mW at 70°C at the rate of 8.0 mW/°C.

recommended operating conditions

		MIN	NOM	MAX	UNIT
Supply voltage, V_{CC}		4.75	5	5.25	V
Voltage at any bus terminal (separately or common-mode), V_I or V_{IC}		−7		12	V
High-level input voltage, V_{IH}	D, DE, and $\overline{RE}$	2			V
Low-level input voltage, V_{IL}	D, DE, and $\overline{RE}$			0.8	V
Differential input voltage, V_{ID} (see Note 3)				±12	V
High-level output current, I_{OH}	Driver			−60	mA
	Receiver			−400	μA
Low-level output current, I_{OL}	Driver			60	mA
	Receiver			8	
Operating free-air temperature, T_A		0		70	°C

NOTE 3: Differential-input/output bus voltage is measured at the noninverting terminal A with respect to the inverting terminal B.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

DRIVER SECTION

driver electrical characteristics over recommended ranges of supply voltage and operating free-air temperature (unless otherwise noted).

PARAMETER		TEST CONDITIONS		MIN	TYP†	MAX	UNIT
V_{IK}	Input clamp voltage	I_I = −18 mA				−1.5	V
V_{OH}	High-level output voltage	V_{IH} = 2 V, V_{IL} = 0.8 V, I_{OH} = −33 mA			3.7		V
V_{OL}	Low-level output voltage	V_{IH} = 2 V, V_{IL} = 0.8 V, I_{OL} = 33 mA			1.1		V
$\|V_{OD1}\|$	Differential output voltage	I_O = 0				2 V_{OD2}	V
$\|V_{OD2}\|$	Differential output voltage	R_L = 100 Ω,	See Figure 1	2	2.7		V
		R_L = 54 Ω,	See Figure 1	1.5	2.4		
$\Delta\|V_{OD}\|$	Change in magnitude of differential output voltage‡					±0.2	V
V_{OC}	Common-mode output voltage§	R_L = 54 Ω or 100 Ω,	See Figure 1			3	V
$\Delta\|V_{OC}\|$	Change in magnitude of common-mode output voltage‡					±0.2	V
I_O	Output current	Output disabled, See Note 4	V_O = 12 V			1	mA
			V_O = −7 V			−0.8	
I_{IH}	High-level input current	V_I = 2.4 V				20	μA
I_{IL}	Low-level input current	V_I = 0.4 V				−400	μA
I_{OS}	Short-circuit output current	V_O = −7 V				−250	mA
		V_O = V_{CC}				250	
		V_O = 12 V				500	
I_{CC}	Supply current (total package)	No load	Outputs enabled		35	50	mA
			Outputs disabled		26	40	

†All typical values are at V_{CC} = 5 V and T_A = 25°C.
‡$\Delta|V_{OD}|$ and $\Delta|V_{OC}|$ are the changes in magnitude of V_{OD} and V_{OC} respectively, that occur when the input is changed from a high level to a low level.
§In EIA Standard RS-422A, V_{OC}, which is the average of the two output voltages with respect to ground, is called output offset voltage, V_{OS}.
NOTE 4: This applies for both power on and power off. Refer to EIA Standard RS-422A for exact conditions.

driver switching characteristics, V_{CC} = 5 V, T_A = 25°C

PARAMETER		TEST CONDITIONS		MIN	TYP	MAX	UNIT
t_{DD}	Differential-output delay time	R_L = 60 Ω,	See Figure 3		40	60	ns
t_{TD}	Differential-output transition time				65	95	ns
t_{PZH}	Output enable time to high level	R_L = 110 Ω,	See Figure 4		55	90	ns
t_{PZL}	Output enable time to low level	R_L = 110 Ω,	See Figure 5		30	50	ns
t_{PHZ}	Output disable time from high level	R_L = 110 Ω,	See Figure 4		85	130	ns
t_{PLZ}	Output disable time from low level	R_L = 110 Ω,	See Figure 5		20	40	ns

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

RECEIVER SECTION

receiver electrical characteristics over recommended ranges of common-mode input voltage, supply voltage, and operating free-air temperature (unless otherwise noted)

PARAMETER		TEST CONDITIONS		MIN	TYP†	MAX	UNIT
V_{TH}	Differential-input high-threshold voltage	V_O = 2.7 V,	I_O = −0.4 mA			0.2	V
V_{TL}	Differential-input low-threshold voltage	V_O = 0.5 V,	I_O = 8 mA	−0.2‡			V
$V_{T+} - V_{T-}$	Hysteresis§				50		mV
V_{IK}	Enable-input clamp voltage	I_I = −18 mA				−1.5	V
V_{OH}	High-level output voltage	V_{ID} = −200 mV, See Figure 2	I_{OH} = −400 μA,	2.7			V
V_{OL}	Low-level output voltage	V_{ID} = −200 mV, See Figure 2	I_{OL} = 8 mA,			0.45	V
I_{OZ}	High-impedance-state output current	V_O = 0.4 V to 2.4 V				±20	μA
I_I	Line input current	Other input = 0 V, See Note 4	V_I = 12 V			1	mA
			V_I = −7 V			−0.8	
I_{IH}	High-level enable-input current	V_{IH} = 2.7 V				20	μA
I_{IL}	Low-level enable-input current	V_{IL} = 0.4 V				−100	μA
r_i	Input resistance			12			kΩ
I_{OS}	Short-circuit output current			−15		−85	mA
I_{CC}	Supply current (total package)	No load	Outputs enabled		35	50	mA
			Outputs disabled		26	40	

†All typical values are at V_{CC} = 5 V, T_A = 25°C.
‡The algebraic convention, where the less-positive (more-negative) limit is designated minimum, is used in this data sheet for common-mode input voltage and threshold voltage levels only.
§Hysteresis is the difference between the positive-going input threshold voltage, V_{T+}, and the negative-going input threshold voltage, V_{T-}. See Figure 4.
NOTE 4: This applies for both power on and power off. Refer to EIA Standard RS-422A for exact conditions.

receiver switching characteristics, V_{CC} = 5 V, T_A = 25°C

PARAMETER		TEST CONDITIONS		MIN	TYP	MAX	UNIT
t_{PLH}	Propagation delay time, low-to-high-level output	V_{ID} = −1.5 V to 1.5 V,			21	35	ns
t_{PHL}	Propagation delay time, high-to-low-level output	C_L = 15 pF,	See Figure 6		23	35	ns
t_{PZH}	Output enable time to high level	C_L = 15 pF,	See Figure 7		10	30	ns
t_{PZL}	Output enable time to low level				12	30	ns
t_{PHZ}	Output disable time from high level	C_L = 15 pF,	See Figure 7		20	35	ns
t_{PLZ}	Output disable time from low level				17	35	ns

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

PARAMETER MEASUREMENT INFORMATION

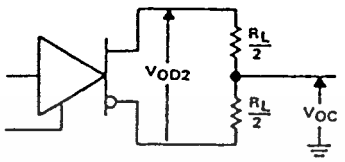


FIGURE 1. DRIVER V_{OD} AND V_{OC}

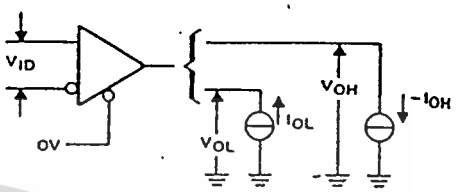
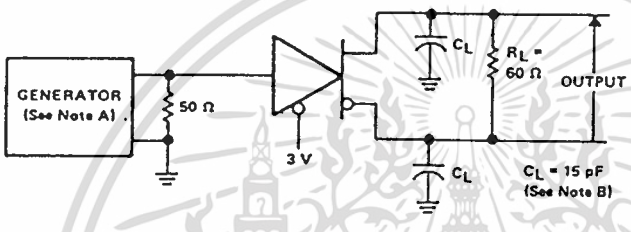
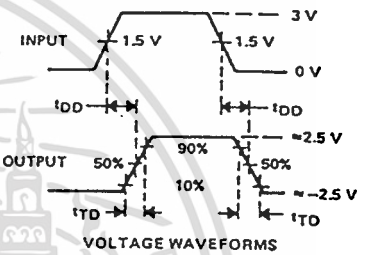


FIGURE 2. RECEIVER V_{OH} AND V_{OL}

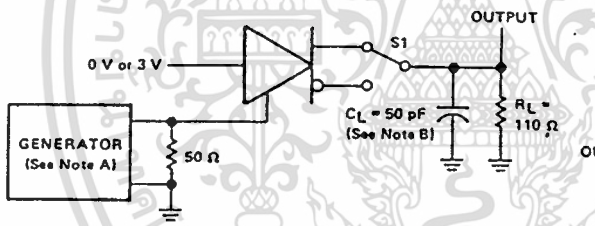


TEST CIRCUIT

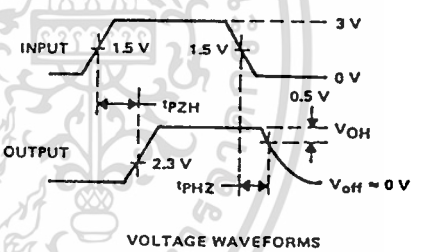


VOLTAGE WAVEFORMS

FIGURE 3. DRIVER DIFFERENTIAL-OUTPUT DELAY AND TRANSITION TIMES

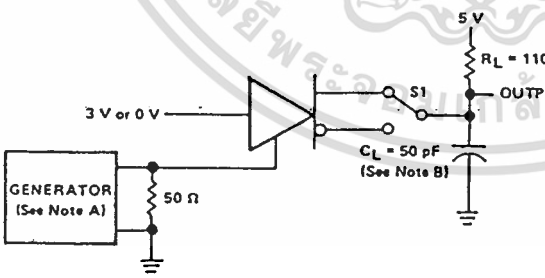


TEST CIRCUIT

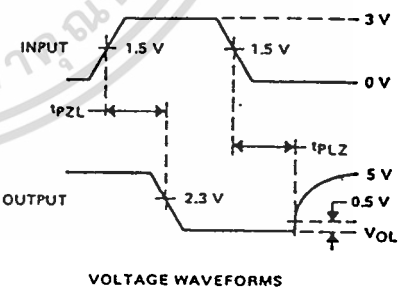


VOLTAGE WAVEFORMS

FIGURE 4. DRIVER ENABLE AND DISABLE TIMES



TEST CIRCUIT



VOLTAGE WAVEFORMS

FIGURE 5. DRIVER ENABLE AND DISABLE TIMES

NOTES: A. The input pulse is supplied by a generator having the following characteristics: PRR = 1 MHz, 50% duty cycle, t_r ≤6 ns, t_f ≤6 ns, Z_{out} = 50 Ω.
B. C_L includes probe and jig capacitance.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

PARAMETER MEASUREMENT INFORMATION

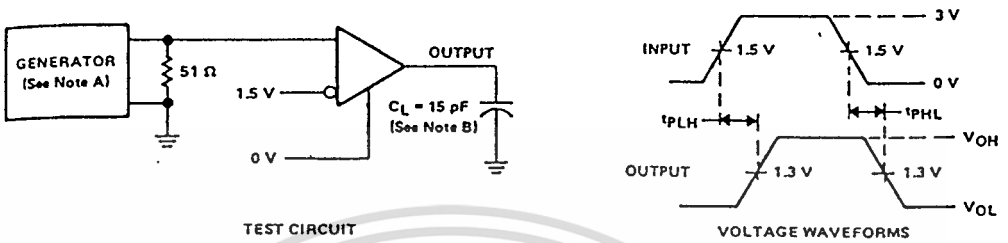


FIGURE 6. RECEIVER PROPAGATION DELAY TIMES

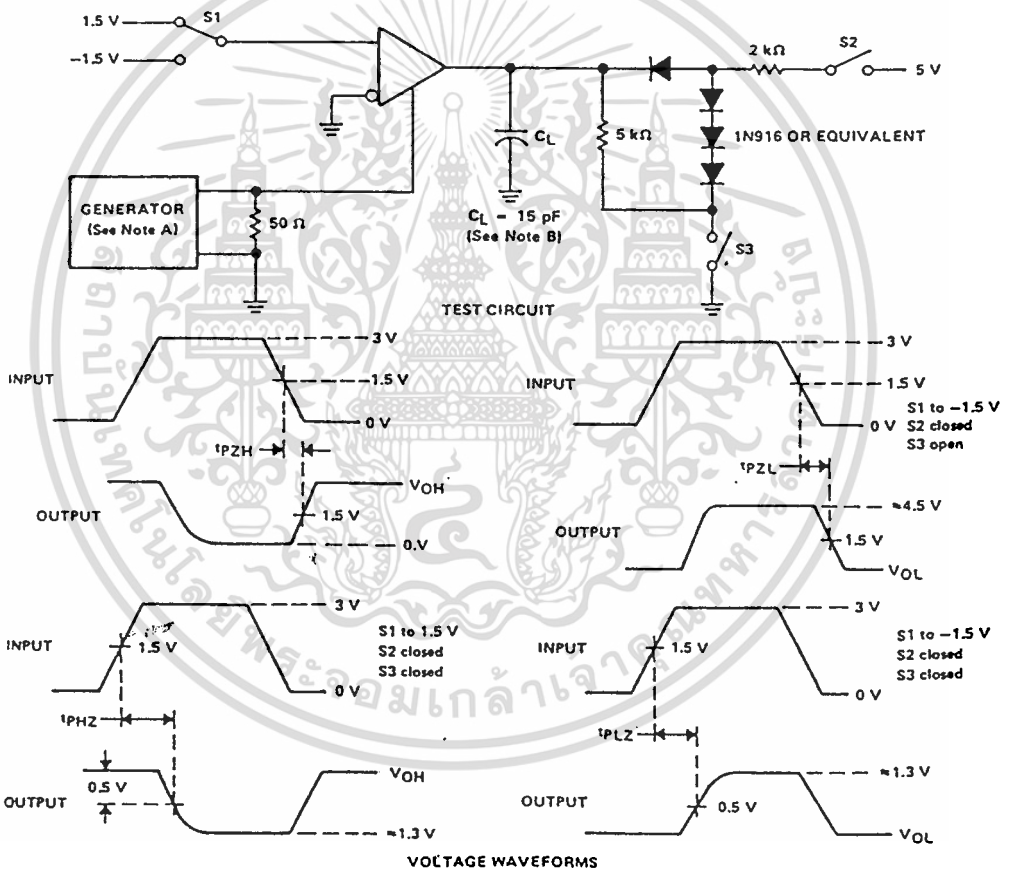


FIGURE 7. RECEIVER OUTPUT ENABLE AND DISABLE TIMES

NOTES: A. The input pulse is supplied by a generator having the following characteristics: PRR = 1 MHz, 50% duty cycle, t_r ≤ 6 ns, t_f ≤ 6 ns, Z_{out} = 50 Ω.
B. C_L includes probe and jig capacitance.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

TYPICAL CHARACTERISTICS

DRIVER HIGH-LEVEL OUTPUT VOLTAGE
vs
DRIVER HIGH-LEVEL OUTPUT CURRENT

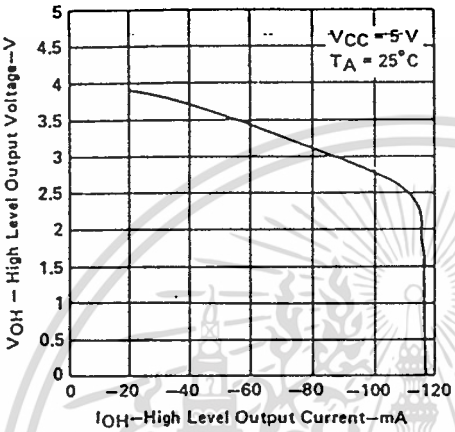


FIGURE 8

DRIVER LOW-LEVEL OUTPUT VOLTAGE
vs
DRIVER LOW-LEVEL OUTPUT CURRENT

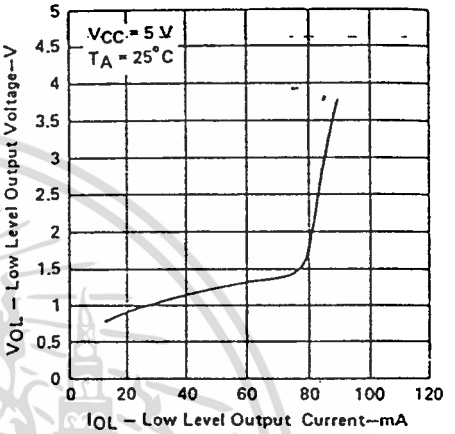


FIGURE 9

DRIVER DIFFERENTIAL OUTPUT VOLTAGE
vs
DRIVER OUTPUT CURRENT

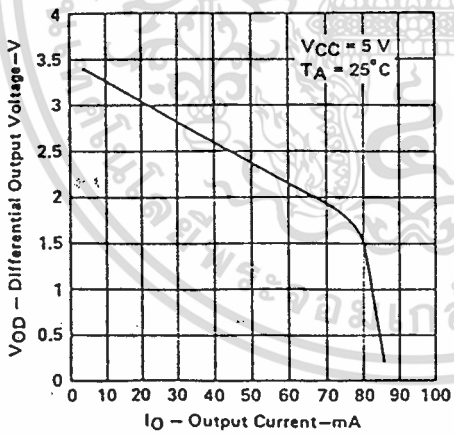


FIGURE 10

RECEIVER LOW-LEVEL OUTPUT VOLTAGE
vs
RECEIVER LOW-LEVEL OUTPUT CURRENT

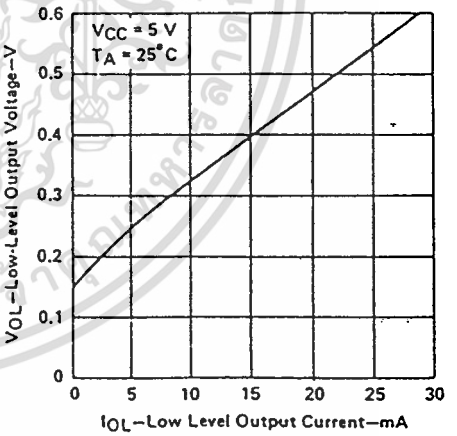


FIGURE 11

TEXAS INSTRUMENTS
POST OFFICE BOX 655303 • DALLAS, TEXAS 75265

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

TYPICAL CHARACTERISTICS

RECEIVER LOW-LEVEL OUTPUT VOLTAGE
vs
FREE-AIR TEMPERATURE

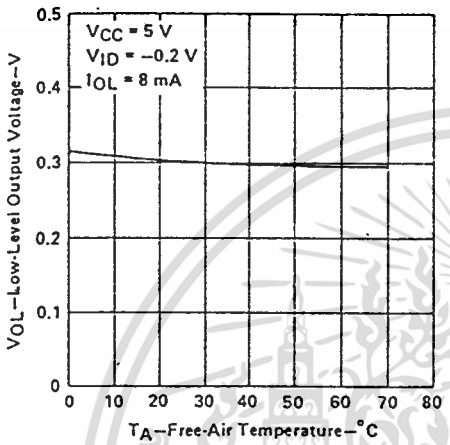


FIGURE 12

RECEIVER OUTPUT VOLTAGE
vs
ENABLE VOLTAGE

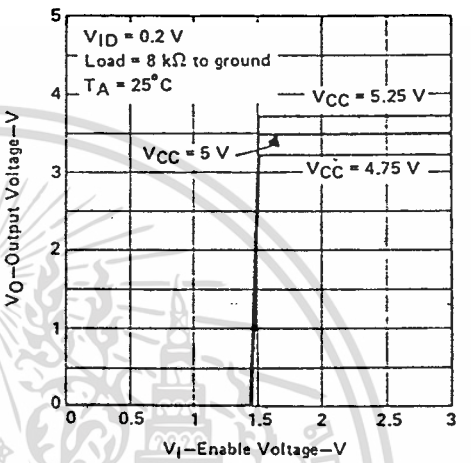


FIGURE 13

RECEIVER OUTPUT VOLTAGE
vs
ENABLE VOLTAGE

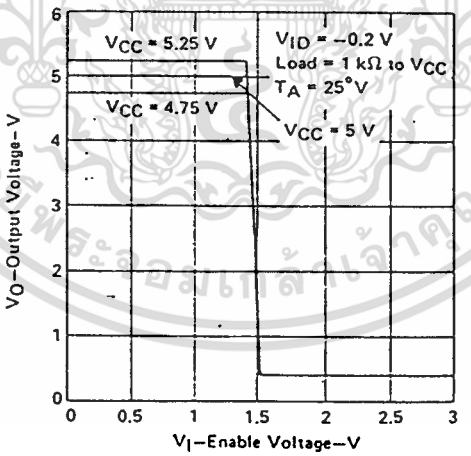
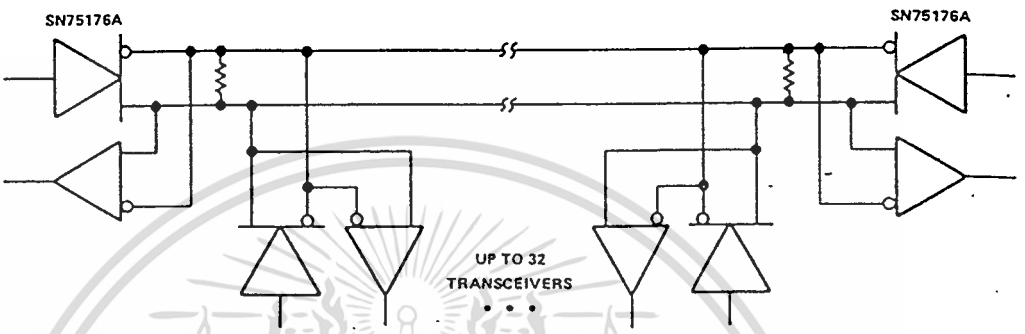


FIGURE 14

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

TYPICAL APPLICATION



NOTE: The line should be terminated at both ends in its characteristic impedance. Stub lengths off the main line should be kept as short as possible.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รายละเอียดของ DS1232 หรือ MAX1232

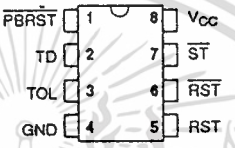
DALLAS SEMICONDUCTOR

DS1232LP/LPS
Low Power MicroMonitor Chip

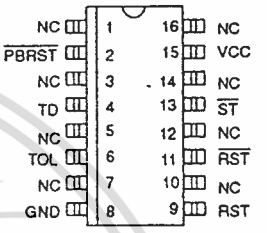
FEATURES

- Super low-power version of DS1232
- 50 µA quiescent current
- Halts and restarts an out-of-control microprocessor
- Automatically restarts microprocessor after power failure
- Monitors pushbutton for external override
- Accurate 5% or 10% microprocessor power supply monitoring
- 8-pin DIP or 8-pin SOIC package
- Optional 16-pin SOIC package available
- Industrial temperature -40°C to +85°C available, designated N

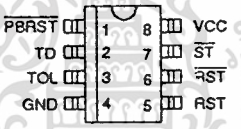
PIN ASSIGNMENT



DS1232LP 8-Pin DIP
(300 Mil)



DS1232LPS 16-Pin SOIC
(300 Mil)



DS1232LPS-2 8-Pin SOIC
(150 Mil)

PIN DESCRIPTION

PBRST	–	Pushbutton Reset Input
TD	–	Time Delay Set
TOL	–	Selects 5% or 10% V_{CC} Detect
GND	–	Ground
RST	–	Reset Output (Active High)
RST	–	Reset Output (Active Low, open drain)
ST	–	Strobe Input
V_{CC}	–	+5 Volt Power

DESCRIPTION

The DS1232LP/LPS Low Power MicroMonitor Chip monitors three vital conditions for a microprocessor: power supply, software execution, and external override. First, a precision temperature-compensated reference and comparator circuit monitors the status of V_{CC}. When an out-of-tolerance condition occurs, an internal power fail signal is generated which forces reset to the active state. When V_{CC} returns to an in-tolerance condition, the reset signals are kept in the active state for a minimum of 250 ms to allow the power supply and processor to stabilize.

The second function the DS1232LP/LPS performs is pushbutton reset control. The DS1232LP/LPS debounces the pushbutton input and guarantees an active reset pulse width of 250 ms minimum. The third function

©Copyright 1995 by Dallas Semiconductor Corporation.
All Rights Reserved. For important information regarding patents and other intellectual property rights, please refer to Dallas Semiconductor databooks.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

is a watchdog timer. The DS1232LP/LPS has an internal timer that forces the reset signals to the active state if the strobe input is not driven low prior to time-out. The watchdog timer function can be set to operate on time-out settings of approximately 150 ms, 600 ms, and 1.2 seconds.

OPERATION - POWER MONITOR

The DS1232LP/LPS detects out-of-tolerance power supply conditions and warns a processor-based system of impending power failure. When V_{CC} falls below a preset level as defined by TOL, the V_{CC} comparator outputs the signals RST and $\overline{RST}$. When TOL is connected to ground, the RST and $\overline{RST}$ signals become active as V_{CC} falls below 4.75 volts. When TOL is connected to V_{CC}, the RST and $\overline{RST}$ signals become active as V_{CC} falls below 4.5 volts. The RST and $\overline{RST}$ are excellent control signals for a microprocessor, as processing is stopped at the last possible moments of valid V_{CC}. On power-up, RST and $\overline{RST}$ are kept active for a minimum of 250 ms to allow the power supply and processor to stabilize.

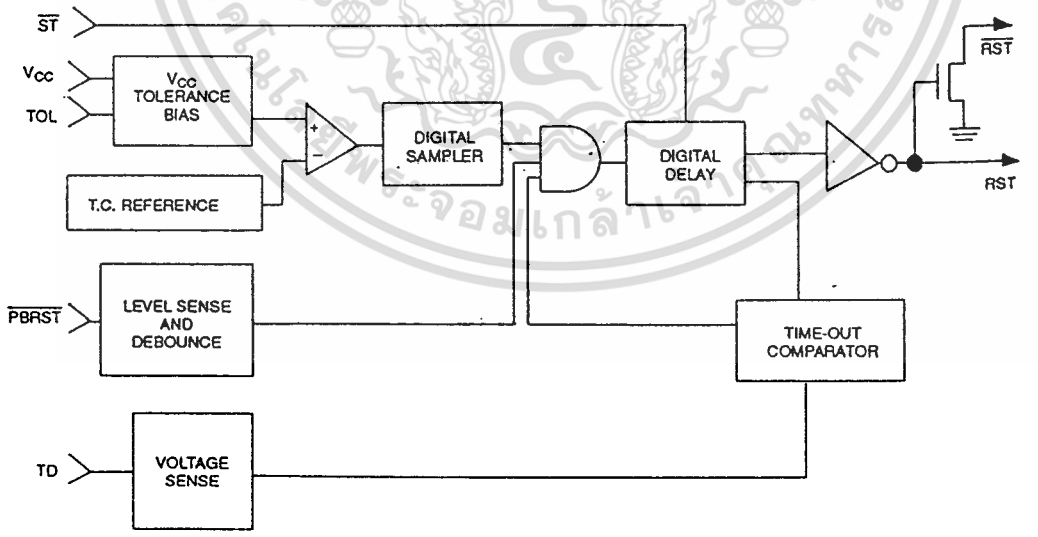
OPERATION - PUSHBUTTON RESET

The DS1232LP/LPS provides an input pin for direct connection to a pushbutton (Figure 1). The pushbutton reset input requires an active low signal. Internally, this input is debounced and timed such that RST and $\overline{RST}$ signals of at least 250 ms minimum are generated. The 250 ms delay starts as the pushbutton reset input is released from low level.

OPERATION - WATCHDOG TIMER

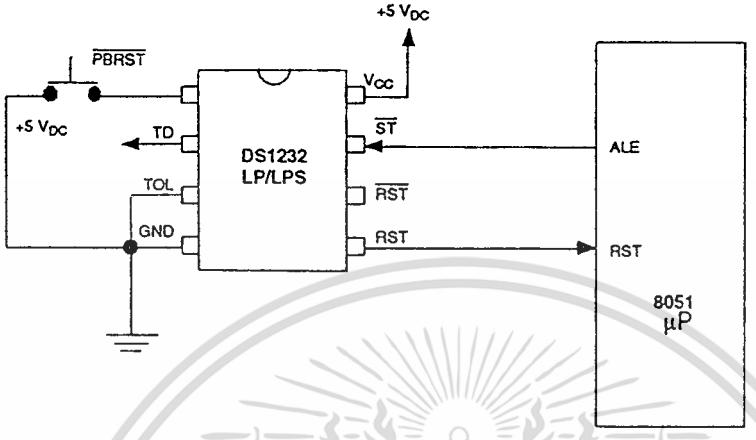
The watchdog timer function forces RST and $\overline{RST}$ signals to the active state when the $\overline{ST}$ input is not stimulated for a predetermined time period. The time period is set by the TD input to be typically 150 ms with TD connected to ground, 600 ms with TD left unconnected, and 1.2 seconds with TD connected to V_{CC}. The watchdog timer starts timing out from the set time period as soon as RST and $\overline{RST}$ are inactive. If a high-to-low transition occurs on the $\overline{ST}$ input pin prior to time-out, the watchdog timer is reset and begins to time-out again. If the watchdog timer is allowed to time-out, then the RST and $\overline{RST}$ signals are driven to the active state for 250 ms minimum. The $\overline{ST}$ input can be derived from microprocessor address signals, data signals, and/or control signals. When the microprocessor is functioning normally, these signals would, as a matter of routine, cause the watchdog to be reset prior to time-out. To guarantee that the watchdog timer does not time-out, a high-to-low transition must occur at or less than the minimum shown in Table 1. A typical circuit example is shown in Figure 2.

MICROMONITOR BLOCK DIAGRAM

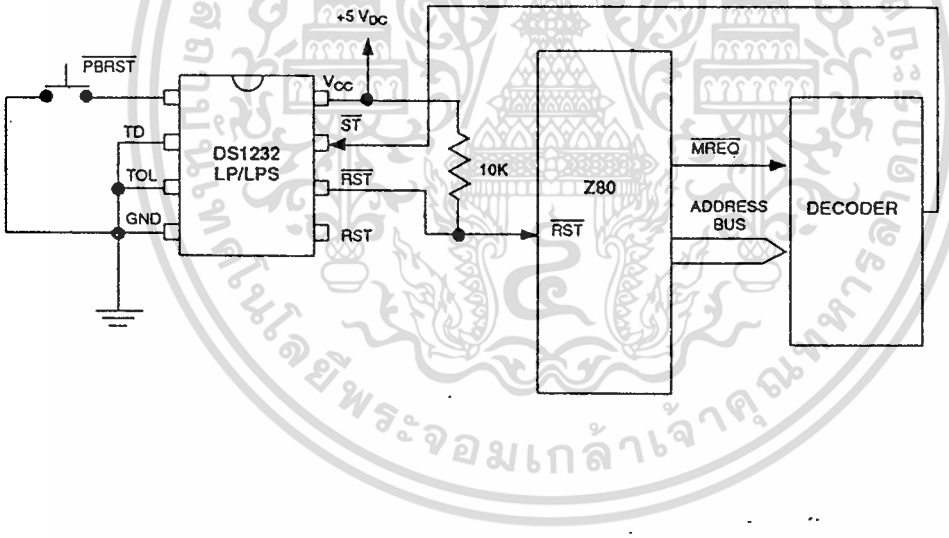


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

PUSHBUTTON RESET Figure 1

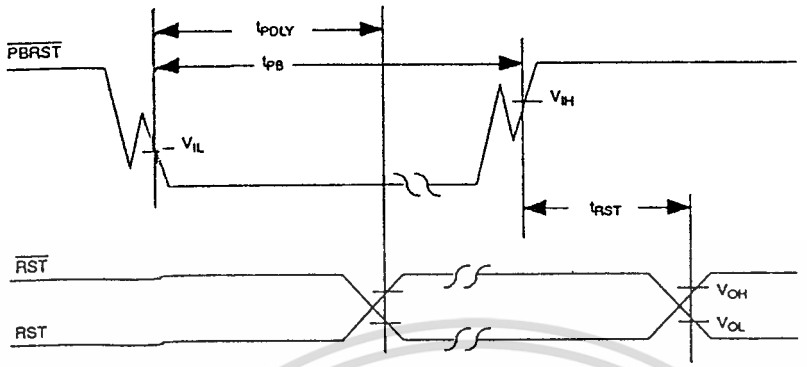


WATCHDOG TIMER Figure 2

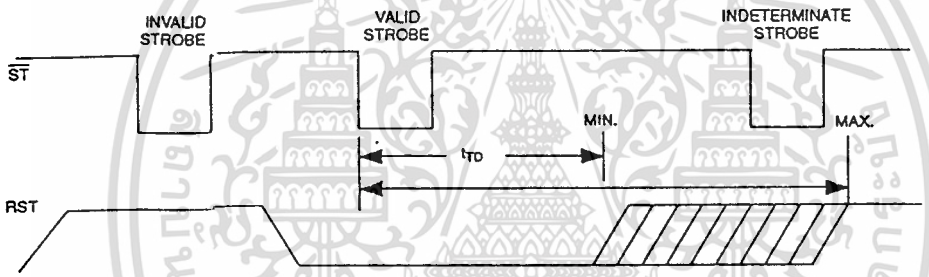


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

TIMING DIAGRAM: PUSHBUTTON RESET Figure 3



TIMING DIAGRAM: STROBE INPUT Figure 4

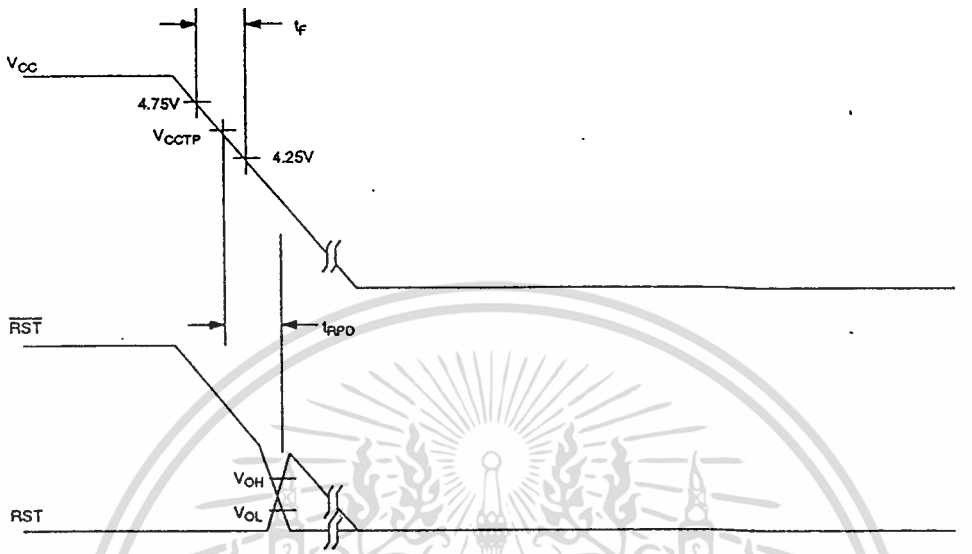


WATCHDOG TIME-OUTS Table 1

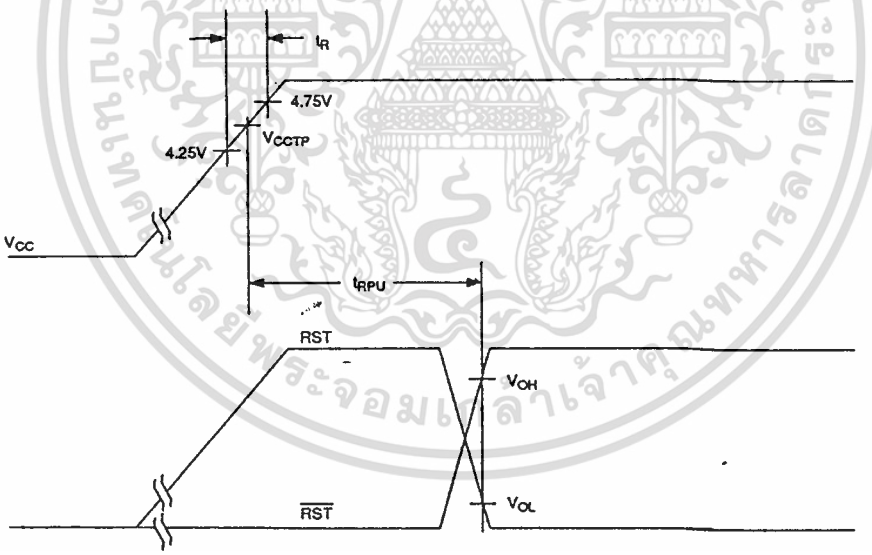
TD	TIME-OUT		
	MIN	TYP	MAX
GND	62.5 ms	150 ms	250 ms
Float	250 ms	600 ms	1000 ms
V_{CC}	500 ms	1200 ms	2000 ms

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

TIMING DIAGRAM: POWER DOWN Figure 5



TIMING DIAGRAM: POWER UP Figure 6



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ABSOLUTE MAXIMUM RATINGS*

Voltage on V_{CC} Pin Relative to Ground	-0.5V to +7.0V
Voltage on I/O Relative to Ground	–0.5V to V_{CC} + 0.5V
Operating Temperature	0°C to 70°C
Operating Temperature (Industrial Version)	–40°C to +85°C
Storage Temperature	-55°C to +125°C
Soldering Temperature	260°C for 10 seconds

* This is a stress rating only and functional operation of the device at these or any other conditions above those indicated in the operation sections of this specification is not implied. Exposure to absolute maximum rating conditions for extended periods of time may affect reliability.

RECOMMENDED DC OPERATING CONDITIONS

(0°C to 70°C)

PARAMETER	SYMBOL	MIN	TYP	MAX	UNITS	NOTES
Supply Voltage	V_{CC}	4.5	5.0	5.5	V	1
$\overline{ST}$ and $\overline{PBRST}$ Input High Level	V_{IH}	2.0		V_{CC}+0.3	V	1
$\overline{ST}$ and $\overline{PBRST}$ Input Low Level	V_{IL}	-0.3		+0.8	V	1

DC ELECTRICAL CHARACTERISTICS

(0°C to 70°C; V_{CC} = 4.5 to 5.5V)

PARAMETER	SYMBOL	MIN	TYP	MAX	UNITS	NOTES
Input Leakage	I_{IL}	-1.0		+1.0	µA	3
Output Current @ 2.4V	I_{OH}	-8	-10		mA	5
Output Current @ 0.4V	I_{OL}	10			mA	
Low Level @ RST	V_{OL}			0.4	V	1
Output Voltage @ -500 uA	V_{OH}	V_{CC} -0.5V	V_{CC} -0.1V		V	1, 7
Operating Current (CMOS)	I_{CC1}			50	µA	2
Operating Current (TTL)	I_{CC2}		200	500	µA	8
V_{CC} Trip Point (TOL = GND)	V_{CCTP}	4.50	4.62	4.74	V	1
V_{CC} Trip Point (TOL = V_{CC})	V_{CCTP}	4.25	4.37	4.49	V	1

CAPACITANCE

(t_A = 25°C)

PARAMETER	SYMBOL	MIN	TYP	MAX	UNITS	NOTES
Input Capacitance	C_{IN}			5	pF	
Output Capacitance	C_{OUT}			7	pF	

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

AC ELECTRICAL CHARACTERISTICS

(0°C to 70°C; V_{CC} = 5V ± 10%)

PARAMETER	SYMBOL	MIN	TYP	MAX	UNITS	NOTES
$\overline{PBRST}$ = V_{IL}	t_{PB}	20			ms	
RESET Active Time	t_{RST}	250	610	1000	ms	
$\overline{ST}$ Pulse Width	t_{ST}	20			ns	6, 9
V_{CC} Fail Detect to RST and $\overline{RST}$	t_{RPD}		50	175	µs	
V_{CC} Slew Rate 4.75V to 4.25V	t_F	300			µs	
V_{CC} Detect to RST and $\overline{RST}$ Inactive	t_{RPU}	250	610	1000	ms	4
V_{CC} Slew Rate 4.25V to 4.75V	t_R	0			ns	
$\overline{PBRST}$ Stable Low to $\overline{RST}$ and RST	t_{PDLY}			20	ms	

NOTES:

1. All voltages referenced to ground.
2. Measured with outputs open and $\overline{ST}$ and $\overline{PBRST}$ within 0.5V of supply rails.
3. $\overline{PBRST}$ is internally pulled up to V_{CC} with an internal impedance of 40K typical.
4. t_R = 5 µs.
5. $\overline{RST}$ is an open drain output.
6. Must not exceed t_{TD} minimum. See Table 1.
7. RST remains within 0.5V of V_{CC} on power-down until V_{CC} drops below 2.0V. $\overline{RST}$ remains within 0.5V of GND on power-down until V_{CC} drops below 2.0V.
8. Measured with outputs open and $\overline{ST}$ and $\overline{PBRST}$ at TTL levels.
9. Watchdog can not be disabled. It must be strobed to avoid resets.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ค.

โปรแกรมการจัดการและควบคุมระบบควบคุมการเข้าออกโดยใช้บัตรแม่เหล็ก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โปรแกรมจัดการและควบคุมระบบส่วนเทอร์มินอลโมดูล

(ไฟล์ T485.ASM)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
            ORG   0000H
BANK0:      DS    8
BANK1:      DS    8
BANK2:      DS    8
BANK3:      DS    8
BITADB:     DS    2
MAGBUF:     DS    40
MAGWRK:     DS    40
STACK:      DS    22
            ORG   MAGWRK
RFCOUNT:    DS    1                    ;**RFTABLE**
RFTABLE:    DS    7
TEST:       DS    1
            ORG   BANK2
SFTABLE:    DS    7                    ;**SFTABLE**
CONTACT:    DS    1

ADDRESS     EQU   00000001B            ;**TM1 ADDRESS**
SERVER      EQU   10000001B
IBMPC       EQU   11111111B
BROADCAST   EQU   11111111B
                                       ;**PORT 1**
CARDTD      EQU   P1.0                 ;DATA
CARDTC      EQU   P1.1                 ;CLOCK
CARDCP      EQU   P1.2                 ;PRESENT OF MC
LED         EQU   P1.3
SIGNAL      EQU   P1.4
                                       ;**BIT ADDRESS**
CARDFF      EQU   00H                  ;CARD FIRST   FLAG
CARDER      EQU   01H                  ;CARD ERROR   FLAG
CARDRF      EQU   02H                  ;CARD REVERSE FLAG
CONNECT     EQU   03H                  ;STATUS CONNECTED
BUSY        EQU   04H                  ;STATUS SERVER BUSY!
REQ         EQU   05H                  ;ERROR FLAG
RFINISH     EQU   06H                  ;RECEIVE SUCCESS
POLL        EQU   07H
B7          EQU   08H                  ;FOR CRC
TEMP        EQU   09H                  ;FOR CRC
                                       ;**RAM ADDRESS**
RFSF        EQU   RFTABLE
RFAF        EQU   RFTABLE+1
RFCF        EQU   RFTABLE+2
RFIF        EQU   RFTABLE+3
RFCRC       EQU   RFTABLE+4
RFXF        EQU   RFTABLE+6
SFSF        EQU   SFTABLE
SFAF        EQU   SFTABLE+1
SFCF        EQU   SFTABLE+2
SFIF        EQU   SFTABLE+3
SFCRC       EQU   SFTABLE+4
SFXF        EQU   SFTABLE+6
                                       ;**VALUE**
BR96        EQU   0FDH
BR48        EQU   0FAH
BR24        EQU   0F4H
BR12        EQU   0E8H
DOUBLE      EQU   80H
NORMAL      EQU   00H
                                       ;**PROTOCAL VALUE**
DM          EQU   00001111B
UA          EQU   01100011B
BZ          EQU   11100011B
FREE        EQU   00011111B
SABM        EQU   00111111B
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น ห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
DISC    EQU  01000011B
MTU     EQU  01001111B
SIM     EQU  10000011B
NRGT    EQU  10111111B
PCBC    EQU  00011011B
DBYTE   EQU  00000000B

        ORG  000H
        LJMP PREPARE

        ORG  0023H
        LJMP 0600H

        ORG  0200H
;-------------------------------------------------------------------------
PREPARE:
        MOV  R2,#00H
POWERUP:
        MOV  R3,#00H
        DJNZ R3,$
        DJNZ R2,POWERUP
        MOV  SP,#STACK

        SETB P1.0                ;SET INPUT
        SETB P1.1
        SETB P1.2
        MOV  TL1,#BR24           ;2400 BAUD RATE
        MOV  TH1,#BR24
        MOV  SCON,#11110000B     ;SERIAL MODE 3 (9BIT UART)
        MOV  TMOD,#00100000B     ;TIMER 1 MODE 2(AUTORELOAD)
        MOV  PCON,#NORMAL        ;NORMAL BUAD RATE
        SETB EA
        SETB ES
        CLR  T0                  ;RECEIVE STAND-BY
        CLR  CONNECT             ;STATUS DISCONNECT
        CLR  BUSY
        CLR  REQ
        CLR  RFINISH
        CLR  SIGNAL
        MOV  SFSF,#01111110B
        MOV  SFAF,#ADDRESS       ;PROVIDE SELF-ADDR TO SERVER
        MOV  SFXF,#01111110B
        MOV  TCON,#01000000B     ;TR1 ON
        CLR  P1.3
        CLR  P1.4
        CLR  P1.5
        CLR  P1.6
        CLR  P1.7
;-------------------------------------------------------------------------

        ;*** MAIN ***
        SETB   CARDCP
MAIN:
        JNB    CARDCP,MAG_RD     ;WAIT FOR CP=0
        JNB    RFINISH,MAIN      ;POLL BROADCAST
        CLR    RFINISH
        CALL   EDIT
        SJMP   MAIN
;-------------------------------------------------------------------------
; *** MAG READ ***
MAG_RD:
        CALL   CARDRD
        CALL   CARDAJ
        JB     CARDER,MAIN
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น ห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                CALL    CARDCL
                SJMP    MAIN
;-------------------------------------------------------------------------
;               SUBROUTINE
;-------------------------------------------------------------------------
; *** CARDCL SUB ***
; CLEAR MAGBUF

CARDCL:
                MOV     R0,#MAGBUF
                MOV     R1,#40
CARDCL1:        MOV     @R0,#0
                INC     R0
                DJNZ    R1,CARDCL1
                RET

; *** CARDRD SUB ***
; MAGNETIC CARD READ (RAW DATA)
; OUT = MAGBUF
; REG = A,R0,R1,R2

CARDRD:         MOV     R0,#MAGBUF
                SETB    CARDFF                  ;SET FISRT FLAG

CARDRD2:        MOV     R2,#5                   ;5 BIT DATA
                CLR     A
CARDRD4:        JB      CARDCP,CARDRD8          ;CHECK CP=1 (END CARD)
                JB      CARDTC,CARDRD4          ;WAIT FOR CLOCK=0
                MOV     R1,#2                   ;DELAY
                DJNZ    R1,$
                MOV     C,CARDTD                ;READ DATA
                JNB     CARDFF,CARDRD5
                JC      CARDRD4                 ;CHECK FIRST=0 (IF BIT C=1
                                                ;IT INDICATES START BIT,
                                                ;NOTHING TO DO)
                CLR     CARDFF
CARDRD5:        RRC     A
                JNB     CARDTC,$                ;WAIT FOR CLOCK=1
                DJNZ    R2,CARDRD4

                RRC     A
                RRC     A
                RRC     A
                CPL     A
                ANL     A,#00011111B
                MOV     @R0,A
                INC     R0
                CJNE    R0,#MAGBUF+40,CARDRD2
CARDRD8:        RET

; *** CARDAJ SUB ***
; MAGNETIC CARD ADJUST DATA (CHECK REVERSE,
; CHECK PARITY,CHANGE TO ASCII)
; IN  = MAGBUF  RAW DATA
; OUT = MAGBUF  ASCII DATA (40 BYTE)
; CARDER  0=OK 1=ERROR
; REG = A,R0,R1,R2,R3,R4

CARDAJ:         MOV     R0,#MAGBUF              ;BACKUP TO MAGWRK
                MOV     R1,#MAGWRK
                MOV     R2,#40
CARDAJ1:        MOV     A,@R0
                MOV     @R1,A
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                INC     R0
                INC     R1
                DJNZ    R2,CARDAJ1

                CLR     CARDRF              ;FORWARD
                CALL    CARDCK
                JB      CARDER,CARDAJ8
                CALL    CARDAX
                JB      CARDER,CARDAJ8
                RET

CARDAJ8:        SETB    CARDRF              ;REVERSE
                CALL    CARDRE
                CALL    CARDCK
                CALL    CARDAX
                RET
; *** CARDAX SUB ***
; MAGNETIC CARD LAST CHECK (FIRST & LAST DATA)
; REG = A,R0

CARDAX:         CLR     CARDER              ;CHECK DATA (B)
                MOV     R0,#MAGBUF
                MOV     A,@R0
                CJNE    A,#'B',CARDAX5

                MOV     R0,#MAGBUF+39       ;CHECK DATA (F)
CARDAX1:        MOV     A,@R0
                CJNE    A,#0,CARDAX2
                DEC     R0
                CJNE    R0,#MAGBUF-1,CARDAX1
                SJMP    CARDAX5

CARDAX2:        DEC     R0
                MOV     A,@R0
                CJNE    A,#'F',CARDAX5
                RET                         ;EXIT OK

CARDAX5:        SETB    CARDER              ;EXIT DATA ERROR
                RET

; *** CARDRE SUB ***
; MAGNETIC CARD REVERSE CHECK
; IN  = MAGWRK  RAW DATA (REVERSE)
; OUT = MAGBUF  RAW DATA (FORWARD)
; REG = A,R0,R1,R2,R3,R4

CARDRE:         CALL    CARDCL
                MOV     R0,#MAGWRK+39       ;REVERSE DATA
                MOV     R1,#MAGBUF          ;FORWARD DATA
                MOV     R2,#5               ;MAGWRK BIT COUNT
                MOV     R3,#5               ;MAGBUF BIT COUNT
                MOV     R4,#0               ;WORK DATA
                CLR     CARDFF

CARDRE1:        MOV     A,@R0               ;SKIP ZERO BYTE
                CJNE    A,#0,CARDRE2
                DEC     R0
                CJNE    R0,#MAGWRK-1,CARDRE1
                RET                         ;ERROR (ALL MAGWRK ZERO)

CARDRE2:        MOV     A,@R0
                RLC     A
                RLC     A
                RLC     A
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น รวมทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
CARDRE3:        RLC     A                       ;FIND FIRST BIT = 1
                JC      CARDRE5
                DJNZ    R2,CARDRE3
                RET                             ;ERROR (NOTHING)

CARDRE5:        PUSH    ACC                     ;SHIFT BIT DATA *** LOOP ***
                MOV     A,R4
                RRC     A
                MOV     R4,A
                POP     ACC

                DJNZ    R2,CARDRE6              ;NEXT BYTE OF MAGWRK
                DEC     R0
                CJNE    R0,#MAGWRK-1,CARDRE51
                SETB    CARDFF
                CLR     A
                SJMP    CARDRE52
CARDRE51:       MOV     A,@R0
                RLC     A                       ;DON'T CARE 3 BIT LEFT
                RLC     A
                RLC     A
CARDRE52:       MOV     R2,#5

CARDRE6:        DJNZ    R3,CARDRE8              ;NEXT BYTE OF MAGBUF
                PUSH    ACC                     ;--\
                MOV     A,R4
                RRC     A
                RRC     A
                RRC     A
                ANL     A,#1FH
                MOV     @R1,A
                INC     R1
                POP     ACC                     ;--/
                JNB     CARDFF,CARDRE61
                RET                             ;EXIT OK
CARDRE61:       MOV     R3,#5

CARDRE8:        RLC     A                       ;NEXT BIT
                SJMP    CARDRE5

;*** CARDCK SUB ***
;MAGNETIC CARD CHECK PARITY & CONVERT TO ASCII
;IN  = MAGBUF  RAW DATA
;OUT = MAGBUF  ASCII DATA (40 BYTE)
;CARDER  0=OK 1=ERROR (PARITY ERROR)
;REG = A,R0,R1

CARDCK:         CLR     CARDER                  ;ERROR=0
                MOV     R0,#MAGBUF
                MOV     R1,#40
CARDCK2:        MOV     A,@R0
                CJNE    A,#0,CARDCK3
                RET                             ;EXIT NO ERROR
                                                ;(UNDER 40 BYTE)

CARDCK3:        JNB     PSW.0,CARDCK8
                ANL     A,#0FH                  ;CONVERT TO ASCII
                CJNE    A,#0AH,$+3
                JNC     CARDCK4
                ORL     A,#30H
                SJMP    CARDCK6
CARDCK4:        SUBB    A,#9
                ORL     A,#40H
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
CARDCK6:        MOV     @R0,A           ;NEXT
                INC     R0
                DJNZ    R1,CARDCK2
                RET                     ;EXIT NO ERROR

CARDCK8:        SETB    CARDER          ;ERROR=1
                RET
;---------------END CARD READ--------------------------------
; *** LINK ***
LINK:
                CLR     P1.5
                CLR     P1.6
                CLR     P1.7

                JB      BUSY,LINKS              ;WAIT FOR THE LINE
                SJMP    LINKT
LINKS:
                JNB     RFINISH,$
                CLR     RFINISH
                MOV     A,RFIF
                CJNE    A,#FREE,LINKS
LINKT:
                MOV     CONTACT,#SERVER         ;SEND DM
                MOV     SFSF,#7FH
                MOV     SFAF,#ADDRESS
                MOV     SFCF,#DM
                MOV     SFIF,#0
                MOV     SFCRC,SFIF
                MOV     SFCRC+1,SFIF
                MOV     SFXF,#7FH
                ACALL   SEND
                SETB    P1.7

                JNB     RFINISH,$               ;WAIT UA
                CLR     RFINISH
                MOV     A,RFCF
                CJNE    A,#UA,LINKE

    ;-------------------------------------------
                MOV     CONTACT,#SERVER         ;SEND DUMB
                ACALL   SENDFULL
    ;-------------------------------------------
                MOV     SFCF,#DBYTE             ;SEND 40 DATA
                MOV     CONTACT,#SERVER
                MOV     R4,#40
                MOV     R0,#MAGBUF
                MOV     A,SFCF
LINK1:
                MOV     SFIF,@R0
                INC     R0
                INC     A
                MOV     SFCF,A
                CALL    SEND
                DJNZ    R4,LINK1
LINK2:                                          ;WAIT SABM OR DISC OR MTU
                JNB     RFINISH,$
                CLR     RFINISH
                MOV     A,RFCF
                CJNE    A,#SABM,LINK3           ;CONNECT
                SETB    SIGNAL                  ;IMPORTANT !
                SJMP    LINKX
LINK3:          CJNE    A,#DISC,LINK4           ;DISCONNECT
                CLR     SIGNAL
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                SJMP   LINKX

LINK4:          CJNE   A,#MTU,LINK5          ;MULTIUSER
                SJMP   LINKX
LINK5:
                CJNE   A,#NRGT,LINKE         ;UNREGISTERED USER
                SJMP   LINKX

LINKE:          CPL    P1.5                  ;ERROR
                CALL   DELAYX
                SJMP   LINKX
LINKX:
                RET

; *** EDIT ***
EDIT:
                CLR    P1.5
                CLR    P1.7

                MOV    A,RFIF                ;CHECK INFORMATION
COMMAND_TABLE:
                CJNE   A,#BZ,CT2             ;LINE BUSY
                SETB   BUSY
                SJMP   EDITX
        CT2:    CJNE   A,#FREE,CT3           ;LINE FREE
                CLR    BUSY
                SJMP   EDITX
        CT3:    CJNE   A,#PCBC,CT4           ;PC_BROADCAST(PC)
                CALL   BRCAST
                SJMP   EDITX
        CT4:    CJNE   A,#SIM,CT5            ;SIMULTANEOUS
                SJMP   EDITX
        CT5:
                SJMP   EDITX
EDITX:
                MOV    C,BUSY
                MOV    P1.6,C

                RET

; *** BROADCAST ***
BRCAST:
                CPL    P1.5
                CALL   DELAYX
                RET

; *** SEND ***
;REG = R1,R2,A,CONTACT

SEND:
                PUSH ACC
                SETB T0                      ;ENABLE DRIVER
                SETB TB8                     ;SEND ACTIVATE ADDR
                MOV  A,CONTACT
                MOV  SBUF,A

                CLR  TB8                     ;SEND DATA
                MOV  R2,#7                   ;QUEUE FOR FRAME
                MOV  R1,#SFTABLE
   SENDNEXT:    JNB  TI,SENDNEXT
                CLR  TI
                MOV  A,@R1
                MOV  SBUF,A
                INC  R1
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น รวมทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                DJNZ R2,SENDNEXT
                JNB  TI,$
                CLR  TI
                SETB TB8
                MOV  A,CONTACT          ;ADDRESS FINISH
                MOV  SBUF,A
                JNB  TI,$
                CLR  TI
                CALL DELAY
                CLR  TB8
                CLR  T0                 ;ENABLE RECEIVER (STAND BY)
                CLR  TI
                POP  ACC
                RET
SENDFULL:
                PUSH ACC
                SETB T0                 ;ENABLE DRIVER
                SETB TB8                ;SEND ACTIVATE ADDR
                MOV  A,CONTACT
                MOV  SBUF,A

                CLR  TB8                ;SEND DATA
                MOV  R2,#7              ;QUEUE FOR FRAME
                MOV  R1,#SFTABLE
SENDNEXTFULL:   JNB  TI,$
                CLR  TI
                MOV  A,@R1
                MOV  SBUF,A
                INC  R1
                DJNZ R2,SENDNEXTFULL
                JNB  TI,$
                CLR  TI

                SETB TB8
                MOV  A,CONTACT          ;ADDRESS FINISH
                MOV  SBUF,A
                JNB  TI,$
                CLR  TI
                MOV  SBUF,A

                JNB  TI,$
                CLR  TI
                CALL DELAY
                CLR  TB8
                CLR  T0                 ;ENABLE RECEIVER (STAND BY)
                CLR  TI
                POP  ACC
                RET

;*** DELAY ***
DELAY:
                MOV  R3,#0FFH
                DJNZ R3,$
                MOV  R3,#0FFH
                DJNZ R3,$
                MOV  R3,#0FFH
                DJNZ R3,$
                MOV  R3,#0FFH
                DJNZ R3,$
                RET
;*** DELAY LONG ***
;REG = R3
DELAYX:
                MOV  R2,#0FFH
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
DELAYX1:        MOV   R3,#0FFH
                DJNZ  R3,$
                MOV   R3,#0FFH
                DJNZ  R3,$
                MOV   R3,#0FFH
                DJNZ  R3,$
                MOV   R3,#0FFH
                DJNZ  R3,$
                DJNZ  R2,DELAYX1
                RET

;-------------------------------------------------------------------------
;       INTERUPT SERVICE ROUTINE : SI VECTOR
;-------------------------------------------------------------------------

                ORG     0600H

                PUSH    ACC
                PUSH    B
                PUSH    PSW
                SETB    RS0                     ;BANK1
                CLR     RS1
                JNB     RI,TI_SESSION
;***RI SESSION***
; 2 CASES  1. SM2=1,RB8=1 : POLL
;          2. SM2=0       : RECEIVING DATA
                MOV     A,SBUF
                CLR     RI                      ;MY BUG
                MOV     B,A                     ;BACK UP
                JNB     SM2,CHK_DAT             ;SM2=0 => RECEIVE DATA
                CJNE    A,#ADDRESS,CHK_ADD      ;SM2=1 => CHECK ADDRESS
                CLR     SM2                     ;PREPARE
                MOV     RFCOUNT,#00H
                SJMP    RETURN
CHK_DAT:        JNB     RB8,REC_DAT
                SETB    SM2                     ;RECEIVE FINISH
                SETB    RFINISH
                SJMP    RETURN
REC_DAT:
                MOV     A,RFCOUNT
                ADD     A,#RFTABLE
                MOV     R0,A
                MOV     A,B                     ;RECOVER DATA
                MOV     @R0,A
                INC     RFCOUNT
                SJMP    RETURN
CHK_ADD:        CJNE    A,#11111111B,RETURN
                CLR     SM2                 ;PREPARE
                MOV     RFCOUNT,#00H
                SJMP    RETURN

;*** TI SESSION ***
TI_SESSION:     SJMP    RETURN

RETURN:
                POP     PSW
                POP     B
                POP     ACC
                RETI
;-------------------------------------------------------------------------
                END
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โปรแกรมจัดการและควบคุมระบบส่วนเซิร์ฟเวอร์โมดูล

(ไฟล์ S485.ASM)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
;COMPOSE OF IBM4.ASM AND INTFACE.ASM

              ORG   0000H
      BANK0:  DS    8
      BANK1:  DS    8
      BANK2:  DS    8
      BANK3:  DS    8
      BITADB: DS    3
      MAGBUF: DS    40
      MAGWRK: DS    40
      STACK:  DS    21
              ORG   MAGWRK
      RFCOUNT: DS   1                    ;** RFTABLE **
      RFTABLE: DS   7
      CONTACT: DS   1
      COUNT:  DS    1
              ORG   BANK2
      SFTABLE: DS   7                    ;** SFTABLE **
      DATA:   DS    1

                                         ;** IDENTITY **
      SERVER  EQU   10000001B
      IBMPC   EQU   11111111B            ;FOR RECEIVE
    BROADCAST EQU   11111111B            ;FOR SEND
                                         ;** PORT **
      CARDTD  EQU   P1.0                 ;DATA
      CARDTC  EQU   P1.1                 ;CLOCK
      CARDCP  EQU   P1.2                 ;PRESENT OF MC
      LED     EQU   P1.3
      SIGNAL  EQU   P1.4
      DRIVER  EQU   T0
                                         ;** BIT ADDRESS **
      CARDFF  EQU   00H                  ;CARD FIRST   FLAG
      CARDER  EQU   01H                  ;CARD ERROR   FLAG
      CARDRF  EQU   02H                  ;CARD REVERSE FLAG
      CONNECT EQU   03H                  ;STATUS CONNECTED
      BUSY    EQU   04H                  ;STATUS SERVER BUSY!
      REQ     EQU   05H                  ;ERROR FLAG
      RFINISH EQU   06H                  ;RECEIVE SUCCESS
      POLL    EQU   07H
      B7      EQU   08H                  ;FOR CRC
      TEMP    EQU   09H                  ;FOR CRC
                                         ;** RAM ADDRESS **
      RFSF    EQU   RFTABLE
      RFAF    EQU   RFTABLE+1
      RFCF    EQU   RFTABLE+2
      RFIF    EQU   RFTABLE+3
      RFCRC   EQU   RFTABLE+4
      RFXF    EQU   RFTABLE+6
      SFSF    EQU   SFTABLE
      SFAF    EQU   SFTABLE+1
      SFCF    EQU   SFTABLE+2
      SFIF    EQU   SFTABLE+3
      SFCRC   EQU   SFTABLE+4
      SFXF    EQU   SFTABLE+6
                                         ;** VALUE **
      BR96    EQU   0FDH
      BR48    EQU   0FAH
      BR24    EQU   0F4H
      BR12    EQU   0E8H
      DOUBLE  EQU   80H
      NORMAL  EQU   00H
                                         ;** PROTOCAL VALUE **
      DM      EQU   00001111B
      UA      EQU   01100011B
```

เอกสารนี้เป็นการที่ได้สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของผลงานทุกครั้งที่มีการนำไปใช้

```
BZ      EQU  11100011B
MTU     EQU  01001111B
SIM     EQU  10000011B
FREE    EQU  00011111B
PCBC    EQU  00011011B
SABM    EQU  00111111B
DISC    EQU  01000011B
NRGT    EQU  10111111B

                                    ;** FILE ADRRESS **
STATUS_TABLE  EQU  0028H
  MCRAM       EQU  0B00H
  SOPHOMORE   EQU  1000H
  JUNIOR      EQU  1960H
  SENIOR      EQU  22C0H
  MASTER      EQU  2C20H
  TEACHER     EQU  3260H


;-------------------------------------------------------------------------
            ORG  000H
            LJMP PREPARE

            ORG  0023H
            LJMP 0A00H

            ORG  0200H
;-------------------------------------------------------------------------
PREPARE:    CLR  LED
            MOV  R1,#02H
POWERUP1:   MOV  R2,#00H
POWERUP2:   MOV  R3,#00H
            DJNZ R3,$
            DJNZ R2,POWERUP2
            DJNZ R1,POWERUP1
            MOV  SP,#STACK
            SETB LED

            SETB P1.0               ;SET INPUT
            SETB P1.1
            SETB P1.2
            MOV  TL1,#BR24          ;2400 BAUD RATE
            MOV  TH1,#BR24
            MOV  SCON,#11110000B    ;SERIAL MODE 3 (9BIT UART)
            MOV  TMOD,#00100000B    ;TIMER 1 MODE 2(AUTORELOAD)
            MOV  PCON,#NORMAL       ;NORMAL BUAD RATE
            SETB EA
            SETB ES
            CLR  T0                 ;RECEIVE STAND-BY
            CLR  CONNECT            ;STATUS DISCONNECT
            CLR  BUSY
            CLR  REQ
            CLR  RFINISH
            MOV  SFSF,#01111110B
            MOV  SFXF,#01111110B
            MOV  TCON,#01000000B    ;TR1 ON
            MOV  COUNT,#00H
;-------------------------------------------------------------------------
INITIALIZE:
            SETB P1.3  ;FREE
            SETB P1.4
            SETB P1.5
            SETB P1.6
            SETB P1.7
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น ไม่อนุญาตให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                                                    ;CLEAR RAM 16K
                MOV     DPTR,#0
INITIALIZE1:    CLR     A
                MOVX    @DPTR,A
                INC     DPTR
                MOV     A,DPL
                CJNE    A,#0FFH,INITIALIZE1
                MOV     A,DPH
                CJNE    A,#3FH,INITIALIZE1

;-------------------------------------------------------------------------
                ;******MAIN******
;-------------------------------------------------------------------------

MAIN:
                                                    ;WAIT FRAME
                MOV     COUNT,#0
                JNB     RFINISH,$
                CLR     RFINISH
                MOV     A,RFCF                      ;CHECK COMMAND
                MOV     B,RFIF
COMMAND_TABLE:
                CJNE    A,#10000011B,CT2            ;ADD_NEW_USER(PC)
                ACALL   NEW_USER
                SJMP    MAIN
        CT2:    CJNE    A,#10101011B,CT3            ;DOWNLOAD_DATA(PC)
                ACALL   DOWNLOAD
                SJMP    MAIN
        CT3:    CJNE    A,#00011011B,CT4            ;PC_BROADCAST(PC)
                ACALL   BRCAST
                SJMP    MAIN
        CT4:    CJNE    A,#00001111B,CT5            ;USER_REQUEST(TM)=DM
                ACALL   EDIT
                SJMP    MAIN

        CT5:
                JMP     MAIN
;-------------------------------------------------------------------------
NEW_USER:       CALL    CARDCL
                SETB    P1.0                ;SET FOR INPUT
                SETB    P1.1
                SETB    P1.2

MAG_RD:
                JB      CARDCP,MAG_RD       ;WAIT FOR CP=0
                CALL    CARDRD
                CALL    CARDAJ
                JB      CARDER,MAG_RD
                CALL    COMMU
                JB      REQ,MAG_RD2
                CALL    CARDCL
                RET
MAG_RD2:
                CLR     REQ
                SJMP    MAG_RD
;-------------------------------------------------------------------------
;                SUBROUTINE
;-------------------------------------------------------------------------
;*** CARDCL SUB ***
; CLEAR MAGBUF

CARDCL:         MOV     R0,#MAGBUF
                MOV     R1,#40
```

เอกสารนี้สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น และห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
CARDCL1:        MOV     @R0,#0
                INC     R0
                DJNZ    R1,CARDCL1
                RET

; *** CARDRD SUB ***
; MAGNETIC CARD READ (RAW DATA)
; OUT = MAGBUF
; REG = A,R0,R1,R2

CARDRD:
                MOV     R0,#MAGBUF
                SETB    CARDFF              ;SET FISRT FLAG

CARDRD2:        MOV     R2,#5               ;5 BIT DATA
                CLR     A
CARDRD4:        JB      CARDCP,CARDRD8      ;CHECK CP=1 (END CARD)
                JB      CARDTC,CARDRD4      ;WAIT FOR CLOCK=0
                MOV     R1,#2               ;DELAY
                DJNZ    R1,$
                MOV     C,CARDTD            ;READ DATA
                JNB     CARDFF,CARDRD5
                JC      CARDRD4             ;CHECK FIRST=0 (IF BIT C=1
                                            ;IT INDICATES START BIT,
                                            ;NOTHING TO DO)
                CLR     CARDFF
CARDRD5:        RRC     A
                JNB     CARDTC,$            ;WAIT FOR CLOCK=1
                DJNZ    R2,CARDRD4

                RRC     A
                RRC     A
                RRC     A
                CPL     A
                ANL     A,#00011111B
                MOV     @R0,A
                INC     R0
                CJNE    R0,#MAGBUF+40,CARDRD2
CARDRD8:        RET

; *** CARDAJ SUB ***
; MAGNETIC CARD ADJUST DATA (CHECK REVERSE,
;CHECK PARITY,CHANGE TO ASCII)
; IN  = MAGBUF  RAW DATA
; OUT = MAGBUF  ASCII DATA (40 BYTE)
;       CARDER  0=OK 1=ERROR
; REG = A,R0,R1,R2,R3,R4

CARDAJ:         MOV     R0,#MAGBUF          ;BACKUP TO MAGWRK
                MOV     R1,#MAGWRK
                MOV     R2,#40
CARDAJ1:        MOV     A,@R0
                MOV     @R1,A
                INC     R0
                INC     R1
                DJNZ    R2,CARDAJ1

                CLR     CARDRF              ;FORWARD
                CALL    CARDCK
                JB      CARDER,CARDAJ8
                CALL    CARDAX
                JB      CARDER,CARDAJ8
                RET
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
CARDAJ8:        SETB    CARDRF          ;REVERSE
                CALL    CARDRE
                CALL    CARDCK
                CALL    CARDAX
                RET
; *** CARDAX SUB ***
; MAGNETIC CARD LAST CHECK (FIRST & LAST DATA)
; REG = A,R0

CARDAX:         CLR     CARDER          ;CHECK DATA (B)
                MOV     R0,#MAGBUF
                MOV     A,@R0
                CJNE    A,#'B',CARDAX5

                MOV     R0,#MAGBUF+39   ;CHECK DATA (F)
CARDAX1:        MOV     A,@R0
                CJNE    A,#0,CARDAX2
                DEC     R0
                CJNE    R0,#MAGBUF-1,CARDAX1
                SJMP    CARDAX5

CARDAX2:        DEC     R0
                MOV     A,@R0
                CJNE    A,#'F',CARDAX5
                RET                     ;EXIT OK

CARDAX5:        SETB    CARDER          ;EXIT DATA ERROR
                RET

; *** CARDRE SUB ***
; MAGNETIC CARD REVERSE CHECK
; IN  = MAGWRK  RAW DATA (REVERSE)
; OUT = MAGBUF  RAW DATA (FORWARD)
; REG = A,R0,R1,R2,R3,R4

CARDRE:         CALL    CARDCL
                MOV     R0,#MAGWRK+39   ;REVERSE DATA
                MOV     R1,#MAGBUF      ;FORWARD DATA
                MOV     R2,#5           ;MAGWRK BIT COUNT
                MOV     R3,#5           ;MAGBUF BIT COUNT
                MOV     R4,#0           ;WORK DATA
                CLR     CARDFF

CARDRE1:        MOV     A,@R0           ;SKIP ZERO BYTE
                CJNE    A,#0,CARDRE2
                DEC     R0
                CJNE    R0,#MAGWRK-1,CARDRE1
                RET                     ;ERROR (ALL MAGWRK ZERO)

CARDRE2:        MOV     A,@R0
                RLC     A
                RLC     A
                RLC     A
CARDRE3:        RLC     A               ;FIND FIRST BIT = 1
                JC      CARDRE5
                DJNZ    R2,CARDRE3
                RET                     ;ERROR (NOTHING)

CARDRE5:        PUSH    ACC             ;SHIFT BIT DATA *** LOOP ***
                MOV     A,R4
                RRC     A
                MOV     R4,A
                POP     ACC
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น ไม่อนุญาตให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                DJNZ    R2,CARDRE6          ;NEXT BYTE OF MAGWRK
                DEC     R0
                CJNE    R0,#MAGWRK-1,CARDRE51
                SETB    CARDFF
                CLR     A
                SJMP    CARDRE52
CARDRE51:       MOV     A,@R0
                RLC     A                   ;DON'T CARE 3 BIT LEFT
                RLC     A
                RLC     A
CARDRE52:       MOV     R2,#5

ARDRE6:         DJNZ    R3,CARDRE8          ;NEXT BYTE OF MAGBUF
                PUSH    ACC                 ;--\
                MOV     A,R4
                RRC     A
                RRC     A
                RRC     A
                ANL     A,#1FH
                MOV     @R1,A
                INC     R1
                POP     ACC                 ;--/
                JNB     CARDFF,CARDRE61
                RET                         ;EXIT OK
CARDRE61:       MOV     R3,#5

CARDRE8:        RLC     A                   ;NEXT BIT
                SJMP    CARDRE5

; *** CARDCK SUB ***
; MAGNETIC CARD CHECK PARITY & CONVERT TO ASCII
; IN  = MAGBUF  RAW DATA
; OUT = MAGBUF  ASCII DATA (40 BYTE)
;       CARDER  0=OK 1=ERROR (PARITY ERROR)
; REG = A,R0,R1

CARDCK:         CLR     CARDER              ;ERROR=0
                MOV     R0,#MAGBUF
                MOV     R1,#40
CARDCK2:        MOV     A,@R0
                CJNE    A,#0,CARDCK3
                RET                         ;EXIT NO ERROR (UNDER 40 BYTE

CARDCK3:        JNB     PSW.0,CARDCK8
                ANL     A,#0FH              ;CONVERT TO ASCII
                CJNE    A,#0AH,$+3
                JNC     CARDCK4
                ORL     A,#30H
                SJMP    CARDCK6
CARDCK4:        SUBB    A,#9
                ORL     A,#40H

CARDCK6:        MOV     @R0,A               ;NEXT
                INC     R0
                DJNZ    R1,CARDCK2
                RET                         ;EXIT NO ERROR

CARDCK8:        SETB    CARDER              ;ERROR=1
                RET
;----------------END CARD READ---------------------------------
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
; *** COMMU ***
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้
COMMU:
                CLR     LED                 ;BUSY!
```

```
                MOV     R1,#40
                MOV     R0,#MAGBUF
COMMU2:
                MOV     A,@R0
                CALL    DSEND
                INC     R0
                DJNZ    R1,COMMU2

                CLR     P1.4
                CLR     P1.5
                CLR     P1.6
                CLR     P1.7
                CALL    DELAYX
                MOV     P1,#11111111b
                SETB    LED                             ;FREE!
                RET
; ** DOWNLOAD **
DOWNLOAD:
                CLR     LED                             ;BUSY!
                MOV     CONTACT,#IBMPC
                MOV     SFSF,#7FH
                MOV     SFAF,#SERVER
                MOV     SFCF,#UA
                MOV     SFIF,#BZ
                MOV     SFCRC,SFIF
                MOV     SFCRC+1,SFIF
                MOV     SFXF,#7FH
                ACALL   SENDFULL

                CLR     RFINISH
                JNB     RFINISH,$
                CLR     RFINISH
                MOV     A,RFCF
                CJNE    A,#01010111B,DOWNLOADXX         ;START_FILE
                MOV     B,RFIF                          ;FILE
                MOV     A,B

                CJNE    A,#1,$+8
                MOV     DPTR,#SOPHOMORE
                SJMP    DOWNLOAD1
                DJNE    A,#2,$+8
                MOV     DPTR,#JUNIOR
                SJMP    DOWNLOAD1
                CJNE    A,#3,$+8
                MOV     DPTR,#SENIOR
                SJMP    DOWNLOAD1
                CJNE    A,#4,$+8
                MOV     DPTR,#MASTER
                SJMP    DOWNLOAD1
                CJNE    A,#5,DOWNLOADXX
                MOV     DPTR,#TEACHER
DOWNLOAD1:
                JNB     RFINISH,$
                CLR     RFINISH
                MOV     A,RFCF
                MOV     B,RFIF                          ;BACKUP DATA
                CJNE    A,#00000011B,DOWNLOAD2          ;INDIVIDUAL
                MOV     A,COUNT
                INC     A
                MOV     COUNT,A
                SWAP    A
                MOV     P1,A
                SJMP    DOWNLOAD1
DOWNLOADXX:
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น ห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                SJMP    DOWNLOADX
DOWNLOAD2:
                CJNE    A,#10010011B,DOWNLOAD3      ;END_OF_FILE
                SETB    LED                         ;FREE!
                JMP     DOWNLOADQ
DOWNLOAD3:
                MOV     A,B                         ;DATA
                MOVX    @DPTR,A                     ;STORED!
                INC     DPTR
                SJMP    DOWNLOAD1
DOWNLOADX:                                          ;ERROR
                MOV     COUNT,#0
                CLR     LED
                CALL    DELAYX
                SETB    LED
                CALL    DELAYX
                SJMP    DOWNLOADX
DOWNLOADQ:
                MOV     CONTACT,#IBMPC              ;CLEAR BUSY
                MOV     SFSF,#7FH
                MOV     SFAF,#SERVER
                MOV     SFCF,#UA
                MOV     SFIF,#FREE
                MOV     SFCRC,SFIF
                MOV     SFCRC+1,SFIF
                MOV     SFXF,#7FH
                ACALL   SENDFULL

                MOV     COUNT,#0
                MOV     p1,#11111111b
                RET
      ;--------- FOR CHECK DATA IN RAM --------
       F6:
                MOV     DPTR,#SOPHOMORE
                MOVX    A,@DPTR
                MOV     P1,A
                CALL    DELAYX
                SWAP    A
                MOV     P1,A
                CALL    DELAYX
                SJMP    F6
      ;-----------------------------------------
  *** BROADCAST ***
BRCAST:

                RET
  *** EDIT ***
EDIT:
                CLR     LED                 ;BUSY!

                MOV     A,RFAF              ;JUMP TO TM# RAM
                ANL     A,#00011111B        ;VALID LOW 5 BITS
                CLR     C
                RLC     A
                MOV     DPTR,#TERM_TAB
                JMP     @A+DPTR
      ;-------------------------------------------------
TERM_TAB:       SJMP    RGT_TERM            ;TERMINAL 1 : 0000000B
                SJMP    RGT_TERM            ;TERMINAL 2 : 0000001B
                SJMP    RGT_TERM            ;TERMINAL 3 : 0000010B
                SJMP    NOT_TERM            ;TERMINAL 4 : 0000011B
                SJMP    NOT_TERM            ;TERMINAL 5 : 0000100B
                SJMP    NOT_TERM            ;TERMINAL 6 : 0000101B
                SJMP    RGT_TERM            ;TERMINAL 7 : 0000110B
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น ไม่อนุญาตให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                SJMP    RGT_TERM        ;TERMINAL 8 : 0000111B
                SJMP    RGT_TERM        ;TERMINAL 9 : 0001000B
                SJMP    RGT_TERM        ;TERMINAL 10: 0001001B
                SJMP    RGT_TERM        ;TERMINAL 11: 0001010B
                SJMP    RGT_TERM        ;TERMINAL 12: 0001011B
                SJMP    RGT_TERM        ;TERMINAL 13: 0001100B
                SJMP    NOT_TERM        ;TERMINAL 14: 0001101B
                SJMP    RGT_TERM        ;TERMINAL 15: 0001110B
                SJMP    RGT_TERM        ;TERMINAL 16: 0001111B
                SJMP    RGT_TERM        ;TERMINAL 17: 0010000B
                SJMP    RGT_TERM        ;TERMINAL 18: 0010001B
                SJMP    RGT_TERM        ;TERMINAL 19: 0010010B
                SJMP    RGT_TERM        ;TERMINAL 20: 0010011B
                SJMP    RGT_TERM        ;TERMINAL 21: 0010100B
                SJMP    NOT_TERM        ;TERMINAL 22: 0010101B
                SJMP    RGT_TERM        ;TERMINAL 23: 0010110B
                SJMP    RGT_TERM        ;TERMINAL 24: 0010111B
                SJMP    RGT_TERM        ;TERMINAL 25: 0011000B
                SJMP    RGT_TERM        ;TERMINAL 26: 0011001B
                SJMP    RGT_TERM        ;TERMINAL 27: 0011010B
                SJMP    NOT_TERM        ;TERMINAL 28: 0011011B
                SJMP    RGT_TERM        ;TERMINAL 29: 0011100B
                SJMP    RGT_TERM        ;TERMINAL 30: 0011101B
                SJMP    RGT_TERM        ;TERMINAL 31: 0011110B
                SJMP    RGT_TERM        ;TERMINAL 32: 0011111B
        ;-----------------------------------------------
RGT_TERM:       SJMP    EDIT1
NOT_TERM:       JMP     EDITX           ;INVALID TERMINAL
EDIT1:                                  ;SEND UA
                MOV     DPTR,#0000H
                                        ;DPTR POINT TO THE 1ST DATA
                MOV     CONTACT,#BROADCAST
                MOV     A,CONTACT
                MOV     SFSF,#7FH
                MOV     SFAF,RFAF
                MOV     SFCF,#UA
                MOV     SFIF,#BZ        ;BUSY! BUSY! BUSY!
                MOV     SFCRC,SFIF
                MOV     SFCRC+1,SFIF
                MOV     SFXF,#7FH
                ACALL   SEND

                MOV     P1,#10010000B
        ;-----------------------------------------------
                CLR     RFINISH
                JNB     RFINISH,$       ;DUMB FRAME!!!
                CLR     RFINISH
        ;-----------------------------------------------

                MOV     R0,#40
EDIT2:                                  ;WAIT 40 DATA
                JNB     RFINISH,$
                CLR     RFINISH
                MOV     A,RFIF
                MOVX    @DPTR,A
                INC     DPTR
                DJNZ    R0,EDIT2

                MOV     P1,#01010000B
                CALL    DELAYX
                                        ;DEFINE TM# STATUS
                MOV     DPTR,#STATUS_TABLE  ;FF = ON
                MOV     A,RFAF              ;00 = OFF
                MOV     B,#0
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น ห้ามมิให้ทำซ้ำ ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                CALL   ADD_DPTR
                MOVX   A,@DPTR
                CJNE   A,#0FFH,EDIT3          ;OFF
                CALL   CASE_ON                ;ON  -> CASE 1.MULTI
                                              ;       CASE 2.LOGOUT
                CJNE   A,#0FFH,EDIT_C1
                MOV    DPTR,#STATUS_TABLE     ;CLEAR STATUS
                MOV    A,RFAF
                MOV    B,#0
                CALL   ADD_DPTR
                MOV    A,#0
                MOVX   @DPTR,A
                                              ;CLEAR MCRAM BUFFER
                MOV    DPTR,#MCRAM
                MOV    A,RFAF
                MOV    B,#28
                MUL    AB
                CALL   ADD_DPTR
                CLR    A

                MOV    R0,#40
EDIT_CLR:       MOVX   @DPTR,A
                INC    DPTR
                DJNZ   R0,EDIT_CLR

                MOV    CONTACT,BROADCAST      ;SEND DISC
                MOV    SFSF,#7FH
                MOV    SFAF,RFAF
                MOV    SFCF,#DISC
                MOV    SFIF,#FREE
                MOV    SFCRC,SFIF
                MOV    SFCRC+1,SFIF
                MOV    SFXF,#7FH
                ACALL  SENDFULL

                MOV    P1,#10010000B
                CALL   DELAYX

                JMP    EDITX                  ;WITH LOGOUT

EDIT_C1:                                      ;MULTIUSER CASE
                MOV    CONTACT,BROADCAST      ;SEND MTU
                MOV    SFSF,#7FH
                MOV    SFAF,RFAF
                MOV    SFCF,#MTU
                MOV    SFIF,#FREE
                MOV    SFCRC,SFIF
                MOV    SFCRC+1,SFIF
                MOV    SFXF,#7FH
                ACALL  SENDFULL

                MOV    P1,#01100000B
                CALL   DELAYX
                JMP    EDITX                  ;WITH MULTIUSER

EDIT3:                                        ;OFF

                CALL   SEARCH
                JZ     EDIT5                  ;EXIT [UNREGISTERED]

                CALL   FIND                   ;FOUND! [SABM,SIM]
                JNZ    EDIT6                  ;CHECK SIMUL. OR RIGHT USER
                MOV    DPTR,#MCRAM
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น ไม่อนุญาตให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                MOV     A,#40
                MOV     B,RFAF
                MUL     AB                      ;B-HI, A-LO
                CALL    ADD_DPTR

                MOV     R1,#40
                MOV     R0,#0

EDIT4:          MOV     P2,#0
                MOVX    A,@R0                   ;WRITE DATA TO MCRAM
                MOVX    @DPTR,A
                INC     R0
                INC     DPTR
                DJNZ    R1,EDIT4

                MOV     CONTACT,#BROADCAST      ;SEND SABM
                MOV     SFSF,#7FH
                MOV     SFAF,RFAF
                MOV     SFCF,#SABM
                MOV     SFIF,#FREE
                MOV     SFCRC,SFIF
                MOV     SFCRC+1,SFIF
                MOV     SFXF,#7FH
                CALL    SENDFULL
                                                ;SET TM#STATUS
                MOV     DPTR,#STATUS_TABLE      ;FF = ON
                MOV     A,RFAF                  ;00 = OFF
                MOV     B,#0
                CALL    ADD_DPTR
                MOV     A,#0FFH
                MOVX    @DPTR,A
                MOV     P1,#01010000B
                CALL    DELAYX
                SJMP    EDITX
EDIT5:                                          ;TM# OFF AND UNAUTHORIZED
                MOV     P1,#00110000B
                CALL    DELAYX
                MOV     CONTACT,BROADCAST       ;SEND NRGT
                MOV     SFSF,#7FH
                MOV     SFAF,RFAF
                MOV     SFCF,#NRGT
                MOV     SFIF,#FREE
                MOV     SFCRC,SFIF
                MOV     SFCRC+1,SFIF
                MOV     SFXF,#7FH
                CALL    SENDFULL
                SJMP    EDITX
EDIT6:                                          ;USER LOGIN MANY TMS
                MOV     P1,#10000000B
                CALL    DELAYX
                MOV     CONTACT,BROADCAST       ;SEND SIM
                MOV     SFSF,#7FH
                MOV     SFAF,RFAF
                MOV     SFCF,#SIM
                MOV     SFIF,#FREE
                MOV     SFCRC,SFIF
                MOV     SFCRC+1,SFIF
                MOV     SFXF,#7FH
                CALL    SENDFULL
                SJMP    EDITX
EDITX:          CLR     RFINISH                 ;MAKE SURE!
                MOV     P1,#11111111B
                SETB    LED                     ;FREE!
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น ห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                RET

; *** CASE ON ***
CASE_ON:
                MOV     DPTR,#MCRAM
                MOV     A,RFAF
                MOV     B,#40
                MUL     AB
                CALL    ADD_DPTR

                MOV     R4,#40
                MOV     R0,#0
CASE_ON_1:                                      ;COMPARE
                MOV     P2,#0
                MOVX    A,@R0
                MOV     B,A
                MOVX    A,@DPTR
                CJNE    A,B,CASE_ON_2
                INC     R0
                INC     DPTR
                DJNZ    R4,CASE_ON_1
                MOV     A,#0FFH                 ;EXIT [FOUND=LOGOUT]
                MOV     P1,#10100000B
                CALL    DELAYX
                SJMP    CASE_ON_X
CASE_ON_2:
                MOV     A,#00H                  ;EXIT [FAILED=MULTI]
                SJMP    CASE_ON_X
CASE_ON_X:
                RET

; *** SEARCH ***
;IN        = KEY
;DATABASE  = SOPHOMORE,JUNIOR,SENIOR,MASTER,TEACHER
;OUT       = A   ; 0FFH = FOUND, 00H = FAILED.

SEARCH:
                MOV     DPTR,#SOPHOMORE
                CALL    ENGINE
                JNZ     SEARCHX
                MOV     DPTR,#JUNIOR
                CALL    ENGINE
                JNZ     SEARCHX
                MOV     DPTR,#SENIOR
                CALL    ENGINE
                JNZ     SEARCHX
                MOV     DPTR,#MASTER
                CALL    ENGINE
                JNZ     SEARCHX
                MOV     DPTR,#TEACHER
                CALL    ENGINE
                JNZ     SEARCHX
SEARCHX:
                RET
; *** FIND ***
FIND:
                MOV     DPTR,#MCRAM
                CALL    ENGINE
                RET

; *** ENGINE ***
ENGINE:
                MOVX    A,@DPTR
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                JZ      ENGINEz         ;EXIT [END OF DATA]
ENGINE1:
                MOV     R4,#40
                MOV     R0,#0
ENGINE2:                                ;COMPARE
                MOV     P2,#0
                MOVX    A,@R0
                MOV     B,A
                MOVX    A,@DPTR
                CJNE    A,B,ENGINE3
                INC     R0
                INC     DPTR
                DJNZ    R4,ENGINE2
                MOV     A,#0FFH         ;EXIT [FOUND!]
                MOV     P1,#10010000B
                CALL    DELAYX
                SJMP    ENGINEX

ENGINE3:                                ;NEXT
                MOV     A,R4
                MOV     B,#0
                CALL    ADD_DPTR
                SJMP    ENGINE

ENGINEZ:        MOV     A,#0
ENGINEX:
                RET

; *** ADD DPTR ***
;IN  = A(LO),B(HI)
;OUT = DPTR
ADD_DPTR:
                CLR     C
                ADD     A,DPL
                MOV     DPL,A
                MOV     A,DPH
                JNC     ADD_DPTR1
                INC     A
ADD_DPTR1:      ADD     A,B
                MOV     DPH,A
                RET

; *** DSEND ***
;REG = A

DSEND:          SETB T0
                CLR  TI
                CLR  TB8
                MOV  SBUF,A
                JNB  TI,$
                CLR  TI
                CALL DELAY
                CLR  T0
                RET
; *** XSEND ***
;REG = A

XSEND:          SETB T0
                CLR  TI
                SETB TB8
                MOV  SBUF,A
                JNB  TI,$
                CLR  TI
                CALL DELAY
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น ห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                CLR  T0
                RET

 *** SEND ***
;REG = R1,R2,A,CONTACT

SEND:
                SETB T0                 ;ENABLE DRIVER
                SETB TB8                ;SEND ACTIVATE ADDR
                MOV  A,CONTACT
                MOV  SBUF,A
                JNB  TI,$
                CLR  TI
                MOV  SBUF,A

                CLR  TB8                ;SEND DATA
                MOV  R2,#7              ;QUEUE FOR FRAME
                MOV  R1,#SFTABLE
    SENDNEXT:   JNB  TI,SENDNEXT
                CLR  TI
                MOV  A,@R1
                MOV  SBUF,A
                INC  R1
                DJNZ R2,SENDNEXT
                JNB  TI,$
                CLR  TI
                SETB TB8
                MOV  A,CONTACT          ;ADDRESS FINISH
                MOV  SBUF,A
                JNB  TI,$
                CLR  TI
                CALL DELAY
                CLR  TB8
                CLR  T0                 ;ENABLE RECEIVER (STAND BY)
                CLR  TI
                RET

 *** SENDFULL ***
;REG = R1,R2,A,CONTACT

SENDFULL:
                SETB T0                 ;ENABLE DRIVER
                SETB TB8                ;SEND ACTIVATE ADDR
                MOV  A,CONTACT
                MOV  SBUF,A

                CLR  TB8                ;SEND DATA
                MOV  R2,#7              ;QUEUE FOR FRAME
                MOV  R1,#SFTABLE
 SENDNEXTFULL:  JNB  TI,SENDNEXTFULL
                CLR  TI
                MOV  A,@R1
                MOV  SBUF,A
                INC  R1
                DJNZ R2,SENDNEXTfull
                JNB  TI,$
                CLR  TI
                SETB TB8
                MOV  A,CONTACT          ;ADDRESS FINISH
                MOV  SBUF,A
                JNB  TI,$
                CLR  TI
                CALL DELAY
                CLR  TB8
                CLR  T0                 ;ENABLE RECEIVER (STAND BY)
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น ไม่อนุญาตให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                CLR  TI
                RET

;*** DELAY ***
DELAY:
                MOV  R3,#0FFH
                DJNZ R3,$
                MOV  R3,#0FFH
                DJNZ R3,$
                MOV  R3,#0FFH
                DJNZ R3,$
                MOV  R3,#0FFH
                DJNZ R3,$
                RET
;*** DELAY LONG ***
;REG = R3
DELAYX:
                MOV  R2,#0FFH
DELAYX1:        MOV  R3,#0FFH
                DJNZ R3,$
                MOV  R3,#0FFH
                DJNZ R3,$
                MOV  R3,#0FFH
                DJNZ R3,$
                MOV  R3,#0FFH
                DJNZ R3,$
                DJNZ R2,DELAYX1
                RET
;-------------------------------------------------------------------------
;         INTERUPT SERVICE ROUTINE : SI VECTOR
;-------------------------------------------------------------------------

                ORG    0A00H
                PUSH   ACC
                PUSH   B
                PUSH   PSW
                SETB   RS0                      ;BANK1
                CLR    RS1
                JNB    RI,TI_SESSION
;***RI SESSION***
; 2 CASES  1. SM2=1,RB8=1 : POLL
;          2. SM2=0       : RECEIVING DATA
                MOV    A,SBUF
                CLR    RI                       ;MY BUG
                MOV    B,A                      ;BACK UP
                JNB    SM2,CHK_DAT              ;SM2=0 => RECEIVE DATA
                CJNE   A,#SERVER,CHK_ADD        ;SM2=1 => CHECK ADDRESS
                CLR    SM2                      ;PREPARE
                MOV    RFCOUNT,#00H
                SJMP   RETURN
  CHK_DAT:      JNB    RB8,REC_DAT
                SETB   SM2                      ;RECEIVE FINISH
                SETB   RFINISH
                SJMP   RETURN
  REC_DAT:
                MOV    A,RFCOUNT
                ADD    A,#RFTABLE
                MOV    R0,A
                MOV    A,B                      ;RECOVER DATA
                MOV    @R0,A
                INC    RFCOUNT
                SJMP   RETURN
  CHK_ADD:      CJNE   A,#11111111B,RETURN
                CLR    SM2                      ;PREPARE
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น ไม่อนุญาตให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                MOV     RFCOUNT,#00H
                SJMP    RETURN

;*** TI SESSION ***
 TI_SESSION:    SJMP    RETURN

 RETURN:
                POP     PSW
                POP     B
                POP     ACC
                RETI
;-------------------------------------------------------------------------

                END
```


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โปรแกรมจัดการและควบคุมระบบส่วนไมโครคอมพิวเตอร์

โดยใช้ภาษาปาสคาล

(ไฟล์ ACCESS.PAS)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
Program ACCESS_CONTROL;
uses crt,dos,graph,
   grdriver,grfont1,grfont2,grfont3,grfont4,
   SoundFX,V24,AccessGP,MouseGP,CommuV24,DumbHDLC,KBThai;
label sub1a,sub1b,sub2b,sub2x,sub4b,sub4x;
const
    VersionName   ='Access.exe';
    VersionVersion ='1.0';
    VersionFor    ='Access Control';
    VersionDate   ='21 March 1998';
    VersionC1     ='Jurachart Jongusuk';
    VersionC2     ='Roengrut  Rujanakraikan';
    VersionAdv    ='Kiatiwan  Songsat';

    IBMPC        = 255; {FFH}
    Activate     = 171; {10101011B}
    StartFile    = 87 ; {01010111B}
    EndFile      = 147; {10010011B}
    DataByte     = 0;   {00H}
    Individual   = 3;   {00000011B}
    Add_New_User  = 131; {10000011B}
    Brcast       = 27;  {00011011B}
    DM           = 15;  {00001111B}

type
  Person = record
             FirstName: string[15];
             LastName : string[25];
             Address  : string[35];
           end;
  PersonFile = file of Person;
var
    GraphDriver,GraphMode : Integer;
  butt,row,col,lin,position : Integer;
                  regs : Registers;
                  f,f1 : text;
                  data : byte;
   logout,special,pass,o_x : boolean;
   not_search,clear,cancel : boolean;
      mouseCLICK,Install : boolean;
              ch1,ch : char;
        PersonSur,st,key : string;
    PersonCode,PersonName : string;
          Frame,RFrame : ProtoFrame;
           P,Int1CSave : Pointer;
                size : word;
  {sophom.txt, junior.txt, senior.txt, master.txt, teacher.txt}
{$F+,S-,W-}
procedure TimerHandler; interrupt;
 begin
  { Timer ISR }
  {***butt: no=0 left=1 right=2 simul=3***}
  MouseInfo(butt,col,row);
  if (butt<>1)or(keypressed) then begin
    if keypressed then ch:=readkey;
    row:=0; col:=0;
    mouseCLICK:=false
  end else begin
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
            mouseCLICK:=true;
          end;
 end;
{$F-,S+}
function Area(a,b,c,d:integer):boolean;
begin
  if  (col>a)and(col<c)and(row>b)and(row<d) then
      Area:=true
  else Area:=false
end;

procedure MainMenu;
var  ver    : word;
     version,versionLo,versionHi : string;
begin
     Beep;
     hidemouse;
     ver:=Dosversion;
     str(Lo(ver),versionLo);
     str(Hi(ver),versionHi);
     version:=versionLo+'.'+versionHi;
     setcolor(15);
     Subwin(75,390,200,20,0,0,15);
     setcolor(9);
     outtextXY(85,396,'Running on DOS ver '+version);
     setcolor(11);
     outtextXY(84,397,'Running on DOS ver '+version);
     setfillstyle(1,0);
     bar(386,455,570,470);
     setcolor(4);
     outtextxy(391,461,'Main Menu');
     setcolor(12);
     outtextxy(390,462,'Main Menu');setcolor(15);
     WindowShow(50,80,250,290,9,0,15);
     SubWin(128,85,93,20,13,8,15);
     if install then begin
       press(287,83,297,93);
       setcolor(0); outtextxy(289,85,'X');
     end;
     setcolor(3);
     outtextxy(140,91,'Main Menu');
     setcolor(11);
     outtextxy(139,92,'Main Menu');
     setcolor(15);
     DrawStar(35,15,14,1);
     DrawStar(35,55,14,1);
     DrawStar(35,95,14,1);
     DrawStar(35,135,14,1);
     DrawStar(35,175,14,1);
     DrawStar(35,215,14,1);
     setcolor(15);
     SetTextStyle(0,0,1);
     OutTextXY(120,120,'New User');
     OutTextXY(120,160,'Download Data');
     OutTextXY(120,200,'Broadcast');
     OutTextXY(120,240,'Delete Card');
     OutTextXY(120,280,'Version');
     OutTextXY(120,320,'Upload Byte');
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    settextstyle(0,0,1);
    showmouse;
end;
procedure TestKey(var ch:char;var special:boolean);
begin
    special:=false;
    ch:=readkey;
    if ch=#0 then
    begin
      ch:=readkey;
      special:=true;
    end;
end;
procedure o_x_test(ch:char;var o_x:boolean);
begin
   case ch of
     #72,#75:begin
             setcolor(15);
             outtextxy(194,331,'ancel');
             setcolor(14);
             outtextxy(134,331,'.K.');
             o_x:=true;
            end;
     #80,#77:begin
             setcolor(15);
             outtextxy(134,331,'.K.');
             setcolor(14);
             outtextxy(194,331,'ancel');
             o_x:=false;
            end
   else end
end;

procedure Exit_Win1(exit_what:string;var o_x:boolean;var pass:boolean);
var P    : pointer;
   size : word;
begin
   {***** Algorithm for create 'exit' window *****}
   hidemouse;
   setcolor(14);
   subwin(100,290,150,70,11,0,15);
   subwin(130,294,92,15,15,0,15);
   settextstyle(2,0,4);
   setcolor(4);
   outtextxy(134,295,'Exit ');
   setcolor(1);
   outtextxy(162,295,exit_what);
   setcolor(15);
   settextstyle(0,0,1);
   Press(115,328,165,340);
   Press(184,328,235,340);
   setcolor(12);
   outtextxy(126,331,'O');
   setcolor(15);
   outtextxy(134,331,'.K.');
   setcolor(12);
   outtextxy(186,331,'C');
   setcolor(15);
```

เอกสารนี้จัดทำขึ้นเพื่อการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    outtextxy(194,331,'ancel');
    showmouse;
  {**********  Algorithm for Exit Window  **************}
        pass :=false;
        o_x  :=false;
        setcolor(14);
        outtextxy(194,331,'ancel');
        repeat
         if not(install) then
           Testkey(ch,special);
         if (ch=#13)and(special=false) then
             pass:=true
         else if (ch='o')or(ch='O')or(area(115,328,165,340)) then begin
                 pass :=true;
                 o_x  :=true;
            end
            else if (ch='c')or(ch='C')or(area(184,328,235,340)) then
               begin
                    pass :=true;
                    o_x  :=false;
               end else begin
                      o_x_Test(ch,o_x);
                    end;
        until (pass);

        if o_x = true then begin
          HideMouse;
          setcolor(15);
          outtextxy(194,331,'ancel');
          AfterPress(115,328,165,340);
          setcolor(0);
          outtextxy(127,331,'O.K.');
          setcolor(15);
          outtextxy(126,332,'O.K.');
          delay(100);
          ShowMouse;

        end
        else begin
           HideMouse;
           setcolor(15);
           outtextxy(134,331,'.K.');
           AfterPress(184,328,235,340);
           setcolor(0);
           outtextxy(187,331,'Cancel');
           setcolor(15);
           outtextxy(186,332,'Cancel');
           delay(100);
           ShowMouse;
           end;
        MainMenu;
        position:=1;
        Hilight(120,120,'New User');
end;

procedure ChooseMain(var Position:integer);
var  pass:boolean;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
procedure rehilightALL;
begin
        settextstyle(0,0,1);
        ReHilight(120,120,'New User');
        ReHilight(120,160,'Download Data');
        ReHilight(120,200,'Broadcast');
        ReHilight(120,240,'Delete Card');
        ReHilight(120,280,'Version');
        ReHilight(120,320,'Upload Byte');
        settextstyle(0,0,1);
end;
begin
  pass:=false;
  repeat
   if (not(install))or(keypressed) then begin
     testkey(ch,special);
     ch1:=ch;
    if (ch1=#13) and (special=false) then    {Confirm Choice}
      pass:=true;
    if (ch1=#27) and (special=false) then
      begin
        Exit_Win1(' Program',o_x,pass);
        if o_x=true then begin
          pass:=true;
          position:=10
        end
        else begin
             pass:=false;
             {Position:=11;
             pass:=true;}
            end;
      end
    else
    if ((ch1=#72)or(ch1=#75)or(ch1=#52)or(ch1=#56)) then begin   {Go Up Case}
      if (position=1) then begin
        hilight(120,320,'Upload Byte');
        ReHilight(120,120,'New User');
        position:=6
      end
      else
      if position=2 then begin
        hilight(120,120,'New User');
        ReHilight(120,160,'Download Data');
        position:=1
      end
      else
      if position=3 then begin
        hilight(120,160,'Download Data');
        ReHilight(120,200,'Broadcast');
        position:=2
      end
      else
      if position=4 then begin
        hilight(120,200,'Broadcast');
        ReHilight(120,240,'Delete Card');
        position:=3
      end
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
      else
      if position=5 then begin
        hilight(120,240,'Delete Card');
        ReHilight(120,280,'Version');
        position:=4
      end
      else
      if position=6 then begin
        hilight(120,280,'Version');
        ReHilight(120,320,'Upload Byte');
        position:=5
      end
    end;

    if ((ch1=#80)or(ch1=#77)or(ch1=#54)or(ch1=#50)) then begin     {Go Down Case}
      if position=1 then begin
        hilight(120,160,'Download Data');
        ReHilight(120,120,'New User');
        position:=2;
      end
      else
      if position=2 then begin
        hilight(120,200,'Broadcast');
        ReHilight(120,160,'Download Data');
        position:=3;
      end
      else
      if position=3 then begin
        hilight(120,240,'Delete Card');
        ReHilight(120,200,'Broadcast');
        position:=4;
      end
      else
      if position=4 then begin
        hilight(120,280,'Version');
        ReHilight(120,240,'Delete Card');
        position:=5;
      end
      else
      if position=5 then begin
        hilight(120,320,'Upload Byte');
        ReHilight(120,280,'Version');
        position:=6;
      end
      else
      if position=6 then begin
        hilight(120,120,'New User');
        ReHilight(120,320,'Upload Byte');
        position:=1;
      end;
    end;
  end else begin
        repeat until (mouseCLICK)or(keypressed);
        if area(287,83,297,93) then
        begin
          hidemouse;
          settextstyle(0,0,1);
          afterpress(287,83,297,93);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
            setcolor(12); outtextxy(289,85,'X');
            delay(100);
            press(287,83,297,93);
            setcolor(0); outtextxy(289,85,'X');
            position:=10;
            pass:=true;
            settextstyle(7,0,1);
            showmouse;
          end else
          if (area(120,115,280,135)) then
          begin
            position:=1;
            rehilightALL;
            hilight(120,120,'New User');
            position:=1;
            pass:=true;
          end else
          if area(120,155,280,175) then
          begin
            rehilightALL;
            hilight(120,160,'Download Data');
            position:=2;
            pass:=true;
          end else
          if area(120,195,280,215) then
          begin
            rehilightALL;
            hilight(120,200,'Broadcast');
            position:=3;
            pass:=true;
          end else
          if area(120,235,280,255) then
          begin
            rehilightALL;
            hilight(120,240,'Delete Card');
            position:=4;
            pass:=true;
          end else
          if area(120,275,280,295) then
          begin
            rehilightALL;
            hilight(120,280,'Version');
            position:=5;
            pass:=true;
          end else
          if area(120,315,280,335) then
          begin
            rehilightALL;
            hilight(120,320,'Upload Byte');
            position:=6;
            pass:=true;
          end else pass:=false;
          end;
    until pass;
    settextstyle(0,0,1);
  end;
  procedure Optimize;
   procedure Optim(t:string);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
  var  r:string;
  begin
     assign(f,t);
     assign(f1,'temp.txt');
     reset(f);
     rewrite(f1);
     repeat
      readln(f,r);
      if (r[length(r)] <> '!')and(r[length(r)-1]<>'!') then
        writeln(f1,r);
     until eof(f);
     rewrite(f);
     reset(f1);
     repeat
      readln(f1,r);
      writeln(f,r);
     until eof(f1);
     close(f1);
     close(f);
  end;
begin
     Optim('Sophom.txt');
     Optim('Junior.txt');
     Optim('Senior.txt');
     Optim('Master.txt');
     Optim('Teacher.txt');
end;
procedure SubMenu1;
begin
     Beep;
     hidemouse;
     setfillstyle(1,0);
     bar(386,455,570,470);
     setcolor(4);
     outtextxy(391,461,'New User');
     setcolor(12);
     outtextxy(390,462,'New User');
     setcolor(15);
     Subwin(350,80,200,190,9,0,15);   {new user}
     Subwin(398,86,111,20,13,0,15);   {title new user}
     if (install) then begin
       press(534,85,544,95);
       setcolor(0);outtextxy(536,87,'X');
     end;
     setfillstyle(1,8);
     Bar(355,240,545,265);          {choose file}
     setcolor(15);
     rectangle(358,243,542,262);
     setcolor(9);
     outtextxy(408,92,'Add New User');
     setcolor(11);
     outtextxy(407,93,'Add New User');setcolor(15);
     Subwin(350,290,200,80,4,0,15);    {result}
     Subwin(421,296,56,12,15,0,15);    {title result}
     setcolor(7);
     outtextxy(427,298,'Result');
     setcolor(0);
     outtextxy(426,299,'Result');setcolor(15);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    setfillstyle(3,15);
    bar(370,315,530,360);          {result show}
    showmouse;
end;
procedure SubMenu1a;
begin
    hidemouse;
    settextstyle(0,0,1);
    outtextxy(400,117,'Sophomore');
    press(370,115,380,125);
    outtextxy(400,142,'Junior');
    press(370,140,380,150);
    outtextxy(400,167,'Senior');
    press(370,165,380,175);
    outtextxy(400,192,'Master');
    press(370,190,380,200);
    outtextxy(400,217,'Teacher');
    press(370,215,380,225);
    setcolor(0);
    outtextxy(372,117,'1');
    outtextxy(372,142,'2');
    outtextxy(372,167,'3');
    outtextxy(372,192,'4');
    outtextxy(372,217,'5');
    setfillstyle(1,0);
    bar(504,246,537,259);          {choice}
    settextstyle(0,0,1);
    setcolor(6);
    outtextxy(366,250,'Choose File :');
    setcolor(14);
    outtextxy(365,249,'Choose File :');setcolor(15);
    showmouse;
end;
procedure DoSub1a(var st:string);
var  mouse1a:boolean;
 procedure Result1a(c:char;s:string);
 begin
    hidemouse;
    setfillstyle(1,0);
    bar(504,246,537,259);          {choice}
    bar(370,315,530,360);          {result show}
    setcolor(15);
    outtextxy(519,250,c);
    outtextxy(405,345,s);
    setcolor(7);
    outtextxy(380,325,'You choose file :');
    showmouse;
 end;
 procedure press1a(a,b,c,d:integer;h:char;j:string);
 begin
      hidemouse;
      afterpress(a,b,c,d);
      setcolor(12);
      outtextxy(400,b+2,j);
      outtextxy(a+2,b+2,h);
      delay(100);
      press(a,b,c,d);
      setcolor(0);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
      outtextxy(a+2,b+2,h);
      setcolor(15);
      outtextxy(400,b+2,j);
      showmouse;
      ch1:=h;
      mouse1a:=true;
    end;
  begin
    cancel:=false;
    mouse1a:=false;
    st:='No Plan';
    SubMenu1;
    Submenu1a;
    repeat
    ch1:=#0;
    ch:=ch1;
    if not(install) then begin
      CursorNoMouse(519,258);
      TestKey(ch,special);
      ch1:=ch;
    end
    else begin ch1:=#0;ch:=ch1;
          repeat until (mouseCLICK)or(ch<>ch1);
          ch1:=ch;
       end;
    if area(370,115,380,125) then
      press1a(370,115,380,125,'1','Sophomore')
    else
    if area(370,140,380,150) then
      press1a(370,140,380,150,'2','Junior')
    else
    if area(370,165,380,175) then
      press1a(370,165,380,175,'3','Senior')
    else
    if area(370,190,380,200) then
      press1a(370,190,380,200,'4','Master')
    else
    if area(370,215,380,225) then
      press1a(370,215,380,225,'5','Teacher')
    else
    if area(534,85,544,95) then begin
      afterpress(534,85,544,95);
      setcolor(12);outtextxy(536,87,'X');
      delay(100);
      press(534,85,544,95);
      setcolor(0);outtextxy(536,87,'X');
      setcolor(15);
      mouse1a:=true;
      ch1:=#27;
    end;

    case ch1 of
    '1':begin st:='Sophom.txt';
          Result1a('1',st);
       end;
    '2':begin st:='Junior.txt';
          Result1a('2',st);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        end;
      '3':begin st:='Senior.txt';
            Result1a('3',st);
        end;
      '4':begin st:='Master.txt';
            Result1a('4',st);
        end;
      '5':begin st:='Teacher.txt';
            Result1a('5',st);
        end;
      #13:begin if (st='No Plan')or(st='') then
            ch1:=#0;
        end;
      #27:begin Cancel:=true;
            st:='';
        end
      else
        begin Result1a(' ','  None  ');
            st:=''
        end;
      end;
      until (ch1=#13)or(cancel=true)or(mouse1a);
      if mouse1a then delay(200);
      assign(f,st);
  end;
  procedure SubMenu1b;
  begin
    SubMenu1;
    hidemouse;
    setcolor(6);
    outtextxy(381,126,'Name :');
    outtextxy(357,146,'Surname :');
    outtextxy(373,166,'Login :');
    setcolor(14);
    outtextxy(380,127,'Name :');
    outtextxy(356,147,'Surname :');
    outtextxy(372,167,'Login :');
    setfillstyle(1,0);
    bar(430,123,540,138);
    bar(430,143,540,158);
    bar(430,163,540,178);
    press(376,205,408,220);
    press(434,205,466,220);
    press(492,205,524,220);
    outtextxy(385,209,'OK');
    outtextxy(439,209,'Clr');
    outtextxy(505,209,'X');
    setcolor(0);
    outtextxy(374,227,'Enter');
    outtextxy(443,227,'F1');
    outtextxy(498,227,'Esc');
    setcolor(6);
    outtextxy(366,250,'Add to :');
    setcolor(14);
    outtextxy(365,249,'Add to :');setcolor(15);
    showmouse;
  end;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
procedure Rec_Text1(x,y,L:integer;var Text:string);
 var i,n,bet,long : integer;
      RecArray : array[1..20] of char;
         acc : boolean;
 begin
  for i:=1 to L do
    RecArray[i]:=' ';
  long:=0; Text:=''; bet:=0; acc:=false;
  Not_Search:=false; clear:=false; i:=0;
  repeat
   if not(install) then begin
     CursorNoMouse(x+bet,y);
     Testkey(ch,special);
     ch1:=ch
   end else begin
            CursorStop(x+bet,y);
            repeat until (keypressed or mouseCLICK);
            if  keypressed then  Testkey(ch,special)
            else begin delay(50);
               if area(534,85,544,95) then begin
                 text:='EXIT'; ch:=#0;
               end else
               if area(376,205,408,220) then begin
                 ch:=#13; special:=false
               end else
               if area(434,205,466,220) then begin
                 ch:=#59; special:=true
               end else
               if area(492,205,524,220) then begin
                 ch:=#27; special:=false
               end else ch:=#0;
               end;
            ch1:=ch;
        end;
   n:=ord(ch1);
   hidemouse;
   if i=L then begin
           setfillstyle(1,0);
           bar(370,315,530,360);          {result show}
           setcolor(15);
           outtextxy(385,345,'Line is too long.');
           setcolor(7);
           outtextxy(380,325,'Error :');
           acc:=true;
   end else if acc then
   begin
     setfillstyle(1,0);
     bar(370,315,530,360);
     acc:=false;
   end;
if long<L then begin
   if ((ch1<>#13)and(special=false))and((((n>=65)and(n<=90))or
   ((n>=97)and(n<=122))or((n>=48)and(n<=57)))and(special=false))
   or(n=45) then begin
    setfillstyle(1,0);
    bar(x+bet-1,y-12,x+bet+6,y);
    outtextxy(x+bet,y-10,ch);
    i:=i+1;long:=long+1;bet:=bet+8;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        RecArray[i]:=ch1;
      end;
  end;
      if not(special) then
      case ch1 of
      #32:begin
          if long<L then begin
            RecArray[i+1]:=' ';
            i:=i+1;bet:=bet+8;long:=long+1;
            setfillstyle(1,0);bar(x+bet-8,y-12,x+bet,y+1);
          end
         end;
      #8 :begin
          if i<>0 then begin
            setcolor(0);
            line(x+bet,y,x+bet+5,y);
            line(x+bet,y-1,x+bet+5,y-1);
            bet:=bet-8;long:=long-1;
            RecArray[i]:=' ';i:=i-1;
            setfillstyle(1,0);bar(x+bet,y-12,x+bet+8,y+1);
          end
         end;
      #27:begin
          Not_Search:=true;
         end;
      #13:begin
          hidemouse;
          SetLineStyle(0,15,1);
          setcolor(0);
          line(x+bet,y,x+5+bet,y);
          line(x+bet,y-1,x+5+bet,y-1);
          showmouse;
        end
      end;
      if special then
      case ch1 of
      #75:begin  {left}
          if i>0 then begin
            setcolor(0);
            line(x+bet,y,x+bet+5,y);
            line(x+bet,y-1,x+bet+5,y-1);
            i:=i-1;bet:=bet-8;
          end
         end;
      #77:begin  {right}
          if i<long-1 then begin
            setcolor(0);
            line(x+bet,y,x+bet+5,y);
            line(x+bet,y-1,x+bet+5,y-1);
            i:=i+1;bet:=bet+8;
          end
         end;

      #59:begin  {F1}
          clear:=true;
          i:=0; long:=0; bet:=0;
         end
      end;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
      showmouse;
     until (ch1=#13)or(Not_Search)or(clear)or(Text='EXIT');
     if ch1=#13 then
      for i:=1 to long do Text:=Text+RecArray[i];

   end;

  procedure Rec_Text(x,y:integer;var Text:string;var Not_Search:boolean);
   var i,n,bet,long : integer;
        RecArray : array[1..20] of char;
   begin
    for i:=1 to 20 do
      RecArray[i]:=' ';
    long:=0;Text:='';textcolor(15);bet:=0;
    Not_Search:=false;i:=0;
    repeat
     if not(install) then begin
       CursorNoMouse(x+bet,y);
       Testkey(ch,special);
       ch1:=ch;
     end else begin
               CursorStop(x+bet,y);
               repeat until (keypressed or mouseCLICK);
               if  keypressed then  Testkey(ch,special)
               else begin delay(50);
                  if area(534,85,544,95) then begin
                    Afterpress(534,85,544,95);
                    setcolor(12);outtextxy(536,87,'X');
                    delay(100);
                    Press(534,85,544,95);
                    setcolor(0);outtextxy(536,87,'X');
                    Not_search:=true;
                  end else
                  if area(376,205,408,220) then begin
                    ch:=#13; special:=false
                  end else
                  if area(434,205,466,220) then begin
                    ch:=#59; special:=true
                  end else
                  if area(492,205,524,220) then begin
                    ch:=#27; special:=false
                  end else ch:=#0;
                  end;
              ch1:=ch;
          end;
    n:=ord(ch1);
    hidemouse;
    if long=13 then begin
            setfillstyle(1,0);
            bar(370,315,530,360);          {result show}
            setcolor(15);
            outtextxy(385,345,'Line is too long.');
            setcolor(7);
            outtextxy(380,325,'Warning :');

        {--clear what ?--}
    end;
  if long<13 then begin
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        if ((ch1<>#13)and(special=false))and((((n>=65)and(n<=90))or
        ((n>=97)and(n<=122))or((n>=48)and(n<=57)))and(special=false))
        or(n=45)
        {or((n>32)and(n<127))} then begin
         setfillstyle(1,0);
         bar(x+bet-1,y-12,x+bet+6,y);
         outtextxy(x+bet,y-10,ch);
         {setcolor(8);
         ThaiCharRW(x+bet+3,y-18,ch);
         setcolor(15);
         ThaiCharRW(x+bet+2,y-17,ch);}
         i:=i+1;long:=long+1;bet:=bet+8;
         RecArray[i]:=ch;
       end;
    end;
        if not(special) then
        case ch1 of
        #32:begin
            if long<13 then begin
              RecArray[i+1]:=' ';
              i:=i+1;bet:=bet+8;long:=long+1;
              setfillstyle(1,0);bar(x+bet-8,y-12,x+bet,y+1);
            end
           end;
        #8 :begin
            if i<>0 then begin
             setcolor(0);
             line(x+bet,y,x+bet+5,y);
             line(x+bet,y-1,x+bet+5,y-1);
             bet:=bet-8;long:=long-1;
             RecArray[i]:=' ';i:=i-1;
             setfillstyle(1,0);bar(x+bet,y-12,x+bet+8,y+1);
            end
           end;
        #27:begin
            AfterPress(492,205,524,220);
            setcolor(12);outtextxy(505,209,'X');
            delay(100);
            Press(492,205,524,220);
            setcolor(15);outtextxy(505,209,'X');
            Not_Search:=true;
           end;
        #13:begin
            AfterPress(376,205,408,220);
            setcolor(12);outtextxy(385,209,'OK');
            delay(100);
            Press(376,205,408,220);
            setcolor(15);outtextxy(385,209,'OK');
            SetLineStyle(0,15,1);
            setcolor(0);
            line(x+bet,y,x+5+bet,y);
            line(x+bet,y-1,x+5+bet,y-1);
           end
        end;
        if special then
        case ch1 of
        #75:begin  {left}
            if i>0 then begin
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                    setcolor(0);
                    line(x+bet,y,x+bet+5,y);
                    line(x+bet,y-1,x+bet+5,y-1);
                    i:=i-1;bet:=bet-8;
                  end
          end;
        #77:begin  {right}
            if i<long-1 then begin
              setcolor(0);
              line(x+bet,y,x+bet+5,y);
              line(x+bet,y-1,x+bet+5,y-1);
              i:=i+1;bet:=bet+8;
            end
           end;

        #59:begin   {F1}
            clear:=true;
            Afterpress(434,205,466,220);
            setcolor(12);outtextxy(439,209,'Clr');
            delay(100);
            press(434,205,466,220);
            setcolor(15);outtextxy(439,209,'Clr');
            setfillstyle(1,0);
            i:=0;long:=0;bet:=0;
            bar(430,123,540,138);
            bar(430,143,540,158);
            bar(430,163,540,178);
           end
        end;
        showmouse;
      until (ch1=#13)or(Not_Search)or(clear);
      for i:=1 to long do
      Text:=Text+RecArray[i];
    end;

  procedure DoSub1b(st:string);
  var      ns : boolean;
           t : string;
  begin {DoSub1b}
      Cancel:=false;
      Clear:=false;
      SubMenu1b;
      setcolor(15);
      outtextxy(435,249,st);  {text file name}
      Rec_text(435,137,t,ns);
      if (not(ns))and(not(clear)) then begin
       hidemouse;
       setfillstyle(1,0);
       bar(370,315,530,360);          {result show}
       setcolor(15);
       outtextxy(405,345,t);
       setcolor(7);
       outtextxy(380,325,'Your name :');
       showmouse;
       PersonName:=t;
      end  else cancel:=true;
    if (not(cancel))and(not(clear)) then begin
      Rec_text(435,157,t,ns);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
      if (not(ns))and(not(clear)) then begin
       hidemouse;
       setfillstyle(1,0);
       bar(370,315,530,360);          {result show}
       setcolor(15);
       outtextxy(405,345,t);
       setcolor(7);
       outtextxy(380,325,'Surname :');
       showmouse;
       PersonSur:=t;
      end  else cancel:=true;
    end;
    if (not(cancel))and(not(clear)) then begin
      Rec_text(435,177,t,ns);
      if (not(ns))and(not(clear)) then begin
       hidemouse;
       setfillstyle(1,0);
       bar(370,315,530,360);          {result show}
       setcolor(15);
       outtextxy(405,345,t);
       setcolor(7);
       outtextxy(380,325,'Login name :');
       showmouse;
       PersonCode:=t;
      end  else cancel:=true;
    end;
  end;
  procedure Submenu1c;
  begin
     Submenu1;
     hidemouse;
     settextstyle(3,0,4);
     setcolor(12);
     outtextxy(370,126,'Slide Now !');
     settextstyle(0,0,1);
     outtextxy(363,249,'Push any key to cancel');
     Press(415,190,485,215);
     outtextxy(425,198,'Cancel');
     setfillstyle(1,0);
     bar(370,315,530,360);          {result show}
     showmouse;
  end;

  procedure DoSub1c;
  var  data  : byte;
       dat   : string;
       count : integer;
       fact  : boolean;
    procedure noblank(var x:boolean);
    var i:integer;
    begin
       x:=false;
       dat:='';
       for i:=1 to Length(PersonName) do
          if PersonName[i]<>' ' then
            dat:=dat+PersonName[i];
       PersonName:=dat;
       if dat='' then x:=x or true;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        dat:='';
        for i:=1 to Length(PersonSur) do
          if PersonSur[i]<>' ' then
            dat:=dat+PersonSur[i];
        PersonSur:=dat;
        if dat='' then x:=x or true;
        dat:='';
        for i:=1 to Length(PersonCode) do
          if PersonCode[i]<>' ' then
            dat:=dat+PersonCode[i];
        PersonCode:=dat;
        if dat='' then x:=x or true;

    end;
  begin
    Cancel:=false;
    count:=0;
    Noblank(fact);
    if not(fact) then begin
     dat:=st;
     delete(dat,length(dat)-2,3);
     dat:=dat+'id';
     assign(f1,dat);
     append(f1);
     write(f1,PersonCode+':'+PersonName+':'+PersonSur);
     append(f);
     Submenu1c;
     write(f,PersonCode+':');
     setcolor(7);
     outtextxy(380,325,'Data Recieve :');
     SendFrame(Frame,Add_New_User,0);   {Add_New_User=U_Frame=10000011=131}
     setcolor(15);
     repeat
     if V24DataAvail then
       begin
          inc(count);hidemouse;
          data:=V24GetByte;
          str(data,dat);
           if length(dat)=1 then dat:='0'+dat;
          write(f,' '+dat);    {1st blank is placed before 1st DATA}
          bar(430,340,470,360);
          outtextxy(435,345,dat);showmouse;
       end;
    until (count=40)or(keypressed)or(area(415,190,485,215));

    if (count<40) then begin
      hidemouse;
      writeln(f,'!'); {Mark no use}
      close(f);
      writeln(f1,'!');
      AfterPress(415,190,485,215);
      setcolor(14);
      outtextxy(425,198,'Cancel');
      delay(200);
      Press(415,190,485,215);
      setcolor(15);
      outtextxy(425,198,'Cancel');
      setfillstyle(1,0);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        bar(370,315,530,360);          {result show}
        outtextxy(380,325,'Data error :');
        bar(430,340,470,360);
        outtextxy(395,345,'Add again');
        showmouse;
        delay(400);
      end else begin
            writeln(f,' '); close(f);
            writeln(f1,' ');
            hidemouse;
            setfillstyle(1,0);
            bar(370,315,530,360);          {result show}
            outtextxy(375,325,PersonCode+' registered :');
            bar(430,340,470,360);
            outtextxy(395,345,'Welcome!');
            setfillstyle(1,9);
            bar(360,120,540,180);
            Press(415,190,485,215);
            setcolor(15);
            outtextxy(425,198,' Done ');
            settextstyle(3,0,4);
            setcolor(12);
            outtextxy(370,126,'Registered!');
            settextstyle(0,0,1);
            showmouse;
            delay(400);
          end;
    close(f1);
    if not(install) then testkey(ch,special) else keypressed;
    AfterPress(415,190,485,215);
    setcolor(14);
    outtextxy(425,198,' Done ');
    delay(200);
    Press(415,190,485,215);
    setcolor(15);
    outtextxy(425,198,' Done ');

  end else clear:=true;
end;

procedure SubMenu2;
begin
    Beep;
    hidemouse;
    setfillstyle(1,0);
    bar(386,455,570,470);
    setcolor(4);
    outtextxy(391,461,'Download Data');
    setcolor(12);
    outtextxy(390,462,'Download Data');
    setcolor(15);
    Subwin(350,80,200,190,9,0,15);    {download}
    Subwin(390,86,125,20,13,0,15);    {title download}
    if (install) then begin
      press(534,85,544,95);
      setcolor(0);outtextxy(536,87,'X');
    end;
    setcolor(9);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    outtextxy(403,92,'Download Data');
    setcolor(11);
    outtextxy(402,93,'Download Data');setcolor(15);
    setfillstyle(1,8);
    Bar(355,240,545,265);          {choose file}
    rectangle(358,243,542,262);
    Subwin(350,290,200,80,4,0,15);   {result}
    Subwin(421,296,56,12,15,0,15);   {title result}
    setcolor(7);
    outtextxy(427,298,'Result');
    setcolor(0);
    outtextxy(426,299,'Result');setcolor(15);
    setfillstyle(3,15);
    bar(370,315,530,360);          {result show}
    showmouse;
    Optimize;
end;
procedure SubMenu2a;
begin
    hidemouse;
    settextstyle(0,0,1);
    outtextxy(400,117,'Sophomore');
    press(370,115,380,125);
    outtextxy(400,137,'Junior');
    press(370,135,380,145);
    outtextxy(400,157,'Senior');
    press(370,155,380,165);
    outtextxy(400,177,'Master');
    press(370,175,380,185);
    outtextxy(400,197,'Teacher');
    press(370,195,380,205);
    outtextxy(400,217,'All');
    press(370,215,380,225);
    setcolor(0);
    outtextxy(372,117,'1');
    outtextxy(372,137,'2');
    outtextxy(372,157,'3');
    outtextxy(372,177,'4');
    outtextxy(372,197,'5');
    outtextxy(372,217,'6');
    setfillstyle(1,0);
    bar(504,246,537,259);          {choice}
    settextstyle(0,0,1);
    setcolor(6);
    outtextxy(366,250,'Choose File :');
    setcolor(14);
    outtextxy(365,249,'Choose File :');setcolor(15);
    showmouse;
end;

procedure DoSub2a;
var mouse2a:boolean;
 procedure Result1a(c:char;s:string);
  begin
    hidemouse;
    setfillstyle(1,0);
    bar(504,246,537,259);          {choice}
    bar(370,315,530,360);          {result show}
```

เอกสารนี้จัดทำขึ้นสำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    setcolor(15);
    outtextxy(519,250,c);
    outtextxy(405,345,s);
    setcolor(7);
    outtextxy(380,325,'You choose file :');
    showmouse;
  end;
procedure press1a(a,b,c,d:integer;h:char;j:string);
  begin
      hidemouse;
      afterpress(a,b,c,d);
      setcolor(12);
      outtextxy(400,b+2,j);
      outtextxy(a+2,b+2,h);
      delay(100);
      press(a,b,c,d);
      setcolor(0);
      outtextxy(a+2,b+2,h);
      setcolor(15);
      outtextxy(400,b+2,j);
      showmouse;
      ch1:=h;
      mouse2a:=true;
end;

begin
    SubMenu2;
    SubMenu2a;
    cancel:=false;
    mouse2a:=false;
    st:='No Plan';
    repeat
    ch1:=#0;
    ch:=ch1;
    if not(install) then begin
      CursorNoMouse(519,258);
      TestKey(ch,special);
      ch1:=ch;
    end else begin
          repeat until (ch<>ch1)or(mouseCLICK);
          ch1:=ch;
        end;
    if area(370,115,380,125) then
      press1a(370,115,380,125,'1','Sophomore')
    else
    if area(370,135,380,145) then
      press1a(370,135,380,145,'2','Junior')
    else
    if area(370,155,380,165) then
      press1a(370,155,380,165,'3','Senior')
    else
    if area(370,175,380,185) then
      press1a(370,175,380,185,'4','Master')
    else
    if area(370,195,380,205) then
      press1a(370,195,380,205,'5','Teacher')
    else
    if area(370,215,380,225) then
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        press1a(370,215,380,225,'6','All')
      else
      if area(534,85,544,95) then begin
        afterpress(534,85,544,95);
        setcolor(12);outtextxy(536,87,'X');
        delay(100);
        press(534,85,544,95);
        setcolor(0);outtextxy(536,87,'X');
        ch1:=#27;
        mouse2a:=true;
      end;

      case ch1 of
      '1':begin st:='Sophom.txt';
            Result1a('1',st);
        end;
      '2':begin st:='Junior.txt';
            Result1a('2',st);
        end;
      '3':begin st:='Senior.txt';
            Result1a('3',st);
        end;
      '4':begin st:='Master.txt';
            Result1a('4',st);
        end;
      '5':begin st:='Teacher.txt';
            Result1a('5',st);
        end;
      '6':begin st:='ALL';
            Result1a('6',st);
            st:='All.txt';
        end;
      #13:begin if (st='No Plan')or(st='') then
              ch1:=#0;
        end;
      #27:begin Cancel:=true;
            st:='';
        end
      else
        begin Result1a(' ','  None  ');
            st:='';
        end;
      end;
      until (ch1=#13)or(cancel=true)or(mouse2a);
      if mouse2a then delay(200);
      assign(f,st);
      reset(f);
      {out = cancel,st}
    end;
    procedure DoSub2b(st:string);
     function fileTxt(st:string):byte;
     begin
        if st='Sophom.txt' then
          fileTxt:=1
        else
        if st='Junior.txt' then
          fileTxt:=2
        else
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    if st='Senior.txt' then
      fileTxt:=3
    else
    if st='Master.txt' then
      fileTxt:=4
    else
    if st='Teacher.txt' then
      fileTxt:=5;
end;
procedure Download;
var   BTS,para,numS      : string;
      BTV,code,i,time,numV : integer;
      inv              : integer;
 begin
    hidemouse;
    setfillstyle(1,0);
    bar(370,315,530,360);
    setfillstyle(1,8);
    Bar(355,240,545,265);
    setcolor(15);
    rectangle(358,243,542,262);
    setcolor(12);outtextxy(364,249,'Download');
    setcolor(6) ;outtextxy(440,248,st);
    setcolor(14);outtextxy(439,249,st);
    setcolor(7);outtextxy(375,320,'Byte No:');
    showmouse;
    reset(f); time:=0;
    repeat
      readln(f,para);
      if para[length(para)]<>'!' then time:=time+1;
    until eof(f);
    str(time,para);
    setcolor(15);
    outtextxy(400,207,'0');
    setfillstyle(1,9);bar(490,205,520,219);outtextxy(495,207,para);
    inv:=(100 div time);
    time:=time-1;
    SendFrame(Frame,Activate,0);          {Activate =U_Frame=10101011=171}
    RecFrame(RFrame);
    delay(200);
    SendFrame(Frame,StartFile,fileTxt(st)); {StartFile=U_Frame=01010111=87}
    delay(200);
    reset(f);
    repeat
        readln(f,para);
        i:=length(para)+1;
        repeat
          dec(i);
        until para[i]<>' ';
     if para[length(para)]<>'!' then begin
        Delete(para,i+1,length(para)-i);
        i:=0;
        repeat
          inc(i);
        until para[i]=':';
        i:=i+2;
        numV:=0;
        repeat
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
            inc(numV);
            str(numV,numS);
            setfillstyle(1,0);bar(440,320,460,330);
            setcolor(7);outtextxy(440,320,numS);
            BTS:=para[i]+para[i+1];
            val(BTS,BTV,code);
            hidemouse;
            bar(440,340,460,350);
            setcolor(15);outtextxy(440,340,BTS);
            showmouse;
            delay(5); {recommend=20}
            SendFrame(Frame,DataByte,BTV);    {DataByte=0}
            repeat
              i:=i+1;
            until para[i]=' ';
            inc(i);
          until (i)>=length(para);
          Delay(5); {recommend=150}
          SendFrame(Frame,Individual,0);      {Individual=00000011b=3}
        end;
        hidemouse;
        setfillstyle(1,10);
        bar(401,181,500-(time*inv),199);   {progess bar}
        showmouse;
        dec(time);
       until eof(f);
       delay(100);
       SendFrame(Frame,EndFile,0);          {EndFile=10010011=147}
       InitMouse(install); Showmouse;
    end;
  procedure  ClearBar;
  begin
            hidemouse;
            delay(300);
            setfillstyle(1,9);
            bar(400,180,500,200);
            setcolor(7);
            rectangle(400,180,502,202);
            setcolor(15);
            rectangle(398,178,500,200);
            rectangle(399,179,501,201);
            setfillstyle(1,8);
            Bar(355,240,545,265);
            showmouse;
   end;
  begin
     hidemouse;
     setfillstyle(1,9);
     bar(368,107,537,240);
     setfillstyle(1,8);
     Bar(355,240,545,265);
     setcolor(15);
     rectangle(358,243,542,262);
     setfillstyle(3,15);
     bar(370,315,530,360);          {result show}
     setcolor(12);outtextxy(364,250,'Press Esc or Enter');
     Press(385,180,435,195);
     Press(465,180,515,195);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
outtextxy(396,184,'O.K.');
outtextxy(467,184,'Cancel');
setcolor(15);outtextxy(364,120,'Downloading');
        outtextxy(415,135,'to SERVER');
setcolor(7); outtextxy(460,120,st);
setcolor(15);outtextxy(459,121,st);
showmouse;
repeat
 if not(install) then
   testkey(ch,special)
 else begin ch1:=#0; ch:=ch1;
        repeat until (ch<>ch1)or(mouseCLICK);
    end;
 if (ch=#13)or(area(385,180,435,195)) then
   begin
       hidemouse;
       AfterPress(385,180,435,195);
       setcolor(12);outtextxy(396,184,'O.K.');
       delay(100);
       Press(385,180,435,195);
       setcolor(15);outtextxy(396,184,'O.K.');
       setfillstyle(1,9);
       bar(380,170,520,200);
       setcolor(7);
       rectangle(400,180,502,202);
       setcolor(15);
       rectangle(398,178,500,200);
       rectangle(399,179,501,201);
       setcolor(15);
       rectangle(358,243,542,262);
       showmouse;
       if st<>'All.txt' then
         Download
       else begin  st:='Sophom.txt'; assign(f,st) ;
                Download; ClearBar;
              st:='Junior.txt'; assign(f,st) ;
                Download; ClearBar;
              st:='Senior.txt'; assign(f,st) ;
                Download; ClearBar;
              st:='Master.txt'; assign(f,st) ;
                Download; ClearBar;
              st:='Teacher.txt';assign(f,st) ;
                Download;
          end;
       ch:=#13;
    end;

 if (ch=#27)or(area(465,180,515,195)) then
   begin
       hidemouse;
       AfterPress(465,180,515,195);
       setcolor(12);outtextxy(467,184,'Cancel');
       delay(100);
       Press(465,180,515,195);
       setcolor(15);outtextxy(467,184,'Cancel');
       showmouse;
       ch:=#27;
   end;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    until ch in [#13,#27];
end;

Procedure SubMenu3;
begin
    Beep;
    hidemouse;
    setfillstyle(1,0);
    bar(386,455,570,470);
    setcolor(4);
    outtextxy(391,461,'Broadcast');
    setcolor(12);
    outtextxy(390,462,'Broadcast');
    setcolor(15);
    Subwin(350,80,200,190,9,0,15);
    Subwin(405,86,90,20,13,0,15);
    if install then begin
      press(534,85,544,95);
      setcolor(0);outtextxy(536,87,'X');
    end;
    setcolor(9);
    outtextxy(415,92,'Broadcast');
    setcolor(11);
    outtextxy(414,93,'Broadcast');setcolor(15);
    setfillstyle(1,8);
    Bar(355,240,545,265);          {choose file}
    rectangle(358,243,542,262);
    Subwin(350,290,200,80,4,0,15);   {result}
    Subwin(421,296,56,12,15,0,15);   {title result}
    setcolor(7);
    outtextxy(427,298,'Result');
    setcolor(0);
    outtextxy(426,299,'Result');setcolor(15);
    setfillstyle(3,15);
    bar(370,315,530,360);          {result show}
    showmouse;
end;
procedure SubMenu3a;
begin
    hidemouse;
    settextstyle(0,0,1);
    outtextxy(400,117,'Disconnect All');
    press(370,115,380,125);
    outtextxy(400,137,'[Reserve]');
    press(370,135,380,145);
    setcolor(0);
    outtextxy(372,117,'1');
    outtextxy(372,137,'2');
    setfillstyle(1,0);
    bar(504,246,537,259);          {choice}
    settextstyle(0,0,1);
    setcolor(6);
    outtextxy(366,250,'Choose Command :');
    setcolor(14);
    outtextxy(365,249,'Choose Command :');setcolor(15);
    showmouse;
end;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
procedure DoSub3a(var st:string);
var mouse3a:boolean;
 procedure Result3a(c:char;s:string);
  begin
    hidemouse;
    setfillstyle(1,0);
    bar(504,246,537,259);          {choice}
    bar(370,315,530,360);          {result show}
    setcolor(15);
    outtextxy(519,250,c);
    outtextxy(405,345,s);
    setcolor(7);
    outtextxy(372,325,'You choose command :');
    showmouse;
  end;
procedure press3a(a,b,c,d:integer;h:char;j:string);
  begin
      hidemouse;
      afterpress(a,b,c,d);
      setcolor(12);
      outtextxy(400,b+2,j);
      outtextxy(a+2,b+2,h);
      delay(100);
      press(a,b,c,d);
      setcolor(0);
      outtextxy(a+2,b+2,h);
      setcolor(15);
      outtextxy(400,b+2,j);
      showmouse;
      ch1:=h;
      mouse3a:=true;
end;

begin
    SubMenu3;
    SubMenu3a;
    cancel:=false;
    mouse3a:=false;
    st:='No Plan';

    repeat
    ch1:=#0;
    ch:=ch1;
    if not(install) then begin
      CursorNoMouse(519,258);
      TestKey(ch,special);
      ch1:=ch;
    end else begin
            repeat until (ch<>ch1)or(mouseCLICK);
            ch1:=ch;
           end;
    if area(370,115,380,125) then
      press3a(370,115,380,125,'1','Disconnect All')
    else
    if area(370,135,380,145) then
      press3a(370,135,380,145,'2','[Reserve]')
    else
    if area(534,85,544,95) then begin
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        afterpress(534,85,544,95);
        setcolor(12);outtextxy(536,87,'X');
        delay(100);
        press(534,85,544,95);
        setcolor(0);outtextxy(536,87,'X');
        ch1:=#27;
        mouse3a:=true;
      end;

      case ch1 of
      '1':begin st:='Disconnect All';
            Result3a('1',st);
        end;
      '2':begin st:='[Reserve]';
            Result3a('2',st);
        end;
      #13:begin if (st='No Plan')or(st='') then
              ch1:=#0;
        end;
      #27:begin Cancel:=true;
            st:='';
        end
      else
        begin Result3a(' ',' None ');
            st:='';
        end;
      end;
      until (ch1=#13)or(cancel=true)or(mouse3a);
      {out = cancel,st}

  end;
  Procedure DoSub3b(st:string);
  begin

  end;
  procedure SubMenu4;
  begin
     Beep;
     hidemouse;
     setfillstyle(1,0);
     bar(386,455,570,470);
     setcolor(4);
     outtextxy(391,461,'Delete Data');
     setcolor(12);
     outtextxy(390,462,'Delete Data');
     setcolor(15);
     Subwin(350,80,200,190,9,0,15);    {delete card}
     Subwin(398,86,111,20,13,0,15);    {title delete card}
     if install then begin
       press(534,85,544,95);
       setcolor(0);outtextxy(536,87,'X');
     end;
     setcolor(9);
     outtextxy(410,92,'Delete Card');
     setcolor(11);
     outtextxy(409,93,'Delete Card');setcolor(15);
     setfillstyle(1,8);
     Bar(355,240,545,265);          {choose file}
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
      rectangle(358,243,542,262);
      Subwin(350,290,200,80,4,0,15);   {result}
      Subwin(421,296,56,12,15,0,15);   {title result}
      setcolor(7);
      outtextxy(427,298,'Result');
      setcolor(0);
      outtextxy(426,299,'Result');setcolor(15);
      setfillstyle(3,15);
      bar(370,315,530,360);            {result show}
      showmouse;
    end;
    procedure DoSub4a(var st:string);
    var  mouse4a:boolean;
     procedure Result1a(c:char;s:string);
     begin
       hidemouse;
       setfillstyle(1,0);
       bar(504,246,537,259);           {choice}
       bar(370,315,530,360);           {result show}
       setcolor(15);
       outtextxy(519,250,c);
       outtextxy(405,345,s);
       setcolor(7);
       outtextxy(380,325,'You choose file :');
       showmouse;
     end;
     procedure press1a(a,b,c,d:integer;h:char;j:string);
     begin
         hidemouse;
         afterpress(a,b,c,d);
         setcolor(12);
         outtextxy(400,b+2,j);
         outtextxy(a+2,b+2,h);
         delay(100);
         press(a,b,c,d);
         setcolor(0);
         outtextxy(a+2,b+2,h);
         setcolor(15);
         outtextxy(400,b+2,j);
         showmouse;
         ch1:=h;
         mouse4a:=true;
     end;

   begin
      mouse4a:=false;
      cancel:=false;
      st:='No Plan';
      SubMenu4;
      Submenu1a;
      repeat
      if not(install) then begin
        CursorNoMouse(519,258);
        TestKey(ch,special);
        ch1:=ch;
      end else begin
            ch1:=#0;ch:=ch1;
            repeat until (mouseCLICK)or(ch<>ch1);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
            ch1:=ch;
          end;
    if area(370,115,380,125) then
      press1a(370,115,380,125,'1','Sophomore')
    else
    if area(370,140,380,150) then
      press1a(370,140,380,150,'2','Junior')
    else
    if area(370,165,380,175) then
      press1a(370,165,380,175,'3','Senior')
    else
    if area(370,190,380,200) then
      press1a(370,190,380,200,'4','Master')
    else
    if area(370,215,380,225) then
      press1a(370,215,380,225,'5','Teacher')
    else
    if area(534,85,544,95) then begin
      afterpress(534,85,544,95);
      setcolor(12);outtextxy(536,87,'X');
      delay(100);
      press(534,85,544,95);
      setcolor(0);outtextxy(536,87,'X');
      ch1:=#27;
      mouse4a:=true;
    end;
    case ch1 of
    '1':begin st:='Sophom.txt';
          Result1a('1',st);
      end;
    '2':begin st:='Junior.txt';
          Result1a('2',st);
      end;
    '3':begin st:='Senior.txt';
          Result1a('3',st);
      end;
    '4':begin st:='Master.txt';
          Result1a('4',st);
      end;
    '5':begin st:='Teacher.txt';
          Result1a('5',st);
      end;
    #13:begin if (st='No Plan')or(st='') then
            ch1:=#0;
      end;
    #27:begin Cancel:=true;
          st:='';
      end
    else
      begin Result1a(' ','  None  ');
          st:='';
      end;
    end;
    until (ch1=#13)or(cancel=true)or(mouse4a);
    if mouseCLICK then delay(200);
    assign(f,st);
    reset(f);
    {out = cancel,st}
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
end;
procedure SubMenu4b(st:string);
begin
  hidemouse;
  setcolor(6);
  outtextxy(391,136,'Login Name :');
  outtextxy(364,248,'Del from :');
  setcolor(14);
  outtextxy(390,137,'Login Name :');
  outtextxy(363,249,'Del from :');
  setcolor(15);
  outtextxy(445,249,st);  {text file name}
  setfillstyle(1,0);
  bar(395,163,505,178);
  press(376,205,408,220);
  press(434,205,466,220);
  press(492,205,524,220);
  outtextxy(385,209,'OK');
  outtextxy(439,209,'Clr');
  outtextxy(505,209,'X');
  setcolor(0);
  outtextxy(374,227,'Enter');
  outtextxy(443,227,'F1');
  outtextxy(498,227,'Esc');
  showmouse;
end;
procedure DoSub4b;
var  t:string;
begin
 Submenu4;
 repeat
  SubMenu4b(st);
  Rec_text1(399,177,8,t);
  hidemouse;
  if t='EXIT' then
  begin
    Afterpress(534,85,544,95);
    setcolor(12);outtextxy(536,87,'X');
    delay(100);
    Press(534,85,544,95);
    setcolor(0);outtextxy(536,87,'X');
  end else
  if ch=#13 then
  begin
    AfterPress(376,205,408,220);
    setcolor(12);outtextxy(385,209,'OK');
    delay(100);
    Press(376,205,408,220);
    setcolor(15);outtextxy(385,209,'OK');
    key:=t
  end else
  if not_search then
  begin
    AfterPress(492,205,524,220);
    setcolor(12);outtextxy(505,209,'X');
    delay(100);
    Press(492,205,524,220);
    setcolor(15);outtextxy(505,209,'X');
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    end else
    if clear then
    begin
      Afterpress(434,205,466,220);
      setcolor(12);outtextxy(439,209,'Clr');
      delay(100);
      press(434,205,466,220);
      setcolor(15);outtextxy(439,209,'Clr');
      setfillstyle(1,0);
      bar(395,163,505,178);
    end;
    showmouse;
  until (ch=#13)or(not_search)or(t='EXIT');
end;
procedure DelBlank(var t:string);
var   i  :integer;
      tx :String;
begin
      tx:='';
      for i:=1 to length(t) do begin
         if t[i]<>' ' then
           tx:=tx+t[i];
      end;
      t:=tx;
end;
procedure Search(var l:integer);
{out=l,  in=key,st:filename}
var   para,temp : string;
          i,j ; integer;
    success,yet : boolean;
begin
  l:=0; j:=0; yet:=true; success:=false;
  repeat
    temp:=''; i:=0; yet:=true;
    readln(f,para); j:=j+1;
    DelBlank(para);
    if para[length(para)]<>'!' then begin
      while (i<length(para))and(yet) do begin
         i:=i+1;
         if   para[i]<>':' then temp:=temp+para[i]
         else yet:=false;
      end;
      if not(yet) then
        if temp=key then begin success:=true; l:=j; end;
    end;
  until (success)or(eof(f));
end;
procedure DoSub4c;
begin
    hidemouse;
    setfillstyle(1,0);
    bar(370,315,530,360);          {result show}
    setcolor(7);
    outtextxy(380,325,'Data '+key);
    reset(f);
    Search(lin);
    setcolor(15);
    if lin=0 then begin
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        outtextxy(405,345,'not found');
      end else begin
                outtextxy(424,345,'found');
              end;
      showmouse;
      repeat
       if not(install) then begin
         testkey(ch,special);
         ch1:=ch;
       end
       else begin
            ch1:=#0; ch:=ch1;
            repeat until (ch<>ch1)or(mouseCLICK);
            ch1:=ch;
            if mouseCLICK then
              if area(376,205,408,220) then ch1:=#13 else
              if area(492,205,524,220) then ch1:=#27 else
              if area(434,205,466,220) then ch1:=#59;
          end;
       hidemouse;
       case ch1 of
       #13:begin Not_search:=false;
             AfterPress(376,205,408,220);
             setcolor(12);outtextxy(385,209,'OK');
             delay(100);
             Press(376,205,408,220);
             setcolor(15);outtextxy(385,209,'OK');
         end;
       #27:begin Not_search:=true;
             AfterPress(492,205,524,220);
             setcolor(12);outtextxy(505,209,'X');
             delay(100);
             Press(492,205,524,220);
             setcolor(15);outtextxy(505,209,'X');
         end;
       #59:begin Clear:=true;
             Afterpress(434,205,466,220);
             setcolor(12);outtextxy(439,209,'Clr');
             delay(100);
             press(434,205,466,220);
             setcolor(15);outtextxy(439,209,'Clr');
         end;
       end;
       showmouse;
     until ch1 in [#13,#27,#59];
end;
procedure DoSub4d;
type   para=string[155];
var
      i,j:integer;
     ftemp:array[1..75] of para;
begin
 if lin<>0 then
 begin
   hidemouse;
   setfillstyle(1,9);
   bar(368,107,537,240);
   setfillstyle(1,8);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
Bar(355,240,545,265);
setcolor(15);
rectangle(358,243,542,262);
setfillstyle(3,15);
bar(370,315,530,360);          {result show}

Press(385,180,435,195);
Press(465,180,515,195);
outtextxy(396,184,'O.K.');
outtextxy(467,184,'Cancel');
setcolor(15);outtextxy(364,120,'Are you sure, you want');
        outtextxy(364,134,'to delete       ?');
setcolor(7); outtextxy(448,133,key);
setcolor(15);outtextxy(447,134,key);
setcolor(12);outtextxy(364,250,'Press Esc or Enter');
showmouse;
repeat
 if not(install) then begin
   testkey(ch,special); ch1:=ch;
 end
 else begin ch1:=#0; ch:=ch1;
        repeat until (ch<>ch1)or(mouseCLICK);
        ch1:=ch;
        if mouseCLICK then
          if area(385,180,435,195) then ch1:=#13 else
          if area(465,180,515,195) then ch1:=#27;
    end;
 hidemouse;
 if ch1=#13 then
   begin  AfterPress(385,180,435,195);
        setcolor(12);outtextxy(396,184,'O.K.');
        delay(100);
        Press(385,180,435,195);
        setcolor(15);outtextxy(396,184,'O.K.');
        reset(f); i:=1;
        while not(eof(f)) do begin
            readln(f,ftemp[i]);
            i:=i+1;
        end;
        ftemp[lin]:=ftemp[lin]+'!';
        rewrite(f);
        for j:=1 to i-1 do writeln(f,ftemp[j]);
    end;

 if ch1=#27 then
   begin  AfterPress(465,180,515,195);
        setcolor(12);outtextxy(467,184,'Cancel');
        delay(100);
        Press(465,180,515,195);
        setcolor(15);outtextxy(467,184,'Cancel');
    end;
 showmouse;
until ch1 in [#13,#27];
hidemouse;
setfillstyle(1,0);
bar(370,315,530,360);
setcolor(7);
outtextxy(380,325,'Data '+key);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    setcolor(15);
    if ch1=#13 then begin
      outtextxy(385,345,'has been deleted.');
      delay(300);
    end else  begin
            outtextxy(385,345,'has been deleted.');
            delay(300);
          end;
    showmouse;
  end;
end;
procedure SubMenu5;
var    i:integer;
begin
    Beep;
    hidemouse;
    setfillstyle(1,0);
    bar(386,455,570,470);
    setcolor(4);
    outtextxy(391,461,'Version');
    setcolor(12);
    outtextxy(390,462,'Version');
    setcolor(15);
    Subwin(350,80,200,220,4,0,15);    {Version}
    Subwin(415,86,70,20,13,0,15);    {title Version}
    if (install) then begin
      press(534,85,544,95);
      setcolor(0);outtextxy(536,87,'X');
    end;
    setcolor(9);
    outtextxy(423,92,'Version');
    setcolor(11);
    outtextxy(422,93,'Version');setcolor(15);
     setcolor(15);
     outtextxy(360,115,'Name    :');
     outtextxy(360,135,'Version :');
     outtextxy(360,155,'For     :');
     outtextxy(360,175,'Date    :');
     outtextxy(360,195,'Creator :');
     outtextxy(360,255,'Advisor :');
     setcolor(9);
     outtextxy(435,115,VersionName);
     outtextxy(435,135,VersionVersion);
     outtextxy(435,155,VersionFor);
     outtextxy(435,175,VersionDate);
     outtextxy(360,215,VersionC1);
     outtextxy(360,235,VersionC2);
     outtextxy(360,275,VersionAdv);
     setcolor(11);
     outtextxy(434,116,VersionName);
     outtextxy(434,136,VersionVersion);
     outtextxy(434,156,VersionFor);
     outtextxy(434,176,VersionDate);
     outtextxy(359,216,VersionC1);
     outtextxy(359,236,VersionC2);
     outtextxy(359,276,VersionAdv);
     showmouse;
     if not(install) then
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
          testkey(ch,special)
        else begin ch1:=#0; ch:=ch1;
               repeat until (ch<>ch1)or(area(534,85,544,95));
               if mouseCLICK then begin
                 hidemouse;
                 afterpress(534,85,544,95);
                 setcolor(12);outtextxy(536,87,'X');
                 delay(100);
                 press(534,85,544,95);
                 setcolor(0);outtextxy(536,87,'X');
                 showmouse;
               end;
          end;
      hidemouse;
      setfillstyle(1,0);
      bar(348,260,552,300);
      showmouse;
end;
Procedure Submenu6;
var   i:integer;
begin
    Beep;
    hidemouse;
    setfillstyle(1,0);
    bar(386,455,570,470);
    setcolor(4);
    outtextxy(391,461,'Upload Data');
    setcolor(12);
    outtextxy(390,462,'Upload Data');
    setcolor(15);
    Subwin(350,80,200,190,9,0,15);   {upload data}
    Subwin(398,86,111,20,13,0,15);   {upload data}
    setcolor(9);
    outtextxy(410,92,'Upload Card');
    setcolor(11);
    outtextxy(409,93,'Upload Card');
    setfillstyle(1,8);
    Bar(355,240,545,265);
    setcolor(15);
    rectangle(358,243,542,262);
    setcolor(14);
    outtextxy(385,249,'Mouse is disable');
    setcolor(15);
    outtextxy(360,130,'Preparing for TEXT mode');
    setfillstyle(1,0);
    bar(365,173,535,195);
    setcolor(10);
    showmouse;
    i:=1;
    while (i<>20)and(not(keypressed))and(not(mouseCLICK)) do begin
      hidemouse;
      outtextxy(358+(8*i)+4,180,' ');
      inc(i);
      showmouse;
      delay(50);
    end;
    if i=20 then begin
      RestoreCrtMode;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
      InitMouse(install);
      showmouse;
      DumbOrHDLC;
      SetGraphMode(GetGraphMode);
      Basics;
      InitMouse(install);
      ShowMouse;
      MainMenu;
      MouseActive;
      if not(install) then MouseInActive;
    end;
end;
procedure DeleteApplication;
begin
   hidemouse;
   setfillstyle(1,0);
   bar(350,75,560,275);
   bar(350,285,560,455);
   showmouse;
end;
BEGIN
  clrscr;
  {---------------------- Constant Frame ----------------------}
                Frame[1]:=$7E;
                Frame[2]:=IBMPC;
                Frame[7]:=$7E;
  {------------------------------------------------------------}
  Initial;
  Basics;
  InitCom(Com1,b2400,none,d8,s1);
  InitMouse(Install);
  MouseActive;
  if install then begin
    ShowMouse;
    GetIntVec($1C,Int1CSave);
    SetIntVec($1C,Addr(TimerHandler));
  end else MouseInActive;
  Optimize;
  MainMenu;
  Hilight(120,120,'New User');
  position:=1;
  repeat
    setfillstyle(1,0);bar(386,455,570,470);
    setcolor(4); outtextxy(391,461,'Main Menu');
    setcolor(12);outtextxy(390,462,'Main Menu');
    ChooseMain(position);
    logout:=true;
        case position of
          1 : begin
           sub1a: DoSub1a(st);
                If not(cancel) then begin
           sub1b:   DoSub1b(st);
                  if clear then goto sub1b;
                  if cancel then goto sub1a
                  else DoSub1c;
                  if clear then goto sub1b;
                end else begin assign(f,'senior.txt');
                              reset(f);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                    end;
                DeleteApplication;
              end;
            2 : begin
                  DoSub2a;
                  if cancel then goto sub2x;
             sub2b:   DoSub2b(st);
             sub2x: DeleteApplication;
                  close(f);
              end;
            3 : begin
                  DoSub3a(st);
                  DoSub3b(st);
                  DeleteApplication;
              end;
            4 : begin
                  DoSub4a(st);
                  if cancel then goto sub4x;
             sub4b:   DoSub4b;
                  if not(not_search) then begin
                    DoSub4c;
                    if clear then goto sub4b
                    else if not(not_search) then
                        DoSub4d;
                  end;
             sub4x: DeleteApplication;
                  close(f);
              end;
            5 : begin
                  SubMenu5;
                  DeleteApplication;
              end;
            6 : begin
                  SubMenu6;
                  DeleteApplication;
              end;
           10 : begin
                 Logout:=false;
                 {exit}
              end;
           11 : begin
                  {logout:=true,Nothing to do!}
              end
         end;{case}
  until  Logout = false;
      ClearDevice;
      CloseGraph;

END.
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้