



ระบบแสดงค่า และแจ้งเตือน

SYSTEM MONITORING



ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาอุตสาหกรรมศาสตรบัณฑิต

สาขาเทคโนโลยีโทรคมนาคม

คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2537

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไปโดยไม่ได้รับอนุญาต หรือใช้เพื่อวัตถุประสงค์อื่นใดโดยไม่ได้รับอนุญาตจากสถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

หัวข้อปริญญาโท

ระบบแสดงค่า และแจ้งเตือน
SYSTEM MONITORING

ชื่อนักศึกษา

นาย วิวัฒน์ สุชีวกุล

เลขประจำตัว 35103157

นาย สมชาย ไช้ประภาย

เลขประจำตัว 35103159

นาย สราวุธ ปราสาทวัฒนา

เลขประจำตัว 35103165

อาจารย์ที่ปรึกษา

อาจารย์ไพศาล สิทธิโยภาสกุล

ภาควิชา

เทคนิคอุตสาหกรรม

ปีการศึกษา

2537

คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้า เจ้าคุณทหารลาดกระบัง
อนุมัติให้นับปริญญาโทฉบับนี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาอุตสาหกรรมศาสตรบัณฑิต

คณะกรรมการตรวจสอบปริญญาโท

ประธานกรรมการ

()

กรรมการ

()

กรรมการ

()

กรรมการ

()

กรรมการ

()

เอชลิทธีของคณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ระบบแสดงค่า และแจ้งเตือน

โดย

นายวิวัฒน์ สุชีวกุล

นายสมชาย ไช้ประกาย

นายสรารุท ปราสาทวัฒนา

อาจารย์ที่ปรึกษา

อาจารย์ไพศาล สิทธิโยภาสกุล

บทคัดย่อ

ปริญญาบัตรฉบับนี้ได้นำเสนอการออกแบบ ระบบแสดงค่า และแจ้งเตือน การทำงานของเครื่องจักร ในงานอุตสาหกรรม (SYSTEM MONITORING) ระบบนี้สามารถใช้งานกับเครื่อง PERSONAL COMPUTER (PC) ทั่วๆ ไปได้ โดยจะมี 2 ส่วนคือ ส่วนของ HARDWARE เสียบที่ SLOT ของเครื่อง PC และส่วนของ SOFTWARE ที่ควบคุมการทำงานของระบบ ที่สามารถทำงานบน WINDOWS โดยการควบคุมผ่านทาง PORT RS-232 และแปลงเป็น RS-422 เพื่อให้สามารถส่งสัญญาณได้ระยะทางไกลๆ การแสดงผลสามารถแสดงผลทางจอภาพ MONOCHROME , VGA , EGA พร้อมทั้งเก็บข้อมูลลงบนแผ่นดิสค์ ตามเวลาที่กำหนด สามารถขอดูข้อมูลในเวลาที่ผ่านมาได้ และพิมพ์ข้อมูลที่จัดเก็บไว้ ออกทางเครื่องพิมพ์ได้

SYSTEM MONITORING

BY

MR.VIVAT SUCHEVAKUL
 MR.SOMCHAI KAIPAPAI
 MR.SARAWUTH PRASATWATTANA

ADVISER

MR.PISAL SITTIYOPASAKUL

ABSTRACT

This paper to presented " SYSTEM MONITORING " . Monitor about system of machine in industrial job. We design it use on personal computer which consist 2 paths (Hardware and software) . Hardware is interface between input and PC . Software to control system (run on windows) we can sent data to long distance by converse signal from port RS-232 to RS-422 .

Output display on monitor (monochrome , VGA , EGA) .We can set time interval to save data on disk and sent data to printer.

กิตติกรรมประกาศ

วิทยานิพนธ์ฉบับนี้ ได้สำเร็จลุล่วงไปได้ด้วยความช่วยเหลืออย่างดียิ่งของอาจารย์ไพศาล สิทธิโยภาสกุล อาจารย์ที่ปรึกษาวิทยานิพนธ์ ซึ่งท่านได้ให้คำแนะนำ และข้อคิดเห็นต่างๆ ของการวิจัยมาด้วยดีตลอด และได้รับความช่วยเหลือจาก บริษัท วิสโก้ จำกัด และบริษัท พีซีทีที จำกัด ที่ได้ช่วยเหลือทางด้านอุปกรณ์อิเล็กทรอนิกส์ และอุปกรณ์ในการทำปริญญาวิทยานิพนธ์ฉบับนี้



นายวิวัฒน์ สุชีวกุล
นายสมชาย ไช้ประกาย
นายสรายุทธ ปราสาทวัฒนา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ

	หน้า
บทคัดย่อ	ก
ABSTRACT	ข
กิตติกรรมประกาศ	ค
บทที่ 1 บทนำ	1
1.1 วัตถุประสงค์ของโครงการ	1
1.2 ความเป็นมาของโครงการ	1
1.3 ขอบเขตของโครงการ	2
1.4 ขั้นตอน และวิธีดำเนินงาน	2
บทที่ 2 ทฤษฎี และหลักการ	4
2.1 การสื่อสารข้อมูลอนุกรมมาตรฐาน RS-232	4
2.2 การสื่อสารข้อมูลอนุกรมมาตรฐาน RS-422	7
2.3 โครงสร้างของระบบ และการทำงาน	9
บทที่ 3 การออกแบบซอฟต์แวร์	10
3.1 ขอบเขตของซอฟต์แวร์ ของส่วนประมวลผลข้อมูล	13
3.2 FLOWCHART การทำงานของ SOFTWARE	14
3.3 การนำเอาระบบ SYSTEM MONITORING ไปใช้งาน	22
3.4 แผนผังสมมุติของการนำไปใช้งาน	23
บทที่ 4 การออกแบบ และการทำงานในส่วนต่างๆ ของระบบ	24
4.1 การเชื่อมต่อสัญญาณ	24
4.2 การเชื่อมต่อข้อมูล	25

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

	หน้า
4.2.1 การออกแบบ ANALOG INPUT MODULE	27
4.2.2 การออกแบบ ANALOG OUTPUT MODULE	32
4.2.3 การออกแบบ DIGITAL INPUT MODULE	37
4.2.4 การออกแบบ DIGITAL OUTPUT MODULE	42
4.3 การเปลี่ยนระดับสัญญาณ RS-232 เป็น RS-422	46
เอกสารอ้างอิง	50
ภาคผนวก ก การต่อทั้งระบบของ SYSTEM MONITORING	52
ภาคผนวก ข การใช้งาน SOFTWARE	53
ภาคผนวก ค โปรแกรม	64
ภาคผนวก ง DATA SHEET	128
ประวัติผู้เขียน	157

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 1

บทนำ

1.1 วัตถุประสงค์ของโครงการ

1. เพื่อเป็นต้นแบบในการพัฒนาระบบ System Monitoring ให้มีการควบคุมซับซ้อนขึ้น
2. เพื่อประยุกต์ใช้งานคอมพิวเตอร์ในงานควบคุมอุตสาหกรรม
3. ฝึกทักษะในการใช้งาน ไมโครคอนโทรลเลอร์ เพื่อการควบคุมทางอุตสาหกรรม

1.2 ความเป็นมาของโครงการ

ในระบบงานทางด้านอุตสาหกรรมสมัยก่อน การจัดเก็บข้อมูลในกระบวนการผลิตค่อนข้างจะเป็นปัญหาหลักของช่างโรงงาน เพราะใช้การจดตัวเลขตามเวลาที่กำหนดไว้ ระบบการจัดการข้อมูลการผลิตในระบบอุตสาหกรรม เริ่มมีการพัฒนาขึ้นมาเรื่อย ๆ จากการเดินจดตัวเลขข้อมูลการผลิต พัฒนามาเป็นการใช้เครื่องบันทึกข้อมูล (Recorder) เป็นตัวบันทึกข้อมูลของกราฟ ปัจจุบันระบบคอมพิวเตอร์ได้เริ่มเข้ามามีบทบาททางด้าน การควบคุมการบวนการผลิต ทางด้านอุตสาหกรรมมากยิ่งขึ้น เพราะคอมพิวเตอร์สามารถทำการวิเคราะห์ข้อมูลที่ได้จากกระบวนการผลิตได้รวดเร็ว และแม่นยำกว่าการใช้คนงาน

ระบบคอมพิวเตอร์ ที่ถูกนำเข้ามาควบคุมกระบวนการผลิต ในโรงงานอุตสาหกรรม ปัจจุบัน เช่น ระบบ DCS (Distributed Control System) ซึ่งใช้คอมพิวเตอร์ควบคุมกระบวนการผลิตทั้งหมด ระบบนี้ใช้ส่วนมากในโรงงานอุตสาหกรรมขนาดใหญ่ มีราคาแพง ต้องใช้ผู้เชี่ยวชาญที่ผ่านการอบรมมาเป็นอย่างดี จึงจะสามารถใช้งานระบบนี้ อย่างมีประสิทธิภาพได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โครงการนี้ มีแนวทางการคิดในการออกแบบสร้างระบบการแจ้งเตือน และควบคุมกระบวนการผลิตในอุตสาหกรรมโดยใช้คอมพิวเตอร์การควบคุมที่สามารถทำได้ เป็นการควบคุมแบบ เปิด - ปิด (On - Off Control) การออกแบบมุ่งเน้นใช้อุปกรณ์ที่มีขายในตลาดเมืองไทยปัจจุบัน

โครงการปริญญาโทฉบับนี้ ออกแบบเพื่อการใช้งานจริง ในระบบอุตสาหกรรมขนาดเล็กที่มีระบบการควบคุมไม่ซับซ้อน ทางกลุ่มผู้ออกแบบสร้างเรียกระบบนี้ว่า System Monitoring

1.3 ขอบเขตของโครงการ

1. ใช้เครื่องคอมพิวเตอร์ส่วนบุคคลเป็นตัวควบคุมการทำงานของระบบ
2. ใช้ไมโครคอนโทรลเลอร์ ตระกูล 68HC11 ในการแปลงสัญญาณจากจุดวัดต่าง ๆ เพื่อจัดรูปแบบของข้อมูลให้เป็นรูปแบบมาตรฐานในการรับส่งข้อมูลกับคอมพิวเตอร์
3. สามารถตั้งเวลาในการบันทึกข้อมูลจากจุดวัดได้
4. สามารถรับข้อมูลที่จะบันทึกได้ทั้ง สัญญาณอนาลอก และดิจิตอล
5. สามารถส่งสัญญาณออกเพื่อการควบคุมได้ทั้งสัญญาณอนาลอก และดิจิตอล
6. สามารถแสดงผลข้อมูลที่จัดเก็บในรูปแบบของตัวเลข หรือกราฟได้

1.4 ขั้นตอน และวิธีดำเนินงาน

ขั้นตอนการดำเนินงานของโครงการนี้แบ่งออกเป็น 2 ช่วง การดำเนินการในช่วงที่ 1 จะเป็นการศึกษาถึงความเป็นไปได้ของแนวความคิด การออกแบบที่วางไว้ ดังรายละเอียดต่อไปนี้

1. ศึกษาต้นคว้าข้อมูลในส่วนที่เกี่ยวข้องกับโครงการ

- ระบบการสื่อสารข้อมูลอนุกรมมาตรฐาน RS-232

- ระบบการสื่อสารข้อมูลอนุกรมมาตรฐาน RS-422

- ศึกษาการออกแบบใช้งาน ไมโครคอนโทรลเลอร์ 68HC11 ทั้ง Software

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่นิยมนำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Software และ Hardware

- ศึกษาการทำงานของระบบ ที่ใช้งานได้จริงของต่างประเทศ เพื่อการออกแบบที่เหมาะสม

2. ออกแบบ และทดลองวงจรในแต่ละภาคการทำงาน



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 2

ทฤษฎี และหลักการ

2.1 การสื่อสารข้อมูลอนุกรมมาตรฐาน RS-232

ผู้ที่กำหนดมาตรฐาน RS-232 ขึ้นเมื่อกว่า 20 ปีมาแล้ว คือ สถาบันแห่งหนึ่งในสหรัฐอเมริกาซึ่งเป็นที่ยอมรับกัน มีชื่อว่า Electronic Industrials Association (EIA) โดยกำหนดเป็นมาตรฐานสำหรับการเชื่อมต่อคอมพิวเตอร์ ซึ่งมาตรฐานนี้เป็นการส่งข้อมูลแบบอซิงโครนัส เป็นที่ยอมรับของบริษัทผู้ผลิตคอมพิวเตอร์เป็นส่วนใหญ่

มาตรฐาน RS-232

มาตรฐานการสื่อสาร RS-232 ถูกพัฒนาขึ้นมาเพื่อรองรับการสื่อสารข้อมูลผ่านทางชุมสายตัวพทโดยโมเด็ม ซึ่งโมเด็มจะทำการเปลี่ยนระดับแรงดัน " 0 " และ " 1 " ไปเป็นความถี่สูงและต่ำ เพื่อสามารถส่งไปตามสายโทรศัพท์ได้ ส่วนโมเด็มที่ปลายทางก็จะทำแปลงความถี่สูง และต่ำให้กลับเป็นระดับแรงดัน " 0 " และ " 1 "

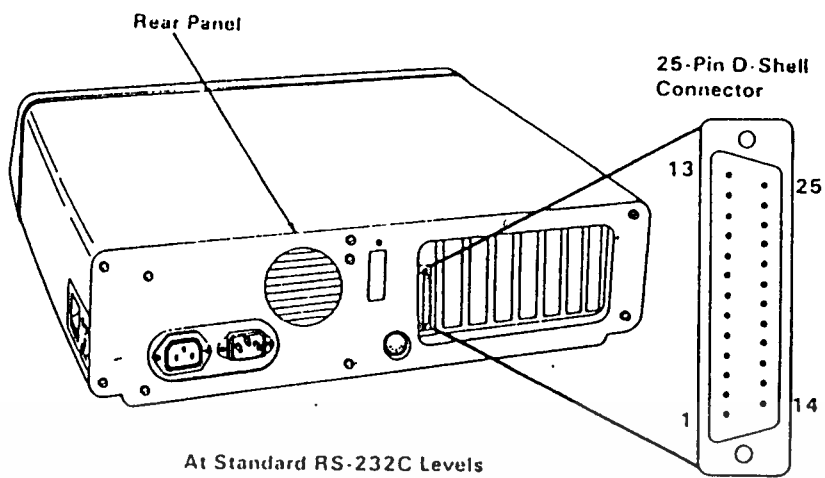
ในปี 1980 โมโครคอมพิวเตอร์ , เทอร์มินอล ต่างก็ใช้มาตรฐาน RS-232 เชื่อมต่อกันโดยตรง โดยไม่ใช้โครงข่ายโทรศัพท์ หรือ โมเด็ม ยกเว้นการควบคุมระยะไกล

รูปที่ 1 เป็นการแสดงการเชื่อมต่อโมโครคอมพิวเตอร์ หรือเทอร์มินอล เข้ากับโมเด็มตามรูป จะพบว่ามิกรราวด์อยู่ 2 เส้น เส้นแรกเป็นกราวด์ของตัวถัง จะทำการต่อเข้ากับโดยตรงกับซิลด์ของระบบ กราวด์เส้นนี้จะต่อระหว่างอุปกรณ์ต้นทาง และปลายทางเท่านั้น ในกรณีที่ต้องการความปลอดภัยในการเชื่อมต่อ กราวด์อีกเส้นเป็นจุดที่ใช้อ้างอิง สำหรับสัญญาณเส้นอื่นๆ

RS-232 มีข้อเสียคือ ไม่สามารถส่งข้อมูลโดยใช้สายในระยะทางที่ไกล ๆ ได้ เพราะ

ปัญหาเรื่อง กราวด์รูป RS-232 จะสามารถส่งข้อมูลได้ไกลเพียง 50 ฟุต

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



At Standard RS-232C Levels
(With Exception of Current Loops)

Description	Pin
NC	1
Transmitted Data	2
Received Data	3
Request to Send	4
Clear to Send	5
Data Set Ready	6
Signal Ground	7
Received Line Signal Detector	8
Transmit Current Loop Data	9
NC	10
Transmit Current Loop Data	11
NC	12
NC	13
NC	14
NC	15
NC	16
NC	17
Receive Current Loop Data	18
NC	19
Data Terminal Ready	20
NC	21
Ring Indicator	22
NC	23
NC	24
-Receive Current Loop Return	25

External Device

Asynchronous Communications Adapter (RS-232C)

Note: To avoid inducing voltage surges on interchange circuits, signals from interchange circuits shall not be used to drive inductive devices, such as relay coils.

Connector Specifications

รูปที่ 1 แสดงการเชื่อมต่อเทอร์มินอล กับไมโครคอลโทรเลอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สัญญาณที่สำคัญของ RS-232

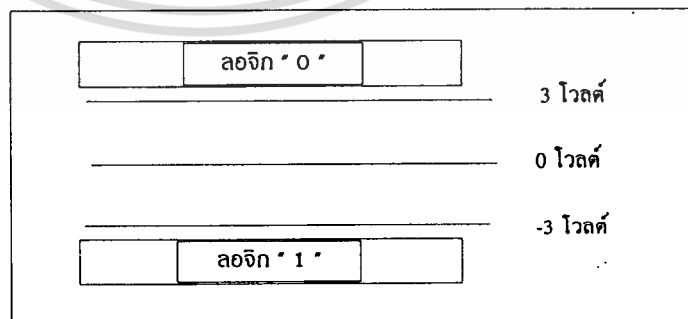
- CLR TO SEND และ REQ TO SEND เมื่อต้นทางต้องการส่งข้อมูลไปยังปลายทาง ต้นทางจะส่งสัญญาณไปที่ REQ TO SEND เมื่อปลายทางพร้อมที่จะรับข้อมูลก็จะตอบกลับมาที่ CLR TO SEND พร้อมๆ กับเริ่มส่งข้อมูลจากต้นทางไปปลายทางทันที

- DATA SET READY และ DATA TERM READY สัญญาณ DATA SET READY เป็นตัวแสดงว่าไม่เต็มกำลังทำงาน ส่วน DATA TERM READY เป็นตัวแสดงความพร้อมสำหรับคอมพิวเตอร์ หรือ เทอร์มินัล

- CARRIER DETECT และ RING INDICATOR สัญญาณ CARRIER DETECT เป็นสัญญาณที่แสดงว่า ปลายทางยังถูกเชื่อมต่ออยู่ ส่วนสัญญาณ RING INDICATOR จะแสดงเมื่อมีเครื่องโทรศัพท์เข้ามาในตู้สาย

ในปัจจุบัน RS-232 จะมีใช้กันเป็นมาตรฐาน ในเครื่องไมโครคอมพิวเตอร์ ซึ่ง มีตัวอย่างการเชื่อมต่อ (ดังรูปด้านล่าง) ซึ่งระดับแรงดัน จะแตกต่างกันไปจากระดับแรงดัน ทีทีแอล และซีมอส ระดับแรงดัน

RS-232 จะสมมาตรกันเมื่อเทียบกับกราวด์ โดยแรงดันมากกว่า 3 โวลต์ ลอจิก " 0 " และ มากกว่า -3 โวลต์ สำหรับลอจิก " 1 " เครื่องไมโครคอมพิวเตอร์จะใช้ระดับแรงดันของ RS-232 +-12 โวลต์



รูปแสดงการสวิงแรงดันสูง ๆ ของ RS-232 เพื่อลดสัญญาณรบกวนในระบบการสื่อสาร

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

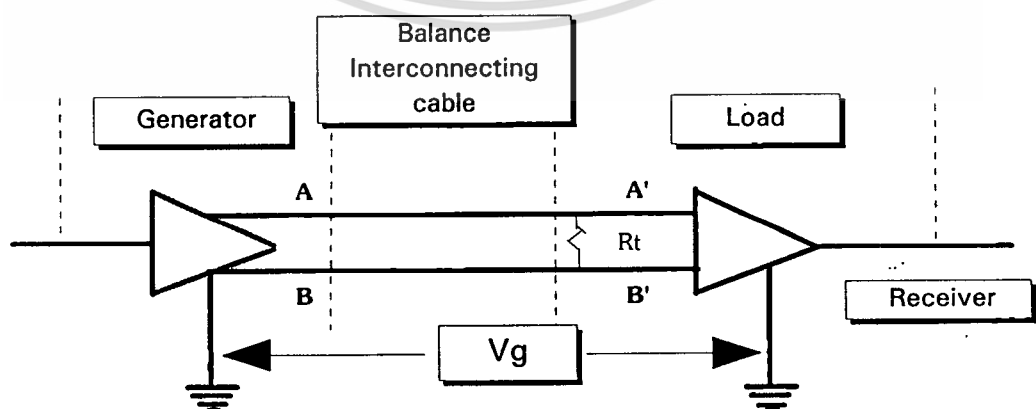
2.2 การสื่อสารข้อมูลอนุกรมมาตรฐาน RS-422

เมื่อไม่นานมานี้เอง EIA ก็ได้ออกแบบมาตรฐานชนิดใหม่ โดยการพัฒนามาจากของเดิม เพื่อใช้งานแทน RS-232 ในส่วนที่ขีดจำกัดของ RS-232 โดยได้ออกแบบให้ RS-422 มีความสามารถในการส่งข้อมูลได้ระยะทางไกล ๆ คือ สามารถส่งได้ไกลถึง 4,000 ฟุต และความเร็วในการส่งข้อมูลสูงถึง 10 ล้านบิตต่อวินาที (10 Mbps) ความสามารถสูงเช่นนี้ เพราะใช้การแยกสายสัญญาณเป็น 2 เส้น ต่อ 1 สัญญาณ โดยไม่ใช้กราวด์ร่วม เรียกว่าเป็นระบบสมดุล อีกทั้งยังใช้สัญญาณที่ +5 โวลต์ จึงไม่จำเป็นต้องใช้ไฟ 12 โวลต์เพิ่ม เหมือน RS-232

ที่ความเร็ว 90 Kbps ยอมให้ส่งได้ไกลเต็มที่ 4,000 ฟุต ความเร็ว 2 Mbps ที่ระยะทาง 200 ฟุต และที่ความเร็ว 10 Mbps ที่ 35 ฟุต

ในการส่งสัญญาณแบบนี้ ขณะที่สัญญาณเส้นหนึ่งมีระดับสัญญาณ 5 โวลต์ อีกเส้นที่คู่กัน จะมีระดับแรงดัน 0 โวลต์ สัญญาณ RS-422 จะแบ่งเป็น 1 คู่สายต่อ 1 สัญญาณ เพื่อลดค่าอิมพีแดนซ์ในสาย ทำให้ผลของความไวต่อสัญญาณรบกวนมีน้อยมาก

ถึงแม้ RS-422 จะมีข้อดีเหนือกว่า RS-232 ก็ตาม แต่ยังมีควมนิยมกันน้อย จะมีใช้กันในโรงงานอุตสาหกรรมเป็นส่วนมาก เพราะข้อดีในด้านความเร็วในการรับส่งข้อมูล และความสามารถในการลดสัญญาณรบกวนของ RS-422



รูปแสดงโมเดลของวงจรกำเนิดสัญญาณ RS-422

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากโมเดลที่ภาคเจเนอเรเตอร์ มีหน้าที่กำเนิดแรงดันความแตกต่าง ระหว่าง 2-6 โวลต์ ส่งผ่านทางสายเคเบิล พิจารณาทางลอจิก ที่ลอจิก 1 ที่จุด A ของภาคเจเนอเรเตอร์ จะมีค่าเป็นลบเมื่อเทียบกับจุด B ส่วนที่ลอจิก "0" จุด A จะมีค่าเป็นบวกเมื่อเทียบกับจุด B

ส่วนที่ภาค Receiver ก็จะทำหน้าที่แปลงแรงดันแตกต่างที่รับเข้ามา เป็นระดับแรงดันดิจิทัล

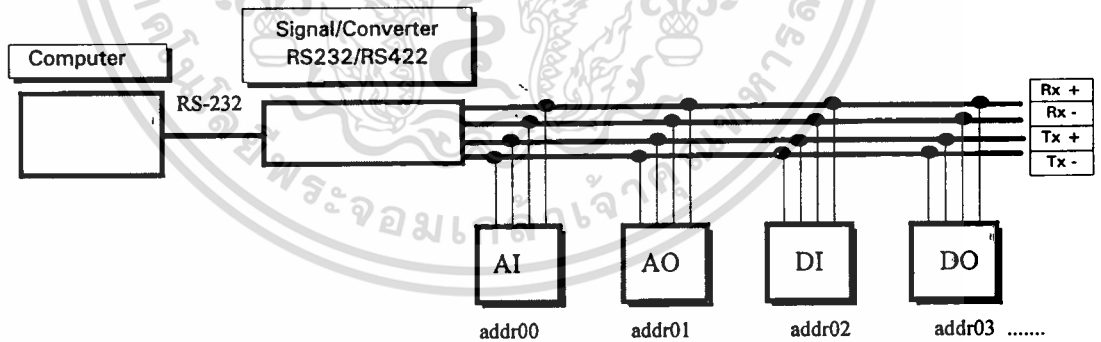
Parameter	Condition	Min	Max	Units
Vo Driver Unloaded Output Voltage			6	V.
Vo Driver Loaded Output	Rt = 100 Ohm.	2		V.
Vt Voltage				
Vt Driver Output Rs Resistance	Per output		50	Ohm.
Ios Driver Output Short Circuit Current Driveroutput risetime	Vo = 0 V.		150	mA.
Iox Driver Power OFF Current	-0.25 < Vo < 6 V.		±100	UA.
Vth Receiver Sensitivity	Vcm = + 7 V.		200	mA
Vcm Receiver Common mode Voltage		-12	12	V.
Receiver Input Offset		± 3		V.



2.3 โครงสร้างของระบบ และการทำงาน

โครงสร้างการทำงานของ System Monitoring แบ่งออกเป็น 3 ส่วน ดังต่อไปนี้

1. ส่วนการประมวลผลข้อมูล ในส่วนการทำงานนี้ จะใช้คอมพิวเตอร์ส่วนบุคคล เป็นตัวประมวลผลข้อมูล และสั่งการควบคุมระบบ ที่สามารถใช้โปรแกรม Microsoft Windows ได้
2. ส่วนการเชื่อมต่อสัญญาณ ส่วนการทำงานนี้จะทำหน้าที่เปลี่ยนระดับสัญญาณ จากมาตรฐาน RS-232 เป็น RS-422 เพื่อเพิ่มระยะทางในการสื่อสารข้อมูล และเพื่อรองรับ การเชื่อมต่อแบบ Multidrop ที่ใช้ในโรงงานนี้
3. ส่วนการเชื่อมต่อกับข้อมูล ส่วนการทำงานนี้มีหน้าที่ในการรับข้อมูลจากจุดวัด ซึ่งวัดได้ทั้งสัญญาณอนาลอก และสัญญาณดิจิทัล เมื่อข้อมูลมาแล้วก็ทำการแปลงให้อยู่ใน รูปแบบมาตรฐานที่พร้อมจะส่งให้กับคอมพิวเตอร์ได้ อีกหน้าที่หนึ่งของส่วนการทำงานนี้คือ การส่งข้อมูลควบคุมให้กับจุดที่จะทำการควบคุม โดยสัญญาณที่ส่งออกนี้ สามารถส่งได้ทั้ง สัญญาณอนาลอก และสัญญาณดิจิทัล ไปยังจุดควบคุมได้



บล็อกไดอะแกรม แสดงระบบของ System Monitoring

บทที่ 3

การออกแบบซอฟต์แวร์

การออกแบบซอฟต์แวร์ที่ใช้ในระบบ SYSTEM MONITORING

ในระบบนี้ซอฟต์แวร์สามารถแยกออกได้เป็น 2 ส่วนคือ

- 1.ซอฟต์แวร์ที่ใช้ในส่วนประมวลผลข้อมูล ซึ่งจะแสดงผลภายใต้ Microsoft Windows
- 2.ซอฟต์แวร์ที่ใช้ในส่วนเชื่อมต่อข้อมูล

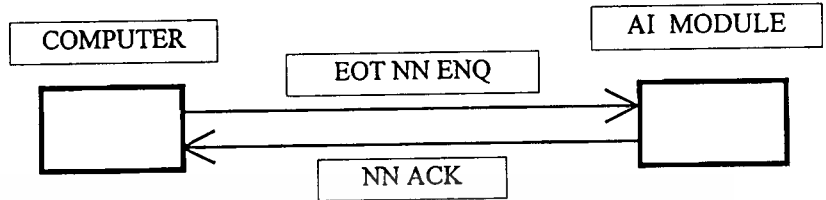
ก่อนการเริ่มเขียนซอฟต์แวร์ ต้องทำการกำหนดรูปแบบของการติดต่อสื่อสารกันระหว่างส่วนเชื่อมต่อข้อมูล กับส่วนประมวลผลข้อมูล ก่อน ซึ่งเรียกว่า โปรโตคอล ซึ่งโปรโตคอลของแต่ละโมดูลมีดังนี้

โปรโตคอลการสื่อสารแต่ละโมดูล

STX	02H
ETX	03H
EOT	04H
ENQ	05H
ACK	06H
Sp	20H

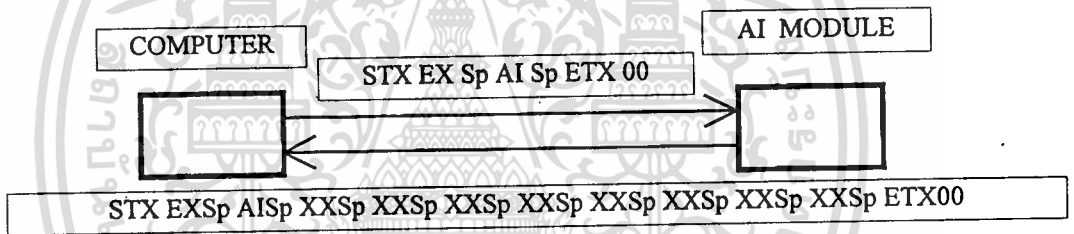
โปรโตคอลของ Analog Input Module

การเริ่มต้นการสื่อสาร



NN : คือ รหัส ASC II ของหมายเลขโมดูล มีค่า 00 - 31

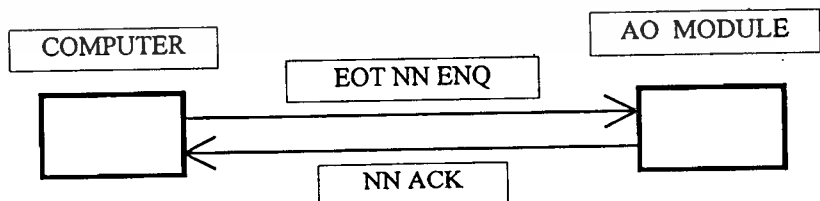
การขอข้อมูลจาก AI โมดูล



XX : คือ ตัววัดของแต่ละช่อง เป็นรหัส ASC II ของตัวเลขตั้งแต่ 00 - FF

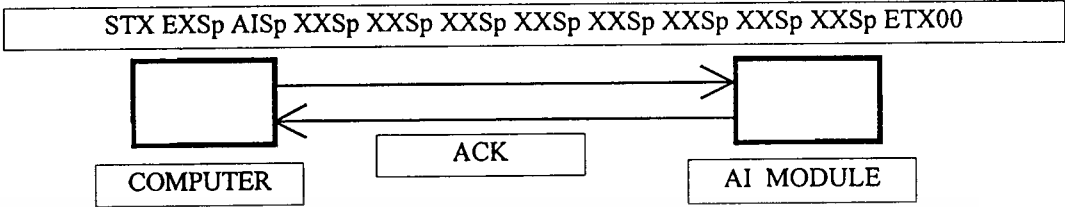
โปรโตคอลของ Analog Output Module

การเริ่มต้นการสื่อสาร



NN : คือ รหัส ASC II ของหมายเลขโมดูล มีค่า 00 - 31

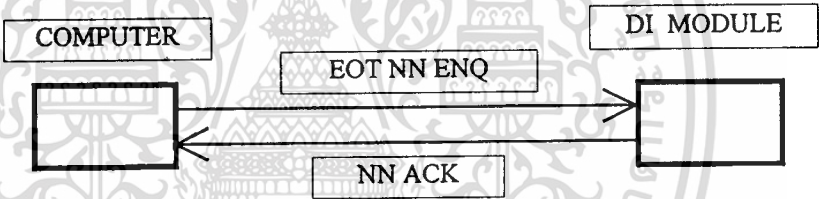
การส่งข้อมูลไปยัง AO โมดูล



XX : คือ ค่าเอาต์พุตที่ส่งไปยัง AO โมดูล เป็นรหัส ASC II ตั้งแต่ 00 - FF

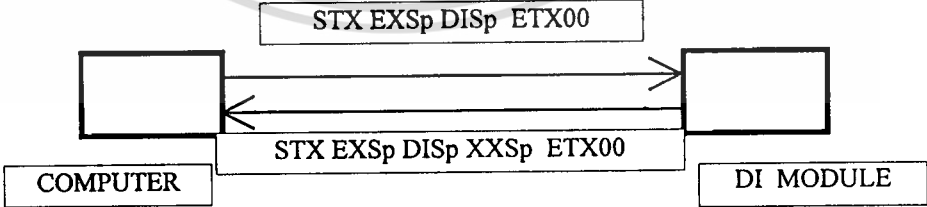
ไปต่อดอลของ Digital Input Module

การเริ่มต้นการสื่อสาร



NN : คือ รหัส ASC II ของหมายเลขโมดูล มีค่า 00 - FF

การขอข้อมูลจาก DI โมดูล



XX : คือ รหัส ASC II ของอินพุตที่เข้ามา

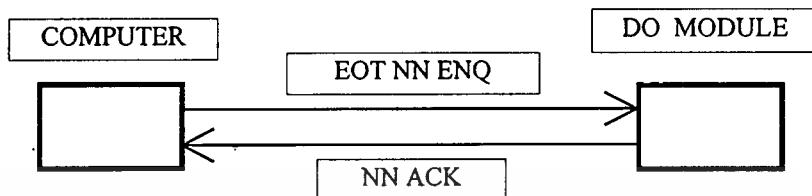
XX = X2 X1

X1 บิตที่ 0 , 1 , 2 , 3 แทนอินพุตหมายเลข 0 , 1 , 2 , 3

X2 บิตที่ 4 , 5 , 6 , 7 แทนอินพุตหมายเลข 4 , 5 , 6 , 7

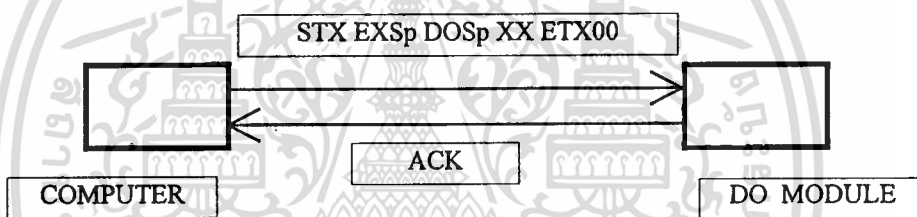
โปรโตคอลของ Digital Output Module

การเริ่มต้นการสื่อสาร



NN : คือ รหัส ASC II ของหมายเลขโมดูล มีค่า 00 - 31

การส่งข้อมูลไปยัง DO โมดูล



XX : คือ รหัส ASC II ของเอาต์พุตที่เข้ามา

XX = X2 X1

X1 บิตที่ 0 , 1 , 2 , 3 แทนอินพุตหมายเลข 0 , 1 , 2 , 3

X2 บิตที่ 4 , 5 , 6 , 7 แทนอินพุตหมายเลข 4 , 5 , 6 , 7

3.1 ขอบเขตของซอฟต์แวร์ของส่วนประมวลผลข้อมูล

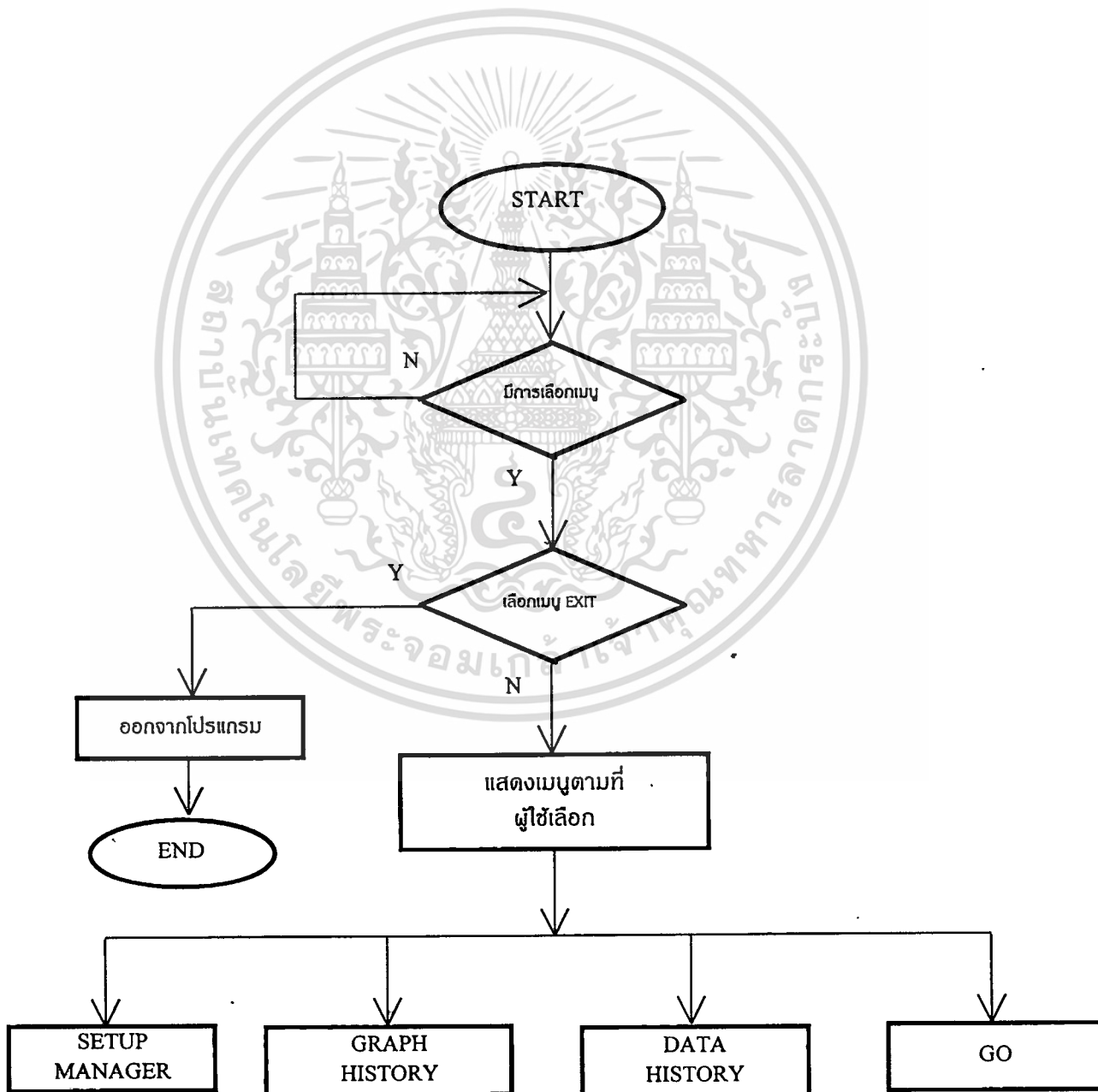
- แสดงผลภายใต้ Microsoft Windows 3.1
- สามารถแสดงข้อมูลเป็นตัวเลข และกราฟได้ 100 ช่อง
- สามารถตั้งค่าการแจ้งเตือนได้ทั้งต่ำกว่าและสูงกว่า
- สามารถเก็บข้อมูลเป็นไฟล์ได้
- สามารถนำข้อมูลที่เก็บไว้มาแสดงดูภายหลังได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- สามารถพิมพ์กราฟได้
- สามารถแก้ไขสเกลของค่าวัดได้

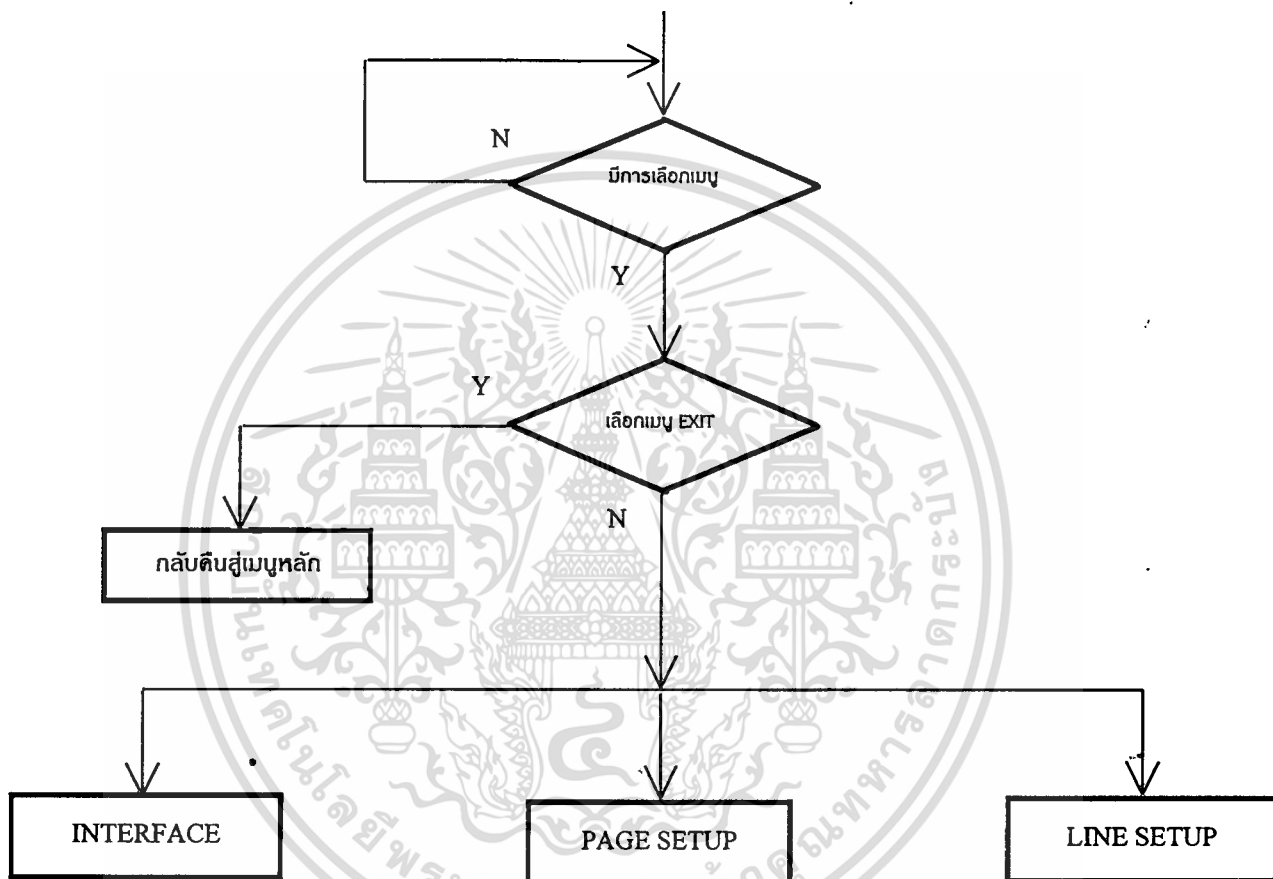
3.2 FLOWCHART การทำงานของ SOFTWARE

MAIN โปรแกรม



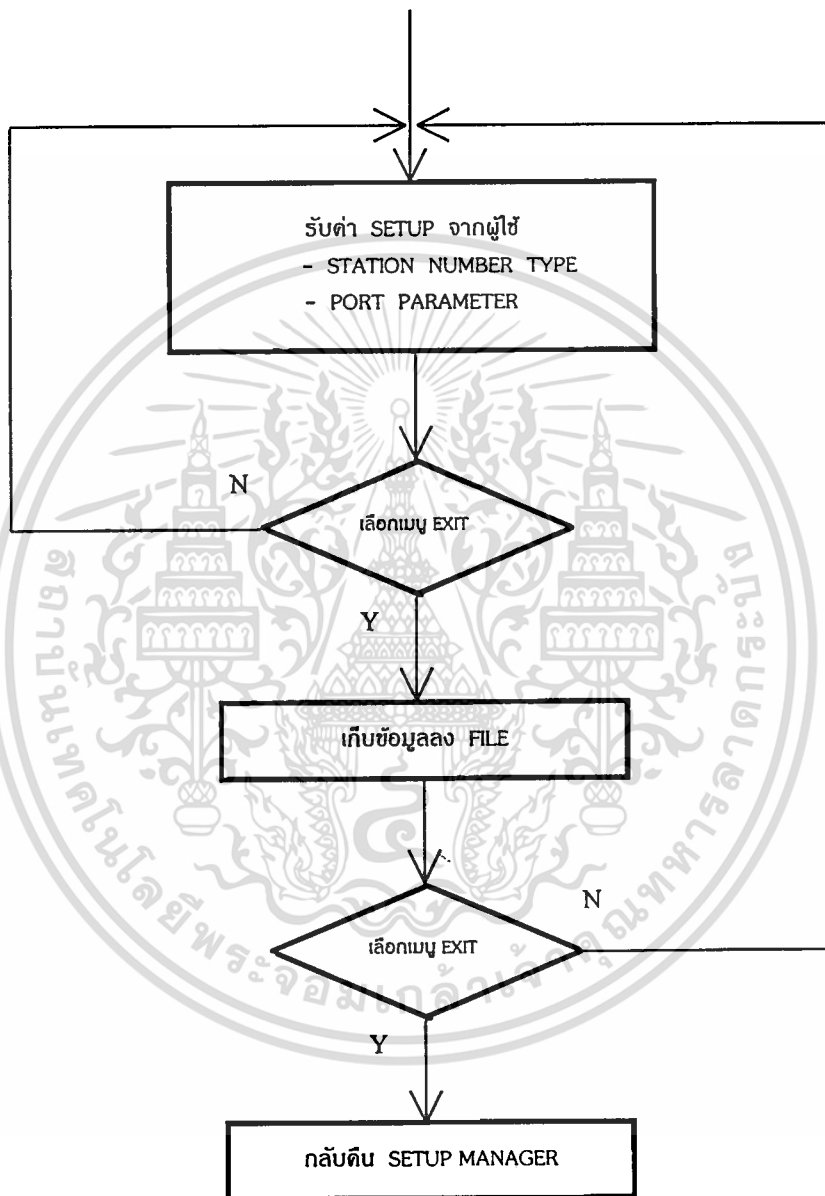
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

FLOWCHART ของ SETUP MANAGER



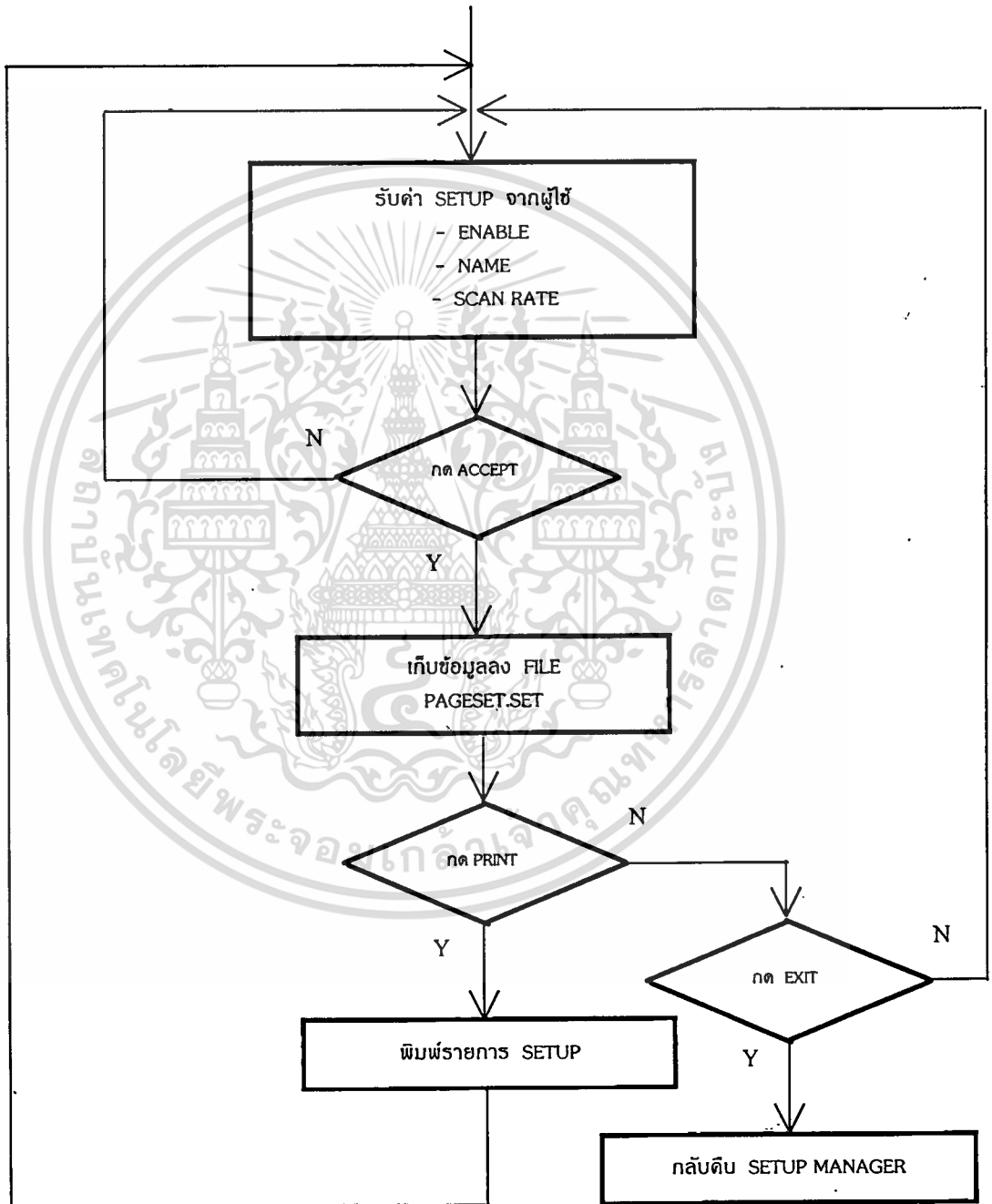
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

FLOWCHART ของ INTERFACE



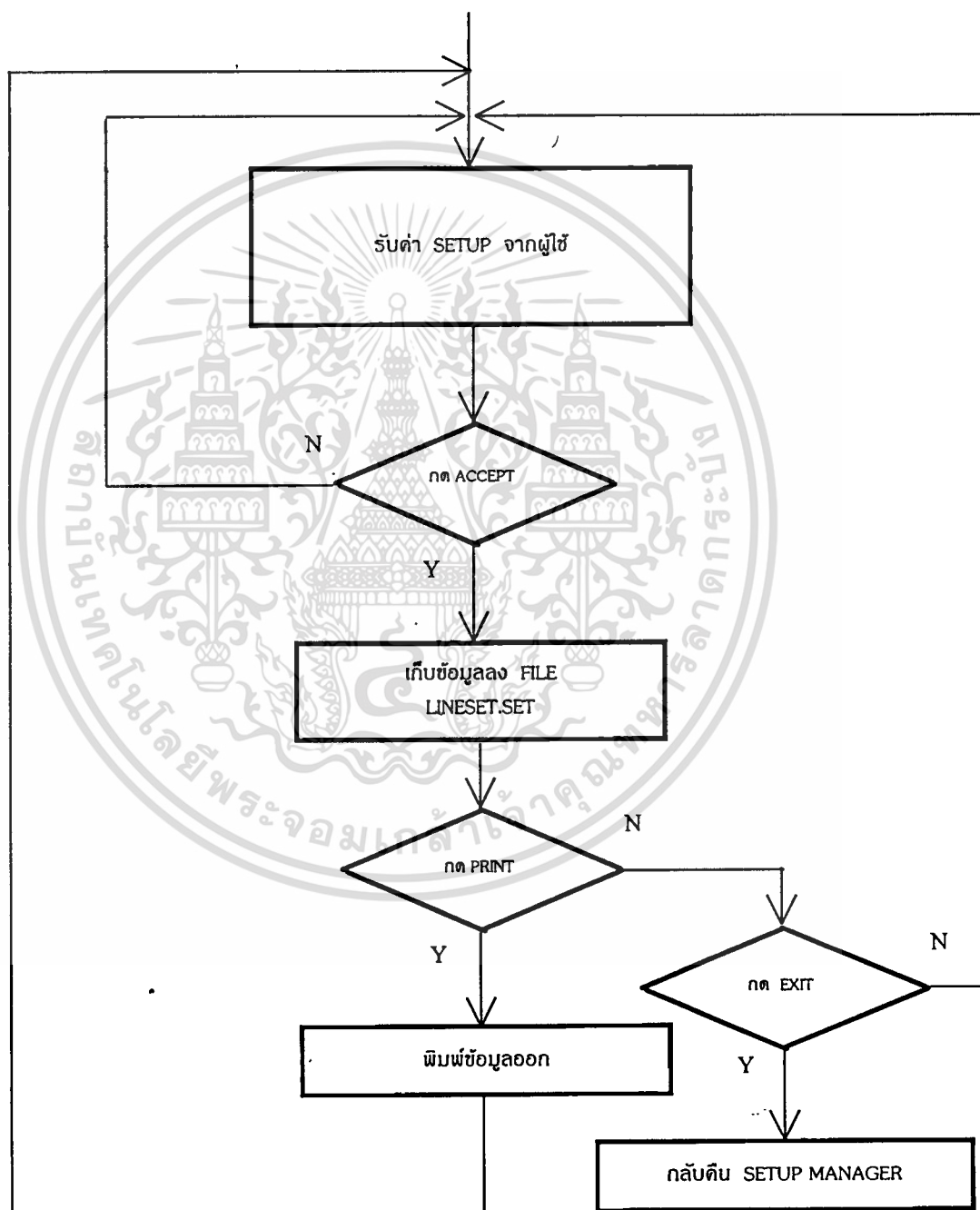
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

FLOWCHART ของ PAGE SETUP



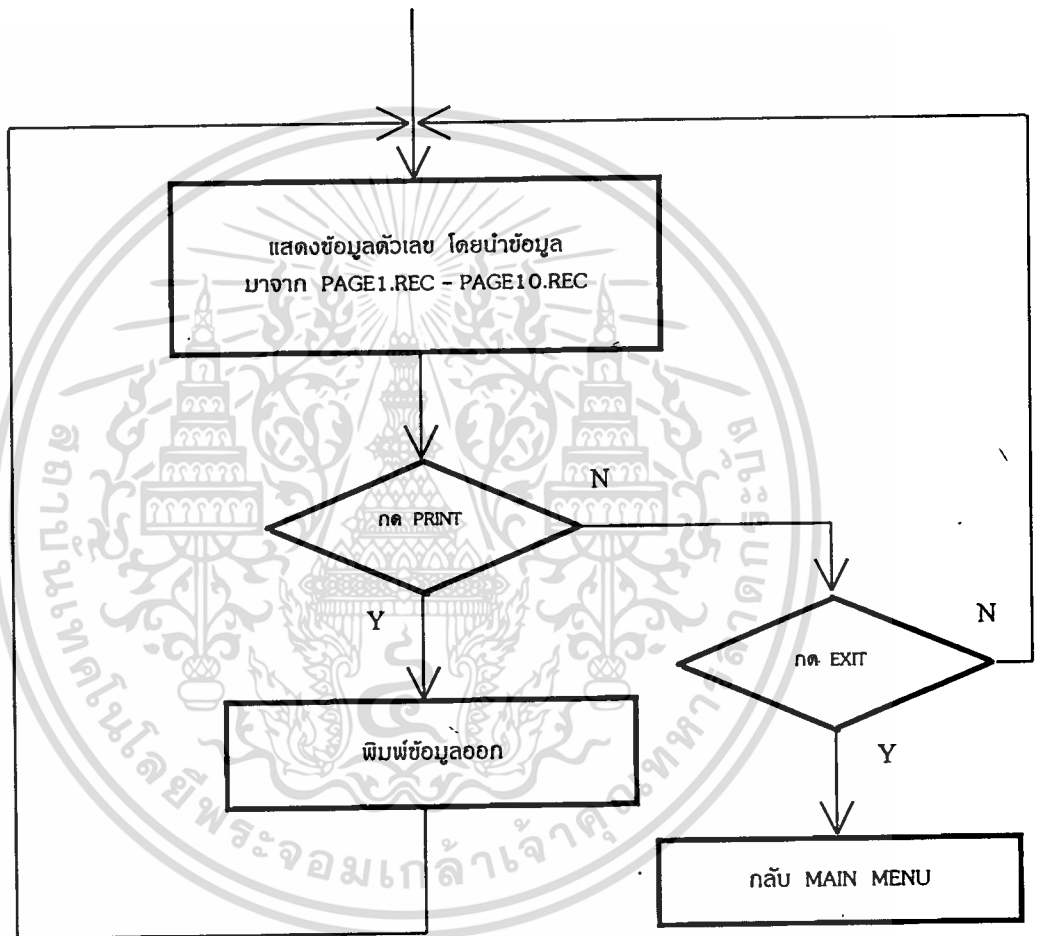
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

FLOWCHART การทำงานของ LINE SETUP



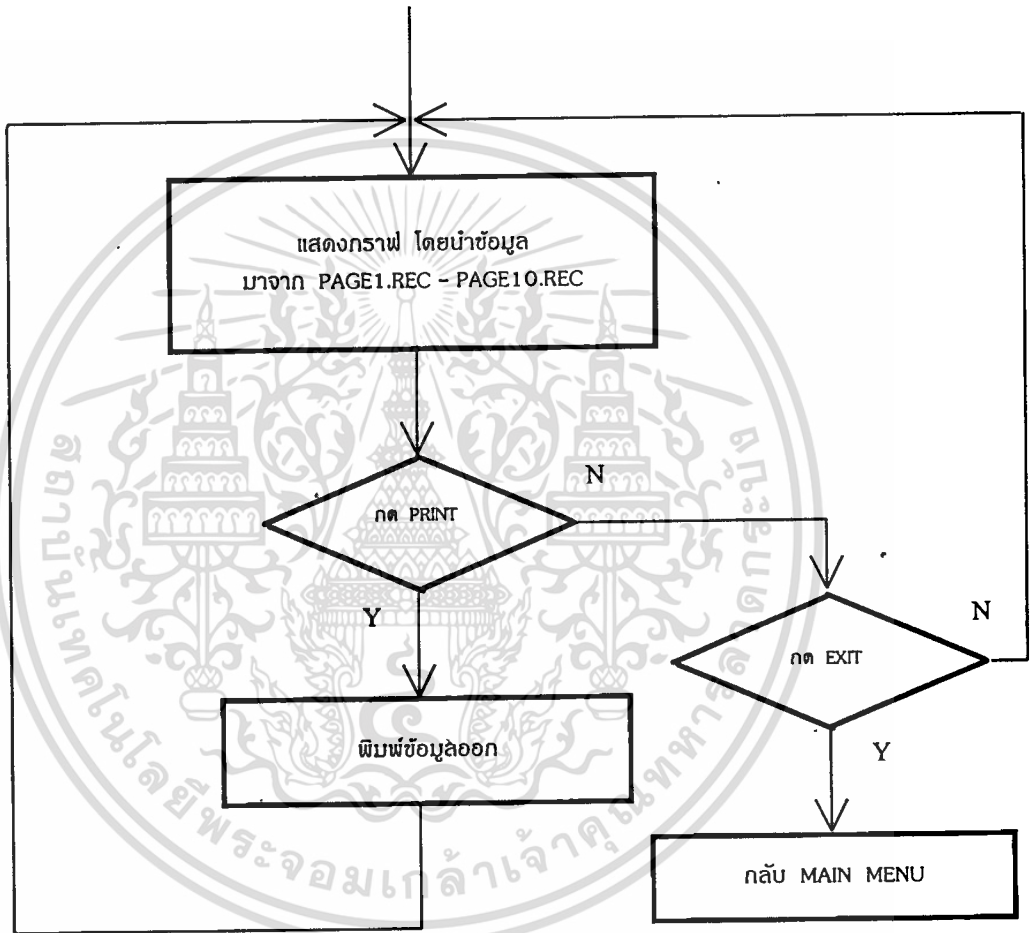
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

FLOWCHART ของ DATA HISTORY



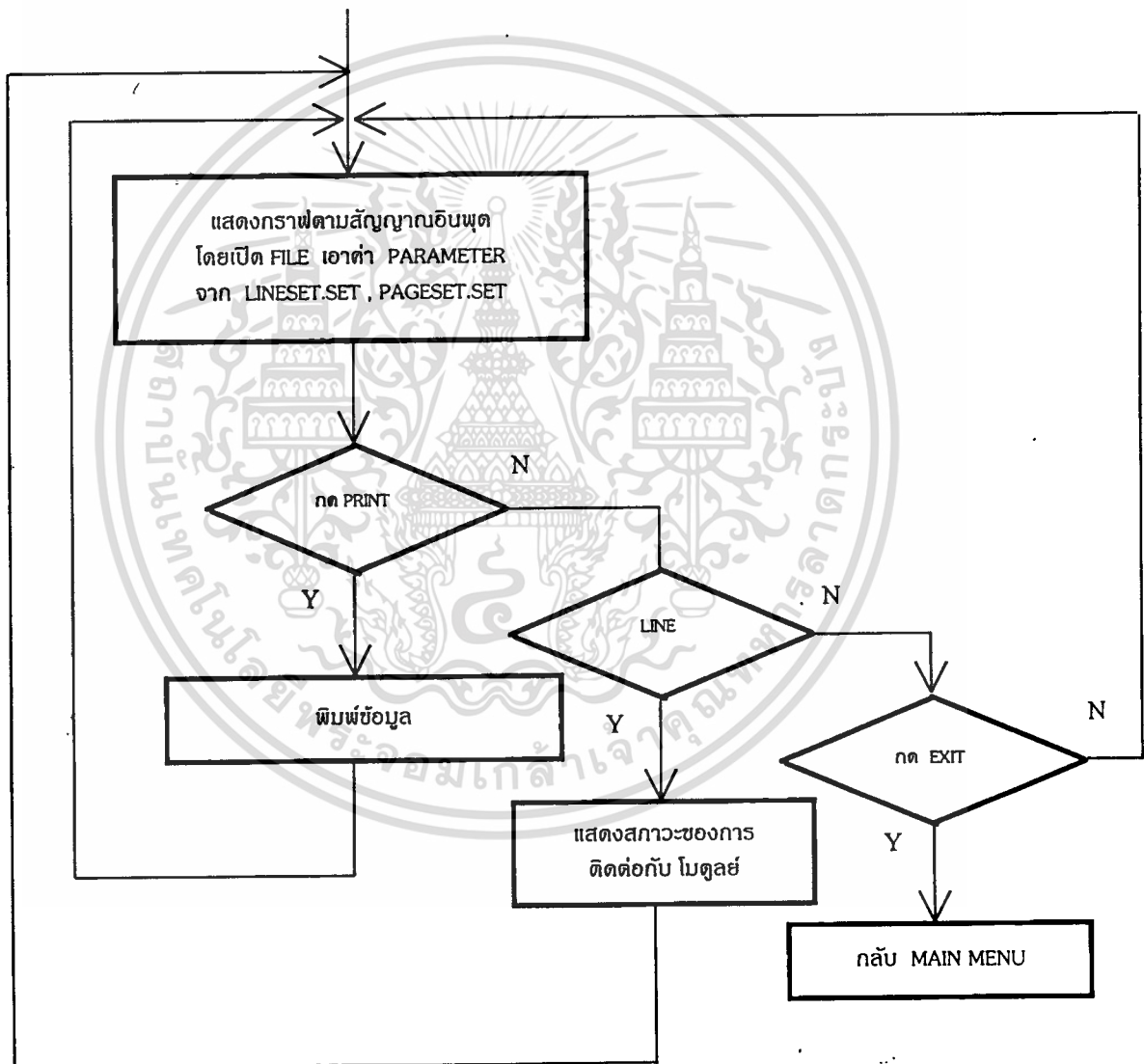
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

FLOWCHART ของ GRAPH HISTORY



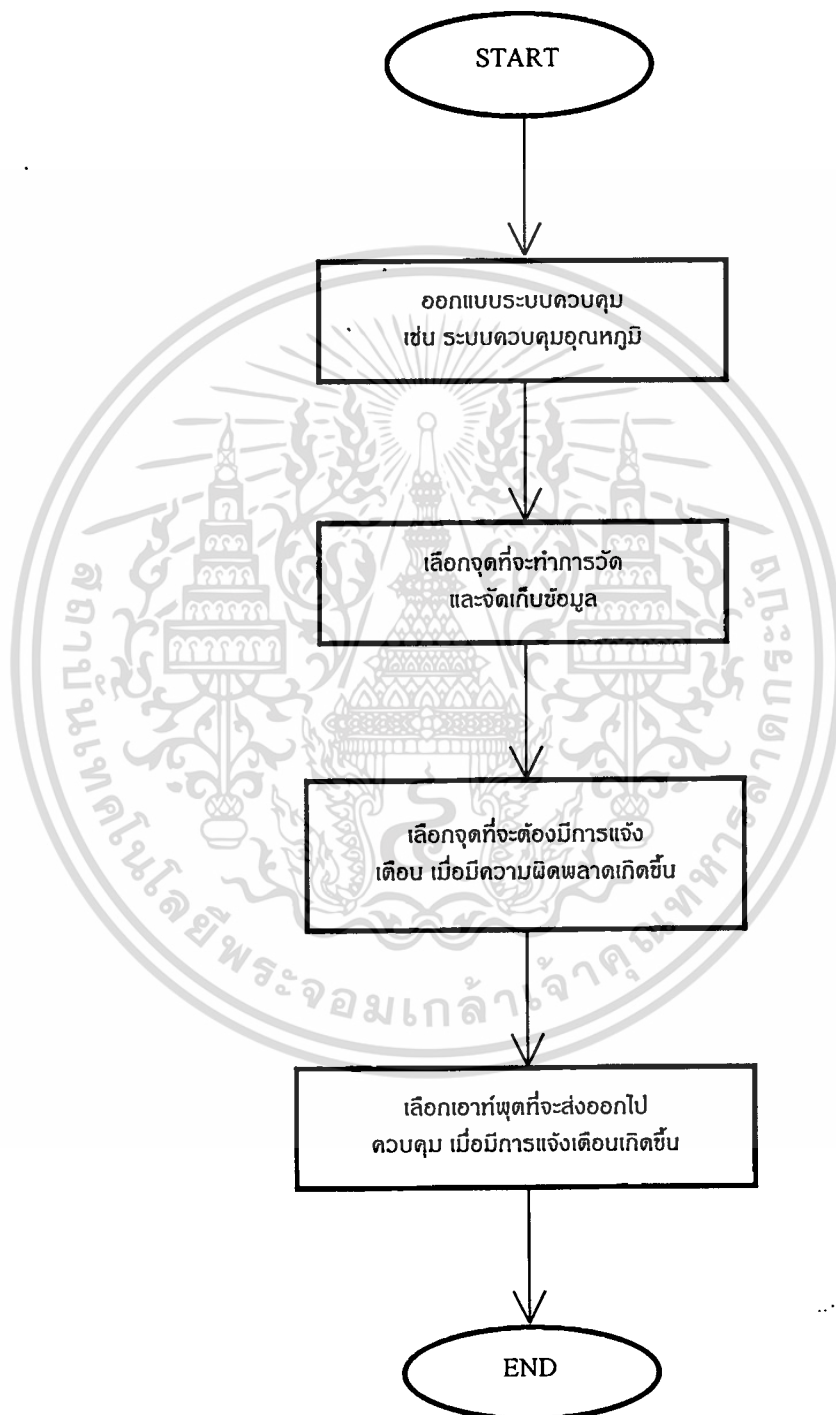
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

FLOWCHART ของ GO



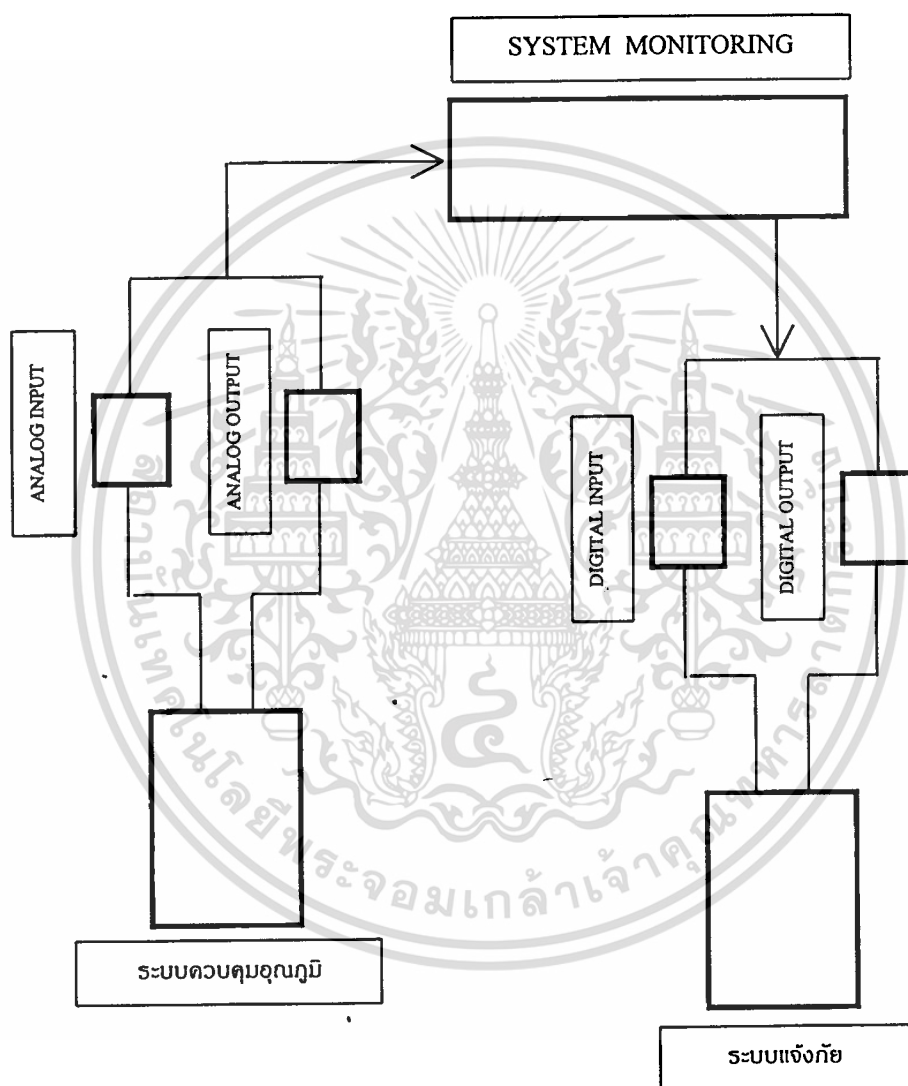
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.3 การนำระบบ SYSTEM MONITORING ไปใช้งาน



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.4 แผนผังสมมุติ ของการนำไปใช้งาน



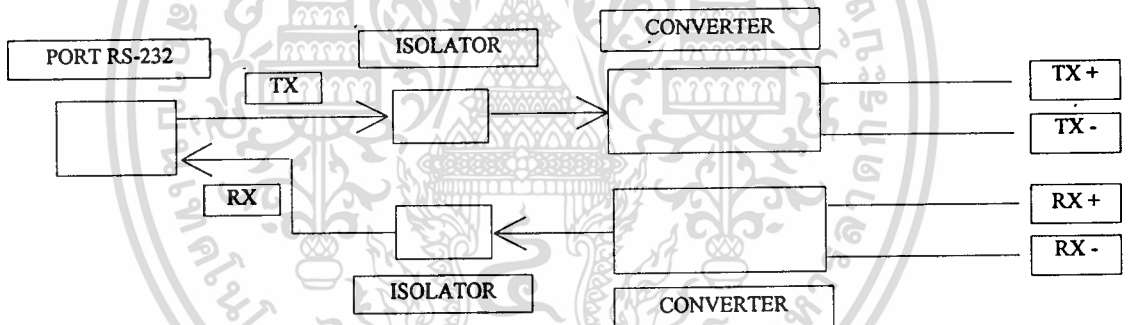
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 4

การออกแบบ และการทำงานในส่วนต่าง ๆ ของระบบ

4.1 ส่วนของการเชื่อมต่อสัญญาณ

หน้าที่การทำงานของส่วนนี้คือ เปลี่ยนระดับสัญญาณระหว่างมาตรฐาน RS-232 และ RS-422 บล็อกไดอะแกรมของส่วนนี้ ดังแสดงรูปด้านล่าง



รูปแสดงบล็อกไดอะแกรมของส่วนการเชื่อมต่อสัญญาณ

สัญญาณ TX จาก Prot RS-232 จะถูกส่งผ่านทาง Isolator เพื่อทำการแยกสัญญาณคอมพิวเตอรื ออกจากสัญญาณของระบบ เพื่อป้องกันสัญญาณรบกวน เมื่อสัญญาณ TX ผ่านออกมาจาก Isolator ก็จะถูกส่งเข้าไปยัง Converter เพื่อทำหน้าที่แปลงสัญญาณเป็น RS-422 ส่งออกไปยังบัซของระบบ

ทางด้านรับสัญญาณ RX+ และ RX- จากบัซของระบบ จะถูกส่งเข้าไปยัง Converter เพื่อแปลงจากสัญญาณ RS-422 เป็น TTL จากนั้นก็ถูกส่งไป Isolator เพื่อทำการแยกเอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เพื่อทำการแยกสัญญาณ RX จาก Isolator ก็จะถูกส่งเข้าสู่ Prot RS-232

การทำงานของวงจร

วงจรของภาคการทำงานนี้ แสดงดังรูปด้านล่าง สัญญาณ TX จากคอมพิวเตอร์ จะถูกส่งผ่านมายัง D1 ผ่าน R1 ป้อนเข้าสู่ O1 เบอร์ 4N26 ซึ่งทำหน้าที่เป็นไอโซเลเตอร์ สัญญาณที่ผ่านออกจาก O1 จะถูกป้อนเข้าสู่ IC1 เบอร์ MC3487 ของโมโตโรล่า ซึ่งทำหน้าที่แปลงสัญญาณจาก TTL ไปเป็นสัญญาณมาตรฐาน RS-422

IC2 ทำหน้าที่เปลี่ยนสัญญาณมาตรฐาน RS-422 เป็นสัญญาณ TTL สัญญาณเอาต์พุตจาก IC2 จะถูกป้อนเพื่อขับ Q1BC547 สัญญาณ RX ที่ผ่านออกจากไอโซเลเตอร์ จะถูกส่งเข้าไปยังวงจรที่ทำหน้าที่เปลี่ยนระดับแรงดัน TTL เป็นแรงดันมาตรฐาน RS-232 ซึ่งวงจรนี้ประกอบด้วย Q2 , R8 , R9 , R10 , C1 , D5 , D6

ภาคเพาเวอร์ซัพพลายของภาคการทำงานนี้ จะแบ่งเป็น 2 ชุด คือ ชุดแรกจ่ายให้กับวงจรส่วนที่ใช้เปลี่ยนแรงดันจาก TTL เป็น RS-422 ชุดที่ 2 จ่ายให้กับวงจรเปลี่ยนแรงดัน TTL เป็น RS-232

4.2 ส่วนของการเชื่อมต่อข้อมูล

ในระบบ System Monitor มีโมดูลย์ที่ทำหน้าที่รับส่งข้อมูลต่างๆ ระหว่างจุดวัด และจุดควบคุมต่างๆ ดังต่อไปนี้

1. Analog Input Module
2. Analog Output Module
3. Digital Input Module
4. Digital Output Module

คุณสมบัติของแต่ละโมดูลย์

Analog Input Module

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับใช้เพื่อการศึกษาเท่านั้น เมื่ออนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.2.1 การออกแบบ ANALOG INPUT MODULE

- สัญญาณอินพุตอยู่ในช่วง 0-5 Vdc
- ความละเอียดของ A/D 8 บิต
- ติดต่อสื่อสารแบบอนุกรมมาตรฐาน RS-422

Analog Output Module

- ให้สัญญาณเอาต์พุตแบบอนาลอกได้ 8 ช่อง
- สัญญาณเอาต์พุตอยู่ในช่วง 0-5 Vdc
- ความละเอียดของ D/A 8 บิต
- ติดต่อสื่อสารแบบอนุกรมมาตรฐาน RS-422

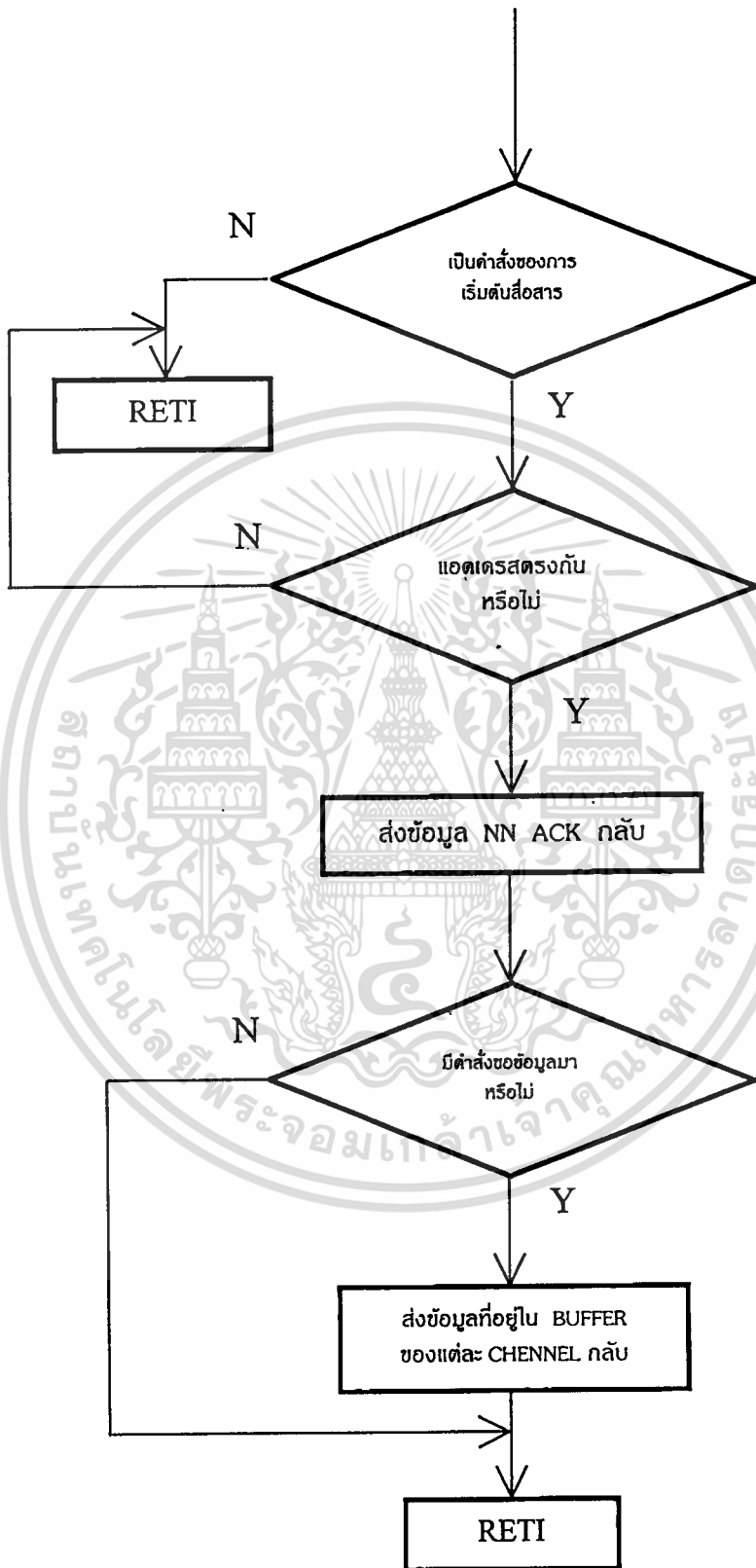
Digital Input Module

- รับสัญญาณอินพุตแบบดิจิตอลได้ 8 ช่อง
- ติดต่อสื่อสารแบบอนุกรมมาตรฐาน RS-422

Digital Output Module

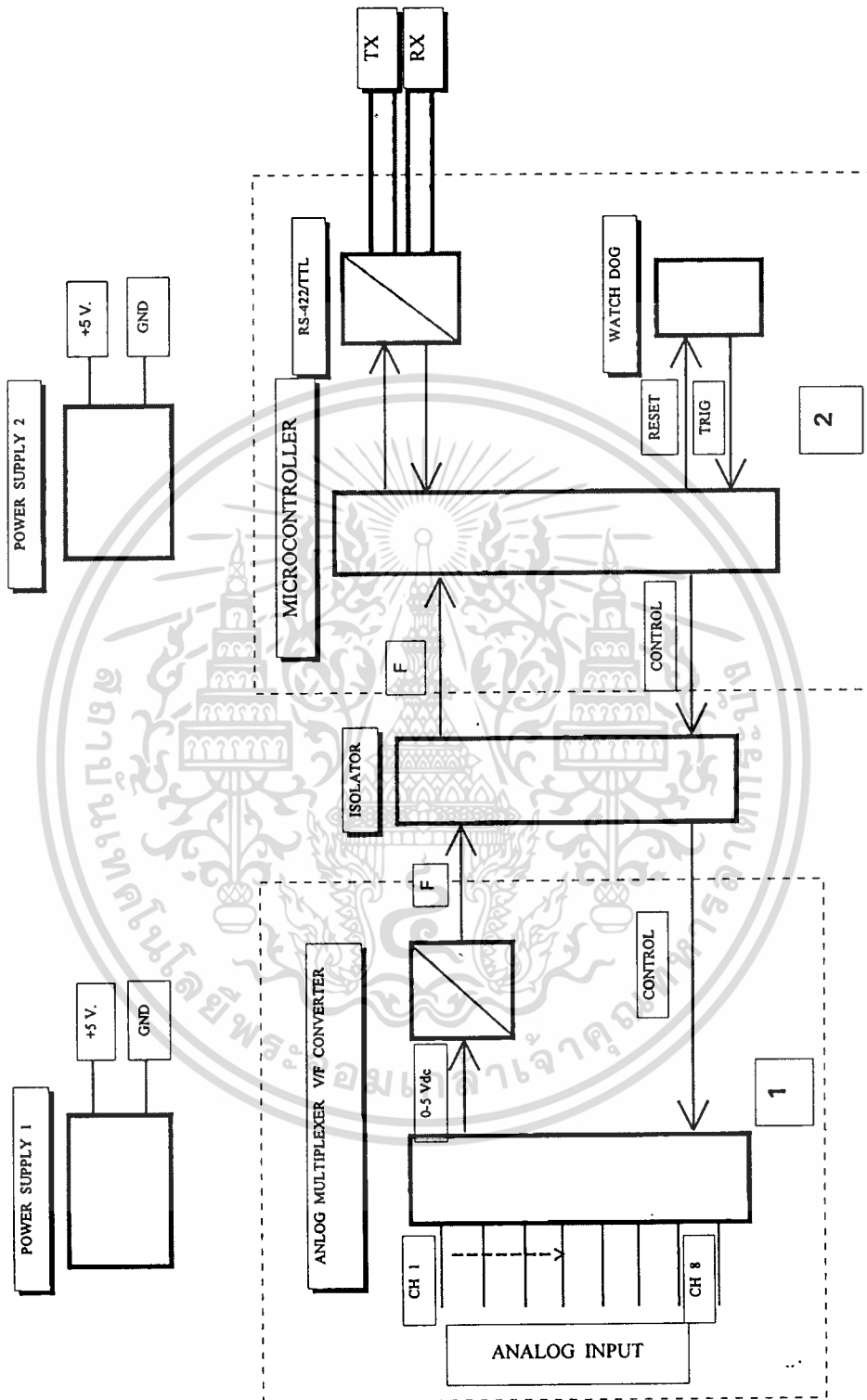
- ให้สัญญาณเอาต์พุตเพื่อใช้ในการควบคุมแบบดิจิตอลได้ 8 ช่อง
- ติดต่อสื่อสารแบบอนุกรมมาตรฐาน RS-422

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



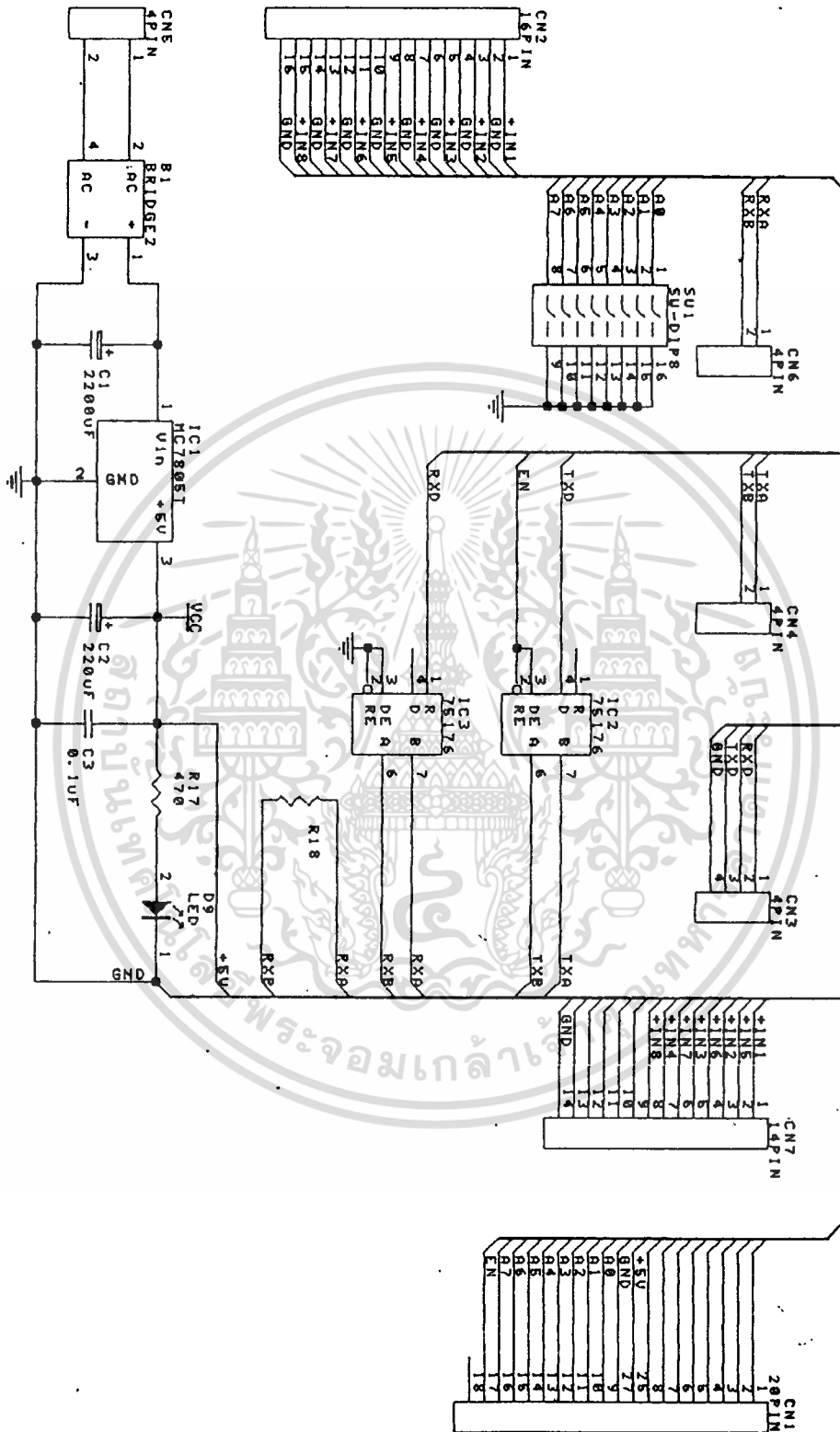
Interrupt Service Routing ของ AO

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



ANALOG INPUT MODULE (AI)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้拿去ใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



วงจรของ ANALOG INPUT MODULE (AI)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การทำงานของ AI

การเปลี่ยนสัญญาณอนาล็อกของ AI จะใช้ A/D ภายในตัวของ CPU โดยต่อสัญญาณตรงเข้าไปที่ AD0 - AD7 มีความละเอียด 8 บิต

สัญญาณเอาต์พุตจาก MUX จะถูกส่งต่อมายัง V/F คือส่วนที่ทำหน้าที่เปลี่ยนแรงดันอินพุตที่รับเข้ามาให้เป็นสัญญาณความถี่ส่งออกไป

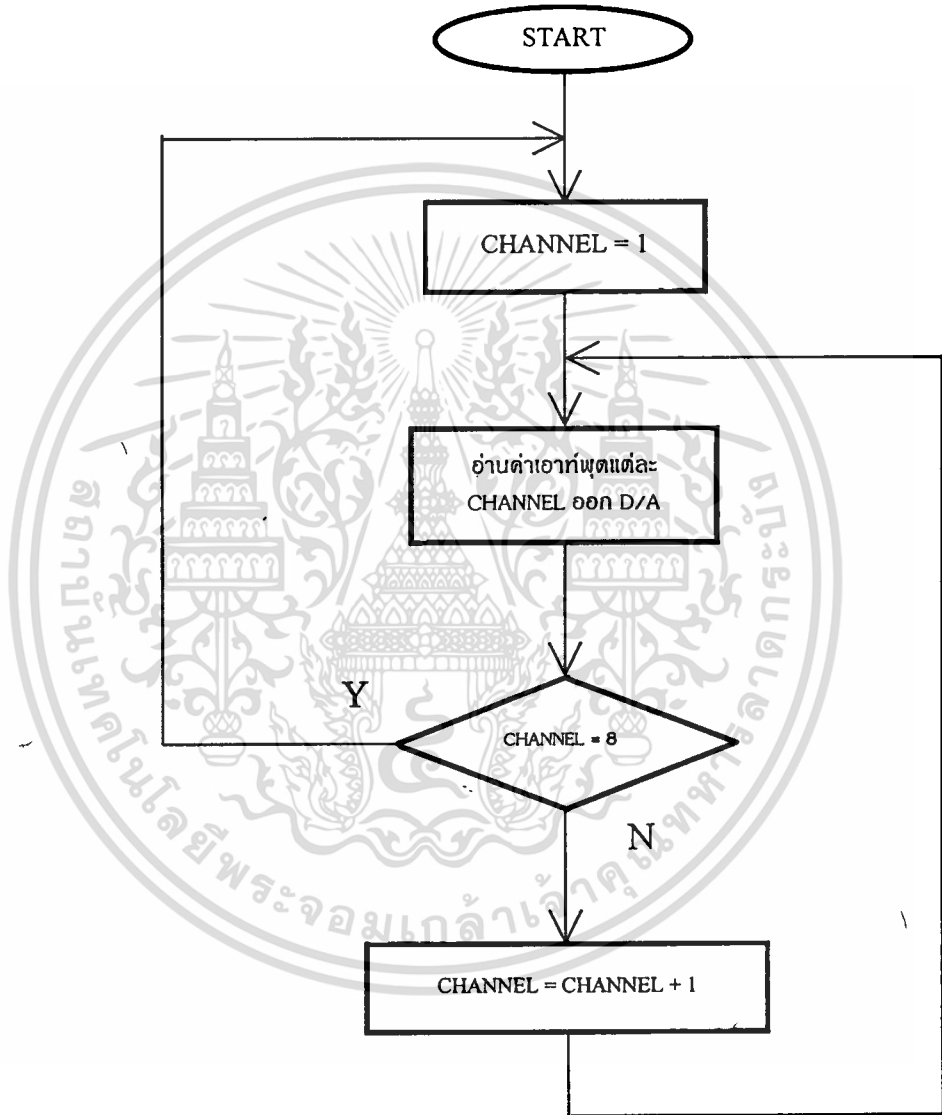
ไอโซเลเตอร์ (Isolator) ทำหน้าที่เชื่อมโยงสัญญาณทางแสง ระหว่าง ภาคการทำงานส่วนหน้า กับภาคส่วนหลัง คือตัวไมโครคอนโทรลเลอร์ 8 บิต สาเหตุที่ต้องใช้ไอโซเลเตอร์ เพื่อป้องกันการรบกวนกันระหว่าง จุดที่วัดสัญญาณ กับตัวไมโครคอนโทรลเลอร์ 8 บิต

ไมโครคอนโทรลเลอร์ 8 บิต ทำหน้าที่ควบคุมการทำงานของ AI ทั้งหมด ตั้งแต่การเลือกสัญญาณที่จะวัด , การวัดความถี่จาก V/F , การรับส่งข้อมูล กับเครื่องคอมพิวเตอร์ และการควบคุมเวลาในการรีเซ็ต Watch Dog

TTL/RS-422 ทำหน้าที่เปลี่ยนระดับแรงดันจาก TTL ให้เป็น RS-422 ตอนส่ง และเปลี่ยนจาก RS-422 ตอนรับ เป็น TTL ตอนรับ

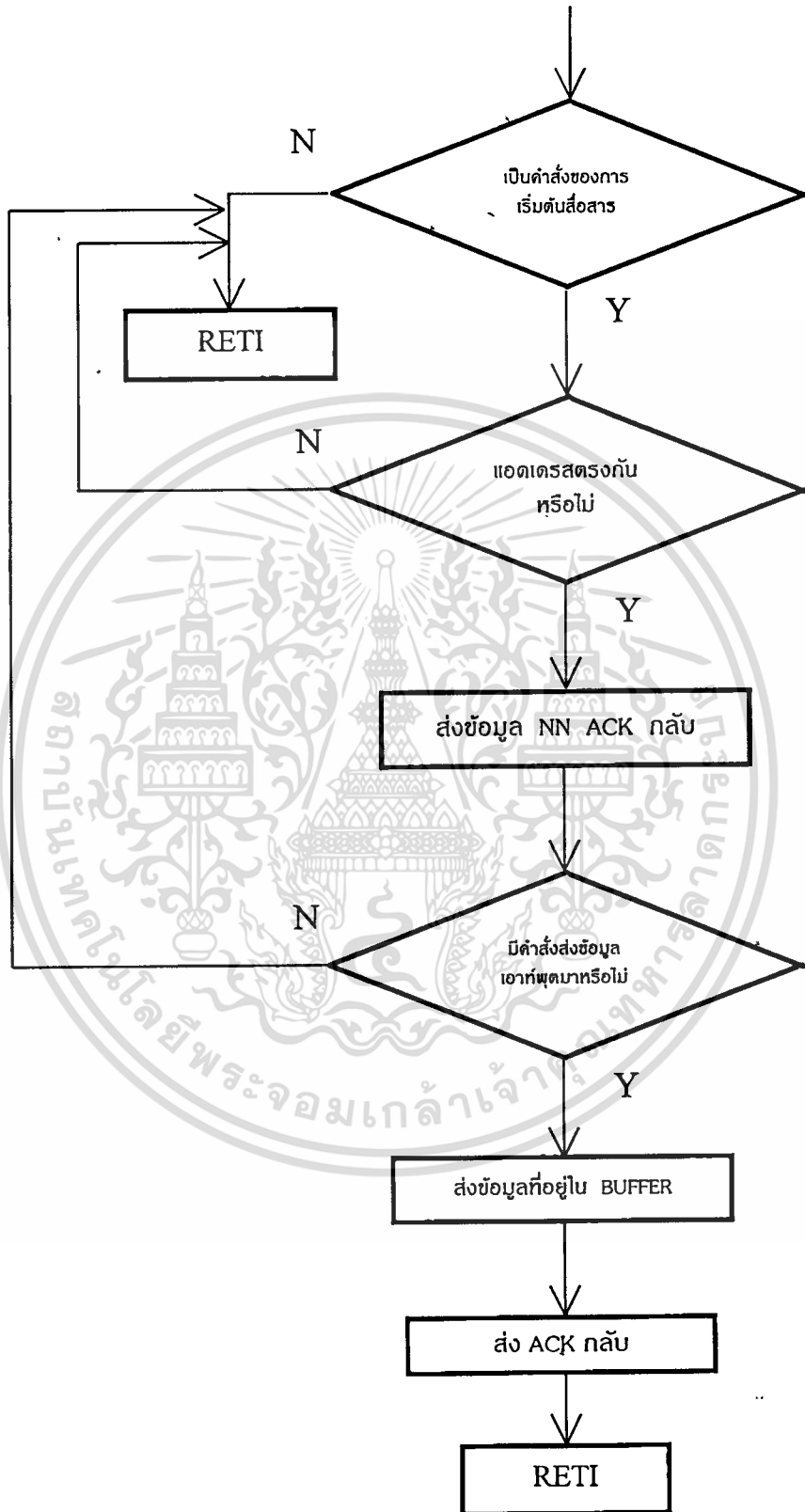
Power Supply ทำหน้าที่จ่ายแรงดัน +5 โวลต์ เลี้ยงวงจร แยกเป็น 2 ชุด เพื่อจ่ายให้กับวงจรส่วนที่ 1 กับส่วนที่ 2

4.2.2 การออกแบบ ANALOG OUTPUT MODULE



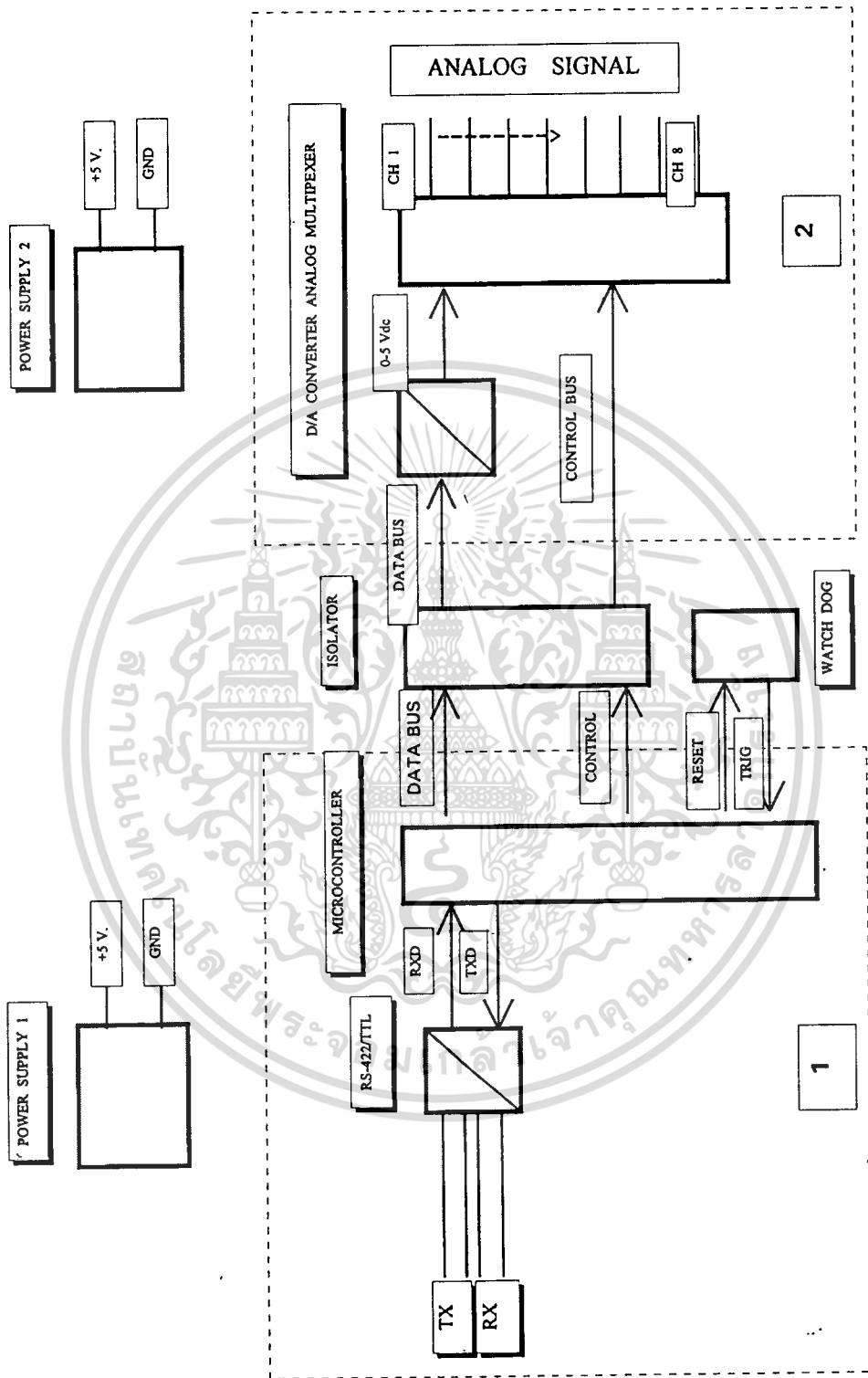
Flowchart ของ AO

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



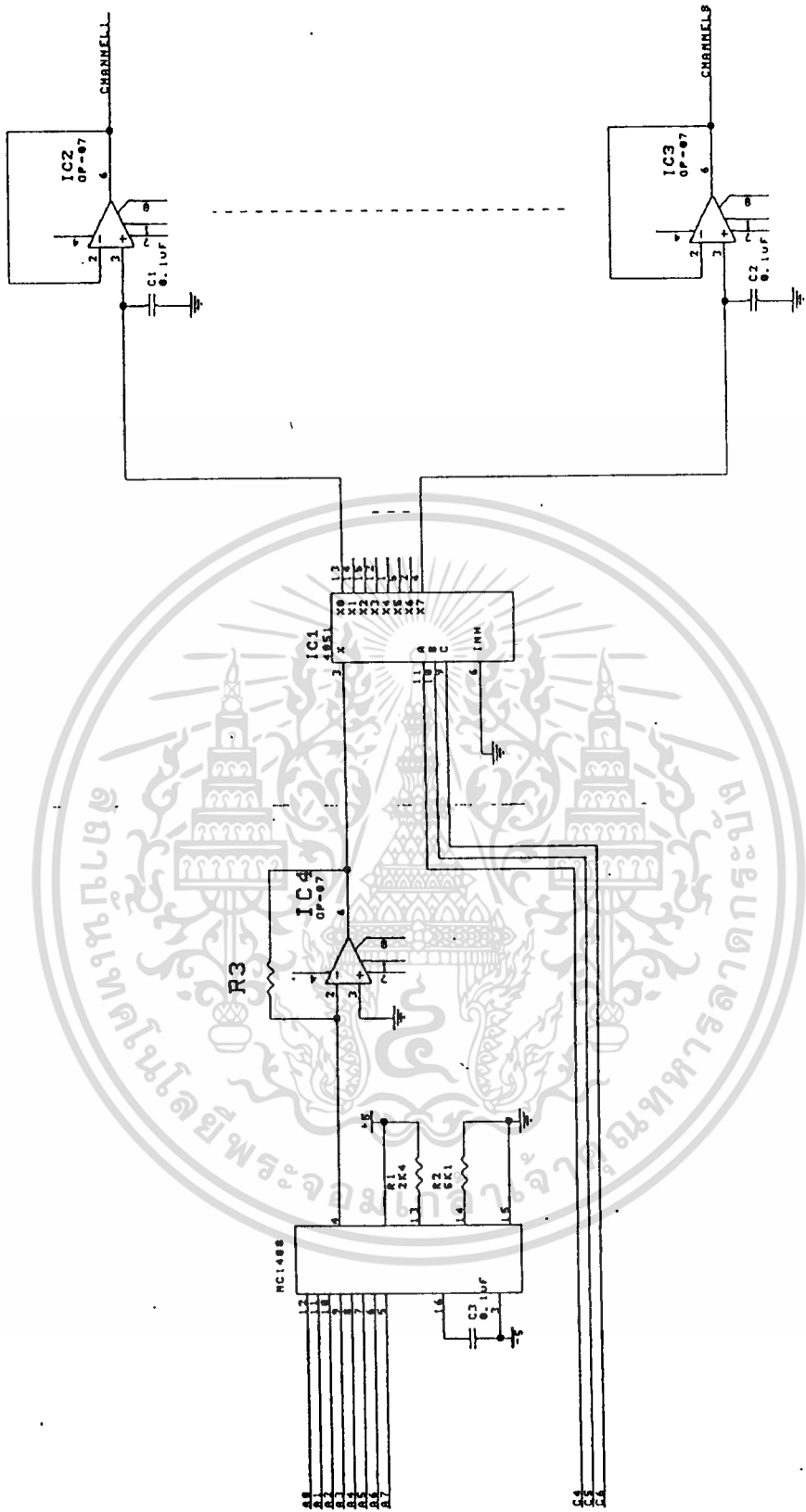
Interrupt Service Routine ของ AO

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



ANALOG OUTPUT MODULE (AO)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้拿去ใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



วงจรของ ANALOG OUTPUT MODULE (AO)

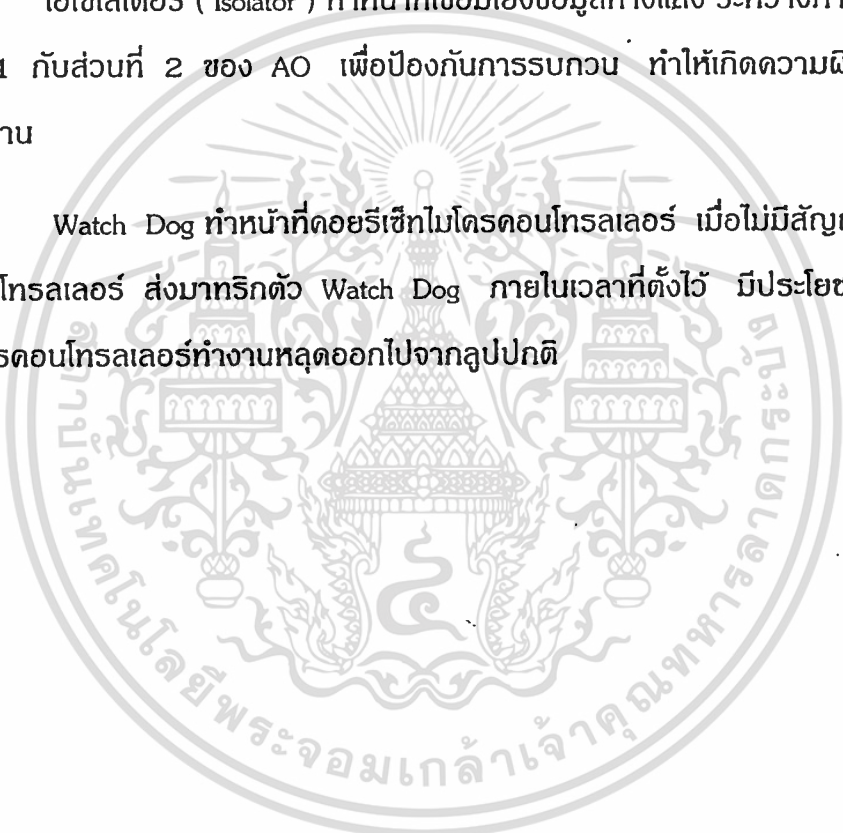
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้拿去ใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การทำงานของ AO

AO จะใช้ D/A เบอร์ MC1408 ในการเปลี่ยนจาก ดิจิตอล เป็นสัญญาณอะนาล็อก สัญญาณออกมาจาก MC1408 จะถูกป้อนเข้าวงจร อะนาล็อกมัลติเพล็กซ์ เบอร์ 4051 เพื่อทำการจ่ายให้กับวงจรเอาต์พุตทั้ง 8 บิต โดยมี IC เบอร์ 08-07 ทำหน้าที่เป็น Buffer เพื่อจ่ายให้กับ Load

ไอโซเลเตอร์ (Isolator) ทำหน้าที่เชื่อมโยงข้อมูลทางแสง ระหว่างภาคการทำงาน ส่วนที่ 1 กับส่วนที่ 2 ของ AO เพื่อป้องกันการรบกวน ทำให้เกิดความผิดพลาดขึ้นในการทำงาน

Watch Dog ทำหน้าที่คอยรีเซ็ตไมโครคอนโทรลเลอร์ เมื่อไม่มีสัญญาณจากไมโครคอนโทรลเลอร์ ส่งมาทริกตัว Watch Dog ภายในเวลาที่ตั้งไว้ มีประโยชน์ในการป้องกันไมโครคอนโทรลเลอร์ทำงานหลุดออกไปจากลูปปกติ



4.2.3 การออกแบบ DIGITAL INPUT MODULE

บล็อกไดอะแกรมของ Digital Input Module แสดงดังรูป

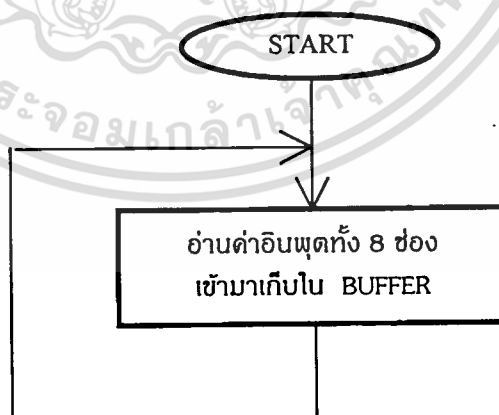
Isolator ทำหน้าที่เชื่อมโยงสัญญาณไฟฟ้าโดยทางแสง เพื่อแยกส่วนระหว่างจุดวัด และโมดูลย์ ออกจากกัน เพื่อป้องกันการรบกวนกัน

Microcontroller ทำหน้าที่ประมวลผลจากสัญญาณอินพุต และทำหน้าที่ติดต่อสื่อสารกับส่วนประมวลผลข้อมูล เพื่อส่งข้อมูลไปให้ทำการวิเคราะห์

RS-422/TTL ทำหน้าที่เปลี่ยนระดับสัญญาณ TTL เป็น RS-422 และเปลี่ยนจาก RS-422 เป็น TTL

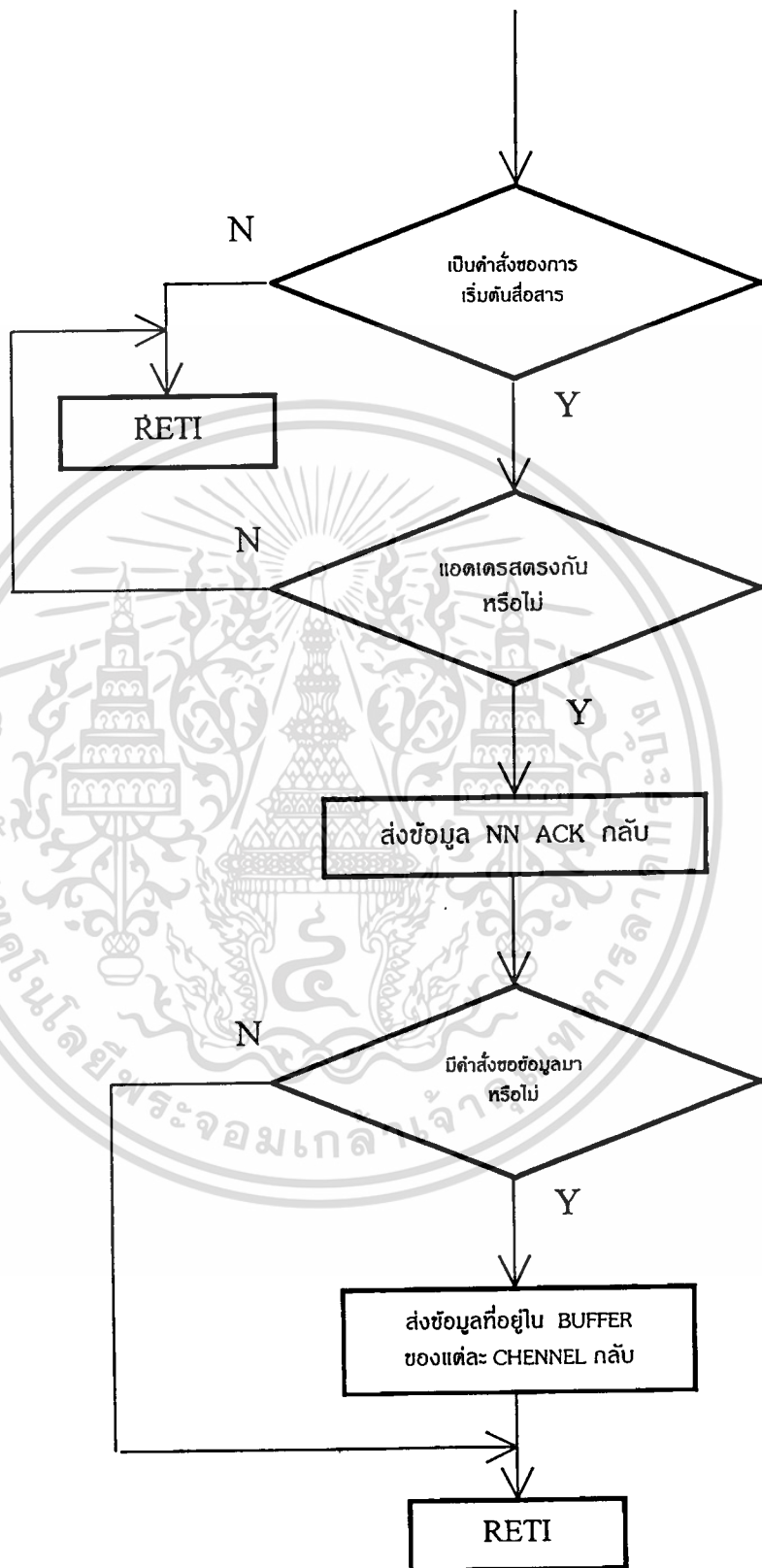
Watch dog ทำหน้าที่ป้องกันการดำเนินงานผิดพลาดของโมดูลย์ ซึ่งอาจทำงานหลุดจากอุปกรณ์ทำงานปกติ ให้กลับสู่การทำงานตามเดิม

Power Supply ทำหน้าที่จ่ายไฟ +5 โวลต์ เลี้ยงโมดูลย์



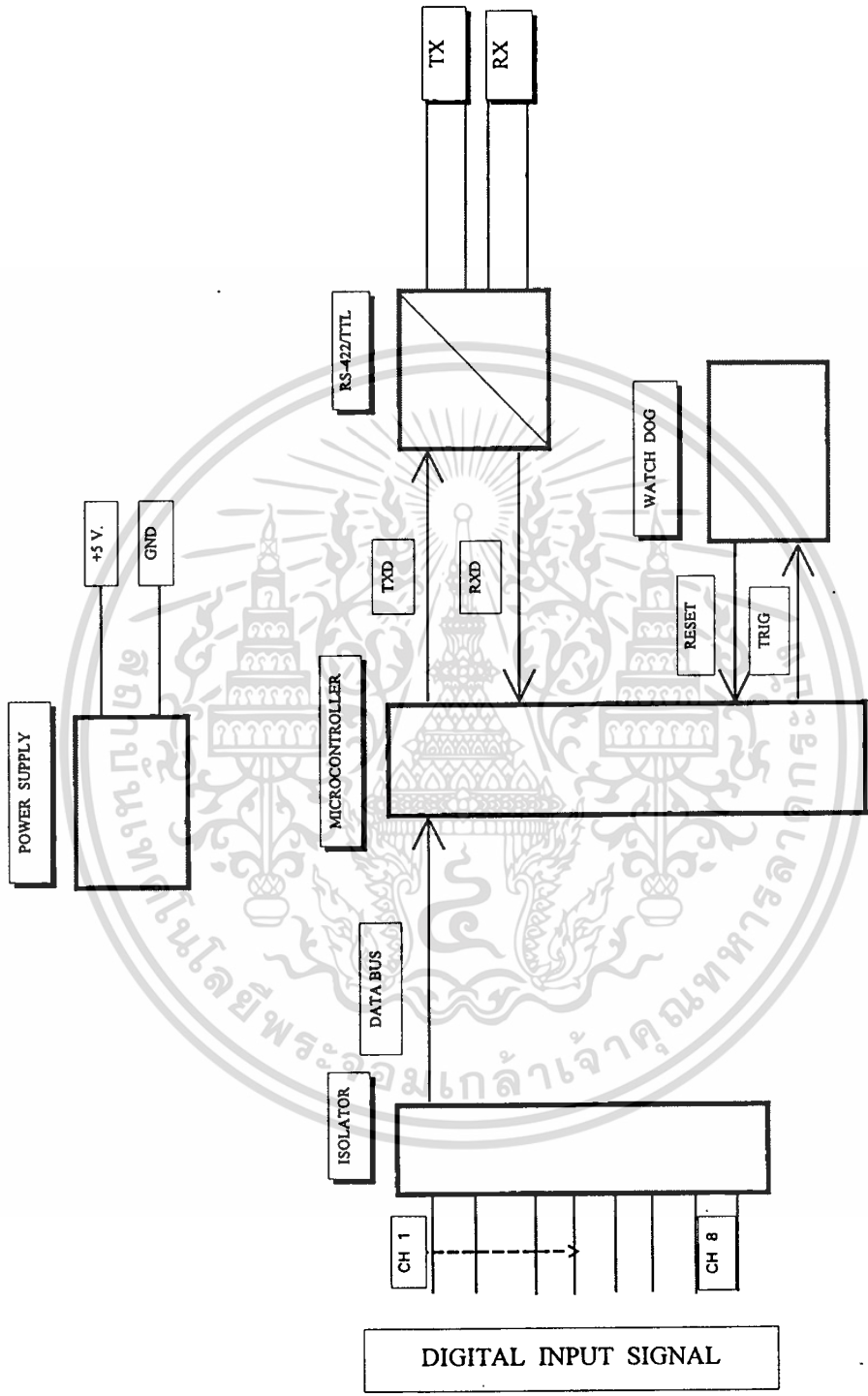
Flowchart ของ DI

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



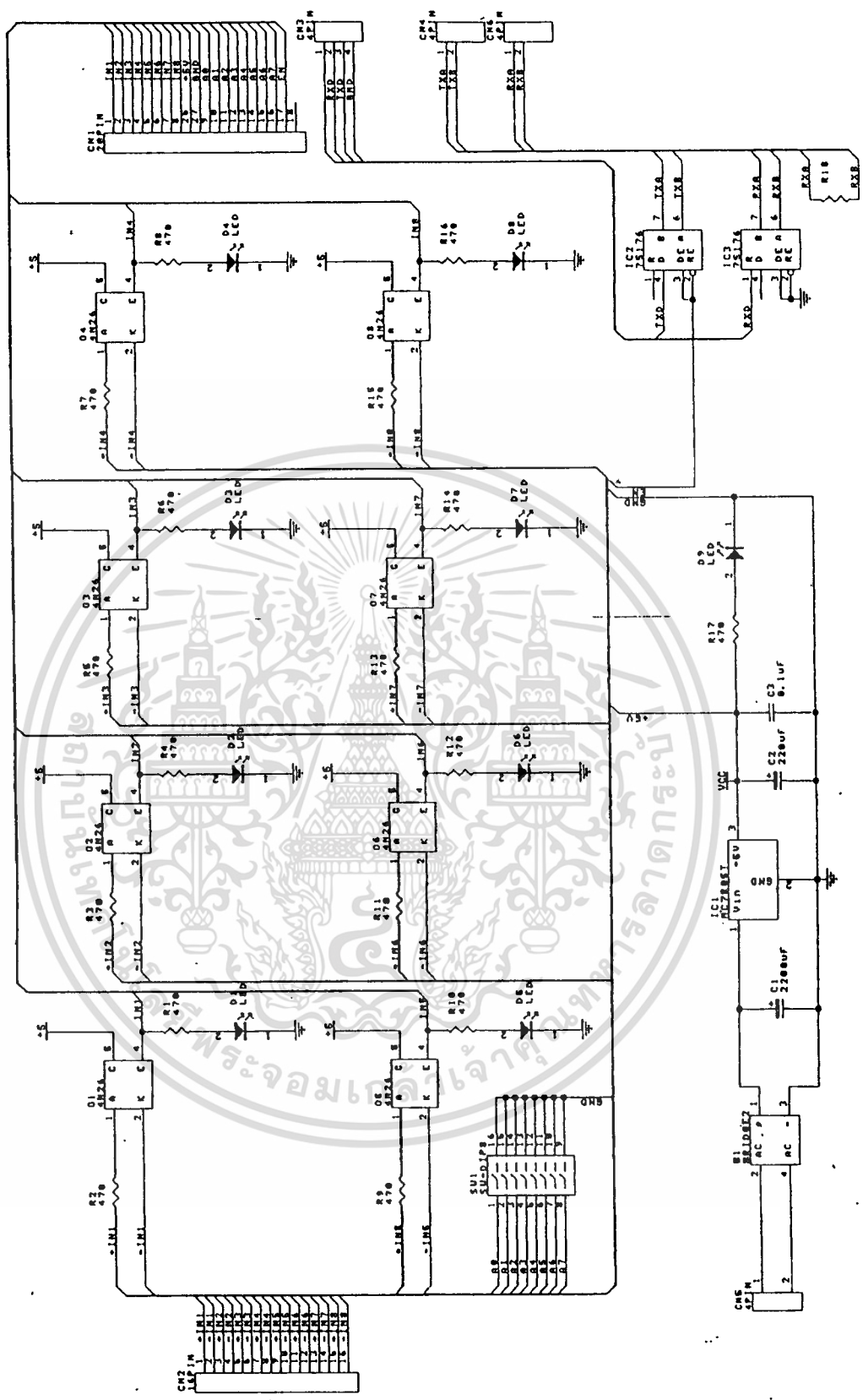
Interrupt Service Routine ของ DI

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



DIGITAL INPUT MODULE (DI)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



วงจรของ DIGITAL INPUT MODULE (DI)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้拿去ใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การทำงานของวงจร DI

สัญญาณอินพุตจากจุดวัด จะถูกป้อนเข้ามาทาง CN2 เพื่อป้อนเข้าสู่ไอซีเตอร์ O1 - O8 สัญญาณที่ออกมาจาก O1 - O8 จะมีระดับแรงดัน TTL เพื่อป้อนเข้าไปยังส่วนไมโครคอนโทรลเลอร์ ผ่านทาง CN1 LED D1 - D8 ทำหน้าที่เป็นตัวแสดงสภาวะของสัญญาณอินพุต

สัญญาณ TTL ถูกป้อนผ่าน CN1 ซึ่งเป็น Conector ที่อยู่บนบอร์ดพัฒนา CP-68 H C11 ของบริษัท อีทีที สัญญาณจะถูกส่งผ่าน IC เบอร์ 8255 ผ่านเข้าไปสู่อิซีคอนโทรลเลอร์เบอร์ 68HC11 เพื่อทำการประมวลผลสัญญาณ เตรียมส่งให้กับส่วนประมวลผลข้อมูลทางบัส RS-422

SW1 ทำหน้าที่เป็น สวิตช์ตั้งหมายเลข Address ของโมดูลย ว่ามีหมายเลขเท่าใด โดยมีค่าตั้งแต่ 0-31

IC2 และ IC3 ทำหน้าที่แปลงสัญญาณ TTL เป็น RS-422 และ RS-422 เป็น TTL ตามลำดับ เพื่อขยายขอบเขตความสามารถของการสื่อสารให้เป็นแบบ 'Multidrop' และเพิ่มระยะทางในการติดต่อสื่อสาร

สำหรับ Watch dog จะใช้ Watch dog ภายในตัวของ 68HC11 ซึ่งเป็นการสะดวกในการทำงานยิ่งขึ้น

4.2.4 การออกแบบ DIGITAL OUTPUT MODULE

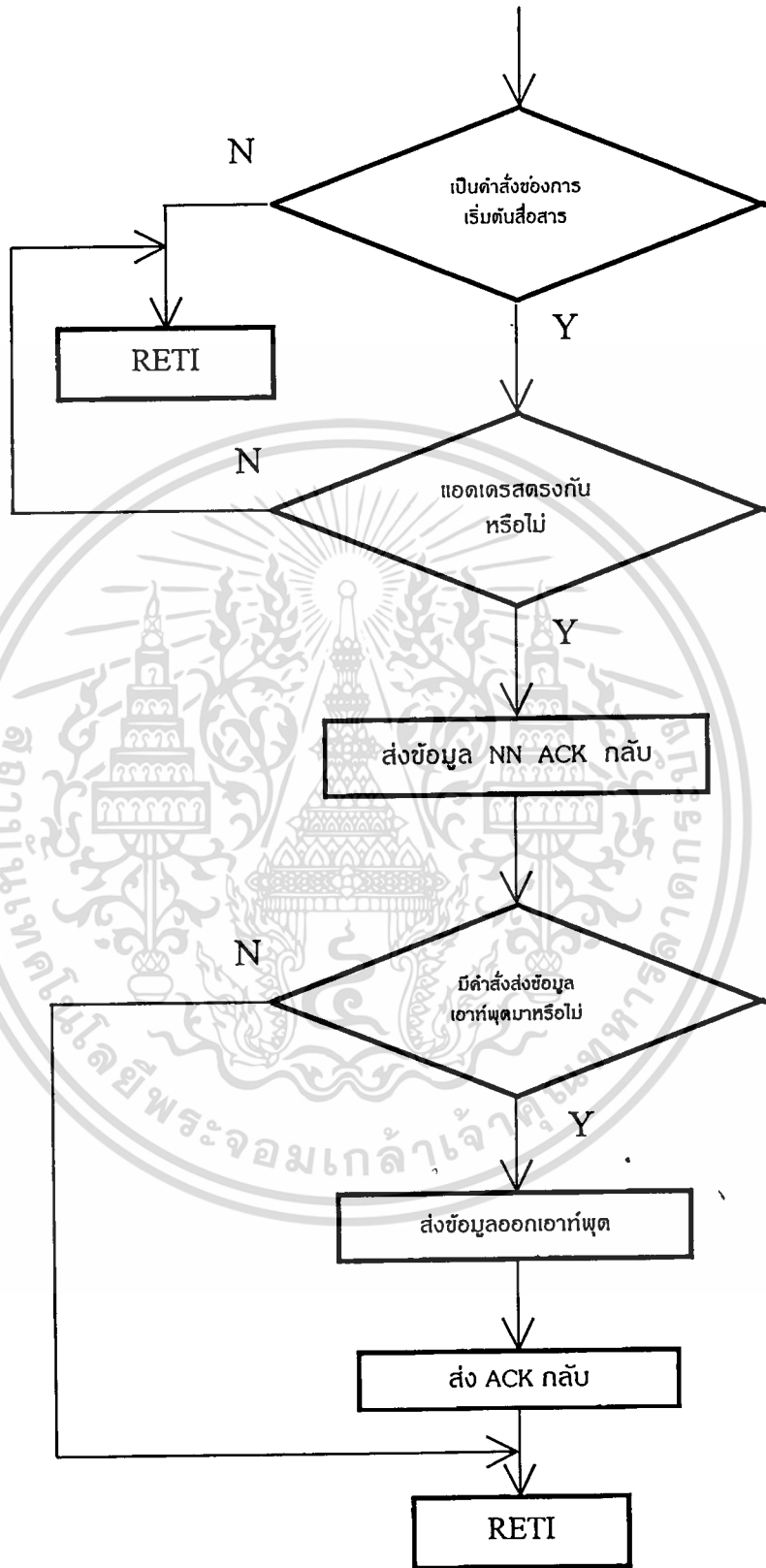
บล็อกไดอะแกรมของ Digital output module แสดงดังรูปด้านล่าง RS-422/TTL ทำหน้าที่เปลี่ยนสัญญาณ RS-422 เป็น TTL และเปลี่ยนสัญญาณ TTL เป็น RS-422 ไมโครคอนโทรลเลอร์ ทำหน้าที่ประมวลข้อมูลที่ส่งมาจาก ส่วนประมวลผลข้อมูลว่าจะทำการส่ง Output เป็นเท่าใด

Isolator ทำหน้าที่ แยกการเชื่อมโยงระหว่าง ตัวโมดูลกับจุดที่ต้องการควบคุม Watch dog ทำหน้าที่ป้องกันการ ทำงานของโมดูลผิดพลาดจากการหลุดรูป ซึ่ง Watch dog จะช่วยให้การทำงานกลับสู่รูปเดิม

Power Supply ทำหน้าที่จ่ายไฟ +5 โวลต์ เลี้ยงโมดูล

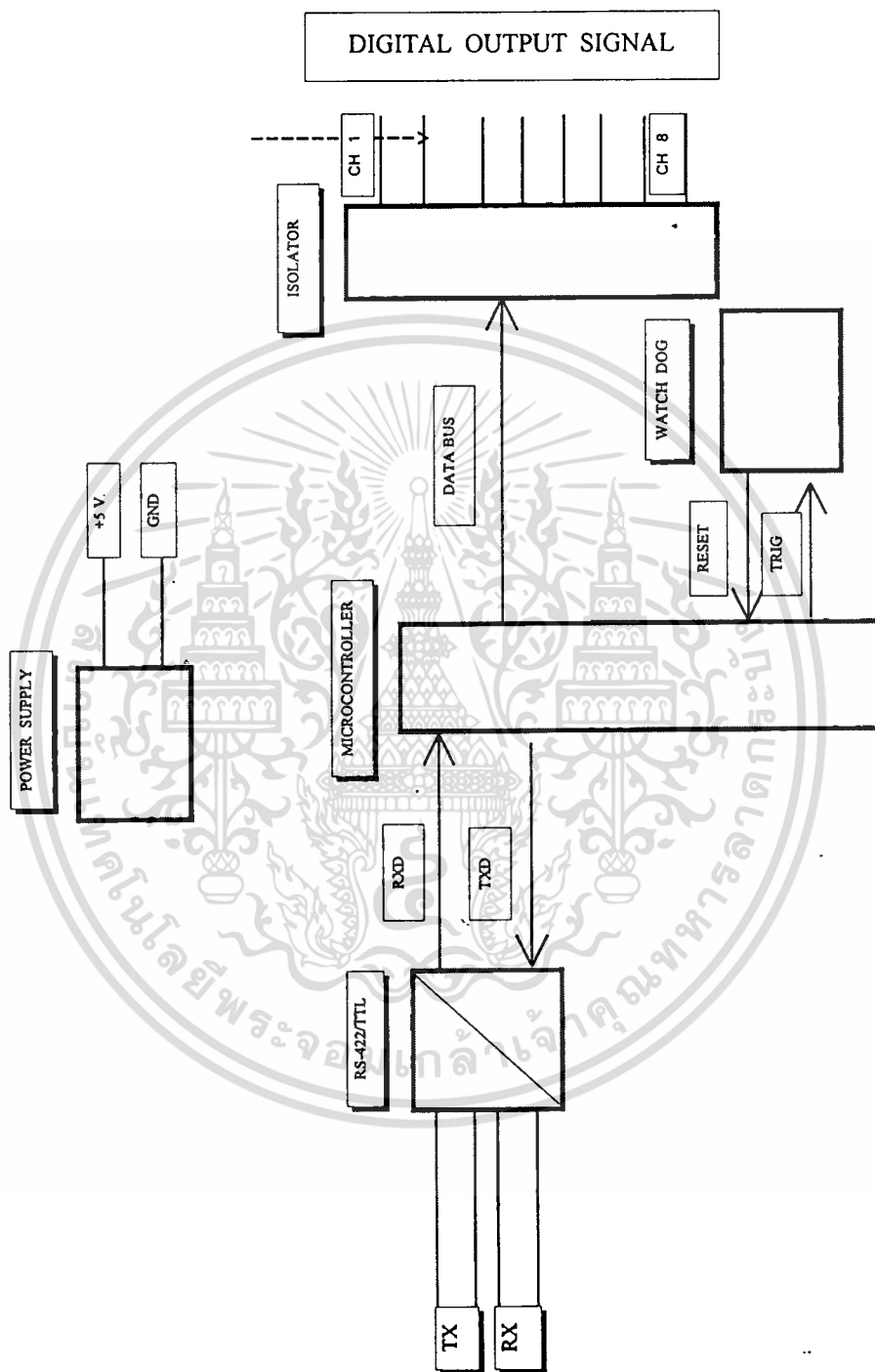


Flowchart ของ DO



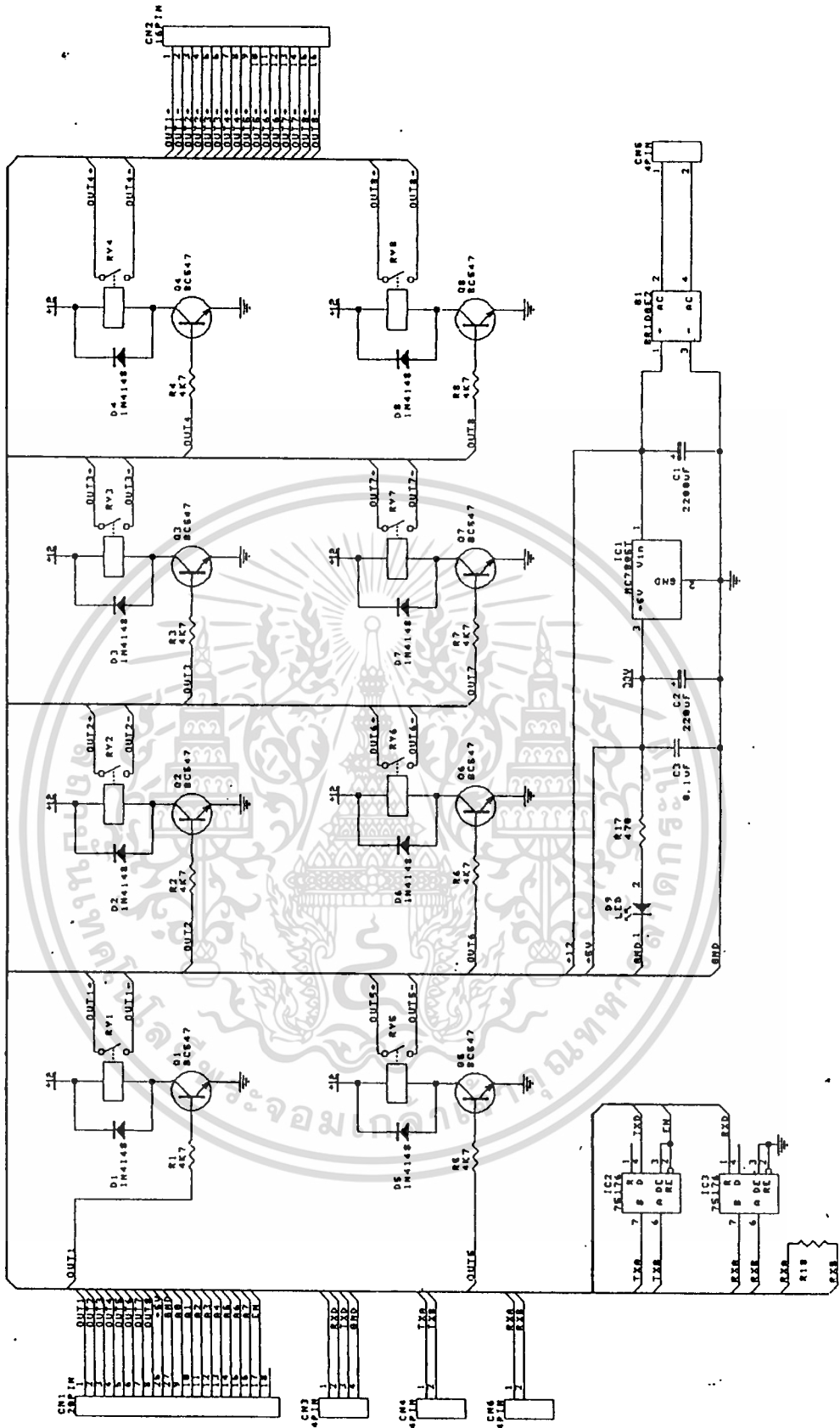
Flowchart Service Routine ของ DO

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



DIGITAL OUTPUT MODULE (DO)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



วงจรของ DIGITAL OUTPUT MODULE (DO)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การทำงานของวงจร DO

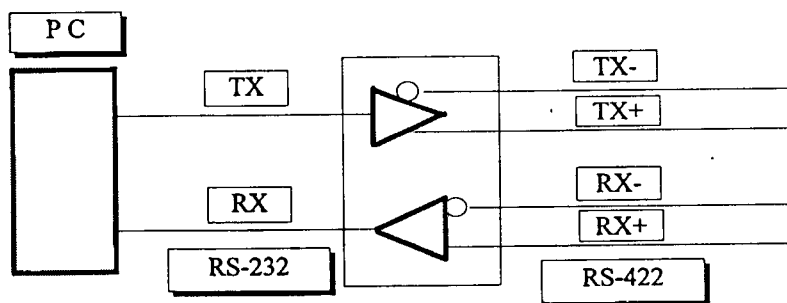
ไมโครจะติดต่อกับส่วนประมวลผลข้อมูล โดยใช้ IC2 และ IC3 เป็นตัวทำหน้าที่เปลี่ยนระดับแรงดัน TTL เป็น RS-422 และจาก RS-422 เป็น TTL มีไมโครคอนโทรลเลอร์ 68#C11 ทำการวิเคราะห์ข้อมูลที่ถูส่งมาจากส่วนประมวลผลข้อมูลแล้ว ไมโครคอนโทรลเลอร์ ก็จะทำการส่งสัญญาณ Output ออกมาที่ IC 8255

สัญญาณจาก IC 8255 จะถูกส่งออกมาที่ Q1 - Q8 ซึ่งทำหน้าที่ขับ Relay สัญญาณ Output จะใช้หน้าสัมผัสของ Relay ในการติดต่อโหลด

4.3 ส่วนของการเปลี่ยนระดับสัญญาณ RS-232 เป็น RS-422

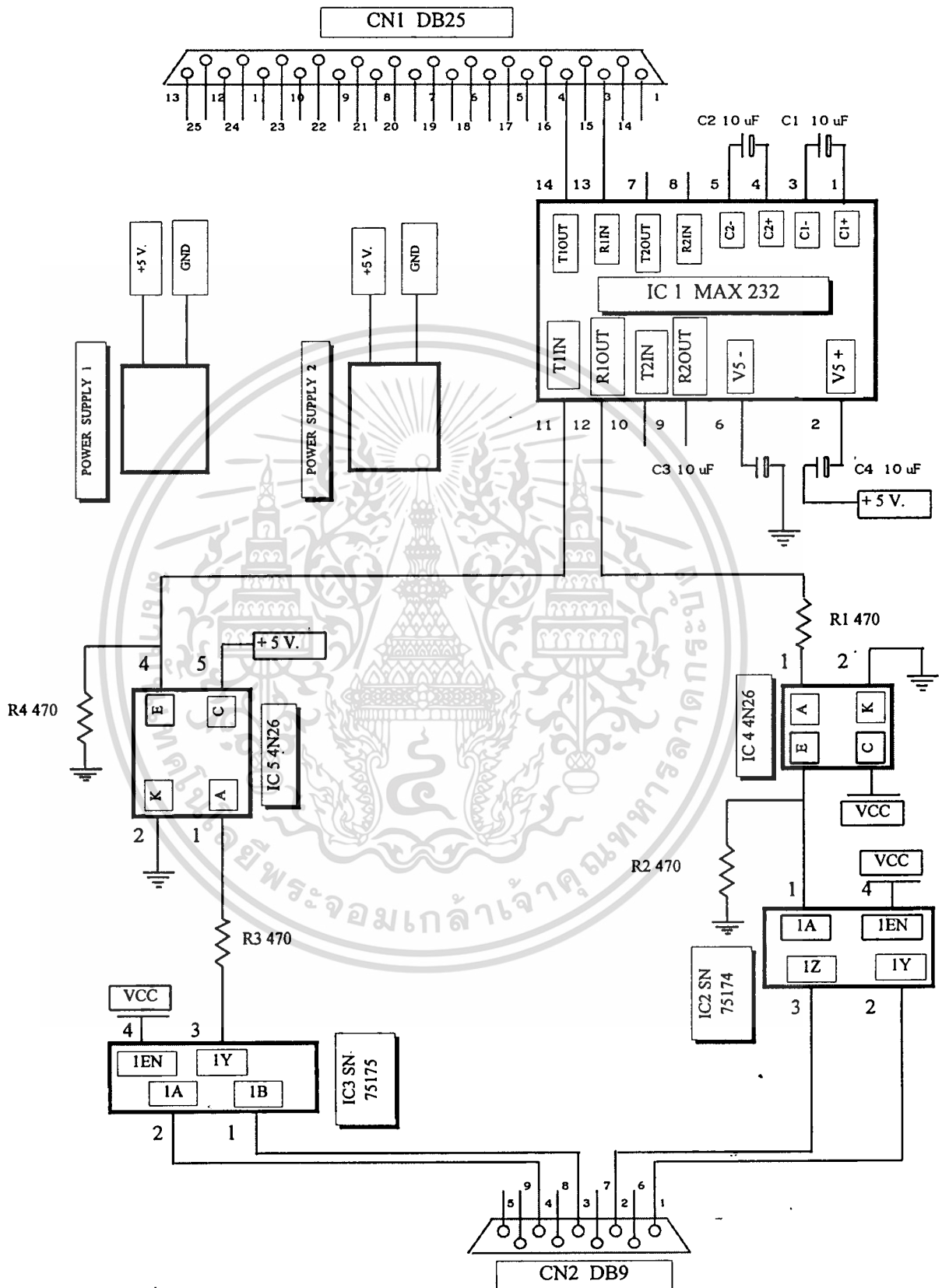
ภาคการทำงานนี้ทำหน้าที่เปลี่ยนระดับแรงดันสัญญาณจากมาตรฐาน RS-232 เป็น มาตรฐาน RS-422 ซึ่งการออกแบบในส่วนนี้ได้ใช้ไอซีเปลี่ยนสัญญาณ ที่มีขายในตลาดเมืองไทย มีดังนี้

- ไอซีเบอร์ MAX 232 ซึ่งทำหน้าที่เป็นตัวเปลี่ยนสัญญาณจากแรงดัน TTL เป็นแรงดันมาตรฐาน RS-232 และเปลี่ยนจากแรงดันมาตรฐาน RS-232 เป็นแรงดัน TTL
- ไอซีเบอร์ SN75174 ทำหน้าที่เปลี่ยนระดับแรงดันจาก TTL เป็นสัญญาณมาตรฐาน RS-422
- ไอซีเบอร์ SN75175 ทำหน้าที่เปลี่ยนระดับแรงดันมาตรฐาน RS-422 เป็น TTL



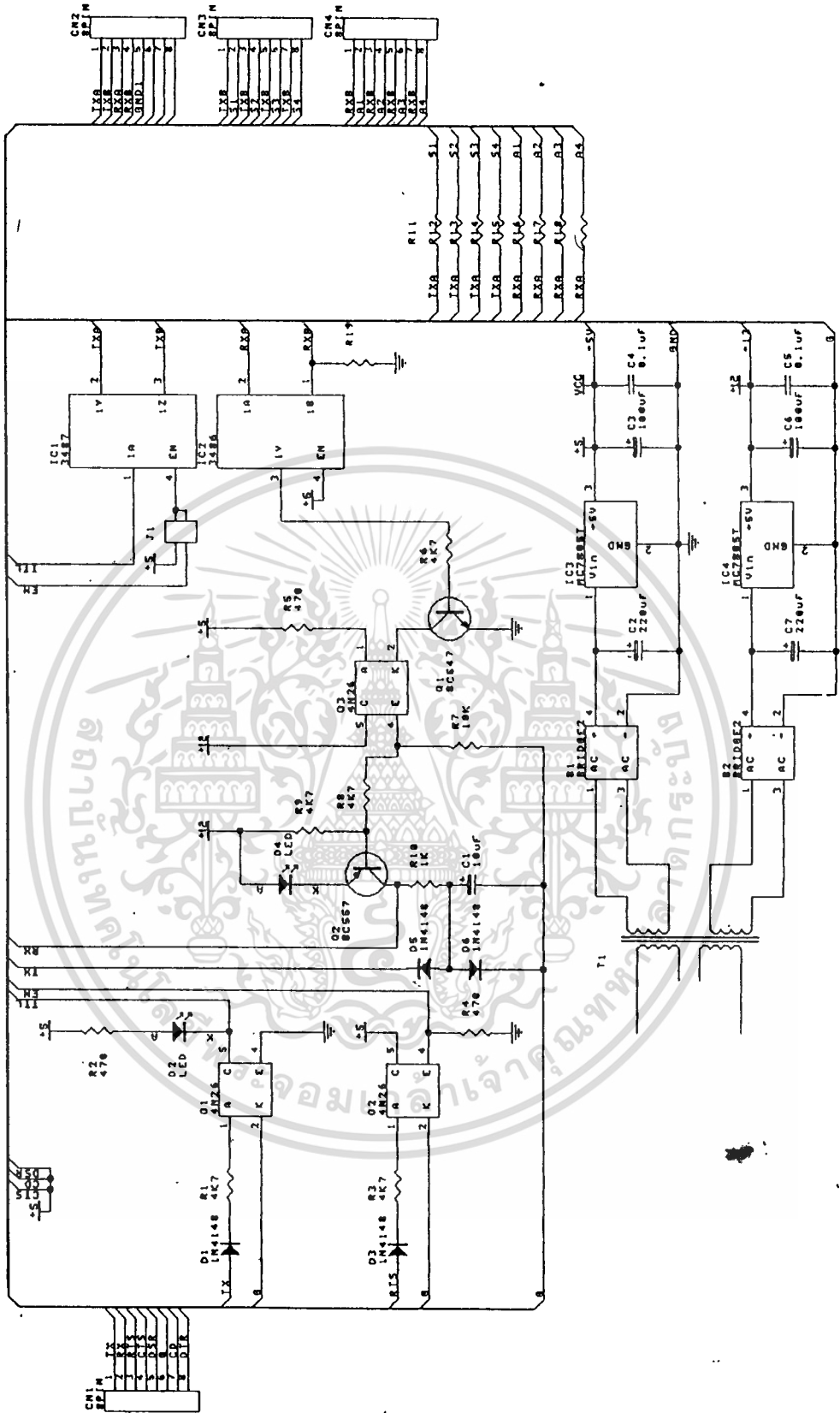
รูปแสดงแนวความคิดในการเปลี่ยนระดับสัญญาณระหว่าง RS-232 และ RS-422

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



บล็อกไดอะแกรมของ CONVERSE 232/422

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



วงจรของ CONVERSE RS-232 / RS-422

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้拿去ใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การทำงานของวงจร CONVERSE ระหว่าง RS-232 กับ RS-422

IC1 ทำหน้าที่เปลี่ยนระดับแรงดัน RS-232 ที่ถูกส่งมาจากคอมพิวเตอร์ ให้เป็นระดับแรงดัน TTL โดยรับสัญญาณจากขา 2 ของ DB25 ซึ่งเป็นขา TX ของพอร์ตอนุกรม สัญญาณ RS-232 เมื่อถูกเปลี่ยนเป็นสัญญาณ TTL แล้ว จะถูกส่งออกที่ขา 12 ของ IC1 ป้อนเข้าสู่ IC4 ซึ่งเป็นตัวเชื่อมโยงทางแสง สาเหตุที่ต้องใช้ เพื่อป้องกันสัญญาณรบกวนจากข้างนอกเข้ามารบกวนคอมพิวเตอร์

IC2 ทำหน้าที่เปลี่ยนระดับสัญญาณจาก TTL ไปเป็นสัญญาณมาตรฐาน RS-422 เพื่อส่งออกสู่ RS-422 Bus ต่อไป

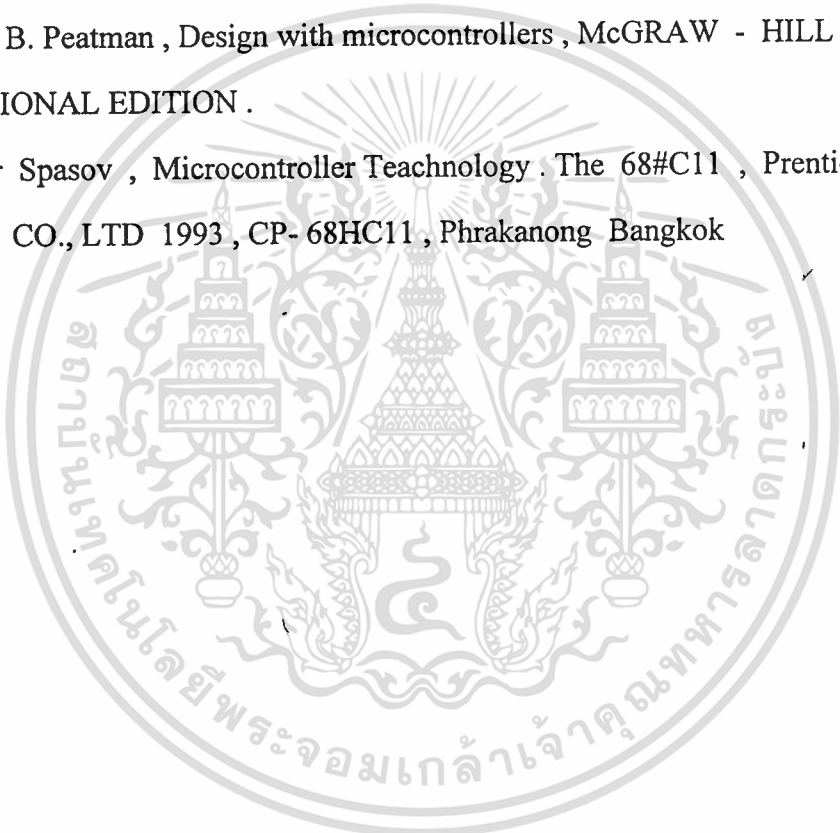
พิจารณาทางด้านรับสัญญาณ สัญญาณที่ส่งมาเพื่อจะเข้าสู่คอมพิวเตอร์ที่อยู่บน RS-422 Bus จะต่อเข้ากับ IC3 ซึ่งทำหน้าที่เปลี่ยนระดับแรงดันจาก RS-422 เป็นระดับแรงดัน TTL ออกที่ขา 3 ของ IC3 สัญญาณนี้จะถูกส่งเข้าที่ IC5 ซึ่งเป็นตัวเชื่อมโยงทางแสง

สัญญาณ TTL ที่ผ่านมาจากตัวเชื่อมโยงทางแสงจะถูกส่งเข้าที่ขา 11 ของ IC1 เพื่อทำการเปลี่ยนสัญญาณจาก TTL เป็น RS-232 ส่งให้กับคอมพิวเตอร์ต่อไป

Power Supply ที่ใช้ในวงจรนี้แบ่งเป็น 2 ชุด เพื่อจ่ายให้กับวงจร 2 ส่วน ซึ่งแต่ละส่วนวงจรจะถูกแยกออกโดย IC4 และ IC5

เอกสารอ้างอิง

- Harold S. Stone , Micro computer Interfacing , University of Massachnsetts , Amherst
- Motorola Inc ., 1991 , CMOS Logit Data
- Motorola Inc., 1993 , Linear/Interface IIs Vol. II
- John B. Peatman , Design with microcontrollers , McGRAW - HILL INTERNATIONAL EDITION .
- Peter Spasov , Microcontroller Teachnology . The 68#C11 , Prentice hall
- ETT CO., LTD 1993 , CP- 68HC11 , Phrakanong Bangkok

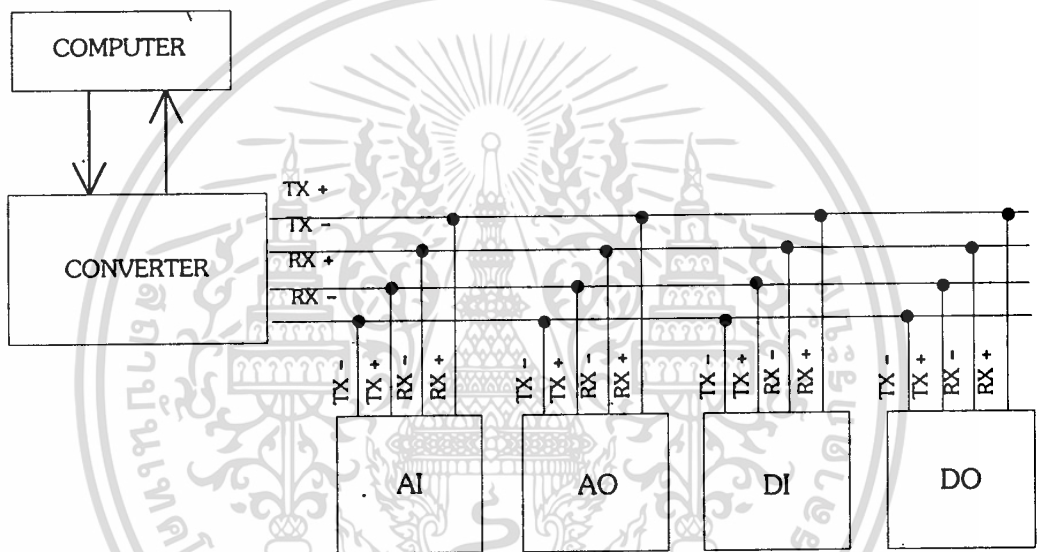




เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ก

การต่อทั้งระบบของ SYSTEM MONITORING



รูปแสดงการต่อทั้งระบบของ SYSTEM MONITORING

การกำหนดขาต่างๆ ของ RS-232 / RS-422 CONVERTER

DB9 (RS-422)

- 1 = TX +
- 2 = TX -
- 3 = RX +
- 4 = RX -
- 5 = GND

DB25 (RS-232)

- 2 = TXD
- 3 = RXD
- 7 = GND

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ข

การใช้งาน SOFTWARE

ซอฟต์แวร์ชุดนี้รับภายใต้วินโดวส์ 3.1 ผู้ใช้สามารถติดตั้งโปรแกรมได้ง่าย โดยใช้แผ่น Setup ทำตามขั้นตอนดังต่อไปนี้

- เริ่มเข้าสู่ Microsoft Windows
- เลือก Main Icon
- เลือก Drive ที่แผ่น Setup อยู่
- เลือกไฟล์ Setup.EXE มารัน
- จากนั้นการติดตั้งโปรแกรมก็จะไปอย่างอัตโนมัติ

System Monitoring

V 2.5

Telecommunication Technology
Faculty Of Engineering

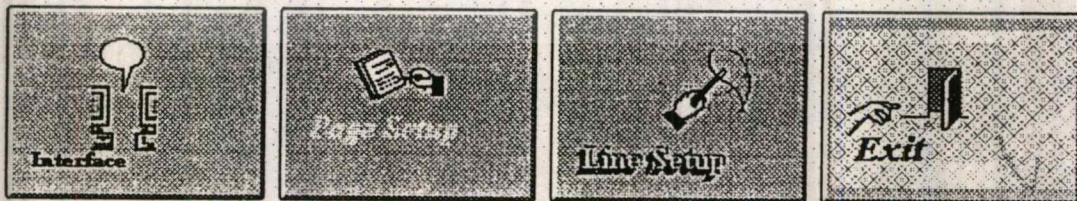
เมื่อเข้าสู่ System Monitoring ก็จะมี Logo ของโปรแกรมปรากฏประมาณ 8

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



เมนูโปรแกรม

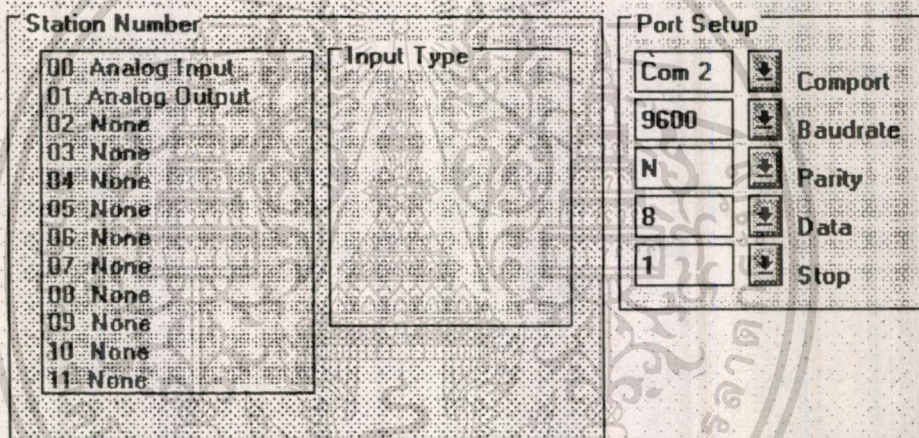
- Setup Manager เลือกเพื่อการ Set พอร์ตอนุกรม และการเซตคุณสมบัติของแต่ละโมดูลย์ว่าจะให้ทำหน้าที่อะไร
- Graph History เป็นเมนูเพื่อใช้เลือกการแสดงผลค่าที่ได้จัดเก็บไว้ เป็นกราฟ ซึ่งผู้ใช้นำข้อมูลที่ถูกจัดเก็บจากขบวนการผลิตมาวิเคราะห์ได้
- Data History เมนูเลือกการแสดงผลค่าที่จัดเก็บจากขบวนการผลิตให้อยู่ในรูปของตัวเลข
- Exit ออกจากโปรแกรม System monitoring
- Go เริ่มทำการแสดงผล และจัดเก็บข้อมูล



เมื่อกดเลือกเมนูที่ Setup Manager ก็จะปรากฏเมนูขึ้นมาใหม่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่นอนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- Interface เมื่อนี้ใช้เลือกคุณสมบัติของ โมดูลต่าง ๆ และพอร์ตอนุกรม
- Page Setup เมื่อนี้ใช้เลือกแต่ละหน้าของการจัดเก็บข้อมูล เช่น ชื่อ และเวลาในการจัดเก็บ
- Line Setup เมื่อนี้ใช้เลือก เพื่อเซตคุณสมบัติของแต่ละเส้นกราฟ และเซตการแสดงผลค่าแรงเดือน เมื่อเกิดความผิดพลาดในระบบ
- Quit ออกจาก Setup Manager



INTERFACE SETUP

เมื่อกดเมนู Interface จะปรากฏเมนูใหม่ดังแสดงในรูปด้านบน

- Station Number แสดงว่าแต่ละโมดูลจะทำหน้าที่เป็นอะไร เช่น Analog input Analog output , Digital Input , Digital Output
- Port Setup เลือกการทำงานของพอร์ตอนุกรม และคุณสมบัติต่าง ๆ ของพอร์ตอนุกรม

Page

Enable

Somchai Name

3 Scan Rate (Second)

PAGE SETUP

- Page ใช้เลือกที่จะเซตที่หน้าไหน มีตั้งแต่ 1 - 10
- Enable เลือกที่จะให้หน้านั้นๆ แสดงรูปภาพหรือไม่
- Name ชื่อของแต่ละหน้า
- Scan Rate ช่วงเวลาในการจัดเก็บแต่ละครั้ง

Page 01 Line 01dfd

dfd Name

Active

Alarm Setup

Alarm Enable

250.0000 High Alarm

-250.0000 Low Alarm

1.000000 Differential Alarm

Address

0 Station NO.

0 Slot

0 Channel

0 Bit

Alarm Output

01 Station No

01 Channel

5.00 Value

Volts Units

-500.000000 Zero

500.000000 Span

Accept

Exit

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับใช้ภายในระบบเท่านั้น ระเบียบด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

LINE SETUP

- Line Select เลือกว่าจะทำการเซตที่ Line ไหน มีตั้งแต่ Line 1 Page 1 ถึง

Line 10 Page 10

- Active เลือกว่าจะให้ Line นั้น ๆ แสดงหรือไม่
- Address เลือกว่าจะให้ Line นั้น ๆ นำค่าที่จะแสดงมาจากไหน
- Unit ไล่หน่วยการวัด
- Zero ค่าที่แสดงเมื่อ Input = 0 %
- Span ค่าที่แสดงเมื่อ Input = 100 %
- Alarm Setup ใช้เลือกค่าตัวแปรต่าง ๆ ที่จะใช้ในการแจ้งเตือน
- High Alarm เลือกค่าที่จะแจ้งเตือน เมื่อ Input สูงเกินกว่าค่าที่กำหนด
- Low Alarm เลือกค่าที่จะแจ้งเตือนเมื่อ Input ต่ำกว่าค่าที่กำหนด
- Differential Alarm เลือกค่าฮิสเตอร์รีซิสของการแจ้งเตือน
- Alarm Output เลือกจุดที่จะส่งค่าออกไปควบคุมเมื่อมีการแจ้งเตือน

Station NO.	Type	Data
00	Analog Input	00
01	Analog Output	
02	Digital Input	00
03	Digital Output	00
04	None	00
05		00
06		00
		00
		00
		00
		00

SIMULATOR

- ใช้เพื่อการจำลองข้อมูลที่ส่งมาจากโมดูลต่าง ๆ กรณีไม่ใช่โมดูลข้างนอก

	Slot 0	Slot 1	Slot 2
Station (00)			
Channel 0	00	00	00
Channel 1	00	00	00
Channel 2	00	00	
Channel 3	00	00	
Channel 4	00	00	
Channel 5	00	00	
Channel 6	00	00	
Channel 7	00	00	

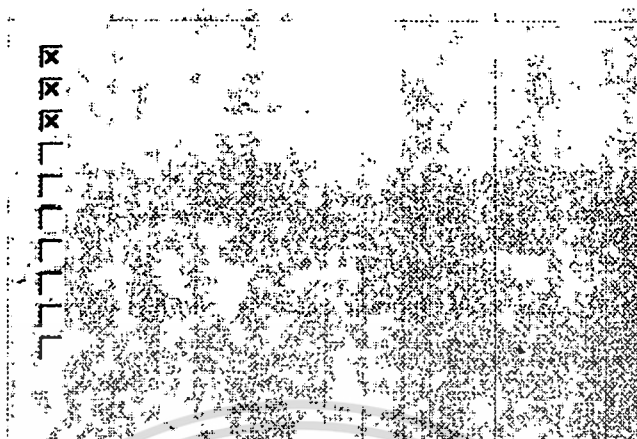
TX Status

RX Status



VIEW

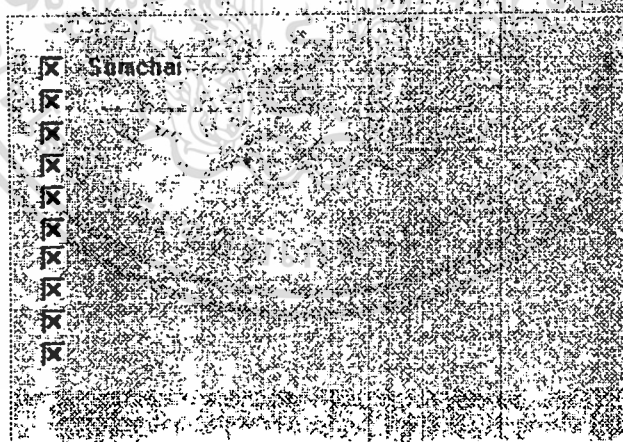
- ใช้ดูการเปลี่ยนแปลงของข้อมูลในแต่ละ Station Number รวมทั้งดูผลของการเชื่อมต่อข้อมูล กับโมดูลต่างๆ ด้วย ว่ามีการผิดพลาดหรือไม่



LINE DETAIL

- ใช้ดูรายละเอียดของแต่ละ Page ว่า เส้นกราฟแต่ละเส้นชื่ออะไร และ Active

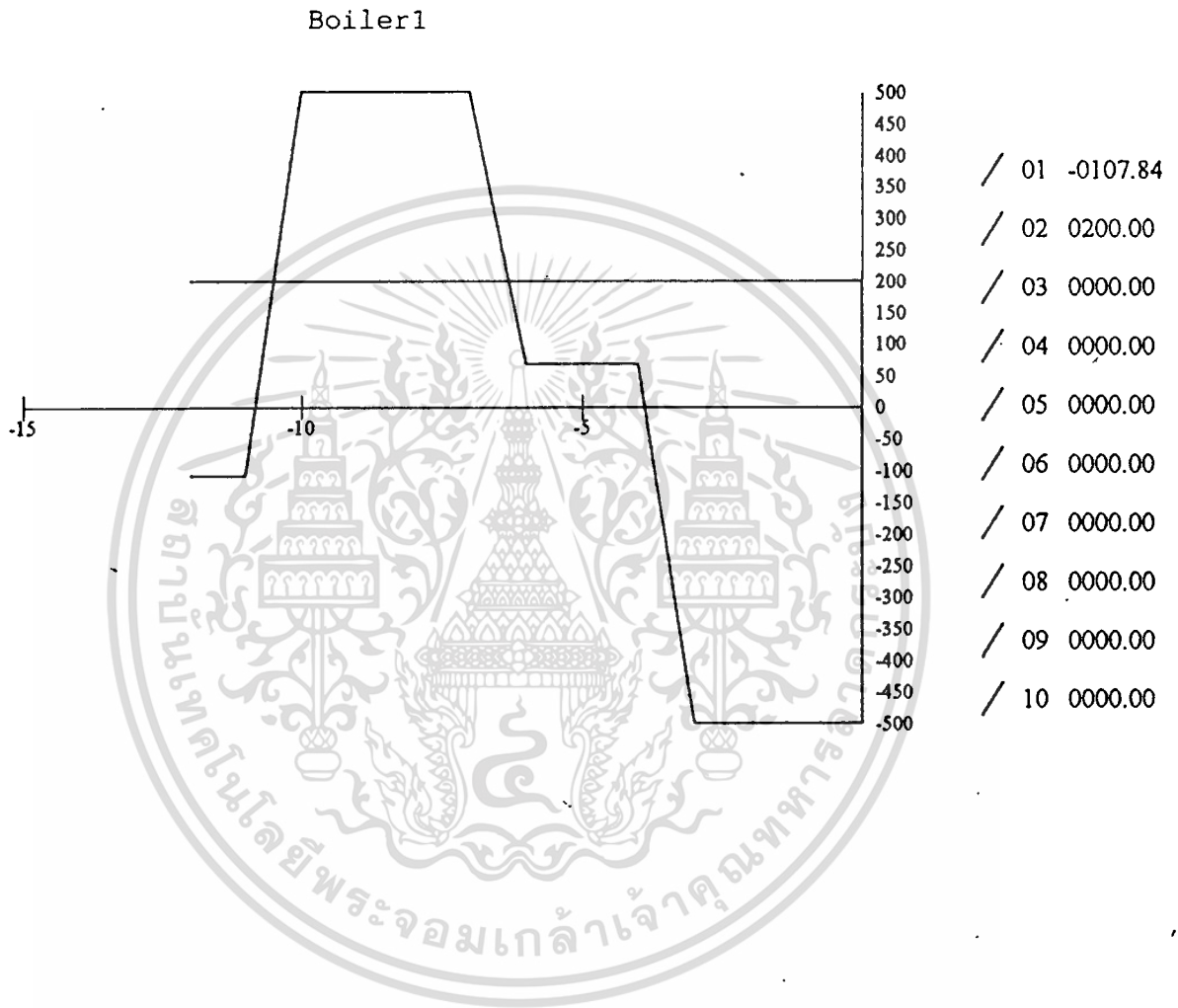
หรือไม่



PAGE DETAIL

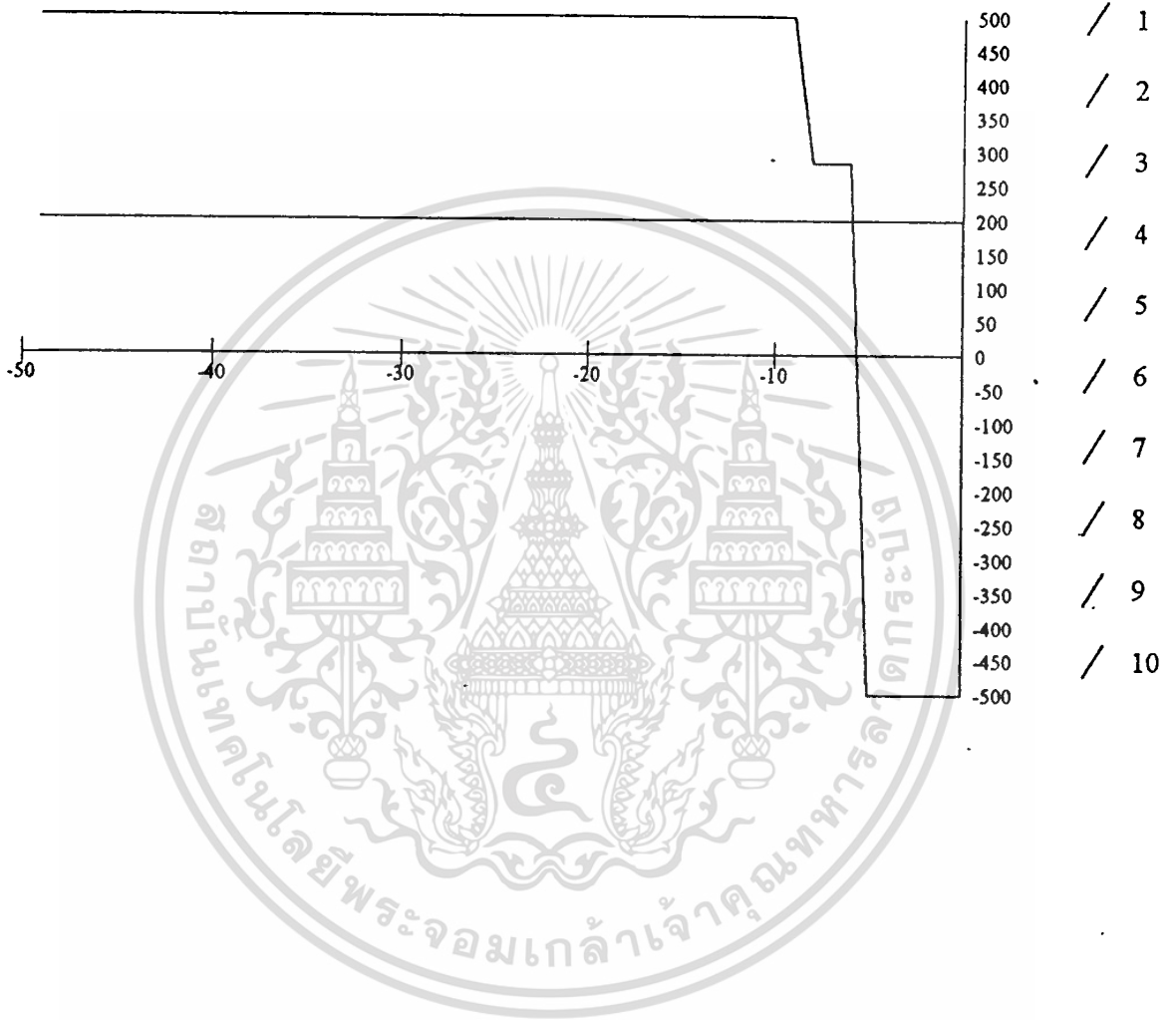
- ใช้ดูรายละเอียดเกี่ยวกับ แต่ละหน้าของการแสดงผล ว่าชื่ออะไร และ Active

หรือไม่



ผลจากการพิมพ์ข้อมูลขณะที่มีการรันปกติ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



ผลจากการพิมพ์ข้อมูลจาก เมนู Graph History

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Data History Report

Print Mode :Alarm Only

Page: 1 Line: 1

Date	Time	Value	Status
10-27-1994	10:33:05	-500.0000	Low Alarm
10-27-1994	10:33:11	-500.0000	Low Alarm
10-27-1994	10:33:16	-500.0000	Low Alarm
10-27-1994	10:33:20	-500.0000	Low Alarm
10-27-1994	10:33:47	-500.0000	Low Alarm
10-27-1994	10:33:52	-500.0000	Low Alarm
10-27-1994	10:33:57	284.3137	High Alarm
10-27-1994	10:34:02	284.3137	High Alarm
10-27-1994	10:34:07	284.3137	High Alarm
10-27-1994	10:34:12	500.0000	High Alarm
10-27-1994	10:34:18	500.0000	High Alarm
10-27-1994	10:34:23	500.0000	High Alarm
10-27-1994	10:34:28	500.0000	High Alarm
10-27-1994	10:34:33	500.0000	High Alarm
10-27-1994	10:34:38	500.0000	High Alarm
10-27-1994	10:34:43	500.0000	High Alarm
10-27-1994	10:34:48	500.0000	High Alarm
10-27-1994	10:34:53	500.0000	High Alarm
10-27-1994	10:34:59	500.0000	High Alarm
10-27-1994	10:35:04	500.0000	High Alarm
10-27-1994	10:35:09	500.0000	High Alarm
10-27-1994	10:35:14	500.0000	High Alarm
10-27-1994	10:35:19	500.0000	High Alarm
10-27-1994	10:35:25	500.0000	High Alarm
10-27-1994	10:35:30	500.0000	High Alarm
10-27-1994	10:35:35	500.0000	High Alarm
10-27-1994	10:35:40	500.0000	High Alarm
10-27-1994	10:35:46	500.0000	High Alarm
10-27-1994	10:35:51	500.0000	High Alarm
10-27-1994	10:35:56	500.0000	High Alarm
10-27-1994	10:36:01	500.0000	High Alarm
10-27-1994	10:36:07	500.0000	High Alarm
10-27-1994	10:36:12	500.0000	High Alarm
10-27-1994	10:36:17	500.0000	High Alarm
10-27-1994	10:36:22	500.0000	High Alarm
10-27-1994	10:36:28	500.0000	High Alarm
10-27-1994	10:36:33	500.0000	High Alarm
10-27-1994	10:36:38	500.0000	High Alarm
10-27-1994	10:36:44	500.0000	High Alarm
10-27-1994	10:36:49	500.0000	High Alarm
10-27-1994	10:36:54	500.0000	High Alarm
10-27-1994	10:37:00	500.0000	High Alarm
10-27-1994	10:37:05	500.0000	High Alarm
10-27-1994	10:37:10	500.0000	High Alarm
10-27-1994	10:37:16	500.0000	High Alarm

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Data History Report

Print Mode :All Data
Page: 1 Line: 1

Date	Time	Value	Status
10-27-1994	10:33:05	-500.0000	Low Alarm
10-27-1994	10:33:11	-500.0000	Low Alarm
10-27-1994	10:33:16	-500.0000	Low Alarm
10-27-1994	10:33:20	-500.0000	Low Alarm
10-27-1994	10:33:47	-500.0000	Low Alarm
10-27-1994	10:33:52	-500.0000	Low Alarm
10-27-1994	10:33:57	284.3137	High Alarm
10-27-1994	10:34:02	284.3137	High Alarm
10-27-1994	10:34:07	284.3137	High Alarm
10-27-1994	10:34:12	500.0000	High Alarm
10-27-1994	10:34:18	500.0000	High Alarm
10-27-1994	10:34:23	500.0000	High Alarm
10-27-1994	10:34:28	500.0000	High Alarm
10-27-1994	10:34:33	500.0000	High Alarm
10-27-1994	10:34:38	500.0000	High Alarm
10-27-1994	10:34:43	500.0000	High Alarm
10-27-1994	10:34:48	500.0000	High Alarm
10-27-1994	10:34:53	500.0000	High Alarm
10-27-1994	10:34:59	500.0000	High Alarm
10-27-1994	10:35:04	500.0000	High Alarm
10-27-1994	10:35:09	500.0000	High Alarm
10-27-1994	10:35:14	500.0000	High Alarm
10-27-1994	10:35:19	500.0000	High Alarm
10-27-1994	10:35:25	500.0000	High Alarm
10-27-1994	10:35:30	500.0000	High Alarm
10-27-1994	10:35:35	500.0000	High Alarm
10-27-1994	10:35:40	500.0000	High Alarm
10-27-1994	10:35:46	500.0000	High Alarm
10-27-1994	10:35:51	500.0000	High Alarm
10-27-1994	10:35:56	500.0000	High Alarm
10-27-1994	10:36:01	500.0000	High Alarm
10-27-1994	10:36:07	500.0000	High Alarm
10-27-1994	10:36:12	500.0000	High Alarm
10-27-1994	10:36:17	500.0000	High Alarm
10-27-1994	10:36:22	500.0000	High Alarm
10-27-1994	10:36:28	500.0000	High Alarm
10-27-1994	10:36:33	500.0000	High Alarm
10-27-1994	10:36:38	500.0000	High Alarm
10-27-1994	10:36:44	500.0000	High Alarm
10-27-1994	10:36:49	500.0000	High Alarm
10-27-1994	10:36:54	500.0000	High Alarm
10-27-1994	10:37:00	500.0000	High Alarm
10-27-1994	10:37:05	500.0000	High Alarm
10-27-1994	10:37:10	500.0000	High Alarm
10-27-1994	10:37:16	500.0000	High Alarm

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ค

โปรแกรม

```
#include <68hc11.h>
#include <stdio.h>
#include <stdlib.h>
#include <intrpt.h>
#include <reg11.h>

#define A_8255 (*(unsigned char*)0x1200)
#define B_8255 (*(unsigned char*)0x1201)
#define C_8255 (*(unsigned char*)0x1202)
#define CNT_8255 (*(unsigned char*)0x1203)
#define SOH 0x01
#define STX 0x02
#define ETX 0x03
#define EOT 0x04
#define ENQ 0x05
#define ACK 0x06
#define NAK 0x15
#define sp 0x20

/*----- Prototype function ----- */

set_sci(void);
send_data(unsigned char);
read_data();
void interrupt sci_int();

/*-----

unsigned char addr[2];
unsigned char input[20];
unsigned char dataout[2];
char ac;
main()
{
    unsigned char data;
    ROM_VECTOR(SCI_VEC,sci_int);
    CONFIG=0x0d;
    OPTION=0x93;
    CNT_8255=0x92;/* A=in B=in C=out */
    set_sci();
    ei();
    do{
        data=B_8255;
        addr[0]=(data/10)+0x30;
        addr[1]=(data%10)+0x30;
    }
    /*----- */
}
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่นิยมนำไปใช้ประโยชน์ด้านการค้า

ไปว่ากรณีใดบ้าง สิ่งนี้หมายถึงให้คัดลอกไปเพื่อ และต่อว่าจริงสิ่งนี้ของเอกสารที่การนำไปใช้

```

data=A_8255&0xf0;
data=data>>4;
if(data<=9)
{
    dataout[0]=data+0x30;
}
else if(data>9)
{
    dataout[0]=(data-10)+0x41;
}
}
-----
data=A_8255&0x0f;
if(data<=9)
{
    dataout[1]=data+0x30;
}
else if(data>9)
{
    dataout[1]=(data-10)+0x41;
}
}while(1);
}
set_sci()
{
    SCCR1=0x00;
    SCCR2=0x2c;
    BAUD =0x30;
}
send_data(data)
unsigned char data;
{
    C_8255=0x01;
    SCDR=data;
    do{
        |while(!((SCSR&0x80)));
        C_8255=0;
    }
}
read_data()
{
    unsigned char status;
    do{
        |while(!((SCSR&0x20)));
        status=SCDR;
        return status;
    }
}
interrupt sci_int()
{
    char a;
    char *b=NULL;
    di();
    input[0]=read_data();
    switch(input[0]){
        case EOT:for(a=1;a<=3;a++){
            {
                input[a]=read_data();
            }
            if((input[1]==addr[0])&&(input[2]==addr[1]))
            {
                send_data(input[1]);
                send_data(input[2]);
                send_data(ACK);
                ac=1;
            }
        }
        break;
    }
}

```

```

case STX:for(a=1;a<=8;a++)
    {
        input[a]=read_data();
    }
    if(ac!=1)
    {
        break;
    }
    else if((input[4]=='D')&&(input[5]=='I'))
    {
        sprintf(b,"%cEX DI %c%c %c",STX,dataout[0],dataout[1],ETX);
        for(a=0;*(b+a)!='\0';a++)
        {
            send_data(*(b+a));
        }
        ac=0;
    }
    break;
default :break;
}
ei();
}

```



```

#include <68hc11.h>
#include <stdio.h>
#include <stdlib.h>
#include <intrpt.h>
#include <reg11.h>
#include <math.h>

#define A_8255 (*(unsigned char*)0x1200)
#define B_8255 (*(unsigned char*)0x1201)
#define C_8255 (*(unsigned char*)0x1202)
#define CNT_8255 (*(unsigned char*)0x1203)
#define SOH 0x01
#define STX 0x02
#define ETX 0x03
#define EOT 0x04
#define ENQ 0x05
#define ACK 0x06
#define NAK 0x15
#define sp 0x20

/*----- Prototype function ----- */
set_sci(void);
send_data(unsigned char);
read_data();
void interrupt sci_int();
/*----- */

unsigned char addr[2];
unsigned char input[20];
char ac;
main()
{
    unsigned char data;
    ROM_VECTOR(SCI_VEC,sci_int);
    CONFIG=0x0d;
    OPTION=0x93;
    CNT_8255=0x82;/* A=out B=in C=out */
    set_sci();
    ei();
    do{
        data=B_8255;
        addr[0]=(data/10)+0x30;
        addr[1]=(data%10)+0x30;
    }while(1);
}
set_sci()
{
    SCCR1=0x00;
    SCCR2=0x2c;
    BAUD =0x30;
}
send_data(data)
unsigned char data;
{
    C_8255=0x01;
    SCDR=data;
    do{
    }while(!(SCSR&0x80));
    C_8255=0;
    read_data()

```

```

{
unsigned char status;
do{
}while(!(SCSR&0x20));
status=SCDR;
return status;
}
interrupt sci_int()
{
char a;
di();
input[0]=read_data();
switch(input[0]){
case EOT:for(a=1;a<=3;a++)
{
input[a]=read_data();
}
if((input[1]==addr[0])&&(input[2]==addr[1]))
{
send_data(input[1]);
send_data(input[2]);
send_data(ACK);
ac=1;
}
break;
case STX:for(a=1;a<=9;a++)
{
input[a]=read_data();
}
if(ac!=1)
{
break;
}
else if((input[4]=='D')&&(input[5]=='O'))
{
send_data(ACK);
A_8255=input[8];
ac=0;
}
break;
default :break;
}
ei();
}

```

```

#include <68hc11.h>
#include <stdio.h>
#include <stdlib.h>
#include <intrpt.h>
#include <reg11.h>

#define A_8255 (*(unsigned char*)0x1200)
#define B_8255 (*(unsigned char*)0x1201)
#define C_8255 (*(unsigned char*)0x1202)
#define CNT_8255 (*(unsigned char*)0x1203)
#define SOH 0x01
#define STX 0x02
#define ETX 0x03
#define EOT 0x04
#define ENQ 0x05
#define ACK 0x06
#define NAK 0x15
#define sp 0x20

/*----- Prototype function ----- */

set_sci(void);
send_data(unsigned char);
read_data();
void interrupt sci_int();

/*----- */

unsigned char addr[2];
unsigned char input[10];
unsigned char dataout[8];
char output[35];
char ac;
main()
{
    unsigned char data;
    ROM_VECTOR(SCI_VEC,sci_int);
    CONFIG=0x0d;
    OPTION=0x93;
    CNT_8255=0x92;/* A=in B=in C=out */
    set_sci();
    ei();
    do{
        data=B_8255;
        addr[0]=(data/10)+0x30;
        addr[1]=(data%10)+0x30;
        /*----- */
        ADCTL=0x10;
        do{
            }while(!(ADCTL&0x80));
        dataout[0]=ADR1;
        dataout[1]=ADR2;
        dataout[2]=ADR3;
        dataout[3]=ADR4;
        /*----- */
        ADCTL=0x14;
        do{
            }while(!(ADCTL&0x80));
        dataout[4]=ADR1;
        dataout[5]=ADR2;
        dataout[6]=ADR3;
        dataout[7]=ADR4;
    }while(1);
}

```

```

}
set_sci()
{
    SCCR1=0x00;
    SCCR2=0x2c;
    BAUD =0x30;
}
send_data(data)
unsigned char data;
{
    C_8255=0x01;
    SCDR=data;
    do{
        }while(!(SCSR&0x80));
    C_8255=0;
}
read_data()
{
    unsigned char status;
    do{
        }while(!(SCSR&0x20));
    status=SCDR;
    return status;
}
interrupt sci_int()
{
    char a;
    char *b;
    di();
    input[0]=read_data();
    switch(input[0]){
        case EOT:for(a=1;a<=3;a++)
            {
                input[a]=read_data();
            }
            if((input[1]==addr[0])&&(input[2]==addr[1]))
            {
                send_data(input[1]);
                send_data(input[2]);
                send_data(ACK);
                ac=1;
            }
            break;
        case STX:for(a=1;a<=8;a++)
            {
                input[a]=read_data();
            }
            if(ac!=1)
            {
                break;
            }
            else if((input[4]=='A')&&(input[5]=='I'))
            {
                b=output;
                sprintf(b,"%cEX AI %02X %02X %02X %02X %02X %02X %02X %02X %c",
                    STX,dataout[0],dataout[1],dataout[2],
                    dataout[3],dataout[4],dataout[5],
                    dataout[6],dataout[7],ETX);
                for(a=0;*(b+a)!='\0';a++)
                {
                    send_data(*(b+a));
                }
            }
            ac=0;
    }
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    }  
    break;  
default :break;  
    }  
    ei();  
}
```



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

#include <68hc11.h>
#include <stdio.h>
#include <stdlib.h>
#include <intrpt.h>
#include <reg11.h>
#include <math.h>

#define A_8255 (*(unsigned char*)0x1200)
#define B_8255 (*(unsigned char*)0x1201)
#define C_8255 (*(unsigned char*)0x1202)
#define CNT_8255 (*(unsigned char*)0x1203)
#define SOH 0x01
#define STX 0x02
#define ETX 0x03
#define EOT 0x04
#define ENQ 0x05
#define ACK 0x06
#define NAK 0x15
#define sp 0x20

/*----- Prototype function ----- */
set_sci(void);
send_data(unsigned char);
read_data();
void interrupt sci_int();
delay();

/*----- */
unsigned char addr[2];
unsigned char input[25];
unsigned char output[8];
char ac;
main()
{
    unsigned char data,channel;
    ROM_VECTOR(SCI_VEC,sci_int);
    CONFIG=0x0d;
    OPTION=0x93;
    CNT_8255=0x82; /* A=out B=in C=out */
    set_sci();
    ei();
    .do{
        data=B_8255;
        addr[0]=(data/10)+0x30;
        addr[1]=(data%10)+0x30;
        /*----- */
        A_8255=0xff;
        for(channel=0;channel<=7;channel++)
        {
            C_8255=channel<<4;
            A_8255=output[channel];
            delay();
        }
    }while(1);
}
delay()
{
    int a;
    for(a=0;a<=500;a++)
    {

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

}
set_sci()
{
  SCCR1=0x00;
  SCCR2=0x2c;
  BAUD =0x30;
}
send_data(data)
unsigned char data;
{
  C_8255=0x01;
  SCDR=data;
  do{
  }while(!(SCSR&0x80));
  C_8255=0;
}
read_data()
{
  unsigned char status;
  do{
  }while(!(SCSR&0x20));
  status=SCDR;
  return status;
}
interrupt sci_int()
{
  char a;
  di();
  input[0]=read_data();
  switch(input[0]){
    case EOT:for(a=1;a<=3;a++)
      {
        input[a]=read_data();
      }
    if((input[1]==addr[0])&&(input[2]==addr[1]))
      {
        send_data(input[1]);
        send_data(input[2]);
        send_data(ACK);
        ac=1;
      }
    break;
    case STX:for(a=1;a<=24;a++)
      {
        input[a]=read_data();
      }
    if(ac!=1)
      {
        break;
      }
    else if((input[4]=='A')&&(input[5]=='O'))
      {
        send_data(ACK);
        for(a=0;a<=7;a++)
          {
            output[a]=input[(a*2)+7];
          }
        ac=0;
      }
    break;
    default :break;
  }
}

```

เอกสารนี้;เอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

MONITOR.BAS - 1

Option Explicit

Global StationSelect As Integer
Global AddressSelect As Integer
Global ParaSelect As Integer
Global Getchange As Integer

Type InterfaceInfo

StationNumber(0 To 31) As String * 25
Port As String * 5
Baud As Integer
Parity As String * 4
Data As Integer
Stop As Integer

End Type

Type PageInfo

Name As String * 30
Scan As Integer
Enable As Integer

End Type

Type LineInfo

FirstName As String * 25
LastName As String * 30
Units As String * 10
StationNumber As Integer
Active As Integer
Slot As Integer
Channel As Integer
Bit As Integer
AlarmEnable As Integer
HighAlarm As Single
LowAlarm As Single
Diff As Single
Zero As Single
Span As Single
'Alarm
Astation As Integer
Achannel As Integer
Avalue As Single

End Type

Type DviewInfo

Slot0(0 To 7) As Integer
Slot1(0 To 7) As Integer
Slot2(0 To 1) As Integer
StationInfo As Integer

End Type

Type PdataInfo

Station(1 To 10) As Integer
Slot(1 To 10) As Integer
Channel(1 To 10) As Integer
Bit(1 To 10) As Integer
Alarm(1 To 10) As Integer
Display(1 To 10) As Integer
ZeroData(1 To 10) As Single
SpanData(1 To 10) As Single
Dif(1 To 10) As Single
LAlarm(1 To 10) As Single
HAlarm(1 To 10) As Single

Astation(1 To 10) As Integer

Achannel(1 To 10) As Integer

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

    Avalue(1 To 10) As Single
End Type
Type GraphData
MONITOR.BAS - 2

```

```

    GhData(1 To 2000) As Single
End Type

```

```

Type Drecord
    Ddate As String * 10
    Dtime As String * 8
    Dvalue(1 To 10) As Single
End Type

```

```

Type Alarmset
    StNumber As Integer
    Channel As Integer
    Value As Single
End Type

```

```

Global LinkStatus(0 To 31) As String * 40
Global Station(0 To 31) As DviewInfo
Global Pdata(1 To 10) As PdataInfo
Global RxCount
Global TxCount
Global Tcount1(1 To 10) As Integer
Global PageSelect%

```



FRMHISTO.FRM - 1

```
Option Explicit
Dim Gdata As GraphData
Dim PageSelect As Integer
Dim CurrentRecord As Long
Dim LastRecord As Long
Dim FileNum As Long
Dim Drec As Drecord
Dim Start As Integer
Dim Stop1 As Integer
Dim Point1 As Integer
Dim File(1 To 10) As String * 12
```

```
Sub de ()
frmHistory.Caption = Str$(Gdata.GhData(CurrentRecord))
End Sub
```

```
Sub DisplayGraph ()
Dim RecLen As Long
Dim Count As Long
Dim LineCount As Long
Dim Count2 As Integer
'
For Count = 1 To 2000
  Gdata.GhData(Count) = 0#
Next
Graph1.DataReset = 9
FileNum = FreeFile
RecLen = Len(Drec)
Open File(PageSelect) For Random As FileNum Len = RecLen
LastRecord = FileLen(File(PageSelect)) / RecLen
If LastRecord = 0 Then
  LastRecord = 1
End If
Select Case LastRecord
  Case Is <= 50
    For Count = 1 To LastRecord
      Get #FileNum, Count, Drec
      For LineCount = 1 To 10
        Gdata.GhData(((LineCount - 1) * 200) + Count) = Drec.Dvalue(LineCount)
      Next LineCount
    Next Count
    DrawGraph LastRecord
  Case Is > 50
    Count2 = 0
    Select Case ((Point1 - 1) * 50) + 50
      Case Is <= LastRecord
        For Count = ((Point1 - 1) * 50) + 1 To ((Point1 - 1) * 50) + 50
          Count2 = Count2 + 1
          Get #FileNum, Count, Drec
          For LineCount = 1 To 10
            Gdata.GhData(((LineCount - 1) * 200) + Count2) = Drec.Dvalue(LineCount)
          Next LineCount
        Next Count
        DrawGraph 50
      Case Is > LastRecord
        For Count = ((Point1 - 1) * 50) + 1 To ((Point1 - 1) * 50) + 50
          Count2 = Count2 + 1
          Get #FileNum, Count, Drec
          For LineCount = 1 To 10
            Gdata.GhData(((LineCount - 1) * 200) + Count2) = Drec.Dvalue(LineCount)
```

เอกสารนี้เป็นเอกสารสงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่นิยมนำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

        Next LineCount
    Next Count
    DrawGraph (((Point1 - 1) * 50) + 50) - LastRecord
FRMHISTO.FRM - 2

    End Select
End Select
Close #FileNum
Point1 = Point1 + 1
End Sub

```

```

Sub DrawGraph (Tcount1 As Long)
Dim S As Integer
Dim p As Integer
Dim Color As Integer
Dim Ran As Integer
Dim i As Integer

```

```

    Graph1.IndexStyle = 1
    Graph1.NumSets = 10
    Graph1.NumPoints = 200
    For i = 1 To Graph1.NumSets
        Graph1.ThisSet = i
        For p = 1 To Tcount1
            Graph1.ThisPoint = Tcount1 - p + 1
            Graph1.XPosData = 1 - p
            Graph1.GraphData = Gdata.GhData(((i - 1) * 200) + p)
        Next p
    Next i
    Color = 5
    S = 0
    For i = 1 To Graph1.NumSets
        Graph1.ThisSet = i
        Graph1.ColorData = Color
        Color = Color + 1
        'Graph1.SymbolData = s
        Graph1.PatternData = 1
        Graph1.LegendText = Str$(i)
        S = S + 1
    Next i
    Graph1.Ticks = 2
    Graph1.PatternData = 1
    Graph1.YAxisStyle = 2
    Graph1.YAxisTicks = 10
    Graph1.TickEvery = 50
    Graph1.DrawMode = 3
End Sub

```

```

Sub cmdFforward_Click ()
Dim Count As Long
Dim LineCount As Long
Dim Count2 As Integer
Dim ReLen As Long
Point1 = Point1 - 2
If Point1 < 1 Then
    Point1 = 1
End If

For Count = 1 To 2000
    Gdata.GhData(Count) = 0#
Next
Graph1.DataReset = 9
FileNum = FreeFile

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

RecLen = Len(Drec)
Open File(PageSelect) For Random As FileNum Len = RecLen
LastRecord = FileLen(File(PageSelect)) / RecLen
If LastRecord = 0 Then
  LastRecord = 1
End If
FRMHISTO.FRM - 3

Select Case LastRecord
  Case Is <= 50
    For Count = 1 To LastRecord
      Get #FileNum, Count, Drec
      For LineCount = 1 To 10
        Gdata.GhData(((LineCount - 1) * 200) + Count) = Drec.Dvalue(LineCount)
      Next LineCount
    Next Count
    DrawGraph LastRecord
  Case Is > 50
    Count2 = 0
    Select Case ((Point1 - 1) * 50) + 50
      Case Is <= LastRecord
        For Count = ((Point1 - 1) * 50) + 1 To ((Point1 - 1) * 50) + 50
          Count2 = Count2 + 1
          Get #FileNum, Count, Drec
          For LineCount = 1 To 10
            Gdata.GhData(((LineCount - 1) * 200) + Count2) = Drec.Dvalue(LineCount)
          Next LineCount
        Next Count
        DrawGraph 50
      Case Is > LastRecord
        For Count = ((Point1 - 1) * 50) + 1 To ((Point1 - 1) * 50) + 50
          Count2 = Count2 + 1
          Get #FileNum, Count, Drec
          For LineCount = 1 To 10
            Gdata.GhData(((LineCount - 1) * 200) + Count2) = Drec.Dvalue(LineCount)
          Next LineCount
        Next Count
        DrawGraph (((Point1 - 1) * 50) + 50) - LastRecord
    End Select
  End Select
Close #FileNum
End Sub

```

```

Sub cmdForward_Click (Index As Integer)
Dim RecLen As Long
Dim Count As Long
Dim LineCount As Long
Dim Count2 As Integer
Point1 = Point1 - 1
If Point1 < 1 Then
  Point1 = 1
End If

For Count = 1 To 2000
  Gdata.GhData(Count) = 0#
Next
Graph1.DataReset = 9
FileNum = FreeFile
RecLen = Len(Drec)
Open File(PageSelect) For Random As FileNum Len = RecLen
LastRecord = FileLen(File(PageSelect)) / RecLen
If LastRecord = 0 Then
  LastRecord = 1

```

```

End If
Select Case LastRecord
  Case Is <= 50
    For Count = 1 To LastRecord
      Get #FileNum, Count, Drec
      For LineCount = 1 To 10
        Gdata.GhData(((LineCount - 1) * 200) + Count) = Drec.Dvalue(LineCount)
      Next LineCount
    Next Count
FRMHISTO.FRM - 4

```

```

DrawGraph LastRecord
Case Is > 50
Count2 = 0
Select Case ((Point1 - 1) * 50) + 50
  Case Is <= LastRecord
    For Count = ((Point1 - 1) * 50) + 1 To ((Point1 - 1) * 50) + 50
      Count2 = Count2 + 1
      Get #FileNum, Count, Drec
      For LineCount = 1 To 10
        Gdata.GhData(((LineCount - 1) * 200) + Count2) = Drec.Dvalue(LineCount)
      Next LineCount
    Next Count
  DrawGraph 50
  Case Is > LastRecord
    For Count = ((Point1 - 1) * 50) + 1 To ((Point1 - 1) * 50) + 50
      Count2 = Count2 + 1
      Get #FileNum, Count, Drec
      For LineCount = 1 To 10
        Gdata.GhData(((LineCount - 1) * 200) + Count2) = Drec.Dvalue(LineCount)
      Next LineCount
    Next Count
  DrawGraph (((Point1 - 1) * 50) + 50) - LastRecord
End Select
End Select
Close #FileNum
End Sub

```

```

Sub cmdfrevers_Click (Index As Integer)
Dim RecLen As Long
Dim Count As Long
Dim LineCount As Long
Dim Count2 As Integer

Point1 = Point1 + 2
For Count = 1 To 2000
  Gdata.GhData(Count) = 0#
Next
Graph1.DataReset = 9
FileNum = FreeFile
RecLen = Len(Drec)
Open File(PageSelect) For Random As FileNum Len = RecLen
LastRecord = FileLen(File(PageSelect)) / RecLen
If LastRecord = 0 Then
  LastRecord = 1
End If
Select Case LastRecord
  Case Is <= 50
    For Count = 1 To LastRecord
      Get #FileNum, Count, Drec
      For LineCount = 1 To 10
        Gdata.GhData(((LineCount - 1) * 200) + Count) = Drec.Dvalue(LineCount)
      Next LineCount

```

```

Next Count
DrawGraph LastRecord
Case Is > 50
Count2 = 0
Select Case ((Point1 - 1) * 50) + 50
Case Is <= LastRecord
For Count = ((Point1 - 1) * 50) + 1 To ((Point1 - 1) * 50) + 50
Count2 = Count2 + 1
Get #FileNum, Count, Drec
For LineCount = 1 To 10
Gdata.GhData(((LineCount - 1) * 200) + Count2) = Drec.Dvalue(LineCount)
Next LineCount
FRMHISTO.FRM -/5

```

```

Next Count
DrawGraph 50
Case Is > LastRecord
For Count = ((Point1 - 1) * 50) + 1 To ((Point1 - 1) * 50) + 50
Count2 = Count2 + 1
Get #FileNum, Count, Drec
For LineCount = 1 To 10
Gdata.GhData(((LineCount - 1) * 200) + Count2) = Drec.Dvalue(LineCount)
Next LineCount
Next Count
DrawGraph (((Point1 - 1) * 50) + 50) - LastRecord
End Select
End Select
Close #FileNum
End Sub

Sub cmdPage1_Click ()
PageSelect = 1
Point1 = 1
DisplayGraph
End Sub

Sub cmdPage10_Click ()
PageSelect = 10
Point1 = 1
DisplayGraph

End Sub

Sub cmdPage2_Click ()
PageSelect = 2
Point1 = 1
DisplayGraph

End Sub

Sub cmdPage3_Click ()
PageSelect = 3
Point1 = 1
DisplayGraph

End Sub

Sub cmdPage4_Click ()
PageSelect = 4
Point1 = 1
DisplayGraph

End Sub

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
Sub cmdPage5_Click ()
  PageSelect = 5
  Point1 = 1
  DisplayGraph
```

```
End Sub
```

```
Sub cmdPage6_Click ()
  PageSelect = 6
  Point1 = 1
  DisplayGraph
```

```
End Sub
```

```
FRMHISTO.FRM - 6
```

```
Sub cmdPage7_Click ()
  PageSelect = 7
  Point1 = 1
  DisplayGraph
```

```
End Sub
```

```
Sub cmdPage8_Click ()
  PageSelect = 8
  Point1 = 1
  DisplayGraph
```

```
End Sub
```

```
Sub cmdPage9_Click ()
  PageSelect = 9
  Point1 = 1
  DisplayGraph
```

```
End Sub
```

```
Sub cmdPrint_Click ()
  Graph1.DrawMode = 5
  End Sub
```

```
Sub cmdReverse_Click ()
  Dim RecLen As Long
  Dim Count As Long
  Dim LineCount As Long
  Dim Count2 As Integer
```

```
  Point1 = Point1 + 1
  For Count = 1 To 2000
    Gdata.GhData(Count) = 0#
  Next
  Graph1.DataReset = 9
  FileNum = FreeFile
  RecLen = Len(Drec)
  Open File(PageSelect) For Random As FileNum Len = RecLen
  LastRecord = FileLen(File(PageSelect)) / RecLen
  If LastRecord = 0 Then
    LastRecord = 1
  End If
  Select Case LastRecord
    Case Is <= 50
      For Count = 1 To LastRecord
        Get #FileNum, Count, Drec
```

```

For LineCount = 1 To 10
  Gdata.GhData(((LineCount - 1) * 200) + Count) = Drec.Dvalue(LineCount)
Next LineCount
Next Count
DrawGraph LastRecord
Case Is > 50
  Count2 = 0
  Select Case ((Point1 - 1) * 50) + 50
  Case Is <= LastRecord
    For Count = ((Point1 - 1) * 50) + 1 To ((Point1 - 1) * 50) + 50
      Count2 = Count + 1
      Get #FileNum, Count, Drec
      For LineCount = 1 To 10
        Gdata.GhData(((LineCount - 1) * 200) + Count2) = Drec.Dvalue(LineCount)
      Next LineCount
    Next Count
  DrawGraph 50
FRMHISTO.FRM - 7

Case Is > LastRecord
  For Count = ((Point1 - 1) * 50) + 1 To ((Point1 - 1) * 50) + 50
    Count2 = Count2 + 1
    Get #FileNum, Count, Drec
    For LineCount = 1 To 10
      Gdata.GhData(((LineCount - 1) * 200) + Count2) = Drec.Dvalue(LineCount)
    Next LineCount
  Next Count
  DrawGraph (((Point1 - 1) * 50) + 50) - LastRecord
End Select
End Select
Close #FileNum
End Sub

Sub cmdExit_Click ()
  Unload Me
End Sub

Sub Form_Load ()
  File(1) = "Page1.rec"
  File(2) = "Page2.rec"
  File(3) = "Page3.rec"
  File(4) = "Page4.rec"
  File(5) = "Page5.rec"
  File(6) = "Page6.rec"
  File(7) = "Page7.rec"
  File(8) = "Page8.rec"
  File(9) = "Page9.rec"
  File(10) = "Page10.rec"
  Point1 = 1
End Sub

```

FRMLDE.FRM - 1

Option Explicit

```
Sub cmdOK_Click ()
    Unload Me
End Sub
```

```
Sub Form_Load ()
    Dim Count%
    Dim Count1%
    Dim FileNum1 As Long
    Dim Ldata As LineInfo
    Dim RecordCount As Long
    frame3d1.Caption = "Page " + Str$(PageSelect)
    FileNum1 = FreeFile
    RecordCount = Len(Ldata)
    Open "Lineset.set" For Random As FileNum1 Len = RecordCount
    For Count = 1 To 10
        For Count1 = 1 To 10
            Get #FileNum1, (PageSelect - 1) * 10 + Count1, Ldata
            Check3d(Count1).Value = Ldata.Active
            Check3d(Count1).Caption = Ldata.Lastname
        Next Count1
    Next Count
    Close #FileNum1
End Sub
```



FRMPDE.FRM - 1

```
Option Explicit
Dim Page As PageInfo
Dim FileNum As Long
Dim RecordCount As Long
```

```
Sub cmdOK_Click ()
Unload Me
End Sub
```

```
Sub Form_Load ()
Dim Count%
FileNum = FreeFile
RecordCount = Len(Page)
Open "Pageset.set" For Random As FileNum Len = RecordCount
For Count = 1 To 10
Get #FileNum, Count, Page
Check3D(Count).Value = Page.Enable
Label(Count).Caption = Trim(Page.Name)
Next Count
Close #FileNum
End Sub
```



GETADD.FRM - 1

Option Explicit

Dim Addr As Integer

Dim AddSelect As Integer

Dim Stype As Integer

Dim Spara As Integer

Sub cboAddress_Click ()

 AddressSelect = cboAddress.ListIndex

End Sub

Sub cmdExit_Click ()

 Unload Me

End Sub

Sub cmdReturn_Click ()

 Getchange = 1

 Unload Me

End Sub

Sub Form_Load ()

Dim Account As Integer

 lbStationType.AddItem "Analog Input"

 lbStationType.AddItem "Analog Output"

 lbStationType.AddItem "Digital Input"

 lbStationType.AddItem "Digital Output"

 For Account = 0 To 31 Step 1

 cboAddress.AddItem Format\$(Account, "00")

 Next Account

End Sub

Sub lbStationPara_Click ()

 Paraselect = lbStationPara.ListIndex

End Sub

Sub lbStationType_Click ()

Dim SeCount As Integer

 StationSelect = lbStationType.ListIndex

 Select Case StationSelect

 Case 0

 StationSelect = lbStationType.ListIndex

 lbStationPara.Clear

 For SeCount = 1 To 8 Step 1

 lbStationPara.AddItem "Analog Input Channel " + Str\$(SeCount)

 Next SeCount

 Case 1

 lbStationPara.Clear

 For SeCount = 1 To 8 Step 1

 lbStationPara.AddItem "Analog Output Channel " + Str\$(SeCount)

 Next SeCount

 Case 2

 lbStationPara.Clear

 For SeCount = 1 To 8 Step 1

 lbStationPara.AddItem "Digital Input Channel " + Str\$(SeCount)

 Next SeCount

 Case 3

 lbStationPara.Clear

 For SeCount = 1 To 8 Step 1

 lbStationPara.AddItem "Digital Output Channel " + Str\$(SeCount)

 Next SeCount

 End Select

เอกสารนี้เป็นเอกสารต้นฉบับสำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

GRAPH.FRM - 1

Option Explicit

Dim FileNum As Long
 Dim FileNum1 As Long
 Dim FileNum2 As Long
 Dim RecordCount As Long
 Dim Page As PageInfo
 Dim Gdata(1 To 10) As GraphData
 Dim Tcount As Integer
 Dim DataRecord As DviewInfo
 Dim Inter As InterfaceInfo
 Dim Drec As Drecord
 Dim Ldata As LineInfo
 Dim InputSpan As Integer
 Dim Regend(1 To 10) As Single

Sub GetDataIn (Cns As Integer, PointsN As Integer)

Dim Cns As Integer
 Dim p As Integer
 Dim temp As Single
 For Cns = 0 To 9
 Select Case Pdata(Cns + 1).Slot(Cns + 1)
 Case 0
 temp = Station(Pdata(PageSelect).Station(Cns + 1)).Slot0(Pdata(PageSelect).Channel(Cns + 1))
 Gdata(PageSelect).GhData(Cns + ((Cns) * PointsN)) = temp
 Case 1
 temp = Station(Pdata(PageSelect).Station(Cns + 1)).Slot1(Pdata(PageSelect).Channel(Cns + 1))
 Gdata(PageSelect).GhData(Cns + ((Cns) * PointsN)) = temp
 End Select
 Next Cns
 End Sub

Sub Timeserv ()

Dim S As Integer
 Dim p As Integer
 Dim Color As Integer
 Dim Ran As Integer
 Dim i As Integer
 If Tcount1(PageSelect) = 1 Then
 Graph1.DataReset = 9
 Graph1.DrawMode = 1
 End If
 Graph1.IndexStyle = 1
 Graph1.NumSets = 10
 Graph1.NumPoints = 200
 For i = 1 To Graph1.NumSets
 Graph1.ThisSet = i
 For p = 1 To Tcount1(PageSelect)
 Graph1.ThisPoint = Tcount1(PageSelect) - p + 1
 Graph1.XPosData = 1 - p
 Graph1.GraphData = Gdata(PageSelect).GhData(((i - 1) * 200) + p)
 Next p
 Regend(i) = Gdata(PageSelect).GhData(((i - 1) * 200) + Tcount1(PageSelect))
 Next i
 Color = 5
 S = 0

เอกสารนี้เป็นทรัพย์สินสงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไปว่ากรณิดยั้งสี่ง ลึกข้งห้งนปีให้ดอเบงโลงบ้อหา และต้องอ้งลึงถึงอ้วของเอกสารทคครั้งทีมีการงอไปให้

```

For i = 1 To Graph1.NumSets
    Graph1.ThisSet = i
    Graph1.ColorData = Color
GRAPH.FRM - 2

    Color = Color + 1
    'Graph1.SymbolData = s
    Graph1.PatternData = 1
    Graph1.LegendText = Format$(i, "00") + " " + Format$(Regend(i), "0000.00
")
    S = S + 1
Next i
Graph1.Ticks = 2
Graph1.PatternData = 1
Graph1.YAxisStyle = 2
Graph1.YAxisTicks = 10
Graph1.TickEvery = 50
Graph1.DrawMode = 3
End Sub

Sub cmdExit_Click ()
    Close
    Unload frmSim
    Unload frmview
    Unload Me
End Sub

Sub cmdLink_Click ()
    'frmlink.Show
End Sub

Sub cmdPage1_Click ()
    PageSelect = 1
    mnuPage.Enabled = True
    mnuLine.Enabled = True
    Graph1.DataReset = 9
    Graph1.Visible = True
    Get #FileNum, 1, Page
    frmGraph.Caption = "Recorder " + "Page[1]"
    Graph1.GraphTitle = Page.Name
    Timeserv
End Sub

Sub cmdPage10_Click ()
    mnuPage.Enabled = True
    mnuLine.Enabled = True
    PageSelect = 10
    Graph1.DataReset = 9
    Graph1.Visible = True

    Get #FileNum, 10, Page
    frmGraph.Caption = "Recorder " + "Page[10]"
    Graph1.GraphTitle = Page.Name
    Timeserv
End Sub

Sub cmdPage2_Click ()
    mnuPage.Enabled = True
    mnuLine.Enabled = True
    PageSelect = 2
    Graph1.DataReset = 9
    Graph1.Visible = True
    Get #FileNum, 2, Page

```

```

frmGraph.Caption = "Recorder " + "Page[2]"
Graph1.GraphTitle = Page.Name
Timeserv
End Sub

```

```

Sub cmdPage3_Click ()
GRAPH.FRM - 3

```

```

mnuPage.Enabled = True
mnuLine.Enabled = True
Graph1.DataReset = 9
Graph1.Visible = True
PageSelect = 3
Get #FileNum, 3, Page
frmGraph.Caption = "Recorder " + "Page[3]"
Graph1.GraphTitle = Page.Name
Timeserv
End Sub

```

```

Sub cmdPage4_Click ()
mnuPage.Enabled = True
mnuLine.Enabled = True
Graph1.DataReset = 9
Graph1.Visible = True
PageSelect = 4
Get #FileNum, 4, Page
frmGraph.Caption = "Recorder " + "Page[4]"
Graph1.GraphTitle = Page.Name
Timeserv
End Sub

```

```

Sub cmdPage5_Click ()
mnuPage.Enabled = True
mnuLine.Enabled = True
Graph1.DataReset = 9
Graph1.Visible = True
PageSelect = 5
Get #FileNum, 5, Page
frmGraph.Caption = "Recorder " + "Page[5]"
Graph1.GraphTitle = Page.Name
Timeserv
End Sub

```

```

Sub cmdPage6_Click ()
mnuPage.Enabled = True
mnuLine.Enabled = True
Graph1.DataReset = 9
Graph1.Visible = True
PageSelect = 6
Get #FileNum, 6, Page
frmGraph.Caption = "Recorder " + "Page[6]"
Graph1.GraphTitle = Page.Name
Timeserv
End Sub

```

```

Sub cmdPage7_Click ()
mnuPage.Enabled = True
mnuLine.Enabled = True
Graph1.DataReset = 9
Graph1.Visible = True
PageSelect = 7
Get #FileNum, 7, Page
frmGraph.Caption = "Recorder " + "Page[7]"

```

```

Graph1.GraphTitle = Page.Name
Timeserv
End Sub

```

```

Sub cmdPage8_Click ()
    mnuPage.Enabled = True
    mnuLine.Enabled = True
    Graph1.DataReset = 9
    Graph1.Visible = True
GRAPH.FRM - 4

```

```

PageSelect = 8
Get #FileNum, 8, Page
frmGraph.Caption = "Recorder " + "Page[8]"
Graph1.GraphTitle = Page.Name
Timeserv
End Sub

```

```

Sub cmdPage9_Click ()
    mnuPage.Enabled = True
    mnuLine.Enabled = True
    Graph1.DataReset = 9
    Graph1.Visible = True
    PageSelect = 9
    Get #FileNum, 9, Page
    frmGraph.Caption = "Recorder " + "Page[9]"
    Graph1.GraphTitle = Page.Name
    Timeserv
End Sub

```

```

Sub cmdPrint_Click ()
    Timer1.Enabled = False
    Timer2.Enabled = False
    Timer3.Enabled = False
    Timer4.Enabled = False
    Timer5.Enabled = False
    Timer6.Enabled = False
    Timer7.Enabled = False
    Timer8.Enabled = False
    Timer9.Enabled = False
    Timer10.Enabled = False
    Unload frmview
    Graph1.DrawMode = 5
    Load frmview
    Timer1.Enabled = True
    Timer2.Enabled = True
    Timer3.Enabled = True
    Timer4.Enabled = True
    Timer5.Enabled = True
    Timer6.Enabled = True
    Timer7.Enabled = True
    Timer8.Enabled = True
    Timer9.Enabled = True
    Timer10.Enabled = True
End Sub

```

```

Sub Form_Load ()
    Dim Count As Integer
    Dim Count1 As Integer
    Dim Count2 As Integer
    Dim Selected As Integer

```

```

Load frmSim
Load frmview

```

```

mnuPage.Enabled = False
mnuLine.Enabled = False
Selected = 0
InputSpan = &HFF
'
FileNum2 = FreeFile
RecordCount = Len(Inter)
Open "Inter.set" For Random As FileNum2 Len = RecordCount
Get #FileNum2, 1, Inter
'
FileNum = FreeFile
RecordCount = Len(Page)
GRAPH.FRM - 5

```

```

Open "Pageset.set" For Random As FileNum Len = RecordCount
For Count = 1 To 10
  Get #FileNum, Count, Page
  Select Case Count
    Case 1
      If Page.Enable = False Then
        cmdPage1.Enabled = False
        Timer1.Enabled = False
      End If
      Timer1.Interval = Page.Scan * 1000
    Case 2
      If Page.Enable = False Then
        cmdPage2.Enabled = False
        Timer2.Enabled = False
      End If
      Timer2.Interval = Page.Scan * 1000
    Case 3
      If Page.Enable = False Then
        cmdPage3.Enabled = False
        Timer3.Enabled = False
      End If
      Timer3.Interval = Page.Scan * 1000
    Case 4
      If Page.Enable = False Then
        cmdPage4.Enabled = False
        Timer4.Enabled = False
      End If
      Timer4.Interval = Page.Scan * 1000
    Case 5
      If Page.Enable = False Then
        cmdPage5.Enabled = False
        Timer5.Enabled = False
      End If
      Timer5.Interval = Page.Scan * 1000
    Case 6
      If Page.Enable = False Then
        cmdPage6.Enabled = False
        Timer6.Enabled = False
      End If
      Timer6.Interval = Page.Scan * 1000
    Case 7
      If Page.Enable = False Then
        cmdPage7.Enabled = False
        Timer7.Enabled = False
      End If
      Timer7.Interval = Page.Scan * 1000
    Case 8
      If Page.Enable = False Then
        cmdPage8.Enabled = False

```

```

    Timer8.Enabled = False
End If
Timer8.Interval = Page.Scan * 1000
Case 9
If Page.Enable = False Then
    cmdPage9.Enabled = False
    Timer9.Enabled = False
End If
Timer9.Interval = Page.Scan * 1000
Case 10
If Page.Enable = False Then
    cmdPage10.Enabled = False
    Timer10.Enabled = False
End If
Timer10.Interval = Page.Scan * 1000
GRAPH.FRM - 6

```

```

End Select
Next Count

FileNum1 = FreeFile
RecordCount = Len(Ldata)
Open "Lineset.set" For Random As FileNum1 Len = RecordCount
Count2 = 1
For Count = 1 To 10
    For Count1 = 1 To 10
        Get #FileNum1, Count2, Ldata
        Pdata(Count).Station(Count1) = Ldata.StationNumber
        Pdata(Count).Slot(Count1) = Ldata.Slot
        Pdata(Count).Channel(Count1) = Ldata.Channel
        Pdata(Count).Bit(Count1) = Ldata.Bit
        Pdata(Count).Alarm(Count1) = Ldata.AlarmEnable
        Pdata(Count).Display(Count1) = Ldata.Active
        Pdata(Count).LAlarm(Count1) = Ldata.LowAlarm
        Pdata(Count).HAlarm(Count1) = Ldata.HighAlarm
        Pdata(Count).Dif(Count1) = Ldata.Diff
        Pdata(Count).ZeroData(Count1) = Ldata.Zero
        Pdata(Count).SpanData(Count1) = Ldata.Span
        Pdata(Count).Astation(Count1) = Ldata.Astation
        Pdata(Count).AChannel(Count1) = Ldata.AChannel
        Pdata(Count).Avalue(Count1) = Ldata.Avalue
        Count2 = Count2 + 1
    Next Count1
Next Count
Close #FileNum1
End Sub

Sub mnuLine_Click ()
    frmlinedetails.Show
End Sub

Sub mnuPage_Click ()
    frmPageDetail.Show
End Sub

Sub mnuSim_Click ()
    frmSim.Show
    frmview.Show
End Sub

Sub Timer1_Timer ()
    Dim Cns%
    Dim temp%

```

```

Dim InputData#
Dim tempo#
Dim Al As String
Dim F As Long
Dim LastRecord As Long
Dim ReLen As Long

Tcount1(1) = Tcount1(1) + 1
If Tcount1(1) > 200 Then
  Tcount1(1) = 1
  For Cns = 1 To 2000
    Gdata(1).GhData(Cns) = 0#
  Next Cns
End If
For Cns = 0 To 9
  Select Case Pdata(1).Slot(Cns + 1)
  Case 0
    InputData = 0#
GRAPH.FRM - 7

    If Pdata(1).Display(Cns + 1) > 0 Then
      temp = Station(Pdata(1).Station(Cns + 1)).Slot0(Pdata(1).Channel(Cns + 1))
    )
      InputData = ((temp / InputSpan) * (Pdata(1).SpanData(Cns + 1) - Pdata(1).ZeroData(Cns + 1)) + Pdata(1).ZeroData(Cns + 1))
      Gdata(1).GhData(Tcount1(1) + ((Cns) * 200)) = InputData
      If Pdata(1).Alarm(Cns + 1) > 0 Then
        Select Case InputData
          Case Is > Pdata(1).HAlarm(Cns + 1)
            If Trim(Inter.StationNumber(Pdata(1).Astation(Cns + 1))) = "Digital Output" Then
              Select Case Pdata(1).Avalue(Cns + 1)
                Case Is >= 1
                  Station(Pdata(1).Astation(Cns + 1)).Slot2(1) = Station(Pdata(1).Astation(Cns + 1)).Slot2(1) Or 2 ^ Pdata(1).AChannel(Cns + 1)
                Case Is < 1
                  Station(Pdata(1).Astation(Cns + 1)).Slot2(1) = Station(Pdata(1).Astation(Cns + 1)).Slot2(1) And (&HFF - 2 ^ Pdata(1).AChannel(Cns + 1))
                End Select
              End If
            If Trim(Inter.StationNumber(Pdata(1).Astation(Cns + 1))) = "Analog Output" Then
              tempo = (Pdata(1).Avalue(Cns + 1) / 5) * &HFF
              Station(Pdata(1).Astation(Cns + 1)).Slot1(Pdata(1).AChannel(Cns + 1)) = CInt(tempo)
            End If
          Case Is < Pdata(1).LAlarm(Cns + 1)
            If Trim(Inter.StationNumber(Pdata(1).Astation(Cns + 1))) = "Digital Output" Then
              Select Case Pdata(1).Avalue(Cns + 1)
                Case Is >= 1
                  Station(Pdata(1).Astation(Cns + 1)).Slot2(1) = Station(Pdata(1).Astation(Cns + 1)).Slot2(1) Or 2 ^ Pdata(1).AChannel(Cns + 1)
                Case Is < 1
                  Station(Pdata(1).Astation(Cns + 1)).Slot2(1) = Station(Pdata(1).Astation(Cns + 1)).Slot2(1) And (&HFF - 2 ^ Pdata(1).AChannel(Cns + 1))
                End Select
              End If
            If Trim(Inter.StationNumber(Pdata(1).Astation(Cns + 1))) = "Analog Output" Then
              tempo = (Pdata(1).Avalue(Cns + 1) / 5) * &HFF

```

```

        Station(Pdata(1).Astation(Cns + 1)).Slot1(Pdata(1).AChannel(Cns + 1)) = CInt(tempo)
    End If
    Case Is < Pdata(1).HAlarm(Cns + 1) - Pdata(1).Dif(Cns + 1)
        If Trim(Inter.StationNumber(Pdata(1).Astation(Cns + 1))) = "Digital Output" Then
            Select Case Pdata(1).Avalue(Cns + 1)
                Case Is >= 1
                    Station(Pdata(1).Astation(Cns + 1)).Slot2(1) = Station(Pdata(1).Astation(Cns + 1)).Slot2(1) And 2 ^ Pdata(1).Avalue(Cns + 1)
                Case Is < 1
                    Station(Pdata(1).Astation(Cns + 1)).Slot2(1) = Station(Pdata(1).Astation(Cns + 1)).Slot2(1) Or (&HFF - 2 ^ Pdata(1).Avalue(Cns + 1))
            End Select
        End If
        If Trim(Inter.StationNumber(Pdata(1).Astation(Cns + 1))) = "Analog Output" Then
            Station(Pdata(1).Astation(Cns + 1)).Slot1(Pdata(1).AChannel(Cns + 1)) = 0
        End If
    GRAPH.FRM - 8
    Case Is > Pdata(1).HAlarm(Cns + 1) + Pdata(1).Dif(Cns + 1)
        If Trim(Inter.StationNumber(Pdata(1).Astation(Cns + 1))) = "Digital Output" Then
            Select Case Pdata(1).Avalue(Cns + 1)
                Case Is >= 1
                    Station(Pdata(1).Astation(Cns + 1)).Slot2(1) = Station(Pdata(1).Astation(Cns + 1)).Slot2(1) And 2 ^ Pdata(1).Avalue(Cns + 1)
                Case Is < 1
                    Station(Pdata(1).Astation(Cns + 1)).Slot2(1) = Station(Pdata(1).Astation(Cns + 1)).Slot2(1) Or (&HFF - 2 ^ Pdata(1).Avalue(Cns + 1))
            End Select
        End If
        If Trim(Inter.StationNumber(Pdata(1).Astation(Cns + 1))) = "Analog Output" Then
            Station(Pdata(1).Astation(Cns + 1)).Slot1(Pdata(1).AChannel(Cns + 1)) = 0
        End If
    End Select
End If
End If
Drec.Dvalue(Cns + 1) = InputData
Case 1
    InputData = 0#
    If Pdata(1).Display(Cns + 1) > 0 Then
        temp = Station(Pdata(1).Station(Cns + 1)).Slot1(Pdata(1).Channel(Cns + 1))
    End If
    InputData = ((temp / InputSpan) * (Pdata(1).SpanData(Cns + 1) - Pdata(1).ZeroData(Cns + 1)) + Pdata(1).ZeroData(Cns + 1))
    Gdata(1).GhData(Tcount1(1) + ((Cns) * 200)) = InputData
End If
Drec.Dvalue(Cns + 1) = InputData
Case 2
    InputData = 0#
    If Pdata(1).Display(Cns + 1) > 0 Then
        temp = Station(Pdata(1).Station(Cns + 1)).Slot2(Pdata(1).Channel(Cns + 1)) And 2 ^ Pdata(1).Bit(Cns + 1)
        If temp Then
            temp = InputSpan
        End If
    End If

```

```

        InputData = ((temp / InputSpan) * (Pdata(1).SpanData(Cns + 1) - Pdata(1)
        ).ZeroData(Cns + 1)) + Pdata(1).ZeroData(Cns + 1)
        Gdata(1).GhData(Tcount1(1) + ((Cns) * 200)) = InputData
        If Pdata(1).Alarm(Cns + 1) > 0 Then
            Select Case InputData
                Case Is > Pdata(1).HAlarm(Cns + 1)
                    If Trim(Inter.StationNumber(Pdata(1).Astation(Cns + 1)))
                    = "Digital Output" Then
                        Select Case Pdata(1).Avalue(Cns + 1)
                            Case Is >= 1
                                Station(Pdata(1).Astation(Cns + 1)).Slot2(1) = Sta
                                tion(Pdata(1).Astation(Cns + 1)).Slot2(1) Or 2 ^ Pdata(1).AChannel(Cns + 1)
                            Case Is < 1
                                Station(Pdata(1).Astation(Cns + 1)).Slot2(1) = Sta
                                tion(Pdata(1).Astation(Cns + 1)).Slot2(1) And (&HFF - 2 ^ Pdata(1).AChannel(Cns
                                + 1))
                        End Select
                    End If
                    If Trim(Inter.StationNumber(Pdata(1).Astation(Cns + 1)))
                    = "Analog Output" Then
                        tempo = (Pdata(1).Avalue(Cns + 1) / 5) * &HFF
                        Station(Pdata(1).Astation(Cns + 1)).Slot1(Pdata(1).AC
                        hannel(Cns + 1)) = CInt(tempo)
                    End If
                Case Is < Pdata(1).LAlarm(Cns + 1)
                    If Trim(Inter.StationNumber(Pdata(1).Astation(Cns + 1)))
                    = "Digital Output" Then
                        Select Case Pdata(1).Avalue(Cns + 1)
                            Case Is >= 1
                                Station(Pdata(1).Astation(Cns + 1)).Slot2(1) = Sta
                                tion(Pdata(1).Astation(Cns + 1)).Slot2(1) Or 2 ^ Pdata(1).AChannel(Cns + 1)
                            Case Is < 1
                                Station(Pdata(1).Astation(Cns + 1)).Slot2(1) = Sta
                                tion(Pdata(1).Astation(Cns + 1)).Slot2(1) And (&HFF - 2 ^ Pdata(1).AChannel(Cns
                                + 1))
                        End Select
                    End If
                    If Trim(Inter.StationNumber(Pdata(1).Astation(Cns + 1)))
                    = "Analog Output" Then
                        tempo = (Pdata(1).Avalue(Cns + 1) / 5) * &HFF
                        Station(Pdata(1).Astation(Cns + 1)).Slot1(Pdata(1).ACha
                        nnel(Cns + 1)) = CInt(tempo)
                    End If
                End Select
            End If
        End If
        Drec.Dvalue(Cns + 1) = InputData
    End Select
Next Cns
Drec.Ddate = Date$
Drec.Dtime = Time$
F = FreeFile
RecLen = Len(Drec)
Open "Page1.rec" For Random As F Len = RecLen
LastRecord = FileLen("Page1.rec") / RecLen
Put #F, LastRecord + 1, Drec
Close #F
If PageSelect = 1 Then
    Timeserv
End If
End Sub

```



```

) = "Analog Output" Then
    tempo = (Pdata(10).Avalue(Cns + 1) / 5) * &HFF
    Station(Pdata(10).Astation(Cns + 1)).Slot1(Pdata(10).AC
hannel(Cns + 1)) = CInt(tempo)
    End If
    Case Is < Pdata(10).HAlarm(Cns + 1) - Pdata(10).Dif(Cns + 1)
    If Trim(Inter.StationNumber(Pdata(10).Astation(Cns + 1))
) = "Digital Output" Then
    Select Case Pdata(10).Avalue(Cns + 1)
    Case Is >= 1
        Station(Pdata(10).Astation(Cns + 1)).Slot2(1) = St
ation(Pdata(10).Astation(Cns + 1)).Slot2(1) And 2 ^ Pdata(10).Avalue(Cns + 1)
    Case Is < 1
        Station(Pdata(10).Astation(Cns + 1)).Slot2(1) = St
ation(Pdata(10).Astation(Cns + 1)).Slot2(1) Or (&HFF - 2 ^ Pdata(10).Avalue(Cns
+ 1))
    End Select
    End If
    If Trim(Inter.StationNumber(Pdata(10).Astation(Cns + 1))
) = "Analog Output" Then
    Station(Pdata(10).Astation(Cns + 1)).Slot1(Pdata(10).A
Channel(Cns + 1)) = 0
    End If
    Case Is > Pdata(10).HAlarm(Cns + 1) + Pdata(10).Dif(Cns + 1)
    If Trim(Inter.StationNumber(Pdata(10).Astation(Cns + 1))
) = "Digital Output" Then
    Select Case Pdata(10).Avalue(Cns + 1)
    Case Is >= 1
        Station(Pdata(10).Astation(Cns + 1)).Slot2(1) = St
ation(Pdata(10).Astation(Cns + 1)).Slot2(1) And 2 ^ Pdata(10).Avalue(Cns + 1)
    Case Is < 1
        Station(Pdata(10).Astation(Cns + 1)).Slot2(1) = St
ation(Pdata(10).Astation(Cns + 1)).Slot2(1) Or (&HFF - 2 ^ Pdata(10).Avalue(Cns
+ 1))
    End Select
    End If
    If Trim(Inter.StationNumber(Pdata(10).Astation(Cns + 1))
) = "Analog Output" Then
    Station(Pdata(10).Astation(Cns + 1)).Slot1(Pdata(10).A
Channel(Cns + 1)) = 0
    End If
    End Select
    End If
    End If
    Drec.Dvalue(Cns + 1) = InputData
    Case 1
    InputData = 0#
    If Pdata(10).Display(Cns + 1) > 0 Then
        temp = Station(Pdata(10).Station(Cns + 1)).Slot1(Pdata(10).Channel(Cns +
1))
        InputData = ((temp / InputSpan) * (Pdata(10).SpanData(Cns + 1) - Pdata(1
).ZeroData(Cns + 1)) + Pdata(10).ZeroData(Cns + 1))
        Gdata(10).GhData(Tcount1(10) + ((Cns) * 200)) = InputData
    End If
    Drec.Dvalue(Cns + 1) = InputData
    Case 2
    InputData = 0#
    If Pdata(10).Display(Cns + 1) > 0 Then
        temp = Station(Pdata(10).Station(Cns + 1)).Slot2(Pdata(10).Channel(Cns
+ 1)) And 2 ^ Pdata(10).Bit(Cns + 1)
        If temp Then

```

เอกสารนี้เป็นเอกสารราชการในการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

    temp = InputSpan
    End If
    InputData = ((temp / InputSpan) * (Pdata(10).SpanData(Cns + 1) - Pdata(
10).ZeroData(Cns + 1)) + Pdata(10).ZeroData(Cns + 1))
    Gdata(10).GhData(Tcount1(10) + ((Cns) * 200)) = InputData
    End If
    Drec.Dvalue(Cns + 1) = InputData
    End Select
Next Cns
Drec.Ddate = Date$
Drec.Dtime = Time$
F = FreeFile
RecLen = Len(Drec)
Open "Page10.rec" For Random As F Len = RecLen
LastRecord = FileLen("Page10.rec") / RecLen
Put #F, LastRecord + 1, Drec
Close #F
If PageSelect = 10 Then
    Timeserv
End If
End Sub

Sub Timer2_Timer ()
Dim Cns As Integer
Dim temp As Single
Dim LastRecord As Long
Dim F As Long
Dim RecLen As Long
Dim InputData As Single
Tcount1(2) = Tcount1(2) + 1
If Tcount1(2) > 200 Then
    Tcount1(2) = 1
    GRAPH.FRM - 12

    For Cns = 1 To 2000
        Gdata(2).GhData(Cns) = 0#
    Next Cns
End If
    For Cns = 0 To 9
        Select Case Pdata(2).Slot(Cns + 1)
            Case 0
                InputData = 0#
                If Pdata(2).Display(Cns + 1) > 0 Then
                    temp = Station(Pdata(2).Station(Cns + 1)).Slot0(Pdata(2).Channel(Cns + 1
))
                    InputData = ((temp / InputSpan) * (Pdata(2).SpanData(Cns + 1) - Pdata(1)
.ZeroData(Cns + 1)) + Pdata(2).ZeroData(Cns + 1))
                    Gdata(2).GhData(Tcount1(2) + ((Cns) * 200)) = InputData
                    End If
                    Drec.Dvalue(Cns + 1) = InputData
                Case 1
                    InputData = 0#
                    If Pdata(2).Display(Cns + 1) > 0 Then
                        temp = Station(Pdata(2).Station(Cns + 1)).Slot1(Pdata(2).Channel(Cns + 1
))
                        InputData = ((temp / InputSpan) * (Pdata(2).SpanData(Cns + 1) - Pdata(1)
.ZeroData(Cns + 1)) + Pdata(2).ZeroData(Cns + 1))
                        Gdata(2).GhData(Tcount1(2) + ((Cns) * 200)) = InputData
                        End If
                        Drec.Dvalue(Cns + 1) = InputData
                    Case 2
                        InputData = 0#

```

```

        temp = Station(Pdata(2).Station(Cns + 1)).Slot2(Pdata(2).Channel(Cns +
1)) And 2 ^ Pdata(2).Bit(Cns + 1)
        If temp Then
            temp = InputSpan
        End If
        InputData = ((temp / InputSpan) * (Pdata(2).SpanData(Cns + 1) - Pdata(2)
).ZeroData(Cns + 1)) + Pdata(2).ZeroData(Cns + 1)
        Gdata(2).GhData(Tcount1(2) + ((Cns) * 200)) = InputData
    End If
    Drec.Dvalue(Cns + 1) = InputData

End Select
Next Cns
Drec.Ddate = Date$
Drec.Dtime = Time$
F = FreeFile
RecLen = Len(Drec)
Open "Page2.rec" For Random As F Len = RecLen
LastRecord = FileLen("Page2.rec") / RecLen
Put #F, LastRecord + 1, Drec
Close #F
If PageSelect = 2 Then
    Timeserv
End If
End Sub

Sub Timer3_Timer ()
Dim Cns As Integer
Dim temp As Integer
Dim LastRecord As Long
Dim F As Long
Dim RecLen As Long
Dim InputData As Single
Tcount1(3) = Tcount1(3) + 1
If Tcount1(3) > 200 Then
GRAPH.FRM - 13

    Tcount1(3) = 1
    For Cns = 1 To 2000
        Gdata(3).GhData(Cns) = 0#
    Next Cns
End If
For Cns = 0 To 9
    Select Case Pdata(3).Slot(Cns + 1)
        Case 0
            InputData = 0#
            If Pdata(3).Display(Cns + 1) > 0 Then
                temp = Station(Pdata(3).Station(Cns + 1)).Slot0(Pdata(3).Channel(Cns + 1
))
                InputData = ((temp / InputSpan) * (Pdata(3).SpanData(Cns + 1) - Pdata(1)
).ZeroData(Cns + 1)) + Pdata(3).ZeroData(Cns + 1)
                Gdata(3).GhData(Tcount1(3) + ((Cns) * 200)) = InputData
            End If
            Drec.Dvalue(Cns + 1) = InputData
        Case 1
            InputData = 0#
            If Pdata(3).Display(Cns + 1) > 0 Then
                temp = Station(Pdata(3).Station(Cns + 1)).Slot1(Pdata(3).Channel(Cns + 1
))
                InputData = ((temp / InputSpan) * (Pdata(3).SpanData(Cns + 1) - Pdata(1)
).ZeroData(Cns + 1)) + Pdata(3).ZeroData(Cns + 1)
                Gdata(3).GhData(Tcount1(3) + ((Cns) * 200)) = InputData
            End If
        Case 2
            InputData = 0#
            If Pdata(3).Display(Cns + 1) > 0 Then
                temp = Station(Pdata(3).Station(Cns + 1)).Slot2(Pdata(3).Channel(Cns + 1
))
                InputData = ((temp / InputSpan) * (Pdata(3).SpanData(Cns + 1) - Pdata(1)
).ZeroData(Cns + 1)) + Pdata(3).ZeroData(Cns + 1)
                Gdata(3).GhData(Tcount1(3) + ((Cns) * 200)) = InputData
            End If
        Case 3
            InputData = 0#
            If Pdata(3).Display(Cns + 1) > 0 Then
                temp = Station(Pdata(3).Station(Cns + 1)).Slot3(Pdata(3).Channel(Cns + 1
))
                InputData = ((temp / InputSpan) * (Pdata(3).SpanData(Cns + 1) - Pdata(1)
).ZeroData(Cns + 1)) + Pdata(3).ZeroData(Cns + 1)
                Gdata(3).GhData(Tcount1(3) + ((Cns) * 200)) = InputData
            End If
        Case 4
            InputData = 0#
            If Pdata(3).Display(Cns + 1) > 0 Then
                temp = Station(Pdata(3).Station(Cns + 1)).Slot4(Pdata(3).Channel(Cns + 1
))
                InputData = ((temp / InputSpan) * (Pdata(3).SpanData(Cns + 1) - Pdata(1)
).ZeroData(Cns + 1)) + Pdata(3).ZeroData(Cns + 1)
                Gdata(3).GhData(Tcount1(3) + ((Cns) * 200)) = InputData
            End If
        Case 5
            InputData = 0#
            If Pdata(3).Display(Cns + 1) > 0 Then
                temp = Station(Pdata(3).Station(Cns + 1)).Slot5(Pdata(3).Channel(Cns + 1
))
                InputData = ((temp / InputSpan) * (Pdata(3).SpanData(Cns + 1) - Pdata(1)
).ZeroData(Cns + 1)) + Pdata(3).ZeroData(Cns + 1)
                Gdata(3).GhData(Tcount1(3) + ((Cns) * 200)) = InputData
            End If
        Case 6
            InputData = 0#
            If Pdata(3).Display(Cns + 1) > 0 Then
                temp = Station(Pdata(3).Station(Cns + 1)).Slot6(Pdata(3).Channel(Cns + 1
))
                InputData = ((temp / InputSpan) * (Pdata(3).SpanData(Cns + 1) - Pdata(1)
).ZeroData(Cns + 1)) + Pdata(3).ZeroData(Cns + 1)
                Gdata(3).GhData(Tcount1(3) + ((Cns) * 200)) = InputData
            End If
        Case 7
            InputData = 0#
            If Pdata(3).Display(Cns + 1) > 0 Then
                temp = Station(Pdata(3).Station(Cns + 1)).Slot7(Pdata(3).Channel(Cns + 1
))
                InputData = ((temp / InputSpan) * (Pdata(3).SpanData(Cns + 1) - Pdata(1)
).ZeroData(Cns + 1)) + Pdata(3).ZeroData(Cns + 1)
                Gdata(3).GhData(Tcount1(3) + ((Cns) * 200)) = InputData
            End If
        Case 8
            InputData = 0#
            If Pdata(3).Display(Cns + 1) > 0 Then
                temp = Station(Pdata(3).Station(Cns + 1)).Slot8(Pdata(3).Channel(Cns + 1
))
                InputData = ((temp / InputSpan) * (Pdata(3).SpanData(Cns + 1) - Pdata(1)
).ZeroData(Cns + 1)) + Pdata(3).ZeroData(Cns + 1)
                Gdata(3).GhData(Tcount1(3) + ((Cns) * 200)) = InputData
            End If
        Case 9
            InputData = 0#
            If Pdata(3).Display(Cns + 1) > 0 Then
                temp = Station(Pdata(3).Station(Cns + 1)).Slot9(Pdata(3).Channel(Cns + 1
))
                InputData = ((temp / InputSpan) * (Pdata(3).SpanData(Cns + 1) - Pdata(1)
).ZeroData(Cns + 1)) + Pdata(3).ZeroData(Cns + 1)
                Gdata(3).GhData(Tcount1(3) + ((Cns) * 200)) = InputData
            End If
    End Select
    Drec.Dvalue(Cns) = InputData
Next Cns
Drec.Ddate = Date$
Drec.Dtime = Time$
F = FreeFile
RecLen = Len(Drec)
Open "Page3.rec" For Random As F Len = RecLen
LastRecord = FileLen("Page3.rec") / RecLen
Put #F, LastRecord + 1, Drec
Close #F

```

เอกสารนี้เป็นเอกสารของงานวิจัยเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

Drec.Dvalue(Cns + 1) = InputData
Case 2
InputData = 0#
If Pdata(3).Display(Cns + 1) > 0 Then
temp = Station(Pdata(3).Station(Cns + 1)).Slot2(Pdata(3).Channel(Cns +
1)) And 2 ^ Pdata(3).Bit(Cns + 1)
If temp Then
temp = InputSpan
End If
InputData = ((temp / InputSpan) * (Pdata(3).SpanData(Cns + 1) - Pdata(3)
).ZeroData(Cns + 1)) + Pdata(3).ZeroData(Cns + 1)
Gdata(3).GhData(Tcount1(3) + ((Cns) * 200)) = InputData
End If
Drec.Dvalue(Cns + 1) = InputData

End Select
Next Cns
Drec.Ddate = Date$
Drec.Dtime = Time$
F = FreeFile
RecLen = Len(Drec)
Open "Page3.rec" For Random As F Len = RecLen
LastRecord = FileLen("Page3.rec") / RecLen
Put #F, LastRecord + 1, Drec
Close #F
If PageSelect = 3 Then
Timeserv
End If

End Sub

Sub Timer4_Timer ()
Dim Cns As Integer
Dim temp As Integer
Dim LastRecord As Long
Dim F As Long
Dim RecLen As Long
Dim InputData As Single
GRAPH.FRM - 14

Tcount1(4) = Tcount1(4) + 1
If Tcount1(4) > 200 Then
Tcount1(4) = 1
For Cns = 1 To 2000
Gdata(4).GhData(Cns) = 0#
Next Cns
End If
For Cns = 0 To 9
Select Case Pdata(4).Slot(Cns + 1)
Case 0
InputData = 0#
If Pdata(4).Display(Cns + 1) > 0 Then
temp = Station(Pdata(4).Station(Cns + 1)).Slot0(Pdata(4).Channel(Cns + 1
))
InputData = ((temp / InputSpan) * (Pdata(4).SpanData(Cns + 1) - Pdata(1)
).ZeroData(Cns + 1)) + Pdata(4).ZeroData(Cns + 1)
Gdata(4).GhData(Tcount1(4) + ((Cns) * 200)) = InputData
End If
Drec.Dvalue(Cns + 1) = InputData
Case 1
InputData = 0#
If Pdata(4).Display(Cns + 1) > 0 Then
temp = Station(Pdata(4).Station(Cns + 1)).Slot1(Pdata(4).Channel(Cns + 1

```

```

))
    InputData = ((temp / InputSpan) * (Pdata(4).SpanData(Cns + 1) - Pdata(1)
.ZeroData(Cns + 1)) + Pdata(4).ZeroData(Cns + 1))
    Gdata(4).GhData(Tcount1(4) + ((Cns) * 200)) = InputData
    End If
    Drec.Dvalue(Cns + 1) = InputData
    Case 2
    InputData = 0#
    If Pdata(4).Display(Cns + 1) > 0 Then
        temp = Station(Pdata(4).Station(Cns + 1)).Slot2(Pdata(4).Channel(Cns +
1)) And 2 ^ Pdata(4).Bit(Cns + 1)
        If temp Then
            temp = InputSpan
        End If
        InputData = ((temp / InputSpan) * (Pdata(4).SpanData(Cns + 1) - Pdata(4)
).ZeroData(Cns + 1)) + Pdata(4).ZeroData(Cns + 1))
        Gdata(4).GhData(Tcount1(4) + ((Cns) * 200)) = InputData
    End If
    Drec.Dvalue(Cns + 1) = InputData

    End Select
Next Cns
Drec.Ddate = Date$
Drec.Dtime = Time$
F = FreeFile
Reclen = Len(Drec)
Open "Page4.rec" For Random As F Len = Reclen
LastRecord = FileLen("Page4.rec") / Reclen
Put #F, LastRecord + 1, Drec
Close #F
If PageSelect = 4 Then
    Timeserv
End If

End Sub

Sub Timer5_Timer ()
Dim Cns As Integer
Dim temp As Integer
Dim LastRecord As Long
Dim F As Long
GRAPH.FRM - 15

Dim Reclen As Long
Dim InputData As Single
Tcount1(5) = Tcount1(5) + 1
If Tcount1(5) > 200 Then
    Tcount1(5) = 1
    For Cns = 1 To 2000
        Gdata(5).GhData(Cns) = 0#
    Next Cns
End If
For Cns = 0 To 9
    Select Case Pdata(5).Slot(Cns + 1)
    Case 0
        InputData = 0#
        If Pdata(5).Display(Cns + 1) > 0 Then
            temp = Station(Pdata(5).Station(Cns + 1)).Slot0(Pdata(5).Channel(Cns + 1
))
            InputData = ((temp / InputSpan) * (Pdata(5).SpanData(Cns + 1) - Pdata(1)
).ZeroData(Cns + 1)) + Pdata(5).ZeroData(Cns + 1))
            Gdata(5).GhData(Tcount1(5) + ((Cns) * 200)) = InputData
        End If

```

เอกสารนี้เป็นเอกสารลิขสิทธิ์ของมหาวิทยาลัยเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง ไม่อนุญาติให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

Drec.Dvalue(Cns + 1) = InputData
Case 1
InputData = 0#
If Pdata(5).Display(Cns + 1) > 0 Then
temp = Station(Pdata(5).Station(Cns + 1)).Slot1(Pdata(5).Channel(Cns + 1
*)
InputData = ((temp / InputSpan) * (Pdata(5).SpanData(Cns + 1) - Pdata(1)
.ZeroData(Cns + 1)) + Pdata(5).ZeroData(Cns + 1))
Gdata(5).GhData(Tcount1(5) + ((Cns) * 200)) = InputData
End If
Drec.Dvalue(Cns + 1) = InputData
Case 2
InputData = 0#
If Pdata(5).Display(Cns + 1) > 0 Then
temp = Station(Pdata(5).Station(Cns + 1)).Slot2(Pdata(5).Channel(Cns +
1)) And 2 ^ Pdata(5).Bit(Cns + 1)
If temp Then
temp = InputSpan
End If
InputData = ((temp / InputSpan) * (Pdata(5).SpanData(Cns + 1) - Pdata(5)
).ZeroData(Cns + 1)) + Pdata(5).ZeroData(Cns + 1))
Gdata(5).GhData(Tcount1(5) + ((Cns) * 200)) = InputData
End If
Drec.Dvalue(Cns + 1) = InputData

End Select
Next Cns
Drec.Ddate = Date$
Drec.Dtime = Time$
F = FreeFile
RecLen = Len(Drec)
Open "Page5.rec" For Random As F Len = RecLen
LastRecord = FileLen("Page5.rec") / RecLen
Put #F, LastRecord + 1, Drec
Close #F
If PageSelect = 5 Then
Timeserv
End If

End Sub

Sub Timer6_Timer ()
Dim Cns As Integer
Dim temp As Integer
GRAPH.FRM - 16

Dim LastRecord As Long
Dim F As Long
Dim RecLen As Long
Dim InputData As Single
Tcount1(6) = Tcount1(6) + 1
If Tcount1(6) > 200 Then
Tcount1(6) = 1
For Cns = 1 To 2000
Gdata(6).GhData(Cns) = 0#
Next Cns
End If
For Cns = 0 To 9
Select Case Pdata(6).Slot(Cns + 1)
Case 0
InputData = 0#
If Pdata(6).Display(Cns + 1) > 0 Then
temp = Station(Pdata(6).Station(Cns + 1)).Slot0(Pdata(6).Channel(Cns + 1

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

))
    InputData = ((temp / InputSpan) * (Pdata(6).SpanData(Cns + 1) - Pdata(1)
.ZeroData(Cns + 1)) + Pdata(6).ZeroData(Cns + 1))
    Gdata(6).GhData(Tcount1(6) + ((Cns) * 200)) = InputData
    End If
    Drec.Dvalue(Cns + 1) = InputData
    Case 1
    InputData = 0#
    If Pdata(6).Display(Cns + 1) > 0 Then
        temp = Station(Pdata(6).Station(Cns + 1)).Slot1(Pdata(6).Channel(Cns + 1)
))
        InputData = ((temp / InputSpan) * (Pdata(6).SpanData(Cns + 1) - Pdata(1)
.ZeroData(Cns + 1)) + Pdata(6).ZeroData(Cns + 1))
        Gdata(6).GhData(Tcount1(6) + ((Cns) * 200)) = InputData
        End If
        Drec.Dvalue(Cns + 1) = InputData
        Case 2
        InputData = 0#
        If Pdata(6).Display(Cns + 1) > 0 Then
            temp = Station(Pdata(6).Station(Cns + 1)).Slot2(Pdata(6).Channel(Cns +
1)) And 2 ^ Pdata(6).Bit(Cns + 1)
            If temp Then
                temp = InputSpan
            End If
            InputData = ((temp / InputSpan) * (Pdata(6).SpanData(Cns + 1) - Pdata(6)
).ZeroData(Cns + 1)) + Pdata(6).ZeroData(Cns + 1))
            Gdata(6).GhData(Tcount1(6) + ((Cns) * 200)) = InputData
            End If
            Drec.Dvalue(Cns + 1) = InputData

        End Select
    Next Cns
    Drec.Ddate = Date$
    Drec.Dtime = Time$
    F = FreeFile
    RecLen = Len(Drec)
    Open "Page6.rec" For Random As F Len = RecLen
    LastRecord = FileLen("Page6.rec") / RecLen
    Put #F, LastRecord + 1, Drec
    Close #F
    If PageSelect = 6 Then
        Timeserv
    End If

End Sub

```

```

Sub Timer7_Timer ()
GRAPH.FRM - 17

```

```

Dim Cns As Integer
Dim temp As Integer
Dim LastRecord As Long
Dim F As Long
Dim RecLen As Long
Dim InputData As Single
Tcount1(7) = Tcount1(7) + 1
If Tcount1(7) > 200 Then
    Tcount1(7) = 1
    For Cns = 1 To 2000
        Gdata(7).GhData(Cns) = 0#
    Next Cns
End If

```

เอกสารนี้เป็นลิขสิทธิ์ของสำนักงานส่งเสริมการค้าในต่างประเทศ ณ นครเชียงใหม่ ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

Select Case Pdata(7).Slot(Cns + 1)
Case 0
  InputData = 0#
  If Pdata(7).Display(Cns + 1) > 0 Then
    temp = Station(Pdata(7).Station(Cns + 1)).Slot0(Pdata(7).Channel(Cns + 1))
  ))
  InputData = ((temp / InputSpan) * (Pdata(7).SpanData(Cns + 1) - Pdata(1)
.ZeroData(Cns + 1)) + Pdata(7).ZeroData(Cns + 1))
  Gdata(7).GhData(Tcount1(7) + ((Cns) * 200)) = InputData
  End If
  Drec.Dvalue(Cns + 1) = InputData
Case 1
  InputData = 0#
  If Pdata(7).Display(Cns + 1) > 0 Then
    temp = Station(Pdata(7).Station(Cns + 1)).Slot1(Pdata(7).Channel(Cns + 1))
  ))
  InputData = ((temp / InputSpan) * (Pdata(7).SpanData(Cns + 1) - Pdata(1)
.ZeroData(Cns + 1)) + Pdata(7).ZeroData(Cns + 1))
  Gdata(7).GhData(Tcount1(7) + ((Cns) * 200)) = InputData
  End If
  Drec.Dvalue(Cns + 1) = InputData
Case 2
  InputData = 0#
  If Pdata(7).Display(Cns + 1) > 0 Then
    temp = Station(Pdata(7).Station(Cns + 1)).Slot2(Pdata(7).Channel(Cns +
1)) And 2 ^ Pdata(7).Bit(Cns + 1)
    If temp Then
      temp = InputSpan
    End If
    InputData = ((temp / InputSpan) * (Pdata(7).SpanData(Cns + 1) - Pdata(7)
).ZeroData(Cns + 1)) + Pdata(7).ZeroData(Cns + 1))
    Gdata(7).GhData(Tcount1(7) + ((Cns) * 200)) = InputData
    End If
    Drec.Dvalue(Cns + 1) = InputData

  End Select
Next Cns
Drec.Ddate = Date$
Drec.Dtime = Time$
F = FreeFile
RecLen = Len(Drec)
Open "Page7.rec" For Random As F Len = RecLen
LastRecord = FileLen("Page7.rec") / RecLen
Put #F, LastRecord + 1, Drec
Close #F
If PageSelect = 7 Then
  Timeserv
End If

End Sub
GRAPH.FRM - 18

```

```

Sub Timer8_Timer ()
Dim Cns As Integer
Dim temp As Integer
Dim LastRecord As Long
Dim F As Long
Dim RecLen As Long
Dim InputData As Single
Tcount1(8) = Tcount1(8) + 1

```

เอกสารนี้เป็นเอกสารสงวนลิขสิทธิ์ของมหาวิทยาลัยเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่าในรูปแบบใดก็ตาม หากมีข้อผิดพลาดประการใดขออภัยเป็นอย่างสูงและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้


```

Dim InputData As Single
Tcount1(9) = Tcount1(9) + 1
If Tcount1(9) > 200 Then
    Tcount1(9) = 1
    For Cns = 1 To 2000
        Gdata(9).GhData(Cns) = 0#
    Next Cns
End If
For Cns = 0 To 9
    Select Case Pdata(9).Slot(Cns + 1)
        Case 0
            InputData = 0#
            If Pdata(9).Display(Cns + 1) > 0 Then
                temp = Station(Pdata(9).Station(Cns + 1)).Slot0(Pdata(9).Channel(Cns + 1))
            )
                InputData = ((temp / InputSpan) * (Pdata(9).SpanData(Cns + 1) - Pdata(1)
                .ZeroData(Cns + 1)) + Pdata(9).ZeroData(Cns + 1))
                Gdata(9).GhData(Tcount1(9) + ((Cns) * 200)) = InputData
            End If
            Drec.Dvalue(Cns + 1) = InputData
        Case 1
            InputData = 0#
            If Pdata(9).Display(Cns + 1) > 0 Then
                temp = Station(Pdata(9).Station(Cns + 1)).Slot1(Pdata(9).Channel(Cns + 1))
            )
                InputData = ((temp / InputSpan) * (Pdata(9).SpanData(Cns + 1) - Pdata(1)
                .ZeroData(Cns + 1)) + Pdata(9).ZeroData(Cns + 1))
                Gdata(9).GhData(Tcount1(9) + ((Cns) * 200)) = InputData
            End If
            Drec.Dvalue(Cns + 1) = InputData
        Case 2
            InputData = 0#
            If Pdata(9).Display(Cns + 1) > 0 Then
                temp = Station(Pdata(9).Station(Cns + 1)).Slot2(Pdata(9).Channel(Cns + 1))
            1) And 2 ^ Pdata(9).Bit(Cns + 1)
                If temp Then
                    temp = InputSpan
                End If
                InputData = ((temp / InputSpan) * (Pdata(9).SpanData(Cns + 1) - Pdata(9)
                ).ZeroData(Cns + 1)) + Pdata(9).ZeroData(Cns + 1))
                Gdata(9).GhData(Tcount1(9) + ((Cns) * 200)) = InputData
            End If
            Drec.Dvalue(Cns + 1) = InputData

    End Select
Next Cns
Drec.Ddate = Date$
Drec.Dtime = Time$
F = FreeFile
RecLen = Len(Drec)
Open "Page9.rec" For Random As F Len = RecLen
LastRecord = FileLen("Page9.rec") / RecLen
Put #F, LastRecord + 1, Drec
Close #F
If PageSelect = 9 Then
    GRAPH.FRM - 20

    Timeserv
End If

End Sub

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

LINESET.FRM - 1

```
Option Explicit
Dim Linesel As LineInfo
Dim Interface As InterfaceInfo
Dim FileNum As Long
Dim FileNum1 As Long
Dim FileNum2 As Long
Dim RecordCount As Long
Dim Lineselect As Integer
Dim Alarm As Alarmset
```

```
Sub Command3D1_Click ()
```

```
End Sub
```

```
Sub cboLineselect_Click ()
```

```
Lineselect = cboLineselect.ListIndex + 1
'cmdExit.Caption = Str$(Lineselect)
Get #FileNum, Lineselect, Linesel
lbName.Text = Linesel.Lastname
tbStation.Text = Linesel.StationNumber
tbSlot.Text = Linesel.Slot
tbUnits.Text = Linesel.Units
tbChannel.Text = Linesel.Channel
tbBit.Text = Linesel.Bit
chkEnable = Linesel.AlarmEnable
chkActive = Linesel.Active
lbZero.Text = Format$(Linesel.Zero, "0.000000")
lbSpan.Text = Format$(Linesel.Span, "0.000000")
lbHigh = Format$(Linesel.HighAlarm, "0.000000")
lbLow.Text = Format$(Linesel.LowAlarm, "0.000000")
lbDiff.Text = Format$(Linesel.Diff, "0.000000")
cboLineselect.Text = Trim(Linesel.Firstname) + " " + Trim(Linesel.Lastname)

tbAstation.Text = Format$(Linesel.Astation, "00")
tbAchannel.Text = Format$(Linesel.Achannel, "00")
tbAvalue.Text = Format$(Linesel.Avalue, "0.00")
```

```
End Sub
```

```
Sub chkEnable_Click ()
```

```
Select Case chkEnable.Value
Case 0
frame2.Visible = False
Case 1
frame2.Visible = True
```

```
End Select
```

```
End Sub
```

```
Sub cmdAccept_Click ()
```

```
Get #FileNum, Lineselect, Linesel
Linesel.Lastname = lbName.Text
Linesel.StationNumber = tbStation.Text
Linesel.Units = tbUnits.Text
Linesel.Slot = tbSlot.Text
Linesel.Channel = tbChannel.Text
Linesel.Bit = tbBit.Text
Linesel.AlarmEnable = chkEnable
Linesel.Active = chkActive
Linesel.Zero = lbZero.Text
Linesel.Span = lbSpan.Text
```

```

Sub Form_Load ()
Dim Count1 As Integer
    FileName = FreeFile
INTER.FRM - 2

    RecordCount = Len(Interface)
    Open "Inter.set" For Random As FileName Len = RecordCount
    Get #FileName, 1, Interface
    For Count1 = 0 To 31 Step 1
        IsbStation.AddItem Format(Count1, "00") + Interface.StationNumber(Count1)
    Next Count1
    cboPort.AddItem "Com 1"
    cboPort.AddItem "Com 2"
    cboPort.AddItem "Com 3"
    cboPort.AddItem "Com 4"
    cboBaud.AddItem "9600"
    cboBaud.AddItem "4800"
    cboBaud.AddItem "2400"
    cboBaud.AddItem "1200"
    cboParity.AddItem "N"
    cboParity.AddItem "O"
    cboParity.AddItem "E"
    cboStop.AddItem "1"
    cboStop.AddItem "1.5"
    cboStop.AddItem "2"
    cboData.AddItem "7"
    cboData.AddItem "8"
    cboPort.Text = Interface.Port
    cboBaud.Text = Interface.Baud
    cboParity.Text = Interface.Parity
    cboData.Text = Interface.Data
    cboStop.Text = Interface.Stop
End Sub

Sub IsbStation_Click ()
    Station = IsbStation.ListIndex
End Sub

```

INTER.FRM - 1

```
Option Explicit
Dim FileNum As Long
Dim Interface As InterfaceInfo
Dim RecordCount As Long
Dim Station As Integer
Dim InputType As Integer
Dim Itype(0 To 4) As String * 25
```

```
Sub chkAI_Click (Index As Integer)
    InputType = 0
End Sub

Sub chkAO_Click (Index As Integer)
    InputType = 1
End Sub

Sub chkDI_Click (Index As Integer)
    InputType = 2
End Sub

Sub chkDO_Click (Index As Integer)
    InputType = 3
End Sub

Sub chkNone_Click (Index As Integer)
    InputType = 4
End Sub

Sub cmdAccept_Click ()
    Dim Output1 As String
    Itype(0) = " Analog Input "
    Itype(1) = " Analog Output "
    Itype(2) = " Digital Input "
    Itype(3) = " Digital Output "
    Itype(4) = " None "
    Output1 = Format$(Station, "00")
    lsbStation.RemoveItem Station
    lsbStation.AddItem Output1 + Itype(InputType)
    Interface.StationNumber(Station) = Itype(InputType)
End Sub

Sub cmdCancel_Click ()
    Close
    Unload Me
End Sub

Sub cmdSave_Click ()
    Interface.Port = cboPort.Text
    Interface.Baud = cboBaud.Text
    Interface.Parity = cboParity.Text
    Interface.Data = cboData.Text
    Interface.Stop = cboStop.Text
    Put #FileNum, 1, Interface
End Sub
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

Close
Unload Me
End Sub

```

```

Sub cmdGetaddress_Click ()
    Getchange = 0
    frmGetaddress.Show 1
    If Getchange = 1 Then
        tbStation.Text = AddressSelect
        Select Case StationSelect
            Case 0
                tbSlot.Text = Str$(0)
                tbChannel.Text = Str$(ParaSelect)
                tbBit.Text = 0
            Case 1
                tbSlot.Text = Str$(1)
                tbChannel.Text = Str$(ParaSelect)
                tbBit.Text = 0
            Case 2
                tbSlot.Text = Str$(2)
                tbChannel.Text = Str$(0)
                tbBit.Text = Str$(ParaSelect)
            Case 3
                tbSlot.Text = Str$(2)
                tbChannel.Text = Str$(1)
                tbBit.Text = Str$(ParaSelect)
        End Select
    End If
End Sub

```

```

Sub Form_Load ()
    Dim Page As Integer
    Dim Line1 As Integer
    Dim Count2 As Integer
    FileNum1 = FreeFile
    RecordCount = Len(Alarm)
    Open "Alarm.set" For Random As FileNum1 Len = RecordCount
    FileNum2 = FreeFile
    RecordCount = Len(Interface)
    Open "Inter.set" For Random As FileNum2 Len = RecordCount
    FileNum = FreeFile
    RecordCount = Len(Linesel)
    Open "Lineset.set" For Random Lock Read Write As FileNum Len = RecordCount
    Lineselect = 1
    Count2 = 1
    For Page = 1 To 10 Step 1
        For Line1 = 1 To 10 Step 1
            Get #FileNum, Count2, Linesel
            Linesel.FirstName = "Page " + Format$(Page, "00") + " Line " + Format$(Line1, "00")
            Put #FileNum, Count2, Linesel
            Count2 = Count2 + 1
        Next Line1
    Next Page
    Next Page
    For Page = 1 To 100 Step 1
        Get #FileNum, Page, Linesel
        cboLineselect.AddItem Trim(Linesel.FirstName) + " " + Trim(Linesel.LastName)
    Next Page

```

```

Get #FileNum, 1, Linesel
lbName.Text = Linesel.Lastname
tbStation.Text = Linesel.StationNumber
tbSlot.Text = Linesel.Slot
tbChannel.Text = Linesel.Channel
tbUnits.Text = Linesel.Units
tbBit.Text = Linesel.Bit
chkEnable = Linesel.AlarmEnable
chkActive = Linesel.Active
lbZero.Text = Format$(Linesel.Zero, "0.000000")
lbSpan.Text = Format$(Linesel.Span, "0.000000")
lbHigh = Format$(Linesel.HighAlarm, "0.000000")
lbLow.Text = Format$(Linesel.LowAlarm, "0.000000")
lbDiff.Text = Format$(Linesel.Diff, "0.000000")
cboLineselct.Text = Trim(Linesel.FirstName) + Trim(Linesel.Lastname)

tbAstation.Text = Format$(Linesel.Astation, "00")
tbAchannel.Text = Format$(Linesel.Achannel, "00")
tbAvalue.Text = Format$(Linesel.Avalue, "0.00")
frame2.Visible = False
If chkEnable.Value Then
    frame2.Visible = True
End If
End Sub

```

PAGESET.FRM - 1

```
Option Explicit
Dim Page As PageInfo
Dim FileNum As Long
Dim RecordCount As Long
Dim PageSelect As Integer
```

```
Sub chk1_Click (Index As Integer)
    PageSelect = 1
    Get #FileNum, 1, Page
    lbName = Page.Name
    lbScan = Page.Scan
    chkEnable.Value = Page.Enable
End Sub
```

```
Sub chk10_Click (Index As Integer)
    PageSelect = 10
    Get #FileNum, 10, Page
    lbName = Page.Name
    lbScan = Page.Scan
    chkEnable.Value = Page.Enable
End Sub
```

```
Sub chk2_Click (Index As Integer)
    PageSelect = 2
    Get #FileNum, 2, Page
    lbName = Page.Name
    lbScan = Page.Scan
    chkEnable.Value = Page.Enable
End Sub
```

```
Sub chk3_Click (Index As Integer)
    PageSelect = 3
    Get #FileNum, 3, Page
    lbName = Page.Name
    lbScan = Page.Scan
    chkEnable.Value = Page.Enable
End Sub
```

```
Sub chk4_Click (Index As Integer)
    PageSelect = 4
    Get #FileNum, 4, Page
    lbName = Page.Name
    lbScan = Page.Scan
    chkEnable.Value = Page.Enable
End Sub
```

```
Sub chk5_Click (Index As Integer)
    PageSelect = 5
    Get #FileNum, 5, Page
    lbName = Page.Name
    lbScan = Page.Scan
    chkEnable.Value = Page.Enable
End Sub
```

```
Sub chk6_Click (Index As Integer)
    PageSelect = 6
    Get #FileNum, 6, Page
    lbName = Page.Name
    lbScan = Page.Scan
    chkEnable.Value = Page.Enable
```

End Sub

Sub chk7_Click (Index As Integer)
PAGESET.FRM - 2

```

    PageSelect = 7
    Get #FileNum, 7, Page
    lbName = Page.Name
    lbScan = Page.Scan
    chkEnable.Value = Page.Enable

```

End Sub

Sub chk8_Click (Index As Integer)

```

    PageSelect = 8
    Get #FileNum, 8, Page
    lbName = Page.Name
    lbScan = Page.Scan
    chkEnable.Value = Page.Enable

```

End Sub

Sub chk9_Click (Index As Integer)

```

    PageSelect = 9
    Get #FileNum, 9, Page
    lbName = Page.Name
    lbScan = Page.Scan
    chkEnable.Value = Page.Enable

```

End Sub

Sub cmdAccept_Click ()

```

    Page.Name = lbName.Text
    Page.Scan = lbScan.Text
    Page.Enable = chkEnable.Value
    Put #FileNum, PageSelect, Page

```

End Sub

Sub cmdExit_Click ()

```

    Close
    Unload Me

```

End Sub

Sub Form_Load ()

```

    FileNum = FreeFile
    RecordCount = Len(Page)
    Open "Pageset.set" For Random As FileNum Len = RecordCount
    Get #FileNum, 1, Page
    lbName = Page.Name
    lbScan = Page.Scan
    chkEnable.Value = Page.Enable
    PageSelect = 1

```

End Sub

Sub Label1_Click ()

```

    PageSelect = 1

```

End Sub

SETUP.FRM - 1

```
Sub Command3D1_Click ()  
    frmInterface.Show 1  
End Sub
```

```
Sub Command3D2_Click ()  
    frmPageset.Show 1  
End Sub
```

```
Sub Command3D3_Click ()  
    frmLineset.Show 1  
End Sub
```

```
Sub Command3D4_Click ()  
    Unload Me  
End Sub
```



SIM.FRM - 1

```
Option Explicit
Dim Stsel As Integer
Dim Stype As Integer
```

```
Sub Command3D1_Click ()
    Unload Me
End Sub
```

```
Sub cmdAccept_Click ()
Dim Co As Integer
Dim A As Integer
Dim B As Integer
frmView.Show
Select Case Stype
    Case 0
        For Co = 0 To 7 Step 1
            Station(Stsel).Slot0(Co) = Text1(Co).Text
        Next Co
    Case 1
        For Co = 0 To 7 Step 1
            Station(Stsel).Slot1(Co) = Text1(Co).Text
        Next Co
    Case 2
        A = 0
        B = 0
        For Co = 0 To 7 Step 1
            A = Text1(Co).Text
            B = B + ((2 ^ Co) * A)
        Next Co
        Station(Stsel).Slot2(0) = B
    Case 3
        A = 0
        B = 0
        For Co = 0 To 7 Step 1
            A = Text1(Co).Text
            B = B + ((2 ^ Co) * A)
        Next Co
        Station(Stsel).Slot2(1) = B
End Select
End Sub
```

```
Sub cmdExit_Click ()
    Unload frmView
    Unload Me
End Sub
```

```
Sub Form_Load ()
Dim St As Integer
    For St = 0 To 31 Step 1
        List1.AddItem Format$(St, "00")

    Next St
    List2.AddItem "Analog Input"
    List2.AddItem "Analog Output"
    List2.AddItem "Digital Input"
    List2.AddItem "Digital Output"
    List2.AddItem "None"
    For St = 0 To 7 Step 1
        Text1(St).Text = Format$(0, "00")
    Next St
    frmView.Show
```

End Sub

Sub List1_Click ()
SIM.FRM - 2

 Stsel = List1.ListIndex
End Sub

Sub List2_Click ()
 Stype = List2.ListIndex
End Sub



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

SYSTEM.FRM - 1

```
Sub cmdExit_Click ()
    End
End Sub
```

```
Sub cmdGo_Click ()
    frmGraph.Show
End Sub
```

```
Sub cmdHistory_Click ()
    frmHistory.Show
End Sub
```

```
Sub cmdSetup_Click ()
    frmSetup.Show 1
End Sub
```

```
Sub Form_Load ()
    Dim c As Long
    frmstart.Show
    For c = 0 To 1000000
        Next
    Unload frmstart
    frmssystem.MousePointer = 0
End Sub
```



VIEW.FRM - 1

```
Option Explicit
Dim Pselect As Integer
Dim Nselect As Integer
Dim Interface As InterfaceInfo
Dim RecLen As Long
Dim FileNum As Long
Dim Buff(1 To 60) As String
Dim Inputdata As String
Dim Counter As Integer
```

```
Sub Display ()
Dim Ch As Integer
Select Case Station(Nselect).StationInfo
Case 0
GrdTable.Col = 1
For Ch = 0 To 7 Step 1
GrdTable.Row = Ch + 2
GrdTable.Text = Format$(Hex$(Station(Nselect).Slot0(Ch)), "00")
Next Ch
GrdTable.Col = 2
For Ch = 0 To 7 Step 1
GrdTable.Row = Ch + 2
GrdTable.Text = Format$(0, "00")
Next Ch
GrdTable.Col = 3
For Ch = 0 To 1 Step 1
GrdTable.Row = Ch + 2
GrdTable.Text = Format$(0, "00")
Next Ch
Case 1
GrdTable.Col = 1
For Ch = 0 To 7 Step 1
GrdTable.Row = Ch + 2
GrdTable.Text = Format$(0, "00")
Next Ch
GrdTable.Col = 2
For Ch = 0 To 7 Step 1
GrdTable.Row = Ch + 2
GrdTable.Text = Format$(Hex$(Station(Nselect).Slot1(Ch)), "00")
Next Ch
GrdTable.Col = 3
For Ch = 0 To 1 Step 1
GrdTable.Row = Ch + 2
GrdTable.Text = Format$(0, "00")
Next Ch
Case 2
GrdTable.Col = 1
For Ch = 0 To 7 Step 1
GrdTable.Row = Ch + 2
GrdTable.Text = Format$(0, "00")
Next Ch
GrdTable.Col = 2
For Ch = 0 To 7 Step 1
GrdTable.Row = Ch + 2
GrdTable.Text = Format$(0, "00")
Next Ch
GrdTable.Col = 3
GrdTable.Row = 3
GrdTable.Text = "00"
GrdTable.Col = 3
```

```

        GrdTable.Row = 2
        GrdTable.Text = Format$(Hex$(Station(Nselect).Slot2(0)), "00")
    Case 3
VIEW.FRM - 2

        GrdTable.Col = 1
        For Ch = 0 To 7 Step 1
            GrdTable.Row = Ch + 2
            GrdTable.Text = Format$(0, "00")
        Next Ch
        GrdTable.Col = 2
        For Ch = 0 To 7 Step 1
            GrdTable.Row = Ch + 2
            GrdTable.Text = Format$(0, "00")
        Next Ch
        GrdTable.Col = 3
        GrdTable.Row = 2
        GrdTable.Text = "00"
        GrdTable.Col = 3
        GrdTable.Row = 3
        GrdTable.Text = Format$(Hex$(Station(Nselect).Slot2(1)), "00")
    End Select
End Sub

```

```
Function HtoDec (Fir$, Sec$, Thi$)
```

```
Dim a%, b%, c%, Counter%
```

```
Select Case Thi
```

```
Case Is = "1"
```

```
a = 1
```

```
Case Is = "2"
```

```
a = 2
```

```
Case Is = "3"
```

```
a = 3
```

```
Case Is = "4"
```

```
a = 4
```

```
Case Is = "5"
```

```
a = 5
```

```
Case Is = "6"
```

```
a = 6
```

```
Case Is = "7"
```

```
a = 7
```

```
Case Is = "8"
```

```
a = 8
```

```
Case Is = "9"
```

```
a = 9
```

```
Case Is = "A"
```

```
a = &HA
```

```
Case Is = "B"
```

```
a = &HB
```

```
Case Is = "C"
```

```
a = &HC
```

```
Case Is = "D"
```

```
a = &HD
```

```
Case Is = "E"
```

```
a = &HE
```

```
Case Is = "F"
```

```
a = &HF
```

```
End Select
```

```
Select Case Sec
```

```
Case Is = "1"
```

```
b = 1 * 16
```

```
Case Is = "2"
```

```

    b = 2 * 16
Case Is = "3"
    b = 3 * 16
Case Is = "4"
    b = 4 * 16
Case Is = "5"
VIEW.FRM - 3

```

```

    b = 5 * 16
Case Is = "6"
    b = 6 * 16
Case Is = "7"
    b = 7 * 16
Case Is = "8"
    b = 8 * 16
Case Is = "9"
    b = 9 * 16
Case Is = "A"
    b = &HA * 16
Case Is = "B"
    b = &HB * 16
Case Is = "C"
    b = &HC * 16
Case Is = "D"
    b = &HD * 16
Case Is = "E"
    b = &HE * 16
Case Is = "F"
    b = &HF * 16
End Select
Select Case Fir
Case Is = "1"
    c = 1 * 16 * 16
Case Is = "2"
    c = 2 * 16 * 16
Case Is = "3"
    c = 3 * 16 * 16
Case Is = "4"
    c = 4 * 16 * 16
Case Is = "5"
    c = 5 * 16 * 16
Case Is = "6"
    c = 6 * 16 * 16
Case Is = "7"
    c = 7 * 16 * 16
Case Is = "8"
    c = 8 * 16 * 16
Case Is = "9"
    c = 9 * 16 * 16
Case Is = "A"
    c = &HA * 16 * 16
Case Is = "B"
    c = &HB * 16 * 16
Case Is = "C"
    c = &HC * 16 * 16
Case Is = "D"
    c = &HD * 16 * 16
Case Is = "E"
    c = &HE * 16 * 16
Case Is = "F"
    c = &HF * 16 * 16
End Select

```

เอกสารนี้ HtoDec = a + b + c ทรัพยากรใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
End Function
```

```
Sub cmdExit_Click ()
    Unload Me
End Sub
```

```
Sub cmdNext_Click ()
    Dim Ch As Integer
    Nselect = Nselect + 1
VIEW.FRM - 4
```

```
    If Nselect > 31 Then
        Nselect = 31
    End If
    cmdPrev.Enabled = True
    GrdTable.Row = 1
    GrdTable.Col = 0
    GrdTable.Text = "Station (" + Format(Nselect, "00") + ")"
    Display
```

```
End Sub
```

```
Sub cmdPrev_Click ()
    Dim Ch As Integer
    Nselect = Nselect - 1
    If Nselect < 0 Then
        cmdPrev.Enabled = False
        Nselect = 0
    End If
    GrdTable.Row = 1
    GrdTable.Col = 0
    GrdTable.Text = "Station (" + Format(Nselect, "00") + ")"
    If cmdPrev.Enabled = True Then
        Display
    End If
End Sub
```

```
Sub Form_Load ()
    Dim RowCount As Integer
    Dim Ch As Integer
    Dim Dataout As String
    Nselect = 0
    For RowCount = 0 To 7 Step 1
        GrdTable.Col = 0
        GrdTable.Row = RowCount + 2
        GrdTable.Text = "Channel " + Str$(RowCount)
        GrdTable.RowHeight(RowCount + 2) = 300
    Next RowCount
    GrdTable.RowHeight(0) = 300
    GrdTable.RowHeight(1) = 300

    GrdTable.ColWidth(0) = 1200
    GrdTable.ColWidth(1) = 2000
    GrdTable.ColWidth(2) = 2000
    GrdTable.ColWidth(3) = 2000

    For RowCount = 0 To 2 Step 1
        GrdTable.Row = 0
        GrdTable.Col = RowCount + 1
        GrdTable.Text = "Slot " + Str$(RowCount)
    Next RowCount
    cmdPrev.Enabled = False
    GrdTable.Row = 1
    GrdTable.Col = 0
```

```

GrdTable.Text = "Station (00)"
GrdTable.ColAlignment(1) = 2
GrdTable.ColAlignment(2) = 2
GrdTable.ColAlignment(3) = 2
Display
'
FileNum = FreeFile
RecLen = Len(Interface)
Open "Inter.set" For Random As FileNum Len = RecLen
Get #FileNum, 1, Interface
Select Case Interface.Port
    Case Is = "Com 1"
VIEW.FRM - 5

        Comm1.CommPort = 1
    Case Is = "Com 2"
        Comm1.CommPort = 2
    Case Is = "Com 3"
        Comm1.CommPort = 3
    Case Is = "Com 4"
        Comm1.CommPort = 4
End Select
Comm1.PortOpen = True
Comm1.Settings = Str$(Interface.Baud) + "," + Trim(Interface.Parity) + "," + S
tr(Interface.Data) + "," + Str(Interface.Stop)
Get #FileNum, 1, Interface
For Ch = 0 To 31
    Select Case Trim(Interface.StationNumber(Ch))
        Case "Analog Input"
            Station(Ch).StationInfo = 0
        Case "Analog Output"
            Station(Ch).StationInfo = 1
        Case "Digital Input"
            Station(Ch).StationInfo = 2
        Case "Digital Output"
            Station(Ch).StationInfo = 3
    End Select
Next Ch
Close #FileNum
End Sub

Sub Form_Unload (Cancel As Integer)
    Close #FileNum
End Sub

Sub Timer1_Timer ()
    Display
End Sub

Sub Timer2_Timer ()
    Dim Startup$
    Dim Dummy%
    Dim StationCount%
    Dim Pu%
    Dim RecCount%
    Timer2.Enabled = False
    For StationCount = 0 To 31
        Select Case Trim(Interface.StationNumber(StationCount))
            Case Is = "Analog Input"
                timer1.Enabled = False
                Station(StationCount).StationInfo = 0
                Inputdata = " "
                RecCount = 0

```

```

Comm1.InBufferCount = 0
TxCount = TxCount + 1
If TxCount > 6 Then
    lbTx.Clear
    lbRx.Clear
    TxCount = 1
End If
lbTx.AddItem Chr$(&H4) + Format$(StationCount, "00") + Chr$(&H5)
Comm1.Output = Chr$(&H4) + Format$(StationCount, "00") + Chr$(&H5)
Do
    Dummy = DoEvents()
    If Comm1.InBufferCount Then
        Inputdata = Inputdata + Comm1.Input
        RecCount = 0
        If InStr(Inputdata, Chr(&H6)) Then
VIEW.FRM - 6

            lbRx.AddItem Inputdata
            If Len(Trim(Inputdata)) > 3 Then
                Inputdata = Right(Inputdata, 3)
            End If
            Exit Do
        End If
    End If
    RecCount = RecCount + 1
    If RecCount > 200 Then
        lbRx.AddItem "Station Number " + Format$(StationCount, "00") + " cannot found !"
        Exit Do
    End If
Loop
If Trim(Inputdata) = Format$(StationCount, "00") + Chr(&H6) Then
    Startup = Chr$(&H2) + "EX" + " " + "AI" + " " + Chr$(&H3) + Chr$(&H0)
    Comm1.InBufferCount = 0
    Inputdata = " "
    RecCount = 0
    TxCount = TxCount + 1
    If TxCount > 6 Then
        lbRx.Clear
        lbTx.Clear
        TxCount = 1
    End If
    lbTx.AddItem Startup
    Comm1.Output = Startup
    Do
        Dummy = DoEvents()
        If Comm1.InBufferCount Then
            Inputdata = Inputdata + Comm1.Input
            RecCount = 0
            If InStr(Inputdata, Chr$(&H3)) Then
                lbRx.AddItem Inputdata
                Inputdata = Trim(Inputdata)
                Inputdata = Mid$(Inputdata, InStr(Inputdata, Chr$(&H2)), 30)
                Station(StationCount).Slot0(0) = HtoDec("0", Mid$(Inputdata, 8, 1)
, Mid$(Inputdata, 9, 1))
                Station(StationCount).Slot0(1) = HtoDec("0", Mid$(Inputdata, 11, 1)
), Mid$(Inputdata, 12, 1))
                Station(StationCount).Slot0(2) = HtoDec("0", Mid$(Inputdata, 14, 1)
), Mid$(Inputdata, 15, 1))
                Station(StationCount).Slot0(3) = HtoDec("0", Mid$(Inputdata, 17, 1)
), Mid$(Inputdata, 18, 1))
                Station(StationCount).Slot0(4) = HtoDec("0", Mid$(Inputdata, 20, 1)
), Mid$(Inputdata, 21, 1))

```

```

RecCount = RecCount + 1
If RecCount > 200 Then
lbRx.AddItem "Station Number " + Format$(StationCount, "00") + " Timeou
t error !"
Exit Do
End If
Loop
If Trim(Inputdata) = Format$(StationCount, "00") + Chr(&H6) Then
Inputdata = Format$(Hex$(Station(StationCount).Slot1(0)), "00") + " "
+ Format$(Hex$(Station(StationCount).Slot1(1)), "00") + " " + Format$(Hex$(Stati
on(StationCount).Slot1(2)), "00") + " " + Format$(Hex$(Station(StationCount).Sl
ot1(3)), "00") + " " + Format$(Hex$(Station(StationCount).Slot1(4)), "00") + " "
+ Format$(Hex$(Station(StationCount).Slot1(5)), "00") + " " + Format$(Hex$(Stat
ion(StationCount).Slot1(6)), "00") + " " + Format$(Hex$(Station(StationCount).Sl
ot1(7)), "00")
Startup = Chr(&H2) + "EX" + " " + "AO" + " " + Inputdata + Chr(&H3)
+ Chr(&H0)
Comm1.InBufferCount = 0
Inputdata = " "
RecCount = 0
TxCount = TxCount + 1
If TxCount > 6 Then
lbRx.Clear
lbTx.Clear
TxCount = 1
End If
lbTx.AddItem Startup
VIEW.FRM - 10
Comm1.Output = Chr(&H2) + "EX" + " " + "AO" + " " + Chr$(Station(Sta
tionCount).Slot1(0)) + " " + Chr$(Station(StationCount).Slot1(1)) + " " + Chr$(S
tation(StationCount).Slot1(2)) + " " + Chr$(Station(StationCount).Slot1(3)) + "
" + Chr$(Station(StationCount).Slot1(4)) + " " + Chr$(Station(StationCount).Slot
1(5)) + " " + Chr$(Station(StationCount).Slot1(6)) + " " + Chr$(Station(StationC
ount).Slot1(7)) + " " + Chr(&H3) + Chr(&H0)
Do
Dummy = DoEvents()
If Comm1.InBufferCount Then
Inputdata = Inputdata + Comm1.Input
RecCount = 0
If InStr(Inputdata, Chr(&H6)) Then
lbRx.AddItem Inputdata
Exit Do
End If
End If
RecCount = RecCount + 1
If RecCount > 400 Then
lbRx.AddItem "Station Number " + Format$(StationCount, "00") + " in
correct type !"
Exit Do
End If
Loop
End If
timer1.Enabled = True

End Select
Next StationCount
Timer2.Enabled = True
End Sub

```

```

Inputdata = Format$(Hex$(Station(StationCount).Slot2(1)), "00")
Startup = Chr$(&H2) + "EX" + " " + "DO" + " " + Inputdata + Chr$(&H3)
+ Chr$(&H0)
Comm1.InBufferCount = 0
Inputdata = " "
RecCount = 0
TxCount = TxCount + 1
If TxCount > 6 Then
  lbRx.Clear
  lbTx.Clear
  TxCount = 1
End If
lbTx.AddItem Startup
Comm1.Output = Chr$(&H2) + "EX" + " " + "DO" + " " + "0" + Chr$(Stati
on(StationCount).Slot2(1)) + Chr$(&H3) + Chr$(&H0)
Do
  Dummy = DoEvents()
  If Comm1.InBufferCount Then
    Inputdata = Inputdata + Comm1.Input
    RecCount = 0
    If InStr(Inputdata, Chr$(&H6)) Then
      lbRx.AddItem Inputdata
      Exit Do
    End If
  End If
  RecCount = RecCount + 1
  If RecCount > 400 Then
    lbRx.AddItem "Station Number " + Format$(StationCount, "00") + " in
correct type !"
    Exit Do
  End If
Loop
End If
timer1.Enabled = True
Case Is = "Analog Output"
timer1.Enabled = False
Station(StationCount).StationInfo = 1
Comm1.InBufferCount = 0
Inputdata = " "
RecCount = 0
TxCount = TxCount + 1
If TxCount > 6 Then
  lbTx.Clear
  lbRx.Clear
  TxCount = 1
End If
lbTx.AddItem Chr$(&H4) + Format$(StationCount, "00") + Chr$(&H5)
Comm1.Output = Chr$(&H4) + Format$(StationCount, "00") + Chr$(&H5)
Do
  Dummy = DoEvents()
  If Comm1.InBufferCount Then
    Inputdata = Inputdata + Comm1.Input
    RecCount = 0
    If InStr(Inputdata, Chr$(&H6)) Then
      lbRx.AddItem Inputdata
      If Len(Trim(Inputdata)) > 3 Then
        Inputdata = Right(Inputdata, 3)
      End If
    End If
    Exit Do
  End If
End If

```

VIEW.FRM - 9

```

        Station(StationCount).Slot0(5) = HtoDec("0", Mid$(Inputdata, 23, 1
), Mid$(Inputdata, 24, 1))
        Station(StationCount).Slot0(6) = HtoDec("0", Mid$(Inputdata, 26, 1
), Mid$(Inputdata, 27, 1))
        Station(StationCount).Slot0(7) = HtoDec("0", Mid$(Inputdata, 29, 1
), Mid$(Inputdata, 30, 1))
        Exit Do
    End If
End If
RecCount = RecCount + 1
If RecCount > 800 Then
    lbRx.AddItem "Station Number " + Format$(StationCount, "00") + " inc
orrect type !"
    Exit Do
End If
Loop
End If
timer1.Enabled = True
VIEW.FRM - 7

```

```

Case Is = "Digital Input"
    timer1.Enabled = False
    Comm1.InBufferCount = 0
    Inputdata = " "
    RecCount = 0
    TxCount = TxCount + 1
    If TxCount > 6 Then
        lbRx.Clear
        lbTx.Clear
        TxCount = 1
    End If
    lbTx.AddItem Chr$(&H4) + Format$(StationCount, "00") + Chr$(&H5)
    Comm1.Output = Chr$(&H4) + Format$(StationCount, "00") + Chr$(&H5)
    Do
        Dummy = DoEvents()
        If Comm1.InBufferCount > 0 Then
            Inputdata = Inputdata + Comm1.Input
            RecCount = 0
            If InStr(Inputdata, Chr(&H6)) Then
                lbRx.AddItem Inputdata
                If Len(Trim(Inputdata)) > 3 Then
                    Inputdata = Right$(Inputdata, 3)
                End If
            End If
            Exit Do
        End If
    End If
    RecCount = RecCount + 1
    If RecCount > 400 Then
        lbRx.AddItem "Station Number " + Format$(StationCount, "00") + " Timeo
ut error !"
        Exit Do
    End If
    Loop
    If Trim(Inputdata) = Format$(StationCount, "00") + Chr(&H6) Then
        Startup = Chr$(&H2) + "EX" + " " + "DI" + " " + Chr$(&H3)..+ Chr$(&H0)
        Comm1.InBufferCount = 0
        Inputdata = " "
        RecCount = 0
        TxCount = TxCount + 1
        If TxCount > 6 Then
            lbRx.Clear
            lbTx.Clear
            TxCount = 1

```

```

End If
  lbTx.AddItem Startup
  Comm1.Output = Startup
  Do
    Dummy = DoEvents()
    If Comm1.InBufferCount Then
      Inputdata = Inputdata + Comm1.Input
      RecCount = 0
      If InStr(Inputdata, Chr$(&H3)) Then
        lbRx.AddItem Inputdata
        Inputdata = Trim(Inputdata)
        Inputdata = Mid$(Inputdata, InStr(Inputdata, Chr$(&H2)), 12)
        Station(StationCount).Slot2(0) = HtoDec("0", Mid$(Inputdata, 8, 1)
, Mid$(Inputdata, 9, 1))
        Exit Do
      End If
    End If
    RecCount = RecCount + 1
    If RecCount > 400 Then
      lbRx.AddItem "Station Number " + Format$(StationCount, "00") + " in
correct type !"
VIEW.FRM - 8
    End Do
  End If
Loop
End If
timer1.Enabled = True
Case Is = "Digital Output"
  timer1.Enabled = False
  Station(StationCount).StationInfo = 3
  Comm1.InBufferCount = 0
  Inputdata = " "
  RecCount = 0
  TxCount = TxCount + 1
  If TxCount > 6 Then
    lbTx.Clear
    lbRx.Clear
    TxCount = 1
  End If
  lbTx.AddItem Chr$(&H4) + Format$(StationCount, "00") + Chr$(&H5)
  Comm1.Output = Chr$(&H4) + Format$(StationCount, "00") + Chr$(&H5)
  Do
    Dummy = DoEvents()
    If Comm1.InBufferCount Then
      Inputdata = Inputdata + Comm1.Input
      RecCount = 0
      If InStr(Inputdata, Chr$(&H6)) Then
        lbRx.AddItem Inputdata
        If Len(Trim(Inputdata)) > 3 Then
          Inputdata = Right(Inputdata, 3)
        End If
        Exit Do
      End If
    End If
    RecCount = RecCount + 1
    If RecCount > 200 Then
      lbRx.AddItem "Station Number " + Format$(StationCount, "00") + " Timeou
t error !"
    End Do
  End If
Loop
If Trim(Inputdata) = Format$(StationCount, "00") + Chr$(&H6) Then

```

```

End If
lbTx.AddItem Startup
Comm1.Output = Startup
Do
  Dummy = DoEvents()
  If Comm1.InBufferCount Then
    Inputdata = Inputdata + Comm1.Input
    RecCount = 0
    If InStr(Inputdata, Chr(&H3)) Then
      lbRx.AddItem Inputdata
      Inputdata = Trim(Inputdata)
      Station = Mid$(Inputdata, InStr(Inputdata, Chr(&H2)), 12)
      Station(StationCount).Slot2(0) = HtoDec("0", Mid$(Inputdata, 8, 1)
, Mid$(Inputdata, 9, 1))
      Exit Do
    End If
  End If
  RecCount = RecCount + 1
  If RecCount > 400 Then
    lbRx.AddItem "Station Number " + Format$(StationCount, "00") + " in
correct type !"
VIEW.FRM - 8
  Exit Do
End If
Loop
End If
timer1.Enabled = True
Case Is = "Digital Output"
timer1.Enabled = False
Station(StationCount).StationInfo = 3
Comm1.InBufferCount = 0
Inputdata = " "
RecCount = 0
TxCount = TxCount + 1
If TxCount > 6 Then
  lbTx.Clear
  lbRx.Clear
  TxCount = 1
End If
lbTx.AddItem Chr(&H4) + Format$(StationCount, "00") + Chr(&H5)
Comm1.Output = Chr(&H4) + Format$(StationCount, "00") + Chr(&H5)
Do
  Dummy = DoEvents()
  If Comm1.InBufferCount Then
    Inputdata = Inputdata + Comm1.Input
    RecCount = 0
    If InStr(Inputdata, Chr(&H6)) Then
      lbRx.AddItem Inputdata
      If Len(Trim(Inputdata)) > 3 Then
        Inputdata = Right(Inputdata, 3)
      End If
    End If
  Exit Do
End If
RecCount = RecCount + 1
If RecCount > 200 Then
  lbRx.AddItem "Station Number " + Format$(StationCount, "00") + " Timeou
t error !"
  Exit Do
End If
Loop
If Trim(Inputdata) = Format$(StationCount, "00") + Chr(&H6) Then

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ง

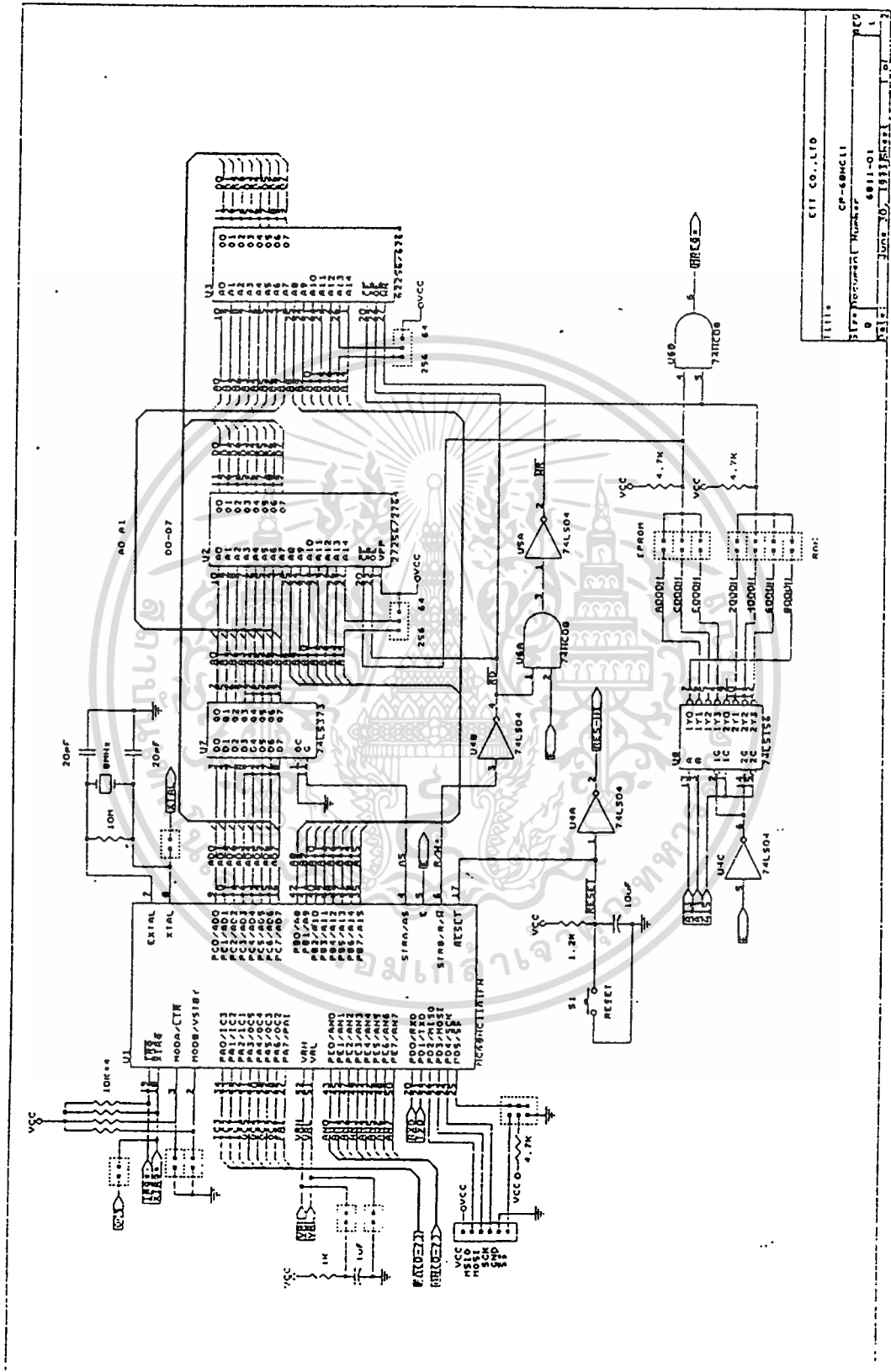
DATA SHEET

SPECIFICATION

CPU/MC#HC11A1FN	32 PIN PLCC
MEMORY	6264/62256 (MAX 32K ADDRESS 52000-59FFF) 2764/27256 (MAX 24K ADDRESS 5A000-5FFFF)
INTERNAL EPROM	512 BYTE
INTERNAL RAM	256 BYTE
PORT	8255 I/O PORT 24 BIT
PRINTER PORT	1 PORT
KEYBOARD	1 PORT 2x4 OR 4x4
LCD	1 LCD MODULE DOT/GRAPHIC
TIMER	3 INPUT/4 OUTPUT AND 1 INPUT/OUTPUT
A/D CHANNEL	
SERIAL PORT	2 CHANNEL SCI (RS232/RS422) & SPI
CLOCK RATE	8 MHz
POWER SUPPLY	INPUT 9V DC
CONNECTOR	
1 40-PIN EXPANSION HEADER-STRIP	(SYSTEM BUS)
1 34-PIN EXPANSION HEADER-STRIP	(PERIPHERAL)
1 20-PIN EXPANSION HEADER-STRIP	(PRINTER)
1 20-PIN EXPANSION HEADER-STRIP	(LCD)
1 14-PIN EXPANSION HEADER-STRIP	(A/D)
1 10-PIN EXPANSION HEADER-STRIP	(TIMER)
1 10-PIN EXPANSION HEADER-STRIP	(KEYBOARD)
1 6-PIN EXPANSION HEADER	(SPI)
1 4-PIN EXPANSION HEADER	(SCI)
1 8-PIN JUMPER	(RAM)
1 6-PIN JUMPER	(EPROM)
1 6-PIN JUMPER	(SS)
1 4-PIN JUMPER	(MODE)
1 4-PIN JUMPER	(VREF)
1 4-PIN JUMPER	(2x3/4x4)
1 3-PIN JUMPER	(2764/27256)
1 3-PIN JUMPER	(6254/62255)
1 3-PIN JUMPER	(CLCD/GLCD)
1 2-PIN JUMPER	(STEP)
1 2-PIN JUMPER	(XTAL)
LED	1 POWER 3AD LED 1 PC7 PORT 6255 ORANGE LED

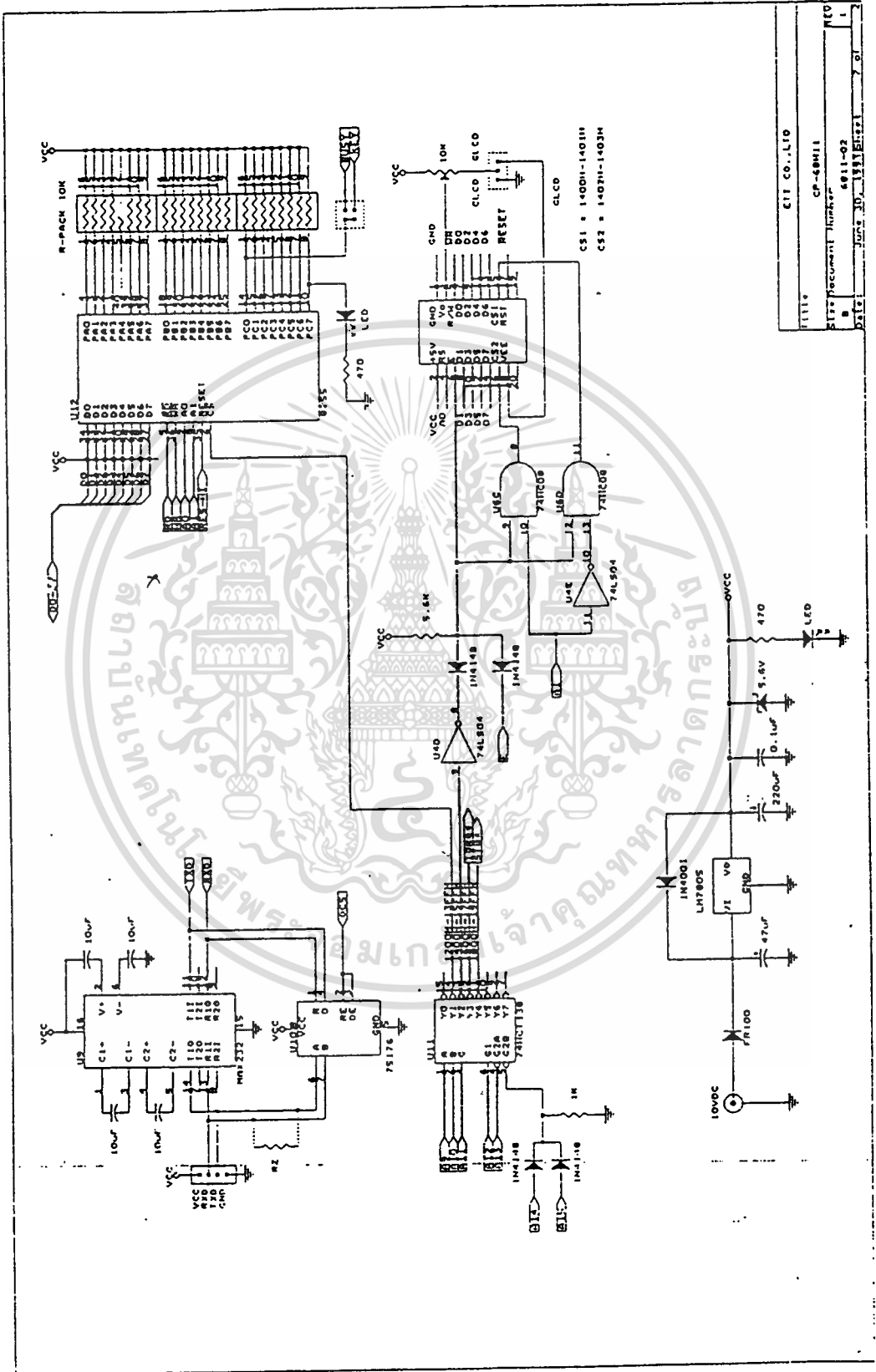
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้ในท้องถิ่น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



Title: CIT CO., LTD
 Cp-60MCL1
 Drawn/Document Number: 6011-01
 Date: June 20, 1981/5/22

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
 ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

MC3487

FIGURE 3 - OUTPUT TRANSITION TIMES TEST CIRCUIT AND WAVEFORMS

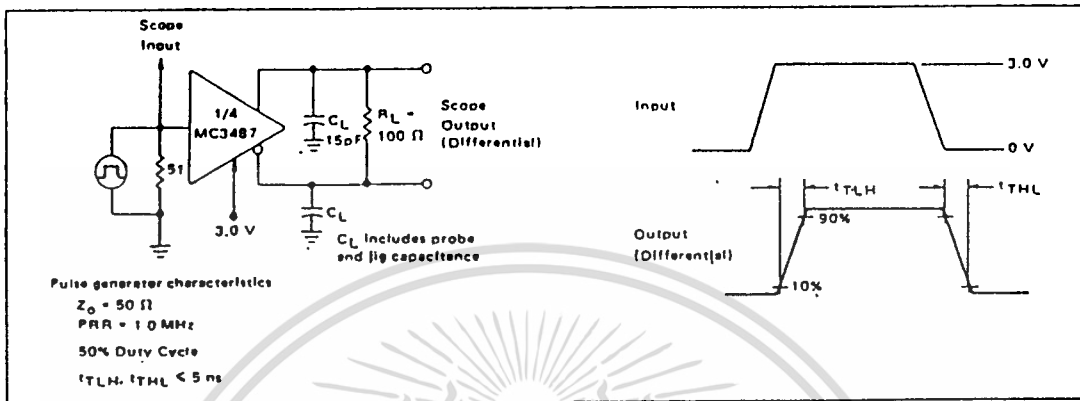


FIGURE 4 - OUTPUT CURRENT versus OUTPUT VOLTAGE

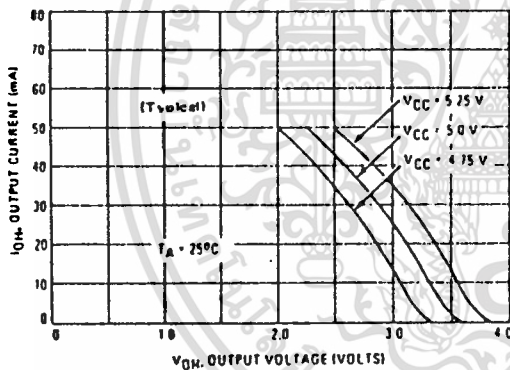
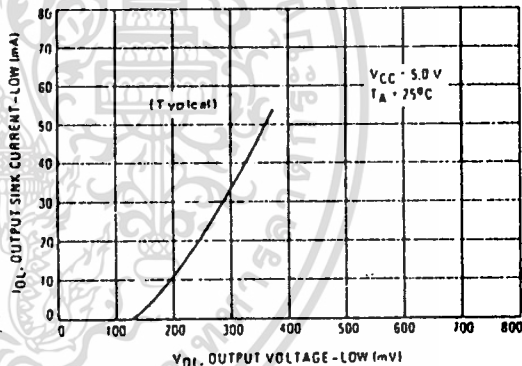


FIGURE 5 - OUTPUT SINK CURRENT versus OUTPUT VOLTAGE



MC3487

FIGURE 1 - THREE-STATE ENABLE TEST CIRCUIT AND WAVEFORMS

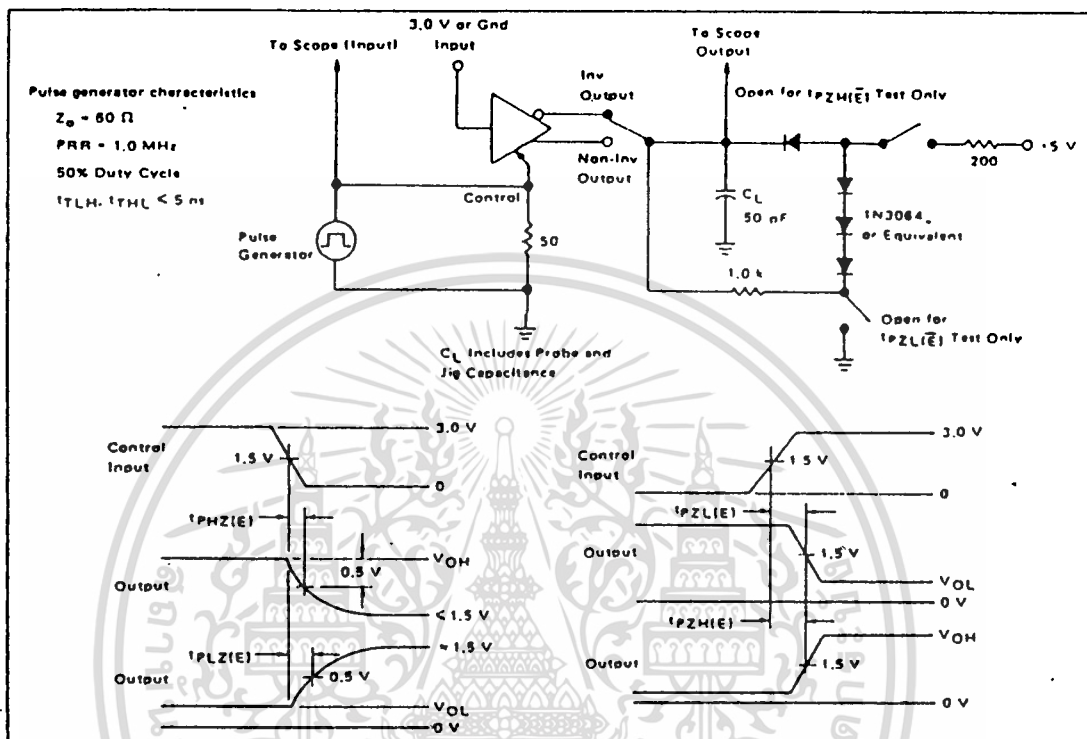
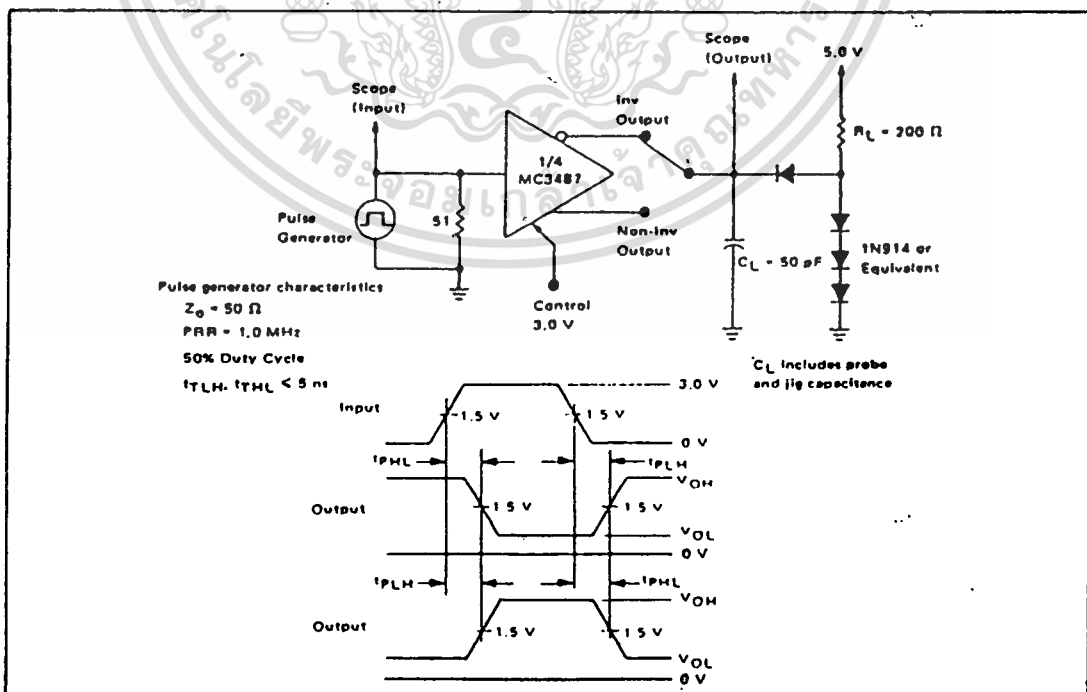


FIGURE 2 - PROPAGATION DELAY TIMES INPUT TO OUTPUT WAVEFORMS AND TEST CIRCUIT



MC3487

MAXIMUM RATINGS

Rating	Symbol	Value	Unit
Power Supply Voltage	V _{CC}	8.0	Vdc
Input Voltage	V _I	5.5	Vdc
Operating Ambient Temperature Range	T _A	0 to +70	°C
Operating Junction Temperature Range	T _J	175 150	°C
Storage Temperature Range	T _{stg}	-85 to +150	°C

ELECTRICAL CHARACTERISTICS (Unless otherwise noted specifications apply 4.75 V < V_{CC} < 5.25 V and 0°C < T_A < 70°C.
Typical values measured at V_{CC} = 5.0 V, and T_A = 25°C.)

Characteristic	Symbol	Min	Typ	Max	Unit
Input Voltage – Low Logic State	V _{IL}	–	–	0.8	Vdc
Input Voltage – High Logic State	V _{IH}	2.0	–	–	Vdc
Input Current – Low Logic State (V _{IL} = 0.5 V)	I _{IL}	–	–	-400	μA
Input Current – High Logic State (V _{IH} = 2.7 V) (V _{IH} = 5.5 V)	I _{IH}	–	–	+50 +100	μA
Input Clamp Voltage (I _{IK} = -18 mA)	V _{IK}	–	–	-1.5	V
Output Voltage – Low Logic State (I _{OL} = 48 mA)	V _{OL}	–	–	0.5	V
Output Voltage – High Logic State (I _{OH} = -20 mA)	V _{OH}	2.5	–	–	V
Output Short-Circuit Current (V _{IH} = 2.0 V, Note 1)	I _{OS}	-40	–	-140	mA
Output Leakage Current – Hi-Z State (V _{IL} = 0.5 V, V _{IL(Z)} = 0.8 V) (V _{IH} = 2.7 V, V _{IH(Z)} = 0.8 V)	I _{OL(Z)}	–	–	+100 +100	μA
Output Leakage Current – Power OFF (V _{OH} = 6.0 V, V _{CC} = 0 V) (V _{OL} = -0.25 V, V _{CC} = 0 V)	I _{OL(off)}	–	–	+100 -100	μA
Output Offset Voltage Difference (Note 2)	V _{OS} - V _{OS}	–	–	+0.4	V
Output Differential Voltage (Note 2)	V _{OD}	2.0	–	–	V
Output Differential Voltage Difference (Note 2)	ΔV _{OD}	–	–	+0.4	V
Power Supply Current (Control Pins = Gnd, Note 3) (Control Pins = 2.0 V)	I _{CCX} I _{CC}	–	–	105 85	mA

- Notes: 1. Only one output may be shorted at a time.
2. See EIA Specification EIA-422 for exact test conditions.
3. Circuit in three-state condition.

SWITCHING CHARACTERISTICS (V_{CC} = 5.0 V, T_A = 25°C unless otherwise noted.)

Characteristic	Symbol	Min	Typ	Max	Unit
Propagation Delay Times					ns
High to Low Output	t _{PHL}	–	–	20	
Low to High Output	t _{PLH}	–	–	20	
Output Transition Times – Differential					ns
High to Low Output	t _{THL}	–	–	20	
Low to High Output	t _{TLH}	–	–	20	
Propagation Delay – Control to Output (R _L = 200 Ω, C _L = 50 pF) (R _L = 200 Ω, C _L = 50 pF) (R _L = ∞, C _L = 50 pF) (R _L = 200 Ω, C _L = 50 pF)	t _{PHZ(E)} t _{PLZ(E)} t _{PZH(E)} t _{PZL(E)}	–	–	25 25 30 30	ns

MOTOROLA
SEMICONDUCTOR
 TECHNICAL DATA

MC3487

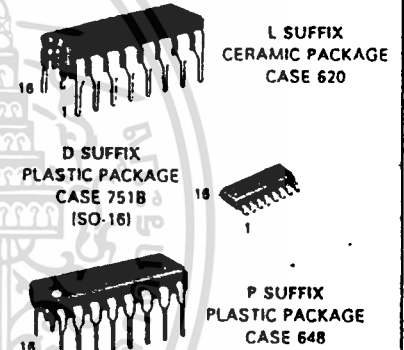
**QUAD LINE DRIVER WITH
 THREE-STATE OUTPUTS**

Motorola's Quad EIA-422 Driver features four independent driver chains which comply with EIA Standards for the Electrical Characteristics of Balanced Voltage Digital Interface Circuits. The outputs are three-state structures which are forced to a high impedance state when the appropriate output control pin reaches a logic zero condition. All input pins are PNP buffered to minimize input loading for either logic one or logic zero inputs. In addition, internal circuitry assures a high impedance output state during the transition between power up and power down. A summary of MC3487 features include:

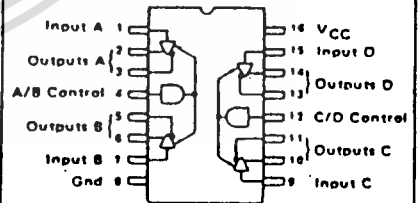
- Four Independent Driver Chains
- Three-State Outputs
- PNP High Impedance Inputs (PIA Compatible)
- Fast Propagation Times (Typ 15 ns)
- TTL Compatible
- Single 5 V Supply Voltage
- Output Rise and Fall Times Less Than 20 ns
- DS 3487 Provides Second Source

**QUAD EIA-422 LINE DRIVER
 WITH THREE-STATE
 OUTPUTS**

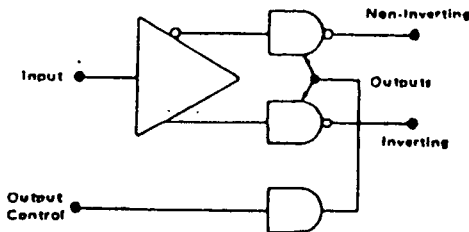
**SILICON MONOLITHIC
 INTEGRATED CIRCUIT**



PIN CONNECTIONS



DRIVER BLOCK DIAGRAM



TRUTH TABLE

Input	Control Input	Non-Inverting Output	Inverting Output
H	H	H	L
L	H	L	H
X	L	Z	Z

L = Low Logic State
 H = High Logic State
 X = Irrelevant
 Z = Third-State (High Impedance)

MC3486

SWITCHING CHARACTERISTICS (Unless otherwise noted, $V_{CC} = 5.0\text{ V}$ and $T_A = 25^\circ\text{C}$.)

Characteristic	Symbol	Min	Typ	Max	Unit
Propagation Delay Time – Differential Inputs to Output (Output High to Low) (Output Low to High)	$t_{PHL(D)}$	–	–	35	ns
	$t_{PLH(D)}$	–	–	30	ns
Propagation Delay Time – Three-State Control to Output (Output Low to Third State) (Output High to Third State) (Output Third State to High) (Output Third State to Low)	t_{PLZ}	–	–	35	ns
	t_{PHZ}	–	–	35	ns
	t_{PZH}	–	–	30	ns
	t_{PZL}	–	–	30	ns

FIGURE 1 – SWITCHING TEST CIRCUIT AND WAVEFORMS

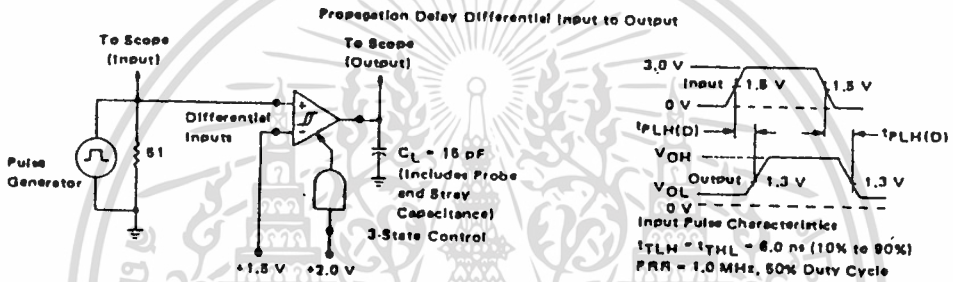
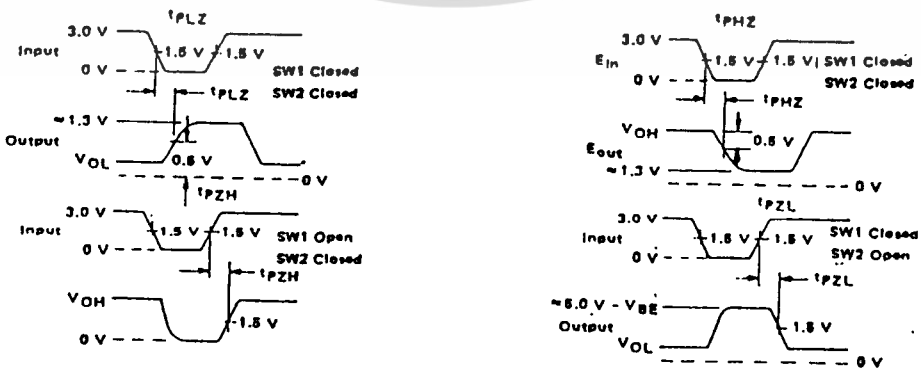
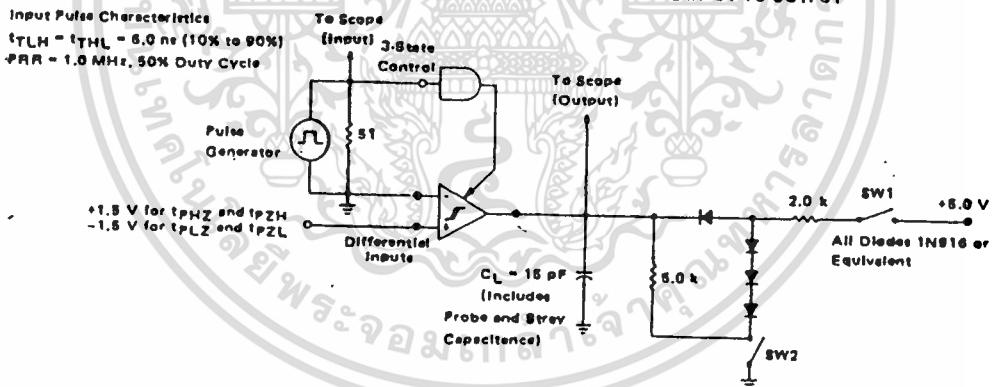


FIGURE 2 – PROPAGATION DELAY THREE-STATE CONTROL INPUT TO OUTPUT



MC3486

MAXIMUM RATINGS

Rating	Symbol	Value	Unit
Power Supply Voltage	V _{CC}	8.0	V _{dc}
Input Common Mode Voltage	V _{ICM}	±15	V _{dc}
Input Differential Voltage	V _{ID}	±25	V _{dc}
Three-State Control Input Voltage	V _I	8.0	V _{dc}
Output Sink Current	I _O	50	mA
Storage Temperature	T _{stg}	-65 to +150	°C
Operating Junction Temperature	T _J		°C
	Ceramic Package	+175	
	Plastic Package	+180	

RECOMMENDED OPERATING CONDITIONS

Rating	Symbol	Value	Unit
Power Supply Voltage	V _{CC}	4.75 to 5.25	V _{dc}
Operating Ambient Temperature	T _A	0 to +70	°C
Input Common Mode Voltage Range	V _{ICR}	-7.0 to +7.0	V _{dc}
Input Differential Voltage Range	V _{IDR}	8.0	V _{dc}

ELECTRICAL CHARACTERISTICS (Unless otherwise noted minimum and maximum limits apply over recommended temperature and power supply voltage ranges. Typical values are for T_A = 25°C, V_{CC} = 5.0 V and V_{IK} = 0 V. See Note 1.)

Characteristic	Symbol	Min	Typ	Max	Unit
Input Voltage — High Logic State (Three-State Control)	V _{IH}	2.0	—	—	V
Input Voltage — Low Logic State (Three-State Control)	V _{IL}	—	—	0.8	V
Differential Input Threshold Voltage, Note 2 (-7.0 V ≤ V _{IC} ≤ 7.0 V, V _{IH} = 2.0 V) (I _O = -0.4 mA, V _{OH} ≥ 2.7 V) (I _O = 8.0 mA, V _{OL} ≥ 0.5 V)	V _{TH(D)}	—	—	0.2 -0.2	V
Input Bias Current (V _{CC} = 0 V or 5.25) (Other inputs at 0 V) (V _I = -10 V) (V _I = -3.0 V) (V _I = +3.0 V) (V _I = +10 V)	I _{B(D)}	—	—	-3.25 -1.50 +1.50 +3.25	mA
Input Balance and Output Level (-7.0 V ≤ V _{IC} ≤ 7.0 V, V _{IH} = 2.0 V, Note 3) (I _O = -0.4 mA, V _{ID} = 0.4 V) (I _O = 8.0 mA, V _{ID} = -0.4 V)	V _{OH} V _{OL}	2.7 —	— —	— 0.5	V
Output Third State Leakage Current (V _{I(D)} = +3.0 V, V _{IL} = 0.8 V, V _{OL} = 0.5 V) (V _{I(D)} = -3.0 V, V _{IH} = 0.8 V, V _{OH} = 2.7 V)	I _{OZ}	—	—	-40 40	μA
Output Short-Circuit Current (V _{I(D)} = 3.0 V, V _{IH} = 2.0 V, V _O = 0 V, Note 4)	I _{OS}	-15	—	-100	mA
Input Current — Low Logic State (Three-State Control) (V _{IL} = 0.5 V)	I _{IL}	—	—	-100	μA
Input Current — High Logic State (Three-State Control) (V _{IH} = 2.7 V) (V _{IH} = 5.25 V)	I _{IH}	—	—	20 100	μA
Input Clamp Diode Voltage (Three-State Control) (I _{IK} = -10 mA)	V _{IK}	—	—	-1.5	V
Power Supply Current (V _{IL} = 2.0 V)	I _{CC}	—	—	85	mA

NOTES:

- All currents into device pins are shown as positive, out of device pins are negative. All voltages referenced to ground unless otherwise noted.
- Differential input threshold voltage and guaranteed output levels are done simultaneously for worst case.
- Refer to EIA-422/3 for exact conditions. Input balance and guaranteed output levels are done simultaneously for worst case.
- Only one output at a time should be shorted.

MOTOROLA
SEMICONDUCTOR
 TECHNICAL DATA

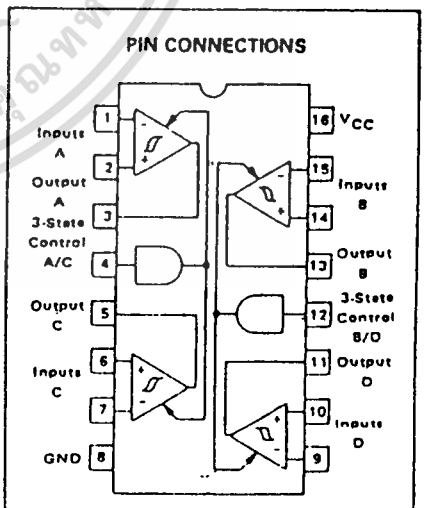
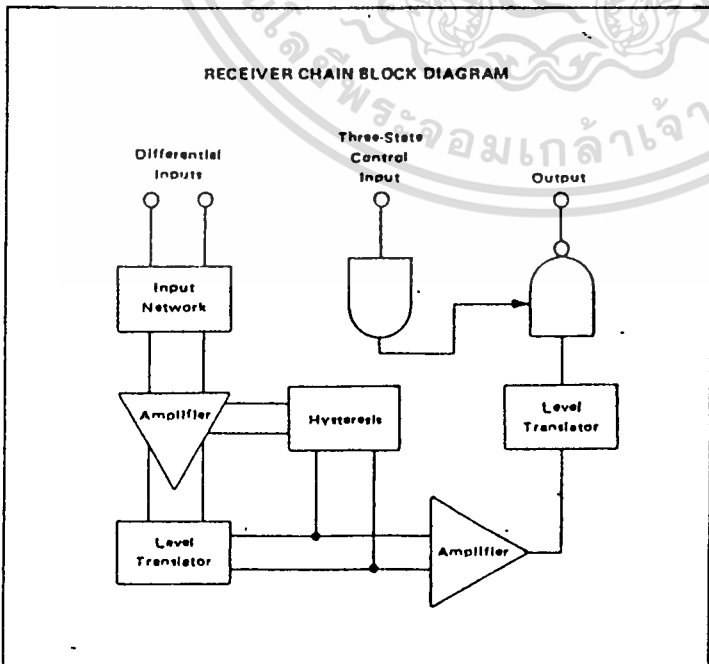
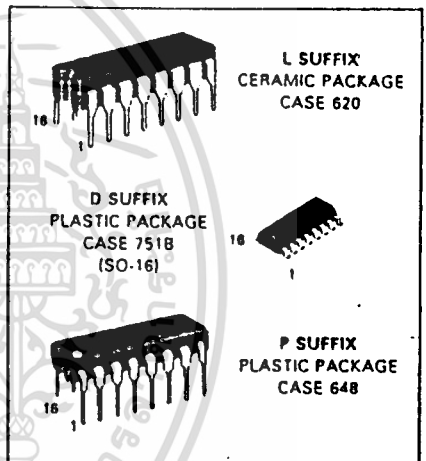
MC3486

QUAD EIA-422/423 LINE RECEIVER WITH THREE-STATE OUTPUTS
 SILICON MONOLITHIC INTEGRATED CIRCUIT

QUAD EIA-422/423 LINE RECEIVER

Motorola's Quad EIA-422/3 Receiver features four independent receiver chains which comply with EIA Standards for the Electrical Characteristics of Balanced/Unbalanced Voltage Digital Interface Circuits. Receiver outputs are 74LS compatible, three-state structures which are forced to a high impedance state when the appropriate output control pin reaches a logic zero condition. A PNP device buffers each output control pin to assure minimum loading for either logic one or logic zero inputs. In addition, each receiver chain has internal hysteresis circuitry to improve noise margin and discourage output instability for slowly changing input waveforms. A summary of MC3486 features include:

- Four Independent Receiver Chains
- Three-State Outputs
- High Impedance Output Control Inputs (PIA Compatible)
- Internal Hysteresis - 30 mV (Typ) @ Zero Volts Common Mode
- Fast Propagation Times - 25 ns (Typ)
- TTL Compatible
- Single 5.0 V Supply Voltage
- DS 3486 Provides Second Source



ORDERING INFORMATION

Device	Temperature Range	Package
MC3486L	0 to +70°C	Ceramic DIP
MC3486P		Plastic DIP
MC3486D		SO-16



MC14051B
MC14052B
MC14053B

ANALOG MULTIPLEXERS/DEMULTIPLEXERS

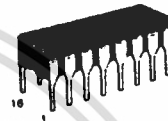
The MC14051B, MC14052B, and MC14053B analog multiplexers are digitally-controlled analog switches. The MC14051B effectively implements an SP8T solid state switch, the MC14052B a DP4T, and the MC14053B a Triple SPDT. All three devices feature low ON impedance and very low OFF leakage current. Control of analog signals up to the complete supply voltage range can be achieved.

- Triple Diode Protection on Control Inputs
- Switch Function is Break Before Make
- Supply Voltage Range - 3.0 Vdc to 18 Vdc
- Analog Voltage Range ($V_{DD} - V_{EE}$) = 3.0 to 18 V
 Note: V_{EE} must be $\leq V_{SS}$
- Linearized Transfer Characteristics
- Low-noise - 12 nV/ $\sqrt{\text{Cycle}}$, $f \geq 1.0$ kHz Typical
- Pin-for-Pin Replacement for CD4051, CD4052, and CD4053
- For 4PDT Switch, See MC14551B
- For Lower R_{ON} , Use the HC4051, HC4052, or HC4053 High-Speed CMOS Devices

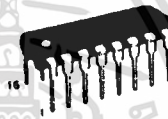
MAXIMUM RATINGS*

Symbol	Parameter	Value	Unit
V_{DD}	DC Supply Voltage (Referenced to V_{EE} , $V_{SS} \geq V_{EE}$)	-0.5 to +18.0	V
V_{in}, V_{out}	Input or Output Voltage (DC or Transient) (Referenced to V_{SS} for Control Inputs and V_{EE} for Switch I/O)	-0.5 to $V_{DD} + 0.5$	V
I_{in}	Input Current (DC or Transient), per Control Pin	± 10	mA
I_{sw}	Switch Through Current	± 25	mA
P_D	Power Dissipation, per Package†	500	mW
T_{stg}	Storage Temperature	-65 to +150	°C
T_L	Lead Temperature (8-Second Soldering)	260	°C

*Maximum Ratings are those values beyond which damage to the device may occur.
 †Temperature Derating Plastic "P" and "DDW" Packages: -7.0 mW/°C From 65°C To 125°C
 Ceramic "L" Packages: -12 mW/°C From 100°C To 125°C



L SUFFIX
 CERAMIC
 CASE 620



P SUFFIX
 PLASTIC
 CASE 648



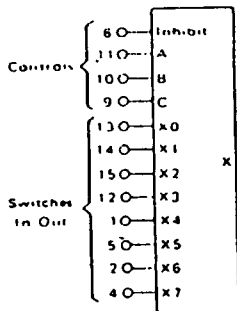
D SUFFIX
 SOIC
 CASE 751B

ORDERING INFORMATION

MC14XXXBCP Plastic
 MC14XXXBCL Ceramic
 MC14XXXBD SOIC

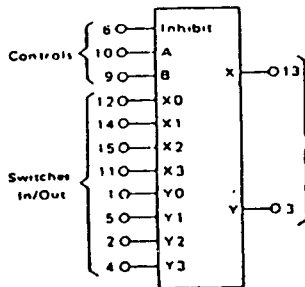
T_A - 55 to 125°C for all packages.

MC14051B
 8-Channel Analog
 Multiplexer/Demultiplexer



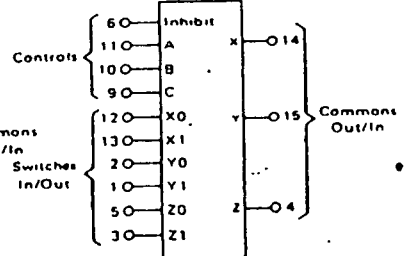
V_{DD} - Pin 16
 V_{SS} - Pin 8
 V_{EE} - Pin 7

MC14052B
 Dual 4-Channel Analog
 Multiplexer/Demultiplexer



V_{DD} - Pin 16
 V_{SS} - Pin 8
 V_{EE} - Pin 7

MC14053B
 Triple 2-Channel Analog
 Multiplexer/Demultiplexer



V_{DD} - Pin 16
 V_{SS} - Pin 8
 V_{EE} - Pin 7

Note: Control inputs referenced to V_{SS} . Analog inputs and outputs reference to V_{EE} . V_{EE} must be $\leq V_{SS}$.

MC14051B•MC14052B•MC14053B

ELECTRICAL CHARACTERISTICS

Characteristic	Symbol	V _{DD}	Test Conditions	-55°C		25°C			125°C		Unit
				Min	Max	Min	Typ #	Max	Min	Max	

SUPPLY REQUIREMENTS (Voltages Referenced to V_{EE})

Power Supply Voltage Range	V _{DD}	—	V _{DD} 3.0 ~ V _{SS} V _{EE}	3.0	18	3.0	—	18	3.0	18	V		
Quiescent Current Per Package	I _{DD}	5.0	Control Inputs: V _{in} = V _{SS} or V _{DD} . Switch I/O: V _{EE} ~ V _{I/O} ~ V _{DD} , and ΔV _{switch} < 500 mV**	—	5.0	—	0.005	5.0	—	150	μA		
		10		—	10	—	0.010	10	—	300			
		15		—	20	—	0.015	20	—	600			
Total Supply Current (Dynamic Plus Quiescent, Per Package)	I _{DD(AV)}	5.0 15	T _A = 25°C only (The channel component, (V _{in} - V _{out}) R _{on} , is not included)	Typical			(0.07 μA/kHz)†	I _{DD}	(0.20 μA/kHz)†	I _{DD}	(0.36 μA/kHz)†	I _{DD}	μA

CONTROL INPUTS — INHIBIT, A, B, C (Voltages Referenced to V_{SS})

Low-Level Input Voltage	V _{IL}	5.0	R _{on} = per spec. I _{off} = per spec	—	1.5	—	2.25	1.5	—	1.5	V
		10		—	3.0	—	4.50	3.0	—	3.0	
		15		—	4.0	—	6.75	4.0	—	4.0	
High-Level Input Voltage	V _{IH}	5.0	R _{on} = per spec. I _{off} = per spec	3.5	—	3.5	2.75	—	3.5	—	V
		10		7.0	—	7.0	5.50	—	7.0	—	
		15		11	—	11	8.25	—	11	—	
Input Leakage Current	I _{in}	15	V _{in} = 0 or V _{DD}	—	± 0.1	—	± 0.00001	± 0.1	—	± 1.0	μA
Input Capacitance	C _{in}	—	—	—	—	—	5.0	7.5	—	—	pF

SWITCHES IN/OUT AND COMMONS OUT/IN — X, Y, Z (Voltages Referenced to V_{EE})

Recommended Peak-to-Peak Voltage Into or Out of the Switch	V _{I/O}	—	Channel On or Off	0	V _{DD}	0	—	V _{DD}	0	V _{DD}	V _{PP}
Recommended Static or Dynamic Voltage Across the Switch** (Figure 5)	ΔV _{switch}	—	Channel On	0	600	0	—	600	0	300	mV
Output Offset Voltage	V _{OO}	—	V _{in} = 0 V, No Load	—	—	—	10	—	—	—	μV
ON Resistance	R _{on}	5.0	ΔV _{switch} < 500 mV**, V _{in} = V _{IL} or V _{IH} (Control), and V _{in} = 0 to V _{DD} (Switch)	—	800	—	250	1050	—	1200	Ω
		10		—	400	—	120	500	—	520	
		15		—	220	—	80	280	—	300	
ΔON Resistance Between Any Two Channels in the Same Package	ΔR _{on}	5.0		—	70	—	25	70	—	135	Ω
		10		—	50	—	10	50	—	95	
		15		—	45	—	10	45	—	65	
Off-Channel Leakage Current (Figure 10)	I _{off}	15	V _{in} = V _{IL} or V _{IH} (Control) Channel to Channel or Any One Channel	—	± 100	—	± 0.05	± 100	—	± 1000	nA
Capacitance, Switch I/O	C _{I/O}	—	Inhibit = V _{DD}	—	—	—	10	—	—	—	pF
Capacitance, Common O/I	C _{O/I}	—	Inhibit = V _{DD} (MC14051B) (MC14052B) (MC14053B)	—	—	—	60	—	—	—	pF
		—		—	—	32	—	—			
		—		—	—	17	—	—			
Capacitance, Feedthrough (Channel Off)	C _{I/O}	—	Pins Not Adjacent	—	—	—	0.15	—	—	—	pF
		—		Pins Adjacent	—	—	—	0.47	—	—	

Data labeled "Typ" is not to be used for design purposes, but is intended as an indication of the IC's potential performance.

** For voltage drops across the switch (ΔV_{switch}) > 600 mV (> 300 mV at high temperature), excessive V_{DD} current may be drawn; i.e. the current out of the switch may contain both V_{DD} and switch input components. The reliability of the device will be unaffected unless the Maximum Ratings are exceeded. (See first page of this data sheet.)

MC14051B•MC14052B•MC14053B

ELECTRICAL CHARACTERISTICS* (C_L = 50 pF, T_A = 25°C) (V_{EE} < V_{SS} unless otherwise indicated)

Characteristic	Symbol	V _{DD} -V _{EE} V _{dC}	Typ # All Types	Max	Unit
Propagation Delay Times (Figure 6) Switch Input to Switch Output (R _L = 10 kΩ)	¹ PLH, ¹ PHL				ns
MC14051					
¹ PLH, ¹ PHL = (0.17 ns/pF) C _L + 26.5 ns		5.0	35	90	
¹ PLH, ¹ PHL = (0.08 ns/pF) C _L + 11 ns		10	15	40	
¹ PLH, ¹ PHL = (0.06 ns/pF) C _L + 9.0 ns		15	12	30	
MC14052					ns
¹ PLH, ¹ PHL = (0.17 ns/pF) C _L + 21.6 ns		5.0	30	76	
¹ PLH, ¹ PHL = (0.08 ns/pF) C _L + 8.0 ns		10	12	30	
¹ PLH, ¹ PHL = (0.06 ns/pF) C _L + 7.0 ns		15	10	25	
MC14053					ns
¹ PLH, ¹ PHL = (0.17 ns/pF) C _L + 16.6 ns		5.0	25	65	
¹ PLH, ¹ PHL = (0.08 ns/pF) C _L + 4.0 ns		10	8.0	20	
¹ PLH, ¹ PHL = (0.06 ns/pF) C _L + 3.0 ns		15	6.0	15	
Inhibit to Output (R _L = 10 kΩ, V _{EE} = V _{SS}) Output "1" or "0" to High Impedance, or High Impedance to "1" or "0" Level	¹ PHZ, ¹ PLZ, ¹ PZH, ¹ PZL				ns
MC14051B		5.0	350	700	
		10	170	340	
		15	140	280	
MC14052B		5.0	300	600	ns
		10	156	310	
		15	125	250	
MC14053B		5.0	275	550	ns
		10	140	280	
		15	110	220	
Control Input to Output (R_L = 10 kΩ, V_{EE} = V_{SS})	¹ PLH, ¹ PHL				ns
MC14051B		5.0	360	720	
		10	180	320	
		15	120	240	
MC14052B		5.0	325	650	ns
		10	130	260	
		15	90	180	
MC14053B		5.0	300	600	ns
		10	120	240	
		15	80	160	
Second Harmonic Distortion (R _L = 10kΩ, f = 1kHz, V _{in} = 5 V _{pp})	—	10	0.07	—	%
Bandwidth (Figure 7) (R _L = 1 kΩ, V _{in} = 1/2 (V _{DD} - V _{EE}) p-p, C _L = 50pF) 20 Log $\frac{V_{out}}{V_{in}}$ = -3 dB	BW	10	17	—	MHz
Off Channel Feedthrough Attenuation (Figure 7) R _L = 1kΩ, V _{in} = 1/2 (V _{DD} - V _{EE}) p-p f _{in} = 4.5 MHz — MC14051B f _{in} = 30 MHz — MC14052B f _{in} = 55 MHz — MC14053B	—	10	-50	—	dB
Channel Separation (Figure 8) R _L = 1 kΩ, V _{in} = 1/2 (V _{DD} - V _{EE}) p-p. f _{in} = 30 MHz	—	10	-50	—	dB
Crosstalk, Control Input to Common Off (Figure 9) (R ₁ = 1 kΩ, R _L = 10 kΩ) Control ¹ T _{LH} = ¹ T _{HL} = 20 ns, Inhibit = V _{SS})	—	10	75	—	mV

*The formulas given are for the typical characteristics only at 25°C
 †Data labelled "Typ" is not to be used for design purposes but is intended as an indication of the IC's potential performance

This device contains protection circuitry to guard against damage due to high static voltages or electric fields. However, precautions must be taken to avoid applications of any voltage higher than maximum rated voltages to this-high-impedance circuit. For proper operation, V_{in} and V_{out} should be constrained to the range V_{SS} < (V_{in} or V_{out}) < V_{DD}. Unused inputs must always be tied to an appropriate logic voltage level (e.g., either V_{SS}, V_{EE}, or V_{DD}). Unused outputs must be left open.

MC14051B•MC14052B•MC14053B

FIGURE 1 – SWITCH CIRCUIT SCHEMATIC

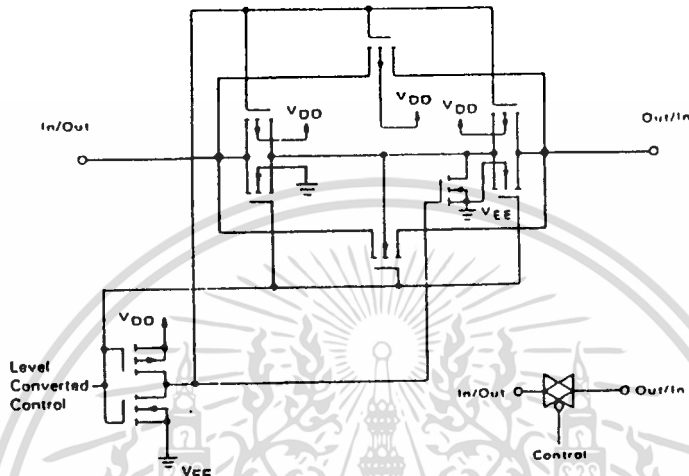


FIGURE 2 – MC14051B FUNCTIONAL DIAGRAM

TRUTH TABLE

Control Inputs			ON Switches		
Inhibit	Select		MC14051B	MC14052B	MC14053B
	C*	B A			
0	0	0	X0	Y0	X0
0	0	0	X1	Y1	X1
0	0	1	X2	Y2	X2
0	0	1	X3	Y3	X3
0	1	0	X4	Z1	X0
0	1	0	X5	Z1	X1
0	1	1	X6	Z1	X0
0	1	1	X7	Z1	X1
1	x	x	None	None	None

*Not applicable for MC14052
x = Don't Care

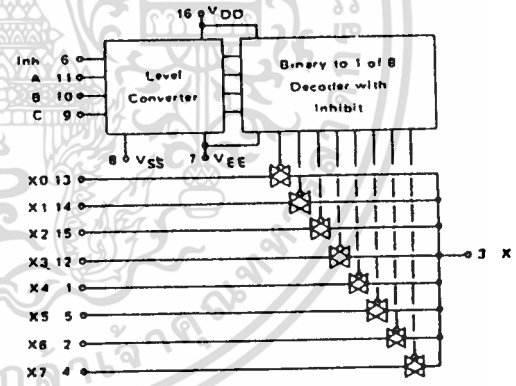


FIGURE 3 – MC14052B FUNCTIONAL DIAGRAM

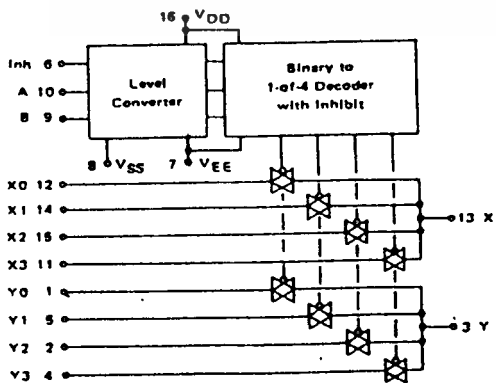
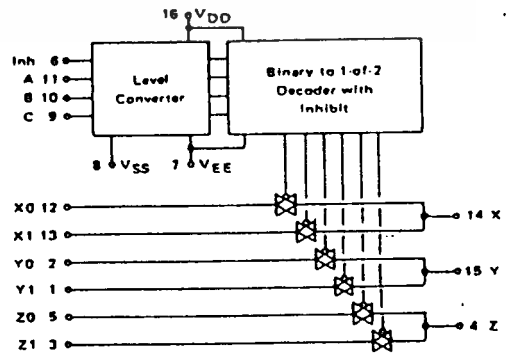


FIGURE 4 – MC14053B FUNCTIONAL DIAGRAM



MC14051B • MC14052B • MC14053B

TEST CIRCUITS

FIGURE 5 — JV ACROSS SWITCH

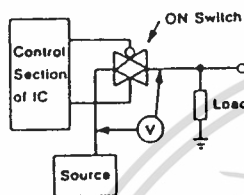


FIGURE 6 — PROPAGATION DELAY TIMES, CONTROL AND INHIBIT TO OUTPUT

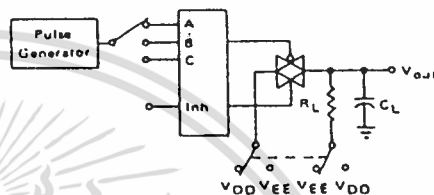


FIGURE 7 — BANDWIDTH AND OFF-CHANNEL FEEDTHROUGH ATTENUATION

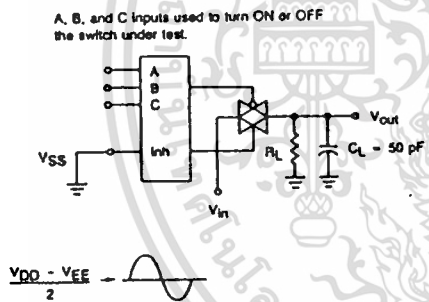


FIGURE 8 — CHANNEL SEPARATION (ADJACENT CHANNELS USED FOR SETUP)

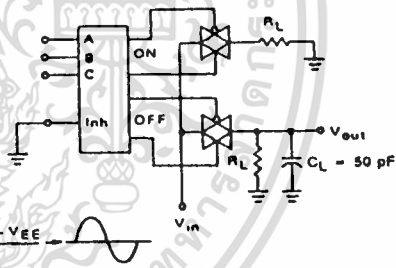


FIGURE 9 — CROSSTALK, CONTROL INPUT TO COMMON O/I

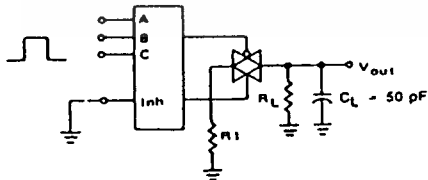
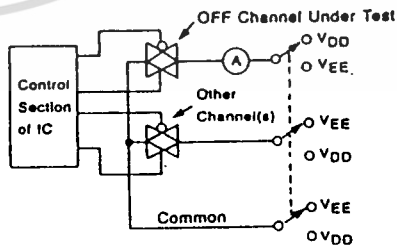


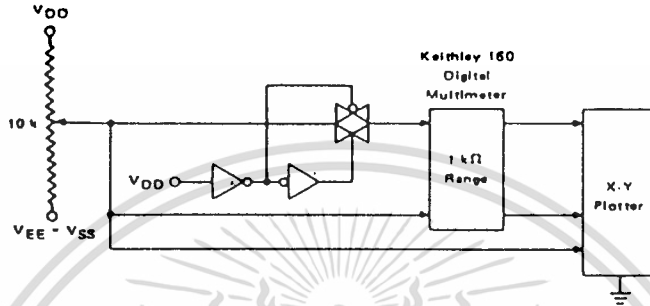
FIGURE 10 — OFF CHANNEL LEAKAGE



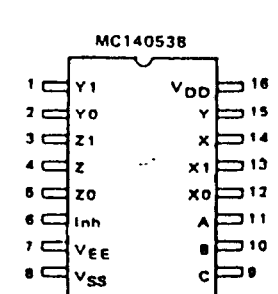
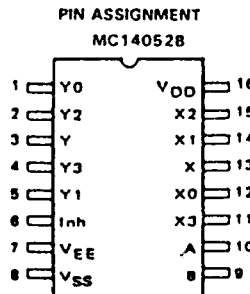
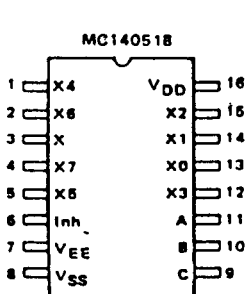
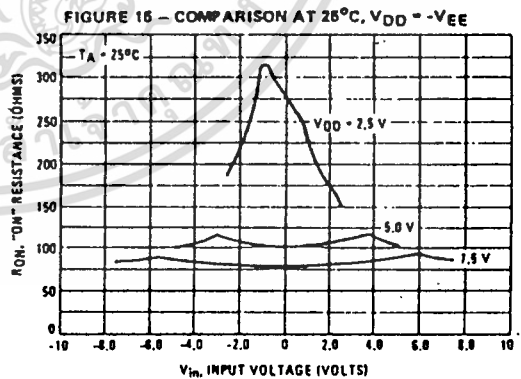
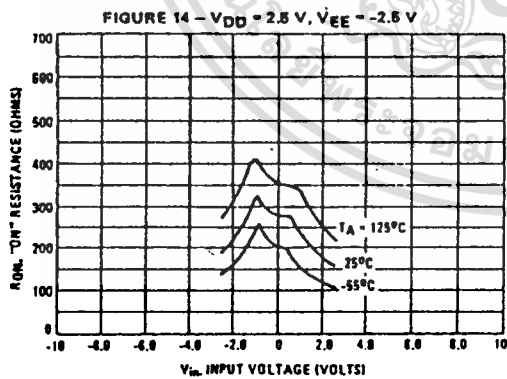
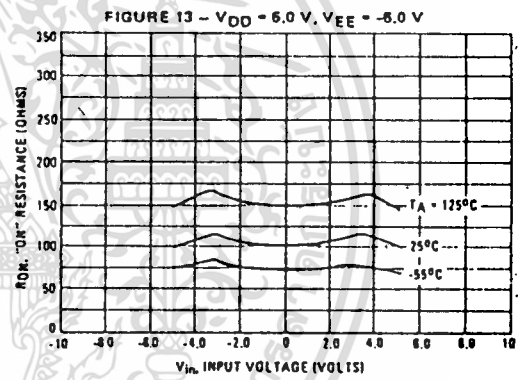
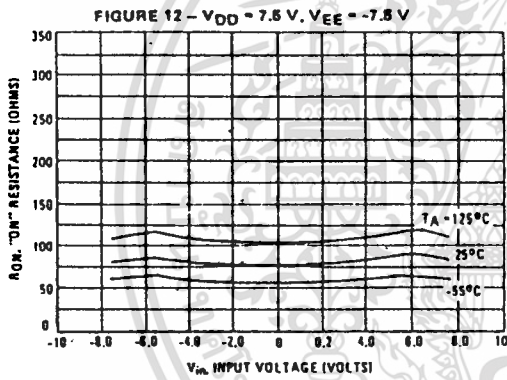
NOTE: See also Figures 7 and 8 on Page 6-51.

MC14051B • MC14052B • MC14053B

FIGURE 11 - CHANNEL RESISTANCE (R_{ON}) TEST CIRCUIT



TYPICAL RESISTANCE CHARACTERISTICS



MC14051B•MC14052B•MC14053B

APPLICATIONS INFORMATION

Figure A illustrates use of the on-chip level converter detailed in Figures 2, 3, and 4. The 0-to-5 V Digital Control signal is used to directly control a 9 V_{p-p} analog signal.

The digital control logic levels are determined by VDD and VSS. The VDD voltage is the logic high voltage; the VSS voltage is logic low. For the example, VDD = +5 V = logic high at the control inputs; VSS = GND = 0 V = logic low.

The maximum analog signal level is determined by VDD and VEE. The VDD voltage determines the maximum recommended peak above VSS. The VEE voltage determines the maximum swing below VSS. For the example, VDD - VSS = 5 V maximum swing above VSS; VSS - VEE = 5 V maximum swing below VSS. The example shows a ± 4.5 V

signal which allows a ½ volt margin at each peak. If voltage transients above VDD and/or below VEE are anticipated on the analog channels, external diodes (D_x) are recommended as shown in Figure B. These diodes should be small signal types able to absorb the maximum anticipated current surges during clipping.

The absolute maximum potential difference between VDD and VEE is 18.0 V. Most parameters are specified up to 15 V which is the recommended maximum difference between VDD and VEE.

Balanced supplies are not required. However, VSS must be greater than or equal to VEE. For example, VDD = +10 V, VSS = +5 V, and VEE = -3 V is acceptable. See the Table below.

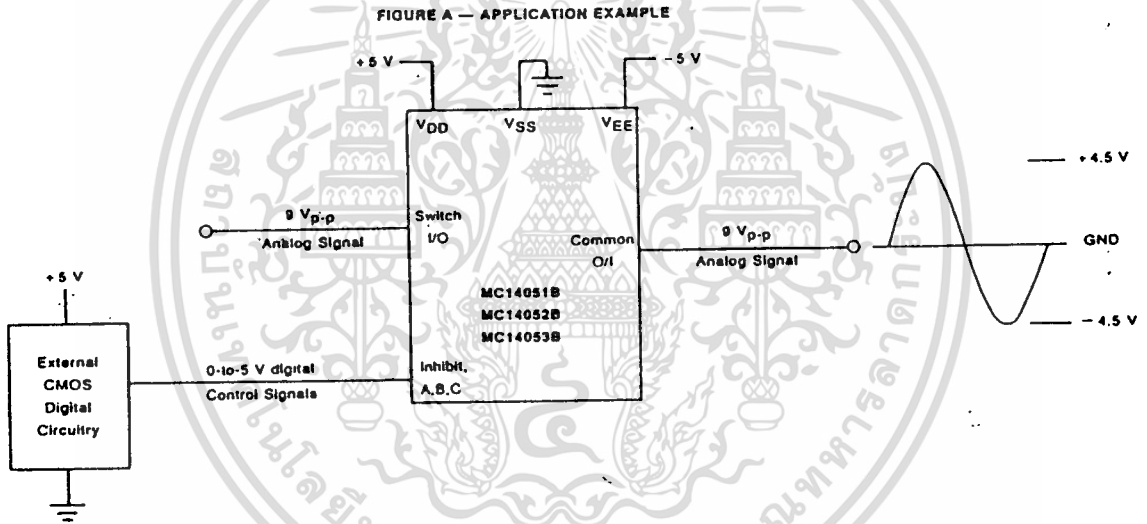
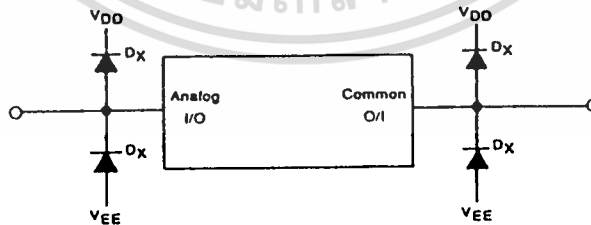


FIGURE B — EXTERNAL GERMANIUM OR SCHOTTKY CLIPPING DIODES



POSSIBLE SUPPLY CONNECTIONS

VDD In Volts	VSS In Volts	VEE In Volts	Control Inputs Logic High/Logic Low In Volts	Maximum Analog Signal Range in Volts
+8	0	-8	+8/0	+8 to -8 = 16 V _{p-p}
+5	0	-12	+5/0	+5 to -12 = 17 V _{p-p}
+5	0	0	+5/0	+5 to 0 = 5 V _{p-p}
+5	0	-5	+5/0	+5 to -5 = 10 V _{p-p}
+10	+5	-5	+10/+5	+10 to -5 = 15 V _{p-p}

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่นอนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

SN65176B, SN75176B
DIFFERENTIAL BUS TRANSCEIVERS

SYMBOL EQUIVALENTS

DATA SHEET PARAMETER	RS-422-A	RS-485
V_O	V_{OH}, V_{OL}	V_{OD}, V_{OL}
V_{OD1}	V_O	V_O
V_{OD2}	$V_I (R_L = 100 \Omega)$	$V_I (R_L = 54 \Omega)$
V_{OD3}		V_I (Test Termination Measurement 2)
ΔV_{OD}	V_I, \bar{V}_I	V_I, \bar{V}_I
V_{OC}	V_{OS}	V_{OS}
ΔV_{OC}	V_{OS}, \bar{V}_{OS}	V_{OS}, \bar{V}_{OS}
I_{OS}	I_{SA}, I_{SB}	
I_O	I_{SA}, I_{SB}	I_A, I_B

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 ¹	MAX	UNIT	
V_{IH} Differential input high threshold voltage	$V_O = 2.7 \text{ V}, I_I = 0.4 \text{ mA}$			0.2	V	
V_{IL} Differential input low threshold voltage	$V_O = 0.5 \text{ V}, I_I = 8 \text{ mA}$	0.2			V	
V_{hys} Hysteresis ²			50		mV	
V_{IK} Enable input clamp voltage	$I_I = 18 \text{ mA}$			1.5	V	
V_{OH} High level output voltage	$V_{ID} = 200 \text{ mV}, I_{OH} = 400 \mu\text{A}$ See Figure 2	2.7			V	
V_{OL} Low-level output voltage	$V_{ID} = 200 \text{ mV}, I_{OL} = 8 \text{ mA}$ See Figure 2			0.45	V	
I_{OZ} High-impedance state output current	$V_O = 0.4 \text{ V to } 2.4 \text{ V}$			20	μA	
I_I Line input current	Other input 0 V, $V_I = 12 \text{ V}$ See Note 6, $V_I = 7 \text{ V}$			1	mA	
I_{IH} High-level enable input current	$V_{IH} = 2.7 \text{ V}$			20	μA	
I_{IL} Low-level enable input current	$V_{IL} = 0.4 \text{ V}$			100	μA	
r_i Input resistance			12		k Ω	
I_{OS} Short circuit output current		15		85	mA	
I_{CC} Supply current (total package)	No load			42	55	mA
				26	35	mA

¹All typical values are at $V_{CC} = 5 \text{ V}, T_A = 25^\circ\text{C}$

²The algebraic convention, in which 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

³Hysteresis is the difference between the positive going input threshold voltage, V_{I+} , and the negative going input threshold voltage, V_{I-} . See Figure 4.

NOTE 6: This applies for both power on and power off. Refer to EIA Standard RS-485 for exact conditions.

receiver switching characteristics, $V_{CC} = 5 \text{ V}, T_A = 25^\circ\text{C}$

PARAMETER	TEST CONDITIONS	MIN	TYP	MAX	UNIT
t_{PLH} Propagation delay time, low to high level output	$V_{ID} = 0 \text{ to } 3 \text{ V}$		21	35	ns
t_{PHL} Propagation delay time, high to low level output	$C_L = 15 \text{ pF}$ See Figure 6		23	35	ns
t_{PZH} Output enable time to high level	$C_L = 15 \text{ pF}$ See Figure 7		10	20	ns
t_{PZL} Output enable time to low level			12	20	ns
t_{PHZ} Output disable time from high level			20	35	ns
t_{PLZ} Output disable time from low level	$C_L = 15 \text{ pF}$ See Figure 7		17	35	ns



SN65176B, SN75176B
DIFFERENTIAL BUS TRANSCEIVERS

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			-1.5	V	
V_O	Output voltage		0	6	V	
$ V_{OD1} $	Differential output voltage		1.5	6	V	
$ V_{OD2} $	Differential output voltage	$R_L = 100 \Omega$, See Figure 1	V_{OD1}			
			2		V	
$ V_{OD3} $	Differential output voltage	$R_L = 54 \Omega$, See Figure 1	1.5	2.5	5	V
		See Note 4	1.5		5	V
$\Delta V_{OD} $	Change in magnitude of differential output voltage [§]			0.2	V	
V_{OC}	Common-mode output voltage	$R_L = 54 \Omega$ or 100Ω , See Figure 1		-3 -1	V	
$\Delta V_{OC} $	Change in magnitude of common mode output voltage [§]			0.2	V	
I_O	Output current	Output disabled, See Note 5	$V_O = 12 \text{ V}$		1	mA
			$V_O = -7 \text{ V}$		-0.8	
I_{IH}	High level input current	$V_I = 2.4 \text{ V}$		20	μA	
I_{IL}	Low level input current	$V_I = 0.4 \text{ V}$		-400	μA	
I_{OS}	Short circuit output current	$V_O = -7 \text{ V}$		-250	mA	
		$V_O = 0$		-150		
		$V_O = V_{CC}$		250		
		$V_O = 12 \text{ V}$		250		
I_{CC}	Supply current (total package)	No load	Outputs enabled	42	55	mA
			Outputs disabled	26	35	

[†] The power-off measurement in EIA Standard RS-422 A applies to disabled outputs only and is not applied to combined inputs and outputs.

[‡] All typical values are at $V_{CC} = 5 \text{ V}$ and $T_A = 25^\circ\text{C}$.

[§] $\Delta|V_{OD}|$ 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.

NOTES: 4. See EIA Standard RS-485 Figure 3.5, Test Termination Measurement 2.

5. This applies for both power on and off; refer to EIA Standard RS-485 for exact conditions. The RS-422-A limit does not apply for a combined driver and receiver terminal.

driver switching characteristics, $V_{CC} = 5 \text{ V}$, $T_A = 25^\circ\text{C}$

PARAMETER	TEST CONDITIONS	MIN	TYP	MAX	UNIT
t_{DD}	Differential-output delay time		15	22	ns
t_{TD}	Differential output transition time	$R_L = 54 \Omega$, See Figure 3	20	30	ns
t_{PZH}	Output enable time to high level	$R_L = 110 \Omega$, See Figure 4	85	120	ns
t_{PLZ}	Output enable time to low level	$R_L = 110 \Omega$, See Figure 5	40	60	ns
t_{PHZ}	Output disable time from high level	$R_L = 110 \Omega$, See Figure 4	150	250	ns
t_{PLZ}	Output disable time from low level	$R_L = 110 \Omega$, See Figure 5	20	30	ns

**SN65176B, SN75176B
DIFFERENTIAL BUS TRANSCEIVERS**

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 power dissipation (see Note 2)	See Dissipation Rating Table
Operating free-air temperature range: SN65176B	40°C to 105°C
SN75176B	0°C to 70°C
Storage temperature range	65°C to 150°C
Lead temperature 1.6 mm (1/16 inch) from the case for 10 seconds: D or P package	260°C
Lead temperature 1.6 mm (1/16 inch) from case for 60 seconds: JG package	300°C

- NOTES: 1. All voltage values, except differential input/output bus voltage, are with respect to network ground terminal.
2. In the JG package, the chips are glass mounted.

DISSIPATION RATING TABLE

PACKAGE	$T_A = 25^\circ\text{C}$	DERATING FACTOR	$T_A = 70^\circ\text{C}$	$T_A = 105^\circ\text{C}$
	POWER RATING	ABOVE $T_A = 25^\circ\text{C}$	POWER RATING	POWER RATING
D	725 mW	5.8 mW/°C	464 mW	261 mW
JG	825 mW	6.6 mW/°C	528 mW	297 mW
P	1100 mW	8.8 mW/°C	702 mW	396 mW

recommended operating conditions

		MIN	TYP	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_O				12	V
High-level input voltage, V_{IH}		D, DE, and RE		2	V
Low-level input voltage, V_{IL}		D, DE, and 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	mA
Operating free air temperature, T_A	SN65176B	40		105	°C
	SN75176B	0		70	°C

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



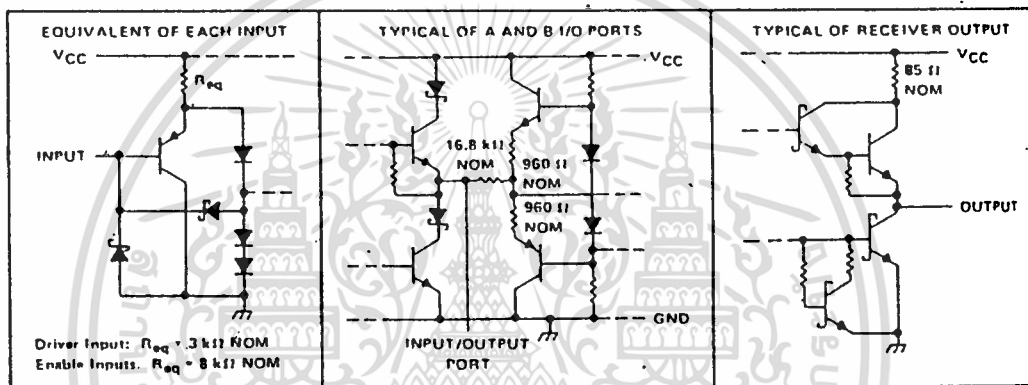
SN65176B, SN75176B DIFFERENTIAL BUS TRANSCEIVERS

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 k Ω , an input sensitivity of +200 mV, and a typical input hysteresis of 50 mV.

The SN65176B and SN75176B can be used in transmission line applications employing the SN75172 and SN75174 quadruple differential line drivers and SN75173 and SN75175 quadruple differential line receivers.

The SN65176B is characterized for operation from -40°C to 105°C and the SN75176B is characterized for operation from 0°C to 70°C.

schematics of inputs and outputs

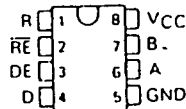


SN65176B, SN75176B DIFFERENTIAL BUS TRANSCEIVERS

D2619 JULY 1985 REVISED SEPTEMBER 1989

- Bidirectional Transceiver
- Meets EIA Standards RS-422-A and RS-485 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, JG, OR P PACKAGE
(TOP VIEW)



FUNCTION TABLE (DRIVER)

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

FUNCTION TABLE (RECEIVER)

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

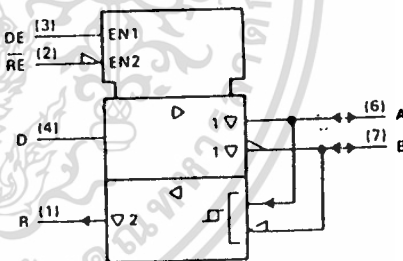
H: high level, L: low level, ? : indeterminate, X: irrelevant, Z: high impedance (off)

description

The SN65176B and SN75176B differential bus transceivers are monolithic integrated circuits designed for bidirectional data communication on multipoint bus transmission lines. They are designed for balanced transmission lines and meet EIA Standard RS-422-A and RS-485 and CCITT Recommendations V.11 and X.27.

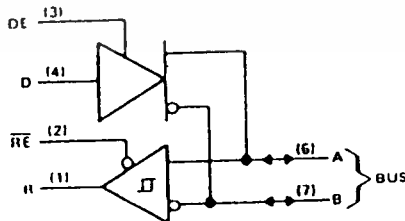
The SN65176B and SN75176B combine a 3-state differential line driver and a differential input line receiver both of which operate from a single 5 V power supply. The driver and receiver have active high and active low enables, respectively, that can be externally connected together to function as a direction 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



This symbol is in accordance with ANSI Std Std 91-1984 and IEC Publication 617-12

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

SN65176B, SN75176B
DIFFERENTIAL BUS TRANSCEIVERS

PARAMETER MEASUREMENT INFORMATION

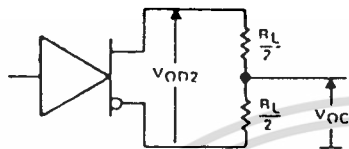


FIGURE 1. DRIVER V_{OD} AND V_{OC}

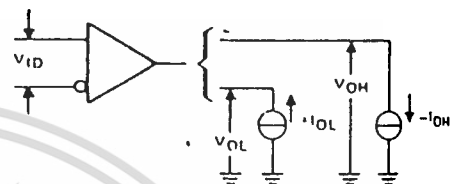


FIGURE 2. RECEIVER V_{OH} AND V_{OL}

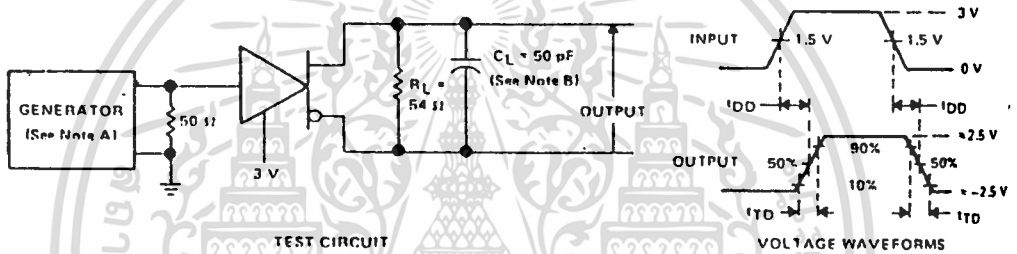


FIGURE 3. DRIVER DIFFERENTIAL-OUTPUT DELAY AND TRANSITION TIMES

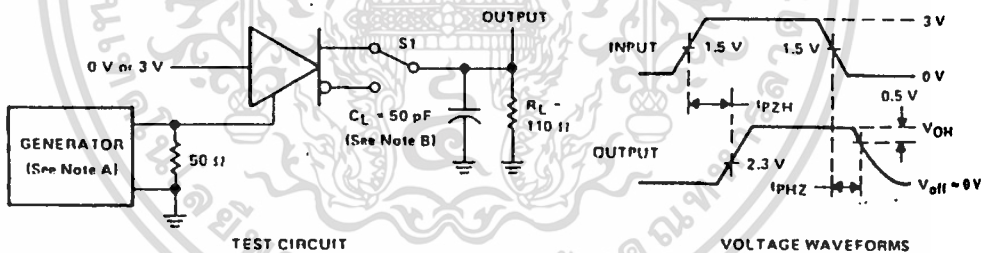


FIGURE 4. DRIVER ENABLE AND DISABLE TIMES

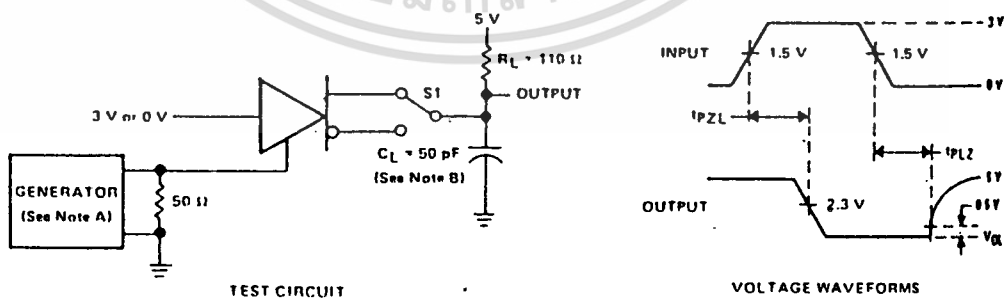


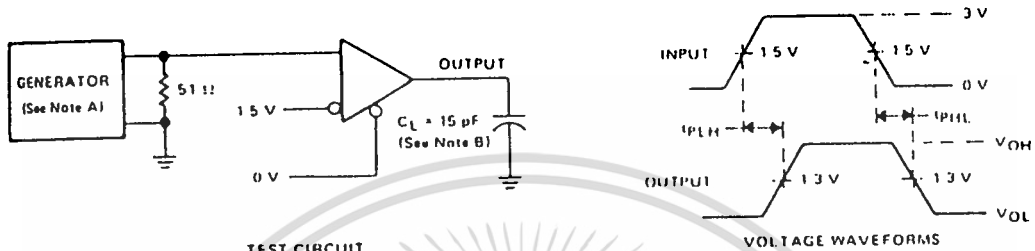
FIGURE 5. DRIVER ENABLE AND DISABLE TIMES

NOTES: A The input pulse is supplied by a generator having the following characteristics: PRR \leq 1 MHz, 50% duty cycle, $t_r < 4$ ns, $t_f < 6$ ns, $Z_{out} = 50 \Omega$.
B C_L includes probe and jig capacitance.



SN65176B, SN75176B
DIFFERENTIAL BUS TRANSCEIVERS

PARAMETER MEASUREMENT INFORMATION



TEST CIRCUIT
FIGURE 6. RECEIVER PROPAGATION DELAY TIMES

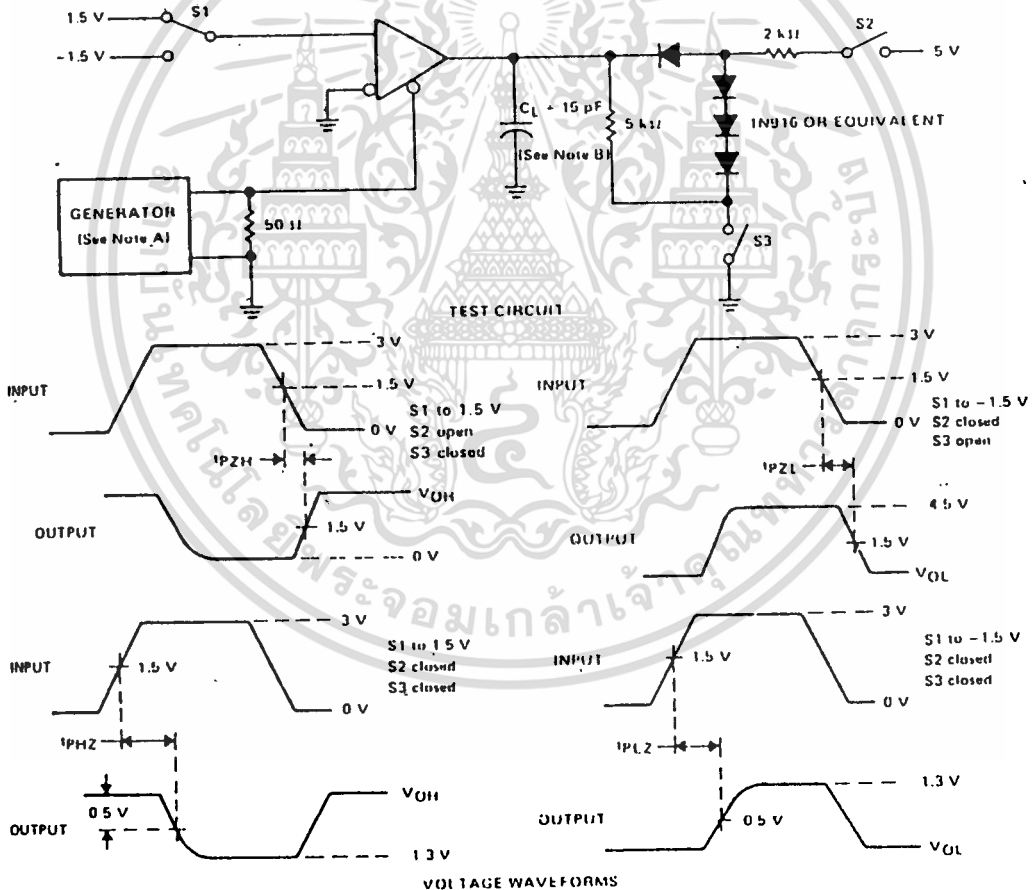


FIGURE 7. RECEIVER OUTPUT ENABLE AND DISABLE TIMES.

NOTES. A The input pulse is supplied by a generator having the following characteristics: $f_{PWH} = 1 \text{ MHz}$, 50% duty cycle, $t_r = 0.05 \text{ ns}$, $t_f = 6 \text{ ns}$, $Z_{out} = 50 \Omega$.
B C_L includes probe and jig capacitance.



SN65176B, SN75176B
DIFFERENTIAL BUS TRANSCEIVERS

TYPICAL CHARACTERISTICS

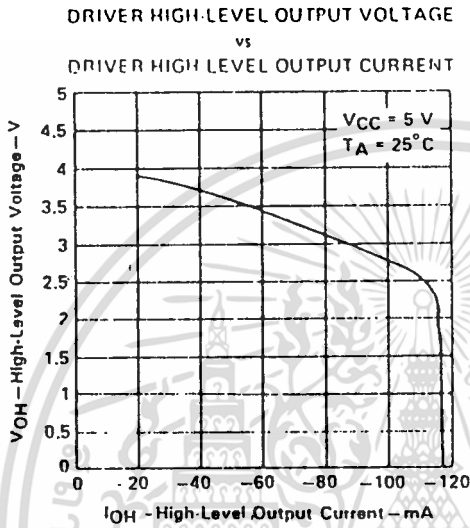


FIGURE 8

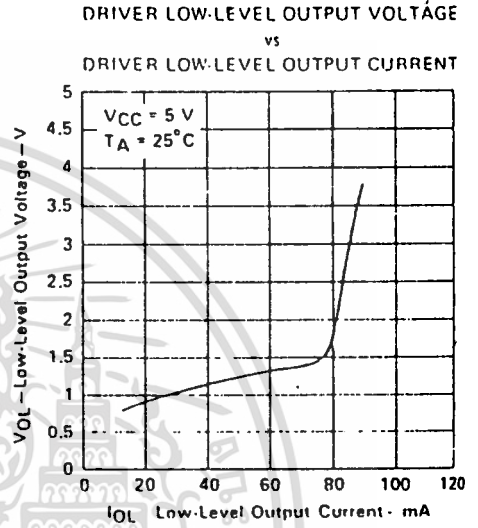


FIGURE 9

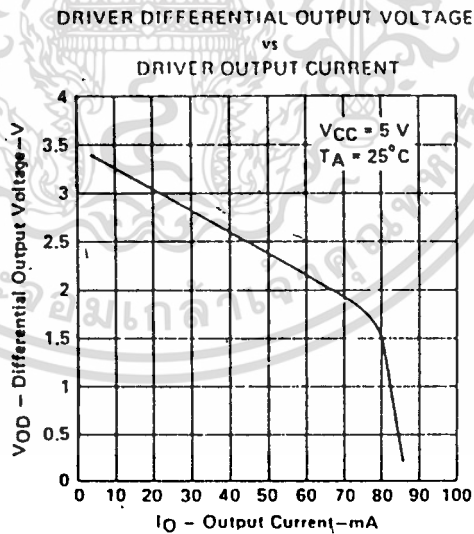


FIGURE 10



SN65176B, SN75176B
DIFFERENTIAL BUS TRANSCEIVERS

TYPICAL CHARACTERISTICS

RECEIVER HIGH-LEVEL OUTPUT VOLTAGE
vs
HIGH LEVEL OUTPUT CURRENT

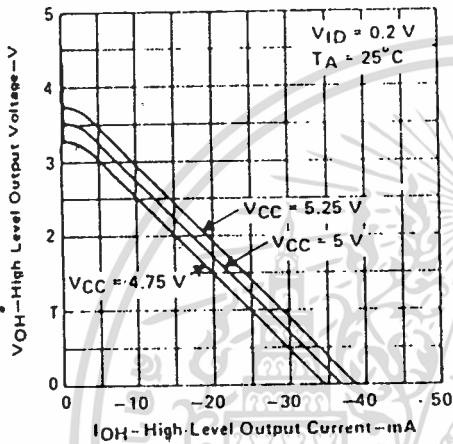


FIGURE 11

RECEIVER HIGH LEVEL OUTPUT
vs
FREE AIR TEMPERATURE

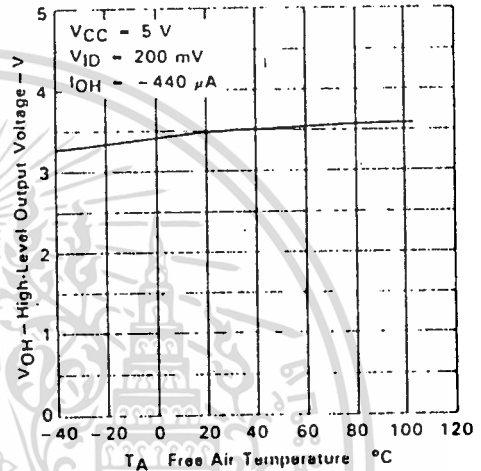


FIGURE 12

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

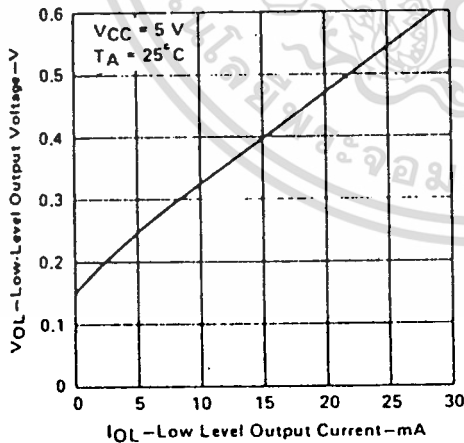


FIGURE 13

RECEIVER LOW LEVEL OUTPUT VOLTAGE
vs
FREE-AIR TEMPERATURE

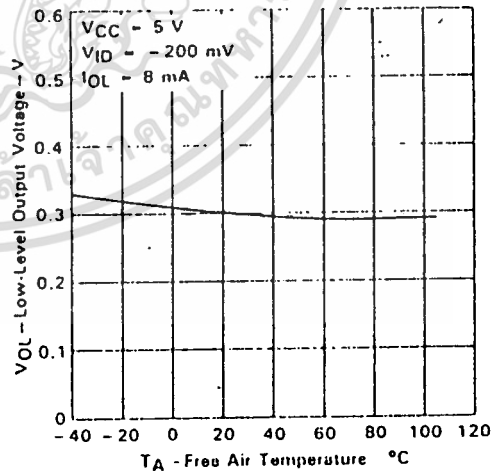


FIGURE 14

SN65176B, SN75176B
DIFFERENTIAL BUS TRANSCEIVERS

TYPICAL CHARACTERISTICS

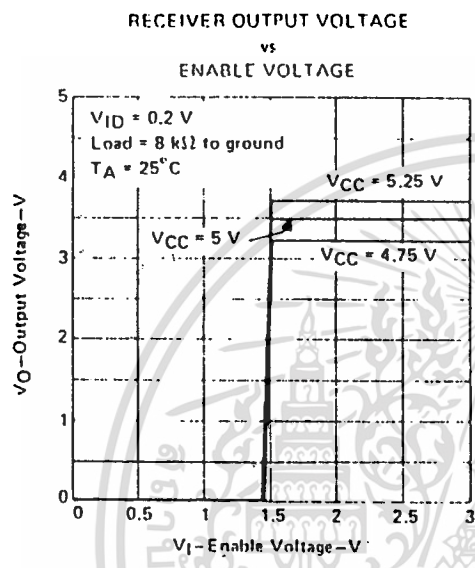


FIGURE 15

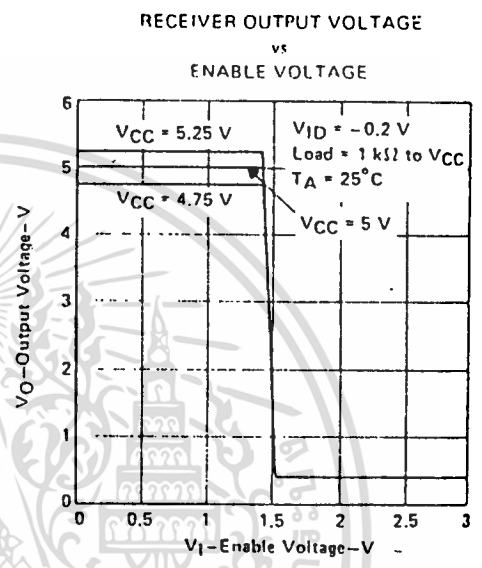


FIGURE 16

TYPICAL APPLICATION

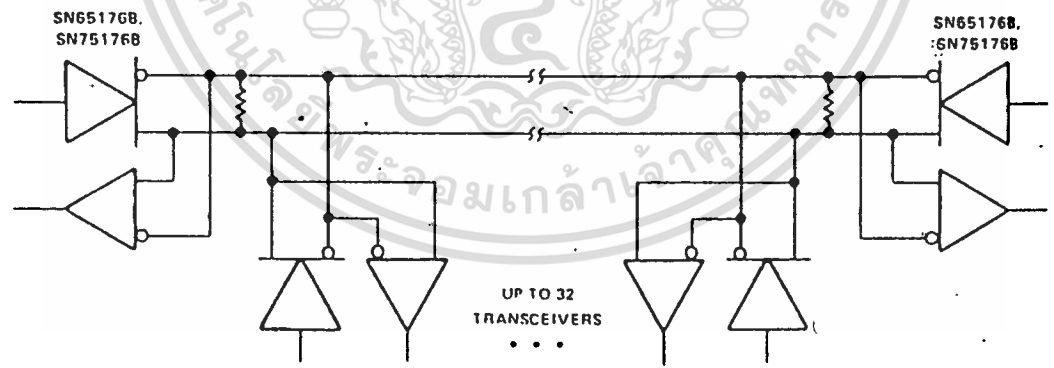


FIGURE 17. TYPICAL APPLICATION CIRCUIT

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.



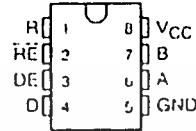
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
 ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

SN65ALS176, SN75ALS176, SN75ALS176A, SN75ALS176B DIFFERENTIAL BUS TRANSCEIVERS

D3042 AUGUST 1987 (REVISIION MAY 1990)

- Meets EIA Standards RS-422A and RS-485, CCITT Recommendations V.11 and X.27, and ISO 8482:1987(E)
- Designed and Tested for Data Rates up to 35 MBaud
- SN65ALS176 Operating Temperature – 40°C to 85°C
- Three Skew Limits Available:
 - ALS176 . . . 10 ns
 - ALS176A . . . 7.5 ns
 - ALS176B . . . 5 ns
- Designed for Multipoint Transmission on Long Bus Lines in Noisy Environments
- Low Supply Current Requirements
30 mA Max
- Wide Positive and Negative Input/Output Bus Voltage Ranges
- Thermal Shutdown Protection
- Driver Positive and Negative Current Limiting
- Receiver Input Hysteresis
- Glitch-Free Power-Up and Power-Down Protection
- Receiver Open-Circuit Fail-Safe Design

D OR P PACKAGE
(TOP VIEW)



FUNCTION TABLE (DRIVER)

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

FUNCTION TABLE (RECEIVER)

DIFFERENTIAL INPUTS A-B	ENABLE RE	OUTPUT R
$V_{DD} = 0.2V$	L	H
$0.2V < V_{DD} < 0.2V$	L	?
$V_{DD} = 0.2V$	L	L
X	H	Z

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

AVAILABLE OPTIONS

T _A	I _{pmax} - I _{pmin} I _{s(kHz)} [†]	PACKAGE	
		SMALL OUTLINE (D) [†]	PLASTIC DIP (P)
0°C	10	SN75ALS176D	SN75ALS176P
to	7.5	SN75ALS176AD	SN75ALS176AP
70°C	5	SN75ALS176BD	SN75ALS176BP
to	10	SN65ALS176D	SN65ALS176P
85°C			

[†] The D package is available taped and reeled. Add the suffix R to the device type, (e.g., SN75ALS176DR).
^{††} I_{s(kHz)} is the greater of 1) the difference between the maximum and minimum specified values of I_{pmax} (or I_{pmin}), and 2) the difference between the maximum and minimum specified values of I_{pmax} (or I_{pmin}). This is the maximum range that the driver or receiver delay time will vary over temperature, V_{CC}, and device to device.

description

The SN65ALS176 and SN75ALS176 series Differential Bus Transceivers are monolithic integrated circuits designed for bidirectional data communication on multipoint bus transmission lines. They are designed for balanced transmission lines and meet EIA Standards RS-422-A and RS-485, CCITT recommendations V.11 and X.27, and ISO 8482:1987(E).

The SN65ALS176 and SN75ALS176 series combine 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, which can be externally connected together to function as a direction control. The driver differential outputs and the receiver differential inputs are connected internally to form a differential input-output (I²O bus port) that is designed to offer minimum loading to the bus whenever the driver is disabled or V_{CC} = 0. This port features wide positive and negative common-mode voltage ranges making the device suitable for party-line applications.

The SN65ALS176 is characterized for operation from -40°C to 85°C and the SN75ALS176 series is characterized for operation from 0°C to 70°C.

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 © 1990 Texas Instruments Incorporated

ประวัติผู้เขียน

ชื่อ นายวิวัฒน์ สุชีวกุล
 วันเดือนปีเกิด วันที่ 28 เมษายน พ.ศ. 2503
 ภูมิลำเนา 49/388 ก. สุขากีบบาล 2 ดลองคุ้ม เขต บึงกุ่ม
 กรุงเทพมหานคร
 ประวัติการศึกษา มศ.3 จาก รร. ราชสีมาวิทยาลัย จ. นครราชสีมา
 ปวช. สาขาช่างไฟฟ้ากำลัง จากวิทยาลัยเทคนิคนครราชสีมา
 ปวส. สาขาช่างไฟฟ้ากำลัง จากวิทยาลัยช่างกลปทุมวัน
 ประวัติการทำงาน องค์การโทรศัพท์แห่งประเทศไทย ตั้งแต่ พ.ศ. 2525 ถึง ปัจจุบัน

ชื่อ นายสมชาย ไชยประกาย
 วันเดือนปีเกิด วันที่ 28 มกราคม พ.ศ. 2514
 ภูมิลำเนา 367 ซ. หลังเทศบาล อ. เมือง จ. สกลนคร
 ประวัติการศึกษา ม. 3 จาก รร. สกลราชวิทยานุกูล จ. สกลนคร
 ปวช. สาขาอิเล็กทรอนิกส์ จากสถาบันเทคโนโลยีราชมงคล วิทยาเขต
 ขอนแก่น (ไทย - เยอรมัน) จ. ขอนแก่น
 ปวส. สาขาอิเล็กทรอนิกส์ จากสถาบันเทคโนโลยีพระจอมเกล้า
 พระนครเหนือ
 ประวัติการทำงาน ปี พ.ศ. 2533 ถึง ปี พ.ศ. 2534 ฝ่ายวิจัย และพัฒนา บริษัท
 ไทเทค ออดิโอ จำกัด
 ปี พ.ศ. 2535 ถึง ปี พ.ศ. 2537 บริษัท วิสโก้ จำกัด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่นิยมนำไปใช้ประโยชน์ด้านการค้า
 ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ชื่อ นายสรายุธ ปราสาทวัฒนา
 วันเดือนปีเกิด วันที่ 10 มิถุนายน พ.ศ. 2510
 ภูมิลำเนา 105/9 ต. ดอนกรวย อ. ดำเนินสะดวก จ. ราชบุรี
 ประวัติการศึกษา ม. 3 จาก รร. สายธรรมจันทร์ จ. ราชบุรี
 ปวช. สาขาช่างไฟฟ้ากำลัง จากวิทยาลัยเทคนิคราชบุรี
 ปวส. สาขาช่างไฟฟ้ากำลัง จากวิทยาลัยเทคนิคราชบุรี
 ประวัติการทำงาน บริษัท พีซีทีที จำกัด ตั้งแต่ พ.ศ. 2531 ถึง ปัจจุบัน



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
 ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้