

โปรแกรมการเชื่อมโยงระหว่างคอมพิวเตอร์และการควบคุมแบบตรรก  
Computer – PLC Interfacing Program



นายวุฒิพล ก่อพัฒน์วงศ์  
Mr. Wutipol Korputanawong

เลขหมู่.....  
เลขทะเบียน 42435  
วัน, เดือน, ปี 23 พ.ค. 2545

b.....  
i.....

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต  
สาขาวิชาวิศวกรรมอุตสาหการ  
คณะวิศวกรรมศาสตร์  
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง  
ปีการศึกษา 2543

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หัวข้อวิทยานิพนธ์      โปรแกรมการเชื่อมโยงระหว่างคอมพิวเตอร์และการควบคุมแบบตรรก  
Computer – PLC Interfacing Program  
นักศึกษา                      นายวุฒิพล ก่อพัฒน์วงศ์  
รหัสประจำตัว                40010757  
ปริญญา                        วิศวกรรมศาสตรบัณฑิต  
สาขาวิชา                    วิศวกรรมอุตสาหกรรม  
ปีการศึกษา                    2543  
อาจารย์ผู้ควบคุมวิทยานิพนธ์

  
(อ.พลชัย โชติปราชญ์กุล)

  
(อ.อุดม จันทร์จรัสสุข)



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ชื่อปริญญาบัตร โปรแกรมการเชื่อมโยงระหว่างคอมพิวเตอร์และตัวควบคุมแบบตรรก  
โดย นักศึกษาชั้นปีที่4 ภาควิชาวิศวกรรมอุตสาหกรรม คณะ วิศวกรรมศาสตร์  
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง  
ชื่อนักศึกษา นาย วุฒิพล ก่อพัฒน์วงศ์  
รหัสประจำตัว 40010757  
ชื่ออาจารย์ที่ปรึกษาอาจารย์ อุดม จันทร์จรีสสุข  
อาจารย์ พลชัย โชติปราชญ์กุล

### บทคัดย่อ

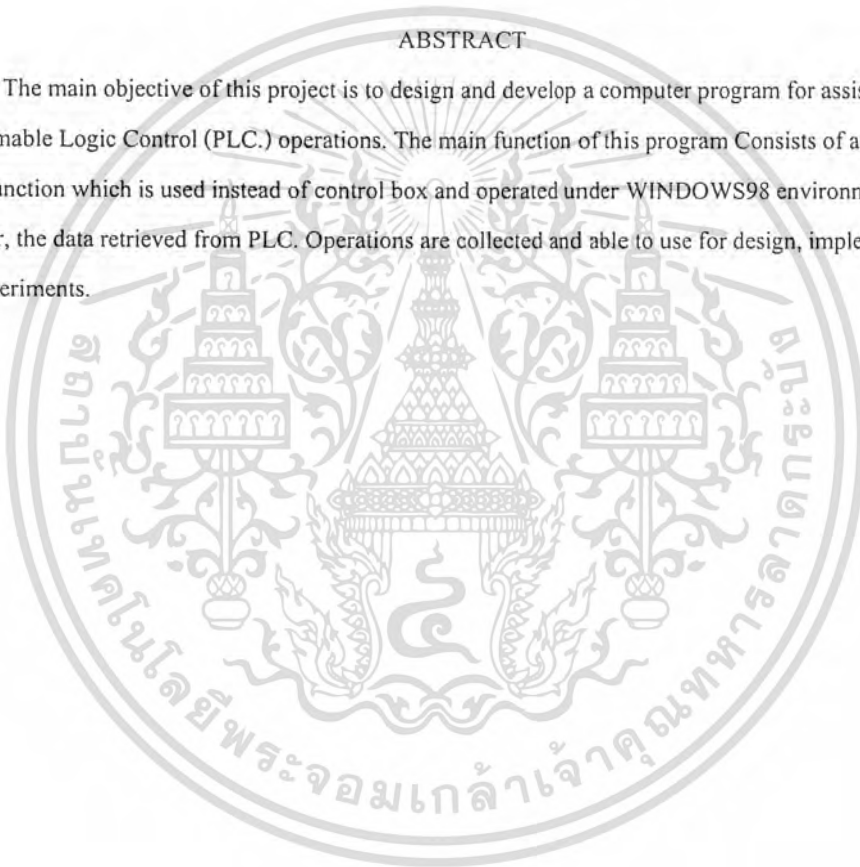
ปริญญาบัตรฉบับนี้เป็นการเขียน โปรแกรมการสั่งงานและแสดงสถานะของตัวควบคุมแบบตรรก (Programmable Logic Controller ; PLC.) ด้วยคอมพิวเตอร์โดยใช้โปรแกรมภาษา Delphi5 ในการพัฒนาเพื่อเป็นแนวทางในการเรียนรู้ทางการเขียน โปรแกรมระหว่าง Delphi5และตัวควบคุมแบบตรรกโดยโปรแกรมจะสามารถควบคุมการทำงานของควบตัวควบคุมแบบตรรกแทนตัวควบคุมสามารถแสดงผลแบบทันเหตุการณ์ (Real time) และสามารถจัดเก็บข้อมูลการทำงานเป็นเวลา,จำนวนครั้งไว้ใน Data memory และข้อมูลดิบเพื่อให้โปรแกรมจัดเก็บข้อมูล,พิมพ์ข้อมูล และนำข้อมูล ไปวิเคราะห์ในกระบวนการต่อเนื่องต่อไป ข้าพเจ้าหวังว่า ปริญญาบัตรฉบับนี้จะเป็นประโยชน์ต่อนักพัฒนาโปรแกรม และ ผู้สนใจ ค่ะไปในอนาคต

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Thesis Title	Computer – PLC. Interfacing Program
Student	Mr. Wutipol Korputanawong
Level of study	Bachelor in Industrial Engineer King Mongkut's Institute of Technology Ladkabang
Academic Year	2000
Advisor	Mr. Udom Janjarutsuk Mr. Pholchai Chotpraynakun

#### ABSTRACT

The main objective of this project is to design and develop a computer program for assisting Programmable Logic Control (PLC.) operations. The main function of this program Consists of an ON-LINE control function which is used instead of control box and operated under WINDOWS98 environment. Moreover, the data retrieved from PLC. Operations are collected and able to use for design, implementations of PLC. experiments.



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## กิตติกรรมประกาศ

การจัดทำปฏิญานិพนธ์ฉบับนี้สำเร็จลงได้ด้วยดีโดยได้รับความอนุเคราะห์และช่วยเหลือจากบุคคล  
หลายๆท่าน ข้าพเจ้าใคร่ขอนำนามของท่านเหล่านั้นมาปรากฏไว้ ณ. ที่นี้ เพื่อเป็นการขอบพระคุณและระลึกถึง

1. อาจารย์พลชัย โชติปรายนกุล อาจารย์ที่ปรึกษาโครงการ : ให้คำปรึกษาในการดำเนินโครงการพร้อม  
คำชี้แนะ และให้การเอื้อเฟื้ออุปกรณ์ ตัวควบคุมแบบตรรก( Programmable Logic Controller ; PLC.)
2. อาจารย์อุดม จันทร์จรัสสุข อาจารย์ที่ปรึกษาโครงการ : ให้คำปรึกษาด้านโปรแกรม และ  
ให้คำแนะนำในการค้นคว้าโครงการด้านต่างๆ
3. อาจารย์ทวีพล ชื้อสัตย์ อาจารย์ภาควิชาวิศวกรรมการวัดคุม : กรุณาให้คำปรึกษาและคำชี้แนะใน  
การค้นคว้าเป็นอย่างดีพร้อมทั้งเอื้อเฟื้ออุปกรณ์ PLC เพื่อการค้นคว้า
4. ผศ.พรศักดิ์ อรรถวานิช หัวหน้าภาควิชาวิศวกรรมอุตสาหกรรม : ให้คำแนะนำและให้ความ  
อนุเคราะห์ในการจัดหางบซื้ออุปกรณ์ PLC. เพื่อความสะดวกในการทำโครงการ
5. อาจารย์สรพลสิทธิ์ ลีมนรัตน์ อาจารย์ภาควิชาวิศวกรรมอุตสาหกรรม : ให้คำปรึกษาและช่วยเหลือดูแล  
ด้านการเรียบเรียงปฏิญานิพนธ์
6. อาจารย์เอกพจน์ ดันตราภิวัฒน์ อาจารย์ภาควิชาวิศวกรรมอุตสาหกรรม : ให้ความเอื้อเฟื้ออุปกรณ์และ  
เครื่องมือต่างๆ ในการทำโครงการ
7. คุณ Erik Salaj โปรแกรมเมอร์ ผู้สร้าง Comport ให้ความอนุเคราะห์ด้าน โปรแกรม Comport และ คำ  
ปรึกษาด้าน โปรแกรม

ตลอดทั้งอาจารย์ทุกท่านที่ได้ประสิทธิ์ประสาทวิชาแก่ข้าพเจ้า เพื่อนๆ ภาควิชาวิศวกรรมอุตสาหกรรม  
คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยี พระจอมเกล้าเจ้าคุณทหารลาดกระบังทุกท่านที่มีน้ำใจช่วยเหลือ  
ข้าพเจ้า และ ขอบขอบพระคุณคุณพ่อ คุณแม่ ญาติพี่น้อง ที่เป็นกำลังใจให้แก่ข้าพเจ้า และเอาใจใส่ด้วยดีตลอดมา

ผู้จัดทำ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

# สารบัญ

บทคัดย่อภาษาไทย	I
บทคัดย่อภาษาอังกฤษ	II
กิตติกรรมประกาศ	III
สารบัญ	IV
สารบัญตาราง	V
สารบัญรูป	VI
บทที่1 บทนำ	1
1.1 ความเป็นมา	1
1.2 จุดมุ่งหมายและวัตถุประสงค์	1
1.3 ขอบเขตของโครงการ	1
1.4 ข้อจำกัดของการศึกษา	1
บทที่2 ทฤษฎีที่เกี่ยวข้อง	2
2.1 ระบบอัตโนมัติ	2
2.2 PLC	3
2.3 เลขฐาน	6
2.4 วงจรตรรก	8
2.5 การสื่อสารข้อมูล	10
2.6 สายสัญญาณ(RS-232)	12
2.7 โพรโทคอลHost-Link	13
2.8 โปรแกรมภาษาDelphi5	16
บทที่3 วิธีการดำเนินการ	23
3.3 แผนการดำเนินการ	23
3.4 ขั้นตอนการสร้างโปรแกรมแบ่งได้ดังนี้	23
3.5 แผนการทดสอบโปรแกรม	40
บทที่4 ผลการดำเนินงาน	41
4.1 ตารางเวลาดำเนินการ	41
4.2 ผลการทดสอบโปรแกรม	41
4.3 ปัญหาและวิธีแก้ไข	47
บทที่5 สรุปและอภิปราย	48
5.1 สรุปผลการทำงาน	48
5.2 อภิปราย	48
ภาคผนวก	
เอกสารอ้างอิง	

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## สารบัญตาราง

ตารางที่ 2.1 ตารางแสดงพื้นที่หน่วยความจำของ PLC.	5
ตารางที่ 2.2 ตารางเปรียบเทียบค่าของเลขฐาน	7
ตารางที่ 2.3 ตารางเปรียบเทียบหน้าที่ในการรับและส่งข้อมูล	11
ตารางที่ 2.4 ตารางแสดงรายละเอียดสัญญาณต่างๆของพอร์ต RS-232	12
ตารางที่ 2.5 ตารางแสดงวิธีการต่อสายสัญญาณ	13
ตารางที่ 2.6 แสดงรูปแบบProtocolมาตรฐานของ Omron	13
ตารางที่ 2.7 แสดงคำสั่งในส่วนของ Header	14
ตารางที่ 2.8 คำสั่งในการอ่านข้อมูล	14
ตารางที่ 2.9 คำสั่งในการเขียนข้อมูล	14
ตารางที่ 2.10 แสดงวิธีหาค่า FCS	15
ตารางที่ 2.11 ตารางแสดงรหัสของการรับและส่งข้อมูลของโปรโตคอล Host-Link	16



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## สารบัญรูป

รูปที่ 2.1 ระบบควบคุมแบบลูบเปิด	2
รูปที่ 2.2 ระบบควบคุมแบบลูบปิด	2
รูปที่ 2.3 โครงสร้างของ PLC.	3
รูปที่ 2.4 แสดง โครงสร้างของหน่วยประมวลผลกลาง	4
รูปที่ 2.5 ภาพแสดง โครงสร้างเลขฐาน 2	6
รูปที่ 2.6 ภาพแสดงการแปลงเลขฐาน 2	7
รูปที่ 2.7 ภาพแสดงการแปลงเลขฐาน 16	8
รูปที่ 2.8 ภาพแสดงการแปลงเลขฐาน 16 เป็นฐาน 2	8
รูปที่ 2.9 ภาพแสดงสัญลักษณ์ และตารางความจริงของ And Gate	8
รูปที่ 2.10 ภาพแสดงสัญลักษณ์ และตารางความจริงของ Or Gate	9
รูปที่ 2.11 ภาพแสดงสัญลักษณ์ และตารางความจริงของ Not Gate	9
รูปที่ 2.12 ภาพแสดงสัญลักษณ์ และตารางความจริงของ Nand Gate	10
รูปที่ 2.13 ภาพแสดงสัญลักษณ์ และตารางความจริงของ Nor Gate	10
รูปที่ 2.14 รูปภาพแสดงการรับ-ส่งข้อมูลทางคอมพิวเตอร์	11
รูปที่ 2.15 เว็บไซต์ของบริษัท Inprise	17
รูปที่ 2.16 ตัวอย่างแอปพลิเคชันที่สร้างจาก Delphi	17
รูปที่ 2.17 การใช้เครื่องมือในการสร้างแอปพลิเคชันของ Delphi	18
รูปที่ 2.18 ตัวอย่างแอปพลิเคชันฐานข้อมูล	18
รูปที่ 2.19 ตัวอย่างแอปพลิเคชันอินเทอร์เน็ต	19
รูปที่ 2.20 ผลการจัดอันดับเครื่องมือพัฒนาแอปพลิเคชันสำหรับ Windows	19
รูปที่ 2.21 ลดความยุ่งยากในการเขียน โปรแกรมใน Delphi 5.0	20
รูปที่ 2.22 การตรวจสอบข้อผิดพลาดและแก้ไข โปรแกรมใน Delphi 5.0	20
รูปที่ 2.23 การตรวจสอบข้อผิดพลาดและแก้ไข โปรแกรมใน Delphi 5.0	21
รูปที่ 2.24 ตัวอย่างแอปพลิเคชันอินเทอร์เน็ตในฝั่งเซิร์ฟเวอร์	21
รูปที่ 2.25 ตัวอย่างแอปพลิเคชันที่ใช้งานร่วมกับ MS Office	22
รูปที่ 3.1 รูปภาพแสดงแผนการสร้าง โปรแกรม	23
รูปที่ 3.2 รูปภาพแสดงภาพความสัมพันธ์ระหว่างอุปกรณ์กับ โปรแกรมที่ได้ออกแบบไว้	24
รูปที่ 3.3 แสดงความสัมพันธ์ระหว่าง โปรแกรมสื่อสารที่ ทำขึ้นมา	25
รูปที่ 3.4 รูปภาพแสดงการเขียน โปรแกรมสื่อสารผ่าน port และใช้โปร โทคอล Host Link ของ Omron	25
รูปที่ 3.5 แสดง โครงสร้างความสัมพันธ์ระหว่าง โปรแกรม	26
รูปที่ 3.6 รูปภาพแสดง โปรแกรม คิดตั้ง	26
รูปที่ 3.7 รูปภาพแสดงการสร้าง Database โดยผ่าน โปรแกรม Database Desktop	27
รูปที่ 3.8 รูปภาพแสดงการสร้าง Database ขึ้นตอนการกำหนด Aliases ใน BDE.Administrator	28

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้ในเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้เผยแพร่ประโยชน์ในการค้า

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 3.9	รูปภาพแสดงการเขียนโปรแกรมโปรแกรมหลักเบื้องต้น	28
รูปที่ 3.10	แสดงโครงสร้างความสัมพันธ์ระหว่างโปรแกรม LSS กับ PLC.	29
รูปที่ 3.11	รูปภาพแสดงโปรแกรม LSS ของ Omron	29
รูปที่ 3.12	รูปภาพแสดงการเขียนโปรแกรมในรูปแบบแผนภาพเชิงเส้น (Ladder Diagram)	30
รูปที่ 3.13	รูปภาพแสดงการเขียนโปรแกรมในรูปแบบเชิงตัวเลข(Numerical Language)	30
รูปที่ 3.14	รูปภาพแสดงFunction ในโปรแกรม LSS	31
รูปที่ 3.15	รูปภาพแสดงการปรับสถานะ การเขียน,อ่าน,เก็บข้อมูล	31
รูปที่ 3.16	รูปภาพแสดงการเลือกสัญลักษณ์ต่างๆ	32
รูปที่ 3.17	รูปภาพแสดงการจบโปรแกรม LSS	32
รูปที่ 3.18	รูปภาพแสดงการเก็บโปรแกรมที่ได้เขียนไว้ใน LSS	33
รูปที่ 3.19	รูปภาพแสดงขั้นตอนการเลือกชนิดการติดต่อกับ โปรแกรม LSS	33
รูปที่ 3.20	รูปภาพแสดงรายละเอียดวิธีการขนถ่ายโปรแกรม	34
รูปที่ 3.21	รูปภาพแสดงรายละเอียดวิธีการขนถ่ายโปรแกรม.	34
รูปที่ 3.22	แสดงความสัมพันธ์ระหว่างโปรแกรมที่ทำขึ้น	35
รูปที่ 3.23	รูปภาพแสดงเส้นทางการเขียนMenu	35
รูปที่ 3.24	รูปภาพแสดงรูปแบบของHelp	36
รูปที่ 3.25	รูปภาพการกำหนดค่าคงที่การสื่อสาร	36
รูปที่ 3.26	รูปภาพแสดง ตำแหน่งของ DataMemory & PLC Command	37
รูปที่ 3.27	รูปภาพแสดงการเตือนว่าสายสัญญาณขาด	37
รูปที่ 3.28	รูปภาพแสดงการเตือนก่อนการกำหนดค่าคงที่การส่ง	38
รูปที่ 3.29	รูปภาพแสดงผลของBit Command กับ DataMemory	38
รูปที่ 3.30	รูปภาพแสดงหน้าต่างการพิมพ์	39
รูปที่ 3.31	รูปภาพแสดงการ Save & Load ผ่านหน้าต่างการพิมพ์	39
รูปที่ 4.1	รูปภาพแสดงการเปลี่ยนแปลงที่เกิดของหน้าคอนเทกต์	41
รูปที่ 4.2	รูปภาพแสดงการสั่งงานและการอ่านข้อมูล	42
รูปที่ 4.3	รูปภาพแสดงการออกแบบ Ladder Diagram	42
รูปที่ 4.4	รูปภาพแสดงการออกแบบ Interface โดยโปรแกรมติดตั้ง	43
รูปที่ 4.5	รูปภาพแสดงการออกแบบ Interfaceเสร็จและสร้างฐานข้อมูล	43
รูปที่ 4.6	รูปภาพแสดงการ โปรแกรม LSS.	44
รูปที่ 4.7	รูปภาพแสดงการถ่ายโปรแกรมสู่PLC.	44
รูปที่ 4.8	รูปภาพแสดงการเปลี่ยนแปลงของ โปรแกรมหลักตามลำดับขั้นตอนของ Ladder Diagram	45
รูปที่ 4.9	รูปภาพแสดงผลการเปลี่ยนแปลงของข้อมูล	46
รูปที่ 4.10	รูปภาพแสดงการพิมพ์รายงาน	46
รูปที่ 4.11	รูปภาพแสดงการเก็บรายงานในรูปแบบเพิ่มตัวอักษร	47

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

# บทที่ 1

## บทนำ

### 1.1 ความเป็นมา

ในปัจจุบันตัวควบคุมแบบตรรกะ (Programmable Logic Controller ; PLC.) ได้มีบทบาท และ ความสำคัญเพิ่มขึ้นอย่างมาก ในวงการอุตสาหกรรมโดยจะสามารถสังเกต ได้จาก มีการใช้ PLC กันอย่างแพร่หลาย โดย PLC จะทำหน้าที่เป็นตัวควบคุม (CONTROLLER) ได้ภายในเครื่องจักรที่มีการทำงานแบบอัตโนมัติ (Automation) และ สามารถที่หลากหลาย และง่ายต่อการใช้งาน แต่ในขณะเดียวกันปัจจุบันนี้โปรแกรมที่สามารถใช้ได้กับ PLC มีราคาสูงเป็นการเพิ่มต้นทุนให้กับโรงงาน และประกอบกับ โปรแกรม ที่จะใช้กับ PLC ยังไม่เป็นที่แพร่หลายในกลุ่มโรงงานอุตสาหกรรม การศึกษาและจัดทำโปรแกรมคอมพิวเตอร์ควบคุม PLC ขึ้นเพื่อใช้เป็นประโยชน์ ในการควบคุมเครื่องจักรสำหรับเปิดปิดและแสดงสถานะ การทำงานในลักษณะทันเหตุการณ์ (Real-Time) รวมทั้งการตั้งงานในลักษณะของชุดคำสั่งให้บรรทัดข้อมูลการทำงานสำหรับการวิเคราะห์ปัญหาด้านเวลาได้ต่อไปในอนาคต

### 1.2 จุดมุ่งหมายและวัตถุประสงค์

1. การศึกษากระบวนการทำงานด้านระบบอัตโนมัติ
2. การศึกษาการใช้งาน PLC.
3. การศึกษากระบวนการเขียน โปรแกรมภาษา Delphi สำหรับควบคุม PLC.
4. ทำการสร้าง โปรแกรมที่มีความสามารถควบคุม และแสดงสถานะของ PLC.
5. จำลองรูปแบบการทำงานของกระบวนการผลิต

### 1.3 ขอบเขตของโครงการ

1. โปรแกรมการควบคุม PLC ผ่าน Host-Link ของ OMRON
2. ภาษาในการเขียน โปรแกรมใช้ Delphi
3. โปรแกรมสามารถควบคุม, แสดงสถานะ และบันทึกข้อมูลการทำงานได้

### 1.4 ข้อจำกัดของการศึกษา

1. ความสามารถของโปรแกรมไม่สามารถใช้ได้กับ PLC. นอกกลุ่ม OMRON ได้
2. การสื่อสารผ่านมาตรฐาน RS-232
3. รูปแบบการทำงานของโปรแกรมใช้การจำลองรูปแบบการผลิตและพร้อมใช้งานกับตัวเครื่องจักร และระบบจับสัญญาณจริงได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

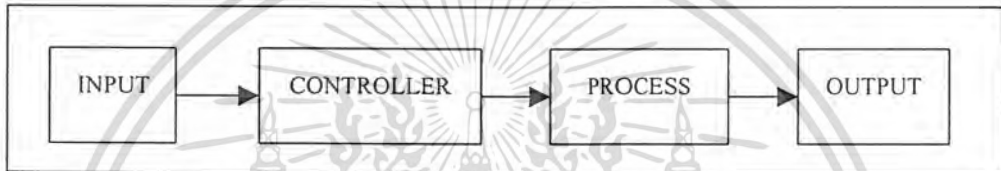
## บทที่ 2 ทฤษฎีที่เกี่ยวข้อง

### 2.1 ระบบอัตโนมัติ

ระบบอัตโนมัติคือระบบการทำงานของเครื่องจักรหรือระบบที่ทำงานได้ต่อเนื่อง ด้วยตนเองเมื่อเริ่มให้สัญญาณเริ่มต้น โดยจุดประสงค์ทั่วไปของระบบอัตโนมัติคือการ คงประสิทธิภาพ ควบคุมอัตราการการผลิตให้คงที่ผลผลิต ลดต้นทุน การควบคุม และการวางแผนการผลิตได้ง่าย

#### 2.1.1 ประเภทของการควบคุมแบ่งเป็น 2 ระบบคือ

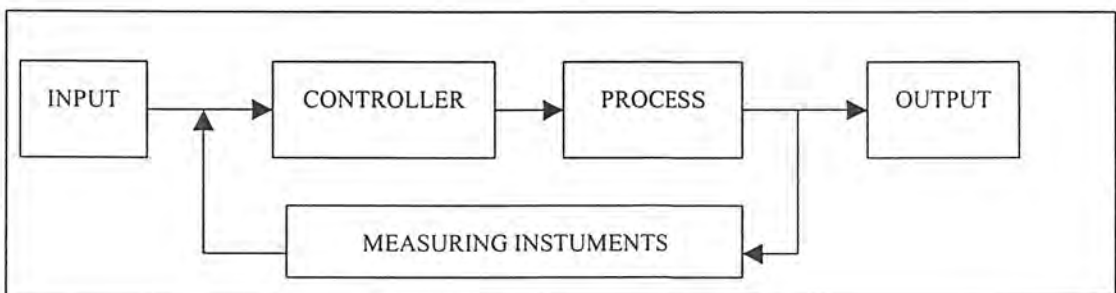
ระบบเปิด



รูปที่ 2.1 ระบบควบคุมแบบลูบเปิด

- ไม่มีการนำเอาที่พหุมาเปรียบเทียบกับอินพุท
- เอาท์พุท ไม่มีผลโดยตรงต่อการควบคุม
- ไม่สามารถควบคุมเอาท์พุท ให้เป็นไปตามต้องการได้ตลอดเวลา หรือเมื่อมีการเปลี่ยนแปลงเกิดขึ้น
- ไม่สามารถจัดผลของการรบกวนได้
- ความเที่ยงตรงของเอาท์พุทขึ้นอยู่กับอุปกรณ์ที่ใช้ และการปรับเทียบ
- ใช้งานง่าย ไม่ต้องกังวลเรื่องเสถียรภาพเพราะจะทราบได้ทันที
- ระบบควบคุมแบบลูบเปิด จะใช้กับงานที่ไม่ต้องการความเที่ยงตรงสูง และ ใช้งานง่าย เช่น พัดลม เคาเริด เครื่องซักผ้า

ระบบปิด



รูปที่ 2.2 ระบบควบคุมแบบลูบปิด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้ภายในของหน่วยงานนี้ มิอนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- มีการนำเอาที่พุดมาเปรียบเทียบกับอินพุท
- สามารถควบคุมเอาที่พุด ให้เป็นไปตามต้องการได้ตลอดเวลา หรือเมื่อมีการเปลี่ยนแปลงเกิดขึ้น
- สามารถจัดผล หรือลดการรบกวนได้
- อุปกรณ์ที่ใช้ไม่จำเป็นต้องมีความเที่ยงตรงสูง
- ต้องระวังเรื่องเสถียรภาพอันเกิดจากผลของการป้อนกลับ
- ระบบควบคุมแบบลูปปิด จะใช้กับงานที่ต้องการความเที่ยงตรงสูง และใช้งานทั่วไปในโรงงาน เช่น การควบคุมอุณหภูมิห้อง การควบคุมกระบวนการทางอุตสาหกรรม เป็นต้น

## 2.2 Programmable Logic Controller(PLC.)

PLC เป็นอุปกรณ์ควบคุม ที่นำมาแทนการควบคุมที่ใช้รีเลย์ ทำให้สะดวกขึ้น เป็นระบบไฟฟ้าอิเล็กทรอนิกส์ และใช้การเขียน โปรแกรมแทนการเดินสายไฟฟ้า

PLC มีหน่วยอินพุตและเอาต์พุตแบบลอจิกและแบบอนาล็อกสามารถควบคุมเครื่องจักร โดยการเปิด ปิด วงจรไฟฟ้า และมีความน่าเชื่อถือสูง

PLC เป็นอุปกรณ์ที่ประกอบกันขึ้นเป็น โมดูล สามารถตรวจสอบด้วยตัวเองเพราะฉะนั้นเมื่อเสียหายก็สามารถเปลี่ยนเฉพาะ โมดูลที่เสีย

PLC. เป็นอุปกรณ์ทางไฟฟ้าอิเล็กทรอนิกส์ทำหน้าที่ควบคุมเครื่องจักรหรือกระบวนการผลิตโดยใช้โปรแกรมในหน่วยความจำกำหนดเงื่อนไขการควบคุมผ่านทางหน่วยอินพุตและเอาต์พุต ดังแสดงในรูปที่2.3



รูปที่2.3 โครงสร้างของPLC.

ข้อดีของการใช้งาน PLC.

1. แก้ไขได้ง่าย
2. ติดตั้งง่าย
3. ติดต่อกับระบบอื่นง่าย
4. ลดการเดินสายไฟควบคุม
5. เนื้อที่ติดตั้งน้อยกว่า
6. บำรุงรักษาง่าย และ ซ่อมแซมง่าย
7. มีประสิทธิภาพการทำงานสูงกว่า
8. ความสามารถของPLC

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## PLC สามารถควบคุมระบบได้ 3 รูปแบบ คือ

1. งานที่ทำตามระดับก่อนหลัง ได้แก่ การทำงานของระบบรีเลย์, การทำงานของไทเมอร์และคาน์เตอร์, การทำงานของ P.C.B card, การทำงานในระบบกึ่งอัตโนมัติ ระบบอัตโนมัติ หรืองานที่เป็นกระบวนการทำงานของเครื่องจักรกลต่างๆ
2. งานควบคุมสมัยใหม่ ได้แก่ การทำงานทางคณิตศาสตร์, การควบคุมแบบอนาล็อก, การควบคุม P.I., การควบคุมเซอร์โวมอเตอร์, การควบคุม Stepper-Motor, Information Handling
3. การควบคุมเกี่ยวกับงานอำนวยความสะดวก ได้แก่ งานสัญญาณเตือน, Fault Diagnostic and Monitoring, งานต่อร่วมกับคอมพิวเตอร์, Printer / ASCII interfacing, งานควบคุมอัตโนมัติในโรงงานอุตสาหกรรม

### 2.2.1 หน่วยอินพุตและเอาต์พุต

หน่วยอินพุตและเอาต์พุตทำหน้าที่ติดต่อกับ PLC กับ อุปกรณ์ภายนอก, เครื่องจักรหรือกระบวนการผลิตต่างๆ หน่วยอินพุต ทำหน้าที่รับคำสั่งสัญญาณในรูปแบบต่างๆจากภายนอก เช่น สวิตช์ หรืออุปกรณ์ตรวจจับต่างๆ หน่วยเอาต์พุต ทำหน้าที่ส่งสัญญาณต่างๆออกไปควบคุมเครื่องจักร หรืออุปกรณ์ภายนอก เช่น หลอดไฟ กระดิ่ง มอเตอร์ วาล์วควบคุม ระบายเชื้อของ CPU ซึ่งประกอบด้วยหน่วยประมวลผล หน่วยความจำและหน่วยจ่ายกำลังงาน แสดงดังรูป 2.4



รูปที่ 2.4 แสดง โครงสร้างของหน่วยประมวลผลกลาง

CPU เป็นส่วนสำคัญของ PLC. ทำหน้าที่ตัดสินใจ และควบคุมการทำงานทั้งหมดของ PLC. โดยรับคำสั่งจากสถานะต่างๆของเครื่องจักรผ่านทางหน่วยอินพุต ประมวลผลตาม โปรแกรมของผู้ใช้ที่เก็บไว้ในหน่วยความจำ และนำผลลัพธ์ที่ได้ส่งไปควบคุมเครื่องจักรทางหน่วยเอาต์พุต การทำงานของ PLC. ทั้งหมดเรียกว่า การสแกน หน่วยจ่ายกำลังงานมีหน้าที่จ่ายไฟฟ้าให้กับ CPU และหน่วยความจำทำงานปกติ

### 2.2.2 หน่วยประมวลผล

หน่วยประมวลผลทำหน้าที่ดูแลการทำงานทั้งหมดของ PLC. คือนำโปรแกรมผู้ใช้ มาปฏิบัติเพื่อควบคุมอุปกรณ์ภายนอกตามเงื่อนไขการควบคุมที่ผู้เขียน โปรแกรมต้องการควบคุมการติดต่อรับส่งข้อมูลระหว่าง CPU กับหน่วยอินพุต/เอาต์พุต และหน่วยอินพุต/เอาต์พุตกับอุปกรณ์ภายนอกติดต่อกับผู้ใช้และอุปกรณ์ร่วม ตรวจสอบสภาพการทำงาน of PLC. โดยมีโปรแกรมบริหารระบบ เป็นผู้ควบคุมอีกทีหนึ่ง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 2.2.3 หน่วยความจำ

หน่วยความจำทำหน้าที่เก็บ โปรแกรมบริหารระบบและข้อมูลต่างๆที่ PLC ใช้ในการประมวลผล แบ่งออกเป็น 2 ส่วนคือ

1. หน่วยความจำระบบ เก็บ โปรแกรมบริหารระบบและข้อมูลของระบบ
2. หน่วยความจำผู้ใช้ เก็บ โปรแกรมผู้ใช้ ข้อมูลหน่วยอินพุต/เอาต์พุต

ตารางที่ 2.1 ตารางแสดงพื้นที่หน่วยความจำของ PLC. [5; 58]

พื้นที่ของข้อมูล		เวิร์ด	บิต	หน้าที่
IR area <sup>1</sup>	Input area	IR 000 to IR 009 (10 words)	IR 00000 to IR 00915 (160 bits)	บิตเหล่านี้ใช้สำหรับคอกับ อินพุต/ เอาต์พุตภายนอก
	Output area	IR 010 to IR 019 (10 words)	IR 01000 to IR 01915 (160 bits)	
	Work area	IR 200 to IR 231 (32 words)	IR 20000 to IR 23115 (512 bits)	เวิร์กบิตเป็นบิตอิสระที่ใช้ในโปรแกรม
SR area		SR 232 to SR 255 (24 words)	SR 23200 to SR 25507 (384 bits)	บิตเหล่านี้ใช้สำหรับหน้าที่เฉพาะ เช่น flags และการควบคุมบิต
TR area		---	TR 0 to TR 7 (8 bits)	บิตเหล่านี้ใช้สำหรับเก็บสถานะ "ON" "OFF" ชั่วคราวที่สาขาของโปรแกรม
HR area <sup>2</sup>		HR 00 to HR 19 (20 words)	HR 0000 to HR 1915 (320 bits)	บิตเหล่านี้ใช้สำหรับเก็บข้อมูล และ รักษาสถานะ "ON" - "OFF" เมื่อ ไม่มีกระแสไฟฟ้า
AR area <sup>2</sup>		AR 00 to AR 15 (16 words)	AR 0000 to AR 1515 (256 bits)	บิตเหล่านี้ใช้สำหรับหน้าที่เฉพาะ เช่น flags และการควบคุมบิต
LR area <sup>1</sup>		LR 00 to LR 15 (16 words)	LR 0000 to LR 1515 (256 bits)	ใช้สำหรับเชื่อมต่อข้อมูล 1:1 กับ PLC ตัวอื่นๆ
Timer/Counter area <sup>2</sup>		TC 000 to TC 127 (timer/counter numbers) <sup>1</sup>		หมายเลขเดียวกันใช้สำหรับไทมเมอร์ และเคาน์เตอร์
DM area	Read/write <sup>2</sup>	DM 0000 to DM 0999 DM 1022 to DM 1023 (1,002 words)	---	ข้อมูลของพื้นที่ DM สามารถเข้าถึง หน่วยของเวิร์ดเท่านั้น และค่าของ เวิร์ดยังคงมีอยู่เมื่อไม่มีกระแสไฟฟ้า
	Error log <sup>4</sup>	DM 1000 to DM 1021 (22 words)	---	ใช้สำหรับเก็บเวลาที่เกิดขึ้นและค่าคิด พลาดที่เกิดขึ้น เวิร์ดเหล่านี้สามารถ ใช้เป็น read/write DM เมื่อ log function ผิดพลาดจะไม่ใช้
	Read-only <sup>4</sup>	DM 6144 to DM 6599 (456 words)	---	ไม่สามารถเขียนมากเกินไปจาก โปรแกรม
	PC Setup <sup>4</sup>	DM 6600 to DM 6655 (56 words)	---	ใช้สำหรับเก็บพารามิเตอร์ต่าง ๆ ที่ ควบคุมการทำงานของ PLC

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## 2.2.4 หน่วยป้อนโปรแกรมให้PLC.

หน่วยป้อนโปรแกรมทำหน้าที่ป้อนโปรแกรมลงหน่วยความจำซึ่งโปรแกรมบริหารระบบจะนำไปประมวลผลเพื่อควบคุมอุปกรณ์ภายนอกอีกทีหนึ่ง ผู้ผลิต PLC พยายามให้ความสะดวกกับผู้ใช้โดยการจัดหาหน่วยป้อนโปรแกรมที่ใช้งานง่ายและมีความสามารถพิเศษต่างๆเพิ่มขึ้น เช่นการตรวจสอบแก้ไขโปรแกรมในหน่วยความจำขณะที่ PLC กำลังปฏิบัติงานการติดต่อกับระบบโครงข่ายและอุปกรณ์ร่วมอื่นๆ หน่วยป้อนโปรแกรมของ PLC มี 5 ประเภท คือ

1. เครื่องป้อนโปรแกรมแบบจอภาพ
2. เครื่องป้อนโปรแกรมขนาดเล็ก
3. เครื่องบันทึกโปรแกรม
4. เครื่องบันทึกข้อมูลลงหน่วยความจำ
5. คอมพิวเตอร์

## 2.2.5 หลักการทำงานของPLC

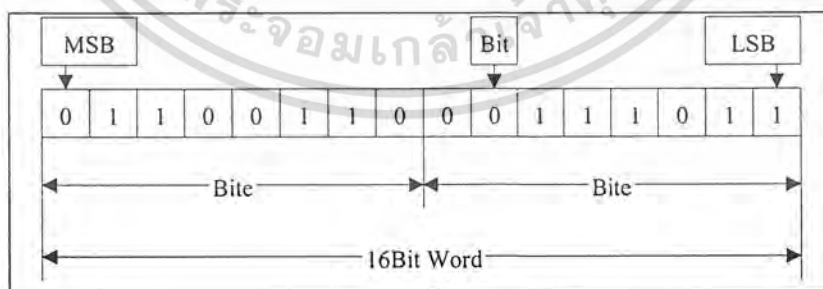
เมื่อมีสัญญาณ อินพุตเข้ามาถูกเก็บเป็นความจำไว้ในหน่วยความจำหลังจากนั้นแลคเคอร์ไดโอดก็จะสรุปผลรวมกับคอนแทคภายในว่าให้เป็นคอนแทคเปิดหรือปิดขึ้นอยู่กับผลการบันทึกของหน่วยความจำ

## 2.3 ระบบเลขฐาน

PLCใช้เลขฐานสองในการประมวลผลและใช้เลขฐานสิบติดต่อกับผู้ใช้ PLC ตัวเลขทุกระบบจะมีค่าเท่ากับเลขฐาน และค่าสูงสุดของเลขฐานนั้นจะมีค่าน้อยกว่าเลขฐานอยู่เสมอ

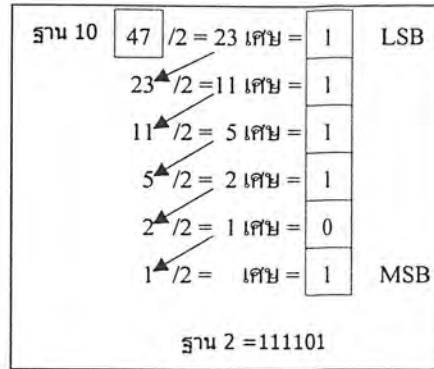
### 2.3.1 เลขฐานสอง

ไบนารี คือ เลขฐานสองที่มีค่าคือเป็น 0 และ 1 แต่ละคิขของเลขฐานสองเรียกว่า บิต (Bit) ใน PLC จะมีโพสเซเซอร์-เมโมรี่ ซึ่งประกอบด้วยจำนวน หนึ่งร้อยหรือหนึ่งพัน โลเกชั่น แต่ละ โลเกชั่น หรือรีจิสเตอร์อ้างถึง Word แต่ละWord จะเก็บ Data ในรูปของ ไบนารี จำนวนของ Bit ที่ให้ Word เก็บข้อมูล ได้นั้นจะขึ้นอยู่กับชนิดของ PLC ที่ใช้



รูปที่ 2.5 ภาพแสดง โครงสร้างเลขฐานสอง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.6 ภาพแสดงการแปลงเลขฐานสอง

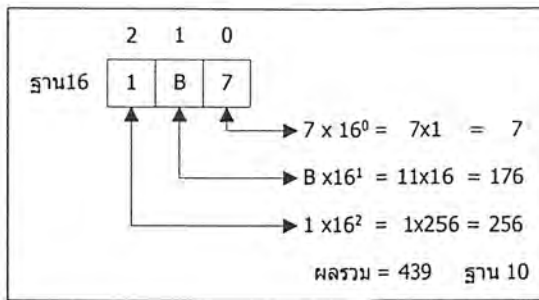
### 2.3.2 เลขฐานสิบหก

ตัวเลขจะเริ่มที่ตัวเลข 0 ถึง 15 แต่ 10-15 จะแทนด้วย A-F

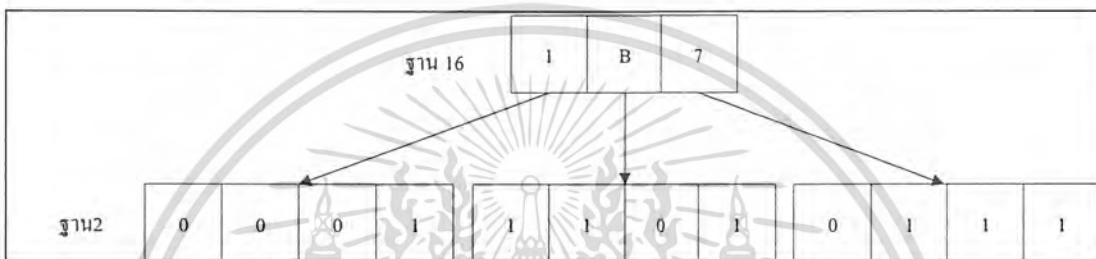
ตารางที่ 2.2 ตารางเปรียบเทียบค่าของเลขฐาน

ฐาน 16	ฐานสอง	ฐานสิบ
0	0000	0
1	0001	1
2	0010	2
3	0011	3
4	0100	4
5	0101	5
6	0110	6
7	0111	7
8	1000	8
9	1001	9
A	1010	10
B	1011	11
C	1100	12
D	1101	13
E	1110	14
F	1111	15

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.7 ภาพแสดงการแปลงเลขฐาน 16



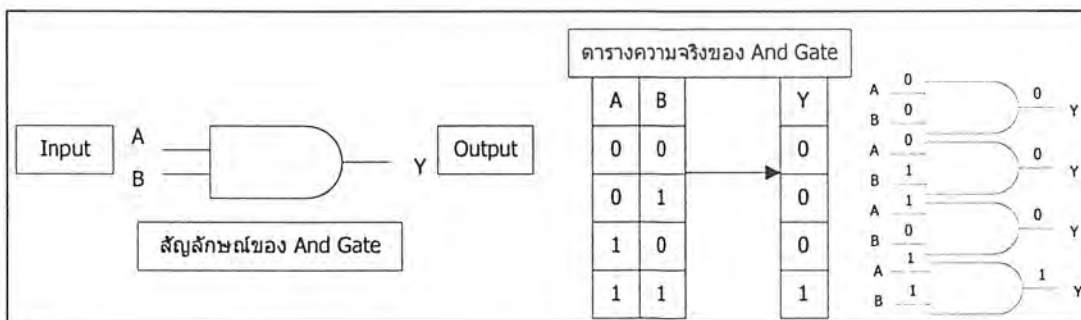
รูปที่ 2.8 ภาพแสดงการแปลงเลขฐาน 16 เป็นฐาน 2

2.4 วงจรตรรก

วงจรรวม หมายถึงวงจรไฟฟ้าที่ประกอบด้วยอุปกรณ์อิเล็กทรอนิกส์ หรือระบบปริเลขที่มีสัญญาณ 2 ระดับ วงจรรวมมีสองชนิด คือ แบบ บวก (จะใช้สัญญาณไฟระดับสูงแทน โลจิกเป็น 1 ระดับไฟต่ำแทน โลจิกเป็น 0)แบบลบ(จะใช้สัญญาณไฟระดับสูงแทน โลจิกเป็น 0 ระดับไฟต่ำแทน โลจิกเป็น 1) สภาวะทาง โลจิกคือ สภาวะเป็น 0 หรือ 1 แทนการทำงานของอุปกรณ์ที่เปลี่ยนแปลง 2 สถานะ

2.4.1 หลักการ AND Gate

AND Gate ทำให้เกิดสัญญาณเอาต์พุต ได้ก็ต่อเมื่ออินพุต A และ B มีค่าเป็น 1

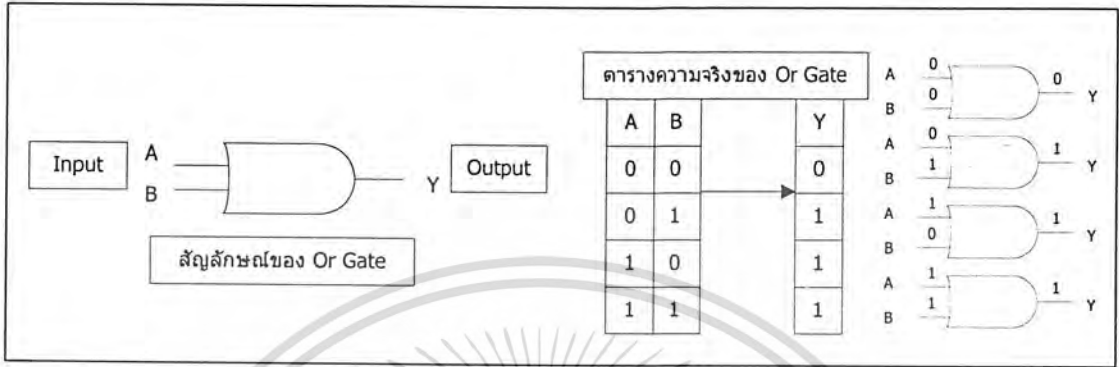


รูปที่ 2.9 ภาพแสดงสัญลักษณ์ และตารางความจริงของ And Gate

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 2.4.2 หลักการ OR Gate

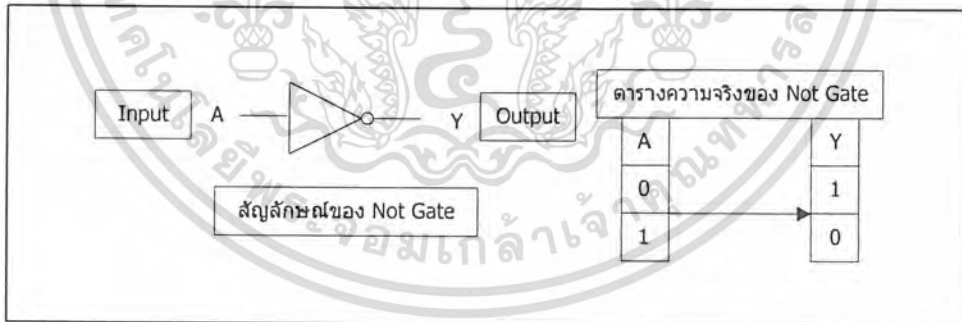
OR Gate สามารถมีอินพุตหลายตัวแต่มีเอาต์พุตเพียงตัวเดียว เอาต์พุตจะมีค่าเท่ากับ 1 ก็ต่อเมื่อ มีอินพุตตัวหนึ่งเป็น 1



รูปที่ 2.10 ภาพแสดงสัญลักษณ์ และตารางความจริงของ Or Gate

### 2.4.3 หลักการ NOT Gate

NOT Gate จะไม่เหมือน AND OR ตรงที่ NOTจะมีอินพุตเพียงตัวเดียว เอาต์พุตจะออกค่าตรงข้ามกับอินพุต

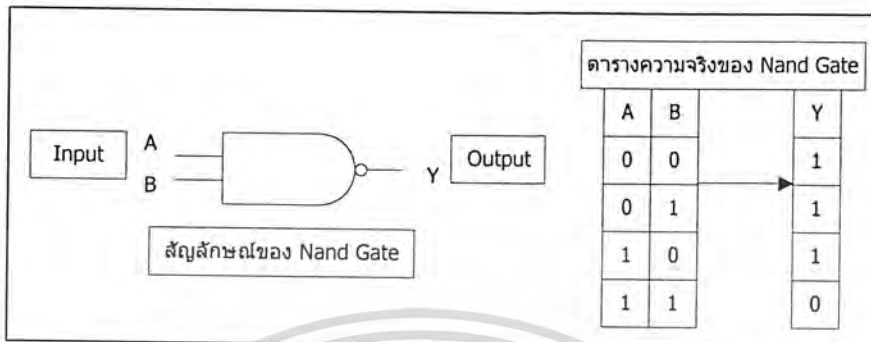


รูปที่ 2.11 ภาพแสดงสัญลักษณ์ และตารางความจริงของ Not Gate

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## 2.4.4 หลักการ NAND Gate

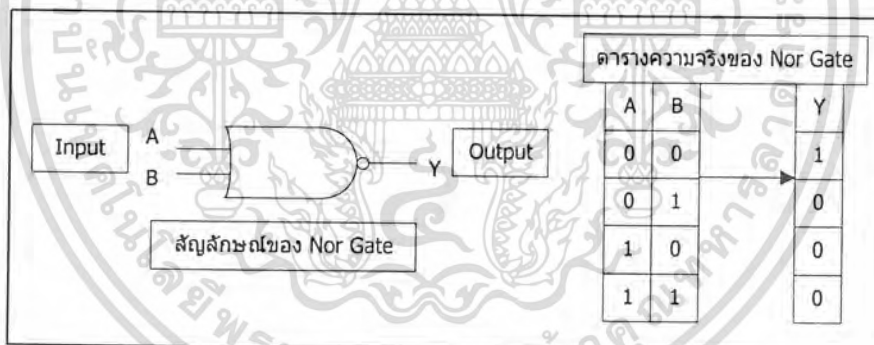
NAND Gate ทำงานตรงข้ามกับ AND Gate



รูปที่ 2.12 ภาพแสดงสัญลักษณ์ และตารางความจริงของ NAND Gate

## 2.4.5 หลักการ NOR Gate

NOR Gate หลักการทำงานตรงข้ามกับ OR Gate



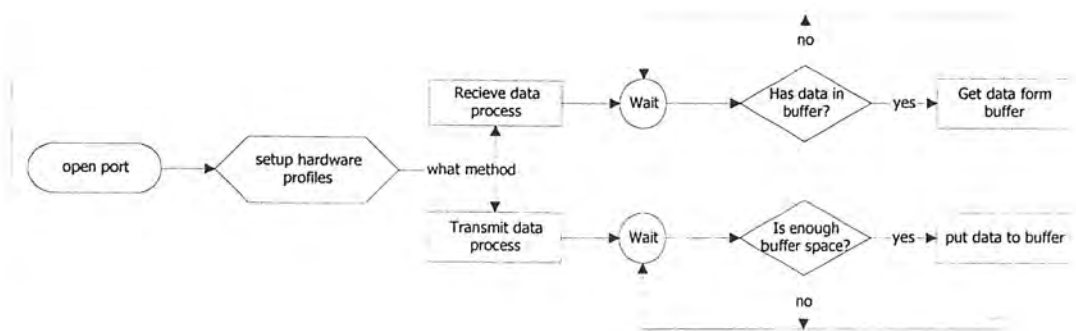
รูปที่ 2.13 ภาพแสดงสัญลักษณ์ และตารางความจริงของ Nor Gate

## 2.5 การสื่อสารข้อมูล

### 2.5.1 การรับ-ส่งข้อมูลทางคอมพิวเตอร์

ในการส่งข้อมูลทางคอมพิวเตอร์เราจะต้องทราบว่าทางด้าน PLC จะรับ-ส่งข้อมูลแบบใดและคอมพิวเตอร์จะรับ-ส่งข้อมูลแบบใด เนื่องจากในการรับส่งใช้มาตรฐานของการรับส่งข้อมูลแบบอนุกรม (RS-232) โดยจะจำแนกตามหน้าที่ของการรับและการส่ง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.14 รูปภาพแสดงการรับ-ส่งข้อมูลทางคอมพิวเตอร์

ตารางที่ 2.3 ตารางเปรียบเทียบหน้าที่ในการรับและส่งข้อมูล

เครื่องค้นทางทำหน้าที่รับข้อมูล	เครื่องปลายทางทำหน้าที่ส่งข้อมูล
1.เปิดพอร์ตสื่อสาร	1.เปิดพอร์ตสื่อสาร
2.กำหนดค่าเบื้องต้น	2.กำหนดค่าเบื้องต้น
3.ตรวจสอบว่ามี Input Buffer (RX)	3.ตรวจสอบว่ามี Output Buffer(TX)เพียงพอที่จะส่ง
4.ถ้ามี ทำการเคลื่อนย้ายข้อมูลออกจาก Buffer	ข้อมูลหรือไม่
ดำเนินการประมวลผลและรอรับข้อมูลชุดต่อไป	4.ถ้ามีดำเนินการส่งข้อมูล

### 2.5.2 ค่าคงที่ที่ใช้สื่อสาร

รูปแบบที่ใช้สื่อสารมาจากมาตรฐานของ PLC OMRON คือ

- Parity เป็นแบบ Even
- Stop bit เป็นแบบ 1
- Data bit เป็นแบบ 7
- Baud Rate เป็นแบบ 9600

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## 2.6 สายสัญญาณ(RS-232)

มาตรฐานของการรับส่งข้อมูลแบบอนุกรม (RS-232) นี้ ได้มีการกำหนดขึ้นมาเพื่อให้ คอมพิวเตอร์ ต่าง ๆ หือกัน หรืออุปกรณ์ต่อพ่วงแต่ละชนิดรับส่งข้อมูลกันได้เมื่อทำตามมาตรฐานนี้ โดยไม่สนใจอุปกรณ์หรือคอมพิวเตอร์นั้นจะผลิตมาจากที่ใด โดยมีการกำหนดรายละเอียดในการรับส่งข้อมูลเช่นข้อต่อที่ใช้เป็นแบบใดมีสัญญาณที่ใช้กี่เส้นแต่ละสัญญาณทำหน้าที่อะไรและใช้ระดับแรงดันไฟเท่าไรความเร็วในการรับส่งเท่าไรใช้ข้อมูลกี่bit ในการรับส่ง ฯลฯ อุปกรณ์ที่ทำตามมาตรฐานนี้จะติดต่อได้อย่างไม่มีปัญหาโดยมีรายละเอียดดังตาราง 2.4 และตาราง 2.5

ตารางที่ 2.4 ตารางแสดงรายละเอียดสัญญาณต่างๆของพอร์ต RS-232

1.ขาต่างๆของ DB25	2.ขาต่างๆของ DB9
1.1 Protective Ground	2.1 Carrier Detect
1.2 Transmitted Data	2.2 Receive Data
1.3 Receive Data	2.3 Transmit Data
1.4 Request to Sent	2.4 Data Terminal Ready
1.5 Clear to Sent	2.5 Signal Ground
1.6 Data set Ready	2.6 Data Set Ready
1.7 Signal Ground	2.7 Request to Send
1.8 Data Carry Detect	2.8 Clear to Send
1.9 Reserve	2.9 Ring Indicator
1.10 Reserve	
1.11 Unassigned	
1.12 Secondary Data Carrier Detect	
1.13 Secondary Clear to Sent	
1.14 Secondary Transmitted Data	
1.15 Transmit Clock	
1.16 Secondary Receive Data	
1.17 Receiver Clock	
1.18 Unassigned	
1.19 Secondary Request to Sent	
1.20 Data Terminal Ready	
1.21 Signal Quality Detector	
1.22 Ring Indicator	
1.23 Data Rate Select	
1.24 External Clock	
1.25 Unassigned	

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 2.5 ตารางแสดงวิธีการต่อสายสัญญาณ

DB25/DB25	DB25/DB25	DB9/DB25	DB9/DB25	DB9/DB9	DB9/DB9
2:3	2:3	8:1	2:3	1:1	2:3
3:2	3:2	3:2	3:2	2:3	3:2
4:5	7:7	2:3	7:5	3:2	5:5
5:4		20:4		4:4	
6:20		7:5		5:5	
7:7		6:6		6:6	
20:6		4:7		7:7	
		5:8		8:8	
		22:9		9:9	

\*หมายเหตุ การต่อสายสัญญาณของมาตรฐาน Host Link จะมีลักษณะดังนี้ com-plc5-9,3-3,2-2

## 2.7 โพรโทคอลHost-Link

Protocolที่ใช้ในการสื่อสารใช้ Host-Link Protocol ของ OMRON โดยมีรายละเอียดดังนี้  
 บล็อกคำสั่งในการอ่าน-เขียนข้อมูลแสดงดังตาราง2.6

ตารางที่ 2.6 แสดงรูปแบบProtocolมาตรฐานของ Omron

@	XX	XX	XX...	FCS	*(CR)
	หมายเลขเครื่อง	Header	Text		

### 2.7.1 หมายเลขเครื่อง

ในการเชื่อมต่อที่เป็นโครงข่ายแบบหลายจุดจะมีเครื่องควบคุมที่อยู่มากกว่าเครื่องแต่ละเครื่องจะมี  
 หมายเลขเครื่องที่ต่างกันไป เช่น @05 คือ หมายเลขที่ 05

### 2.7.2 ส่วน Header

เป็นส่วนของคำสั่งหลักที่จะกำหนดว่าต้องการกระทำกับข้อมูลส่วนใด เช่น ถ้าต้องการอ่านข้อมูลอินพุท  
 จากเครื่องควบคุมจะใช้ตัวอักษรพิมพ์ใหญ่ว่า "RR" ถ้าต้องการเขียนข้อมูลให้กับเครื่องควบคุมจะใช้ว่า "WR"  
 เป็นต้น คำสั่งของ Headerแสดงในตาราง 2.7

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
 ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### ตารางที่ 2.7 แสดงคำสั่งในส่วนของ Header

หน่วยความจำ	คำสั่งอ่าน	คำสั่งเขียน
IR	RR	WR
HR	RH	WH
LR	RL	WL
CNT/TIM	RC	WC
AR	RJ	WJ
DM	RD	WD

#### 2.7.3 ส่วน TEXT

เป็นส่วนของข้อมูล เช่น คำที่อ่าน ได้จากอินพุท หรือ คำที่ต้องเขียนลงในพื้นที่ต่างๆ โดยดูตัวอย่างจาก ตาราง 2.8 และ ตาราง 2.9

#### ตารางที่ 2.8 คำสั่งในการอ่านข้อมูล

@	00	RR	0000	0002	FCS	*(CR)
---	----	----	------	------	-----	-------

จากตารางที่ 2.8 เป็นการอ่านข้อมูลแบบ "IR" จากที่แอสซีหมายเลข 00 โดยเริ่มจาก word ที่ 0000 เป็นจำนวน 2 คำ หรือ IR 00 และ IR 01

#### ตารางที่ 2.9 คำสั่งในการเขียนข้อมูล

@	00	WR	0001	AF02	FCS	*(CR)
---	----	----	------	------	-----	-------

จากตารางที่ 2.9 เป็นการเขียนข้อมูลลงใน IR 01 ด้วยค่า AF02

#### 2.7.4 ส่วนของ FCS

เป็นส่วนของการควบคุมความผิดพลาดของข้อมูลซึ่งได้จากการคำนวณ การหา FCS จะทำการแปลงตัวอักษรทุกตัวก่อนหน้าเป็นเลขฐาน 16 สองตัว จากตารางรหัส ASCII นำเลขที่ได้มาเขียนเรียงกันตามแนวนิ่งแล้ว จับตัวหน้าของทุกตัวมาทำการ xor กันและจับตัวหลังของทุกตัวมาทำการ xor กันจากนั้นนำผลของตัวหน้ามาใส่เลข 3 ข้างหน้าแล้วนำผลของตัวหลังมาใส่เลข 3 ข้างหน้าจากนั้นแปลงกลับมาเป็นตัวอักษรแล้วจากนั้นจะได้ค่า FCS ที่ต้องการ

ตัวอย่างเช่น @10RH00310001XX\*(CR)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 2.10 แสดงวิธีหาค่า FCS

@	0100	0000
1	0011	0001
0	0011	0000
R	0101	0010
H	0100	1000
0	0011	0000
0	0011	0000
3	0011	0011
1	0011	0001
0	0011	0000
0	0011	0000
0	0011	0000
1	0011	0001
เมื่อ XOR กันแล้วได้ ผล	0101	1000
แปลงกลับเป็นตัว เลข โดยใส่ 0011 ข้างหน้า	0011 0101 5	0011 1000 8

\*หมายเหตุ ตารางรหัส ASCII สามารถดูได้จากภาคผนวก

### 2.7.5 ส่วนของ TERMINAL(\*)

เป็นส่วนปิดท้ายบอกให้ทราบว่าจบบล็อกและตามด้วย carrier return (CR)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## 2.7.6 ผลตอบสนองที่มาจาก PLC

ผลตอบสนองนั้นจะแบ่งเป็น 2 ลักษณะคือ คอบสนองที่ถูกต้องกับผลตอบสนองที่เป็นรหัสการผิดพลาด โดยรูปแบบคล้ายกับการส่งข้อมูลรหัสสามารถดูจากตารางที่ 2.11

ตารางที่ 2.11 ตารางแสดงรหัสของการรับและส่งข้อมูลของ โปรโตคอล Host-Link [7]

x16 <sup>1</sup>	x16 <sup>0</sup>	Description
0	0	Command completed normally.
0	1	Execution was not possible because the PC is in RUN mode. Change the PC mode.
0	2	Execution was not possible because the PC is in MONITOR mode. Change the PC mode.
0	3	Execution was not possible because PROM is mounted. Change the unit to RAM or EEPROM.
0	4	Address overflow (data overflow). Check the program.
0	B	Execution was not possible because the PC is in PROGRAM mode. Change the PC mode.
0	C	Execution was not possible because the PC is in DEBUG mode. Change the PC mode.
0	D	Execution was not possible because the Host Link Unit's keyswitch is set to LOCAL mode or because the command was sent to a C2000H CPU that was on standby. Change the mode or send the command to the active CPU.
1	0	Parity error.
1	1	Framing error (stop bit(s) not detected).
1	2	Overrun (the next command was received too soon).
1	3	FCS error (checksum error).
1	4	Command format error.
1	5	An incorrect data area designation was made for READ or WRITE.
1	6	Instruction not found.
1	8	Frame length error (maximum length exceeded).
1	9	Execution was not possible because of an unexecutable error clear, memory error, EEPROM write disabled, etc.
2	0	I/O table generation was not possible (unrecognized Remote I/O Unit, word overflow, duplicated word allocation).
2	1	An error occurred in the PC's CPU.
2	2	The specified Memory Unit does not exist.
2	3	The specified Memory Unit is write-protected.
A	0	Aborted due to parity error in transmit data.
A	1	Aborted due to framing error in transmit data.
A	2	Aborted due to overrun in transmit data.
A	3	Aborted due to FCS (checksum) error in transmit data.
A	4	Aborted due to format error in transmit data.
A	5	Aborted due to entry number data error in transmit data.
A	8	Aborted due to frame length error in transmit data.
B	0	Not executable because the program area is not 16 kbytes.
Other		Remove any possible causes of noise and resend the command.

## 2.8 โปรแกรมภาษา Delphi5 [1; 22-27]

Delphi5 ได้ชื่อว่าเป็นเครื่องมือที่ใช้ในการสร้างแอปพลิเคชันเพื่อใช้ในระบบปฏิบัติการ ตระกูล windows ซึ่งได้รับการปรับปรุงมาอย่างต่อเนื่อง จนมาถึงเวอร์ชัน 5.0 ซึ่งโดดเด่นในเรื่องความเร็วในการทำงาน และให้ผลลัพธ์เป็นแอปพลิเคชันที่มีประสิทธิภาพการทำงานที่เยี่ยมยอดจึงได้รับการยอมรับจากนักพัฒนาแอปพลิเคชันทั่วโลก

### 2.8.1 เครื่องมือสนับสนุนของโปรแกรมภาษา Delphi

Delphi เป็นเครื่องมือสำหรับสร้างแอปพลิเคชันสำหรับรันบน Windows 95/98/2000 ที่ผลิต โดยบริษัท

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Inprise ชื่อเดิมคือ Borland ซึ่งเป็นบริษัทที่แควงของนักพัฒนาแอปพลิเคชันรู้จัก และยอมรับในตัวผลิตภัณฑ์เป็นอย่างดี



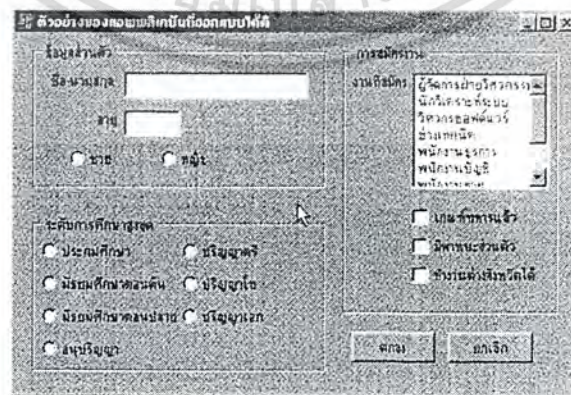
รูปที่ 2.15 เว็บไซต์ของบริษัท Inprise [1; 22]

Delphi นั้นเป็นเครื่องมือสำหรับสร้างแอปพลิเคชันแบบ Visual Programming ซึ่งทำให้เราสามารถเห็นผลลัพธ์ในการทำงานไปพร้อมกับการลงมือสร้างแอปพลิเคชัน

จุดเด่นที่สำคัญมากของความเป็น Visual Program ก็คือช่วยลดเวลาของการสร้างแอปพลิเคชัน นั่นเพราะแทนที่จะต้องไปแต่งส่วนที่ติดต่อกับผู้ใช้ โดยมอบหน้าที่ให้ Delphi เป็นต้นจัดการแทนความสามารถของ Delphi 5.0

Delphi ได้พัฒนาจนถึงเวอร์ชัน 5 ซึ่งประกอบไปด้วยเครื่องมือช่วยในการออกแบบ สร้าง ทดสอบแอปพลิเคชันที่หลากหลายช่วยให้ผลงานออกมาอย่างรวดเร็ว สำหรับความสามารถของ Delphi 5.0 หลักๆมีดังนี้

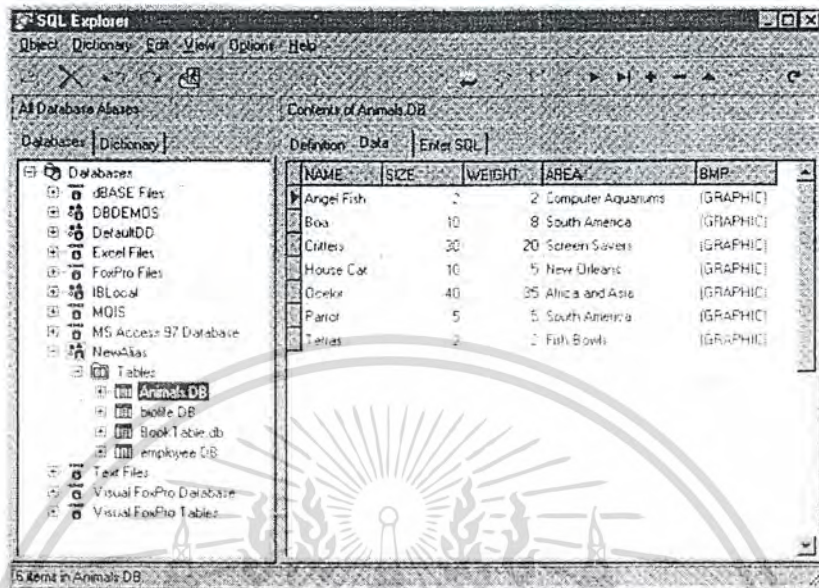
สร้างแอปพลิเคชันแบบ Visual Programming นั้นเกิดจากการเอาออบเจกต์ต่างๆมาประกอบเป็นแอปพลิเคชันซึ่งออบเจกต์เหล่านั้นส่วนใหญ่จะอยู่ในรูปของส่วนติดต่อกับผู้ใช้ ซึ่งทำให้เราสร้างแอปพลิเคชันอย่างรวดเร็ว ใช้งานง่าย และสวยงาม



รูปที่ 2.16 ตัวอย่างแอปพลิเคชันที่สร้างจาก Delphi [1; 23]

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Delphi ได้ช่วยให้การพัฒนาแอปพลิเคชัน เป็นเรื่องที่ไม่ยุ่งยากอีกต่อไป เพราะมีการรวบรวมเครื่องมือที่จำเป็นต่างๆ ไว้ให้ใช้งานกันอย่างครบถ้วน



รูปที่ 2.17 การใช้เครื่องมือในการสร้างแอปพลิเคชันของ Delphi [1; 23]

## 2.8.2 การสร้างแอปพลิเคชันใช้งานฐานข้อมูล

Delphi สามารถสร้างแอปพลิเคชันร่วมกับฐานข้อมูลที่ได้รับการยอมรับเป็นอย่างมาก ประกอบไปด้วยเครื่องมืออำนวยความสะดวกในการออกแบบ การสร้าง และการทดสอบแอปพลิเคชันฐานข้อมูล

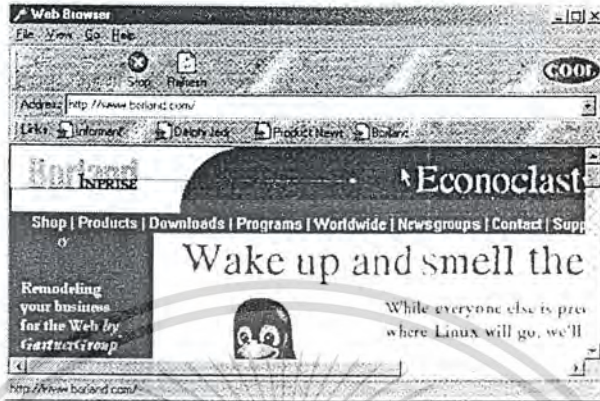


รูปที่ 2.18 ตัวอย่างแอปพลิเคชันฐานข้อมูล [1; 24]

## 2.8.3 การสร้างแอปพลิเคชันใช้งานร่วมกับ Internet

อินเทอร์เน็ตนั้น ไม่ได้มีเพียงเว็บเพจ ที่สร้างจาก HTML เท่านั้น ยังมีแอปพลิเคชันที่สามารถทำงานบนเอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

อินเทอร์เน็ตการที่จะได้รับการยอมรับ ไม่เพียงความสวยงาม สิ่งที่ต้องเพิ่มเข้ามาคือความสามารถในการทำงานของแอปพลิเคชัน Delphi5.0 ได้เตรียมความสะดวกสบาย สำหรับการสร้างแอปพลิเคชันร่วมกับอินเทอร์เน็ตไว้อย่างมากมาย



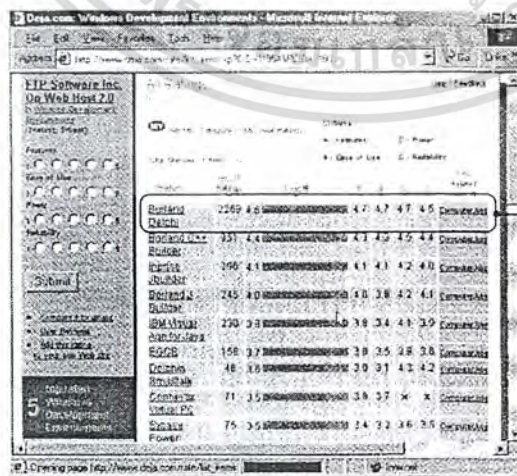
รูปที่ 2.19 ตัวอย่างแอปพลิเคชันอินเทอร์เน็ต [1; 24]

### 2.8.4 จุดเด่นของ Delphi5

การพัฒนาในเวอร์ชันที่ผ่านมาเราจะพบว่า Delphi 5.0 มีความสามารถเพิ่มมากขึ้น โดยเฉพาะความสามารถด้านการสร้างแอปพลิเคชันด้านอินเทอร์เน็ต อีกทั้งจุดเด่นของการปรับปรุงประสิทธิภาพของออบเจกต์ระบบเอกสารช่วยเหลือ

### 2.8.5 ความเร็วของแอปพลิเคชันที่สร้างจาก Delphi5

Delphi5 นั้นเป็นคอมไพเลอร์ที่สมบูรณ์ในตัว คือสามารถที่จะแปลงเป็นไฟล์ (.EXE) ที่ใช้งานได้ทันที และสามารถใช้งานได้ทุกเครื่องโดยไม่ต้องมีไฟล์เพิ่มเติมเลขทำให้แอปพลิเคชันที่สร้างจาก Delphi มีขนาดเล็กกินทรัพยากรน้อยจึงทำงานได้รวดเร็ว

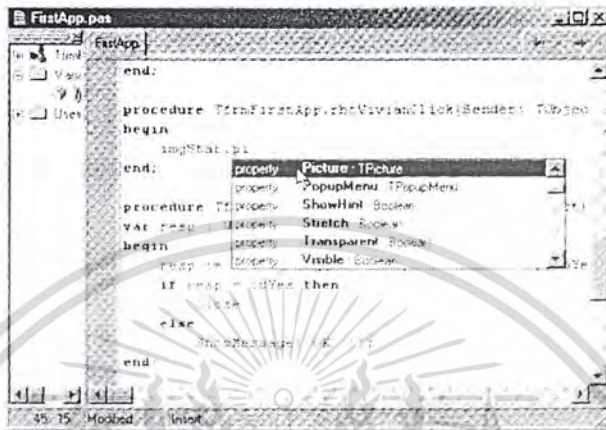


รูปที่ 2.20 ผลการจัดอันดับเครื่องมือพัฒนาแอปพลิเคชันสำหรับ Windows [1; 25]

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 2.8.6 ความสามารถทางการเขียนโปรแกรม

การเขียนโปรแกรมเป็นเรื่องความมีระเบียบ โดยธรรมชาติอยู่แล้วแต่เมื่อมีความซับซ้อนมากขึ้นทำให้การเขียนโปรแกรมเป็นเรื่องยุ่งยากอย่างหลีกเลี่ยงไม่ได้ ทำให้มีหลายๆวิธีที่คิดขึ้นมาเพื่อลดความซับซ้อนลงซึ่ง Delphi มีหลายวิธีเขียนและหลายวิธีทดสอบ



รูปที่ 2.21 ลดความยุ่งยากในการเขียน โปรแกรมใน Delphi 5.0 [1; 25]

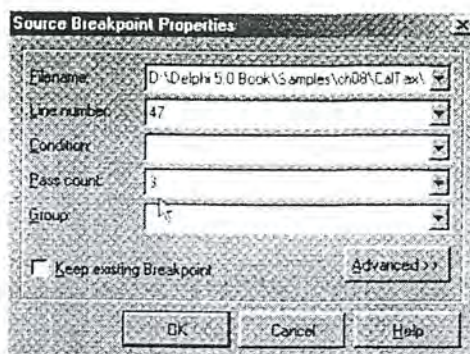
### 2.8.7 ความสามารถด้านการตรวจสอบโปรแกรม

เมื่อความซับซ้อนของโปรแกรมมีมาก ความผิดพลาดก็มากตามไปด้วย ทำให้เราต้องมีวิธีการที่ช่วยนำเอาความผิดพลาดออกมามากที่สุด และเมื่อแอปพลิเคชันที่ผู้เขียน โปรแกรมนั้นอยู่กับคนละที่เช่นการทำงานแบบเครือข่ายการตรวจสอบและแก้ไขข้อผิดพลาดต่างๆเป็นเรื่องยาก แต่ Delphi ก็มีความสามารถในการตรวจสอบและแก้ไขความผิดพลาดจากระยะไกลด้วย



รูปที่ 2.22 การตรวจสอบข้อผิดพลาดและแก้ไข โปรแกรมใน Delphi5.0 [1; 26]

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.23 การตรวจสอบข้อผิดพลาดและแก้ไขโปรแกรมใน Delphi 5.0 [1; 26]

### 2.8.8 ความสามารถในการสร้างแอปพลิเคชันอินเทอร์เน็ต

อินเทอร์เน็ตในวันนี้ไม่ใช่เรื่องที่จะมองข้าม ถือว่าเป็นหนทางรอดของเราเลยทีเดียว Delphi 5.0 นั้นถึงกับเน้นความเป็นเครื่องมือพัฒนาแอปพลิเคชันอินเทอร์เน็ต จากเดิมที่จะเน้นทางด้านฐานข้อมูลและแอปพลิเคชันใช้งานทั่วไป

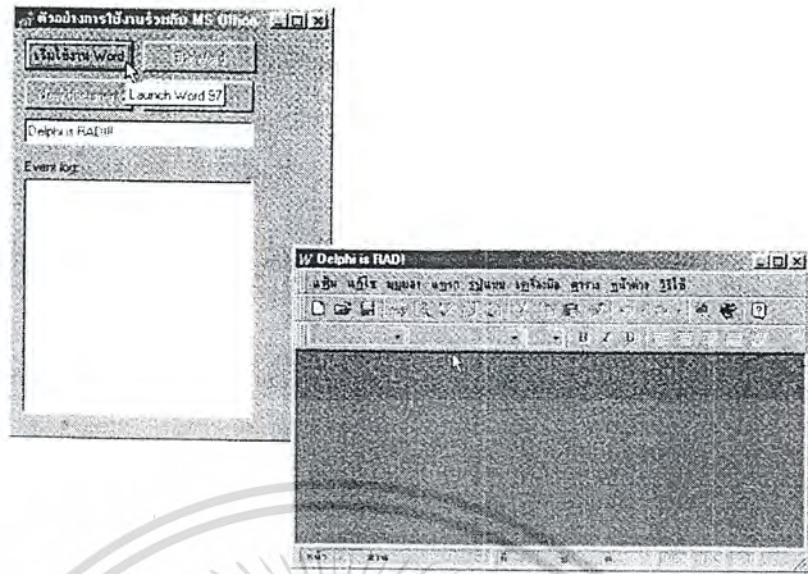


รูปที่ 2.24 ตัวอย่างแอปพลิเคชันอินเทอร์เน็ตในฝั่งเซิร์ฟเวอร์ [1; 27]

### 2.8.9 เข้ากับเทคโนโลยีของไมโครซอฟท์

อย่างที่ทราบกันว่าไมโครซอฟท์เข้าไปซื้อหุ้นของ inprise และนั่นก็ทำให้ Delphi ใกล้กับเทคโนโลยีไมโครซอฟท์มากขึ้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.25 ตัวอย่างแอปพลิเคชันที่ใช้งานร่วมกับ MS Office [1; 27]

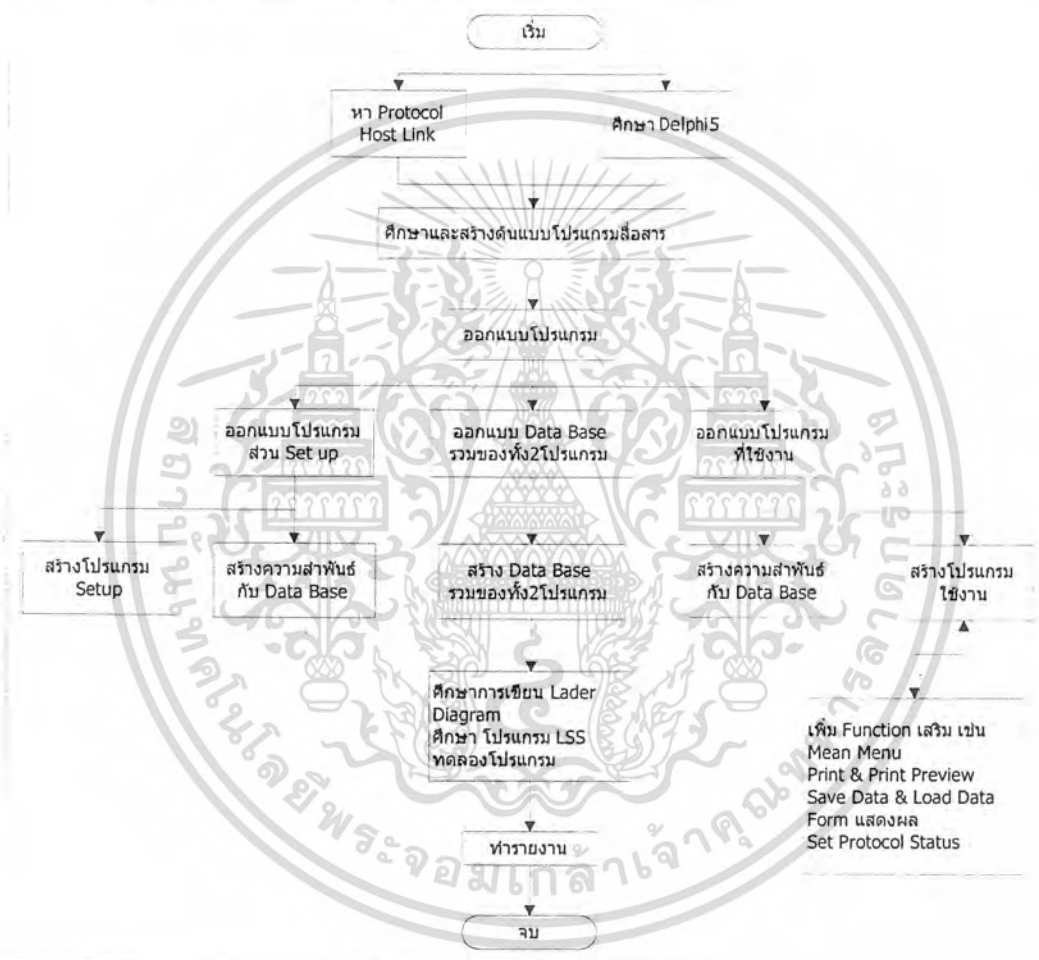


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## บทที่ 3 วิธีการดำเนินการ

### 3.1 แผนการดำเนินการ

แผนการดำเนินการสร้างโปรแกรมการเขียนและอ่านข้อมูลจากเครื่องควบคุมแบบตรรก  
โปรแกรมได้โดยคอมพิวเตอร์ ดังแผนภาพต่อไปนี้



รูปที่ 3.1 รูปภาพแสดงแผนการสร้าง โปรแกรม

### 3.2 ขั้นตอนการสร้างโปรแกรมแบ่งได้ดังนี้

- |             |  |
|-------------|--|
| ขั้นตอนที่1 | การออกแบบ  |
| ขั้นตอนที่2 | สร้างโปรแกรมการติดต่อระหว่าง PLC และ คอมพิวเตอร์ |
| ขั้นตอนที่3 | สร้างโปรแกรมหลัก                                 |
| ขั้นตอนที่4 | ศึกษาโปรแกรมLSS                                  |
| ขั้นตอนที่5 | สร้างโปรแกรมเสริม                                |

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานในเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 3.2.1 ขั้นตอนการออกแบบขั้นตอนแบบโปรแกรม

การออกแบบโปรแกรมจะยึดตามหลักของวิชา Product Design โดยใช้วิธี Build Form Function คือออกแบบตามหน้าที่ที่ได้ตั้งไว้

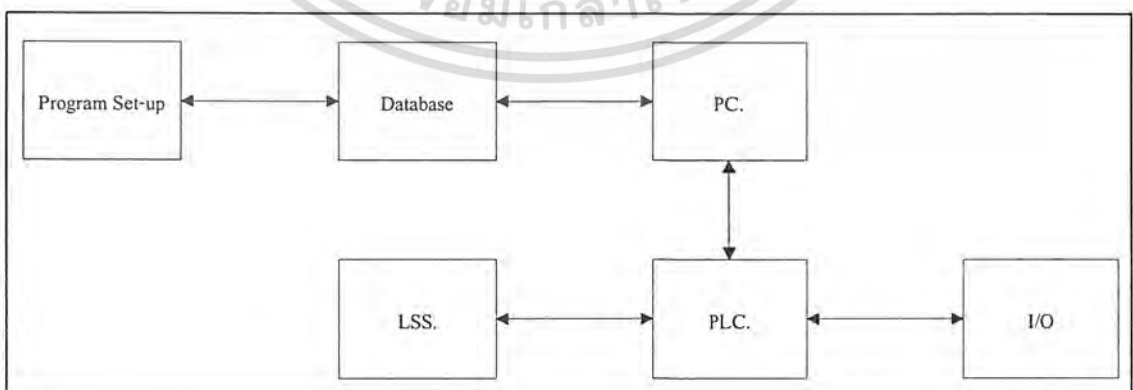
หน้าที่และข้อกำหนดของโปรแกรมของโปรแกรมที่ได้ตั้งไว้

- win98
- delphi5
- โปรแกรมสามารถปรับเปลี่ยนได้ตามต้องการ
- โปรแกรมสามารถพัฒนาต่อได้
- โปรแกรมแบบ On-Line
- สามารถบอกวิธีใช้และวิธีแก้ไขเบื้องต้นได้
- ใช้รหัสผ่านได้
- แยกโปรแกรมตามประเภท(การออกแบบและการใช้งาน)
- สามารถพิมพ์สรุปผล
- สามารถเก็บข้อมูลของผลิตภัณฑ์
- โปรแกรมสามารถใช้ไปโตคอลได้ครบ
- สามารถทำงานผ่าน Mouse
- แสดงผลแบบ Real-Time

ดังนั้นรูปแบบของ โปรแกรมหลักสามารถทำเป็นขั้นตอนโดย

1. จะต้องมี2 โปรแกรมคือ 1. โปรแกรม Set up 2. โปรแกรมที่ใช้งาน โดยจะทำการสื่อสารกันด้วยระบบ Database ที่ต้องใช้ระบบ Database เพราะว่าในอนาคตอาจให้เครื่องอื่นสามารถเปิด
2. ใน โปรแกรมจะต้องมีการเก็บข้อมูลและนำข้อมูล ไปพิมพ์เนื่องจากว่าการเปิดไฟล์ Data จะต้องไม่เกี่ยวกับโปรแกรมที่กำลังทำงานดังนั้นการเปิดดูไฟล์จะต้องเป็นเพียงเอกสารเท่านั้น

โครงสร้างของโปรแกรมหลักแสดงดังรูป3.2

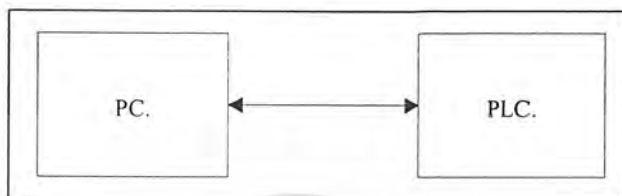


รูปที่ 3.2 รูปภาพแสดงภาพความสัมพันธ์ระหว่างอุปกรณ์กับ โปรแกรมที่ได้ออกแบบไว้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

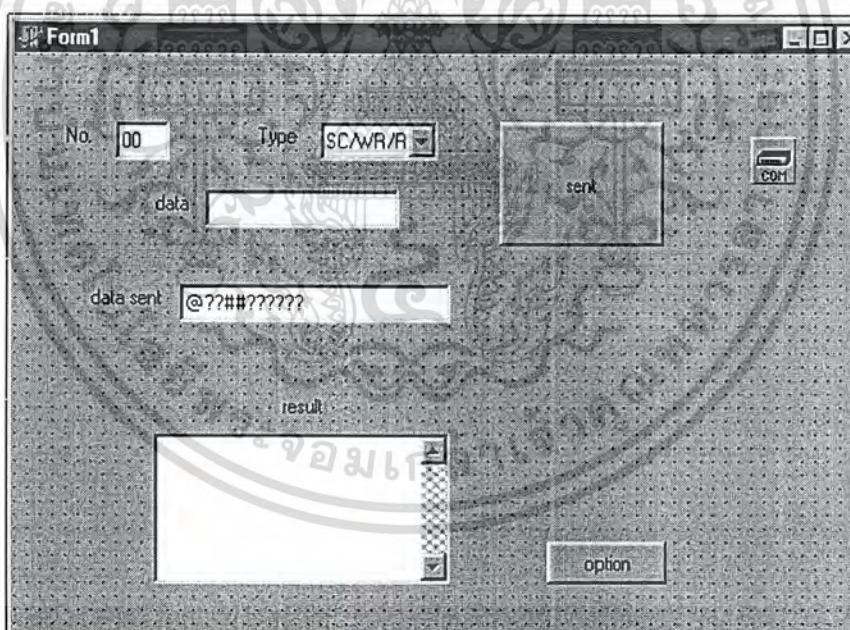
### 3.2.2 ขั้นตอนการสร้างโปรแกรมทดลองการติดต่อระหว่าง PLC และคอมพิวเตอร์

ทำการเขียน โปรแกรมติดต่อระหว่าง PLC กับ คอมพิวเตอร์ (ไม่มีการสื่อสารแบบทันทีเหตุการณ์) ขั้นตอนนี้ ได้ทำการติดตั้งเครื่องมือช่วยเหลือเป็นที่เรียบร้อย โดยใช้ Comport เป็น Component ของ Delphi 5 ที่ถูกพัฒนาโดยนาย Erik Salaj โครงสร้าง โปรแกรมแสดงดังรูป 3.3



รูปที่3.3 แสดงความสัมพันธ์ระหว่าง โปรแกรมสื่อสารที่ทำขึ้นมา

โปรแกรมทดลองนี้ เป็นการสื่อสารกับPLC. โดยการกดปุ่ม Sent แล้ว โปรแกรมจะทำการถามไปยัง PLC. เพื่อให้ PLC. จะตอบกลับมาโดยคำตอบแสดงผ่าน Memo แสดงดังรูป3.4

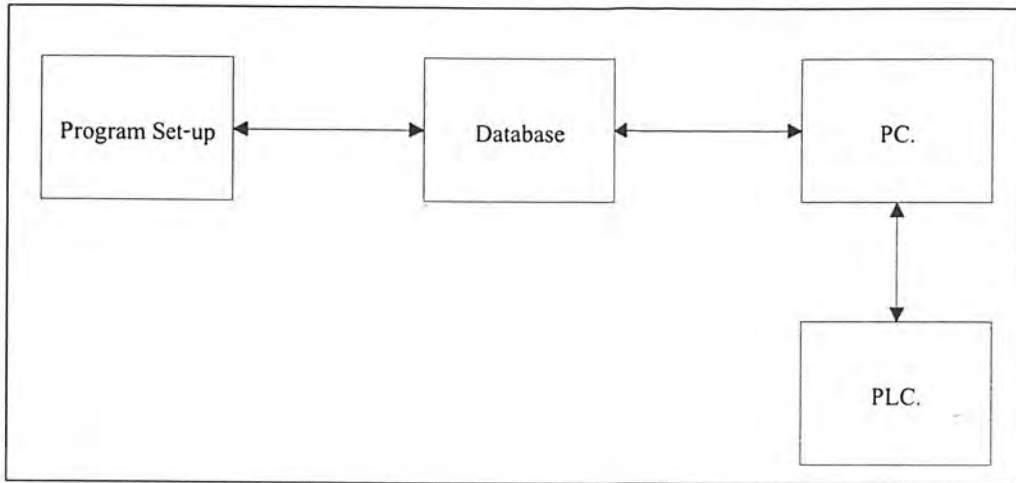


รูปที่ 3.4 รูปภาพแสดงการเขียน โปรแกรมสื่อสารผ่าน port และใช้โปรโตคอล Host Link ของOmron

### 3.2.3 ขั้นตอนการสร้างโปรแกรมหลัก

ทำการเขียน โปรแกรมหลักเพื่อการติดต่อสื่อสารระหว่าง PLC. กับคอมพิวเตอร์ และ โปรแกรมติดตั้ง แสดงโครงสร้างความสัมพันธ์ระหว่าง โปรแกรมดังรูป3.5

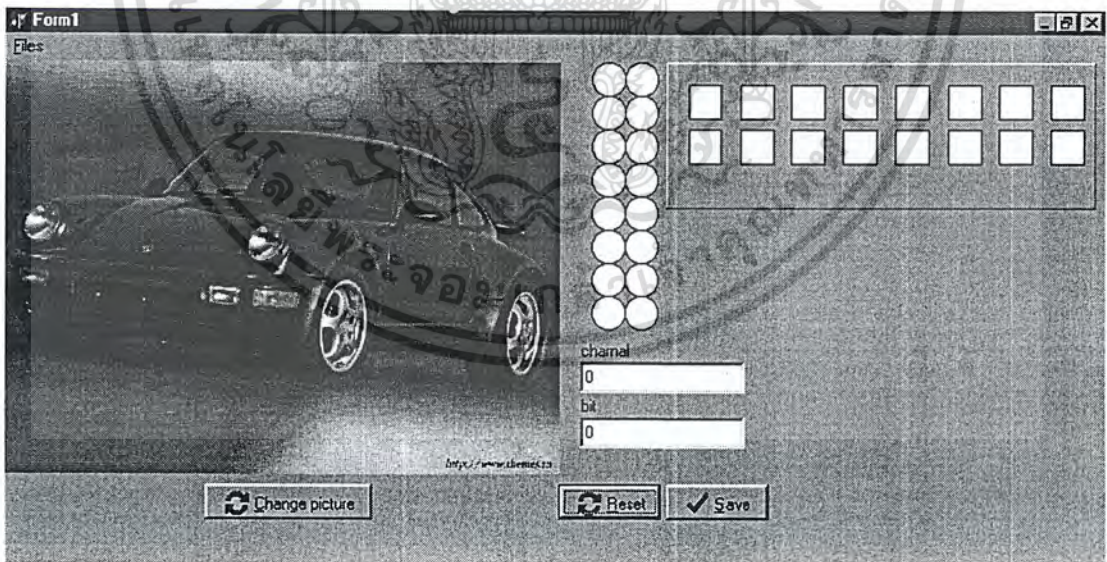
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.5 แสดง โครงสร้างความสัมพันธ์ระหว่างโปรแกรม

โปรแกรมที่ออกแบบสำหรับการ เชื่อมโยงสั่งการแบบ Bit และ Word และเชื่อมต่อกันกับฐานข้อมูล

- การสั่งการแบบ Bit จะแบ่งเป็น 2 ส่วน คือส่วนของการแสดงผล และส่วนของการสั่งงาน
- การสั่งการแบบ Word ทำการเขียน โปรแกรมแบบการสั่งงาน โดยใช้โปรโตคอล Host Link ของ Omron โดยจำกัดความสามารถด้วยคำสั่ง “WR” (Write Relay) ที่สามารถสั่งให้หน้า Contract ของ หน่วยอินพุท หรือ เอาท์พุท เปิด หรือ ปิด พร้อมๆกัน
- ในด้านความสัมพันธ์ของโปรแกรมติดตั้ง และ โปรแกรมหลักใช้ Database เป็นตัวกลาง



รูปที่ 3.6 รูปภาพแสดง โปรแกรม ติดตั้ง

หน้าที่ของโปรแกรม ติดตั้ง มีหลักการทำงานดังนี้

- ทำการ Load Database

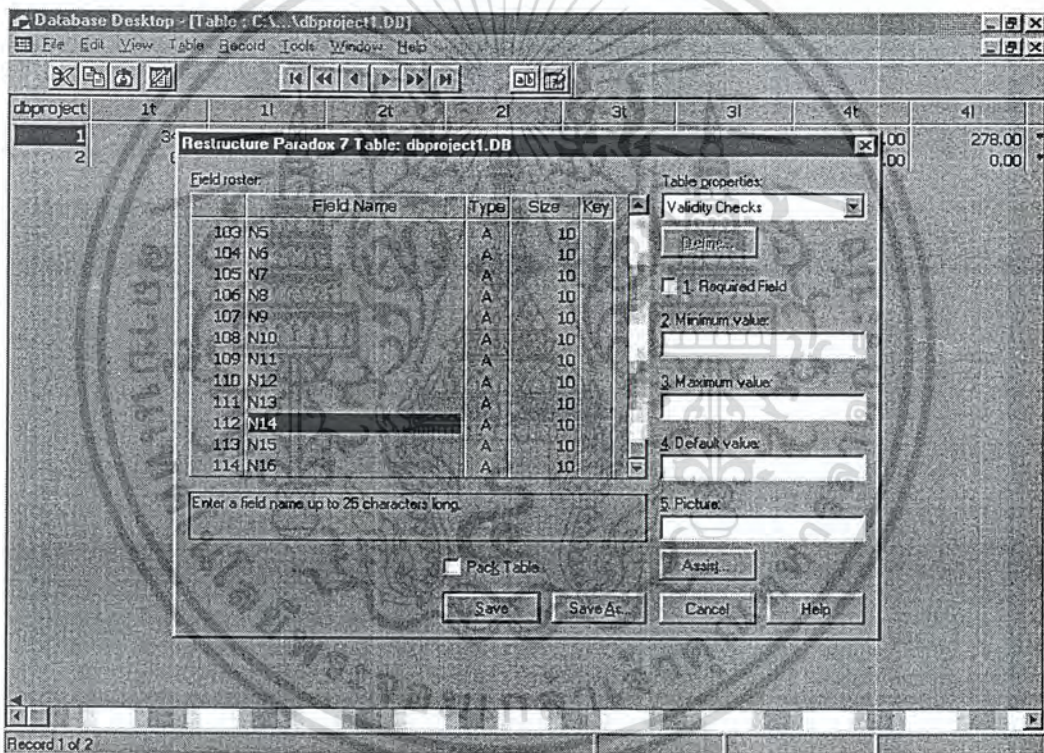
เอกสารนี้เป็นเอกสารที่เผยแพร่เพื่อการศึกษานี้เท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- ทำการ Save Database

การเชื่อมโยง Database เป็นการสร้างเรคคอร์ดให้ค่าครบเท่ากับจำนวนตัวแปรที่จะเก็บข้อมูลดังนั้นจึงจำแนกข้อมูลเป็น

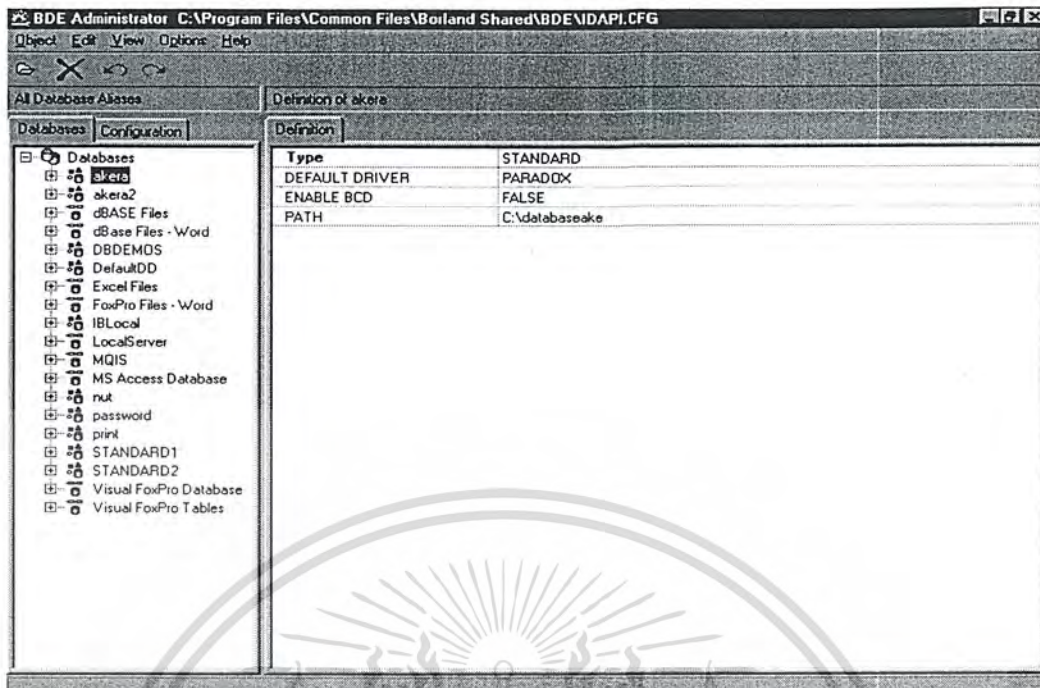
- การเก็บค่าตำแหน่ง 32 เรคคอร์ด
  - การเก็บค่าBiและChannel 64 เรคคอร์ด
  - การเก็บค่ารายชื่อปุมอีก 16 เรคคอร์ด
  - การเก็บค่ารายชื่อรูปอีก 1 เรคคอร์ด
  - การเก็บค่าทั่วไปอีก 1 เรคคอร์ด
- รวม 114 เรคคอร์ด

โดยใช้ Database แบบ Paradox แสดงรูปแบบการฐานข้อมูลในรูปที่ 3.7 และรูปที่ 3.8



รูปที่ 3.7 รูปภาพแสดงการสร้าง Database โดยผ่านโปรแกรม Database Desktop

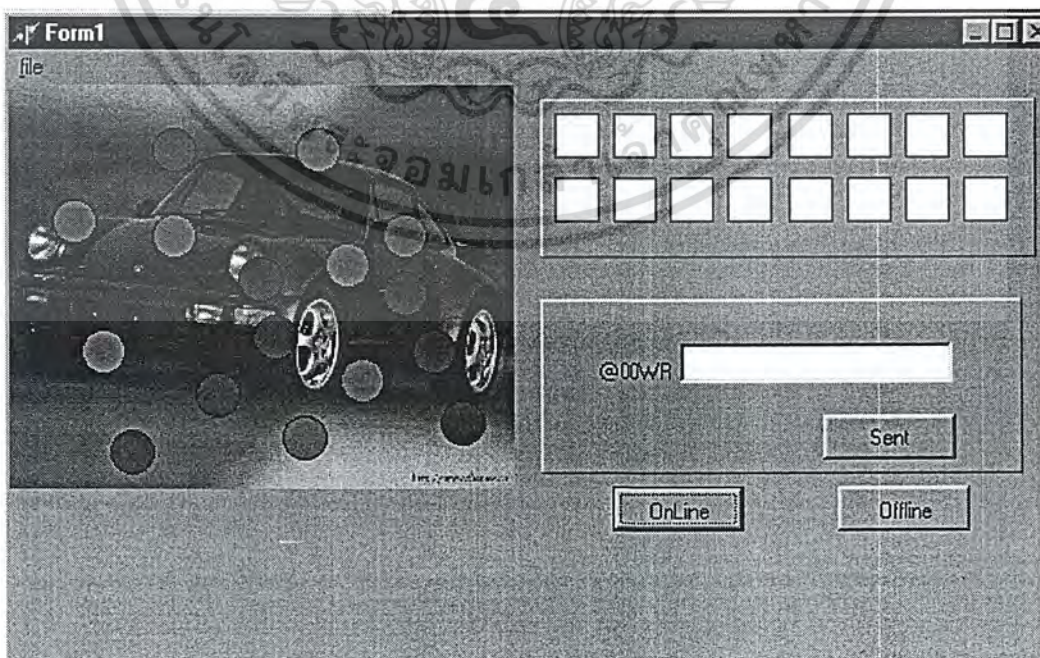
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.8 รูปภาพแสดงการสร้าง Database ขั้นตอนการกำหนด Aliases ใน BDE.Administrator

โปรแกรม ComLinkPLC มีหลักการทำงานคือ

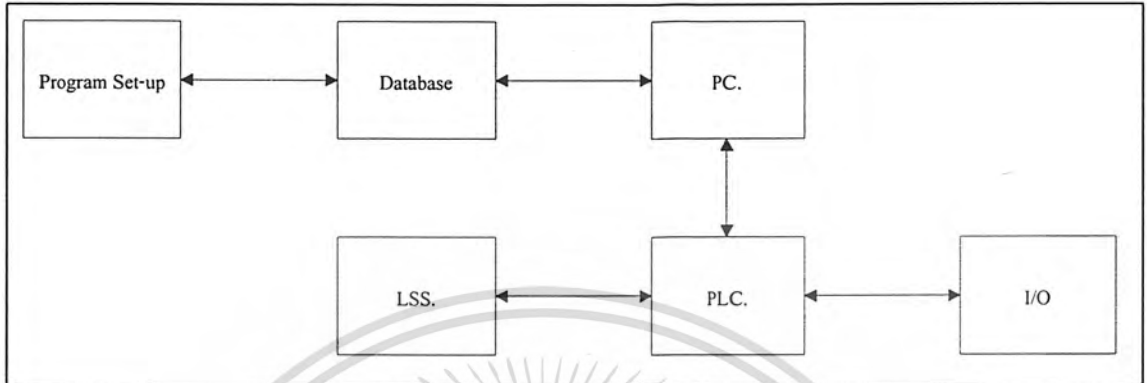
- Load File Database ที่โปรแกรมติดตั้ง ได้ Save ไว้
- ทำการสื่อสารแบบ On-Line ได้
- ทำการสั่งการ PLC. ผ่านโปรแกรมได้



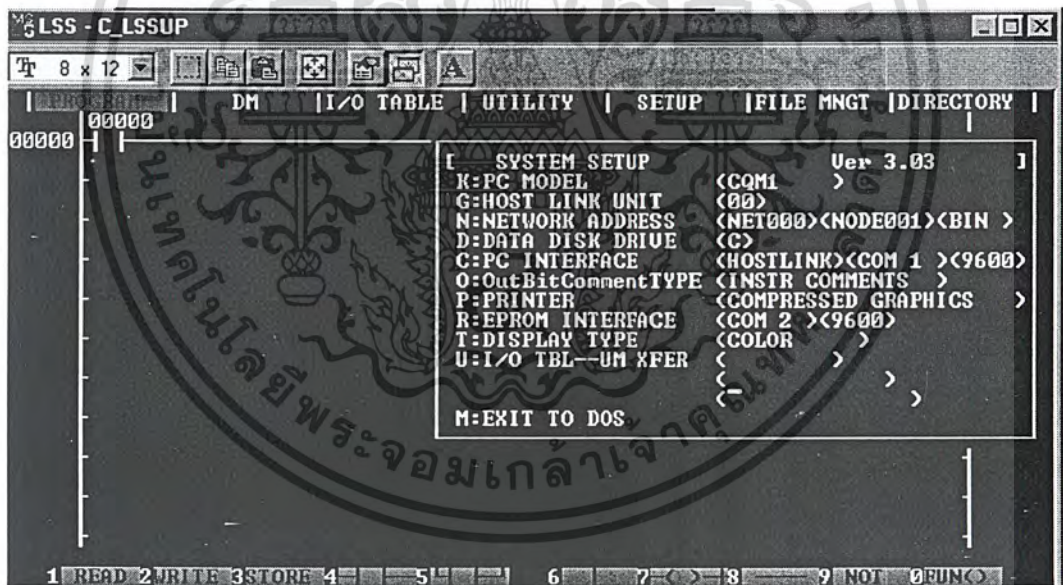
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับใช้เฉพาะในวงจำกัดเท่านั้น ไม่ควรเผยแพร่ไปยังบุคคลอื่นโดยไม่ได้รับอนุญาต  
รูปที่ 3.9 รูปภาพแสดงการเขียนโปรแกรมโปรแกรมหลักเบื้องต้น  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 3.2.4 ขั้นตอนการศึกษาโปรแกรมLSS

โปรแกรม LSS เป็นตัวเขียนโปรแกรมLadder Diagram ให้กับ PLC แสดงดังรูปที่3.10และรูปที่3.11



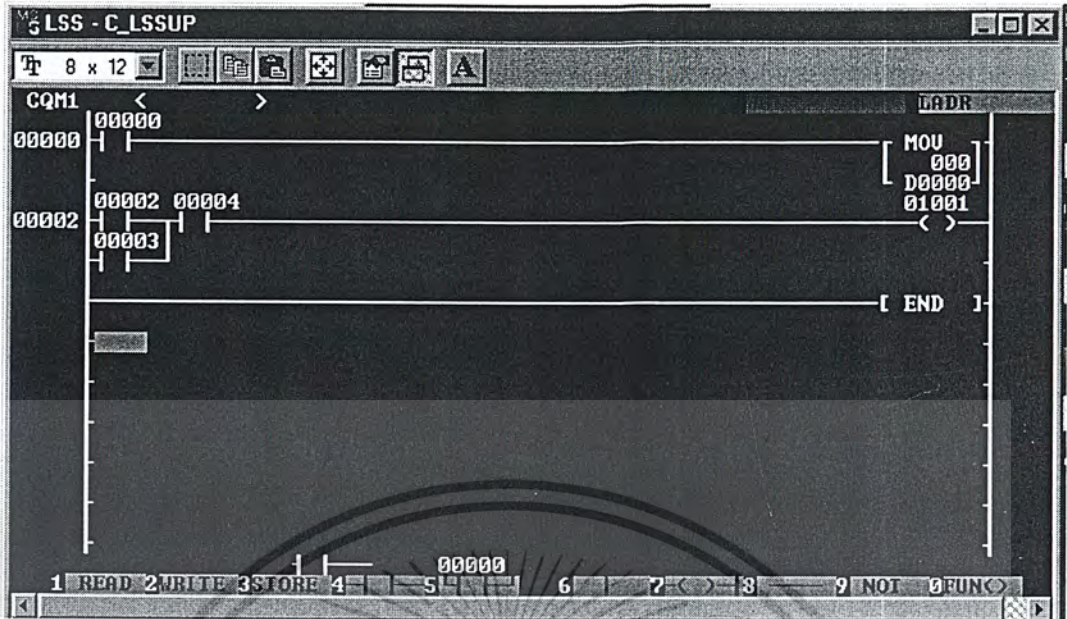
รูปที่3.10 แสดงโครงสร้างความสัมพันธ์ระหว่างโปรแกรม LSS กับ PLC.



รูปที่ 3.11 รูปภาพแสดงโปรแกรม LSS ของ Omron

ภายในโปรแกรม LSS มีรูปแบบรูปแบบการเขียนแบบแผนภาพเชิงเส้น (Ladder Diagram) และการเขียนเชิงตัวเลข(Numerical Language) ดังรูป 3.12 และ 3.13

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



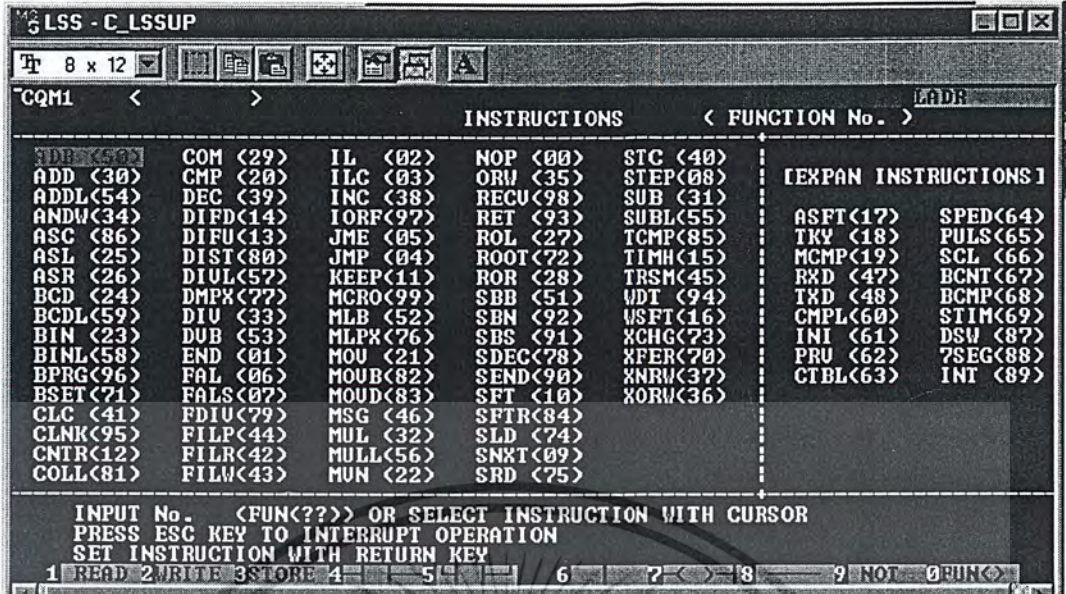
รูปที่ 3.12 รูปภาพแสดงการเขียนโปรแกรมในรูปแบบแผนภาพเชิงเส้น (Ladder Diagram)

รูปที่ 3.13 รูปภาพแสดงการเขียนโปรแกรมในรูปแบบเชิงตัวเลข(Numerical Language)

การเรียกใช้ Function ของโปรแกรม LSS เรียกใช้โดยการกด F10 หน้าี่การทำงานทั้งหมดแสดงในรูป

3.14

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.14 รูปภาพแสดงFunction ในโปรแกรม LSS

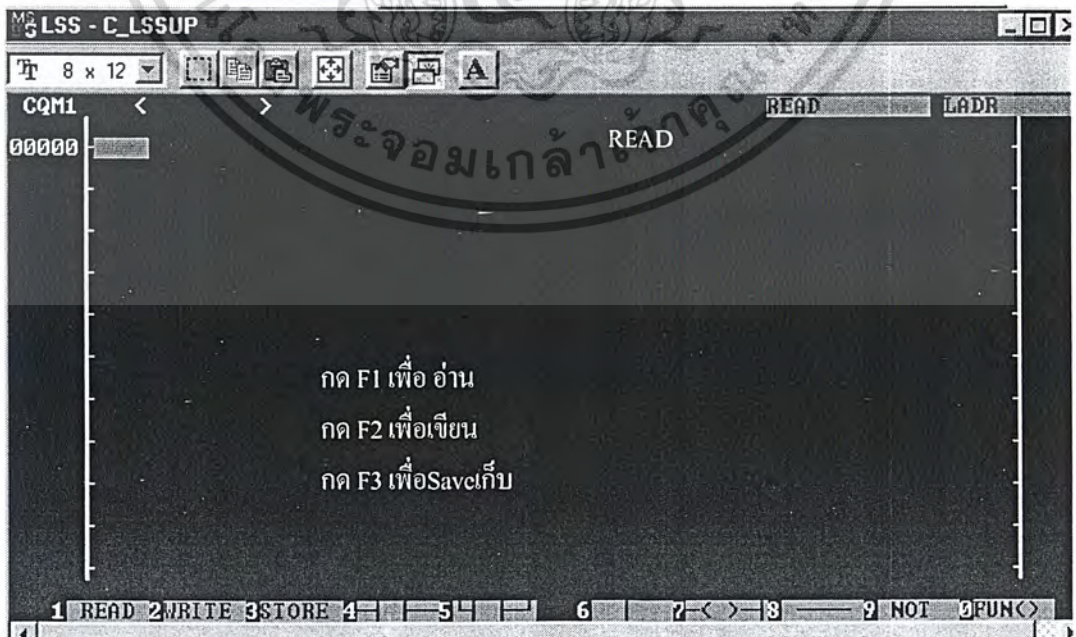
การเขียนโปรแกรม LSS

เริ่มจากออกแบบ Ladder Diagram แล้วจึงนำ Ladder มาเขียนลงไปใน โปรแกรม LSS แล้วจากนั้นจึงทำการถ่ายข้อมูลลงไปยัง PLC.

ขั้นตอนการทำงานของโปรแกรม LSS.

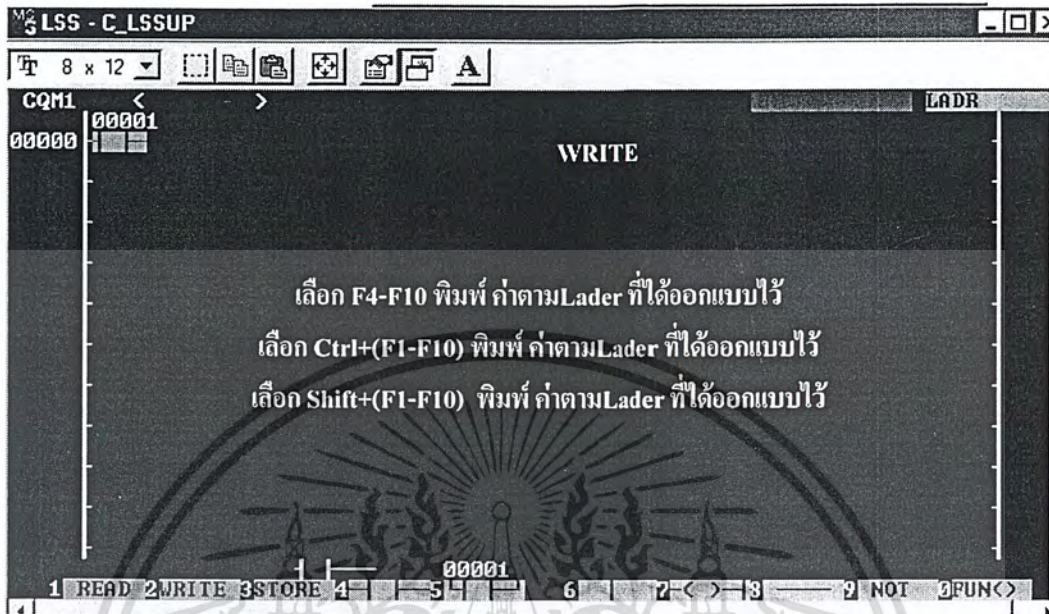
ขั้นตอนที่ 1 การเปิด โปรแกรมแล้วปรับสถานะ การเขียน,อ่าน,เก็บข้อมูล โดยการกด F1,F2,F3

ขั้นตอนที่ 2 เลือกการเขียน โปรแกรมโดยการกด F1



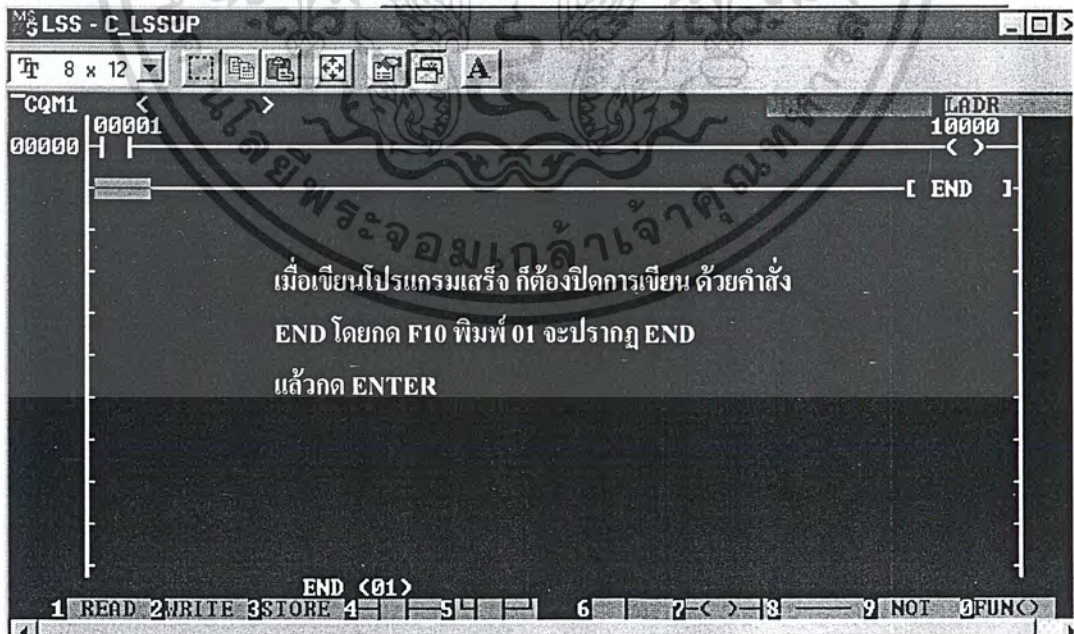
เอกสารนี้เป็นเอกสารที่สงวนไว้รูปที่ 3.15 รูปภาพแสดงการปรับสถานะ การเขียน,อ่าน,เก็บข้อมูลให้ใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ขั้นตอนที่ 3 การเขียน โปรแกรม Ladder Diagram สามารถเลือกสัญลักษณ์ต่างๆ โดยการกด F4-F10 , Ctrl+(F4-F10),Shift+(F4-F10)



รูปที่ 3.16 รูปภาพแสดงการเลือกสัญลักษณ์ต่างๆ

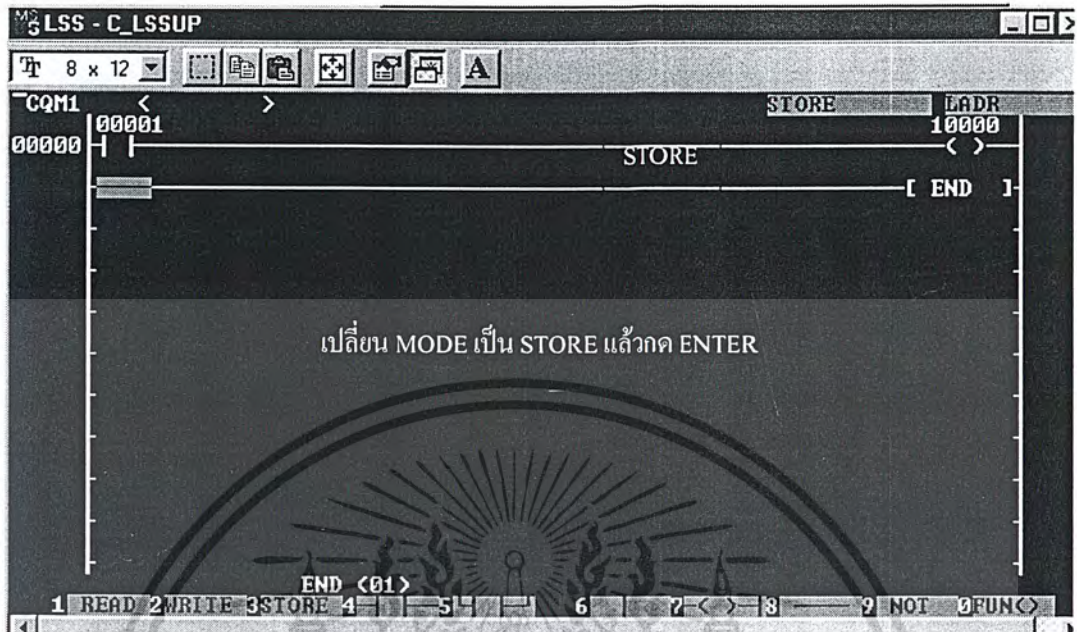
ขั้นตอนที่ 4 ใช้ Function 01 ในการจบ โปรแกรม โดยการกด F10



รูปที่ 3.17 รูปภาพแสดงการจบโปรแกรม LSS

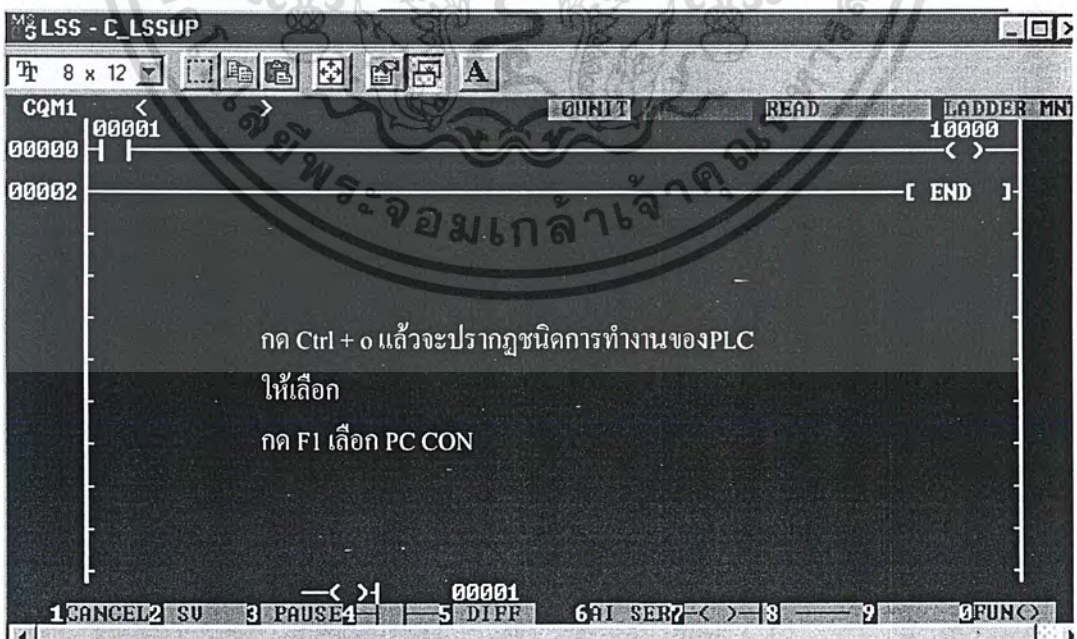
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ขั้นตอนที่ 5 ทำการเปลี่ยนรูปแบบการทำงานแบบเก็บข้อมูล โดยการกด F3



รูปที่ 3.18 รูปภาพแสดงการเก็บ โปรแกรมที่ได้เขียนไว้ใน LSS

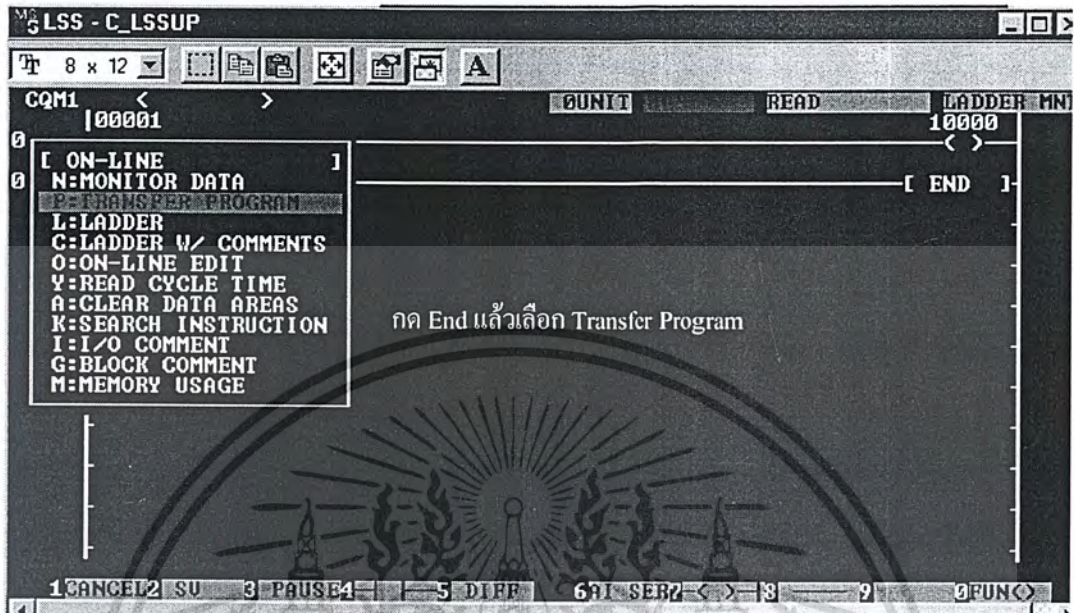
ขั้นตอนที่ 6 ทำการติดต่อกับ PLC เพื่อการขนถ่ายข้อมูลจากคอมพิวเตอร์ไปยัง PLC. โดยการกด Ctrl + O เลือก PC CON



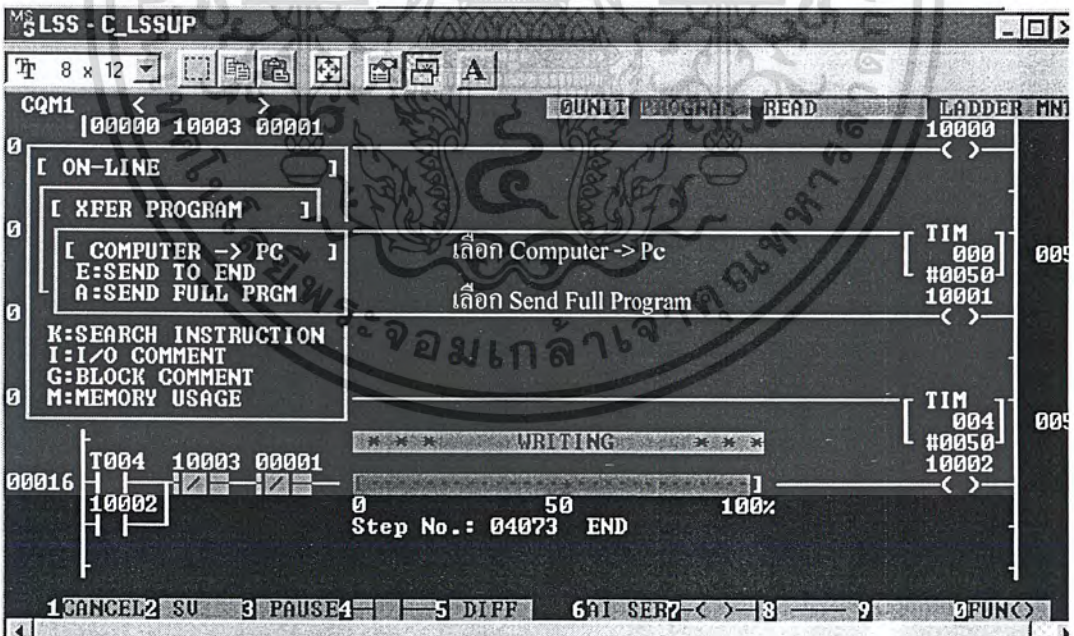
รูปที่ 3.19 รูปภาพแสดงขั้นตอนการเลือกชนิดการติดต่อกับ โปรแกรม LSS

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ขั้นตอนสุดท้าย การขนถ่ายข้อมูลจากคอมพิวเตอร์ไปยัง PLC. โดยกด End เลือก Menu – Transfer Program – Computer->PC - Send Full Programแสดงรายละเอียดวิธีการขนถ่ายโปรแกรมในรูปที่ 3.20 และรูปที่ 3.21



รูปที่ 3.20 รูปภาพแสดงรายละเอียดวิธีการขนถ่ายโปรแกรม

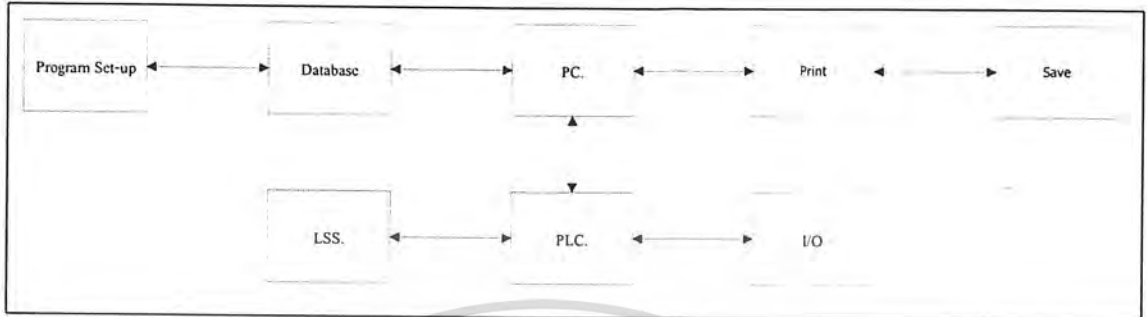


รูปที่ 3.21 รูปภาพแสดงรายละเอียดวิธีการขนถ่ายโปรแกรม.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 3.2.5 ขั้นตอนการสร้างโปรแกรมเสริม

ขั้นตอนนี้เขียน โปรแกรมเพื่อเพิ่มส่วนขยายของ โปรแกรม

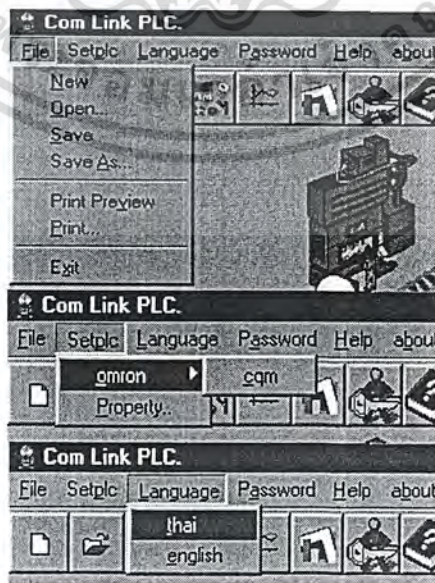


รูปที่ 3.22 แสดงความสัมพันธ์ระหว่างโปรแกรมที่สร้างขึ้น

ชุดคำสั่งและหน้าที่ของโปรแกรมย่อได้แก่

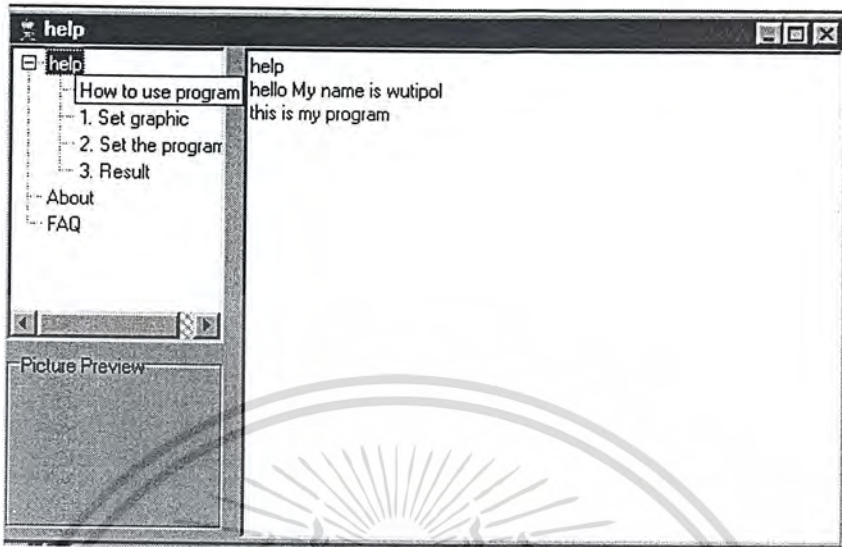
1 Menu : หลักการสร้างเมนูจะอาศัยรูปแบบมาตรฐานของ โปรแกรมทั่วไปซึ่งประกอบด้วย

- File : มีหน้าที่เปิด,สร้าง,จัดเก็บข้อมูล และ ฐานข้อมูล รวมทั้งการจัดพิมพ์
- Setplc : มีหน้าที่ในการกำหนดลักษณะการติดต่อของ PLC. กับระบบ
- Language : มีหน้าที่ในการเปลี่ยนภาษาที่ใช้
- Password : เป็นการติดตั้งระบบรักษาความปลอดภัยสำหรับการแก้ไขเปลี่ยนแปลงของโปรแกรมและระบบ
- Help : เป็นตัวช่วยเหลือและทำความเข้าใจต่อผู้ใช้โปรแกรม
- About : เป็นข้อมูลส่วนตัวของผู้สร้างโปรแกรม



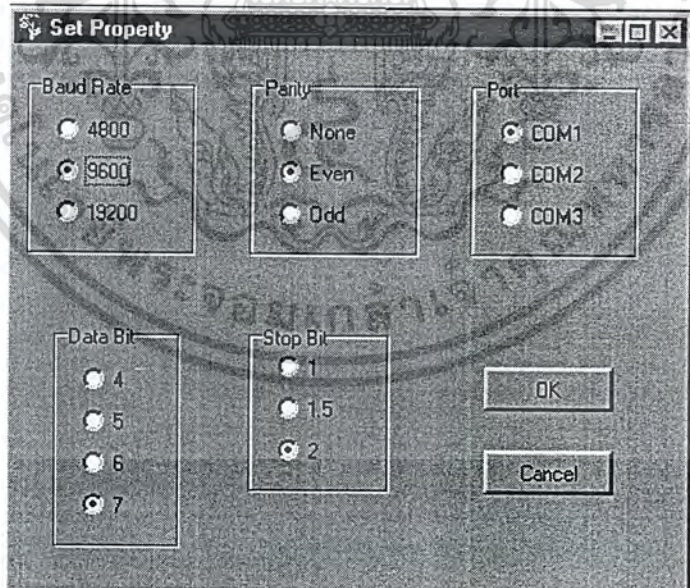
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับครูใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
 รูปที่ 3.23 รูปภาพแสดงเส้นทางในการเขียน Menu  
 ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2 Help : การสร้างhelp ทำการออกแบบตามหลักการของรากต้นไม้(Tree Diagram)



รูปที่ 3.24 รูปภาพแสดงรูปแบบของHelp

การส่ง 3 ติดตั้งค่าคงที่การส่ง : มีไว้สำหรับรองรับการเปลี่ยน PLC. โดย PLC. ตัวอื่นอาจจะมีการเปลี่ยนค่าคงที่

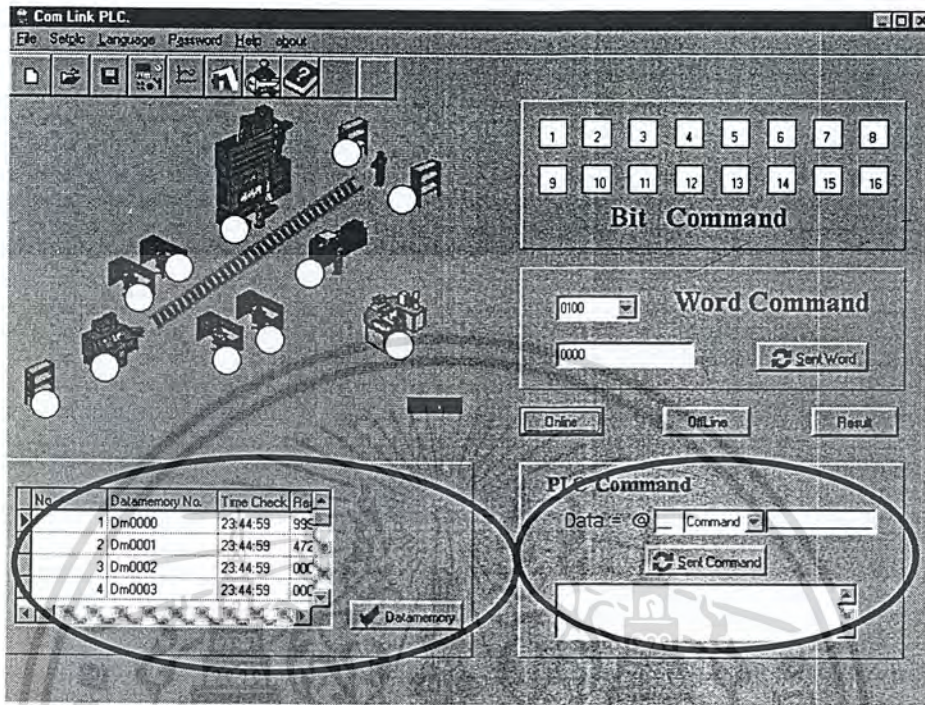


รูปที่ 3.25 รูปภาพการกำหนดค่าคงที่การสื่อสาร

4 DataMemory+PLC. Command(การส่งสัญญาณแบบภาษา Host-Link) : 2โปรแกรมนี้มีลักษณะ

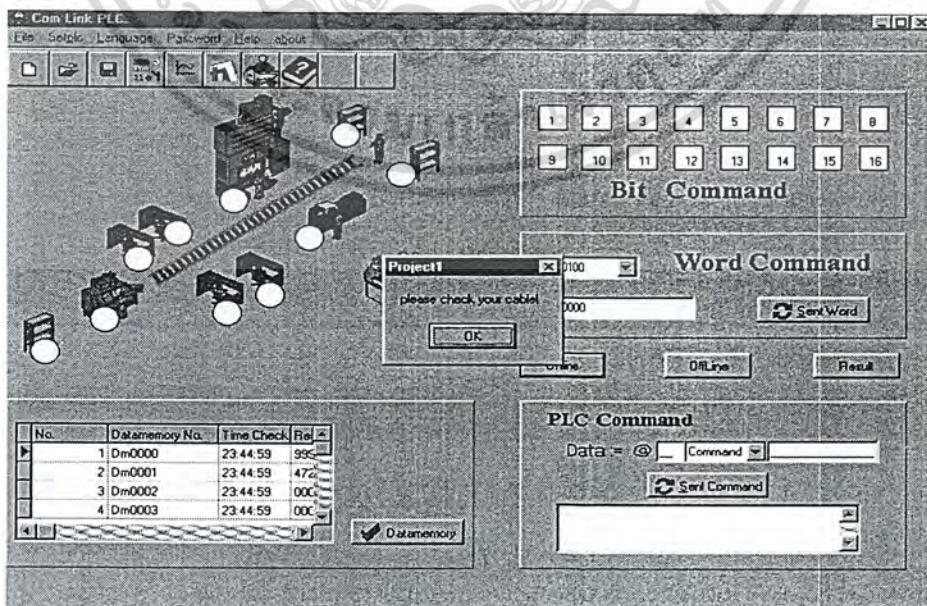
คล้ายกับ Word Command ที่ไม่Online แต่ต่างกันตรงที่การใช้งานคือ DataMemory จะเข้า Check เฉพาะเอกสารนี้เป็นเอกสารที่ส่งวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

DataMemory และสามารถทำการ Save และ Print ส่วน PLC Command จะสามารถส่งข้อมูลไปยัง PLC ได้ครบทุกโปรโตคอลที่ใช้ แต่จะไม่สามารถ Print และ Save



รูปที่ 3.26 รูปภาพแสดง ตำแหน่งของ DataMemory & PLC Command

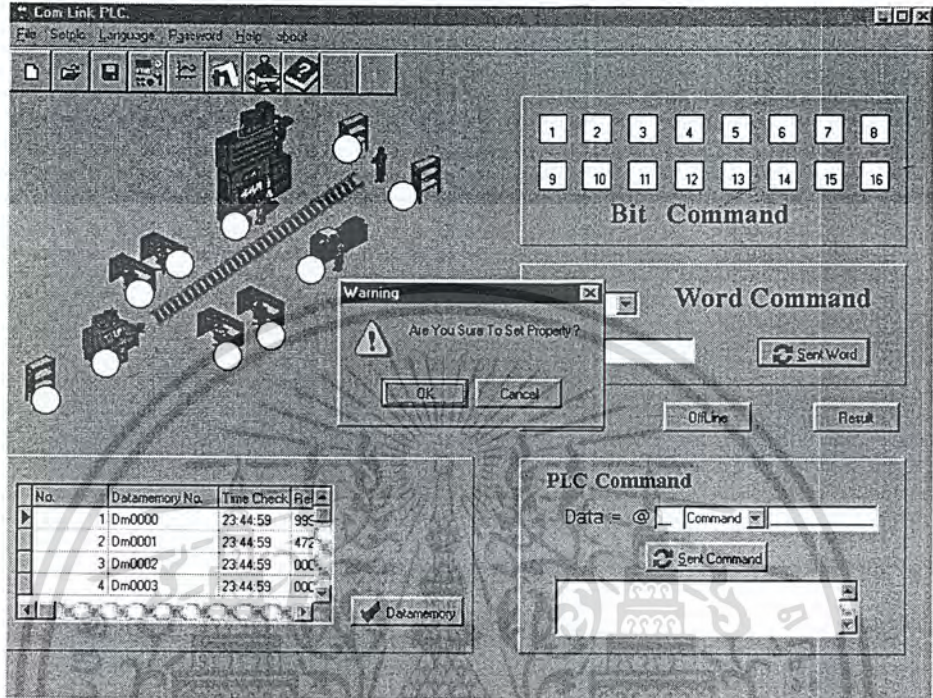
5 การตรวจสอบข้อผิดพลาดโปรแกรม : ในส่วนของการ Online ต้องมีการดูว่าสายที่สื่อสารกันอยู่นั้นเสียหายหรือไม่ถ้าเสียหายก็ต้องมีการเดินและหยุด Online ทันที



รูปที่ 3.27 รูปภาพแสดงการเตือนว่าสายสัญญาณขาด

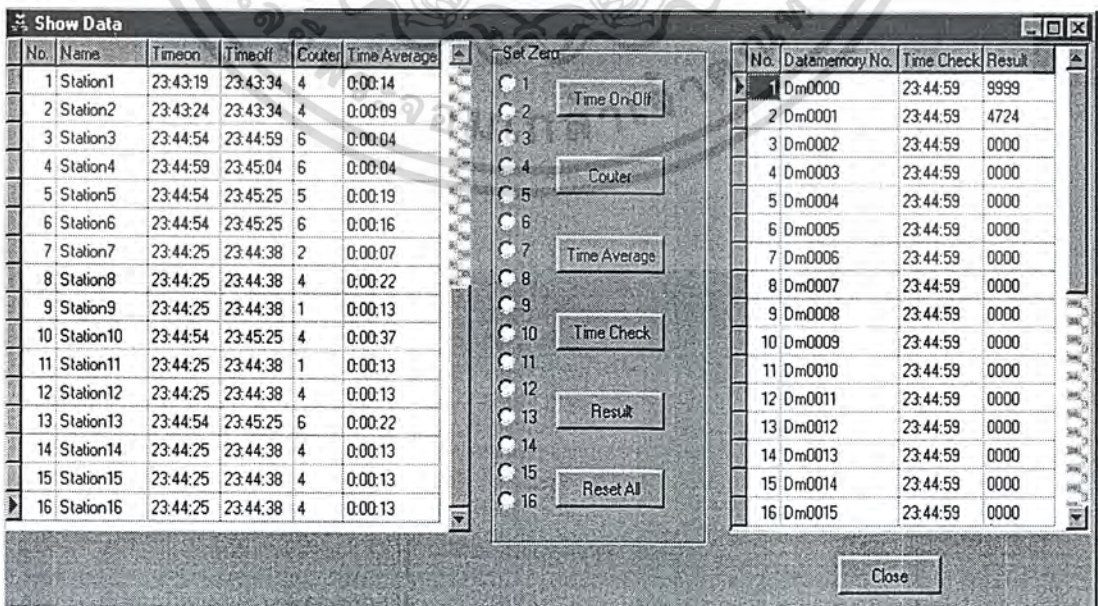
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับใช้เฉพาะในวงจำกัดเท่านั้น ไม่ควรเผยแพร่ให้วงกว้างไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

6 Dialogแจ้งเตือน : ต้องมี Dialogเตือนเนื่องจากใน โปรแกรมการสั่งงานบางอย่างอาจส่งผลกระทบต่อระบบเช่นการปรับค่าคงที่การสื่อสาร



รูปที่ 3.28 รูปภาพแสดงการเตือนก่อนการกำหนดค่าคงที่การส่ง

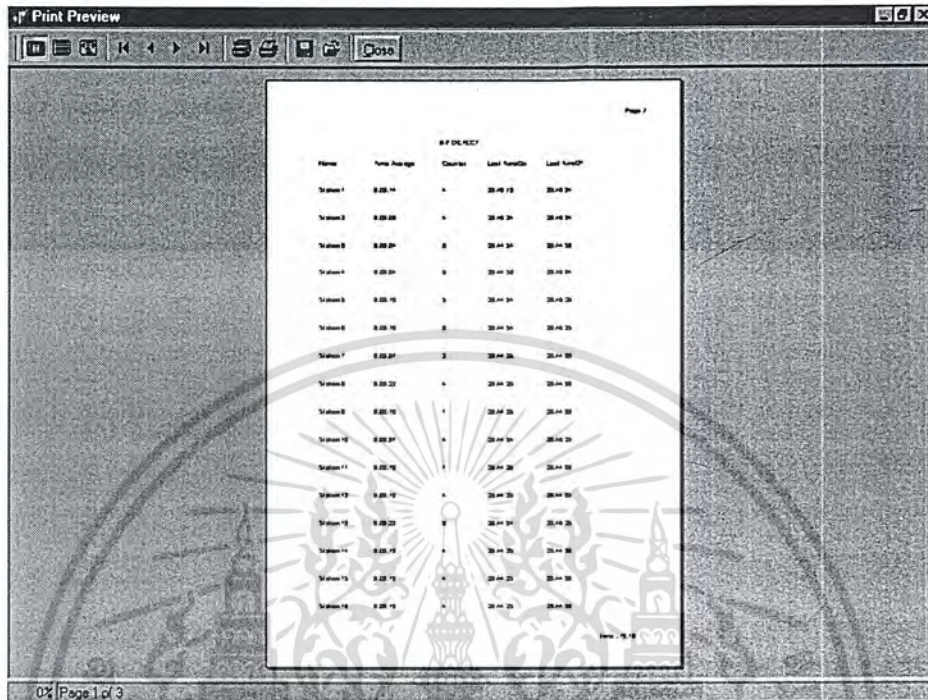
7 การแสดงผล : จะแสดงผลของBit Command กับ DataMemory



รูปที่ 3.29 รูปภาพแสดงผลของBit Command กับ DataMemory

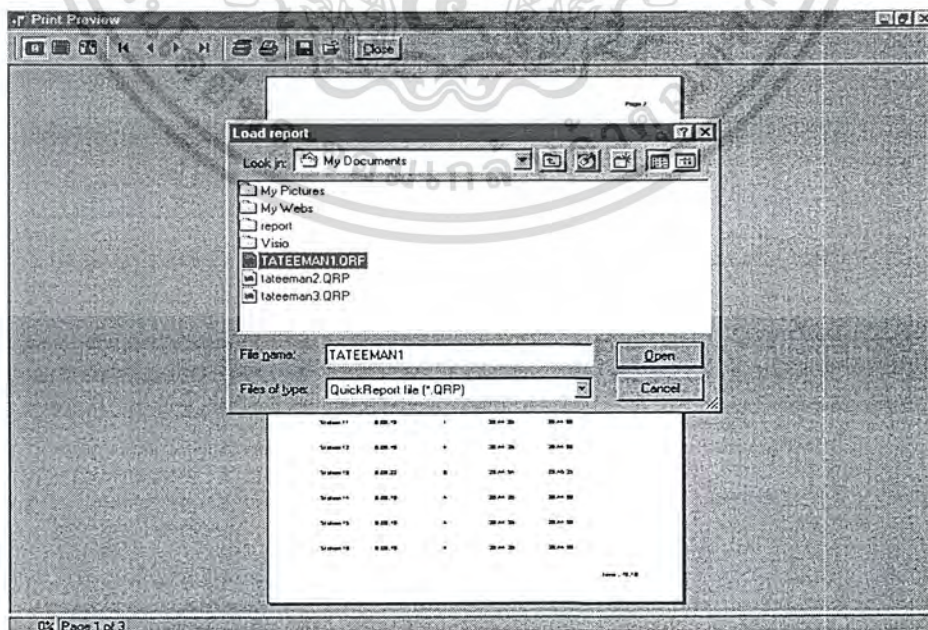
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับใช้เฉพาะงานที่องค์กรสื่อสารเท่านั้น ไม่ควรเผยแพร่ไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## 8 การ Print : การพิมพ์จะเอาผลของ result มาพิมพ์



รูปที่ 3.30 รูปภาพแสดงหน้าต่างการพิมพ์

## 9 การ Save & Load : การ Save ต้องผ่านทาง Print เพราะจะรายงานเพียงผลเท่านั้น



เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์ของ บริษัท เทคโนโลยีสารสนเทศ จำกัด เมื่ออนุญาตให้ผู้อื่นไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 3.3 แผนการทดสอบ

หลังจากที่ได้สร้าง โปรแกรมสำหรับติดต่อสั่งงานระหว่าง Com กับ PLC การตรวจสอบโปรแกรมเพื่อหาจุดบกพร่องได้ทำตามลำดับแผนการดังนี้

#### 3.3.1 ตรวจสอบการสั่งงาน

การสั่งงานจากสวิตช์สามารถตรวจสอบโดยการกดสวิตช์แล้วที่ Input ของ PLC. ไฟแสดงสถานะ การทำงานจะติดการสั่งงานจากโปรแกรมหลักสามารถตรวจจากไฟแสดงสถานะ การทำงานจะติดที่ Output ของ PLC.

#### 3.3.2 ตรวจสอบการอ่านข้อมูล

การอ่านข้อมูลจากPLC. โดยโปรแกรมหลักสามารถตรวจสอบโดยการเทียบระหว่างไฟแสดงสถานะ การทำงานจะติดที่ Output ของ PLC.กับกราฟฟิคที่หน้าจอคอมพิวเตอร์

#### 3.3.3 ตรวจสอบการทำงานประสานกับโปรแกรม Ladder Diagram

การทำงานของ PLC. จะต้องทำงานตรงกับLadder Diagram ที่ได้ทำการออกแบบไว้สามารถตรวจสอบ โดยการเปรียบเทียบไฟแสดงสถานะการทำงานระหว่าง Input กับ Output

#### 3.3.4 ตรวจสอบการจัดเก็บข้อมูลจาก Datamemory

ต้องทำการเปรียบเทียบค่าที่กำหนดให้กับ Datamemory กับ ข้อมูลที่จัดเก็บจากโปรแกรมหลัก

## บทที่ 4

### ผลการดำเนินงาน

#### 4.1 ตารางเวลา

ในการดำเนินการได้ทำตามแผนการที่วางไว้โดยจะแสดงช่วงเวลาการทำงานไว้ในตารางเวลาโดยจะแสดงตามตารางที่ 4.1

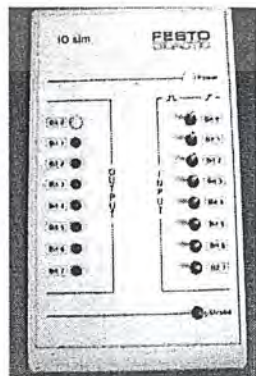
ตารางที่ 4.1 แสดงตารางเวลาการทำงาน

ID	Task Name	Start	End	Duration	2000							2001					
					Jan	Feb	Aug	Sep	Oct	Nov	Dec	Jan	Feb	Mar			
1	ศึกษาข้อมูลเกี่ยวกับโปรแกรม	30/5/00	21/8/00	12w													
2	ศึกษาหาข้อมูลเกี่ยวกับ PLC	3/7/00	25/8/00	8w													
3	หาตัวอย่างที่ใช้กับ PLC	3/7/00	25/8/00	8w													
4	สร้างโปรแกรมส่วน PLC-INTERFACE	7/8/00	29/9/00	8w													
5	สร้างโปรแกรมส่วน USER-INTERFACE	7/9/00	1/11/00	8w													
6	สร้างโปรแกรมส่วน ฐานข้อมูล	4/10/00	19/12/00	11w													
7	ทำการเพิ่มและรวมโปรแกรม	28/11/00	19/2/01	12w													
8	ทดลองและแก้ไขโปรแกรม	11/8/00	1/3/01	29w													
9	เรียบเรียง วิทยานิพนธ์	19/2/01	13/4/01	8w													

#### 4.2 ผลการทดสอบโปรแกรม

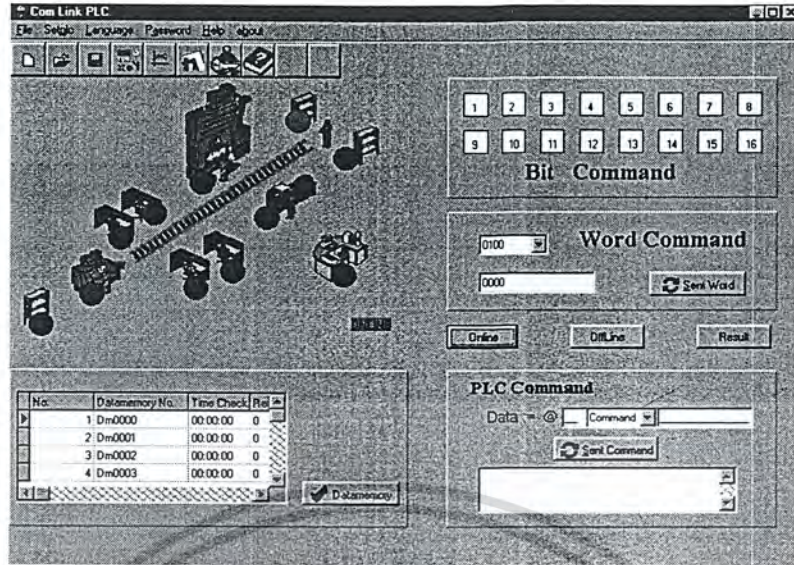
##### 4.2.1 ผลการตรวจสอบการสั่งงานและผลการตรวจสอบการอ่าน

โปรแกรมได้ทำการทดสอบการสั่งงาน โดยการสั่งของโปรแกรมเพื่อให้ตำแหน่งของ Output ทำการปิดหน้าคอนเทคต์และโปรแกรมได้ทำการอ่านข้อมูลกลับมาแสดงที่ตัวโปรแกรมดังจะเห็นได้จากรูปที่ 4.2 และการคิดของหลอดไฟดังรูปที่ 4.1



รูปที่ 4.1 รูปภาพแสดงการเปลี่ยนแปลงที่เกิดขึ้นของหน้าคอนเทคต์

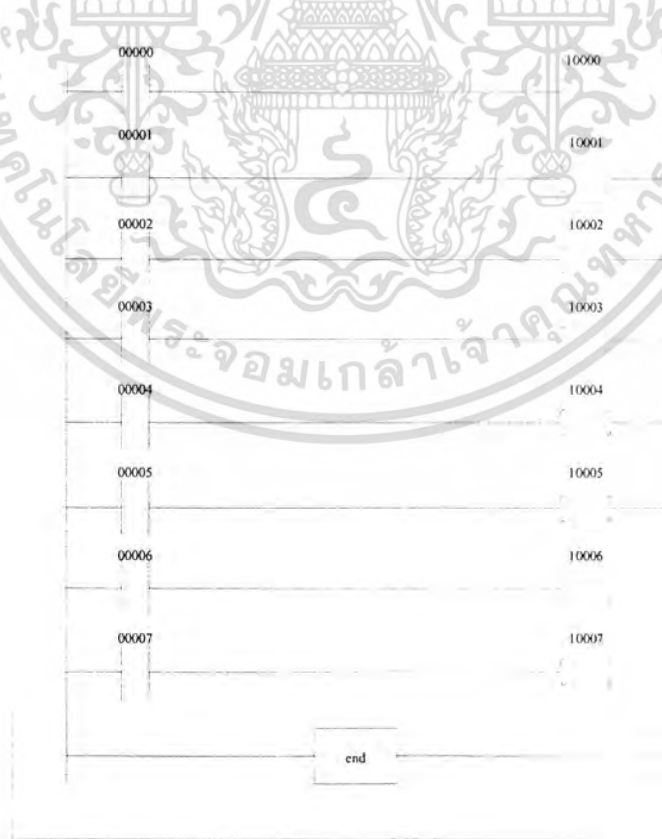
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับบริการเชิงงานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.2 รูปภาพแสดงการทำงานและการอ่านข้อมูล

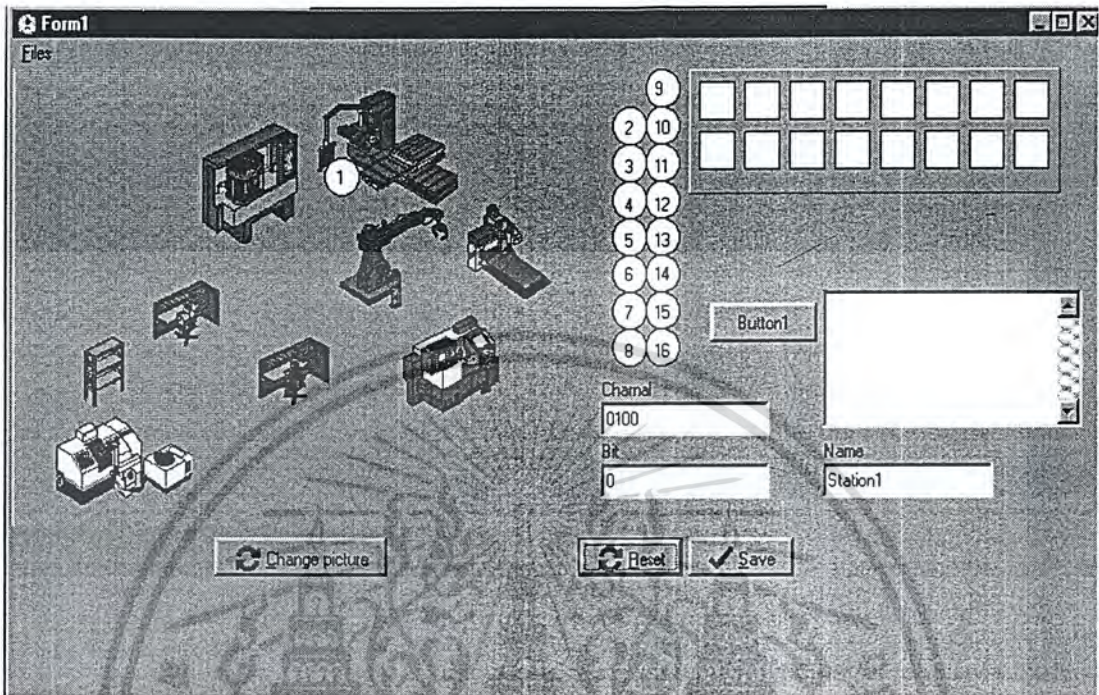
#### 4.2.2 ผลการตรวจสอบการทำงานประสานกับโปรแกรม Ladder Diagram

ได้มีการออกแบบ Ladder Diagram สำหรับการทดสอบและจำลองการทำงานของ PLC. ไว้ดังรูปที่ 4.3

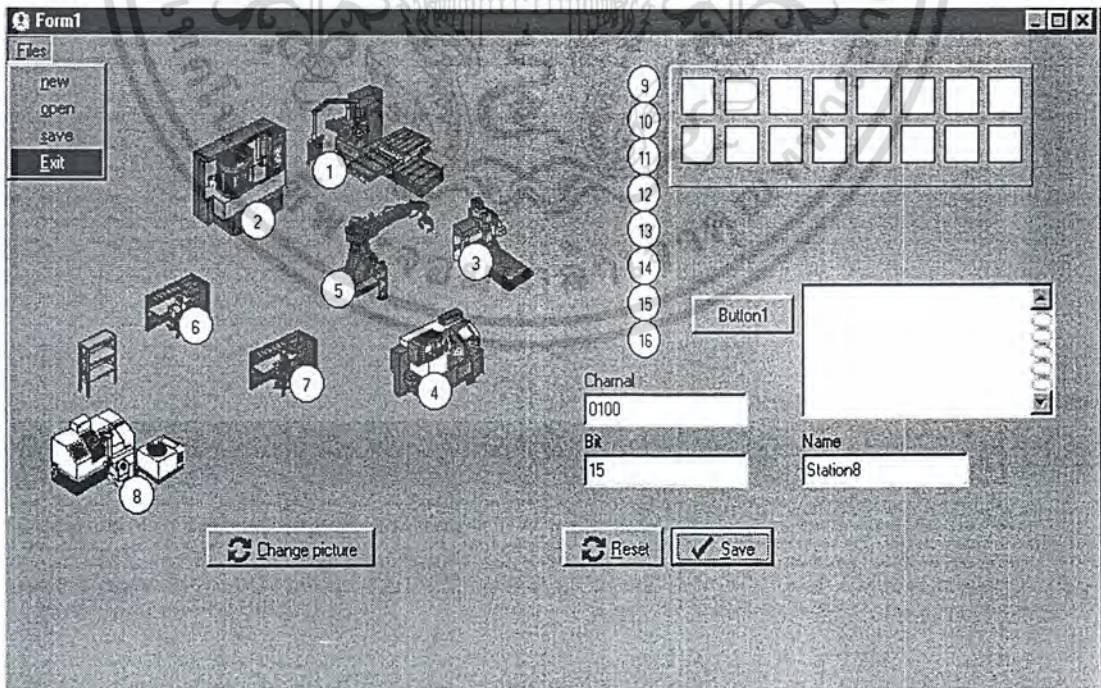


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับรูปที่ 4.3 รูปภาพแสดงการออกแบบ Ladder Diagram หน้าไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ทำการสร้างแบบจำลองสายการผลิตไว้ในโปรแกรมติดตั้งและเป็นการสร้างฐานข้อมูลสำหรับการติดต่อระหว่างคอมพิวเตอร์กับ PLC. ดังรูปที่ 4.4 และรูปที่ 4.5



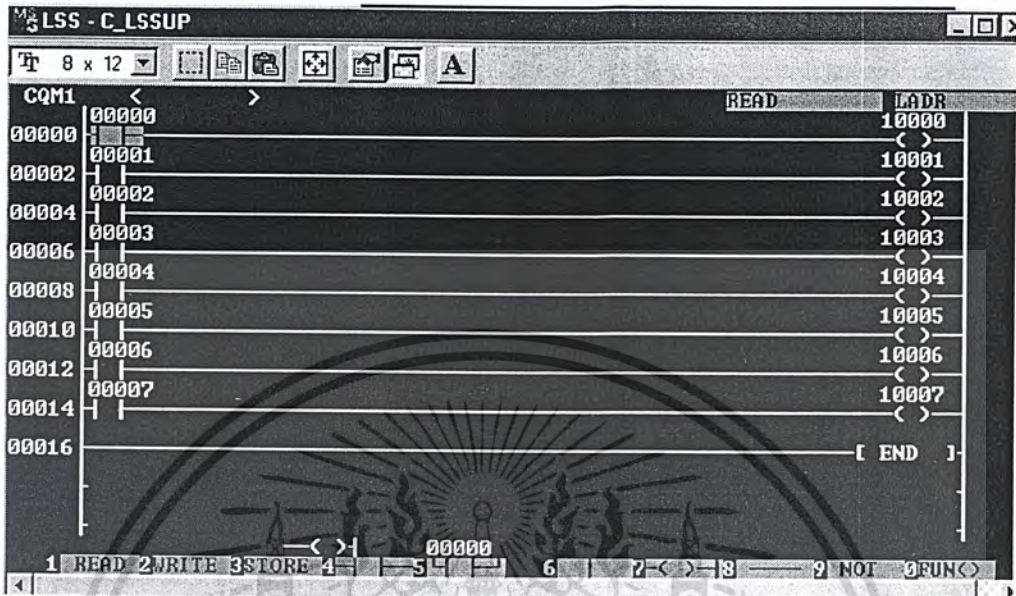
รูปที่ 4.4 รูปภาพแสดงการออกแบบ Interface โดย โปรแกรมติดตั้ง



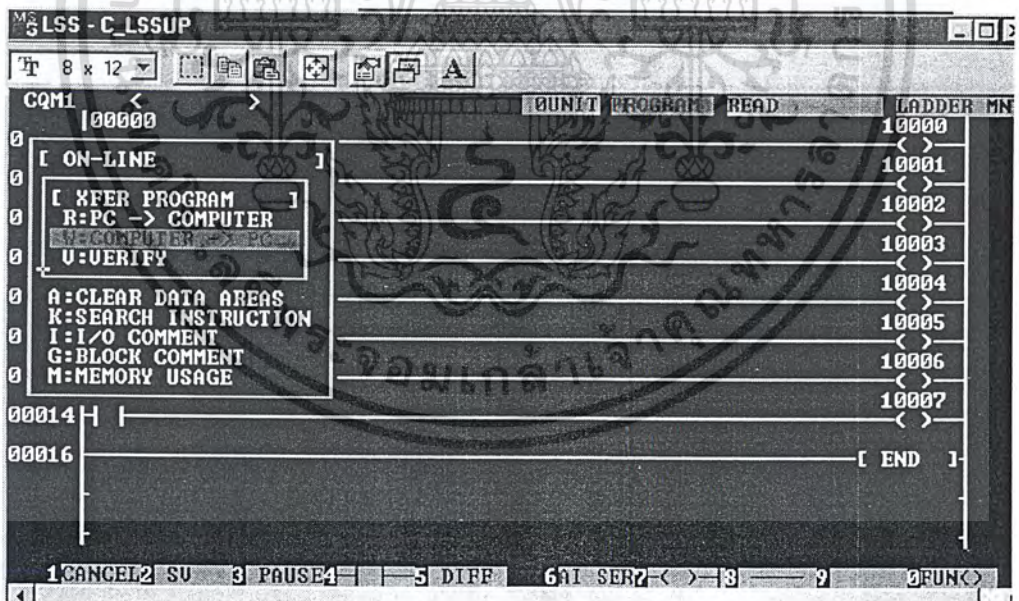
รูปที่ 4.5 รูปภาพแสดงการออกแบบ Interface เสร็จและสร้างฐานข้อมูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ทำการเขียนโปรแกรม Ladder Diagram ไปยัง PLC. โดยผ่านโปรแกรม LSS. เพื่อให้สามารถทำงานตามลำดับขั้นของ Ladder Diagram ดังจะเห็นผลได้จากโปรแกรม LSS. ดังรูปที่ 4.6 และรูปที่ 4.7



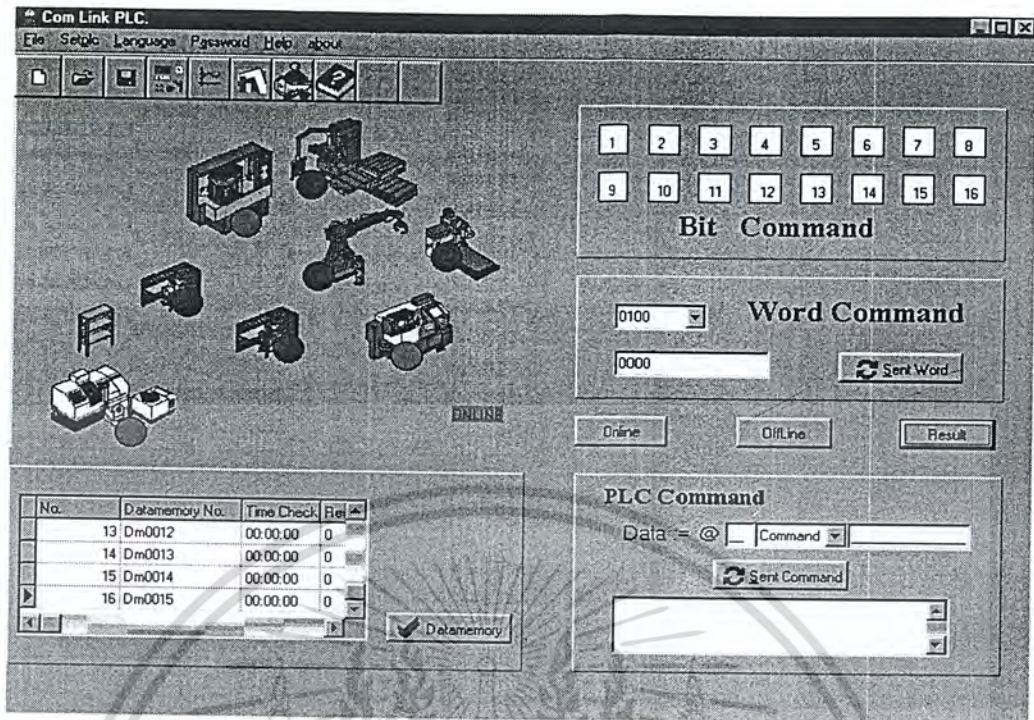
รูปที่ 4.6 รูปภาพแสดงการโปรแกรม LSS.



รูปที่ 4.7 รูปภาพแสดงการถ่ายโปรแกรมสู่ PLC.

ทำการเชื่อมต่อโปรแกรมหลัก กับ PLC. เพื่อตรวจสอบการทำงานตามขั้นตอนของ Ladder Diagram ดังจะเห็นผลได้จากโปรแกรมหลักดังรูปที่ 4.8

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

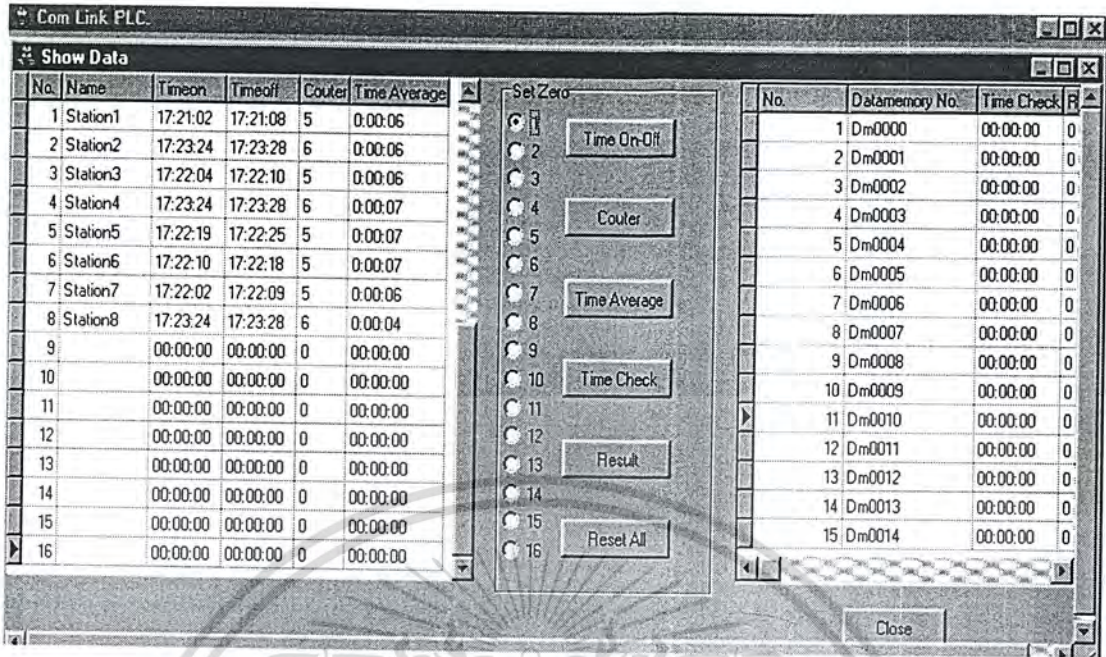


รูปที่ 4.8 รูปภาพแสดงการเปลี่ยนแปลงของ โปรแกรมหลักตามลำดับขั้นตอนของ Ladder Diagram

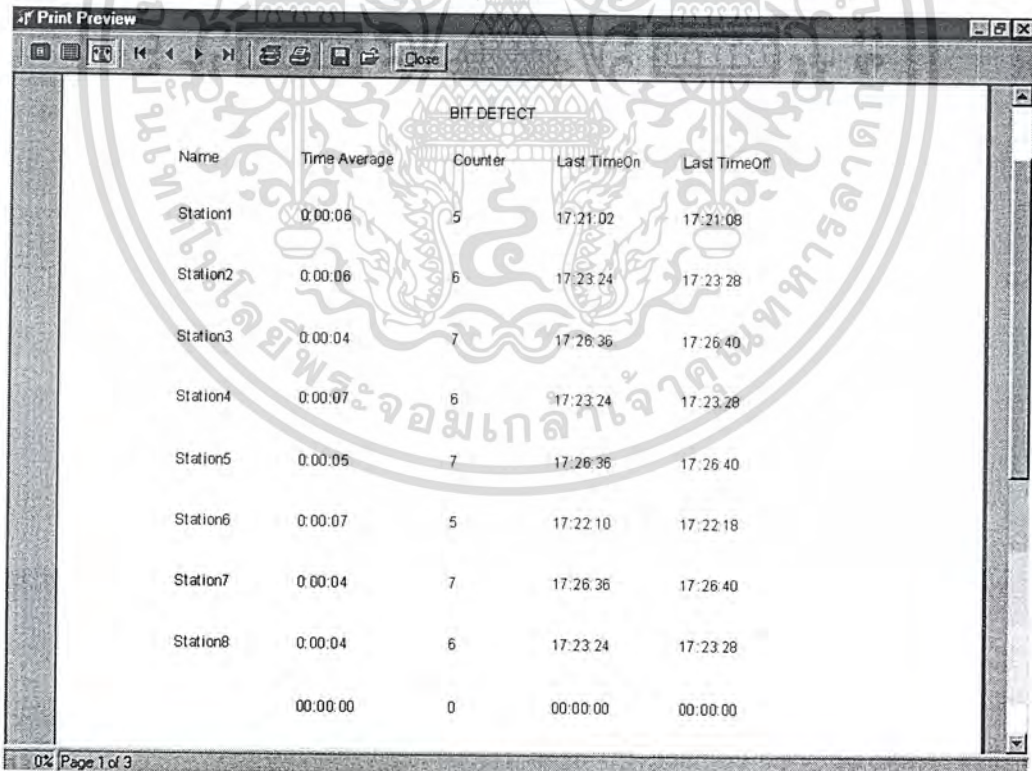
#### 4.2.3 ผลการตรวจสอบการจับเก็บข้อมูลออกจาก DataMemory

การอ่านค่า DataMemoryทำได้โดยการกดปุ่มสั่งงานจากโปรแกรมหลักเพื่อให้ข้อมูลจาก DataMemory ถ่ายโอนมาจับเก็บยังระบบฐานข้อมูล หลังจากทำการทดสอบตามข้อ 4.2.2 ตัวโปรแกรมหลักได้ทำการเก็บข้อมูล การสั่งงานและการเปิดปิดของอินพุทและเอาต์พุทและ DataMemory ดังรูปที่ 4.9 เพื่อการวิเคราะห์ทางด้าน เวลาหรือความถี่การทำงาน โดยข้อมูลเหล่านี้สามารถทำการพิมพ์ออกเป็นรายงานดังรูปที่ 4.10รวมทั้งการจับเก็บ ฐานข้อมูลข้อมูลในรูปแบบแฟ้มตัวอักษร (Text File) ดังรูปที่ 4.11

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

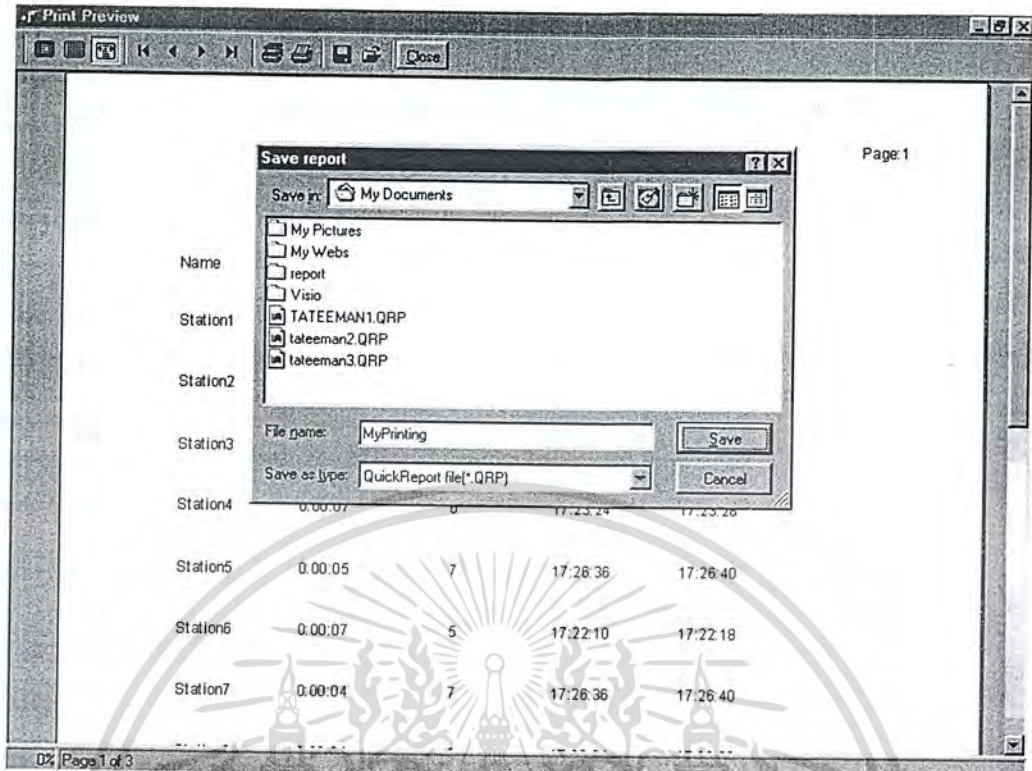


รูปที่4.9 รูปภาพแสดงผลการเปลี่ยนแปลงของข้อมูล



รูปที่4.10 รูปภาพแสดงการพิมพ์รายงาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.11 ภาพแสดงการเก็บรายงานในรูปแบบเพิ่มตัวอักษร

### 4.3 ปัญหาที่พบและวิธีแก้ไข

#### 4.3.1 ปัญหาที่พบ

- โปรแกรม Delphi ไม่มี Component สำหรับการสื่อสารออก Port
- การขาดแคลนอุปกรณ์ PLC. ในการทำงาน
- การใช้เวลาสำหรับการออกแบบความสัมพันธ์ระหว่าง โปรแกรมหลักและโปรแกรมติดตั้ง
- การตั้งงาน โดย Protocol Host-Link ไม่มีการตั้งงานแบบ Bit
- ต้องมีการออกแบบวงจรที่ใช้ต่อร่วมกับ PLC.

#### 4.3.2 วิธีแก้ไข

- แก้ปัญหาโดยการ Download Component Comport มาใช้งาน
- ได้ทำการสั่งซื้อ PLC รุ่น CPM1
- ขอคำแนะนำสำหรับการออกแบบจากอาจารย์ที่ปรึกษา
- ได้ทำการสร้าง โปรแกรมในการแปลงระบบเลขฐานให้อยู่ในรูปแบบที่สามารถตั้งงานได้
- ทำการออกแบบวงจร โดยดูตัวอย่างจากคู่มือกำกับของ PLC.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## บทที่ 5

### สรุปและอภิปราย

#### 5.1 สรุปผลการทำงาน

โครงการที่ได้ดำเนินการสร้างโปรแกรมสั่งงานและแสดงสถานะของ PLC ผ่านคอมพิวเตอร์เพื่อทดแทนระบบการสั่งงาน โดยการกดสวิตช์หรือการสร้างแผงสวิตช์ควบคุมรวมทั้งเพิ่มเติมความสามารถในการจัดเก็บข้อมูลของการทำงาน ทำให้เห็นรูปแบบของการเขียนโปรแกรมได้ชัดเจน

จากผลการทำงานของโปรแกรมแสดงให้เห็นถึงความสามารถของการสั่งงานจากโปรแกรมที่ได้เขียนขึ้น มีการตอบสนองได้เป็นอย่างดีระหว่างคอมพิวเตอร์กับ PLC.

#### 5.2 อภิปราย

โปรแกรมนี้มีข้อบกพร่องตรงไหนสามารถปรับปรุงในส่วนใดบ้าง

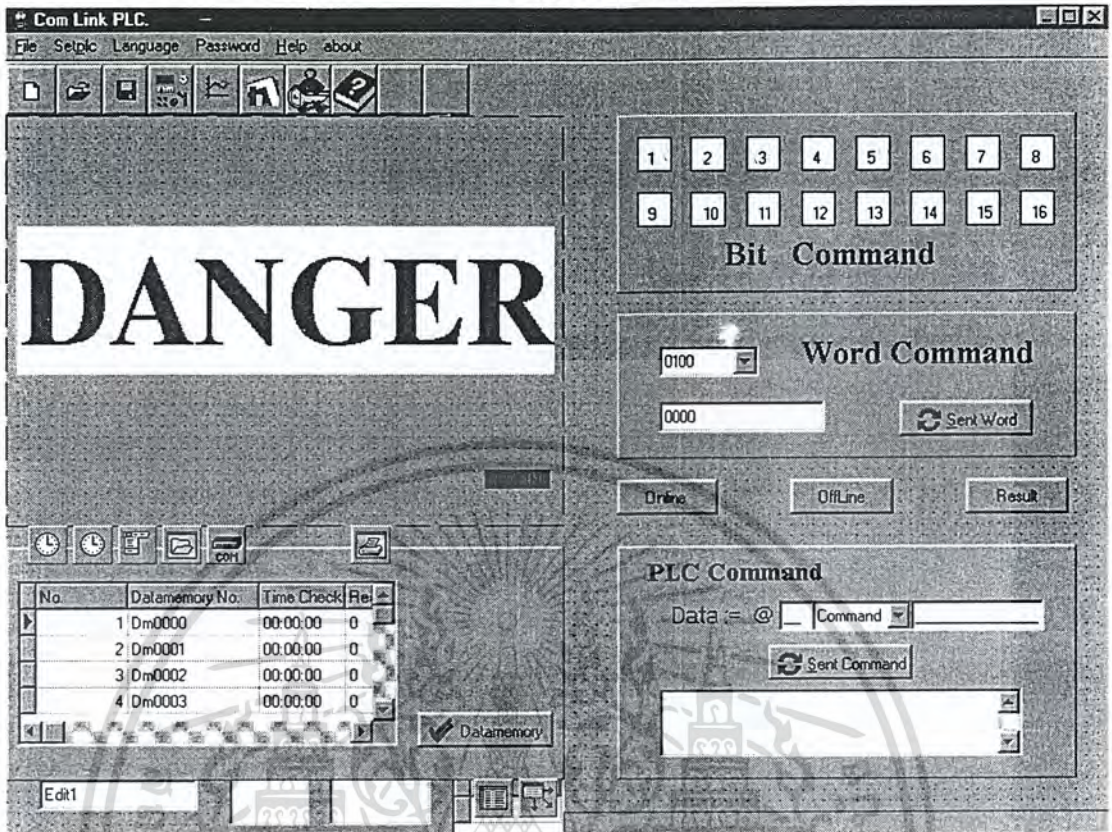
1. การใช้ Database ปรกติแล้ว 1 Record ของ Database จะสามารถเก็บข้อมูลได้หลายชุดข้อมูล แต่การเก็บข้อมูลของปุ่มและรหัสต่างๆภายใน โปรแกรมจะใช้เพียงแค่ Field เดียวสมารถพัฒนาต่อไปในอนาคตได้
2. โปรแกรมจะใช้ Database มากกว่า 1 ตัวในการเก็บค่าต่างๆควรใช้ Database ให้น้อยที่สุด
3. เพิ่มโปรแกรมเสริมการทำงาน เช่น การวิเคราะห์ Ladder Diagrams
4. เพิ่มจำนวน Bit ในการแสดงและการสั่งงาน
5. เพิ่มเติมความสามารถในการทำงานแบบ PID
6. เพิ่มเติมความสามารถในการทำงานกับ Servo-Motor
7. ปรับปรุง User Interface ให้มีรูปแบบกระชับและสวยงามมากขึ้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

\*\*\*\*\*Tform1(ComLinkPLC.)\*\*\*\*\*



รูปที่ ๘1 ภาพแสดง Form ที่ 1 ของโปรแกรม ComLinkPLC.

unit Unit1;

interface

uses

Windows, Messages, SysUtils, Classes, Graphics, Controls, Forms, Dialogs,  
ExtCtrls, Grids, DBGrids, Db, DBTables, Menus, jpeg, StdCtrls, Buttons,  
ComPort, ComCtrls, ToolWin, ImgList, BkGround, Mask;

type

TForm1 = class(TForm)

MainMenu1: TMainMenu;

DataSource1: TDataSource;

Table1: TTable;

DBGrid1: TDBGrid;

Image1: TImage;

OpenDialog1: TOpenDialog;

File1: TMenuItem;

Exit1: TMenuItem;

N1: TMenuItem;

Print1: TMenuItem;

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

N2: TMenuItem;  
SaveAs1: TMenuItem;  
Save1: TMenuItem;  
Open1: TMenuItem;  
New1: TMenuItem;  
Button1: TButton;  
Button2: TButton;  
Panel1: TPanel;  
Shape100: TShape;  
Shape101: TShape;  
Shape102: TShape;  
Shape103: TShape;  
Shape104: TShape;  
Shape105: TShape;  
Shape106: TShape;  
Shape107: TShape;  
Label1: TLabel;  
Panel2: TPanel;  
BitBtn1: TBitBtn;  
Label2: TLabel;  
Label3: TLabel;  
PrintPreview1: TMenuItem;  
setp1: TMenuItem;  
Help1: TMenuItem;  
about1: TMenuItem;  
Langure1: TMenuItem;  
thai: TMenuItem;  
ComPort1: TComPort;  
omron1: TMenuItem;  
cqm1: TMenuItem;  
Property1: TMenuItem;  
english1: TMenuItem;  
Edit1: TEdit;  
Timer1: TTimer;  
ProgressBar1: TProgressBar;  
ToolBar1: TToolBar;  
SpeedButton1: TSpeedButton;



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับครูใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

SpeedButton2: TSpeedButton;  
SpeedButton3: TSpeedButton;  
SpeedButton4: TSpeedButton;  
SpeedButton5: TSpeedButton;  
SpeedButton6: TSpeedButton;  
SpeedButton7: TSpeedButton;  
SpeedButton8: TSpeedButton;  
SpeedButton9: TSpeedButton;  
SpeedButton10: TSpeedButton;  
password1: TMenuItem;  
newuser1: TMenuItem;  
EditPassword1: TMenuItem;  
DeleatPassword1: TMenuItem;  
Memo1: TMemo;  
PrintDialog1: TPrintDialog;  
Shape108: TShape;  
Shape109: TShape;  
Shape110: TShape;  
Shape111: TShape;  
Shape112: TShape;  
Shape113: TShape;  
Shape114: TShape;  
Shape115: TShape;  
Edit2: TEdit;  
ComboBox1: TComboBox;  
Button3: TButton;  
Label4: TLabel;  
Timer2: TTimer;  
Memo2: TMemo;  
bc1: TLabel;  
bc2: TLabel;  
Label5: TLabel;  
Label6: TLabel;  
Label7: TLabel;  
Label8: TLabel;  
Label9: TLabel;  
Label10: TLabel;



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

Label11: TLabel;
Label12: TLabel;
Label13: TLabel;
Label14: TLabel;
Label15: TLabel;
Label16: TLabel;
Label17: TLabel;
Label18: TLabel;
Panel3: TPanel;
DBGrid2: TDBGrid;
BitBtn2: TBitBtn;
Panel4: TPanel;
Label19: TLabel;
Label20: TLabel;
MaskEdit1: TMaskEdit;
ComboBox2: TComboBox;
MaskEdit2: TMaskEdit;
BitBtn3: TBitBtn;
Memo3: TMemo;
procedure FormCreate(Sender: TObject);
procedure Exit1Click(Sender: TObject);
procedure Button2Click(Sender: TObject);
procedure about1Click(Sender: TObject);
procedure thai1Click(Sender: TObject);
procedure english1Click(Sender: TObject);
procedure Button1Click(Sender: TObject);
procedure Timer1Timer(Sender: TObject);
procedure Shape100MouseDown(Sender: TObject; Button: TMouseButton;
  Shift: TShiftState; X, Y: Integer);
procedure Shape101MouseDown(Sender: TObject; Button: TMouseButton;
  Shift: TShiftState; X, Y: Integer);
procedure Shape102MouseDown(Sender: TObject; Button: TMouseButton;
  Shift: TShiftState; X, Y: Integer);
procedure Shape103MouseDown(Sender: TObject; Button: TMouseButton;
  Shift: TShiftState; X, Y: Integer);
procedure Shape104MouseDown(Sender: TObject; Button: TMouseButton;
  Shift: TShiftState; X, Y: Integer);

```

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์ไว้เพื่อการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

procedure Shape105MouseDown(Sender: TObject; Button: TMouseButton;
  Shift: TShiftState; X, Y: Integer);
procedure Shape106MouseDown(Sender: TObject; Button: TMouseButton;
  Shift: TShiftState; X, Y: Integer);
procedure Shape107MouseDown(Sender: TObject; Button: TMouseButton;
  Shift: TShiftState; X, Y: Integer);
procedure Shape108MouseDown(Sender: TObject; Button: TMouseButton;
  Shift: TShiftState; X, Y: Integer);
procedure Shape109MouseDown(Sender: TObject; Button: TMouseButton;
  Shift: TShiftState; X, Y: Integer);
procedure Shape110MouseDown(Sender: TObject; Button: TMouseButton;
  Shift: TShiftState; X, Y: Integer);
procedure Shape111MouseDown(Sender: TObject; Button: TMouseButton;
  Shift: TShiftState; X, Y: Integer);
procedure Shape112MouseDown(Sender: TObject; Button: TMouseButton;
  Shift: TShiftState; X, Y: Integer);
procedure Shape113MouseDown(Sender: TObject; Button: TMouseButton;
  Shift: TShiftState; X, Y: Integer);
procedure Shape114MouseDown(Sender: TObject; Button: TMouseButton;
  Shift: TShiftState; X, Y: Integer);
procedure Shape115MouseDown(Sender: TObject; Button: TMouseButton;
  Shift: TShiftState; X, Y: Integer);
procedure BitBtn1Click(Sender: TObject);
procedure Button3Click(Sender: TObject);
procedure Open1Click(Sender: TObject);
procedure Print1Click(Sender: TObject);
procedure PrintPreview1Click(Sender: TObject);
procedure Property1Click(Sender: TObject);
procedure cqm1Click(Sender: TObject);
procedure Timer2Timer(Sender: TObject);
procedure BitBtn2Click(Sender: TObject);
procedure BitBtn3Click(Sender: TObject);
private
  { Private declarations }
public
  { Public declarations }

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
 end,  
 ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

var
  Form1: TForm1;
shape1,shape2,shape3,shape4,shape5,shape6,shape7,shape8,shape9,shape10,shape11,shape12,shape13,shape14,
shape15,shape16:tshape;
  chanal,bit : array[1..32]of string;
  inpstring,recieve,onebitsend1,onebitsend2 :string;
  juvenus :integer;
  onebitchange : boolean;
procedure ake (btn:integer;onoff:boolean);
PROCEDURE TATEEMAN(STR:STRING;str2:string);
procedure data_rec_avr (btn:string);
function dectohex(charinp:INTEGER):CHAR;//yyy
function hextodec(charinp:char):integer;//xxx
FUNCTION STATUSBIT(ST11:STRING) :INTEGER;
implementation
uses unit2, unit3 ,unit4, Unit5,unit6;
{$R *.DFM}

procedure TForm1.FormCreate(Sender: TObject);
var i,j :integer;
  Filename1,a :string;
begin
  j:=1;Filename1:="";
  JUVENTUS:=1;
  table1.Active:=false;
  memo2.Lines.LoadFromFile('c:\name.txt');
  a:=memo2.Lines.Text;
  while a[j]<>'*' do
  begin
    filename1:=filename1+a[j];
    j:=j+1;
  end;
  //table1.TableName:=filename1;
  table1.TableName:='C:\databaseake\dbproject.DB';
  table1.Active:=true;
  if not((table1.FieldByName('1b').asinteger = 0)and

```

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์ไว้เพื่อใช้ในการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

        (table1.FieldByName('1c').asinteger = 0))
then
begin
    shape1:=tshape.Create(self);
    shape1.Parent := self;
    shape1.top:= table1.FieldByName('1t').asinteger+40;
    shape1.left:=table1.FieldByName('1l').asinteger;
    shape1.shape :=stcircle;
    shape1.Width := 25;
    shape1.Height:= 25;
end;
if not((table1.FieldByName('2b').asinteger = 0)and
        (table1.FieldByName('2c').asinteger = 0))
then
begin
    shape2:=tshape.Create(self);
    shape2.Parent := self;
    shape2.top:= table1.FieldByName('2t').asinteger+40;
    shape2.left:=table1.FieldByName('2l').asinteger;
    shape2.shape :=stcircle;
    shape2.Width := 25;
    shape2.Height:= 25;
end;
if not((table1.FieldByName('3b').asinteger = 0)and
        (table1.FieldByName('3c').asinteger = 0))
then
begin
    shape3:=tshape.Create(self);
    shape3.Parent := self;
    shape3.top:= table1.FieldByName('3t').asinteger+40;
    shape3.left:=table1.FieldByName('3l').asinteger;
    shape3.shape :=stcircle;
    shape3.Width := 25;
    shape3.Height:= 25;
end;
if not((table1.FieldByName('4b').asinteger = 0)and
        (table1.FieldByName('4c').asinteger = 0))

```



เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์ไว้เพื่อใช้ในการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
 ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

then
begin
shape4:=tshape.Create(self);
shape4.Parent := self;
shape4.top:= table1.FieldByName('4t').asinteger+40;
shape4.left:=table1.FieldByName('4l').asinteger;
shape4.shape :=stcircle;
shape4.Width := 25;
shape4.Height:= 25;
end;
if not((table1.FieldByName('5b').asinteger = 0)and
(table1.FieldByName('5c').asinteger = 0))
then
begin
shape5:=tshape.Create(self);
shape5.Parent := self;
shape5.top:= table1.FieldByName('5t').asinteger+40;
shape5.left:=table1.FieldByName('5l').asinteger;
shape5.shape :=stcircle;
shape5.Width := 25;
shape5.Height:= 25;
end;
if not((table1.FieldByName('6b').asinteger = 0)and
(table1.FieldByName('6c').asinteger = 0))
then
begin
shape6:=tshape.Create(self);
shape6.Parent := self;
shape6.top:= table1.FieldByName('6t').asinteger+40;
shape6.left:=table1.FieldByName('6l').asinteger;
shape6.shape :=stcircle;
shape6.Width := 25;
shape6.Height:= 25;
end;
if not((table1.FieldByName('7b').asinteger = 0)and
(table1.FieldByName('7c').asinteger = 0))

```



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
 ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

begin
shape7:=tshape.Create(self);
shape7.Parent := self;
shape7.top:= table1.FieldByName('7t').asinteger+40;
shape7.left:=table1.FieldByName('7l').asinteger;
shape7.shape :=stcircle;
shape7.Width := 25;
shape7.Height:= 25;
end;
if not((table1.FieldByName('8b').asinteger = 0)and
(table1.FieldByName('8c').asinteger = 0))
then
begin
shape8:=tshape.Create(self);
shape8.Parent := self;
shape8.top:= table1.FieldByName('8t').asinteger+40;
shape8.left:=table1.FieldByName('8l').asinteger;
shape8.shape :=stcircle;
shape8.Width := 25;
shape8.Height:= 25;
end;
if not((table1.FieldByName('9b').asinteger = 0)and
(table1.FieldByName('9c').asinteger = 0))
then
begin
shape9:=tshape.Create(self);
shape9.Parent := self;
shape9.top:= table1.FieldByName('9t').asinteger+40;
shape9.left:=table1.FieldByName('9l').asinteger;
shape9.shape :=stcircle;
shape9.Width := 25;
shape9.Height:= 25;
end;
if not((table1.FieldByName('10b').asinteger = 0)and
(table1.FieldByName('10c').asinteger = 0))
then

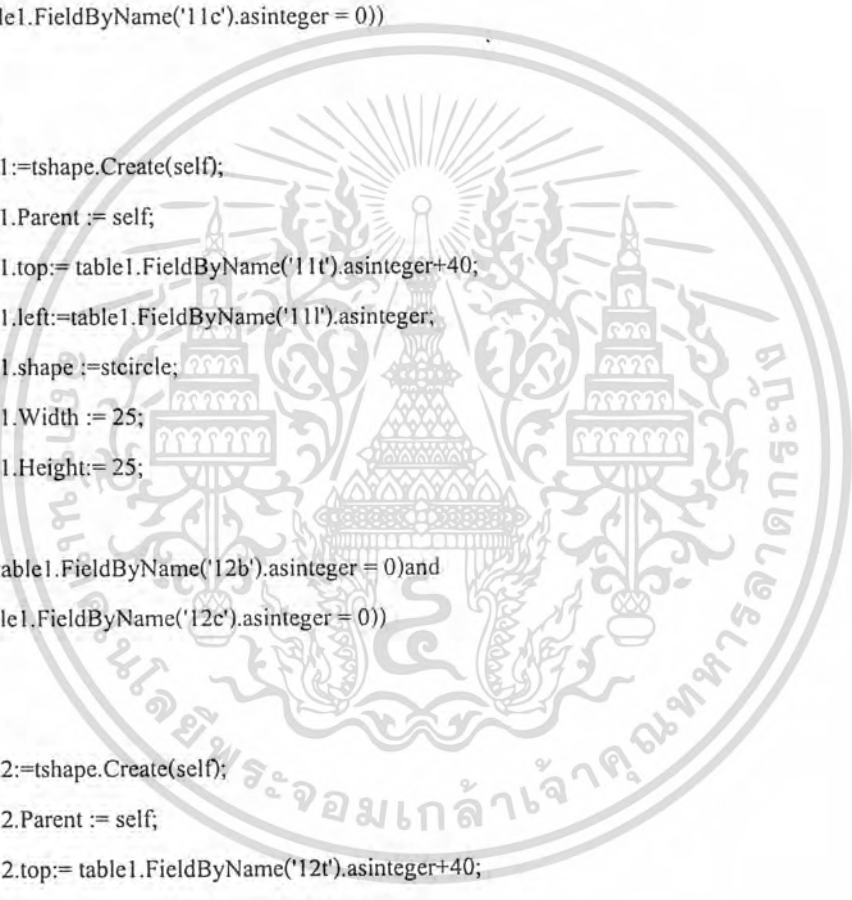
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

shape10:=tshape.Create(self);
shape10.Parent := self;
shape10.top:= table1.FieldByName('10t').asinteger+40;
shape10.left:=table1.FieldByName('10l').asinteger;
shape10.shape :=stcircle;
shape10.Width := 25;
shape10.Height:= 25;
end;
if not((table1.FieldByName('11b').asinteger = 0)and
(table1.FieldByName('11c').asinteger = 0))
then
begin
shape11:=tshape.Create(self);
shape11.Parent := self;
shape11.top:= table1.FieldByName('11t').asinteger+40;
shape11.left:=table1.FieldByName('11l').asinteger;
shape11.shape :=stcircle;
shape11.Width := 25;
shape11.Height:= 25;
end;
if not((table1.FieldByName('12b').asinteger = 0)and
(table1.FieldByName('12c').asinteger = 0))
then
begin
shape12:=tshape.Create(self);
shape12.Parent := self;
shape12.top:= table1.FieldByName('12t').asinteger+40;
shape12.left:=table1.FieldByName('12l').asinteger;
shape12.shape :=stcircle;
shape12.Width := 25;
shape12.Height:= 25;
end;
if not((table1.FieldByName('13b').asinteger = 0)and
(table1.FieldByName('13c').asinteger = 0))
then
begin
shape13:=tshape.Create(self);

```



เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์การใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

shape13.Parent := self;
shape13.top:= table1.FieldByName('13t').asinteger+40;
shape13.left:=table1.FieldByName('13l').asinteger;
shape13.shape :=stcircle;
shape13.Width := 25;
shape13.Height:= 25;
end;
if not((table1.FieldByName('14b').asinteger = 0)and
      (table1.FieldByName('14c').asinteger = 0))
then
begin
shape14:=tshape.Create(self);
shape14.Parent := self;
shape14.top:= table1.FieldByName('14t').asinteger+40;
shape14.left:=table1.FieldByName('14l').asinteger;
shape14.shape :=stcircle;
shape14.Width := 25;
shape14.Height:= 25;
end;
if not((table1.FieldByName('15b').asinteger = 0)and
      (table1.FieldByName('15c').asinteger = 0))
then
begin
shape15:=tshape.Create(self);
shape15.Parent := self;
shape15.top:= table1.FieldByName('15t').asinteger+40;
shape15.left:=table1.FieldByName('15l').asinteger;
shape15.shape :=stcircle;
shape15.Width := 25;
shape15.Height:= 25;
end;
if not((table1.FieldByName('16b').asinteger = 0)and
      (table1.FieldByName('16c').asinteger = 0))
then
begin
shape16:=tshape.Create(self);
shape16.Parent := self;

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

shape16.top:= table1.FieldByName('16t').asinteger+40;
shape16.left:=table1.FieldByName('16l').asinteger;
shape16.shape :=stcircle;
shape16.Width := 25;
shape16.Height:= 25;
end;
for i:= 1 to 16 do
begin
chanal[i]:=table1.fieldbyname(inttostr(i)+'c').asstring;
bit[i]:=table1.fieldbyname(inttostr(i)+'b').asstring;
chanal[i+16]:=table1.fieldbyname(inttostr(i+99)+'c').asstring;
bit[i+16]:=table1.fieldbyname(inttostr(i+99)+'b').asstring;
end;
shape1.Hint:=table1.fieldbyname('N1').asstring;
shape2.Hint:=table1.fieldbyname('N2').asstring;
shape3.Hint:=table1.fieldbyname('N3').asstring;
shape4.Hint:=table1.fieldbyname('N4').asstring;
shape5.Hint:=table1.fieldbyname('N5').asstring;
shape6.Hint:=table1.fieldbyname('N6').asstring;
shape7.Hint:=table1.fieldbyname('N7').asstring;
shape8.Hint:=table1.fieldbyname('N8').asstring;
shape9.Hint:=table1.fieldbyname('N9').asstring;
shape10.Hint:=table1.fieldbyname('N10').asstring;
shape11.Hint:=table1.fieldbyname('N11').asstring;
shape12.Hint:=table1.fieldbyname('N12').asstring;
shape13.Hint:=table1.fieldbyname('N13').asstring;
shape14.Hint:=table1.fieldbyname('N14').asstring;
shape15.Hint:=table1.fieldbyname('N15').asstring;
shape16.Hint:=table1.fieldbyname('N16').asstring;
shape1.showhint:=true;
shape2.showhint:=true;
shape3.showhint:=true;
shape4.showhint:=true;
shape5.showhint:=true;
shape6.showhint:=true;
shape7.showhint:=true;
shape8.showhint:=true;

```

เอกสารนี้เป็นเอกสารที่จัดทำขึ้นสำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
shape9.showhint:=true;
shape10.showhint:=true;
shape11.showhint:=true;
shape12.showhint:=true;
shape13.showhint:=true;
shape14.showhint:=true;
shape15.showhint:=true;
shape16.showhint:=true;
shape100.Hint:='chanal '+chanal[17]+' bit '+bit[17];
shape100.showhint:=true;
shape101.Hint:='chanal '+chanal[18]+' bit '+bit[18];
shape101.showhint:=true;
shape102.Hint:='chanal '+chanal[19]+' bit '+bit[19];
shape102.showhint:=true;
shape103.Hint:='chanal '+chanal[20]+' bit '+bit[20];
shape103.showhint:=true;
shape104.Hint:='chanal '+chanal[21]+' bit '+bit[21];
shape104.showhint:=true;
shape105.Hint:='chanal '+chanal[22]+' bit '+bit[22];
shape105.showhint:=true;
shape106.Hint:='chanal '+chanal[23]+' bit '+bit[23];
shape106.showhint:=true;
shape107.Hint:='chanal '+chanal[24]+' bit '+bit[24];
shape107.showhint:=true;
shape108.Hint:='chanal '+chanal[25]+' bit '+bit[25];
shape108.showhint:=true;
shape109.Hint:='chanal '+chanal[26]+' bit '+bit[26];
shape109.showhint:=true;
shape110.Hint:='chanal '+chanal[27]+' bit '+bit[27];
shape110.showhint:=true;
shape111.Hint:='chanal '+chanal[28]+' bit '+bit[28];
shape111.showhint:=true;
shape112.Hint:='chanal '+chanal[29]+' bit '+bit[29];
shape112.showhint:=true;
shape113.Hint:='chanal '+chanal[30]+' bit '+bit[30];
shape113.showhint:=true;
shape114.Hint:='chanal '+chanal[31]+' bit '+bit[31];
```



เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์ไว้สำหรับใช้ภายในเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

shape114.showhint:=true;
shape115.Hint:='chanal '+chanal[32]+' bit '+bit[32];
shape115.showhint:=true;
if table1.FieldByName('pic').asstring = " then
begin
  showmessage('please select picture in first program');
  form1.Close;
end
else
image1.Picture.LoadFromFile(table1.fieldbyname('pic').asstring);
end;

procedure TForm1.Exit1Click(Sender: TObject);
begin
  if MessageDlg('Do you want to save?',mtwarning,[mbok,mbcancel],0) = mrok then
  begin
  end;
  self.Close;
end;

procedure TForm1.Button2Click(Sender: TObject);
var i:integer;
begin
  datamodule3.table1.First;
  for i:= 1 to 16 do
  begin
    datamodule3.table1.Edit;
    datamodule3.Table1.FieldByName('Name').asstring:=table1.fieldbyname('N'+inttostr(i)).asstring;
    datamodule3.table1.post;
    datamodule3.table1.Next;
  end;
  form2.show;
end;

procedure TForm1.about1Click(Sender: TObject);
begin

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
form4.show;  
end;
```

```
procedure TForm1.thai1Click(Sender: TObject);
```

```
begin
```

```
file1.caption:='เพิ่ม';  
setplc1.caption:='เลือกชนิดปล';  
new1.Caption:='เริ่ม';  
open1.Caption:='เปิด';  
save1.Caption:='จัดเก็บ';  
saveas1.Caption:='จัดเก็บที่';  
exit1.Caption:='ออก';  
print1.Caption:='พิมพ์';  
printpreview1.Caption:='รูปก่อนพิมพ์';  
cqm1.Caption:='cqm';  
property1.Caption:='คุณสมบัติ';  
thai1.Caption:='ไทย';  
english1.Caption:='อังกฤษ';  
help1.Caption:='คำแนะนำ';  
about1.Caption:='ผู้จัดทำ';  
langure1.caption:='ภาษา';  
button1.Caption:='สื่อสาร';  
button3.Caption:='หยุดการสื่อสาร';  
button2.Caption:='แสดงผล';  
bitbtn1.Caption:='ส่งข้อมูล';  
end;
```

```
procedure TForm1.english1Click(Sender: TObject);
```

```
begin
```

```
file1.caption:='File';  
setplc1.caption:='Setplc';  
new1.Caption:='New';  
open1.Caption:='Caption';  
save1.Caption:='Save';  
saveas1.Caption:='Save as..';  
exit1.Caption:='Exit';
```

```
print1.Caption:='Print';
```

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์หรือการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

printpreview1.Caption:='Print Preview';
cqm1.Caption:='CQM';
property1.Caption:='Property';
thai1.Caption:='Thai';
english1.Caption:='English';
help1.Caption:='Help';
about1.Caption:='About';
langure1.caption:='Langure';
button1.Caption:='OnLine';
button3.Caption:='OffLine';
button2.Caption:='Result';
bitbtn1.Caption:='&Sent data';
end;

FUNCTION addFCS(STRINGINP:STRING):STRING;
VAR collect,collect2,collect3,c1,c2 : string;
    i :INTEGER;
    CH : CHAR;
BEGIN
    for i:=1 to length(stringinp) do
    begin
        collect:=collect+inttohex(ord(stringinp[i]),2);//เรียงลำดับตัวเลข
        collect2:=collect2+collect[(2*i)-1];//แยกตัวเลข
        collect3:=collect3+collect[2*i];//แยกตัวเลข
    end;
    c1:=collect2[1];
    c2:=collect3[1];
    for i:=2 to length(collect2) do
    begin
        c1:=inttohex(strtoint('$'+c1[1]) xor strtoint('$'+collect2[i],1);
        c2:=inttohex(strtoint('$'+c2[1]) xor strtoint('$'+collect3[i],1);
    end;
    CH:=chr(strtoint('$3'+c2[1]));
    CASE CH OF
        ':': CH:='A';
        ';': CH:='B';
        '<': CH:='C';
    end;
END;

```

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์ไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

    '=' : CH:='D';
    '>' : CH:='E';
    '?' : CH:='F';
END;
result:=(stringinp+chr(strtoint('$3'+c1[1]))+ch+'*');
end;
function hextodec(charinp:char):integer;
begin
case charinp of
'0' :result:=0;
'1' :result:=1;
'2' :result:=2;
'3' :result:=3;
'4' :result:=4;
'5' :result:=5;
'6' :result:=6;
'7' :result:=7;
'8' :result:=8;
'9' :result:=9;
'A' :result:=10;
'B' :result:=11;
'C' :result:=12;
'D' :result:=13;
'E' :result:=14;
'F' :result:=15;
END;
END;
function dectohex(charinp:INTEGER):CHAR;
begin
case charinp of
0 :result:='0';
1 :result:='1';
2 :result:='2';
3 :result:='3';
4 :result:='4';
5 :result:='5';
6 :result:='6';

```



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

7 :result:='7';
8 :result:='8';
9 :result:='9';
10 :result:='A';
11 :result:='B';
12 :result:='C';
13 :result:='D';
14 :result:='E';
15 :result:='F';
END;
END;
function bitstatus(intinp:integer) :string;
begin
case intinp of
0 : result:='0000';
1 : result:='1000';
2 : result:='0100';
3 : result:='1100';
4 : result:='0010';
5 : result:='1010';
6 : result:='0110';
7 : result:='1110';
8 : result:='0001';
9 : result:='1001';
10: result:='0101';
11: result:='1101';
12: result:='0011';
13: result:='1011';
14: result:='0111';
15: result:='1111';
end;
end;
FUNCTION STATUSBIT(ST11:STRING) :INTEGER;
BEGIN
IF ST11= '0000'THEN result:=0;
IF ST11= '1000'THEN result:=1;
IF ST11= '0100'THEN result:=2;

```



เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์และใช้ในงานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

IF ST11= '1100'THEN result:=3;
IF ST11= '0010'THEN result:=4;
IF ST11= '1010'THEN result:=5;
IF ST11= '0110'THEN result:=6;
IF ST11= '1110'THEN result:=7;
IF ST11= '0001'THEN result:=8;
IF ST11= '1001'THEN result:=9;
IF ST11= '0101'THEN result:=10;
IF ST11= '1101'THEN result:=11;
IF ST11= '0011'THEN result:=12;
IF ST11= '1011'THEN result:=13;
IF ST11= '0111'THEN result:=14;
IF ST11= '1111'THEN result:=15;
end;
procedure datarecord (btn:string;onoff:boolean);
begin
datamodule3.Table1.First;
if onoff then
begin
datamodule3.Table1.MoveBy(strtoint(btn)-1);
datamodule3.Table1.edit;
datamodule3.Table1.FieldName('Timeon').asString:=timetostr(time);

datamodule3.Table1.FieldName('Couter').asinteger:=datamodule3.Table1.FieldName('Couter').asinteger+
1;
end
else
begin
datamodule3.Table1.MoveBy(strtoint(btn)-1);
datamodule3.Table1.edit;
datamodule3.Table1.FieldName('Timeoff').asString:=timetostr(time);
end;
datamodule3.Table1.Post;
datamodule3.Table1.First;
end;
procedure data_rec_avr (btn:string);
begin

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

datamodule3.Table1.First;
datamodule3.Table1.MoveBy(strtoint(btn)-1);
datamodule3.Table1.edit;
IF datamodule3.Table1.FieldByName('Time Average').asstring = '00:00:00' then
datamodule3.Table1.FieldByName('Time Average').asstring:=timetostr((
strtotime(datamodule3.Table1.FieldByName('Timeoff').asstring)-
strtotime(datamodule3.Table1.FieldByName('Timeon').asstring) ) )
else
//VVV( timeav:= (timeav:=((timeav*cout-1)+(timeoff-timein))/(count) )
datamodule3.Table1.FieldByName('Time Average').asstring:=timetostr(
((strtotime(datamodule3.Table1.FieldByName('Time
Average').asstring)*(datamodule3.Table1.FieldByName('Couter').asinteger-
1))+strtotime(datamodule3.Table1.FieldByName('TimeoFF').asstring)-
strtotime(datamodule3.Table1.FieldByName('TimeoN').asstring)))/datamodule3.Table1.FieldByName('Couter').
asinteger );
datamodule3.Table1.Post;
datamodule3.Table1.First;
end;
procedure ake (btn:integer;onoff:boolean);
begin
case btn of
1:if not(onoff) then
begin //off
if shape1.Brush.Color<>clred then
begin
datarecord(inttostr(btn),onoff);
if shape1.Brush.Color<>clwhite then data_rec_avr(inttostr(btn));
shape1.Brush.Color:=clred;
end;
end
else
begin //on
if shape1.Brush.Color<>cllime then
begin
datarecord(inttostr(btn),onoff);
shape1.Brush.Color:=clLIME;
end;
end;
end;
end;
end;
end;

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

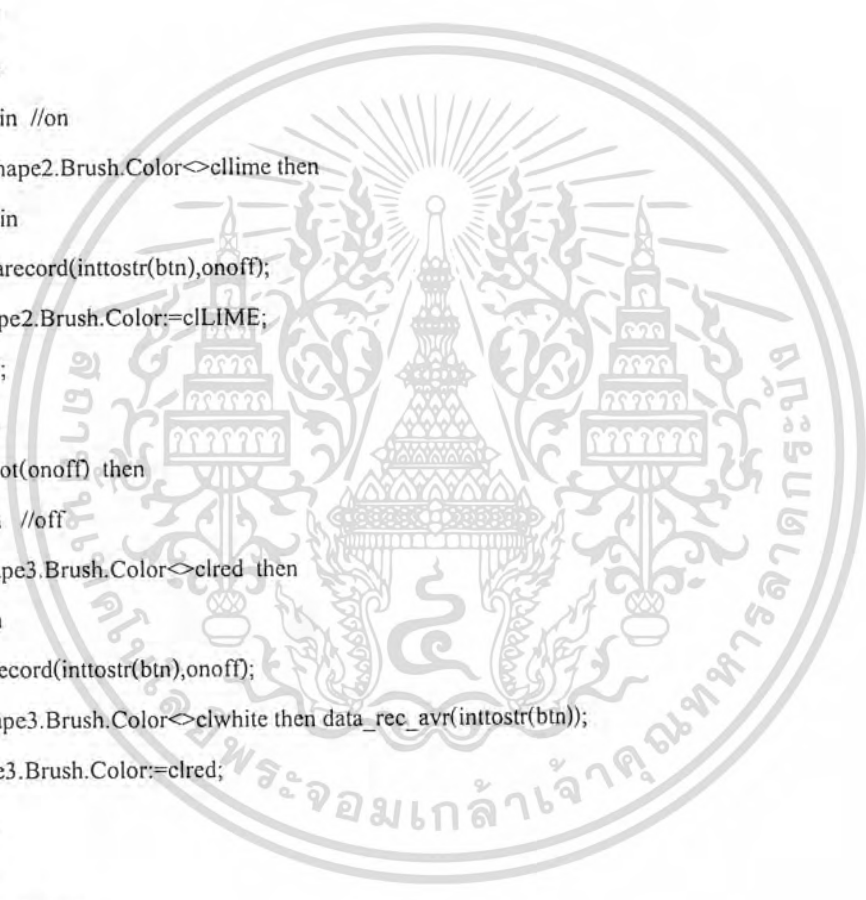


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

2:if not(onoff) then
begin //off
if shape2.Brush.Color<>clred then
begin
datarecord(inttostr(btn),onoff);
if shape2.Brush.Color<>clwhite then data_rec_avr(inttostr(btn));
shape2.Brush.Color:=clred;
end;
end
else
begin //on
if shape2.Brush.Color<>cllime then
begin
datarecord(inttostr(btn),onoff);
shape2.Brush.Color:=clLIME;
end;
end;
3:if not(onoff) then
begin //off
if shape3.Brush.Color<>clred then
begin
datarecord(inttostr(btn),onoff);
if shape3.Brush.Color<>clwhite then data_rec_avr(inttostr(btn));
shape3.Brush.Color:=clred;
end;
end
else begin //on
if shape3.Brush.Color<>cllime then
begin
datarecord(inttostr(btn),onoff);
shape3.Brush.Color:=clLIME;
end;
end;
4:if not(onoff) then
begin //off
if shape4.Brush.Color<>clred then

```



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับอาจารย์งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

begin
datarecord(inttostr(btn),onoff);
if shape4.Brush.Color<>clwhite then data_rec_avr(inttostr(btn));
shape4.Brush.Color:=clred;
end;
end
else begin //on
  if shape4.Brush.Color<>cllime then
  begin
  datarecord(inttostr(btn),onoff);
  shape4.Brush.Color:=clLIME;
  end;
  end;
5:if not(onoff) then
begin //off
if shape5.Brush.Color<>clred then
begin
datarecord(inttostr(btn),onoff);
if shape5.Brush.Color<>clwhite then data_rec_avr(inttostr(btn));
shape5.Brush.Color:=clred;
end;
end
else begin //on
  if shape5.Brush.Color<>cllime then
  begin
  datarecord(inttostr(btn),onoff);
  shape5.Brush.Color:=clLIME;
  end;
  end;
6:if not(onoff) then
begin //off
if shape6.Brush.Color<>clred then
begin
datarecord(inttostr(btn),onoff);
if shape6.Brush.Color<>clwhite then data_rec_avr(inttostr(btn));
shape6.Brush.Color:=clred;
end;
end;

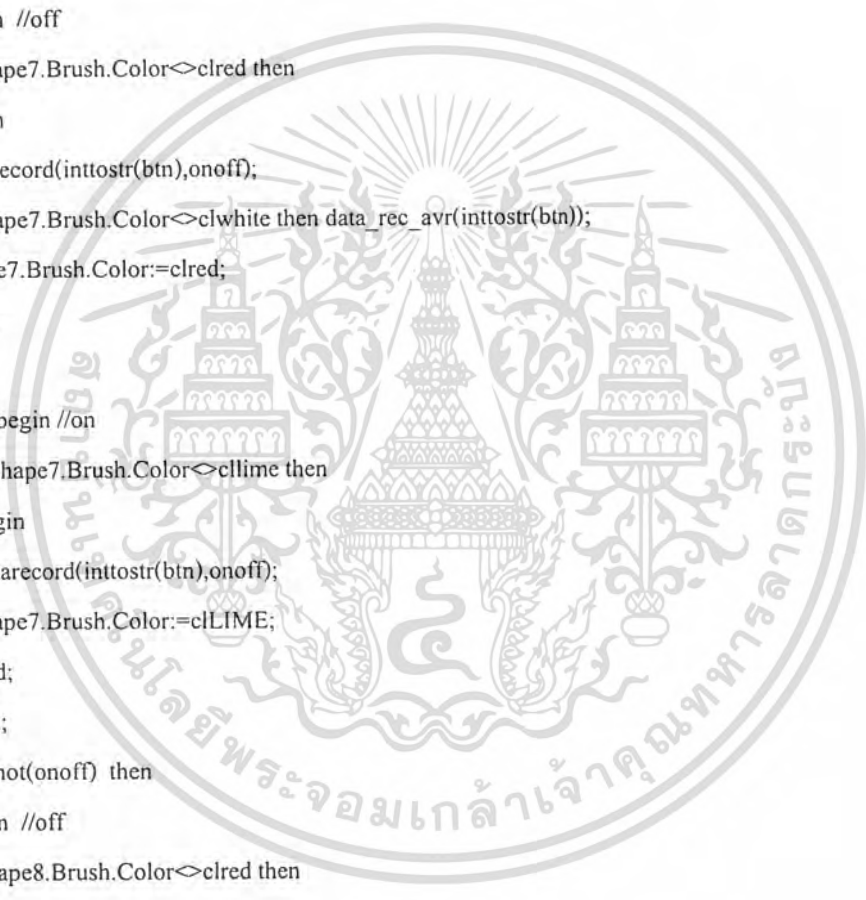
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

end
else begin //on
    if shape6.Brush.Color<>cllime then
        begin
            datarecord(inttostr(btn),onoff);
            shape6.Brush.Color:=clLIME;
            end;
            end;
7:if not(onoff) then
begin //off
if shape7.Brush.Color<>clred then
begin
datarecord(inttostr(btn),onoff);
if shape7.Brush.Color<>clwhite then data_rec_avr(inttostr(btn));
shape7.Brush.Color:=clred;
end;
end
else begin //on
    if shape7.Brush.Color<>cllime then
        begin
            datarecord(inttostr(btn),onoff);
            shape7.Brush.Color:=clLIME;
            end;
            end;
8:if not(onoff) then
begin //off
if shape8.Brush.Color<>clred then
begin
datarecord(inttostr(btn),onoff);
if shape8.Brush.Color<>clwhite then data_rec_avr(inttostr(btn));
shape8.Brush.Color:=clred;
end;
end
else begin //on
    if shape8.Brush.Color<>cllime then
        begin
            datarecord(inttostr(btn),onoff);

```



```

shape8.Brush.Color:=clLIME;
end;
end;
9:if not(onoff) then
begin //off
if shape9.Brush.Color<>clred then
begin
datarecord(inttostr(btn),onoff);
if shape9.Brush.Color<>clwhite then data_rec_avr(inttostr(btn));
shape9.Brush.Color:=clred;
end;
end
else begin //on
if shape9.Brush.Color<>cllime then
begin
datarecord(inttostr(btn),onoff);
shape9.Brush.Color:=clLIME;
end;
end;
10:if not(onoff) then
begin //off
if shape10.Brush.Color<>clred then
begin
datarecord(inttostr(btn),onoff);
if shape10.Brush.Color<>clwhite then data_rec_avr(inttostr(btn));
shape10.Brush.Color:=clred;
end;
end
else begin //on
if shape10.Brush.Color<>cllime then
begin
datarecord(inttostr(btn),onoff);
shape10.Brush.Color:=clLIME;
end;
end;

```

```

11:if not(onoff) then

```

```

begin //off

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

if shape11.Brush.Color<>clred then
begin
datarecord(inttostr(btn),onoff);
if shape11.Brush.Color<>clwhite then data_rec_avr(inttostr(btn));
shape11.Brush.Color:=clred;
end;
end
else begin //on
if shape11.Brush.Color<>cllime then
begin
datarecord(inttostr(btn),onoff);
shape11.Brush.Color:=clLIME;
end;
end;
12:if not(onoff) then
begin //off
if shape12.Brush.Color<>clred then
begin
datarecord(inttostr(btn),onoff);
if shape12.Brush.Color<>clwhite then data_rec_avr(inttostr(btn));
shape12.Brush.Color:=clred;
end;
end
else begin //on
if shape12.Brush.Color<>cllime then
begin
datarecord(inttostr(btn),onoff);
shape12.Brush.Color:=clLIME;
end;
end;
13:if not(onoff) then
begin //off
if shape13.Brush.Color<>clred then
begin
datarecord(inttostr(btn),onoff);
if shape13.Brush.Color<>clwhite then data_rec_avr(inttostr(btn));
shape13.Brush.Color:=clred;

```



เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์การใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

end;
end
else begin //on
    if shape13.Brush.Color<>cllime then
        begin
            datarecord(inttostr(btn),onoff);
            shape13.Brush.Color:=clLIME;
            end;
        end;
14:if not(onoff) then
begin //off
    if shape14.Brush.Color<>clred then
        begin
            datarecord(inttostr(btn),onoff);
            if shape14.Brush.Color<>clwhite then data_rec_avr(inttostr(btn));
            shape14.Brush.Color:=clred;
            end;
        end
    else begin //on
        if shape14.Brush.Color<>cllime then
            begin
                datarecord(inttostr(btn),onoff);
                shape14.Brush.Color:=clLIME;
                end;
            end;
15:if not(onoff) then
begin //off
    if shape15.Brush.Color<>clred then
        begin
            datarecord(inttostr(btn),onoff);
            if shape15.Brush.Color<>clwhite then data_rec_avr(inttostr(btn));
            shape15.Brush.Color:=clred;
            end;
        end
    else
        begin //on
            if shape15.Brush.Color<>cllime then

```



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับกรใช้วงมเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

begin
datarecord(inttostr(btn),onoff);
shape15.Brush.Color:=clLIME;
end;
end;
16:if not(onoff) then
begin //off
if shape16.Brush.Color<>clred then
begin
datarecord(inttostr(btn),onoff);
if shape16.Brush.Color<>clwhite then data_rec_avr(inttostr(btn));
shape16.Brush.Color:=clred;
end;
end
else
begin //on
if shape16.Brush.Color<>cllime then
begin
datarecord(inttostr(btn),onoff);
shape16.Brush.Color:=clLIME;
end;
end;
end;
end;
end;

```

```

procedure TForm1.Button1Click(Sender: TObject);

```

```

var i,j,b:integer;

```

```

a,recieve2:string;

```

```

begin

```

```

IF LABEL3.CAPTION = 'ONLINE' THEN EXIT;

```

```

LABEL3.Caption:='ONLINE';

```

```

LABEL3.Color:=CLLIME;

```

```

//at first time you will sent data (set mode plc)

```

```

repeat

```

```

a:=addfcs('@00SC02');

```

```

comport1.Active:=true;

```

```

comport1.WriteString(a+chr(13));

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

// comport1.Active:=false;
i:=0;
RECIEVE:=";
REPEAT
try
// comport1.Active:=true;
recieve:=RECIEVE+comport1.ReadChar;
// comport1.Active:=false;
except
on E: EComError do
begin
MessageDlg('Error ' + IntToStr(E.ErrorCode) + ': ' + E.Message, mtError,[mbOK], 0);
showmessage('please check your cable!');
LABEL3.Caption:='OFF LINE';
LABEL3.Color:=CLred;
exit;
end;
end;
i:=i+1;
UNTIL (RECIEVE[i]='*');
until (recieve='@00SC0050*');
//SENT SC OPEN MONITERMODE ALLREADY
comport1.Active:=false;
{progressbar1.StepBy(0);}
//now sent about wr&rr-----CHECK RESPONT {WR RR}
if juventus <10 then
inpstring:='@00RR'+0100'{TABLE1.FIELDBYNAME(INTTOSTR(J)+'C').ASSTRING}+000'{1}'+inttostr(ju
ventus)
else
inpstring:='@00RR'+0100'{TABLE1.FIELDBYNAME(INTTOSTR(J)+'C').ASSTRING}+00{01}'+inttostr(ju
ventus);
INPSTRING:=ADDFCS(INPSTRING);
REPEAT
edit1.Text:=recieve;
RECIEVE:=";
RECIEVE2:=";
comport1.Active:=true;

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

comport1.WriteString(a+chr(13));
// comport1.Active:=false;

i:=0;
REPEAT
  try
// comport1.Active:=true;
  recieve:=RECIEVE+comport1.ReadChar;
// comport1.Active:=false;
  except
  on E: EComError do
  begin
  timer1.Enabled:=false;
  MessageDlg('Error ' + IntToStr(E.ErrorCode) + ': ' + E.Message, mtError,[mbOK], 0);
  showmessage('please check your cable!');
  LABEL3.Caption:='OFF LINE';
  LABEL3.Color:=CLred;
  exit;
  end;
  end;
  i:=i+1;
  UNTIL (RECIEVE[i]='*');
  comport1.Active:=false;
  for i:=1 to length(recieve)-3 do //CHECKFCS
  recieve2:=recieve2+recieve[i];
  RECIEVE2:=ADDFCS(RECIEVE2);
  UNTIL (RECIEVE2 = RECIEVE)AND(RECIEVE[6] = '0')AND(RECIEVE[7] = '0');
  //progressbar1.Position:=0;
  //progressbar1.Refresh;
  TIMER1.Interval:=1000;
  TIMER1.Enabled:=TRUE;
  end;

PROCEDURE TATEEMAN(STR:STRING;str2:string);
var Z,W,I :INTEGER;
    RECIEVE1,recieve2,B100,B1001,SEND1,ST : STRING;

```

ST2 :CHAR;

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
 begin  
 ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

IF FORM1.Label3.Caption = 'OFF LINE' THEN EXIT;
Z:=0;W:=0;I:=0;RECIEVE1:="";RECIEVE2:="";B100:="";B1001:="";SEND1:="";ST:="";
FORM1.TIMER1.Enabled:=FALSE;
form1.TIMER1.Interval:=1000;
repeat
SEND1:="";
REPEAT
SEND1:=@00RR'+{form1.Table1.fieldbyname(str2).asString}'0100'+'0001';
SEND1:=ADDFCS(SEND1);
FORM1.comport1.Active:=true;
FORM1.comport1.WriteString(SEND1+CHR(13));
// FORM1.comport1.Active:=false;
Z:=0;
repeat
try
// FORM1.comport1.Active:=true;
recieve1:=RECIEVE1+FORM1.comport1.ReadChar;
// FORM1.comport1.Active:=false;
except
on E: EComError do
begin
form1.timer1.Enabled:=false;
MessageDlg('Error' + IntToStr(E.ErrorCode) + ': ' + E.Message, mtError,[mbOK], 0);
showmessage('please check your cable!');
form1.LABEL3.Caption:='OFF LINE';
form1.LABEL3.Color:=CLred;
exit;
end;
end;
Z:=Z+1;
UNTIL (RECIEVE1[Z]='*');
for i:=1 to length(recieve1)-3 do
recieve2:=recieve2+recieve1[i];
RECIEVE2:=ADDFCS(RECIEVE2);
UNTIL (RECIEVE2 = RECIEVE)AND(RECIEVE1[6] = '0')AND(RECIEVE1[7] = '0');
FORM1.comport1.Active:=false;

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
 case FORM1.TABLE1.FieldByName(STR).asINTEGER of

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

0.3 :begin

```
W:=hextodec(recieve[11]); @@  
B100:=BITSTATUS(W);  
CASE FORM1.TABLE1.FieldByName(STR).asINTEGER OF  
0:IF B100[1] = '0' THEN B1001:='1'+B100[2]+B100[3]+B100[4] ELSE  
B1001:='0'+B100[2]+B100[3]+B100[4];  
1:IF B100[2] = '0' THEN B1001:=B100[1]+'1'+B100[3]+B100[4] ELSE  
B1001:=B100[1]+'0'+B100[3]+B100[4];  
2:IF B100[3] = '0' THEN B1001:=B100[1]+B100[2]+'1'+B100[4] ELSE  
B1001:=B100[1]+B100[2]+'0'+B100[4];  
3:IF B100[4] = '0' THEN B1001:=B100[1]+B100[2]+B100[3]+'1' ELSE  
B1001:=B100[1]+B100[2]+B100[3]+'0';  
END;  
I:=STATUSBIT(B1001);  
ST2:=dectohex(I);  
st:=recieve[8]+recieve[9]+recieve[10]+st2;  
end;
```

4.7 :begin

```
W:=hextodec(recieve[10]); @@  
B100:=BITSTATUS(W);  
CASE FORM1.TABLE1.FieldByName(STR).asINTEGER OF  
4:IF B100[1] = '0' THEN B1001:='1'+B100[2]+B100[3]+B100[4] ELSE  
B1001:='0'+B100[2]+B100[3]+B100[4];  
5:IF B100[2] = '0' THEN B1001:=B100[1]+'1'+B100[3]+B100[4] ELSE  
B1001:=B100[1]+'0'+B100[3]+B100[4];  
6:IF B100[3] = '0' THEN B1001:=B100[1]+B100[2]+'1'+B100[4] ELSE  
B1001:=B100[1]+B100[2]+'0'+B100[4];  
7:IF B100[4] = '0' THEN B1001:=B100[1]+B100[2]+B100[3]+'1' ELSE  
B1001:=B100[1]+B100[2]+B100[3]+'0';  
END;  
I:=STATUSBIT(B1001);  
ST2:=dectohex(I);  
st:=recieve[8]+recieve[9]+st2+recieve[11];  
end;
```

8..11:begin

```
W:=hextodec(recieve[9]); @@  
B100:=BITSTATUS(W);
```

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์ไว้สำหรับอาจารย์ใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

CASE FORM1.TABLE1.FieldByName(STR).asINTEGER OF
8:IF B100[1] = '0' THEN B1001:='1'+B100[2]+B100[3]+B100[4] ELSE
B1001:='0'+B100[2]+B100[3]+B100[4];
9:IF B100[2] = '0' THEN B1001:=B100[1]+'1'+B100[3]+B100[4] ELSE
B1001:=B100[1]+'0'+B100[3]+B100[4];
10:IF B100[3] = '0' THEN B1001:=B100[1]+B100[2]+'1'+B100[4] ELSE
B1001:=B100[1]+B100[2]+'0'+B100[4];
11:IF B100[4] = '0' THEN B1001:=B100[1]+B100[2]+B100[3]+'1' ELSE
B1001:=B100[1]+B100[2]+B100[3]+'0';
END;
I:=STATUSBIT(B1001);
ST2:=dectohex(I);
st:=recieve1[8]+st2+recieve1[10]+recieve1[11];
end;
12..15:begin
W:=hextodec(recieve[8]); //@@
B100:=BITSTATUS(W);
CASE FORM1.TABLE1.FieldByName(STR).asINTEGER OF
12:IF B100[1] = '0' THEN B1001:='1'+B100[2]+B100[3]+B100[4] ELSE
B1001:='0'+B100[2]+B100[3]+B100[4];
13:IF B100[2] = '0' THEN B1001:=B100[1]+'1'+B100[3]+B100[4] ELSE
B1001:=B100[1]+'0'+B100[3]+B100[4];
14:IF B100[3] = '0' THEN B1001:=B100[1]+B100[2]+'1'+B100[4] ELSE
B1001:=B100[1]+B100[2]+'0'+B100[4];
15:IF B100[4] = '0' THEN B1001:=B100[1]+B100[2]+B100[3]+'1' ELSE
B1001:=B100[1]+B100[2]+B100[3]+'0';
END;
I:=STATUSBIT(B1001);
ST2:=dectohex(I);
st:=st2+recieve1[9]+recieve1[10]+recieve1[11];
end;
END;
SEND1:='@00WR'+{form1.Table1.fieldbyname(str2).asString}'0100'+ST;
FORM1.MEMO1.Lines.Add(SEND1);
SEND1:=ADDFCS(SEND1);

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
 Z:=0;  
 ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

RECIEVE:="";
FORM1.MEMO1.Lines.Add(SEND1);
FORM1.COMPORT1.Active:=TRUE;
FORM1.COMPORT1.WRITESTRING(SEND1+CHR(13));
// FORM1.COMPORT1.Active:=false;

repeat
  try
// FORM1.COMPORT1.Active:=TRUE;
  recieve:=RECIEVE+FORM1.comport1.ReadChar;
// FORM1.COMPORT1.Active:=false;

  except
  on E: EComError do
  begin
  form1.timer1.Enabled:=false;
  MessageDlg('Error ' + IntToStr(E.ErrorCode) + ': ' + E.Message, mtError,[mbOK], 0);
  showmessage('please check your cable!');
  form1.LABEL3.Caption:='OFF LINE';
  form1.LABEL3.Color:=CLred;
  exit;
  end;
  end;
  Z:=Z+1;
  UNTIL (RECIEVE[Z]='*');
  until (recieve = '@00WR0045*');
  FORM1.COMPORT1.ACTIVE:=FALSE;
  FORM1.TIMER1.Enabled:=TRUE;
  onebitchange:=false;

  end;

  procedure TForm1.Shape100MouseDown(Sender: TObject; Button: TMouseButton;
  Shift: TShiftState; X, Y: Integer);
  BEGIN
  IF FORM1.Label3.Caption = 'OFF LINE' THEN EXIT;
  onebitchange:=true;onebitsend1:='100B';onebitsend2:='100c';
  end;

  procedure TForm1.Shape101MouseDown(Sender: TObject; Button: TMouseButton;
  Shift: TShiftState; X, Y: Integer);

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับอาจารย์ใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

begin
IF FORM1.Label3.Caption = 'OFF LINE' THEN EXIT;
onebitchange:=true;onebitsend1:='101B';onebitsend2:='101c';
end;
procedure TForm1.Shape102MouseDown(Sender: TObject; Button: TMouseButton;
  Shift: TShiftState; X, Y: Integer);
begin
IF FORM1.Label3.Caption = 'OFF LINE' THEN EXIT;
onebitchange:=true;onebitsend1:='102B';onebitsend2:='102c';
end;
procedure TForm1.Shape103MouseDown(Sender: TObject; Button: TMouseButton;
  Shift: TShiftState; X, Y: Integer);
begin
IF FORM1.Label3.Caption = 'OFF LINE' THEN EXIT;
onebitchange:=true;onebitsend1:='103B';onebitsend2:='103c';
end;
procedure TForm1.Shape104MouseDown(Sender: TObject; Button: TMouseButton;
  Shift: TShiftState; X, Y: Integer);
begin
IF FORM1.Label3.Caption = 'OFF LINE' THEN EXIT;
onebitchange:=true;onebitsend1:='104B';onebitsend2:='104c';
end;
procedure TForm1.Shape105MouseDown(Sender: TObject; Button: TMouseButton;
  Shift: TShiftState; X, Y: Integer);
begin
IF FORM1.Label3.Caption = 'OFF LINE' THEN EXIT;
onebitchange:=true;onebitsend1:='105B';onebitsend2:='105c';
end;
procedure TForm1.Shape106MouseDown(Sender: TObject; Button: TMouseButton;
  Shift: TShiftState; X, Y: Integer);
begin
IF FORM1.Label3.Caption = 'OFF LINE' THEN EXIT;
onebitchange:=true;onebitsend1:='106B';onebitsend2:='106c';
end;
procedure TForm1.Shape107MouseDown(Sender: TObject; Button: TMouseButton;
  Shift: TShiftState; X, Y: Integer);
begin

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

IF FORM1.Label3.Caption = 'OFF LINE' THEN EXIT;
onebitchange:=true;onebitsend1:='107B';onebitsend2:='107c';
end;
procedure TForm1.Shape108MouseDown(Sender: TObject; Button: TMouseButton;
  Shift: TShiftState; X, Y: Integer);
begin
IF FORM1.Label3.Caption = 'OFF LINE' THEN EXIT;
onebitchange:=true;onebitsend1:='108B';onebitsend2:='108c';
end;
procedure TForm1.Shape109MouseDown(Sender: TObject; Button: TMouseButton;
  Shift: TShiftState; X, Y: Integer);
begin
IF FORM1.Label3.Caption = 'OFF LINE' THEN EXIT;
onebitchange:=true;onebitsend1:='109B';onebitsend2:='109c';
end;
procedure TForm1.Shape110MouseDown(Sender: TObject; Button: TMouseButton;
  Shift: TShiftState; X, Y: Integer);
begin
IF FORM1.Label3.Caption = 'OFF LINE' THEN EXIT;
onebitchange:=true;onebitsend1:='110B';onebitsend2:='110c';
end;
procedure TForm1.Shape111MouseDown(Sender: TObject; Button: TMouseButton;
  Shift: TShiftState; X, Y: Integer);
begin
IF FORM1.Label3.Caption = 'OFF LINE' THEN EXIT;
onebitchange:=true;onebitsend1:='111B';onebitsend2:='111c';
end;
procedure TForm1.Shape112MouseDown(Sender: TObject; Button: TMouseButton;
  Shift: TShiftState; X, Y: Integer);
begin
IF FORM1.Label3.Caption = 'OFF LINE' THEN EXIT;
onebitchange:=true;onebitsend1:='112B';onebitsend2:='112c';
end;
procedure TForm1.Shape113MouseDown(Sender: TObject; Button: TMouseButton;
  Shift: TShiftState; X, Y: Integer);
begin

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับกรใช้เฉพาะที่องค์กรศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

onebitchange:=true;onebitsend1:='113B';onebitsend2:='113c';
end;
procedure TForm1.Shape114MouseDown(Sender: TObject; Button: TMouseButton;
  Shift: TShiftState; X, Y: Integer);
begin
IF FORM1.Label3.Caption = 'OFF LINE' THEN EXIT;
onebitchange:=true;onebitsend1:='114B';onebitsend2:='114c';
end;
procedure TForm1.Shape115MouseDown(Sender: TObject; Button: TMouseButton;
  Shift: TShiftState; X, Y: Integer);
begin
IF FORM1.Label3.Caption = 'OFF LINE' THEN EXIT;
onebitchange:=true;onebitsend1:='115B';onebitsend2:='115c';
end;
procedure TForm1.BitBtn1Click(Sender: TObject);
var z,l:integer;
  TXT,RECIEVE1,RECIEVE2 : STRING;
begin
IF FORM1.Label3.Caption = 'OFF LINE' THEN EXIT;
if MessageDlg('Are You Sure To Sent Word',mtwarning,[mbok,mbcancel],0) = mrOk then
begin
timer1.Enabled:=false;
TIMER1.Interval:=500;
CASE COMBOBOX1.ITEMINDEX OF
  -1 :begin
    txt:=combobox1.text;
    end;
  0 :begin
    txt:='0000';
    end;
  1 :begin
    txt:='0100';
    end;
end;
end;
txt:='@00WR'+TXT+EDIT2.TEXT;

```

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

RECIEVE1:="";
REPEAT
Z:=0;
RECIEVE1:="";
RECIEVE2:="";
comport1.active:=true;
comport1.WriteString(txt+chr(13));
// comport1.active:=false;
REPEAT
try
// comport1.active:=true;
recieve1:=RECIEVE1+comport1.ReadChar;
// comport1.active:=false;
except
on E: EComError do
begin
timer1.Enabled:=false;
MessageDlg('Error ' + IntToStr(E.ErrorCode) + ' ' + E.Message, mtError,[mbOK], 0);
showmessage('please check your cable!');
LABEL3.Caption:='OFF LINE';
LABEL3.Color:=CLred;
exit;
end;
end;

Z:=Z+1;
UNTIL (RECIEVE1[Z]='*');
for i:=1 to length(recieve1)-3 do //CHECKFCS
recieve2:=recieve2+recieve1[i];
RECIEVE2:=ADDFCS(RECIEVE2);
UNTIL (RECIEVE2 = RECIEVE1)AND(RECIEVE1[6] = '0')AND(RECIEVE1[7] = '0');
TIMER1.Enabled:=TRUE;
end;
end;

```

```

procedure TForm1.Button3Click(Sender: TObject);

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

LABEL3.Caption:='OFF LINE';
LABEL3.Color:=CLred;
TIMER1.Enabled:=FALSE;
end;

procedure TForm1.Timer1Timer(Sender: TObject);
var Z,X,Y,W,i,J,b,k:integer;
    a,recieve2,c2:string;
    aa : array[1..10,1..4,1..4] of boolean ;
begin
if label4.Visible then exit;
TIMER1.Enabled:=FALSE;
if onebitchange then
begin
TATEEMAN(onebitsend1,onebitsend2);
onebitchange:=false;
end;
TIMER1.Interval:=500;
LABEL3.Caption:='ONLINE';
//edit1.text:='online';
LABEL3.Color:=CLLIME;
if juvenus <10 then
inpstring:='@00RR'+0100'{TABLE1.FIELDDBYNAME(INTTOSTR(J)+'C').ASSTRING}+'000'{1}'+inttostr(ju
ventus)
else
inpstring:='@00RR'+0100'{TABLE1.FIELDDBYNAME(INTTOSTR(J)+'C').ASSTRING}+'00'{01}'+inttostr(ju
ventus);
INPSTRING:=ADDFCS(INPSTRING);
REPEAT
i:=0;
RECIEVE:=";
RECIEVE2:=";
comport1.active:=true;
comport1.WriteString(inpstring+chr(13));//SENT DATA ALLREADY
// comport1.active:=false;
REPEAT

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

// comport1.active:=true;
recieve:=RECIEVE+comport1.ReadChar;
// comport1.active:=false;
except
on E: EComError do
begin
timer1.Enabled:=false;
MessageDlg('Error ' + IntToStr(E.ErrorCode) + ': ' + E.Message, mtError,[mbOK], 0);
showmessage('please check your cable!');
exit;
end;
end;
end;
i:=i+1;
UNTIL (RECIEVE[i]='*');
comport1.Active:=false;
// progressbar1.StepBy(25);
for i:=1 to length(recieve)-3 do //CHECKFCS
recieve2:=recieve2+recieve[i];
RECIEVE2:=ADDFCS(RECIEVE2);
UNTIL (RECIEVE2 = RECIEVE)AND(RECIEVE[6] = '0')AND(RECIEVE[7] = '0');
//at first time you will sent data (set mode plc)
//now sent about wr&rr-----CHECK RESPONT {WR RR}
//CHECK FCS AND CHECK DATA ALLREADY
for i:= 1 to juventus do
for j:=4 downto 1 do
begin
b:=hexdec(recieve[3+(i*4)+(5-j)]);
if b>=8 then
begin
b:=b-8;
aa[i,j,4]:= true;
end
else aa[i,j,4]:=false;
if b>=4 then
begin
b:=b-4;
aa[i,j,3]:= true;

```

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์การใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

        end
        else aa[i,j,3]:=false;
    if b>=2 then
        begin
            b:=b-2;
            aa[i,j,2]:= true;
        end
        else aa[i,j,2]:=false;
    if b>=1 then
        begin
            b:=b-1;
            aa[i,j,1]:= true;
        end
        else aa[i,j,1]:=false;
    end;
//progressbar1.StepBy(50);
for i:=1 to 16 do begin
case strtoint(bit[i]) of
    0 :begin j:=1; k:=1; end;
    1 :begin j:=1; k:=2; end;
    2 :begin j:=1; k:=3; end;
    3 :begin j:=1; k:=4; end;
    4 :begin j:=2; k:=1; end;
    5 :begin j:=2; k:=2; end;
    6 :begin j:=2; k:=3; end;
    7 :begin j:=2; k:=4; end;
    8 :begin j:=3; k:=1; end;
    9 :begin j:=3; k:=2; end;
    10 :begin j:=3; k:=3; end;
    11 :begin j:=3; k:=4; end;
    12 :begin j:=4; k:=1; end;
    13 :begin j:=4; k:=2; end;
    14 :begin j:=4; k:=3; end;
    15 :begin j:=4; k:=4; end;
end;
//progressbar1.StepBy(25);

```

เอกสารนี้เผยแพร่โดยคณะเทคโนโลยีสารสนเทศ มหาวิทยาลัยเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง  
 เอกสารนี้เผยแพร่โดยคณะเทคโนโลยีสารสนเทศ มหาวิทยาลัยเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง  
 อนุญาตให้ใช้ประโยชน์ด้านการค้า  
 ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

END;
//progressbar1.Position:=0;
//progressbar1.Refresh;
FOR Z:=1 TO JUVENTUS DO
FOR X:=1 TO 4 DO
FOR Y:=1 TO 4 DO
BEGIN
IF AA[Z,X,Y]=TRUE THEN W:=1 ELSE W:=0;
{MEMO1.Lines.Add('AA['+INTTOSTR(Z)+'+',+INTTOSTR(X)+'+',+INTTOSTR(Y)+'']='+INTTOSTR(W));}
TIMER1.Enabled:=TRUE;
END;
end;

```

```

procedure TForm1.Open1Click(Sender: TObject);
var i:integer;
begin
TIMER1.Enabled:=FALSE;
table1.Active:=false;
if opendirialog1.Execute then
begin
table1.TableName:=opendirialog1.FileName;
memo2.Lines.Clear;
memo2.Lines.Add(opendirialog1.filename+'*');
memo2.Lines.SaveToFile('c:\name.txt');
end;
table1.Active:=true;
if opendirialog1.FileName = " then
begin
table1.Active:=false;
table1.TableName:='C:\databaseake\dbproject.DB';
table1.Active:=true;
end;
if not((table1.FieldByName('1b').asinteger = 0)and
(table1.FieldByName('1c').asinteger = 0))
then
begin

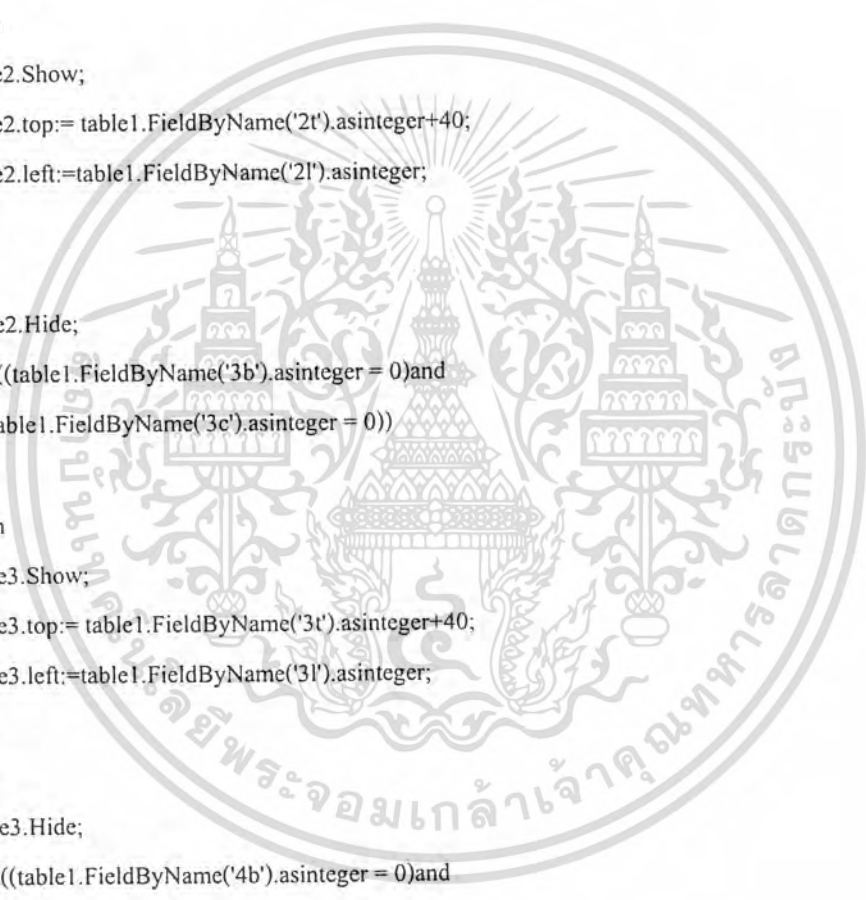
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

shape1.Show;
shape1.top:= table1.FieldByName('1t').asinteger+40;
shape1.left:=table1.FieldByName('1l').asinteger;
end
else
shape1.Hide;
if not((table1.FieldByName('2b').asinteger = 0)and
      (table1.FieldByName('2c').asinteger = 0))
then
begin
shape2.Show;
shape2.top:= table1.FieldByName('2t').asinteger+40;
shape2.left:=table1.FieldByName('2l').asinteger;
end
else
shape2.Hide;
if not((table1.FieldByName('3b').asinteger = 0)and
      (table1.FieldByName('3c').asinteger = 0))
then
begin
shape3.Show;
shape3.top:= table1.FieldByName('3t').asinteger+40;
shape3.left:=table1.FieldByName('3l').asinteger;
end
else
shape3.Hide;
if not((table1.FieldByName('4b').asinteger = 0)and
      (table1.FieldByName('4c').asinteger = 0))
then
begin
shape4.Show;
shape4.top:= table1.FieldByName('4t').asinteger+40;
shape4.left:=table1.FieldByName('4l').asinteger;
end
else
shape4.Hide;
if not((table1.FieldByName('5b').asinteger = 0)and

```



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับกรณีใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

    (table1.FieldName('5c').asinteger = 0))
then
begin
shape5.Show;
shape5.top:= table1.FieldName('5t').asinteger+40;
shape5.left:=table1.FieldName('5l').asinteger;
end
else
shape5.Hide;
if not((table1.FieldName('6b').asinteger = 0)and
    (table1.FieldName('6c').asinteger = 0))
then
begin
shape6.Show;
shape6.top:= table1.FieldName('6t').asinteger+40;
shape6.left:=table1.FieldName('6l').asinteger;
end
else
shape6.Hide;
if not((table1.FieldName('7b').asinteger = 0)and
    (table1.FieldName('7c').asinteger = 0))
then
begin
shape7.Show;
shape7.top:= table1.FieldName('7t').asinteger+40;
shape7.left:=table1.FieldName('7l').asinteger;
end
else
shape7.Hide;
if not((table1.FieldName('8b').asinteger = 0)and
    (table1.FieldName('8c').asinteger = 0))
then
begin
shape8.Show;
shape8.top:= table1.FieldName('8t').asinteger+40;
shape8.left:=table1.FieldName('8l').asinteger;
end

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

else
    shape8.Hide;
if not((table1.FieldByName('9b').asinteger = 0)and
    (table1.FieldByName('9c').asinteger = 0))
then
    begin
    shape9.Show;
    shape9.top:= table1.FieldByName('9t').asinteger+40;
    shape9.left:=table1.FieldByName('9l').asinteger;
    end
else
    shape9.Hide;
if not((table1.FieldByName('10b').asinteger = 0)and
    (table1.FieldByName('10c').asinteger = 0))
then
    begin
    shape10.Show;
    shape10.top:= table1.FieldByName('10t').asinteger+40;
    shape10.left:=table1.FieldByName('10l').asinteger;
    end
else
    shape10.Hide;
if not((table1.FieldByName('11b').asinteger = 0)and
    (table1.FieldByName('11c').asinteger = 0))
then
    begin
    shape11.Show;
    shape11.top:= table1.FieldByName('11t').asinteger+40;
    shape11.left:=table1.FieldByName('11l').asinteger;
    end
else
    shape11.Hide;
if not((table1.FieldByName('12b').asinteger = 0)and
    (table1.FieldByName('12c').asinteger = 0))
then
    begin
    shape12.Show;

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

shape12.top:= table1.FieldByName('12t').asinteger+40;
shape12.left:=table1.FieldByName('12l').asinteger;
end
else
shape12.Hide;
if not((table1.FieldByName('13b').asinteger = 0)and
(table1.FieldByName('13c').asinteger = 0))
then
begin
shape13.Show;
shape13.top:= table1.FieldByName('13t').asinteger+40;
shape13.left:=table1.FieldByName('13l').asinteger;
end
else
shape13.Hide;
if not((table1.FieldByName('14b').asinteger = 0)and
(table1.FieldByName('14c').asinteger = 0))
then
begin
shape14.Show;
Shape14.top:= table1.FieldByName('14t').asinteger+40;
Shape14.left:=table1.FieldByName('14l').asinteger;
End
Else
shape14.Hide;
if not((table1.FieldByName('15b').asinteger = 0)and
(table1.FieldByName('15c').asinteger = 0))
then
begin
shape15.Show;
shape15.top:= table1.FieldByName('15t').asinteger+40;
shape15.left:=table1.FieldByName('15l').asinteger;
end
else
shape15.Hide;
if not((table1.FieldByName('16b').asinteger = 0)and
(table1.FieldByName('16c').asinteger = 0))

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับอาจารย์และบุคลากรที่ศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

then
begin
shape16.Show;
shape16.top:= table1.FieldByName('16t').asinteger+40;
shape16.left:=table1.FieldByName('16l').asinteger;
end
else
shape16.Hide;
for i:= 1 to 16 do
begin
chanal[i]:=table1.fieldbyname(inttostr(i)+'c').asstring;
bit[i]:=table1.fieldbyname(inttostr(i)+'b').asstring;
chanal[i+16]:=table1.fieldbyname(inttostr(i+99)+'c').asstring;
bit[i+16]:=table1.fieldbyname(inttostr(i+99)+'b').asstring;
end;
shape1.Hint:=table1.fieldbyname('N1').asstring;
shape2.Hint:=table1.fieldbyname('N2').asstring;
shape3.Hint:=table1.fieldbyname('N3').asstring;
shape4.Hint:=table1.fieldbyname('N4').asstring;
shape5.Hint:=table1.fieldbyname('N5').asstring;
shape6.Hint:=table1.fieldbyname('N6').asstring;
shape7.Hint:=table1.fieldbyname('N7').asstring;
shape8.Hint:=table1.fieldbyname('N8').asstring;
shape9.Hint:=table1.fieldbyname('N9').asstring;
shape10.Hint:=table1.fieldbyname('N10').asstring;
shape11.Hint:=table1.fieldbyname('N11').asstring;
shape12.Hint:=table1.fieldbyname('N12').asstring;
shape13.Hint:=table1.fieldbyname('N13').asstring;
shape14.Hint:=table1.fieldbyname('N14').asstring;
shape15.Hint:=table1.fieldbyname('N15').asstring;
shape16.Hint:=table1.fieldbyname('N16').asstring;
shape1.showhint:=true;
shape2.showhint:=true;
shape3.showhint:=true;
shape4.showhint:=true;
shape5.showhint:=true;
shape6.showhint:=true;

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
shape7.showhint:=true;
shape8.showhint:=true;
shape9.showhint:=true;
shape10.showhint:=true;
shape11.showhint:=true;
shape12.showhint:=true;
shape13.showhint:=true;
shape14.showhint:=true;
shape15.showhint:=true;
shape16.showhint:=true;
shape100.Hint:='chanal '+chanal[17]+' bit '+bit[17];
shape100.showhint:=true;
shape101.Hint:='chanal '+chanal[18]+' bit '+bit[18];
shape101.showhint:=true;
shape102.Hint:='chanal '+chanal[19]+' bit '+bit[19];
shape102.showhint:=true;
shape103.Hint:='chanal '+chanal[20]+' bit '+bit[20];
shape103.showhint:=true;
shape104.Hint:='chanal '+chanal[21]+' bit '+bit[21];
shape104.showhint:=true;
shape105.Hint:='chanal '+chanal[22]+' bit '+bit[22];
shape105.showhint:=true;
shape106.Hint:='chanal '+chanal[23]+' bit '+bit[23];
shape106.showhint:=true;
shape107.Hint:='chanal '+chanal[24]+' bit '+bit[24];
shape107.showhint:=true;
shape108.Hint:='chanal '+chanal[25]+' bit '+bit[25];
shape108.showhint:=true;
shape109.Hint:='chanal '+chanal[26]+' bit '+bit[26];
shape109.showhint:=true;
shape110.Hint:='chanal '+chanal[27]+' bit '+bit[27];
shape110.showhint:=true;
shape111.Hint:='chanal '+chanal[28]+' bit '+bit[28];
shape111.showhint:=true;
shape112.Hint:='chanal '+chanal[29]+' bit '+bit[29];
shape112.showhint:=true;
shape113.Hint:='chanal '+chanal[30]+' bit '+bit[30];
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับใช้ในการเรียนการสอนที่โรงเรียนเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

shape113.showhint:=true;
shape114.Hint:='chanal '+chanal[31]+' bit '+bit[31];
shape114.showhint:=true;
shape115.Hint:='chanal '+chanal[32]+' bit '+bit[32];
shape115.showhint:=true;
if table1.FieldName('pic').asString = " then
begin
  showmessage('please select picture in first program');
  form1.Close;
end
else
  image1.Picture.LoadFromFile(table1.fieldbyname('pic').asString);
end;

procedure TForm1.Print1Click(Sender: TObject);
var i: integer;
begin
  datamodule3.table1.First;
  for i:= 1 to 16 do
  begin
    datamodule3.table1.Edit;
    datamodule3.Table1.FieldName('Name').asString:=table1.fieldbyname('N'+inttostr(i)).asString;
    datamodule3.table1.post;
    datamodule3.table1.Next;
  end;
  form5.QuickRep1.Print;
end;

procedure TForm1.PrintPreview1Click(Sender: TObject);
var i: integer;
begin
  datamodule3.table1.First;
  for i:= 1 to 16 do
  begin
    datamodule3.table1.Edit;
    datamodule3.Table1.FieldName('Name').asString:=table1.fieldbyname('N'+inttostr(i)).asString;
    datamodule3.table1.post;

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
datamodule3.table1.Next;  
end;  
form5.QuickRep1.Preview;  
end;
```

```
procedure TForm1.Property1Click(Sender: TObject);  
begin  
form1.Button3.Click;  
if MessageDlg('Are You Sure To Set Property ?',mtwarning,[mbok,mbcancel],0) = mrok then  
begin  
form6.Show;  
end;  
end;
```

```
procedure TForm1.cqm1Click(Sender: TObject);  
begin  
form1.Button3.Click;  
if MessageDlg('Are You Sure To Set Property ?',mtwarning,[mbok,mbcancel],0) = mrok then  
begin  
form1.ComPort1.BaudRate:=br9600;  
form1.ComPort1.DataBits:=db7;  
form1.ComPort1.Parity:=paeven;  
form1.ComPort1.StopBits:=sb2;  
form1.ComPort1.DeviceName:='com1';  
end;  
end;
```

```
procedure TForm1.Timer2Timer(Sender: TObject);  
begin  
timer2.Enabled:=false;  
end;
```

```
procedure TForm1.BitBtn2Click(Sender: TObject);  
var Datamemsent ,DMRECIEVE ,DMRECIEVE2 : string;  
i,j : integer;  
begin
```

```
if label3.caption = 'ON LINE' Then form1.Timer1.Enabled:=false;  
datamemsent := '@00RD00000016';
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับครูใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

datamemsent := addfcs(datamemsent);
REPEAT
i:=0;
DMRECIEVE:="";
comport1.active:=true;
comport1.WriteString(datamemsent+chr(13));//SENT DATA ALLREADY
//comport1.active:=false;
REPEAT
try
//comport1.active:=true;
DMrecieve:=DMRECIEVE+comport1.ReadChar;
//comport1.active:=false;
except
on E: EComError do
begin
timer1.Enabled:=false;
MessageDlg('Error ' + IntToStr(E.ErrorCode) + ': ' + E.Message, mtError,[mbOK], 0);
showmessage('please check your cable!');
exit;
end;
end;
i:=i+1;
UNTIL (dmRECIEVE[i]='*');
//showmessage(dmrecieve);
dmrecieve2:="";
for j:=1 to length(dmrecieve)-3 do
begin
dmrecieve2:=dmrecieve2+dmrecieve[j];
end;
dmrecieve2:=addfcs(dmrecieve2);
until ((dmRECIEVE[6]='0') and (dmRECIEVE[7]='0') and (dmrecieve=dmrecieve2));
comport1.Active:=false;
DataModule3.Table2.First;
for I:=0 to 15 do
begin
dmrecieve2:="";
DataModule3.Table2.First;

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

DataModule3.Table2.MoveBy(i);
for j:=1 to 4 do
begin
dmrecieve2:=dmrecieve2+dmrecieve[7+(i*4)+j];
end;
DataModule3.Table2.Edit;
DataModule3.Table2.FieldName('Time Check').asString:=timetostr(time);
DataModule3.Table2.FieldName('result').asString:=dmrecieve2;
DataModule3.Table2.post;
end;
if label3.caption = 'ON LINE' THEN form1.Timer1.Enabled:=true;
end;

procedure TForm1.BitBtn3Click(Sender: TObject);
var COMBO,Datamemsent,DATAMEMSENT2,dmrecieve,dmrecieve2 : string;
    I,J:INTEGER;
begin
CASE COMBOBOX2.ItemIndex OF
-1 : COMBO:=COMBOBOX2.TEXT;
0 : COMBO:='RR';
1 : COMBO:='WR';
2 : COMBO:='RH';
3 : COMBO:='WH';
4 : COMBO:='RL';
5 : COMBO:='WL';
6 : COMBO:='RC';
7 : COMBO:='WC';
8 : COMBO:='RJ';
9 : COMBO:='WJ';
10 : COMBO:='RD';
11 : COMBO:='WD';
END;
IF COMBO = 'Command' THEN EXIT;
Datamemsent:='@'+maskedit1.Text+combo+maskedit2.Text ;
DATAMEMSENT2:='';
FOR I:=1 TO LENGTH(DATAMEMSENT) DO

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
 BEGIN  
 ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

IF DATAMEMSENT[I]<>' ' THEN DATAMEMSENT2:=DATAMEMSENT2+DATAMEMSENT[I];
END;
DATAMEMSENT:=DATAMEMSENT2;
Datamemsent:=ADDFCS(Datamemsent);
REPEAT
i:=0;
DMRECIEVE:="";
comport1.active:=true;
comport1.WriteString(datamemsent+chr(13));//SENT DATA ALLREADY
// comport1.active:=false;
REPEAT
try
// comport1.active:=true;
DMrecieve:=DMRECIEVE+comport1.ReadChar;
// comport1.active:=false;
except
on E: EComError do
begin
timer1.Enabled:=false;
MessageDlg('Error ' + IntToStr(E.ErrorCode) + ': ' + E.Message, mtError,[mbOK], 0);
showmessage('please check your cable!');
exit;
end;
end;
i:=i+1;
UNTIL (dmRECIEVE[i]='*');
dmrecieve2:="";
for j:=1 to length(dmrecieve)-3 do
begin
dmrecieve2:=dmrecieve2+dmrecieve[j];
end;
dmrecieve2:=addfcs(dmrecieve2);
until (dmrecieve=dmrecieve2);
comport1.Active:=false;
MEMO3.Lines.Add(DMRECIEVE);
end;

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

\*\*\*\*\*Tform2(ComLinkPLC.)\*\*\*\*\*

รูปที่ ๒ ภาพแสดง Form ที่ ๒ ของโปรแกรม ComLinkPLC.

unit Unit2;

interface

uses

Windows, Messages, SysUtils, Classes, Graphics, Controls, Forms, Dialogs,  
Grids, DBGrids, Db, DBTables, StdCtrls;

type

TForm2 = class(TForm)

DBGrid1: TDBGrid;

Button1: TButton;

GroupBox1: TGroupBox;

RadioButton1: TRadioButton;

RadioButton2: TRadioButton;

RadioButton3: TRadioButton;

RadioButton4: TRadioButton;

RadioButton5: TRadioButton;

RadioButton6: TRadioButton;

RadioButton7: TRadioButton;

RadioButton8: TRadioButton;

RadioButton9: TRadioButton;

RadioButton10: TRadioButton;

RadioButton11: TRadioButton;

RadioButton12: TRadioButton;

RadioButton13: TRadioButton;

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

RadioButton14: TRadioButton;
RadioButton15: TRadioButton;
RadioButton16: TRadioButton;

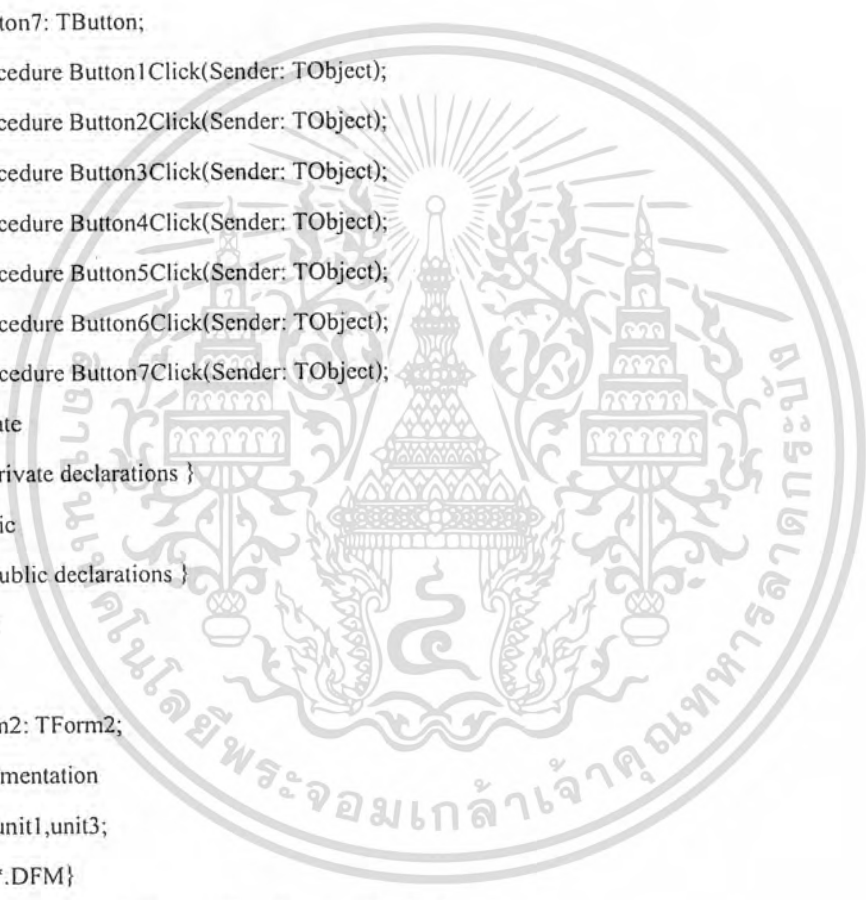
Button2: TButton;
Button3: TButton;
Button4: TButton;
DBGrid2: TDBGrid;
Button5: TButton;
Button6: TButton;
Button7: TButton;

procedure Button1Click(Sender: TObject);
procedure Button2Click(Sender: TObject);
procedure Button3Click(Sender: TObject);
procedure Button4Click(Sender: TObject);
procedure Button5Click(Sender: TObject);
procedure Button6Click(Sender: TObject);
procedure Button7Click(Sender: TObject);
private
  { Private declarations }
public
  { Public declarations }
end;
var
  Form2: TForm2;
implementation
  uses unit1,unit3;
  {SR *.DFM}

  procedure TForm2.Button1Click(Sender: TObject);
  begin
    form2.Hide;
    form1.show;
  end;

  procedure TForm2.Button2Click(Sender: TObject);
  var ij:integer;
  begin
    form1.Timer1.Enabled:=false;

```



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

for i:=1 to 10000 do //delay
begin
j:=(((((((((((i+1)+i)+i)+i)+i)+i)+i)+i)+i)+i)+i);
end;
datamodule3.Table1.First;
if radiobutton1.Checked then
begin
datamodule3.Table1.Edit;
datamodule3.Table1.FieldName('Couter').asinteger:=0;
end;
if radiobutton2.Checked then
begin
datamodule3.Table1.MoveBy(1);
datamodule3.Table1.Edit;
datamodule3.Table1.FieldName('Couter').asinteger:=0;
datamodule3.Table1.Post;
end;
if radiobutton3.Checked then
begin
datamodule3.Table1.MoveBy(2);
datamodule3.Table1.Edit;
datamodule3.Table1.FieldName('Couter').asinteger:=0;
datamodule3.Table1.Post;
end;
if radiobutton4.Checked then
begin
datamodule3.Table1.MoveBy(3);
datamodule3.Table1.Edit;
datamodule3.Table1.FieldName('Couter').asinteger:=0;
datamodule3.Table1.Post;
end;
if radiobutton5.Checked then
begin
datamodule3.Table1.MoveBy(4);
datamodule3.Table1.Edit;
datamodule3.Table1.FieldName('Couter').asinteger:=0;
datamodule3.Table1.Post;

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับผู้ใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

end;
if radiobutton6.Checked then
begin
    datamodule3.Table1.MoveBy(5);
    datamodule3.Table1.Edit;
    datamodule3.Table1.FieldByName('Couter').asinteger:=0;
    datamodule3.Table1.Post;
end;
if radiobutton7.Checked then
begin
    datamodule3.Table1.MoveBy(6);
    datamodule3.Table1.Edit;
    datamodule3.Table1.FieldByName('Couter').asinteger:=0;
    datamodule3.Table1.Post;
end;
if radiobutton8.Checked then
begin
    datamodule3.Table1.MoveBy(7);
    datamodule3.Table1.Edit;
    datamodule3.Table1.FieldByName('Couter').asinteger:=0;
    datamodule3.Table1.Post;
end;
if radiobutton9.Checked then
begin
    datamodule3.Table1.MoveBy(8);
    datamodule3.Table1.Edit;
    datamodule3.Table1.FieldByName('Couter').asinteger:=0;
    datamodule3.Table1.Post;
end;
if radiobutton10.Checked then
begin
    datamodule3.Table1.MoveBy(9);
    datamodule3.Table1.Edit;
    datamodule3.Table1.FieldByName('Couter').asinteger:=0;
    datamodule3.Table1.Post;
end;
if radiobutton11.Checked then

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

begin
    datamodule3.Table1.MoveBy(10);
    datamodule3.Table1.Edit;
    datamodule3.Table1.FieldName('Couter').asinteger:=0;
    datamodule3.Table1.Post;
end;
if radiobutton12.Checked then
begin
    datamodule3.Table1.MoveBy(11);
    datamodule3.Table1.Edit;
    datamodule3.Table1.FieldName('Couter').asinteger:=0;
    datamodule3.Table1.Post;
end;
if radiobutton13.Checked then
begin
    datamodule3.Table1.MoveBy(12);
    datamodule3.Table1.Edit;
    datamodule3.Table1.FieldName('Couter').asinteger:=0;
    datamodule3.Table1.Post;
end;
if radiobutton14.Checked then
begin
    datamodule3.Table1.MoveBy(13);
    datamodule3.Table1.Edit;
    datamodule3.Table1.FieldName('Couter').asinteger:=0;
    datamodule3.Table1.Post;
end;
if radiobutton15.Checked then
begin
    datamodule3.Table1.MoveBy(14);
    datamodule3.Table1.Edit;
    datamodule3.Table1.FieldName('Couter').asinteger:=0;
    datamodule3.Table1.Post;
end;
if radiobutton16.Checked then
begin
    datamodule3.Table1.MoveBy(15);

```



```

datamodule3.Table1.Edit;
datamodule3.Table1.FieldName('Couter').asinteger:=0;
datamodule3.Table1.Post;
end;
end;
procedure TForm2.Button3Click(Sender: TObject);
var i,j:integer;
begin
form1.Timer1.Enabled:=false;
for i:=1 to 10000 do //delay
begin
j:=(((((((((((i+1)+i)+i)+i)+i)+i)+i)+i)+i)+i)+i);
end;
datamodule3.Table1.First;
if radiobutton1.Checked then
begin
datamodule3.Table1.Edit;
datamodule3.Table1.FieldName('Time Average').asString:='00:00:00';
datamodule3.Table1.Post;
end;
if radiobutton2.Checked then
begin
datamodule3.Table1.MoveBy(1);
datamodule3.Table1.Edit;
datamodule3.Table1.FieldName('Time Average').asString:='00:00:00';
datamodule3.Table1.Post;
end;
if radiobutton3.Checked then
begin
datamodule3.Table1.MoveBy(2);
datamodule3.Table1.Edit;
datamodule3.Table1.FieldName('Time Average').asString:='00:00:00';
datamodule3.Table1.Post;
end;
if radiobutton4.Checked then
begin
datamodule3.Table1.MoveBy(3);

```

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์ไว้เพื่อใช้ในการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

datamodule3.Table1.Edit;
datamodule3.Table1.FieldName('Time Average').asString:='00:00:00';
datamodule3.Table1.Post;
end;
if radiobutton5.Checked then
begin
datamodule3.Table1.MoveBy(4);
datamodule3.Table1.Edit;
datamodule3.Table1.FieldName('Time Average').asString:='00:00:00';
datamodule3.Table1.Post;
end;
if radiobutton6.Checked then
begin
datamodule3.Table1.MoveBy(5);
datamodule3.Table1.Edit;
datamodule3.Table1.FieldName('Time Average').asString:='00:00:00';
datamodule3.Table1.Post;
end;
if radiobutton7.Checked then
begin
datamodule3.Table1.MoveBy(6);
datamodule3.Table1.Edit;
datamodule3.Table1.FieldName('Time Average').asString:='00:00:00';
datamodule3.Table1.Post;
end;
if radiobutton8.Checked then
begin
datamodule3.Table1.MoveBy(7);
datamodule3.Table1.Edit;
datamodule3.Table1.FieldName('Time Average').asString:='00:00:00';
datamodule3.Table1.Post;
end;
if radiobutton9.Checked then
begin
datamodule3.Table1.MoveBy(8);

```

```
datamodule3.Table1.Edit;
```

```
datamodule3.Table1.FieldName('Time Average').asString:='00:00:00';
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

    datamodule3.Table1.Post;
end;
if radiobutton10.Checked then
begin
    datamodule3.Table1.MoveBy(9);
    datamodule3.Table1.Edit;
    datamodule3.Table1.FieldName('Time Average').asString:='00:00:00';
    datamodule3.Table1.Post;
end;
if radiobutton11.Checked then
begin
    datamodule3.Table1.MoveBy(10);
    datamodule3.Table1.Edit;
    datamodule3.Table1.FieldName('Time Average').asString:='00:00:00';
    datamodule3.Table1.Post;
end;
if radiobutton12.Checked then
begin
    datamodule3.Table1.MoveBy(11);
    datamodule3.Table1.Edit;
    datamodule3.Table1.FieldName('Time Average').asString:='00:00:00';
    datamodule3.Table1.Post;
end;
if radiobutton13.Checked then
begin
    datamodule3.Table1.MoveBy(12);
    datamodule3.Table1.Edit;
    datamodule3.Table1.FieldName('Time Average').asString:='00:00:00';
    datamodule3.Table1.Post;
end;
if radiobutton14.Checked then
begin
    datamodule3.Table1.MoveBy(13);
    datamodule3.Table1.Edit;
    datamodule3.Table1.FieldName('Time Average').asString:='00:00:00';
    datamodule3.Table1.Post;

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
 ้  
 ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

if radiobutton15.Checked then
begin
    datamodule3.Table1.MoveBy(14);
    datamodule3.Table1.Edit;
    datamodule3.Table1.FieldName('Time Average').asString:='00:00:00';
    datamodule3.Table1.Post;
end;
if radiobutton16 Checked then
begin
    datamodule3.Table1.MoveBy(15);
    datamodule3.Table1.Edit;
    datamodule3.Table1.FieldName('Time Average').asString:='00:00:00';
    datamodule3.Table1.Post;
end;
end;

procedure TForm2.Button4Click(Sender: TObject);
VAR I,j,k:INTEGER;
begin
    form1.Timer1.Enabled:=false;
    for k:=1 to 10000 do    for i:=1 to 10000 do //delay
    begin
        j:=(((((((((((i+1)+i)+i)+i)+1)+1)+1)+1)+1)+1)+1)+1);
    end;
    datamodule3.Table1.first;
    datamodule3.Table1.Edit;
    datamodule3.Table1.FieldName('Time Average').asString:='00:00:00';
    datamodule3.Table1.FieldName('Timeon').asString:='00:00:00';
    datamodule3.Table1.FieldName('TimeOFF').asString:='00:00:00';
    datamodule3.Table1.FieldName('Couter').asinteger:=0;
    datamodule3.Table1.Post;
    datamodule3.Table2.first;
    datamodule3.Table2.Edit;
    datamodule3.Table2.FieldName('Time Check').asString:='00:00:00';
    datamodule3.Table2.FieldName('Result').asinteger:=0;
    datamodule3.Table2.Post;

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
**FOR I:=1 TO 15 DO**  
 ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

BEGIN
datamodule3.Table1.Next;
datamodule3.Table1.Edit;
datamodule3.Table1.FieldName('Time Average').asString:='00:00:00';
datamodule3.Table1.FieldName('Couter').asinteger:=0;
datamodule3.Table1.FieldName('Timeon').asString:='00:00:00';
datamodule3.Table1.FieldName('TimeOFF').asString:='00:00:00';
datamodule3.Table1.Post;
datamodule3.Table2.Next;
datamodule3.Table2.Edit;
datamodule3.Table2.FieldName('Time Check').asString:='00:00:00';
datamodule3.Table2.FieldName('Result').asinteger:=0;
datamodule3.Table2.Post;
END;
end;

procedure TForm2.Button5Click(Sender: TObject);
var i,j:integer;
begin
form1.Timer1.Enabled:=false;
for i:=1 to 10000 do //delay
begin
j:=(((((((((((i+1)+i)+j)+i)+1)+1)+1)+1)+1)+1)+1)+1);
end;
begin
datamodule3.Table1.first;
if radiobutton1.Checked then
begin
datamodule3.Table1.Edit;
datamodule3.Table1.FieldName('Timeon').asString:='00:00:00';
datamodule3.Table1.FieldName('TimeOFF').asString:='00:00:00';
datamodule3.Table1.Post;
end;
if radiobutton2.Checked then
begin
datamodule3.Table1.MoveBy(1);
datamodule3.Table1.Edit;

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

datamodule3.Table1.FieldByName('Timeon').asstring:='00:00:00';
datamodule3.Table1.FieldByName('TimeOFF').asstring:='00:00:00';
datamodule3.Table1.Post;
end;
if radiobutton3.Checked then
begin
datamodule3.Table1.MoveBy(2);
datamodule3.Table1.Edit;
datamodule3.Table1.FieldByName('Timeon').asstring:='00:00:00';
datamodule3.Table1.FieldByName('TimeOFF').asstring:='00:00:00';
datamodule3.Table1.Post;
end;
if radiobutton4.Checked then
begin
datamodule3.Table1.MoveBy(3);
datamodule3.Table1.Edit;
datamodule3.Table1.FieldByName('Timeon').asstring:='00:00:00';
datamodule3.Table1.FieldByName('TimeOFF').asstring:='00:00:00';
datamodule3.Table1.Post;
end;
if radiobutton5.Checked then
begin
datamodule3.Table1.MoveBy(4);
datamodule3.Table1.Edit;
datamodule3.Table1.FieldByName('Timeon').asstring:='00:00:00';
datamodule3.Table1.FieldByName('TimeOFF').asstring:='00:00:00';
datamodule3.Table1.Post;
end;
if radiobutton6.Checked then
begin
datamodule3.Table1.MoveBy(5);
datamodule3.Table1.Edit;
datamodule3.Table1.FieldByName('Timeon').asstring:='00:00:00';
datamodule3.Table1.FieldByName('TimeOFF').asstring:='00:00:00';
datamodule3.Table1.Post;
end;

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมีเหตุดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

begin
datamodule3.Table1.MoveBy(6);
datamodule3.Table1.Edit;
datamodule3.Table1.FieldName('Timeon').asString:='00:00:00';
datamodule3.Table1.FieldName('TimeOFF').asString:='00:00:00';
datamodule3.Table1.Post;
end;
if radiobutton8.Checked then
begin
datamodule3.Table1.MoveBy(7);
datamodule3.Table1.Edit;
datamodule3.Table1.FieldName('Timeon').asString:='00:00:00';
datamodule3.Table1.FieldName('TimeOFF').asString:='00:00:00';
datamodule3.Table1.Post;
end;
if radiobutton9.Checked then
begin
datamodule3.Table1.MoveBy(8);
datamodule3.Table1.Edit;
datamodule3.Table1.FieldName('Timeon').asString:='00:00:00';
datamodule3.Table1.FieldName('TimeOFF').asString:='00:00:00';
datamodule3.Table1.Post;
end;
if radiobutton10.Checked then
begin
datamodule3.Table1.MoveBy(9);
datamodule3.Table1.Edit;
datamodule3.Table1.FieldName('Timeon').asString:='00:00:00';
datamodule3.Table1.FieldName('TimeOFF').asString:='00:00:00';
datamodule3.Table1.Post;
end;
if radiobutton11.Checked then
begin
datamodule3.Table1.MoveBy(10);
datamodule3.Table1.Edit;
datamodule3.Table1.FieldName('Timeon').asString:='00:00:00';
datamodule3.Table1.FieldName('TimeOFF').asString:='00:00:00';

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

datamodule3.Table1.Post;
end;
if radiobutton12.Checked then
begin
datamodule3.Table1.MoveBy(11);
datamodule3.Table1.Edit;
datamodule3.Table1.FieldName('Timeon').asstring:='00:00:00';
datamodule3.Table1.FieldName('TimeOFF').asstring:='00:00:00';
datamodule3.Table1.Post;
end;
if radiobutton13.Checked then
begin
datamodule3.Table1.MoveBy(12);
datamodule3.Table1.Edit;
datamodule3.Table1.FieldName('Timeon').asstring:='00:00:00';
datamodule3.Table1.FieldName('TimeOFF').asstring:='00:00:00';
datamodule3.Table1.Post;
end;
if radiobutton14.Checked then
begin
datamodule3.Table1.MoveBy(13);
datamodule3.Table1.Edit;
datamodule3.Table1.FieldName('Timeon').asstring:='00:00:00';
datamodule3.Table1.FieldName('TimeOFF').asstring:='00:00:00';
datamodule3.Table1.Post;
end;
if radiobutton15.Checked then
begin
datamodule3.Table1.MoveBy(14);
datamodule3.Table1.Edit;
datamodule3.Table1.FieldName('Timeon').asstring:='00:00:00';
datamodule3.Table1.FieldName('TimeOFF').asstring:='00:00:00';
datamodule3.Table1.Post;
end;
if radiobutton16.Checked then
begin
datamodule3.Table1.MoveBy(15);

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

datamodule3.Table1.Edit;
datamodule3.Table1.FieldName('Timeon').asstring:='00:00:00';
datamodule3.Table1.FieldName('TimeOFF').asstring:='00:00:00';
datamodule3.Table1.Post;
end;
end;
end;
procedure TForm2.Button6Click(Sender: TObject);
var i,j:integer;
begin
form1.Timer1.Enabled:=false;
for i:=1 to 10000 do //delay
begin
j:=(((((((((((i+1)+i)+i)+i)+I)+I)+I)+I)+I)+I)+I);
end;
begin
datamodule3.Table2.first;
if radiobutton1.Checked then
begin
datamodule3.Table2.Edit;
datamodule3.Table2.FieldName('Time Check').asstring:='00:00:00';
datamodule3.Table2.Post;
end;
if radiobutton2.Checked then
begin
datamodule3.Table2.MoveBy(1);
datamodule3.Table2.Edit;
datamodule3.Table2.FieldName('Time Check').asstring:='00:00:00';
datamodule3.Table2.Post;
end;
if radiobutton3.Checked then
begin
datamodule3.Table2.MoveBy(2);
datamodule3.Table2.Edit;
datamodule3.Table2.FieldName('Time Check').asstring:='00:00:00';
datamodule3.Table2.Post;

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

if radiobutton4.Checked then
begin
datamodule3.Table2.MoveBy(3);
datamodule3.Table2.Edit;
datamodule3.Table2.FieldByName('Time Check').asstring:='00:00:00';
datamodule3.Table2.Post;
end;
if radiobutton5.Checked then
begin
datamodule3.Table2.MoveBy(4);
datamodule3.Table2.Edit;
datamodule3.Table2.FieldByName('Time Check').asstring:='00:00:00';
datamodule3.Table2.Post;
end;
if radiobutton6.Checked then
begin
datamodule3.Table2.MoveBy(5);
datamodule3.Table2.Edit;
datamodule3.Table2.FieldByName('Time Check').asstring:='00:00:00';
datamodule3.Table2.Post;
end;
if radiobutton7.Checked then
begin
datamodule3.Table2.MoveBy(6);
datamodule3.Table2.Edit;
datamodule3.Table2.FieldByName('Time Check').asstring:='00:00:00';
datamodule3.Table2.Post;
end;
if radiobutton8.Checked then
begin
datamodule3.Table2.MoveBy(7);
datamodule3.Table2.Edit;
datamodule3.Table2.FieldByName('Time Check').asstring:='00:00:00';
datamodule3.Table2.Post;
end;
if radiobutton9.Checked then
begin

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

datamodule3.Table2.MoveBy(8);
datamodule3.Table2.Edit;
datamodule3.Table2.FieldName('Time Check').asString:='00:00:00';
datamodule3.Table2.Post;
end;
if radiobutton10.Checked then
begin
datamodule3.Table2.MoveBy(9);
datamodule3.Table2.Edit;
datamodule3.Table2.FieldName('Time Check').asString:='00:00:00';
datamodule3.Table2.Post;
end;
if radiobutton11.Checked then
begin
datamodule3.Table2.MoveBy(10);
datamodule3.Table2.Edit;
datamodule3.Table2.FieldName('Time Check').asString:='00:00:00';
datamodule3.Table2.Post;
end;
if radiobutton12.Checked then
begin
datamodule3.Table2.MoveBy(11);
datamodule3.Table2.Edit;
datamodule3.Table2.FieldName('Time Check').asString:='00:00:00';
datamodule3.Table2.Post;
end;
if radiobutton13.Checked then
begin
datamodule3.Table2.MoveBy(12);
datamodule3.Table2.Edit;
datamodule3.Table2.FieldName('Time Check').asString:='00:00:00';
datamodule3.Table2.Post;
end;
if radiobutton14.Checked then
begin

```

```
datamodule3.Table2.MoveBy(13);
```

```
datamodule3.Table2.Edit;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

datamodule3.Table2.FieldByName('Time Check').asstring:='00:00:00';
datamodule3.Table2.Post;

end;

if radiobutton15.Checked then
begin
datamodule3.Table2.MoveBy(14);
datamodule3.Table2.Edit;
datamodule3.Table2.FieldByName('Time Check').asstring:='00:00:00';
datamodule3.Table2.Post;

end;

if radiobutton16.Checked then
begin
datamodule3.Table2.MoveBy(15);
datamodule3.Table2.Edit;
datamodule3.Table2.FieldByName('Time Check').asstring:='00:00:00';
datamodule3.Table2.Post;

end;
end;
end;

procedure TForm2.Button7Click(Sender: TObject);
var i,j:integer;
begin
form1.Timer1.Enabled:=false;
for i:=1 to 10000 do //delay
begin
j:=((((((((((((((i+1)+i)+i)+i)+i)+i)+i)+i)+i)+i)+i)+i);

end;

begin
datamodule3.Table2.first;
if radiobutton1.Checked then
begin
datamodule3.Table2.Edit;
datamodule3.Table2.FieldByName('Result').asstring:='0';
datamodule3.Table2.Post;

end;

if radiobutton2.Checked then

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
 ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

datamodule3.Table2.MoveBy(1);
datamodule3.Table2.Edit;
datamodule3.Table2.FieldName('Result').asstring:='0';
datamodule3.Table2.Post;
end;
if radiobutton3.Checked then
begin
datamodule3.Table2.MoveBy(2);
datamodule3.Table2.Edit;
datamodule3.Table2.FieldName('Result').asstring:='0';
datamodule3.Table2.Post;
end;
if radiobutton4.Checked then
begin
datamodule3.Table2.MoveBy(3);
datamodule3.Table2.Edit;
datamodule3.Table2.FieldName('Result').asstring:='0';
datamodule3.Table2.Post;
end;
if radiobutton5.Checked then
begin
datamodule3.Table2.MoveBy(4);
datamodule3.Table2.Edit;
datamodule3.Table2.FieldName('Result').asstring:='0';
datamodule3.Table2.Post;
end;
if radiobutton6.Checked then
begin
datamodule3.Table2.MoveBy(5);
datamodule3.Table2.Edit;
datamodule3.Table2.FieldName('Result').asstring:='0';
datamodule3.Table2.Post;
end;
if radiobutton7.Checked then
begin
datamodule3.Table2.MoveBy(6);
datamodule3.Table2.Edit;

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

datamodule3.Table2.FieldByName('Result').asString:='0';
datamodule3.Table2.Post;
end;
if radiobutton8.Checked then
begin
datamodule3.Table2.MoveBy(7);
datamodule3.Table2.Edit;
datamodule3.Table2.FieldByName('Result').asString:='0';
datamodule3.Table2.Post;
end;
if radiobutton9.Checked then
begin
datamodule3.Table2.MoveBy(8);
datamodule3.Table2.Edit;
datamodule3.Table2.FieldByName('Result').asString:='0';
datamodule3.Table2.Post;
end;
if radiobutton10.Checked then
begin
datamodule3.Table2.MoveBy(9);
datamodule3.Table2.Edit;
datamodule3.Table2.FieldByName('Result').asString:='0';
datamodule3.Table2.Post;
end;
if radiobutton11.Checked then
begin
datamodule3.Table2.MoveBy(10);
datamodule3.Table2.Edit;
datamodule3.Table2.FieldByName('Result').asString:='0';
datamodule3.Table2.Post;
end;
if radiobutton12.Checked then
begin
datamodule3.Table2.MoveBy(11);
datamodule3.Table2.Edit;
datamodule3.Table2.FieldByName('Result').asString:='0';
datamodule3.Table2.Post;

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

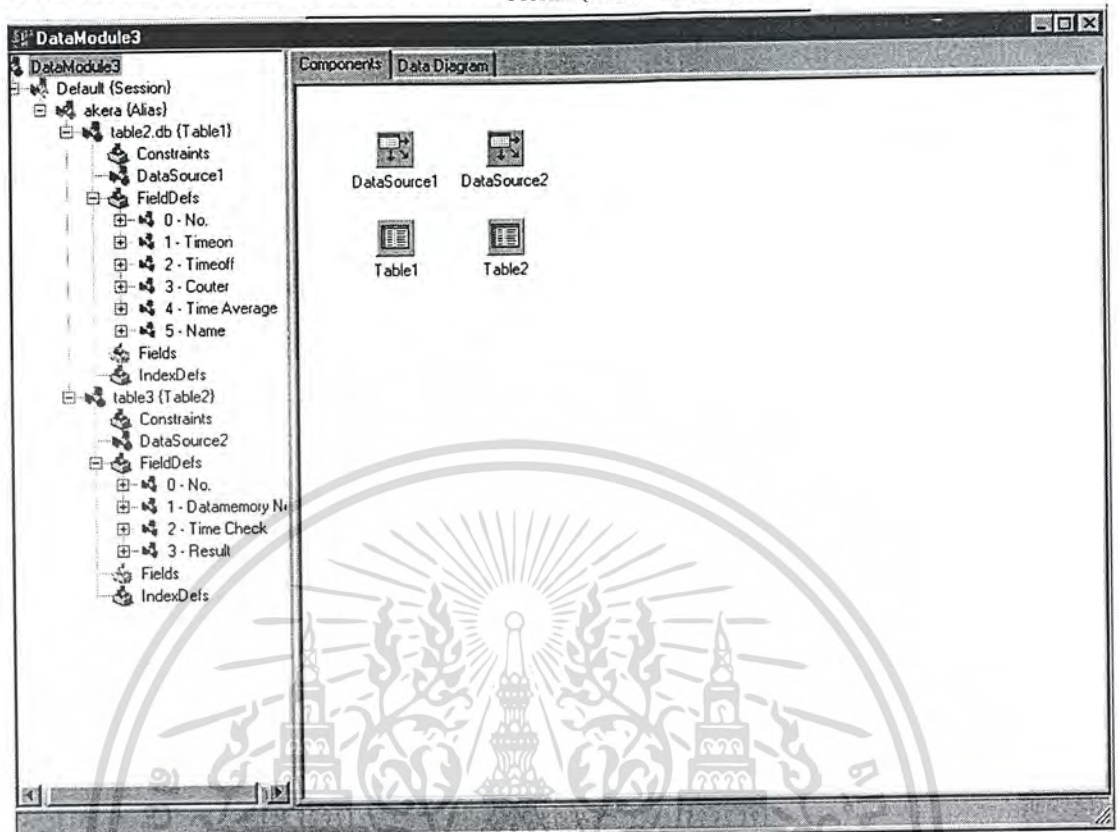
end;
if radiobutton13.Checked then
begin
datamodule3.Table2.MoveBy(12);
datamodule3.Table2.Edit;
datamodule3.Table2.FieldName('Result').asString:='0';
datamodule3.Table2.Post;
end;
if radiobutton14.Checked then
begin
datamodule3.Table2.MoveBy(13);
datamodule3.Table2.Edit;
datamodule3.Table2.FieldName('Result').asString:='0';
datamodule3.Table2.Post;
end;
if radiobutton15.Checked then
begin
datamodule3.Table2.MoveBy(14);
datamodule3.Table2.Edit;
datamodule3.Table2.FieldName('Result').asString:='0';
datamodule3.Table2.Post;
end;
if radiobutton16.Checked then
begin
datamodule3.Table2.MoveBy(15);
datamodule3.Table2.Edit;
datamodule3.Table2.FieldName('Result').asString:='0';
datamodule3.Table2.Post;
end;
end;
end;
end;
end.

```



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

\*\*\*\*\*Tform3(ComLinkPLC.)\*\*\*\*\*



รูปที่ ๓3 ภาพแสดง (DataMemory) Form ที่ 3 ของโปรแกรม ComLinkPLC.

```

unit Unit3;
interface
uses
  Windows, Messages, SysUtils, Classes, Graphics, Controls, Forms, Dialogs,
  Db, DBTables;
type
  TDataModule3 = class(TDataModule)
    DataSource1: TDataSource;
    Table1: TTable;
    Table2: TTable;
    DataSource2: TDataSource;
  procedure DataModuleCreate(Sender: TObject);
  private
    { Private declarations }
  public
    { Public declarations }
  end;

```

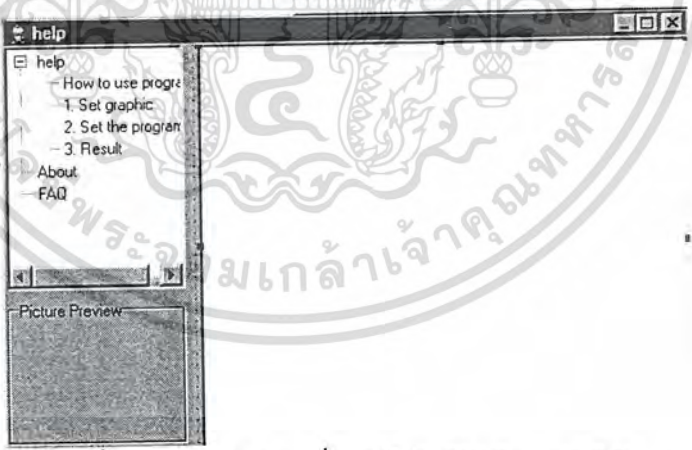
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

DataModule3: TDataModule3;
implementation
uses unit1;
{SR *.DFM}

procedure TDataModule3.DataModuleCreate(Sender: TObject);
var j:integer;
begin
for j:= 1 to 16 do
begin
datamodule3.Table1.First;
datamodule3.Table1.MoveBy(j-1);
datamodule3.Table1.Edit;
datamodule3.Table1.FieldName('Name').asString:=form1.table1.FieldName('N'+inttostr(j)).asString;
datamodule3.Table1.Post;
end;
end;
end.
*****Tform4(ComLinkPLC.)*****

```



รูปที่ ๗4 ภาพแสดง Form ที่4 ของโปรแกรม ComLinkPLC.

```

unit Unit4;
interface
uses
Windows, Messages, SysUtils, Classes, Graphics, Controls, Forms, Dialogs,
ExtCtrls, StdCtrls, ComCtrls;
type

```

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
 ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

TreeView1: TTreeView;
Memo1: TMemo;
GroupBox1: TGroupBox;
procedure TreeView1Change(Sender: TObject; Node: TTreeNode);
private
  { Private declarations }
public
  { Public declarations }
end;
var
  Form4: TForm4;
implementation
{$SR *.DFM}
procedure TForm4.TreeView1Change(Sender: TObject; Node: TTreeNode);
begin
  if treeview1.Selected.Level = 1 then
  begin
    case treeview1.Selected.Index of
      0 : begin
          memo1.Lines.Clear;
          memo1.Lines.LoadFromFile('c:\1.txt');
          end;
      1 : begin
          memo1.Lines.Clear;
          memo1.Lines.LoadFromFile('c:\2.txt');
          end;
      2 : begin
          memo1.Lines.Clear;
          memo1.Lines.LoadFromFile('c:\3.txt');
          end;
      3 : begin
          memo1.Lines.Clear;
          memo1.Lines.LoadFromFile('c:\netlog.txt');
          end;
    end;
  end;
end;
end;

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
 ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
begin
case treeview1.Selected.Index of
0 : begin
memo1.Lines.Clear;
memo1.Lines.LoadFromFile('c:\1.txt');
end;
1 : begin
memo1.Lines.Clear;
memo1.Lines.LoadFromFile('c:\2.txt');
end;
2 : begin
memo1.Lines.Clear;
memo1.Lines.LoadFromFile('c:\3.txt');
end;
3 : begin
memo1.Lines.Clear;
memo1.Lines.LoadFromFile('c:\netlog.txt');
end;
end;
end;
end;
end.
```



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Print Preview

1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20

Page: (Page:)

BIT DETECT

Name	Time Average	Counter	Last Time On	Last Time Off
Title				

DATAMEMORY DETECT

Name	Result	Time Check
ColumnHeaderBand1: TQRBand		

Name	Time Average	Counter	Time on	Time off
Sub Detail				
Datamemory No	Result	Time Check		
Sub Detail				

Time : (Time)

Page Footer

รูปที่ ๘5 ภาพแสดง Form ที่ 5 ของโปรแกรม ComLinkPLC.

unit Unit5;

interface

uses

Windows, Messages, SysUtils, Classes, Graphics, Controls, Forms, Dialogs,  
ExtCtrls, QuickRpt, QrCtrls;

type

TForm5 = class(TForm)

QuickRep1: TQuickRep;

PageHeaderBand1: TQRBand;

TitleBand1: TQRBand;

ColumnHeaderBand1: TQRBand;

QRSysData1: TQRSysData;

QRLabel1: TQRLabel;

QRLabel2: TQRLabel;

QRLabel4: TQRLabel;

QRLabel6: TQRLabel;

QRSubDetail1: TQRSubDetail;

QRDBText1: TQRDBText;

เอกสารนี้เป็น QRDBText1: TQRDBText; การใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

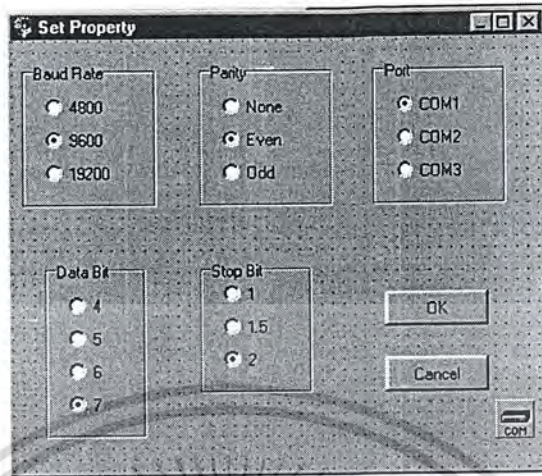
QRDBText2: TQRDBText;
QRDBText3: TQRDBText;
QRDBText4: TQRDBText;
QRDBText5: TQRDBText;
PageFooterBand1: TQRBand;
QRSysData2: TQRSysData;
QRLabel8: TQRLabel;
QRLabel9: TQRLabel;
QRLabel10: TQRLabel;
QRLabel11: TQRLabel;
QRLabel12: TQRLabel;
QRSubDetail2: TQRSubDetail;
QRDBText6: TQRDBText;
QRDBText7: TQRDBText;
QRDBText8: TQRDBText;
QRLabel3: TQRLabel;
private
  { Private declarations }
public
  { Public declarations }
end;
var
  Form5: TForm5;
implementation
uses unit1,unit3;
{$R *.DFM}
end.

```



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

\*\*\*\*\*Tform6(ComLinkPLC.)\*\*\*\*\*



รูปที่ ผ6 ภาพแสดง Form ที่6 ของโปรแกรม ComLinkPLC.

```
unit Unit6;  
interface  
uses  
  Windows, Messages, SysUtils, Classes, Graphics, Controls, Forms, Dialogs,  
  StdCtrls, ComPort;  
type  
  TForm6 = class(TForm)  
    GroupBox1: TGroupBox;  
    RadioButton1: TRadioButton;  
    RadioButton2: TRadioButton;  
    RadioButton3: TRadioButton;  
    GroupBox2: TGroupBox;  
    RadioButton4: TRadioButton;  
    RadioButton5: TRadioButton;  
    RadioButton6: TRadioButton;  
    GroupBox3: TGroupBox;  
    RadioButton7: TRadioButton;  
    RadioButton8: TRadioButton;  
    RadioButton9: TRadioButton;  
    GroupBox4: TGroupBox;  
    RadioButton10: TRadioButton;  
    RadioButton11: TRadioButton;  
    RadioButton12: TRadioButton;  
    RadioButton13: TRadioButton;
```

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์ไว้สำหรับใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

GroupBox5: TGroupBox;
RadioButton14: TRadioButton;
RadioButton15: TRadioButton;
RadioButton16: TRadioButton;
Button1: TButton;
Button2: TButton;
ComPort1: TComPort;
procedure Button1Click(Sender: TObject);
procedure Button2Click(Sender: TObject):
private
  { Private declarations }
public
  { Public declarations }
end;
var
  Form6: TForm6;
implementation
uses unit1;
{$R *.DFM}
procedure TForm6.Button1Click(Sender: TObject);
begin
  if radiobutton1.Checked then form1.ComPort1.BaudRate := br4800;
  if radiobutton2.Checked then form1.ComPort1.BaudRate := br9600;
  if radiobutton3.Checked then form1.ComPort1.BaudRate := br10400;
  if radiobutton4.Checked then form1.ComPort1.Parity := panone;
  if radiobutton5.Checked then form1.ComPort1.Parity := paeven;
  if radiobutton6.Checked then form1.ComPort1.Parity := paodd;
  if radiobutton7.Checked then form1.ComPort1.DeviceName := 'com1';
  if radiobutton8.Checked then form1.ComPort1.DeviceName := 'com2';
  if radiobutton9.Checked then form1.ComPort1.DeviceName := 'com3';
  if radiobutton10.Checked then form1.ComPort1.DataBits := Db4;
  if radiobutton11.Checked then form1.ComPort1.DataBits := db5;
  if radiobutton12.Checked then form1.ComPort1.DataBits := db6;
  if radiobutton13.Checked then form1.ComPort1.DataBits := db7;
  if radiobutton14.Checked then form1.ComPort1.StopBits := sb1;
  if radiobutton15.Checked then form1.ComPort1.StopBits := sb1_5;
  if radiobutton16.Checked then form1.ComPort1.StopBits := sb2;

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

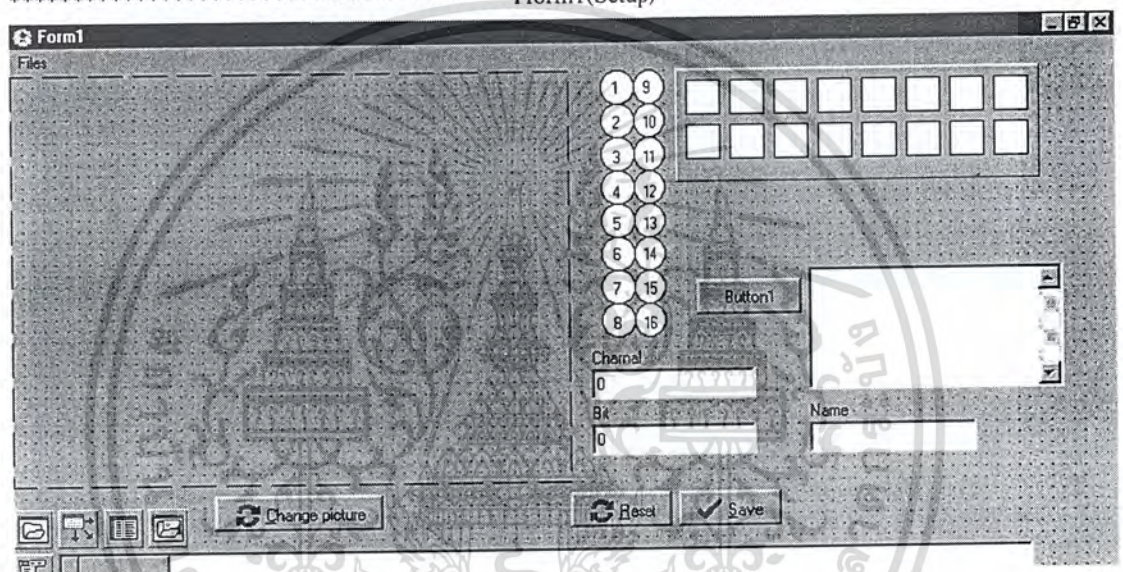
form6.Hide;
form1.Show;
end;
procedure TForm6.Button2Click(Sender: TObject);
begin
form6.Hide;
form1.Show;
end;
end.

```

```

*****Tform1(Setup)*****

```



รูปที่ ๗7 ภาพแสดง Form ที่ 1 ของโปรแกรม Setup

```

unit Unit1;

```

```

interface

```

```

uses

```

```

Windows, Messages, SysUtils, Classes, Graphics, Controls, Forms, Dialogs,
AppEvnts, ExtCtrls, StdCtrls, Buttons, jpeg, ActnList, Grids, DBGrids,
Db, DBTables, Menus, ExtDlgs;

```

```

type

```

```

TForm1 = class(TForm)

```

```

    BitBtn1: TBitBtn;

```

```

    BitBtn2: TBitBtn;

```

```

    Edit1: TEdit;

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Edit2: TEdit;  
Image1: TImage;  
Shape1: TShape;  
Shape2: TShape;  
Shape3: TShape;  
Shape4: TShape;  
Shape5: TShape;  
Shape6: TShape;  
Shape7: TShape;  
Shape8: TShape;  
Shape9: TShape;  
Shape10: TShape;  
Shape11: TShape;  
Shape12: TShape;  
Shape13: TShape;  
Shape14: TShape;  
Shape15: TShape;  
Shape16: TShape;  
DataSource1: TDataSource;  
Table1: TTable;  
DBGrid1: TDBGrid;  
OpenPictureDialog1: TOpenPictureDialog;  
MainMenu1: TMainMenu;  
Files1: TMenuItem;  
Open1: TMenuItem;  
open2: TMenuItem;  
save1: TMenuItem;  
BitBtn3: TBitBtn;  
Exit1: TMenuItem;  
Label1: TLabel;  
Label2: TLabel;  
OpenDialog1: TOpenDialog;  
Panel1: TPanel;  
Shape100: TShape;  
Shape101: TShape;  
Shape102: TShape;  
Shape103: TShape;



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

Shape104: TShape;
Shape105: TShape;
Shape106: TShape;
Shape107: TShape;
Shape108: TShape;
Shape109: TShape;
Shape110: TShape;
Shape111: TShape;
Shape112: TShape;
Shape113: TShape;
Shape114: TShape;
Shape115: TShape;
Button1: TButton;
Memo1: TMemo;
I1: TLabel;
L2: TLabel;
L3: TLabel;
L4: TLabel;
L5: TLabel;
L6: TLabel;
L7: TLabel;
L8: TLabel;
L9: TLabel;
L10: TLabel;
L11: TLabel;
L12: TLabel;
L13: TLabel;
L14: TLabel;
L15: TLabel;
L16: TLabel;
Edit3: TEdit;
Label3: TLabel;

procedure Shape1MouseDown(Sender: TObject; Button: TMouseButton;
  Shift: TShiftState; X, Y: Integer);

procedure ShapeMouseDown(Sender: TObject; Button: TMouseButton;
  Shift: TShiftState; X, Y: Integer);
procedure ShapMouseDown(Sender: TObject; Button: TMouseButton;

```



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
Shift: TShiftState; X, Y: Integer);
procedure Shape4MouseDown(Sender: TObject; Button: TMouseButton;
  Shift: TShiftState; X, Y: Integer);
procedure Shape5MouseDown(Sender: TObject; Button: TMouseButton;
  Shift: TShiftState; X, Y: Integer);
procedure Shape6MouseDown(Sender: TObject; Button: TMouseButton;
  Shift: TShiftState; X, Y: Integer);
procedure Shape7MouseDown(Sender: TObject; Button: TMouseButton;
  Shift: TShiftState; X, Y: Integer);
procedure Shape8MouseDown(Sender: TObject; Button: TMouseButton;
  Shift: TShiftState; X, Y: Integer);
procedure Shape9MouseDown(Sender: TObject; Button: TMouseButton;
  Shift: TShiftState; X, Y: Integer);
procedure Shape10MouseDown(Sender: TObject; Button: TMouseButton;
  Shift: TShiftState; X, Y: Integer);
procedure Shape11MouseDown(Sender: TObject; Button: TMouseButton;
  Shift: TShiftState; X, Y: Integer);
procedure Shape12MouseDown(Sender: TObject; Button: TMouseButton;
  Shift: TShiftState; X, Y: Integer);
procedure Shape13MouseDown(Sender: TObject; Button: TMouseButton;
  Shift: TShiftState; X, Y: Integer);
procedure Shape15MouseDown(Sender: TObject; Button: TMouseButton;
  Shift: TShiftState; X, Y: Integer);
procedure Shape16MouseDown(Sender: TObject; Button: TMouseButton;
  Shift: TShiftState; X, Y: Integer);
procedure Shape14MouseDown(Sender: TObject; Button: TMouseButton;
  Shift: TShiftState; X, Y: Integer);
procedure Shape1MouseDownMove(Sender: TObject; Shift: TShiftState; X,
  Y: Integer);
procedure ShapeMouseDownMove(Sender: TObject; Shift: TShiftState; X,
  Y: Integer);
procedure ShapeMouseDownMove(Sender: TObject; Shift: TShiftState; X,
  Y: Integer);
procedure Shape4MouseDownMove(Sender: TObject; Shift: TShiftState; X,
  Y: Integer);
procedure Shape5MouseDownMove(Sender: TObject; Shift: TShiftState; X,
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
procedure Shape6MouseMove(Sender: TObject; Shift: TShiftState; X,
  Y: Integer);
procedure Shape7MouseMove(Sender: TObject; Shift: TShiftState; X,
  Y: Integer);
procedure Shape8MouseMove(Sender: TObject; Shift: TShiftState; X,
  Y: Integer);
procedure Shape9MouseMove(Sender: TObject; Shift: TShiftState; X,
  Y: Integer);
procedure Shape10MouseMove(Sender: TObject; Shift: TShiftState; X,
  Y: Integer);
procedure Shape11MouseMove(Sender: TObject; Shift: TShiftState; X,
  Y: Integer);
procedure Shape12MouseMove(Sender: TObject; Shift: TShiftState; X,
  Y: Integer);
procedure Shape13MouseMove(Sender: TObject; Shift: TShiftState; X,
  Y: Integer);
procedure Shape16MouseMove(Sender: TObject; Shift: TShiftState; X,
  Y: Integer);
procedure Shape15MouseMove(Sender: TObject; Shift: TShiftState; X,
  Y: Integer);
procedure Shape14MouseMove(Sender: TObject; Shift: TShiftState; X,
  Y: Integer);
procedure ShapeMouseUp(Sender: TObject; Button: TMouseButton;
  Shift: TShiftState; X, Y: Integer);
procedure Shape2MouseUp(Sender: TObject; Button: TMouseButton;
  Shift: TShiftState; X, Y: Integer);
procedure ShapMouseUp(Sender: TObject; Button: TMouseButton;
  Shift: TShiftState; X, Y: Integer);
procedure Shape4MouseUp(Sender: TObject; Button: TMouseButton;
  Shift: TShiftState; X, Y: Integer);
procedure Shape5MouseUp(Sender: TObject; Button: TMouseButton;
  Shift: TShiftState; X, Y: Integer);
procedure Shape6MouseUp(Sender: TObject; Button: TMouseButton;
  Shift: TShiftState; X, Y: Integer);
procedure Shape7MouseUp(Sender: TObject; Button: TMouseButton;
  Shift: TShiftState; X, Y: Integer);
procedure Shape8MouseUp(Sender: TObject; Button: TMouseButton;
```

เอกสารนี้เป็นเอกสารที่สามารถนำส่วนใดส่วนหนึ่งหรือทั้งหมดไปเผยแพร่โดยไม่ก่อให้เกิดความเสียหายทางวิชาการ  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
Shift: TShiftState; X, Y: Integer);
procedure Shape9MouseUp(Sender: TObject; Button: TMouseButton;
Shift: TShiftState; X, Y: Integer);
procedure Shape10MouseUp(Sender: TObject; Button: TMouseButton;
Shift: TShiftState; X, Y: Integer);
procedure Shape11MouseUp(Sender: TObject; Button: TMouseButton;
Shift: TShiftState; X, Y: Integer);
procedure Shape12MouseUp(Sender: TObject; Button: TMouseButton;
Shift: TShiftState; X, Y: Integer);
procedure Shape13MouseUp(Sender: TObject; Button: TMouseButton;
Shift: TShiftState; X, Y: Integer);
procedure Shape16MouseUp(Sender: TObject; Button: TMouseButton;
Shift: TShiftState; X, Y: Integer);
procedure Shape15MouseUp(Sender: TObject; Button: TMouseButton;
Shift: TShiftState; X, Y: Integer);
procedure Shape14MouseUp(Sender: TObject; Button: TMouseButton;
Shift: TShiftState; X, Y: Integer);
procedure BitBtn2Click(Sender: TObject);
procedure BitBtn3Click(Sender: TObject);
procedure FormCreate(Sender: TObject);
procedure BitBtn1Click(Sender: TObject);
procedure Exit1Click(Sender: TObject);
procedure open2Click(Sender: TObject);
procedure Shape1MouseUp(Sender: TObject; Button: TMouseButton;
Shift: TShiftState; X, Y: Integer);
procedure Shape100MouseUp(Sender: TObject; Button: TMouseButton;
Shift: TShiftState; X, Y: Integer);
procedure Shape101MouseUp(Sender: TObject; Button: TMouseButton;
Shift: TShiftState; X, Y: Integer);
procedure Shape102MouseUp(Sender: TObject; Button: TMouseButton;
Shift: TShiftState; X, Y: Integer);
procedure Shape103MouseUp(Sender: TObject; Button: TMouseButton;
Shift: TShiftState; X, Y: Integer);
procedure Shape104MouseUp(Sender: TObject; Button: TMouseButton;
Shift: TShiftState; X, Y: Integer);
procedure Shape105MouseUp(Sender: TObject; Button: TMouseButton;
Shift: TShiftState; X, Y: Integer);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับใช้ในการเรียนเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
procedure Shape106MouseUp(Sender: TObject; Button: TMouseButton;
```

```
  Shift: TShiftState; X, Y: Integer);
```

```
procedure Shape107MouseUp(Sender: TObject; Button: TMouseButton;
```

```
  Shift: TShiftState; X, Y: Integer);
```

```
procedure Shape108MouseUp(Sender: TObject; Button: TMouseButton;
```

```
  Shift: TShiftState; X, Y: Integer);
```

```
procedure Shape109MouseUp(Sender: TObject; Button: TMouseButton;
```

```
  Shift: TShiftState; X, Y: Integer);
```

```
procedure Shape110MouseUp(Sender: TObject; Button: TMouseButton;
```

```
  Shift: TShiftState; X, Y: Integer);
```

```
procedure Shape111MouseUp(Sender: TObject; Button: TMouseButton;
```

```
  Shift: TShiftState; X, Y: Integer);
```

```
procedure Shape112MouseUp(Sender: TObject; Button: TMouseButton;
```

```
  Shift: TShiftState; X, Y: Integer);
```

```
procedure Shape113MouseUp(Sender: TObject; Button: TMouseButton;
```

```
  Shift: TShiftState; X, Y: Integer);
```

```
procedure Shape114MouseUp(Sender: TObject; Button: TMouseButton;
```

```
  Shift: TShiftState; X, Y: Integer);
```

```
procedure Shape115MouseUp(Sender: TObject; Button: TMouseButton;
```

```
  Shift: TShiftState; X, Y: Integer);
```

```
procedure Button1Click(Sender: TObject);
```

```
private
```

```
  { Private declarations }
```

```
public
```

```
  { Public declarations }
```

```
end;
```

```
var
```

```
  Form1: TForm1;
```

```
  startx,starty:integer;
```

```
b1,b2,b3,b4,b5,b6,b7,b8,b9,b10,b11,b12,b13,b14,b15,b16,b17,b18,b19,b20,b21,b22,b23,b24,b25,b26,b27,b28,b
```

```
29,b30,b31,b32,
```

```
c1,c2,c3,c4,c5,c6,c7,c8,c9,c10,c11,c12,c13,c14,c15,c16,c17,c18,c19,c20,c21,c22,c23,c24,c25,c26,c27,c28,c29,
```

```
c30,c31,c32:string;
```

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
d : array[1..16]of string;
```

```
implementation
```

```
{SR *.DFM}
```

```
procedure TForm1.Shape1MouseDown(Sender: TObject; Button: TMouseButton;
```

```
  Shift: TShiftState; X, Y: Integer);
```

```
begin
```

```
  startx :=x;
```

```
  starty :=y;
```

```
end;
```

```
procedure TForm1.ShapeMouseDn(Sender: TObject; Button: TMouseButton;
```

```
  Shift: TShiftState; X, Y: Integer);
```

```
begin
```

```
  startx :=x;
```

```
  starty :=y;
```

```
end;
```

```
procedure TForm1.ShapMouseDown(Sender: TObject; Button: TMouseButton;
```

```
  Shift: TShiftState; X, Y: Integer);
```

```
begin
```

```
  startx :=x;
```

```
  starty :=y;
```

```
end;
```

```
procedure TForm1.Shape4MouseDown(Sender: TObject; Button: TMouseButton;
```

```
  Shift: TShiftState; X, Y: Integer);
```

```
begin
```

```
  startx :=x;
```

```
  starty :=y;
```

```
end;
```

```
procedure TForm1.Shape5MouseDown(Sender: TObject; Button: TMouseButton;
```

```
  Shift: TShiftState; X, Y: Integer);
```

```
begin
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
startx :=x;  
starty :=y;  
end;
```

```
procedure TForm1.Shape6MouseDown(Sender: TObject; Button: TMouseButton;  
  Shift: TShiftState; X, Y: Integer);  
begin  
startx :=x;  
starty :=y;  
end;
```

```
procedure TForm1.Shape7MouseDown(Sender: TObject; Button: TMouseButton;  
  Shift: TShiftState; X, Y: Integer);  
begin  
startx :=x;  
starty :=y;  
end;
```

```
procedure TForm1.Shape8MouseDown(Sender: TObject; Button: TMouseButton;  
  Shift: TShiftState; X, Y: Integer);  
begin  
startx :=x;  
starty :=y;  
end;
```

```
procedure TForm1.Shape9MouseDown(Sender: TObject; Button: TMouseButton;  
  Shift: TShiftState; X, Y: Integer);  
begin  
startx :=x;  
starty :=y;  
end;
```

```
procedure TForm1.Shape10MouseDown(Sender: TObject; Button: TMouseButton;  
  Shift: TShiftState; X, Y: Integer);  
begin
```

```
startx :=x;  
starty :=y;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
end;
```

```
procedure TForm1.Shape11MouseDown(Sender: TObject; Button: TMouseButton;
```

```
  Shift: TShiftState; X, Y: Integer);
```

```
begin
```

```
  startx :=x;
```

```
  starty :=y;
```

```
end;
```

```
procedure TForm1.Shape12MouseDown(Sender: TObject; Button: TMouseButton;
```

```
  Shift: TShiftState; X, Y: Integer);
```

```
begin
```

```
  startx :=x;
```

```
  starty :=y;
```

```
end;
```

```
procedure TForm1.Shape13MouseDown(Sender: TObject; Button: TMouseButton;
```

```
  Shift: TShiftState; X, Y: Integer);
```

```
begin
```

```
  startx :=x;
```

```
  starty :=y;
```

```
end;
```

```
procedure TForm1.Shape15MouseDown(Sender: TObject; Button: TMouseButton;
```

```
  Shift: TShiftState; X, Y: Integer);
```

```
begin
```

```
  startx :=x;
```

```
  starty :=y;
```

```
end;
```

```
procedure TForm1.Shape16MouseDown(Sender: TObject; Button: TMouseButton;
```

```
  Shift: TShiftState; X, Y: Integer);
```

```
begin
```

```
  startx :=x;
```

```
  starty :=y;
```

```
end;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

procedure TForm1.Shape14MouseDown(Sender: TObject; Button: TMouseButton;
  Shift: TShiftState; X, Y: Integer);
begin
  startx :=x;
  starty :=y;
end;

```

```

procedure TForm1.Shape1MouseMove(Sender: TObject; Shift: TShiftState; X,
  Y: Integer);

```

```

begin
  if ssLeft in Shift then

```

```

begin
  shape1.Left := shape1.Left + x-startx;
  shape1.Top := shape1.Top +y-starty;
  l1.Top :=l1.Top+y-starty;
  l1.Left:=l1.Left+x-startx;
end;
end;

```

```

procedure TForm1.ShapeMouseMove(Sender: TObject; Shift: TShiftState; X,
  Y: Integer);

```

```

begin
  if ssLeft in Shift then
begin
  shape2.Left := shape2.Left + x-startx;
  shape2.Top := shape2.Top +y-starty;
  l2.Top :=l2.Top+y-starty;
  l2.Left:=l2.Left+x-startx;
end;
end;

```

```

procedure TForm1.ShapMouseMove(Sender: TObject; Shift: TShiftState; X,
  Y: Integer);

```

```

begin
  if ssLeft in Shift then

```

```

begin
  shape3.Left := shape3.Left + x-startx;

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

shape3.Top := shape3.Top +y-starty;
l3.Top :=l3.Top+y-starty;
l3.Left:=l3.Left+x-startx;
end;
end;

```

```

procedure TForm1.Shape4MouseMove(Sender: TObject; Shift: TShiftState; X,
Y: Integer);

```

```

begin
if ssLeft in Shift then
begin
shape4.Left := shape4.Left + x-startx;
shape4.Top := shape4.Top +y-starty;
l4.Top :=l4.Top+y-starty;
l4.Left:=l4.Left+x-startx;
end;
end;

```

```

procedure TForm1.Shape5MouseMove(Sender: TObject; Shift: TShiftState; X,
Y: Integer);

```

```

begin
if ssLeft in Shift then
begin
shape5.Left := shape5.Left + x-startx;
shape5.Top := shape5.Top +y-starty;
l5.Top :=l5.Top+y-starty;
l5.Left:=l5.Left+x-startx;
end;
end;

```

```

procedure TForm1.Shape6MouseMove(Sender: TObject; Shift: TShiftState; X,
Y: Integer);

```

```

begin
if ssLeft in Shift then
begin

```

```

shape6.Left := shape6.Left + x-startx;
shape6.Top := shape6.Top +y-starty;

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
l6.Top :=l6.Top+y-starty;  
l6.Left:=l6.Left+x-startx;  
end;  
end;
```

```
procedure TForm1.Shape7MouseMove(Sender: TObject; Shift: TShiftState; X,  
  Y: Integer);  
begin  
if ssLeft in Shift then  
begin  
shape7.Left := shape7.Left + x-startx;  
shape7.Top := shape7.Top +y-starty;  
l7.Top :=l7.Top+y-starty;  
l7.Left:=l7.Left+x-startx;  
end;  
end;
```

```
procedure TForm1.Shape8MouseMove(Sender: TObject; Shift: TShiftState; X,  
  Y: Integer);  
begin  
if ssLeft in Shift then  
begin  
shape8.Left := shape8.Left + x-startx;  
shape8.Top := shape8.Top +y-starty;  
l8.Top :=l8.Top+y-starty;  
l8.Left:=l8.Left+x-startx;  
end;  
end;
```

```
procedure TForm1.Shape9MouseMove(Sender: TObject; Shift: TShiftState; X,  
  Y: Integer);  
begin  
if ssLeft in Shift then  
begin  
shape9.Left := shape9.Left + x-startx;
```

```
shape9.Top := shape9.Top +y-starty;
```

```
l9.Top :=l9.Top+y-starty;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
l9.Left:=l9.Left+x-startx;  
end;  
end;
```

```
procedure TForm1.Shape10MouseMove(Sender: TObject; Shift: TShiftState; X,  
  Y: Integer);  
begin  
if ssLeft in Shift then  
begin  
shape10.Left := shape10.Left + x-startx;  
shape10.Top := shape10.Top + y-starty;  
l10.Top :=l10.Top+y-starty;  
l10.Left:=l10.Left+x-startx;  
end;  
end;
```

```
procedure TForm1.Shape11MouseMove(Sender: TObject; Shift: TShiftState; X,  
  Y: Integer);  
begin  
if ssLeft in Shift then  
begin  
shape11.Left := shape11.Left + x-startx;  
shape11.Top := shape11.Top + y-starty;  
l11.Top :=l11.Top+y-starty;  
l11.Left:=l11.Left+x-startx;  
end;  
end;
```

```
procedure TForm1.Shape12MouseMove(Sender: TObject; Shift: TShiftState; X,  
  Y: Integer);  
begin  
if ssLeft in Shift then  
begin  
shape12.Left := shape12.Left + x-startx;  
shape12.Top := shape12.Top + y-starty;
```

```
l12.Top :=l12.Top+y-starty;
```

```
l12.Left:=l12.Left+x-startx;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

end;

end;

```
procedure TForm1.Shape13MouseMove(Sender: TObject; Shift: TShiftState; X,
```

```
Y: Integer);
```

```
begin
```

```
if ssLeft in Shift then
```

```
begin
```

```
shape13.Left := shape13.Left + x-startx;
```

```
shape13.Top := shape13.Top + y-starty;
```

```
l13.Top :=l13.Top+y-starty;
```

```
l13.Left:=l13.Left+x-startx;
```

```
end;
```

```
end;
```

```
procedure TForm1.Shape16MouseMove(Sender: TObject; Shift: TShiftState; X,
```

```
Y: Integer);
```

```
begin
```

```
if ssLeft in Shift then
```

```
begin
```

```
shape16.Left := shape16.Left + x-startx;
```

```
shape16.Top := shape16.Top + y-starty;
```

```
l16.Top :=l16.Top+y-starty;
```

```
l16.Left:=l16.Left+x-startx;
```

```
end;
```

```
end;
```

```
procedure TForm1.Shape15MouseMove(Sender: TObject; Shift: TShiftState; X,
```

```
Y: Integer);
```

```
begin
```

```
if ssLeft in Shift then
```

```
begin
```

```
shape15.Left := shape15.Left + x-startx;
```

```
shape15.Top := shape15.Top + y-starty;
```

```
l15.Top :=l15.Top+y-starty;
```

```
l15.Left:=l15.Left+x-startx;
```

```
end;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
end;
```

```
procedure TForm1.Shape14MouseMove(Sender: TObject; Shift: TShiftState; X,
```

```
Y: Integer);
```

```
begin
```

```
if ssLeft in Shift then
```

```
begin
```

```
shape14.Left := shape14.Left + x-startx;
```

```
shape14.Top := shape14.Top + y-starty;
```

```
l14.Top :=l14.Top+y-starty;
```

```
l14.Left:=l14.Left+x-startx;
```

```
end;
```

```
end;
```

```
procedure TForm1.ShapeMouseUp(Sender: TObject; Button: TMouseButton;
```

```
Shift: TShiftState; X, Y: Integer);
```

```
begin
```

```
edit1.Text := table1.FieldByName('2c').asstring;
```

```
edit2.Text := table1.FieldByName('2b').asstring;
```

```
edit3.Text := table1.FieldByName('N2').asstring;
```

```
c2:=inputbox('inputchanal','aa','0');
```

```
edit1.Text := c2;
```

```
b2:=inputbox('inputbit','aa','0');
```

```
edit2.Text := b2;
```

```
d[2]:=inputbox('InputName','Name','Station2');
```

```
edit3.text := d[2];
```

```
end;
```

```
procedure TForm1.Shape2MouseUp(Sender: TObject; Button: TMouseButton;
```

```
Shift: TShiftState; X, Y: Integer);
```

```
begin
```

```
c2:=inputbox('inputchanal','aa','0');
```

```
edit1.Text := c2;
```

```
b2:=inputbox('inputbit','aa','0');
```

```
edit2.Text := b2;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
procedure TForm1.ShapMouseUp(Sender: TObject; Button: TMouseButton;
```

```
Shift: TShiftState; X, Y: Integer);
```

```
begin
```

```
edit1.Text := table1.FieldByName('3c').asstring;
```

```
edit2.Text := table1.FieldByName('3b').asstring;
```

```
edit3.Text := table1.FieldByName('N3').asstring;
```

```
c3:=inputbox('inputchanal','aa','0');
```

```
edit1.Text := c3;
```

```
b3:=inputbox('inputbit','aa','0');
```

```
edit2.Text := b3;
```

```
d[3]:=inputbox('InputName','Name','Station3');
```

```
edit3.text := d[3];
```

```
end;
```

```
procedure TForm1.Shape4MouseUp(Sender: TObject; Button: TMouseButton;
```

```
Shift: TShiftState; X, Y: Integer);
```

```
begin
```

```
edit1.Text := table1.FieldByName('4c').asstring;
```

```
edit2.Text := table1.FieldByName('4b').asstring;
```

```
edit3.Text := table1.FieldByName('N4').asstring;
```

```
c4:=inputbox('inputchanal','aa','0');
```

```
edit1.Text := c4;
```

```
b4:=inputbox('inputbit','aa','0');
```

```
edit2.Text := b4;
```

```
d[4]:=inputbox('InputName','Name','Station4');
```

```
edit3.text := d[4];
```

```
end;
```

```
procedure TForm1.Shape5MouseUp(Sender: TObject; Button: TMouseButton;
```

```
Shift: TShiftState; X, Y: Integer);
```

```
begin
```

```
edit1.Text := table1.FieldByName('5c').asstring;
```

```
edit2.Text := table1.FieldByName('5b').asstring;
```

```
edit3.Text := table1.FieldByName('N5').asstring;
```

```
c5:=inputbox('inputchanal','aa','0');
```

```
edit1.Text := c5;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

b5:=inputbox('inputbit','aa','0');
edit2.Text := b5;
d[5]:=inputbox('InputName','Name','Station5');
edit3.text := d[5];
end;

```

```

procedure TForm1.Shape6MouseUp(Sender: TObject; Button: TMouseButton;
Shift: TShiftState; X, Y: Integer);
begin
edit1.Text := table1.FieldByName('6c').asstring;
edit2.Text := table1.FieldByName('6b').asstring;
edit3.Text := table1.FieldByName('N6').asstring;
c6:=inputbox('inputchanal','aa','0');
edit1.Text := c6;
b6:=inputbox('inputbit','aa','0');
edit2.Text := b6;
d[6]:=inputbox('InputName','Name','Station6');
edit3.text := d[6];
end;

```

```

procedure TForm1.Shape7MouseUp(Sender: TObject; Button: TMouseButton;
Shift: TShiftState; X, Y: Integer);
begin
edit1.Text := table1.FieldByName('7c').asstring;
edit2.Text := table1.FieldByName('7b').asstring;
edit3.Text := table1.FieldByName('N7').asstring;
c7:=inputbox('inputchanal','aa','0');
edit1.Text := c7;
b7:=inputbox('inputbit','aa','0');
edit2.Text := b7;
d[7]:=inputbox('InputName','Name','Station7');
edit3.text := d[7];
end;

```

```

procedure TForm1.Shape8MouseUp(Sender: TObject; Button: TMouseButton;

```

```

Shift: TShiftState; X, Y: Integer);

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
 ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

edit1.Text := table1.FieldName('8c').asstring;
edit2.Text := table1.FieldName('8b').asstring;
edit3.Text := table1.FieldName('N8').asstring;
c8:=inputbox('inputchanal','aa','0');
edit1.Text := c8;
b8:=inputbox('inputbit','aa','0');
edit2.Text := b8;
d[8]:=inputbox('InputName','Name','Station8');
edit3.text := d[8];
end;

```

```

procedure TForm1.Shape9MouseUp(Sender: TObject; Button: TMouseButton;
  Shift: TShiftState; X, Y: Integer);
begin
edit1.Text := table1.FieldName('9c').asstring;
edit2.Text := table1.FieldName('9b').asstring;
edit3.Text := table1.FieldName('N9').asstring;
c9:=inputbox('inputchanal','aa','0');
edit1.Text := c9;
b9:=inputbox('inputbit','aa','0');
edit2.Text := b9;
d[9]:=inputbox('InputName','Name','Station9');
edit3.text := d[9];
end;

```

```

procedure TForm1.Shape10MouseUp(Sender: TObject; Button: TMouseButton;
  Shift: TShiftState; X, Y: Integer);
begin
edit1.Text := table1.FieldName('10c').asstring;
edit2.Text := table1.FieldName('10b').asstring;
edit3.Text := table1.FieldName('N10').asstring;
c10:=inputbox('inputchanal','aa','0');
edit1.Text := c10;
b10:=inputbox('inputbit','aa','0');
edit2.Text := b10;
d[10]:=inputbox('InputName','Name','Station10');
edit3.text := d[10];
end;

```

เอกสารนี้เป็นเอกสารที่จัดทำขึ้นเพื่อการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
end;
```

```
procedure TForm1.Shape11MouseDown(Sender: TObject; Button: TMouseButton;  
  Shift: TShiftState; X, Y: Integer);  
begin  
  edit1.Text := table1.FieldByName('11c').asString;  
  edit2.Text := table1.FieldByName('11b').asString;  
  edit3.Text := table1.FieldByName('N11').asString;  
  c11:=inputbox('inputchanal','aa','0');  
  edit1.Text := c11;  
  b11:=inputbox('inputbit','aa','0');  
  edit2.Text := b11;  
  d[11]:=inputbox('InputName','Name','Station11');  
  edit3.text := d[11];  
end;
```

```
procedure TForm1.Shape12MouseDown(Sender: TObject; Button: TMouseButton;  
  Shift: TShiftState; X, Y: Integer);  
begin  
  edit1.Text := table1.FieldByName('12c').asString;  
  edit2.Text := table1.FieldByName('12b').asString;  
  edit3.Text := table1.FieldByName('N12').asString;  
  c12:=inputbox('inputchanal','aa','0');  
  edit1.Text := c12;  
  b12:=inputbox('inputbit','aa','0');  
  edit2.Text := b12;  
  d[12]:=inputbox('InputName','Name','Station12');  
  edit3.text := d[12];  
end;
```

```
procedure TForm1.Shape13MouseDown(Sender: TObject; Button: TMouseButton;  
  Shift: TShiftState; X, Y: Integer);  
begin  
  edit1.Text := table1.FieldByName('13c').asString;  
  edit2.Text := table1.FieldByName('13b').asString;  
  edit3.Text := table1.FieldByName('N13').asString;  
  c13:=inputbox('inputchanal','aa','0');
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับงานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

edit1.Text := c13;
b13:=inputbox('inputbit','aa','0');
edit2.Text := b13;
d[13]:=inputbox('InputName','Name','Station13');
edit3.text := d[13];
end;

```

```

procedure TForm1.Shape16MouseUp(Sender: TObject; Button: TMouseButton;
  Shift: TShiftState; X, Y: Integer);
begin
edit1.Text := table1.FieldByName('16c').asString;
edit2.Text := table1.FieldByName('16b').asString;
edit3.Text := table1.FieldByName('N16').asString;
c16:=inputbox('inputchanal','aa','0');
edit1.Text := c16;
b16:=inputbox('inputbit','aa','0');
edit2.Text := b16;
d[16]:=inputbox('InputName','Name','Station14');
edit3.text := d[16];
end;

```

```

procedure TForm1.Shape15MouseUp(Sender: TObject; Button: TMouseButton;
  Shift: TShiftState; X, Y: Integer);
begin
edit1.Text := table1.FieldByName('15c').asString;
edit2.Text := table1.FieldByName('15b').asString;
edit3.Text := table1.FieldByName('N15').asString;
c15:=inputbox('inputchanal','aa','0');
edit1.Text := c15;
b15:=inputbox('inputbit','aa','0');
edit2.Text := b15;
d[15]:=inputbox('InputName','Name','Station15');
edit3.text := d[15];
end;

```

```

procedure TForm1.Shape14MouseUp(Sender: TObject; Button: TMouseButton;
  Shift: TShiftState; X, Y: Integer);

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

begin
edit1.Text := table1.FieldName('14c').asString;
edit2.Text := table1.FieldName('14b').asString;
edit3.Text := table1.FieldName('N14').asString;
c14:=inputbox('inputchanal','aa','0');
edit1.Text := c14;
b14:=inputbox('inputbit','aa','0');
edit2.Text := b14;
d[14]:=inputbox('InputName','Name','Station 16');
edit3.text := d[14];
end;

```

```

procedure TForm1.BitBtn2Click(Sender: TObject);
var i:integer;
begin
table1.First;
table1.edit;
table1.FieldName('1t').asinteger := shape1.Top;
table1.FieldName('1l').asinteger := shape1.Left;
table1.FieldName('2t').asinteger := shape2.Top;
table1.FieldName('2l').asinteger := shape2.Left;
table1.FieldName('3t').asinteger := shape3.Top;
table1.FieldName('3l').asinteger := shape3.Left;
table1.FieldName('4t').asinteger := shape4.Top;
table1.FieldName('4l').asinteger := shape4.Left;
table1.FieldName('5t').asinteger := shape5.Top;
table1.FieldName('5l').asinteger := shape5.Left;
table1.FieldName('6t').asinteger := shape6.Top;
table1.FieldName('6l').asinteger := shape6.Left;
table1.FieldName('7t').asinteger := shape7.Top;
table1.FieldName('7l').asinteger := shape7.Left;
table1.FieldName('8t').asinteger := shape8.Top;
table1.FieldName('8l').asinteger := shape8.Left;
table1.FieldName('9t').asinteger := shape9.Top;
table1.FieldName('9l').asinteger := shape9.Left;
table1.FieldName('10t').asinteger := shape10.Top;
table1.FieldName('10l').asinteger := shape10.Left;

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเฉพาะเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

table1.FieldByName('11t').asinteger := shape11.Top;  
table1.FieldByName('11l').asinteger := shape11.Left;  
table1.FieldByName('12t').asinteger := shape12.Top;  
table1.FieldByName('12l').asinteger := shape12.Left;  
table1.FieldByName('13t').asinteger := shape13.Top;  
table1.FieldByName('13l').asinteger := shape13.Left;  
table1.FieldByName('14t').asinteger := shape14.Top;  
table1.FieldByName('14l').asinteger := shape14.Left;  
table1.FieldByName('15t').asinteger := shape15.Top;  
table1.FieldByName('15l').asinteger := shape15.Left;  
table1.FieldByName('16t').asinteger := shape16.Top;  
table1.FieldByName('16l').asinteger := shape16.Left;  
table1.FieldByName('1b').asString := b1;  
table1.FieldByName('1c').asString := c1;  
table1.FieldByName('2b').asString := b2;  
table1.FieldByName('2c').asString := c2;  
table1.FieldByName('3b').asString := b3;  
table1.FieldByName('3c').asString := c3;  
table1.FieldByName('4b').asString := b4;  
table1.FieldByName('4c').asString := c4;  
table1.FieldByName('5b').asString := b5;  
table1.FieldByName('5c').asString := c5;  
table1.FieldByName('6b').asString := b6;  
table1.FieldByName('6c').asString := c6;  
table1.FieldByName('7b').asString := b7;  
table1.FieldByName('7c').asString := c7;  
table1.FieldByName('8b').asString := b8;  
table1.FieldByName('8c').asString := c8;  
table1.FieldByName('9b').asString := b9;  
table1.FieldByName('9c').asString := c9;  
table1.FieldByName('10b').asString := b10;  
table1.FieldByName('10c').asString := c10;  
table1.FieldByName('11b').asString := b11;  
table1.FieldByName('11c').asString := c11;  
table1.FieldByName('12b').asString := b12;  
table1.FieldByName('12c').asString := c12;  
table1.FieldByName('13b').asString := b13;

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

table1.FieldByName('13c').asString := c13;  
table1.FieldByName('14b').asString := b14;  
table1.FieldByName('14c').asString := c14;  
table1.FieldByName('15b').asString := b15;  
table1.FieldByName('15c').asString := c15;  
table1.FieldByName('16b').asString := b16;  
table1.FieldByName('16c').asString := c16;  
table1.FieldByName('100b').asString := b17;  
table1.FieldByName('100c').asString := c17;  
table1.FieldByName('101b').asString := b18;  
table1.FieldByName('101c').asString := c18;  
table1.FieldByName('102b').asString := b19;  
table1.FieldByName('102c').asString := c19;  
table1.FieldByName('103b').asString := b20;  
table1.FieldByName('103c').asString := c20;  
table1.FieldByName('104b').asString := b21;  
table1.FieldByName('104c').asString := c21;  
table1.FieldByName('105b').asString := b22;  
table1.FieldByName('105c').asString := c22;  
table1.FieldByName('106b').asString := b23;  
table1.FieldByName('106c').asString := c23;  
table1.FieldByName('107b').asString := b24;  
table1.FieldByName('107c').asString := c24;  
table1.FieldByName('108b').asString := b25;  
table1.FieldByName('108c').asString := c25;  
table1.FieldByName('109b').asString := b26;  
table1.FieldByName('109c').asString := c26;  
table1.FieldByName('110b').asString := b27;  
table1.FieldByName('110c').asString := c27;  
table1.FieldByName('111b').asString := b28;  
table1.FieldByName('111c').asString := c28;  
table1.FieldByName('112b').asString := b29;  
table1.FieldByName('112c').asString := c29;  
table1.FieldByName('113b').asString := b30;  
table1.FieldByName('113c').asString := c30;  
table1.FieldByName('114b').asString := b31;  
table1.FieldByName('114c').asString := c31;

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

table1.FieldName('115b').asString := b32;
table1.FieldName('115c').asString := c32;
for i:=1 to 16 do
begin
table1.FieldName('N'+inttostr(i)).asString := d[i];
end;
if openpictorialog1.FileName = " then
begin
if table1.FieldName('pic').asString = " then
begin
table1.Post;
openpictorialog1.Execute;
image1.Picture.LoadFromFile(openpictorialog1.filename);
table1.First;
table1.edit;
table1.FieldName('pic').asString:=openpictorialog1.FileName;
end;
end
else
begin
table1.FieldName('pic').asString:=openpictorialog1.FileName;
end;
{able1.post;
table1.next;
table1.edit;
table1.FieldName('1t').asinteger := strtoint(b1);
table1.FieldName('1l').asinteger := strtoint(c1);
table1.FieldName('2t').asinteger := strtoint(b2);
table1.FieldName('2l').asinteger := strtoint(c2);
table1.FieldName('3t').asinteger := strtoint(b3);
table1.FieldName('3l').asinteger := strtoint(c3);
table1.FieldName('4t').asinteger := strtoint(b4);
table1.FieldName('4l').asinteger := strtoint(c4);
table1.FieldName('5t').asinteger := strtoint(b5);
table1.FieldName('5l').asinteger := strtoint(c5);
table1.FieldName('6t').asinteger := strtoint(b6);
table1.FieldName('6l').asinteger := strtoint(c6);

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

table1.FieldByName('7t').asinteger := strtoint(b7);
table1.FieldByName('7l').asinteger := strtoint(c7);
table1.FieldByName('8t').asinteger := strtoint(b8);
table1.FieldByName('8l').asinteger := strtoint(c8);
table1.FieldByName('9t').asinteger := strtoint(b9);
table1.FieldByName('9l').asinteger := strtoint(c9);
table1.FieldByName('10t').asinteger := strtoint(b10);
table1.FieldByName('10l').asinteger := strtoint(c10);
table1.FieldByName('11t').asinteger := strtoint(b11);
table1.FieldByName('11l').asinteger := strtoint(c11);
table1.FieldByName('12t').asinteger := strtoint(b12);
table1.FieldByName('12l').asinteger := strtoint(c12);
table1.FieldByName('13t').asinteger := strtoint(b13);
table1.FieldByName('13l').asinteger := strtoint(c13);
table1.FieldByName('14t').asinteger := strtoint(b14);
table1.FieldByName('14l').asinteger := strtoint(c14);
table1.FieldByName('15t').asinteger := strtoint(b15);
table1.FieldByName('15l').asinteger := strtoint(c15);
table1.FieldByName('16t').asinteger := strtoint(b16);
table1.FieldByName('16l').asinteger := strtoint(c16);
}
table1.Post;
table1.First;
end;

procedure TForm1.BitBtn3Click(Sender: TObject);
begin
openpictorialog1.Execute;
image1.Picture.LoadFromFile(openpictorialog1.FileName);
end;

procedure TForm1.FormCreate(Sender: TObject);
var i:integer;
begin
b1 := '0';
b2 := '0';
b3 := '0';

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

b4 := '0';  
b5 := '0';  
b6 := '0';  
b7 := '0';  
b8 := '0';  
b9 := '0';  
b10 := '0';  
b11 := '0';  
b12 := '0';  
b13 := '0';  
b14 := '0';  
b15 := '0';  
b16 := '0';  
b17 := '0';  
b18 := '0';  
b19 := '0';  
b20 := '0';  
b21 := '0';  
b22 := '0';  
b23 := '0';  
b24 := '0';  
b25 := '0';  
b26 := '0';  
b27 := '0';  
b28 := '0';  
b29 := '0';  
b30 := '0';  
b31 := '0';  
b32 := '0';  
c1 := '0';  
c2 := '0';  
c3 := '0';  
c4 := '0';  
c5 := '0';  
c6 := '0';  
c7 := '0';  
c8 := '0';



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

c9 := '0';
c10 := '0';
c11 := '0';
c12 := '0';
c13 := '0';
c14 := '0';
c15 := '0';
c16 := '0';
c17 := '0';
c18 := '0';
c19 := '0';
c20 := '0';
c21 := '0';
c22 := '0';
c23 := '0';
c24 := '0';
c25 := '0';
c26 := '0';
c27 := '0';
c28 := '0';
c29 := '0';
c30 := '0';
c31 := '0';
c32 := '0';
for i:=1 to 16 do
d[i]:= "";
end;

```

```

procedure TForm1.BitBtn1Click(Sender: TObject);
var
j:integer;
i:string;
begin
openpictorialog1.Execute;
image1.Picture.LoadFromFile(openpictorialog1.FileName);

```

```

table1.First;
table1.Edit;

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



```

for j:= 1 to 16 DO
begin
i:=inttostr(j);
table1.FieldByName(i+'l').asinteger:=0;
table1.FieldByName(i+'t').asinteger:=0;
end;
table1.FieldByName('pic').asString:=openpicturedialog1.FileName;
for j:= 1 to 16 DO
begin
i:=inttostr(j);
table1.FieldByName(i+'b').asinteger := strtoint('0');
table1.FieldByName(i+'c').asinteger := strtoint('0');
table1.FieldByName(inttostr(j+99)+'b').asinteger := strtoint('0');
table1.FieldByName(inttostr(j+99)+'c').asinteger := strtoint('0');
end;
{table1.FieldByName('1b').asinteger := strtoint('0');
table1.FieldByName('1c').asinteger := strtoint('0');
table1.FieldByName('2b').asinteger := strtoint('0');
table1.FieldByName('2c').asinteger := strtoint('0');
table1.FieldByName('3b').asinteger := strtoint('0');
table1.FieldByName('3c').asinteger := strtoint('0');
table1.FieldByName('4b').asinteger := strtoint('0');
table1.FieldByName('4c').asinteger := strtoint('0');
table1.FieldByName('5b').asinteger := strtoint('0');
table1.FieldByName('5c').asinteger := strtoint('0');
table1.FieldByName('6b').asinteger := strtoint('0');
table1.FieldByName('6c').asinteger := strtoint('0');
table1.FieldByName('7b').asinteger := strtoint('0');
table1.FieldByName('7c').asinteger := strtoint('0');
table1.FieldByName('8b').asinteger := strtoint('0');
table1.FieldByName('8c').asinteger := strtoint('0');
table1.FieldByName('9b').asinteger := strtoint('0');
table1.FieldByName('9c').asinteger := strtoint('0');
table1.FieldByName('10b').asinteger := strtoint('0');
table1.FieldByName('10c').asinteger := strtoint('0');
table1.FieldByName('11b').asinteger := strtoint('0');
table1.FieldByName('11c').asinteger := strtoint('0');
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับกรใช้ภายในมหาวิทยาลัยเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

table1.FieldByName('12b').asinteger := strtoint('0');
table1.FieldByName('12c').asinteger := strtoint('0');
table1.FieldByName('13b').asinteger := strtoint('0');
table1.FieldByName('13c').asinteger := strtoint('0');
table1.FieldByName('14b').asinteger := strtoint('0');
table1.FieldByName('14c').asinteger := strtoint('0');
table1.FieldByName('15b').asinteger := strtoint('0');
table1.FieldByName('15c').asinteger := strtoint('0');
table1.FieldByName('16b').asinteger := strtoint('0');
table1.FieldByName('16c').asinteger := strtoint('0');}

```

```
table1.Post;
```

```
table1.First;
```

```
shape1.Top:=0;
```

```
shape2.Top:=24;
```

```
shape3.Top:=48;
```

```
shape4.Top:=72;
```

```
shape5.Top:=96;
```

```
shape6.Top:=120;
```

```
shape7.Top:=144;
```

```
shape8.Top:=168;
```

```
shape9.Top:=0;
```

```
shape10.Top:=24;
```

```
shape11.Top:=48;
```

```
shape12.Top:=72;
```

```
shape13.Top:=96;
```

```
shape14.Top:=120;
```

```
shape15.Top:=144;
```

```
shape16.Top:=168;
```

```
11.Top:=5;
```

```
12.Top:=30;
```

```
13.Top:=55;
```

```
14.Top:=80;
```

```
15.Top:=103;
```

```
16.Top:=125;
```

```
17.Top:=150;
```

```
18.Top:=175;
```

```
19.Top:=5;
```



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

110.Top:=30;  
111.Top:=55;  
112.Top:=80;  
113.Top:=103;  
114.Top:=125;  
115.Top:=150;  
116.Top:=175;  
shape1.Left:=424;  
shape2.Left:=424;  
shape3.Left:=424;  
shape4.Left:=424;  
shape5.Left:=424;  
shape6.Left:=424;  
shape7.Left:=424;  
shape8.Left:=424;  
shape9.Left:=448;  
shape10.Left:=448;  
shape11.Left:=448;  
shape12.Left:=448;  
shape13.Left:=448;  
shape14.Left:=448;  
shape15.Left:=448;  
shape16.Left:=448;  
11.Left:=432;  
12.Left:=432;  
13.Left:=432;  
14.Left:=432;  
15.Left:=432;  
16.Left:=432;  
17.Left:=432;  
18.Left:=432;  
19.Left:=455;  
110.Left:=455;  
111.Left:=455;  
112.Left:=455;  
113.Left:=455;  
114.Left:=455;



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

l15.Left:=455;

l16.Left:=455;

b1 := '0';

b2 := '0';

b3 := '0';

b4 := '0';

b5 := '0';

b6 := '0';

b7 := '0';

b8 := '0';

b9 := '0';

b10 := '0';

b11 := '0';

b12 := '0';

b13 := '0';

b14 := '0';

b15 := '0';

b16 := '0';

b17 := '0';

b18 := '0';

b19 := '0';

b20 := '0';

b21 := '0';

b22 := '0';

b23 := '0';

b24 := '0';

b25 := '0';

b26 := '0';

b27 := '0';

b28 := '0';

b29 := '0';

b30 := '0';

b31 := '0';

b32 := '0';

c1 := '0';

c2 := '0';

c3 := '0';



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

c4 := '0';
c5 := '0';
c6 := '0';
c7 := '0';
c8 := '0';
c9 := '0';
c10 := '0';
c11 := '0';
c12 := '0';
c13 := '0';
c14 := '0';
c15 := '0';
c16 := '0';
c17 := '0';
c18 := '0';
c19 := '0';
c20 := '0';
c21 := '0';
c22 := '0';
c23 := '0';
c24 := '0';
c25 := '0';
c26 := '0';
c27 := '0';
c28 := '0';
c29 := '0';
c30 := '0';
c31 := '0';
c32 := '0';
for j:=1 to 16 do
d[j] := "";
end;

```

```

procedure TForm1.Exit1Click(Sender: TObject);

```

```

begin

```

```

self.Close;

```

```

end;

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



```

procedure TForm1.open2Click(Sender: TObject);
var i:integer;
begin
table1.active := false;
opendialog1.Execute;
table1.TableName:=opendialog1.FileName;
table1.Active := true;
table1.First;
if table1.FieldByName('pic').asString <> " then
image1.Picture.LoadFromFile(table1.FieldByName('pic').asString);
shape1.Top := table1.FieldByName('1t').asinteger ;
shape1.Left := table1.FieldByName('1l').asinteger;
11.top:=shape1.Top+6;
11.Left:=shape1.Left+9;
shape2.Top := table1.FieldByName('2t').asinteger ;
shape2.Left := table1.FieldByName('2l').asinteger;
12.top:=shape2.Top+6;
12.Left:=shape2.Left+9;
shape3.Top := table1.FieldByName('3t').asinteger ;
shape3.Left := table1.FieldByName('3l').asinteger;
13.top:=shape3.Top+6;
13.Left:=shape3.Left+9;
shape4.Top := table1.FieldByName('4t').asinteger ;
shape4.Left := table1.FieldByName('4l').asinteger;
14.top:=shape4.Top+6;
14.Left:=shape4.Left+9;
shape5.Top := table1.FieldByName('5t').asinteger ;
shape5.Left := table1.FieldByName('5l').asinteger;
15.top:=shape5.Top+6;
15.Left:=shape5.Left+9;
shape6.Top := table1.FieldByName('6t').asinteger ;
shape6.Left := table1.FieldByName('6l').asinteger;
16.top:=shape6.Top+6;
16.Left:=shape6.Left+9;
shape7.Top := table1.FieldByName('7t').asinteger ;
shape7.Left := table1.FieldByName('7l').asinteger;

```

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์ของมหาวิทยาลัยเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

17.top:=shape7.Top+6;
17.Left:=shape7.Left+9;
shape8.Top := table1.FieldByName('8t').asinteger ;
shape8.Left := table1.FieldByName('8l').asinteger;
18.top:=shape8.Top+6;
18.Left:=shape8.Left+9;
shape9.Top := table1.FieldByName('9t').asinteger ;
shape9.Left := table1.FieldByName('9l').asinteger;
19.top:=shape9.Top+6;
19.Left:=shape9.Left+9;
shape10.Top := table1.FieldByName('10t').asinteger;
shape10.Left := table1.FieldByName('10l').asinteger;
110.top:=shape10.Top+6;
110.Left:=shape10.Left+7;
shape11.Top := table1.FieldByName('11t').asinteger;
shape11.Left := table1.FieldByName('11l').asinteger;
111.top:=shape11.Top+6;
111.Left:=shape11.Left+7;
shape12.Top := table1.FieldByName('12t').asinteger ;
shape12.Left := table1.FieldByName('12l').asinteger;
112.top:=shape12.Top+6;
112.Left:=shape12.Left+7;
shape13.Top := table1.FieldByName('13t').asinteger ;
shape13.Left := table1.FieldByName('13l').asinteger;
113.top:=shape13.Top+6;
113.Left:=shape13.Left+7;
shape14.Top := table1.FieldByName('14t').asinteger ;
shape14.Left := table1.FieldByName('14l').asinteger;
114.top:=shape14.Top+6;
114.Left:=shape14.Left+7;
shape15.Top := table1.FieldByName('15t').asinteger ;
shape15.Left := table1.FieldByName('15l').asinteger;
115.top:=shape15.Top+6;
115.Left:=shape15.Left+7;
shape16.Top := table1.FieldByName('16t').asinteger ;
shape16.Left := table1.FieldByName('16l').asinteger;
116.top:=shape16.Top+6;

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

l16.Left:=shape16.Left+7;  
b1:=table1.FieldByName('1b').asstring;  
b2:=table1.FieldByName('2b').asstring;  
b3:=table1.FieldByName('3b').asstring;  
b4:=table1.FieldByName('4b').asstring;  
b5:=table1.FieldByName('5b').asstring;  
b6:=table1.FieldByName('6b').asstring;  
b7:=table1.FieldByName('7b').asstring;  
b8:=table1.FieldByName('8b').asstring;  
b9:=table1.FieldByName('9b').asstring;  
b10:=table1.FieldByName('10b').asstring;  
b11:=table1.FieldByName('11b').asstring;  
b12:=table1.FieldByName('12b').asstring;  
b13:=table1.FieldByName('13b').asstring;  
b14:=table1.FieldByName('14b').asstring;  
b15:=table1.FieldByName('15b').asstring;  
b16:=table1.FieldByName('16b').asstring;  
b17:=table1.FieldByName('100b').asstring;  
b18:=table1.FieldByName('101b').asstring;  
b19:=table1.FieldByName('102b').asstring;  
b20:=table1.FieldByName('103b').asstring;  
b21:=table1.FieldByName('104b').asstring;  
b22:=table1.FieldByName('105b').asstring;  
b23:=table1.FieldByName('106b').asstring;  
b24:=table1.FieldByName('107b').asstring;  
b25:=table1.FieldByName('108b').asstring;  
b26:=table1.FieldByName('109b').asstring;  
b27:=table1.FieldByName('110b').asstring;  
b28:=table1.FieldByName('111b').asstring;  
b29:=table1.FieldByName('112b').asstring;  
b30:=table1.FieldByName('113b').asstring;  
b31:=table1.FieldByName('114b').asstring;  
b32:=table1.FieldByName('115b').asstring;  
c1:=table1.FieldByName('1c').asstring;  
c2:=table1.FieldByName('2c').asstring;

c3:=table1.FieldByName('3c').asstring;

c4:=table1.FieldByName('4c').asstring;

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

c5:=table1.FieldByName('5c').asstring;
c6:=table1.FieldByName('6c').asstring;
c7:=table1.FieldByName('7c').asstring;
c8:=table1.FieldByName('8c').asstring;
c9:=table1.FieldByName('9c').asstring;
c10:=table1.FieldByName('10c').asstring;
c11:=table1.FieldByName('11c').asstring;
c12:=table1.FieldByName('12c').asstring;
c13:=table1.FieldByName('13c').asstring;
c14:=table1.FieldByName('14c').asstring;
c15:=table1.FieldByName('15c').asstring;
c16:=table1.FieldByName('16c').asstring;
c17:=table1.FieldByName('100c').asstring;
c18:=table1.FieldByName('101c').asstring;
c19:=table1.FieldByName('102c').asstring;
c20:=table1.FieldByName('103c').asstring;
c21:=table1.FieldByName('104c').asstring;
c22:=table1.FieldByName('105c').asstring;
c23:=table1.FieldByName('106c').asstring;
c24:=table1.FieldByName('107c').asstring;
c25:=table1.FieldByName('108c').asstring;
c26:=table1.FieldByName('109c').asstring;
c27:=table1.FieldByName('110c').asstring;
c28:=table1.FieldByName('111c').asstring;
c29:=table1.FieldByName('112c').asstring;
c30:=table1.FieldByName('113c').asstring;
c31:=table1.FieldByName('114c').asstring;
c32:=table1.FieldByName('115c').asstring;
for i:= 1 to 16 do
d[i]:= table1.FieldByName('N'+inttostr(i)).asstring;
shape1.Hint:='chanel '+c1+' bit '+b1;
shape2.Hint:='chanel '+c2+' bit '+b2;
shape3.Hint:='chanel '+c3+' bit '+b3;
shape4.Hint:='chanel '+c4+' bit '+b4;
shape5.Hint:='chanel '+c5+' bit '+b5;
shape6.Hint:='chanel '+c6+' bit '+b6;
shape7.Hint:='chanel '+c7+' bit '+b7;

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

shape8.Hint:='chanel '+c8+' bit '+b8;  
shape9.Hint:='chanel '+c9+' bit '+b9;  
shape10.Hint:='chanel '+c10+' bit '+b10;  
shape11.Hint:='chanel '+c11+' bit '+b11;  
shape12.Hint:='chanel '+c12+' bit '+b12;  
shape13.Hint:='chanel '+c13+' bit '+b13;  
shape14.Hint:='chanel '+c14+' bit '+b14;  
shape15.Hint:='chanel '+c15+' bit '+b15;  
shape16.Hint:='chanel '+c16+' bit '+b16;  
shape100.Hint:='chanel '+c17+' bit '+b17;  
shape101.Hint:='chanel '+c18+' bit '+b18;  
shape102.Hint:='chanel '+c19+' bit '+b19;  
shape103.Hint:='chanel '+c20+' bit '+b20;  
shape104.Hint:='chanel '+c21+' bit '+b21;  
shape105.Hint:='chanel '+c22+' bit '+b22;  
shape106.Hint:='chanel '+c23+' bit '+b23;  
shape107.Hint:='chanel '+c24+' bit '+b24;  
shape108.Hint:='chanel '+c25+' bit '+b25;  
shape109.Hint:='chanel '+c26+' bit '+b26;  
shape110.Hint:='chanel '+c27+' bit '+b27;  
shape111.Hint:='chanel '+c28+' bit '+b28;  
shape112.Hint:='chanel '+c29+' bit '+b29;  
shape113.Hint:='chanel '+c30+' bit '+b30;  
shape114.Hint:='chanel '+c31+' bit '+b31;  
shape115.Hint:='chanel '+c32+' bit '+b32;  
11.Hint:='chanel '+c1+' bit '+b1;  
12.Hint:='chanel '+c2+' bit '+b2;  
13.Hint:='chanel '+c3+' bit '+b3;  
14.Hint:='chanel '+c4+' bit '+b4;  
15.Hint:='chanel '+c5+' bit '+b5;  
16.Hint:='chanel '+c6+' bit '+b6;  
17.Hint:='chanel '+c7+' bit '+b7;  
18.Hint:='chanel '+c8+' bit '+b8;  
19.Hint:='chanel '+c9+' bit '+b9;  
110.Hint:='chanel '+c10+' bit '+b10;

111.Hint:='chanel '+c11+' bit '+b11;

112.Hint:='chanel '+c12+' bit '+b12;

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

l13.Hint:='chanel '+c13+' bit '+b13;
l14.Hint:='chanel '+c14+' bit '+b14;
l15.Hint:='chanel '+c15+' bit '+b15;
l16.Hint:='chanel '+c16+' bit '+b16;
end;

```

```

procedure TForm1.Shape1MouseDown(Sender: TObject; Button: TMouseButton;

```

```

  Shift: TShiftState; X, Y: Integer);

```

```

begin

```

```

  edit1.Text := table1.FieldByName('1c').asstring;

```

```

  edit2.Text := table1.FieldByName('1b').asstring;

```

```

  edit3.Text := table1.FieldByName('N1').asstring;

```

```

  c1:=inputbox('inputchanel','aa','0');

```

```

  edit1.Text := c1;

```

```

  b1:=inputbox('inputbit','aa','0');

```

```

  edit2.Text := b1;

```

```

  d[1]:=inputbox('InputName','Name','Station1');

```

```

  edit3.text := d[1];

```

```

end;

```

```

procedure TForm1.Shape100MouseDown(Sender: TObject; Button: TMouseButton;

```

```

  Shift: TShiftState; X, Y: Integer);

```

```

begin

```

```

  edit1.Text := table1.FieldByName('100c').asstring;

```

```

  edit2.Text := table1.FieldByName('100b').asstring;

```

```

  c17:=inputbox('inputchanel','aa','0');

```

```

  edit1.Text := c17;

```

```

  b17:=inputbox('inputbit','aa','0');

```

```

  edit2.Text := b17;

```

```

end;

```

```

procedure TForm1.Shape101MouseDown(Sender: TObject; Button: TMouseButton;

```

```

  Shift: TShiftState; X, Y: Integer);

```

```

begin

```

```

  edit1.Text := table1.FieldByName('101c').asstring;

```

```

  edit2.Text := table1.FieldByName('101b').asstring;

```

```

  c18:=inputbox('inputchanel','aa','0');

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

edit1.Text := c18;
b18:=inputbox('inputbit','aa','0');
edit2.Text := b18;
end;

procedure TForm1.Shape102MouseUp(Sender: TObject; Button: TMouseButton;
  Shift: TShiftState; X, Y: Integer);
begin
edit1.Text := table1.FieldByName('102c').asstring;
edit2.Text := table1.FieldByName('102b').asstring;
c19:=inputbox('inputchanal','aa','0');
edit1.Text := c19;
b19:=inputbox('inputbit','aa','0');
edit2.Text := b19;
end;

procedure TForm1.Shape103MouseUp(Sender: TObject; Button: TMouseButton;
  Shift: TShiftState; X, Y: Integer);
begin
edit1.Text := table1.FieldByName('103c').asstring;
edit2.Text := table1.FieldByName('103b').asstring;
c20:=inputbox('inputchanal','aa','0');
edit1.Text := c20;
b20:=inputbox('inputbit','aa','0');
edit2.Text := b20;
end;

procedure TForm1.Shape104MouseUp(Sender: TObject; Button: TMouseButton;
  Shift: TShiftState; X, Y: Integer);
begin
edit1.Text := table1.FieldByName('104c').asstring;
edit2.Text := table1.FieldByName('104b').asstring;
c21:=inputbox('inputchanal','aa','0');
edit1.Text := c21;
b21:=inputbox('inputbit','aa','0');

```

```
edit2.Text := b21;
```

```
end;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

procedure TForm1.Shape105MouseUp(Sender: TObject; Button: TMouseButton;
  Shift: TShiftState; X, Y: Integer);
begin
edit1.Text := table1.FieldByName('105c').asString;
edit2.Text := table1.FieldByName('105b').asString;
c22:=inputbox('inputchanal','aa','0');
edit1.Text := c22;
b22:=inputbox('inputbit','aa','0');
edit2.Text := b22;
end;

```

```

procedure TForm1.Shape106MouseUp(Sender: TObject; Button: TMouseButton;
  Shift: TShiftState; X, Y: Integer);
begin
edit1.Text := table1.FieldByName('106c').asString;
edit2.Text := table1.FieldByName('106b').asString;
c23:=inputbox('inputchanal','aa','0');
edit1.Text := c23;
b23:=inputbox('inputbit','aa','0');
edit2.Text := b23;
end;

```

```

procedure TForm1.Shape107MouseUp(Sender: TObject; Button: TMouseButton;
  Shift: TShiftState; X, Y: Integer);
begin
edit1.Text := table1.FieldByName('107c').asString;
edit2.Text := table1.FieldByName('107b').asString;
c24:=inputbox('inputchanal','aa','0');
edit1.Text := c24;
b24:=inputbox('inputbit','aa','0');
edit2.Text := b24;
end;

```

```

procedure TForm1.Shape108MouseUp(Sender: TObject; Button: TMouseButton;

```

```

  Shift: TShiftState; X, Y: Integer);

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
 begin

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

edit1.Text := table1.FieldByName('108c').asString;
edit2.Text := table1.FieldByName('108b').asString;
c25:=inputbox('inputchanal','aa','0');
edit1.Text := c25;
b25:=inputbox('inputbit','aa','0');
edit2.Text := b25;
end;

```

```

procedure TForm1.Shape109MouseUp(Sender: TObject; Button: TMouseButton;
  Shift: TShiftState; X, Y: Integer);
begin
edit1.Text := table1.FieldByName('109c').asString;
edit2.Text := table1.FieldByName('109b').asString;
c26:=inputbox('inputchanal','aa','0');
edit1.Text := c26;
b26:=inputbox('inputbit','aa','0');
edit2.Text := b26;
end;

```

```

procedure TForm1.Shape110MouseUp(Sender: TObject; Button: TMouseButton;
  Shift: TShiftState; X, Y: Integer);
begin
edit1.Text := table1.FieldByName('110c').asString;
edit2.Text := table1.FieldByName('110b').asString;
c27:=inputbox('inputchanal','aa','0');
edit1.Text := c27;
b27:=inputbox('inputbit','aa','0');
edit2.Text := b27;
end;

```

```

procedure TForm1.Shape111MouseUp(Sender: TObject; Button: TMouseButton;
  Shift: TShiftState; X, Y: Integer);
begin
edit1.Text := table1.FieldByName('111c').asString;
edit2.Text := table1.FieldByName('111b').asString;

```

```

c28:=inputbox('inputchanal','aa','0');

```

```

edit1.Text := c28;

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
b28:=inputbox('inputbit','aa','0');
edit2.Text := b28;
end;
```

```
procedure TForm1.Shape112MouseUp(Sender: TObject; Button: TMouseButton;
  Shift: TShiftState; X, Y: Integer);
begin
edit1.Text := table1.FieldByName('112c').asstring;
edit2.Text := table1.FieldByName('112b').asstring;
c29:=inputbox('inputchanal','aa','0');
edit1.Text := c29;
b29:=inputbox('inputbit','aa','0');
edit2.Text := b29;
end;
```

```
procedure TForm1.Shape113MouseUp(Sender: TObject; Button: TMouseButton;
  Shift: TShiftState; X, Y: Integer);
begin
edit1.Text := table1.FieldByName('113c').asstring;
edit2.Text := table1.FieldByName('113b').asstring;
c30:=inputbox('inputchanal','aa','0');
edit1.Text := c30;
b30:=inputbox('inputbit','aa','0');
edit2.Text := b30;
end;
```

```
procedure TForm1.Shape114MouseUp(Sender: TObject; Button: TMouseButton;
  Shift: TShiftState; X, Y: Integer);
begin
edit1.Text := table1.FieldByName('114c').asstring;
edit2.Text := table1.FieldByName('114b').asstring;
c31:=inputbox('inputchanal','aa','0');
edit1.Text := c31;
b31:=inputbox('inputbit','aa','0');
edit2.Text := b31;
end;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

procedure TForm1.Shape115MouseDown(Sender: TObject; Button: TMouseButton;
  Shift: TShiftState; X, Y: Integer);
begin
edit1.Text := table1.FieldByName('115c').asstring;
edit2.Text := table1.FieldByName('115b').asstring;
c32:=inputbox('inputchanal','aa','0');
edit1.Text := c32;
b32:=inputbox('inputbit','aa','0');
edit2.Text := b32;
end;

```

```

procedure TForm1.Button1Click(Sender: TObject);
begin
memo1.Lines.Add('chanal');
memo1.Lines.Add(c16);
memo1.Lines.Add(c17);
memo1.Lines.Add(c18);
memo1.Lines.Add(c19);
memo1.Lines.Add(c20);
memo1.Lines.Add(c21);
memo1.Lines.Add(c22);
memo1.Lines.Add(c23);
memo1.Lines.Add(c24);
memo1.Lines.Add(c25);
memo1.Lines.Add(c26);
memo1.Lines.Add(c27);
memo1.Lines.Add(c28);
memo1.Lines.Add(c29);
memo1.Lines.Add(c30);
memo1.Lines.Add(c31);
memo1.Lines.Add(c32);
memo1.Lines.Add('bit');
memo1.Lines.Add(b16);
memo1.Lines.Add(b17);
memo1.Lines.Add(b18);

```

```
memo1.Lines.Add(b19);
```

```
memo1.Lines.Add(b20);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

memo1.Lines.Add(b21);  
memo1.Lines.Add(b22);  
memo1.Lines.Add(b23);  
memo1.Lines.Add(b24);  
memo1.Lines.Add(b25);  
memo1.Lines.Add(b26);  
memo1.Lines.Add(b27);  
memo1.Lines.Add(b28);  
memo1.Lines.Add(b29);  
memo1.Lines.Add(b30);  
memo1.Lines.Add(b31);  
memo1.Lines.Add(b32);  
end;  
  
end.



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## SECTION 6

### Host Link Commands

This section explains the methods and procedures for using host link commands, which can be used for host link communications via the CQM1 and CPM1 ports.

6-1	Communications Procedure .....	314
6-2	Command and Response Formats .....	316
6-2-1	Commands from the Host Computer .....	316
6-2-2	Commands from the PC (CQM1 Only) .....	319
6-2-3	Response End Codes .....	319
6-3	Host Link Commands .....	320
6-3-1	IR/SR AREA READ — RR .....	320
6-3-2	LR AREA READ — RL .....	320
6-3-3	HR AREA READ — RH .....	321
6-3-4	PV READ — RC .....	321
6-3-5	TC STATUS READ — RG .....	322
6-3-6	DM AREA READ — RD .....	322
6-3-7	AR AREA READ — RJ .....	323
6-3-8	IR/SR AREA WRITE — WR .....	323
6-3-9	LR AREA WRITE — WL .....	324
6-3-10	HR AREA WRITE — WH .....	324
6-3-11	PV WRITE — WC .....	325
6-3-12	TC STATUS WRITE — WG .....	325
6-3-13	DM AREA WRITE — WD .....	326
6-3-14	AR AREA WRITE — WJ .....	327
6-3-15	SV READ 1 — R# .....	327
6-3-16	SV READ 2 — RS .....	328
6-3-17	SV READ 3 — R% (CQM1 PCs Only) .....	329
6-3-18	SV CHANGE 1 — W# .....	330
6-3-19	SV CHANGE 2 — WS .....	331
6-3-20	SV CHANGE 3 — W% (CQM1 PCs Only) .....	332
6-3-21	STATUS READ — MS .....	333
6-3-22	STATUS WRITE — SC .....	334
6-3-23	ERROR READ — MF .....	334
6-3-24	FORCED SET — KS .....	335
6-3-25	FORCED RESET — KR .....	336
6-3-26	MULTIPLE FORCED SET/RESET — FK .....	337
6-3-27	FORCED SET/RESET CANCEL — KC .....	338
6-3-28	PC MODEL READ — MM .....	338
6-3-29	TEST — TS .....	339
6-3-30	PROGRAM READ — RP .....	339
6-3-31	PROGRAM WRITE — WP .....	340
6-3-32	COMPOUND COMMAND — QQ .....	340
6-3-33	ABORT — XZ .....	342
6-3-34	INITIALIZE — IC .....	342
6-3-35	Undefined Command — IC .....	342

เอกสารนี้เป็นเอกสารที่จัดทำขึ้นเพื่อการศึกษาค้นคว้าเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ใดๆ  
 ไม่ว่ากรณีใดๆ ทั้งสิ้น ยกเว้นที่เห็นเหตุเปลี่ยนแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## Communications Procedure

## Command Chart

The commands listed in the chart below can be used for host link communications with the CQM1 and CPM1. These commands are all sent from the host computer to the PC.

Header code	Name	PC mode			Applicable PCs	Page
		RUN	MON	PRG		
RR	IR/SR AREA READ	Valid	Valid	Valid	All	320
RL	LR AREA READ	Valid	Valid	Valid	All	320
RH	HR AREA READ	Valid	Valid	Valid	All	321
RC	PV READ	Valid	Valid	Valid	All	321
RG	TC STATUS READ	Valid	Valid	Valid	All	322
RD	DM AREA READ	Valid	Valid	Valid	All	322
RJ	AR AREA READ	Valid	Valid	Valid	All	323
WR	IR/SR AREA WRITE	Not valid	Valid	Valid	All	323
WL	LR AREA WRITE	Not valid	Valid	Valid	All	324
WH	HR AREA WRITE	Not valid	Valid	Valid	All	324
WC	PV WRITE	Not valid	Valid	Valid	All	325
WG	TC STATUS WRITE	Not valid	Valid	Valid	All	325
WD	DM AREA WRITE	Not valid	Valid	Valid	All	326
WJ	AR AREA WRITE	Not valid	Valid	Valid	All	327
R#	SV READ 1	Valid	Valid	Valid	All	327
R\$	SV READ 2	Valid	Valid	Valid	All	328
R%	SV READ 3	Valid	Valid	Valid	CQM1 PCs only	329
R%	SV READ 3	Valid	Valid	Valid	All	330
W#	SV CHANGE 1	Not valid	Valid	Valid	All	331
W\$	SV CHANGE 2	Not valid	Valid	Valid	All	331
W%	SV CHANGE 3	Not valid	Valid	Valid	CQM1 PCs only	332
MS	STATUS READ	Valid	Valid	Valid	All	333
SC	STATUS WRITE	Valid	Valid	Valid	All	334
MF	ERROR READ	Valid	Valid	Valid	All	334
KS	FORCED SET	Not valid	Valid	Valid	All	335
KR	FORCED RESET	Not valid	Valid	Valid	All	336
FK	MULTIPLE FORCED SET/RESET	Not valid	Valid	Valid	All	337
KC	FORCED SET/RESET CANCEL	Not valid	Valid	Valid	All	338
MM	PC MODEL READ	Valid	Valid	Valid	All	338
TS	TEST	Valid	Valid	Valid	All	339
RP	PROGRAM READ	Valid	Valid	Valid	All	339
WP	PROGRAM WRITE	Not valid	Not valid	Valid	All	340
QQ	COMPOUND COMMAND	Valid	Valid	Valid	All	340
XZ	ABORT (command only)	Valid	Valid	Valid	All	342
*:	INITIALIZE (command only)	Valid	Valid	Valid	All	342
IC	Undefined command (response only)	---	---	---	All	342

## 6-1 Communications Procedure

Host link communications are executed by means an exchange of commands and responses between the host computer and the PC.

เอกสารนี้เป็นเอกสารที่ส่งจากโฮสต์ไปยังพีซี (host to PC) ซึ่งเป็นการสื่อสารแบบปกติ (normal method) ซึ่งคำสั่งจะถูกส่งจากโฮสต์ไปยังพีซี (host to PC) และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้ (reference to the owner of the command every time it is used).

With the CQM1, there are two communications methods that can be used. One is the normal method, in which commands are issued from the host computer to the PC. The other method allows commands to be issued from the PC to the host computer.

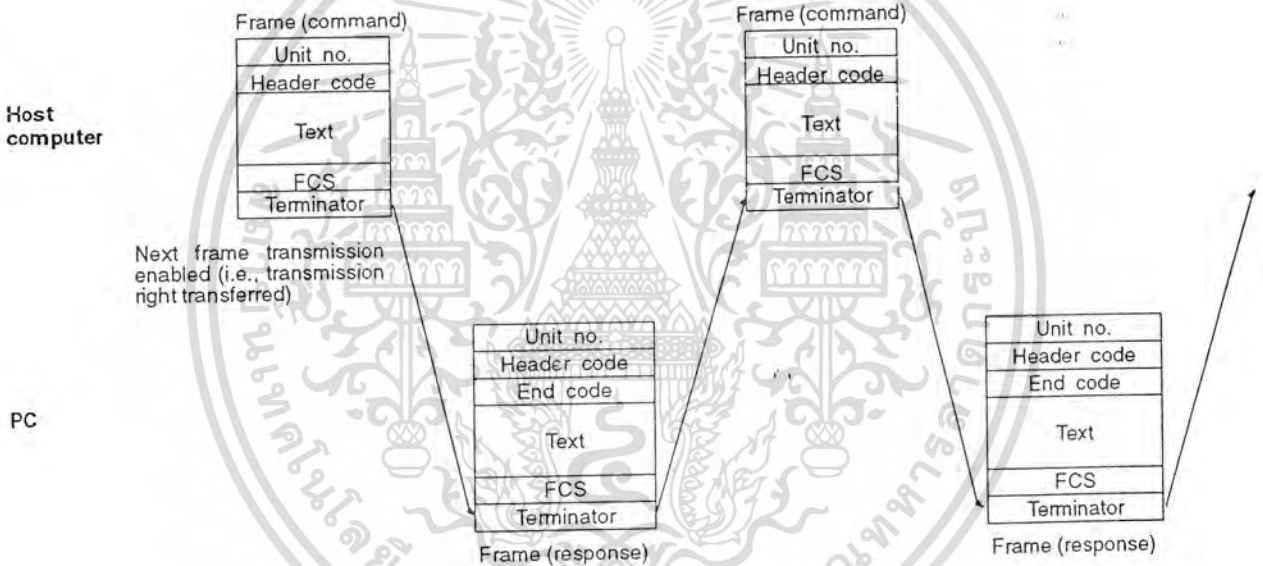
**Frame Transmission and Reception**

Commands and responses are exchanged in the order shown in the illustration below. The block of data transferred in a single transmission is called a "frame." A single frame is configured of a maximum of 131 characters of data.

The right to send a frame is called the "transmission right." The Unit that has the transmission right is the one that can send a frame at any given time. The transmission right is traded back and forth between the host computer and the PC each time a frame is transmitted. The transmission right is passed from the transmitting Unit to the receiving Unit when either a terminator (the code that marks the end of a command or response) or a delimiter (the code that sets frames apart) is received.

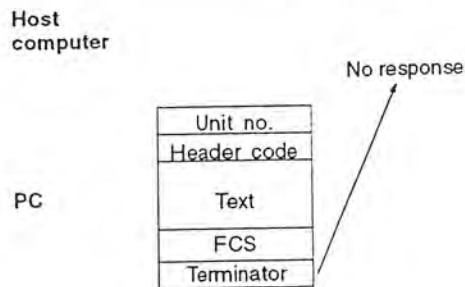
**Commands from Host**

In host link communications, the host computer ordinarily has the transmission right first and initiates the communications. The PC then automatically sends a response.



**Commands from PC (CQM1 PCs Only)**

With CQM1 PCs, it is also possible in host link communications for the PC to send commands to the host computer. In this case it is the PC that has the transmission right and initiates the communications.



When commands are issued to the host computer, the data is transmitted in one direction from the PC to the host computer. If a response to a command is required use a host link communications command to write the response from the host computer to the PC.

## 6-2 Command and Response Formats

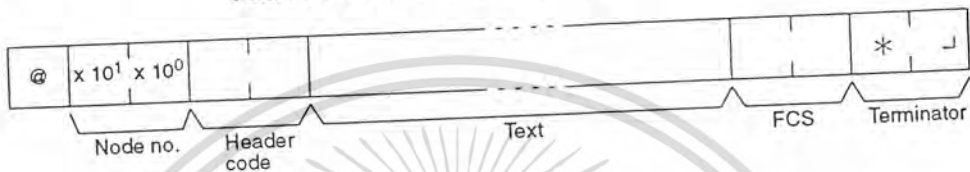
This section explains the formats for the commands and responses that are exchanged in host link communications.

### 6-2-1 Commands from the Host Computer

When a command is issued from the host computer, the command and response formats are as shown below.

#### Command Format

When transmitting a command from the host computer, prepare the command data in the format shown below.



@  
An "@" symbol must be placed at the beginning.

#### Node No.

Identifies the PC communicating with the host computer.

Specify the node number set for the PC in the PC Setup (DM 6648, DM 6653).

#### Header Code

Set the 2-character command code.

#### Text

Set the command parameters.

#### FCS

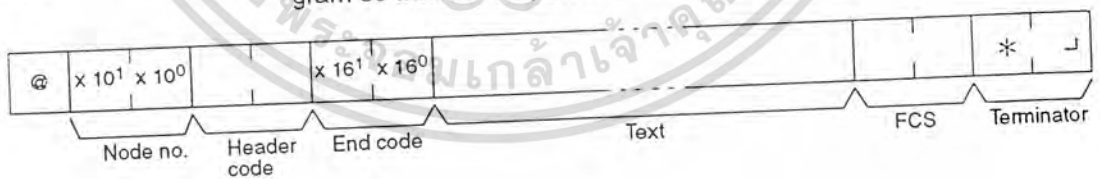
Set a 2-character Frame Check Sequence code. See page 318.

#### Terminator

Set two characters, "\*" and the carriage return (CHRS(13)) to indicate the end of the command.

#### Response Format

The response from the PC is returned in the format shown below. Prepare a program so that the response data can be interpreted and processed.



#### @, Node No., Header Code

Contents identical to those of the command are returned.

#### End Code

The completion status of the command (e.g., whether or not an error has occurred) is returned.

#### Text

Text is returned only when there is data such as read data.

#### FCS, Terminator

Refer to the corresponding explanations under "Command Format."

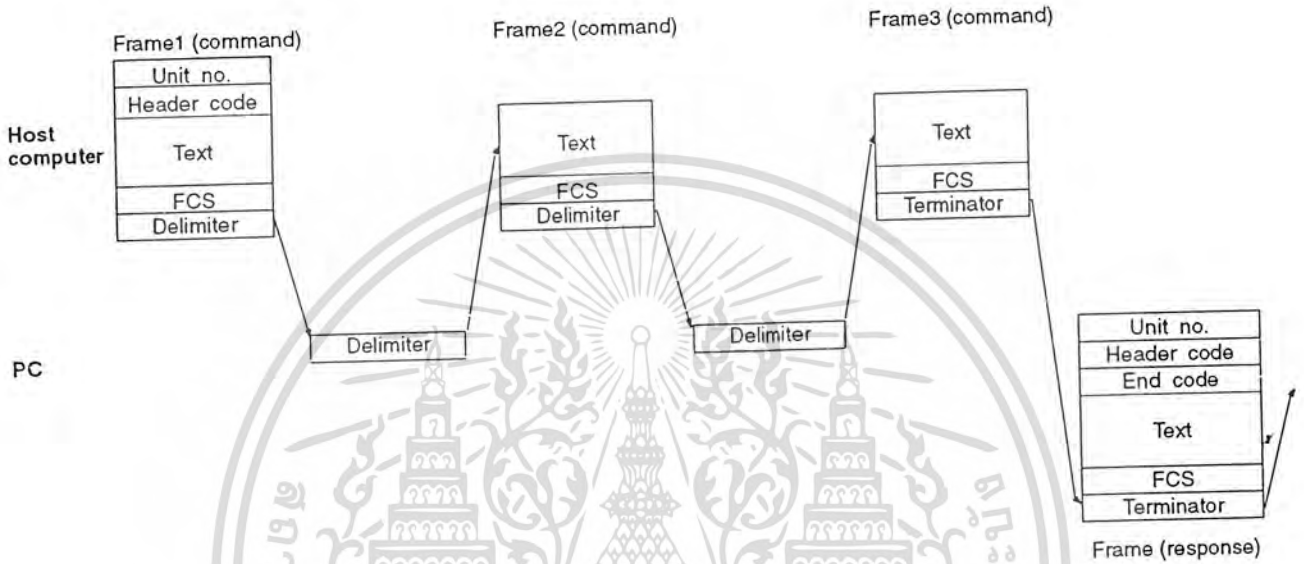
#### Long Transmissions

The largest block of data that can be transmitted as a single frame is 131 characters. A command or response of 132 characters or more must therefore be divided into more than one frame before transmission. When a transmission is split, the ends of the first and intermediate frames are marked by a delimiter instead of a terminator. อิงถึงเลขของเอกสารทุกครั้งที่มีการนำไปใช้

Command and Response Formats

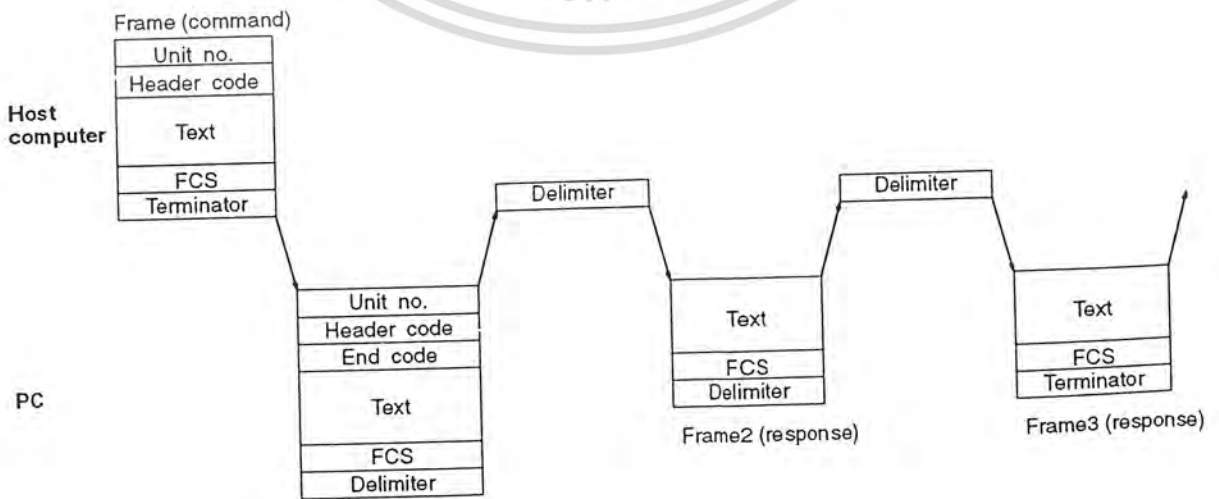
**Dividing Commands (Host Computer to PC)**

As each frame is transmitted by the host computer, the computer waits for the delimiter to be transmitted from the PC. After the delimiter has been transmitted, the next frame will then be sent. This procedure is repeated until the entire command has been transmitted.



**Dividing Responses (PC to Host Computer)**

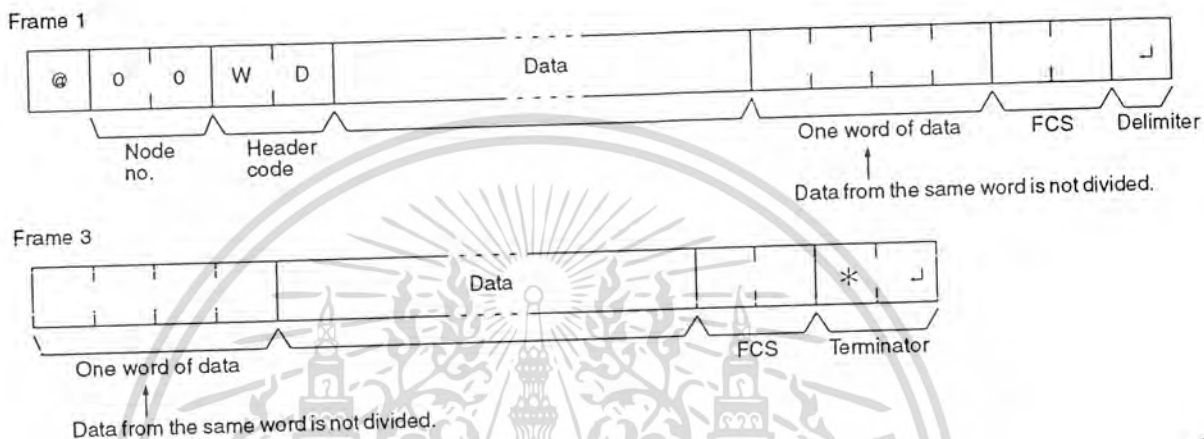
As each frame is received by the host computer, a delimiter is transmitted to the PC. After the delimiter has been transmitted, the PC will transmit the next frame. This procedure is repeated until the entire response has been transmitted.



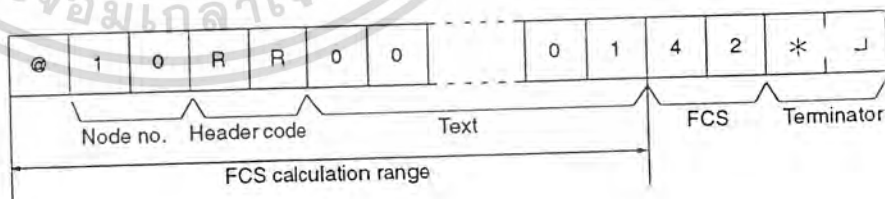
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับใช้เพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

**Precautions for Long Transmissions**

When dividing commands such as WR, WL, WC, or WD that execute write operations, be careful not to divide into separate frames data that is to be written into a single word. As shown in the illustration below, be sure to divide frames so that they coincide with the divisions between words.



**FCS (Frame Check Sequence)** When a frame is transmitted, an FCS is placed just before the delimiter or terminator in order to check whether any data error has been generated. The FCS is 8-bit data converted into two ASCII characters. The 8-bit data is the result of an EXCLUSIVE OR performed on the data from the beginning of the frame until the end of the text in that frame (i.e., just before the FCS). Calculating the FCS each time a frame is received and checking the result against the FCS that is included in the frame makes it possible to check for data errors in the frame.



Character	ASCII code	Binary	Binary
@	40	0100	0000
		XOR	
1	31	0011	0001
		XOR	
0	30	0011	0000
		XOR	
R	52	0101	0010
		...	
1	31	0011	0001
		0100	0010
Calculation result		4	2

Converted to hexadecimal. Handled as ASCII characters.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้ในการเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้เผยแพร่เป็นการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Example Program for FCS

This example shows a BASIC subroutine program for executing an FCS check on a frame received by the host computer.

```

400 *FCSCHECK
410 L=LEN (RESPONSE$) ' ..... Data transmitted and received
420 Q=0:FCCK$=" "
430 A$=RIGHT$ (RESPONSE$,1)
440 PRINT RESPONSE$,A$,L
450 IF A$="*" THEN LENG$=LEN (RESPONSE$)-3
      ELSE LENG$=LEN (RESPONSE$)-2
460 FCSP$=MID$ (RESPONSE$,LENG$+1,2) ' .... FCS data received
470 FOR I=1 TO LENG$ ' ..... Number of characters in FCS
480 Q=ASC (MID$ (RESPONSE$,I,1)) XOR Q
490 NEXT I
500 FCSD$=HEX$ (Q)
510 IF LEN (FCSD$)=1 THEN FCSD$="0"+FCSD$ ' ..... FCS result
520 IF FCSD$<>FCSP$ THEN FCCK$="ERR"
530 PRINT "FCSD$=";FCSD$,"FCSP$=";FCSP$,"FCCK$=";FCCK$
540 RETURN
    
```

- Note**
1. Normal reception data includes the FCS, delimiter or terminator, and so on. When an error occurs in transmission, however the FCS or some other data may not be included. Be sure to program the system to cover this possibility.
  2. In this program example, the CR code (CHR\$(13)) is not entered for RESPONSE\$. When including the CR code, make the changes in lines 430 and 450.

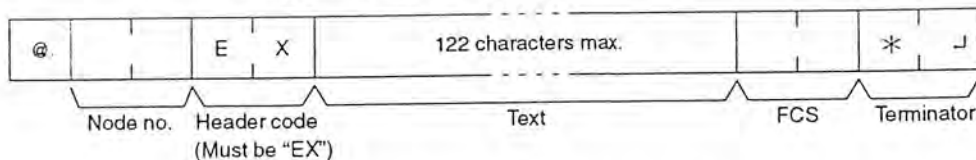
6-2-2 Commands from the PC (CQM1 Only)

In host link communications, commands are ordinarily sent from the host computer to the PC, but it is also possible for commands to be sent from the PC to the host computer. In Host Link Mode, any data can be transmitted from the PC to the host computer. To send a command to the host computer, use the TRANSMIT instruction (TXD(48)) in the PC program in Host Link Mode.

TXD(48) outputs data from the specified port (the RS-232C port or the peripheral port). Refer to page 308 for details on using TXD(48).

Reception Format

When TXD(48) is executed, the data stored in the words beginning with the first send word is converted to ASCII and output to the host computer as a host link command in the format shown below. The "@" symbol, node number, header code, FCS, and delimiter are all added automatically when the transmission is sent. At the host computer, it is necessary to prepare in advance a program for interpreting and processing this format.



One byte of data (2 digits hexadecimal) is converted to two characters in ASCII for transmission, the amount of data in the transmission is twice the amount of words specified for TXD(48). The maximum number of characters for transmission is 122 and the maximum number of bytes that can be designated for TXD(48) is one half of that, or 61.

6-2-3 Response End Codes

Refer to 8-7 Host Link Errors for a table listing the response end codes that may be returned in host link communications. An end code of 00 indicates normal completion of a command.

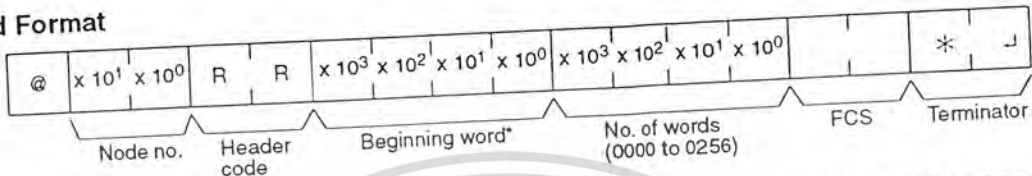
### 6-3 Host Link Commands

This section explains the commands that can be issued from the host computer to the PC.

#### 6-3-1 IR/SR AREA READ — RR

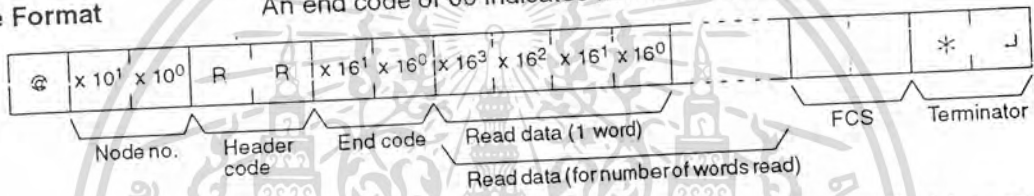
Reads the contents of the specified number of IR and SR words, starting from the specified word.

**Command Format**



**Note** Beginning word: 0000 to 0255 in CQM1 PCs, 0000 to 0019 and 0200 to 0255 in CPM1 PCs.

**Response Format**



An end code of 00 indicates normal completion.

**Note** 1. Words 0020 to 0199 in CPM1 PCs cannot be specified. If an attempt to read any of these words is made, a response of 0000 will be returned.  
2. The response will be divided when reading more than 30 words of data.

**Parameters**

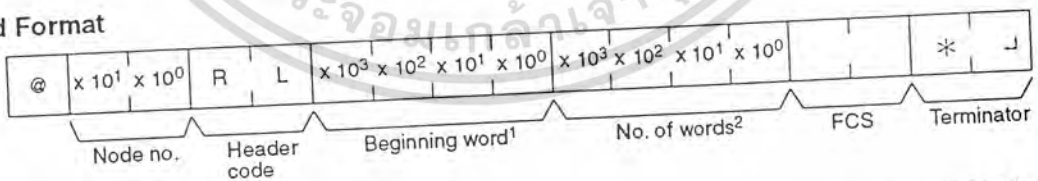
**Read Data (Response)**

The contents of the number of words specified by the command are returned in hexadecimal as a response. The words are returned in order, starting with the specified beginning word.

#### 6-3-2 LR AREA READ — RL

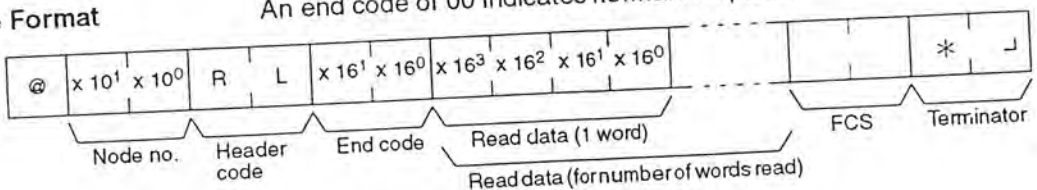
Reads the contents of the specified number of LR words, starting from the specified word.

**Command Format**



**Note** 1. Beginning word: 0000 to 0063 in CQM1 PCs, 0000 to 0015 in CPM1 PCs  
2. No. of words: 0001 to 0064 in CQM1 PCs, 0001 to 0016 in CPM1 PCs

**Response Format**



**Read Data (Response)**

The contents of the number of words specified by the command are returned in hexadecimal as a response. The words are returned in order, starting with the specified beginning word.

**Parameters**

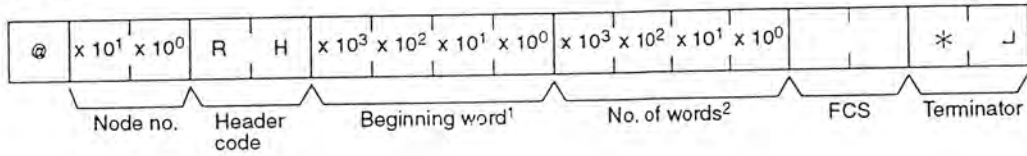
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับ...  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัด...

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับ...  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัด...

### 6-3-3 HR AREA READ — RH

Reads the contents of the specified number of HR words, starting from the specified word.

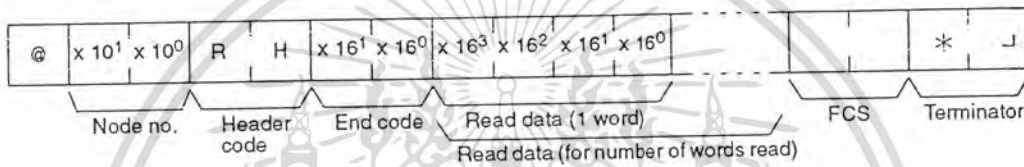
#### Command Format



- Note**
1. Beginning word: 0000 to 0099 in CQM1 PCs, 0000 to 0019 in CPM1 PCs
  2. No. of words: 0001 to 0100 in CQM1 PCs, 0001 to 0020 in CPM1 PCs

#### Response Format

An end code of 00 indicates normal completion.



#### Parameters

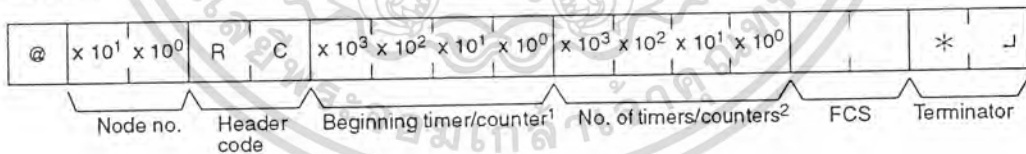
##### Read Data (Response)

The contents of the number of words specified by the command are returned in hexadecimal as a response. The words are returned in order, starting with the specified beginning word.

### 6-3-4 PV READ — RC

Reads the contents of the specified number of timer/counter PVs (present values), starting from the specified timer/counter.

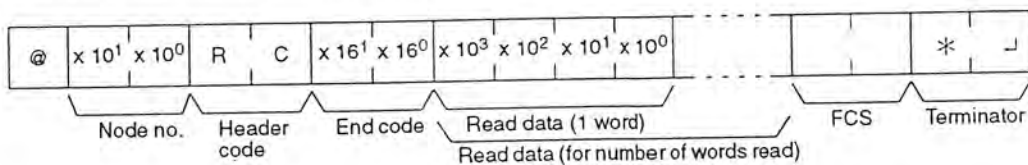
#### Command Format



- Note**
1. Beginning T/C: 0000 to 0511 in CQM1 PCs, 0000 to 0127 in CPM1 PCs
  2. No. of T/Cs: 0001 to 0512 in CQM1 PCs, 0001 to 0128 in CPM1 PCs

#### Response Format

An end code of 00 indicates normal completion.



The response will be divided when reading more than 30 words of data.

#### Parameters

##### Read Data (Response)

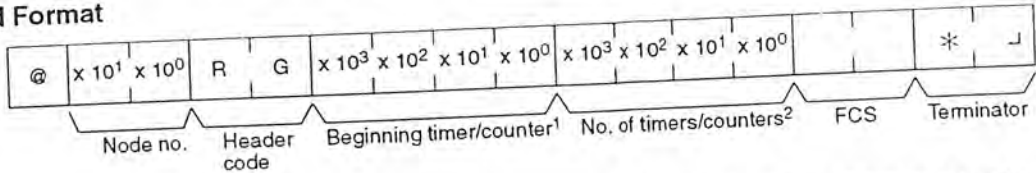
The number of present values specified by the command is returned in hexadecimal as a response. The PVs are returned in order, starting with the specified beginning timer/counter.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับใช้ภายในเท่านั้น การนำออกเผยแพร่โดยไม่ได้รับอนุญาตถือว่าผิดกฎหมาย  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 6-3-5 TC STATUS READ — RG

Reads the status of the Completion Flags of the specified number of timers/counters, starting from the specified timer/counter.

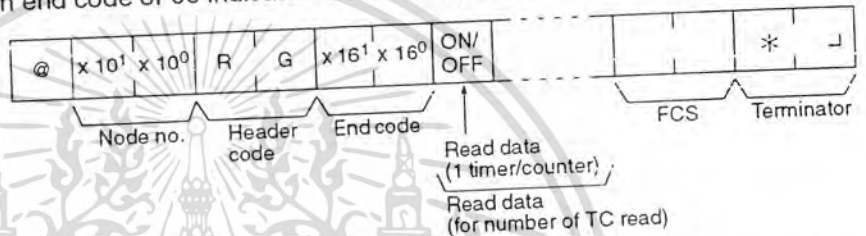
#### Command Format



- Note**
1. Beginning T/C: 0000 to 0511 in CQM1 PCs, 0000 to 0127 in CPM1 PCs
  2. No. of T/Cs: 0001 to 0512 in CQM1 PCs, 0001 to 0128 in CPM1 PCs

#### Response Format

An end code of 00 indicates normal completion.



The response will be divided when reading the status of more than 123 timer/counters.

#### Parameters

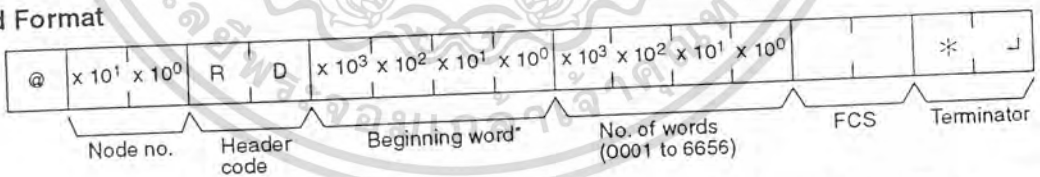
##### Read Data (Response)

The status of the number of Completion Flags specified by the command is returned as a response. "1" indicates that the Completion Flag is ON.

### 6-3-6 DM AREA READ — RD

Reads the contents of the specified number of DM words, starting from the specified word.

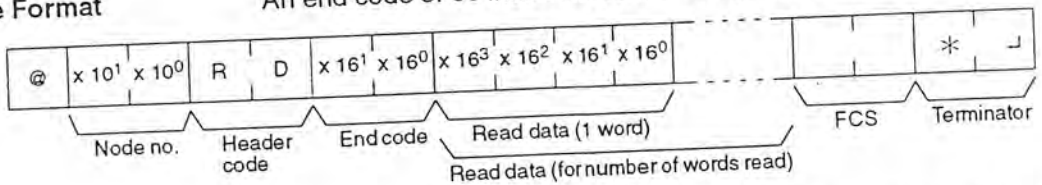
#### Command Format



- Note**
- Beginning word: 0000 to 6655 in CQM1 PCs, 0000 to 1023 and 6144 to 6655 in CPM1 PCs.

#### Response Format

An end code of 00 indicates normal completion.



- Note**
1. Words 1024 to 6143 in CPM1 PCs cannot be specified. If an attempt to read any of these words is made, a response of 0000 will be returned.
  2. The response will be divided when reading more than 30 words of data.

##### Read Data (Response)

The contents of the number of words specified by the command are returned hexadecimal as a response. The words are returned in order, starting with the specified beginning word.

#### Parameters

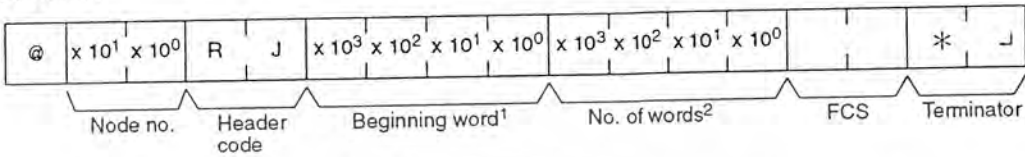
เอกสารที่สงวนไว้สำหรับกา...  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัด...  
ของเอกสารทุกครั้งที่มีการนำไปใช้

**Note** Be careful about the configuration of the DM area, as it varies depending on the CPU model.

### 6-3-7 AR AREA READ — RJ

Reads the contents of the specified number of AR words, starting from the specified word.

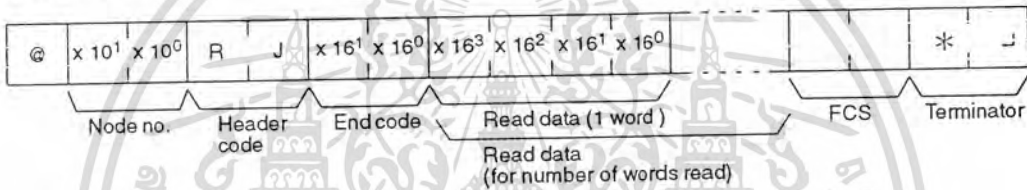
**Command Format**



**Note** 1. Beginning word: 0000 to 0027 in CQM1 PCs, 0000 to 0015 in CPM1 PCs  
 2. No. of words: 0001 to 0028 in CQM1 PCs, 0001 to 0016 in CPM1 PCs

**Response Format**

An end code of 00 indicates normal completion.



**Parameters**

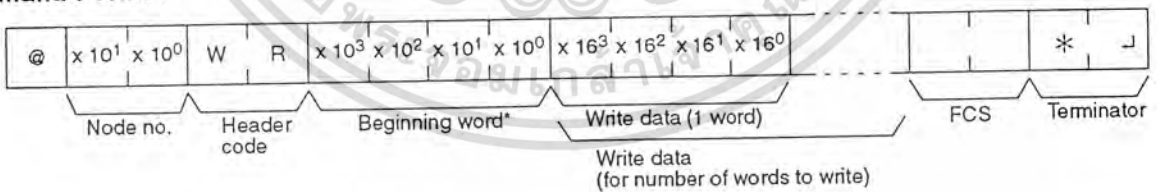
**Read Data (Response)**

The contents of the number of words specified by the command are returned in hexadecimal as a response. The words are returned in order, starting with the specified beginning word.

### 6-3-8 IR/SR AREA WRITE — WR

Writes data to the IR and SR areas, starting from the specified word. Writing is done word by word.

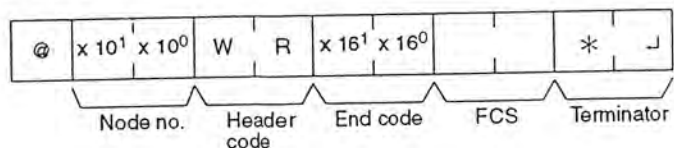
**Command Format**



**Note** 1. Beginning word: 0000 to 0252 in CQM1 PCs, 0000 to 0019 and 0200 to 0252 in CPM1 PCs.  
 2. Divide the command when writing more than 30 words of data.

**Response Format**

An end code of 00 indicates normal completion.



**Note** Words 0020 to 0199 in CPM1 PCs cannot be specified. If an attempt to write to any of these words is made, the writing operation will not be executed and normal completion occurs.

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้พิมพ์คำสั่งนี้อีก และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Host Link Commands

Parameters

Write Data (Command)

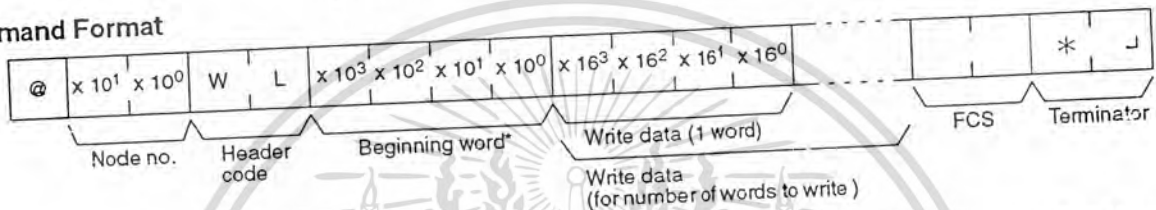
Specify in order the contents of the number of words to be written to the LR or SR area in hexadecimal, starting with the specified beginning word.

**Note** If data is specified for writing which exceeds the allowable range, an error will be generated and the writing operation will not be executed. If, for example, 252 is specified as the beginning word for writing, and two words of data are specified, then 253 will become the last word for writing data, and the command will not be executed because SR 253 is beyond the writeable range.

6-3-9 LR AREA WRITE — WL

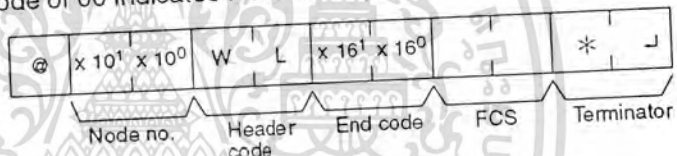
Writes data to the LR area, starting from the specified word. Writing is done word by word.

Command Format



**Note** Beginning word: 0000 to 0063 in CQM1 PCs, 0000 to 0015 in CPM1 PCs  
An end code of 00 indicates normal completion.

Response Format



Parameters

Write Data (Command)

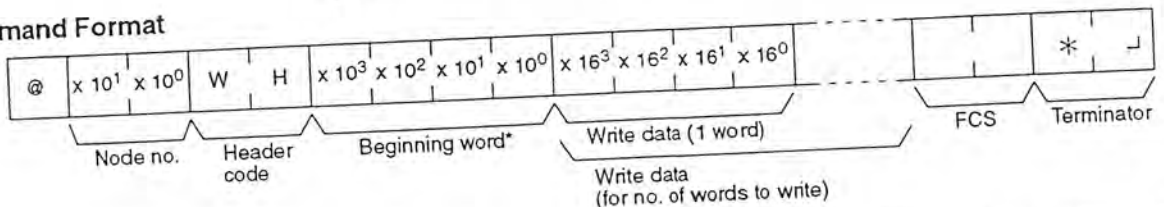
Specify in order the contents of the number of words to be written to the LR area in hexadecimal, starting with the specified beginning word.

**Note** If data is specified for writing which exceeds the allowable range, an error will be generated and the writing operation will not be executed. If, for example, 60 is specified as the beginning word for writing and five words of data are specified, then 64 will become the last word for writing data, and the command will not be executed because LR 64 is beyond area boundary.

6-3-10 HR AREA WRITE — WH

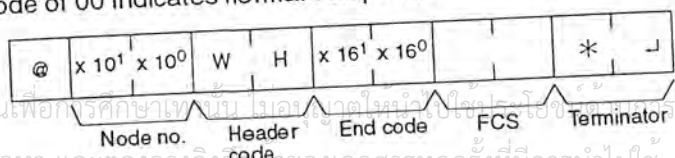
Writes data to the HR area, starting from the specified word. Writing is done word by word.

Command Format



**Note** Beginning word: 0000 to 0063 in CQM1 PCs, 0000 to 0019 in CPM1 PCs  
An end code of 00 indicates normal completion.

Response Format



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่ควรนำข้อมูลไปเผยแพร่โดยไม่ได้รับอนุญาต  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Parameters

**Write Data (Command)**

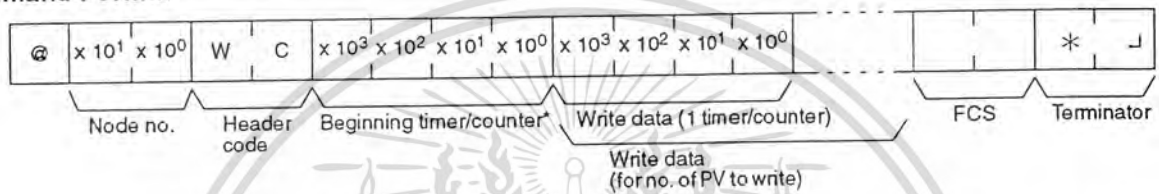
Specify in order the contents of the number of words to be written to the HR area in hexadecimal, starting with the specified beginning word.

**Note** If data is specified for writing which exceeds the allowable range, an error will be generated and the writing operation will not be executed. If, for example, 98 is specified as the beginning word for writing, and three words of data are specified, then 100 will become the last word for writing data, and the command will not be executed because HR 100 is beyond area boundary.

**6-3-11 PV WRITE — WC**

Writes the PVs (present values) of timers/counters starting from the specified timer/counter.

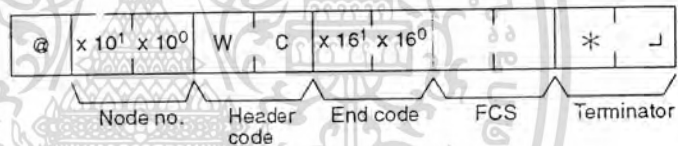
Command Format



**Note** 1. Beginning T/C: 0000 to 0511 in CQM1 PCs, 0000 to 0127 in CPM1 PCs  
 2. Divide the command when writing more than 29 words of data.

Response Format

An end code of 00 indicates normal completion.



Parameters

**Write Data (Command)**

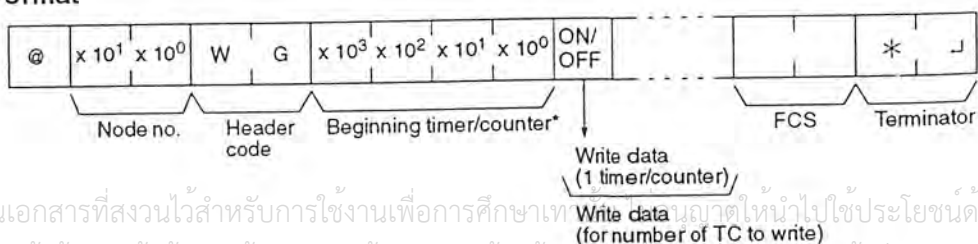
Specify in decimal numbers (BCD) the present values for the number of timers/counters that are to be written, starting from the beginning timer/counter.

**Note** 1. When this command is used to write data to the PV area, the Completion Flags for the timers/counters that are written will be turned OFF.  
 2. If data is specified for writing which exceeds the allowable range, an error will be generated and the writing operation will not be executed. If, for example, 510 is specified as the beginning word for writing, and three words of data are specified, then 512 will become the last word for writing data, and the command will not be executed because TC 512 is beyond area boundary.

**6-3-12 TC STATUS WRITE — WG**

Writes the status of the Completion Flags for timers and counters in the TC area, starting from the specified timer/counter (number). Writing is done number by number.

Command Format



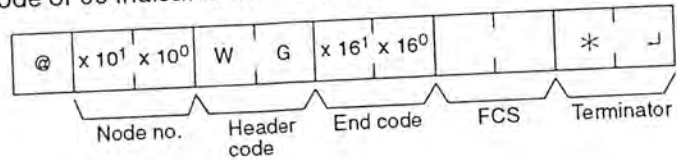
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น กรุณาอย่าให้นำไปใช้ประโยชน์ด้านการค้า  
 ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Host Link Commands

- Note**
1. Beginning T/C: 0000 to 0511 in CQM1 PCs, 0000 to 0127 in CPM1 PCs
  2. Divide the command when writing the status of more than 118 timer/counters.

Response Format

An end code of 00 indicates normal completion.



Parameters

**Write Data (Command)**

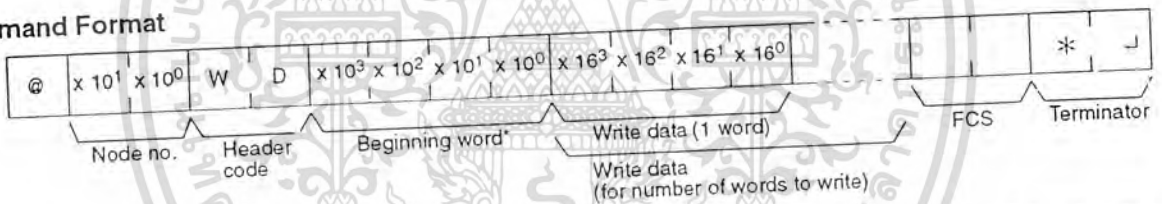
Specify the status of the Completion Flags, for the number of timers/counters to be written, in order (from the beginning word) as ON (i.e., "1") or OFF (i.e., "0"). When a Completion Flag is ON, it indicates that the time or count is up.

- Note** If data is specified for writing which exceeds the allowable range, an error will be generated and the writing operation will not be executed. If, for example, 510 is specified as the beginning word for writing, and three words of data are specified, then 512 will become the last word for writing data, and the command will not be executed because TC 512 is beyond area boundary.

6-3-13 DM AREA WRITE — WD

Writes data to the DM area, starting from the specified word. Writing is done word by word.

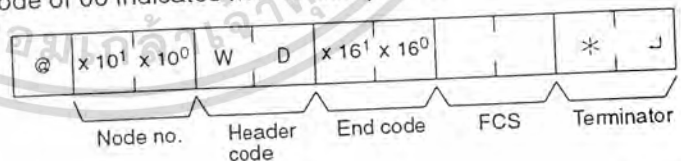
Command Format



- Note**
1. Beginning word: 0000 to 6143 in CQM1 PCs, 0000 to 1023 and 6144 to 6655 in CPM1 PCs
  2. Divide the command when writing more than 29 words of data.

Response Format

An end code of 00 indicates normal completion.



- Note** Words 1024 to 6143 in CPM1 PCs cannot be specified. If an attempt to write to any of these words is made, the writing operation will not be executed and normal completion occurs.

Parameters

**Write Data (Command)**

Specify in order the contents of the number of words to be written to the DM area in hexadecimal, starting with the specified beginning word.

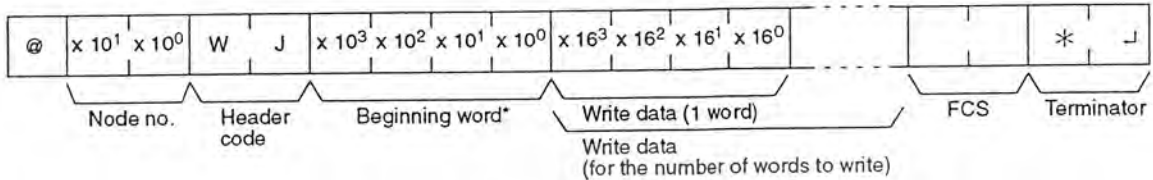
- Note**
1. If data is specified for writing which exceeds the allowable range, an error will be generated and the writing operation will not be executed. If, for example, 6142 is specified as the beginning word for writing, and three words of data are specified, then 6144 will become the last word for writing data, and the command will not be executed because DM 6144 is beyond the write-able range.
  2. Be careful about the configuration of the DM area, as it varies depending on the CPU model.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้การศึกษาเท่านั้น ไม่อนุญาตให้นำไปเผยแพร่โดยไม่ได้รับอนุญาต  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปดเปลี่ยนแปลงข้อมูลใดๆ และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 6-3-14 AR AREA WRITE — WJ

Writes data to the AR area, starting from the specified word. Writing is done word by word.

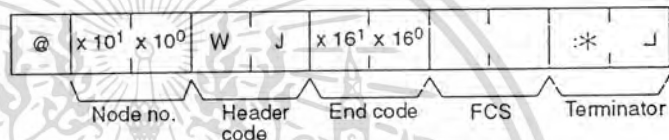
**Command Format**



**Note** Beginning word: 0000 to 0027 in CQM1 PCs, 0000 to 0015 in CPM1 PCs

**Response Format**

An end code of 00 indicates normal completion.



**Parameters**

**Write Data (Command)**

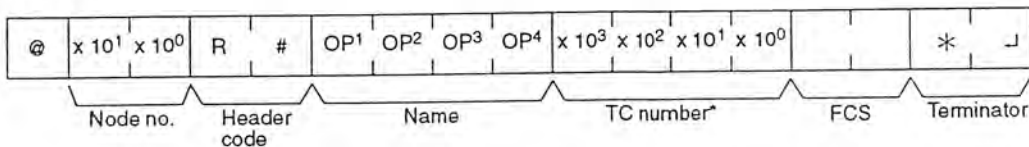
Specify in order the contents of the number of words to be written to the AR area in hexadecimal, starting with the specified beginning word.

**Note** If data is specified for writing which exceeds the allowable range, an error will be generated and the writing operation will not be executed. If, for example, 26 is specified as the beginning word for writing, and three words of data are specified, then 28 will become the last word for writing data, and the command will not be executed because AR 28 is beyond the writeable range.

### 6-3-15 SV READ 1 — R#

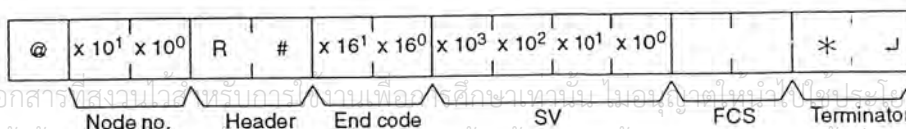
Searches for the first instance of a TIM, TIMH(15), CNT, and CNTR(12) instruction with the specified TC number in the user's program and reads the PV, which assumed to be set as a constant. The SV that is read is a 4-digit decimal number (BCD). The program is searched from the beginning, which may take as much as 10 seconds to produce a response.

**Command Format**



**Note** TC number: 0000 to 0511 in CQM1 PCs, 0000 to 0127 in CPM1 PCs

**Response Format**



เอกสารนี้เป็นเอกสารสงวนไว้สำหรับกรใช้งานเพื่อศึกษาเท่านั้น ไม่อนุญาตให้นำไปประโชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้เผยแพร่เนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Host Link Commands

Parameters

Name, TC Number (Command)

Specify the instruction for reading the SV in "Name." Make this setting in 4 characters. In "TC number," specify the timer/counter number used for the instruction.

Instruction name				Classification
OP1	OP2	OP3	OP4	
T	I	M	(Space)	TIMER
T	I	M	H	HIGH-SPEED TIMER
C	N	T	(Space)	COUNTER
C	N	T	R	REVERSIBLE COUNTER

SV (Response)

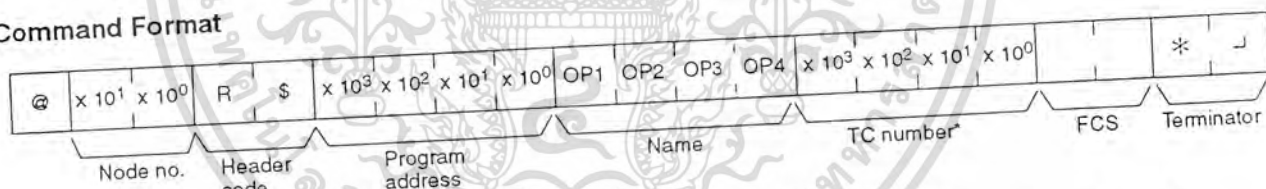
The constant SV is returned.

- Note**
1. The instruction specified under "Name" must be in four characters.
  2. If the same instruction is used more than once in a program, only the first one will be read.
  3. Use this command only when it is definite that a constant SV has been set.
  4. The response end code will indicate an error (16) if the SV wasn't entered as a constant.

6-3-16 SV READ 2 — R\$

Reads the constant SV or the word address where the SV is stored. The SV that is read is a 4-digit decimal number (BCD) written as the second operand for the TIM, TIMH(15), CNT, or CNTR(12) instruction at the specified program address in the user's program. This can only be done with a program of less than 10K.

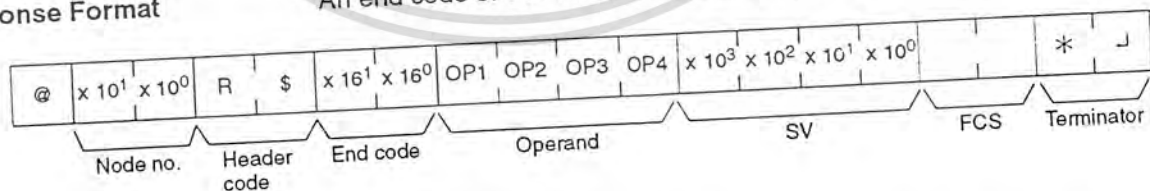
Command Format



**Note** TC number: 0000 to 0511 in CQM1 PCs, 0000 to 0127 in CPM1 PCs

Response Format

An end code of 00 indicates normal completion.



Parameters

Name, TC Number (Command)

Specify the name of the instruction for reading the SV in "Name." Make this setting in 4 characters. In "TC number," specify the timer/counter number used for the instruction.

Instruction name				Classification
OP1	OP2	OP3	OP4	
T	I	M	(Space)	TIMER
T	I	M	H	HIGH-SPEED TIMER
C	N	T	(Space)	COUNTER
C	N	T	R	REVERSIBLE COUNTER

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้เผยแพร่หรือทำซ้ำโดยไม่ได้รับอนุญาต  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปดเนื้อหา และต้องวางอินลิ้งค์ของเอกสารทุกครั้งที่มีการนำไปใช้

**Operand, SV (Response)**

The name that indicates the SV classification is returned to "Operand," and either the word address where the SV is stored or the constant SV is returned to "SV."

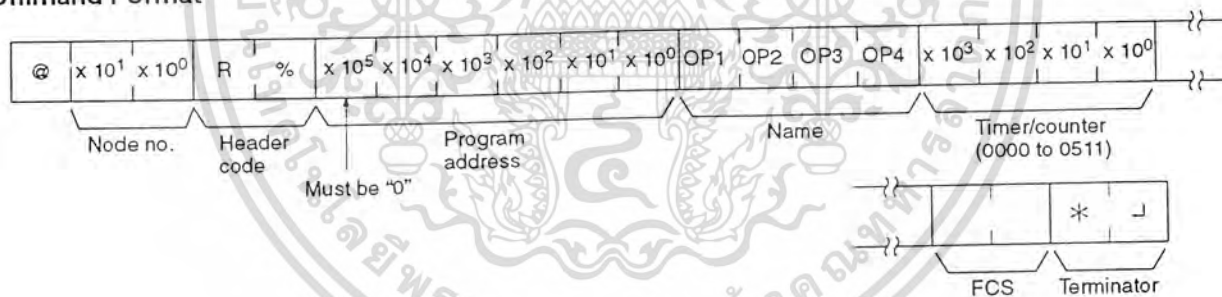
Operand				Classification	Constant or word address	
OP1	OP2	OP3	OP4		CQM1 PCs	CPM1 PCs
C	I	O	(Space)	IR or SR	0000 to 0255	0000 to 0019 0200 to 0255
L	R	(Space)	(Space)	LR	0000 to 0063	0000 to 0015
H	R	(Space)	(Space)	HR	0000 to 0099	0000 to 0019
A	R	(Space)	(Space)	AR	0000 to 0027	0000 to 0015
D	M	(Space)	(Space)	DM	0000 to 6655	0000 to 6655
D	M	*	(Space)	DM (indirect)	0000 to 6655	0000 to 6655
C	O	N	(Space)	Constant	0000 to 9999	0000 to 9999

**Note** The instruction name specified under "Name" must be in four characters. Fill any gaps with spaces to make a total of four characters.

**6-3-17 SV READ 3 — R% (CQM1 PCs Only)**

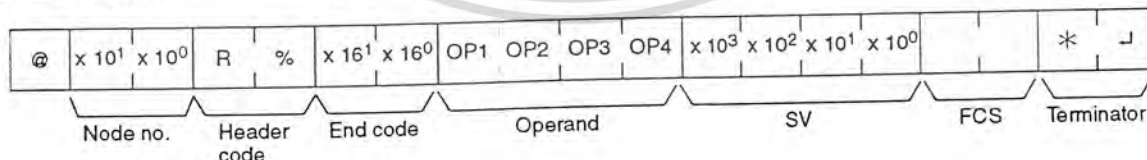
Reads the constant SV or the word address where the SV is stored. The SV that is read is a 4-digit decimal number (BCD) written in the second word of the TIM, TIMH(15), CNT, or CNTR(12) instruction at the specified program address in the user's program. With this command, program addresses can be specified for a program of 10K or more.

**Command Format**



**Response Format**

An end code of 00 indicates normal completion.



**Parameters**

**Name, TC Number (Command)**

Specify the name of the instruction for reading the SV in "Name." Make this setting in 4 characters. In "TC number," specify the timer/counter number used by the instruction.

Instruction name				Classification	TC number range
OP1	OP2	OP3	OP4		
T	I	M	(Space)	TIMER	0000 to 0511
T	I	M	H	HIGH-SPEED TIMER	
C	N	T	(Space)	COUNTER	
C	N	T	R	REVERSIBLE COUNTER	

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้ภายในเพื่อการศึกษาและการวิจัยเท่านั้น ไม่สามารถนำออกจำหน่ายหรือทำซ้ำโดยไม่ได้รับอนุญาต

**Operand, SV (Response)**

The name that indicates the SV classification is returned to "Operand," and either the word address where the SV is stored or the constant SV is returned to "SV."

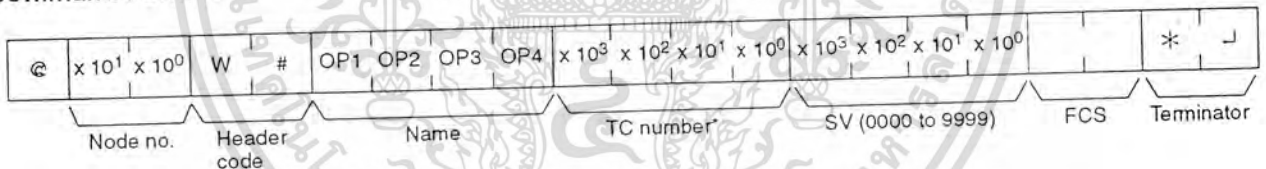
Operand				Classification	Constant or word address
OP1	OP2	OP3	OP4		
C	I	O	(Space)	IR or SR	0000 to 0255
L	R	(Space)	(Space)	LR	0000 to 0063
H	R	(Space)	(Space)	HR	0000 to 0099
A	R	(Space)	(Space)	AR	0000 to 0027
D	M	(Space)	(Space)	DM	0000 to 6655
D	M	*	(Space)	DM (indirect)	0000 to 6655
C	O	N	(Space)	Constant	0000 to 9999

**Note** The instruction name specified under "Name" must be in four characters. Fill any gaps with spaces to make a total of four characters.

**6-3-18 SV CHANGE 1 — W#**

Searches for the first instance of the specified TIM, TIMH(15), CNT, or CNTR(12) instruction in the user's program and changes the SV to new constant SV specified in the second word of the instruction. The program is searched from the beginning, and it may therefore take approximately 10 seconds to produce a response.

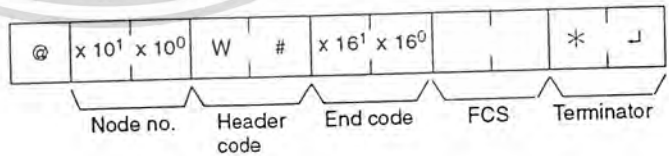
**Command Format**



**Note** TC number: 0000 to 0511 in CQM1 PCs, 0000 to 0127 in CPM1 PCs

**Response Format**

An end code of 00 indicates normal completion.



**Parameters**

**Name, TC Number (Command)**

In "Name," specify the name of the instruction, in four characters, for changing the SV. In "TC number," specify the timer/counter number used for the instruction.

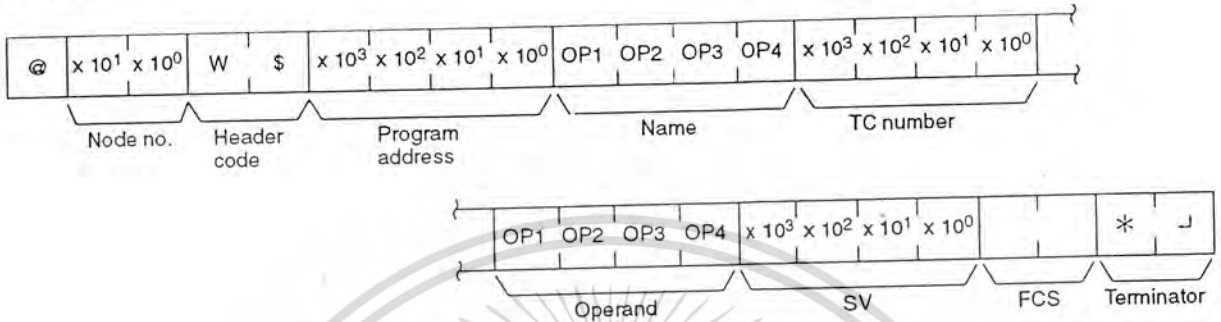
Instruction name				Classification
OP1	OP2	OP3	OP4	
T	I	M	(Space)	TIMER
T	I	M	H	HIGH-SPEED TIMER
C	N	T	(Space)	COUNTER
C	N	T	R	REVERSIBLE COUNTER

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับใช้ภายในงานเพื่อการศึกษาเท่านั้น ไม่สามารถเผยแพร่โดยไม่ได้รับอนุญาตจากทางบริษัทฯ  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดลอกหรือเผยแพร่ข้อมูลใดๆ ไปยังผู้อื่นโดยไม่ได้รับอนุญาตจากทางบริษัทฯ

### 6-3-19 SV CHANGE 2 — W\$

Changes the contents of the second word of the TIM, TIMH(15), CNT, or CNTR(12) at the specified program address in the user's program. This can only be done with a program of less than 10K.

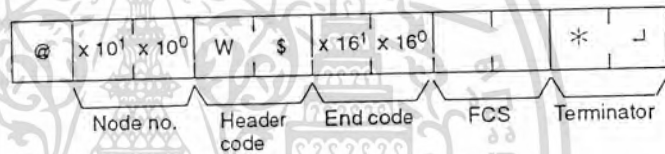
#### Command Format



Note TC number: 0000 to 0511 in CQM1 PCs, 0000 to 0127 in CPM1 PCs

#### Response Format

An end code of 00 indicates normal completion.



#### Parameters

##### Name, TC Number (Command)

In "Name," specify the name of the instruction, in four characters, for changing the SV. In "TC number," specify the timer/counter number used for the instruction.

Instruction name				Classification
OP1	OP2	OP3	OP4	
T	I	M	(Space)	TIMER
T	I	M	H	HIGH-SPEED TIMER
C	N	T	(Space)	COUNTER
C	N	T	R	REVERSIBLE COUNTER

##### Operand, SV (Response)

In "Operand," specify the name that indicates the SV classification. Specify the name in four characters. In "SV," specify either the word address where the SV is stored or the constant SV.

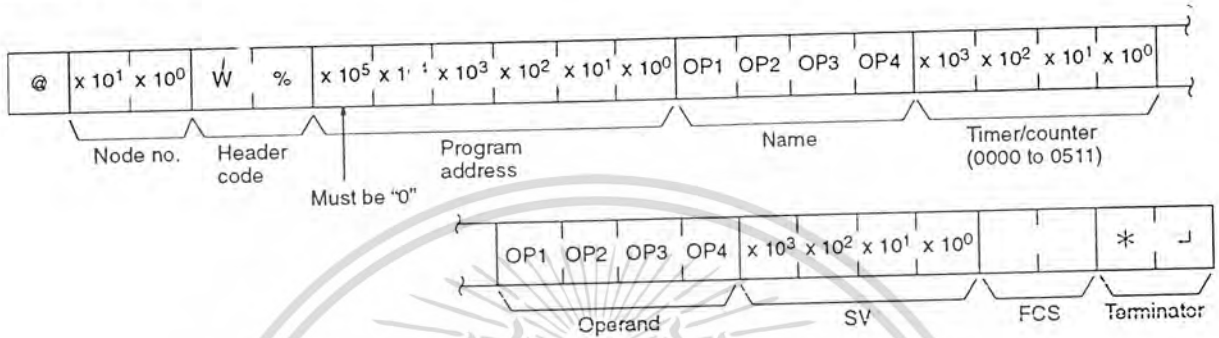
Operand				Classification	Constant or word address	
OP1	OP2	OP3	OP4		CQM1 PCs	CPM1 PCs
C	I	O	(Space)	IR or SR	0000 to 0252	0000 to 0019 0200 to 0252
L	R	(Space)	(Space)	LR	0000 to 0063	0000 to 0015
H	R	(Space)	(Space)	HR	0000 to 0099	0000 to 0019
A	R	(Space)	(Space)	AR	0000 to 0027	0000 to 0015
D	M	(Space)	(Space)	DM	0000 to 6655	0000 to 1023 6144 to 6655
D	M	*	(Space)	DM (indirect)	0000 to 6655	0000 to 1023 6144 to 6655
C	O	N	(Space)	Constant	0000 to 9999	0000 to 9999

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษเท่านั้น ไม่อนุญาตให้ทำประโยชน์อื่นใด  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดลอกหรือเผยแพร่เอกสารนี้โดยไม่ได้รับอนุญาตจากทางบริษัทฯ

### 6-3-20 SV CHANGE 3 — W% (CQM1 PCs Only)

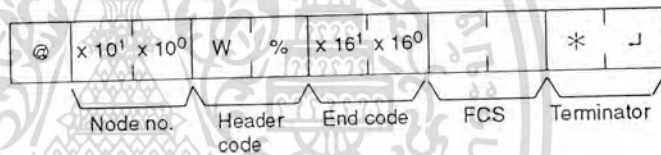
Changes the contents of the second word of the TIM, TIMH(15), CNT, or CNTR(12) at the specified program address in the user's program. With this command, program address can be specified for a program of more than 10K.

#### Command Format



#### Response Format

An end code of 00 indicates normal completion.



#### Parameters

##### Name, TC Number (Command)

In "Name," specify the name of the instruction, in four characters, for changing the SV. In "TC number," specify the timer/counter number used for the instruction.

Instruction name				Classification	TC number range
OP1	OP2	OP3	OP4		
T	I	M	(Space)	TIMER	0000 to 0511
T	I	M	H	HIGH-SPEED TIMER	
C	N	T	(Space)	COUNTER	
C	N	T	R	REVERSIBLE COUNTER	

##### Operand, New SV (Response)

In "Operand," specify the name that indicates the SV classification. Specify the name in four characters. In "SV," specify either the word address where the SV is stored or the constant SV.

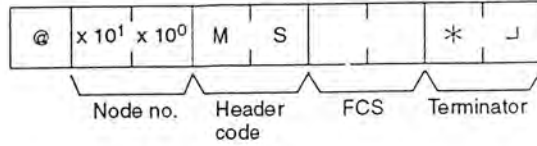
Operand				Classification	Constant or word address
OP1	OP2	OP3	OP4		
C	I	O	(Space)	IR or SR	0000 to 0252
L	R	(Space)	(Space)	LR	0000 to 0063
H	R	(Space)	(Space)	HR	0000 to 0099
A	R	(Space)	(Space)	AR	0000 to 0027
D	M	(Space)	(Space)	DM	0000 to 6655
D	M	*	(Space)	DM (indirect)	0000 to 6655
C	O	N	(Space)	Constant	0000 to 9999

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับใช้เฉพาะภายในเท่านั้น ไม่ควรเผยแพร่ไปใช้ประโยชน์อื่นใด  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้เปลี่ยนแปลงเนื้อหาในเอกสารทุกครั้งที่มีการแก้ไข

### 6-3-21 STATUS READ — MS

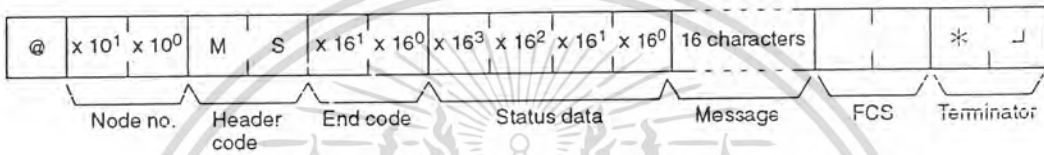
Reads the PC operating conditions.

#### Command Format



#### Response Format

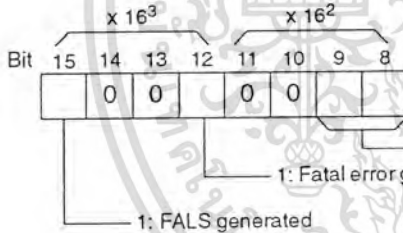
An end code of 00 indicates normal completion.



#### Parameters

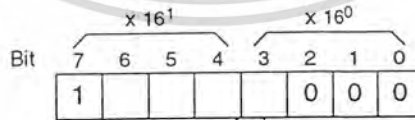
##### Status Data, Message (Response)

“Status data” consists of four digits (two bytes) hexadecimal. The leftmost byte indicates CPU operation mode, and the rightmost byte indicates the size of the program area.



Bit	Operation mode
9 8	PROGRAM mode
1 0	RUN mode
1 1	MONITOR mode

This area is different from that of STATUS WRITE.



Bit	Program area
6 5 4	None
0 0 1	4 Kbytes
0 1 0	8 Kbytes

Program area write-protection  
0: Write-protected  
1: Not write-protected

(In CQM1 PCs, turn DIP switch pin 1 ON to write-protect the program area.)

In CQM1 PCs, the “Message” parameter is a FAL/FALS number that exists when the command is executed. When there is no message, this parameter is omitted.

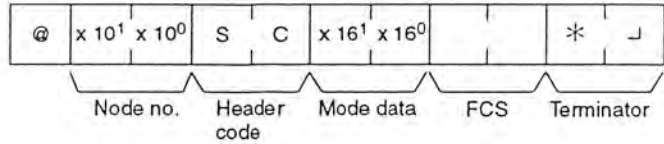
In CPM1 PCs, the “Message” parameter is a 16-character message that exists when the command is executed. When there is no message, this parameter is omitted.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับใช้ภายในเท่านั้น ไม่สามารถเผยแพร่โดยไม่ได้รับอนุญาต  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้นำเนื้อหา และตองอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 6-3-22 STATUS WRITE — SC

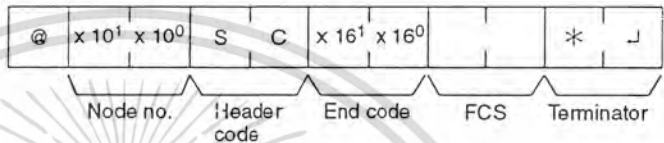
Changes the PC operating mode.

**Command Format**



**Response Format**

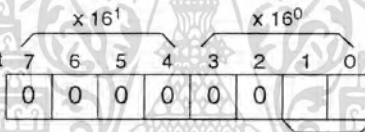
An end code of 00 indicates normal completion.



**Parameters**

**Mode Data (Command)**

"Mode data" consists of two digits (one byte) hexadecimal. With the leftmost two bits, specify the PC operating mode. Set all of the remaining bits to "0."



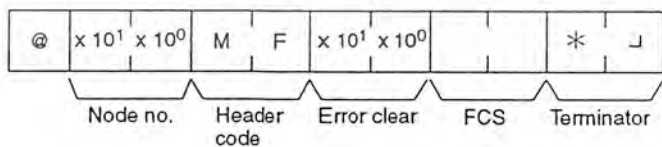
Bit	Operation mode
1 0	
0 0	PROGRAM mode
1 0	MONITOR mode
1 1	RUN mode

This area is different from that of STATUS READ.

### 6-3-23 ERROR READ — MF

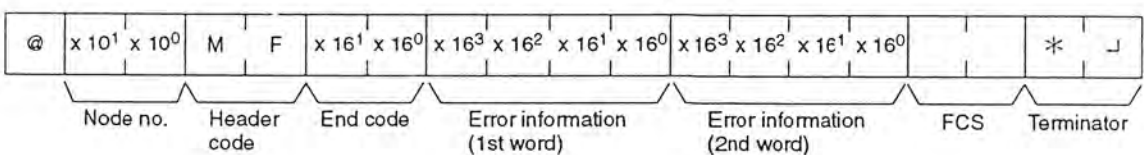
Reads and clears errors in the PC. Also checks whether previous errors have been cleared.

**Command Format**



**Response Format**

An end code of 00 indicates normal completion.



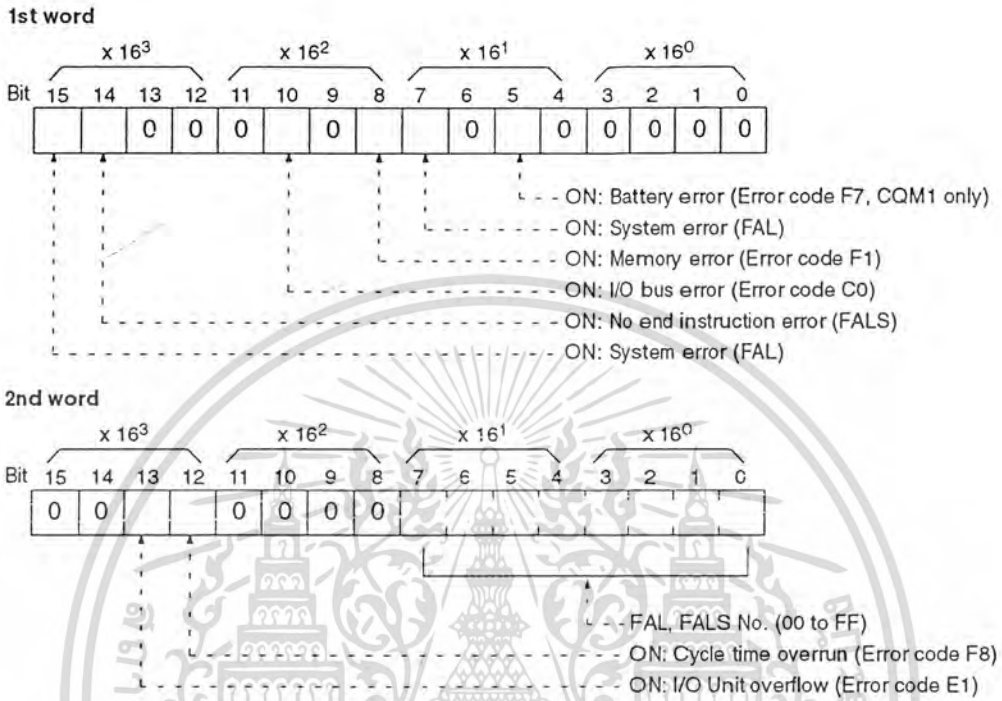
**Parameters**

**Error Clear (Command)**

Specify 01 to clear errors and 00 to not clear errors (BCD). Fatal errors can be cleared only when the PC is in PROGRAM mode.

**Error Information (Response)**

The error information comes in two words.

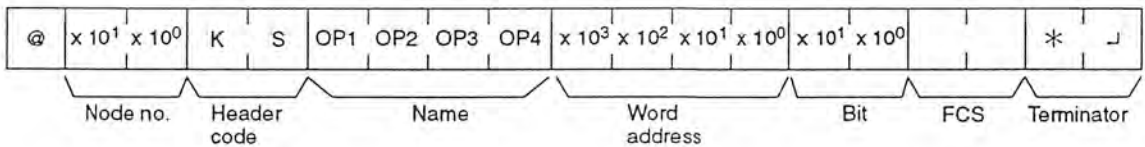


**6-3-24 FORCED SET — KS**

Force sets a bit in the IR, SR, LR, HR, AR, or TC area. Just one bit can be force set at a time.

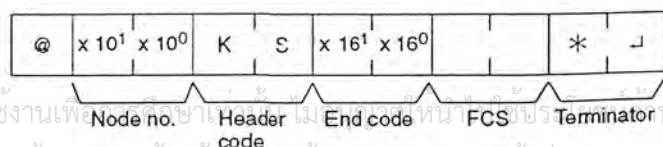
Once a bit has been forced set or reset, that status will be retained until a FORCED SET/RESET CANCEL (KG) command or the next FORCED SET/RESET command is transmitted.

**Command Format**



**Response Format**

An end code of 00 indicates normal completion.



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ในการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Host Link Commands

Parameters

Name, Word address, Bit (Command)

In "Name," specify the area (i.e., IR, SR, LR, HR, AR, or TC) that is to be forced set. Specify the name in four characters. In "Word address," specify the address of the word, and in "Bit" the number of the bit that is to be forced set.

Name				Classification	Word address setting range		Bit
OP1	OP2	OP3	OP4		CQM1 PCs	CPM1 PCs	
C	I	O	(Space)	IR or SR	0000 to 0252	0000 to 0019 0200 to 0252	00 to 15 (decimal)
L	R	(Space)	(Space)	LR	0000 to 0063	0000 to 0015	
H	R	(Space)	(Space)	HR	0000 to 0099	0000 to 0019	
A	R	(Space)	(Space)	AR	0000 to 0027	0000 to 0015	
T	I	M	(Space)	Completion Flag (timer)	0000 to 0511	0000 to 0127	
T	I	M	H	Completion Flag (high-speed timer)			
C	N	T	(Space)	Completion Flag (counter)			
C	N	T	R	Completion Flag (reversible counter)			

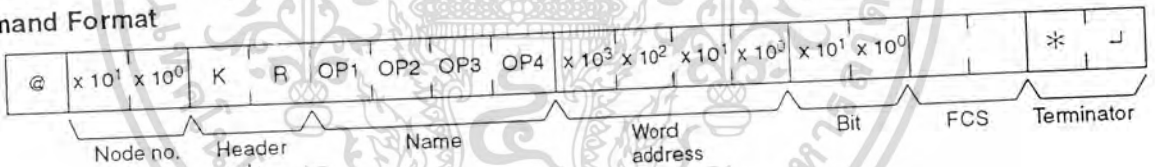
Note The area specified under "Name" must be in four characters. Add spaces after the data area name if it is shorter than four characters.

6-3-25 FORCED RESET — KR

Force resets a bit in the IR, SR, LR, HR, AR, or TC area. Just one bit can be forced reset at a time.

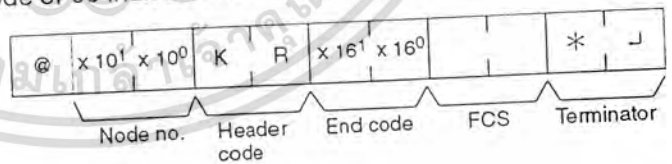
Once a bit has been forced set or reset, that status will be retained until a FORCED SET/RESET CANCEL (KC) command or the next FORCED SET/RESET command is transmitted.

Command Format



Response Format

An end code of 00 indicates normal completion.



Parameters

Name, Word address, Bit (Command)

In "Name," specify the area (i.e., IR, SR, LR, HR, AR, or TC) that is to be forced reset. Specify the name in four characters. In "Word address," specify the address of the word, and in "Bit" the number of the bit that is to be forced reset.

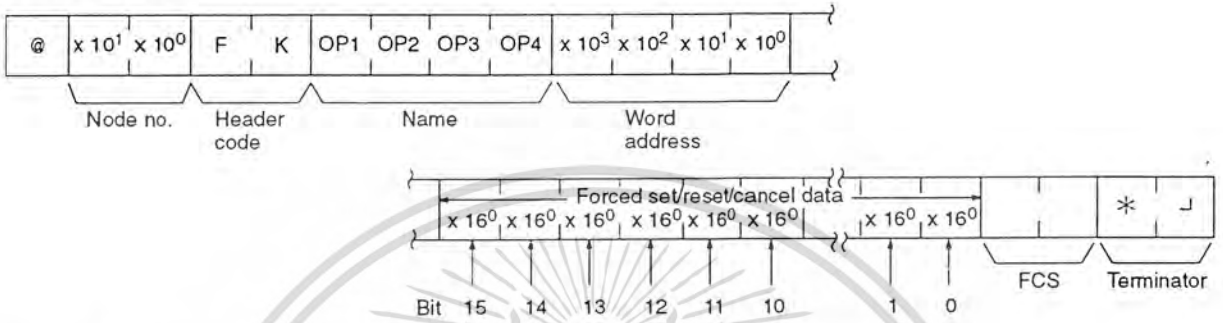
Name				Classification	Word address setting range		Bit
OP1	OP2	OP3	OP4		CQM1 PCs	CPM1 PCs	
C	I	O	(Space)	IR or SR	0000 to 0252	0000 to 0019 0200 to 0252	00 to 15 (decimal)
L	R	(Space)	(Space)	LR	0000 to 0063	0000 to 0015	
H	R	(Space)	(Space)	HR	0000 to 0099	0000 to 0019	
A	R	(Space)	(Space)	AR	0000 to 0027	0000 to 0015	
T	I	M	(Space)	Completion Flag (timer)	0000 to 0511	0000 to 0127	
T	I	M	H	Completion Flag (high-speed timer)			
C	N	T	(Space)	Completion Flag (counter)			
C	N	T	R	Completion Flag (reversible counter)			

**Note** The area specified under "Name" must be in four characters. Add spaces after the data area name if it is shorter than four characters.

### 6-3-26 MULTIPLE FORCED SET/RESET — FK

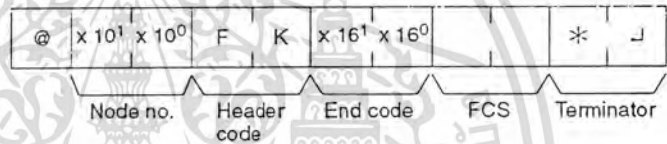
Force sets, force resets, or cancels the status of the bits in one word in the IR, SR, LR, HR, AR, or TC area.

**Command Format**



**Response Format**

An end code of 00 indicates normal completion.



**Parameters**

**Name, Word address (Command)**

In "Name," specify the area (i.e., IR, SR, LR, HR, AR, or TC) that is to be forced set or reset. Specify the name in four characters. In "Word address," specify the address of the word that is to be forced set or reset.

Name				Classification	Word address setting range	
OP1	OP2	OP3	OP4		CQM1 PCs	CPM1 PCs
C	I	O	(Space)	IR or SR	0000 to 0252	0000 to 0019 0200 to 0252
L	R	(Space)	(Space)	LR	0000 to 0063	0000 to 0015
H	R	(Space)	(Space)	HR	0000 to 0099	0000 to 0019
A	R	(Space)	(Space)	AR	0000 to 0027	0000 to 0015
T	I	M	(Space)	Completion Flag (timer)	0000 to 0511	0000 to 0127
T	I	M	H	Completion Flag (high-speed timer)		
C	N	T	(Space)	Completion Flag (counter)		
C	N	T	R	Completion Flag (reversible counter)		

**Forced set/Reset/Cancel data (Command)**

If a timer or counter completion flag is specified, only bit 15 is effective and all other bits will be ignored. Only force-setting and force-resetting are possible for timers/counters.

If a word address is specified, the content of the word specifies the desired process for each bit in the specified word, as shown in the following table.

Hexadecimal setting	Process
0000	No action (bit status not changed)
0002	Reset
0003	Set
0004	Forced-reset
0005	Forced-set
0008	Forced set/reset status cancel

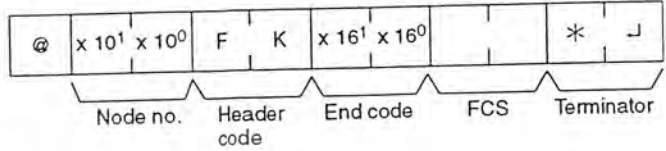
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับใช้ในงานเพื่อการศึกษาเท่านั้น ไม่สามารถนำออกเผยแพร่โดยไม่ได้รับอนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

Host Link Commands

The bits that are merely set or reset may change status the next time the program is executed, but bits that are force-set or force-reset will maintain the forced status until it is cleared.

Response Format

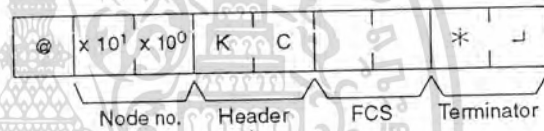
An end code of 00 indicates normal completion.



6-3-27 FORCED SET/RESET CANCEL — KC

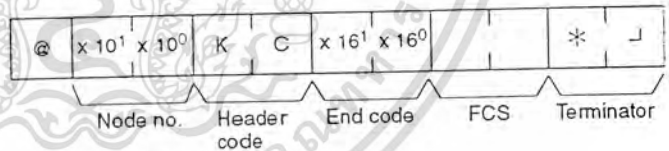
Cancels all forced set and forced reset bits (including those set by FORCED SET, FORCED RESET, and MULTIPLE FORCED SET/RESET). If multiple bits are set, the forced status will be cancelled for all of them. It is not possible to cancel bits one by one using KC.

Command Format



Response Format

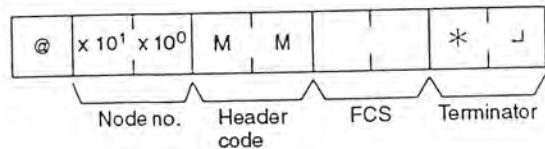
An end code of 00 indicates normal completion.



6-3-28 PC MODEL READ — MM

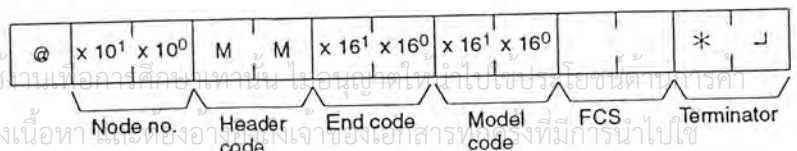
Reads the model type of the PC.

Command Format



Response Format

An end code of 00 indicates normal completion.



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้ภายในของสำนักงานเท่านั้น ไม่อนุญาตให้ไปเผยแพร่ภายนอกสำนักงาน  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องแจ้งเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Parameters

Model Code

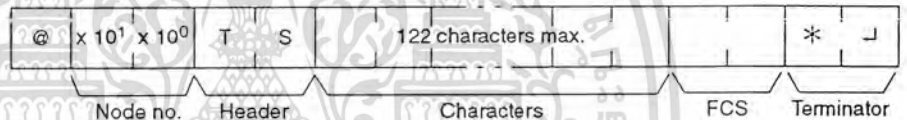
"Model code" indicates the PC model in two digits hexadecimal.

Model code	Model
01	C250
02	C500
03	C120
0E	C2000
10	C1000H
11	C2000H/CQM1/CPM1
12	C20H/C28H/C40H/C200H/C200HS
20	CV500
21	CV1000
22	CV2000
40	CVM1-CPU01-E
41	CVM1-CPU11-E
42	CVM1-CPU21-E

6-3-29 TEST — TS

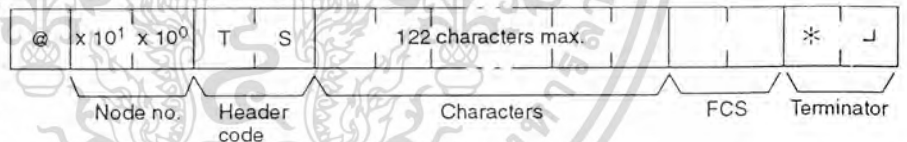
Returns, unaltered, one block of data transmitted from the host computer.

Command Format



Response Format

An end code of 00 indicates normal completion.



Parameters

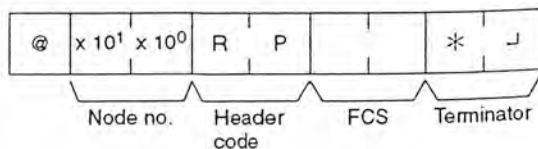
Characters (Command, Response)

For the command, this setting specifies any characters other than the carriage return (CHR\$(13)). For the response, the same characters as specified by the command will be returned unaltered if the test is successful.

6-3-30 PROGRAM READ — RP

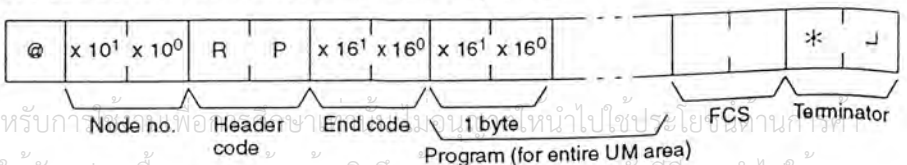
Reads the contents of the PC user's program area in machine language (object code). The contents are read as a block, from the beginning to the end.

Command Format



Response Format

An end code of 00 indicates normal completion.



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานภายในเท่านั้น ไม่ควรเผยแพร่โดยไม่ได้รับอนุญาต  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Parameters

**Program (Response)**

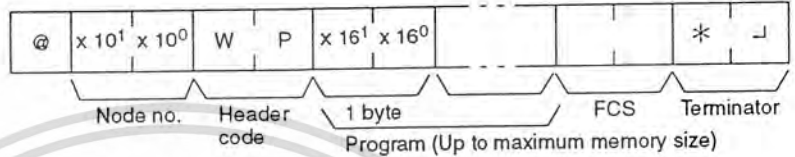
The program is read from the entire program area.

**Note** To stop this operation in progress, execute the ABORT (XZ) command.

**6-3-31 PROGRAM WRITE — WP**

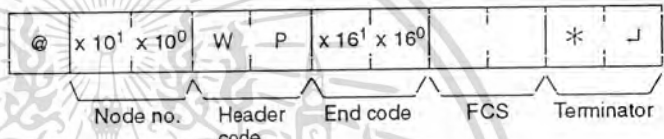
Writes to the PC user's program area the machine language (object code) program transmitted from the host computer. The contents are written as a block, from the beginning.

Command Format



Response Format

An end code of 00 indicates normal completion.



Parameters

**Program (Command)**

Program data up to the maximum memory size.

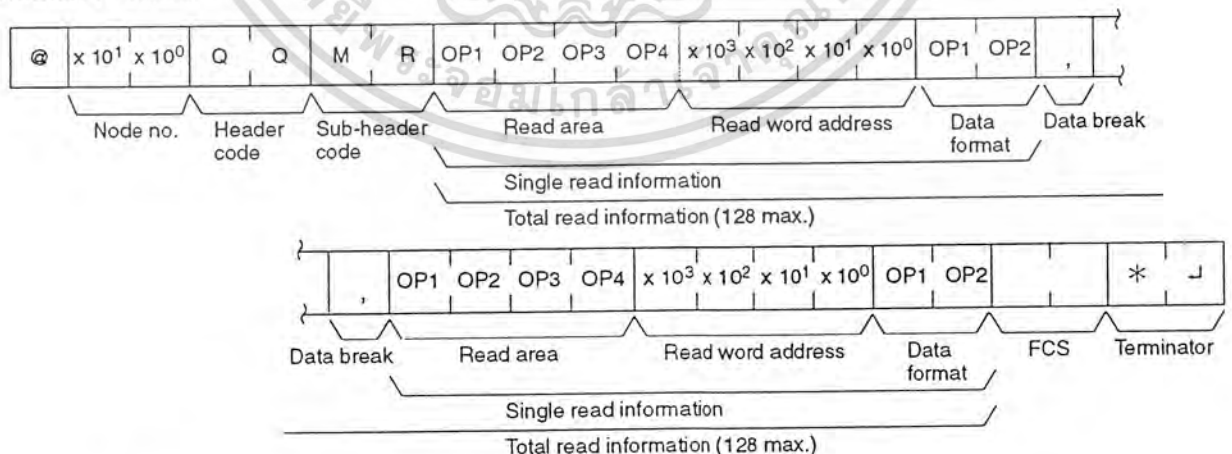
**6-3-32 COMPOUND COMMAND — QQ**

Registers at the PC all of the bits, words, and timers/counters that are to be read, and reads the status of all of them as a batch.

Registering Read Information

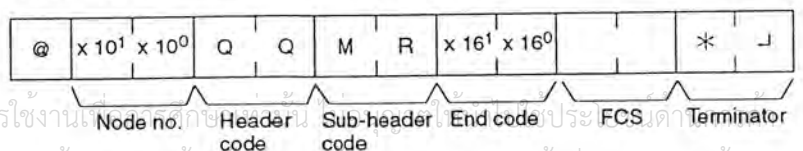
Register the information on all of the bits, words, and timers/counters that are to be read.

Command Format



Response Format

An end code of 00 indicates normal completion.



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานขององค์กรศึกษาเท่านั้น ไม่ควรเผยแพร่โดยไม่ได้รับอนุญาต  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Parameters

**Read Area (Command)**

Specify in four-character code the area that is to be read. The codes that can be specified are listed in the following table.

**Read Word address, Data Format (Command)**

Depending on the area and type of data that are to be read, the information to be read is as shown in the following table. The "read data" is specified in four digits BCD, and the data format is specified in two digits BCD.

Area classification	Read data	Read area	Read word		Data format
			CQM1 PCs	CPM1 PCs	
IR or SR	Bit	C I O (S)	0000 to 0255	0000 to 0019	00 to 15 (decimal)
	Word			0200 to 0255	"CH"
LR	Bit	L R (S) (S)	0000 to 0063	0000 to 0015	00 to 15 (decimal)
	Word				"CH"
HR	Bit	H R (S) (S)	0000 to 0099	0000 to 0019	00 to 15 (decimal)
	Word				"CH"
AR	Bit	A R (S) (S)	0000 to 0027	0000 to 0015	00 to 15 (decimal)
	Bit				"CH"
Timer	Completion Flag	T I M (S)	0000 to 0511	0000 to 0127	2 characters other than "CH"
	PV				"CH"
High-speed timer	Completion Flag	T I M H	0000 to 0511	0000 to 0127	2 characters other than "CH"
	PV				"CH"
Counter	Completion Flag	C N T (S)	0000 to 0511	0000 to 0127	2 characters other than "CH"
	PV				"CH"
Reversible counter	Completion Flag	C N T R	0000 to 0511	0000 to 0127	2 characters other than "CH"
	PV				"CH"
DM	Word	D M (S) (S)	0000 to 6655	0000 to 1023 6144 to 6655	Any 2 characters

(S): Space

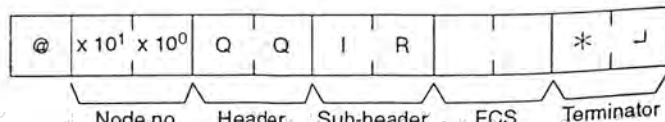
**Data Break (Command)**

The read information is specified one item at a time separated by a break code (.). The maximum number of items that can be specified is 128. (When the PV of a timer/counter is specified, however, the status of the Completion Flag is also returned, and must therefore be counted as two items.)

**Batch Reading**

The bit, word, and timer/counter status is read as a batch according to the read information that was registered with QQ.

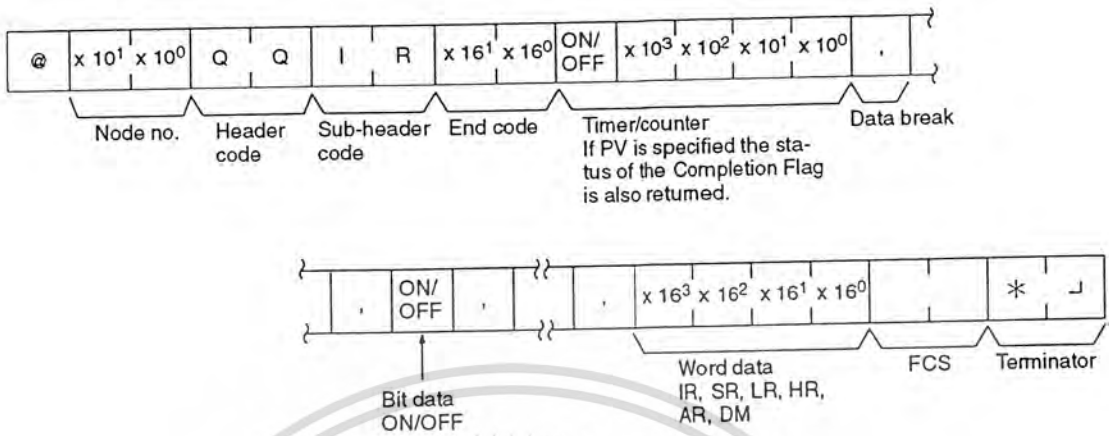
**Command Format**



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่ควรนำไปเผยแพร่ภายนอกการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Response Format

An end code of 00 indicates normal completion.



Parameters

Read Data (Response)

Read data is returned according to the data format and the order in which read information was registered using QQ. If "Completion Flag" has been specified, then bit data (ON or OFF) is returned. If "Word" has been specified, then word data is returned. If "PV" has been specified for timers/counters, however, then the PV is returned following the Completion Flag.

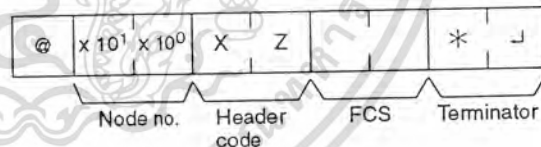
Data Break (Response)

The break code (.) is returned between sections that are read.

6-3-33 ABORT — XZ

Aborts the Host Link operation that is currently being processed, and then enables reception of the next command. The ABORT command does not receive a response.

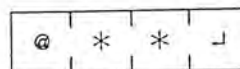
Command Format



6-3-34 INITIALIZE — \*\*

Initializes the transmission control procedure of all the PCs connected to the host computer. The INITIALIZE command does not use node numbers or FCS, and does not receive a response.

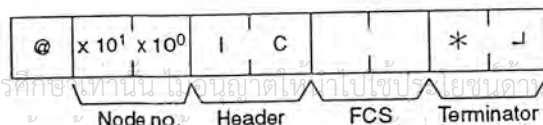
Command Format



6-3-35 Undefined Command — IC

This response is returned if the header code of a command cannot be decoded. Check the header code.

Command Format



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาค้นคว้าเท่านั้น ไม่อนุญาตให้นำไปเผยแพร่โดยไม่ผ่านการคัดค้าน  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## เอกสารอ้างอิง

1. เริ่มต้นอย่างมืออาชีพด้วย Delphi 5 ฉบับสมบูรณ์  
ผู้แต่ง สัจจะ จรัสรุ่งรวีวร ,จักรพงษ์ สุขประเสริฐ  
สำนักพิมพ์ Infopress
2. คู่มือการเขียนเครื่องมือช่วยพัฒนาโปรแกรมด้วยเคลไฟต์  
ผู้แต่ง จิรายุ วิริยะพิบูล  
สำนักพิมพ์ ไทยวัฒนาพานิช
3. คู่มือ Borland Delphi 5 ฉบับสมบูรณ์  
ผู้แต่ง กมลมาศ กำจรกิจการ  
สำนักพิมพ์ Provision
4. Delphi Episode II เทคนิคและการพัฒนาโปรแกรมด้วยเคลไฟต์  
ผู้แต่ง ประพนธ์ อิศวภาณุวัฒน์  
สำนักพิมพ์ ซีเอ็ด
5. ระบบ PLC (Programmable Logic Controller)  
ผู้แต่ง ณรงค์ ตันชีวะวงศ์  
สำนักพิมพ์ สมาคมส่งเสริมเทคโนโลยี (ไทย-ญี่ปุ่น)
6. วิศวกรรมระบบควบคุม Control System Engineer และ ระบบควบคุมป้อนกลับ Feedback Control System  
ผู้แต่ง รศ.กิตติ ศิริมศรชฐ , รศ.วิทยา ทิพย์สุวรรณพร  
สำนักพิมพ์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
7. คู่มือการใช้งาน PLC. Cqmi

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้