สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

# โปรแกรมสนับสนุนการทำงานของระบบพีซีลิงค์
# (PC LINK SUPPORT SOFTWARE)

นาย บัญชา ปะถีละเตสัง
น.ส.ระพีพรรณ ดูอนุพงศ์
น.ส.วิดาวี วงศ์เดโช

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต
สาขาวิศวกรรมการวัดคุม
คณะวิศวกรรมศาสตร์
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
ปีการศึกษา 2541

เลขหมู่.....
เลขทะเบียน 33969
วัน, เดือน, ปี 23 ก.ย. 2542

เอกสารนี้สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่อนุญาตให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ปริญญานิพนธ์ปีการศึกษา 2541

ภาควิชาเทคโนโลยีการวัดคุมทางอุตสาหกรรม

สาขาวิชาวิศวกรรมการวัดคุม

คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เรื่อง โปรแกรมสนับสนุนการทำงานของระบบพีซีลิงค์

PC LINK SUPPORT SOFTWARE


ผู้จัดทำ

นาย บัญชา ปะสีละเตสัง รหัสประจำตัว 38014257

น.ส.ระพีพรรณ คูอนุพงศ์ รหัสประจำตัว 38014411

น.ส.วิภาวี วงศ์เดโช รหัสประจำตัว 38014468


..... อาจารย์ที่ปรึกษา

(อาจารย์ทวีพล ซื่อสัตย์)


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หัวข้อวิทยานิพนธ์ : โปรแกรมสนับสนุนการทำงานของระบบพีซีลิงค์
โดย : นายบัญชา ปะสีละเตสัง
: น.ส.ระพีพรรณ คูอนุพงศ์
: น.ส.วิภาวี วงศ์เดโช
ภาควิชา : เทคโนโลยีการวัดคุมทางอุตสาหกรรม
อาจารย์ที่ปรึกษา : อ. ทวีพล ซื่อสัตย์
ปีการศึกษา : 2541

## บทคัดย่อ

ปริญญานิพนธ์ฉบับนี้นำเสนอการออกแบบโปรแกรมบนเครื่องคอมพิวเตอร์เพื่อสนับสนุนการทำงานของเครื่องควบคุมที่โปรแกรมได้ ( Programmable Controller ) ซึ่งต่อกันในลักษณะโครงข่ายแบบพีซีลิงค์ ( PC Link ) โปรแกรมที่ได้ทำการออกแบบนี้ สามารถตรวจสอบสภาวะ, อ่านและเขียนข้อมูลลงในเครื่องควบคุมแต่ละเครื่องที่เชื่อมต่ออยู่ในระบบ ตลอดจนช่วยให้การติดตั้งระบบการเชื่อมโยงซึ่งยุ่งยากทำได้ง่ายขึ้น โดยที่โปรแกรมนี้ได้พัฒนาบนไมโครซอฟท์ วิชวล เบสิค เวอร์ชั่น 5.0 ( Microsoft Visual Basic 5.0 ) ภายใต้ระบบปฏิบัติการไมโครซอฟท์ วินโดวส์ 95 ( Microsoft Windows 95 ) ซึ่งแสดงผลในรูปแบบของกราฟฟิกอินเตอร์เฟส( Graphic Interface )

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Project Report Title : PC Link Support Software

By : Mr. Banchar Paseelatesang

: Miss Rapeepan Kooanupong

: Miss Viphavee Wongdecho

Dapartment : Industrial Instrumentation Technology

Project Report Advisor : Mr. Taweepol Suesut

Academic : 1998

## Abstract

This thesis presents designing program for supporting the operation of Programmable Controllers ( PCs ) connected in PC Link System. This program can monitor the PC status , read/write data to each PC connected in system, including make it easy to install Link System which is complicated and difficult. This program is developed by using Microsoft Visual Basic 5.0 under Microsoft Windows 95 and displays by graphic interface.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

# สารบัญ


| | หน้า |
|---|---|
| **บทที่ 1 บทนำ** | **1** |
| 1.1 ที่มาของโครงงาน | 1 |
| 1.2 วัตถุประสงค์ของโครงงาน | 1 |
| 1.3 ขอบเขตของโครงงาน | 2 |
| **บทที่ 2 ทฤษฎีและหลักการ** | **3** |
| 2.1 ความรู้เบื้องต้นเกี่ยวกับ PLC | 3 |
| 2.2 การสื่อสารแบบ PC Link | 9 |
| 2.3 การสื่อสารแบบ Host Link | 15 |
| 2.4 ทฤษฎีการทำงาน Visual Basic | 23 |
| **บทที่ 3 การออกแบบโปรแกรม** | **46** |
| 3.1 คุณสมบัติทั่วไปของโปรแกรมสนับสนุนการทำงาน | 46 |
| 3.2 การออกแบบโปรแกรมเพื่อตรวจสอบสถานะการทำงานของเครื่องควบคุม PLC | 46 |
| **บทที่ 4 การใช้งานของโปรแกรม** | **51** |
| **บทที่ 5 บทสรุปและแนวทางการพัฒนาต่อ** | **66** |
| 5.1 แนวทางในการพัฒนาต่อ | 66 |
| 5.2 ข้อจำกัดของโครงงาน | 66 |
| 5.3 บทสรุป | 66 |
| **กิตติกรรมประกาศ** | **67** |
| **บรรณานุกรม** | **68** |



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

# สารบัญภาพ


| | | หน้า |
|---|---|---|
| **ภาพที่ 2.1** | Block Diagram ของ PLC | 4 |
| **ภาพที่ 2.2** | Input / Output Scan และ Program Scan | 5 |
| **ภาพที่ 2.3** | แผนภาพแสดง การทำงาน 1 การสแกนของ หน่วยประมวลผลกลาง (CPU) | 5 |
| **ภาพที่ 2.4** | ช่องต่ออุปกรณ์ภายนอก ( Peripheral Port ) ของ PLC | 8 |
| **ภาพที่ 2.5** | Function ส่วนแสดงผลของ PLC รุ่น SYSMAC C200HS | 12 |
| **ภาพที่ 2.6** | การติดต่อสื่อสารระหว่าง PLC กับ คอมพิวเตอร์ | 15 |
| **ภาพที่ 2.7** | หน้าจอของ Visual Basic | 23 |
| **ภาพที่ 2.8** | วินโดว์ออกแบบเมนูของ Visual Basic | 27 |
| **ภาพที่ 2.9** | แบบฟอร์มพร้อมกับเมนูบาร์และเมนูแบบดึงลง | 28 |
| **ภาพที่ 2.10** | เครื่องมือสร้างเท็กซ์บ็อกซ์ | 29 |
| **ภาพที่ 2.11** | เครื่องมือสร้างเลเบล | 30 |
| **ภาพที่ 2.12** | เครื่องมือสร้างปุ่มคำสั่ง | 31 |
| **ภาพที่ 2.13** | เครื่องมือสร้างเช็คบ็อกซ์ | 32 |
| **ภาพที่ 2.14** | เครื่องมือสร้างออปชั่นบัตทอน | 32 |
| **ภาพที่ 2.15** | เครื่องมือสร้างเฟรม | 33 |
| **ภาพที่ 2.16** | การใช้เฟรมจัดกลุ่มคอนโทรล | 33 |
| **ภาพที่ 2.17** | เครื่องมือสร้างตัวจับเวลา | 34 |
| **ภาพที่ 2.18** | เครื่องมือสร้างคอมโบบ็อกซ์ | 34 |
| **ภาพที่ 2.19** | เครื่องมือสร้างเชพคอนโทรล | 35 |
| **ภาพที่ 2.20** | เครื่องมือสร้างโพรเกรสบาร์ | 36 |
| **ภาพที่ 2.21** | เครื่องมือสร้างไดร์ฟลิสต์บ็อกซ์ | 36 |
| **ภาพที่ 2.22** | เครื่องมือสร้างไดเรกทอรีลิสต์บ็อกซ์ | 37 |
| **ภาพที่ 2.23** | เครื่องมือสร้างไฟล์ลิสต์บ็อกซ์ | 37 |
| **ภาพที่ 2.24** | เครื่องมือสร้างอิมเมจคอนโทรล | 38 |
| **ภาพที่ 2.25** | เครื่องมือสร้างพิกเจอร์บ็อกซ์ | 38 |



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หน้า

**ภาพที่ 2.26** เครื่องมือสร้างแท็บสตริปพ์ 39
**ภาพที่ 2.27** ไดอะล็อกซ์คุณสมบัติ Custom ของแทบสตริปพ์ 39
**ภาพที่ 2.28** เครื่องมือสร้างทูลบาร์ 40
**ภาพที่ 2.29** เครื่องมือสร้างอิมเมจลิสต์ 40
**ภาพที่ 2.30** การใช้ทูลบาร์และอิมเมจลิสต์ 41
**ภาพที่ 2.31** ไดอะล็อกซ์คุณสมบัติ Custom ของทูลบาร์ 41
**ภาพที่ 2.32** ไดอะล็อกซ์คุณสมบัติ Custom ของอิมเมจลิสต์ 43
**ภาพที่ 2.33** MSCOMM CONTROL 44
**ภาพที่ 3.1** แผนผังการสื่อสารข้อมูล 47
**ภาพที่ 3.2** แผนผังการคำนวณ FCS 48
**ภาพที่ 3.3** แผนผังการตรวจสอบสถานะการทำงาน 49
**ภาพที่ 3.4** แผนผังการกำหนดค่าสถานะของ PLC 50
**ภาพที่ 4.1** หน้าจอการตั้งค่า (Setting Part) 51
**ภาพที่ 4.2** การตั้งค่าพารามิเตอร์ในการสื่อสารแบบโฮสลิงค์ 52
**ภาพที่ 4.3** หน้าจอสำหรับเลือกพื้นที่ที่ต้องการ 53
**ภาพที่ 4.4** หน้าจอสำหรับแสดงค่าในพื้นที่ Internal Relay และ Special Relay 54
**ภาพที่ 4.5** หน้าจอสำหรับแสดงค่าในพื้นที่ Link Relay 55
**ภาพที่ 4.6** หน้าจอสำหรับแสดงค่าในพื้นที่ Data Memory 56
**ภาพที่ 4.7** หน้าจอการหาค่าในพื้นที่ Auxiliary Relay 57
**ภาพที่ 4.8** หน้าจอสำหรับแสดงค่าในพื้นที่ Timer และ Counter 58
**ภาพที่ 4.9** การเชื่อมต่อสายในระบบพีซีลิงค์ 59
**ภาพที่ 4.10** หน้าจอสำหรับอ่านค่าในพื้นที่ Data Memory 60
**ภาพที่ 4.11** หน้าจอสำหรับเขียนค่าในพื้นที่ Data Memory 61
**ภาพที่ 4.12** การอ่านค่าในพื้นที่ Link Relay 62
**ภาพที่ 4.13** แสดงการเขียนค่าลงในพื้นที่ Link Relay 63
**ภาพที่ 4.14** หน้าจอหลักของ Help 64

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

# สารบัญตาราง


| | หน้า |
|---|---|
| **ตารางที่ 1** เปรียบเทียบระหว่างระบบซีเควนซ์ ( Sequence ) กับระบบ PLC | 3 |
| **ตารางที่ 2** ประเภทและหน้าที่ของอุปกรณ์ติดต่อภายนอก | 7 |
| **ตารางที่ 3** ข้อกำหนดของเชื่อมโยงข้อมูล ( Data Link Specification ) | 10 |
| **ตารางที่ 4** การกำหนดพื้นที่ในการเชื่อมโยง | 11 |
| **ตารางที่ 5** การกำหนดจำนวน Nodes ในการเชื่อมโยงข้อมูล | 11 |
| **ตารางที่ 6** เลขฐานสิบหกเพื่อนำไปตั้งค่าในพื้นที่ AR 0007 | 12 |
| **ตารางที่ 7** รายละเอียดการกำหนด DIP SWITCH ของ SW1 | 14 |
| **ตารางที่ 8** รายละเอียดการกำหนด DIP SWITCH ของ SW2 | 14 |
| **ตารางที่ 9** ความหมายของ End Code | 16 |



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

# บทที่ 1
# บทนำ

## 1.1 ที่มาของโครงงาน

ระบบโรงงานแบบอัตโนมัติ (Factory Automation System) กำลังเข้ามามีบทบาทในโรงงานอุตสาหกรรมของประเทศมากขึ้น ประกอบกับการแข่งขันในเชิงกลยุทธ์ที่สูงขึ้น เป็นผลให้โรงงานอุตสาหกรรมต้องลดต้นทุนการผลิต และในขณะเดียวกันคุณภาพของผลผลิตต้องเป็นที่ยอมรับได้ของผู้บริโภคหรือมีคุณภาพที่สูงขึ้นด้วย ระบบโรงงานแบบอัตโนมัติเป็นระบบที่อำนวยความสะดวกแก่ผู้ปฏิบัติงานในโรงงานอุตสาหกรรมเป็นอย่างมาก ทำให้สามารถเข้าถึงข้อมูลในกระบวนการ การตรวจสอบคุณภาพ การแก้ไขสถานะ ตลอดจนการส่งผ่านข้อมูลเพื่อทำแผนการผลิตให้เป็นไปอย่างมีประสิทธิภาพ อันเป็นผลให้สูญเสียวัตถุดิบน้อยลง ต้นทุนการผลิตต่ำลง และผลผลิตมีคุณภาพสูงขึ้น

แต่เนื่องจากกระบวนการผลิตในแต่ละโรงงานอาจมีขนาดใหญ่การใช้ PLC เพียงตัวเดียวอาจไม่สามารถควบคุมทั้งโรงงานได้จึงได้มีการนำเอา PLC หลายๆตัวมาเชื่อมต่อกันแลกเปลี่ยนข้อมูลระหว่างกันและสามารถทำงานร่วมกันได้ รวมทั้งปัจจุบันได้มีการนำเอาซอฟท์แวร์มาประยุกต์ใช้กันมากขึ้นเพื่อให้การเชื่อมโยงข้อมูลนั้นง่ายขึ้นเพราะสามารถติดต่อกับ PLC หลายๆตัวได้จากคอมพิวเตอร์เพียงแค่ตัวเดียวจึงทำให้สะดวกในการสร้างศูนย์กลางการควบคุมเพื่อควบคุม PLC ทั้งระบบได้

ภาษาที่เลือกใช้ในการพัฒนาซอฟท์แวร์ในโครงงานนี้ ได้แก่ Microsoft Visual Basic 5.0 เนื่องจากมีการติดต่อกับผู้ใช้ในแบบกราฟฟิก (Graphic User Interface) ทำให้สะดวกในการใช้งานมากกว่ารูปแบบการติดต่อแบบใช้ Text เพียงอย่างเดียว

## 1.2 วัตถุประสงค์ของโครงงาน

1. เพื่อพัฒนาการเชื่อมต่อของ PLC ในรูปแบบของ PC Link ให้ทำงานได้ง่ายขึ้นโดยอาศัยการเขียนโปรแกรมจาก Microsoft Visual Basic บนเครื่องคอมพิวเตอร์

2. เพื่อการติดต่อเชื่อมโยงข้อมูลระหว่าง PLC ที่มีการเชื่อมต่อแบบ PC Link ตลอดจนเป็นแนวทางในการพัฒนาการเชื่อมต่อในระบบ LAN สำหรับ PLC ต่อไป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

**1.3 ขอบเขตของโครงงาน**

ในโครงงานนี้ได้ศึกษาเกี่ยวกับระบบการเชื่อมต่อคอมพิวเตอร์กับ PLC หลาย ๆ ตัวและการใช้ภาษา Microsoft Visual Basic 5.0 เพื่อพัฒนาโปรแกรมสนับสนุนการทำงานบนระบบที่มีการเชื่อมต่อคอมพิวเตอร์กับ PLC หลาย ๆ ตัว โดยโปรแกรมนี้จะช่วยอำนวยความสะดวกแก่ผู้ใช้งานในการตั้งค่าพารามิเตอร์ต่าง ๆ สำหรับการเชื่อมต่อ เช่น การตั้งค่า Baud rate, การเลือกพื้นที่ที่ต้องการเชื่อมโยงข้อมูลระหว่าง PLC แต่ละตัว เป็นต้น นอกจากนี้ยังได้พัฒนาโปรแกรมในส่วนแสดงและแก้ไขค่าสถานะในรูปแบบที่เป็นกราฟฟิก ( Graphic User Interface ) เพื่อให้ผู้ใช้สามารถใช้งานได้ง่ายขึ้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

# บทที่ 2
# ทฤษฎีและหลักการ

## 2.1 ความรู้เบื้องต้นเกี่ยวกับ PLC

### 2.1.1โปรแกรมเมเบิ้ลโลจิก คอนโทรลเลอร์ ( PLC )

PLC เป็นอุปกรณ์ที่ใช้สำหรับควบคุมเครื่องจักรหรือระบบกระบวนการให้ทำงานตามคำสั่งของผู้ใช้ และข้อมูลต่าง ๆที่ได้รับจากหน่วยอินพุท/เอาท์พุทของ PLC การทำงานของ PLC เป็นได้ทั้งการทำงานตามช่วงเวลา ตามลำดับขั้นตอนฟังก์ชั่นทางคณิตศาสตร์และอื่น ๆ

การใช้ PLC สำหรับควบคุมเครื่องจักรหรืออุปกรณ์ต่าง ๆ ในโรงงานอุตสาหกรรมจะมีข้อได้เปรียบกว่าการใช้ระบบของรีเลย์ (Relay) ซึ่งจำเป็นจะต้องเดินสายไฟ หรือที่เรียกว่า Hard-Wired ฉะนั้นเมื่อมีความจำเป็นที่ต้องเปลี่ยนกระบวนการผลิต หรือลำดับการทำงานใหม่ ก็ต้องเดินสายไฟใหม่ ซึ่งเสียและเสียค่าใช้จ่ายสูง แต่เมื่อเปลี่ยนมาใช้ PLC แล้ว การเปลี่ยนกระบวนการผลิตหรือลำดับการทำงานใหม่นั้น ทำได้โดยการเปลี่ยนใหม่เท่านั้น นอกจากนี้แล้ว PLC ยังใช้ระบบโซลิด-สเตท ซึ่งน่าเชื่อถือกว่าระบบเดิม การกินกระแสไฟฟ้าน้อยกว่า และสะดวกกว่าเมื่อต้องการขยายขั้นตอนการทำงานของเครื่องจักร

**ตารางที่ 1** เปรียบเทียบระหว่างระบบซีเควนซ์ ( Sequence ) กับระบบ PLC

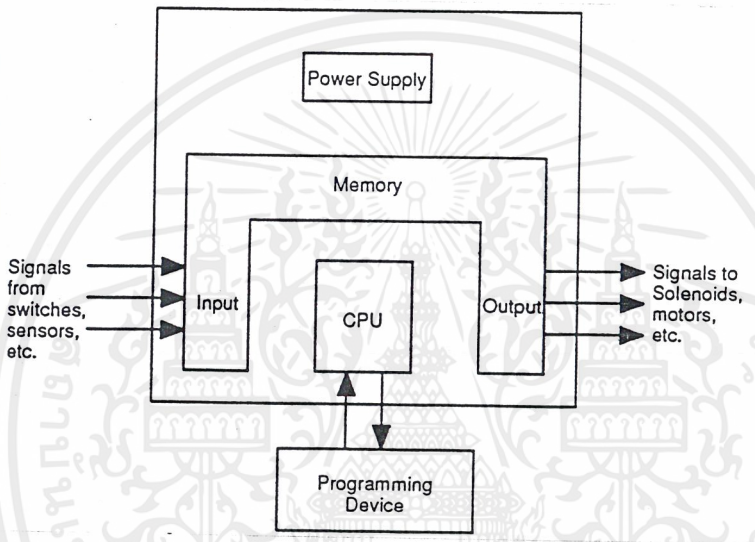
| | ระบบซีเควนซ์ หรือ ใช้การเดินสายไฟ | ระบบโปรแกรมเมเบิล ลอจิก คอนโทรลเลอร์ |
|---|---|---|
| การควบคุมระบบ | ปรับแก้ไขเพิ่มเติมได้ยาก | ปรับแก้ไขเพิ่มเติมได้ง่าย |
| การซ่อม หรือ แก้ไข | ทำได้ยาก | ทำได้ง่าย |
| การติดต่อกับอุปกรณ์ภายนอก | ทำได้ยาก | ทำได้ง่าย |
| อายุการใช้งาน | น้อยกว่า เพราะมีส่วนของการเคลื่อนที่มาก | มากกว่า เพราะส่วนที่เคลื่อนที่มีน้อย |
| ติดต่อกับอุปกรณ์ไกล ๆ | ยุ่งยาก เพราะต้องเดินสายไฟยาว | ง่าย การเดินสายไฟน้อย |
| ความเร็วในการทำงาน | ช้า | เร็ว |
| ขนาด | ใหญ่ | เล็ก |
| ทนต่อสัญญาณรบกวน | ดีมาก | ดี |
| การติดตั้ง | ใช้เวลามาก | ใช้เวลาน้อย |
| การทำงานที่ระบบซับซ้อน | ยาก ใช้รีเลย์จำนวนมาก | ง่าย สะดวก |

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

**2.1.2 ส่วนประกอบของ PLC**

แบ่งได้ 4 ส่วนด้วยกัน คือ

1. ส่วนที่เป็นหน่วยประมวลผลกลาง ( Central Processing Unit : CPU)
2. หน่วยความจำ (Memory Unit)
3. ส่วนที่เป็นอินพุท /เอาท์พุท ( Input /Output : I/O )
4. ส่วนที่เป็นอุปกรณ์การโปรแกรม ( Programming Device)




**ภาพที่ 2.1** Block Diagram ของ PLC

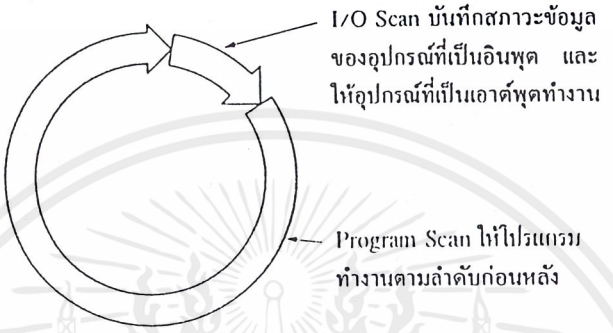
**2.1.3หน่วยประมวลผลกลาง (CPU)**

หน่วยประมวลผลกลาง เป็นส่วนมันสมองของระบบ โดยภายในประกอบด้วยวงจร Logic Gate ชนิดต่างๆหลายชนิด และมี Microprocessor-based ใช้สำหรับแทนอุปกรณ์จำพวกรีเลย์ (Relay) เคาท์เตอร์ (Counter) ไทเมอร์ (Timer) และซีเควนเซอร์ (Sequencer) เพื่อให้ผู้ใช้ได้ออกแบบใช้วงจรรีเลย์ แลดเดอร์ ลอจิก (Relay Ladder Logic) เข้าไปได้

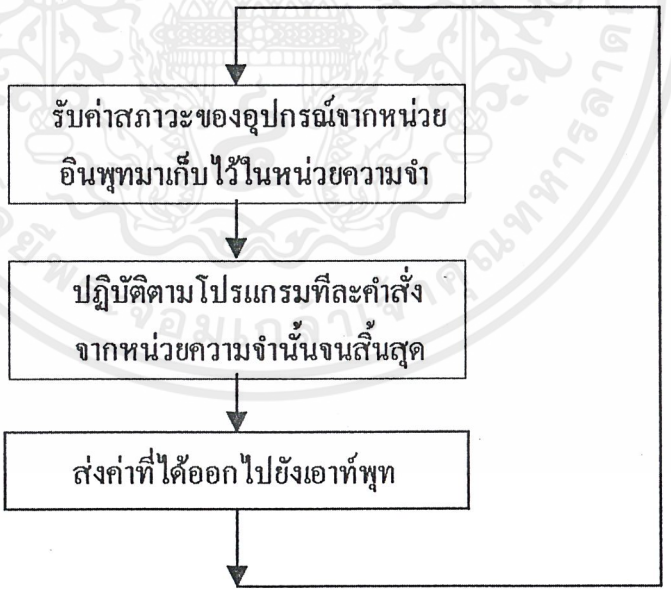
หน่วยประมวลผลกลาง จะรับ (Read) ข้อมูล (Input Data) จากอุปกรณ์ให้สัญญาณ (Sensing Device) ต่าง ๆ จากนั้นจะปฏิบัติการเก็บข้อมูลโดยใช้โปรแกรมจากหน่วยความจำ และส่งข้อมูลที่เหมาะสมถูกต้องไปยังอุปกรณ์ควบคุม (Control Device) แหล่งของกระแสไฟฟ้าตรง (DC Current) สำหรับใช้สร้างแรงดันไฟฟ้าต่ำ ๆ (Low Level Voltage) ซึ่งใช้โดยโปรเซสเซอร์ (Processor) และ I/O Modules ซึ่งแหล่งจ่ายไฟนี้จะเก็บไว้ที่ หน่วยประมวลผลกลาง หรือแยกออกไปติดตั้งที่จุดอื่นก็ได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การประมวลผลของ หน่วยประมวลผลกลาง จากโปรแกรมทำได้โดยรับข้อมูลจากหน่วยอินพุทเข้ามาทำการประมวลผล แล้วส่งข้อมูลที่ได้ออกไปยังเอาท์พุท จากนั้นก็วกกลับไปรับข้อมูลอินพุทเข้ามาอีก การทำในลักษณะนี้เรียกว่า การสแกน (Scan Time)




**ภาพที่ 2.2** Input / Output Scan และ Program Scan




**ภาพที่ 2.3** แผนภาพแสดง การทำงาน 1 การสแกนของ หน่วยประมวลผลกลาง (CPU)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

**2.1.4 หน่วยความจำ (Memory Unit)**

เป็นองค์ประกอบที่สำคัญของระบบ เพราะใช้เป็นที่เก็บโปรแกรมและข้อมูล ขนาดของหน่วยความจำจะเป็นสิ่งกำหนดความสามารถของระบบ ระบบที่มีหน่วยความจำมากจะทำให้ผู้ใช้สามารถเขียนโปรแกรมที่มีความซับซ้อนได้มากขึ้น

หน่วยความจำของ PLC ประกอบด้วย หน่วยความจำชนิด RAM และ ROM หน่วยความจำชนิด RAM ทำหน้าที่เก็บโปรแกรมของผู้ใช้และข้อมูลสำหรับใช้ในการปฏิบัติงานของ PLC ส่วน ROM ทำหน้าที่เก็บโปรแกรมสำหรับใช้ในการปฏิบัติงานของ PLC ตามโปรแกรมของผู้ใช้ ROM ย่อมาจาก Read Only Memory สามารถโปรแกรมได้แต่ลบไม่ได้ ถ้าชำรุดแล้วซ่อมไม่ได้
หน่วยความจำชนิดต่าง ๆ

1. RAM ( Random Access Memory ) หน่วยความจำประเภทนี้จะมีแบตเตอรี่เล็ก ๆ ต่อไว้เพื่อใช้เลี้ยงข้อมูลเมื่อเกิดไฟดับ การอ่านและเขียนโปรแกรมลงใน RAM ทำได้ง่ายมาก จึงเหมาะกับการใช้งานในระยะทดลองเครื่องที่มีการเปลี่ยนแปลงแก้ไขโปรแกรมบ่อย ๆ
2. EPROM ( Erasable Programmable Read Only Memory ) หน่วยความจำชนิด EPROMนี้จะต้องใช้เครื่องมือพิเศษในการเขียนโปรแกรม การลบโปรแกรมทำได้โดยใช้แสงอัลตราไวโอเลตหรือตากแดดร้อนๆ นาน ๆ มีข้อดีตรงที่โปรแกรมจะไม่สูญหายแม้ไฟดับ จึงเหมาะกับการใช้งานที่ไม่ต้องการเปลี่ยนโปรแกรม
3. EEPROM ( Electrical Erasable Programmable Read Only Memory ) หน่วยความจำชนิดนี้ไม่ต้องใช้เครื่องมือพิเศษในการเขียนและลบโปรแกรม โดยใช้วิธีการทางไฟฟ้าเหมือนกับ RAM นอกจากนั้นไม่จำเป็นต้องมีแบตเตอรี่สำรองไฟเมื่อไฟดับ ราคาแพงกว่า แต่จะรวมคุณสมบัติที่ดีของ RAM และ EPROM เอาไว้ด้วยกัน

**2.1.5 ส่วนของอินพุทและเอาท์พุท (I/O UNIT)**

หน่วยอินพุท ทำหน้าที่รับสัญญาณจากอุปกรณ์ภายนอกที่เป็นสวิทช์และตัวตรวจจับชนิดต่าง ๆ (sensor) ของเครื่องจักรหรือกระบวนการ แล้วแปลงเป็นสัญญาณ AC หรือ DC ที่เหมาะสมเพื่อส่งให้แก่หน่วยประมวลผลกลาง

ในส่วนของเอาท์พุท จะทำหน้าที่รับค่าสภาวะที่ได้จากการประมวลผลของหน่วยประมวลผลกลาง แล้วนำค่าที่ได้นี้ไปขยายสัญญาณออกให้มีขนาดใหญ่พอจะขับอุปกรณ์ภายนอก เช่นรีเลย์,

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โซลินอยด์ หรือหลอดไฟ, มอเตอร์,วาล์ว, ปั๊ม เป็นต้น นอกจากนั้นแล้ว ยังทำหน้าที่แยกสัญญาณของหน่วยประมวลผลกลางออกจากอุปกรณ์เอาท์พุทเพื่อป้องกันความเสียหายที่อาจเกิดขึ้นได้

โดยปกติเอาท์พุทนี้จะสามารถขับโหลดได้ด้วยกระแสไฟฟ้าประมาณ 1-2 แอมแปร์ ถ้าโหลดต้องการกระแสไฟฟ้ามากกว่านี้ จะต้องต่อเข้ากับอุปกรณ์ขับ หรือขยายอีกทีหนึ่ง เช่น รีเลย์ โซลิทสเตทรีเลย์ และคอนแทรคเตอร์ เป็นต้น

### 2.1.6 อุปกรณ์ติดต่อภายนอก (Peripheral Device)

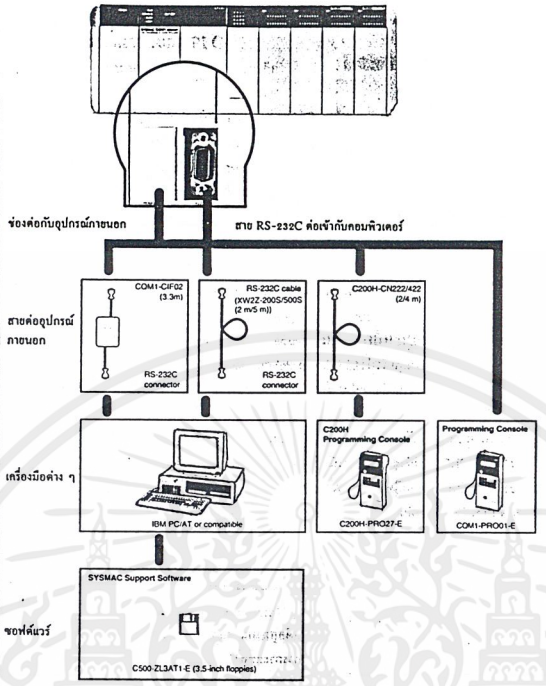
เป็นอุปกรณ์แบบต่าง ๆ ที่อำนวยความสะดวกในการพัฒนาโปรแกรม หน้าที่ของอุปกรณ์ติดต่อภายนอก ได้แก่

1. ป้อนโปรแกรมเข้าไปใน memoryของระบบ
2. ใช้ในการแก้ไข (Debug) โปรแกรม
3. ใช้ในการเก็บรักษาโปรแกรม
4. ใช้ในการพิมพ์โปรแกรม
5. ใช้ในการแสดงสภาวะการควบคุม

**ตารางที่ 2** แสดงประเภทและหน้าที่ของอุปกรณ์ติดต่อภายนอก

| อุปกรณ์ต่อร่วม | หน้าที่การใช้งานเกี่ยวกับโปรแกรม | | | | |
|---|---|---|---|---|---|
| | ป้อน | แก้ไข | โหลดใหม่ | พิมพ์ | สภาวะ |
| 1. Programming Console | * | * | | | * |
| 2. EPROM Writer | | | * | | |
| 3. Printer | | | | * | |
| 4. Graphic Programming | * | * | | | * |
| 5. CRT Monitor | * | * | | | * |
| 6. Audio Cassette | | | * | | |
| 7. Ladder Software | * | * | * | * | * |

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้




**ภาพที่ 2.4** ช่องต่ออุปกรณ์ภายนอก ( Peripheral Port ) ของ PLC

### 2.1.7 ความแตกต่างระหว่างคอมพิวเตอร์กับ PLC

1) PLC ถูกออกแบบให้มีความทนทานต่อสภาพแวดล้อมของโรงงานอุตสาหกรรม เช่น ความร้อน ความเย็น ระบบไฟฟ้ารบกวน การสั่นสะเทือน การกระแทก
2) การใช้โปรแกรมของ PLC จะไม่ยุ่งยากเหมือนของคอมพิวเตอร์ เนื่องจากPLC มีระบบตรวจสอบตัวเอง ทำให้ใช้งานง่าย และบำรุงรักษาง่าย
3) PLC ทำงานตามที่โปรแกรมเอาไว้เพียงโปรแกรมเดียว ทำให้ไม่ยุ่งยาก ส่วนคอมพิวเตอร์จะทำงานที่โปรแกรมหลาย ๆ โปรแกรมพร้อมกัน จึงมีความยุ่งยากมากกว่า
4) PLC ใช้ควบคุมกระบวนการผลิตทุกชนิด ทั้งแบบอนาล็อก และแบบลอจิก ( ON-OFF )

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

**2.2 การสื่อสารแบบ PC Link**

การสื่อสารแบบ PC Link หรือ ที่เรียกกันสำหรับ PLC ของบริษัท OMRON ว่า Sysmac Link เป็นชนิดหนึ่งของการควบคุมการสื่อสารระหว่าง PLC เป็น Communication Path Control System ของ OMRON ซึ่ง communication path จะมีการแลกเปลี่ยนข้อมูลข่าวสารระหว่างกันเป็นจำนวนมาก และมีความเร็วสูงจึงมีประโยชน์อย่างมากต่อการสื่อสารของ PC

การสื่อสารโดยใช้ PC Link System นั้นจะมีประโยชน์ต่อการรับ-ส่งข้อมูล(Data Link Function) แบบ Programmable ระหว่าง PC อิสระ ที่จำเป็นต้องมีการจัดการอย่างมีประสิทธิภาพ การรับส่งข้อมูล (Massage Service Function) สามารถทำได้ขึ้นอยู่กับ User Program ใน System Link Network นั้นจะมี Node ที่เรียกกันว่าเป็น Control Station เพื่อควบคุมเกี่ยวกับการสื่อสารภายใน Network โดยจำเป็นจะต้องมีการติดตั้งไว้ 1 Node ใน 1 Network โดยทั่วไปจะกำหนดให้ Node ที่มี Node Address น้อยที่สุดใน Network นั้นทำหน้าที่เป็น Control Station Node ที่นอกเหนือจาก Control Station ในกรณีที่ Control Station เกิด down ไปด้วยเหตุใดก็ตาม Node ที่มี Node Address น้อยสุดตัวถัดไปจะทำหน้าที่เป็น Control Station โดยอัตโนมัติเพื่อป้องกันมิให้ Network ทั้งหมดต้องหยุดลง (แต่ถ้าหยุดจะ Link ไม่สมบูรณ์)

- Data Link Function

สามารถ Link ข้อมูลระหว่าง PLC บน Network เดียวกันได้ Data Link Function นั้น วิธีการติดต่อสื่อสารโดยใช้ Data Link Area ที่ถูกกำหนดโดยอัตโนมัติโดยขึ้นอยู่กับ Area Network Relay (AR) และวิธีการติดต่อสื่อสารโดยการใช้ Data Link Table ซึ่งถูกกำหนดจาก Tool ซึ่งได้มีการเตรียมไว้แล้ว

- Massage Service Function

ใน Sysmac Link Unit นั้นจะมีคำสั่ง ( Command ) เพื่อตอบสนองการควบคุมทั้งหลาย เช่น คำสั่ง ส่ง/รับ เพื่อเขียนเข้าไปอ่านข้อมูลออกมาจาก PLC Node อื่นๆหรือ PLC ที่ใช้ Sysmac CV Service หรือ FA Computer (LUNA FA และอื่นๆ) ถ้าใช้ฟังก์ชั่นเหล่านี้ Program เพื่อการสื่อสารและการควบคุมจะสามารถใช้ในการสื่อสารที่ซับซ้อนได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

**2.2.1 Data Link Function ของ Sysmac Link Unit**

เป็นฟังก์ชั่นที่เพื่อการแลกเปลี่ยนข้อมูลโดยอัตโนมัติระหว่าง PLC ที่อยู่ใน Network เดียวกันโดยจะต้องกำหนด Data Link Area ซึ่งจะทำให้การแลกเปลี่ยนข้อมูลระหว่างกันนั้นกระทำ (Execute)ได้ใน Data Link Area แต่ละ Node นั้นจะต้องมีการเก็บพื้นที่เพื่อการตอบสนองซึ่งรายละเอียดของ Node นั้นจะต้องสามารถทำการติดต่อสื่อสารได้อย่างอัตโนมัติเช่นเดียวกับ Node ที่กำลังทำการ Link ข้อมูลอยู่

ข้อมูลที่ถูกเขียนเข้าไปใน Area สำหรับ Self-Node จะเข้าไปอยู่ใน Data Link Area ของ PLC นั้นจะถูก report ให้ Node อื่นๆทราบถึงข้อมูลนั้นและการตรวจสอบยืนยันข้อมูลจาก Area สำหรับ Node อื่นๆก็สามารถกระทำได้เช่นเดียวกัน

**ตารางที่ 3** ข้อกำหนดของเชื่อมโยงข้อมูล(Data Link Specification)

<table>
<tr><th>Item</th><th colspan="6">Specification</th></tr>
<tr><td>No. of Data Link Node</td><td colspan="6">Max 62 Nodes : Min 2 Nodes</td></tr>
<tr><td>No. of Area Channel</td><td colspan="6">Max ต่อ 1 Node : LR Area 64 CH<br>DM Area 254 CH</td></tr>
<tr><td>No. of Max Link Channel</td><td colspan="6">2966 CH (รวมทั้ง LR และ DM Area)</td></tr>
<tr><td rowspan="5">การแบ่งส่วนของ Data Link Area</td><td colspan="6">กรณีการกำหนดแบบอัตโนมัติ</td></tr>
<tr><td rowspan="2">Area</td><td rowspan="2">Area Channel No.</td><td colspan="4">หน่วยของการแบ่ง</td></tr>
<tr><td>2</td><td>4</td><td>8</td><td>16</td></tr>
<tr><td>LR</td><td>LR 00-LR 63</td><td>32</td><td>16</td><td>8</td><td>4</td></tr>
<tr><td>DM</td><td>DM0000-DM0127</td><td>64</td><td>32</td><td>16</td><td>8</td></tr>
<tr><td colspan="6">กรณีกำหนดโดยอิสระ กำหนด Area ตาม Tool (FIT และอื่นๆ)</td></tr>
<tr><td>Backup Function ของ Data Link Table</td><td colspan="6">Backup ไว้ใน EEPROM ใน Sysmac Link เฉพาะกรณีกำหนดโดยอิสระเท่านั้น</td></tr>
</table>

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

**2.2.2 SYSMAC LINK System Data Link Settings**

ในการเซตพื้นที่การเชื่อมต่อของ PLC นั้นเราจะทำการเซตพื้นที่ AR 0700 ถึง AR 0703 และ AR 0704 ถึง AR 0707 ให้มีค่าเป็น 1 หรือ 0 ตามต้องการเราสามารถจะเลือกได้ว่าจะให้พื้นที่ใหนที่สามารถเชื่อมโยงข้อมูลถึงกันได้บ้างหรือจะให้เชื่อมโยงข้อมูลกันทั้งพื้นที่ LR (Link Relay: เป็นพื้นที่ในการส่งผ่านข้อมูลข่าวสารกันระหว่าง PLC แต่ละตัว) และ DM (Data Memory:เป็นพื้นที่หน่วยความจำที่ใช้ในการเก็บข้อมูลข่าวสาร)

**ตารางที่ 4** การกำหนดพื้นที่ในการเชื่อมโยง

| Operating Level 0 | | Operating Level 1 | | Setting | |
|---|---|---|---|---|---|
| AR 0700 | AR 0701 | AR 0704 | AR 0705 | | |
| 0 | 0 | 0 | 0 | Words set externally (FIT) | |
| 1 | 0 | 1 | 0 | Automatic allocation | LR Area Only |
| 0 | 1 | 0 | 1 | | DM Area Only |
| 1 | 1 | 1 | 1 | | LR and DM Area |

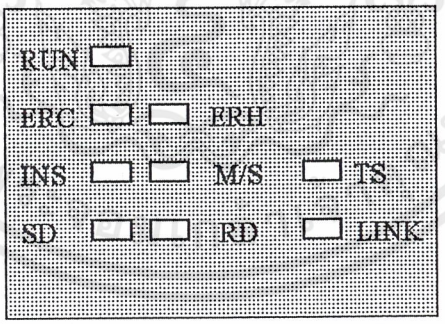
**ตารางที่ 5** การกำหนดจำนวน Nodes ในการเชื่อมโยงข้อมูล

| Operating Level 0 | | Operating level 1 | | Words per node | | Max. No. of Nodes |
|---|---|---|---|---|---|---|
| AR 0702 | AR 0703 | AR 0706 | AR 0707 | LR Area | DM Area | |
| 0 | 0 | 0 | 0 | 4 | 8 | 16 |
| 1 | 0 | 1 | 0 | 8 | 16 | 8 |
| 0 | 1 | 0 | 1 | 16 | 32 | 4 |
| 1 | 1 | 1 | 1 | 32 | 64 | 2 |

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

**ตารางที่ 6** เลขฐานสิบหกเพื่อนำไปตั้งค่าในพื้นที่ AR 0007

| Selecting Areas | Max. No. of Nodes | Hex. For Settings | LR Area CH. | DM Area CH. |
|---|---|---|---|---|
| LR Only | 2 | A0DD | 32 | - |
| | 4 | A099 | 16 | - |
| | 8 | A055 | 8 | - |
| | 16 | A011 | 4 | - |
| DM Only | 2 | A0EE | - | 64 |
| | 4 | A0AA | - | 32 |
| | 8 | A066 | - | 16 |
| | 16 | A022 | - | 8 |
| DM and LR | 2 | A0FF | 32 | 64 |
| | 4 | A0BB | 16 | 32 |
| | 8 | A077 | 8 | 16 |
| | 16 | A033 | 4 | 8 |




**ภาพที่ 2.5** Function ส่วนแสดงผลของ PLC รุ่น SYSMAC C200HS

สภาวะ LED ของส่วนแสดงผลที่อยู่ในสภาวะติด-ดับ หรือ กระพริบจะสามารถบอกถึงสภาพของ Unit ได้ดังนี้คือ

○ หมายถึง ไฟติด

◆ หมายถึง ไฟกระพริบ

☆ หมายถึง ไฟไม่ติด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

**RUN** กำลังทำงาน

○ ทำงานอยู่ในสภาวะปกติ

☆ มีสิ่งผิดปกติในการทำงาน

**ECR** มีสิ่งผิดปกติในการสื่อสาร

○ Communication Controller ผิดปกติการกำหนด node address นั้นมีขอผิดพลาด หรือ อาจเกิดขาดการกำหนด node address ซ้อนกัน

☆ สภาพการสื่อสารที่ไม่มีอะไรผิดปกติ

**INS** กำลัง insert ให้ network

○ กำลัง insert ให้ network

☆ ไม่ได้ Insert ให้ network

**SD** กำลังส่งข้อมูล

○ กำลังส่งข้อมูล

☆ ไม่ได้ส่งข้อมูล

**RD** กำลังรับข้อมูล

○ กำลังรับข้อมูล

☆ ไม่ได้รับข้อมูลในขณะนั้น

**TS** กำลังทดสอบ

○ กำลังทำการทดสอบ

◆ การกำหนดการทดสอบมีสิ่งผิดปกติ

☆ ไม่ได้ทำการทดสอบ

**M/S** Control Station

○ เป็น control station

☆ เป็น station ธรรมดา หรือกำลัง insert ใส่ใน network

**LINK** Data Link

○ กำลังทำการ link ข้อมูล

◆ มีสิ่งผิดปกติในการ link ข้อมูล

☆ ไม่ได้ทำการ link ข้อมูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 2.2.3 การปรับ Node ในการเชื่อมต่อ

ให้ปรับสวิตซ์ที่ Node No. ตามลำดับของ unit ที่เชื่อมต่อ เช่น unit ที่ 1 ก็ให้หมุนสวิตซ์ SW1 เป็น 01 เป็นต้นส่วนตัวอื่นๆก็ทำในลักษณะเดียวกัน ส่วน SW2 ให้ OFF ทั้งหมด

**ตารางที่ 7** รายละเอียดการกำหนด DIP SWITCH ของ SW1

| DIP SWITCH | รายละเอียดของการกำหนด | |
|---|---|---|
| | OFF("0") | ON("1") |
| SW1-4 | สำหรับ Maintenance (ให้กำหนดที่ 0) | |
| SW 1-3 | ให้ SYSMAC LINK UNIT เป็น system # 1 | ให้ SYSMAC LINK UNIT เป็น system #0 |
| SW 1-2 | Data Link Stop | Data Link Start |
| SW 1-1 | Test Stop | Test Start |

**ตารางที่ 8** รายละเอียดการกำหนด DIP SWITCH ของ SW2

| DIP SWITCH | รายละเอียดการกำหนด | |
|---|---|---|
| | OFF("0") | ON("1") |
| SW 2-4 | Control Station Mode(ปกติ) | Ordinary Station Mode |
| SW 2-3 | Network Parameter ไม่เป็น Initial (ปกติ) | ปรับ Network parameter ให้เป็น Initial |
| SW 2-2 | Not Use (กำหนดไว้ที่ OFF) | - |
| SW 1-1 | Not Use (กำหนดไว้ที่ OFF) | - |

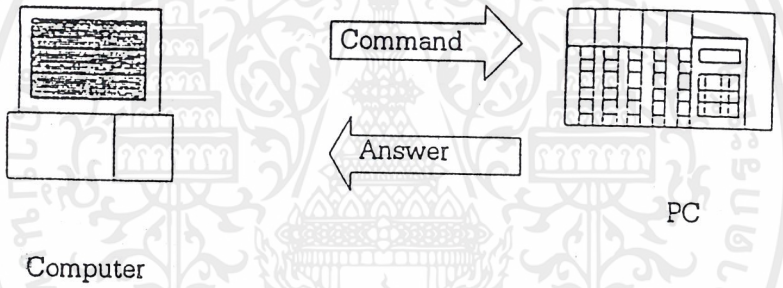
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

**2.3 การสื่อสารแบบ Host Link**

การสื่อสารแบบ Host Link เป็นการเชื่อมต่อ PLC กับ เครื่องคอมพิวเตอร์ทั่วไปผ่านสาย RS-232 เพื่อให้สามารถควบคุม PLC จากคอมพิวเตอร์ได้ สำหรับคอมพิวเตอร์ 1 เครื่องสามารถต่อเข้ากับ PLC ได้จำนวนมาก โดยใช้การเชื่อมต่อ PLC หลาย ๆ ตัวเข้าด้วยกัน เรียก PLC Link

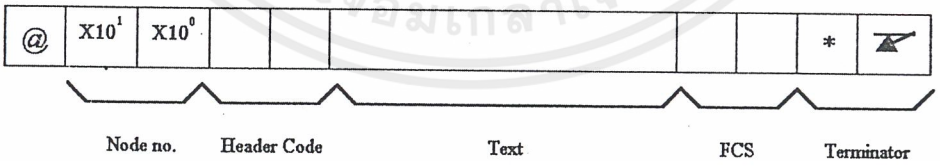
**2.3.1 การติดต่อสื่อสารระหว่าง คอมพิวเตอร์กับ PLC**

การสื่อสารเริ่มจากการส่งข้อมูลในรูปแบบคำสั่งจาก คอมพิวเตอร์ ( Host Computer ) ไปยัง PLC หลังจากที่ PLC ได้รับคำสั่ง ( Command ) จาก Host computer แล้ว PLCจะส่งข้อมูลกลับไปยังคอมพิวเตอร์ในรูปแบบของผลตอบสนอง(Response) ซึ่งจะมีส่วนของ End Code ที่จะใช้ตรวจสอบความผิดพลาดที่อาจเกิดในการส่งคำสั่ง




**ภาพที่ 2.6** การติดต่อสื่อสารระหว่าง PLC กับ คอมพิวเตอร์

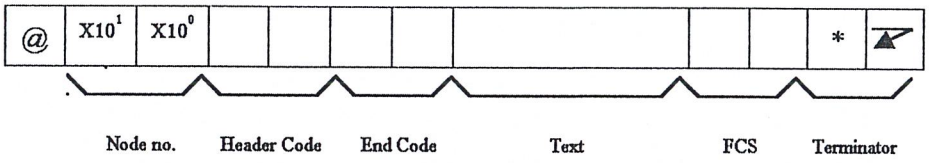
รูปแบบคำสั่ง (Command Format )




| | |
|---|---|
| @ | เป็นสัญลักษณ์ที่แทนการเริ่มต้นคำสั่ง |
| Node no. | หมายเลขเครื่องคอมพิวเตอร์ที่ต้องการติดต่อ เริ่มจาก 00 ถึง 31 |
| Header Code | เป็นส่วนของคำสั่งความยาว 2 ตัวอักษร |
| Text | ค่าพารามิเตอร์ที่จะทำการส่ง |
| FCS | Frame Check Sequence ความยาว 2 ตัวอักษร |
| Terminator | เป็นส่วนแสดงการสิ้นสุดของคำสั่ง |

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปแบบผลตอบสนอง




@ ,Node no. , Header Code จะเหมือนกับคำสั่งที่ส่งไป

End Code (เลขฐานสิบหก) เป็นส่วนที่แสดงสถานะ ปกติจะแสดง “00”
ถ้ามีข้อผิดพลาดจะแสดงเป็น Error code

Text ส่วนนี้จะถูกส่งในกรณีเป็นคำสั่งอ่านข้อมูลเท่านั้น

**ตารางที่ 9** แสดงความหมายของ End Code

| End Code | ความหมายของ End Code | เหตุผลที่เกิดขึ้น | การปฏิบัติงาน (Process) |
|---|---|---|---|
| 00 | ภาวะปกติ | - | - |
| 01 | ไม่สามารถปฏิบัติงานได้ได้ใน Run Mode | คำสั่งที่ถูกส่งไปไม่สามารถปฏิบัติงานได้ใน Run Mode | ตรวจดูความสัมพันธ์ของคำสั่งที่ใช้กับ Mode |
| 02 | ไม่สามารถปฏิบัติงานได้ได้ใน Monitor Mode | คำสั่งที่ถูกส่งไปไม่สามารถปฏิบัติงานได้ใน Monitor Mode | |
| 0B | ไม่สามารถปฏิบัติงานได้ได้ใน Program Mode | คำสั่งที่ถูกส่งไปไม่สามารถปฏิบัติงานได้ใน Program Mode | Codeนี้ไม่สามารถใช้ได้ในขณะนี้ |
| 13 | ความผิดพลาดของ FCS | FCSผิดพลาดเนื่องจากการคำนวณหรือจากอิทธิพลของสัญญาณรบกวน | -ตรวจดูวิธีการคำนวณ<br>-กรณีมีสัญญาณรบกวนให้ส่งคำสั่งอีกครั้ง |
| 14 | เกิดความผิดพลาดในรูปแบบคำสั่ง ( Format Error) | รูปแบบคำสั่งผิดพลาด | ให้ตรวจดูรูปแบบอีกครั้งแล้วส่งคำสั่งอีกครั้ง |
| End Code | ความหมายของ End Code | เหตุผลที่เกิดขึ้น | การปฏิบัติงาน (Process) |
| 15 | เกิดความผิดพลาดในข้อมูลที่ใส่ | การกำหนดพื้นที่สำหรับการอ่านและการเขียนไม่ถูกต้อง | กำหนดพื้นที่ให้ถูกต้องแล้วส่งคำสั่งอีกครั้ง |
| 16 | คำสั่งไม่สมบูรณ์พอ | ไม่มีคำสั่งที่ระบุไว้อยู่จริง | ตรวจเช็ค command code |
| 18 | เกิดความผิดพลาดในความยาวของ Frame | ความยาว Frame มากกว่าความยาวสูงสุดที่กำหนดไว้ | แบ่งFrameนั้นออกเป็นหลาย ๆ Frame |

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง



| | | | |
|---|---|---|---|
| 19 | ไม่สามารถ ปฏิบัติงานได้ | - | - |
| 23 | ในส่วนความจำของผู้ใช้มีการป้องกันการเขียนไว้(write-protected) | Pin1 ที่ Dip Switch ของ C200HS อยู่ในสถานะ ON | ตั้งค่า pin 1 ให้อยู่ในสถานะ OFF |
| A3 | มีการ Abort เนื่องจากความผิดพลาดใน FCS ของข้อมูลที่ส่ง | ความผิดพลาดเกิดขึ้นเพราะความยาวของFrame นั้นมากเกินกว่าจะสามารถปฏิบัติงาน(execute)ได้ในFrame เดียว | ตรวจสอบคำสั่งแล้วทำการส่งอีกครั้ง |
| A4 | มีการ Abort เพราะเกิดการผิดพลาดในรูปแบบของข้อมูลที่ส่ง | | |
| A5 | มีการ Abort เพราะมีการใส่ข้อมูลที่ผิดพลาดในข้อมูลที่ส่ง | | |
| A8 | มีการ Abort เพราะเกิดความผิดพลาดเกี่ยวกับความยาว Frame ของข้อมูลที่ส่ง | | |
| อื่น ๆ | - | เป็นอิทธิพลของสัญญาณรบกวน | ให้ส่งคำสั่งใหม่อีกครั้ง |

**2.3.2 Frame Check Sequence ( FCS )**

FCS เป็นสิ่งที่ใช้สำหรับการตรวจเช็คความผิดพลาดของการสื่อสาร เป็นส่วนที่เกิดจากการเปลี่ยนข้อมูล 8 บิท เป็น 2 ตัวอักษรของข้อมูล ASCII โดยการทำ Exclusive OR ของทุกตัวอักษรจากส่วนของคำสั่งและผลตอบสนอง โดยเริ่มตั้งแต่ เครื่องหมาย " @ " จนถึงตัวอักษรตัวสุดท้ายของ Text

ตัวอย่างการหาค่า FCS

| @10 | RH | 00310001 | 58 | *CR |
|---|---|---|---|---|
| Unit no. | Header | Text | FCS | Terminator |

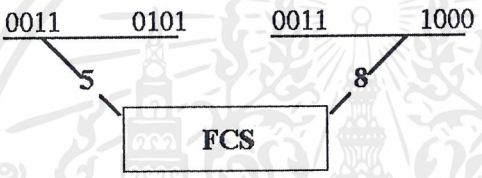
| | | |
|---|---|---|
| @ | 0100 | 0000 |
| | XOR | |
| 1 | 0011 | 0001 |
| | XOR | |
| 0 | 0011 | 0000 |
| | XOR | |

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

33969

| | | |
|---|---|---|
| **R** | 0101 | 0010 |
| .. | ..... | ..... |
| **0** | 0011 | 0000 |
| | XOR | |
| **1** | 0011 | 0001 |
| ผลการคำนวณ | 0101 | 1000 |

แปลงเป็น ASCII




ขั้นตอนการตรวจความผิดพลาดโดยใช้ FCS

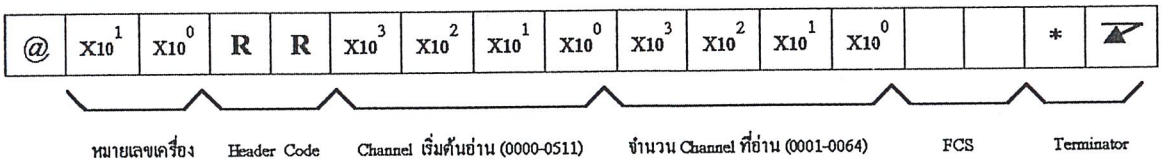
1. FCSที่ได้จากการคำนวณจะถูกส่งพร้อมกับคำสั่งไปยัง PLC
2. PLC จะทำการเปรียบเทียบ FCS ที่ PLC คำนวณได้กับ FCS ที่ถูกส่งมาจากคอมพิวเตอร์ ถ้าตรงกันจะส่งผลตอบสนองเป็น OK และถ้าไม่ตรงกันจะส่งผลตอบสนองเป็น NG

### 2.3.3 Host Link Command

รูปแบบคำสั่งของ Host Link เพื่อใช้ติดต่อกับพื้นที่ต่าง ๆ ของ PLC มีดังนี้

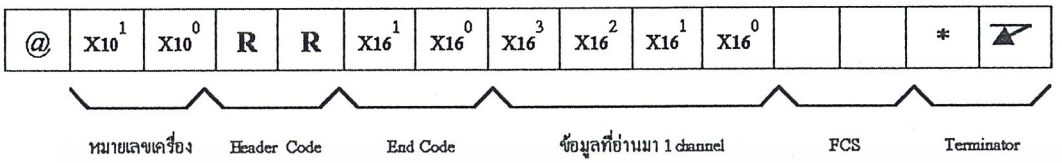
**1) พื้นที่ Internal Relay/special Relay**

รูปแบบ คำสั่งการอ่าน ( Read Command Format)

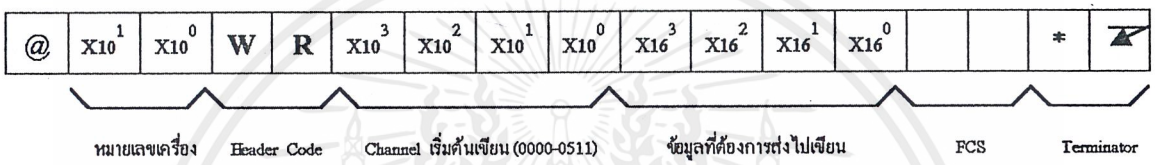



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

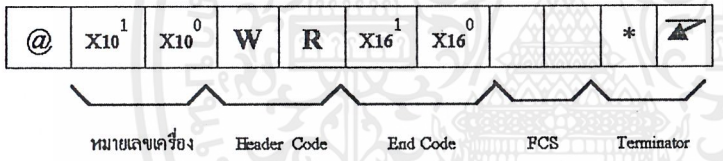
รูปแบบผลตอบสนอง ( Response Format )




รูปแบบคำสั่งการเขียน ( Write Command Format )

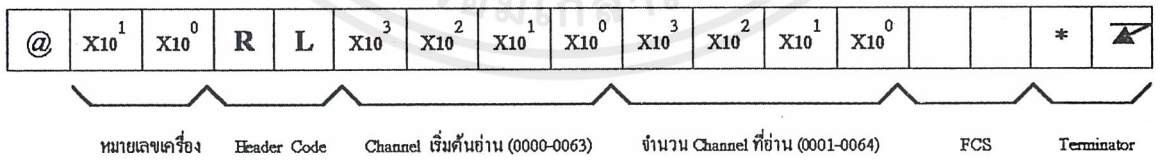



รูปแบบผลตอบสนอง ( Response Format )

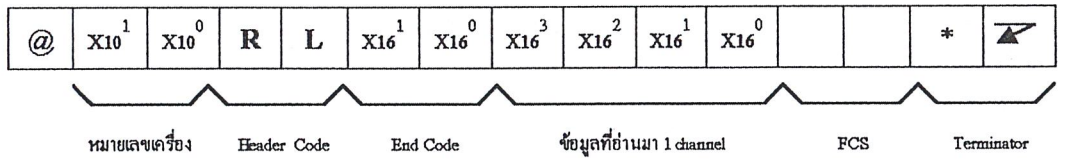



**2) พื้นที่ Link Relay**

รูปแบบ คำสั่งการอ่าน ( Read Command Format)

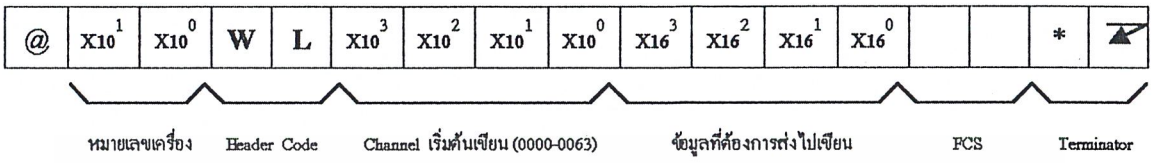



รูปแบบผลตอบสนอง ( Response Format )

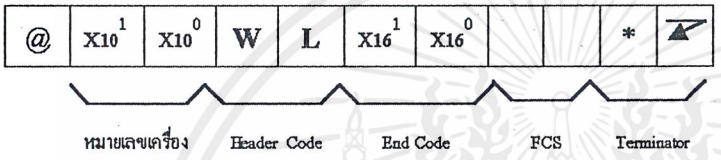



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปแบบคำสั่งการเขียน ( Write Command Format )

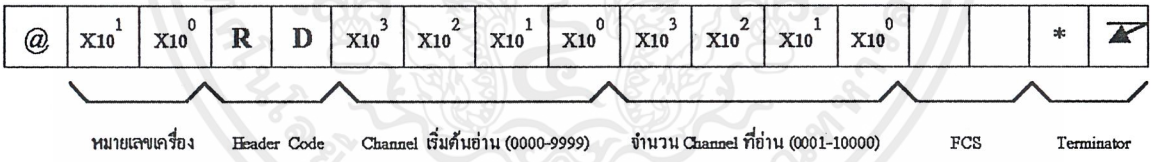



รูปแบบผลตอบสนอง ( Response Format )

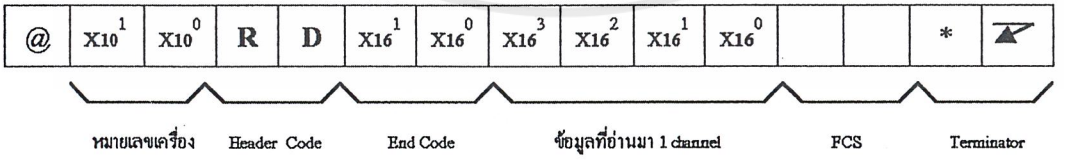



**3) พื้นที่ Data Memory**

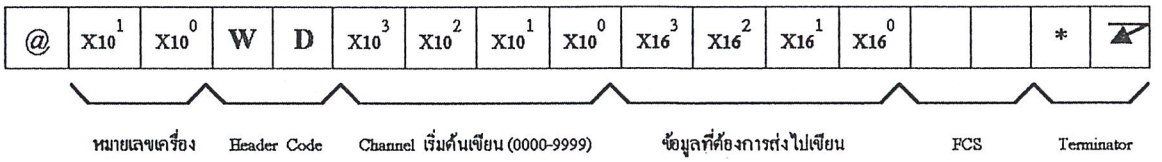
รูปแบบ คำสั่งการอ่าน ( Read Command Format )




รูปแบบผลตอบสนอง ( Response Format )

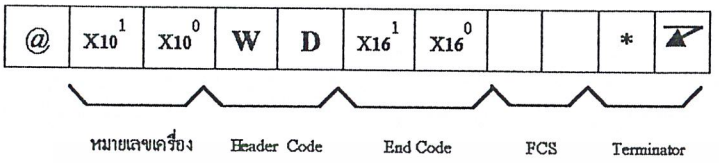



รูปแบบคำสั่งการเขียน ( Write Command Format )



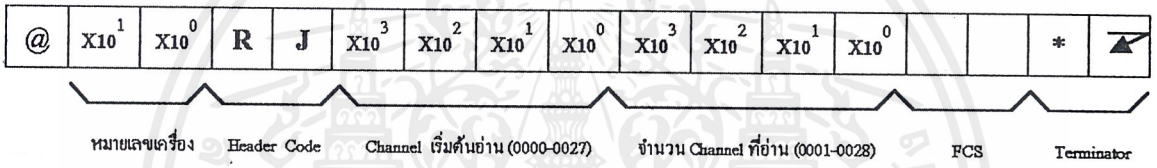

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปแบบผลตอบสนอง ( Response Format )

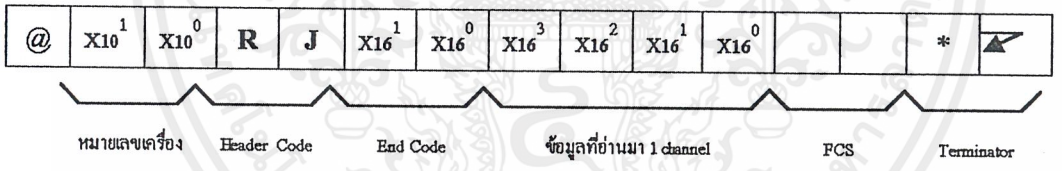



**4) พื้นที่ Auxiliary Relay**

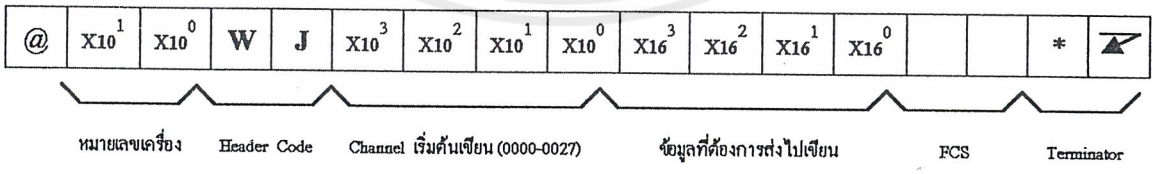
รูปแบบ คำสั่งการอ่าน ( Read Command Format )



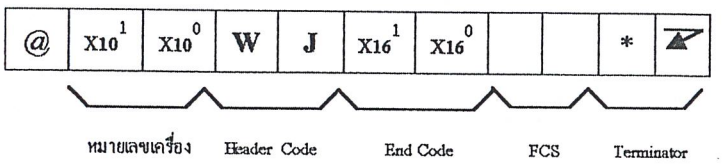

รูปแบบผลตอบสนอง ( Response Format )




รูปแบบคำสั่งการเขียน ( Write Command Format )



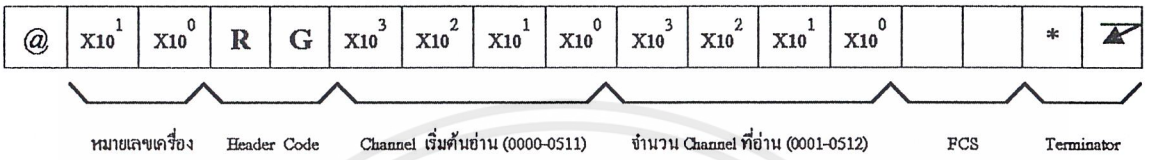

รูปแบบผลตอบสนอง ( Response Format )




เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

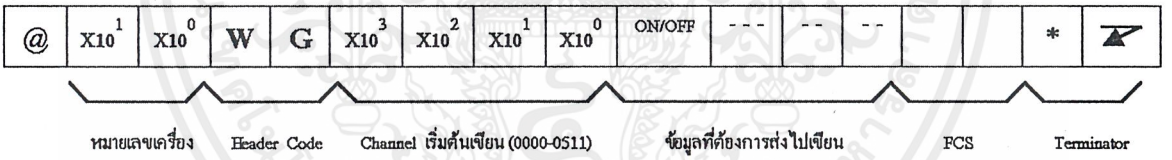
**5) พื้นที่ Timer / Counter**

รูปแบบ คำสั่งการอ่าน ( Read Command Format )

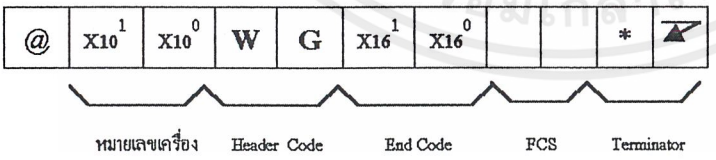



รูปแบบผลตอบสนอง ( Response Format )

รูปแบบคำสั่งการเขียน ( Write Command Format )




รูปแบบผลตอบสนอง ( Response Format )



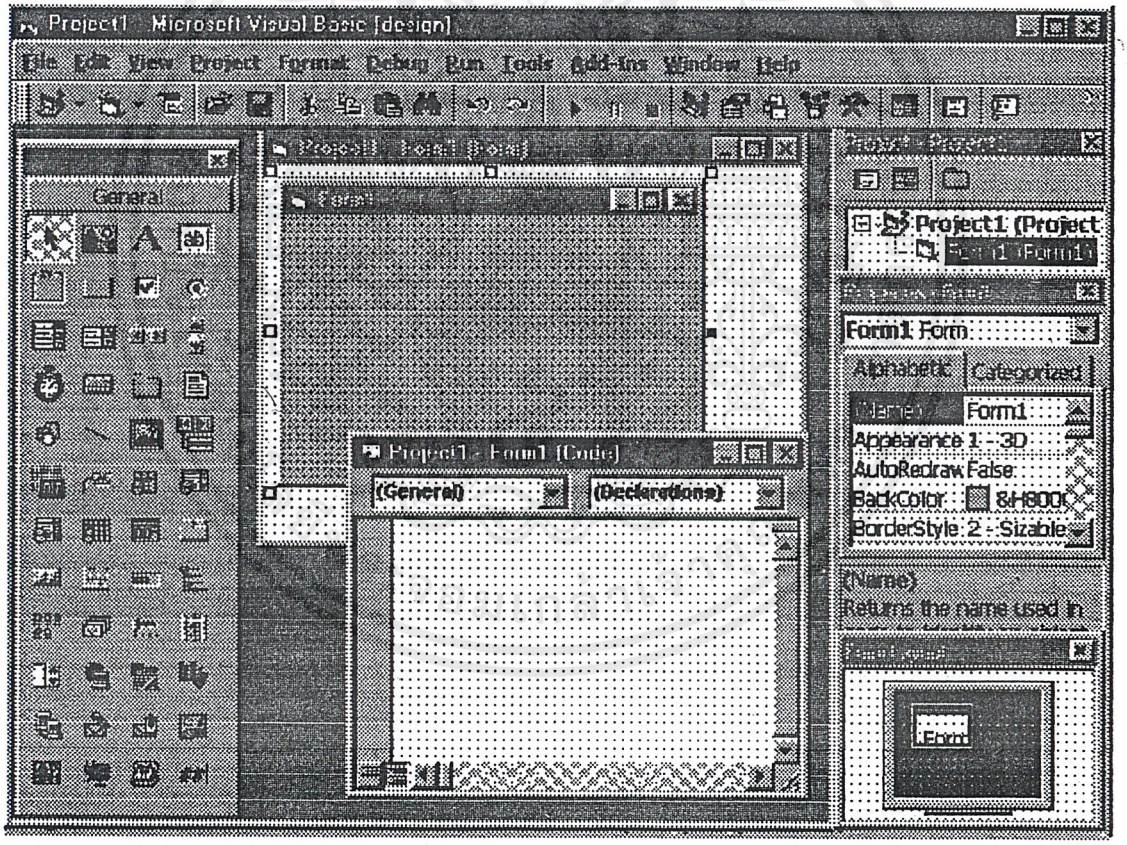

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## 2.4 ทฤษฎีการทำงาน Visual Basic

การพัฒนาแอพพลิเคชั่นใน VB5 จะไม่เหมือนกับการเขียนโปรแกรมที่เราเคยเขียนมาก่อนในภาษาโปรแกรมแบบอื่นๆ บน DOS ที่โปรแกรมจะทำงานตั้งแต่คำสั่งแรกจนถึงคำสั่งสุดท้าย ซึ่งอาจจะมีทางเลือกการทำซ้ำ หรือการเรียกโปรแกรมย่อยมาทำงาน

ใน VB5 จะเป็นการเขียนโปรแกรมแบบ Event-driven การเขียนโปรแกรมแบบนี้เป็นการเขียนโปรแกรมที่ต้องตอบสนองต่อเหตุการณ์ ( Event ) เป็นเหตุการณ์ๆ ไป

### 2.4.1 รายละเอียดของส่วนประกอบต่างๆ ของหน้าจอของ VB




**ภาพที่ 2.7** หน้าจอของ Visual Basic

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เมนูบาร์ ( Menu Bar )

เป็นคำสั่งที่เราสามารถใช้งานได้ทั้งหมดใน VB

ทูลบาร์ ( Toolbar)

เป็นส่วนที่รวบรวมเอาไอคอนเล็กๆ มากมายเอาไว้ ซึ่งแต่ละไอคอน ( Icon )เหล่านี้ก็เปรียบเสมือนคำสั่งหนึ่งๆ ของเมนู นั่นคือไอคอนเหล่านี้ได้ถูกออกแบบมาเพื่อให้การใช้งานคำสั่งต่างๆ ทำได้รวดเร็วและมีลักษณะที่สื่อความหมายกับผู้ใช้มากขึ้น

ทูลบ็อกซ์ ( Toolbox )

เป็นแถบเครื่องมือที่ประกอบไปด้วยไอคอนต่างๆ ซึ่งเรียกว่า " คอนโทรล " ( Control ) โดยจะนำมาใช้ร่วมกับฟอร์มเพื่อสร้างจอภาพของโปรเจ็กต์ แต่ละคอนโทรลจะใช้เป็นเครื่องมือในการสร้างส่วนติดต่อกับผู้ใช้ ( User Interface )โดยการคลิ๊กเลือกที่คอนโทรลนั้นแล้วนำมาวางลงบนฟอร์มด้วยการลากแล้ววาง ( Drag and Drop ) หรือโดยการดับเบิ้ลคลิ๊กที่ไอคอน ในทูลบ็อกซ์

หน้าต่างโปรเจ็กต์( Project Explorer Window )

เป็นหน้าต่างที่รวบรวมรายชื่อของฟอร์มโมดูลไฟล์ ( Custom Control ) โมดูลคลาส ( Class Module ) หรือไฟล์ทรัพยากร ( Resource File ) สำหรับการสร้างแอพพลิเคชั่นหนึ่งๆ ซึ่งการรวมเอาไฟล์เหล่านี้เข้าด้วยกันเพื่อสร้าง แอพพลิเคชั่นภายใต้ VB เรียกว่า โปรเจ็กต์ ( Project )

หน้าต่างคุณสมบัติ ( Properties Window )

เป็นหน้าต่างที่รวบรวมคุณสมบัติทั้งหมดของฟอร์มหรือคอนโทรลเอาไว้ ซึ่งคุณสมบัติทั้งหมดที่ปรากฏในหน้าต่างนี้จะเป็นคุณสมบัติที่ผู้อ่านสามารถกำหนดค่าได้ในขณะกำลังออกแบบ เมื่อผู้อ่านทำการแก้ไขค่าคุณสมบัติต่างๆ ในหน้าต่างคุณสมบัตินี้ก็จะส่งผลต่อคอนโทรลตัวนั้นทันที ซึ่งบางคุณสมบัติสามารถแสดงให้เห็นการเปลี่ยนแปลงคุณลักษณะได้ทันที ส่วนคุณสมบัติบางอย่างจะแสดงผลให้เห็นก็ต่อเมื่อผู้อ่านมีการรันแอพพลิเคชั่นเท่านั้น

ฟอร์ม ( Form )

เป็นหน้าต่างที่ใช้ในการออกแบบส่วนติดต่อกับผู้ใช้ของแอพพลิเคชั่นของเรา เนื่องจาก VB ได้รับการออกแบบให้ผู้ใช้นำมาออกแบบแอพพลิเคชั่นในลักษณะของการสื่อสารด้วยรูป ( Visual ) ดั้งนั้นการสื่อการทำงานต่างๆ ระหว่างแอพพลิเคชั่นกับผู้ใช้จะต้องกระทำผ่านฟอร์ม โดยในแอพพลิเคชั่นหนึ่งสามารถมีได้หลายฟอร์ม และภายในฟอร์มก็จะถูกใช้ในการบรรจุคอนโทรลต่างๆ ฟอร์มจึงทำหน้าที่เป็นตัวบรรจุ ( Container ) และฟอร์มยังเป็นออบเจ็กต์ ( Object ) ตัวหนึ่งของ VB

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ที่อนุญาตให้ ผู้ใช้แก้ไขคุณสมบัติได้ในขณะออกแบบจากหน้าต่างคุณสมบัติและสามารถควบคุมพฤติกรรมต่างๆ ของฟอร์มด้วยวิธีเดียวกับออบเจ็กต์อื่นๆ

หน้าต่างโค้ดอิดิเตอร์ ( Code Editor Window )

เป็นหน้าต่างที่ใช้ในการพิมพ์คำสั่งโปรแกรมเข้าไปในแอพพลิเคชั่นของเรา

หน้าต่างฟอร์มเลย์เอาท์ ( Form Layout Window )

เป็นหน้าต่างแสดงตำแหน่งฟอร์มบนจอ ช่วยให้เรากำหนดตำแหน่งของฟอร์มบนหน้าจอได้สะดวกขึ้น โดยการคลิ้กเมาส์ ลากฟอร์มที่แสดงบนหน้าต่างให้เคลื่อนไปยังตำแหน่งที่ต้องการ

**คุณสมบัติ ( Property )**

เป็นสิ่งที่ใช้บรรยายลักษณะต่างๆ ของคอนโทรล ทำให้คอนโทรลสามารถใช้งานที่หลายรูปแบบมากขึ้น และมีลักษณะแปลกตาไม่ซ้ำกัน การกำหนดค่าคุณสมบัติอาจทำในขณะออกแบบผ่านทางหน้าต่างคุณสมบัติ หรือกำหนดขณะรันแอพพลิเคชั่นโดยเขียนคำสั่งโปรแกรมให้ตั้งคุณสมบัติขณะโปรแกรมทำงาน

**เมดออด ( Method )**

เป็นโปรแกรมย่อยประเภทหนึ่ง ซึ่งเป็นสมาชิกของคอนโทรลนั้นๆ เหมือนกับคุณสมบัติของคอนโทรล เมื่อเรียกใช้เมดถอดจะเป็นการสั่งให้คอนโทรลทำงานให้ ซึ่งอาจจะมีการเปลี่ยนแปลงค่าคุณสมบัติต่างๆ ของคอนโทรลนั้น

**อีเว็นต์ ( Event )**

เป็นการตอบสนองต่อเหตุการณ์ภายนอกของคอนโทรล และเป็นโปรแกรมย่อยที่ทำงานทันทีที่เกิดเหตุการณ์นั้นขึ้นมา

### 2.4.2 คอนโทรลของ VB

คอนโทรลเป็นเครื่องมืออย่างหนึ่งที่ช่วยให้การสร้างแอพพลิเคชั่นด้วย VB ทำได้ง่ายและรวดเร็วยิ่งขึ้น แต่เมื่อได้นำไปวางบนฟอร์มแล้ว จะเรียกว่าออบเจ็กต์ ( Object ) ซึ่งจริงๆแล้วคอนโทรลก็คือไฟล์ไดนามิกลิงค์ไลบรารีทั่วๆไปของวินโดว์ ไฟล์เหล่านี้จะมีนามสกุล .VBX หรือ .OCX ซึ่งเราเรียกว่า Custom Control ซึ่งไฟล์เหล่านี้จะถูกจัดเก็บในไดเร็คทอรี่ย่อย System ของ Window

คอนโทรลพื้นฐานใน VB จะเป็นคอนโทรลติดต่อกับผู้ใช้ในการทำงานพื้นฐานต่างๆ เช่น การแสดงข้อความและรับข้อความจากผู้ใช้ การแสดงรูปภาพ การแสดงตัวเลือก เป็นต้น ซึ่งจะกล่าวถึงเฉพาะคอนโทรลที่ได้นำมาใช้งานดังต่อไปนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

**ฟอร์ม ( Form )**

ฟอร์มเป็นส่วนประกอบพื้นฐานในการติดต่อกับผู้ใช้ ที่ใช้แสดงผลการทำงาน รวมทั้งรับอินพุทต่างๆจากผู้ใช้

คุณสมบัติที่สำคัญของฟอร์ม

AutoRedraw เป็นคุณสมบัติที่บอกว่าต้องมีการวาดสิ่งที่อยู่บนฟอร์มนั้นใหม่หรือไม่ เมื่อจำเป็นต้องมีการวาดใหม่ เช่น มีหน้าต่างอื่นมาบัง เป็นต้น ถ้าเป็น True เมื่อจำเป็นจะต้องมีการปรับปรุงหน้าจอ VB จะจัดการเรื่องนี้เองโดยอัตโนมัติ แต่ถ้าเป็น False เราจำเป็นต้องจัดการเอง

BackColor ใช้กำหนดสีส่วนหลัง ( Background ) ของฟอร์ม

Caption ใช้กำหนดข้อความที่แสดงบนไตเติ้ลบาร์ของฟอร์ม

ControlBox เป็นคุณสมบัติที่บอกว่าให้มีเมนูที่ใช้ย่อ ขยาย หรือปิดหน้าต่างอยู่ที่มุมบนด้านซ้ายของหน้าต่างหรือไม่ ถ้าเป็น True จะมีการแสดงปุ่มเมนู ถ้าเป็น False จะไม่มีการแสดงปุ่มเมนู

ForeColor เป็นคุณสมบัติที่ใช้กำหนดสีของข้อความหรือกราฟฟิกบนฟอร์ม

Height,Width เป็นคุณสมบัติที่ใช้กำหนดความสูงและความกว้างของฟอร์มซึ่งวัดอยู่ในหน่วยที่เรียกว่า ทวิพ ( 1 ทวิพ = 1/1440 นิ้ว )

MaxButton,Minbutton เป็นคุณสมบัติที่กำหนดว่าจะให้มีปุ่มย่อขยายฟอร์มที่มุมบนขวาหรือไม่ซึ่งจะมีผลในขณะรันเท่านั้น

Name เป็นคุณสมบัติที่ใช้กำหนดชื่อของฟอร์มในรหัสโปรแกรม ซึ่งเปลี่ยนแปลงได้เฉพาะในขณะออกแบบเท่านั้น

Picture เป็นคุณสมบัติที่ใช้แสดงภาพบนหน้าต่างเสมอ ในการเปลี่ยนแปลงคุณสมบัติในขณะรันต้องใช้ฟังก์ชัน LoadPicture

Visible เป็นคุณสมบัติที่กำหนดว่าจะให้เห็นฟอร์มหรือไม่ในขณะรัน

WindowState เป็นคุณสมบัติที่กำหนดว่าจะให้แสดงฟอร์มในขณะรันในสถานะปกติ ( ค่ากำหนดเป็น 0 ) ในสถานะลดขนาดเป็นไอคอน ( ค่ากำหนดเป็น 1 ) หรือในสถานะขยายขนาด ( ค่ากำหนดเป็น 2 )

อีเว็นต์ ( Event )

Click เป็นอีเว็นต์ที่เกิดขึ้นเมื่อมีการคลิ้กเมาส์บนฟอร์ม โดยถ้าผู้ใช้คลิ้กเมาส์บนออบเจ็กต์ที่บรรจุอยู่ในฟอร์ม อีเว็นต์ Click ก็จะไม่ถูกส่งผ่านไปให้แก่ฟอร์ม

Load,Unload อีเว็นต์ Load จะเกิดขึ้นเมื่อฟอร์มนั้นโหลดขึ้นมา โดยมากอีเว็นต์นี้จะมีคำ-สั่งเกี่ยวกับการเริ่มต้นต่างๆของคอนโทรล และตัวแปรในโมดูลฟอร์มนี้ ส่วน Unload จะเป็นอีเว็นต์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ที่เกิดขึ้นยกเลิกฟอร์มไปแล้ว

เมดถอด ( Method )

Cls เป็นเมดถอดที่ใช้เคลียร์สิ่งที่วาดบนฟอร์ม

Hide เป็นเมดถอดที่ใช้ซ่อนฟอร์มไว้ชั่วคราว

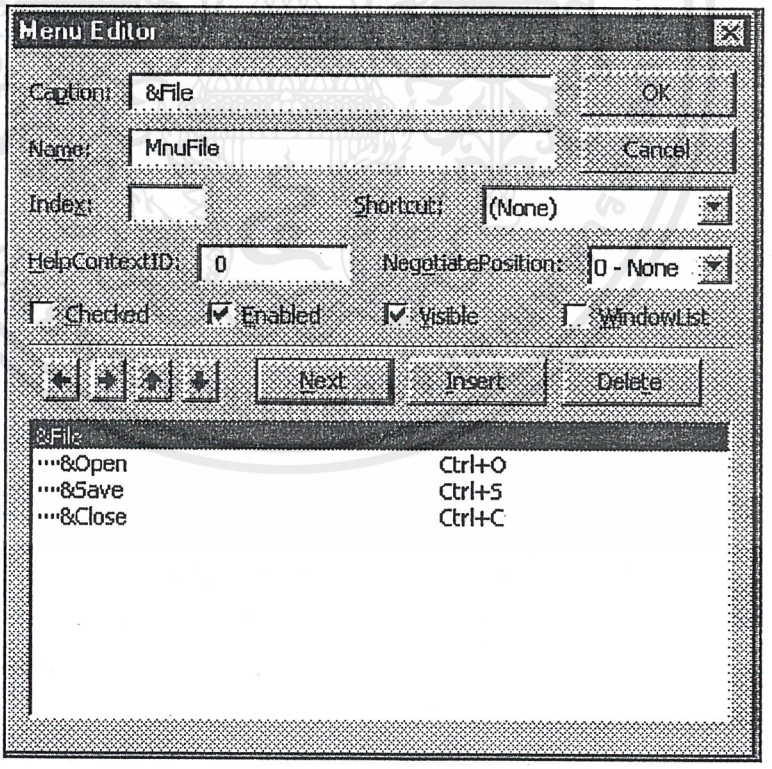
Show เป็นเมดถอดที่ใช้ในการแสดงฟอร์มที่ต้องการ

LoadPicture เป็นเมดถอดที่ใช้ตั้งคุณสมบัติ Picture ของฟอร์ม

SetFocus เป็นเมดถอดที่ใช้ในการโฟกัสมายังฟอร์มนี้เพื่อให้ผู้ใช้สามารถทำงานด้วยได้

**เมนู (MANU)**

แม้ว่าในตอนนี้คุณได้สร้างแอพพลิเคชั่นขนาดเล็กหลายๆโปรแกรมขึ้น แต่คุณก็อาจรู้สึกว่าไม่มีโปรแกรมใดที่ดูเหมือน " แอพพลิเคชั่นจริงๆ " เหตุใดจึงเป็นเช่นนี้ สาเหตุก็มาจากการที่แอพพลิเคชั่นที่สร้างขึ้นนั้นไม่มีเมนูบาร์ อย่างไรก็ตาม คุณสามารถแก้ไขสถานการณ์นี้ได้เพราะ Visual Basic ให้แนวทางที่ใช้สร้างเมนูต่างๆได้โดยง่ายดาย เมนูดำเนินการด้วยแนวทางแบบอาศัยอีเว็นต์คล้ายกับออบเจ็กต์ที่เราพิจารณามาแล้ว

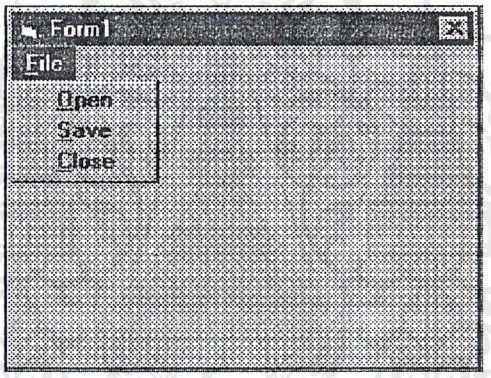



**ภาพที่ 2.8** วินโดว์ออกแบบเมนูของ Visual Basic

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ในการที่จะสร้างเมนูสำหรับเมนูสำหรับแอพพลิเคชั่นหนึ่งๆ คุณจะต้องใช้วินโดว์ Menu Design ของ Visual Basic ตามที่แสดงอยู่ในรูป สร้างโครงการใหม่ขึ้นหนึ่งโครงการ และเปิดวินโดว์นี้โดยการเลือกคำสั่ง Menu Design จากเมนู Windows หรือโดยการกดปุ่มเมาส์บนปุ่ม Menu Design Windows บนทูลบาร์ของ Visual Basic

คุณสร้างเมนูบาร์ของแอพพลิเคชั่นหนึ่งๆ ด้วยการสร้างลำดับชั้นของชิ้นข้อมูลในวินโดว์ Menu Design สำหรับชิ้นเมนูแต่ละชิ้น คุณจะต้องนิยามคุณสมบัติ Caption และคุณสมบัติ Name ข้อความจ่าหน้าเป็นข้อความที่จะปรากฏขึ้นในเมนูบาร์หรือบนเมนูแบบดึงลง ซึ่งของชิ้นเมนูจะระบุชิ้น เมนูนั้นไว้ในรหัสโปรแกรมของคุณ การที่จำเป็นต้องใช้ชื่อก็เพราะว่าชิ้นเมนูแต่ละชิ้นจะมีกระบวนงาน Click




**ภาพที่ 2.9** แบบฟอร์มพร้อมกับเมนูบาร์และเมนูแบบดึงลง

ถ้าคุณกดปุ่มเมาส์บนหน่วยเมนูในแบบฟอร์ม Visual Basic จะเปิดวินโดว์รหัสสำหรับกระบวนงาน Click ที่สอดคล้องกัน ในวินโดว์นี้ คุณเขียนรหัสสำหรับการดำเนินการที่จะเกิดขึ้นเมื่อผู้ใช้เลือกชิ้นเมนู เช่น กดปุ่มเมาส์บนชิ้นเมนู Exit ของเมนู File เพื่อเปิดวินโดว์รหัสสำหรับกระบวนการงานงานอีเว้นต์ mnuFileExit_Click คำสั่งของ Visual Basic ที่ใช้ยุติแอพพลิเคชั่น คือ End ดังนั้นกระบวนการงานสำหรับชิ้นเมนู Exit ควรจะเป็นดังนี้

```
Sub mnuFileExit_Click()
        End
End Sub
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บ็อกซ์ใส่ข้อคำสั่ง Debug.Pint ในกระบวนงานเมนูต่างๆ และหลังจากนี้ดำเนินงานแอพพลิเคชั่นอีกครั้ง ในขณะที่เลือกชิ้นเมนูต่างๆ นั้นให้จับตาดูที่วินโดว์ เพื่อเป็นการยืนยันว่ากำลังเอ็กซีคิวต์กระบวนงานเมนูนั้นอยู่

**เท็กซ์บ็อกซ์( Text Box )**

เป็นคอนโทรลที่ผู้ใช้สามารถเติม แก้ไขข้อความในเท็กซ์บ็อกซ์นี้ได้โดยผ่านคีย์บอร์ดเข้าสู่โปรแกรม หรือแสดงข้อมูลต่างๆในโปรแกรมของเราได้

**ภาพที่ 2.10** เครื่องมือสร้างเท็กซ์บ็อกซ์

คุณสมบัติ

Alignment เป็นคุณสมบัติที่กำหนดรูปแบบของข้อความที่แสดงในคอนโทรลว่าจะให้ข้อความอยู่ชิดซ้าย ( LeftJustify ) ,ชิดขวา( RightJustify ) หรืออยู่ตรงกลาง ( Center )โดยคุณสมบัตินี้จะใช้ได้ก็ต่อเมื่อคณสมบัติ Multiline มีค่าเป็น True

FontBold,FontItalic,FontName,FontSize,FontUnderline เป็นคุณสมบัติที่กำหนดรูปแบบของข้อความที่แสดงในคอนโทรลว่าจะใช้ตัวอักษรแบบใด ( FontName ) ขนาดเท่าใด ( FontSize ) เป็นตัวหนา ( FontBold ) ตัวเอียง ( FontItalic) หรือขีดเส้นใต้ (FontUnderline)

Maxlength เป็นคุณสมบัติที่กำหนดจำนวนอักขระสูงสุดของข้อความ โดยถ้ากำหนดเป็น 0 จะรับข้อความได้ไม่จำกัด

Multiline เป็นคุณสมบัติที่กำหนดให้เท็กซ์บ็อกซ์แสดงข้อความในรูปแบบหลายบรรทัดได้

ScrollBars เป็นคุณสมบัติที่กำหนดรูปแบบของข้อความที่แสดงในคอนโทรล ว่าจะให้แสดงสกรอลบาร์ด้วยหรือไม่อย่างไร เมื่อข้อความยาวเกินความกว้างหรือความยาวของคอนโทรล

SelLength เป็นคุณสมบัติที่บอกว่าความยาวของข้อความที่เลือกอยู่ในขณะนั้นมีค่าเท่าไร

SelStart เป็นคุณสมบัติที่บอกว่าตำแหน่งแรกของข้อความที่เลือกอยู่ในขณะนั้นอยู่ในตำแหน่งที่เท่าไร ถ้าไม่มีการเลือกข้อความอยู่ จะบอกตำแหน่งที่จะแทรกข้อความอยู่ในขณะนั้น

SelText เป็นคุณสมบัติที่บอกว่าข้อความส่วนที่เลือกอยู่ในขณะนั้นคืออะไร

Text เป็นคุณสมบัติที่เก็บข้อความของเท็กซ์บ็อกซ์ในปัจจุบันไว้ และผู้ใช้ยังสามารถโปรแกรมดัดแปลงแก้ไขข้อความที่แสดงออกมาได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

อีเว็นต์

Change เป็นอีเว็นต์ที่เกิดขึ้นเมื่อมีการเปลี่ยนแปลงคุณสมบัติ Text ของ เท็กซ์บ็อกซ์ ไม่ว่าจะเป็นจากผู้ใช้ป้อนข้อความเข้าไปใหม่ หรือ จากการตั้งค่าคุณสมบัติ Text ใหม่ ในโปรแกรม

LostFocus เป็นอีเว็นต์ที่เกิดขึ้นเมื่อผู้ใช้ตั้งระยะเคลื่อนออกไปจากเท็กซ์บ็อกซ์หรือใช้เมาส์เลือกออบเจ็กต์อื่นๆ

เมดถอด

SetFocus เป็นวิธีการที่มีประโยชน์ที่สุดในการชี้ความสนใจไปยังเท็กซ์บ็อกซ์ที่สำคัญที่สุดในหลายๆเท็กซ์บ็อกซ์ โดยเคอร์เซอร์จะไปอยู่ในเท็กซ์บ็อกซ์ที่เรากำหนด

**เลเบล ( Label )**

เป็นคอนโทรลที่ใช้ในการแสดงข้อความตามที่เราต้องการ เพื่อแสดงข้อมูลบางอย่างแก่ผู้ใช้ โดยผู้ใช้ไม่สามารถแก้ไขข้อความนั้นได้ในตอนรันโปรแกรม การแก้ไขข้อความในเลเบลจะต้องทำในขณะออกแบบ หรือในคำสั่งโปรแกรมตอนรันเท่านั้น

**ภาพที่ 2.11** เครื่องมือสร้างเลเบล

คุณสมบัติ

คุณสมบัติดังที่ได้กล่าวไปแล้วที่มีในเลเบลได้แก่ Alignment, BackColor, FontBold, FontItalic, FontName, FontSize, FontUnderline, Name, Visible คุณสมบัติอื่นๆ ได้แก่

Autosize เป็นคุณสมบัติที่กำหนดให้คอนโทรลปรับขนาดตามข้อความที่แสดง

Caption เป็นคุณสมบัติที่บอกถึงข้อความที่แสดงบนคอนโทรลในขณะนั้น

WordWarp เป็นคุณสมบัติที่กำหนดให้ข้อความขึ้นบรรทัดใหม่ เมื่อยาวเกินความยาวของคอนโทรล

Enabled ถ้ามีค่าเป็น True จะใช้งานตามปกติ แต่ถ้าเป็น False ข้อความจะเป็นสีเทาและอีเว็นต์ของเมาส์จะใช้งานบนคอนโทรลนี้ไม่ได้

อีเว็นต์

อีเว็นต์ของเลเบลจะเป็น Click และ DblClick เช่นเดียวกับที่ฟอร์มทำได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

**ปุ่มคำสั่ง ( Command Button )**

เป็นคอนโทรล ที่ใช้ในการทำงานอย่างหนึ่งตามที่เราต้องการ โดยเราสามารถกำหนดสิ่งที่เราต้องการให้ทำงานในอีเว็นต์ Click

**ภาพที่ 2.12** เครื่องมือสร้างปุ่มคำสั่ง

การกำหนดข้อความที่ให้แสดงบนคอนโทรล

ในการกำหนดข้อความที่ต้องการให้แสดงผลบนคอนโทรล ให้เรากำหนดที่คุณสมบัติ Caption ของคอนโทรลในหน้าต่างคุณสมบัติ หรือในคำสั่งโปรแกรมก็ได้

คุณสมบัติ

Default เป็นคุณสมบัติที่บอกว่าเมื่อกดปุ่ม < Enter > จะเป็นการบอกว่าทำงานเหมือนกับการคลิ้กปุ่มคำสั่งนี้

Cancel จะเป็นคุณสมบัติบอกว่าเมื่อกดปุ่ม < ESC > จะเป็นการบอกว่าทำงานเหมือนกับการคลิ้กปุ่มคำสั่งนี้

Style จะเป็นคุณสมบัติที่บอกว่า ต้องการให้ คอนโทรลเป็นแบบธรรมดา ( Standard ) หรือแบบมีรูปภาพ ( Graphical ) ด้วยหรือไม่

Picture คุณสมบัตินี้จะเป็นการบอกว่า ให้แสดงรูปใดในคอนโทรล

DownPicture คุณสมบัตินี้จะเป็นการบอกว่า ให้แสดงรูปใดในปุ่มคำสั่งเมื่อปุ่มคำสั่งถูกกดลงไป

DisabledPicture คุณสมบัตินี้จะเป็นการบอกว่า ให้แสดงรูปใดในปุ่ม คำสั่งเมื่อปุ่มคำสั่งนั้นไม่สามารถทำงานได้ ( Disable อยู่ )

การสร้างฮอตคีย์

กำหนดฮอตคีย์ให้กับคอนโทรลตามที่เราต้องการได้โดยการเพิ่มสัญลักษณ์ & ให้อยู่หน้าตัวอักษรในข้อความที่ปรากฏบนคอนโทรลให้เป็นฮอตคีย์ ทำให้ตัวอักษรตัวนั้นมีเส้นใต้ขีดอยู่และเมื่อเรากดปุ่ม < Alt > + ตัวอักษรที่มีเครื่องหมาย & อยู่ข้างหน้า จะทำให้เราสามารถโฟกัสไปยังคอนโทรลนั้น ๆ ได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

อีเว็นต์

Click จะเป็นอีเว็นต์ที่เกิดขึ้นเมื่อเราได้คลิ้กบนปุ่มคำสั่ง ( คอนโทรลอื่นก็มีอีเว็นต์นี้เช่นกัน และเกิดขึ้นเมื่อมีการคลิ้กที่คอนโทรลนั้นเกิดขึ้น )

GetFocus เป็นอีเว็นต์ที่เกิดขึ้นเมื่อคอนโทรลได้รับโฟกัส

LostFocus เป็นอีเว็นต์ที่เกิดขึ้นเมื่อเลื่อนโฟกัสจากคอนโทรลออกไป

**เช็คบ็อกซ์ ( Check Box )**

เป็นคอนโทรลที่ใช้เป็นตัวเลือกให้ผู้ใช้สามารถเลือกได้ครั้งละหลายๆตัวเลือกในแต่ละกลุ่ม เราสามารถใช้คอนโทรลเฟรมในการแบ่งกลุ่มคอนโทรลเช็คบ็อกซ์

**ภาพที่ 2.13** เครื่องมือสร้างเช็คบ็อกซ์

คุณสมบัติ

Value เป็นคุณสมบัติที่บอกว่า ค่าของคอนโทรลขณะนั้นเป็นเท่าไรได้แก่ ค่า 0 คือ Unchecked ,ค่า 1 คือ Checked และค่า 2 คือ Unavailable

Enable เป็นคุณสมบัติที่บอกว่าจะให้คอนโทรลใช้งานและตอบสนองการทำงานต่ออีเวนท์ได้หรือไม่

Visible เป็นคุณสมบัติที่บอกว่าจะให้คอนโทรลมองเห็นได้หรือไม่ในขณะรันโปรแกรม

**ออปชั่นบัตทอน ( Option Button )**

เป็นคอนโทรลที่ใช้เป็นตัวเลือกให้ผู้ใช้สามารถเลือกได้ครั้งละหนึ่งตัวเลือกเท่านั้น เราสามารถใช้คอนโทรลเฟรมในการแบ่งกลุ่มคอนโทรลเช่นเดียวกับคอนโทรลเช็คบ็อกซ์

**ภาพที่ 2.14** เครื่องมือสร้างออปชั่นบัตทอน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

คุณสมบัติ

Value เป็นคุณสมบัติที่บอกว่าคอนโทรลถูกเลือกในขณะนั้นหรือไม่ โดยจะมีค่าเป็น True เมื่อคอนโทรลถูกเลือกและเป็น False เมื่อไม่ถูกเลือก

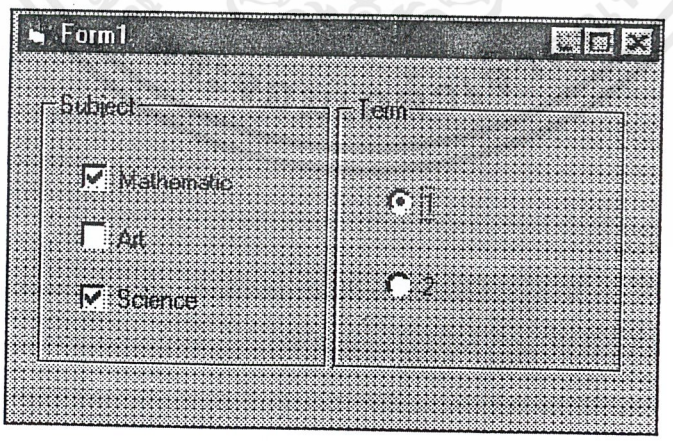
**เฟรม ( Frame )**

เป็นคอนโทรลที่ใช้ในการจัดกลุ่มคอนโทรลที่ต้องใช้ร่วมกัน ทำงานอย่างเดียวกันไว้ด้วยกันซึ่งโดยทั่วไปคอนโทรลเฟรมเป็นคอนโทรลที่ช่วยเพิ่มความเรียบร้อย และความสวยงามของโปรแกรม

**ภาพที่ 2.15** เครื่องมือสร้างเฟรม

คุณสมบัติ

Caption เป็นคุณสมบัติแสดงข้อความบนคอนโทรลนอกจากนี้เรายังสามารถกำหนดคุณสมบัตินี้ให้ใช้ฮอตคีย์ได้เช่นเดียวกับคอนโทรลที่กล่าวมา โดยการใช้อักษร & อยู่หน้าตัวอักษรที่เราต้องการให้เป็นปุ่มที่ใช้ฮอตคีย์




**ภาพที่ 2.16** การใช้เฟรมจัดกลุ่มคอนโทรล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

**ตัวจับเวลา ( Timer )**

เป็นคอนโทรลที่ใช้เพื่อให้ทำงานในทุกๆช่วงเวลาที่กำหนดได้ และสามารถทำงานในแบบหลังฉากได้ ในตอนรันโปรแกรมคอนโทรลนี้จะไม่สามารถมองเห็นได้ ( Non-visual Interface )

**ภาพที่ 2.17** เครื่องมือสร้างตัวจับเวลา

คุณสมบัติ

Interval เป็นคุณสมบัติที่ใช้กำหนดช่วงเวลา มีหน่วยเป็นมิลลิวินาที

อีเว็นต์

Timer เป็นอีเว็นต์ที่เกิดขึ้นในทุกช่วงเวลาที่กำหนดไว้ในคุณสมบัติ Interval

**คอมโบบ็อกซ์ ( ComboBox )**

จะเป็นคอนโทรลที่ใช้ในการแสดงรายการที่ต้องการให้ผู้ใช้เลือกรายการ หรือสามารถแก้ไขรายการที่เลือกได้ คอนโทรลนี้จะใช้พื้นที่ในการวางคอนโทรลน้อยกว่าลิสต์บ็อกซ์ คอนโทรลนี้จะเหมือนกับคอนโทรลเท็กซ์บ็อกซ์ และลิสต์บ็อกซ์รวมกัน

**ภาพที่ 2.18** เครื่องมือสร้างคอมโบบ็อกซ์

การกำหนดรูปแบบของคอมโบบ็อกซ์

เราสามารถกำหนดรูปแบบของคอมโบบ็อกซ์ได้โดยผ่านทางคุณสมบัติ Style เนื่องจากคอมโบบ็อกซ์มีรูปแบบที่ใช้ในการใช้งานอยู่ 3 แบบ คือ

1 ) ดร็อบดาวน์คอมโบบ็อกซ์ ( Drop-down Combo Box ) เป็นคอมโบบ็อกซ์ ซึ่งผู้ใช้สามารถใส่ค่าที่ต้องการลงไปในคอนโทรลโดยตรง หรือ เลือกจากคอนโทรลได้ โดยการคลิ้กที่ปุ่มลูกศรลง เมื่อเลือกแล้วตัวเลือกนั้นจะแสดงอยู่ข้างบนสุดของคอนโทรล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2 ) ซิมเปิ้ลคอมโบบ็อกซ์ ( Simple Combo Box ) เป็นคอมโบบ็อกซ์ ซึ่งรายการที่อยู่ในคอนโทรลจะแสดงอยู่ข้างล่างตลอดเวลา และเมื่อรายการเกินความสูงของคอนโทรล คอนโทรลจะแสดงสกรอลบาร์แนวตั้ง สามารถแก้ไขค่าที่อยู่ข้างบนสุดของคอนโทรลหรือจากรายการก็ได้

3 ) ดร็อบดาวน์ลิสต์บ็อกซ์ ( Drop-down List Box )เป็นคอมโบบ็อกซ์ที่คล้ายกับลิสต์บ็อกซ์ แต่จะประหยัดเนื้อที่ที่ต้องใช้ในการแสดงคอนโทรล ซึ่งจะแสดงรายการเมื่อคลิ้กปุ่มลูกศรลง และคอมโบบ็อกซ์แบบนี้จะไม่สามารถแก้ไขค่าได้ นอกจากเลือกรายการเท่านั้น

**เชพ ( Shape )**

เป็นคอนโทรลทางด้านกราฟฟิกพื้นฐาน ใช้แสดงรูปทรงต่างๆในแอพพลิเคชั่นของเราโดยกำหนดชนิดของรูปทรงในขณะออกแบบได้ 6 รูปแบบคือ สี่เหลี่ยมมุมฉาก, สี่เหลี่ยมจัตุรัส, วงรี , วงกลม, สี่เหลี่ยมมุมฉากมุมมน และสี่เหลี่ยมจัตุรัสมุมมน โดยกำหนดที่คุณสมบัติ Shape

**ภาพที่ 2.19** เครื่องมือสร้างเชพคอนโทรล

คุณสมบัติ

Shape ใช้กำหนดรูปแบบของรูปทรง

BorderWidth และ BorderColor ใช้กำหนดความหนาและสีของเส้นขอบรูปทรง

BorderStyle ใชกำหนดรูปแบบของเส้นขอบ เช่น เส้นประ, เส้นจุด เป็นต้น

FillColor ใช้กำหนดสีที่จะระบายลงบนคอนโทรล

FillStyle ใช้กำหนดรูปแบบของการระบายสีในคอนโทรล

BackStyle ถ้ากำหนดเป็น Opaque พื้นหลังของคอนโทรลจะเป็นสีของคุณสมบัติ BackColor แต่ถ้าเป็น Transparent สีพื้นหลังของคอนโทรลจะเป็นสีของ BackColor โดยจะไม่บังคอนโทรลอื่นที่ซ้อนอยู่กับคอนโทรลนี้

BackColor ใช้กำหนดสีของพื้นหลังของคอนโทรล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

**โพรเกรสบาร์ ( Progress Bar )**

เป็นคอนโทรลขั้นสูงที่ช่วยให้ผู้ใช้รู้ว่า การทำงานในส่วนที่ต้องใช้เวลาในการทำงานนานของโปรแกรมผ่านไปแล้วกี่เปอร์เซ็นต์ ถ้าไม่มีการบอกข้อมูลดังกล่าว อาจทำให้ผู้ใช้เข้าใจผิดว่าเกิดความผิดปกติกับระบบได้ เนื่องจากไม่มีการบอกว่าระบบกำลังทำงานนั้นอยู่

**ภาพที่ 2.20** เครื่องมือสร้างโพรเกรสบาร์

คุณสมบัติ

Min และ Max เป็นคุณสมบัติที่ใช้กำหนดค่าต่ำสุด และสูงสุดของคอนโทรลที่แสดงได้

Value เป็นคุณสมบัติในขณะรันโปรแกรม ที่ช่วยบอกให้ผู้ใช้รู้ว่าการทำงานของโปรแกรมช่วงนั้นอยู่ที่จุดใด โดยไม่สามารถกำหนดค่านี้ในตอนออกแบบได้

**ไดร์ฟลิสต์บ็อกซ์**

เป็นคอนโทรลที่ทำงานแบบคอมโบบ็อกซ์ชนิด Drop-Down List ที่แสดงไดรฟ์ที่มีอยู่ในระบบของเราทั้งหมด ( รวมทั้งไดรฟ์ที่เป็นไดรฟ์ของระบบเครือข่าย ) โดยอัตโนมัติ การเลือกค่าจากคอนโทรลจะทำให้ไดเรกทอรีปัจจุบันของโปรแกรมเปลี่ยนไป

**ภาพที่ 2.21** เครื่องมือสร้างไดร์ฟลิสต์บ็อกซ์

คุณสมบัติ

Drive เป็นคุณสมบัติที่บอกว่า เราได้เลือกไดรฟ์ไหนไว้ในคอนโทรลในขณะนั้น

นอกจากนั้นยังมีคุณสมบัติเหมือนกับคอมโบบ็อกซ์ เช่น คุณสมบัติ List , ListIndex , ListCount เป็นต้น ที่ใช้ในความหมายเดียวกับคอนโทรลลิสต์บ็อกซ์

อีเว็นต์

Change เป็นอีเว็นต์ที่ตอบสนองต่อการเปลี่ยนแปลงค่าของคอนโทรล ไดรฟ์ลิสต์บ็อกซ์ อีเว็นต์นี้เป็นอีเว็นต์ที่เราต้องเขียนคำสั่งจัดการ เมื่อมีการเปลี่ยนแปลงเส้นทางไดเรกทอรีปัจจุบันด้วย เป็นอีเว็นต์ที่เราต้องทำงานด้วยบ่อยมาก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

**ไดเรกทอรีลิสต์บ็อกซ์**

เป็นคอนโทรลที่แสดงเส้นทางปัจจุบันซึ่งรวมทั้งไดร์ฟ และไดเรกทอรีปัจจุบันที่เราทำงานด้วย คอนโทรลนี้จะทำงาน เมื่อเราดับเบิ้ลคลิ้กที่ไดเรกทอรีใดๆ ก็จะเปลี่ยนค่าของไดเรกทอรีปัจจุบัน รวมทั้งคุณสมบัติ Path ด้วยคอนโทรลนี้

**ภาพที่ 2.22** เครื่องมือสร้างไดเรกทอรีลิสต์บ็อกซ์

คุณสมบัติ

Path เป็นคุณสมบัติที่มีอยู่ในคอนโทรลไดเรกทอรีลิสต์บ็อกซ์ และไฟล์ลิสต์บ็อกซ์ เป็นคุณสมบัติที่จะบอกให้ว่าไดเรกทอรีปัจจุบันนั้นอยู่ที่ไหน เป็นคุณสมบัติที่มีในตอนรันโปรแกรมเท่านั้น คุณสมบัตินี้ จะเปลี่ยนแปลงเมื่อเราเปลี่ยนแปลงค่าในคอนโทรลไดเรกทอรีลิสต์บ็อกซ์โดยการดับเบิ้ลคลิ้กการแสดงค่าจะอยู่ในรูป เช่น " c:\Windows\System "เป็นต้น

**ไฟล์ลิสต์บ็อกซ์**

เป็นคอนโทรลที่แสดงไฟล์ทั้งหมดที่มีอยู่ในไดร์ฟ และไดเรกทอรีปัจจุบันตามแอตทริบิวต์ที่เราต้องการ

**ภาพที่ 2.23** เครื่องมือสร้างไฟล์ลิสต์บ็อกซ์

คุณสมบัติ

Pattern เป็นคุณสมบัติที่จะบอกว่าให้คอนโทรลแสดงไฟล์ที่มีส่วนขยายอะไร วิธีการใช้ เช่น FileListBox1.Pattern = " *.exe " จะเป็นการแสดงไฟล์ที่มีส่วนส่วนขยายเป็น exe ทั้งหมดในไดเรกทอรีปัจจุบัน

FileName เป็นคุณสมบัติที่จะบอกว่าให้ เราได้เลือกไฟล์ใดไว้บ้าง และบอกทั้งไดเรกทอรีของไฟล์ที่เราได้เลือกไว้ด้วย

Archive , Hidden , Normal และ System เป็นคุณสมบัติที่บอกว่า เราต้องการแสดงไฟล์ประเภทใดบ้าง ถ้ากำหนดคุณสมบัติดังกล่าวเป็น True จะแสดงไฟล์ประเภทนั้นในคอนโทรลนั้นในคอนโทรลด้วย ถ้าเป็น False จะไม่แสดงในคอนโทรล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

**อิมเมจและพิคเจอร์บ็อกซ์ (Image and Picture Box )**

อิมเมจคอนโทรล และพิกเจอร์บ็อกซ์คอนโทรล ทำให้สามารถวางสารสนเทศทางกราฟฟิกไว้ในตำแหน่งเฉพาะบนฟอร์ม พิกเจอร์บ็อกซ์จะมีความยืดหยุ่นมากกว่า และด้วยเหตุนี้จึงต้องการหน่วยความจำและหน่วยประมวลผลในปริมาณที่มากกว่า พิกเจอร์บ็อกซ์เหมาะอย่างยิ่งสำหรับสภาพแวดล้อมไดนามิก เช่น การวาดกราฟฟิกเข้าไปที่จอภาพโดยตรงในขณะที่โปรแกรมกำลังทำงาน หรือการทำไอคอนให้เคลื่อนไหวได้ ด้วยการเคลื่อนย้ายไอคอนไปรอบๆจอภาพ อิมเมจคอนโทรลเหมาะสำหรับสถานการณ์สถิต นั่นคือ ไม่มีการเปลี่ยนแปลงภาพที่สร้างหรือวางเอาไว้บนจอ

**ภาพที่ 2.24** เครื่องมือสร้างอิมเมจคอนโทรล

**ภาพที่ 2.25** เครื่องมือสร้างพิกเจอร์บ็อกซ์

คุณสมบัติ

คุณสมบัติ Enabled, Height, Left, Name, Picture, Top, Visible และ Width ส่วนใหญ่จะคล้ายกับคุณสมบัติของฟอร์ม เราสามารถตั้งคุณสมบัติ Picture ให้แสดงบิตแม็พหรือไอคอน โดยในเวลาออกแบบให้ใช้หน้าต่างคุณสมบัติและไดอะล็อกบ็อกซ์ Load Picture ในเวลารันจำเป็นต้องใช้ฟังก์ชัน LoadPicture พิกเจอร์บ็อกซ์ซึ่งทำงานเกือบเหมือนกับฟอร์มเล็กๆ ที่อยู่ภายในฟอร์มหลัก มีคุณสมบัติที่ไม่มีอยู่ในอิมเมจคอนโทรล นั่นคือ AutoRedraw, FontBold, FontItalic, FontName, FontSize, FontUnderline ซึ่งคุณสมบัติเหล่านี้ทำงานในทำนองเดียวกับที่มีอยู่ในฟอร์ม อิมเมจคอนโทรลมีคุณสมบัติอย่างหนึ่งที่ไม่มีอยู่ในพิกเจอร์บ็อกซ์คือ คุณสมบัติ Stretch โดยปกติแล้วจะมีค่าเป็น False ทำให้อิมเมจคอนโทรลปรับขนาดตัวเองเพื่อให้พอดีกับภาพที่บรรจุอยู่ แต่เมื่อกำหนดให้เป็น True จะทำให้อิมเมจคอนโทรลปรับขนาดภาพให้พอดีกับขนาดของคอนโทรลที่เราวาดเอาไว้ จอแสดงผลของฟอร์มนั้นประกอบด้วยชั้น 3 ชั้น สารสนเทศทั้งหมดที่แสดงบนฟอร์มโดยตรงจะปรากฏอยู่ชั้นล่างสุด สารสนเทศจากคอนโทรลทางด้านกราฟฟิก 3 ตัว คือ อิมเมจ ( Image ), เชพ ( Shape ) และ ไลน์ ( Line ) จะปรากฏอยู่ตรงชั้นกลาง และอ็อบเจ็กต์อื่นๆทั้งหมด จะอยู่ชั้นบนสุด ด้วยเหตุนี้อิมเมจคอนโทรลจึงอาจบดบังข้อความที่พิมพ์ออกมาบนฟอร์มในขณะเดียวกันพิกเจอร์บ็อกซ์ก็อาจบดบังอิมเมจคอนโทรลได้เช่นกัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

อีเว็นต์

Click, DblClick เป็นอีเว็นต์พื้นฐานที่พบง่ายและบ่อยที่สุด

DragDrop เกิดขึ้นเมื่อผู้ใช้ลากแล้ววาง

MouseMove, MouseDown, MouseUp เกิดขึ้นเมื่อมีการใช้งานเมาส์นั่นคือ เมื่อมีการเคลื่อนเมาส์, กดปุ่มและปล่อยปุ่มของเมาส์

GetFocus, LostFocus เป็นอีเว็นต์ของพิกเจอร์บ็อกซ์ที่ไม่มีในอิมเมจคอนโทรล เกิดขึ้นเมื่อคอนโทรลได้รับโฟกัสและเมื่อเลื่อนโฟกัสออกจากคอนโทรล

ฟังก์ชันและเมดถอด

เราสามารถเรียกฟังก์ชัน LoadPicture เพื่อใช้ตั้งคุณสมบัติ Picture ของพิกเจอร์บ็อกซ์หรืออิมเมจคอนโทรล พิกเจอร์บ็อกซ์ยังสนับสนุนเมดถอด Cls, Point และเมดถอดทางกราฟฟิกอื่นๆ

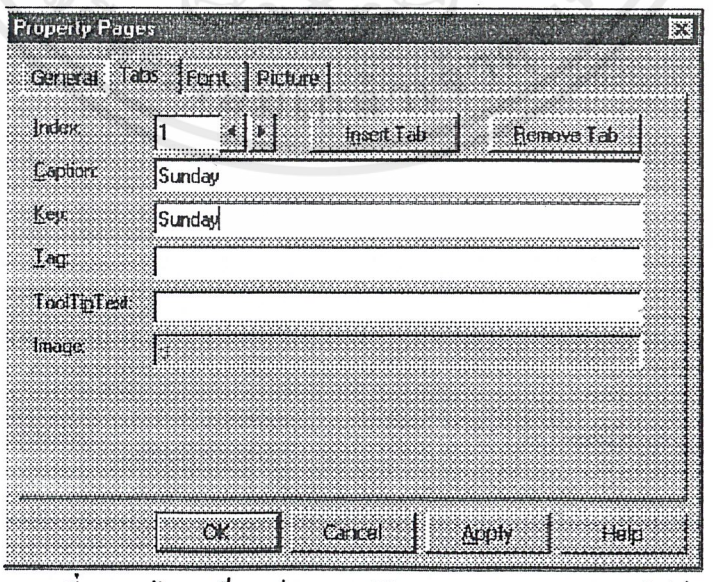
**แท็บสตริปพ์ ( Tab Strip )**

เป็นคอนโทรลขั้นสูงที่ช่วยในการแยกส่วนประกอบต่างๆออกเป็นกลุ่ม ทำให้ประหยัดพื้นที่ของหน้าจอที่ใช้ในการแสดงผล และทำให้สะดวกในการทำงานกับโปรแกรม

**ภาพที่ 2.26** เครื่องมือสร้างแท็บสตริปพ์

คุณสมบัติ

การกำหนดคุณสมบัติของคอนโทรลตัวนี้ ให้เราเลือกที่คุณสมบัติ Custom จะปรากฏไดอะล็อกซ์ดังรูปต่อไปนี




**ภาพที่ 2.27** ไดอะล็อกซ์คุณสมบัติ Custom ของแทบสตริปพ์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ในไดอะล็อกซ์ที่ปรากฏออกมาจะมีแท็บที่ชื่อว่า Tab อยู่ ซึ่งคุณสมบัติที่อยู่ในแท็บนี้จะใช้ในการกำหนดคุณสมบัติต่างๆของแต่ละแท็บที่อยู่ในคอนโทรลแท็บสตริปพ์ มีคุณสมบัติสำคัญดังนี้

Key เป็นคุณสมบัติที่ใช้ในการบ่งบอกว่า เป็นแท็บอันไหนในคอลเล็กชั่น Tabs

ToolTipText เป็นคุณสมบัติที่ใช้ในการแสดง ToolTip ซึ่งเป็นข้อความที่แสดงความหมายของแท็บนั้นให้ผู้ใช้เข้าใจอย่างง่ายๆ เมื่อมีการหยุดตัวชี้เมาส์ที่แท็บของคอนโทรลสักครู่

Caption เป็นคุณสมบัติที่กำหนดข้อความที่จะแสดงบนหัวของแต่ละแท็บ

Image เป็นคุณสมบัติที่บอกว่าจะแสดงรูปภาพไหนของอิมเมจคอนโทรลที่กำหนดไว้ในคุณสมบัติ ImageList

นอกจากนี้แท็บสตริปพ์ยังมีคุณสมบัติที่สำคัญดังต่อไปนี้

SelectItem เป็นคุณสมบัติที่บอกว่าเราได้เลือกรายการอันไหนในคอนโทรล ซึ่งเป็นคุณสมบัติที่มีตอนรันโปรแกรมเท่านั้น

Style เป็นคุณสมบัติที่กำหนดว่าจะแสดงลักษณะของคอนโทรลเป็น Tab หรือ Button

ShowTips เป็นคุณสมบัติที่ใช้ในการบอกว่าจะให้แสดง ToolTip หรือไม่ ถ้าค่าเป็น True จะแสดงToolTip ได้ แต่ถ้าค่าเป็น False จะไม่แสดง ToolTip

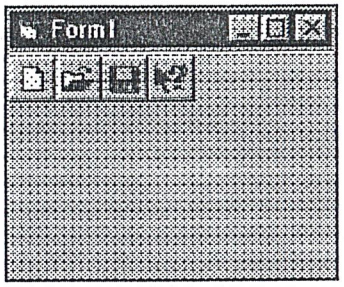
**ทูลบาร์ ( Toolbar ) และอิมเมจลิสต์ ( ImageList )**

เป็นคอนโทรลขั้นสูงซึ่งในโปรแกรมที่เราเคยพบในวินโดวส์ส่วนใหญ่ จะมีทูลบาร์ที่ช่วยให้เราทำงานกับโปรแกรมได้สะดวกและรวดเร็ว ซึ่งทูลบาร์จำพวกนี้จะช่วยให้ผู้ใช้เข้าถึงคำสั่งในเมนูของโปรแกรมได้รวดเร็วยิ่งขึ้น

**ภาพที่ 2.28** เครื่องมือสร้างทูลบาร์

**ภาพที่ 2.29** เครื่องมือสร้างอิมเมจลิสต์

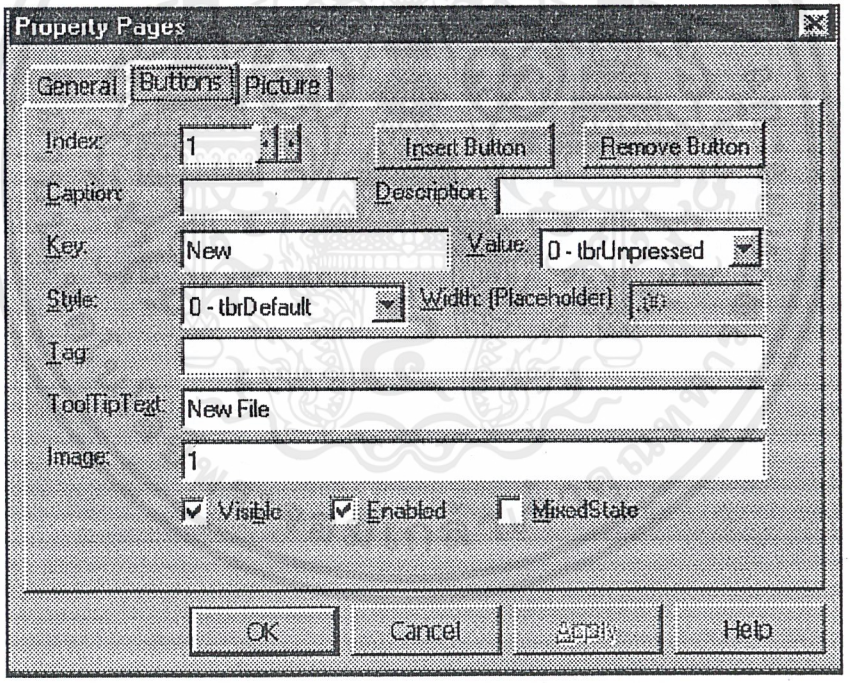
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้




**ภาพที่ 2.30** การใช้ทูลบาร์และอิมเมจลิสต์

คุณสมบัติที่สำคัญของทูลบาร์คอนโทรล

การกำหนดคุณสมบัติของคอนโทรลตัวนี้ ให้เราเลือกที่คุณสมบัติ Custom จะปรากฏไดอะล็อกซ์ดังนี้




**ภาพที่ 2.31** ไดอะล็อกซ์คุณสมบัติ Custom ของทูลบาร์

ในไดอะล็อกซ์นี้จะมีอยู่ 3 แท็บ ในแต่ละแท็บจะมีรายละเอียดดังต่อไปนี้

แท็บ General

จะเป็นแท็บที่มีคุณสมบัติเกี่ยวกับปุ่มบนทูลบาร์ที่มีคุณสมบัติสำคัญดังต่อไปนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ImageList เป็นคุณสมบัติที่บอกว่าให้แสดงรูปจากคอนโทรลอิมเมจลิสต์อันไหน

ButtonWidth , ButtonHeight เป็นคุณสมบัติที่ใช้กำหนดขนาด ความกว้างและความสูงของแต่ละปุ่มบนทูลบาร์

แท็บ Buttons

จะเป็นแท็บที่ใช้ในการเพิ่มหรือลดปุ่มบนทูลบาร์ที่มีคุณสมบัติสำคัญของแต่ละปุ่มดังต่อไปนี้

Key เป็นคุณสมบัติที่ช่วยในการบอกว่าเป็นปุ่มใดในทูลบาร์ โดยกำหนดค่าเป็นสตริง เพื่ออ้างอิงถึงปุ่มนั้นๆ

Style เป็นคุณสมบัติที่ใช้กำหนดรูปแบบของปุ่มบนทูลบาร์ เช่น ให้เป็นปุ่มธรรมดา, เป็นปุ่มที่ทำหน้าที่คล้ายเช็คบ็อกซ์, เป็นปุ่มที่อยู่เป็นกลุ่มกับปุ่มอื่นๆ และทำหน้าที่คล้ายออบชั่นบัตทอน เป็นต้น

ToolTipText เป็นคุณสมบัติที่ใช้แสดงข้อความอธิบายสั้นๆ เมื่อเราหยุดตัวชี้เมาส์บนปุ่มนั้นๆสักครู่โดยที่ไม่ได้คลิ๊กปุ่ม ซึ่งจะเกิดขึ้นเมื่อกำหนดคุณสมบัติ ShowTips ให้เป็น True ด้วย

แท็บ Picture

จะเป็นแท็บที่ใช้กำหนดรูปภาพและคุณสมบัติของรูปภาพของปุ่มบนทูลบาร์ มีคุณสมบัติที่สำคัญดังนี้

AllowCustomize เป็นคุณสมบัติที่กำหนดให้มีการจัดรูปแบบของทูลบาร์ในขณะรันโปรแกรมได้ เมื่อมีค่าเป็น True และไม่ได้เมื่อมีค่าเป็น False

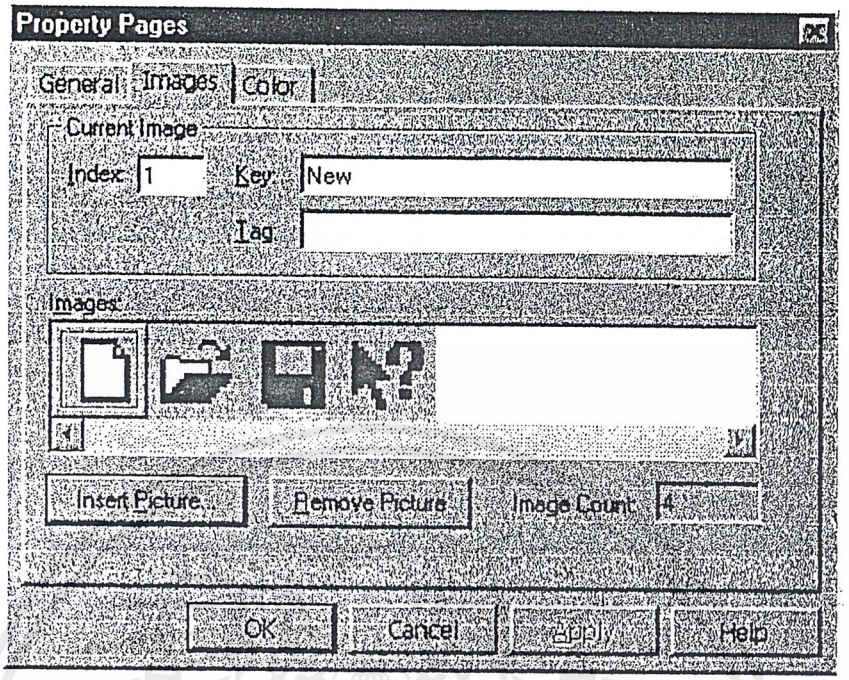
ShowTips เป็นคุณสมบัติที่กำหนดให้มีการแสดง ToolTip ที่อยู่ในคุณสมบัติ ToolTipText เมื่อมีค่าเป็น True และไม่มีการแสดง ToolTip เมื่อมีค่าเป็น False

Wrappable เป็นคุณสมบัติที่กำหนดให้แสดงทูลบาร์คอนโทรลในอีกแถวเมื่อไม่สามารถแสดงได้ครบในแถวเดียว เมื่อมีค่าเป็น True และไม่มีการขึ้นแถวใหม่เมื่อมีค่าเป็น False

คุณสมบัติที่สำคัญของอิมเมจลิสต์คอนโทรล

เมื่อเลือกคุณสมบัติ Custom ของคอนโทรลนี้จะปรากฏไดอะล็อกซ์ดังรูป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้




**ภาพที่ 2.32** ไดอะล็อกซ์คุณสมบัติ Custom ของอิมเมจลิสต์

ในไดอะล็อกซ์นี้จะมีอยู่ 3 แท็บ แต่ละแท็บมีรายละเอียดดังนี้

แท็บ General เป็นแท็บที่ใช้กำหนดว่ารูปที่แสดงจะมีขนาดเท่าใดและจะใช้ MaskColor หรือไม่

แท็บ Images เป็นแท็บที่ใช้ในการเพิ่มและลบรูปภาพในคอนโทรล และมีคุณสมบัติ Key หรือ Index ที่จะช่วยให้เราเข้าถึงรูปภาพที่ต้องการได้

แท็บ Color เป็นแท็บที่ใช้กำหนดสีที่จะแสดงบนรูปภาพว่าสีใดเป็นสี BackGround และ MaskColor

**เครื่องมือกำหนดการสื่อสาร ( Communication Control )**

การติดต่อสื่อสารระหว่าง Computer และ PC จะกระทำผ่าน control ของ Visual Basic ที่เรียกว่า Microsoft communication control หรือ MSComm ผ่าน port อนุกรม RS-232 ซึ่งเป็น port ที่ใช้กันอย่างแพร่หลาย ซึ่งกำหนดมาตรฐานโดยองค์การอุตสาหกรรมอิเล็กทรอนิกส์แห่งอเมริกา (EIA) โดยแบ่งการเชื่อมต่อออกเป็น 2 ลักษณะ คือ DTE (Data Terminal Equipment) และ DCE (Data Communication Equipment) ซึ่งโดยปกติ DTE จะต้องต่อเข้ากับ DCE เสมอ เช่น การต่อเครื่องคอมพิวเตอร์(DTE)เข้ากับโมเด็ม (DCE) เป็นต้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

**ภาพที่ 2.33** MSCOMM CONTROL

พอร์ตอนุกรม RS –232 จะเป็นพอร์ตของเครื่องคอมพิวเตอร์ทั้งประเภท 9 ขา และ 25 ขา ซึ่งเราเรียกว่า พอร์ต Com1 และ Com2 นั่นเองโดยในทางความเป็นจริงพอร์ตอนุกรมไม่ได้ถูกควบคุมโดยเมนบอร์ด แต่การสื่อสารทั้งหมดจะถูกจัดการโดยชิป UART (Universal Asynchronous Receiver/Transmitter) อีกทีหนึ่งซึ่ง UART จะทำหน้าที่ในการรับ/ส่งข้อมูลดังนี้

**การส่งข้อมูล(Data Transmission)**

- รับตัวอักษรจากคอมพิวเตอร์
- แปลงตัวอักษรให้เป็นสายข้อมูลแบบบิต(เราเรียกว่าเป็นขบวนการ Serialization)
- สร้างเฟรมของข้อมูลโดยเพิ่มบิตที่จำเป็นสำหรับการสื่อสารและการตรวจสอบ เช่น start , stop และ parity เป็นต้น
- ส่งผ่านข้อมูลที่สร้างขึ้นมาจากขั้นตอนที่ผ่านมาด้วยความเร็วของ baud rate
- แสดงสถานะพร้อมที่จะรับข้อมูลตัวอักษรถัดไปให้กับเครื่องคอมพิวเตอร์

**การรับข้อมูล (Data Receiver)**

- รับตัวอักษรจากการอินเทอร์เฟส
- ตรวจสอบความถูกต้องของเฟรมข้อมูลตามมาตฐานที่กำหนด โดยหากเฟรมของข้อมูลมีรูปแบบที่ไม่ถูกต้องจะมีการแจ้งข้อผิดพลาดทันที
- ตรวจสอบความถูกต้องของพาริตี้
- แปลงสายข้อมูลแบบบิตให้เป็นตัวอักษร
- ส่งตัวอักษรให้กับเครื่องคอมพิวเตอร์
- แสดงสถานะความพร้อมที่จะรับข้อมูลตัวอักษรถัดไปให้กับอินเทอร์เฟส

การนำ MSComm มาใช้มีรายละเอียดดังนี้คือ

1. ต้องมี MSComm ซึ่งจะมีอยู่แล้วเมื่อตอนที่ติดตั้ง Visual Basic
2. กำหนด Custom Control MSCOMM.OCX(สำหรับ Visual Basic 4 ขึ้นไป) โดยการเลือก MSCOMM.OCX จากเมนู Project/Components/ ที่ Control เลือก Microsoft Comm Control 5.0 (กรณีที่ใช้ version 5) ก็จะได้ MSComm รูปโทรศัพท์ปรากฏในทูลบอกซ์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3. นำคอนโทรลไปใส่ไว้ใน Formที่ต้องการใช้งาน
4. กำหนดคุณสมบัติต่างๆให้กับ Control เขียนฟังก์ชั่นที่เกี่ยวข้องกับ MSComm

ในการใช้ MSComm ติดต่อกับ Serial Port นั้นมีการติดต่อกัน 2 แบบ คือ

1. การสื่อสารแบบกระตุ้นด้วยเหตุการณ์ (Event-Driven Communication) หรือเป็นการเขียนกระตุนด้วยอินเทอร์รัปนั่นเอง โดยเมื่อข้อมูลเข้ามาก็จะเกิด CommEvent ขึ้นกับ OnComm Event
2. การสื่อสารแบบโพลลิ่ง(Polling) โดยการวนรอบตรวจสอบข้อมูลจาก Serial Port ตลอดเวลา

การกำหนด Property ที่สำคัญๆเช่น

- ComPort การกำหนดหมายเลขของ Port ในการติดต่อ RS 232 เช่น Com1, Com2 เป็นต้น

ตัวอย่างการกำหนด ➔ MSComm1.ComPort =1

- Settings เป็นการกำหนดอัตราบอด แพริที่ จำนวนข้อมูลบิตจบ เช่น 9600,n,8,1 สำหรับใน PC Link กำหนดเป็น 9600,e, 7,2 ตัวอย่างการกำหนด ➔ MSComm1.Settings = “9600,e, 7,2”
- PortOpen เป็นการกำหนดสถานะว่าให้เปิด หรือ ปิด พอร์ต ตัวอย่างการกำหนด ➔ MSComm1.PortOpen =True สำหรับการเปิดพอร์ต และ MSComm1.PortZOpen = False สำหรับการสั่งให้ปิดพอร์ต
- Input เป็นการย้ายข้อมูลจากบัฟเฟอร์การรับข้อมูล
- Output ส่งข้อมูลให้กับบัฟเฟอร์ส่งข้อมูลส่งออก

ตัวอย่างการกำหนด ➔ MSComm1.Input = ข้อมูลที่จะรับจากพอร์ต

MSComm1.Output = ข้อมูลที่จะส่งออกพอร์ต

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

# บรรณานุกรม


1. ธาริน สิทธิธรรมชารี , ธนัญชัย จำนงค์ภักดี , "Microsoft Visual Basic Version 5.0" , บริษัท ซีเอ็ดยูเคชั่น จำกัด ( มหาชน )
2. สุทธิศักดิ์ พงศ์ธนาพานิช , "Visual Basic 5.0 Professional การใช้คำสั่งและคอนโทรล ActiveX" , บรัษัท ซีเอ็ดยูเคชั่น จำกัด ( มหาชน )
3. กิตติ ภักดีวัฒนะกุล , ทำลอง ครูอุตสาหะ , "Visual Basic 5 ฉบับ โปรแกรมเมอร์" ,บริษัท ดวงกมลสมัย จำกัด
4. ณรงค์ ตันชีวะวงศ์ , "ระบบ PLC ( Programmable Logic Controller )" , สมาคมส่งเสริมเทคโนโลยี ( ไทย – ญี่ปุ่น )
5. ภาควิชาเทคโนโลยีการวัดคุมทางอุตสาหกรรม , "OMRON FA 2 ADVANCED PLC WITH C200HS", คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
6. OMRON ELECTRONICS CO.,LTD , "ADVANCED OMRON FA2 TRAINING COURSE" , คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
7. OMRON JAPAN , "C200HS Programmable Controllers Operation Manual" , 1995
8. OMRON JAPAN , "SYSMAC Programmable Controllers C200HS Installation Guide", 1995


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

# บทที่ 3
# การออกแบบโปรแกรมสนับสนุนการทำงานของระบบพีซีลิงค์

โปรแกรมสนับสนุนการทำงานของระบบพีซีลิงค์ที่ออกแบบขึ้นมานี้ช่วยอำนวยความสะดวกแก่ผู้ใช้ในการตั้งค่าพารามิเตอร์สำหรับการเชื่อมต่อ, แสดงผลและแก้ไขค่าสถานะของ PLC โดยอาศัยรูปแบบข้อตกลงในการสื่อสารหรือโปรโตคอลของการสื่อสารแบบโฮสต์ลิงค์ เขียนด้วยโปรแกรม Visual Basic 5.0 ภายใต้ระบบจัดการของ Microsoft Windows 95 ในบทนี้จะกล่าวถึงการนำข้อตกลงในการสื่อสาร มาประยุกต์ใช้งานร่วมกับโปรแกรมที่ทำงานบนเครื่องคอมพิวเตอร์ ตลอดจนโครงสร้างและวิธีการในการออกแบบโปรแกรมแต่ละส่วน

## 3.1 คุณสมบัติทั่วไปของโปรแกรมสนับสนุนการทำงาน

1. เป็นโปรแกรมที่ใช้ในการตรวจสอบสถานะ ( Monitoring ) การทำงานของพื้นที่หน่วยความจำส่วนต่างๆ นอกจากนี้ยังสามารถกำหนดค่า ( Force ) ให้กับพื้นที่หน่วยความจำดังกล่าวของเครื่องควบคุม PLC ได้
2. เป็นโปรแกรมที่ใช้จัดการเชื่อมโยงข้อมูลให้กับเครื่องควบคุม PLC ที่เชื่อมต่ออยู่ในระบบเพื่อให้เครื่องควบคุมแต่ละเครื่องที่ต่ออยู่ในระบบ สามารถสื่อสารข้อมูลถึงกันได้

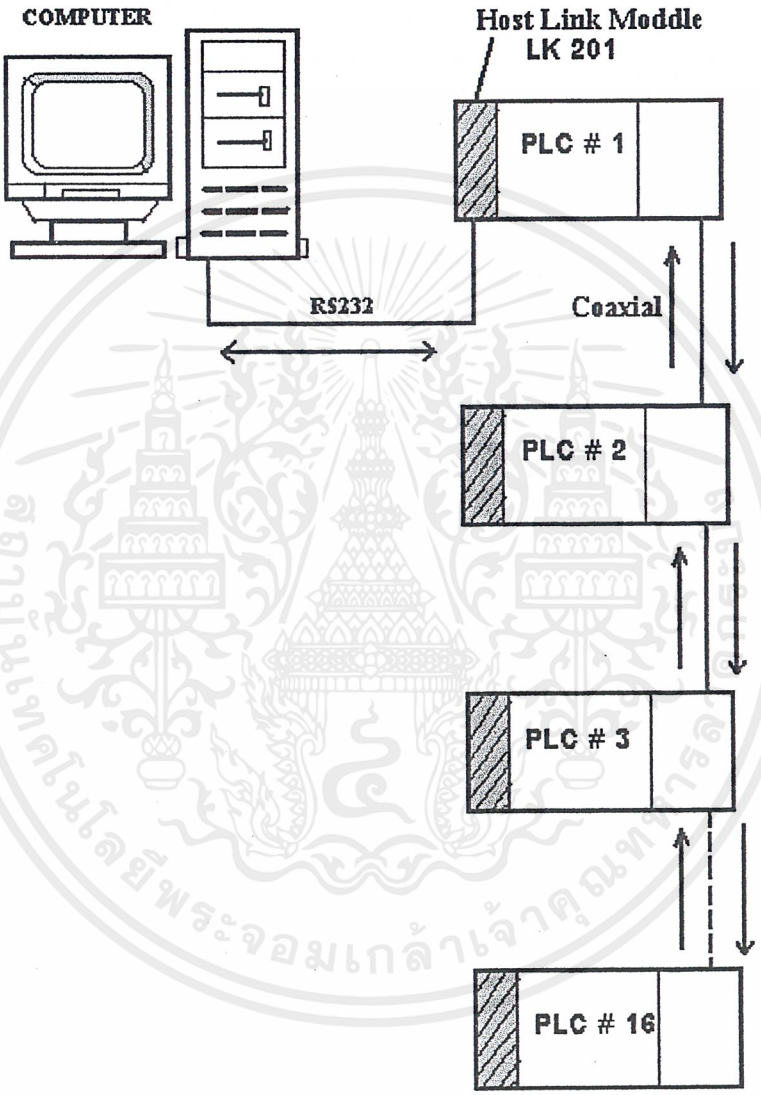
## 3.2 การออกแบบโปรแกรมเพื่อตรวจสอบสถานะการทำงานของเครื่องควบคุม PLC

ในการตรวจสอบสถานะการทำงานของเครื่องควบคุม PLC ได้แบ่งพื้นที่ออกเป็น 5 ส่วน ดังนี้

1. พื้นที่ของ Internal และ Special Relay
2. พื้นที่ของ Link Relay
3. พื้นที่ของ Data Memory
4. พื้นที่ของ Auxiliary Relay
5. พื้นที่ของ Timer และ Counter

ในการเชื่อมต่อ PLC เข้ากับคอมพิวเตอร์เพื่อติดต่อสื่อสารข้อมูลนั้นได้ใช้สาย RS-232 ส่วนการเชื่อมต่อระหว่าง PLC นั้น ได้ใช้สาย Coaxial ซึ่งแสดงเป็นแผนผังการสื่อสารข้อมูลได้ดังภาพที่ 3.1

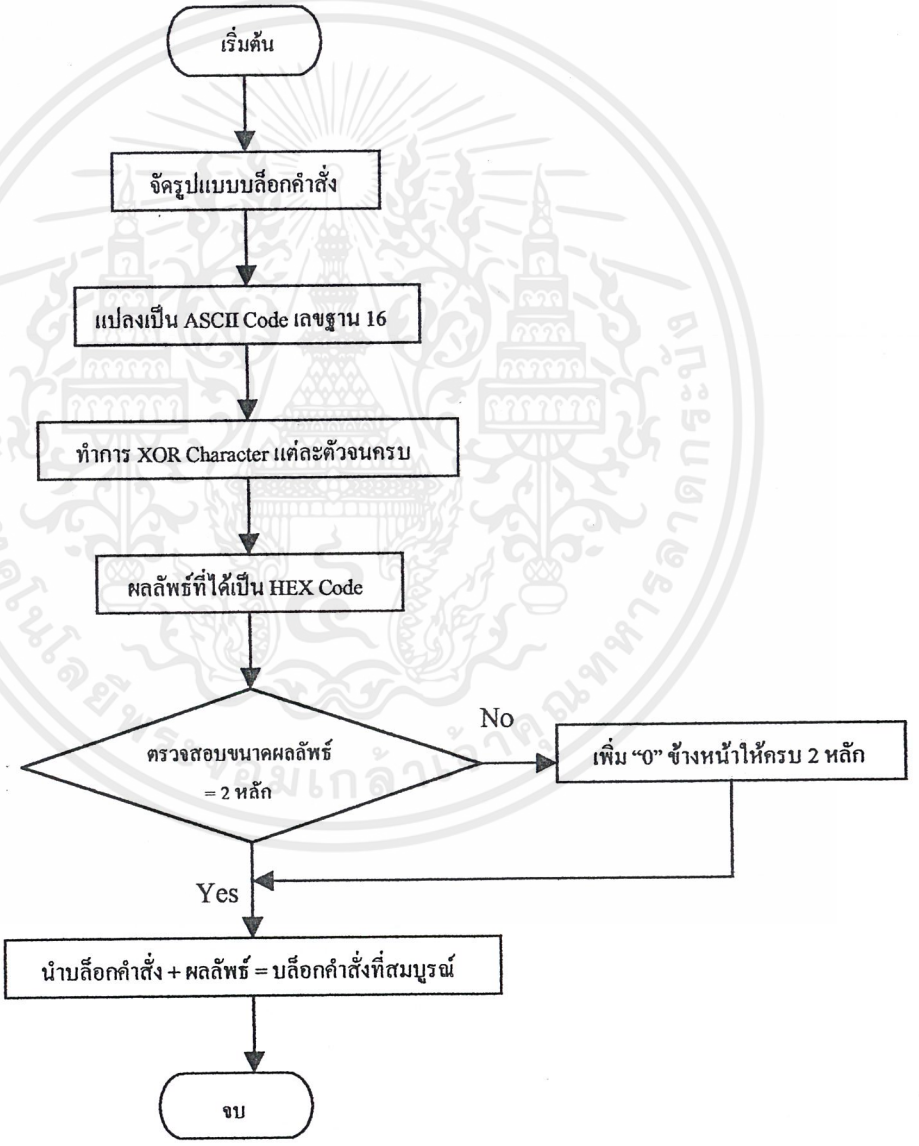
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้




**ภาพที่ 3.1** แผนผังการสื่อสารข้อมูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

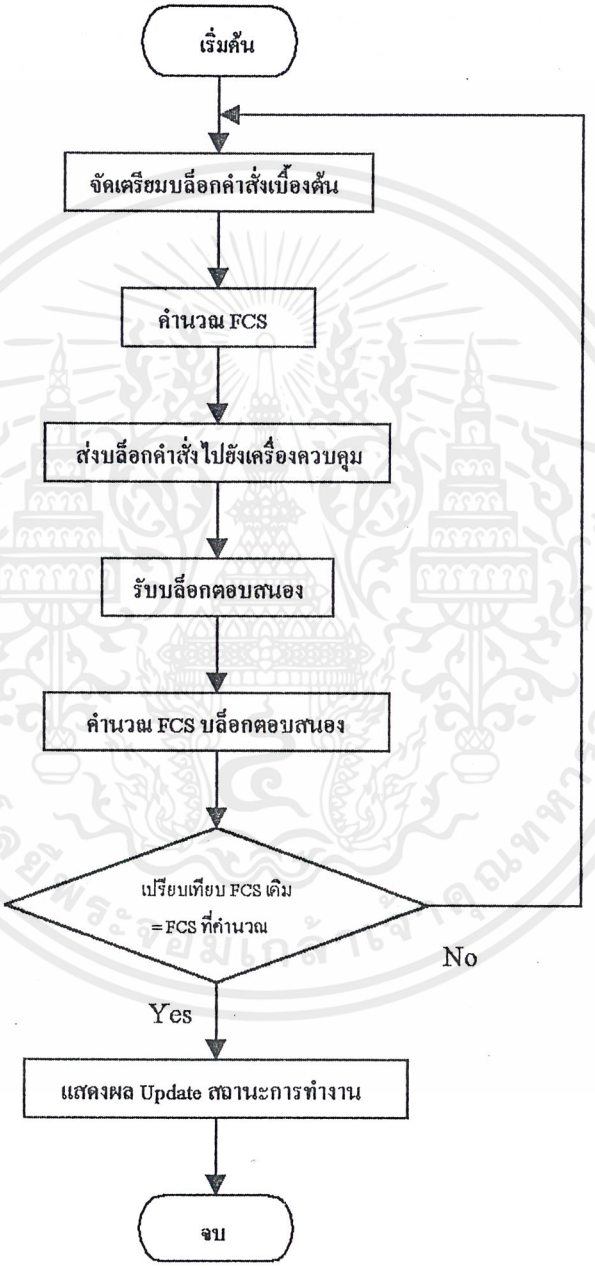
การเขียนโปรแกรมเพื่อจะทำการติดต่อสื่อสารกับเครื่องควบคุม PLC จะต้องมีความเข้าใจเกี่ยวกับโปรโตคอลของเครื่องควบคุมเป็นอย่างดี ซึ่งรายละเอียดของโปรโตคอลได้กล่าวไว้แล้วในบทที่ 2 สำหรับโปรแกรมในส่วนที่ต้องมีการติดต่อสื่อสารนั้น แต่ละโปรแกรมที่ทำการออกแบบต้องมีการควบคุมความผิดพลาดเหมือนกันก็คือ โปรแกรมในการคำนวณ FCS ( Frame Check Sequence ) โครงสร้างและแนวทางในการออกแบบได้แสดงในแผนผังภาพที่ 3.2




**ภาพที่ 3.2** แผนผังการคำนวณ FCS

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

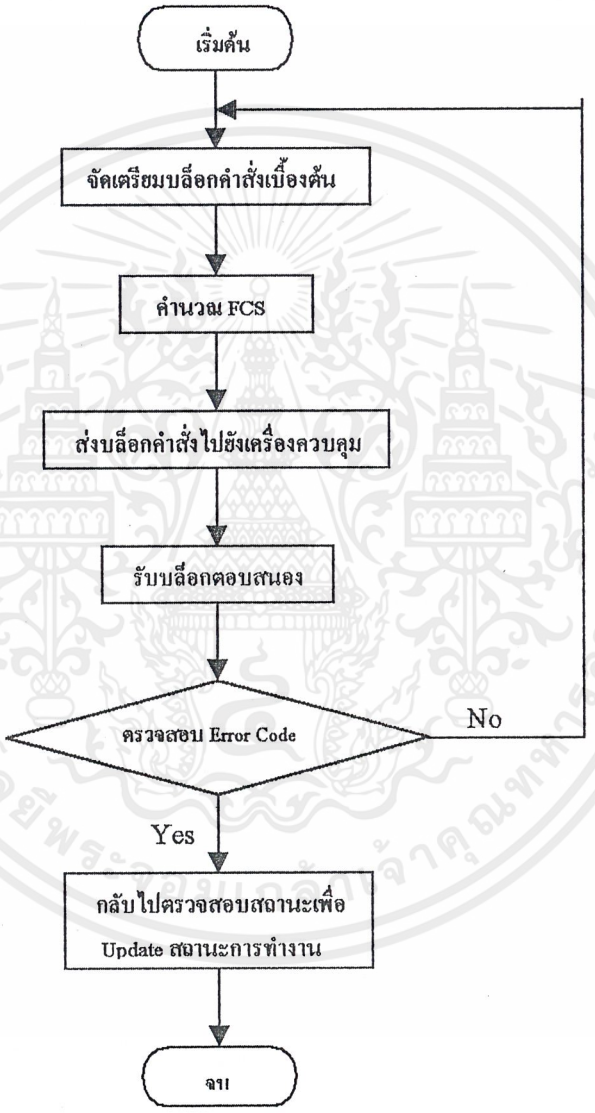
จากโปรโตคอลที่ใช้ในการอ่านข้อมูลในพื้นที่ต่างๆ จะได้บล็อกตอบสนองมาแสดงผลของสถานะการทำงานในรูปแบบที่เข้าใจง่าย ดังแผนผังในภาพที่ 3.3




**ภาพที่ 3.3** แผนผังการตรวจสอบสถานะการทำงาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การออกแบบโปรแกรมสำหรับกำหนดค่า ( Force ) ให้กับพื้นที่บางส่วนของเครื่องควบคุม แสดงเป็นแผนผังการทำงานดังภาพที่ 3.4




**ภาพที่ 3.4** แผนผังการกำหนดค่าสถานะของ PLC

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

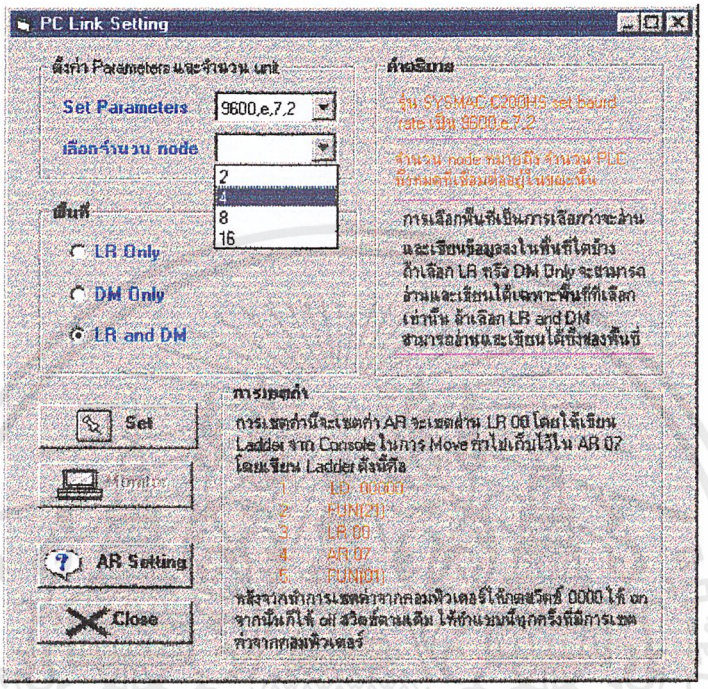
# บทที่ 4
# การใช้งานของโปรแกรม

การทำงานของโปรแกรม แบ่งเป็น 3 ส่วน ดังนี้

1. **ส่วนของการตั้งค่า ( Setting Part )**
   - ส่วนของ PC Link Setting
   - ส่วนของ Host link Setting
2. **ส่วนของ Host Link Monitoring**
   - สามารถตรวจดูค่าสถานะของ Internal Relay และ Special Relay
   - สามารถตรวจดูค่าและกำหนดค่าสถานะใน พื้นที่ Link Relay
   - สามารถตรวจดูค่าและกำหนดค่าสถานะในพื้นที่ Data Memory
   - สามารถตรวจดูค่าบิท ON-OFF ในพื้นที่ Auxiliary Relay
   - สามารถตรวจดูค่าและกำหนดค่าสถานะในพื้นที่ Timer และ Counter
3. **ส่วนของ PC Link Monitoring**
   - สามารถตรวจดูค่าในพื้นที่ Data Memory ของ PLC ที่เชื่อมต่อกัน
   - สามารถตรวจดูค่าในพื้นที่ Link Relay ของ PLC ที่เชื่อมต่อกัน
   - สามารถแก้ไขข้อมูลในพื้นที่ Data Memory ของ PLCตัวที่เชื่อมต่อกับคอมพิวเตอร์
   - สามารถแก้ไขข้อมูลในพื้นที่ Link Relay ของ PLCตัวที่เชื่อมต่อกับคอมพิวเตอร์
4. **ส่วนของ Help**
   - การปรับเซตPC
   - การเซตพื้นที่AR
   - การแบ่งพื้นที่ LR และ DM
   - การเซตARจากConsole
   - การMonitorผ่านConsole

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

**ขั้นตอนการใช้งานในส่วนของการตั้งค่า ( Setting Part)**




**ภาพที่ 4.1** แสดงหน้าจอการตั้งค่า (Setting)

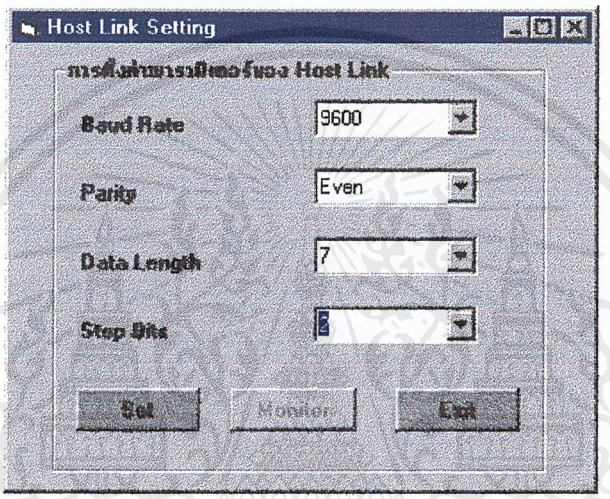
**ขั้นตอนการตั้งค่าในส่วนของ PC Link Setting**

1. ตั้งค่าพารามิเตอร์สำหรับการติดต่อสื่อสารแบบ Host Link โดยเลือกค่าจาก Combo Box สำหรับการติดต่อแบบ Host link จะต้องตั้งค่า ดังนี้ Baud rate 9600 bps , Even Parity, Data Length = 7 และ Stop Bits = 2
2. ตั้งค่าจำนวนUnitที่ทำการต่อ โดยเลือกค่าจาก Combo Box
3. เลือกพื้นที่ที่จะสามารถติดต่อกันได้ เมื่อเป็นการสื่อสารแบบ PC Link
4. กดปุ่มคำสั่ง **Set** เพื่อนำค่าที่ตั้งไว้ส่งไปตั้งค่าในพื้นที่ AR 07 ของ PLC
5. กดปุ่มคำสั่ง **Monitor** เมื่อการตั้งค่าเรียบร้อย และต้องการจะไปยังหน้าจอถัดไป
6. กดปุ่มคำสั่ง **Close** ใช้เมื่อต้องการออกจากการทำงานในหน้าจอนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

**ขั้นตอนการตั้งค่าในส่วนของ Host Link Setting**

1. เลือก Host Link Setting บนหน้าจอหลัก
2. เลือกค่าพารามิเตอร์ในการสื่อสาร สำหรับ รุ่น C200HS ให้เลือกตามภาพที่ 4.2
3. หลังจากเลือกครบทุกค่าให้เลือกปุ่ม Set
4. เลือก Exit เพื่อออกจากหน้าจอ Setting




**ภาพที่ 4.2** แสดงการตั้งค่าพารามิเตอร์ในการสื่อสารแบบHost Link

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

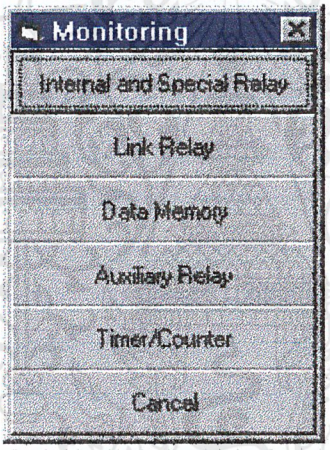
**ขั้นตอนการใช้งานในส่วน Host Link Monitoring**

โปรแกรมในส่วนนี้ใช้ในการตรวจดูค่าในพื้นที่ต่าง ๆ ของ PLC ดังนี้

- Internal and Special Relay
- Link Relay
- Data Memory
- Auxiliary Relay
- Timer and Counter

การเริ่มต้นใช้งาน

เริ่มจากเลือกพื้นที่ที่ต้องการจะให้แสดงค่า (Monitoring )



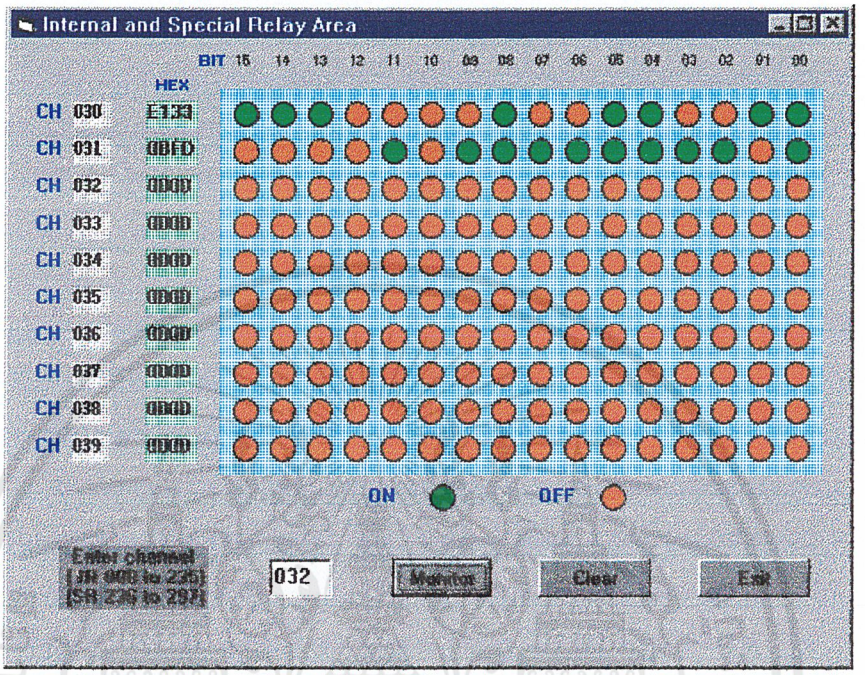

**ภาพที่ 4.3** แสดงหน้าจอสำหรับเลือกพื้นที่ที่ต้องการ

หน้าที่ของปุ่มคำสั่ง

1. **Internal and Special Relay** ใช้เพื่อเข้าไปดูค่าสถานะในพื้นที่ Internal Relayและ Special Relay
2. **Link Relay** ใช้เพื่อเข้าไปดูค่าสถานะในพื้นที่ Link Relay
3. **Data Memory** ใช้เพื่อเข้าไปดูค่าสถานะในพื้นที่ Data Memory
4. **Auxiliary Relay** ใช้เพื่อเข้าไปดูค่าสถานะในพื้นที่ Auxiliary Relay
5. **Timer / Counter** ใช้เพื่อเข้าไปดูค่าสถานะในพื้นที่ Timer and Counter
6. **Cancel** กลับไปหน้าจอหลัก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

**พื้นที่ Internal relay และ Special relay**



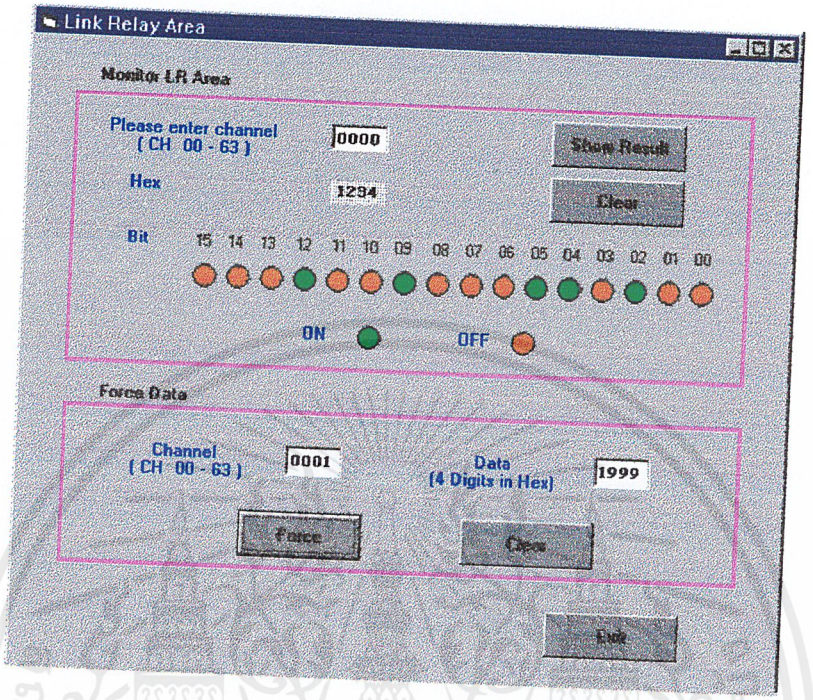

ภาพที่ 4.4 แสดงหน้าจอสำหรับแสดงค่าในพื้นที่ Internal Relay และ Special Relay

ขั้นตอนการดูค่าสถานะ

1. ใส่ค่า Channel ที่ต้องการให้แสดงค่าสถานะลงในช่องรับข้อมูล (Text Box)
   - Channel 000 – 235 จะเป็นพื้นที่ในส่วนของ Internal Relay
   - Channel 236 - 297 จะเป็นพื้นที่ในส่วนของ Special Relay
2. กดปุ่มคำสั่ง **Monitor** เพื่อแสดงค่าสถานะON–OFF และเลขฐานสิบหก โดยจะแสดงได้ครั้งละ 10 Channel
3. กดปุ่มคำสั่ง **Clear** เพื่อลบค่าสถานะเดิม และเตรียมรับค่าChannelที่ต้องการดูถัดไป
4. กดปุ่มคำสั่ง **Exit** เพื่อสิ้นสุดการทำงานในหน้าจอนี้ และกลับไปยังหน้าจอ Monitoring

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

**พื้นที่ Link Relay**




ภาพที่ 4.5 แสดงหน้าจอสำหรับแสดงค่าในพื้นที่ Link Relay

ขั้นตอนการดูค่าสถานะ( Monitor LR Area )

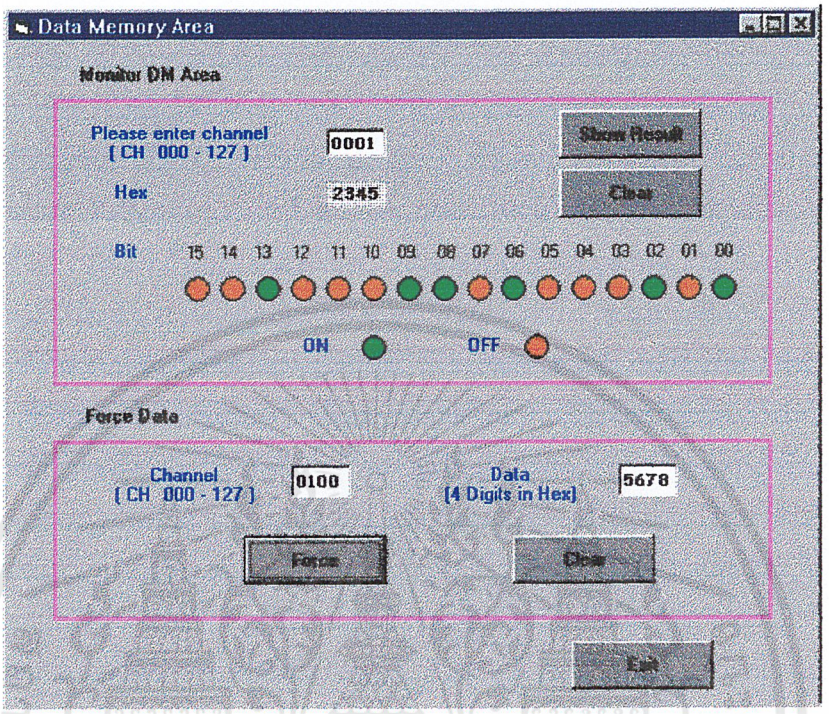
1. ใส่ค่า Channel ที่ต้องการให้แสดงค่าสถานะลงในช่องรับข้อมูล (Text Box)
2. กดปุ่มคำสั่ง **Show Result** เพื่อแสดงค่าสถานะON-OFF และเลขฐานสิบหก
3. กดปุ่มคำสั่ง **Clear** เพื่อลบค่าสถานะเดิม และเตรียมรับค่าChannelที่ต้องการดูถัดไป
4. กดปุ่มคำสั่ง **Exit** เพื่อสิ้นสุดการทำงานในหน้าจอนี้ และกลับไปยังหน้าจอ Monitoring

ขั้นตอนการกำหนดค่าสถานะ ( Force Data )

1. ใส่ค่า Channel ที่ต้องการกำหนดค่าสถานะลงในช่องรับข้อมูล (Text Box)
2. ใส่ค่าข้อมูลด้วยเลขฐานสิบหก ลงในช่องรับข้อมูล (Text Box)
3. กดปุ่มคำสั่ง **Force** เพื่อส่งค่าไปกำหนดสถานะในChannelที่ต้องการ หลังจากส่งค่าไปแล้วจะมีกล่องข้อความ ( Message Box ) แสดงสถานะการส่งโดยจะแสดงข้อความ "Forced data is Successful" เมื่อการส่งถูกต้อง และจะแสดงข้อผิดพลาดเมื่อการส่งผิดพลาด
4. กดปุ่มคำสั่ง **Clear** เพื่อลบค่าเดิมและเตรียมรับค่าChannel กับข้อมูลในการกำหนดค่าครั้งถัดไป
5. กดปุ่มคำสั่ง **Exit** เพื่อสิ้นสุดการทำงานในหน้าจอนี้ และกลับไปยังหน้าจอ Monitoring

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

**พื้นที่ Data Memory**




**ภาพที่ 4.6** แสดงหน้าจอสำหรับแสดงค่าในพื้นที่ Data Memory

ขั้นตอนการดูค่าสถานะ( Monitor DM Area )

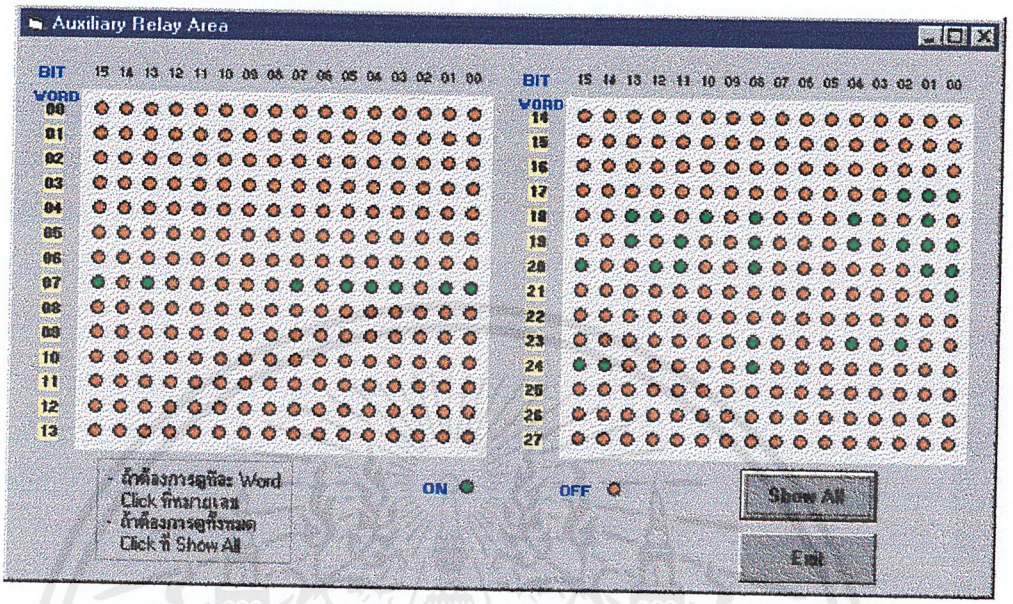
1. ใส่ค่า Channel ที่ต้องการให้แสดงค่าสถานะลงในช่องรับข้อมูล (Text Box)
2. กดปุ่มคำสั่ง **Show Result** เพื่อแสดงค่าสถานะ ON-OFF และเลขฐานสิบหก
3. กดปุ่มคำสั่ง **Clear** เพื่อลบค่าสถานะเดิม และเตรียมรับค่าChannelที่ต้องการดูถัดไป
4. กดปุ่มคำสั่ง **Exit** เพื่อสิ้นสุดการทำงานในหน้าจอนี้ และกลับไปยังหน้าจอ Monitoring

ขั้นตอนการกำหนดสถานะ ( Force Data )

1. ใส่ค่า Channel ที่ต้องการกำหนดสถานะลงในช่องรับข้อมูล (Text Box)
2. ใส่ค่าข้อมูลด้วยเลขฐานสิบหก ลงในช่องรับข้อมูล (Text Box)
3. กดปุ่มคำสั่ง **Force** เพื่อส่งค่าไปกำหนดค่าสถานะใน Channel ที่ต้องการ หลังจากส่งค่าไปแล้ว จะมีกล่องข้อความ ( Message Box ) แสดงสถานะการส่งโดยจะแสดงข้อความ “Forced data is Successful” เมื่อการส่งถูกต้อง และจะแสดงข้อผิดพลาดเมื่อการส่งผิดพลาด
4. กดปุ่มคำสั่ง **Clear** เพื่อลบค่าเดิมและเตรียมรับค่าChannel กับข้อมูลในการกำหนดค่าครั้งถัดไป
5. กดปุ่มคำสั่ง **Exit** เพื่อสิ้นสุดการทำงานในหน้าจอนี้ และกลับไปยังหน้าจอ Monitoring

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

**พื้นที่ Auxiliary Relay**



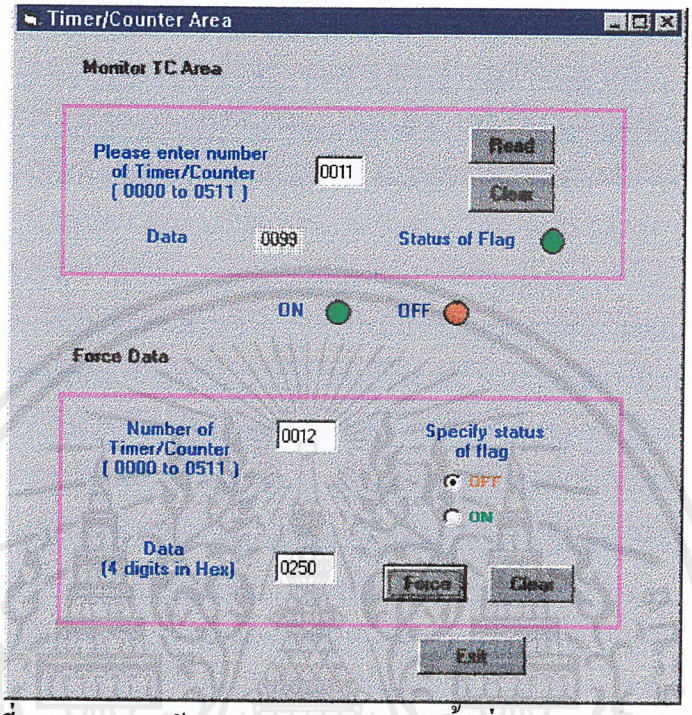

**ภาพที่ 4.7** แสดงหน้าจอการหาค่าในพื้นที่ Auxiliary Relay

ขั้นตอนการดูค่าสถานะ( Auxiliary Relay Monitoring )

1. คลิ้กแถบแสดงหมายเลขของ Word ที่ต้องการให้แสดงค่าสถานะ หลังจากการคลิ้กแถบหมายเลขนั้นจะเปลี่ยนเป็นแถบสี พร้อมทั้งมีการแสดงค่าสถานะผ่านหลอดไฟ ถ้าต้องการดูทั้งหมดให้เลือก **Show All**
   - สีเขียว สำหรับบิทที่มีสถานะ ON
   - สีแดง สำหรับบิทที่มีสถานะ OFF
2. กดปุ่มคำสั่ง **Exit** เพื่อสิ้นสุดการทำงานในหน้าจอนี้ และกลับไปยังหน้าจอ Monitoring

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

**พื้นที่ Timer และ Counter**




ภาพที่ 4.8 แสดงหน้าจอสำหรับแสดงค่าในพื้นที่ Timer และ Counter

ขั้นตอนการดูค่าสถานะ (Timer and Counter Monitoring)

1. ใส่ค่า Channel ที่ต้องการให้แสดงค่าสถานะลงในช่องรับข้อมูล (Text Box)
2. กดปุ่มคำสั่ง **Read** เพื่ออ่านค่าข้อมูล และแสดงค่าสถานะ ON-OFF โดยใช้หลอดไฟ
3. กดปุ่มคำสั่ง **Clear** เพื่อลบค่าสถานะเดิม และเตรียมรับค่าChannelที่ต้องการดูถัดไป
4. กดปุ่มคำสั่ง **Exit** เพื่อสิ้นสุดการทำงานในหน้าจอนี้ และกลับไปยังหน้าจอ Monitoring

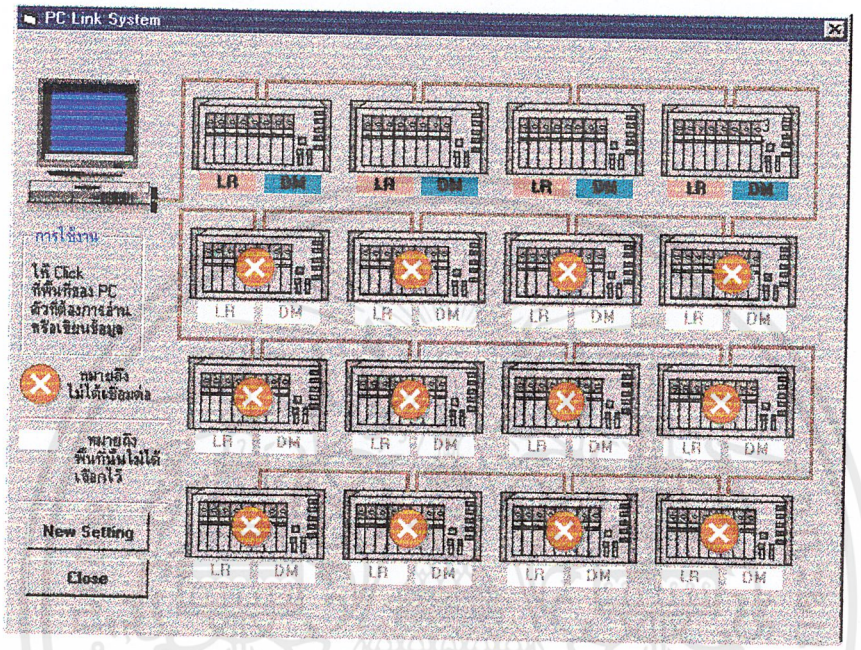
ขั้นตอนการกำหนดสถานะ (Force Data)

1. ใส่ค่า Channel ที่ต้องการกำหนดสถานะลงในช่องรับข้อมูล (Text Box)
2. ใส่ค่าข้อมูลด้วยเลขฐานสิบหก ลงในช่องรับข้อมูล (Text Box)
3. กำหนดค่าสถานะ ON-OFF โดยการคลิ้กเลือกที่ Option Bottom
4. กดปุ่มคำสั่ง **Force** เพื่อส่งค่าไปกำหนดค่าสถานะในChannelที่ต้องการ หลังจากส่งค่าไปแล้วจะมีกล่องข้อความ ( Message Box ) แสดงสถานะการส่งโดยจะแสดงข้อความ “Forced data is Successful” เมื่อการส่งถูกต้อง และจะแสดงข้อผิดพลาดเมื่อการส่งผิดพลาด
5. กดปุ่มคำสั่ง **Clear** เพื่อลบค่าเดิมและเตรียมรับค่าChannel กับข้อมูลในการกำหนดค่าครั้งถัดไป
6. กดปุ่มคำสั่ง **Exit** เพื่อสิ้นสุดการทำงานในหน้าจอนี้ และกลับไปยังหน้าจอ Monitoring

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

**ขั้นตอนการใช้งานส่วนของ PC Link Monitoring**

โปรแกรมในส่วนนี้จะช่วยให้สามารถเขียนและอ่านข้อมูลจาก Data Memory และ Link Relay ซึ่งเป็นพื้นที่ที่สามารถติดต่อถึงกันได้เมื่อ PLC มีการเชื่อมต่อกันแบบ PC Link




ภาพที่ 4.9 การเชื่อมต่อสายในระบบพีซีลิงค์

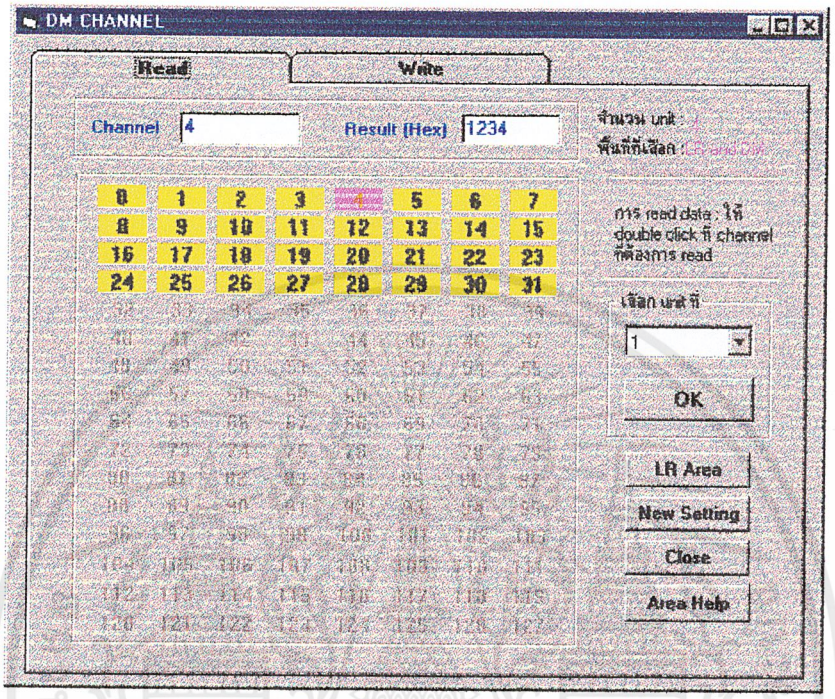
จากรูป

1. แสดงจำนวนพีซีทั้งหมดที่เชื่อมต่ออยู่ซึ่งได้สูงสุดเท่ากับ 16 ตัว
2. พีซีที่เราไม่ได้เซตให้มีการเชื่อมต่อจะมีเครื่องหมายกากบาททับ
3. พีซีที่ไม่มีเครื่องหมายกากบาททับคือจำนวนพีซีที่เราเซตให้มีการเชื่อมต่อทั้งหมดซึ่งเราสามารถเข้าไปมอนิเตอร์ดูค่าได้โดยคลิ้กที่พื้นที่ LR หรือ DM ของพีซีตัวที่เราต้องการ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

**การใช้งานในส่วนของ Data Memory (DM)**

การอ่านค่าในพื้นที่ DM




**ภาพที่ 4.10** แสดงหน้าจอสำหรับอ่านค่าในพื้นที่ Data Memory

ขั้นตอนการอ่านข้อมูล

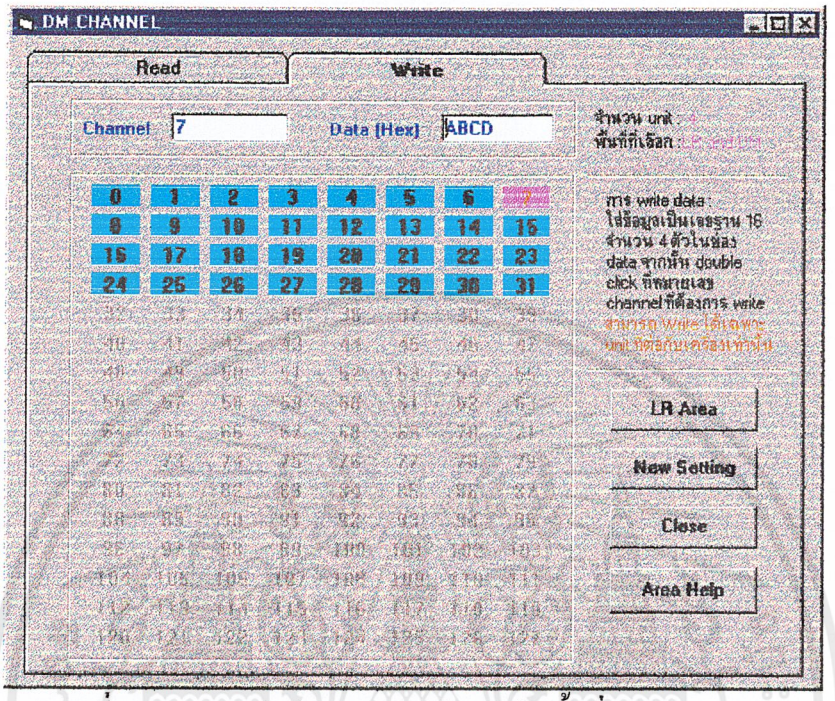
1. คลิ้กเลือกแท็บ **Read**
2. เลือก Unit ที่ต้องการอ่านค่าจาก คอมโบบอกซ์
3. กดปุ่มคำสั่ง **OK** แถบหมายเลขChannel ของ Unit ที่ต้องการอ่านค่าจะแสดงเป็นแถบสีให้สามารถเลือกได้
4. คลิ้กเลือกแถบหมายเลข Channel ที่ต้องการอ่านค่า จะปรากฏค่าของChannel ที่เลือกไปแสดงที่ ช่องแสดงข้อความ พร้อมกับข้อมูลที่อ่านมาได้จาก Channel นั้น ๆ

หน้าที่ของปุ่มคำสั่ง

1. ปุ่มคำสั่ง **LR Area** ใช้เพื่อไปยังหน้าจอของ Link Relay
2. ปุ่มคำสั่ง **New Setting** ใช้เพื่อกลับไปยังหน้าจอของการตั้งค่า ( Setting Part )
3. ปุ่มคำสั่ง **Close** ใช้เมื่อต้องการออกจากการทำงานในหน้าจอนี้
3. ปุ่ม **Area Help** จะเมื่อต้องการดูการจัดแบ่งพื้นที่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การใช้งานในส่วนของการเขียนข้อมูล ( Data Memory Write )




ภาพที่ 4.11 แสดงหน้าจอสำหรับเขียนค่าในพื้นที่ Data Memory

ขั้นตอนการเขียนข้อมูล

1. คลิ้กเลือกแท็บ **Write**
2. แถบหมายเลขChannel ที่สามารถเขียนได้ จะแสดงเป็นแถบสีให้สามารถเลือกได้
3. ใส่ค่าข้อมูลด้วยเลขฐานสิบหกลงในช่องรับข้อมูล (Text Box)
4. คลิ้กเลือกแถบหมายเลข Channel ที่ต้องการอ่านค่า จะปรากฏค่าของChannel ที่เลือกไปแสดงที่ ช่องแสดงข้อความ

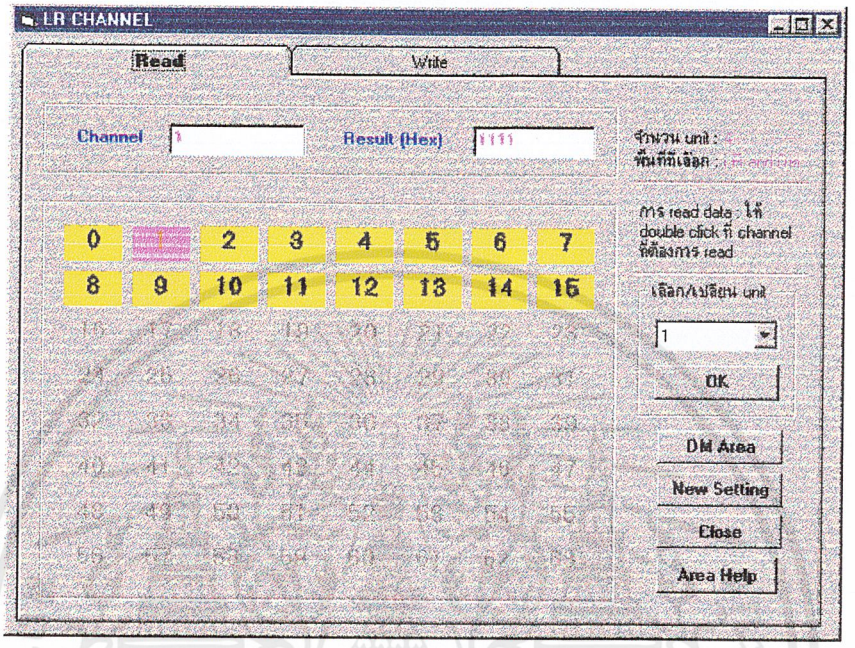
หน้าที่ของปุ่มคำสั่ง

1. ปุ่มคำสั่ง **LR Area** ใช้เพื่อไปยังหน้าจอของ Link Relay
3. ปุ่มคำสั่ง **New Setting** ใช้เพื่อกลับไปยังหน้าจอของการตั้งค่า ( Setting Part )
3. ปุ่มคำสั่ง **Close** ใช้เมื่อต้องการออกจากการทำงานในหน้าจอนี้
4. ปุ่มคำสั่ง **Area Help** ใช้เมื่อต้องการดูการจัดแบ่งพื้นที่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

**การใช้งานในส่วนของ Link Relay**

การใช้งานในส่วนของการอ่านข้อมูล (Link Relay Read)




**ภาพที่ 4.12** แสดงการอ่านค่าในพื้นที่ LR

ขั้นตอนการอ่านข้อมูล

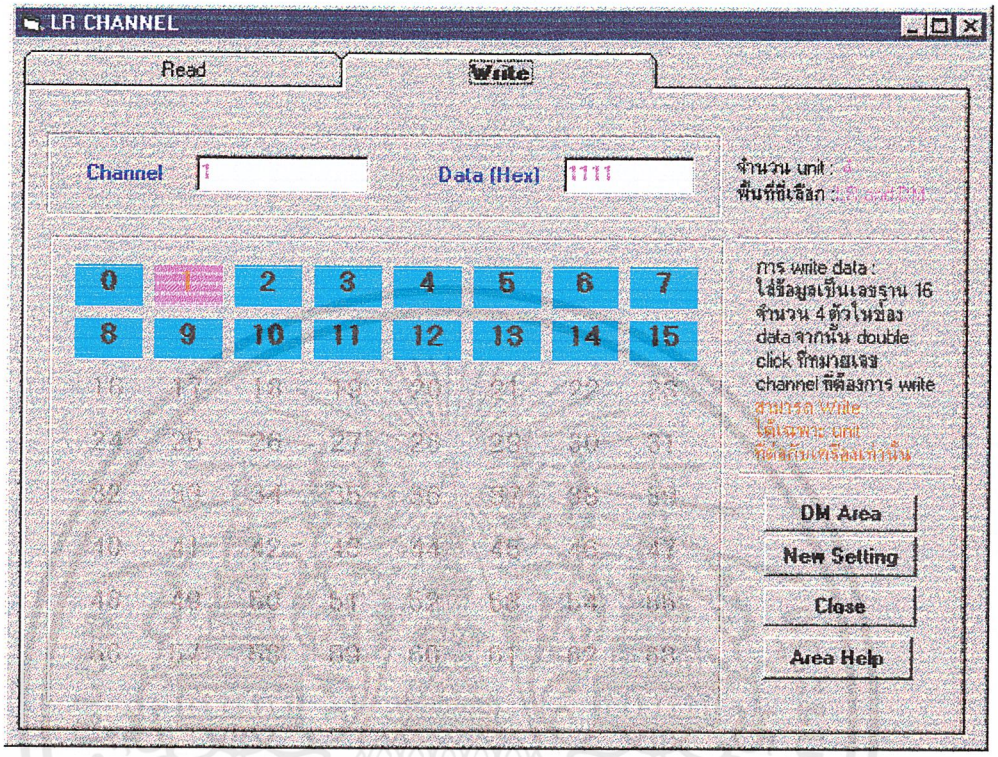
1. คลิ้กเลือกแท็บ **Read**
2. เลือก Unit ที่ต้องการอ่านค่าจาก คอมโบบอกซ์
3. กดปุ่มคำสั่ง **OK** แถบหมายเลขChannel ของ Unit ที่ต้องการอ่านค่าจะแสดงเป็นแถบสีให้สามารถเลือกได้
4. คลิ้กเลือกแถบหมายเลข Channel ที่ต้องการอ่านค่า จะปรากฏค่าของChannel ที่เลือกไปแสดงที่ ช่องแสดงข้อความ พร้อมกับข้อมูลที่อ่านมาได้จาก Channel นั้น ๆ

หน้าที่ของปุ่มคำสั่ง

1. ปุ่มคำสั่ง **DM Area** ใช้เพื่อไปยังหน้าจอของ Data Memory
4. ปุ่มคำสั่ง **New Setting** ใช้เพื่อกลับไปยังหน้าจอของการตั้งค่า ( Setting Part )
3. ปุ่มคำสั่ง **Close** ใช้เมื่อต้องการออกจากการทำงานในหน้าจอนี้
4. ปุ่มคำสั่ง **Area Help** ใช้เมื่อต้องการดูการจัดแบ่งพื้นที่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การใช้งานในส่วนของการเขียนข้อมูล (Link Relay Write )




ภาพที่ 4.13 แสดงการเขียนค่าลงในพื้นที่ LR

ขั้นตอนการเขียนข้อมูล

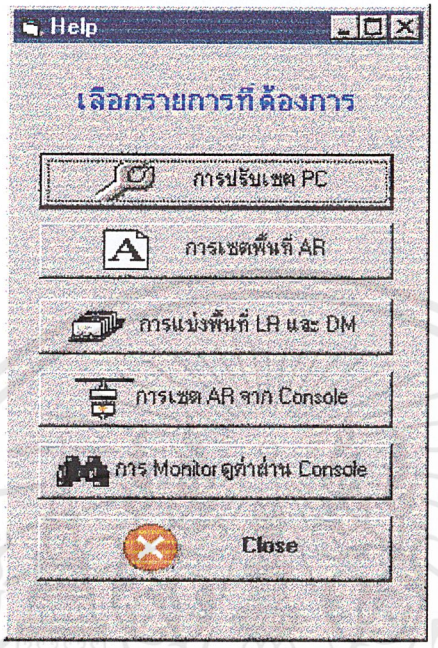
1. คลิ้กเลือกแท็บ **Write**
2. แถบหมายเลขChannel ที่สามารถเขียนได้ จะแสดงเป็นแถบสีให้สามารถเลือกได้
3. ใส่ค่าข้อมูลด้วยเลขฐานสิบหกลงในช่องรับข้อมูล (Text Box)
4. คลิ้กเลือกแถบหมายเลข Channel ที่ต้องการเขียน จะปรากฏค่าของChannel ที่เลือกไปแสดงที่ช่องแสดงข้อความ

หน้าที่ของปุ่มคำสั่ง

1. ปุ่มคำสั่ง **DM Area** ใช้เพื่อไปยังหน้าจอของ Data Memory
2. ปุ่มคำสั่ง **New Setting** ใช้เพื่อกลับไปยังหน้าจอของการตั้งค่า ( Setting Part )
3. ปุ่มคำสั่ง **Close** ใช้เมื่อต้องการออกจากการทำงานในหน้าจอนี้
4. ปุ่มคำสั่ง **Area Help** ใช้เมื่อต้องการดูการจัดแบ่งพื้นที่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

**การใช้งานในส่วนของ Help**




ภาพที่ 4.13 หน้าจอหลักของ Help

หน้าที่ของปุ่มคำสั่ง

1. **การปรับเซตPC** แสดงการปรับเซตทางด้านฮาร์ดแวร์สำหรับการเชื่อมต่อ เช่น การปรับ DIP SWITCH, การปรับ SWITCH ของแต่ละโหนด และการตรวจสอบไฟสถานะต่างๆ
2. **การเซตพื้นที่AR** หน้าจอแสดงการเซตค่าลงบนพื้นที่ AR 07 ให้พีซีรู้ว่าจะเชื่อมต่อกันทั้งหมดกี่ตัว เพื่อจะได้จัดแบ่งพื้นที่ให้ได้
3. **การแบ่งพื้นที่ LR และ DM** แสดงการแบ่งพื้นที่ของ LR และ DM ตามจำนวนยูนิตที่เราต้องการเชื่อมต่อ
4. **การเซตARจากConsole** แสดงวิธีการเซตค่าลงบนพื้นที่ AR 07 โดยการเซตผ่านคอนโซล
5. **การMonitorผ่านConsole** แสดงวิธีการตรวจสอบค่าผ่านคอนโซล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

# บทที่ 5
# บทสรุปและแนวทางในการพัฒนาต่อ

## 5.1 แนวทางในการพัฒนาต่อ

จากโปรแกรมที่ได้สร้างมา เป็นการสนับสนุนการเชื่อมต่อ PC ของ OMRON รุ่นC200HS ดังนั้นสามารถที่จะนำไปประยุกต์ใช้กับ PC รุ่นอื่นๆได้ด้วย นอกจากนี้หากเรารู้โค้ดคำสั่งจากบริษัทผู้ผลิตอาจพัฒนาโปรแกรมให้สามารถรับค่าเป็นบูลีน ( Boolean ) เพื่อส่งต่อไปยัง PC หรือการเพิ่มคำสั่งอื่นๆ นอกเหนือจากคำสั่งการอ่านและเขียน และสามารถที่จะบันทึกข้อมูลในแต่ละ Unit ตามระยะเวลาที่เรากำหนดได้ว่าแต่ละขณะมีการเปลี่ยนแปลงข้อมูลอย่างไร ตลอดจนการพัฒนาไปสู่การเชื่อมต่อในระบบแลน ( LAN ) เพื่อให้สามารถตรวจสอบสภาวะการทำงาน หรือการควบคุม PC แต่ละตัวที่อยู่ห่างออกไปจากศูนย์กลางการควบคุมเพียงแห่งเดียว

## 5.2 ข้อจำกัดของโครงการ

เนื่องจาก PC ที่ใช้ทำโครงงาน สามารถเขียนข้อมูลลงไปได้เฉพาะตัวที่ต่อกับเครื่องคอมพิวเตอรีเท่านั้น ไม่สามารถเขียนข้ามไปยังตัวอื่นๆได้ นอกจากนี้รูปแบบของคำสั่งจากบริษัทผู้ผลิตที่ให้มาก็มีเพียงการอ่านและเขียนข้อมูลเท่านั้น แต่ไม่ได้ให้คำสั่งรูปแบบอื่นๆ เช่น การเคลื่อนย้ายข้อมูล, การตั้งเวลา ฯลฯ ทำให้การควบคุม PC ในรูปแบบอื่นๆ ผ่านเครื่องคอมพิวเตอร์ไม่สามารถทำได้รวมทั้งตำราเกี่ยวกับ PC ยังมีให้ศึกษาน้อยทำให้มุมมองต่อการนำ PC ไปประยุกต์ใช้มีค่อนข้างจำกัด

## 5.3 บทสรุป

ในระบบอุตสาหกรรมในปัจจุบัน มีการใช้ระบบควบคุมแบบอัตโนมัติเป็นส่วนใหญ่ PC จึงเข้ามามีบทบาทมาก และในโรงงานอุตสาหกรรมแต่ละแห่งนั้นจะมี PC อยู่เป็นจำนวนมากดังนั้นการจะควบคุม PC แต่ละตัวโดยตรงนั้นคงทำได้ยาก ต้องใช้คนจำนวนมากทีเดียวจึงจะสามารถควบคุมได้ทั่วถึง ทำให้เสียเวลาและค่าใช้จ่ายมาก การนำ PC มาเชื่อมต่อกันเป็นระบบโครงข่ายโดยมีศูนย์กลางการควบคุมเพียงแห่งเดียวจะทำให้ประหยัดเวลาและค่าใช้จ่ายลงได้มากทีเดียว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

# กิตติกรรมประกาศ

ปริญญานิพนธ์ฉบับนี้คงไม่สามารถสำเร็จลุล่วงไปได้ ถ้าไม่ได้รับความช่วยเหลือจาก อาจารย์ทวีพล ซื่อสัตย์ และ อาจารย์ท่านอื่น ๆ ที่คอยให้คำปรึกษาและช่วยแก้ไขปัญหาต่าง ๆ ด้วยดีตลอดมา

ขอขอบคุณ พ่อและแม่ ที่คอยเป็นกำลังใจ และให้ทุนทรัพย์ในการทำโครงงานนี้

ขอขอบคุณ อาจารย์ ไสว พงศ์สวัสดิ์ ที่ได้เอื้อเฟื้อ Printer

ขอขอบคุณ เพื่อน ๆ นักศึกษาภาควิชาเทคโนโลยีการวัดคุมทางอุตสาหกรรม ที่คอยให้คำปรึกษาพร้อมทั้งกำลังใจ

สุดท้ายขอขอบคุณภาควิชา เทคโนโลยีการวัดคุมทางอุตสาหกรรม สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง ที่ได้เอื้อเฟื้อในเรื่องเครื่องมือและอุปกรณ์ต่าง ๆ ในการทำโครงงานนี้ให้สำเร็จลุล่วงไปด้วยดี

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

# บรรณานุกรม


1. ธาริน สิทธิธรรมชารี , ธนัญชัย จำนงค์ภักดี , "Microsoft Visual Basic Version 5.0" , บริษัท ซีเอ็ดยูเคชั่น จำกัด ( มหาชน )
2. สุทธิศักดิ์ พงศ์ธนาพานิช , "Visual Basic 5.0 Professional การใช้คำสั่งและคอนโทรล ActiveX" , บริษัท ซีเอ็ดยูเคชั่น จำกัด ( มหาชน )
3. กิตติ ภักดีวัฒนะกุล , ทำลอง ครูอุตสาหะ , "Visual Basic 5 ฉบับ โปรแกรมเมอร์" ,บริษัท ดวงกมลสมัย จำกัด
4. ณรงค์ ตันชีวะวงศ์ , "ระบบ PLC ( Programmable Logic Controller )" , สมาคมส่งเสริมเทคโนโลยี ( ไทย – ญี่ปุ่น )
5. ภาควิชาเทคโนโลยีการวัดคุมทางอุตสาหกรรม , "OMRON FA 2 ADVANCED PLC WITH C200HS", คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
6. OMRON ELECTRONICS CO.,LTD , "ADVANCED OMRON FA2 TRAINING COURSE" , คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
7. OMRON JAPAN , "C200HS Programmable Controllers Operation Manual" , 1995
8. OMRON JAPAN , "SYSMAC Programmable Controllers C200HS Installation Guide", 1995


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้