

การพัฒนาระบบสารสนเทศสำหรับเครื่องจัดเก็บผลิตภัณฑ์แบบอัตโนมัติ
DEVELOPMENT OF INFORMATION SYSTEM FOR AUTOMATIC
WAREHOUSE



นายจตุพร รอดคำทวย
นายพิศาล สุนเลี้ยง
นายศักดา สะโดอยู่

เลขหม.....
เลขทะเบียน 45846
วัน, เดือน, ปี 19 ก.พ. 2546

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

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต

สาขาวิชาวิศวกรรมการวัดคุม

ภาควิชาวิศวกรรมการวัดคุม คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2544

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้ง
๒๕๔๖

**DEVELOPMENT OF INFORMATION SYSTEM FOR AUTOMATIC
WAREHOUSE**



**A THESIS SUBMITTED IN PARTIAL FULFILLMENT
OF THE REQUIREMENT FOR THE DEGREE OF
BACHELOR OF ENGINEERING IN INSTRUMENTATION ENGINEERING
DEPARTMENT OF INSTRUMENTATION ENGINEERING
FACULTY OF ENGINEERING
KING MONGKUT'S INSTITUTE OF TECHNOLOGY LADKRABANG**

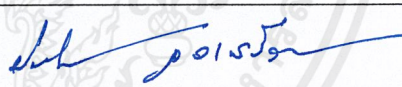
2001

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาควิชาวิศวกรรมการวัดคุม
คณะวิศวกรรมศาสตร์
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
ใบรับรองวิทยานิพนธ์

หัวข้อวิทยานิพนธ์ การพัฒนาระบบสารสนเทศสำหรับเครื่องจัดเก็บผลิตภัณฑ์แบบอัตโนมัติ
DEVELOPMENT OF INFORMATION SYSTEM FOR AUTOMATIC
WAREHOUSE

นักศึกษาผู้จัดทำ นายจตุพร รอดคำทวย รหัสประจำตัว 42015425
นายพิศาล สุนเลี้ยง รหัสประจำตัว 42015445
นายศักดา สะโคอยู่ รหัสประจำตัว 42015453
ปริญญา วิศวกรรมศาสตรบัณฑิต
สาขาวิชา วิศวกรรมการวัดคุม
ปีการศึกษา 2544

| อาจารย์ผู้ควบคุมวิทยานิพนธ์ | ลายมือชื่อ |
|--|--|
| อาจารย์ทวีพล ชื่อสัตย์ อาจารย์อาจินต์ น่วมสำราญ |  |

วัน/เดือน/ปี ที่สอบ วันจันทร์ ที่ 25 มีนาคม พ.ศ. 2545
สถานที่สอบ ณ ห้องสอบวิทยานิพนธ์ ภาควิชาวิศวกรรมการวัดคุม

ภาควิชารับรองแล้ว



(ผศ. ประสิทธิ์ จุลเสรีวงศ์)

หัวหน้าภาควิชาวิศวกรรมการวัดคุม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

| | | |
|--------------------|--|--|
| หัวข้อปริญญานิพนธ์ | การพัฒนาาระบบสารสนเทศสำหรับเครื่องจัดเก็บผลิตภัณฑ์แบบอัตโนมัติ | |
| | DEVELOPMENT OF INFORMATION SYSTEM FOR AUTOMATIC WAREHOUSE | |
| นักศึกษาจัดทำ | นายจตุพร รอดคำทวย | |
| | นายพิศาล โครงงานยง | |
| | นายศักดิ์ดา สะโคอยู่ | |
| อาจารย์ที่ปรึกษา | อาจารย์ทวีพล ชื้อสัตย์ | |
| | อาจารย์อาจินต์ น่วมสำราญ | |
| ปีการศึกษา | 2544 | |

บทคัดย่อ

ปริญญานิพนธ์การพัฒนาาระบบสารสนเทศสำหรับเครื่องจัดเก็บผลิตภัณฑ์ แบบอัตโนมัติ เป็นการศึกษารูปแบบของ โครงสร้างและการจัดเก็บวัสดุ เพื่อให้ประหยัดพื้นที่ในการจัดเก็บและ มีความรวดเร็วในการค้นหา ด้วยการนำ PLC มาควบคุมกระบวนการให้มีการทำงาน ตามต้องการ ปริญญานิพนธ์นี้ ได้ทำการจำลองกระบวนการที่ใช้งานทั่วไป ในคลังจัดเก็บสินค้าอัตโนมัติซึ่ง ประกอบด้วยมอเตอร์กระแสสลับจำนวน 3 ตัว สำหรับขับเคลื่อนชิ้นส่วนในการจัดเก็บสินค้า และ ส่วนเคลื่อนย้ายตำแหน่งในการจัดเก็บสินค้า การส่งงานจะเป็นแบบอัตโนมัติ จากโปรแกรม คอมพิวเตอร์ที่มีการจัดการฐานข้อมูล ผ่านพอร์ตอนุกรม RS 232C ในการจัดเก็บสินค้าแต่ละครั้ง จะเป็นการเก็บ ประเภทสินค้า วัน/เดือน/ปี เวลาที่จัดเก็บ ซึ่งจะบันทึกลงในฐานข้อมูล โดย สามารถแบ่งการใช้ฐานข้อมูลร่วมกัน ระหว่างฝ่ายต่างๆ ในองค์กรผ่านระบบเครือข่ายท้องถิ่น(LAN) เพื่อให้เกิดประสิทธิภาพในการการบริหารวัสดุคงคลัง และการจัดเก็บสินค้าเหล่านี้ทำการจัดเก็บ แบบ (Fisrt In –First Out) เพื่อเป็นการป้องกันสินค้าที่เข้ามาเก็บก่อนถูกเก็บไว้นานจนเสียหาย และ เป็นการนำหลักการบริหารคลังสินค้า (Inventory Management) มาใช้ในเรื่องของพื้นที่ในการ ใช้สอยให้เกิดประโยชน์สูงสุด เพื่อลดต้นทุนและค่าใช้จ่ายในการจัดเก็บ

| | | |
|-----------------------|---|-------------|
| Thesis Title | Development of Information System for Automatic Warehouse | |
| Author | Mr. Jatuporn | Rodcumtui |
| | Mr. Pisan | Soonlieng |
| | Mr.Sakda | Sadoyoo |
| Thesis Adviser | Mr.Taweepol | Suesut |
| | Mr.Arjin | Noumsumlann |
| Year | 2001 | |

ABSTRACT

This thesis presents the development of information system for automatic warehouse. To decrease lead-time and economization is very nessesary in industrial management. Therefore, this project has developed the information system as well as the automatic warehouse machine in order to decrease leadtime and increase network connectibity also. The automatic warehouse consist of 3 set of AC moter for drive 3 axis of mechanical system that controlled by Programmable Logic Controller : PLC. The method to order and storge any product inform of FIFO can be done automatically by computer software which connected to PLC through RS232C Serial commucation. The software base on Microsoft Windows used for controling warehouse machine and also sharing database passing computer network to department.

กิตติกรรมประกาศ

ปริญญาานิพนธ์ฉบับนี้สำเร็จได้ด้วยดีเพราะได้รับความเมตตาจาก อาจารย์ทวีพล ชื่อดัตต์ และอาจารย์อาจินต์ น่วมสำราญ ที่ได้ให้คำแนะนำแก่ผู้ทำปริญญาานิพนธ์ตลอดมา อีกทั้งยังเอื้อเพื่อ อุปกรณ์และเครื่องมือต่างๆในการทำปริญญาานิพนธ์นี้ผู้จัดทำรู้สึกซาบซึ้ง และขอกราบขอบพระคุณ เป็นอย่างสูง

ขอขอบพระคุณ รศ.สุพรรณ กุลพานิชย์ และอาจารย์ภาควิชาวิศวกรรมการวัดคุมทุกท่าน ที่ได้ให้คำแนะนำอันเป็นประโยชน์ต่อการทำปริญญาานิพนธ์ฉบับนี้

และที่ลืมเสียมิได้คือ ขอกราบขอบพระคุณ พ่อ แม่ ย่า อา อันเป็นที่รักยิ่ง ที่สนับสนุนและเป็นแรงบันดาลใจในการทำปริญญาานิพนธ์ฉบับนี้

คุณค่าและประโยชน์อันพึงมีจากปริญญาานิพนธ์ฉบับนี้ ผู้จัดทำขอบแต่ผู้มีพระคุณ ทุกท่าน

คณะผู้จัดทำ

สารบัญ

| | หน้า |
|--|----------|
| บทคัดย่อภาษาไทย..... | I |
| บทคัดย่อภาษาอังกฤษ..... | II |
| กิตติกรรมประกาศ..... | III |
| สารบัญ..... | IV |
| สารบัญตาราง..... | VI |
| สารบัญรูป..... | VII |
| | |
| บทที่1 บทนำ..... | 1 |
| 1.1 ที่มาของปริญญาบัตร..... | 1 |
| 1.2 วัตถุประสงค์ของปริญญาบัตร..... | 1 |
| 1.3 ขอบเขตของปริญญาบัตร..... | 1 |
| 1.4 รายละเอียดของปริญญาบัตร..... | 1 |
| | |
| บทที่2 หลักการและทฤษฎี..... | 2 |
| 2.1 เครื่องควบคุมแบบโปรแกรมได้..... | 2 |
| 2.1.1 Introduction to PLC..... | 2 |
| 2.1.2 โปรโตคอลในการติดต่อสื่อสาร..... | 3 |
| 2.1.3 รูปแบบของบล็อค..... | 4 |
| 2.1.4 คำสั่งต่างๆที่ใช้ในการส่งข้อมูล..... | 5 |
| 2.1.5 คำสั่งต่างๆที่ใช้ในการรับข้อมูล..... | 5 |
| 2.1.6 การคำนวณ Frame Checksum(FCS)..... | 6 |
| 2.1.7 High Speed Counter..... | 7 |
| 2.1.8 การ Setup Datamemory C28H Back Up..... | 9 |
| 2.2 การจัดการฐานข้อมูลโดย MS-Visual Basic และ MS-Access..... | 9 |
| 2.2.1 Microsoft Visual Basic..... | 9 |
| 2.2.2 Microsoft Access..... | 16 |
| 2.2.3 การจัดการฐานข้อมูล..... | 17 |
| 2.3 การติดต่อสื่อสาร..... | 32 |

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ(ต่อ)

| | หน้า |
|---|-----------|
| 2.3.1 พอร์ตอนุกรม..... | 32 |
| 2.3.2 คอนโทรล MSCComm | 35 |
| 2.4 การจัดการระบบเครือข่าย..... | 38 |
| 2.4.1 อุปกรณ์สำคัญที่ต้องใช้ในระบบเน็ตเวิร์ค..... | 38 |
| 2.4.2 วิธีการต่อสายLAN..... | 41 |
| 2.5 การเขียนโปรแกรมอำนวยความสะดวกในระบบเครือข่าย..... | 42 |
| บทที่3 รายละเอียดในการสร้างและประกอบโครงงาน..... | 48 |
| 3.1 ฮาร์ดแวร์..... | 48 |
| 3.1.1 Warehouse..... | 48 |
| 3.1.2 การใช้งาน Driver Unidrive | 50 |
| บทที่4 การทดลองโครงงาน..... | 54 |
| 4.1 ทดสอบโปรแกรม..... | 54 |
| 4.2 ทดสอบฐานข้อมูล..... | 63 |
| บทที่5 สรุปผลและแนวทางการพัฒนา..... | 70 |
| 5.1 สรุปการทำงานของปริณญาณิพนธ์..... | 70 |
| 5.2 แนวทางพัฒนา..... | 70 |
| บรรณานุกรม..... | 71 |

สารบัญตาราง

| ตารางที่ | หน้า |
|--|------|
| 2.1 แสดง Data and Memory areas..... | 2 |
| 2.2 แสดงการเรียงสายสัญญาณแบบธรรมดา..... | 41 |
| 2.3 แสดงการเรียงสายสัญญาณสำหรับการต่อแบบไขว้สาย..... | 42 |
| 2.4 แสดงการเรียงสายสัญญาณสำหรับการต่อแบบไขว้สาย(ของอีกด้าน)..... | 42 |
| 4.1 เป็นตารางที่บันทึกค่าการนำเข้า – ออกของผลิตภัณฑ์ทุกชนิด..... | 66 |
| 4.2 แสดงปริมาณผลิตภัณฑ์ที่ 1 มีอยู่บน Warehouse ในปัจจุบัน..... | 67 |
| 4.3 แสดงปริมาณผลิตภัณฑ์ที่ 2 มีอยู่บน Warehouse ในปัจจุบัน..... | 67 |
| 4.4 แสดงปริมาณผลิตภัณฑ์ที่ 3 มีอยู่บน Warehouse ในปัจจุบัน..... | 68 |
| 4.5 แสดงรายละเอียดเกี่ยวกับผลิตภัณฑ์แต่ละชนิด..... | 68 |
| 4.6 แสดงข้อมูลเกี่ยวกับรหัสแถบและชนิดผลิตภัณฑ์..... | 69 |

สารบัญรูป

| รูปที่ | หน้า |
|---|------|
| 2.1 คำสั่งต่างๆที่ใช้ในการส่งข้อมูล..... | 5 |
| 2.2 คำสั่งต่างๆที่ใช้ในการรับข้อมูล..... | 6 |
| 2.3 การคำนวณ FCS..... | 6 |
| 2.4 The overall operation of the high speed counter..... | 7 |
| 2.5 การต่อเน็ตเวิร์คแบบบัส..... | 38 |
| 2.6 ขั้วต่อแบบ BNC..... | 39 |
| 2.7 การต่อวงจรเครือข่ายแบบ Star..... | 39 |
| 2.8 ขั้วต่อแบบ RJ-45..... | 40 |
| 3.1 warehouse ที่ใช้ในโครงการ..... | 48 |
| 3.2 การควบคุมระยะด้วย Encoder..... | 49 |
| 3.3 ตู้ควบคุมและ Touch Screen..... | 49 |
| 3.4 รูปหน้า KEY PAD ของ UNI1404..... | 52 |
| 4.1 แสดงหน้าจอเกี่ยวกับโปรแกรม..... | 54 |
| 4.2 แสดงการเลือก โหมดการทำงาน..... | 55 |
| 4.3 แสดงการทำงาน โหมดอัตโนมัติ..... | 56 |
| 4.4 แสดง การทำงาน โหมด Manual..... | 57 |
| 4.5 แสดงหน้าจอที่ใช้จัดสถานะของผลิตภัณฑ์..... | 58 |
| 4.6 แสดงผลลัพธ์ที่เปลี่ยนในการเปลี่ยนค่าเครื่องควบคุม..... | 59 |
| 4.7 การตรวจสอบปริมาณการจัดเก็บ-นำออกของผลิตภัณฑ์..... | 60 |
| 4.8 แสดงปริมาณผลิตภัณฑ์ในปัจจุบัน..... | 61 |
| 4.9 แสดงฐานข้อมูลเกี่ยวกับผลิตภัณฑ์..... | 62 |
| 4.10 แสดงรายละเอียดเกี่ยวกับผลิตภัณฑ์..... | 63 |
| 4.11 เครื่องจัดเก็บผลิตภัณฑ์อัตโนมัติขณะอยู่ที่ตำแหน่งเริ่มต้น..... | 64 |
| 4.12 เครื่องจัดเก็บผลิตภัณฑ์อัตโนมัติขณะอยู่ที่ตำแหน่งเริ่มต้น..... | 65 |
| 4.13 เครื่องจัดเก็บผลิตภัณฑ์อัตโนมัตินำผลิตภัณฑ์เก็บหรือนำออก..... | 65 |
| 4.14 เครื่องจัดเก็บผลิตภัณฑ์อัตโนมัตินำผลิตภัณฑ์เก็บหรือนำออก..... | 66 |

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 1

บทนำ

1.1 ความเป็นมาและเหตุจูงใจของการวิจัย

ในปัจจุบันเทคโนโลยีของเครื่องควบคุมอัตโนมัติมีความสามารถสูงขึ้น ทั้งในด้านการควบคุมที่แม่นยำ ความเร็วในการประมวลผล นอกจากนี้ยังถูกพัฒนาความสามารถในด้านการสื่อสารข้อมูล ทำให้สามารถเชื่อมต่อกับอุปกรณ์ต่างๆ ได้อย่างหลากหลายจากพื้นฐานการนำเทคโนโลยีสารสนเทศมาประยุกต์ใช้ในการบริหารจัดการในองค์กรธุรกิจ (Management Information System; MIS) เพื่อลดเวลานำ (Lead Time) ทำให้การดำเนินงานต่างๆ รวดเร็วยิ่งขึ้น ในปฏิญญานิพนธ์นี้ได้นำเทคโนโลยีสารสนเทศมาเชื่อมโยงเครือข่ายคอมพิวเตอร์ในองค์กรธุรกิจ ไปจนถึงกระบวนการผลิตคือตัวระบบควบคุมอัตโนมัติเพื่อเพิ่มความรวดเร็ว และถูกต้องมากยิ่งขึ้น โดยนำระบบควบคุมคลังพัสดุอัตโนมัติมาเป็นกรณีศึกษา

1.2 วัตถุประสงค์ของปฏิญญานิพนธ์

1. เพื่อออกแบบระบบจัดเก็บสินค้าคงเหลืออัตโนมัติด้วยเครื่องควบคุมที่โปรแกรมได้
2. เพื่อออกแบบระบบเครือข่ายคอมพิวเตอร์และการเชื่อมต่อโยงฐานข้อมูลร่วมกับเครื่องควบคุมที่โปรแกรมได้

1.3 ขอบเขตของปฏิญญานิพนธ์

ในปฏิญญานิพนธ์นี้ได้ทำการศึกษาถึง การนำระบบสารสนเทศมาใช้กับเครื่องจัดเก็บผลิตภัณฑ์อัตโนมัติ โดยทำการใช้ระบบแลนในการติดต่อในหน่วยงาน และเขียนโปรแกรมเพื่ออำนวยความสะดวกในการใช้งานเครื่องจัดเก็บผลิตภัณฑ์อัตโนมัติภายในหน่วยงาน

1.4 รายละเอียดของปฏิญญานิพนธ์

ในปฏิญญานิพนธ์นี้ได้ทำ

1. เขียนโปรแกรมแลตเตอร์ควบคุมการทำงานของเครื่องจัดเก็บผลิตภัณฑ์อัตโนมัติ
2. เขียนโปรแกรมการจัดการฐานข้อมูลในการจัดเก็บและการนำออกของผลิตภัณฑ์
3. เขียนโปรแกรมอำนวยความสะดวกในการใช้งานติดต่อกับเครื่องจัดเก็บผลิตภัณฑ์อัตโนมัติ
- เขียนโปรแกรมควบคุมการทำงาน โดย PT
5. ติดตั้งระบบแลนเข้ากับระบบจัดเก็บผลิตภัณฑ์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 2

หลักการและทฤษฎี

2.1 เครื่องควบคุมแบบโปรแกรมได้

2.1.1 Introduction to PLC

OMRON C28H C6DR-DE

Power Supply : 24 VDC

Inputs : 16,24 VDC input

2 commons

One with 2 pts

One with 4 pts

Outputs : 12 relay output

With sockets

6 commons

4 with 1 pt , each 2 with 4 pts

Memory : RAM , IC socket , EEPROM

Calander/clock : Built in

Data and Memory areas

ตารางที่ 2.1 แสดง Data and Memory areas

| Area | Acronym | Range | Function |
|-----------------|---------|---|--|
| Internal Relay | IR | Words: 000 to 246 Bits: 0000 to 24615 | Used to control I/O points, other bits, timers, and counters, and to temporarily store data. |
| Special Relay | SR | Words: 247 to 255 Bits: 247000 to 25507 | Contains system clocks, flags, control bits, and Status information. |
| Auxiliary Relay | AR | Words: AR00 to AR27 Bits: AR00 to AR2715 | Contains flags and bits for special functions. Retains status during power failure. |
| Data Memory | DM | Read/write: DM0000 to DM0999 Read only: DM1000 to DM1999 | Used for internal data storage and manipulation. |

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

| Area | Acronym | Range | Function |
|-----------------|---------|--|--|
| Holding Relay | HR | Word: HR00 to HR99 Bits: HR0000toHR9915 | Used to store data and retain the data values when the power to the PC is turned off. |
| Timer/ Counter | TC | TC000 to TC511 (TC numbers used to access other information) | Used to define timers and counters, and to access completion flags, PV, SV. In general, when used as a bit operand, a TC number accesses the completion flag for the timer or counter defined using the TC number .When used as a word operand, the TC number accesses the present value of the timer or counter |
| Link Relay | LR | Words: LR00 to LR63 Bits: LR0000 to 6315 | Available for use as work bits. |
| Temporaly Relay | TR | TR00 to 07(bits only) | Used to temporarily store and retrieve execution conditions. These bits can only be used in the Load and Output instructions is necessary when programming certain types of branching ladder diagrams. |
| Program Memory | UM | UM:Depends on Memory Unit used. | Contains the program executed by the CPU |

2.1.2 โพรโตคอลในการติดต่อสื่อสาร(Communications Protocol)

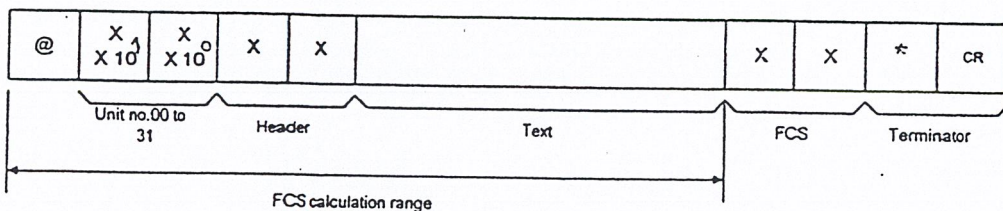
Host Computer จะมีลำดับความสำคัญเป็นตัวเริ่มต้นในการส่งผ่านข้อมูล เมื่อมีการส่งผ่านข้อมูลระหว่าง Host Computer กับ Host Link System ก่อนที่จะมีการส่งผ่านข้อมูลได้ Host Computer จะต้องส่งคำสั่งไปยังเครื่องควบคุมแบบโปรแกรมได้ใน Host Link System ก่อน

ชุดของข้อมูลในการส่งผ่านข้อมูลนั้นเรียกว่า บล็อก(Block) บล็อกของข้อมูลที่ส่งจาก Host Computer ไปยัง Host Link System เรียกว่าบล็อกชุดคำสั่ง (Command Block) ส่วนบล็อกที่ส่งมาจาก Host Link System ไปยัง Host Computer เรียกว่า บล็อกชุดตอบสนอง(Response Block)

ในระบบการติดต่อสื่อสารที่มีการเชื่อมโยงหลายๆจุด แต่ละบล็อกจะเริ่มต้นด้วยค่าหมายเลข Unit และส่วนหัว(Header) และจบท้ายด้วยรหัส Frame Checksum (FCS) และตัวสิ้นสุด (*CR) เมื่อCR ย่อมาจาก Carrier Return(สัญญาณเคาะแป้น)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

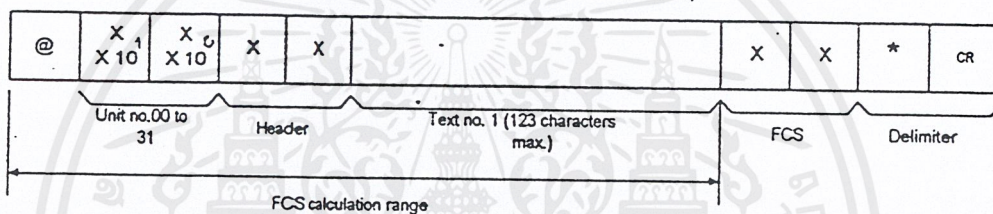
2.1.3 รูปแบบของบล็อค



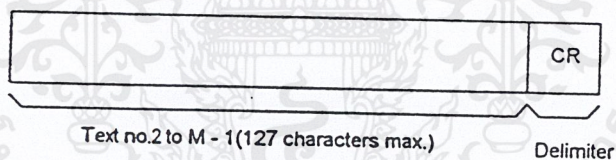
ปกติแล้วในหนึ่งบล็อคจะสร้างขึ้นมาจากหนึ่ง Unit ที่เรียกว่าเฟรม (Frame) แต่ถ้าบล็อคนั้นมีข้อมูลมากกว่า 131 อักขระขึ้นไป บล็อคนั้นจะถูกแบ่งให้มีมากกว่า 1 เฟรมและในเฟรมย่อยสามารถมีข้อมูลได้มากถึง 128 อักขระ แต่ถ้าการติดต่อสื่อสารนั้นเป็นการเชื่อมโยงระหว่างจุด 2 จุด ทุกๆเฟรม (รวมทั้งเฟรมแรกด้วย) สามารถมีข้อมูลได้มากถึง 128 อักขระ

รูปแบบบล็อคที่มากกว่าหนึ่งเฟรม

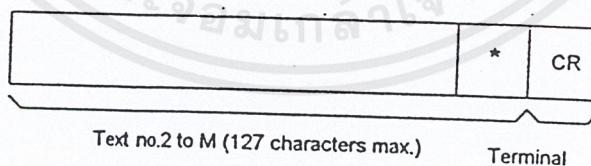
1. ในระบบการติดต่อสื่อสารที่มีการเชื่อมโยงหลายๆจุด



เฟรมต่างๆตรงกลาง (มีความยาวของข้อมูลน้อยกว่าหรือมากกว่า 128)

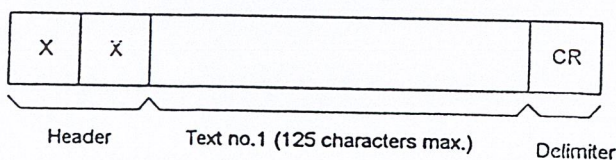


เฟรมสุดท้าย (มีความยาวของข้อมูลน้อยกว่าหรือเท่ากับ 128)



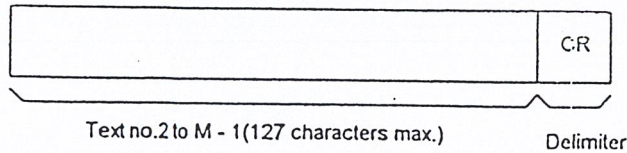
2. ในการติดต่อสื่อสารที่เป็นารเชื่อมโยงระหว่างจุด 2 จุด

เฟรมแรก (มีความยาวของข้อมูลน้อยกว่าหรือเท่ากับ 128)

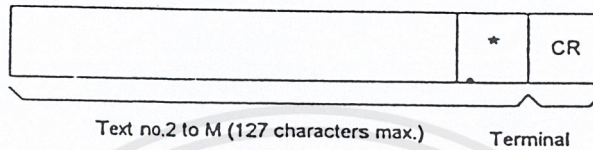


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เฟรมต่างๆตรงกลาง (มีความยาวของข้อมูลน้อยกว่าหรือเท่ากับ 128)

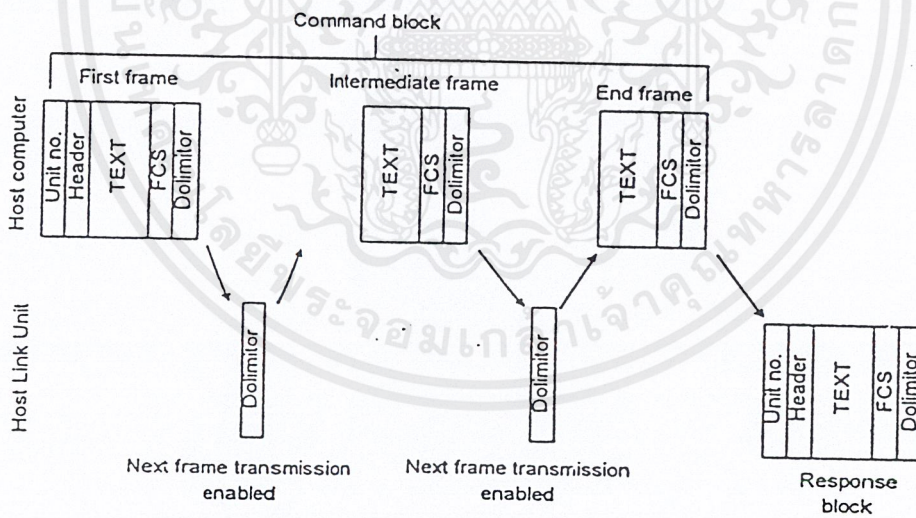


เฟรมสุดท้าย (มีความยาวของข้อมูลน้อยกว่าหรือเท่ากับ 128)



2.1.4 คำสั่งต่างๆที่ใช้ในการส่งข้อมูล(Sending Commands)

การส่งบล็อกคำสั่งที่มีเฟรมมากกว่าหนึ่งเฟรม จากเครื่องคอมพิวเตอร์ส่วนบุคคล จะเริ่มต้นด้วยการส่งเพียงแค่เฟรมแรกเท่านั้น จะไม่มีการส่งเฟรมลำดับถัดไปด้วย จนกว่า Host Computer จะได้รับคำกำหนดขอบเขต (Delimiter) ซึ่งเป็นค่าที่เครื่องควบคุมแบบ โปรแกรม ได้จะส่งกลับมา และในการเขียนคำสั่งแต่ละคำสั่งไม่ควรแยกข้อมูลออกจากเฟรมต่างๆที่แตกต่าง กันให้เป็นเวิร์คเพียงเวิร์คเดียว

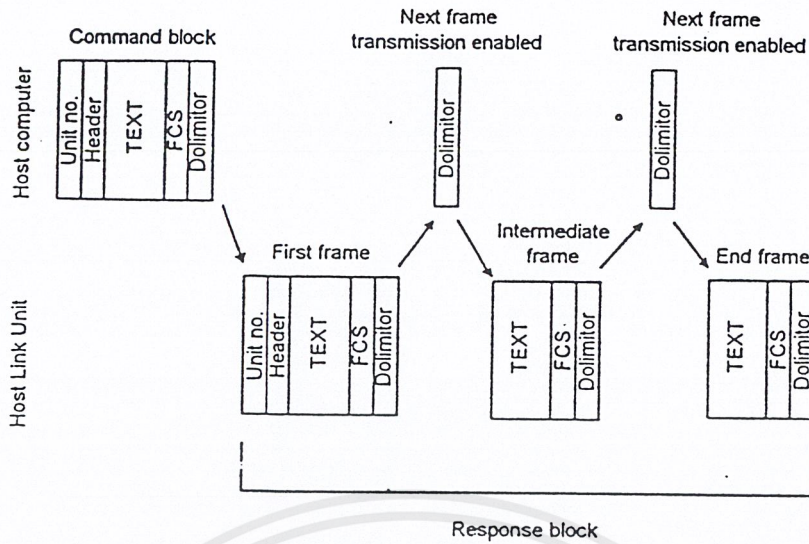


รูปที่ 2.1 คำสั่งต่างๆที่ใช้ในการส่งข้อมูล

2.1.5 คำสั่งต่างๆที่ใช้ในการรับข้อมูล(Receiving Commands)

ในการรับข้อมูล บล็อกตอบสนองจะประกอบไปด้วยเฟรมต่างๆมากกว่าหนึ่งเฟรม จากเครื่องควบคุมแบบ โปรแกรม ได้โดยที่ Host Computer จะต้องส่งคำรหัสย้อนกลับ(Delimiter) ไปให้เครื่องควบคุมแบบ โปรแกรม ได้ เพื่อเป็นการกระตุ้นให้เครื่องควบคุมแบบ โปรแกรม ได้ส่งค่าของเฟรมต่อไปมาให้แก่ Host Computer

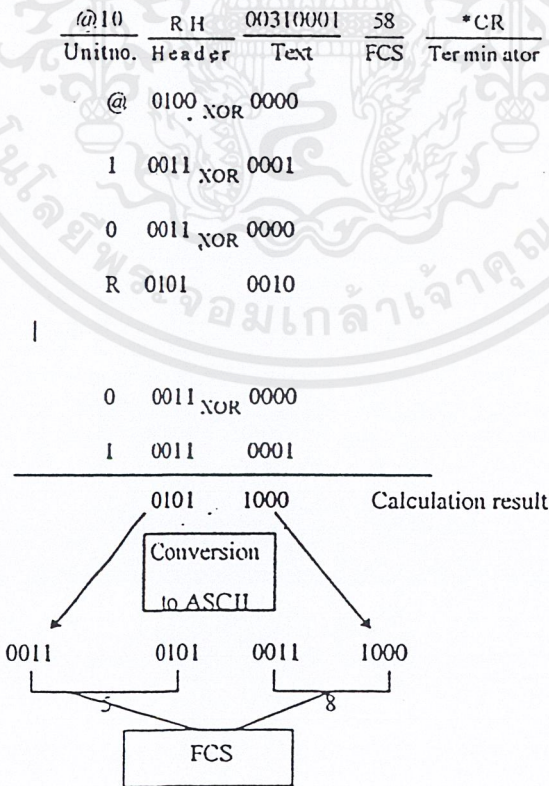
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการเรียนเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.2 คำสั่งต่างๆที่ใช้ในการรับข้อมูล

2.1.6 การคำนวณ Frame Checksum(FCS)

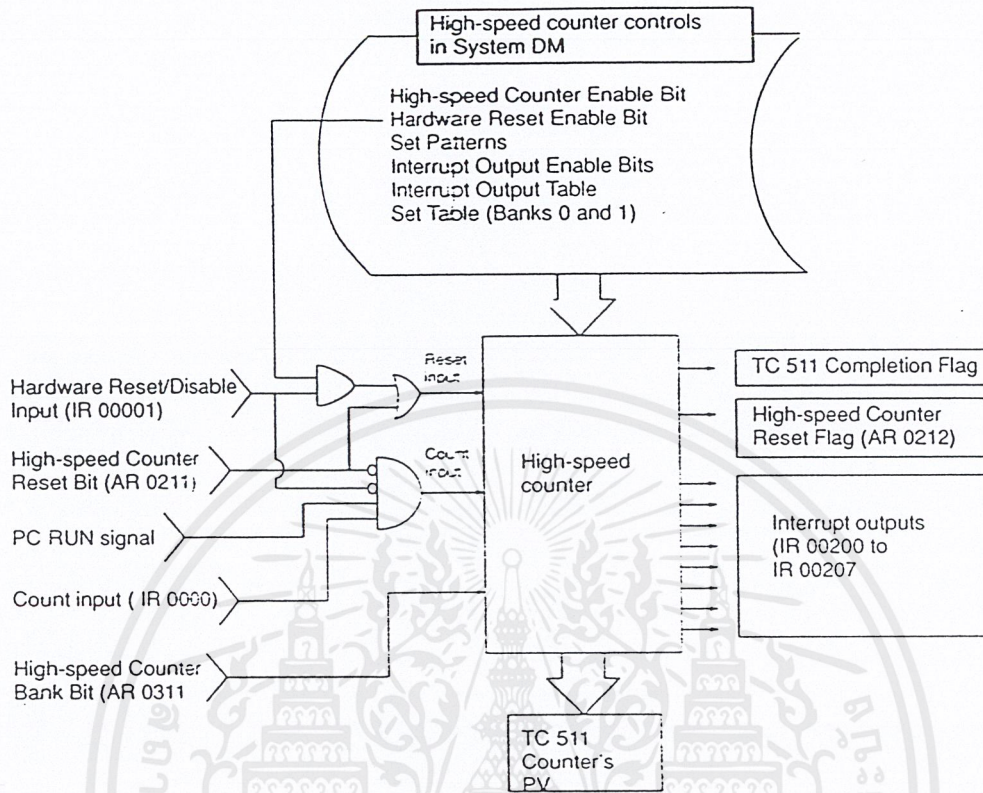
FCS คือข้อมูลจำนวน 8 บิต ที่เป็นค่าผลลัพธ์จากการ EXCLUSIVE OR ค่าข้อมูลของ ASCII จำนวน 2 ชุดอักษร โดยกระทำทีละอักษรเรียงลำดับกันไป เริ่มต้นจากอักษรแรกของทั้งสองชุด จนกระทั่งถึงอักษรสุดท้าย ดังตัวอย่างต่อไปนี้



รูปที่ 2.3 การคำนวณ FCS

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.1.7 High Speed Counter



รูปที่ 2.4 The overall operation of the high speed counter

คำสั่งการใช้งาน High Speed Counter

High Speed Counter – HDM(61)

Ladder Symbol

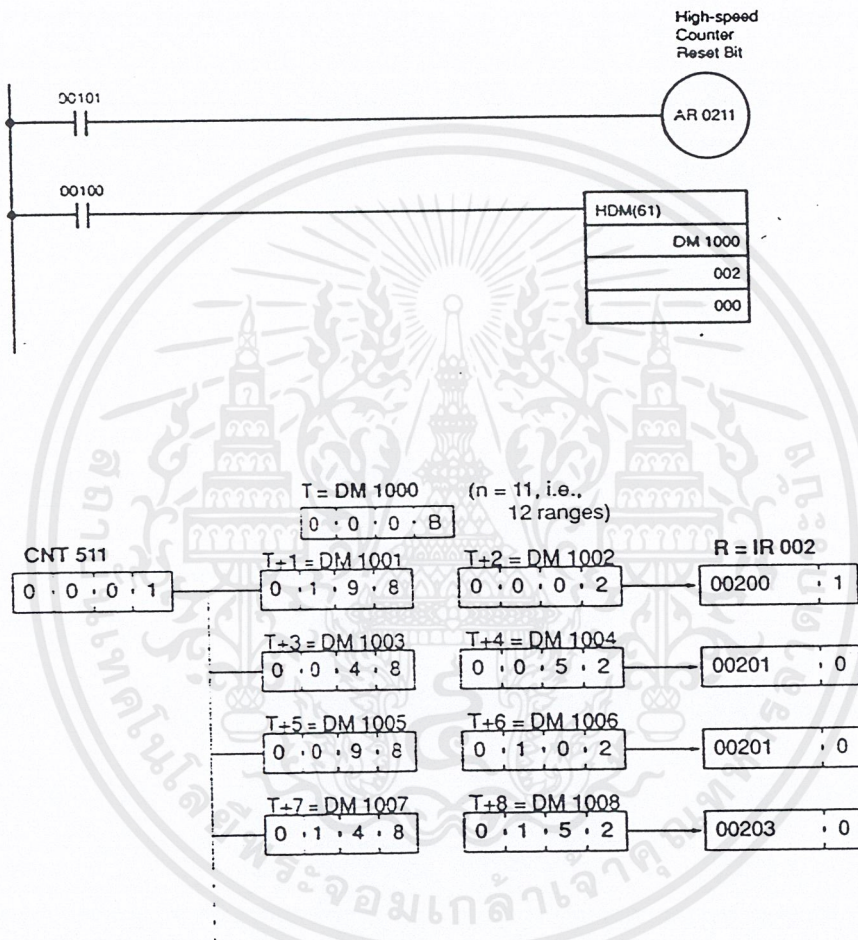
Operation Data Areas

| |
|---------|
| HDM(61) |
| T |
| R |
| --- |

| |
|--|
| T: Beginning table word(BCD) IR, SR, AR, DM, HR, TC, LR |
| R: Beginning result word IR, SR, AR, DM, HR, TC, LR |
| ---: Not used |

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตัวอย่างการใช้งานฟังก์ชัน High Speed Counter – HDM(61)



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.1.8 การ Setup Datamemory C28H Back Up

การที่เราจะใช้ High Speed Counter ของเครื่อง PLC รุ่น C28H นี้ได้จะต้องทำการตั้งค่าให้กับ PLC ตามขั้นตอนดังนี้

| | | | | | | | |
|-------|---------|-------|----|------|-------|-------|-----|
| SHIFT | CONTACT | SHIFT | HR | 1415 | MONTR | SHIFT | SET |
|-------|---------|-------|----|------|-------|-------|-----|

| | | | | | | |
|-------|------|-------|----|----|-------|------|
| SHIFT | CH/* | SHIFT | HR | 14 | MONTR | 8280 |
|-------|------|-------|----|----|-------|------|

2.2 การจัดการฐานข้อมูลโดย MS-Visual Basic และ MS-Access

2.2.1 Microsoft Visual Basic

โปรแกรม Microsoft Visual Basic 6.0 เป็นเครื่องมือในการพัฒนาโปรแกรมที่ได้ได้รับความนิยมเป็นอย่างมาก เนื่องจากมีความง่ายและสะดวกในการใช้งาน อีกทั้งยังมีเครื่องมือในการเขียนโปรแกรมเป็นจำนวนมาก Microsoft Visual Basic 6.0 เป็นเครื่องมือในการสร้างโปรแกรมระบบปฏิบัติการ Windows ที่ใช้งานง่าย โดยการสร้างโปรแกรมนั้นจะเป็นการเลือกเครื่องมือต่างๆ มาออกแบบหน้าจอของโปรแกรมที่จะสร้างขึ้น ซึ่งเรียกการเขียนโปรแกรมในลักษณะนี้ว่า Visual Programming การเขียนโปรแกรมแบบนี้ไม่จำเป็นต้องเขียนคำสั่งต่างๆ มากนัก จึงสามารถสร้างโปรแกรมได้อย่างรวดเร็ว นอกจากนี้ยังสนับสนุนการจัดการการกับฐานข้อมูล ได้เป็นอย่างดี

คำศัพท์ที่ใช้ใน MS- Visual Basic

คอนโทรล(Control)

เป็นเครื่องมืออย่างหนึ่งที่ใช้สร้างออบเจกต์บนฟอร์มของ MS- Visual Basic เลือกคอนโทรลจากทูลบ็อกซ์แล้ววาดคอนโทรลนั้นไว้บนฟอร์ม โดยการใช้เมาส์ เราจะใช้คอนโทรลส่วนใหญ่ในการสร้างยูสเซอร์อินเตอร์เฟซเช่น Imagebox , Commandbutton

ออบเจกต์(Object)

ออบเจกต์เป็นชื่อหนึ่งขององค์ประกอบของยูสเซอร์อินเตอร์เฟซ ที่เราสร้างไว้บนฟอร์มของ MS- Visual Basic โดยใช้คอนโทรลในทูลบ็อกซ์ เราสามารถเคลื่อนย้ายปรับขนาดและปรับแต่ง ออบเจกต์ ได้โดยการกำหนดคุณสมบัติ ออบเจกต์ยังมีสิ่งที่เรา Inherent Functionality ซึ่งหมายถึงการที่ออบเจกต์รู้ว่าต้องทำอะไรบ้างและสามารถตอบสนองต่อสถานะการณ์ต่างๆ ได้เอง (เช่น Listbox รู้จักวิธีเลื่อน Scrollbox เป็นต้น) และสามารถเขียน โปรแกรม MS- Visual Basic ให้กับออบเจกต์ โดยใช้ไวนต์โปรซีเจอร์ที่ปรับแต่งไว้ตามสถานะการณ์ต่างๆ ในโปรแกรมได้ ฟอร์มในวิซวลเบสิกเองก็เป็นออบเจกต์อย่างหนึ่งเหมือนกัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

คุณสมบัติ(Property)

คุณสมบัติเป็นค่าหรือลักษณะที่มีอยู่ในออบเจกต์ของวิซวล ตัวอย่างเช่น คุณสมบัติ Caption และ Forecolour ซึ่งจะสามารถกำหนดคุณสมบัติในระหว่างออกแบบได้ โดยใช้หน้าต่าง Property หรือระหว่างที่รันโปรแกรม โดยใช้สเตตเมนต์ (Statement) ต่างๆ ในโค้ดโปรแกรม ในการเขียน โค้ดโปรแกรมจะใช้รูปแบบการกำหนดคุณสมบัตินี้

Object.Property = Value

โดยที่ Object เป็นชื่อของออบเจกต์ที่ต้องการปรับค่า Property เป็นคุณลักษณะที่ต้องการเปลี่ยน และ Value เป็นค่าที่กำหนดขึ้นใหม่ ตัวอย่างเช่น Command.Caption = "Hello" สามารถใช้โค้ด โปรแกรมเพื่อกำหนดคุณสมบัติ Caption ของ Object Command1 เป็น "HELLO"

อีเวนต์โพรซีเจอร์(Eventprocedure)

เป็นบล็อกของโค้ดที่ทำงานเมื่อออบเจกต์นั้นถูกใช้งานในโปรแกรมเช่น หากมีการคลิกที่ปุ่มCommandButton ที่ชื่อCommand1 จะมีการเรียกโพรซีเจอร์ Comaand1_Click ทันที

โปรแกรมสเตตเมนต์(Statement)

โปรแกรมสเตตเมนต์เป็นคีย์เวิร์ดใน โค้ด ที่ใช้งานของโปรแกรมสเตตเมนต์ของวิซวลเบสิก ใช้สร้างแหล่งเก็บข้อมูล เปิดไฟล์ คำนวณตัวเลข ชื่อหรือคุณสมบัติต่างๆ เก็บไว้ในโปรแกรมได้

วิธีการ(Method)

เป็นสเตตเมนต์พิเศษที่ใช้ทำงานอย่างใดอย่างหนึ่งให้กับออบเจกต์เฉพาะอย่างในโปรแกรม ในการเขียน โค้ด โปรแกรมจะใช้รูปแบบการกำหนดวิธีการดังนี้

Object.Method Value

โดยที่Object เป็นชื่อของออบเจกต์ที่ต้องการเปลี่ยน Method เป็นคำสั่งที่ต้องการใช้เปลี่ยนแปลงออบเจกต์นั้นและ Value เป็นอาร์กิวเมนต์เพิ่มเติมที่ใช้กับวิธีการ ตัวอย่างเช่น

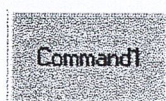
List1.AddItem"Check"

เป็นการใช้วิธีการ AddItem เพิ่มคำว่า Check ไว้ใน List1

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

อุปกรณ์ที่ใช้ในการสร้างยูสเซอร์อินเตอร์เฟซ (User Interface)

คอนโทรล Command Button



เป็นคอนโทรลที่ส่วนใหญ่นำไปใช้ในฟอร์มต่างๆ ฟอร์ม เนื่องจากเป็นคอนโทรลที่ใช้ในการตอบรับการทำงานจากผู้ใช้ เช่นเดียวกับปุ่มคำสั่ง OK, Cancel ทั่วไป

คอนโทรล Text Box



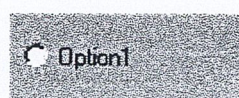
มักจะถูกนำไปใช้ทุกๆ ฟอร์มที่มีการรับกรอกข้อความจากผู้ใช้ เนื่องจากคอนโทรลนี้ทำหน้าที่แสดงข้อมูล และยังอนุญาตให้ผู้ใช้สามารถแก้ไขตัวอักษรต่างๆ ได้อีกด้วย

คอนโทรล Label



เป็นคอนโทรลในลักษณะของกราฟฟิกที่ถูกใช้ในงานด้านการแสดงผลข้อความบนฟอร์มเพื่อใช้ในการสื่อสารข้อความกับผู้ใช้ และคอนโทรลนี้ไม่สามารถแก้ไขได้โดยตรง

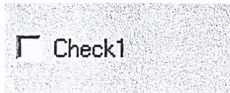
คอนโทรล Option Button



เป็นคอนโทรลที่ถูกออกแบบ เพื่อนำมาใช้ในลักษณะของการเลือกอย่างใดอย่างหนึ่ง และคอนโทรลนี้ผู้อ่านสามารถนำมาใช้โดยแยกออกเป็นกลุ่มๆ ได้โดยอาศัยคอนโทรลชนิดตัวบรรจุบอกลักษณะเป็น True หรือ False

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

คอนโทรล Check Box



เป็นคอนโทรลที่ถูกนำไปใช้ในลักษณะของการกำหนดสถานะเป็น True หรือ False คอนโทรลนี้ไม่มีลักษณะการรวมกลุ่ม ผู้ใช้สามารถเลือกทีละตัวพร้อมๆกันได้คราวละหลายๆตัว

คอนโทรล List Box



โดยที่ลักษณะการใช้งานจะเป็นในรูปแบบการแสดงความหลายๆบรรทัดภายในคอนโทรล List Box

คอนโทรล ToolBar

เป็นคอนโทรลที่จะแสดงปุ่มต่างๆให้ผู้ใช้งานสามารถสั่งให้โปรแกรมทำงานได้ โดยทั่วไปการออกแบบ ToolBar จะเป็นการนำคำสั่งที่ผู้ใช้งานบ่อยๆมารวมกันเป็นชุด เพื่อความสะดวกในการสั่งงานโปรแกรม

คอนโทรล Line และ Shape

มีคุณสมบัติและวิธีการสำหรับใช้ควบคุมการแสดงผล และตำแหน่งและควบคุมพฤติกรรมของคอนโทรลเท่านั้น แต่จะไม่มี การตอบสนองเหตุการณ์กับผู้ใช้ เนื่องจากคอนโทรลนี้ไม่มีเหตุการณ์เป็นคอนโทรลเพิ่มกราฟฟิกให้กับฟอร์มเท่านั้น

คอนโทรล Image

เป็นคอนโทรลที่ช่วยในการแสดงผลกราฟฟิก ในไฟล์กราฟฟิกต่างๆ เช่นเดียวกับคอนโทรล Picture Box แต่คอนโทรล Image จะสามารถทำงานหรือวาดกราฟฟิกภายในคอนโทรลได้รวดเร็วกว่า และที่สำคัญมีการใช้ทรัพยากรระบบน้อยกว่าอีกด้วย แต่มีการสนับสนุนคุณสมบัติเหตุการณ์ และวิธีการน้อยกว่าคอนโทรล Picture Box

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

คอนโทรล Picture Box

เป็นคอนโทรลแสดงกราฟฟิกและภาพกราฟฟิกจากไฟล์

คอนโทรล Time



เป็นคอนโทรลที่ใช้ในการควบคุม และจัดเหตุการณ์ด้านเวลา สามารถเขียนโค้ดเมื่อทำงานใดๆ เมื่อช่วงเวลาก่อนไปตามที่กำหนด คอนโทรลนี้จะตอบสนองเหตุการณ์เพียงเหตุการณ์เดียวเท่านั้น

คอนโทรล Common Dialog



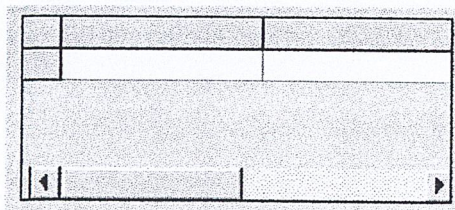
เป็นคอนโทรลที่ช่วยในการแสดงผลไดอะล็อกบ็อกซ์ Open, Save As, Colour, Font, Print Setup, หรือ Print ซึ่งสามารถจะควบคุมการแสดงผลส่วนต่างๆ ของไดอะล็อกบ็อกซ์ได้โดยผ่านทางคุณสมบัติของคอนโทรล Common Dialog ทั้งนี้เพื่อการใช้งานให้เป็นไปได้ด้วยความยืดหยุ่นมากยิ่งขึ้น

คอนโทรล Communication



เป็นเครื่องมือที่ช่วยในการติดต่อกับพอร์ตอนุกรม(Serial Port)ทำให้สามารถรับส่งข้อมูลผ่านทางพอร์ตอนุกรมได้ด้วยคอนโทรลชนิดนี้ เช่นในการติดต่อผ่าน โมเด็ม

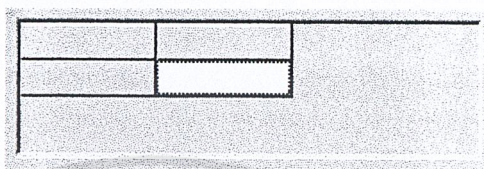
คอนโทรล DataGrid



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

คอนโทรล DataGrid เป็นคอนโทรลที่ถูกจัดอยู่ในกลุ่ม Data-Aware Control มีรูปแบบการแสดงผลข้อมูลแบบตาราง

คอนโทรล MSHFlexGrid



เป็นคอนโทรลพื้นฐานข้อมูล ที่ถูกออกแบบมาให้ใช้ในการแสดงข้อมูลที่มาจากตารางมากกว่า 1 ตารางในเวลาเดียวกัน

คอนโทรล DTPicker

21/3/45

คอนโทรล DateTimePicker เป็นคอนโทรลแสดง Dropdown List เป็นปฏิทินที่สามารถรับข้อมูลวันที่ และเวลาจากผู้ใช้

คอนโทรล CoolBar



คอนโทรล CoolBar จะใช้สำหรับบรรจุคอนโทรล Toolbar ตั้งแต่ 1 ตัวขึ้นไป เก็บรวมกันไว้เพื่อให้สามารถแสดงทุลบาร์ได้มากกว่า 1 แถบในเวลาเดียวกัน

คอนโทรล ADO Data



คอนโทรล ADO Data เป็นคอนโทรลที่ถูกออกแบบมาแทนคอนโทรล Data โดยที่คอนโทรล ADO Data จะใช้เทคโนโลยี OLEDB เป็นกลไกในการเข้าถึงข้อมูลในฐานข้อมูล รูปแบบการติดต่อของคอนโทรล ADO Data กับฐานข้อมูล จะเป็นลักษณะการเชื่อมต่อ (Connection) เข้ากับฐานข้อมูลแทนการกำหนดชื่อฐานข้อมูล

ออบเจ็กต์ OLE

หมายถึงข้อมูลที่ได้รับการแอปพลิเคชัน OLE ซึ่งข้อมูลนี้สามารถเป็นไปได้อีกหลายชนิด เช่น Microsoft Access สนับสนุนออบเจ็กต์ OLE ดังต่อไปนี้ worksheet, macro sheet, chart, cell เป็นต้น และออบเจ็กต์ OLE ที่สร้างขึ้นใหม่ ผู้อ่านสามารถกำหนดให้เป็นออบเจ็กต์ OLE ชนิดเชื่อมโยง (Linked) หรือฝัง (Embedded) ก็ได้

ออบเจ็กต์ Printer

ใช้ในการควบคุมการพิมพ์ข้อความเอกสาร ฟอนต์หรือกราฟฟิก ออกทางเครื่องพิมพ์ โดยที่ข้อมูลทั้งหมดที่ถูกส่งพิมพ์โดยอาศัย ออบเจ็กต์นี้ จะถูกนำไปเข้าคิวรอจัดพิมพ์อีกครั้ง โดย Print Manager ของวินโดวส์อีกครั้ง

คุณสมบัติ(Property)

| | |
|------------|--|
| Alignment | กำหนดตำแหน่งการวางข้อความในคอนโทรล คือ ค่า 0 วางข้อความชิดซ้าย ค่า 1 วางข้อความชิดขวา ค่า 2 วางข้อความไว้ตรงกลาง |
| BackColour | และ ForeColour รายงานหรือกำหนดสีพื้นและสีตัวอักษรของข้อความในคอนโทรล |
| Backstyle | รายงาน หรือกำหนดคุณลักษณะของพื้นหลังของคอนโทรลให้เป็นแบบโปร่งใสหรือแบบทึบ |
| Caption | กำหนดข้อความสำหรับแสดงภายในคอนโทรล |
| Enabled | รายงานหรือกำหนดให้คอนโทรลสามารถมีหรือไม่มีการตอบสนองต่อเหตุการณ์ที่ถูกกำหนดโดยผู้ใช้ เช่น Click โดยจะมีค่าเป็น true หรือ False |
| Font | ใช้ในการกำหนดคุณสมบัติของออบเจ็กต์ฟอนต์ หรือรายงานคุณสมบัติของออบเจ็กต์ฟอนต์ของคอนโทรล |

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์ของมหาวิทยาลัยเทคโนโลยีพระจอมเกล้าธนบุรี ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

| | |
|--------------|---|
| | Height, Width, Left, Top รายงานหรือกำหนดขนาดความกว้าง ความสูงและตำแหน่ง (Left ,Top) |
| | การวางคอนโทรลลงในตัวบรรจุ |
| Name | กำหนดชื่อคอนโทรลสำหรับการอ้างอิงโดยโค้ดในการเขียนโปรแกรม |
| Visible | รายงานหรือกำหนดให้เหตุการณ์ Click ของคอนโทรลโดยการกำหนดให้คุณสมบัตินี้มีค่าเท่ากับ True |
| Value | รายงานหรือกำหนดให้เหตุการณ์ Click ของคอนโทรลโดยการกำหนดให้คุณสมบัตินี้มีค่าเท่ากับ True |
| List Index | รายงานหรือกำหนดค่าลำดับของรายการที่ถูกเลือก (เลือกสำหรับรายการที่คุณสมบัติ List Index มีค่าเท่ากับ 0) |
| Border Style | รายงาน หรือกำหนดการแสดงผลเส้นขอบคอนโทรลหรือกำหนดรูปแบบในการวาดเส้นตรงหรือเส้นรอบรูปในคอนโทรล Shape |
| Border Width | รายงานหรือกำหนดความหนาของเส้นตรงหรือรอบรูปทรง Shape |
| Fillcolour | รายงานหรือกำหนดสีที่ใช้ในการระบายพื้นที่ภายในรูปทรง Shape |
| Fill Style | รายงานหรือกำหนดรูปแบบใช้ในการระบายพื้นที่ภายในรูปทรง Shape |
| Shape | รายงานหรือกำหนดรูปทรงของคอนโทรล Shape เช่น สี่เหลี่ยม วงกลม |
| Picture | รายงานหรือกำหนดกราฟฟิกหรือ ไฟล์กราฟฟิกจะแสดงผลภายในคอนโทรล |
| Text | รายงานหรือกำหนดข้อความให้กับคอนโทรล ซึ่งข้อความที่อยู่ในคุณสมบัตินี้จะถูกนำไปแสดงในคอนโทรล Text Box |
| Stretch | รายงานหรือข้อกำหนดให้กราฟฟิกมีขนาดเปลี่ยนแปลงตามขนาดของคอนโทรล |
| Interval | รายงานหรือกำหนดช่วงเวลาในหน่วยของมิลลิวินาที (1 ใน 1000วินาที) สำหรับทำให้เหตุการณ์ Timer ซึ่งการตรวจสอบเวลาจะถูกกระทำโดย Vb/Win |

2.2.2 Microsoft Access

Microsoft Access เป็นโปรแกรมจัดการฐานข้อมูล (Database Management Program) ที่มีขีดความสามารถสูง และมีการออกแบบให้ง่ายต่อการใช้งาน แต่ก็ยังมีปัญหาเกิดขึ้นบ้างในการใช้งานเนื่องจากคุณสมบัติต่างๆ ที่มีอยู่มากมายในโปรแกรม

นิยามต่างๆ ที่ควรรู้

Database คือ ไฟล์เดี่ยวซึ่งประกอบด้วยข้อมูลและ Tools ทั้งหมด(เช่น Form และ Report)สำหรับการดำเนินการกับข้อมูล

Table คือ ที่ที่จะต้องใช้ในการจัดเก็บข้อมูลดิบ ซึ่งมีการจัดเรียงเช่นเดียวกับ Spreadsheet

แต่ละ คอลัมน์หรือฟิลด์จะประกอบด้วยรายละเอียดแต่ละชนิด เช่น ZIP CODE ในเอกสารนี้เป็นเอกสารที่ส่งวนเวียนสำหรับการแข่งขันเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปเผยแพร่บนสื่อออนไลน์ใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

| | |
|----------------|---|
| | แต่ละแถว(row) หรือ (Record) จะประกอบไปด้วยรายละเอียดในแต่ละรายการหรือเหตุการณ์ เช่น คนไข้ หรือ ข้อมูลต่างๆ ด้านเครื่องมือ เป็นต้น |
| Queries | จะเป็นส่วนที่เรียกใช้และรวบรวมข้อมูลจาก Table ต่างๆ โดยขึ้นอยู่กับหลักเกณฑ์ที่กำหนดไว้ เช่น สามารถใช้ Query ในการเรียกใช้ Record จาก Table ชื่อ Employee สำหรับพนักงานทุกคนซึ่งทำงานล่วงเวลาในสัปดาห์ที่ผ่านมา Query ยังสามารถเรียกใช้ข้อมูลเป็นกลุ่มได้เช่นกัน |
| Form | เป็น Windows ที่ปรากฏขึ้นบน screen และใช้ในการแสดงข้อมูลหรือบันทึกข้อมูลโดยตรง |
| Report | จะทำงานคล้ายกับ Form เว้นแต่ว่า Report ใช้สำหรับพิมพ์ออกมา ไม่แสดงบน screen และไม่ใช้ในการบันทึกข้อมูล |
| Macro | เป็นส่วนที่ดำเนินการตามที่กำหนดโดยอัตโนมัติ และใช้เพื่อทำให้การทำงานใน Access ง่ายขึ้น |
| Modules | คือ sheet ต่างๆ ซึ่งประกอบด้วย Access Basic Code ที่มีการโปรแกรมขึ้นใน Access |

2.2.3 การจัดการฐานข้อมูล

ในการจัดการฐานข้อมูลต่างๆ ทำในลักษณะของการจัดเก็บข้อมูลแล้วนำข้อมูลนั้นออกมาใช้หรือแสดงในภายหลังในรูปแบบของรายงานต่างๆ เช่น งานทางด้านบัญชี จัดการกับสินค้าคงคลัง ข้อมูลบุคลากร เป็นต้น

ความซับซ้อนของกระบวนการต่างๆ ในโปรแกรมส่วนหนึ่งเป็นผลมาจากความสลับซับซ้อนของข้อมูลที่เกี่ยวข้องอีกทั้งรูปแบบของการเก็บข้อมูลในแฟ้มข้อมูลในลักษณะพื้นฐาน เช่น แฟ้มข้อมูลแบบ Sequencetial, Random หรือ Indexed File ก็ล้วนมีจุดอ่อนในการลดทอนประสิทธิภาพการทำงานของผู้พัฒนาโปรแกรมหลายประการ เช่น

1. ผู้พัฒนาโปรแกรมจะต้องมีความเข้าใจในลักษณะของแฟ้มข้อมูลที่จะใช้นั้นเป็นอย่างดีและจะต้องคิดหาวิธีการเรียกใช้ข้อมูลดังกล่าวให้เหมาะสมกับลักษณะของการจัดเก็บในแฟ้มข้อมูลแบบนั้นๆ โดยเฉพาะทำให้โปรแกรมที่ต้องการพัฒนานั้นซับซ้อนขึ้นโดยไม่จำเป็น
2. ขัดแย้งกับแนวคิดในการพัฒนาโปรแกรมในลักษณะของ Modular Programming เพราะคำสั่งต่างๆ ที่ใช้ในการจัดการข้อมูลนั้นจะขึ้นอยู่กับภาษาหรือเครื่องมือในการพัฒนานั้น เช่น คำสั่งที่ใช้ในการอ่านข้อมูลภาษาซี หรือ เบสิก ก็ไม่เหมือนกัน ไม่สามารถที่จะใช้สิ่งที่ได้เรียนรู้และโปรแกรมย่อยต่างๆ ร่วมกันได้ ทำให้สิ้นเปลืองเวลาและเงินทองในการเรียนรู้และพัฒนา
3. ตัวข้อมูลเองก็อาจจะไม่สามารถที่จะใช้ร่วมกันได้ (Share) กับโปรแกรมอื่นได้อย่างสะดวกหากในหน่วยงานหรือองค์กรมีระบบงานที่ใช้ข้อมูลเหมือนกัน แต่ไม่สามารถใช้ข้อมูล

เอกสารนี้เป็นเอกสารสงวนลิขสิทธิ์เพื่อการศึกษาค้นคว้า ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ร่วมกันได้จะทำให้เกิดความซับซ้อนของข้อมูลขึ้นในหน่วยงานหรือองค์กรนั้น ซึ่งต่อมาจะทำให้ไม่สามารถบอกได้ว่าข้อมูลใดถูกต้องและใหม่ที่สุด

เราสามารถที่จะหลีกเลี่ยงปัญหาต่างๆเหล่านี้ได้ด้วยการใช้ระบบจัดการฐานข้อมูล (Database Management System –DBMS) แทนการใช้แฟ้มข้อมูลทั่วไป ลักษณะของฐานข้อมูลนั้นมีหลักอยู่ 3 ประเภท คือแบบ Hierarchy แบบ Relation และแบบ Network แต่ที่ได้รับความนิยมอย่างมากในปัจจุบันจะเป็นฐานข้อมูลแบบสัมพันธ์ (Relation Database) เพราะว่าพื้นฐานข้อมูลที่จัดเก็บจะอยู่ในรูปตาราง (Table) ซึ่งจะทำให้ง่ายต่อการเข้าใจและเรียกใช้งาน

ใน MS-Visual Basic 6.0 สามารถติดต่อกับฐานข้อมูลได้หลากหลาย เช่น Dbase, FoxPro และยังสามารถสร้าง ODBC (Open Database Connectivity) เพื่อเป็นมาตรฐานในการติดต่อกับฐานข้อมูลภายนอกบนเครื่อง Server ต่างๆ เช่น SQL Server , Oracle ซึ่ง MS-Visual Basic จะเตรียมโปรแกรมสำหรับติดต่อกับฐานข้อมูลชนิดนั้นๆ หรือที่เรียกว่า Driver ไว้ให้หากต้องการใช้กับฐานข้อมูลชนิดอื่น นอกจากนี้ต้องหา Driver สำหรับฐานข้อมูลชนิดนั้นมาติดตั้งเพิ่มเติม

ข้อดีของความสามารถทางด้านฐานข้อมูลของ MS-Visual Basic คือช่วยลดข้อจำกัดการพัฒนางาน โดยใช้แฟ้มข้อมูลธรรมดาในการจัดเก็บข้อมูล เช่น

1. โปรแกรมจะเป็นอิสระจากฐานข้อมูลที่ใช้เพราะเมื่อไม่มีความแตกต่างจากฐานข้อมูล หมายความว่าจะเป็นฐานข้อมูลชนิดใด ก็จะใช้ชุดคำสั่งในการอ่านเขียน หรือจัดการต่างๆ ได้เหมือนกัน จะแตกต่างก็ตรงส่วนขยายของคำสั่งสำหรับการเปิด (Open) หรือสร้าง (Create) ฐานข้อมูลเท่านั้น ทำให้ไม่ต้องไปเรียนรู้ถึงคำสั่งที่แตกต่างกันหลายแบบ นอกจากนี้การใช้ชุดคำสั่งย่อย (Module) ร่วมกันก็เป็นไปได้มาก และสร้างระบบงานในลักษณะของ Client/Server ก็ง่ายขึ้น สามารถทำการพัฒนาบนระบบเครื่องคอมพิวเตอร์เครื่องเดียวก็ได้ จากนั้นก็ค่อยเปลี่ยนค่าคงที่กำหนดในเรื่องฐานข้อมูลเท่านั้น ก็สามารถนำไปใช้กับข้อมูลบนเครื่องอื่นได้ทันที

2. ข้อมูลเดิมที่มีอยู่แล้วในฐานข้อมูลชนิดอื่นๆ ก็สามารถนำมาใช้กับ MS-Visual Basic ได้ โดยไม่จำเป็นต้องสร้างชุดใหม่อีกชุดให้ซับซ้อน หรือทำการแปลง (Convert) ให้เป็นฐานข้อมูล Access เสียก่อน เพราะการทำเช่นนั้นจะเสียเวลาไปพัฒนาระบบงานเดิมที่ใช้กับฐานข้อมูลเดิมแล้วซ้ำอีกชุดหนึ่ง สำหรับฐานข้อมูลที่เปลี่ยนแปลงมาใหม่ซึ่งจะเป็นการสิ้นเปลืองเวลาและเงิน โครงสร้างฐานข้อมูล

ฐานข้อมูลแบบสัมพันธ์มีหลักการพื้นฐานมาจากคณิตศาสตร์ ในเรื่องทฤษฎีบทของเซต (Theory of Set) โดยที่การจัดเก็บหรือแสดงข้อมูลให้ผู้ใช้เห็นจะเป็นในรูปแบบของตาราง

ข้อมูลต่างๆ จะแสดงในรูปของตาราง โดยในแต่ละตารางจะเป็นการจัดรวบรวมข้อมูลประเภทเดียวกันไว้ด้วยกัน โดยที่แต่ละแถวที่ประกอบขึ้นเป็นตารางนั้น ก็คือ เรคคอร์ด ซึ่งจะเป็นที่เก็บข้อมูลแต่ละชุดของตารางนั้น และในแต่ละแถวก็จะประกอบด้วยฟิลด์หรือคอลัมน์ซึ่งเป็นหน่วยย่อยที่แสดงคุณสมบัติของข้อมูลแต่ละแถว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ในแต่ละตารางมักจะมีบางคอลัมน์หรืออาจจะหลายคอลัมน์ประกอบกันที่สามารถบอกถึงความแตกต่างของฐานข้อมูลในแต่ละแถวได้ คอลัมน์ดังกล่าวเรียกว่า Primary Key เช่น คอลัมน์เลขประจำตัวพนักงานที่จะแตกต่างกันทุกคน และในการเชื่อมโยงตารางฐานข้อมูลที่มีหลายตารางนั้นจะกระทำได้อีกคือเมื่อแต่ละตารางมีคอลัมน์ที่เกี่ยวข้องกัน ซึ่งคอลัมน์ที่เกี่ยวข้องนี้จะเรียกว่า Foreign Key

ชุดคำสั่งที่ใช้ในการจัดการกับข้อมูลและ โครงสร้างของฐานข้อมูล

แบ่งได้เป็น 2 ประเภทหลักๆ คือ

1. จัดการกับโครงสร้างฐานข้อมูล เป็นการสร้าง ลบ หรือ แก้ไขฐานข้อมูลและ ตาราง เช่น สร้างตารางฐานข้อมูลพนักงานขึ้นมา ชุดคำสั่งต่างๆ ที่เกี่ยวข้องกับการสร้างหรือเปลี่ยนแปลงโครงสร้างของฐานข้อมูลมีศัพท์เรียกว่า Data Definition Language หรือ DDL

2. จัดการฐานข้อมูลที่มีอยู่ในตารางต่างๆ ในฐานข้อมูล เป็นการทำงานในเรื่องของการเพิ่ม ลบ แก้ไขข้อมูลในตารางต่างๆ เช่น การเพิ่มข้อมูลพนักงานใหม่เข้าไปในตารางข้อมูลพนักงาน คำสั่งประเภทนี้เรียกว่า เป็นคำสั่ง Data Management Language หรือ DML

ปัจจุบันชุดคำสั่งฐานข้อมูลที่นิยมกันมาก เรียกว่า Structure Query Language (SQL) โดยจะประกอบด้วยคำสั่งฐานข้อมูล ซึ่งมีมาตรฐานกำหนดทั้งที่เป็นคำสั่งสำหรับจัดการกับโครงสร้าง (DDL) หรือจัดการกับข้อมูลในฐานข้อมูล (DML) ชุดคำสั่ง SQL ของผู้ผลิตแต่ละรายจะเกือบใกล้เคียงกันหมดแต่อาจจะแตกต่างกันบ้างเล็กน้อย เพราะผู้ผลิตในแต่ละรายมักจะมีการเพิ่มเติมคำสั่งหรือตัวเลือกให้มากขึ้นกว่าที่กำหนดไว้ในมาตรฐาน เพื่อให้มีความสะดวกหรือมีประสิทธิภาพที่ดีเหมาะกับฐานข้อมูลตัวเอง

การจัดการฐานข้อมูลโดยใช้ Structure Query Language (SQL)

Structure Query Language (SQL) เป็นภาษามาตรฐานที่ใช้ในการจัดการฐานข้อมูล โดยเฉพาะที่เป็นแบบ RDBMS (Relation Database Management System) SQL เป็นภาษาที่มีประสิทธิภาพสูงสามารถเรียนรู้และใช้งานอย่างง่าย จึงทำให้เป็นที่นิยมได้อย่างรวดเร็ว นอกจากนี้ยังสามารถใช้ในการจัดการกับฐานข้อมูลโดยผ่าน RDBMS เอง หรือจะสั่งผ่าน โปรแกรมก็ได้ เช่น หากเราใช้ MS-Access เราก็สามารถพิมพ์คำสั่ง SQL ได้โดยตรงในโปรแกรม Access หรือจะใช้คำสั่ง SQL ผ่าน VB6 หรือ Comelier อื่นๆ ก็ได้ เราจะใช้ SQL เพื่อทำการจัดการข้อมูลในฐานข้อมูลได้หลายอย่าง เช่น การแสดงข้อมูลจากฐานข้อมูลแบบมีเงื่อนไข, การเพิ่ม, การลบ และการนำข้อมูลจากตารางหลายๆตารางมาแสดงร่วมกันได้ เป็นต้น

โครงสร้างของภาษา SQL

ภาษา SQL ประกอบไปด้วย 3 ส่วนใหญ่ๆคือ

1. Data Definition Language (DDL) เป็นกลุ่มคำสั่งในภาษา SQL ที่ใช้สำหรับจัดการโครงสร้างของฐานข้อมูล เช่น การสร้างฐานข้อมูล, ปรับปรุงโครงสร้างของฐานข้อมูล เป็นต้น ตัวอย่างการใช้งานกลุ่มคำสั่ง DDL นี้ก็คือ การสร้างฐานข้อมูลด้วย MS SQL Server ก็จะมีการใช้งานคำสั่งในกลุ่ม DDL เป็นหลัก

2. Data Manipulation Language (DML) เป็นกลุ่มคำสั่งในภาษา SQL ที่ใช้สำหรับจัดการข้อมูลในฐานข้อมูล เช่น การแสดงข้อมูลแบบมีเงื่อนไข, การลบข้อมูล, การเพิ่มข้อมูล และการแสดงที่มาจากตารางหลายตาราง เป็นต้น

กลุ่มฟังก์ชัน Aggregate Function เป็นฟังก์ชันพิเศษของภาษา SQL ที่ทำหน้าที่เฉพาะอย่าง เช่น หาผลรวมเรคคอร์ด, ค่าสูงสุด, ค่าต่ำสุด เป็นต้น เป็นกลุ่มฟังก์ชันที่มีประโยชน์มาก เพราะจะช่วยลดภาระให้คุณ ไม่ต้องเขียนโค้ดจัดการเอง

สำหรับการใช้งานภาษา SQL ร่วมกับ Visual Basic เพื่อจัดการฐานข้อมูล จะใช้งานกลุ่มคำสั่ง DML เป็นหลัก โดยจะอธิบายการใช้งานกลุ่มคำสั่ง DML ร่วมกับกลุ่มฟังก์ชัน Aggregate และกำหนดเงื่อนไขโดยการใช้ตัวดำเนินการด้านต่างๆ

คำสั่งในกลุ่มของ DML จะมีคำสั่งพื้นฐานอยู่ 4 คำสั่งคือ

1. DELETE เป็นคำสั่งที่ใช้สำหรับลบข้อมูลหรือลบเรคคอร์ดใดๆในตาราง
2. INSERT เป็นคำสั่งที่ใช้สำหรับเพิ่มข้อมูลหรือเพิ่มเรคคอร์ดใดๆเข้าไปในตาราง
3. SELECT เป็นคำสั่งที่ใช้สำหรับเลือกข้อมูลหรือแสดงเรคคอร์ดใดๆที่ต้องการจากตารางอาจจะมาจากตารางเดียว หรือหลายตารางก็ได้
4. UPDATE ใช้สำหรับแก้ไขข้อมูลหรือแก้ไขเรคคอร์ดใดๆในตาราง

ตัวดำเนินการ (Operator)

ตัวดำเนินการเปรียบเทียบที่น่าสนใจ ได้แก่

- | | | |
|------|-----|--|
| = | คือ | เท่ากับ (Equal) |
| <> | คือ | ไม่เท่ากับ (Not Equal) |
| < | คือ | น้อยกว่า (Less Than) |
| > | คือ | มากกว่า (Greater Than) |
| <= | คือ | น้อยกว่าหรือเท่ากับ (Less Than or Equal To) |
| >= | คือ | มากกว่าหรือเท่ากับ (Greater Than or Equal To) |
| Like | คือ | เป็นการเปรียบเทียบโดยใช้ตัวอักษรพิเศษ (Wild Card Character) เข้าร่วม |

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตัวดำเนินการด้านตรรกะ (Logical Operator)

ตัวดำเนินการด้านตรรกะที่นิยมใช้จะมีอยู่ 3 ชนิดคือ And, OR และ Not

กลุ่มฟังก์ชัน Aggregate

กลุ่มฟังก์ชัน Aggregate เป็นฟังก์ชันที่ใช้คำนวณทางคณิตศาสตร์ โดยช่วยให้การนำเสนอผลการค้นหาเป็นไปอย่างมีประสิทธิภาพ ซึ่งมีฟังก์ชันพื้นฐานดังต่อไปนี้

| | | |
|---------|---------|--|
| AVG() | หน้าที่ | หาค่าเฉลี่ยของฟิลด์ จากเรคคอร์ดทั้งหมด |
| COUNT() | หน้าที่ | นับจำนวนเรคคอร์ด |
| FIRST() | หน้าที่ | หาค่าแรกในฟิลด์ |
| LAST() | หน้าที่ | หาค่าสุดท้ายในฟิลด์ |
| MAX() | หน้าที่ | หาค่ามากที่สุด หรือค่าสูงสุด |
| MIN() | หน้าที่ | หาค่าน้อยที่สุด หรือค่าต่ำสุด |
| SUM() | หน้าที่ | หาผลรวมทั้งหมดของฟิลด์ |

ลักษณะการใช้งานของกลุ่มคำสั่ง DML

คำสั่ง DELETE

เป็นคำสั่งที่ใช้สำหรับลบข้อมูล หรือลบเรคคอร์ดใดๆออกจากตาราง มีรูปแบบการใช้งาน 2 ลักษณะใหญ่ๆคือ

| | |
|------------|--------------------------------------|
| รูปแบบที่1 | DELETE FROM ชื่อตาราง WHERE เงื่อนไข |
| รูปแบบที่2 | DELETE *FROM ชื่อตาราง |

ชื่อตารางในที่นี้ หมายถึง ชื่อตารางที่ต้องการลบ ส่วนเงื่อนไขจะหมายถึง เงื่อนไขในการลบข้อมูล หรือลบเรคคอร์ดในตารางนั้นๆ

ถ้าสำหรับเครื่องหมาย * หมายถึง ข้อมูลใดๆ หรือข้อมูลทุกเรคคอร์ด

คำสั่ง INSERT

เป็นคำสั่งที่ใช้สำหรับเพิ่มข้อมูล หรือเพิ่มเรคคอร์ดเข้าไปในตารางในกรณีที่ฟิลด์เป็นข้อมูลชนิด Text ต้องใช้เครื่องหมาย ' กำกับด้วยฟิลด์นั้นด้วย มีรูปแบบการใช้งาน 2 ลักษณะ

| | |
|------------|---|
| รูปแบบที่1 | INSERT INTO tablename(field 1,field2,...) VALUES(value1,'value2',...) |
| รูปแบบที่2 | INSERT INTO tablename2 SELECT * FROM tablename1 WHERE criteria |

ตัวแปร tablename หมายถึง ชื่อตารางที่ต้องการเพิ่มเร็คคอร์ดเข้าไป

ตัวแปร tablename1 หมายถึง เลือกข้อมูลจากตารางที่ชื่อว่า tablename1 ตามเงื่อนไขที่ระบุไว้ในตัวแปร criteria แล้วนำมาเพิ่มที่ตาราง tablename2

ตัวแปร field-fieldn หมายถึง ชื่อของฟิลด์ต่างๆที่อยู่ในตารางที่ต้องการเพิ่มข้อมูล

ตัวแปร value1-valueu หมายถึง ค่าของฟิลด์ที่จะเพิ่มเข้าไป โดยที่ต้องระบุค่าให้ตรงกับ ฟิลด์ด้วย

ตัวแปร criteria หมายถึง เงื่อนไขในการดึงข้อมูลจากตาราง tablename1

คำสั่ง SELECT

ใช้สำหรับเลือกหรือดึงข้อมูล(Retrieve Data) ที่เราต้องการจากตารางที่ระบุไว้ เป็นคำสั่งที่มีความยืดหยุ่นสูงมาก เพราะว่ามีเงื่อนไขในการนำข้อมูลออกมาจากตารางมีมากมายหลายลักษณะ แต่มีรูปแบบการใช้งานพื้นฐานอยู่ 2 ลักษณะคือ

SELECT * FROM ชื่อตาราง

หรือ

SELECT ฟิลด์ที่ 1, ฟิลด์ที่ 2,...,ฟิลด์ที่ n FROM ชื่อตาราง WHERE เงื่อนไข

ชื่อตาราง หมายถึง ชื่อตารางที่ต้องการดึงข้อมูล

ตัวแปร ฟิลด์ที่ 1, ฟิลด์ที่ 2, ..., ฟิลด์ที่ n หมายถึง ชื่อฟิลด์ที่ต้องการดึงข้อมูล ถ้ามีมากกว่า

1 ฟิลด์จะใช้เครื่องหมาย , คั่นระหว่างฟิลด์

เงื่อนไข หมายถึง เงื่อนไขในการดึงข้อมูล อาจเป็นเงื่อนไขทางคณิตศาสตร์ หรือเป็น

คำสั่ง SELECT ซ้อนอยู่ข้างในก็ได้

คำสั่ง UPDATE

เป็นคำสั่งที่ใช้สำหรับแก้ไข หรือเปลี่ยนแปลงข้อมูลในเร็คคอร์ดที่มีอยู่ในตาราง มีรูปแบบการใช้งานดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้


```
UPDATE ชื่อตาราง SET ชื่อฟิลด์ = ค่าที่กำหนด WHERE เงื่อนไข
```

การเชื่อมโยงตาราง

การเชื่อมโยงตารางโดยใช้คำสั่ง SELECT

วิธีนี้เป็นการเชื่อมโยงตาราง โดยระบุชื่อฟิลด์ที่มีความสัมพันธ์กัน เช่น

```
SELECT Student, FirstName, LastName, MajorDesc
FROM student, major
WHERE student.MajorID=majorID
```

การเชื่อมโยงตารางโดยใช้คำสั่ง INNER JOIN

เป็นการเชื่อมโยงข้อมูลจาก 2 ตารางเข้าด้วยกัน โดยระบุชื่อฟิลด์ที่มีความสัมพันธ์กันหลังคำสั่ง ON เป็นเงื่อนไขในการ JOIN ตาราง โดยที่ฟิลด์ดังกล่าว จะต้องเป็นข้อมูลชนิดเดียวกัน มีลักษณะเช่นเดียวกับการใช้คำสั่ง SELECT มีรูปแบบการใช้งานดังนี้

```
SELECT fieldname
FROM tablename1
INNER JOIN tablename2
ON tablename1.fieldname1 OPERATION tablename2.fieldname2
```

ตัวแปร fieldname หมายถึง ชื่อฟิลด์ที่อยู่ในตารางที่ 1 หรือ 2

ตัวแปร tablename1 หมายถึง ชื่อตารางที่ 1

ตัวแปร tablename2 หมายถึง ชื่อตารางที่ 2

ตัวแปร fieldname1 หมายถึง ชื่อฟิลด์ที่อยู่ในตารางที่ 1

ตัวแปร fieldname2 หมายถึง ชื่อฟิลด์ที่อยู่ในตารางที่ 2

ตัวแปร operation หมายถึง เงื่อนไขหรือตัวดำเนินการต่างๆที่ใช้ในการเชื่อมโยง

การเชื่อมโยงคำสั่งโดยให้คำสั่ง LEFT JOIN

เป็นการเชื่อมโยง 2 ตารางเข้าด้วยกัน โดยกำหนดให้ตารางแรก (ตารางทางซ้าย

ของคำสั่ง LEFT JOIN) เป็นหลัก แล้วนำตารางที่ 2 เข้ามาเชื่อมโยง ตามเงื่อนไขที่ได้ระบุไว้ มีรูปแบบการใช้งานดังนี้

```
SELECT fieldname
FROM tablename1 LEFT JOIN tablename2
ON tablename1.fieldname1 OPERATION tablename2.fieldname2
```

ตัวแปร fieldname หมายถึง ชื่อฟิลด์ที่อยู่ในตารางที่ 1 หรือ 2

ตัวแปร tablename1 หมายถึง ชื่อตารางที่ 1

ตัวแปร tablename2 หมายถึง ชื่อตารางที่ 2

ตัวแปร fieldname1 หมายถึง ชื่อฟิลด์ที่อยู่ในตารางที่ 1

ตัวแปร fieldname2 หมายถึง ชื่อฟิลด์ที่อยู่ในตารางที่ 2

ตัวแปร operation หมายถึง เงื่อนไขหรือตัวดำเนินการต่างๆที่ใช้ในการเชื่อมโยง

การเชื่อมโยงตารางโดยใช้คำสั่ง RIGHT JOIN

เป็นการเชื่อมโยง 2 ตารางเข้าด้วยกัน แต่กำหนดให้ตารางที่อยู่ทางด้านขวาของคำสั่ง Right Join เป็นหลัก แล้วนำเรCORDของตารางที่อยู่ทางซ้ายของคำสั่ง Right JOIN เข้ามารวมแสดง เฉพาะเรCORDที่ตรงตามเงื่อนไขที่ระบุไว้ มีรูปแบบการใช้งานดังนี้

```
SELSTC fieldname
FROM tablename1 RIGHT JOIN tablename2
ON tablename1.fieldname1 OPERATION tablename2.fieldname2
```

ตัวแปร fieldname หมายถึง ชื่อฟิลด์ที่อยู่ในตารางที่ 1 หรือ 2

ตัวแปร tablename1 หมายถึง ชื่อตารางที่ 1

ตัวแปร tablename2 หมายถึง ชื่อตารางที่ 2

ตัวแปร fieldname1 หมายถึง ชื่อฟิลด์ที่อยู่ในตารางที่ 1

ตัวแปร fieldname2 หมายถึง ชื่อฟิลด์ที่อยู่ในตารางที่ 2

ตัวแปร operation หมายถึง เงื่อนไขหรือตัวดำเนินการต่างๆที่ใช้ในการเชื่อมโยง

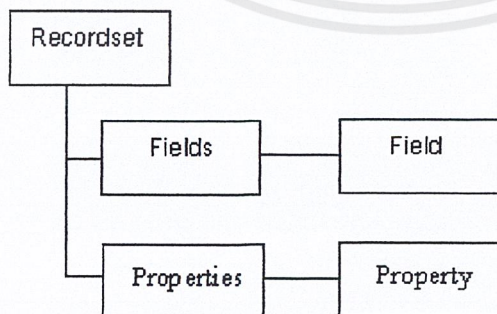
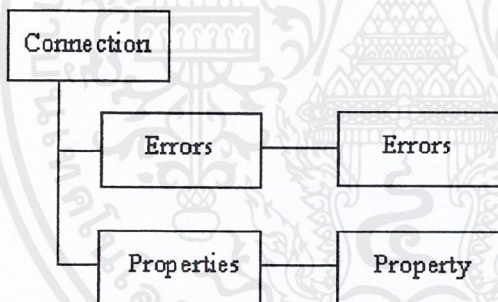
เทคโนโลยี ActiveX Data Object

ADO เป็นเทคโนโลยีที่มีแนวความคิดว่าจาก DAO บางส่วน กล่าวคือ จะมองฐานข้อมูลเป็นออบเจกต์เช่นกัน แต่จะใช้ OLEDB Provider เป็นตัวจัดการ ข้อมูลและโครงสร้างของฐานข้อมูลแทน โดยที่จะไม่นิยามออบเจกต์ขึ้นมา เพื่อแทนโครงสร้างของฐานข้อมูลแต่ละส่วน เหมือนกับ DAO แต่จะใช้วิธีการสร้าง OLEDB Provider ให้กับ RDBMS แต่ละชนิดแทน เช่นเมื่อฐานข้อมูลเป็นชนิด JET ก็จะใช้ Microsoft Jet OLEDB Provider ถ้าเป็น Oracle ก็จะใช้ Microsoft OLEDB Provider for Oracle เป็นต้น ทำให้ออบเจกต์ในโมเดล ADO ไม่ยุ่งยากซับซ้อนเหมือนออบเจกต์ DAO

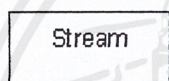
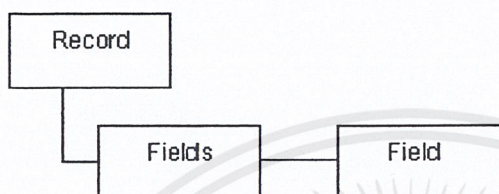
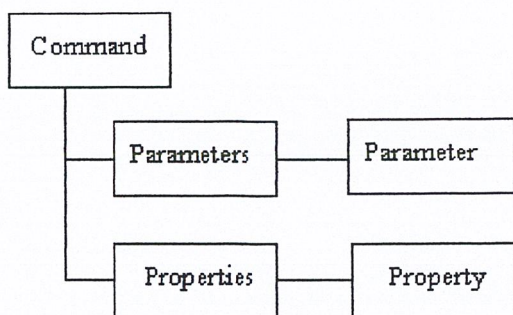
ใน Visual Basic สามารถเรียกใช้ OLEDB Provider ได้ 2 วิธีคือ

1. อาศัยคอนโทรล ADO Data (ADO Data Control) ร่วมกับกลุ่มคอนโทรล Bound & ActiveX Bound Control ที่มีคำว่า OLEDB ต่อท้าย
2. โดยการเรียกใช้งานกลุ่มออบเจกต์ ADO โดยตรง

โครงสร้างของออบเจกต์ ในโมเดลของ ADO 2.5



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



ชุดออบเจกต์ใน ADO 2.5

สำหรับโมเดลของ ADO 2.5 จะประกอบด้วยออบเจกต์ 9 ตัว คือ

| | |
|---------------------|--|
| ออบเจกต์ Command | เป็นออบเจกต์ตัวกลางที่ใช้สำหรับส่งคำสั่ง (Execute) ไปยังฐานข้อมูล เป้าหมายที่ต้องการติดต่อ มีศัพท์เรียกฐานข้อมูลดังกล่าวว่า Data Sources |
| ออบเจกต์ Connection | ใช้สำหรับเริ่มต้นเปิดการเชื่อมต่อกับฐานข้อมูล |
| ออบเจกต์ Error | เป็นออบเจกต์ที่เก็บข้อมูลเกี่ยวกับ ความผิดพลาดของการเข้าถึงข้อมูลของ OLEDB Provider |
| ออบเจกต์ Field | ใช้สำหรับเก็บฟิลด์ใดฟิลด์หนึ่งในฐานข้อมูล |
| ออบเจกต์ Parameter | ใช้สำหรับรับ - ส่งตัวแปร หรืออาร์กิวเมนต์ระหว่างการทำคิวรี หรือ Stored Procedure ของออบเจกต์ Command |
| ออบเจกต์ Property | ใช้สำหรับกำหนดคุณสมบัติต่างๆของ ออบเจกต์ ADO |
| ออบเจกต์ Record | ใช้สำหรับเก็บเรคคอร์ดใดเรคคอร์ดหนึ่ง ที่อยู่ในออบเจกต์ Recordset |
| ออบเจกต์ Recordset | เป็นผลลัพธ์ที่ได้มาจากฐานข้อมูล ประกอบด้วยกลุ่มของเรคคอร์ด และฟิลด์ที่มาจากตารางที่เชื่อมต่อเป็นออบเจกต์ที่มีความสำคัญมากที่สุด |

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ออบเจกต์ Stream เป็นออบเจกต์ที่ทำหน้าที่เป็นตัวกลาง สำหรับรับ - ส่งข้อมูลข้อความ (Stream of Text) ให้กับระบบไฟล์ข้อความ File System Object (FSO) เช่น การสร้างไฟล์ *.txt โดย ASP จะอาศัยออบเจกต์ตัวนี้ ทำหน้าที่ลำเลียงข้อมูลนั่นเอง

วิธีการติดต่อกับฐานข้อมูลของ MS –Visual Basic 6.0

โปรแกรม MS –Visual Basic 6.0 คือมีความสามารถในการจัดการกับฐานข้อมูล ได้ โดยจะมีวิธีการจัดการอยู่ 2 แบบ คือ

1. การใช้คำสั่งคอนโทรล (Data Control) เป็นวิธีการที่ง่ายและสะดวกที่สุดใน การติดต่อกับฐานข้อมูล เนื่องจากคำสั่งคอนโทรลจะติดต่อกับฐานข้อมูลและจัดการกับฐานข้อมูล ในตารางอัตโนมัติเช่น การเปิดฐานข้อมูล การแสดงและแก้ไขข้อมูลในตาราง อย่างไรก็ตามการใช้ คำสั่งคอนโทรลยังมีข้อจำกัดอยู่พอสมควร เช่น ไม่มีฟังก์ชันฝนกสรลบข้อมูล

2. การใช้คำสั่งออบเจกต์ (Data Object) วิธีนี้จะต้องเขียนโปรแกรมเพื่อติดต่อกับ ฐานข้อมูล โดยการใช้ออบเจกต์ต่างๆ ที่ VB6 มีมาให้โดยการเพิ่ม ลบ หรือแก้ไขข้อมูล จะต้อง เขียนโปรแกรมจัดการเอง แต่ข้อดีของวิธีนี้คือสามารถติดต่อกับข้อมูลจากหลายๆ ตารางพร้อมกัน สามารถสร้างคิวรีตอนรันโปรแกรมได้ และสามารถควบคุมความผิดพลาดต่างๆ ได้ดีกว่าใช้คำสั่ง คอนโทรล รวมทั้งสามารถใช้ภาษา SQL เพื่อจัดการกับฐานข้อมูลได้อีกด้วย

การจัดการเกี่ยวกับฐานข้อมูล

โดยปรกติการสร้างฐานข้อมูลเพื่อจะนำมาใช้งานในวิซวลเบสิกนั้น ทำได้ 4 วิธี หลักๆ คือสร้างโดยอาศัยเครื่องมือของระบบการจัดการฐานข้อมูลนั้นวิธีนี้จะแตกต่างกันไปตามรูปแบบของฐานข้อมูลที่ใช้ เช่น หากต้องการใช้งานฐานข้อมูลของ dBASE ก็ใช้เครื่องมือของ dBASE เป็นตัวสร้าง

การใช้โปรแกรม Data Manager

โปรแกรม Data Manager จะมีมากับ VB 6 อยู่แล้ว โปรแกรมนี้จะใช้สร้างฐานข้อมูลในรูปแบบของ Jet 1.1 หรือ Access ได้เพียงอย่างเดียวเท่านั้น โดยที่เราไม่จำเป็นต้องมีโปรแกรม Access ตัวโปรแกรม Data Manager นั้นอาจมีความสามารถจำกัด ทำได้เพียงการสร้าง ลบ แสดง รายละเอียดของตารางและดัชนีต่างๆ ในฐานข้อมูลเท่านั้น ไม่สามารถทำการแก้ไขหรือเปลี่ยนแปลงรายละเอียดของตารางเดิมที่มีอยู่แล้ว

การใช้ Data Control

Data Control นับเป็น คอนโทรลประเภทหนึ่งที่สามารถจัดวางลงฟอร์มของวิซวลเบสิคได้และใช้ในการจัดเก็บข้อมูลได้โดยไม่ต้องเขียน โปรแกรมเพิ่ม วิธีการนี้แม้สะดวกแต่มีข้อจำกัดอยู่บ้าง

การใช้ชุดคำสั่งในการจัดการฐานข้อมูล (DDL) กับออบเจกต์ฐานข้อมูลต่างๆ

วิธีการนี้เหมาะสำหรับเมื่อเราต้องการสร้างฐานข้อมูลใหม่ ในระหว่างการทำงานของโปรแกรม จัดเป็นวิธีที่ยุงยากมากที่สุด แต่ก็ให้ความยืดหยุ่นดี เพราะสามารถสร้าง ลบ หรือแก้ไขรายละเอียดต่างๆ ได้ในระหว่างการทำงานและในการจัดการกับออบเจกต์แบบที่เป็น QueryDef ได้ด้วยเช่นเดียวกัน

การใช้งานดาต้าคอนโทรล (Data Control)

ดาต้าคอนโทรลเป็นเครื่องมือที่ช่วยให้สามารถจัดการกับข้อมูลได้ในตาราง เช่น การเพิ่ม ลบ แก้ไข หรือการค้นหาข้อมูลที่ต้องการ เป็นต้น

คุณสมบัติที่สำคัญสำหรับดาต้าคอนโทรล

DataSource เป็นกากำหนดชื่อของดาต้าคอนโทรลที่ต้องการเชื่อมโยง

DataField เป็นการกำหนดว่าเราต้องการให้แสดงหรือแก้ไขข้อมูลฟิลด์ใดในตาราง

DatabaseName เป็นที่อยู่ของไฟล์ฐานข้อมูลที่ต้องการ โดยมีการระบุชื่อฟิลด์นั้น

Connect เป็นคุณสมบัติที่ให้ออกว่าต้องการติดต่อกับฐานข้อมูลประเภทใด เช่น MS-Access

RecordSetType เป็นชนิดของเรคคอร์ดเซตที่ต้องการเข้าถึงในฐานข้อมูลแบ่งได้เป็น 3 ชนิด

1. Table สามารถติดต่อกับตารางใดตารางหนึ่งในฐานข้อมูลเท่านั้น ซึ่งการเข้าถึงข้อมูลสามารถเข้าถึงได้อย่างรวดเร็ว เนื่องจากจะใช้อินเด็กซ์เป็นตัวค้นหาข้อมูลส่วนข้อจำกัดคือ ไม่สามารถดึงข้อมูลบางส่วนจากตารางได้ จะต้องดึงมาทั้งตารางเท่านั้น

2. Dynaset สามารถใช้ติดต่อกับข้อมูลหลายๆ ตารางได้ เราสามารถใช้คำสั่ง SQL เพื่อดึงข้อมูลบางส่วนจากตาราง รวมทั้งสามารถแก้ไขข้อมูลได้ แต่การทำงานช้ากว่าการใช้ Table

3. Snapshot จะเหมือนกับ Dynaset แต่การทำงานจะเร็วกว่า เนื่องจากจะเก็บข้อมูลทุกอย่างอยู่ในหน่วยความจำ และไม่สามารถแก้ไขข้อมูลได้

RecordSource เป็นการกำหนดชื่อของTable Danset หรือ Snapshot ที่เราต้องการติดต่อ

ออบเจ็กต์เรคคอร์ดเซต (RecordSet) และการจัดเก็บข้อมูล

คุณสมบัติ (Property) ของเรคคอร์ดเซต

| | |
|-----------------|---|
| Connect | โดยปกติจะว่าง แต่หากมีค่าจะหมายถึงค่าของ Attached Table ในกรณีที่เป็น ตาราง |
| Datacreated | ที่ตารางที่เกี่ยวข้องนี้ถูกสร้างขึ้นมา |
| LastUpdate | วันมีการแก้ไขครั้งสุดท้าย |
| Name | แสดงชื่อของตาราง |
| SourceTableName | ชื่อของตารางที่อ้างอิงบนฐานข้อมูลอื่น ในกรณีที่เป็นตารางซึ่งกำหนดบน ตารางฐานข้อมูลอื่น |
| Updatable | หากว่ามีค่าความเป็นจริง ก็หมายความว่ายอมให้มีการแก้ไขเปลี่ยนแปลง ข้อมูลได้ |
| EOF | เป็นคุณสมบัติที่แสดงว่า ได้มีการเลื่อนตำแหน่งเรคคอร์ดไปข้างหน้าผ่าน เรคคอร์ดสุดท้ายในไฟล์หรือไม่ โดยจะมีค่าเป็นจริงหรือเท็จ |
| BOF | เป็นคุณสมบัติที่แสดงว่า ได้มีการเลื่อนตำแหน่งเรคคอร์ดย้อนหลังผ่านเรคคอร์ดแรกในไฟล์หรือไม่ โดยจะมีค่าเป็นจริงหรือเท็จ |

ฟังก์ชันการทำงาน (Method) ของเรคคอร์ดเซต

| | |
|--------------|--|
| MoveFirst | เป็นการสั่งเลื่อนตัวชี้ตำแหน่งในออบเจ็กต์เรคคอร์ดเซตนั้นให้ ไปชี้ที่ตำแหน่ง เรคคอร์ดแรกสุด |
| MoveNext | เพื่อเลื่อนตัวชี้ตำแหน่ง ไปยังเรคคอร์ดถัดไปจากตำแหน่งปัจจุบัน |
| MovePrevious | เพื่อเลื่อนตัวชี้ตำแหน่ง ไปยังเรคคอร์ดก่อนหน้าตำแหน่งปัจจุบัน |
| MoveLast | เลื่อนตำแหน่ง ไปยังเรคคอร์ดสุดท้ายในเรคคอร์ดเซต มีรูปแบบดังนี้ Recordset_Object.[Move first] |
| FindFirst | เลื่อนไปยังตำแหน่งเรคคอร์ดที่มีข้อมูลในลักษณะตามที่ต้องการ โดยเริ่ม ค้นหาจากเรคคอร์ดแรกไปเรคคอร์ดสุดท้าย |
| FindNext | เลื่อนไปยังตำแหน่งเรคคอร์ดถัดไปจากตำแหน่งปัจจุบัน ที่มีข้อมูลตาม ลักษณะที่ต้องการ |
| FindPrevious | เหมือนกับฟังก์ชัน FindNext แต่เป็นการค้นหาในทิศทางย้อนกลับ |

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

| | |
|----------|---|
| Nomatch | <p>หากพบเรคคอร์ดที่มีข้อมูลในลักษณะที่กำหนด ค่าของคุณสมบัตินี้จะเป็น False แต่ถ้าหากค้นหาไม่พบ จะมีค่าเป็น True และตำแหน่งของตัวชี้จะอยู่ที่เดิม</p> <p>มีรูปแบบการใช้งาน</p> <p><code>Recordset_Object.[FindFirst]Criteria</code></p> |
| Criteria | <p>เป็นตัวแปรหรือค่าคงที่ประเภทตัวอักษร(String)เพื่อกำหนดว่าต้องการจะค้นหาข้อมูลในลักษณะอย่างไร วิธีการกำหนดค่าของ Criteria จะมีหลักการเหมือนกับการกำหนดค่าของ Where ในคำสั่ง Select ของ SQL เพียงแต่ไม่มีคำว่า Where ประกอบด้วยเท่านั้น</p> <p>รูปแบบมีดังนี้</p> <p>ฟิลด์ที่มีชนิดเป็นตัวเลข และตัวอักษร</p> <p><code>Field_Name [=, <>, <, >=, <=] Value หรือ Field_Name in (Value,Value,...)</code></p> |
| Seek | <p>ฟังก์ชันนี้ใช้ได้กับเรคคอร์ดเซตประเภทตารางอย่างเดียวเท่านั้น โดยจะทำการค้นหาข้อมูลที่ต้องการจากครรชนี ต้องกำหนดว่าจะให้ค้นหาจากครรชนีตัวใดของตาราง วิธีการนี้จะค้นหาข้อมูลได้เร็วกว่าการใช้ฟังก์ชัน Find เป็นอย่างมาก</p> <p>รูปแบบการใช้งาน</p> <p><code>Table_Object.Index = Index_Name</code></p> <p><code>Table_Object.Seek Comparison_Operator,Comparison_Value</code></p> <p>โดยที่ ;Index_Name คือตัวแปรหรือค่าคงที่ๆเป็นชนิดตัวอักษรมีค่าเป็นชื่อของครรชนี</p> <p>Comparison_Operator เป็นตัวแปรหรือค่าคงที่ชนิดตัวอักษรระบุวิธีการเปรียบเทียบจะกำหนดได้เพียง = , <> , > , < , >= , <= เท่านั้น</p> <p>Comparison_Value เป็นตัวแปรที่มีชนิดตามชนิดของฟิลด์ซึ่งระบุค่าที่ต้องการค้นหา</p> |

การอ่านค่าหรือกำหนดค่าของฟิลด์ในเรคคอร์ดในตำแหน่งนั้นๆ

รูปแบบคำสั่ง

`Recordset_Object.Fields("Field_Name") . Value`

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

อีกรหัสโดยการระบุถึงลำดับของฟิลด์ในออบเจกต์เรคคอร์ดเซ็ทนั้น

รูปแบบคำสั่ง

Recordset_Object.Fields(Field_Ordinal_Number)

การกำหนดค่าให้กับฟิลด์

รูปแบบคำสั่ง

Recordset_Object.Field("Field_Name").Value = Value

การเพิ่มข้อมูลใหม่

ทำได้โดยใช้คำสั่ง AddNew และ Update

รูปแบบคำสั่ง

Recordset_Object.AddNew ; เป็นการสั่งให้เพิ่มเรคคอร์ดว่างๆหนึ่งเรคคอร์ด

Recordset_Object.Fields("Field_Name") = Value ; เป็นคำสั่งสำหรับกำหนดค่าให้กับฟิลด์ต่างๆตามต้องการ

Recordset_Object.Update ; เป็นการสั่งแก้ไขค่าในฐานข้อมูลตามที่กำหนด

การแก้ไขข้อมูลเดิม

ทำได้โดยใช้คำสั่ง Edit และ Update

รูปแบบคำสั่ง

Recordset_Object.Edit ; เป็นการประกาศว่าจะเริ่มแก้ไขข้อมูลที่ตำแหน่งของตัวชี้

Recordset_Object.Field("Field_Name") = New_Value ; เป็นการกำหนดค่าให้กับฟิลด์ที่ต้องการจะเปลี่ยนค่า

Recordset_Object.Update ; เป็นการสั่งให้เปลี่ยนข้อมูลในฐานข้อมูลได้แล้ว

การแก้ไขข้อมูลเดิม

ทำได้โดยใช้คำสั่ง Delete

รูปแบบคำสั่ง

Recordset_Object.Delete

Uptable เป็นการตรวจสอบว่าออบเจกต์เรคคอร์ดเซ็ทนั้นสามารถเปลี่ยนแปลงแก้ไขได้

หรือไม่โดย จะมีค่าเป็น True หรือ False

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.3 การติดต่อสื่อสาร

2.3.1 พอร์ตอนุกรม (Serial Port)

พอร์ตอนุกรม RS-232-C จะเป็นพอร์ตของเครื่องคอมพิวเตอร์ที่มีขาต่อ (connector) ทั้งประเภท 9 ขาและ 25 ขา และเราเรียกกันว่า COM1: และ COM2: นั่นเอง ในความเป็นจริงพอร์ตอนุกรมไม่ได้ถูกควบคุมโดยตรงจาก CPU บนเมนบอร์ด แต่การสื่อสารทั้งหมดจะถูกจัดการโดยชิป UART (Universal Asynchronous Receiver/Transmitter) อีกทีหนึ่ง ซึ่งในปัจจุบันเบอร์ที่ใช้กันมากที่สุดคือเบอร์ 16550C ซึ่งเป็นเวอร์ชันที่ได้รับการแก้ไขข้อผิดพลาดแล้ว ซึ่งชิป UART นี้จะทำหน้าที่ในการรับและการส่งข้อมูลดังต่อไปนี้

การส่งข้อมูล (Data Transmission)

- รับตัวอักษรจากเครื่องคอมพิวเตอร์
- แปลงตัวอักษรให้เป็นสายข้อมูลแบบบิต (เราเรียกว่ากระบวนการ serialization)
- สร้างเฟรมข้อมูลโดยการเพิ่มบิตที่จำเป็นสำหรับการสื่อสารและการตรวจสอบ เช่น บิต START, STOP และ PARITY เป็นต้น
- ส่งผ่านเฟรมข้อมูลที่สร้างขึ้นมาแล้วจากขั้นตอนที่ผ่านมา ด้วยความเร็วของโมเด็มหรือพอร์ตอนุกรม (Data Receiver)

การรับข้อมูล (Data Receiver)

- รับตัวอักษรจากอินเตอร์เฟซ
- ตรวจสอบความถูกต้องของเฟรมข้อมูลตามมาตรฐานที่กำหนด โดยถ้าหากเฟรมข้อมูลมีรูปแบบที่ไม่ถูกต้องก็จะมีอาการแจ้งเตือนทันที
- ตรวจสอบความถูกต้องของพาริตี
- แปลงสายข้อมูลแบบบิตเป็นตัวอักษร
- ส่งตัวอักษรให้กับเครื่องคอมพิวเตอร์
- แสดงสถานะความพร้อมที่จะรับข้อมูลตัวอักษรถัดไปให้กับอินเตอร์เฟซ

สำหรับการเชื่อมต่อสัญญาณต่างๆ ระหว่าง CPU ของเมนบอร์ดของเครื่องคอมพิวเตอร์กับพอร์ตอนุกรมนั้น จะต้องทำผ่านชิป UART

Transmit Data (TD)

เป็นวงจรที่สร้างสัญญาณ transmitted data ซึ่งถูกส่งจาก DTE ไปยัง DCE ซึ่งสัญญาณที่ส่งออกมาจะเป็นไค้คคำสั่ง โมเด็มหรือข้อมูลก็ได้

Received Data (RD)

เป็นวงจรที่สร้างสัญญาณ received data ซึ่งถูกส่งจาก DCE ไปยัง DTE ซึ่งสัญญาณที่ส่งออกมาจะเป็นไค้คคำสั่งของ โมเด็มหรือข้อมูลก็ได้ ซึ่งเป็นสัญญาณที่มีทิศทางตรงข้ามกับสัญญาณ transmitted data

Data Terminal Ready (DTR)

สัญญาณ DTR จะถูกส่งจาก DTE ไปยัง DCE เพื่อเป็นการแจ้งความพร้อมในการสื่อสารให้ โมเด็มได้ทราบ โดยหากโมเด็ม (อุปกรณ์ DCE) มีความสามารถในการตอบรับแบบอัตโนมัติ (automatically answer) โมเด็มก็จะสามารถตอบรับได้เฉพาะเมื่อมีสัญญาณ DTR อยู่สถานะ on เท่านั้น

Carrier Detect (CD)

สัญญาณ CD จะถูกส่งจาก DCE ไปยัง DTE เพื่อเป็นการแจ้งว่าโมเด็มอยู่ในสถานะกำลังติดต่อกับ โมเด็มตัวอื่น หรือโมเด็มกำลังได้รับสัญญาณที่พร้อมสำหรับการติดต่อสื่อสาร สำหรับสัญญาณ Carrier Detect นี้ สามารถเรียกอีกชื่อได้ว่า Received Line Signal Detector

Request To Send (RTS)

สัญญาณ RTS จะถูกส่งออกจาก DTE ไปยัง DCE โดยเมื่อสัญญาณ RTS อยู่ในสถานะ on ก็หมายถึง เครื่องคอมพิวเตอร์พร้อมที่จะรับข้อมูลจากโมเด็ม และในทางกลับกันหากสัญญาณ RTS อยู่ในสถานะ off ก็หมายถึง เครื่องคอมพิวเตอร์ไม่พร้อมที่จะรับข้อมูลจากโมเด็ม

Clear To Send (CTS)

สัญญาณ CTS จะถูกส่งจาก DCE ไปยัง DTE ซึ่งเป็นสัญญาณที่ทำหน้าที่ตรงข้ามกับสัญญาณ RTS โดยเมื่อสัญญาณ CTS อยู่ในสถานะ on ก็หมายถึง โมเด็มพร้อมที่จะรับข้อมูลจากเครื่องคอมพิวเตอร์ และในทางกลับกันหากสัญญาณ CTS อยู่ในสถานะ off ก็หมายถึงโมเด็มไม่พร้อมที่จะรับข้อมูลจากเครื่องคอมพิวเตอร์

Data Set Ready (DSR)

สัญญาณ DSR จะถูกส่งจาก DCE ไปยัง DTE เพื่อเป็นการแจ้งความพร้อมในการสื่อสารจากโมเด็มให้กับเครื่องคอมพิวเตอร์ได้ทราบ โดยสัญญาณ DSR จะอยู่ในสถานะ on ก็ต่อเมื่อโมเด็มได้รับสัญญาณ DTR เท่านั้น

Ring Indicator (RI)

สัญญาณ RI จะถูกส่งจาก DCE ไปยัง DTE เพื่อเป็นการแจ้งให้เครื่องคอมพิวเตอร์ทราบว่าโมเด็มกำลังได้รับสัญญาณกระดิ่ง (ring signal) จากโมเด็มตัวอื่น โดยที่สัญญาณ RI จะอยู่ในสถานะ on โมเด็มได้รับสัญญาณกระดิ่ง และ off เมื่อโมเด็มไม่ได้รับสัญญาณกระดิ่ง เนื่องจากโมเด็มรุ่นใหม่ๆ ในปัจจุบันมักจะมีความสามารถในการตอบรับแบบอัตโนมัติ (automatical answer) ดังนั้นจึงไม่มีความจำเป็นต้องใช้สัญญาณ RI อีกต่อไป

รูปแบบของเฟรมข้อมูล (Data Format)

เฟรมข้อมูลในการส่งหรือรับจะประกอบด้วยบิตข้อมูลที่มีความหมายดังต่อไปนี้

บิตข้อมูล (Data Bit)

เมื่อชิป UART ได้รับตัวอักษรที่ส่งมาจากเครื่องคอมพิวเตอร์แล้ว ก็ต้องทำการแปลงตัวอักษรดังกล่าวให้เป็นสายข้อมูลชนิดบิตที่มีความยาวตั้งแต่ 5 ถึง 8 บิต ซึ่งเราเรียกขบวนการแปลงตัวอักษรให้เป็นสายข้อมูลชนิดนี้ว่า Serialization จากนั้นโมเด็มก็จะทำการส่งแต่ละบิตไปยังโมเด็มปลายทาง โดยจะเริ่มต้นส่งจากบิตที่มีนัยสำคัญต่ำสุด (least significant bit) ไปยังบิตที่มีนัยสำคัญสูงสุด (most significant bit)

บิตเริ่มต้นข้อมูล (START Bit)

ในการส่งข้อมูลแบบ asynchronous นั้น เราจะต้องมีวิธีการบอกโมเด็มให้ทราบถึงจุดเริ่มต้นของข้อมูลที่ต้องการส่ง ดังนั้นก่อนหน้าข้อมูลทุกๆ เฟรมจะต้องถูกนำหน้าด้วยบิตเริ่มต้นข้อมูล (start bit) เสมอ

บิตสิ้นสุดข้อมูล (STOP Bit)

ในการส่งข้อมูลแบบ asynchronous นั้น ในกรณีที่บิตเริ่มต้นข้อมูลเกิดการสูญหายในระหว่างการส่ง โมเด็มก็จะไม่สามารถทราบถึงจุดสิ้นสุดของสายข้อมูลบิต ได้เลย นอกเสียจากโมเด็มจะตรวจบิตเริ่มต้นใหม่อีกครั้งเท่านั้น ดังนั้นจึงมีการเพิ่มบิตสิ้นสุดข้อมูล ต่อท้ายทุกๆ ข้อมูลของแต่ละตัวอักษรเพื่อแจ้งการสิ้นสุดของสายข้อมูลบิต โดยการที่เรียงกันของบิตเริ่มต้นข้อมูล สายข้อมูล และบิตสิ้นสุดข้อมูลนี้เราเรียกว่า เฟรมข้อมูล (data frame)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้คัดลอกเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บิตพาร์ตี (PARTY Bit)

เนื่องจากการส่งผ่านข้อมูลทางสายสัญญาณนั้น สามารถเกิดสัญญาณรบกวนได้ง่าย ด้วยเหตุนี้จึงเป็นไปได้มากที่สถานะของแต่ละบิตของข้อมูลที่ถูกส่งจะมีการเปลี่ยนแปลง เช่น จากบิต 0 เป็นบิต 1 เป็นต้น ในระหว่างการส่งผ่านข้อมูล ด้วยเหตุนี้จึงมีการคิดค้นวิธีที่ง่ายที่สุดที่ช่วยในการตรวจสอบความถูกต้องของสถานะของบิตข้อมูล ซึ่งเรียกกันว่า การตรวจสอบค่าพาร์ตี (Party Check) โดยในการส่งผ่านข้อมูลด้วยโปรโตคอล START/STOP นั้น ในการส่งตัวอักษรข้อความทั่วไปจะใช้เพียง 7 บิตข้อมูลเท่านั้น ดังนั้นจึงมีการเพิ่มบิตพาร์ตีต่อสายเฟรมข้อมูลที่ถูกสร้างขึ้นมาเพื่อตรวจสอบสถานะของผลบวกของบิตที่เป็น 1 ของสายบิตข้อมูลของสายแต่ละตัวอักษร สำหรับการคำนวณค่าของพาร์ตีมีดังนี้

1. ถ้าหากจำนวนของบิตที่มีค่าเท่ากับ 1 ของสายบิตข้อมูลมีค่าเป็นเลขคู่ (even number) บิตพาร์ตี จะมีค่าเท่ากับ 0
2. ถ้าหากจำนวนของบิตที่มีค่าเท่ากับ 1 ของสายบิตข้อมูลมีค่าเป็นเลขคี่ (odd number)

คอนโทรลที่ใช้ในการติดต่อสื่อสารของโปรแกรม Visual Basic

2.3.2 คอนโทรล MSComm

คอนโทรล MSComm (Communications) เป็นคอนโทรลตัวหนึ่งที่ช่วยในการติดต่อสื่อสารกับพอร์ตอนุกรม (serial port) ซึ่งเป็นคอนโทรลที่ทำงานโดยมีการตอบสนองต่อเหตุการณ์แบบ even-driven นั่นก็คือคอนโทรลจะทำหน้าที่ตรวจสอบการเกิดขึ้นหรือการร้องขอให้เกิดเหตุการณ์ต่างๆ กับพอร์ตอนุกรม โดยอัตโนมัติ และจะมีการแจ้งเตือนให้ผู้อ่านได้ทราบโดยผ่านทางโพธิ์ซีเอร์เหตุการณ์ เช่นเดียวกับคอนโทรลทั่วไปของ Visual Basic นั่นเอง การนำ MSComm มาใช้งานมีรายละเอียดดังนี้

1. ต้องมี MSComm ซึ่งจะมียู่แล้วเมื่อตอนติดตั้ง Visual Basic
2. กำหนด Custom Control MSCOMM.OCX (สำหรับ Visual Basic4 ขึ้นไป) โดยการเลือก MSCOMM.OCX จากเมนู Project/Components/ที่ Control เลือก Microsoft Comm Control 6.0 (กรณีที่ใช้ Version6) ก็จะได้ MSComm รูปโทรศัพท์ปรากฏในทูลบ็อกซ์
3. นำคอนโทรลไปใส่ไว้ใน Form ที่ต้องการใช้งาน
4. กำหนดคุณสมบัติต่างๆ ให้กับ Control เขียนฟังก์ชันที่เกี่ยวข้องกับ MSComm ในการใช้ MSComm ติดต่อกับ Serial Port นั้นมีการติดต่อกัน 2 แบบคือ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.1 การสื่อสารกระตุ้นด้วยเหตุการณ์ (Even-Driven Communication) หรือเป็นการเขียนกระตุ้นด้วยการอินเตอร์รัปชั่นเอง โดยเมื่อข้อมูลเข้ามาจะเกิด CommEvent ขึ้นกับ OnComm Event

4.2 การสื่อสารแบบโพลลิ่ง (Polling) โดยการวนรอบตรวจสอบข้อมูลจาก Serial Port ตลอดเวลา

การกำหนด Property ที่สำคัญ

ComPort คือการกำหนดหมายเลขของ Port ในการติดต่อ RS232 เช่น Com1, Com2

รูปแบบการใช้งาน

```
object.ComPort [= value]
```

```
MSComm1.Comport = 1
```

Setting คือการกำหนดพารามิเตอร์ในการสื่อสารผ่านพอร์ตอนุกรม

รูปแบบการใช้งาน

```
object.Setting [= value]
```

```
MSComm1.Setting = "9600,e,7,2"
```

Value หมายถึง ข้อมูลชนิดสตริง ที่กำหนดพารามิเตอร์ในการสื่อสารผ่านพอร์ตอนุกรม สำหรับคุณสมบัติ Setting จะต้องเรียงลำดับดังนี้

```
"BBBB,P,D,S"
```

เมื่อ ; BBBB คือความเร็วของการส่งถ่ายข้อมูลในหน่วยของ baud rate สำหรับค่า baud rate คอนโทรล MSComm สามารถปรับได้มีค่าดังต่อไปนี้ 110,300,600,1200,2400,9600,14400,19200,28800,38400,56000,128000 และ 256000 ซึ่งค่า 9600 จะเป็นค่า default

; P คือบิตพาริตี (Parity Bit) สำหรับใช้ในการตรวจสอบความถูกต้องของข้อมูล ซึ่งมีความหมายดังต่อไปนี้

| พารามิเตอร์ | ความหมาย |
|-------------|---------------|
| E | Even |
| M | Mark |
| N | None(default) |
| O | Odd |

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่นิยมนำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

S Space

; D คือขนาดของบิตข้อมูล ซึ่งสามารถมีค่าได้ดังต่อไปนี้

4,5,6,7 หรือ 8 (default)

; S คือขนาดของบิตหยุด (stop bit) ซึ่งสามารถมีค่าเป็น

1(default) หรือ 2

PortOpen เป็นการกำหนดสถานะการเปิด (open) หรือ ปิด (close) ของพอร์ต
อนุกรมของเครื่องคอมพิวเตอร์

รูปแบบการใช้งาน

Object.PortOpen [=boolean]

MSComm1.PortOpen = true

Boolean หมายถึงข้อมูลชนิดบูลีน ที่กำหนดสถานะของพอร์ต
อนุกรม ดังต่อไปนี้

True หมายถึง พอร์ตอนุกรมถูกเปิด

False หมายถึง พอร์ตอนุกรมถูกปิด

Input เป็นการย้ายข้อมูลจากบัฟเฟอร์รับข้อมูล

รูปแบบการใช้งาน

Object.Input

MSComm1.Input = ข้อมูลที่จะรับจากพอร์ต

Output เป็นการส่งข้อมูลให้กับบัฟเฟอร์ส่งข้อมูลส่งออก

รูปแบบการใช้งาน

Object.Output [=value]

MSComm1.output = เท่ากับข้อมูลที่จะส่งออกพอร์ต

โพลีเมอร์เหตุการณ์ ONComm ; เกิดขึ้นเมื่อมีการเปลี่ยนแปลงค่าของคุณสมบัติ
CommEvent ซึ่งเป็นการบอกถึงการเกิดข้อผิดพลาดหรือมีการสื่อสารเกิดขึ้นก็ได้

รูปแบบ โพลีเมอร์เหตุการณ์

Private Sub object_OnComm0

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.4 การจัดการระบบเครือข่าย (LAN)

LAN ย่อมาจาก Local Area Network ประกอบด้วยส่วนที่เป็นทั้งฮาร์ดแวร์และซอฟต์แวร์ที่ทำงานร่วมกัน เพื่อการส่งถ่ายข้อมูลในระบบดิจิทัลระหว่างคอมพิวเตอร์ ซึ่งประโยชน์ของแลนนีก็มีอยู่หลายอย่างด้วยกัน ตัวอย่างเช่น

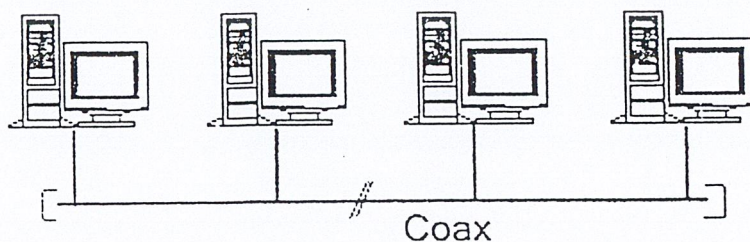
- การแชร์ข้อมูลร่วมกันระหว่างผู้ใช้คอมพิวเตอร์ในระบบเน็ตเวิร์คเดียวกัน
- การใช้ทรัพยากรทางฮาร์ดแวร์ร่วมกัน เช่น การแชร์เครื่องพริ้นเตอร์
- ความถูกต้องของข้อมูล หากมีข้อมูลที่ต้องทำการเปลี่ยนแปลงอยู่เสมอก็จะสามารถอัปเดตข้อมูลเก็บไว้ในเครื่องคอมพิวเตอร์ทำให้ได้ข้อมูลที่ต้องการอยู่เสมอ
- การติดต่อสื่อสารระหว่างผู้ที่เชื่อมต่อเครือข่าย ผู้ที่อยู่ในระบบเน็ตเวิร์คเดียวกันสามารถที่จะสื่อสารกันได้โดยใช้ซอฟต์แวร์ที่จัดการด้านนี้ผ่านทางหน้าจอคอมพิวเตอร์ของตน

2.4.1 อุปกรณ์สำคัญที่ต้องใช้ในระบบเน็ตเวิร์ค

1. การ์ดแลนหรือการ์ดเน็ตเวิร์ค(LAN card or Network adaptor card)

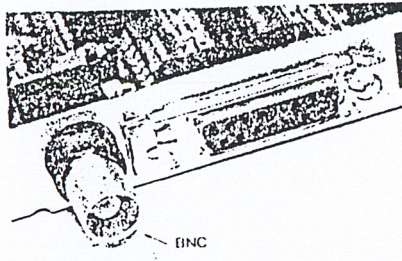
เราต้องติดตั้งการ์ดแลนในเครื่องคอมพิวเตอร์ทุกเครื่อง ที่ต้องการต่อเข้ากับเน็ตเวิร์ค โดยสังเกตที่ด้านหลังการ์ดแลนจะมีขั้วต่อสายเคเบิลหรือสายแลนอยู่ 2 แบบด้วยกันคือ ขั้วต่อแบบ BNC และขั้วต่อแบบ RJ-45 ซึ่งทั้งสองแบบต่างก็มีจุดเด่นและจุดด้อยต่างกันไป

การ์ดแลนที่ใช้ขั้วต่อแบบ BNC นั้นจะมีข้อได้เปรียบที่จะสามารถต่อจากเครื่องคอมพิวเตอร์เครื่องหนึ่งได้โดยไม่ต้องใช้อุปกรณ์เสริมใดๆ ซึ่งเราเรียกวิธีการต่อสายแบบนี้ว่าการต่อเน็ตเวิร์คแบบบัส ข้อจำกัดของการใช้การ์ดแลนแบบนี้คือความยาวของสายแลนที่นำมาต่อซึ่งหากนำมายึดเป็นเส้นตรงแล้วจะต้องมีความยาวไม่เกิน 200 เมตร และหากสายแลนของเครื่องคอมพิวเตอร์ใดมีปัญหาจะทำให้คอมพิวเตอร์เครื่องอื่นๆทุกเครื่องไม่สามารถใช้งานระบบเน็ตเวิร์คได้เลย



รูปที่ 2.5 การต่อเน็ตเวิร์คแบบบัส

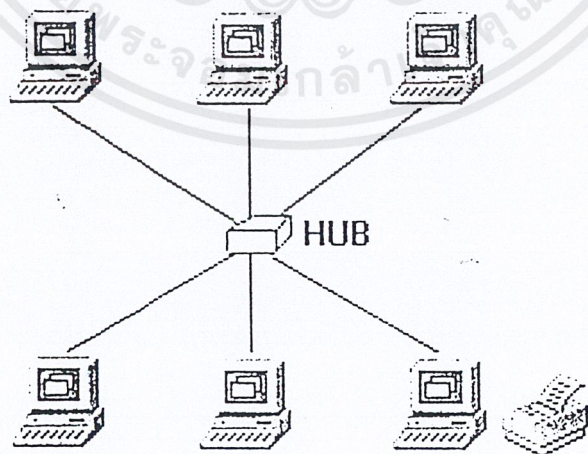
ขั้วต่อสาย BNC จะมีลักษณะเป็นกรวยเหล็กกลมๆที่ยื่นออกมาจากด้านหลังเอกตัวการ์ด ถ้ามองเข้าไปข้างในจะเห็นว่ามีส่วนเหล็กเล็กๆอยู่ตรงกลาง อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.6 ขั้วต่อแบบ BNC

สำหรับสายที่จะใช้กับขั้วต่อแบบ BNC นี้จะมีลักษณะเหมือนสายรับสัญญาณของโทรทัศน์ ซึ่งมีชื่อเรียกเฉพาะคือสาย Coaxial โดยสายประเภทนี้จะมี 4 ชั้น คือ ชั้นนอกสุดเป็นฉนวนชั้นที่สองถัดเข้าไปจะเป็นสายเส้นเล็กๆจำนวนมากซึ่งจะเป็นสายกราวด์ ชั้นที่สามจะเป็นพลาสติกหนาซึ่งทำหน้าที่เป็นฉนวนเช่นกัน และชั้นในสุดจะเป็นลักษณะของลวดทองแดง 1 เส้น เป็นแกนตรงกลางซึ่งส่วนนี้เองคือสายสัญญาณที่แท้จริง การต่อโดยใช้ขั้วต่อแบบ BNC และใช้สายต่อแบบ Coaxial นี้จะถูกเรียกทางเทคนิคว่าเป็นการต่อแบบ 10Base2 ซึ่งในปัจจุบันนี้ไม่ค่อยได้รับความนิยมแล้วเนื่องจากจะสามารถส่งผ่านข้อมูลระหว่างเครื่องคอมพิวเตอร์ได้ที่ความเร็วสูงสุดเพียง 10 Mbps (10 Megabits Per Second) เท่านั้น

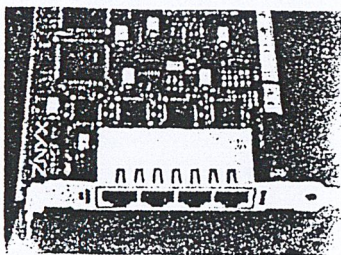
การ์ดแลนแบบ RJ-45 สำหรับการใช้งานการ์ดแลนแบบหัว RJ-45 นั้นจะใช้วิธีการต่อเครือข่ายที่เรียกว่าแบบ Star คือจะใช้อุปกรณ์ตัวกลางที่เรียกว่า HUB ทำหน้าที่เป็นจุดเชื่อมต่ออยู่ตรงศูนย์กลางของระบบ การต่อระบบเครือข่ายแบบนี้เครื่องคอมพิวเตอร์ในระบบทุกเครื่องจะต้องต่อสายแลนเข้ามาเชื่อมต่อกับ HUB ซึ่งข้อดีก็คือ หากสายแลนของเครื่องคอมพิวเตอร์



รูปที่ 2.7 การต่อวงจรเครือข่ายแบบ Star

เครื่องใดเครื่องหนึ่งเกิดขัดข้อง ก็ยังสามารถใช้ระบบเครือข่ายอยู่ได้ เอกสารนี้เป็นเอกสารสงวนลิขสิทธิ์สำหรับการเรียนเพื่อการศึกษาเท่านั้น มิอนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การต่อเครือข่ายแบบ Star นี้สามารถที่จะใช้ร่วมกับการต่อแบบบัสได้อีกด้วยโดยทำการเชื่อมเครือข่ายหลายๆวงเข้าด้วยกัน โดยใช้การพ่วงสายจาก HUB จากเครือข่ายหนึ่งไปยัง HUB ของเครือข่ายหนึ่ง แต่ HUB ที่ใช้นี้จะต้องมีช่องต่อพ่วงสายที่วางอยู่นี้อยู่ด้วย



รูปที่ 2.8 ขั้วต่อแบบ RJ-45

ขั้วต่อแบบ RJ-45 นั้นจะมีลักษณะเหมือนกับขั้วต่อของสายโทรศัพท์ แตกต่างกันที่มีขนาดใหญ่กว่าและจำนวนสายสัญญาณภายในมากกว่าคือ สายโทรศัพท์จะมีจำนวนสาย 4 เส้นและเรียกว่าหัว RJ-11 แต่หัว RJ-45 นี้จะมีสาย 8 เส้นสำหรับสาย LAN ที่จะนำมาต่อกับขั้วต่อแบบ RJ-45 นี้จะเรียกว่าสาย UTP ซึ่งภายในจะประกอบไปด้วยสายสัญญาณเส้นเล็กๆจำนวน 8 เส้นขดกันเป็นเกลียว ซึ่งการขดเป็นเกลียวนี้ก็จะมึลักษณะแตกต่างกันไปหลายลักษณะและการขดเกลียวในแต่ละลักษณะก็จะมีผลต่อความเร็วในการส่งสัญญาณด้วยเช่นกัน ซึ่งเราจะเรียกสายที่มีการขดเกลียวในลักษณะต่างๆกันว่า CAT เช่น CAT3, CAT4, CAT5 เป็นต้น

สายสัญญาณภายในจะมีสีต่างๆกัน คือ สีส้ม , ขาว-ส้ม , น้ำเงิน , ขาว-น้ำเงิน , เขียว , ขาว-เขียว, น้ำตาล และ ขาว-น้ำตาล ซึ่งการแยกสีของสายแต่ละเส้นให้แตกต่างกันนี้จะช่วยให้สามารถต่อสายได้สะดวกและง่ายมากยิ่งขึ้น ถึงแม้ว่าจะมีจำนวนสายมากถึง 8 เส้น แต่ในความเป็นจริงแล้ว จำนวนสายที่จะถูกใช้งานจะมีเพียง 4 เส้นเท่านั้นคือ เส้นที่ 1,2,3 และ 6 การต่อโดยใช้ขั้วต่อแบบ RJ-45 และสายแบบ UTP นี้จะเรียกตามศัพท์เทคนิคว่าเป็นการต่อแบบ 10Base-T ซึ่งเป็นที่นิยมใช้กันมากที่สุดในปัจจุบัน เนื่องจากสะดวกและสามารถรองรับการส่งผ่านข้อมูลในวงเน็ตเวิร์คได้ด้วยความเร็วตั้งแต่ 10Mbps จนถึง 100Mbps เลยทีเดียว แต่ทั้งนี้ความเร็วดังกล่าวจะขึ้นอยู่กับชนิดของการ์ดแลนและ HUB ที่ใช้ว่าสามารถรองรับความเร็วได้ในระดับความเร็วเท่าใดซึ่งหากว่าการ์ดแลนหรือ HUB ตัวใดตัวหนึ่งไม่สามารถสนับสนุนการส่งผ่านข้อมูลที่ความเร็ว 100 Mbps ได้ ความเร็วที่ได้รับก็จะลดลงด้วยเช่นกัน

เราอาจจะเห็นว่ามีการ์ดแลนบางรุ่นอาจจะมีขั้วต่อทั้งแบบ BNC และแบบ RJ-45 รวมอยู่ในตัวเดียวกันซึ่งสามารถใช้ในการเชื่อมต่อได้ทั้งสองแบบดังกล่าวข้างต้น ทั้งนี้ก็ขึ้นอยู่กับการออกแบบของผู้ผลิตแต่ละรายนั่นเอง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2. HUB

HUB เป็นอุปกรณ์สำหรับเชื่อมต่อสัญญาณจากเครื่องคอมพิวเตอร์หลายๆเครื่อง สำหรับการต่อเน็ตเวิร์คแบบ Star โดยเมื่อเราได้ติดตั้งการ์ดแลนในเครื่องคอมพิวเตอร์แล้วก็ให้นำปลายด้านหนึ่งของสาย LAN เสียบเข้ากับช่องต่อด้านหลังการ์ดแลน และปลายสายอีกด้านหนึ่ง เสียบเข้ากับช่องต่อสายของ HUB และทำการตั้งค่าการใช้งานในระบบปฏิบัติการเสียก่อน

3. สาย LAN

คือสายเคเบิล หรือสายสัญญาณที่ใช้เชื่อมต่อระหว่างเครื่องคอมพิวเตอร์ทุกเครื่อง ภายในเครือข่ายเดียวกัน โดยสำหรับการต่อเน็ตเวิร์คในวงเล็กๆตามที่ได้อธิบายมานี้ก็จะใช้สาย LAN แบบ Coaxial หรือแบบ UTP ตามแต่ว่าเราจะเชื่อมต่อวงเน็ตเวิร์คของเราเป็นแบบ Bus หรือ Star ตามที่ได้อธิบายไปข้างต้น

นอกจากนี้ก็ยังมียุโรปกรณ์ปลั๊กย่อยเช่น คีมเข้าหัว RJ-45, หัว RJ-45, ตัวปิดสัญญาณ หรือ Terminator ของสาย BNC

2.4.2 วิธีการต่อสาย LAN

จะขอกล่าวถึงวิธีการต่อสายสำหรับการการ์ดแลนแบบ RJ-45 สำหรับการต่อสายแลน ที่มีหัวต่อแบบ RJ-45 นั้นนอกจากจะต้องมีสาย UTP และหัว RJ-45 แล้วยังจะต้องมีอุปกรณ์ที่จำเป็น อีกตัวหนึ่งซึ่งก็คือ คีมสำหรับเข้าหัว RJ-45

การต่อสาย RJ-45 มีอยู่ 2 วิธีด้วยกันคือการต่อแบบธรรมดาและการต่อแบบไขว้สาย

1. การต่อแบบธรรมดา

ใช้สำหรับการต่อสาย UTP เพื่อเชื่อมต่อระหว่างคอมพิวเตอร์กับ HUB ซึ่งการเรียงสายสัญญาณทั้งสองด้านจะเหมือนกันตามตารางนี้

ตารางที่ 2.2 การเรียงสายสัญญาณแบบธรรมดา

| ลำดับ | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 |
|-------|---------|-----|-----------|---------|-------------|-------|------------|--------|
| สี | ขาว-ส้ม | ส้ม | ขาว-เขียว | น้ำเงิน | ขาว-น้ำเงิน | เขียว | ขาว-น้ำตาล | น้ำตาล |

2. การต่อแบบไขว้สาย

การต่อสาย UTP อีกแบบหนึ่งจะเรียกว่าการต่อแบบไขว้ ซึ่งการต่อแบบไขว้นี้จะมีข้อดีคือ เราสามารถที่จะนำสายที่ต่อไว้เสร็จแล้วเชื่อมต่อกับคอมพิวเตอร์สองเครื่องได้เลย โดยไม่ต้องใช้ HUB การต่อแบบไขว้นี้ จะมีการเรียงสายสัญญาณทั้งสองด้านแตกต่างกันไป โดยมีวิธีการต่อดังตารางคือ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 2.3 การเรียงสายสัญญาณสำหรับการต่อแบบไขว้สาย

| | | | | | | | | |
|-------|---------|-----|-----------|---------|-------------|-------|------------|--------|
| ลำดับ | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 |
| สี | ขาว-ส้ม | ส้ม | ขาว-เขียว | น้ำเงิน | ขาว-น้ำเงิน | เขียว | ขาว-น้ำตาล | น้ำตาล |

ส่วนปลายสายอีกด้านหนึ่งจะต้องต่อโดยเรียงลำดับดังนี้

ตารางที่ 2.4 การเรียงสายสัญญาณสำหรับการต่อแบบไขว้สาย(ของอีกด้าน)

| | | | | | | | | |
|-------|-----------|-------|---------|---------|-------------|-----|------------|--------|
| ลำดับ | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 |
| สี | ขาว-เขียว | เขียว | ขาว-ส้ม | น้ำเงิน | ขาว-น้ำเงิน | ส้ม | ขาว-น้ำตาล | น้ำตาล |

จะเห็นว่าการต่อสายด้านหนึ่ง จะเหมือนกันกับการต่อสายแบบธรรมดา แต่ ปลายสายอีกด้านหนึ่งจะต้องต่อสายด้วยการเรียงลำดับสายต่างออกไป

ส่วนการต่อสายทั้งสองแบบนี้ ทำดังนี้

1. ให้ใช้คีมเข้าหัว RJ-45 ทำการปลอกฉนวนหุ้มสายภายนอกสุดที่เป็นพลาสติก สีขาวออกประมาณ 1/2 นิ้ว
2. คลี่สายด้านใน 8 เส้นออก (ไม่ต้องลอกฉนวนของสายออก) แล้วให้จัดเรียงสีของสายตามวิธีที่ต้องการต่อสายตามตาราง
3. นำสายเสียบเข้ากับหัว RJ-45 ซึ่งจะเป็นช่องเล็ก ๆ
4. ใช้คีมเข้าหัว RJ-45 ทำการย้ำสายสัญญาณกับหัว RJ-45 ให้แน่น

มาตรฐานเครือข่าย

ที่โครงการนี้ใช้เป็นมาตรฐาน 100Base-T เป็นมาตรฐานที่ทันสมัยและได้รับการนำไปใช้งานอย่างแพร่หลาย เนื่องจากความง่ายในการติดตั้งใช้งาน 100Base-T สามารถรองรับ การส่งข้อมูลด้วยอัตราเร็วสูงถึง 100 เมกะบิตต่อวินาทีได้บนคู่สายตีเกลียวแบบไม่มีชีลด์

2.5 การเขียนโปรแกรมอำนวยความสะดวกในระบบเครือข่าย

2.5.1 การนำเสนอข้อมูลในฐานะข้อมูลผ่าน Web Page โดย Macromedia Dreamweaver UltraDev

โปรแกรม Macromedia Dreamweaver UltraDev เป็นเครื่องมือสำหรับออกแบบและสร้าง Web Page รวมทั้งช่วยบริการจัดการ Web Site โปรแกรมการทำงานแบบ WYSSWYG

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

(What You See Is What You Get) หรือเห็นอย่างไรได้อย่างนั้น เราสามารถสร้าง Web Page โดยการพิมพ์ข้อความ วางรูปภาพ หรือจัดองค์ประกอบต่างๆ บนหน้าจอได้ง่าย อีกทั้งโปรแกรม Macromedia Dreamweaver Ultradev มีความสามารถในการเขียน Script ได้ถึง 3 ภาษา คือภาษา ASP ,JSP,และภาษา Cold Fusion โดยอัตโนมัติ รวมทั้งการเขียนโปรแกรมเพื่อติดต่อกับฐานข้อมูล

คำศัพท์ที่ใช้ใน Macromedia Dreamweaver Ultradev

Web Site

ตำแหน่งที่อยู่ของ Web Page ใช้สำหรับการประชาสัมพันธ์ข้อมูล ข่าวสาร และบริการรับส่ง e-mail

Web Page

เอกสาร HTML ซึ่งเป็นเอกสารหลายมิติมีจุดเชื่อมโยงไปยังเอกสาร HTML อื่น ๆ ได้อีกHTML เอกสารที่สามารถเชื่อมโยงข้อมูลไปยังเอกสารอื่น โดยอาศัยการใส่สัญลักษณ์พิเศษ เรียกว่า TAG

Home Page

เอกสาร HTMLธรรมดาที่สามารถเข้าถึงข้อมูล Web Site ได้

Browser

โปรแกรมสำหรับเปิดเอกสารข้อมูลต่างๆ

ASP

Script ที่ทำงานอยู่บนWeb Server มีส่วนขยายเป็น .asp ลักษณะของไฟล์ *.asp จะประกอบด้วยโค้ดที่แทรกไว้ระหว่างภาษา HTML

SCRIPT

โปรแกรมสั้นๆ สำหรับจัดการกับข้อมูลต่างๆ ที่มีอยู่แล้ว โดย SCRIPT จะแทรกอยู่ในไฟล์ HTML

VB Script

คือภาษาที่ใช้เขียน Script มีลักษณะคล้ายกับคำสั่งของ Visual Basic เอกสารนี้เป็นเอกสารที่ส่งงานไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่นิยามให้เห็นาเป็ใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Site

ไคลเอนต์ที่เก็บเว็บเพจต่างๆ ที่สร้างขึ้นเสมือน Web Server สำหรับที่ใช้ในการทำงาน

Data Bindings

เครื่องมือที่ใช้ในการติดต่อกับฐานข้อมูลมี เครื่องมือภายใน 5 ชนิด คือ

1. **Recordset (Query)** ใช้สำหรับกำหนด Recordset ในไฟล์ฐานข้อมูลว่าเราจะนำข้อมูลใดบางจากตารางข้อมูลขึ้นมาการใช้งาน ซึ่งเราอาจจะต้องมีการกำหนดเงื่อนไขบางประการเพื่อให้ข้อมูลนั้นๆ ตรงตามวัตถุประสงค์

2. **Command (Stored Procedure)** เป็นรูปแบบคำสั่งที่เก็บไว้ในฐานข้อมูล เพื่อให้การเปลี่ยนแปลงแก้ไขคำสั่งสามารถทำได้โดยตรงที่ Stored Procedure ซึ่งสะดวกมากกว่าการแก้ไขที่โค้ดของโปรแกรม นอกจากนั้นการใช้ Stored Procedure และ จะเก็บไว้ในฐานข้อมูล แต่การใช้ SQL ในโปรแกรมจะต้องทำการ Compile SQL นั้นก่อนทุกครั้ง

3. **Request Variable** ทำหน้าที่รับค่าที่ส่งมาจากเบราว์เซอร์ เพื่อนำมาประมวลผลที่ฝั่ง Server

4. **Session Variable** ทำหน้าที่ในการเก็บข้อมูลที่ต้องการของผู้ใช้แต่ละคน โดย จะสร้าง Session Objectทันทีที่มีการติดต่อกับ Web Server และจะทำการลบ Session นั้นเมื่อเวลาตามที่ตั้งไว้

5. **Application Variable** ทำหน้าที่ในการกำหนดคุณสมบัติต่างๆ ให้ Application เช่น การกำหนดสิทธิ์ให้ Client สามารถเข้าไปแก้ไข Application นั้น ได้หรือไม่ได้ เป็นต้น

Server Behavior

เครื่องมือที่ใช้ติดต่อกับฐานข้อมูลเพื่อ ลบ เพิ่ม หรือ แก้ไขข้อมูลในฐานข้อมูล

Form

หมายถึงส่วนประกอบบนเว็บเพจหน้าที่ได้รับข้อมูลจาก Client แล้วส่งกลับมายัง Web Server เพื่อการประมวลผล โดยแบบฟอร์มจะประกอบด้วย Object ที่ใช้รับข้อมูลประเภทต่างๆ เรียกว่า ฟิลด์ (Field) ประกอบด้วย

1. Text Field เป็นฟิลด์ที่ใช้รับข้อมูลประเภทตัวอักษรหรือ ข้อความ
2. Button เป็นฟิลด์ที่สั่งให้เบราว์เซอร์ส่งข้อมูลจากแบบฟอร์มไปยังโปรแกรม CG
3. Checkbox เป็นฟิลด์ที่อนุญาตให้ผู้ชมเลือกหัวข้อที่ต้องการหลายหัวข้อ โดยการคลิก

ให้เกิดเครื่องหมายถูกในช่องสี่เหลี่ยมของหัวข้อนั้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการแข่งขันเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Hyperlink

การเชื่อมโยงของเอกสารจากจุดหนึ่งไปยังอีกจุดหนึ่ง หรือจากเอกสารหนึ่งไปยังเอกสารหนึ่ง

Record Insertion Form

เครื่องมือที่ใช้ในการเพิ่มข้อมูลในฐานข้อมูลโดยผ่าน Web Page

Record Update Form

เครื่องมือที่ใช้ในการแก้ไขข้อมูลในฐานข้อมูลโดยผ่าน Web Page

Recordset Navigation Bar

เครื่องมือที่ช่วยในการแสดงข้อมูลที่ละ Recordset โดยแต่ละ Recordset จะมีค่าตามที่ผู้ใช้กำหนด

Recordset Navigation Status

เครื่องมือที่ช่วยในการแสดงข้อมูลว่าข้อมูลที่แสดงเป็นข้อมูลที่เท่าใดจากข้อมูลทั้งหมด

Delete Record

เครื่องมือที่ใช้ในการลบข้อมูลในฐานข้อมูลโดยผ่าน Web Page

Repeat Region

เป็นการกำหนดค่าให้ Records เพื่อให้เกิดการวนรอบแสดงข้อมูลครั้งละกี่รายการก็ได้ตามที่ผู้ใช้ระบุ

Log in user

เครื่องมือที่ใช้ในการกำหนดสิทธิของผู้เข้าใช้ เช่น การกรอก Username หรือ Password โดยเครื่องมือดังกล่าวจะติดต่อกับฐานข้อมูลที่มี Password และ Username อยู่

Table

เครื่องมือที่ใช้ในการสร้างตารางใน WebPages โดยคุณสมบัติต่างๆ ของตารางสามารถกำหนดได้ใน Property ของตาราง เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Link

จุดปลายทางของ Hyperlink จาก WebPages ที่เราต้องการให้เชื่อมโยงกับเอกสาร HTML อื่นๆ

Record

ข้อมูลที่อยู่ในฐานข้อมูลที่เราติดต่อ

การสร้าง WebPages เพื่อนำเสนอข้อมูลจากฐานข้อมูล โดย Macromedia Dreamweaver

Ultradev

ในการสร้าง WebPages เพื่อนำเสนอข้อมูลจากฐานข้อมูลโดย Macromedia Dreamweaver Ultradev นั้นทำในลักษณะการนำข้อมูลจากฐานข้อมูลมาแสดงในรูปแบบของ WebPages เพื่อแสดงข้อมูลที่อยู่ในฐานข้อมูล โดยผู้ใช้สามารถแก้ไขข้อมูลในฐานข้อมูลได้จากทาง WebPages โดยตรงทั้งการ เพิ่ม ลบ หรือแก้ไขข้อมูลในฐานข้อมูลนั้นๆ เช่น การสร้าง WebPages เพื่อแสดงฐานข้อมูลของการจัดเก็บผลิตภัณฑ์ โดยเราสามารถแก้ไข ข้อมูลในฐานข้อมูลของการจัดเก็บผลิตภัณฑ์นั้นได้ เป็นต้น

การสร้าง WebPages ในการแก้ไขฐานข้อมูลนั้นผู้สร้าง WebPages นั้นต้องคำนึงถึงความปลอดภัยของข้อมูลที่อยู่ในฐานข้อมูลนั้นด้วยโดยเฉพาะข้อมูลในส่วนที่ถือว่าเป็นความลับ ควรมีการกำหนดสิทธิของผู้ใช้ เช่น การกำหนด Password และ Username ของผู้ใช้

ข้อดีของโปรแกรม Macromedia Dreamweaver Ultradev ในการติดต่อกับฐานข้อมูลนั้นก็คือความสะดวกของเครื่องมือที่เป็นอุปกรณ์ที่มีอยู่ในตัวโปรแกรม ทั้งเครื่องมือที่ใช้ในการติดต่อกับฐานข้อมูล เครื่องมือที่ช่วยในการตรวจสอบข้อมูลทั้งหมด ,เครื่องมือที่ใช้ในการลบข้อมูล, เครื่องมือ ที่ใช้ในการเพิ่มข้อมูลและเครื่องมือที่ใช้ในการเพิ่มข้อมูล รวมทั้งเครื่องมือที่ช่วยในการกำหนดสิทธิของผู้ใช้ WebPages เพื่อป้องกันการแก้ไขจากบุคคลภายนอก

โปรแกรมฐานข้อมูลที่โปรแกรม Macromedia Dreamweaver Ultradev ใช้ในการติดต่อก็คือ โปรแกรม MS Access

คำสั่งที่ใช้ในการจัดการกับฐานข้อมูลของโปรแกรม Macromedia Dreamweaver

Ultradev

1. คำสั่ง Recordsets และคำสั่ง Repeat Region

เป็นคำสั่งที่ใช้ในการแสดงข้อมูลเพื่อให้เกิดการวนรอบของข้อมูลทั้งหมดที่มีอยู่ เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ทางการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ในฐานะข้อมูลโดยผ่านทาง WebPages

2.คำสั่ง Record Insertion Form

เป็นคำสั่งที่ใช้ในการเพิ่มข้อมูลลงไปในฐานข้อมูลโดยผ่านทาง WebPages

3.คำสั่ง Record Update Form

เป็นคำสั่งที่ใช้ในการแก้ไขข้อมูลลงไปในฐานข้อมูลโดยผ่านทาง WebPages

4.คำสั่ง Log In User

เป็นคำสั่งที่ใช้ในการกำหนดสิทธิ์ของผู้ใช้ที่จะทำการแก้ไขข้อมูลทาง WebPages

วิธีการติดต่อกับฐานข้อมูลของโปรแกรม Macromedia Dreamweaver Ultradev

การติดต่อกับฐานข้อมูลของโปรแกรม Macromedia Dreamweaver Ultradev นั้น เราใช้วิธีการ DSN หรือ Data Source Name

Data Source Name (DSN) คือการกำหนดให้ตารางข้อมูลนั้นๆ ที่ถูกสร้างขึ้นไว้แล้วสามารถใช้ติดต่อกับโปรแกรม Macromedia Dreamweaver Ultradev เพื่อสามารถใช้ในการเพิ่ม ลบ หรือ แก้ไขข้อมูลในฐานข้อมูลนั้นได้

บทที่ 3

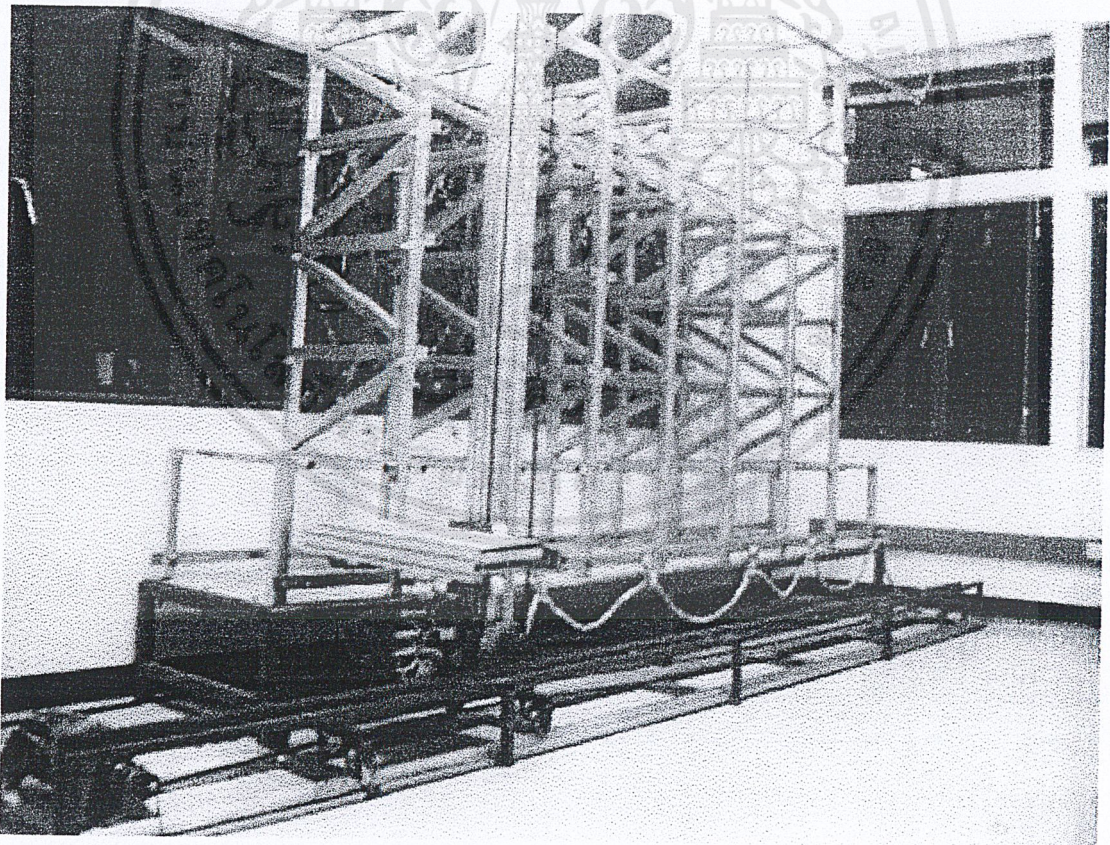
รายละเอียดในการสร้างและประกอบโครงงาน

3.1 Hardware

3.1.1 Warehouse

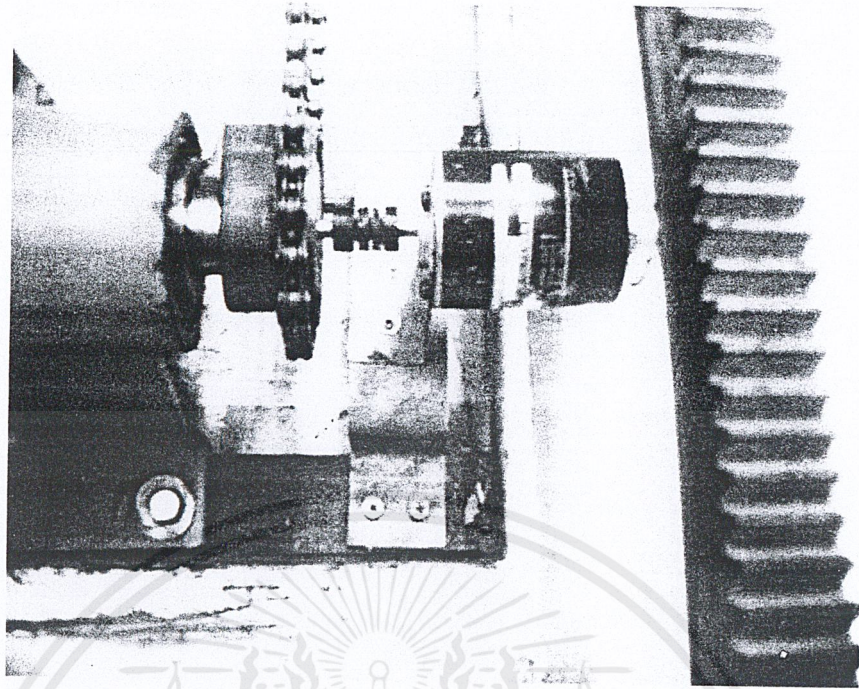
การทำงานของชุดจัดเก็บจะใช้เครื่องควบคุมโปรแกรมได้(PLC) ของบริษัท OMRON รุ่น C28H ตั้งงาน โดยจะมี Inverter ของ UNIDRIVE อยู่ 1 ตัวที่ใช้ควบคุมการหมุนของมอเตอร์ 3 เฟส 380 โวลต์ ของทั้งแกน X และ แกน Y โดยจะใช้วงจรรีเลย์ในการสั่งให้สลับการทำงานของแกน X และแกน Y ส่วนมอเตอร์ของถาดเลื่อนแกน Z นั้น ใช้มอเตอร์ 1 เฟส 220 โวลต์ แบบหมุนกลับทางอัตโนมัติเมื่อแรงบิดสูงขึ้น

ตัว warehouse นั้น ทำจากโครงเหล็ก มี 4 แถว 6 หลัก จะทำการแบ่งเก็บผลิตภัณฑ์ 3 ชนิด โดยจะแบ่งเป็นชนิดละ 2 หลัก

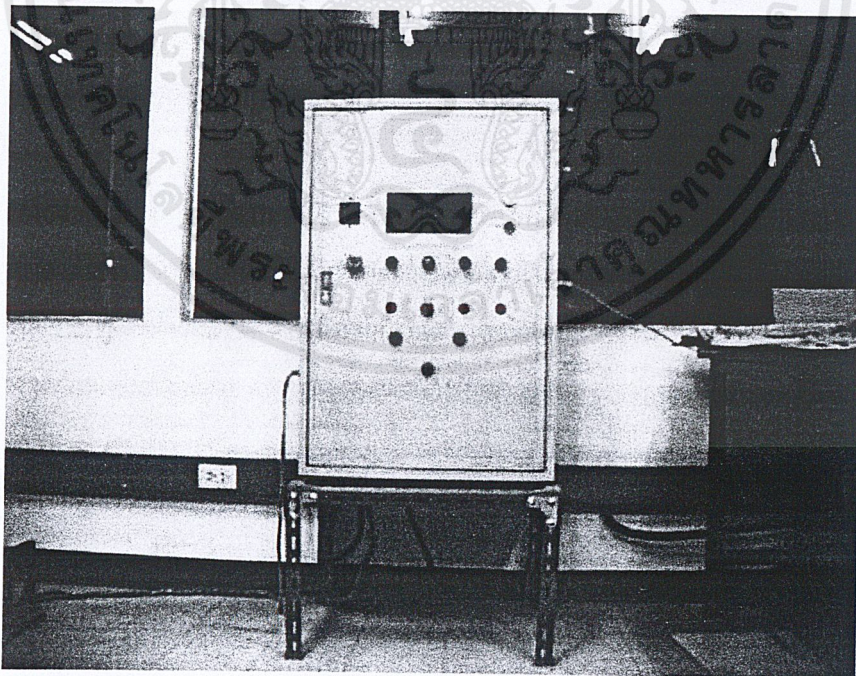


รูปที่ 3.1 warehouse ที่ใช้ในโครงงาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่3.2 การควบคุมระยะด้วย Encoder



รูปที่3.3 ตู้ควบคุมและTouch Screen

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.1.2 การใช้งาน Driver Unidrive

Unidrive เป็น Inverter ที่ใช้ในการควบคุมขับเคลื่อนมอเตอร์ กับชุดขับเคลื่อนในแนวแกน (X) โดยจะรับคำสั่งจากการทำงานมาจาก Output ของ PLC และทำงานตามที่ PLC ประมวลผลมาให้ Unidrive เป็น Inverter ที่สามารถทำการควบคุมมอเตอร์ได้หลายรูปแบบ เช่น Open loop, Close loop, Servo control แต่ที่เลือกใช้ในการควบคุมมอเตอร์ของชุดขับเคลื่อนในแนวแกนนี้จะใช้การควบคุมแบบ Open loop

Model ของ Unidrive

Unidrive ที่ใช้นี้เป็นรุ่น UNI1404 มีข้อมูลทางเทคนิคดังนี้

| | | |
|--------------------------|------------------|-------------|
| 1. Rate unit power | (400 VAC Supply) | 4.0 KVA |
| 2. Max rated motor power | (400 VAC Supply) | 2.2 KW(3HP) |
| 3. Rated Unit current | | 5.6 A |
| 4. Max unit current | | 7.1 A |

Terminal Connection

ในตัว UNI1404 จะมี Terminal อยู่ 2 จุด ตามหน้าที่การทำงาน คือ

1. Power connector

เป็นจุดต่อที่ใช้ต่อไฟฟ้าทางด้านกำลัง (Power) ให้กับ UNI1404 ไฟฟ้าที่ใช้กับขั้ว Power นี้จะมีแรงดันสูงถึง 380 VAC จะต้องทำการตัดไฟออกจากตัว UNI1404 ก่อนที่จะทำการสับเปลี่ยนตำแหน่งของสายที่ขั้วกำลังนี้ จุดต่อของ Power connector จะมีอยู่ 10 จุดด้วยกัน

2. Signal Connector

เป็นจุดที่ใช้ในการควบคุม UNI1404 ซึ่ง Output จาก PLC ที่จะนำมาควบคุมให้ Motor ทำงานก็จะถูกต่อเข้ากับจุดต่อเหล่านี้ สัญญาณที่ใช้กับจุดต่อเหล่านี้ เป็นสัญญาณต่ำ สามารถสับเปลี่ยนถอดสายได้โดยไม่ต้องหยุดจ่ายไฟให้กับ UNI1404 จุดต่อเหล่านี้มีด้วยกัน 2 แถว โดยแยกเป็นแถวบนกับแถวล่าง โดยมีรายละเอียดของจุดต่อแต่ละตัวดังนี้

Terminal 1 Status Relay แสดงค่า Drive Healthy

Terminal 2 Status Relay แสดงค่า Drive Healthy

(Connect load 240/1A (Inductive Load) เป็น N.O. ฉนวน 3KV Update

ทุก 8 ms)

Terminal 3 0 V Signal Ground ใช้ในการต่อ Enable Terminal ต่างๆ

Terminal 4 +10 Volt REF เป็นค่าอ้างอิงในการปรับความถี่เพื่อป้อนให้กับ Analog Input

1 และ 2 โดยต้องเลือกที่ 0.24 และ 0.25 ให้เป็น 0 คือแรงดันจะเป็นการ REF

ค่า 0-10 Volt ให้ถึง Analog เพื่อไปควบคุม Speed(Frequency)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- Terminal 5 Analog Reference Speed 1 (Noninverting)
- Terminal 6 Analog Reference Speed 1 (Inverting)
(เป็น Remote Speed Demand Range -10 V ถึง $+10\text{ V}$ Input Resistor 100K)
- Terminal 7 Analog Reference Speed 2 (Local Speed Demand)
- Terminal 8 NTC Therenistor ใช้ในการต่อวงจรป้องกัน โดยอาศัยอุณหภูมิ
- Terminal 9 Analog Speed Frequency Output
- Terminal 10 Analog Torque Output
(ค่า Output จะออกมาที่ 9 กับ 10 เป็น Bipolar Volt (-10 ถึง $+10\text{V MA}$)
Max Load = 1K มีการป้องกันการ Short Circuit Update ทุก 8 ms)
- Terminal 11 เป็น Signal 0 Volt เหมือน Terminal 3
- Terminal 21 ไม่ใช้
- Terminal 22 +24 Volt Supply Current 200 MA, Max 240 MA
- Terminal 23 0 Volt Signal Ground
- Terminal 24 Digital Output ของ Open Loop Operation “AT SPEED” สถานะ Output เป็น
Negative Logic Digital 0 ถึง 24 Volt 100 MA, Max 120 MA
- Terminal 25 Remote Reset Input
- Terminal 26 Run Jog
- Terminal 27 Run Forward
- Terminal 28 Run Reverse
- Terminal 29 เลือกรหัส Reference Analog(Local/Remote)
- Terminal 30 External Fault Trip(Open = Trip Code ET) เป็นตัวอนุญาตให้ Driveทำงาน

การควบคุมแบบ Open Loop

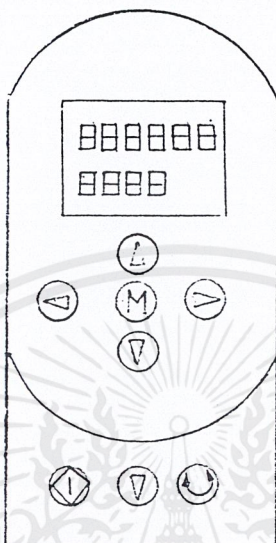
การควบคุมมอเตอร์ของ UNI1404 จะเลือกใช้แบบ Open Loop(V/F) Control โดยการควบคุมจะมีคุณสมบัติดังนี้

1. ค่าความถี่ Output จะคำนวณมาจากค่าความถี่อ้างอิงที่เลือกไว้
2. ค่าแรงดันที่ออกมาจะแปรผันโดยตรงกับค่าความถี่ที่ Unidrive ผลิตมา
3. ที่ความถี่ต่ำๆจะมีการชดเชยแรงดันตกเนื่องจากความต้องการของขดลวด
4. ค่าแรงดันสูงสุดของมอเตอร์จะถูกจำกัดด้วยแรงดันที่จ่ายให้กับ Unidrive
5. มีการชดเชยค่า Slip ให้สำหรับ Induction Motor
6. มีการควบคุมและจำกัดกระแสโดยอัตโนมัติเมื่อกระแสเกินค่าสูงสุด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การใช้งาน UNI1404

ดังที่กล่าวมาแล้วว่า Uni1404 สามารถที่จะใช้งานในการควบคุมมอเตอร์ได้หลายลักษณะ การควบคุมในลักษณะต่างๆ จะถูกกำหนดโดยการโปรแกรมค่าตัวแปรต่างๆ ให้กับ UNI1404 เพื่อให้ UNI1404 ทำการควบคุมมอเตอร์ให้เป็นไปตามรูปแบบที่ต้องการ



รูปที่ 3.4 รูปหน้า KEY PAD ของ UNI1404

Programming Key

เป็น Key Cursor 4 key ไว้ใช้ในการเปลี่ยนตำแหน่งและเปลี่ยนค่าตัวแปรต่างๆ โดยมี Mode Key(M) ในการเลือกเข้าสู่การโปรแกรม

Control Key

เป็นKeyที่ใช้ในการควบคุมการทำงานของUNI1404 ได้โดยวิธีการ Local -Control มีปุ่มให้ใช้ควบคุม 3 ปุ่มคือ

- RUN ใช้ในการควบคุมให้ UNI1404 สั่งให้มอเตอร์ทำงาน
- STOP-RESET ใช้ในการควบคุมให้ UNI1404 สั่งให้มอเตอร์หยุดการทำงาน
- FWD-REV ใช้ในการควบคุมให้ UNI1404 สั่งให้มอเตอร์หมุน Forward หรือ Reverse

การโปรแกรมค่าลงใน UNI1404

เมื่อทำการเปิดเครื่องขึ้นมาถ้ามีความผิดปกติที่จอแสดงผล Code Temp ซึ่งสามารถดูความหมายได้จากตารางและแก้ไขความผิดปกตินั้นที่จอแสดงผลจะแสดง Status ให้รู้ว่าพร้อมใช้งานแล้ว เมื่อต้องการ โปรแกรมค่าตัวแปรให้ทำดังนี้

บทที่ 4

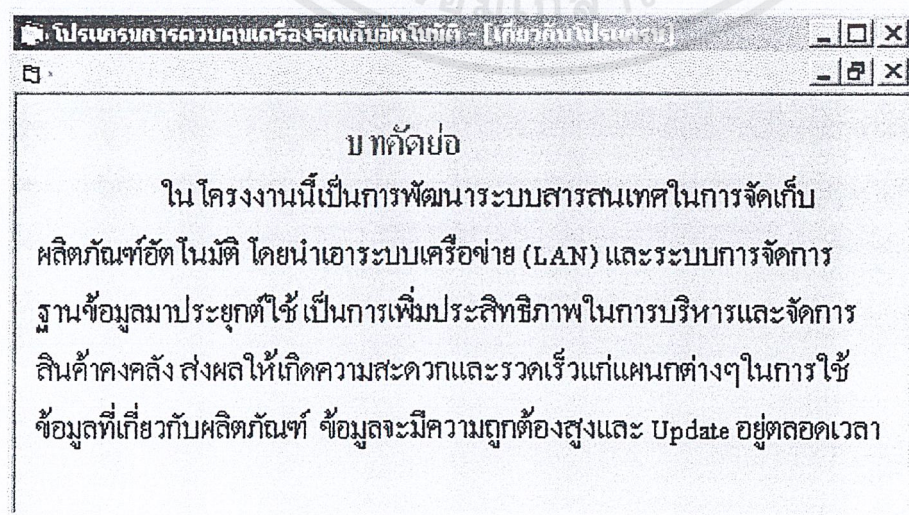
การทดลองโครงการ

4.1 การทดสอบโปรแกรม

ในการทำงานของเครื่องควบคุมแบบโปรแกรมได้ ได้แบ่งการทำงานออกเป็น 3 โหมดการทำงานด้วยกัน คือ

1. โหมดการทำงานอัตโนมัติ (Automatic Mode)
2. โหมดการทำงานด้วยมือ (Manual Mode)
3. โหมดควบคุมการทำงานด้วย PT

โดยการทำงานทั้ง 3 โหมดนี้ ได้ออกแบบให้สามารถทำการจัดเก็บ และ นำออกของผลิตภัณฑ์ซึ่งมีอยู่ 3 ชนิด การทำงานของโหมด อัตโนมัติจะเริ่มจากการรับค่าจากเครื่องอ่านรหัสแถบ หรือการป้อนค่า ในคอมพิวเตอร์ จากนั้นจะต้องทำการป้อนว่าจะนำเก็บ หรือ นำออก จากนั้นโปรแกรมจะส่งคำสั่งเพื่อสั่งให้ เครื่องควบคุมอัตโนมัติทำงาน ตามเงื่อนไขของโปรแกรม Ladder โดยที่โปรแกรมจะอ่านค่าสถานะต่างๆ ของเครื่องควบคุมแบบโปรแกรมได้ตลอดเวลา เพื่อป้องกันการทำงานที่ผิดพลาด เมื่อกระบวนการจัดเก็บหรือนำออกเสร็จแล้ว (เครื่องควบคุมอยู่ที่ตำแหน่งเริ่มต้นจะมีการ การUpdate ข้อมูลลงในฐานข้อมูล หากการทำงานไม่จบสิ้นจะไม่มี การเปลี่ยนแปลงของฐานข้อมูล เพราะถือว่า ผลิตภัณฑ์ไม่มีอยู่บน Warehouse ส่วนการทำงานในโหมด การทำงานด้วยมือ(Manual Mode) จะสามารถเคลื่อนที่ทำการจัดเก็บ หรือ การนำออกได้ แต่ จะไม่มีการ Update ฐานข้อมูล ส่วนการทำงานในโหมดการทำงานด้วย PT จะมีการทำงานเหมือนกับ การทำงานในโหมด อัตโนมัติ(Automatic Mode) และโหมด การทำงานด้วยมือแต่จะเป็นการป้อนค่าทาง PT แทนการป้อนค่าทางคอมพิวเตอร์



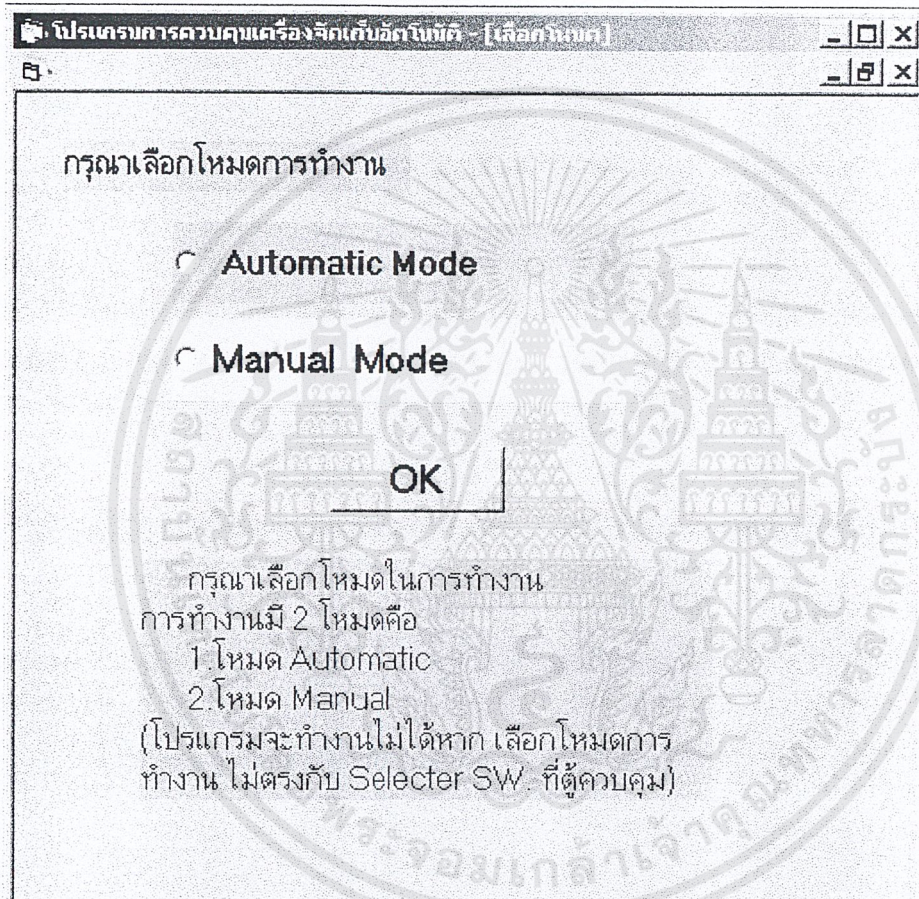
รูปที่ 4.1 แสดงหน้าจอเกี่ยวกับ โปรแกรม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โปรแกรมการควบคุมเริ่มต้นทำงาน ผู้ใช้จะพบกับ หน้าจอยูสเซอร์อินเตอร์เฟซที่เป็นข้อมูลเกี่ยวกับโปรแกรม

จากนั้นจะเป็นการให้เลือก โหมดการทำงาน คือ โหมดอัตโนมัติ(Automatic Mode) และ โหมดการทำงานด้วยมือ (Manual Mode) โดยจะต้องเลือกให้ตรงกับ Switch ที่ผู้ควบคุม หากไม่ตรง

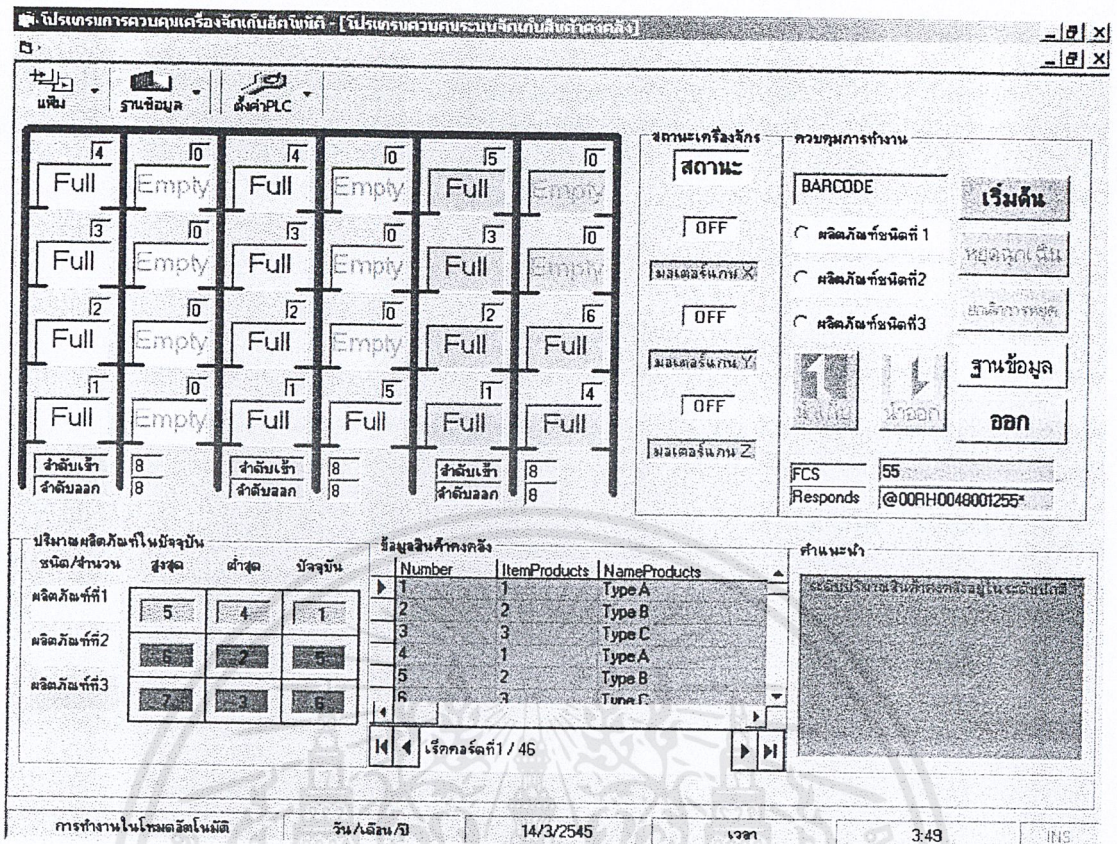
โปรแกรมจะเตือนและไม่ทำงานต่อไป



รูปที่4.2 แสดงการเลือกโหมดการทำงาน

กรณีที่เลือกโหมดอัตโนมัติตรงกับผู้ควบคุม จะพบกับหน้าจอที่ใช้ในการควบคุมแบบอัตโนมัติโดยที่หน้าจอจะพบ สถานะของผลิตภัณฑ์บน Warehouse ว่ามีอยู่หรือไม่ การทำงานจะเริ่มจากการป้อนชนิดผลิตภัณฑ์ที่ Option Button หรือการอ่านค่าจาก Barcode โดยจะรับค่าล่าสุดก่อนการกดปุ่ม นำเก็บ หรือนำออก เมื่อกดปุ่มนำเก็บ หรือนำออกแล้วจะไม่มีกรับค่าชนิดของผลิตภัณฑ์และไม่สามารถกดปุ่มใดๆ ยกเว้นปุ่มหยุดได้ จนกว่ากระบวนการจัดเก็บหรือนำออกจะเสร็จ จึงมีการ Update ฐานข้อมูล

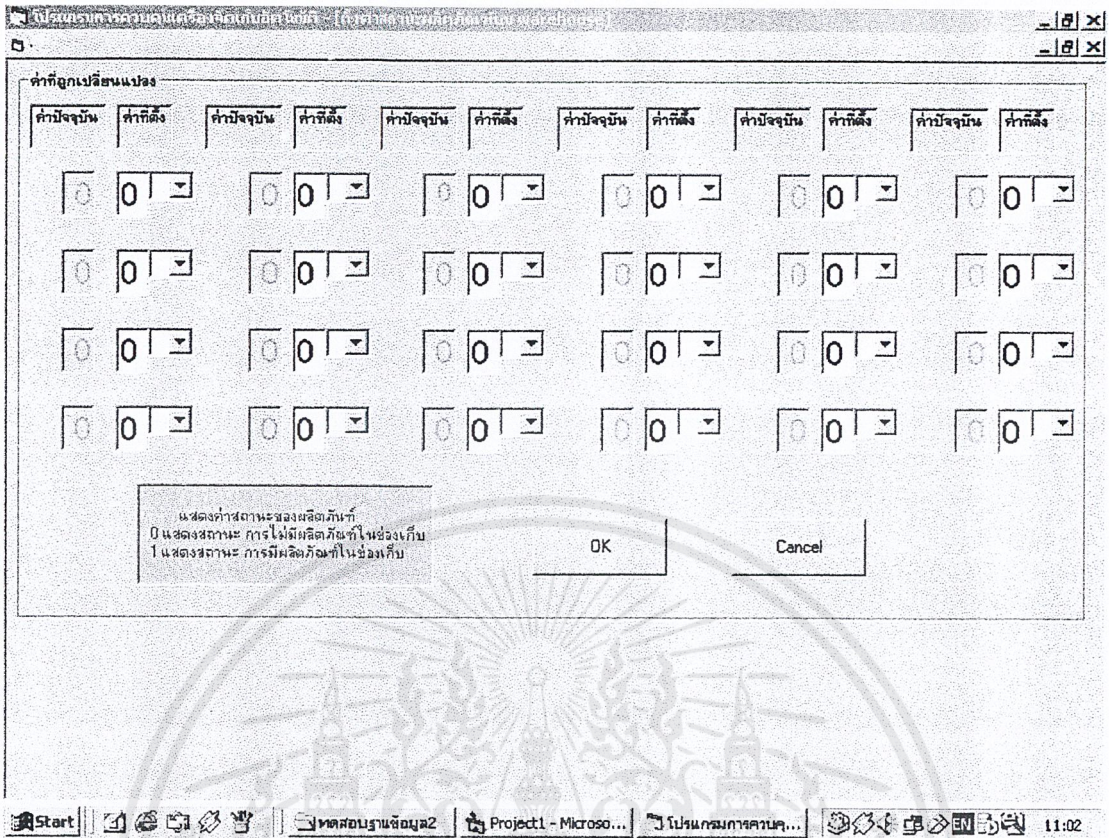
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.3 แสดงการทำงานโหมดอัตโนมัติ

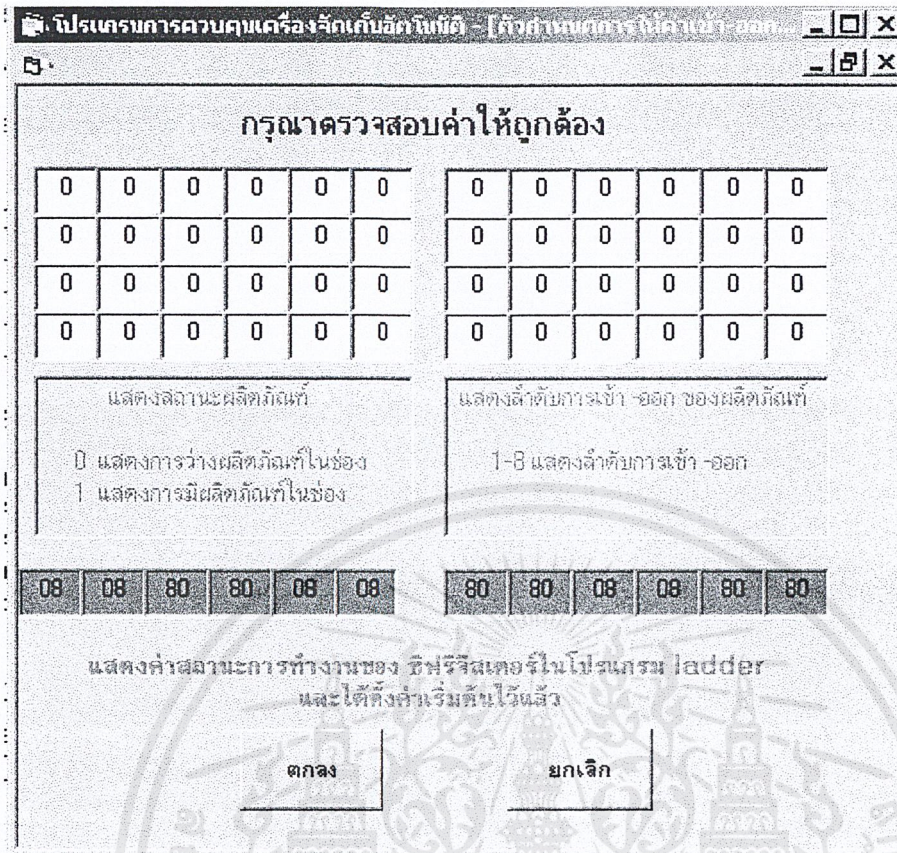
- การปรับเปลี่ยนค่าของ เครื่องควบคุมสามารถเปลี่ยนแปลงได้ ผ่านทางหน้าจอของ โหมดอัตโนมัติสามารถตั้งค่า สถานะผลิตภัณฑ์ว่า มีหรือไม่มีบน Warehouse การตั้งค่าลำดับการเข้าและออก ดังรูปที่ 4.4-4.6 ตามลำดับ
- การตรวจสอบสถิติการนำเข้า และนำออก ของช่วงเวลาที่ผ่านมาได้ โดยผ่านการตรวจสอบฐานข้อมูล ดังรูป 4.7
- การตรวจสอบปริมาณผลิตภัณฑ์ปัจจุบัน โดยผ่านการตรวจสอบผลิตภัณฑ์ทั้งหมด 4.8
- การตรวจสอบคุณลักษณะเฉพาะของผลิตภัณฑ์สามารถตรวจสอบ ทาง การตรวจสอบผลิตภัณฑ์ ดังภาพที่ 4.9-4.10

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



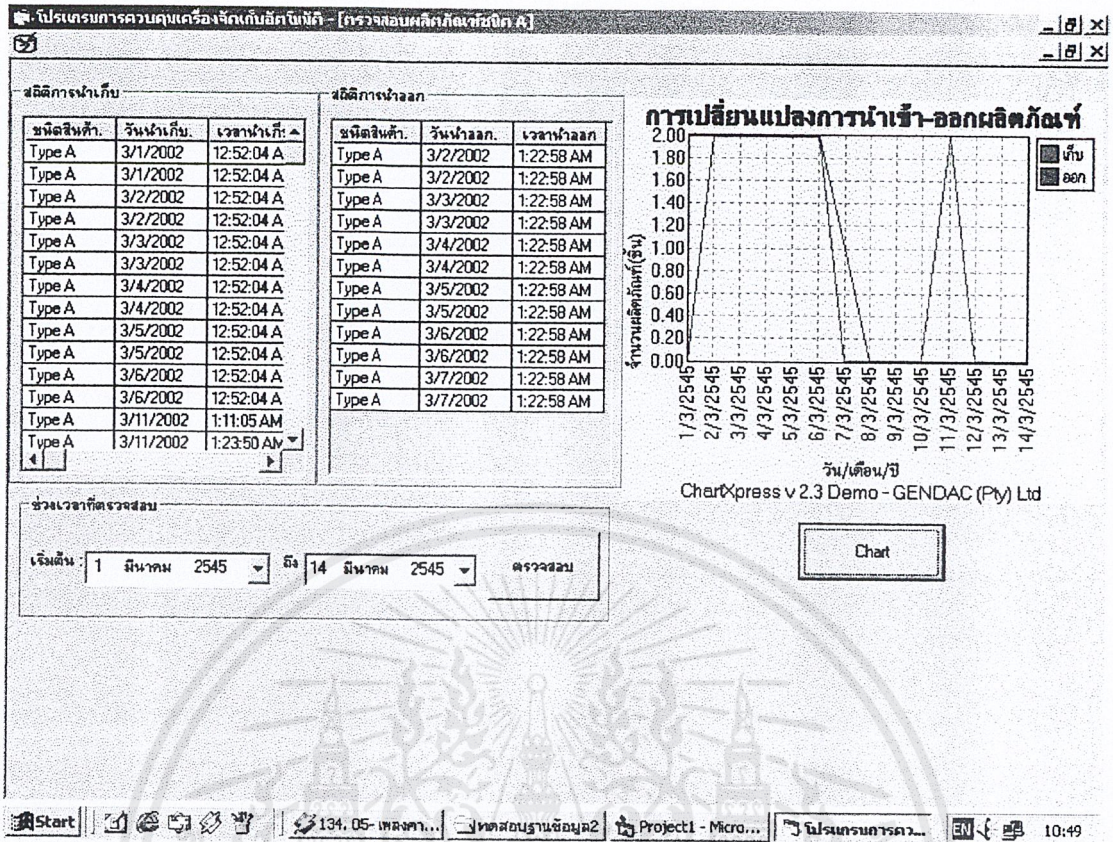
รูปที่ 4.4 แสดง การทำงานโหมด Manual

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
 ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



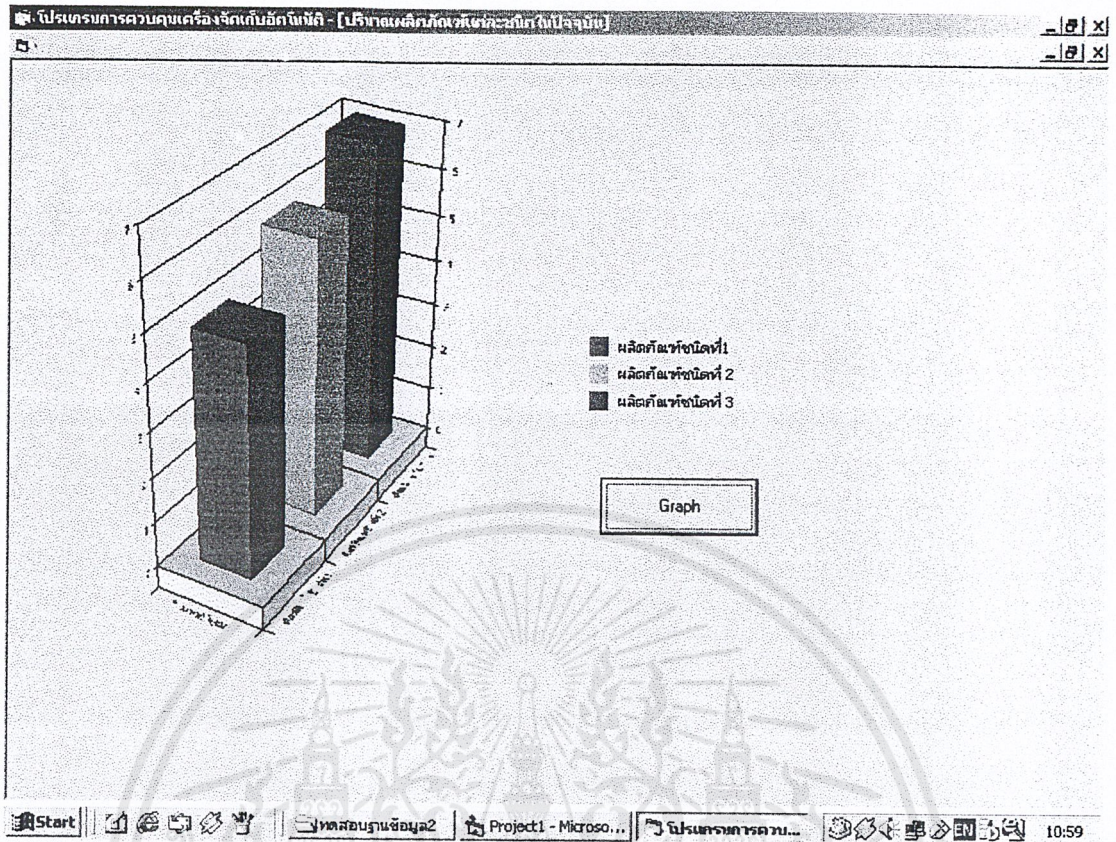
รูปที่ 4.6 แสดงผลลัพธ์ที่เปลี่ยนในการเปลี่ยนค่าเครื่องควบคุม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.7 การตรวจสอบปริมาณการนำเข้า-นำออกของผลิตภัณฑ์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.8 แสดงปริมาณผลผลิตในแต่ละชนิดในปัจจุบัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โปรแกรมการควบคุมเครื่องจักรเก็บผลไม้ - [ระบบคอมพิวเตอร์กลางมหาวิทยาลัย]

ข้อมูลหลัก รายการผลิตภัณฑ์ ตัวแทนจำหน่าย

รายการสินค้า ตรวจสอบผลิตภัณฑ์ จบการทำงาน

เลือกรายการผลิตภัณฑ์

เลือกรายการที่ต้องการ ดูรายการผลิตภัณฑ์

ข้อมูลทั่วไป

รุ่นผลิตภัณฑ์ Type A

ปริมาณคงเหลือ 5

ปริมาณที่กำลังสั่งซื้อ

จำนวนผลิตภัณฑ์ที่น้อยที่สุด ดูรายละเอียด

รูปที่ 4.9 แสดงฐานข้อมูลเกี่ยวกับผลิตภัณฑ์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 4.10 แสดงรายละเอียดเกี่ยวกับผลิตภัณฑ์

4.2 การทดสอบฐานข้อมูล

การตรวจสอบและบันทึกค่าต่างๆฐานข้อมูล ในโครงการมีตารางดังนี้

ตารางที่ 1 (Item Master) ประกอบด้วยคอลัมน์ 8 คอลัมน์ คือ Number(ตำแหน่งการเข้าของผลิตภัณฑ์ชนิดนี้), Item Product(ตัวที่ใช้ชี้ชนิดของผลิตภัณฑ์), Name Products(ชื่อชนิดผลิตภัณฑ์), Date IN(วัน/เดือน/ปี นำเก็บ), Time IN(เวลานำเก็บ), Date OUT(วัน/เดือน/ปี ที่นำออก), Time-Out(เวลาที่นำออก), Status(สถานะของผลิตภัณฑ์บน Warehouse) โดยมีจุดประสงค์เพื่อจัดเก็บข้อมูลการเข้าออกเพื่อเป็นประวัติของคลังจัดเก็บสินค้า โดยจะไม่มี การเคลียร์เมื่อสินค้าออกไป แต่จะเป็นการเช็คที่ Status แทน

ตารางที่ 2 (TempType1OnWarehouse) ประกอบด้วย 4 คอลัมน์ คือ NameProduct(ชื่อชนิดผลิตภัณฑ์), Date IN(วัน/เดือน/ปี ที่นำจัดเก็บ), Time IN(เวลาการจัดเก็บ), Status (สถานะผลิตภัณฑ์บน Warehouse) โดยมีจุดประสงค์เพื่อบอกปริมาณและสถานะของผลิตภัณฑ์ชนิดที่ 1 บน Warehouse โดยจะมีการเคลียร์ค่าเมื่อนำผลิตภัณฑ์ออก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

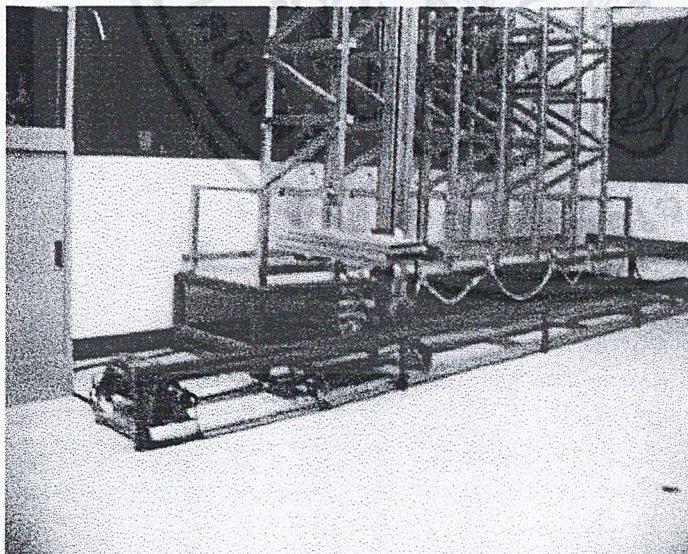
ตารางที่3 (Tem Type2On Warehouse) ประกอบด้วย 4 คอลัมน์ คือ NameProduct(ชนิดผลิตภัณฑ์), Date IN(วัน/เดือน/ปี ที่นำจัดเก็บ) ,Time IN(เวลาการจัดเก็บ),Status (สถานะผลิตภัณฑ์บนWarehouse) โดยมีจุดประสงค์เพื่อบอกปริมาณและสถานะของผลิตภัณฑ์ชนิดที่ 2 บน Warehouseโดยจะมีการเคลียร์ค่าเมื่อนำผลิตภัณฑ์ออก

ตารางที่4 (TempType3On Warehouse) ประกอบด้วย 4 คอลัมน์ คือ NameProduct(ชื่อชนิดผลิตภัณฑ์), Date IN(วัน/เดือน/ปี ที่นำจัดเก็บ) ,Time IN(เวลาการจัดเก็บ), Status (สถานะผลิตภัณฑ์บนWarehouse) โดยมีจุดประสงค์เพื่อบอกปริมาณ และสถานะของผลิตภัณฑ์ชนิดที่ 3 บน Warehouseโดยจะมีการเคลียร์ค่าเมื่อนำผลิตภัณฑ์ออก

ตารางที่5 (Detail Product) ประกอบด้วย 10 คอลัมน์ คือ Item Product(ตัวที่ใช้ชื่อชนิดของผลิตภัณฑ์), NameProducts (ชื่อชนิดผลิตภัณฑ์) ,Width (ความกว้าง) , Long(ความยาว), Height (ความสูง) , MaxProduct (ปริมาณมากที่สุดที่ต้องการให้มีบน Warehouse) , MinProduct (ปริมาณน้อยที่สุดที่ต้องการให้มีบนWarehouse) , PictureFilename (พาสเก็บรูปภาพผลิตภัณฑ์) ,Specification(ลักษณะของผลิตภัณฑ์โดยมีจุดประสงค์เพื่อเก็บข้อมูลต่างๆเกี่ยวกับผลิตภัณฑ์)

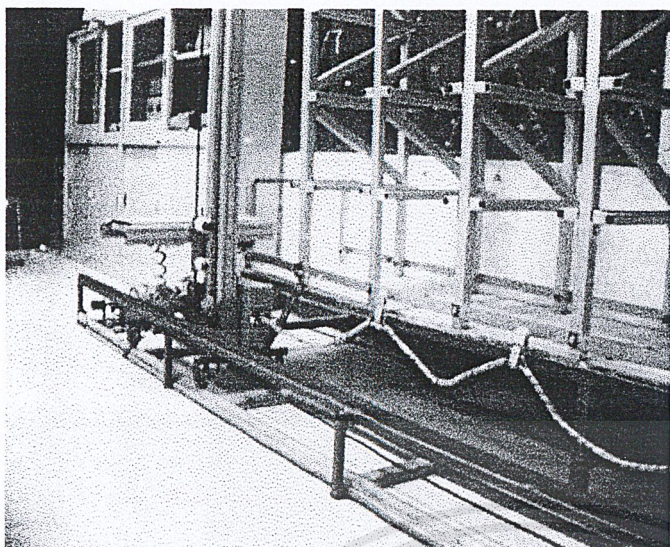
ตารางที่ 6 (Barcode) ประกอบด้วย 2 คอลัมน์คือBarcodeNumber (รหัสบาร์โค้ด) Information(ชนิดของผลิตภัณฑ์ เก็บประวัติผลิตภัณฑ์)

ตัวอย่างการทำงานของเครื่องจัดเก็บอัตโนมัติ

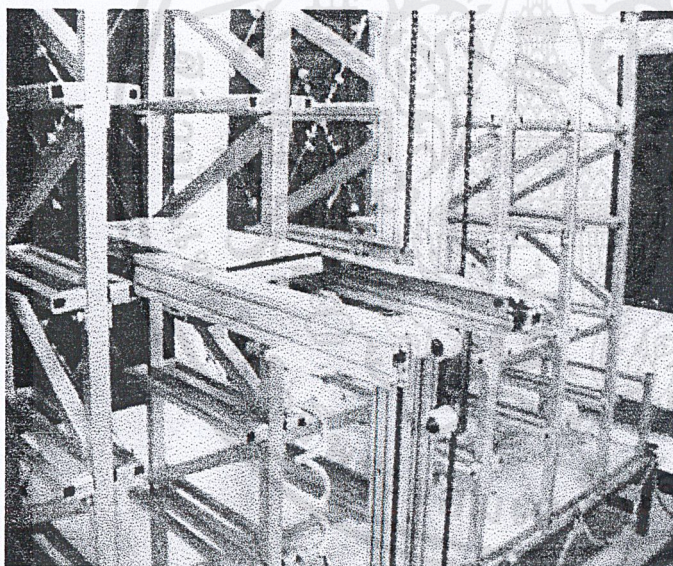


รูปที่4.11 เครื่องจัดเก็บผลิตภัณฑ์อัตโนมัติขณะอยู่ที่ตำแหน่งเริ่มต้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

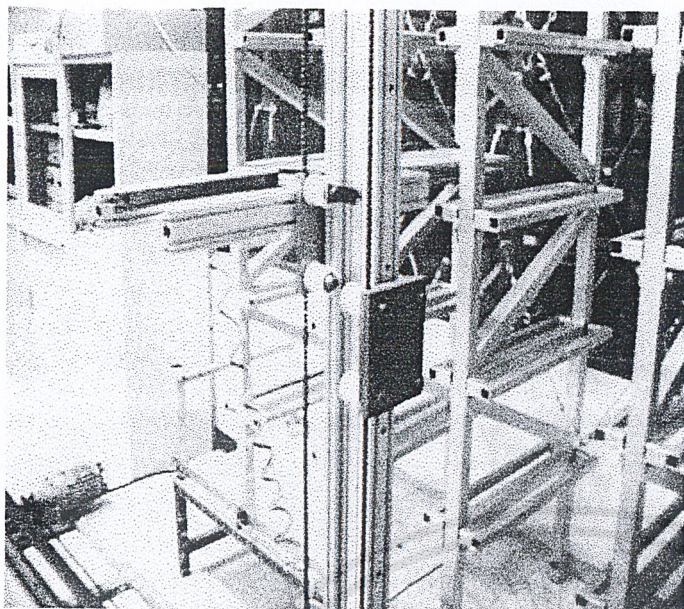


รูปที่4.12 เครื่องจัดเก็บผลิตภัณฑ์อัตโนมัติขณะอยู่ที่ตำแหน่งเริ่มต้น



รูปที่4.13 เครื่องจัดเก็บผลิตภัณฑ์อัตโนมัติขณะนำผลิตภัณฑ์เก็บหรือนำออก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่4.14 เครื่องจัดเก็บผลิตภัณฑ์อัตโนมัติขณะนำผลิตภัณฑ์เก็บหรือนำออก

ตัวอย่างในการเปลี่ยนแปลงข้อมูลในการทดลองโครงการ

ตารางที่4.1 เป็นตารางที่บันทึกค่าการนำเข้า – ออกของผลิตภัณฑ์ทุกชนิด

Microsoft Access

เมนู: แก้ไข, มุมมอง, แทรก, รูปแบบ, ระบุเป็น, เครื่องมือ, ฐานข้อมูล, ใช้

ItemMaster: ตาราง

| Number | ItemProducts | NameProducts | DateIN | TimeIN | DateOUT | TimeOUT | Status |
|--------|--------------|--------------|----------|---------|----------|---------|-------------------------------------|
| 1 | 1 Type A | | 1/3/2545 | 0:52:04 | 2/3/2545 | 1:22:58 | <input checked="" type="checkbox"/> |
| 2 | 2 Type B | | 1/3/2545 | 0:57:08 | 2/3/2545 | 1:14:41 | <input checked="" type="checkbox"/> |
| 3 | 3 Type C | | 1/3/2545 | 1:00:18 | 2/3/2545 | 1:10:49 | <input checked="" type="checkbox"/> |
| * | 0 | 0 | | | | | <input type="checkbox"/> |

มุมมองแผนข้อมูล

Start | DATA1 (E:) | datawarehouse : ... | ItemMaster : ตาราง | NUM | 11:17

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 4.2 แสดงปริมาณผลิตภัณฑ์ที่ 1 มีอยู่บน Warehouse ในปัจจุบัน

The screenshot shows a Microsoft Access window with a table named 'TempType2OnWarehouse : ตาราง'. The table has four columns: NameProduct, DateIN, TimeIN, and Status. There is one row of data for 'Type B' with DateIN '8/3/2545' and TimeIN '1:02:32'. The Status column has two empty checkboxes.

| NameProduct | DateIN | TimeIN | Status |
|-------------|----------|---------|--|
| Type B | 8/3/2545 | 1:02:32 | <input type="checkbox"/> <input type="checkbox"/> |

ตารางที่ 4.3 แสดงปริมาณผลิตภัณฑ์ที่ 2 มีอยู่บน Warehouse ในปัจจุบัน

The screenshot shows a Microsoft Access window with a table named 'TempType2OnWarehouse : ตาราง'. The table has four columns: NameProduct, DateIN, TimeIN, and Status. There is one row of data for 'Type B' with DateIN '8/3/2545' and TimeIN '1:02:32'. The Status column has two empty checkboxes.

| NameProduct | DateIN | TimeIN | Status |
|-------------|----------|---------|--|
| Type B | 8/3/2545 | 1:02:32 | <input type="checkbox"/> <input type="checkbox"/> |

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 4.4 แสดงปริมาณผลิตภัณฑ์ที่ 3 มีอยู่บน Warehouse ในปัจจุบัน

| NameProduct | DateIN | TimeIN | Status |
|-------------|-----------|---------|--------------------------|
| Type C | 11/3/2545 | 1:25:54 | <input type="checkbox"/> |
| | | | <input type="checkbox"/> |

ตารางที่ 4.5 แสดงรายละเอียดเกี่ยวกับผลิตภัณฑ์แต่ละชนิด

| ItemProduct | NameProducts | Width | Long | Height | Weight | MaxProduct | MinProduct | PictureFileName | Specific Che |
|-------------|--------------|-------|------|--------|--------|------------|------------|---------------------------|------------------|
| 1 | TypeA | 10 | 10 | 10 | 100 | 7 | | c:\PictureProducts\01.gif | สารอินทรีย์ |
| 2 | TypeB | 20 | 20 | 20 | 200 | 6 | | c:\PictureProducts\02.gif | สารกึ่งโลหะ |
| 3 | TypeC | 30 | 30 | 30 | 300 | 5 | | c:\PictureProducts\03.gif | สารเร่งปฏิกิริยา |
| 0 | | 0 | 0 | 0 | 0 | 0 | 0 | | |

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 4.6 แสดงข้อมูลเกี่ยวกับรหัสแถบและชนิดผลิตภัณฑ์

The screenshot shows a Microsoft Access window titled "Barcode : ตาราง". The window contains a table with the following data:

| Barcode Number | Information |
|----------------|-------------|
| 1 | Type A |
| 2 | Type B |
| 3 | Type C |
| 0 | |

The table is displayed in a grid view. The status bar at the bottom indicates "Records: 14 | 3" and "Page 3 of 3". The taskbar shows the Start button, a search icon, and several open applications including "DATA1 (E)", "datawarehouse : ...", and "Barcode : ตาราง". The system clock shows "11:34".

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 5

สรุปการทำงานของโครงการและแนวทางการพัฒนา

5.1 สรุปการทำงานของโครงการ

1. การทำงานของเครื่องจัดเก็บผลิตภัณฑ์อัตโนมัติ สามารถทำการจัดเก็บและนำออกได้อย่างแม่นยำและถูกต้อง
2. เมื่อการทำงานของเครื่องจัดเก็บผลิตภัณฑ์อัตโนมัติ เสร็จกระบวนการทำงานเรียบร้อยแล้ว ฐานข้อมูลมีการเปลี่ยนแปลงอย่างถูกต้อง
3. สามารถใช้ฐานข้อมูลร่วมกัน ผ่านระบบเครือข่าย LAN ได้ถูกต้อง

5.2 แนวทางการพัฒนา

1. เนื่องจากที่ช่องจัดเก็บผลิตภัณฑ์ไม่มีตัวตรวจจับสถานะอย่างแท้จริง อาจทำให้สถานะไม่ตรงตามความเป็นจริง จึงควรมีการติดตั้ง Sensor เพื่อทำการตรวจผลิตภัณฑ์
2. เนื่องจากการติดต่อสื่อสารระหว่างคอมพิวเตอร์กับ PLC ใช้พอร์ตอนุกรม RS-232 ทำให้ต้องใช้เวลาในการติดต่อสื่อสาร ควรพัฒนา
3. จะเป็นการดีหากสามารถนำเอาระบบ Internet มาใช้ควบคุมและตรวจสอบสถานะของเครื่องควบคุมอัตโนมัติและผลิตภัณฑ์

บรรณานุกรม

- [1] ฉันทวุฒิ พิษผล และ พิชิต สันติภูตานนท์ “คู่มือเรียน Visual Basic6” บริษัทโปรวิชั่น พ.ศ. 2542
- [2] สุพรรณ กุลพาณิชย์ “PROGRAMMABLE CONTROLLER เทคนิคและการใช้งานเบื้องต้น” บริษัท ออมรอน ตรีศักดิ์ จำกัด พ.ศ.2521
- [3] Omron “Sysmac C20H, C28H, C40H OPERATION MANUAL” Omron Tateisi 1990



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้