

การควบคุมการใช้อินเทอร์เน็ตด้วยบัตรแถบแม่เหล็ก  
CONTROL INTERNET BY MAGNETIC CARD



นายบัณฑิต รักวีรธรรม  
นายวรวิทย์ กุลพงษ์

เลขที่.....  
เลขทะเบียน 42532  
วัน, เดือน, ปี 2 4 พ.ศ. 2545

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

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต  
สาขาวิชาวิศวกรรมการวัดคุม  
ภาควิชาเทคโนโลยีการวัดคุมทางอุตสาหกรรม คณะวิศวกรรมศาสตร์  
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง  
ปีการศึกษา 2543

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำใช้ 64

# CONTROL INTERNET BY MAGNETIC CARD



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

2000


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาควิชาเทคโนโลยีการวัดคุมทางอุตสาหกรรม  
คณะวิศวกรรมศาสตร์  
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง  
ใบรับรองปริญญาโท

หัวข้อปริญญาโท การควบคุมการใช้อินเทอร์เน็ตด้วยบัตรแถบแม่เหล็ก  
CONTROL INTERNET BY MAGNETIC CARD  
นักศึกษาผู้จัดทำ นายบัณฑิต รักวีรธรรม รหัสประจำตัว 41013418  
นายวรวุฒิ กุลพงษ์ รหัสประจำตัว 41013425  
ปริญญา วิศวกรรมศาสตรบัณฑิต  
สาขาวิชา วิศวกรรมการวัดคุม  
ปีการศึกษา 2543

อาจารย์ผู้ควบคุมปริญญาโท	ลายมือชื่อ
ผศ. ประภาส อุดคกิมานันท์	

วัน/เดือน/ปี ที่สอบ วันพฤหัสบดีที่ 15 มีนาคม พ.ศ. 2544  
สถานที่สอบ ณ ห้องสอบปริญญาโท ภาควิชาเทคโนโลยีการวัดคุมทางอุตสาหกรรม

ภาควิชารับรองแล้ว  
  
(ผศ. ประสิทธิ์ จุลเสรีวงศ์)  
หัวหน้าภาควิชา ฯ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หัวข้อปริญาพนธ์                    การควบคุมการใช้อินเทอร์เน็ตด้วยบัตรแม่เหล็ก  
CONTROL INTERNET BY MAGNETIC CARD  
นักศึกษาผู้จัดทำ                    นายบัณฑิต รักวีระธรรม  
    นายวรวิทย์ กุลพงษ์  
อาจารย์ที่ปรึกษา                    ผู้ช่วยศาสตราจารย์ ประภาส อุคคกิมพันธ์  
ปีการศึกษา                            2543

### บทคัดย่อ

ในปัจจุบันอินเทอร์เน็ต(INTERNET) ได้เข้ามามีบทบาทในสังคมการสื่อสารของคนไทยมากขึ้น เทคโนโลยีสมัยใหม่ได้พัฒนารูปแบบของอินเทอร์เน็ตให้ใช้งานได้ง่ายขึ้น จึงมีผู้ให้บริการอินเทอร์เน็ตเกิดขึ้นมากมาย

ดังนั้นในการควบคุมการใช้บริการของผู้ให้บริการแต่ละรายจะมีปัญหาเกิดขึ้นคือการควบคุมเวลาในการใช้งานของลูกค้า จึงได้คิดวิธีที่จะทำการ์ดแถบแม่เหล็ก(MEGNETIC CARD) มาประยุกต์ เพื่อควบคุมเวลาในการให้บริการเพื่อลดความยุ่งยากและเพิ่มความสะดวกให้กับเจ้าของกิจการและลูกค้าที่มาใช้บริการอินเทอร์เน็ต

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

**Thesis Title** CONTROL INTERNET BY MAGNETIC CARD  
**Authers** Mr. BUNDIT RUCKVERATHEM  
Mr. WORAWUT KULAPONG  
**Thesis Advisor** Asst.Prof. PAPHAT AUKKAKIMAPAN  
**Year** 2000

### ABSTRACT

At present, internet has a role on communicative social of Thai people. New technology has developed the forms of internet be more easily user, so there have been many internet user now

The problem of each server in controlling user is time controlling of customer usage, so we applied the magnetic card to control the service time decrease the implication and increase the convenience of the business owners and the customers. This project bring the most efficiency to the business of internet service.

## กิตติกรรมประกาศ

วิทยานิพนธ์ฉบับนี้สำเร็จลุล่วงได้ดีเพราะได้รับคำแนะนำที่ดีเสมอมาจากท่านอาจารย์ที่ให้ปรึกษาโครงการคือ ผู้ช่วยศาสตราจารย์ ประภาส อุคคกิมพันธ์ ทำให้การทำงานสำเร็จได้ด้วยดี โดยท่านอาจารย์ได้ให้แนวความคิดในการดำเนินการเพื่อแก้ไขปัญหาได้ไปในแนวทางที่ถูกต้องทำให้เมื่อเกิดปัญหาในการทำงานก็จะสามารถแก้ไขให้ผ่านไปได้ด้วยดี

ขอขอบคุณองค์การสื่อสารมวลชนแห่งประเทศไทย(อ.ส.ม.ท) ที่เอื้อเฟื้อเครื่องมือและสถานที่ในการทำโครงการในครั้งนี้

และขอกราบขอบพระคุณ คุณแม่ และคุณพ่อ ที่เป็นที่เคารพรักอย่างยิ่ง ที่เป็นแรงกระตุ้นและเป็นแรงผลักดันในมีกำลังใจในการทำงานจนสำเร็จลุล่วงด้วยดี

ประโยชน์อันพึงได้รับจากวิทยานิพนธ์ฉบับนี้ ขอมอบแด่ผู้ที่มีพระคุณทุกท่าน

คณะผู้จัดทำ

# สารบัญ

	หน้า
บทคัดย่อภาษาไทย.....	I
บทคัดย่อภาษาอังกฤษ.....	II
กิตติกรรมประกาศ.....	III
สารบัญ.....	IV
สารบัญภาพ.....	VI
บทที่ 1 บทนำ.....	1
ภาพโดยรวมของโครงการ.....	2
แนวความคิดในการสร้าง.....	2
คุณสมบัติของโครงการ.....	3
บทที่ 2 Hardware.....	4
Magnetic card.....	4
การอ่านข้อมูลจากบัตรแถบแม่เหล็ก.....	5
บทที่ 3 Software.....	9
คุณสมบัติ เมคคอด และอีเวนต์.....	9
ฟอร์ม (Forms).....	11
ปุ่มคำสั่ง (Command Button).....	14
คอนโทรลสำหรับแสดงและรับข้อมูล.....	16
เลเบล (Label).....	16
เท็กบ็อกซ์ (Text Box).....	17
คอนโทรลแสดงตัวเลือก.....	18
เช็ทบ็อกซ์ (Check Box).....	18
โฟกัส (Focus).....	19
ความรูปเกี่ยวกับเมนู (Menu).....	20
Menu Editor.....	20
การใช้งานด้วย Menu Editor.....	21
การสร้างเมนูย่อย (Sub Menu).....	22
การกำหนดคีย์ลัดให้เมนู.....	22
การเขียนคำสั่งในรูปแบบต่างๆ.....	23
การแบ่งคำสั่งบรรทัดหนึ่งออกเป็นหลายบรรทัด.....	24

# สารบัญ(ต่อ)

	หน้า
การใส่คำอธิบายสำหรับโปรแกรม (Comment).....	24
MSComm .....	25
องค์ประกอบในการใช้ MSComm.....	26
การตั้งค่าติดต่อกับพอร์ต.....	26
การกำหนดคุณสมบัติของ MSComm Control ให้สามารถติดต่อกับ พอร์ตได้.....	27
วิธีการรับส่งข้อมูลจาก Serial Port .....	28
Winsock.....	29
ขั้นตอนการเขียนโปรแกรม.....	29
บทที่ 4 การ Set Network ขั้นพื้นฐาน .....	39
บทที่ 5 หลักการทำงานของโปรแกรมในแต่ละส่วน.....	44
อธิบายโปรแกรมย่อยต่างๆ.....	48
ส่วนต่างๆของโปรแกรมหลัก.....	49
Main Program.....	49
บทที่ 6 ผลการทดลอง.....	55
บทที่ 7 สรุปโครงการและข้อเสนอแนะในการพัฒนา.....	58
บรรณานุกรม.....	59
ภาคผนวก.....	60
ภาคผนวก ก.....	61
Source Code เครื่องรูดการ์ด.....	61
Source Code โปรแกรมหลัก ( VB 6.0) Form1 (master.frm).....	67
Source Code โปรแกรมหลัก ( VB 6.0) frprog (master2.frm).....	100
Source Code โปรแกรมหลัก ( VB 6.0) frprogm (master3.frm).....	101
Source Code โปรแกรมหลัก ( VB 6.0) MoDule1(master.bas).....	102
SourceCode โปรแกรมลูกข่าย ( VB 6.0) Form1(slave1.frm).....	102
SourceCode โปรแกรมลูกข่าย ( VB 6.0) Form2(slave2.frm).....	105
วิธีการติดตั้ง.....	108
ภาคผนวก ข.....	109

# สารบัญภาพ

ภาพที่	หน้า
1.1 แสดงภาพโดยรวมของโครงการ.....	2
2.1 แสดงวงจรที่ใช้ในการถอดรหัส.....	7
2.2 แสดงภาพการรูดการ์ดและวงจรถอดรหัส.....	8
2.3 แสดงโครงสร้างของเครื่องรูดการ์ด.....	8
3.1 แสดงการวางปุ่มบน Form.....	9
3.2 แสดงการเลื่อนไปจุดเริ่มต้น.....	10
3.2 แสดงการดับเบิลคลิกเพื่อสร้าง Event Click.....	10
3.4 แสดงการตอบสนองต่อ Textbox .....	15
3.5 แสดง Control ที่ได้รับการ focus.....	19
3.6 เมนู (Menu) .....	20
3.7 แสดง Menu Editor .....	21
3.8 แสดง ปุ่ม ของ Menu ต่างๆ.....	22
3.9 แสดงการกำหนด Menu Editor และผลลัพธ์.....	22
3.10 แสดงคีย์ลัดที่เกิดขึ้น.....	23
3.11 แสดงการตั้งค่าคีย์ลัดให้เมนู.....	23
3.12 แสดงการสร้าง Menu แบบมีเส้นค้น.....	23
3.13 ก แสดงการเพิ่ม Control .....	25
3.13 ข เลือกตามภาพ.....	25
3.13 ค MSComm ที่เพิ่มเข้ามา.....	25
3.14 ก แสดงการเพิ่ม Control .....	29
3.14 ข เลือกตามภาพ.....	29
3.14 ค Winsock ที่เพิ่มเข้ามา.....	29
3.15 การสร้าง Project ใหม่.....	30
3.16 รูปแบบโปรแกรมตัวอย่าง.....	30
3.17 ก แสดงการ Set ค่าต่างๆให้ Control .....	31
3.17 ข แสดงการ Set ค่าต่างๆให้ Control.....	32
3.17 ค แสดงการ Set ค่าต่างๆให้ Control .....	33
3.18 แสดงการ Save Project.....	34
3.19 แสดงโปรแกรมที่จะสร้าง.....	35

## สารบัญภาพ(ต่อ)

ภาพที่	หน้า
3.20 การวาง Control.....	35
3.21 ก แสดงการ set properties.....	35
3.21 ข แสดงการ ser properties .....	36
3.22 แสดงการ Run โปรแกรมทั้งสอง.....	37
3.23 แสดงผลการ Run โปรแกรม.....	38
3.24 แสดงผลการ Run โปรแกรม.....	38
4.1 Network Neighborhood .....	39
4.2 Properties.....	39
4.3 แสดงหลักเลือก properties.....	40
4.4 แสดงรายการ Protocol.....	40
4.5 แสดงการเพิ่ม Protocol.....	41
4.6 Set ตามภาพ.....	41
4.7 แสดงการใส่ Computer Name.....	42
4.8 แสดงการ Set ค่า Protocol.....	42
4.9 แสดงเครื่องลูกข่ายเมื่อทำการ Set เรียบร้อยแล้ว.....	43
5.1 แสดง Flowchart ของหลักการทำงานโดยรวมของ Software.....	44
5.2 แสดง Flowchart การตรวจสอบการเข้า-ออกจากบริการ.....	45
5.3 แสดงส่วนการทำงานของการเข้าใช้บริการ.....	46
5.4 แสดงส่วนการทำงานของการออกจากบริการ.....	47
5.5 ภาพแสดงหน้าหลักโปรแกรม.....	49
5.6 แสดง Menu สำหรับการ SetComPort.....	50
5.7 แสดงการเลือกเครื่องลูกข่ายที่ต้องการใช้งาน.....	50
5.8 แสดง Tap Menu .....	50
5.9 แสดงโปรแกรมในส่วน ofฐานข้อมูลลูกค้า.....	51
5.10 History .....	51
5.11 แสดงส่วน INPUT .....	52
5.12 ComMonitor .....	52
5.13 แสดงในส่วนของการลดจำนวนเงินตามเวลา.....	53
5.14 Logo ของโปรแกรม.....	54

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## สารบัญภาพ(ต่อ)

ภาพที่		หน้า
6.1	แสดงการ Set ค่า Serial Port.....	55
6.2	แสดงการเลือกตำแหน่งของเครื่องที่ Tap ControlPanel.....	56
6.3	แสดงค่าที่รับมาจากเครื่องรูดการ์ด.....	56
6.4	แสดงการแจ้งที่ History.....	56
6.5	แสดง Tap ComMonitor.....	57



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

# บทที่ 1

## บทนำ

วัตถุประสงค์ในการศึกษาโครงการนี้เพื่อเรียนรู้การเชื่อมต่ออุปกรณ์ภายนอกกับคอมพิวเตอร์โดยสามารถเรียนรู้การใช้งานพอร์ทัลต่างๆของคอมพิวเตอร์กับอุปกรณ์ที่กำหนดได้โดยใช้ข้อกำหนดต่างๆตามมาตรฐานที่กำหนดไว้ได้และสามารถนำความรู้ที่ได้มาประยุกต์ใช้ให้เกิดประโยชน์ตามวัตถุประสงค์ที่ต้องการ

ในที่นี่จะนำเครื่องอ่านการ์ดแถบแม่เหล็กมาประยุกต์ใช้งานเชื่อมต่อกับพอร์ทัล RS-232 ของคอมพิวเตอร์ เพื่อนำมาใช้ควบคุมการใช้งานอินเทอร์เน็ต โดยมีขอบเขตของการศึกษาดังนี้

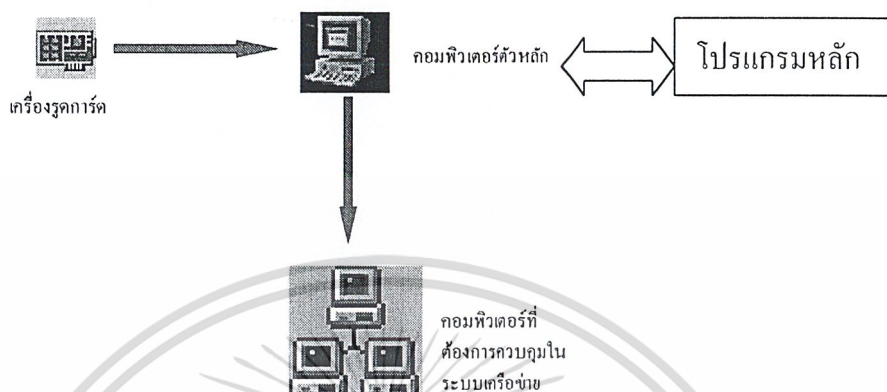
1. สามารถอธิบายหลักการทำงานของเครื่องรูดการ์ดได้
2. สามารถประยุกต์อุปกรณ์รองรับการรูดการ์ดได้
3. สามารถสร้างอุปกรณ์เพื่อทำการเชื่อมต่ออุปกรณ์รองรับการรูดการ์ดกับคอมพิวเตอร์ได้
4. สามารถออกแบบโปรแกรมเพื่อควบคุมการถอดรหัสข้อมูลที่ได้จากการรูดการ์ดได้ โดยควบคุมทางด้านของเครื่องรูดการ์ด
5. สามารถออกแบบโปรแกรมเพื่อควบคุมการรับข้อมูลที่ส่งมาจากอุปกรณ์ภายนอกได้ โดยสามารถแสดงผลที่ถูกต้องได้
6. สามารถออกแบบโปรแกรมเพื่อประยุกต์การใช้งานได้ในส่วนของการเชื่อมต่อกับผู้ใช้งาน
7. ส่วนของโปรแกรมการเชื่อมต่อกับผู้ใช้งานจะสามารถแสดงเวลาคงเหลือของการใช้งานอินเทอร์เน็ตได้ โดยจะแบ่งการทำงานออกเป็นหลายๆส่วน

จากขอบเขตการศึกษาดังกล่าวเราจึงแบ่งการศึกษาออกเป็น 2 ส่วนใหญ่ๆ คือ ส่วนของ Hardware กับส่วนของ Software

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## ภาพโดยรวมของโครงการ

โครงการจะแบ่งตามโครงสร้างได้ 4 ส่วน ดังภาพที่



ภาพที่ 1.1 แสดงภาพโดยรวมของโครงการ

เมื่อทำการรูดการ์ดที่เครื่องอ่านบัตรแม่เหล็ก จะมีการถอดรหัส โดยอ่านค่าที่บันทึกในบัตรออกมา แล้วส่งให้คอมพิวเตอร์ คอมพิวเตอร์จะทำการเปรียบเทียบรหัสที่ได้กับรหัสที่เก็บไว้ในฐานข้อมูล แล้วทำการตั้งค่านิยามข้อมูลมาคำนวณต่อไป

โดยจะมีการเขียนโปรแกรมไว้ส่งค่าออกมาควบคุมอุปกรณ์ภายนอกในกรณีที่ตรงกับเงื่อนไขที่โปรแกรมไว้

### การทำงานเมื่อนำไปใช้งานจริง

บัตรจะต้องมีการกำหนดราคาเหมือนกับบัตรโทรศัพท์ไว้แล้ว โดยบัตรแต่ละใบจะมีรหัสที่ตรงกับฐานข้อมูลที่อยู่ในคอมพิวเตอร์อยู่แล้ว โดยพนักงานมีหน้าที่ จ่ายบัตรอย่างเดียวเมื่อมีลูกค้ามาใช้บริการครั้งแรก ผู้ให้บริการจะทำการออกบัตรที่มีรหัสให้ โดยรหัสนี้จะถูกกำหนดโดยผู้ให้บริการเอง

สมมติว่าลูกค้าซื้อชั่วโมงอินเทอร์เน็ตในราคา 100 บาท ทางร้านจะออกบัตรที่มีราคา 100 บาทให้ โดยเมื่อลูกค้านำบัตรไปรูดที่เครื่องรูด เครื่องรูดจะส่งรหัสไปที่ คอมพิวเตอร์ เพื่อเปรียบเทียบ รหัสกับฐานข้อมูลที่มีอยู่ เมื่อพบว่ามี รหัสนี้อยู่จริง ก็จะแสดงค่าเวลาคงเหลือที่สามารถเล่นอินเทอร์เน็ตได้ และ จะทำการค้นหาเครื่องลูกข่ายที่ว่างเพื่อกำหนดให้ลูกค้าไปใช้งานได้

### แนวความคิดในการสร้าง

ทำไม? ต้องทำต้องประยุกต์เป็นเครื่องควบคุมการใช้งานอินเทอร์เน็ต

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ในปัจจุบันอินเทอร์เน็ต คาเฟ่ เป็นที่แพร่หลายมากในสังคมแห่งเทคโนโลยี เพราะมีความสะดวกทั้งในด้านการธุรกิจ หรือทางด้านการบินทาง จึงมีการเปิดร้านทำธุรกิจอินเทอร์เน็ตขนาดเล็กรูปร่างขึ้นมามากมาย แต่ปัญหาหนึ่งที่พบบ่อยๆในการทำธุรกิจประเภทนี้คือ การคิดเงินค่าเวลาการใช้งาน ซึ่งปัญหาที่เกิดขึ้นนี้

### 1. การเก็บค่าบริการ

เนื่องจากการทำงานจริงๆเจ้าของผู้ได้รับผลประโยชน์อาจไม่ได้ดูแลเรื่องการรับเงินค่าบริการ \* เองแค่ได้ จ้างพนักงานเป็นคนเก็บเงิน จึงอาจจะเกิดปัญหาเรื่องการยกยอกเงินไปทำให้ได้เงินไม่เต็มเม็ดเต็มหน่วย

### 2. เรื่องความทันสมัย

จะทำให้เป็นที่นิยมและเป็นที่ยอมรับได้ง่ายทำให้เกิดผลดีในด้านธุรกิจ

### 3. เรื่องความสะดวกสบาย

จะมีความสะดวกสบายมากขึ้น เพราะไม่ต้องคอยมาจัดการเรื่องการเปิดปิดเครื่อง \*\*\* การเก็บค่าบริการจากการสำรวจจากร้านที่ให้บริการทาง internet จะใช้วิธีการเก็บเงินก่อนเล่น โดยมีอัตราส่วนในการคิดเงิน มีหน่วยเป็น บาท ต่อ นาที แล้วแต่ผู้ให้บริการจะเป็นผู้กำหนด

\*\* เนื่องจากในเวลาเครื่องใดที่ไม่ได้ใช้งานเครื่องนั้นจะถูกให้ STAND BY เพื่อประหยัดพลังงาน

### คุณสมบัติของโครงการ

โครงการนี้มีคุณสมบัติดังนี้

1. สามารถซื้อบัตรเพื่อสร้างฐานข้อมูลของตนเองได้ (สำหรับลูกค้า)
2. ทำให้สามารถเล่นได้โดยมีกำหนดเวลาที่แน่นอน ตามวงเงินที่กำหนด(สำหรับลูกค้า)
3. สามารถป้องกันการทุจริตในการเก็บค่าบริการได้ เพราะจำนวนเงินสามารถกำหนดได้จากจำนวนบัตรที่จ่ายออกไป(สำหรับเจ้าของกิจการ)
4. มีความสะดวกในการควบคุมการใช้งานเพราะเมื่อรู้อัตรเครื่องจะทำการเปิดลูกข่ายในวงเครือข่ายที่ว่างจากการใช้งานขึ้นมาเองทำให้ไม่ต้องคอยไปเปิดปิด(สำหรับเจ้าของกิจการ)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## บทที่ 2

### Hardware

#### Magnetic card

หรือที่เราเรียกกันว่า บัตรแถบแม่เหล็กเราคุ้นเคยกันดีในรูปแบบของบัตรเงินสด บัตร ATM บัตรเครดิต บัตร VISA และบัตรอื่นๆ อีกมากมาย ฯลฯ เป็นการ์ดแถบแม่เหล็ก ซึ่งบนตัวบัตรเองจะบันทึกข้อมูลและรายละเอียดต่างๆ ของบัตรไว้ในรูปของเส้นแรงแม่เหล็กภายในส่วนที่เป็นแถบแม่เหล็กในการ์ด โดยที่แถบแม่เหล็กที่อยู่บนการ์ด ซึ่งเรียกว่า “Track” นั้น ตามปกติแล้วจะถูกแบ่งออกเป็น 3 ส่วนด้วยกัน ซึ่งในแต่ละส่วนจะใช้เก็บข้อมูลซึ่งมีความหนาแน่นและลักษณะของข้อมูลที่มีความแตกต่างกัน ดังนี้คือ

แถบแม่เหล็ก	ความหนาแน่นในการบันทึก	การเข้ารหัสข้อมูล	จำนวนตัวอักษรสูงสุด	ลักษณะของข้อมูลที่เก็บในบัตร
Track 1	210 BPI	ALPHA	79	ชื่อเจ้าของบัตรและหมายเลขบัตร
Track 2	75 BPI	BCD	40	หมายเลขบัตรและวันหมดอายุ
Track 3	210 BPI	BCD	107	หมายเลขบัตรและรหัสพิเศษ

หมายเหตุ ความหนาแน่นของการบันทึกข้อมูลมีหน่วยเป็น BPI ( Byte Per Inch )

ในการทำโครงงานครั้งนี้เราจะใช้เครื่องอ่านแถบแม่เหล็กมาตรฐานเป็นส่วน Hardware ของการอ่านค่าจากการ์ด

เครื่องอ่านแถบแม่เหล็ก TOKIN รุ่น MCS-131P-3 นี้จะอ่านได้เฉพาะข้อมูลที่บันทึกด้วยความละเอียด 75 BPI คือเฉพาะในส่วนของแถบแม่เหล็กแถบที่ 2 (Track2) เท่านั้น โดยมีสัญญาณที่ใช้เชื่อมต่อเพื่ออ่านข้อมูลจากเครื่องอ่านแถบแม่เหล็กทั้งหมด 5 สัญญาณ โดยเป็นสัญญาณเอาต์พุตส่งออกมาจากเครื่องอ่านแถบแม่เหล็กทั้งหมด และมีขาไฟเลี้ยงวงจรที่ต้องต่อให้เครื่องอ่านอีก ดังนี้คือ

1. ไฟเลี้ยงวงจร ( +VCC ) ซึ่งต้องมีค่าเป็น +5VCC ซึ่งต่อจากภายนอกให้เครื่องอ่านแถบเอกสารแม่เหล็กสารที่ส่งวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2. Ground เป็นจุดอ้างอิงของสัญญาณต่างๆ ระหว่างเครื่องอ่านแถบแม่เหล็กกับบอร์ดไมโครฯ

3. สัญญาณข้อมูล ( Data ) เป็นสัญญาณข้อมูลเข้าที่พุทส่งออกมาจากเครื่องอ่านแถบแม่เหล็ก ซึ่งเป็นสัญญาณของข้อมูลที่อ่านได้จากบัตรแถบแม่เหล็ก โดยในการอ่านสัญญาณข้อมูลนี้ต้องอ่านแบบอนุกรมทีละบิต โดยต้องพิจารณาให้สัมพันธ์สอดคล้องกับสัญญาณนาฬิกา ( Clock ) ของบัตรด้วย ซึ่งสัญญาณข้อมูลที่อ่านได้นี้จะกลับสภาวะกับสัญญาณจริง คือมีสภาวะเป็นตรงข้ามกันอยู่ ดังนั้น เมื่ออ่านสัญญาณจากเครื่องอ่านแถบแม่เหล็กได้แล้วจะต้องกลับสภาวะของสัญญาณ ( Complement ) เสียก่อนจึงจะได้ข้อมูลที่ถูกต้อง

4. สัญญาณนาฬิกา ( Clock ) เป็นสัญญาณเข้าที่พุทส่งออกมาจากเครื่องอ่านแถบแม่เหล็ก ใช้เป็นสัญญาณอ้างอิงในการอ่านสัญญาณข้อมูลจากบัตร โดยการอ่านสัญญาณข้อมูลแต่ละบิตนั้นต้องอ่านในขณะที่สัญญาณนาฬิกานี้เป็นขอบขาลง ( Falling Edge ) เสมอ ซึ่งสัญญาณข้อมูลที่อ่านได้นั้นจะเริ่มต้นจากบิตที่มีนัยสำคัญต่ำสุด ( LSB ) ก่อนเป็นอันดับแรก

5. สัญญาณ Present เป็นสัญญาณสภาวะเข้าที่พุทส่งออกมาจากเครื่องอ่านแถบแม่เหล็กทำงานที่ลอจิกศูนย์เมื่อสัญญาณนี้เกิด ( “ 0 ” ) จะบ่งบอกให้ทราบว่าเครื่องอ่านแถบแม่เหล็กเริ่มต้นส่งข้อมูลออกมา หรือมีการนำบัตรแถบแม่เหล็กไปรูดผ่านหัวอ่านสัญญาณของเครื่อง เมื่อสัญญาณนี้หมดลง ( กลับเป็นลอจิกหนึ่ง ) อีกครั้งหนึ่งแสดงว่าข้อมูลจากการอ่านในครั้งนั้นถูกส่งออกไปหมดสิ้นเรียบร้อยแล้ว ซึ่งเราอาจใช้ประโยชน์ จากสัญญาณนี้ ส่งเป็นสัญญาณ Interrupt ให้ CPU เพื่อทำการอ่านข้อมูล

### การอ่านข้อมูลจากบัตรแถบแม่เหล็ก

การใช้เครื่องอ่านแถบแม่เหล็กเพื่อให้อ่านข้อมูลจากบัตรแถบแม่เหล็กนั้น จะใช้วิธีการนำบัตรส่วนที่เป็นแถบแม่เหล็กไป Slide ผ่านหัวอ่านของเครื่องอ่านไปตามทิศทางและตำแหน่งที่กำหนดไว้เท่านั้น โดยให้สังเกตตำแหน่งของหัวอ่านกับส่วนที่เป็นแถบแม่เหล็กของบัตรต้องอยู่ในแนวเดียวกัน และเนื่องจากเครื่องอ่านแถบแม่เหล็กสามารถอ่านข้อมูลจากบัตรแถบแม่เหล็กได้เฉพาะข้อมูลของ Track2 เท่านั้น ดังนั้นส่วนที่ 2 ( track2 ) ซึ่งใช้บันทึกสัญญาณด้วยความหนาแน่นของสนามแม่เหล็ก 75 BPI ซึ่งจะบรรจุจำนวนรหัสของข้อมูลต่างๆ ใน Track นี้ ได้สูงสุดไม่เกิน 40 ตัวอักษร โดยนับรวมรหัสควบคุมและรหัสตรวจสอบด้วย ซึ่งมีรายละเอียดดังนี้ คือ

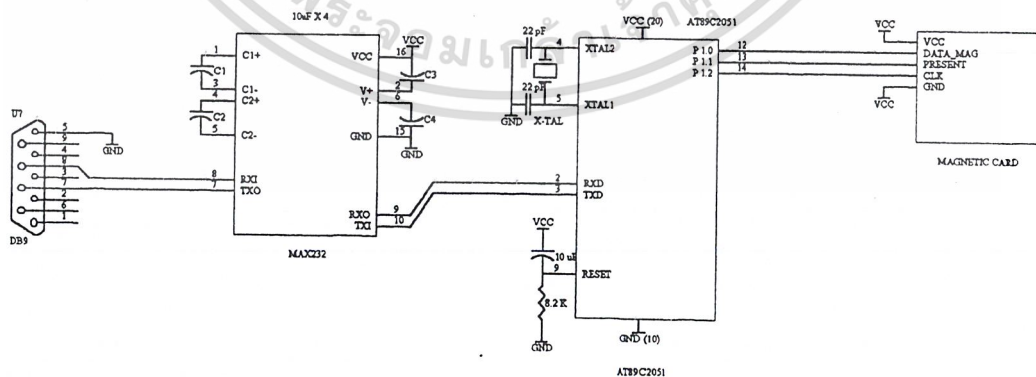
SS	PAN	FS	Addition Data	ES	LRC
----	-----	----	---------------	----	-----

- SS - รหัสข้อมูลแสดงการเริ่มต้นของข้อมูลในบิต ( Start Sentinel ) มีรหัสข้อมูลเป็น OBH ( : )
- PAN - ข้อมูลแสดงหมายเลขของบิต ( มีจำนวนข้อมูลในส่วนนี้สูงสุดไม่เกิน 19 หลัก )
- FS - รหัสข้อมูลแสดงการแบ่งแยกข้อมูล ( Field Sperator ) มีรหัสข้อมูลเป็น ODH ( = )
- Additionnal Data - เป็นข้อมูลเพิ่มเติมอื่นๆ ของบิต เช่น เดือน / ปี ที่ออกบิตและหมดอายุ
- ES - รหัสข้อมูลแสดงการสิ้นสุดของข้อมูลในบิต ( End Sentinel ) มีรหัสข้อมูลเป็น OFH ( ? )
- LRC - ข้อมูลตรวจสอบความผิดพลาด ( Longitudeinal Redundancy Clock )

จากภาพซึ่งใช้แสดง ลักษณะการจัดเรียงข้อมูลที่เก็บไว้ในแถบแม่เหล็กในส่วนของ Track2 ซึ่งจัดเรียงลำดับความสำคัญจากซ้ายไปขวาแบบอนุกรมทีละบิต โดยจะสังเกตเห็นว่าข้อมูลซึ่งเป็นโลจิกศูนย์หรือที่เรียกว่า “ Clocking Bit ” นำหน้าและปิดท้ายข้อมูลจริงอยู่ ซึ่งลักษณะการจัดเก็บข้อมูลของ Track2 จะเก็บข้อมูลด้วยการเข้ารหัสแบบ “ Modulo5 “ ซึ่งในแต่ละชุดข้อมูลจะประกอบด้วยข้อมูล 5 บิต โดย 4 บิตแรก ( D0..D3 ) เป็นรหัสข้อมูลแบบ BCD ส่วนบิตที่ 5 ( D4 ) เป็นพาริตีบิตแบบคี่ ( ODD ) เพื่อใช้ตรวจสอบความถูกต้องของข้อมูลในแต่ละไบท์ ( 5 – Bit ) ที่อ่านได้ ซึ่งหากค่าของบิตพาริตีผิดพลาด แสดงว่าการอ่านข้อมูลนั้นล้มเหลว โดยเริ่มทำการอ่านข้อมูลเมื่อสัญญาณ Present เริ่มเป็นศูนย์ก่อน และทำการอ่านข้อมูลทุกๆ ขอบขาลงของสัญญาณนาฬิกาเสมอ ซึ่งข้อมูลในส่วนก่อนเริ่มต้นและหลังจากสิ้นสุดของการอ่านนี้จะมีค่าเป็นศูนย์ ( สัญญาณ Data = Logic “1” ) นำหน้าและปิดท้ายข้อมูลจริงอยู่ ซึ่งเรียกว่า “ Clocking Bit “ ซึ่งข้อมูลในส่วนนี้เราไม่ต้องสนใจ แต่ให้ตรวจสอบและรองนกว่าจะเริ่ม เป็นข้อมูล Start บิต ( OBH ) หรือ 11010 ในที่นี้เรียงลำดับความสำคัญจากซ้ายไปขวา ซึ่งบิตเริ่มต้น ของรหัส Start หรือ OBH ต้องเริ่มด้วย 1 เป็นบิตแรกเสมอ ดังนั้นในการอ่านเราต้อง รองนกว่าจะพบสัญญาณข้อมูลมีค่าเป็น “1” ( สัญญาณ Data – Logic “0” เพราะกลับสภาวะกันอยู่ ) จึงเริ่มเก็บ ข้อมูล ชุดละ 5 บิต ไปเรื่อยๆ จนถึงรหัสจบ ( OFH ) ซึ่งเมื่อพบรหัสจบแล้วจะมีข้อมูลตามมาอีก 1 ไบท์ ซึ่งเป็นข้อมูลสำหรับตรวจสอบความผิดพลาดของการอ่านข้อมูลทั้งหมดใน Track2 เรียกว่า “LRC” ซึ่งค่าของ “LRC” สามารถหาได้จากการนำเอาข้อมูลในแต่ละไบท์ ( ไม่คิดพาริตีบิต ) ตั้งแต่เริ่มต้น จนถึงสุดมาทำการ XOR กับไบท์ถัดไปเรื่อยๆ ตามลำดับ ซึ่งหากผลลัพธ์สุดท้ายที่ได้ไม่เท่ากับ ค่าของ “ LRC “ ที่อ่านมาได้แสดงว่าการอ่านข้อมูลทั้งหมดล้มเหลว

สำหรับข้อมูลของรหัส BCD ที่ใช้สำหรับเครื่องอ่านแถบแม่เหล็กนี้จะเป็นข้อมูลชุดละ 5 บิต โดยเป็นข้อมูลจริง 4 บิต และเป็นรหัสตรวจสอบพาริตีอีก 1 บิต ซึ่งมีจำนวนทั้งหมด 16 อักขระ ดังนี้คือ

Parity	D3	D2	D1	D0	Character	Function
1	0	0	0	0	0 (0H)	Data
0	0	0	0	1	1 (1H)	Data
0	0	0	1	0	2 (2H)	Data
1	0	0	1	1	3 (3H)	Data
0	0	1	0	0	4 (4H)	Data
1	0	1	0	1	5 (5H)	Data
1	0	1	1	0	6 (6H)	Data
0	0	1	1	1	7 (7H)	Data
0	1	0	0	0	8 (8H)	Data
1	1	0	0	1	9 (9H)	Data
1	1	0	1	0	: (AH)	Control
0	1	0	1	1	; (BH)	Start Sentinel
1	1	1	0	0	< (CH)	Control
0	1	1	0	1	= (DH)	Field Separator
0	1	1	1	0	> (EH)	Control
1	1	1	1	1	? (FH)	End Sentinel



ภาพที่ 2.1 แสดงวงจรที่ใช้ในการถอดรหัส

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โดยจากหลักการทำงานทั้งหมดนี้เราสามารถนำมาสร้างวงจรเพื่อตรวจสอบสัญญาณที่ได้จากเครื่องรูดการ์ด ดังภาพ



ภาพที่ 2.2 แสดงภาพการรูดการ์ดและวงจรถอดรหัส



ภาพที่ 2.3 แสดงโครงสร้างของเครื่องรูดการ์ด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## บทที่ 3

### Software

ในการทำงานของโปรแกรมที่จะต้องเขียนด้วย VisualBasic (ซึ่งต่อไปจะเรียกว่า VB ) ต้องมีการศึกษาเกี่ยวกับการเขียนโปรแกรมด้วย VB ดังนั้นในบทนี้เราจะกล่าวถึงพื้นฐานการเขียนโปรแกรมด้วย VB แบบ ConTrol พื้นฐานในการเขียนโปรแกรม ซึ่งสิ่งที่จะกล่าวถึงนี้เพียงพอที่จะทำโครงการนี้ขึ้นได้

#### การออกแบบหน้าจอ VB6

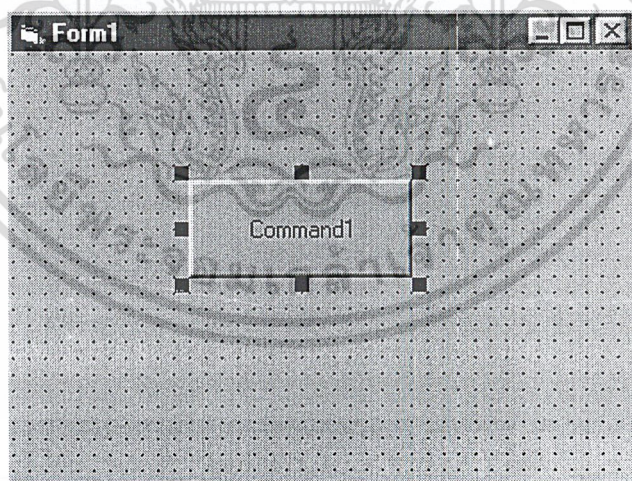
การนำคอนโทรลที่มีอยู่มาออกแบบฟอร์มให้เหมาะสม จะเห็นได้ว่าทั้งฟอร์มและคอนโทรลนั้น เป็นเครื่องมือพื้นฐานในการเขียนโปรแกรมด้วย VB6

เราจะกล่าวถึงหลักการพื้นฐานเกี่ยวกับฟอร์ม และคอนโทรลที่เราจำเป็นต้องทราบ เพื่อให้เราเข้าใจหลักการในการออกแบบหน้าจอมากขึ้น

#### คุณสมบัติ เมตอด และอีเวนต์

ในการทำงานกับฟอร์มและคอนโทรล เราจะต้องทำความเข้าใจกับ 3 คำต่อไปนี้ คือ

1. คุณสมบัติ (Properties) ที่ให้เรากำหนดลักษณะต่าง ๆ ของฟอร์ม และคอนโทรล เช่น ปุ่มคำสั่งชื่อ MyCmd มีคุณสมบัติที่เรากำหนดได้ ดังภาพ



ภาพที่ 3.1 แสดงการวางปุ่มบน Form

เราสามารถกำหนดคุณสมบัติสำหรับคอนโทรลต่าง ๆ ผ่านทางหน้าต่าง Properties หรือโดยใช้คำสั่งที่มีรูปแบบดังต่อไปนี้ได้

MyCmd.Caption = "OK" เป็นการกำหนดค่าให้คุณสมบัติ Caption ของ MyCmd

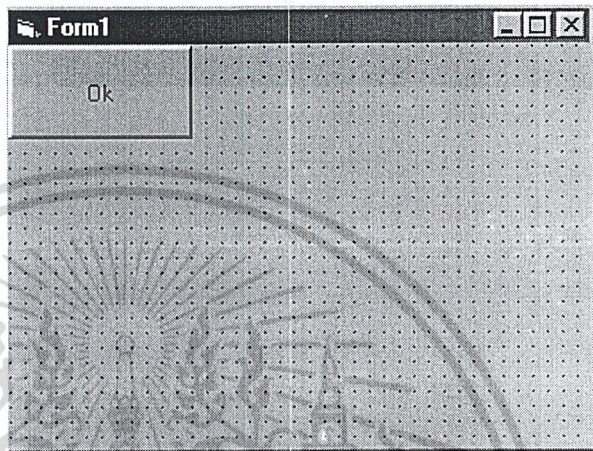
MyCmd.Height = 1000 เป็นการกำหนดค่าให้คุณสมบัติ Height ของ MyCmd

MyCmd.Width = 2000 เป็นการกำหนดค่าให้คุณสมบัติ Width ของ MyCmd

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากข้างต้น ข้อมูลชนิดข้อความจะต้องมีเครื่องหมาย “ คลุมห้วท้ายด้วย ส่วนข้อมูลชนิดตัวเลขไม่จำเป็นต้องมีเครื่องหมายคลุม

2. เมธอด (Method) เป็นการสั่งให้ฟอร์ม และคอนโทรลทำงานตามที่เราร้องขอไป เช่น การสั่งให้ปุ่มคำสั่งชื่อ MyCmd ในตัวอย่างที่ผ่านมาเปลี่ยนไปวางอยู่บนตำแหน่งเริ่มต้นบนฟอร์มดังภาพ



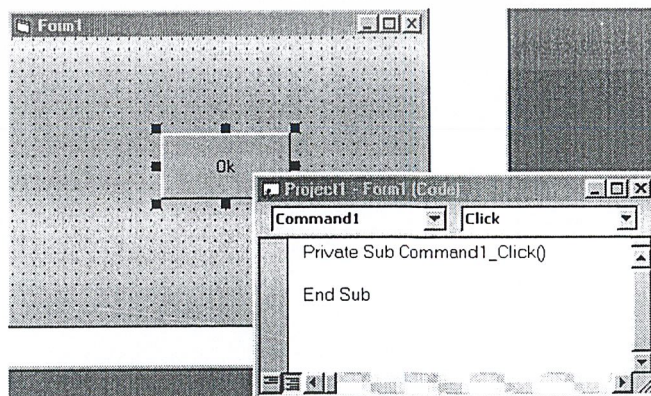
ภาพที่ 3.2 แสดงการเลื่อนไปจุดเริ่มต้น

ในการสั่งให้ปุ่มคำสั่งให้ทำงานตามที่เรต้องการข้างต้น ให้เราใส่คำสั่งต่อไปนี้

```
MyCmd.Move 0, 0
```

โดยที่คำสั่งนี้จะเป็นการสั่งให้คอนโทรลปุ่มคำสั่งชื่อ MyCmd ย้ายไปที่ตำแหน่ง 0, 0 ซึ่งเป็นตำแหน่งดังภาพข้างต้น

3. อีเวนต์ (Events) เป็นเหตุการณ์ที่เกิดขึ้นกับฟอร์มหรือคอนโทรลที่เราสามารถใส่คำสั่งเพื่อตอบสนองได้เช่นถ้าเราต้องการตอบสนองต่ออีเวนต์Clickของปุ่มคำสั่งชื่อMyCmd ดังภาพ



ภาพที่ 3.3 แสดงการดับเบิลคลิกเพื่อสร้าง Event Click

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ในการตอบสนองต่ออีเวนต์ Click ข้างต้น ให้เราใส่คำสั่งต่อไปนี้

```
Sub MyCmd_Click()      ‘ คำสั่งที่ตอบสนองต่ออีเวนต์ Click ของ MyCmd
MyCmd.Caption = “Click”  ‘ เปลี่ยนคุณสมบัติ Caption ของ MyCmd เป็น Click
End Sub                ‘ จบการตอบสนองต่ออีเวนต์ Click ของ MyCmd
```

จากตัวอย่างคำสั่งข้างต้นเมื่อเรา Click mouse ปุ่มคำสั่งแล้วคำว่า OK บนปุ่มคำสั่งจะเปลี่ยนเป็น Click

## ฟอร์ม (Forms)

ฟอร์มจะเป็นเครื่องมือที่เราจะต้องทำงานด้วยบ่อยมาก ในการสร้างโปรแกรมด้วย VB6 โดยที่ฟอร์มจะเป็นหน้าต่างที่ผู้ใช้ติดต่อทำงานด้วย ผ่านทางคอนโทรลต่าง ๆ ที่เราวางบนฟอร์ม ดังนั้นในหัวข้อนี้ เราจะศึกษาคุณสมบัติ เมคกอด และอีเวนต์ของฟอร์มที่จำเป็นก่อน

ในขั้นตอนแรก เราจะต้องกำหนดคุณสมบัติของฟอร์ม ซึ่งอาจจะผ่านทาง Properties Window ในขณะออกแบบ หรือกำหนดผ่านทางคำสั่งโปรแกรมในขณะรันก็ได้

### คุณสมบัติของฟอร์ม

ฟอร์มมีคุณสมบัติต่าง ๆ มากมาย แต่ที่เราจำเป็นต้องรู้จักในตอนนี้ มีดังตารางต่อไปนี้

#### ชื่อคุณสมบัติ

#### คำอธิบาย

Name \*

กำหนดชื่อของฟอร์มที่เราใช้อย่างอิงถึงใน โปรแกรม

BorderStyle

กำหนดลักษณะในการเปลี่ยนขนาดของฟอร์มเป็นอย่างไร

MinButton, MaxButton

กำหนดให้ฟอร์มสามารถ Maximize, Minimize ได้หรือไม่

Caption \*

กำหนดข้อความที่แสดงบนไคเต็ลบาร์ของฟอร์ม

Icon

กำหนดภาพไอคอนของฟอร์มเมื่อ Minimize ฟอร์ม

Height, Width \*

กำหนดความสูง และความกว้างของฟอร์ม

Left, Top \*

กำหนดตำแหน่งของฟอร์ม โดยคิดจากตำแหน่งบนซ้าย

ของหน้าจอ

Movable

กำหนดว่าฟอร์มจะสามารถเคลื่อนย้ายได้หรือไม่

WindowState

กำหนดว่าเมื่อเริ่มต้นเรียกฟอร์มจะให้อยู่ในแบบ

Maximize, Minimize หรือธรรมดา

Enabled \*

กำหนดให้ฟอร์มสามารถตอบสนองต่ออีเวนต์ได้หรือไม่

**Note** คุณสมบัติในตารางที่มีเครื่องหมาย \* เป็นคุณสมบัติที่ใช้กับคอนโทรลอื่นได้ โดยที่จะมีความหมายใกล้เคียงกัน

เพื่อความเข้าใจของคุณ ขอให้ลองกำหนดคุณสมบัติต่าง ๆ ผ่านทาง Properties Window แล้วลองรันโปรแกรม ดูว่ามีผลการทำงานอย่างไร

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## เมตดอด และอีเวนต์ที่สำคัญของฟอร์ม

ฟอร์มมีเมตดอด และอีเวนต์ที่เราจำเป็นต้องรู้จักในตอนนี้ ดังตารางต่อไปนี้

ชื่อเมตดอด, อีเวนต์	คำอธิบาย
Load	เป็นอีเวนต์ที่เกิดเมื่อเรียกฟอร์มขึ้นมาครั้งแรก
Resize	เป็นอีเวนต์ที่เกิดเมื่อมีการเปลี่ยนขนาดของฟอร์ม
Active	เป็นอีเวนต์ที่เกิดขึ้นเมื่อฟอร์มนั้นเป็นฟอร์มที่เราทำงานในขณะนั้น
Deactive	เป็นอีเวนต์ที่เกิดเมื่อฟอร์มอื่น ๆ กลายเป็นฟอร์มที่แอคทีฟแทน
Click	เป็นอีเวนต์ที่เกิดเมื่อมีการ Click mouse บนฟอร์ม
DblClick	เป็นอีเวนต์ที่เกิดเมื่อมีการ Double Click mouse บนฟอร์ม
Show	เป็นเมตดอดที่ใช้แสดงฟอร์มขึ้นมา
Hide	เป็นเมตดอดที่ใช้ซ่อนฟอร์มไว้
Print	เป็นเมตดอดที่ใช้พิมพ์ข้อความออกมาบนฟอร์ม
Cls	เป็นเมตดอดที่ใช้ในการลบสิ่งที่วาด หรือพิมพ์บนฟอร์ม

## ขั้นตอนการสร้างโปรแกรม

1. ให้เราเลือกโปรเจกต์ชนิด Standard.EXE
2. วางคอนโทรลต่าง ๆ บน Form 1 และกำหนดคุณสมบัติต่าง ๆ ของฟอร์ม และคอนโทรลบน Form1 ดังตารางต่อไปนี้

ชนิดคอนโทรล	คุณสมบัติ	ค่าที่กำหนดให้	คำอธิบาย
Form	Name	Form1	ชื่อของฟอร์ม
	AutoRedraw	True	เพื่อไม่ให้ข้อความหายไปเมื่อมีฟอร์มอื่นบัง
Command Button	Name	Command1	ชื่อปุ่มคำสั่ง
	Caption	Show Form2	ข้อความที่ปรากฏบนปุ่มคำสั่ง
Command Button	Name	Command2	ชื่อปุ่มคำสั่ง
	Caption	Clear	ข้อความที่ปรากฏบนปุ่มคำสั่ง

เอกสารนี้เป็นเอกสารลิขสิทธิ์ของสถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง 3. ให้เพิ่มฟอร์มชื่อ Form2 เข้าไป ด้วยการ Click เลือกเมนู Project>Add Form ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4. จะปรากฏฟอร์มใหม่ชื่อ Form2 ให้เรากำหนดคุณสมบัติของฟอร์ม และคอนโทรลดังตารางต่อไปนี้

ชนิดคอนโทรล	คุณสมบัติ	ค่าที่กำหนดให้	คำอธิบาย
Form	Name	Form2	ชื่อของฟอร์ม
	AutoRedraw	True	เพื่อไม่ให้ข้อความหายไปเมื่อมีฟอร์มอื่นบัง
Command Button	Name	Command1	ชื่อปุ่มคำสั่ง
	Caption	Clear	ข้อความที่ปรากฏบนปุ่มคำสั่ง

5. ต่อจากนั้นให้เราใส่คำสั่งโปรแกรมต่อไปนี้ลงไปในโมดูล Form1

```

Private Sub Command1_Click()
    Form2.Show ' เมื่อ Click mouse ให้แสดง Form2
End Sub

-----

Private Sub Command2_Click()
    Form1.Cls ' เมื่อ Click mouse ให้ลบข้อความบน Form1 ออก
End Sub

-----

Private Sub Form_Activate()
    Form1.Print "Form_Activate" ' ตอบสนองต่ออีเวนต์ Activate
End Sub

-----

Private Sub Form_Click()
    Form1.Print "Form_Click" ' ตอบสนองต่ออีเวนต์ Click
End Sub

```

## ปุ่มคำสั่ง (Command Button)

เราคงพอจะได้ทราบกันมาบ้างแล้วว่า ปุ่มคำสั่งนั้นมีหน้าที่หลักคือ การตอบสนองต่อการ Click mouse ของผู้ใช้ที่ทำงานมายัง โปรแกรมว่า ต้องการให้โปรแกรมทำอะไรตอบกลับไป ดังนั้น ในหัวข้อนี้ เราจะกล่าวถึงรายละเอียดเพิ่มเติมของปุ่มคำสั่งซึ่งมีดังต่อไปนี้

การเกิดอีเวนต์ Click ของปุ่มคำสั่ง

สำหรับอีเวนต์ Click ของปุ่มคำสั่งนี้ เกิดได้จากหลาย ๆ กรณีดังต่อไปนี้

เกิดจากการ Click mouse ของผู้ใช้งาน

กดปุ่ม <Enter> ปุ่มคำสั่งที่มีคุณสมบัติ Default เป็น True

กดปุ่ม <Esc> ปุ่มคำสั่งที่มีคุณสมบัติ Cancel เป็น True

เกิดจากผู้ใช้กดปุ่ม <Alt>+ ตัวอักษรขีดเส้นใต้อยู่ (เราสามารถกำหนดใน

คุณสมบัติ Caption โดยการใส่ & หน้าตัวอักษรที่ต้องการให้มีการขีดเส้นใต้)

กำหนดคุณสมบัติ Value ของปุ่มคำสั่งให้เท่ากับ True ในขณะที่รันโปรแกรม โดยใช้คำสั่ง

```
Command1.Value = True
```

### คุณสมบัติของปุ่มคำสั่ง

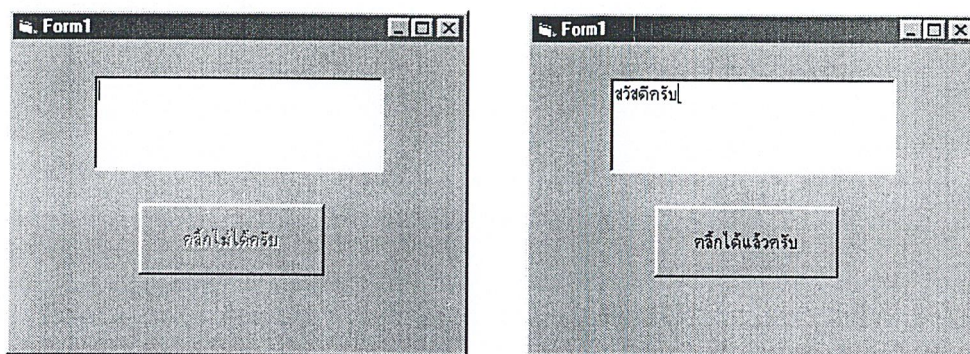
นอกจากนี้ ปุ่มคำสั่งยังมีคุณสมบัติต่าง ๆ ที่น่าสนใจดังต่อไปนี้

ชื่อคุณสมบัติ	คำอธิบาย
Style	กำหนดให้เป็นปุ่มคำสั่งธรรมดา หรือมีรูปภาพบนปุ่มด้วย
Picture	กำหนดรูปภาพที่แสดงบนปุ่มคำสั่ง (ใช้ได้เมื่อคุณสมบัติ Style เท่ากับ 1)
DownPicture	กำหนดรูปภาพที่แสดงบนปุ่มคำสั่ง เมื่อปุ่มถูกกด
DisabledPicture	กำหนดรูปภาพที่แสดงบนปุ่มคำสั่ง เมื่อคุณสมบัติ Enabled เท่ากับ False
ToolTipText	เป็นข้อความที่แสดง เมื่อหยุดตัวชี้เมาส์ที่ปุ่มคำสั่งช่วงเวลาหนึ่ง
Caption	ข้อความที่ปรากฏบนปุ่มคำสั่ง
Enabled	กำหนดให้ปุ่มคำสั่งใช้งานได้ หรือ ไม่ได้
Visible	กำหนดให้คอนโทรลสามารถมองเห็นได้ในตอนรันหรือไม่

### ตัวอย่างโปรแกรมการใช้ปุ่มคำสั่ง

ตัวอย่างนี้จะแสดงการใช้งานปุ่มคำสั่งในการทำงาน โดยให้เราใส่ข้อความในเท็กบ็อกซ์ ถ้าไม่มีข้อความในเท็กบ็อกซ์ ปุ่มคำสั่งจะไม่ตอบสนองการทำงาน แต่ถ้ามีข้อความในเท็กบ็อกซ์ปุ่มคำสั่งจะตอบสนองต่อการ Click mouse ได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



ภาพที่ 3.4 แสดงการตอบสนองต่อ Textbox

1. วางคอนโทรลดังภาพ และกำหนดคุณสมบัติต่างๆ ของคอนโทรลดังตารางต่อไปนี้

ชนิดคอนโทรล	คุณสมบัติ	ค่าที่กำหนดให้	คำอธิบาย
Form	Name	Form1	ชื่อของฟอร์ม
	AutoRedraw	True	เพื่อไม่ให้ข้อความหายไปเมื่อมีฟอร์มอื่นบัง
Command Button	Name	Command1	ชื่อปุ่มคำสั่ง
	Caption	เดิมไม่ได้ครับ	ข้อความที่ปรากฏบนปุ่มคำสั่ง
TextBox	Text1		ชื่อเท็กบ็อกซ์
	Text	""	ข้อความที่ปรากฏในเท็กบ็อกซ์

1. ให้เราในคำสั่งโปรแกรมต่อไปนี้ลงไป

```
Private Sub Text1_Change()
    If Text1.Text = "" _
        Then
            Command1.Caption = "คลิกไม่ได้ครับ"
            Command1.Enabled = False
        Else
            Command1.Caption = "คลิกได้แล้วครับ"
            Command1.Enabled = True
        End If
    End Sub
```

สำหรับคำสั่ง If-Then-Else-End if ข้างต้น เป็นคำสั่งที่ทำงานตามเงื่อนไข โดยถ้าค่าที่อยู่หลัง If เป็นจริง โปรแกรมจะทำตามคำสั่งที่อยู่หลัง Then แต่ถ้าเป็น False จะทำงานตามคำสั่งที่อยู่หลัง Else จนถึง End If

2. จะให้โปรแกรมที่ทำงานตามผลลัพธ์ที่เรากล่าวมาแล้ว

## คอนโทรลสำหรับแสดงและรับข้อมูล

คอนโทรลที่ใช้สำหรับการแสดง และรับข้อมูลก็จะมีเลเบล และเท็กบ็อกซ์ ซึ่งเลเบลจะใช้ในการแสดงข้อมูลอย่างเดียวไม่สามารถให้ผู้ใช้ในข้อมูลเข้ามาได้ ส่วนเท็กบ็อกซ์นั้น จะให้ผู้ใช้ใส่ข้อมูลเข้ามาได้ด้วย เช่น ให้ผู้ใช้ใส่ชื่อ และรหัสผ่าน เป็นต้น ดังนั้นในหัวข้อนี้ เราจะกล่าวถึงคอนโทรลทั้ง 2 นี้กัน

### เลเบล (Label)

เลเบลเป็นคอนโทรลที่ใช้ในการแสดงข้อมูลให้ผู้ใช้เห็น โดยที่ผู้ใช้จะไม่สามารถแก้ไขข้อมูลนั้นได้

### คุณสมบัติของเลเบล

สำหรับเลเบลจะมีคุณสมบัติที่เราจะต้องรู้จักดังตารางต่อไปนี้

ชื่อคุณสมบัติ	คำอธิบาย
Alignment	กำหนดให้ข้อความในเลเบลชิดซ้าย ขวา หรือตรงกลาง
AutoSize	กำหนดให้ขนาดของเลเบลเท่ากับข้อความหรือไม่ (ตามแนวนอน)
WordWrap	กำหนดให้ขนาดของเลเบลเท่ากับข้อความคิดตามแนวตั้ง โดยยังรักษาความยาวตามแนวนอนไว้

สำหรับคุณสมบัติ AutoSize กับ WordWrap จะมีการใช้งานร่วมกันด้วยซึ่งมีกรณีต่าง ๆ ดังนี้

ถ้าไม่ต้องการให้มีการเปลี่ยนแปลงขนาดเลเบลเลย ให้กำหนดคุณสมบัติ AutoSize เท่ากับ False

ถ้าต้องการให้เลเบลมีการขยายตามแนวเท่านั้น ให้กำหนดคุณสมบัติ AutoSize เท่ากับ True

แท็กบ็อกซ์ (TextBox)

คอนโทรลแท็กบ็อกซ์เป็นคอนโทรลที่เราใช้งานบ่อยมาก ในการรับข้อมูลจากผู้ใช้ที่ป้อนเข้ามาในโปรแกรมของเรา รวมทั้งสามารถแสดงผล และให้ผู้แก้ไขข้อมูลได้ด้วย

คุณสมบัติของแท็กบ็อกซ์

ชื่อคุณสมบัติ	คำอธิบาย
MaxLength	กำหนดความยาวสูงสุดของข้อความที่สามารถรับได้ของแท็กบ็อกซ์เป็นเท่าไร
MultiLine	กำหนดให้แสดงและรับข้อความในแท็กบ็อกซ์ในรูปแบบหลายบรรทัดได้หรือไม่
ScrollBars	กำหนดให้แสดงสกรอลบาร์ด้วยหรือไม่และแสดงในแบบใด (แนวนอน แนวตั้ง หรือทั้งสองแนว)
Alignment	กำหนดให้จัดรูปแบบข้อความในคอนโทรลอย่างไร เช่น LeftJustify, Center และ RightJustify จะเป็นอย่างไร คุณสมบัตินี้จะใช้ได้ต่อเมื่อคุณสมบัติ MultiLine มีค่าเป็น True
Locked	กำหนดให้ผู้แก้ไขข้อความในแท็กบ็อกซ์ได้หรือไม่

โดยทั่วไปแล้วแท็กบ็อกซ์จะแสดงข้อความเพียงบรรทัดเดียว ถ้ามีข้อความเกินขนาดของแท็กบ็อกซ์ จะแสดงเท่ากับขนาดของตัวแท็กบ็อกซ์เอง แต่เราสามารถเลื่อนเคอร์เซอร์ไปมาเพื่อดูข้อความทั้งหมดได้ แต่ถ้าเราต้องการให้แท็กบ็อกซ์แสดงข้อความได้หลายบรรทัด ให้เรากำหนดคุณสมบัติ MultiLine เป็น True และถ้าเราต้องการให้มีสกรอลบาร์ด้วย เราต้องกำหนดคุณสมบัติ ScrollBars ให้เป็นตามที่เราต้องการว่าจะมีในแนวใดบ้าง

**Note** ในขณะที่ออกแบบเราไม่สามารถใส่ข้อความในแท็กบ็อกซ์แบบหลายบรรทัดได้ แต่เราสามารถใส่ได้ในคำสั่งโปรแกรม โดยใช้สัญลักษณ์ VbCrLf (อักขระ 13 และ 10 ในรหัส ASCII) คั่นระหว่างข้อความที่จะถูกแยกคนละบรรทัดกัน ดังตัวอย่างคำสั่งต่อไปนี้

```
Sub Command1_Click()
    Text1.Text = "First Name : aha "& VbCrLf & "Last Name : pop"
End Sub
```

เอกสารนี้เป็นเอกสารสงวนลิขสิทธิ์ ห้ามมิให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

First Name : aha

Last Name : pop

### คอนโทรลแสดงตัวเลือก

ในบางครั้งเราต้องการสร้างโปรแกรมที่ให้ผู้ใช้งานเลือกตัวเลือกที่ต้องการได้ ซึ่งจะสะดวกกว่าการคีย์ข้อมูลเข้ามาผ่านเท็กบ็อกซ์ ต่อไปนี้เราจะกล่าวถึงรายละเอียดของคอลโทรลต่อไปนี้

#### เช็kb็อกซ์ (Check Box)

คอนโทรลนี้จะให้ผู้ใช้งานเลือกว่าจะเช็ (ให้มีเครื่องหมายถูกในช่อง) หรือไม่ เพียงอย่างเดียวอย่างหนึ่งเท่านั้น

ค่าของคอนโทรลเช็kb็อกซ์ เราจะได้จากคุณสมบัติ Value ซึ่งจะมีค่าดังตารางต่อไปนี้

การแสดงผลในคอนโทรล	ค่าตัวเลข	ชื่อของค่าคงที่
Unchecked	0	VbUnchecked
Checked	1	VbChecked
Unavailable	2	VbGrayed

โดยที่ค่า 2 นั้น เราจะสามารถกำหนดในโปรแกรมได้เท่านั้น เราอาจจะใช้ชื่อค่าคงที่แทนได้ ดังคำสั่งโปรแกรมต่อไปนี้

Check1.Value = VbChecked

‘ กำหนดให้ Check1 มีการเช็

Check2.Value = VbUnchecked

‘ กำหนดให้ Check2 ไม่มีการเช็

ถ้าเราต้องการกำหนดข้อความที่แสดงบนคอนโทรล ให้เรากำหนดผ่านทางคุณสมบัติ Caption และถ้าเราต้องการตรวจสอบการเช็ ให้เราตรวจสอบผ่านทางอีเวนต์ของเช็kb็อกซ์ คือ Click mouse ซึ่งจะเกิดขึ้นเมื่อผู้ใช้งาน Click mouse ที่คอลโทรลในขณะที่รันโปรแกรม ทำให้เราสามารถตรวจสอบสถานะของเช็kb็อกซ์ได้ ดังตัวอย่างคำสั่งโปรแกรมต่อไปนี้ที่ตรวจสอบการเช็ของเช็kb็อกซ์ชื่อ Check1

```
Sub Check1_Click()
```

```
    If Check1.Value = VbChecked Then
```

```
        ‘ ถ้ามีการเช็เกิดขึ้นมาให้แสดงไดอะล็อกซ์บ็อกซ์บอกว่า มีการเช็
```

```
        MsgBox “Check1 is checked.”, VbOKOnly
```

```
    Else
```

```
        ‘ ถ้ามีการเช็เกิดขึ้นมาให้แสดงไดอะล็อกซ์บ็อกซ์บอกว่า ไม่มีการเช็
```

```
        MsgBox “Check1 is unchecked.”, VbOKOnly
```

```
    End Sub
```

## โฟกัส (Focus)

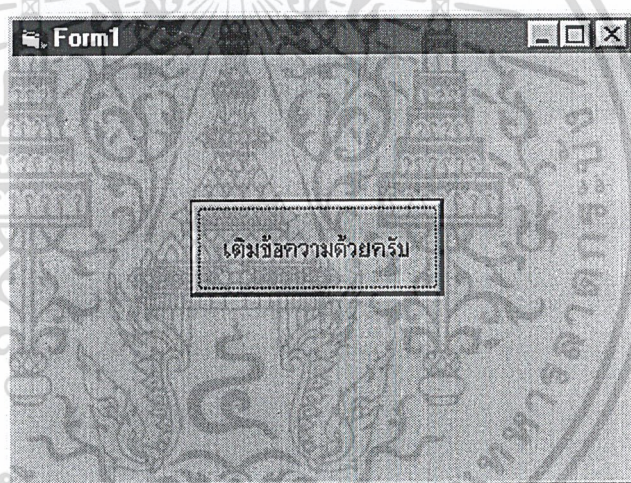
โฟกัสเป็นสิ่งที่ใช้บ่งบอกว่า คอนโทรลใดบนฟอร์มกำลังตอบสนองต่อการใส่ข้อมูลของผู้ใช้โดยคอนโทรลที่ได้รับโฟกัสจะเป็นคอนโทรลที่จะรับข้อมูลจากผู้ใช้ เราสามารถเปลี่ยนโฟกัสไปยังคอนโทรลต่าง ๆ ได้ด้วยวิธีต่าง ๆ ดังต่อไปนี้

เลือกคอนโทรลที่ต้องการให้ถูกรับโฟกัสโดยการกด <TAB> ไปยังคอนโทรล ,Click mouse หรือกดคีย์ลัด (ShortCut) ไปยังคอนโทรลนั้น ๆ

ใช้เมธอด SetFocus ของคอนโทรลที่ต้องการให้ถูกรับโฟกัส ดังคำสั่งต่อไปนี้

```
Sub Command1_Click( )
    Text1.SetFocus
End Sub
```

คอนโทรลที่ได้รับโฟกัสจะมีเส้นกรอบล้อมรอบ ดังภาพ



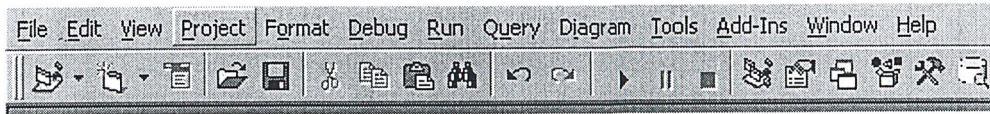
ภาพที่ 3.5 แสดง Control ที่ได้รับการ focus

คอนโทรลจะสามารถรับโฟกัสได้ก็ต่อเมื่อคุณสมบัติ Enabled และ Visible ถูกกำหนดให้เป็น True โดยคุณสมบัติ Enabled จะเป็นการกำหนดว่าคอนโทรลนั้นสามารถใช้งานได้หรือไม่ ส่วนคุณสมบัติ Visible จะเป็นคุณสมบัติที่ใช้กำหนดว่าจะให้คอนโทรลมองเห็นตอนรันโปรแกรมได้หรือไม่

สำหรับอีเวนต์ที่เกี่ยวข้องกับเรื่องโฟกัส คือ GotFocus และ LostFocus ซึ่งจะเกิดขึ้นเมื่อคอนโทรลได้รับโฟกัส และเสียโฟกัสตามลำดับ

## ความรู้เกี่ยวกับเมนู (Menu)

เมนูเป็นเอกลักษณ์อย่างหนึ่งของโปรแกรมบนวินโดวส์ เพราะเป็นที่เก็บคำสั่งต่าง ๆ ที่โปรแกรมมีไว้ให้ผู้ใช้ได้ตอบ ซึ่งส่วนประกอบต่าง ๆ ของเมนูแสดงได้ดังภาพ



ภาพที่ 3.6 เมนู (Menu)

สำหรับการออกแบบเมนูที่ดี เราจำเป็นต้องจัดกลุ่มของคำสั่งต่าง ๆ เป็นหมวดหมู่และนำมาไว้ในแต่ละเมนู โดยอาจจะแบ่งคำสั่งบางขุมออกจากกันด้วย Separator Bar ดังที่เห็นในภาพ คุณสมบัติและอีเวนต์ของเมนู

เมนูเป็นคอนโทรลรูปแบบหนึ่ง ซึ่งมีคุณสมบัติที่สำคัญดังต่อไปนี้

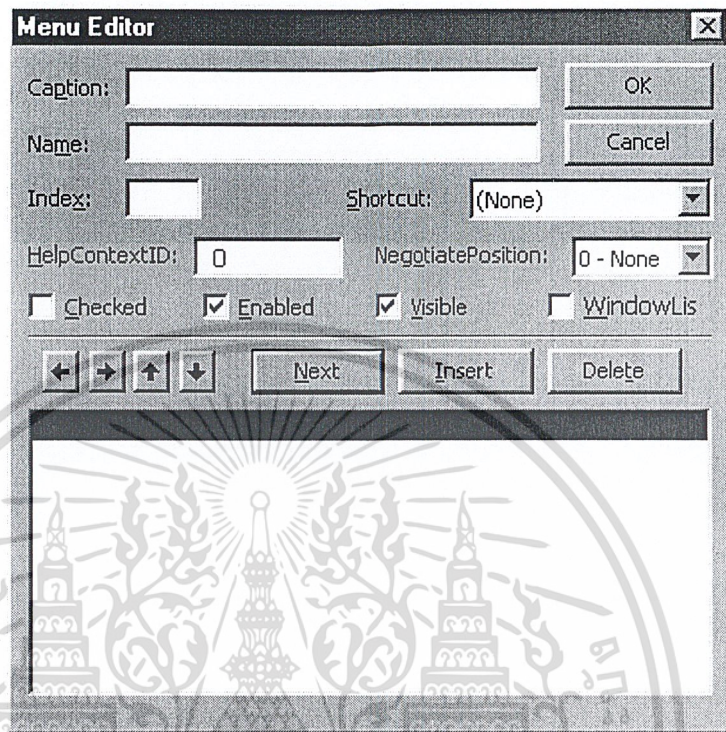
คุณสมบัติ	คำอธิบาย
Name	เป็นชื่อของเมนู
Caption	กำหนดข้อความที่จะให้แสดงบนเมนู
Visible	กำหนดว่าจะให้เมื่อนั้นมองเห็นได้ในรอนรัน หรือไม่
Enabled	กำหนดว่าจะให้เมื่อนั้นตอบสนองต่อผู้ใช้ได้ หรือไม่
Checked	ใช้ตรวจสอบว่า เมื่อนั้นถูกเช็คหรือไม่

สำหรับอีเวนต์ของเมนู จะมีเพียงอีเวนต์เดียว คือ อีเวนต์ Click ที่จะตอบสนองต่อการเลือกเมนูของผู้ใช้

### Menu Editor

เราสามารถ ใช้ Menu Editor สร้างเมนูที่จะนำมาใช้ในโปรแกรมของเราได้โดยที่เราไม่จำเป็นต้องเขียนคำสั่งไปจัดการเลย โดย Menu Editor สามารถจัดการในเรื่องต่าง ๆ เกี่ยวกับเมนู เช่น การเพิ่ม ลบ และแก้ไขรายการที่ปรากฏในเมนู การใช้คีย์ลัด (ShortCuts) สำหรับเรียกใช้เมนู เป็นต้น โดยเมนูที่เราสร้างจะมีลักษณะเหมือนคอนโทรลทั่วไปที่เราสามารถกำหนดคุณสมบัติของเมนูผ่านทาง Properties Window และตอบสนองต่ออีเวนต์ของเมนูได้

สำหรับการเรียกหน้าต่าง Menu Editor ให้เลือกฟอร์มที่จะสร้างเมนู เสร็จแล้วให้เลือกคำสั่ง Tools>Menu Editor จากเมนู ดังภาพ



ภาพที่ 3.7 แสดง Menu Editor

### การใช้งานด้วย Menu Editor





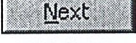
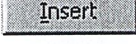

ในหัวข้อนี้เราจะกล่าวถึงวิธีการใช้งาน Menu Editor เพื่อสร้างเมนู ส่วนประกอบต่าง ๆ ของ Menu Editor จะมีรายละเอียด ดังต่อไปนี้

ส่วนที่ 1 เป็นส่วนที่ใช้ในการกำหนดค่าคุณสมบัติของเมนู ซึ่งมีรายละเอียดที่สำคัญดังต่อไปนี้

- Caption เป็นข้อความที่ต้องการให้แสดงบนเมนู
- Name ใช้กำหนดชื่อของรายการในเมื่อนั้นเพื่อใช้อ้างอิงจากในโปรแกรมของเรา
- Shortcut ใช้กำหนดคีย์ลัดให้กับรายการในเมื่อนั้น (กดแล้วจะเป็นการเรียกใช้รายการในเมื่อนั้นเลย)
- Checked ใช้กำหนดให้เมนูแสดงคำสั่งต่าง ๆ ในลักษณะ เหมือนกับตัวเลือก (เช็kb็อกซ์)หรือไม่
- Enabled ใช้กำหนดว่าเมื่อนั้นใช้งานได้หรือไม่ตอนรัน โปรแกรม
- Visible ใช้กำหนดว่าเมื่อนั้นจะมองเห็นเมื่อตอนรันโปรแกรมหรือไม่


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ส่วนที่ 2 เป็นส่วนที่ใช้ในการจัดตำแหน่งและลำดับชั้นของเมนู มีส่วนประกอบ ดังต่อไปนี้

ปุ่ม		ใช้ในการลดลำดับชั้นของเมนูย่อยลง 1 ชั้น
ปุ่ม		ใช้ในการเพิ่มลำดับชั้นของเมนูย่อยขึ้น 1 ชั้น
ปุ่ม		ใช้เลื่อนรายการที่เลือกขึ้น ไปอยู่ข้างบน
ปุ่ม		ใช้เลื่อนรายการที่เลือกกลง ไปอยู่ข้างล่าง
ปุ่ม		ใช้เลื่อนไปยังรายการเมนูถัดไป
ปุ่ม		ใช้เพิ่มรายการใหม่ในเมนู
ปุ่ม		ใช้ลบรายการออกจากเมนู

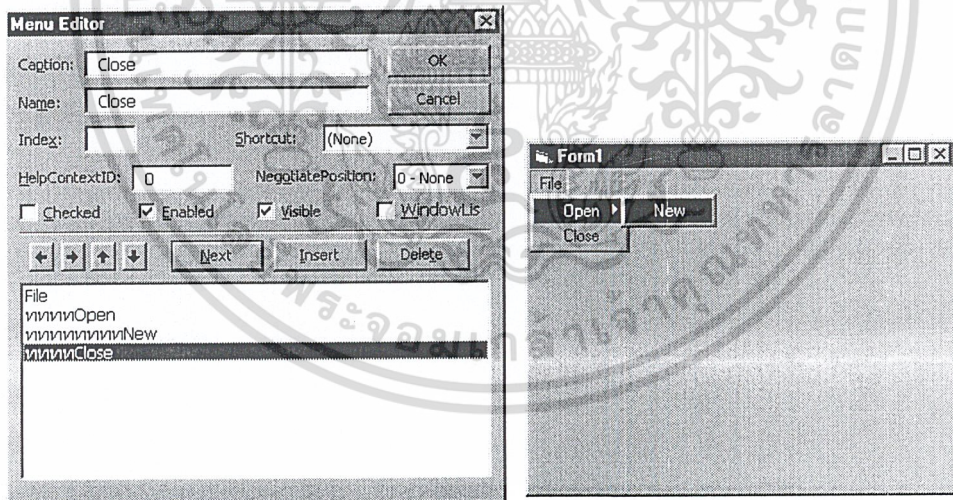
### การสร้างเมนูย่อย (Sub Menu)

ใน VB6 เราสามารถสร้างเมนูย่อยในเมนูหลักได้ถึง 5 ลำดับชั้น โดยมีขั้นตอน ดังนี้

1. Click mouse ปุ่ม  ในการลดลำดับชั้นของเมนูย่อยลง 1 ชั้น โดยเมื่อ Click แล้ว แถวของเมนูจะมี ..... ขึ้นมาแสดงว่า เมื่อนั้นกลายเป็นเมนูย่อย ของเมนูที่อยู่ก่อน

2. หรือ Click mouse ปุ่ม  เพื่อเพิ่มลำดับชั้นของเมนูย่อยขึ้น 1 ชั้น โดยเมื่อ Click แล้วจะลดย่อหน้าลงไป

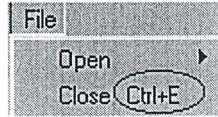
ตัวอย่างต่อไปนี้เป็นกรสร้างเมนูที่มีเมนูย่อย



ภาพที่ 3.9 แสดงการกำหนด Menu Editor และผลลัพธ์

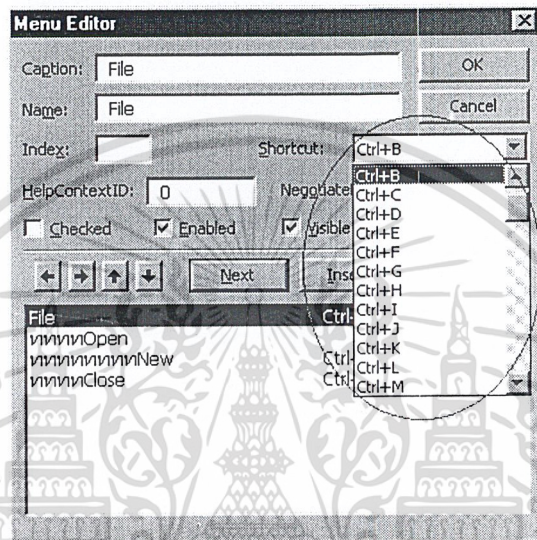
### การกำหนดคีย์ลัดให้เมนู

ShortCut จะเป็นการกำหนดปุ่มคีย์ลัดที่เมื่อกดแล้วจะเป็นการเรียกคำสั่งนั้นขึ้นมาทันที ปุ่มที่กำหนดให้เป็นคีย์ลัดของเมนูจะรวมปุ่มควบคุม เช่น <CTRL>, <ALT> กับปุ่มอื่น ๆ เช่น A, B เอกสาร และเมื่อมีการกำหนดคีย์ลัดให้กับคำสั่งในเมนูแล้ว คีย์ลัดจะปรากฏที่ด้านขวาของคำสั่งดังกล่าว ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



ภาพที่ 3.10 แสดงคีย์ลัดที่เกิดขึ้น

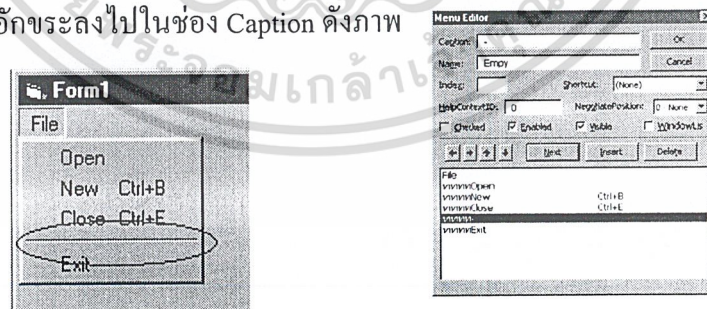
เราสามารถกำหนดคีย์ลัด ให้กับคำสั่งในเมนูได้โดยเลือกจากช่อง ShortCut ดังภาพ



ภาพที่ 3.11 แสดงการตั้งค่าคีย์ลัดให้เมนู

### การใช้เส้นคั่น (Separator Bar)

ในบางครั้งเราต้องการใช้เส้นคั่นแยกรายการคำสั่งในเมนูออกเป็นกลุ่มตามต้องการ ดังภาพ โดยการเติมใส่อักขระลงไปในช่อง Caption ดังภาพ



ภาพที่ 3.12 แสดงการสร้าง Menu แบบมีเส้นคั่น

### การเขียนคำสั่งในรูปแบบต่าง ๆ

ในการเขียนคำสั่งโปรแกรมบางครั้งเราอาจจะต้องการให้มีคำอธิบายสำหรับคำสั่งโปรแกรมในส่วนนั้น หรือต้องการแบ่งคำสั่งในบรรทัดหนึ่งออกเป็นหลายบรรทัด เนื่องจากคำสั่งในบรรทัดนั้นยาวเกินไป เราจะอธิบายการใส่คำสั่งในแต่ละรูปแบบ ดังหัวข้อต่อไปนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### การแบ่งคำสั่งบรรทัดหนึ่งออกเป็นหลายบรรทัด

เนื่องจากคำสั่งต่าง ๆ ใน VB6 นั้น เราต้องใส่คำสั่งต่าง ๆ ให้จบภายใน 1 บรรทัด แต่ในบางครั้งคำสั่งนั้นยาวมาก ทำให้ไม่สะดวกในการทำงาน จึงมีวิธีแตกบรรทัดหนึ่งออกเป็นหลายบรรทัดได้ ด้วยการใส่ตัวอักขระช่องว่าง ตามด้วยอักขระ \_ ดังตัวอย่าง เช่น

MsgBox “Your Name is Tharin Sittitummacharee”, VbOkOnly, “Name”

เราสามารถแยกเป็นหลาย ๆ บรรทัดได้ดังคำสั่งข้างล่าง

MegBox “โครงการ การใช้เครื่องอ่านแถบแม่เหล็กเพื่อควบคุมการใช้ อินเทอร์เน็ต”, \_  
VbOkOnly, “Name”

จากตัวอย่างข้างต้น บรรทัดที่ 1 และ 2 จะเป็นคำสั่งเดียวกัน แต่แยกจากกันเป็น 2 บรรทัด ด้วยอักขระช่องว่างแล้วตามด้วยอักขระ \_

### การรวมคำสั่งหลายบรรทัดในบรรทัดเดียว

เราสามารถรวมคำสั่งมากกว่าหนึ่งคำสั่งให้อยู่ในบรรทัดเดียวกัน โดยใช้เครื่องหมาย : ในการแยกแต่ละคำสั่งออกจากกัน ตัวอย่างเช่น

$$X = 1$$

$$Y = 2$$

$$Z = 3$$

เราสามารถรวมเป็นคำสั่งเดียวกันได้ดังคำสั่งข้างล่าง

$$X = 1 : Y = 2 : Z = X + Y$$

จากตัวอย่างข้างต้น จะเป็นการรวมคำสั่ง  $X = 1$ ,  $Y = 2$  และ  $Z = X + Y$  ให้อยู่ในบรรทัดเดียวกัน โดยเว้นแต่ละคำสั่งด้วยเครื่องหมาย :

### การใส่คำอธิบายสำหรับโปรแกรม (Comment)

เราสามารถเพิ่มคำอธิบายสำหรับคำสั่งโปรแกรมในหน้าต่าง Code Editor ได้เพื่อช่วยให้ทำความเข้าใจกับคำสั่งที่เราพิมพ์ได้ง่ายขึ้น โดยเราสามารถใส่คำอธิบายได้ด้วยการใช้อักขระ ' ซึ่ง

เป็นอักขระที่บอกให้ VB6 ไม่ต้องสนใจข้อความที่ตามหลังอักขระนั้น ตัวอย่างการเพิ่มคำอธิบายในคำสั่งโปรแกรม

' This is Comment 1

X = 1 ' This is Comment 2

จากตัวอย่างข้างต้น บรรทัดที่ 1 จะเป็นคำอธิบายโปรแกรมของเรา ส่วนบรรทัดที่ 2 จะมีคำสั่ง X = 1 และตามด้วยคำอธิบายโปรแกรม

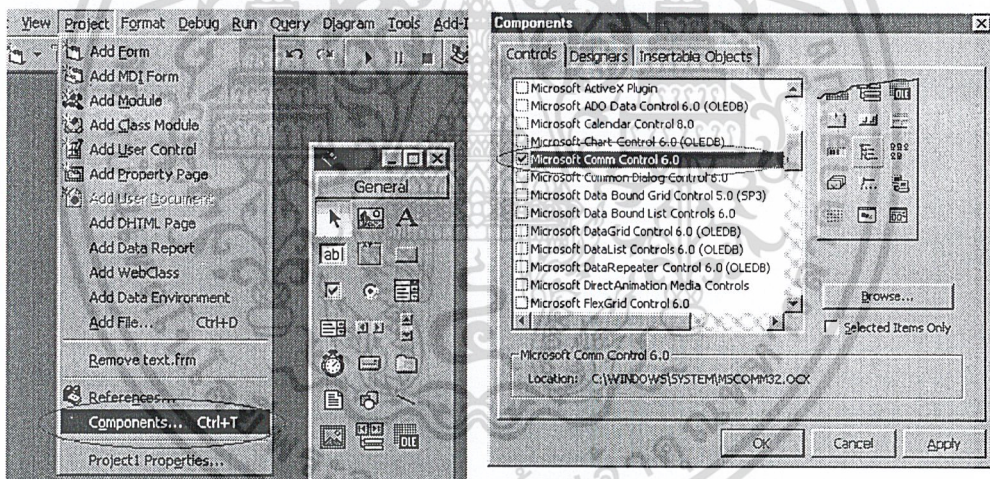
นอกจาก Control ที่ได้อธิบายไว้ดังกล่าวแล้วยังมีอีก 2 ชนิดที่จะกล่าวถึงต่อไปนี้ก็คือ

MSComm กับ Winsock

### MSComm

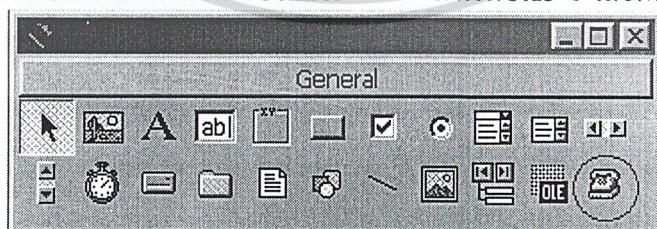
ในการทำโครงงานชิ้นนี้เราจะต้องรับค่าที่ได้จากการรูดการ์ดผ่านมาทาง Port RS232 ซึ่งในการเขียนโปรแกรมด้วย VB นั้นเราจะใช้ Control ที่ชื่อว่า MSComm เป็นตัวช่วยในการรับค่าผ่านทาง SerialPort ของ Computer

โดยปกติแล้ว MSComm จะไม่ปรากฏอยู่ที่ ToolBar จะต้องทำการเพิ่มเข้ามาดังภาพ



ภาพที่ 3.13 ก แสดงการเพิ่ม Control

ภาพที่ 3.13 ข เลือกตามภาพ



ภาพที่ 3.13 ค MSComm ที่เพิ่มเข้ามา

การเขียนโปรแกรมติดต่อกับ Serial Port สามารถทำได้ 2 วิธี คือ

#### 1. การติดต่อแบบอินเทอร์เน็ต

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ขบวนการอินเทอร์เน็ต อุปกรณ์รอบข้างเกือบทุกชิ้นจะต้องปฏิบัติงานอยู่เพื่อส่งสัญญาณไปให้แก่ซีพียูเสมอ ถ้าอุปกรณ์นั้นพร้อมที่จะรับส่ง เราจะเขียนโปรแกรมอินเทอร์เน็ต โดยเมื่อที่ข้อมูลเข้ามาก็จะทำให้มี CommEvent กับ OnComm Event

## 2. การติดต่อแบบ โพลลิง

ในระบบพีซี การโพลมีบ้างที่ใช้การส่งผ่านข้อมูลระหว่าง Terminal กับ CPU กรณีข้อมูลเป็นประเภทไบท์ที่ส่งจากคีย์บอร์ด โดยวิธีการนี้จะตรวจสอบ คีย์บอร์ดว่ามีข้อมูลส่งมาหรือเปล่า โดยจะตรวจสอบตลอดเวลา การทำงานกับข้อมูลที่ได้รับเข้ามาจะตรวจสอบด้วยความเร็วที่สูงกว่า อัตราความเร็วข้อมูลที่ส่งเข้ามาทาง คีย์บอร์ด การที่ CPU ส่งสัญญาณออกไปตรวจสอบพบว่า มีข้อมูลที่ส่งเข้ามา เรียกว่า "Wet Poll" ซึ่งจะเสียช่วงเวลา 90 เปอร์เซ็นต์ คาบเวลาที่เสียไปนั้น เราเลี่ยงไปใช้เทคนิค การโพลแบบ "Round Robin" แทน แต่ใน VB เราจะใช้การตรวจสอบข้อมูลที่มาจาก Serial Port ตลอด โดยจะใช้ Control Timer เข้ามาช่วยในการเขียนโปรแกรมซึ่งสามารถตรวจสอบได้ถึงระดับ 1 มิลลิวินาที

ในตัวคอนโทรล MSComm มี Event ที่ใช้เพียง Event เดียวเท่านั้นเอง ก็คือ OnComm Event ซึ่งจะใช้ในการติดต่อแบบอินเทอร์เน็ต การเขียน โปรแกรมติดต่อ Serial Port แบบธรรมดา จะใช้ comEvent เพียง comEvReceive, comEvSend ถ้าเป็นการติดต่อสื่อสารแบบ โมเด็มจะใช้หลายตัวในการตรวจสอบสัญญาณ

### องค์ประกอบในการใช้ MSComm การตั้งค่าติดต่อกับพอร์ต

ComPort คือ เราต้องกำหนดหมายเลข Port ที่ใช้ต่อ RS-232 (Com1, Com2) รายละเอียดดูในเมนูด้านซ้าย Serial Port Detail

Setting คือ เราต้องกำหนดอัตรา Baud, Parity, Data (จำนวนบิต), Stop ตัวอย่าง 1200, n, 8, 1 เป็นต้น

HandShaking คือ เราจะกำหนดได้ 4 แบบ

1. comNone
2. comXonXoff
3. comRTS
4. comTRSXonXoff

### การใช้ Buffer ในการรับส่งข้อมูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

InBuffersize	คือ การกำหนด Buffer ในการรับข้อมูลเข้ามา
OutBuffersize	คือ การกำหนด Buffer ในการส่งข้อมูลออกไป
Rthreshold	คือ การที่เรากำหนดการเกิด Event-driven ในการรับข้อมูลเข้ามา
Sthreshold	คือ การที่เรากำหนดการเกิด Event-driven ในการรับข้อมูลออกไป
Inputlen	คือ จำนวนของข้อมูลที่ไปอ่านใน Buffer รับข้อมูล
EOFEnable	คือ การที่บอกว่าสิ้นสุดของไฟล์(EOF) End of File

### ด้านฮาร์ดแวร์

ParityReplace	คือ ค่าของคาเลกเตอร์ที่จะแทนในเมื่อเกิด Parity Error
NullDiscard	คือ การกำหนดให้รับหรือไม่รับ NULL CHARACTER
RTSEnable	คือ ทำให้มีสัญญาณ RTS (Request To Send)
DTSEnable	คือ ทำให้มีสัญญาณ DTR(Data Terminal Ready)

การกำหนดคุณสมบัติของ MSComm Control ให้สามารถติดต่อกับพอร์ตได้

1. Property ชื่อ CommPort คือ เลือกคอมพอร์ตที่เราจะต่อใช้งาน

```
MSComm1.CommPort=1
```

ในที่นี้เลือกจะใช้ Com1 อยู่ที่ด้านหลังเครื่องคอมพิวเตอร์

2. Property ชื่อ Settings คือ การตั้งค่าของการรับส่งข้อมูล ซึ่งจะต้องรู้ด้วยว่าอัตราบอด ของอุปกรณ์ที่จะติดต่อด้วยเป็นเท่าไร โดยมีรายละเอียดการใส่ต่างๆค่าดังนี้

```
MSComm1.Setting
```

```
= "Baud(อัตราการรับส่งข้อมูล),Parity(ถ้าไม่ใช้ใส่ N,จำนวนบิตข้อมูล,บิตสตอป"
```

ตัวอย่าง

```
MSComm1.Settings="1200,N,8,1"
```

3. Property ชื่อ InputLen คือ กำหนดขนาดขณะที่มีข้อมูลเข้ามาให้ไปอ่านข้อมูลทั้งหมดที่อยู่ในบัฟเฟอร์

```
MSComm1.InputLen=1
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4. Property ชื่อ PortOpen คือ จะเปิดให้พอร์ตใช้งานหรือไม่ ถ้าเปิด =True ถ้าปิด =False

```
MSComm1.PortOpen=True
```

5. Property ชื่อ Rthreshold คือ ทำให้เกิดการกระตุ้นด้วย Event-driven เมื่อมีข้อมูลในบัฟเฟอร์รับข้อมูล(Comport)มันทำให้เกิดCommEvent ใน OnComm Event

```
MSComm1.Rthreshold =1
```

จากรายละเอียดที่กล่าวมา เราจะมาเขียนใน โพรซีเจอร์ VB ซึ่งจะไว้ที่ Sub Form\_Load()

```
Private Sub Form_Load()
    MSComm1.Settings="1200,N,8,1"
    MSComm1.CommPort=1
    MSComm1.InputLen=1
    MSComm1.PortOpen=True
    MSComm1.Rthreshold =1
End Sub
```

### วิธีของการรับส่งข้อมูลจาก Serial Port

จากวิธีเขียน โค้ดด้านบนเป็นการกำหนดค่าเริ่มต้นให้กับคอมพอร์ตและเปิดใช้การรับและส่งของพอร์ต RS-232 ดังนั้นก็สามารถจะรับและส่งข้อมูลทางพอร์ตได้ โดยใช้ Property ดังนี้

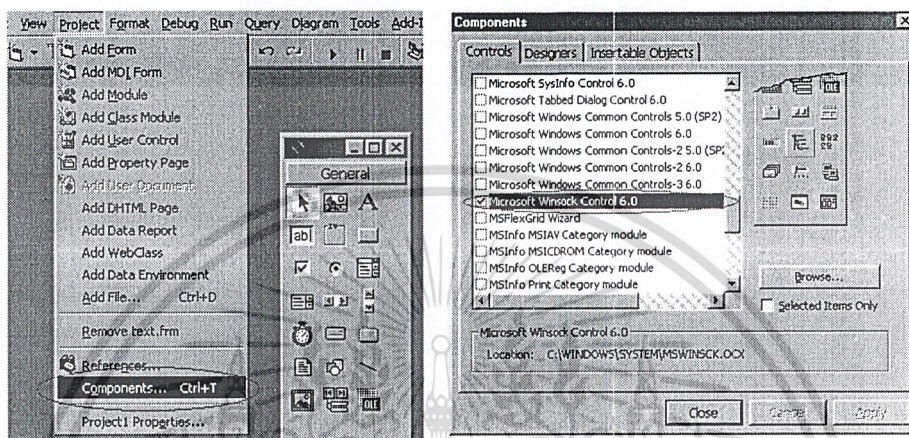
Output = ซึ่งจะเป็นการส่งข้อมูลไปที่พอร์ต

Input = เป็นส่วนของการรับข้อมูลจากพอร์ต แต่ในส่วนนี้จะต้องนำคำสั่งไปเขียนที่ Event Property OnComm จะอยู่ใน Sub MSComm\_OnComm ซึ่ง จะอ่านข้อมูลเข้ามาจากทางพอร์ต RS232 นั้นเอง

## Winsock

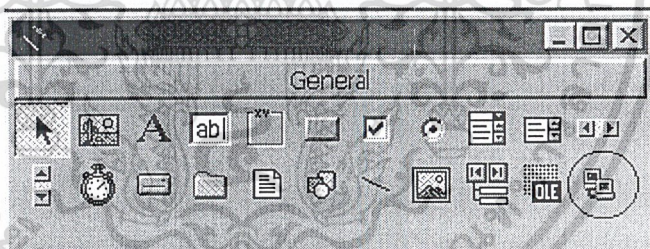
นอกจากเราจะเขียนโปรแกรมเพื่อการรับค่าจาก SerialPort แล้วเรายังต้องทำการส่งค่าภายในระบบ Network ด้วยดังนั้นการเขียนโปรแกรมด้วย VB ก็จะมี Control ที่ชื่อว่า Winsock ไว้ใช้ในการรับส่งค่าภายใน Network

ในการเขียนโปรแกรมต้องทำการเพิ่ม Winsock เข้ามาก่อน โดยมีขั้นตอนดังภาพ



ภาพที่ 3.14 ก แสดงการเพิ่ม Control

ภาพที่ 3.14 ข เลือกตามภาพ



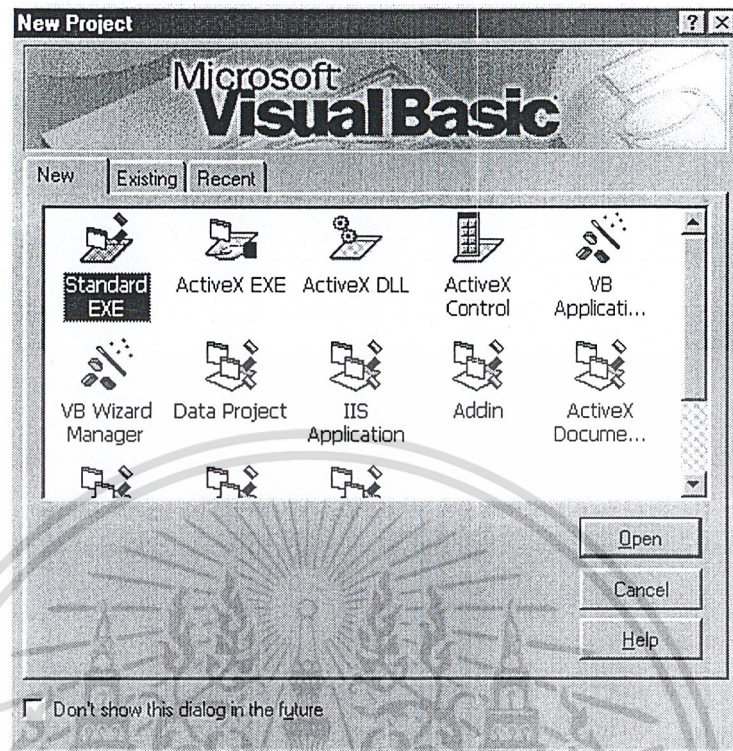
ภาพที่ 3.14 ค Winsock ที่เพิ่มเข้ามา

### ขั้นตอนการเขียนโปรแกรม

ในการเขียนครั้งนี้ เราจะทดลองเขียนให้ติดต่อผ่าน โปรโตคอล TCP/IP โดยลักษณะโปรแกรมจะคล้ายกับโปรแกรมสื่อสารทาง internet (Chat) โดยจะแบ่งโปรแกรมเป็น 2 ส่วน ส่วนแรกเป็นส่วนของการส่งข้อมูลโดยจะทำงานบนเครื่องหลักของเครือข่าย ส่วนที่สองจะทำงานเป็นตัวรับข้อมูลจะทำงานบนเครื่องเครือข่าย โดยที่เราจะประยุกต์การรับข้อความนี้มาเป็นการแปรคำสั่งแทนซึ่งจะอธิบายต่อไป

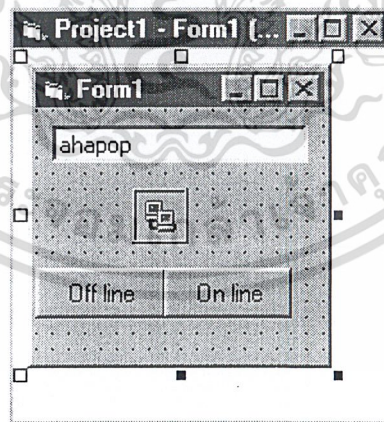
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เริ่มต้นด้วยการเปิดโปรเจกต์ใหม่ดังภาพ



ภาพที่ 3.15 การสร้าง Project ใหม่

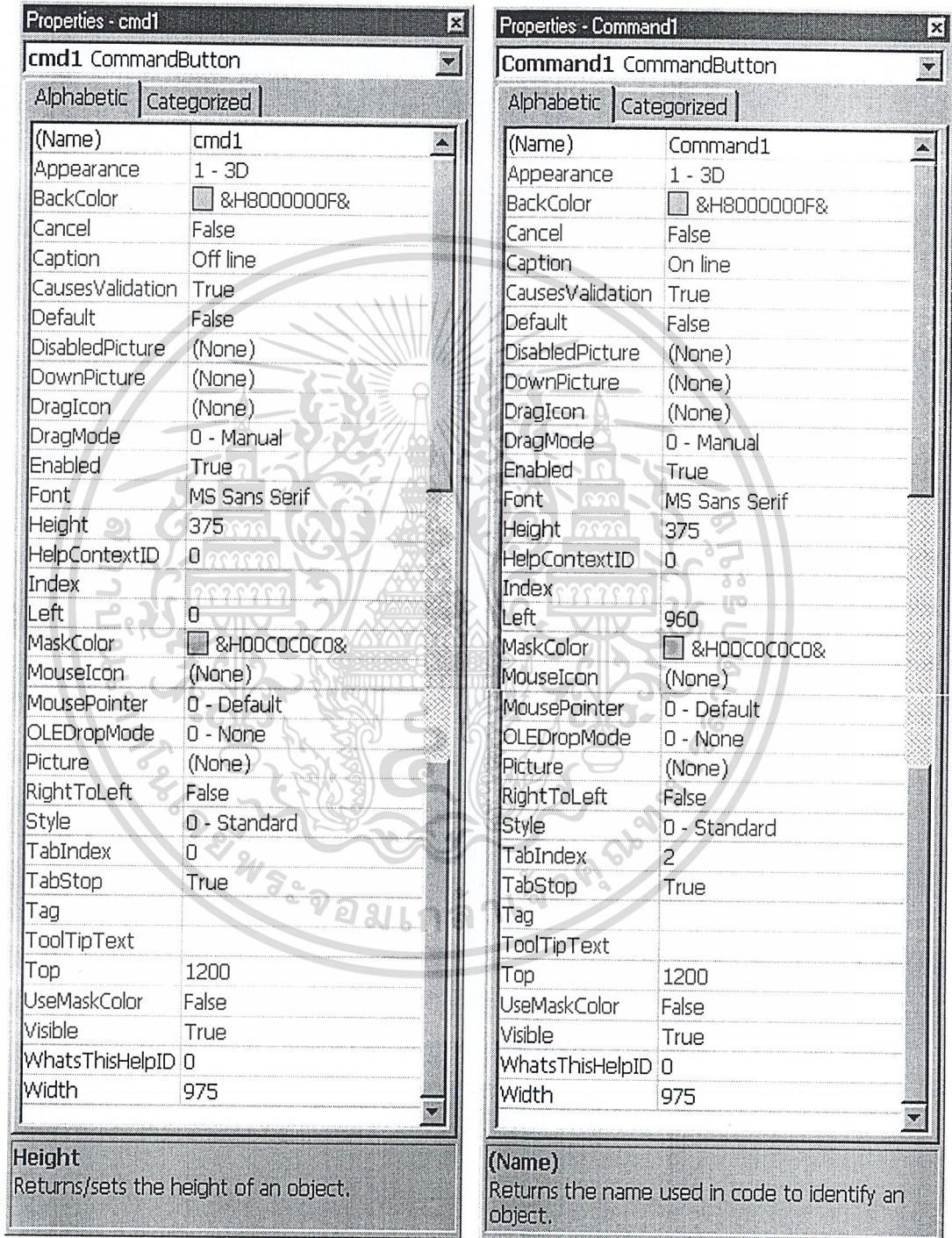
หลังจากนั้นเราก็ทำการวาง Control ต่างไปดังภาพ



ภาพที่ 3.16 รูปแบบโปรแกรมตัวอย่าง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากนั้นก็กำหนดค่าต่างๆดังต่อไปนี้



ภาพที่ 3.17 ก แสดงการ Set ค่าต่างๆให้ Control

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

**Properties - Form1**

Form1 Form

(Name)	Form1
Appearance	1 - 3D
AutoRedraw	False
BackColor	&H8000000F&
BorderStyle	2 - Sizable
Caption	Form1
ClipControls	True
ControlBox	True
DrawMode	13 - Copy Pen
DrawStyle	0 - Solid
DrawWidth	1
Enabled	True
FillColor	&H00000000&
FillStyle	1 - Transparent
Font	MS Sans Serif
FontTransparent	True
ForeColor	&H80000012&
HasDC	True
Height	2295
HelpContextID	0
Icon	(Icon)
KeyPreview	False
Left	4140
LinkMode	0 - None
LinkTopic	Form1
MaxButton	True
MDIChild	False
MinButton	True
MouseIcon	(None)
MousePointer	0 - Default
Moveable	True
NegotiateMenus	True
OLEDropMode	0 - None
Palette	(None)
PaletteMode	0 - Halftone
Picture	(None)
RightToLeft	False
ScaleHeight	1890
ScaleLeft	0
ScaleMode	1 - Twip
ScaleTop	0
ScaleWidth	2175
ShowInTaskbar	True
StartUpPosition	0 - Manual
Tag	
Top	6045
Visible	True
WhatsThisButton	False
WhatsThisHelp	False
Width	2295
WindowState	0 - Normal

**Height**  
Returns/sets the height of an object.

**Properties - txt1**

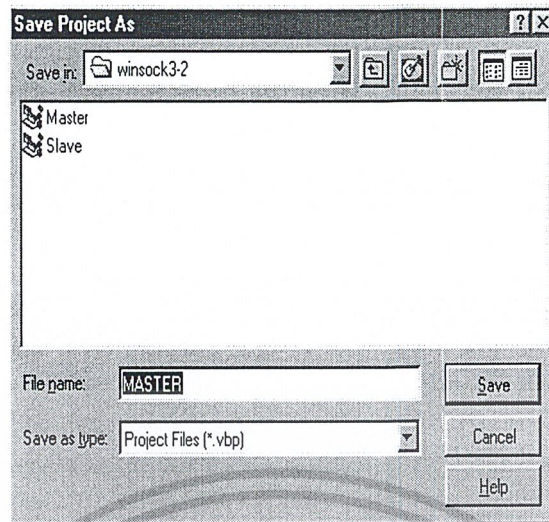
txt1 TextBox

(Name)	txt1
Alignment	0 - Left Justify
Appearance	1 - 3D
BackColor	&H80000005&
BorderStyle	1 - Fixed Single
CausesValidation	True
DataField	
DataFormat	General
DataMember	
DataSource	
DragIcon	(None)
DragMode	0 - Manual
Enabled	True
Font	MS Sans Serif
ForeColor	&H80000008&
Height	285
HelpContextID	0
HideSelection	True
Index	
Left	120
LinkItem	
LinkMode	0 - None
LinkTimeout	50
LinkTopic	
Locked	False
MaxLength	0
MouseIcon	(None)
MousePointer	0 - Default
MultiLine	False
OLEDragMode	0 - Manual
OLEDropMode	0 - None
PasswordChar	
RightToLeft	False
ScrollBars	0 - None
TabIndex	1
TabStop	True
Tag	
Text	ahapop
ToolTipText	
Top	120
Visible	True
WhatsThisHelpID	0
Width	1935

**CausesValidation**  
Returns/sets whether validation occurs on the control which lost focus.

### ภาพที่ 3.17 ข แสดงการ Set ค่าต่างๆให้ Control

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



ภาพที่ 3.17 ค แสดงการ Save ค่า Control

เมื่อทำการ set Properties ของ Control ทุกตัวแล้ว ก็ทำการเขียน Code ดังนี้

```

Sub cmd1_Click()
    wsk1.RemotePort = 1001
    wsk1.RemoteHost = txt1.Text
    wsk1.Connect
    cmd1.Enabled = False
    Command1.Enabled = True
End Sub

-----

Private Sub Command1_Click()

    wsk1.Close

    cmd1.Enabled = True

    Command1.Enabled = False

End Sub

-----

Private Sub Form_Load()

    Command1.Enabled = False

End Sub

```

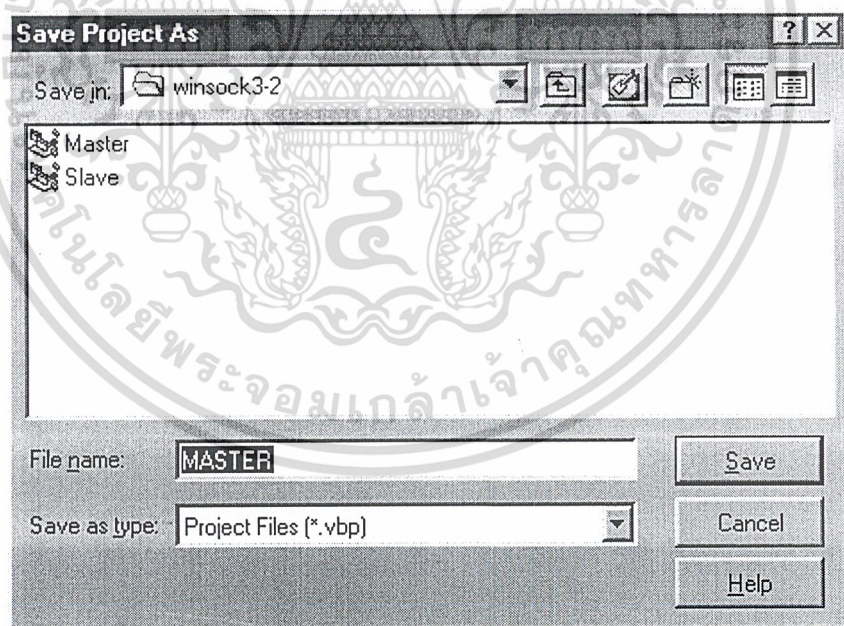
```

-----
Private Sub wsk1_Connect()
    wsk1.SendData "ok"
End Sub
-----

Private Sub wsk1_DataArrival(ByVal bytesTotal As Long)
    Dim datain As String
    wsk1.GetData datain
    txt1.Text = datain
End Sub

```

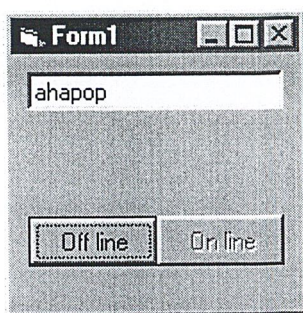
เสร็จแล้วทำการตั้งชื่อและ save โดยเราจะตั้งชื่อว่า Master



ภาพที่ 3.18 แสดงการ Save Project

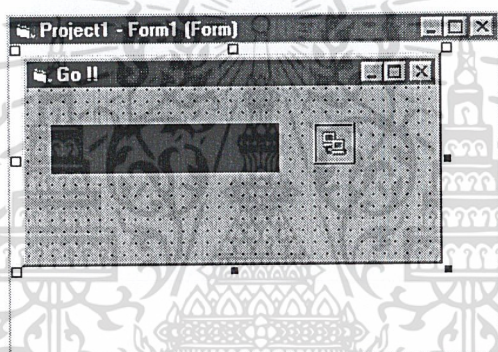
เมื่อ Save แล้วทดลอง Run ดู จะต้องไม่มี Error เกิดขึ้น ถ้ามีให้ทำการแก้ไขตามข้อผิดพลาด ถ้าไม่มีอะไรผิดพลาดจะต้องได้ดังภาพที่ 3.19

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



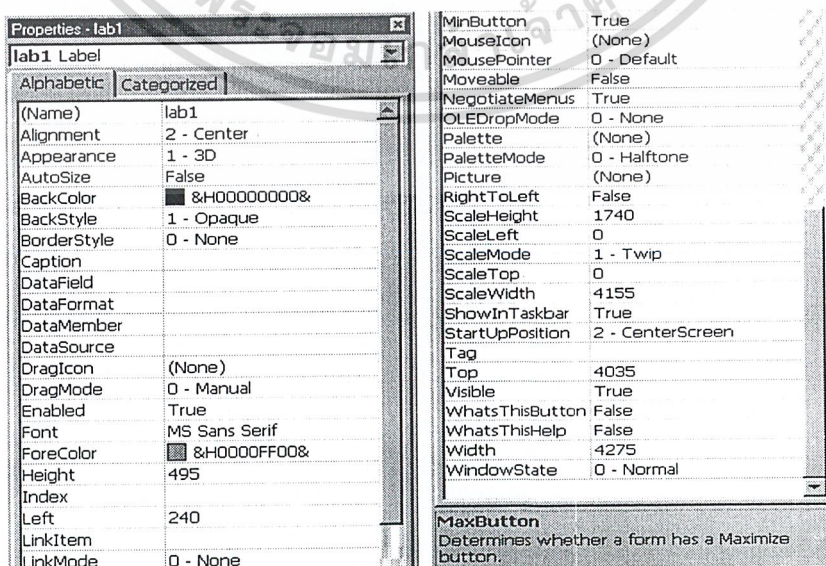
ภาพที่ 3.19 แสดงโปรแกรมที่จะสร้าง

เมื่อทำส่วนของ Master เสร็จแล้วก็ลงมือทำส่วนของการรับข้อมูลโดยเราจะใช้ชื่อโปรแกรมว่า Slave  
เปิดโปรเจ็คใหม่ และวาง Control ดังภาพ 3.20

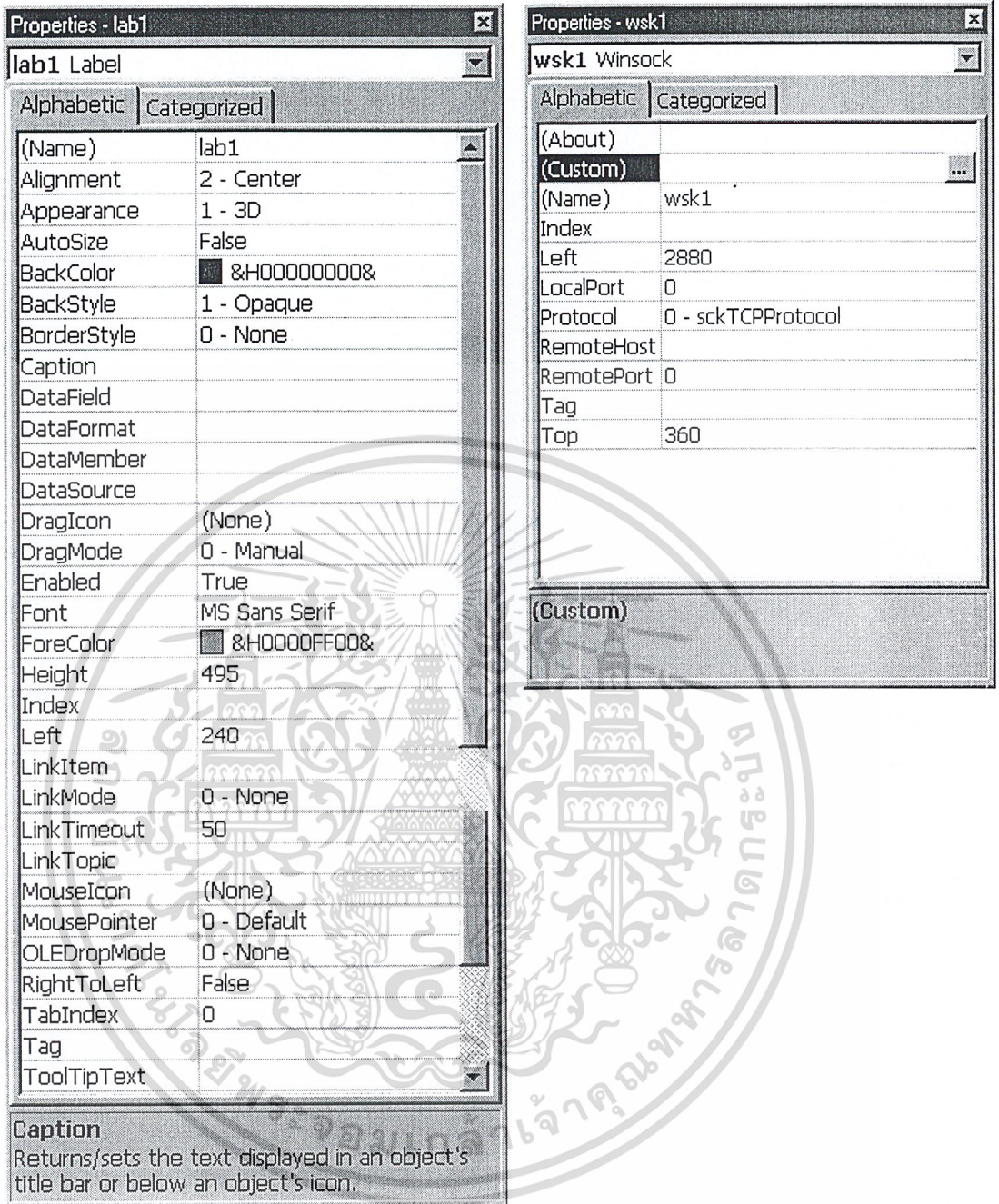


ภาพที่ 3.20 การวาง Control

ทำการ Set Properties ให้กับ Control ทุกตัวตามนี้



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับภาพที่ 3.21 การแสดงการ set properties ให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



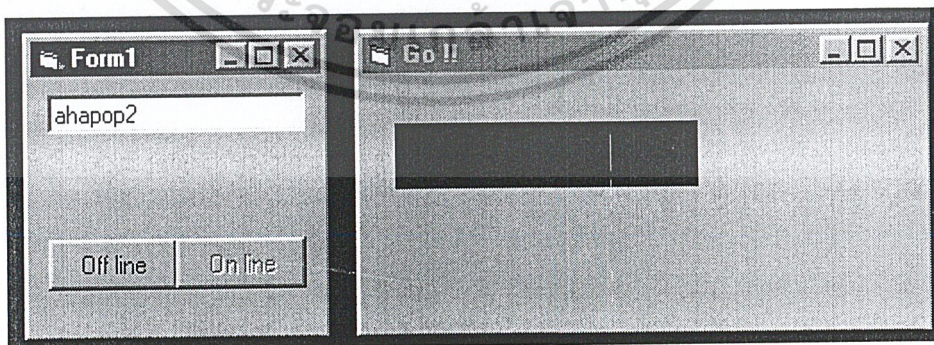
ภาพที่ 3.21 ข แสดงการ set properties

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

แล้วทำการพิมพ์ code ดังนี้

```
Private Sub Form_Load()
    wsk1.LocalPort = 1001
    wsk1.Listen
End Sub
-----
Private Sub wsk1_Close()
    Dim a As Integer
    Wsk1.Close
    Lab1.Caption = "On Line"
    wsk1.Listen
End Sub
-----
Private Sub wsk1_ConnectionRequest(ByVal requestID As long)
    If wsk1.State <> sckClosed Then wsk1.Close
    wsk1.Accept requestID
    lab1.Caption = "TimeOut"
End Sub
```

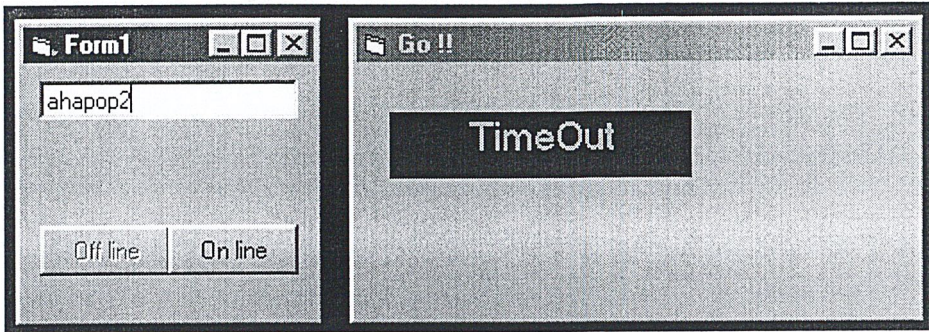
ทำการ save และ run ดู สมมติว่าไม่มี Error เกิดขึ้น ให้ run ทั้ง Master และ Slave พร้อมกันดังภาพ



ภาพที่ 3.22 แสดงการ Run โปรแกรมทั้งสอง

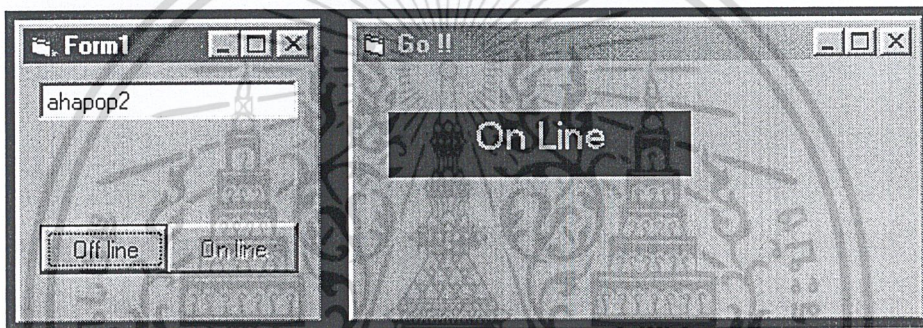
ในช่อง textbox ให้ใส่ชื่อของ computer ตัวที่ต้องการจะติดต่อด้วย แล้วกด Off line เครื่องที่ฝั่ง Slave ต้องได้รับข้อมูล โดยจะแสดงผลเมื่อได้รับข้อมูลดังภาพ 3.23

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์การเชิงานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



ภาพที่ 3.23 แสดงผลการ Run โปรแกรม

จะเห็นว่าฝั่ง Slave เปลี่ยนไปเป็นคำว่า TimeOut ลองกด On line ที่ Master จะได้ผลดังภาพ



ภาพที่ 3.24 แสดงผลการ Run โปรแกรม

จากวิธีการง่ายๆดังนี้เราจะไปประยุกต์เพื่อควบคุมคอมพิวเตอร์ให้ทำงานอย่างที่เราต้องการในการทำโครงงานครั้งนี้ เราจะต้องทำการติดต่อเครื่อง Computer ที่ต่อใช้งานอยู่ในรูปแบบของ Network ดังนั้นเราจึงควรมีความรู้เกี่ยวกับ Network บ้างซึ่งจะอธิบายในบทต่อไป

## บทที่ 4

### การ Set Network ขั้นพื้นฐาน

ในการเขียนโปรแกรมเพื่อการติดต่อในวงแลน บางครั้งก็เกิดปัญหาขึ้นมาโดยไม่ทราบว่าจะเกิดจาก ตัวโปรแกรมเองหรือเกิดจากระบบ network ที่เราทดลองอยู่ ดังนั้นจำเป็นอย่างยิ่งที่เราต้องทำความเข้าใจในการ set network อย่างง่าย เพื่อให้แต่ละเครื่องสามารถ แลกเปลี่ยนข้อมูลหรือสามารถ ทำการ share drive ในการทำงานได้ และเป็นส่วนสำคัญใน โปรเจ็คชิ้นนี้ด้วยเพราะเราต้องการควบคุมเครื่องที่อยู่ในวงแลนให้ได้ดังนั้นสิ่งที่ต้องการคือให้ทุกเครื่องสามารถมองเห็นกันหมด หรือหมายถึงสามารถ share drive กันได้

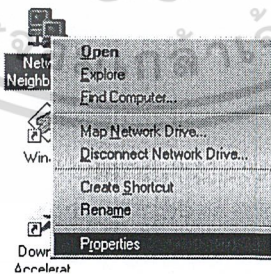
ขั้นตอนดังต่อไปนี้ที่ผู้เขียนได้ใช้ประสบการณ์ของผู้เขียนเท่านั้น โดยศึกษาเฉพาะที่ ต้องการใช้เท่านั้น ดังนั้นถ้าวิธีบางอย่างที่ผู้อ่านสามารถกระทำได้ด้วยวิธีที่ดีกว่า ก็สามารถที่จะกระทำได้

ถ้าผู้อ่านมีเครื่อง computer ที่มี network card อยู่และสามารถใช้งานได้ บน desktop จะมี icon ดังภาพ 4.1



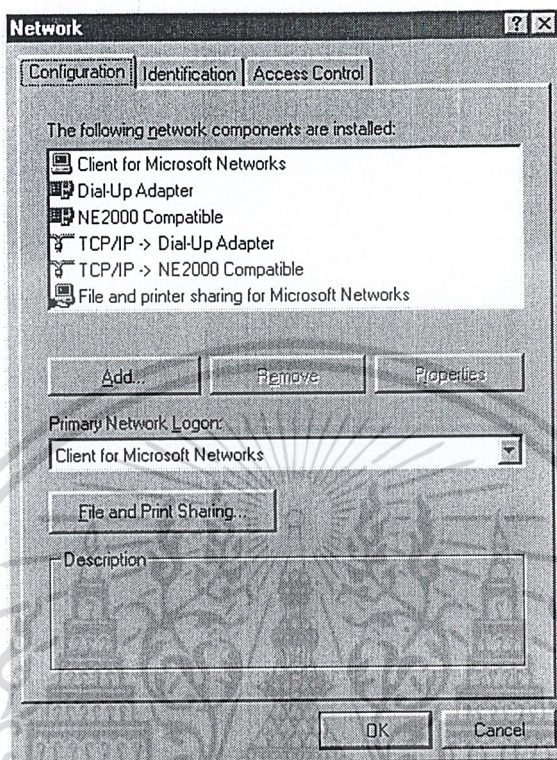
ภาพที่ 4.1 แสดง ICON

ให้คลิกเมาท์ปุ่มขวามือแล้วเลือกที่ properties ดังภาพที่ 4.2



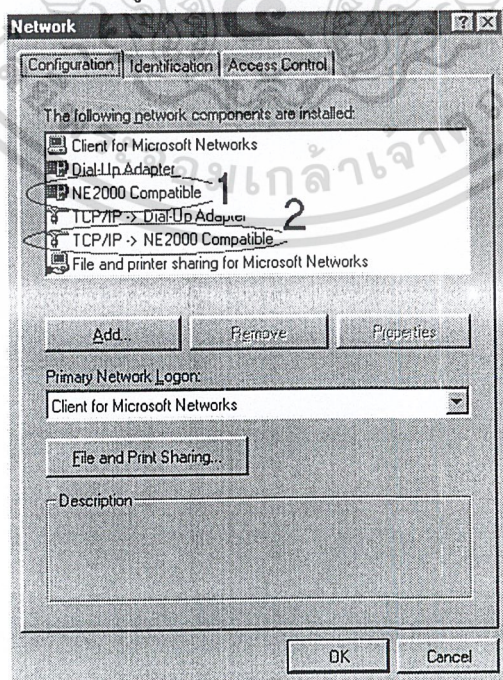
ภาพที่ 4.2 Properties

เมื่อทำการเลือกที่ propertiesแล้วจะมีหน้าต่างแสดงรายละเอียดดังภาพที่ 4.3



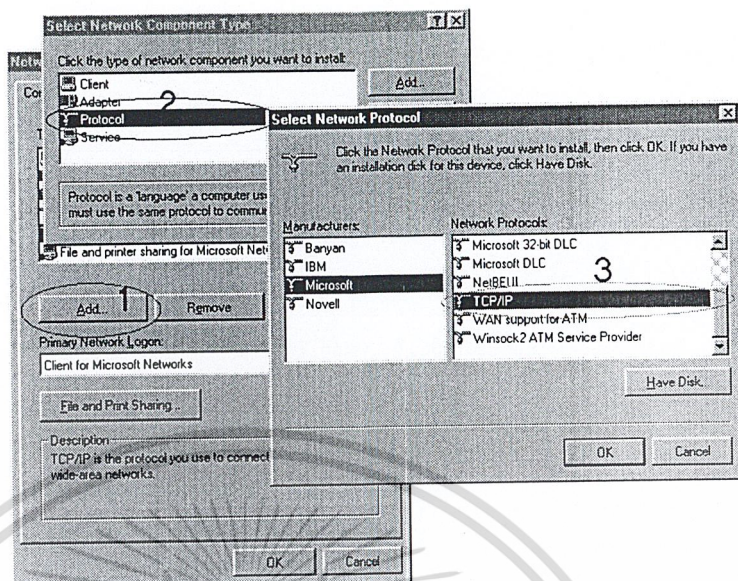
ภาพที่ 4.3 แสดงหลังเลือก properties

ถ้า network card ได้รับการติดตั้งที่ถูกต้องจะมีดังในที่วงด้วย



เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์ของงานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ภาพที่ 4.4 แสดงรายการ Portocol  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

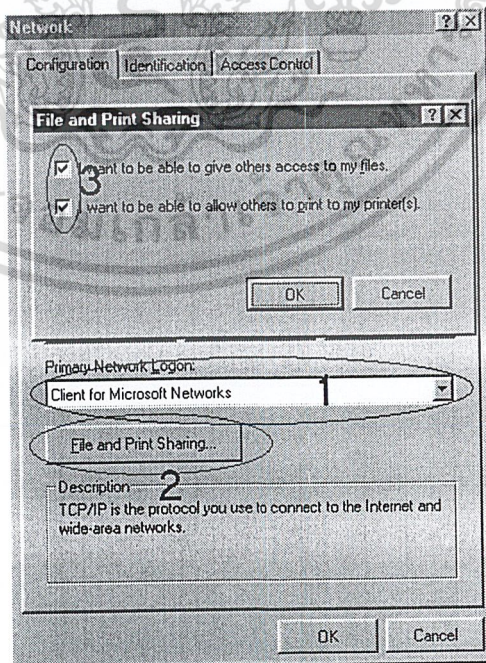
แต่ถ้าไม่มี protocol TCP/IP (ของ network card ) ต้องทำการ Add เข้ามาดังภาพ 4.5



ภาพที่ 4.5 แสดงการเพิ่ม Protocol

โดยอันดับแรกให้ คลิกที่ Add (1) จะปรากฏ หน้าต่างอีกบาน ให้ดับเบิลคลิกที่ Protocol (2) และสุดท้ายจะมีหน้าต่างปรากฏอีกบาน ให้เลือกที่ Microsoft TCP/IP (3) คลิก OK และทำตามขั้นตอนที่ปรากฏขึ้นมาไปจนเสร็จ โดยส่วนใหญ่จะถามหา OS win98 (ถ้าเครื่องนั้นใช้ windows98 อยู่) และ Restart

ส่วนที่สองคือการ set ให้share file และ printer ได้ คลิกดังภาพ 4.6



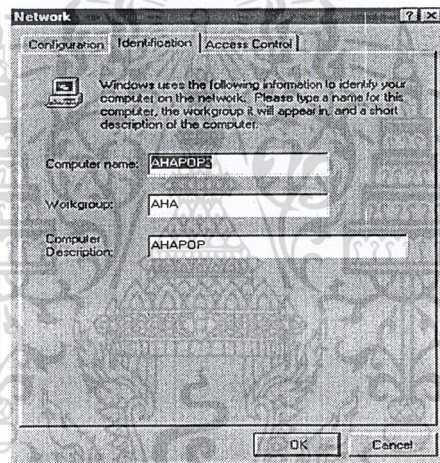
ภาพที่ 4.6 แสดงการ Share Drive

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

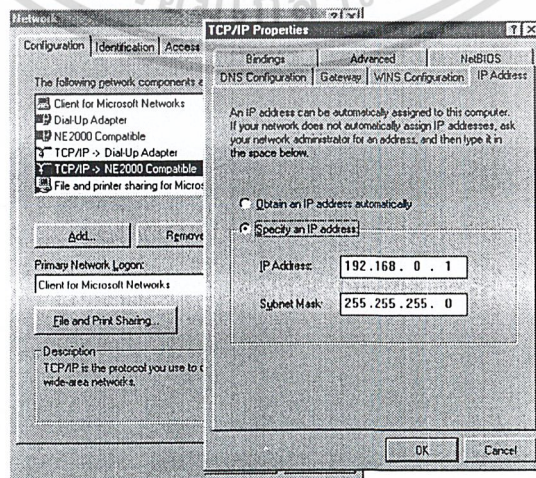
ให้ทำตามภาพได้เลย โดยทำตามอันดับที่ระบุเอาไว้ แล้วคลิก OK และทำไปตามขั้นตอนจนเสร็จ โดยส่วนใหญ่จะถามหา OS win98 (ถ้าเครื่องนั้นใช้ windows98 อยู่) และ Restart

ขั้นตอนต่อไปคือการกำหนดชื่อให้เครื่อง computer ซึ่งในโปรเจกต์นี้ถือว่าเป็นเรื่องที่สำคัญมากๆ เพราะในการควบคุมที่ออกแบบไว้จะยึดเอา Computer Name เป็นหลัก

เริ่มจากเข้าไปที่ properties ของ network และเลือกหน้าต่างดังภาพในช่อง Computer name ให้ใส่ชื่อของ Computer ซึ่งสามารถตั้งได้เองเป็นอะไรก็ได้ ในช่อง Workgroup ใส่ชื่ออะไรก็ได้ แต่ถ้ามีเครื่องอื่นอีกและต้องการให้อยู่ใน Workgroup เดียว กันต้องใส่ Workgroup ให้เหมือนกัน และขอแนะนำว่าควรให้เหมือนกันเพื่อความสะดวกในการ Set นอกนั้นไม่ต้องสนใจให้คลิกเลือกที่ Tap Configuration เมื่อคลิกกลับมาที่ Tap Configuration แล้วให้เลือกที่ TCP/IP ที่ตามด้วยชื่อของ Network card ที่อยู่ในเครื่อง แล้วคลิก properties จะได้หน้าต่างดังภาพที่ 4.7



ภาพที่ 4.7 แสดงการตั้งชื่อ ComputerName



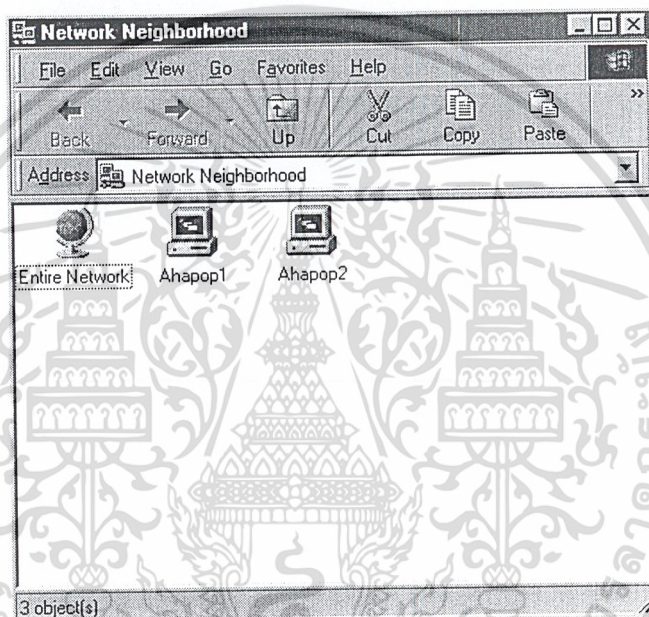
ภาพที่ 4.8 แสดงการใส่ค่า IP ADDRESS

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตรงค่า IP Address จะมี Crass ต่างๆกัน โดยจะให้เครื่องหลักลงท้ายด้วย 1 และให้เครื่องที่เหลือเรียงเลขกันไป

ส่วนค่า Subnet Mask ถ้าอยู่ในวง network เดียวกันต้องเหมือนกันเสมอ เสร็จแล้วก็ คลิก OK เครื่องอาจจะถามหา OS ของเครื่องนั้นๆ ก็ให้จัดการตามขั้นตอนไปเรื่อยๆ และ Restart ใหม่ ถ้าทุกอย่างปกติ หลัง Boot เสร็จจะมี dialog Box ให้ใส่ password ให้ใส่ชื่อ และ password ลงไป ซึ่งแล้วแต่จะตั้ง

เมื่อใส่รหัสเรียบร้อยแล้ว ลองเข้าไปที่ Network Neighborhood ท่านจะพบว่า มี icon ภาพ Computer ที่มีชื่อเครื่องของท่านและเครื่องอื่นอยู่ด้วย



ภาพที่ 4.9 แสดงเครื่องลูกข่ายอื่นๆ ที่อยู่ในเครือข่ายเดียวกัน

เป็นอันเสร็จการ set ขึ้นพื้นฐาน ซึ่งก็เพียงพอที่จะเอาไว้ Run Project ได้

## บทที่ 5

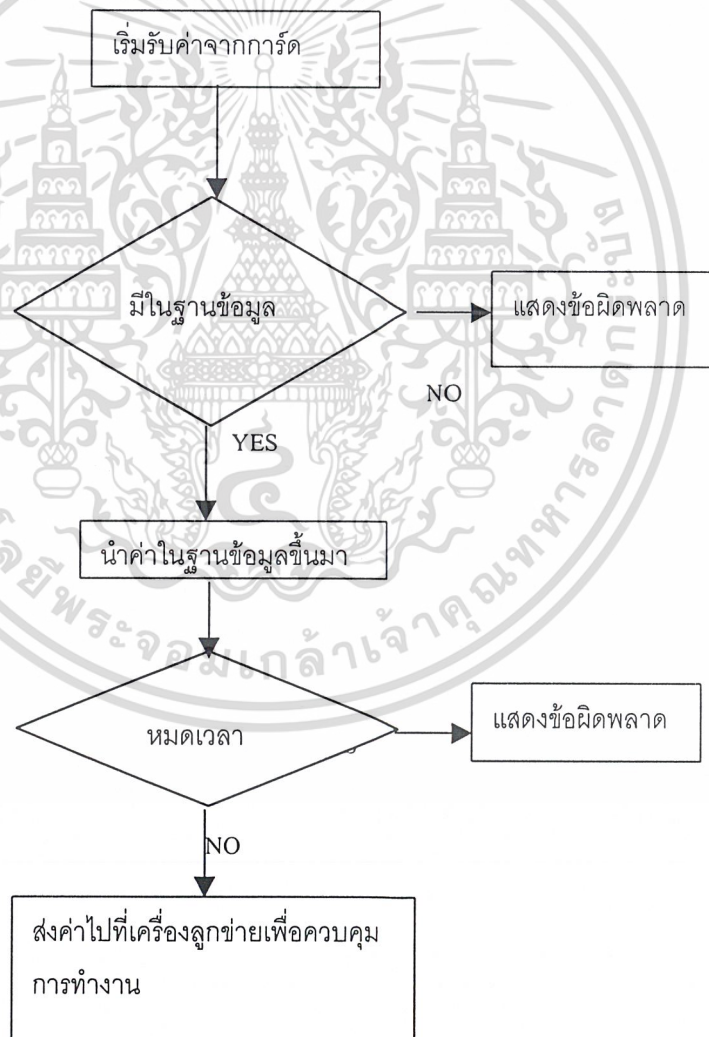
### หลักการการทำงานของโปรแกรมในแต่ละส่วน

การทำโครงการในส่วนของ Software จะแบ่งออกเป็น 2 ส่วนคือส่วนที่ทำงานที่ตัว Computer หลักกับส่วนที่ทำงานบน Computer ที่อยู่บน Network

Software ที่ทำงานบน Computer หลัก จะแบ่งออกเป็น 4 ส่วนใหญ่ ๆ คือ

1. ส่วนของ INPUT ซึ่งรับค่าจาก Hardware ภายนอก
2. ส่วนของ ฐานข้อมูล
3. ส่วนการควบคุมหลัก
4. ส่วนของการควบคุมภายนอกโดยผ่าน Network

โดยมี Flowchart ดังนี้

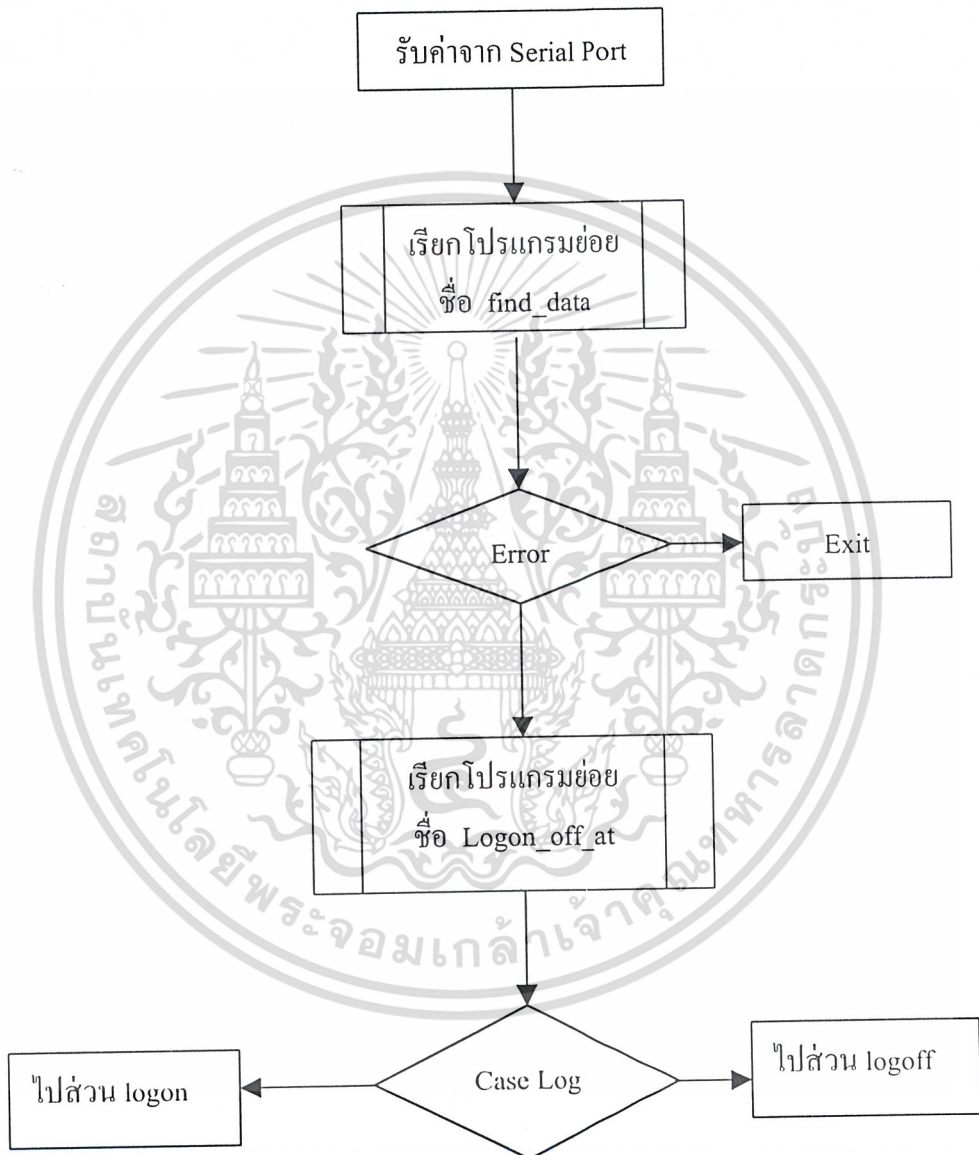


ภาพที่ 5.1 แสดง Flowchart ของหลักการการทำงานโดยรวมของ Software

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จาก Flowchart ที่แสดงเป็นการแสดงในส่วนหลักๆเท่านั้น ซึ่งในส่วนของรายละเอียดจะกล่าวถึงในส่วนต่อไป

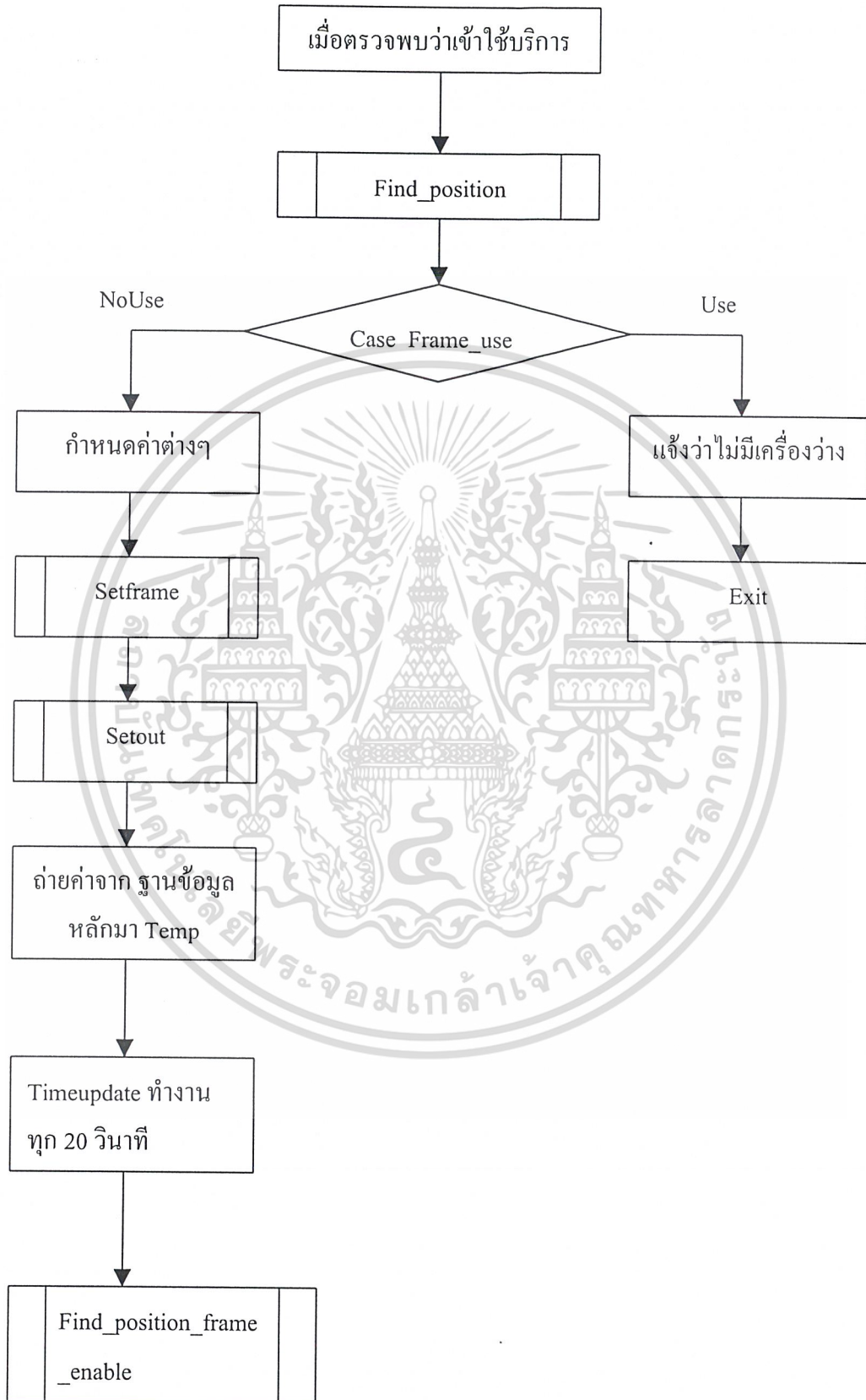
ในส่วนนี้แสดง Flowchart ของการตรวจสอบว่าเป็นการรูดการ์ดเข้ามาใช้บริการหรือออกจากบริการ



ภาพที่ 5.2 แสดง Flowchart การตรวจสอบการเข้า-ออกจากบริการ

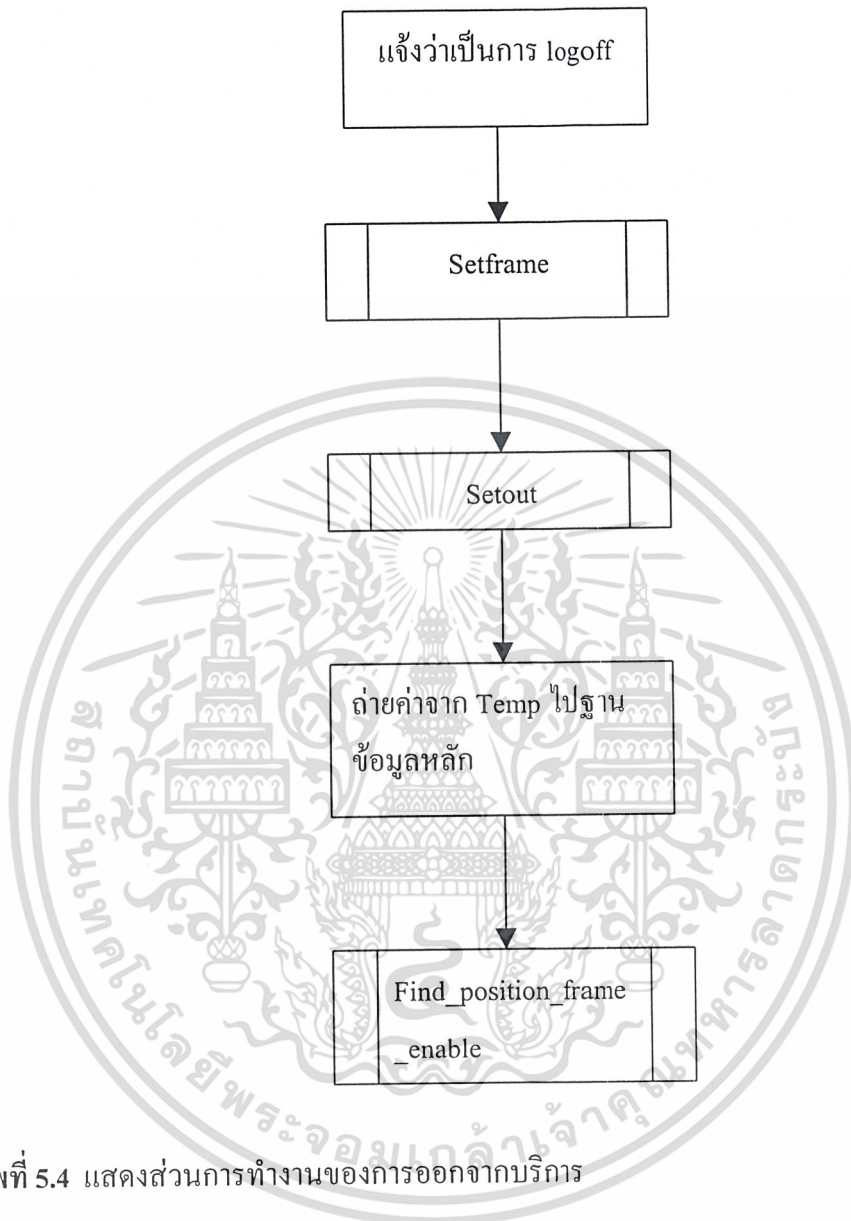
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เมื่อเราตรวจพบว่าเป็นการเข้าใช้บริการ โปรแกรมจะทำงานดังนี้



ภาพที่ 5.3 แสดงส่วนการทำงานของกรเข้าใช้บริการนั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เมื่อเราพบว่าการรูดการ์ดนั้นเป็นการออกจากการใช้บริการ โปรแกรมจะทำงานดังนี้



ภาพที่ 5.4 แสดงส่วนการทำงานของกรออกจาบริการ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## อธิบายโปรแกรมย่อยต่างๆ

Sub find\_data()

โปรแกรมย่อยในส่วนนี้จะทำการตรวจสอบค่าที่ได้จาก Port นำมาเปรียบเทียบกับค่าที่อยู่ในฐานข้อมูลหลักโดยถ้ามีค่าในฐานข้อมูลจะส่งค่าต่างๆที่สำคัญออกไปเช่น ชื่อผู้ใช้บริการหรือจำนวนเงินต่างๆ แต่ที่สำคัญคือตัวแปรที่ชื่อว่า data\_base จะเก็บค่า “ noerror “ ถ้าไม่มีสิ่งผิดปกติ และจะเก็บค่า “ error “ ถ้ามีสิ่งผิดปกติ

Sub logon\_off()

ในส่วนนี้จะตรวจหาว่ามีการ LogOn หรือ Logoff ที่ตำแหน่งที่เท่าไรและตรวจสอบว่ากำลังจะเข้าหรือออกจากบริการ

Sub find\_position()

เป็นการหาตำแหน่งเครื่องที่ว่างจากการใช้งานอยู่เพื่อให้ผู้ใช้บริการได้เข้าไปใช้ ถ้าตรวจพบเครื่องที่ว่างก็จะแจ้งออกมาเป็นตำแหน่งเครื่อง

Sub set\_frame()

เป็นส่วนที่ทำการแจ้งไปที่แผงควบคุมหลักว่าตำแหน่งไหนกำลังทำอะไรอยู่

Sub sutout()

จะเป็นส่วนที่ทำการกำหนดชื่อเครื่องลูกข่ายที่จะทำการควบคุมโดยจะส่งการไปที่ Winsock อีกทีหนึ่ง

ส่วนรายละเอียดปลีกย่อยสามารถดูได้จาก Soucecode โปรแกรมในภาคผนวก

## ส่วนต่างๆของโปรแกรมหลัก

เนื่องจากโปรแกรมนี้ แบ่งออกเป็น 2 ส่วน

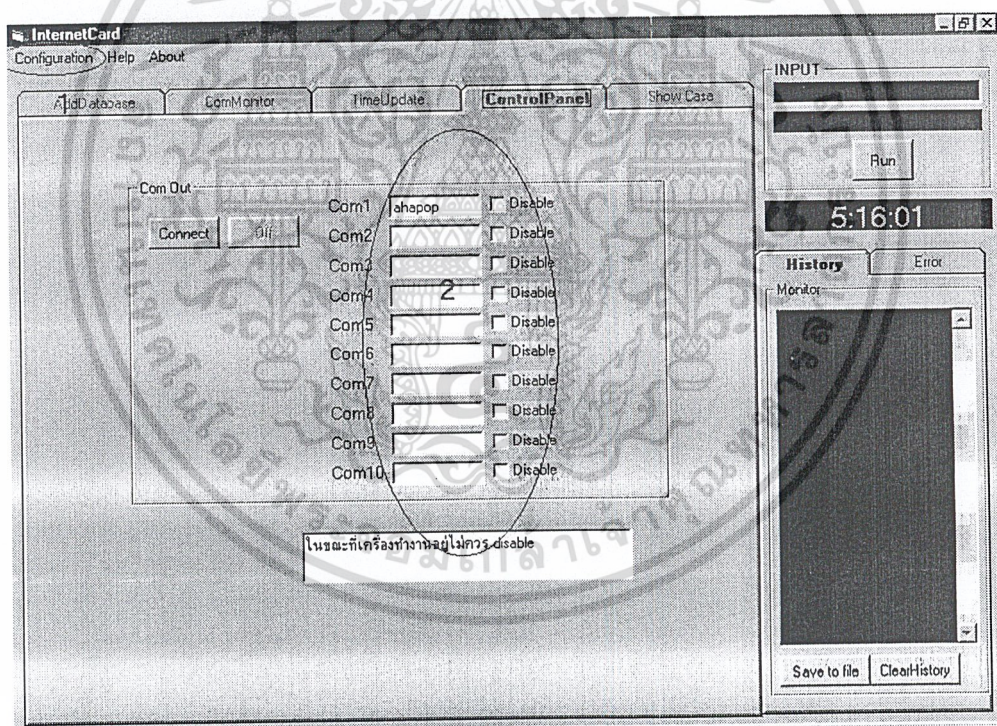
ส่วนที่แรกจะเป็นส่วนหลักของการควบคุมเครื่องลูกข่ายโดยจะติดตั้งโปรแกรมดังกล่าวไว้ที่เครื่องหลักเราขอเรียกว่าเครื่อง Server และเรียกโปรแกรมว่า Main Program

ส่วนที่สองเป็นส่วนของการรับคำสั่งเพื่อการควบคุม โดยจะติดตั้งไว้ที่เครื่องลูกข่าย เราขอเรียกว่า Client และเรียกโปรแกรมว่า Slave

### Main Program

เป็นส่วนที่รับค่าจากการ์ดเข้ามาแล้วตรวจสอบสถานะของการ์ด แล้วส่งงานไปยังตัวลูกข่าย โดยสามารถใช้งานได้ดังนี้

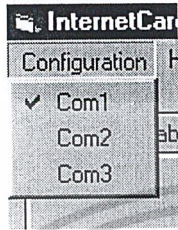
1 เมื่อเปิดโปรแกรมขึ้นมาจะพบกับหน้าต่างดังภาพ



ภาพที่ 5.5 ภาพแสดงหน้าต่างหลักโปรแกรม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

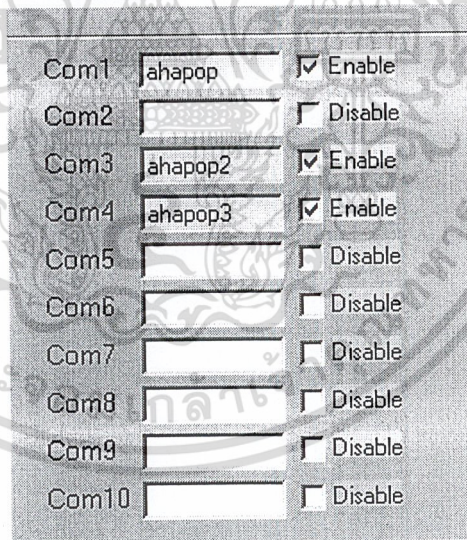
สิ่งที่จะต้องให้ความสำคัญอันดับแรกคือการ Set Port ที่ใช้รับข้อมูลจากเครื่องรูดการ์ด คือ Menu Configuration ที่วงกลมหมายเลข 1 กระทำได้โดยเมื่อคลิกที่ Menu ดังกล่าวแล้วจะมี Port ให้เลือกดังภาพ



ภาพที่ 5.6 แสดง Menu สำหรับการ SetComPort

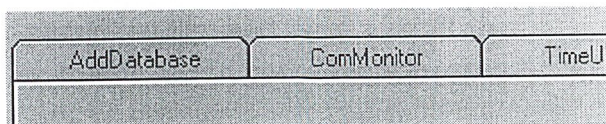
ก็ให้เลือกตามที่ต้องการที่ต่อเครื่องรูดการ์ด

ส่วนที่สองคือการ Set ให้เครื่องลูกข่ายที่ต้องการให้ทำงานภายใต้การควบคุม วิธีใช้คือให้ใส่ชื่อ ComputerName ของเครื่องลูกข่ายที่ต้องการใช้ แล้วคลิก ที่เช็คบ็อกซ์ ดังภาพ 5.7



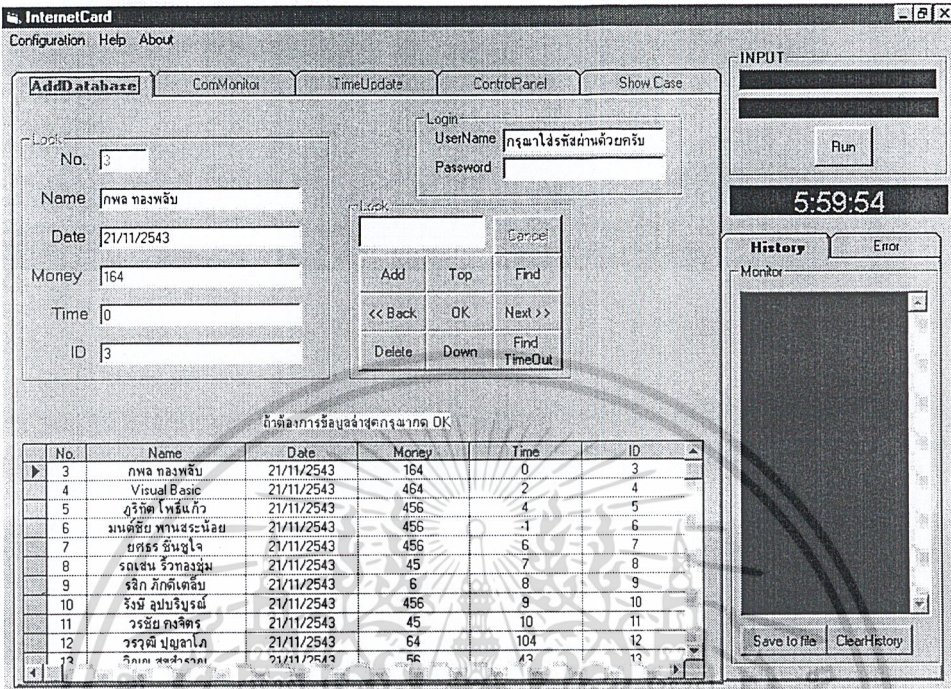
ภาพ 5.7 แสดงการเลือกเครื่องลูกข่ายที่ต้องการใช้งาน

เมื่อทำการ Set ค่าตามที่ต้องการแล้วก็ Click ที่ Tap DataBase ดังภาพ 5.8



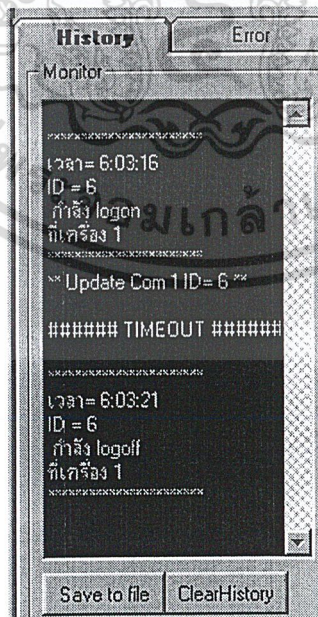
เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์ไว้เพื่อใช้ในการเรียนเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จะพบกับหน้าที่เป็นการรับค่าหรือเปลี่ยนแปลงค่าในฐานข้อมูล ซึ่งการจะเข้าไปแก้ไขได้ ต้องมี Password (ในโปรแกรมนี้ตั้งไว้ว่า ahapop )



ภาพที่ 5.9 แสดงโปรแกรมในส่วนของฐานข้อมูลลูกค้า

สมมติว่ามีการรูดการ์ดเข้ามาเราจะทราบได้โดยการดูช่องของ History ดังภาพ 5.10

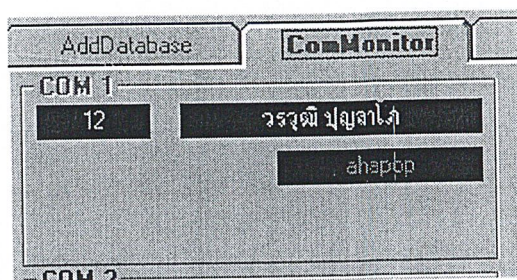


ภาพที่ 5.10 History

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

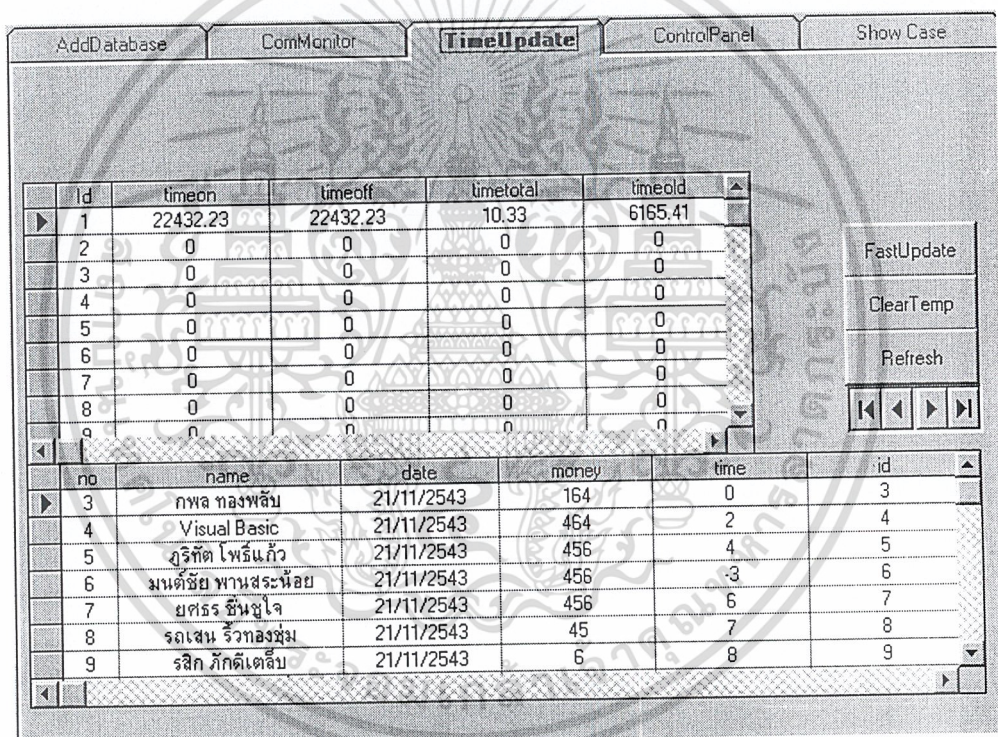


ในส่วน ของ Tap Commonitor จะแสดงว่าเครื่องลูกข่ายใดถูกใช้งานอยู่ ดังภาพ



ภาพที่ 5.12 ComMonitor

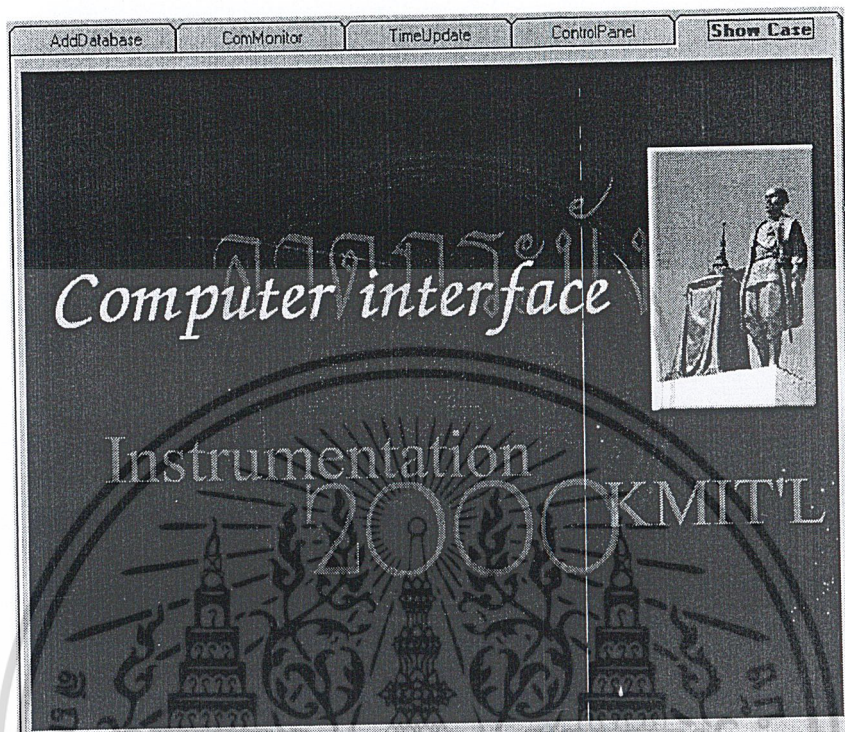
ส่วน Tap TimeUpdate จะเป็นการแสดงเวลาและค่าของเงินของผู้ใช้บริการที่คงเหลืออยู่ ดังภาพ



ภาพที่ 5.13 แสดงในส่วนของการลดจำนวนเงินตามเวลา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ส่วนหน้าสุดท้าย เป็นการแสดง Logo ของโปรแกรม



ภาพที่ 5.14 Logo ของโปรแกรม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## บทที่ 6

### ผลการทดลอง

ในการทดลองได้ทำการต่อComputerในรูปแบบของNetworkจำนวน 3 เครื่อง โดย Set ค่าต่างๆดังนี้

เครื่องที่ 1 (ให้เป็นเครื่องหลัก)

Computer Name : KMITL1  
IP Address : 192.168.0.1  
SubNet Mask : 255.255.255.0

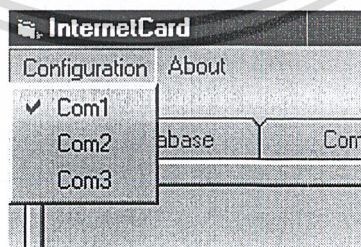
เครื่องที่ 2 (ให้เป็นเครื่องลูกข่าย)

Computer Name : KMITL2  
IP Address : 192.168.0.2  
SubNet Mask : 255.255.255.0

เครื่องที่ 3 (ให้เป็นเครื่องลูกข่าย)

Computer Name : KMITL3  
IP Address : 192.168.0.3  
SubNet Mask : 255.255.255.0

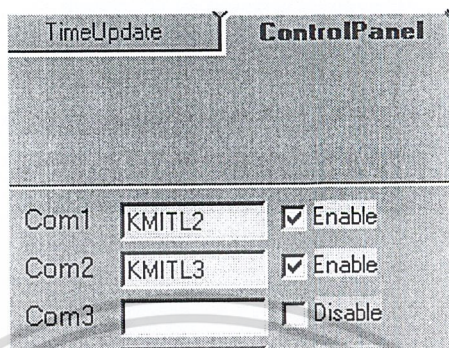
ให้ 1 เครื่องทำการ Run โปรแกรม Main แล้วทำการ Set ค่าต่างๆดังภาพโดยการเริ่ม Set ที่ Menu Configuration แล้วเลือก SerialPort ที่เครื่องรูดการ์ดได้ต่ออยู่



ภาพที่ 6.1 แสดงการ Set ค่า SerialPort

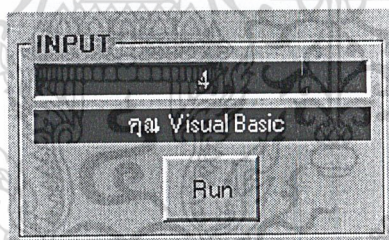
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ขั้นตอนต่อไปก็ทำการเลือกตำแหน่งของเครื่องที่ต้องการ โดยเราสามารถเลือกได้ 10 ตำแหน่ง แต่ในการทดลองนี้เรามีเครื่องที่ควบคุมได้เพียง 2 เครื่องเท่านั้นจึงขอเลือกที่ 2 ตำแหน่งแรกดังภาพ



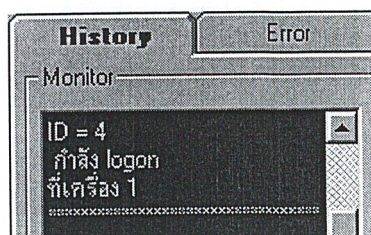
ภาพที่ 6.2 แสดงการเลือกตำแหน่งของเครื่องที่ Tap ControlPanel

เมื่อทำการ Set ค่าต่างๆเรียบร้อยแล้ว และเครื่องลูกข่ายได้ทำการ Run โปรแกรม Slave เรียบร้อยแล้ว ก็ทำการรูดการ์ดที่เครื่องรูดการ์ด ซึ่งค่าที่ได้จะแสดงที่ชุดแสดงผล INPUT โดยจะแสดงรหัส และชื่อบัตร ดังภาพ



ภาพที่ 6.3 แสดงค่าที่รับมาจากเครื่องรูดการ์ด

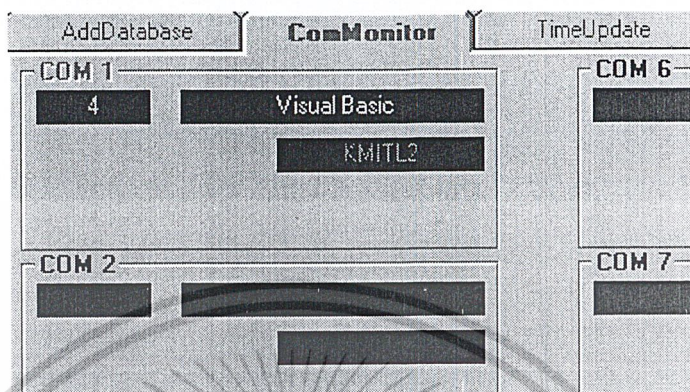
ถ้าบัตรยังไม่หมดอายุ โปรแกรมจะทำการแจ้งไปที่ History โดยบอกรหัสบัตร,การใช้บริการ, ตำแหน่งของเครื่องที่ใช้ ดังภาพ



ภาพที่ 6.4 แสดงการแจ้งที่ History

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

และโปรแกรมก็จะทำการ Set ค่าในส่วนของ Tap ComMonitor เพื่อแสดงผลการควบคุม โดยจะระบุ รหัสผู้ใช้, ชื่อบัตร และชื่อ Computer ที่ถูกควบคุมอยู่ดังภาพที่ 6.5



ภาพที่ 6.5 แสดง Tap ComMonitor

หลังจากนั้นเครื่องลูกข่ายที่ถูกระบุให้ทำงานจะทำการคลายล็อก ทำให้เครื่องสามารถใช้งานได้ตามปกติ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## บทที่ 7

# สรุปโครงการและข้อเสนอแนะในการพัฒนา

ส่วนของการใช้งานด้าน Hardware ไม่มีปัญหาในการทดลองนั้นเราจะใช้การ์ดของโรงภาพยนตร์ ที่เป็นแถบแม่เหล็กมาทำการทดลองแทนการไปจ้างทำบัตรจริงๆ เนื่องจากเราสามารถประยุกต์เพื่อลดต้นทุนในการทำ Project ลงได้ โดยรวมแล้วด้าน Hardware ทำงานได้ดี

ส่วนของการใช้งานด้าน Software ยังมีปัญหาอยู่บ้างเช่น การบังคับให้เครื่องในเครือข่ายหยุดทำงานนั้นสามารถกระทำได้เฉพาะเวลาที่เครื่องลูกข่ายทำงานที่สภาวะปกติเท่านั้น ถ้าเครื่องลูกข่ายยังอยู่ในสภาวะ Load Program อยู่ทำให้เกิด Error ได้

เนื่องจากในปัจจุบันระบบปฏิบัติการได้มีเพิ่มมากขึ้น ดังนั้นการเขียนโปรแกรมในการควบคุมเครื่องลูกข่ายหรือ จะทำการ Run โปรแกรม Main ก็ดี ต้องคำนึงถึงภาษาที่นำมาเขียนด้วยว่าเหมาะสมกับระบบปฏิบัติการหรือไม่ ซึ่งในการเขียน โปรแกรมนั้นควรจะสามารถที่จะรองรับระบบปฏิบัติการต่างๆ ได้เป็นอย่างดี เพราะจะได้ไม่เป็นปัญหาในการควบคุม

ปัญหาที่สองที่อยากจะเสนอแนะคือ การพัฒนาโปรแกรมเพื่อให้มีประสิทธิภาพด้านความเร็วและความแน่นอนในการทำงานควรจะมีการแยกโปรแกรมย่อยออกมาเป็นส่วนๆ โดยทำให้อยู่ในรูปของ File นามสกุล DLL เพราะเราสามารถจะแก้ไขโปรแกรมได้ง่ายกว่าการที่จะต้องไปแก้ไปในโปรแกรมหลักทั้งหมด

และทั้งหมดนี้เป็นความคิดเห็นทั้งหมดโดยอาศัยประสบการณ์ส่วนตัวในที่ได้พบมาและได้จะการทำโครงการนี้ขึ้น ซึ่งถ้าท่านผู้อ่านมีแนวคิดที่ต่างออกไป ท่านก็สามารถที่จะทำโครงการนี้ไปพัฒนาได้เพื่อให้เกิดประโยชน์และประสิทธิภาพสูงสุด

## บรรณานุกรม

สัจจะ จรัสรุ่งรวีจร, คู่มือการสร้างแอปพลิเคชันด้วย VisualBasic 6 Basic & Advanced  
 ธาริน สิทธิธรรมชารี, คู่มือการเขียนโปรแกรม Microsoft Visual Basic Version 6.0

<http://www.freevbcode.com/>

<http://web.domainlx.com/devforum/>

<http://www.planetsourcecode.com/vb/>

<http://thaibit.hypermart.net/>

<http://www.welcome.to/vbstreet>

<http://www.thai.net/ahapop>



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
 ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## ภาคผนวก ก

## Source Code เครื่องรูดการ์ด

```

;*****
;* Program Test Magnetic Card *
;* Read Track2 Modulo5 Format *
;* HardWare : ET CP2051 V2 *
;* Display Data Code to RS232 *
;* CPU Control : MCS51 *
;* Compiler : SXA51 *
;*****
;
DATA_MAG EQU P1.0 ; Data From Magnetic Card
PRESENT EQU P1.1 ; Enable Magnetic Card
CLK_MAG EQU P1.2 ; Clock Sync Magnetic Card

ORG 20H
FLAG_BUF: DS 1 ; Flag Buffer
CHECKSUM: DS 1
BUFFER: DS 40 ; 40 Byte Buffer
;
START EQU FLAG_BUF.0 ; Start Flag Status
STOP EQU FLAG_BUF.1 ; Stop Flag Status
ERROR EQU FLAG_BUF.2 ; Error Flag Status
PARITY EQU FLAG_BUF.3 ; Parity Error Flag Status

ORG 0000H
PROGRAM: MOV SP,#128-32 ; Stack 32 Byte
MOV FLAG_BUF,#0 ; Clear All Flag
SETB DATA_MAG ; Standy Signale
SETB PRESENT

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

SETB  CLK_MAG
;
INIT_SR: MOV  A,#0FDH      ; Set baud rate 9600
        MOV  TH1,A
        MOV  TL1,A
        MOV  TMOD,#0010000B ; TIMER1 MODE 1
        CLR  ES            ; Disable serial interrupt
        CLR  ET1          ; Disable timer1 interrupt
        SETB TR1          ; Set timer1 control
        MOV  SCON,#0101000B ; Serial mode 1
;
TEST_MAG: LCALL PRINT_SER
        DB  'TEST MAGNETIC CARD',0DH,0AH,00
;
DSP_NUM0: LCALL GET_DATA   ; Get Number of Card
        MOV  R0,#BUFFER
        JB  ERROR,SUM_ERR
        JB  PARITY,PAR_ERR
        SJMP DSP_NUM1
;
SUM_ERR: LCALL PRINT_SER   ; Checksum Error
        DB  'Checksum Error!',0DH,0AH,00
        CLR  ERROR
        SJMP DSP_NUM0
;
PAR_ERR: LCALL PRINT_SER   ; Parity Error
        DB  'Parity Error!',0DH,0AH,00
        CLR  PARITY
        SJMP DSP_NUM0
;
DSP_NUM1: MOV  A,@R0

```

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์ไว้เพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

ADD A,#30H
LCALL TX_BYTE ; Display End
SJMP DSP_NUM3
DSP_NUM2: ADD A,#30H
LCALL TX_BYTE
INC R0
SJMP DSP_NUM1
DSP_NUM3: MOV A,#0DH ; Carrier Return
LCALL TX_BYTE
MOV A,#0AH ; Line Feed
LCALL TX_BYTE
SJMP DSP_NUM0
;*****
;* Clear Data Buffer *
;* Register : R0,ACC *
;*****
;
CLR_BUFF: PUSH B
MOV B,#40
MOV R0,#BUFFER ; Clear Data Buffer
CLR_BUF1: CLR A
MOV @R0,A
INC R0
DJNZ B,CLR_BUF1
POP B
RET
;*****
;* Read Data From Magnetic Card *
;* Get Number of Card (Modulo5) *
;* Output : BUFFER (BCD 40Byte) *

```

```

;* Reg. : R1,R2,ACC,CHECKSUM *
;*****
;
GET_DATA: LCALL CLR_BUFF      ; Clear Buffer
          MOV  CHECKSUM,#0
          MOV  R1,#BUFFER-1    ; Pointer to Save Data
          CLR  START           ; Clear Any Flag
          CLR  STOP
          CLR  ERROR
          ;
GET_DAT0: LCALL GET_CLK       ; Get Start Sentinel
          MOV  C,DATA_MAG
          CPL  C
          JB  DATA_MAG,GET_DAT0 ; Loop Until Start Bit
          SETB START          ; Start Bit OK
          ;
GET_DAT1: MOV  R2,#5          ; Modulo 5 Format Counter
          CLR  A
          CLR  PARITY
          INC  R1              ; Point to Next Byte Save
GET_DAT2: JB  START,GET_DAT3  ; Get 1-Byte Data
          LCALL GET_CLK
          MOV  C,DATA_MAG
          CPL  C
GET_DAT3: RRC  A
          CLR  START
          DJNZ R2,GET_DAT2
          RR  A
          RR  A
          RR  A
          JB  P,GET_DAT4      ; Parity Odd OK

```

เอกสารนี้เป็นเอกสารสงวนลิขสิทธิ์ของมหาวิทยาลัยเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
 SETB PARITY รับการใส่ Parity Error ศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
 ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

    SJMP GET_DAT8
;
GET_DAT4: ANL  A,#0FH      ; Ignore Parity
          JB   STOP,GET_DAT5
          PUSH ACC
          XRL  A,CHECKSUM   ; Checksum Data
          MOV  CHECKSUM,A
          POP  ACC
GET_DAT5: MOV  @R1,A       ; Save Data
          JB   STOP,GET_DAT6 ; Stop Operation
          CJNE A,#0FH,GET_DAT1
          SETB STOP
          SJMP GET_DAT1
GET_DAT6: CJNE  A,CHECKSUM,GET_DAT7
          CLR  ERROR ; Checksum OK
          SJMP GET_DAT8
GET_DAT7: SETB  ERROR ; Checksum Error
GET_DAT8: RET

;*****
;
;* Get Magnetic Clock *
;*****
;
GET_CLK: JB   CLK_MAG,$   ; Wait Falling Clock
          NOP
          NOP
          JNB  CLK_MAG,$   ; Wait Rising Clock
          NOP
          NOP
          RET

```

```

;* Send 1-Byte to RS-232 *
;* Input  : ACC      *
;* Output : Serial port *
.*****
;
TX_BYTE: CLR  TI
        MOV  SBUF,A
        JNB  TI,$
        CLR  TI
        RET

.*****
;* Print Data to Serial Port RS232 *
;* Usage  : LCALL PRINT_SER      *
;*        : DB 'xxxx',00      *
;* Register : ACC      *
;* Note    : last byte must be 00 *
.*****
;
PRINT_SER: POP  DPH
          POP  DPL
PRINT1:  CLR  A
          MOVC A,@A+DPTR
          CJNE A,#00H,PRINT2
          SJMP PRINT3

PRINT2:  LCALL TX_BYTE
          INC  DPTR
          SJMP PRINT1

PRINT3:  PUSH  DPL
          PUSH  DPH
          RET

```

เอกสารนี้เป็นเอกสารสงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## Source Code โปรแกรมหลัก ( VisualBasic 6.0)

### Form1 (master.frm)

Option Explicit

```

<-----ส่วนของการเริ่มต้น----->

Private Sub Form_Load()
wsk1.RemotePort = 1001
'ให้ตัวหน่วงเวลาส่งข้อมูลออกไปยังไม่ทำงาน
tmddelayout.Enabled = False
'-----
'ให้ปุ่ม com out ใช้ไม่ได้
cmdout2.Enabled = False
'-----
'ให้ swf1 ทำงาน
swf1.Play
'-----
'กำหนดให้ mscomm ทำงาน
MSComm1.Settings = "9600,n,8,1"
MSComm1.CommPort = 1
MSComm1.InputLen = 1
MSComm1.PortOpen = True
MSComm1.RThreshold = 1
'-----
'ให้ com1 รับค่าจาก port โดยไป set ค่าที่ menu
com1.Checked = True
com2.Checked = False
com3.Checked = False
'-----
'ให้ปุ่ม cancel ไม่ทำงานขณะเริ่ม

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
cmdaddcancel.Enabled = False
```

```
'-----
```

```
'จัด grid ให้ได้อย่างต้องการ
```

```
dt1.RecordSource = "select * from user "
```

```
dt1.Refresh
```

```
'-----
```

```
'จัด grid temp ให้ได้อย่างต้องการ
```

```
dttemp.RecordSource = "select * from temp "
```

```
dttemp.Refresh
```

```
'-----
```

```
'จัด grid user ให้ได้อย่างต้องการ
```

```
dtuser.RecordSource = "select * from user "
```

```
dtuser.Refresh
```

```
'-----
```

```
' ให้ time เริ่มทำการupdate
```

```
timupdate.Enabled = True
```

```
'-----
```

```
' ให้com1รับค่าจากport โดยไป set ค่าที่ menu
```

```
com1.Checked = True
```

```
'-----
```

```
End Sub
```

```
<----->
```

```
'<-----ส่วนนี้จะทำการ run ตัวเองเมื่อมีการรูดการ์ด หรือมีการเปลี่ยนแปลงที่ ช่อง----->
```

```
Private Sub txmain_Change()
```

```
timerefresh.Enabled = True
```

```
End Sub
```

```
Private Sub timerefresh_Timer()
```

```
cbmain.Value = True ' เป็นการ click ปุ่ม
```

```
txmain.Text = ""
```

```
txin.Text = "" ' ให้ส่วนที่ได้จากการรูดการ์ดว่างเปล่าด้วย
```

```
timerefresh.Enabled = False
```

```
End Sub
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

'<----->
'<-----ส่วนหลักอยู่ที่นี้ถ้าสั่งให้ exit sub ที่นี่ จะหยุดทั้งหมด----->
Private Sub cbmain_Click()
    Dim x As Integer
    Call find_data 'เรียก sub หาค่าใน database
    If data_base = "error" Then Exit Sub
    Call logon_off_at
    '-----Select Case log----->
    Select Case log
        '-----Case logon----->
        Case "logon"
            Call find_position
            '-----Select Case frame_use----->
            Select Case frame_use
                Case "nouse"
                    'MsgBox "ID = " & id & " กำลัง " & log & " ที่เครื่อง " & i
                    + 1
                    'แจ้งไปที่ history ว่ามีการ update เกิดขึ้น
                    stmon.Text = stmon.Text _
                        & vbCrLf _
                        & "*****"
                    & vbCrLf _
                    & "เวลา= " & Time & vbCrLf _
                    & "ID = " & id & vbCrLf _
                    & " กำลัง " & log & vbCrLf _
                    & " ที่เครื่อง " & i + 1 & vbCrLf _
                    & "*****"
                    & vbCrLf
                    who1(i) = id 'แทนการเก็บ id
                    who2(1) = i 'แทนการเก็บ i
                    ' ส่งค่า setframe = "off" ให้ไปปิด Frame เพื่อให้รู้ว่ามีการใช้

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
 ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

setfe = "off"

Call setframe

Call setout   ' ของการ logon

#####
'ถึงตรงนี้ก็เอาtime จาก user ----> temp และเอา timeon ใ้
ไปด้วย

On Error GoTo err_aha
' ช่วงนี้ใ้เพื่อค้นหาจำนวนเงินที่เหลือแล้วใ้ในตาราง temp
dtuser.RecordSource = "select * from user where user.id
=" & id

dtuser.Refresh

timeold = dbguser.Columns(4) * 60 ' ทำให้เป็นหน่วย
วินาที
'-----
' ช่วงนี้ใ้ทำให้ตารางตรงกัน การถ่ายค่าจะได้ถูกต้อง
dttemp.RecordSource = "select * from temp where temp.id
=" & i + 1

dttemp.Refresh ' ถ้าไม่ใ้ refresh จะใช้ไม่ได้ เพราะgrid จะ
คงค่าเดิม

' ทำการถ่ายค่า
dbgtemp.Columns(4) = timeold

timeon = Timer
dbgtemp.Columns(1) = timeon

'-----
'หลังจากใ้คำสั่ง sql แล้วจะทำให้ grid ไม่เต็ม จะต้องใ้คำ
สั่งนี้อีกครั้งเพื่อแสดงข้อมูลทั้งหมด

dttemp.RecordSource = "select * from temp "
dttemp.Refresh
dtuser.RecordSource = "select * from user "
dtuser.Refresh
'-----
Exit Sub

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

err\_aha:

```

Exit Sub

#####

Case "use"
<-----เป็นส่วนในการพิมพ์ history ----->
stmon.Text = stmon.Text _
& vbCrLf _
& "*****"

& vbCrLf _
& "ไม่มีเครื่องว่างครับ " & vbCrLf _
& Time & vbCrLf _
& "*****"

& vbCrLf
stmon2.Text = stmon2.Text _
& vbCrLf _
& "*****"

& vbCrLf _
& "ไม่มีเครื่องว่างครับ " & vbCrLf _
& Time & vbCrLf _
& "*****"

& vbCrLf
<----->
Exit Sub

End Select

<-----จบ Select Case frame_use----->

<-----Case logoff----->

Case "logoff"
'MsgBox "ID = " & id & " กำลัง " & log & " เครื่องที่ " & i + 1
'แจ้งไปที่ history ว่ามีการ update เกิดขึ้น
stmon.Text = stmon.Text _
& vbCrLf _

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับ & "\*\*\*\*\*" & vbCrLf  
 ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

& "เวลา= " & Time & vbCrLf _
& "ID = " & id & vbCrLf _
& " กำลัง " & log & vbCrLf _
& "ที่เครื่อง " & i + 1 & vbCrLf _
& "*****" & vbCrLf

```

'ส่งค่าไปเปิด Frame ให้ใช้งานได้อีกครั้ง

```
setfe = "on"
```

```
Call setframe
```

```
Call setout ' ของ logoff
```

```
'$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$
```

'ถึงตรงนี้ก็เอาtime จาก temp ----> user และเอา timeold ใ้ไปด้วย

```
On Error GoTo err_aha2
```

' ช่วงนี้ทำให้ตารางตรงกัน การถ่ายค่าจะได้ถูกต้อง

```
dtuser.RecordSource = _
```

```
"select * from user where user.id =" & id
```

```
dtuser.Refresh
```

```
dttemp.RecordSource = _
```

```
"select * from temp where temp.id = "
```

```
& i + 1
```

dttemp.Refresh ' ถ้าไม่ใ้ refresh จะใ้ไม่ได้ เพราะgrid จะคงค่าเดิม

' ทำการถ่ายค่า

```
dbguser.Columns(4) = Int(Val(dbgtemp.Columns(4)) / 60)
```

```
'-----
```

'หลังจากใ้คำสั่ง sql แล้วจะทำให้ grid "ไม่เต็ม" จะต้องใ้คำสั่งนี้ใ้ีกครั้งเพื่อแสดงข้อ

มูลทั้งหมด

```
dttemp.RecordSource = "select * from temp "
```

```
dttemp.Refresh
```

```
dtuser.RecordSource = "select * from user "
```

```
dtuser.Refresh
```

```
'-----
```

```
Exit Sub
```

err\_aha2:

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$
End Select

'<----- จป Select Case log----->
End Sub

'<----->
'<-----ส่วนนี้เป็น SUB Program ต่างๆ ----->

Sub find_data()
On Error GoTo err_overdata ' ในกรณีที่ไม่ใส่ อักษร
id = txmain.Text ' รับค่าจากการ์ดเข้ามา
dt1.RecordSource = "select * from user where user.id =" & id
dt1.Refresh
lbmain.Caption = " คุณ " & fg1.TextMatrix(1, 1) ' ใส่เพื่อต้องการให้เกิด Errorเมื่อใส่เลข
เกิน Record
'ส่งค่าออกไป name , time_user
namein = fg1.TextMatrix(1, 1)
time_user = fg1.TextMatrix(1, 4)
id = txmain.Text
data_base = "noerror"
'-----
'จัด grid ให้เข้าที่เดิม
dt1.RecordSource = "select * from user "
dt1.Refresh
'-----

Exit Sub

err_overdata:
lbmain.Caption = "DataBase Error"
data_base = "error" ' ส่งค่า Error กลับไป
'จัด grid ให้เข้าที่เดิม
dt1.RecordSource = "select * from user "
dt1.Refresh

```

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

End Sub

<-----ทำการตรวจหาว่าจะ logon หรือ off ที่ตำแหน่งเท่าไร----->

Sub logon\_off\_at() ' ถ้าตำแหน่งที่ตรวจสอบมีค่าตรงกับ id แสดงว่ามีการใช้อยู่แล้วและ

กำลัง logoff

i = 0

For i = 0 To 9

If id = lb1(i).Caption \_

Then

'MsgBox "logoff"

' ส่งค่าออกไป

log = "logoff"

i = i

id = id

'-----

Exit Sub

Else

End If

Next i

'MsgBox "logon"

' ส่งค่าออกไปว่า logon แต่ไม่ต้องบอกตำแหน่ง

log = "logon"

id = id

'-----

End Sub

<----->

Sub find\_position()

i = 0

For i = 0 To 9

If fe1(i).Enabled = True And lbcomenable(i).Caption = "Enable" \_

Then

'MsgBox "No\_user"

' ส่งค่าออกไปว่าว่างที่ = i

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับใช้ภายในเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

        frame_use = "nouse"

        i = i

        id = id

        '-----

    Exit Sub

Else

    End If

Next i

'MsgBox "use"
' ส่งค่าออกไปว่า use หรือแปลว่าไม่มีเครื่องว่าง
frame_use = "use"
id = id
'-----
End Sub
<----->
Sub setframe()
Select Case setfe
    Case "on"
        fe1(i).Enabled = True
        lb1(i).Caption = ""
        lb2(i).Caption = ""
        lb3(i).Caption = ""

    Case "off"
        fe1(i).Enabled = False
        lb1(i).Caption = id
        lb2(i).Caption = namein
        lb3(i).Caption = txcomname(i).Text

End Select

End Sub
<----->
<-----ทำการล้างตาราง temp ให้เป็น 0----->

Private Sub cbtemp_Click()

```

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
If MsgBox("ถ้ามีเครื่องทำงานอยู่จะทำให้ฐานข้อมูลผิดพลาดได้", vbCritical + vbOKCancel, "
ตรวจสอบ") = vbOK _
```

```
Then
```

```
On Error GoTo err_temp
```

```
i = 0
```

```
For i = 0 To 9
```

```
dttemp.RecordSource = _
```

```
"select * from temp where temp.id = " & i + 1
```

```
dttemp.Refresh ' ถ้าไม่ใส่ refresh จะใช้ไม่ได้ เพราะ grid จะคงค่าเดิม
```

```
dbgtemp.Columns(1) = 0
```

```
dbgtemp.Columns(2) = 0
```

```
dbgtemp.Columns(3) = 0
```

```
dbgtemp.Columns(4) = 0
```

```
'หลังจากใช้คำสั่ง sql แล้วจะทำให้ grid ไม่เต็ม จะต้องใช้คำสั่ง นี้อีกครั้งเพื่อแสดงข้อมูลทั้ง
```

```
หมด
```

```
dttemp.RecordSource = "select * from temp "
```

```
dttemp.Refresh
```

```
Next i
```

```
err_temp:
```

```
Exit Sub
```

```
Else
```

```
End If
```

```
End Sub
```

```
'<----->
```

```
' ส่วนนี้เป็นนาฬิกาปกติ
```

```
Private Sub Timclock_Timer()
```

```
lbclock.Caption = Time
```

```
End Sub
```

```
'<----->
```

```
Private Sub timupdate_Timer() ' ส่วนนี้เป็นการ update ข้อมูล ในตาราง temp ทุกๆเวลาที่ตั้งไว้
```

```
' ระหว่างมีการ update จะมีการรูดการ์ดไม่ได้
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
'<---- เฝ้าดูการทำงานทุก 20 sec ----->
```

```
'Static cc As Integer
```

```
'cc = cc + 1
```

```
'txmyself.Text = "เป็นครั้งที่ " & cc
```

```
'<----->
```

```
txmain.Enabled = False
```

```
Call find_position_frame_enable
```

```
txmain.Enabled = True
```

```
End Sub
```

```
'<-----จบ ส่วนนี้เป็น SUB Program ต่างๆ ----->
```

```
'<----- เรียกเครื่องตามชื่อที่กำหนด ----->
```

```
Sub setout()
    ' ส่วนนี้จะทำการหาชื่อของเครื่องที่ต้องการจะcontrol และส่งค่าออกไปที่
    เครื่องชื่อนั้นๆ
```

```
On Error GoTo Error_wsk1
```

```
If off_error <> "error" _
```

```
Then
```

```
wsk1.RemotePort = 1001
```

```
wsk1.RemoteHost = txcomname(i).Text ' กำหนดชื่อโดยอัตโนมัติ
```

```
wsk1.Connect ' เปิด port
```

```
Exit Sub
```

```
Else
```

```
off_error = "ok"
```

```
Exit Sub
```

```
End If
```

```
Error_wsk1:
```

```
wsk1.Close
```

```
End Sub
```

```
'<-----จะจัดการ Error winsock ให้ ----->
```

```
Private Sub wsk1_Error(ByVal Number As Integer, Description As String, ByVal Scode As
Long, ByVal Source As String, ByVal HelpFile As String, ByVal HelpContext As Long,
CancelDisplay As Boolean)
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

<----- พิมพ์ค่าที่จะสั่งให้ logoff เข้าไป ----->
' ถ้าเป็นการerror ตอน logon ให้ บังคับ logoff ไปเลย แต่ถ้าเป็นการ logoff ก็ให้ข้ามการบังคับ
ไป
' ที่จุดนี้ i ได้เท่ากับ who2(1) ไปแล้ว
If log = "logon" _
    Then
        txmain.Text = who1(who2(1))
        off_error = "error"
        <----- เป็นส่วนในการพิมพ์ history ----->
        stmon.Text = stmon.Text _
            & vbCrLf _
            & "*****"
& vbCrLf _
            & "เวลา= " & Time & vbCrLf _
            & "เกิด Error winsock " & vbCrLf _
            & "ที่เครื่อง " & who2(1) + 1 & vbCrLf _
            & "*****"
& vbCrLf _
        stmon2.Text = stmon2.Text _
            & vbCrLf _
            & "*****"
& vbCrLf _
            & "เวลา= " & Time & vbCrLf _
            & "เกิด Error winsock " & vbCrLf _
            & "ที่เครื่อง " & who2(1) + 1 & vbCrLf _
            & "*****"
& vbCrLf _
        <----->
    Else
        ' ถ้าเป็นการ logoff ไม่ต้องสั่งให้ พิมพ์ค่าที่ txmain.text
        <----- เป็นส่วนในการพิมพ์ history ----->

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับใช้ในการใช้งานเพื่อตรวจสอบเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

& vbCrLf _
& "*****"
& vbCrLf _
& "เวลา=" & Time & vbCrLf _
& "เกิด Error winsock " & vbCrLf _
& "ที่เครื่อง " & who2(1) + 1 & vbCrLf _
& "*****"
& vbCrLf
stmon2.Text = stmon2.Text _
& vbCrLf _
& "*****"
& vbCrLf _
& "เวลา=" & Time & vbCrLf _
& "เกิด Error winsock " & vbCrLf _
& "ที่เครื่อง " & who2(1) + 1 & vbCrLf _
& "*****"
& vbCrLf
End If
'-----เมื่อเกิดการ Error เราจะเอา comname เครื่องนั้นออก----->
Select Case who2(1)
Case 0
Checkcom0.Value = 0
Case 1
Checkcom1.Value = 0
Case 2
Checkcom2.Value = 0
Case 3
Checkcom3.Value = 0

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
 ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

Checkcom4.Value = 0

Case 5

Checkcom5.Value = 0

Case 6

Checkcom6.Value = 0

Case 7

Checkcom7.Value = 0

Case 8

Checkcom8.Value = 0

Case 9

Checkcom9.Value = 0

End Select

'----->
wsk1.Close
End Sub

'----->
'-----ส่วนนี้จะส่งคำสั่งไปให้ slave เพื่อคำนวณผลซึ่งแล้วแต่ว่า logon หรือ off----->
Private Sub wsk1_Connect()
wsk1.SendData log ' เปลี่ยนตรงนี้เป็นคำสั่งที่จะควบคุมเครื่อง
End Sub

'----->
'-----เมื่อเครื่องลูกรับคำสั่งเรียบร้อยแล้วจะส่งข้อมูล ok กลับมาให้ตรวจสอบด้วย----->
Private Sub wsk1_DataArrival(ByVal bytesTotal As Long)
Dim datain As String
wsk1.GetData datain
If datain = "ok" _
Then
'MsgBox "ปิด port ได้"
tmddelayout.Enabled = True ' ถ้าไม่หน่วงเวลาจะทำการรับส่งข้อมูลไม่ทัน
Else
'MsgBox "ห้ามปิด port"
End If

```

เอกสารนี้สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

End Sub

```
'<----->
'<----->
```

Private Sub tmdelayout\_Timer()

wsk1.Close ' เมื่อส่งเสร็จแล้วจะตัดการสื่อสารทันทีเพื่อเปิดโอกาสให้เครื่องอื่นได้ทำงานต่อ  
tmdelayout.Enabled = False

End Sub

```
'<-----จบ เรียกเครื่องตามชื่อที่กำหนด ----->
'<-----ส่วน Update ข้อมูล----->
```

Sub find\_position\_frame\_enable()

' ส่วนนี้จะทำการหาว่ามีการใช้เครื่องอยู่ถึงจะอนุญาต

ให้ update

On Error GoTo err\_aha1Minute

i = 0

For i = 0 To 9

If fel(i).Enabled = True

Then

' ถ้าไม่มีการใช้เครื่องเลย จะผ่าน โดยไม่ update เลย

' MsgBox "i=" & i & " ผ่าน ไป"

Else ' ตรงนี้หมายถึงมีการทำงานของเครื่องแล้ว

' แจ้งไปที่ history ว่ามีการ update เกิดขึ้น

stmon.Text = stmon.Text & "\*\* Update Com " & i + 1 & " ID=" & who1(i) & "

\*\*" & vbCrLf

' ใสค่าผลต่างของเวลาใน temp แทน และเปลี่ยนค่าtimein = timeout ด้วย

dttemp.RecordSource = \_

"select \* from temp where temp.id = " & i + 1

dttemp.Refresh ' ถ้าไม่ใส่ refresh จะใช้ไม่ได้ เพราะgrid จะคงค่าเดิม

timeoff = Timer

dbgtemp.Columns(2) = timeoff

timetotal = Val(dbgtemp.Columns(2)) - Val(dbgtemp.Columns(1))

dbgtemp.Columns(3) = timetotal

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์ไว้เพื่อใช้ในการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

timeold = Val(dbgtemp.Columns(4)) - Val(dbgtemp.Columns(3))
dbgtemp.Columns(4) = timeold
'MsgBox "ที่ id= " & who1(i) & "ก่อนไป timecheck มี timeold= " & timeold
Call timecheck

If timeover = "timeout" _
    Then
        txmain.Enabled = True ' เมื่อมีการ update แล้วจะให้
การรับข้อมูลใช้ได้ซึ่งก่อนหน้านี้เราต้องหยุดเพื่อ update
        ' แจ้งไปที่ history ว่ามีการ update เกิดขึ้น
        stmon.Text = stmon.Text & vbCrLf & "#####
TIMEOUT #####" & vbCrLf
        '<-----บรรทัดนี้จะพิมพ์ ผู้ที่หมดเวลาอัตโนมัติ----->
        txmain.Text = who1(i)
        'MsgBox "id= " & who1(i) 'เพราะ id จะเก็บเฉพาะ
ค่าที่ login ครั้งสุดท้าย
        'MsgBox "ถ้าเป็น id ก่อนใช้ who=" & id
        '### ไม่จำเป็นต้องเรียกด้วยคำสั่งนี้แล้วเพราะถ้า#####
        '### มีการเปลี่ยนแปลงในช่องรับข้อมูลจะทำการ Run อัตโนมัติ###
        ' Call cbmain_Click
        '#####
        '<----->
        Exit Sub
    Else
End If

'<----->
' ถึงตรงนี้ timein กับ timeout จะมีค่าเท่ากัน เพราะรอบต่อไปในการคำนวณ
timeout จะเปลี่ยนค่าไปเอง
' ถ้ามองจากตารางอาจจะสงสัยได้ให้ใส่บรรทัดข้างล่างจะเข้าใจ
' ลองใส่บรรทัดนี้ดู----->>>> MsgBox dbgtemp.Columns(1)
dbgtemp.Columns(1) = timeoff ' ถ้ายค่า timeoff ให้ timeon เพื่อป้องกันการคำนวณ
ระยะเวลาห่างเวลาซ้ำซ้อน

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

'หลังจากใช้คำสั่ง sql แล้วจะทำให้ grid ไม่เต็ม จะต้องใช้คำสั่ง นี้อีกครั้งเพื่อ  
แสดงข้อมูลทั้งหมด

```
dttemp.RecordSource = "select * from temp "
```

```
dttemp.Refresh
```

```
'-----
```

```
End If
```

```
Next i
```

```
err_aha1Minute:
```

```
End Sub
```

```
Sub timecheck() ' ส่วนนี้จะทำการตรวจว่าหมดเวลาหรือยัง
```

```
On Error GoTo err_aha2
```

```
If timeold <= 0
```

```
Then
```

```
timeover = "timeout"
```

```
Exit Sub
```

```
Else
```

```
timeover = "timein" ' ถ้าไม่กำหนดตรงนี้ จะทำให้ timeover มีค่าเดิมตลอดทำให้
```

```
เช็คเวลาไม่ผ่าน
```

```
End If
```

```
err_aha2:
```

```
End Sub
```

```
Private Sub cbupdate_Click()
```

```
Call find_position_frame_enable
```

```
End Sub
```

```
'<-----จบส่วน Update ข้อมูล ----->
```

```
'<-----ส่วนนี้อยู่ในส่วนของ AddDataBase----->
```

```
Private Sub cmdadd_Click()
```

```
On Error GoTo err_go
```

```
If MsgBox("คุณต้องการเพิ่ม", vbCritical + vbOKCancel, "ตรวจสอบ") = vbOK _
```

```
Then
```

```
cmdadd.Enabled = False
```

```
cmddel.Enabled = False
```

เอกสารนี้เป็นเอกสารที่จัดทำขึ้นเพื่อการศึกษานานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

cmdaddcancel.Enabled = True

Data1.Recordset.AddNew

txid.Text = "" ' กำหนดให้ ว่างเปล่า เพราะตารางจะกำหนดให้ = 0 มาอยู่แล้วจะทำให้
ตรวจจับไม่พบ

If txname.Text = "" _
    Then
        MsgBox "กรุณาใส่ชื่อด้วยครับ"
    ElseIf txdate.Text = "" _
    Then
        MsgBox " กรุณาใส่วันที่ด้วยครับ"
    ElseIf txmoney.Text = "" _
    Then
        MsgBox "กรุณาใส่จำนวนเงินด้วยครับ"
    ElseIf txtime.Text = "" _
    Then
        MsgBox " กรุณาใส่เวลาด้วยครับ"
    ElseIf txid.Text = "" _
    Then
        MsgBox "กรุณาใส่รหัสบัตรด้วยครับ"
    Else
        Data1.Recordset.MoveFirst
        Data1.Recordset.Update
        cmdadd.Enabled = True
        cmddel.Enabled = True
        cmdaddcancel.Enabled = False
    End If
Else
    End If
err_go:
End Sub

Private Sub cmdref_Click()
    On Error GoTo err_aha

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

If txname.Text = "" _
    Then
        MsgBox "กรุณาใส่ชื่อด้วยครับ"
    ElseIf txdate.Text = "" _
    Then
        MsgBox " กรุณาใส่วันที่ด้วยครับ"
    ElseIf txmoney.Text = "" _
    Then
        MsgBox "กรุณาใส่จำนวนเงินด้วยครับ"
    ElseIf txttime.Text = "" _
    Then
        MsgBox " กรุณาใส่เวลาด้วยครับ"
    ElseIf txid.Text = "" _
    Then
        MsgBox "กรุณาใส่รหัสบัตรด้วยครับ"
    Else
'จัด grid ให้ได้อย่างต้องการ
Data1.RecordSource = "select * from user "
Data1.Refresh
'-----
cmdadd.Enabled = True
cmddel.Enabled = True
Data1.Recordset.MoveFirst
End If

err_aha:
End Sub

Private Sub cmddel_Click()
    If MsgBox("คุณต้องการลบเดี๋ยวนี", vbCritical + vbOKCancel, "ตรวจสอบ") = vbOK _
    Then
        Data1.Recordset.Delete
        Data1.Recordset.MovePrevious
    Else

```

เอกสารนี้เป็นเอกสารสงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

End If
End Sub
Private Sub cmdaddcancel_Click()
    cmdadd.Enabled = True
    cmddel.Enabled = True
    Data1.Recordset.CancelUpdate 1
    cmdaddcancel.Enabled = False
End Sub
Private Sub cmdaddtop_Click()
    Data1.Recordset.MoveFirst
End Sub
Private Sub cmdadddown_Click()
    Data1.Recordset.MoveLast
End Sub
Private Sub cmdnextadd_Click()
    On Error GoTo enddata
    Data1.Recordset.MoveNext
Exit Sub
enddata:
    MsgBox "ไม่มีข้อมูลแล้วครับ", , "แจ้งข้อผิดพลาด"
End Sub
Private Sub cmdbackadd_Click()
    On Error GoTo enddata
    Data1.Recordset.MovePrevious
Exit Sub
enddata:
    MsgBox "ไม่มีข้อมูลแล้วครับ", , "แจ้งข้อผิดพลาด"
End Sub
<-----หมวดส่วน AddData Base ----->
' ทำการrefresh อยู่ในส่วนของ time update
Private Sub cmdreuser_Click()

```

เอกสารนี้เป็จัด grid ให้ได้อย่างต้องการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

dtuser.RecordSource = "select * from user "
dtuser.Refresh
'-----
End Sub
'<-----สอง sub นี้ ให้พิมพ์ วันที่ แบบ auto ----->
Private Sub txdate_Click()
    txdate.Text = Date
End Sub
Private Sub txdate_GotFocus()
    txdate.Text = Date
End Sub
'<-----จบการพิมพ์ auto ----->
'<-----ส่วนนี้เป็นการกำหนด password----->
Private Sub SSTab1_Click(PreviousTab As Integer) ' ส่วนนี้เป็นการตรวจจับการเข้าเขตต้องห้าม
    Select Case SSTab1.Tab
    Case 0
        feadddata.Caption = "Lock"
        fefill.Caption = "Lock"
        txusername.Text = "กรุณาใส่รหัสผ่านด้วยครับ"
        txpassword.Text = ""
        fefill.Enabled = False
        feadddata.Enabled = False
    Case 3
        Exit Sub
    End Select
End Sub
Private Sub txpassword_Change()
    feadddata.Caption = "Lock"
    fefill.Caption = "Lock"
    fefill.Enabled = False
    feadddata.Enabled = False

```

```

If txpassword.Text = "ahapop" _
Then
    feadddata.Enabled = True
    feadddata.Caption = "UnLock"
    fefill.Caption = "ข้อมูลลูกค้า"
    fefill.Enabled = True
Else
    End If
Exit Sub

```

```
End Sub
```

```
'<-----จบส่วน password----->
```

```
'<-----ตรงนี้เป็นส่วนของการเรียก menu----->
```

```
Private Sub program_Click()
```

```
frprog.Show
```

```
End Sub
```

```
Private Sub programmer_Click()
```

```
frprog.Show
```

```
End Sub
```

```
'-----
```

```
'เป็นการ clear history
```

```
Private Sub cmdclear_Click()
```

```
stmon.Text = ""
```

```
End Sub
```

```
'-----
```

```
'เป็นการ clear history
```

```
Private Sub cmdclear2_Click()
```

```
stmon2.Text = ""
```

```
End Sub
```

```
'-----
```

```
'เป็นการ set ให้เป็น com1 อย่างเดียว
```

```
Private Sub com1_Click()
```

```
On Error GoTo Error_comport1
```

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์การใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

com1.Checked = True

If com1.Checked = True Then _
    MSComm1.PortOpen = False
    com2.Checked = False
    com3.Checked = False
    MSComm1.CommPort = 1
    MSComm1.PortOpen = True
    Exit Sub

Error_comport1:
'<-----เป็นส่วนในการแจ้ง Error ไปที่ History และ Error ----->
    stmon.Text = stmon.Text _
        & vbCrLf _
        & "*****"
& vbCrLf _
        & "เวลา=" & Time & vbCrLf _
        & "เกิด Error ComPort 1 " & vbCrLf _
        & "*****"
& vbCrLf _
    stmon2.Text = stmon2.Text _
        & vbCrLf _
        & "*****"
& vbCrLf _
        & "เวลา=" & Time & vbCrLf _
        & "เกิด Error ComPort 1 " & vbCrLf _
        & "*****"
& vbCrLf

'<----->

End Sub

'-----
เป็นการ set ให้เป็น com2 อย่างเดียว
Private Sub com2_Click()

```

```

com2.Checked = True
If com2.Checked = True Then _
    MSComm1.PortOpen = False
    com1.Checked = False
    com3.Checked = False
    MSComm1.CommPort = 2
    MSComm1.PortOpen = True
    Exit Sub
Error_comport2:
'<-----เป็นส่วนในการแจ้ง Error ไปที่ History และ Error ----->
    stmon.Text = stmon.Text _
        & vbCrLf _
        & "*****" & vbCrLf _
        & "เวลา= " & Time & vbCrLf _
        & "เกิด Error ComPort 2 " & vbCrLf _
    & "*****" & vbCrLf
stmon2.Text = stmon2.Text _
    & vbCrLf _
    & "*****"
& vbCrLf _
    & "เวลา= " & Time & vbCrLf _
    & "เกิด Error ComPort 2 " & vbCrLf _
    & "*****"
& vbCrLf
'<----->
End Sub
'-----
'เป็นการ set ให้เป็น com3 อย่างเดียว
Private Sub com3_Click()
On Error GoTo Error_comport3
    com3.Checked = True

```

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์ไว้เพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

MSComm1.PortOpen = False
com1.Checked = False
com2.Checked = False
MSComm1.CommPort = 3
MSComm1.PortOpen = True
Exit Sub

```

Error\_comport3:

```

'<-----เป็นส่วนในการแจ้ง Error ไปที่ History และ Error ----->
    stmon.Text = stmon.Text _
        & vbCrLf _
        & "*****"
& vbCrLf _
        & "เวลา= " & Time & vbCrLf _
        & "เกิด Error ComPort 3 " & vbCrLf _
        & "*****"
& vbCrLf _
    stmon2.Text = stmon2.Text _
        & vbCrLf _
        & "*****"
& vbCrLf _
        & "เวลา= " & Time & vbCrLf _
        & "เกิด Error ComPort 3 " & vbCrLf _
        & "*****"
& vbCrLf

```

'<----->

End Sub

'<----->

'<-----ส่วนนี้จะทำการติดต่อกับเครื่องรูดการ์ด----->

Private Sub MSComm1\_OnComm()

Dim Buffer As Variant

Buffer = MSComm1.Input

ShowData txin, (StrConv(Buffer, vbUnicode))

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับใช้ภายในเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
txmain.Text = Mid(txin.Text, 2, 5)
```

```
End Sub
```

```
Private Static Sub ShowData(Term As Control, Data As String)
```

```
On Error GoTo Handler
```

```
Term.SelText = Data
```

```
Handler:
```

```
Resume Next
```

```
End Sub
```

```
<-----จบส่วนติดต่อการ์ด----->
```

```
' คำสั่งตรงนี้จะทำให้ history ลงมาอยู่ที่บรรทัดล่าสุด
```

```
Private Sub stmon_Change()
```

```
stmon.SelStart = Len(stmon.Text)
```

```
End Sub
```

```
'-----
```

```
Private Sub stmon2_Change()
```

```
stmon2.SelStart = Len(stmon.Text)
```

```
End Sub
```

```
'-----
```

```
<-----ส่วนนี้จะควบคุมรายละเอียดของ ControlPanel----->
```

```
Private Sub Checkcom0_Click()
```

```
' จุดคำสั่งนี้ ใ้ใส่เพื่อแก้ไข การล๊ิมใส่ comname
```

```
If txcomname(0) = "" _
```

```
Then
```

```
Checkcom0.Value = 0
```

```
Exit Sub
```

```
'-----
```

```
ElseIf Checkcom0.Value = 1 _
```

```
Then
```

```
lbcomenable(0).BackColor = &HC0FFC0
```

```
lbcomenable(0).Caption = "Enable"
```

```
txcomname(0).Locked = True
```

```
txcomname(0).BackColor = &HC0FFC0
```

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์ของ บริษัท สยามคอมพิวติ้ง จำกัด ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

Else
    lbcomenable(0).BackColor = &HC0C0FF
    lbcomenable(0).Caption = "Disable"
    txcomname(0).Locked = False
    txcomname(0).BackColor = &H80000005
    txcomname(0).Text = ""

```

```
End If
```

```
End Sub
```

```
Private Sub Checkcom1_Click()
```

```
' จุดคำสั่งนี้ ใส่เพื่อแก้ไข การล๊ิมใส่ comname
```

```
If txcomname(1) = "" _
```

```
Then
```

```
Checkcom1.Value = 0
```

```
Exit Sub
```

```
-----
ElseIf Checkcom1.Value = 1 _
```

```
Then
```

```
lbcomenable(1).BackColor = &HC0FFC0
```

```
lbcomenable(1).Caption = "Enable"
```

```
txcomname(1).Locked = True
```

```
txcomname(1).BackColor = &HC0FFC0
```

```
Else
```

```
lbcomenable(1).BackColor = &HC0C0FF
```

```
lbcomenable(1).Caption = "Disable"
```

```
txcomname(1).Locked = False
```

```
txcomname(1).BackColor = &H80000005
```

```
txcomname(1).Text = ""
```

```
End If
```

```
End Sub
```

```
Private Sub Checkcom2_Click()
```

```
' จุดคำสั่งนี้ ใส่เพื่อแก้ไข การล๊ิมใส่ comname
```

```
If txcomname(2) = ""
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Then

    Checkcom2.Value = 0

    Exit Sub

-----

ElseIf Checkcom2.Value = 1 \_

    Then

        lbcomenable(2).BackColor = &HC0FFC0

        lbcomenable(2).Caption = "Enable"

        txcomname(2).Locked = True

        txcomname(2).BackColor = &HC0FFC0

    Else

        lbcomenable(2).BackColor = &HC0C0FF

        lbcomenable(2).Caption = "Disable"

        txcomname(2).Locked = False

        txcomname(2).BackColor = &H8000005

        txcomname(2).Text = ""

End If

End Sub

Private Sub Checkcom3\_Click()

' ชุดคำสั่งนี้ ไว้เพื่อแก้ไข การลิมใส่ comname

If txcomname(3) = "" \_

    Then

        Checkcom3.Value = 0

        Exit Sub

-----

ElseIf Checkcom3.Value = 1 \_

    Then

        lbcomenable(3).BackColor = &HC0FFC0

        lbcomenable(3).Caption = "Enable"

        txcomname(3).Locked = True

        txcomname(3).BackColor = &HC0FFC0

    Else

เอกสารนี้เป็นเอกสาร ที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

lbcomenable(3).BackColor = &HC0C0FF
lbcomenable(3).Caption = "Disable"
txcomname(3).Locked = False
txcomname(3).BackColor = &H80000005
txcomname(3).Text = ""

```

```
End If
```

```
End Sub
```

```
Private Sub Checkcom4_Click()
```

```
' ชุดคำสั่งนี้ ใ้ใส่เพื่อแก้ไข การลิมใ้ comname
```

```
If txcomname(4) = "" _
```

```
Then
```

```
Checkcom4.Value = 0
```

```
Exit Sub
```

```
-----
ElseIf Checkcom4.Value = 1
```

```
Then
```

```
lbcomenable(4).BackColor = &HC0FFC0
```

```
lbcomenable(4).Caption = "Enable"
```

```
txcomname(4).Locked = True
```

```
txcomname(4).BackColor = &HC0FFC0
```

```
Else
```

```
lbcomenable(4).BackColor = &HC0C0FF
```

```
lbcomenable(4).Caption = "Disable"
```

```
txcomname(4).Locked = False
```

```
txcomname(4).BackColor = &H80000005
```

```
txcomname(4).Text = ""
```

```
End If
```

```
End Sub
```

```
Private Sub Checkcom5_Click()
```

```
' ชุดคำสั่งนี้ ใ้ใส่เพื่อแก้ไข การลิมใ้ comname
```

```
If txcomname(5) = "" _
```

```
Then
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
Checkcom5.Value = 0
```

```
Exit Sub
```

```
ElseIf Checkcom5.Value = 1 _
```

```
Then
```

```
lbcomenable(5).BackColor = &HC0FFC0
```

```
lbcomenable(5).Caption = "Enable"
```

```
txcomname(5).Locked = True
```

```
txcomname(5).BackColor = &HC0FFC0
```

```
Else
```

```
lbcomenable(5).BackColor = &HC0C0FF
```

```
lbcomenable(5).Caption = "Disable"
```

```
txcomname(5).Locked = False
```

```
txcomname(5).BackColor = &H80000005
```

```
txcomname(5).Text = ""
```

```
End If
```

```
End Sub
```

```
Private Sub Checkcom6_Click()
```

```
' ชุดคำสั่งนี้ ใส่เพื่อแก้ไข การลิ้มใส่ comname
```

```
If txcomname(6) = "" _
```

```
Then
```

```
Checkcom6.Value = 0
```

```
Exit Sub
```

```
ElseIf Checkcom6.Value = 1 _
```

```
Then
```

```
lbcomenable(6).BackColor = &HC0FFC0
```

```
lbcomenable(6).Caption = "Enable"
```

```
txcomname(6).Locked = True
```

```
txcomname(6).BackColor = &HC0FFC0
```

```
Else
```

เอกสารนี้เป็นเอกสารที่ส่ง lbcomenable(6).BackColor = &HC0C0FF ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

lbcomenable(6).Caption = "Disable"
txcomname(6).Locked = False
txcomname(6).BackColor = &H80000005
txcomname(6).Text = ""

End If

End Sub

Private Sub Checkcom7_Click()
' ชุดคำสั่งนี้ ใ้ใส่เพื่อแก้ไข การล๊ิมใส่ comname
If txcomname(7) = "" _
    Then
        Checkcom7.Value = 0
        Exit Sub
    -----
ElseIf Checkcom7.Value = 1 _
    Then
        lbcomenable(7).BackColor = &HC0FFC0
        lbcomenable(7).Caption = "Enable"
        txcomname(7).Locked = True
        txcomname(7).BackColor = &HC0FFC0
    Else
        lbcomenable(7).BackColor = &HC0C0FF
        lbcomenable(7).Caption = "Disable"
        txcomname(7).Locked = False
        txcomname(7).BackColor = &H80000005
        txcomname(7).Text = ""
    End If
End Sub

Private Sub Checkcom8_Click()
' ชุดคำสั่งนี้ ใ้ใส่เพื่อแก้ไข การล๊ิมใส่ comname
If txcomname(8) = "" _
    Then

```

เอกสารนี้เป็นเอกสารของมหาวิทยาลัยเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง ไม่ควรเผยแพร่โดยไม่ได้รับอนุญาต การใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Exit Sub

'-----

ElseIf Checkcom8.Value = 1 \_

Then

lbcomenable(8).BackColor = &HC0FFC0

lbcomenable(8).Caption = "Enable"

txcomname(8).Locked = True

txcomname(8).BackColor = &HC0FFC0

Else

lbcomenable(8).BackColor = &HC0C0FF

lbcomenable(8).Caption = "Disable"

txcomname(8).Locked = False

txcomname(8).BackColor = &H8000005

txcomname(8).Text = ""

End If

End Sub

Private Sub Checkcom9\_Click()

' จุดคำสั่งนี้ ใส่เพื่อแก้ไข การล๊ิมใส่ comname

If txcomname(9) = "" \_

Then

Checkcom9.Value = 0

Exit Sub

'-----

ElseIf Checkcom9.Value = 1 \_

Then

lbcomenable(9).BackColor = &HC0FFC0

lbcomenable(9).Caption = "Enable"

txcomname(9).Locked = True

txcomname(9).BackColor = &HC0FFC0

Else

lbcomenable(9).BackColor = &HC0C0FF

lbcomenable(9).Caption = "Disable" เท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

txcomname(9).Locked = False
txcomname(9).BackColor = &H80000005
txcomname(9).Text = ""

```

```
End If
```

```
End Sub
```

```
'<----->
```

```
'<-----ส่วนนี้ทำการบันทึก History ลง file----->
```

```
Private Sub cmdsavehistory_Click()
```

```
Dim fso As New FileSystemObject
```

```
Dim obf As File
```

```
Dim obt As TextStream
```

```
fso.CreateTextFile ("C:\Program Files\internetcard\history.txt")
```

```
Set obf = fso.GetFile("C:\Program Files\internetcard\history.txt")
```

```
Set obt = obf.OpenAsTextStream(ForWriting)
```

```
obt.Write stmon.Text
```

```
obt.WriteLine
```

```
obt.Close
```

```
End Sub
```

```
'<----->
```

```
'<-----ส่วนนี้ทำการบันทึก Error ลง file----->
```

```
Private Sub cmdsavehistory2_Click()
```

```
Dim fso As New FileSystemObject
```

```
Dim obf As File
```

```
Dim obt As TextStream
```

```
fso.CreateTextFile ("C:\Program Files\internetcard\error.txt")
```

```
Set obf = fso.GetFile("C:\Program Files\internetcard\error.txt")
```

```
Set obt = obf.OpenAsTextStream(ForWriting)
```

```
obt.Write stmon2.Text
```

```
obt.WriteLine
```

```
obt.Close
```

```
End Sub
```

```
'<----->
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## Source Code โปรแกรมหลัก ( VisualBasic 6.0)

### frprog (master2.frm)

```

Private Declare Function SetWindowPos Lib "user32" ( _
    ByVal hwnd As Long, _
    ByVal hWndInsertAfter As Long, _
    ByVal x As Long, ByVal y As Long, _
    ByVal cx As Long, _
    ByVal cy As Long, _
    ByVal wFlags As Long _
) As Long
Const HWND_TOPMOST = -1
Const HWND_NOTOPMOST = -2
Const SWP_NOSIZE = &H1
Const SWP_NOMOVE = &H2
Const SWPFLAGS = SWP_NOSIZE Or SWP_NOMOVE
Private Sub SetTopMost(ByVal frm As Form, ByVal bTopMost As Boolean)
    Dim lUseVal As Long
    Dim lRet As Long
    lUseVal = IIf(bTopMost, HWND_TOPMOST, HWND_NOTOPMOST)
    lRet = SetWindowPos(frm.hwnd, lUseVal, 0, 0, 0, 0, SWPFLAGS)
End Sub
Private Sub Command1_Click()
    Unload Me
    Set frprog = Nothing
End Sub
Private Sub Form_Load()
    SetTopMost Me, True
End Sub
Private Sub Form_Unload(Cancel As Integer)
    SetTopMost Me, False
End Sub

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## SourceCode โปรแกรมหลัก ( VisualBasic 6.0)

### frprog (master3.frm)

```

Private Declare Function SetWindowPos Lib "user32" ( _
    ByVal hwnd As Long, _
    ByVal hWndInsertAfter As Long, _
    ByVal x As Long, ByVal y As Long, _
    ByVal cx As Long, _
    ByVal cy As Long, _
    ByVal wFlags As Long _
) As Long
Const HWND_TOPMOST = -1
Const HWND_NOTOPMOST = -2
Const SWP_NOSIZE = &H1
Const SWP_NOMOVE = &H2
Const SWPFLAGS = SWP_NOSIZE Or SWP_NOMOVE
Private Sub SetTopMost(ByVal frm As Form, ByVal bTopMost As Boolean)
Dim lUseVal As Long
Dim lRet As Long
    lUseVal = IIf(bTopMost, HWND_TOPMOST, HWND_NOTOPMOST)
    lRet = SetWindowPos(frm.hwnd, lUseVal, 0, 0, 0, 0, SWPFLAGS)
End Sub
Private Sub Command1_Click()
Unload Me
Set frprog = Nothing
End Sub
Private Sub Form_Load()
swfaboutpgm.Play
    SetTopMost Me, True
End Sub
Private Sub Form_Unload(Cancel As Integer)
    SetTopMost Me, False:End Sub

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## SourceCode โปรแกรมหลัก ( VisualBasic 6.0)

### MoDule1(master.bas)

Option Explicit

Global i As Integer 'เป็น index ซี่

Global id As String 'รับค่าจากการ์ด

Global namein As String 'ชื่อที่ได้จาก grid

Global time\_user As Currency 'เวลาที่ได้จากตาราง user

Global log As String ' logon หรือ off

Global setfe As String ' กำหนดว่า จะทำอะไรกับ frame on,off

Global frame\_use As String ' ใช้อยู่หรือไม่

Global data\_base As String 'ฐานข้อมูลมีปัญหาหรือไม่

Global timeold As Currency 'ไว้ถ่ายค่าเวลาระหว่างฐาน

Global timeon As Currency '-----

Global timeoff As Currency '-----

Global timetotal As Currency '-----

Global timeover As String ' ตั้งค่า timeout ออกมาเมื่อหมดเวลา

Global who1(9), who2(9) As Long 'เพื่อเก็บค่า id ไว้ที่ frame

Global a As Integer 'เป็นการเก็บค่า index ของ checkbox ในcontrolpanel

Global datain As String 'ค่าที่ยืนยันการส่งผ่านระบบ

Global off\_error As String ' เป็นการระบุค่าพิเศษ สำหรับการ logoff เนื่องจากการ Error ของ winsock

## SourceCode โปรแกรมลูกข่าย ( VisualBasic 6.0)

### Form1(slave1.frm)

Private Const EWX\_SHUTDOWN = 1

Private Const EWX\_REBOOT = 2

Private Const EWX\_LOGOFF = 0

Private Declare Function ExitWindowsEx Lib "user32" (ByVal uFlags As Long, ByVal dwReserved As Long) As Long

Private Declare Function SystemParametersInfo Lib "user32" \_

Alias "SystemParametersInfoA" (ByVal uAction As Long, \_

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

ByVal uParam As Long, lpvParam As Any, _
ByVal fuWinIni As Long) As Long
Private Const SPI_SCREENSAVERRUNNING = 97
Private Declare Function SetWindowPos Lib "user32" ( _
    ByVal hwnd As Long, _
    ByVal hWndInsertAfter As Long, _
    ByVal x As Long, ByVal y As Long, _
    ByVal cx As Long, _
    ByVal cy As Long, _
    ByVal wFlags As Long _
) As Long
Const HWND_TOPMOST = -1
Const HWND_NOTOPMOST = -2
Const SWP_NOSIZE = &H1
Const SWP_NOMOVE = &H2
Const SWPFLAGS = SWP_NOSIZE Or SWP_NOMOVE
Private Sub SetTopMost(ByVal frm As Form, ByVal bTopMost As Boolean)
Dim lUseVal As Long
Dim lRet As Long
    lUseVal = IIf(bTopMost, HWND_TOPMOST, HWND_NOTOPMOST)
    lRet = SetWindowPos(frm.hwnd, lUseVal, 0, 0, 0, 0, SWPFLAGS)
End Sub
Private Sub Form_Load()
SetTopMost Me, True
wsk1.LocalPort = 1001
wsk1.Listen
Dim ret As Integer
    Dim pOld As Boolean
    ret = SystemParametersInfo(SPI_SCREENSAVERRUNNING, True, pOld, 0)
    Load Form2
    Form2.Show

```

เอกสารนี้เป็นเอกสารที่ **Form2.SetFocus** ใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

End Sub

Private Sub Form_Unload(Cancel As Integer)
SetTopMost Me, False
Form1.Caption = "คุณกำลังทำผิดกฎ !!!"
MsgBox "แทน function หลังจากนี้ด้วย logoff หรือ shutdown"
'ExitWindowsEx 0, 0
End Sub

Private Sub Timer1_Timer()
lab1.Caption = Now
End Sub

Private Sub wsk1_Close()
Dim a As Integer
wsk1.Close
wsk1.Listen
End Sub

Private Sub wsk1_ConnectionRequest(ByVal requestID As Long)
If wsk1.State <> sckClosed Then wsk1.Close
wsk1.Accept requestID
'lab1.Caption = "มีการติดต่อเข้ามา"
'Form1.Caption = "มีการติดต่อเข้ามา"
'MsgBox "มีการติดต่อเข้ามา"
End Sub

Private Sub wsk1_DataArrival(ByVal bytesTotal As Long)
Dim datain As String
wsk1.GetData datain
'lab1.Caption = "มีคำสั่งให้ " & datain
Form1.Caption = datain
If datain = "logoff" _
Then
Dim ret As Integer
Dim pOld As Boolean
ret = SystemParametersInfo(SPI_SCREENSAVERRUNNING, True, pOld, 0)

```

```

Load Form2
Form2.Show
Form2.SetFocus
wsk1.SendData "ok" 'ส่งข้อมูลกลับว่ารับข้อมูลมาแล้ว
Exit Sub
Else
Unload Form2
Form1.SetFocus
wsk1.SendData "ok" 'ส่งข้อมูลกลับว่ารับข้อมูลมาแล้ว
End If
End Sub

```

### SourceCode โปรแกรมลูกข่าย ( VisualBasic 6.0)

#### Form2(slave2.frm)

```

Private Declare Function SetWindowPos Lib "user32" ( _
    ByVal hwnd As Long, _
    ByVal hWndInsertAfter As Long, _
    ByVal x As Long, ByVal y As Long, _
    ByVal cx As Long, _
    ByVal cy As Long, _
    ByVal wFlags As Long _
) As Long
Const HWND_TOPMOST = -1
Const HWND_NOTOPMOST = -2
Const SWP_NOSIZE = &H1
Const SWP_NOMOVE = &H2
Const SWPFLAGS = SWP_NOSIZE Or SWP_NOMOVE
Private Sub SetTopMost(ByVal frm As Form, ByVal bTopMost As Boolean)
Dim lUseVal As Long
Dim lRet As Long

```

```

lUseVal = IIf(bTopMost, HWND_TOPMOST, HWND_NOTOPMOST)

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

    lRet = SetWindowPos(frm.hwnd, lUseVal, 0, 0, 0, SWPFLAGS)
End Sub
Private Sub Command1_Click()
Unload Form2
Unload Form1
End Sub
Private Sub Form_Load()
    SetTopMost Me, True
End Sub
Private Sub Form_Unload(Cancel As Integer)
    SetTopMost Me, False
End Sub
'<----- แสดงเวลาบน form ----->
Private Sub Timer1_Timer()
    lbltimenow.Caption = Now
End Sub
'<----->

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## วิธีการติดตั้ง

นำ Folder ที่ชื่อว่า internetcard ไปไว้ที่ part C:\Program Files\XXXXX  
(XXXXX คือตำแหน่งที่จะวาง folder)

ภายใน Folder จะประกอบไปด้วย

1. aboutnong.swf
2. Kmitl.swf
3. Temp.mdb
4. User.mdb

นำโปรแกรม Main ที่ทำเป็นจุด EXE แล้ว ไป Run บนเครื่องที่ต้องการจะให้ เป็นเครื่อง Master

นำโปรแกรม Slave ที่ทำเป็นจุด EXE แล้ว ไป Run บนเครื่องที่ต้องการจะให้ เป็นเครื่องลูกข่าย

หมายเหตุ โปรแกรม Slave ไม่สามารถ Run ได้บน ระบบปฏิบัติการ Windows NT



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## ภาคผนวก ข



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## +5V-Powered, Multi-Channel RS-232 Drivers/Receivers

### ELECTRICAL CHARACTERISTICS—MAX220/222/232A/233A/242/243 (continued)

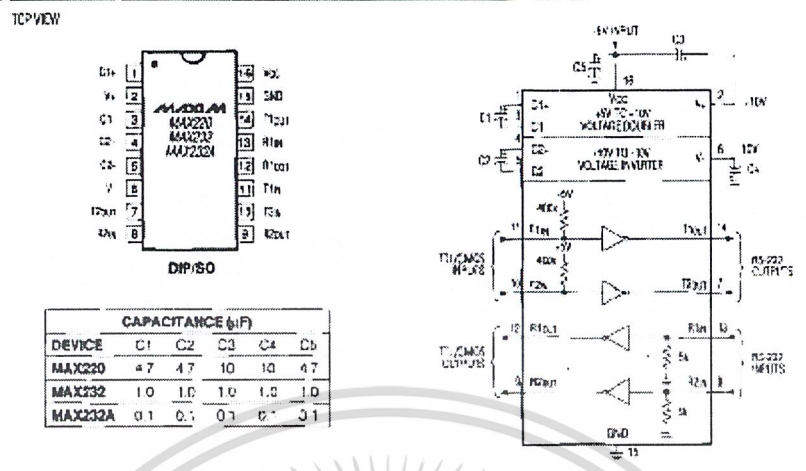
(V<sub>CC</sub> = +5V ±10%, C<sub>1</sub> C<sub>4</sub> = 0.1μF, I<sub>A</sub> = I<sub>MIN</sub> to I<sub>MAX</sub>, unless otherwise noted.)

PARAMETER	CONDITIONS		MIN	TYP	MAX	UNITS
EN Input Threshold Low	MAX242			1.4	0.6	V
EN Input Threshold High	MAX242		2.0	1.4		V
<b>POWER SUPPLY</b>						
Operating Supply Voltage			4.5		5.5	V
V <sub>CC</sub> Supply Current (I <sub>STDBY</sub> - V <sub>CC</sub> ) Figure 5, 6, 9, 19	No load	MAX220		0.5	2	mA
		MAX222/232A/233A/242/243		4	10	
	3kΩ load on all outputs	MAX220		1.2		
		MAX222/232A/233A/242/243		1.5		
Shutdown Supply Current	MAX222/242	T <sub>A</sub> = +25°C		0.1	10	μA
		T <sub>A</sub> = 0° to +70°C		2	50	
		T <sub>A</sub> = -40° to -35°C		2	50	
		T <sub>A</sub> = -55° to -125°C		35	100	
SHDN Input Leakage Current	MAX222/242				±1	μA
SHDN Threshold Low	MAX222/242			1.4	0.8	V
SHDN Threshold High	MAX222/242		2.0	1.4		V
<b>AC CHARACTERISTICS</b>						
Transition Slow Rate	C <sub>L</sub> = 50pF to 2500pF, R <sub>L</sub> = 3kΩ to 7kΩ, V <sub>CC</sub> = 5V, T <sub>A</sub> = +25°C, measured from +3V to -3V or -3V to -3V	MAX222/232A/233A/242/243	6	12	30	V/μs
		MAX220	1.5	3	30	
Transmitter Propagation Delay TLL to RS-232 (normal operation), Figure 1	PHLT	MAX222/232A/233A/242/243		1.9	3.5	ns
		MAX220		4	10	
Receiver Propagation Delay RS-232 to TLL (normal operation), Figure 2	PLHR	MAX222/232A/233A/242/243		1.5	3.5	ns
		MAX220		5	10	
Receiver Propagation Delay RS-232 to TLL (shutdown), Figure 2	PLHR	MAX222/232A/233A/242/243		0.5	1	ns
		MAX220		0.6	3	
Receiver Propagation Delay RS-232 to TLL (shutdown), Figure 2	PLHS	MAX242		0.5	10	ns
		MAX242		2.5	10	
Receiver Output Enable Time, Figure 3	EN	MAX242		125	500	ns
Receiver Output Disable Time, Figure 3	DE	MAX242		150	500	ns
Transmitter Output Enable Time (SHDN goes high), Figure 4	LE	MAX222/242, 0.1μF caps (includes charge-pump start-up)		250		μs
Transmitter Output Disable Time (SHDN goes low), Figure 4	DET	MAX222/242, 0.1μF caps		500		ns
Transmitter + to Propagation Delay Difference (normal operation)	PHLT - PLHT	MAX222/232A/233A/242/243		300		ns
		MAX220		2000		
Receiver + to Propagation Delay Difference (normal operation)	PLHS - PLHF	MAX222/232A/233A/242/243		100		ns
		MAX220		225		

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## +5V-Powered, Multi-Channel RS-232 Drivers/Receivers

MAX220-MAX249



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Flash Programming Modes

Mode	RST/VPP	P3.2/PROG	P3.3	P3.4	P3.5	P3.7
Write Code Data <sup>(1)(2)</sup>	12V		L	H	H	H
Read Code Data <sup>(1)</sup>	H	H	L	L	H	H
Write Lock	Bit - 1		H	H	H	H
	Bit - 2		H	H	L	L
Chip Erase	12V	(2)	H	L	L	L
Read Signature Byte	H	H	L	L	L	L

- Notes: 1. The internal PEROM address counter is reset to 000H on the rising edge of RST and is advanced by a positive pulse at XTAL 1 pin.  
 2. Chip Erase requires a 10 ms PROG pulse.  
 3. P3.1 is pulled Low during programming to indicate RDY/BSY.

Figure 3. Programming the Flash Memory

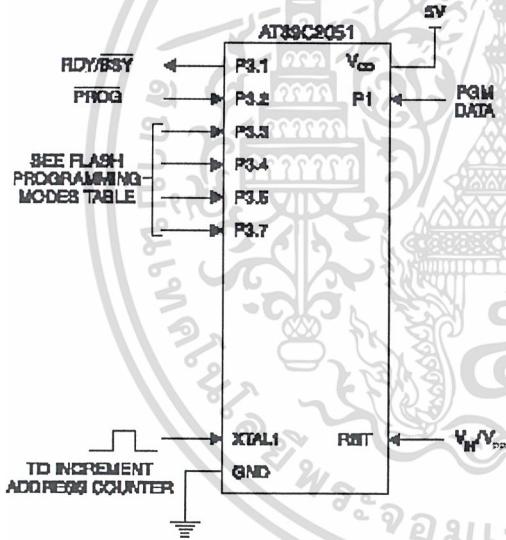
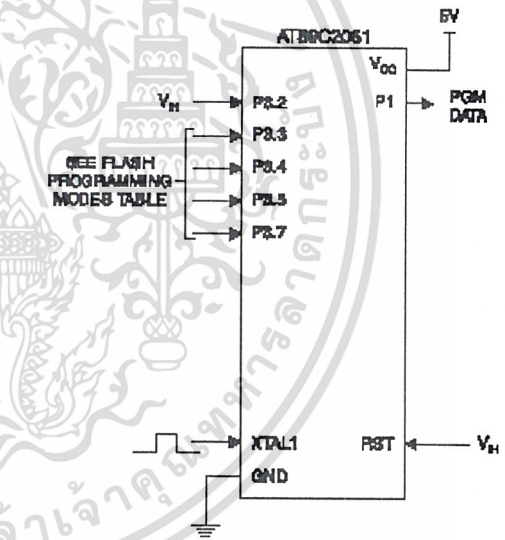


Figure 4. Verifying the Flash Memory



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

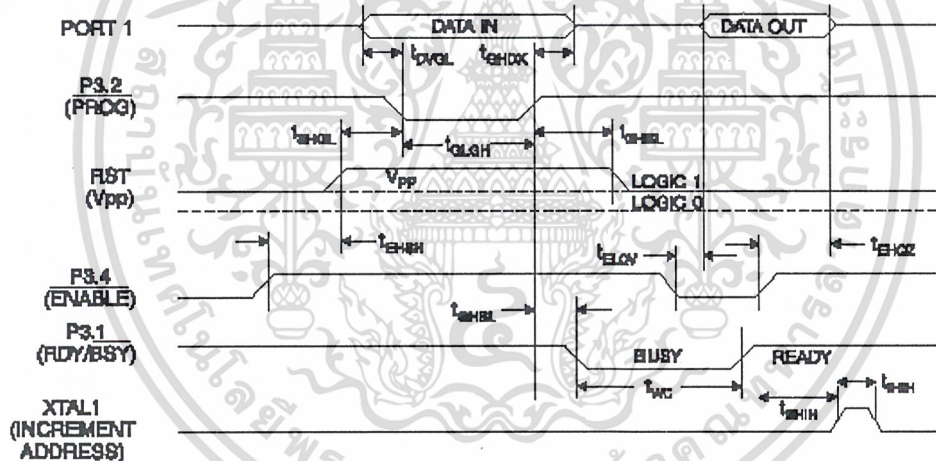
## Flash Programming and Verification Characteristics

$T_A = 0^\circ\text{C}$  to  $70^\circ\text{C}$ ,  $V_{CC} = 5.0 \pm 10\%$

Symbol	Parameter	Min	Max	Units
$V_{PP}$	Programming Enable Voltage	11.5	12.5	V
$I_{PP}$	Programming Enable Current		250	$\mu\text{A}$
$t_{DVGL}$	Data Setup to $\overline{\text{PROG}}$ Low	1.0		$\mu\text{s}$
$t_{GHDX}$	Data Hold after $\overline{\text{PROG}}$	1.0		$\mu\text{s}$
$t_{EHSH}$	P3.4 (ENABLE) High to $V_{PP}$	1.0		$\mu\text{s}$
$t_{SHGL}$	$V_{PP}$ Setup to $\overline{\text{PROG}}$ Low	10		$\mu\text{s}$
$t_{GHSI}$	$V_{PP}$ Hold after $\overline{\text{PROG}}$	10		$\mu\text{s}$
$t_{GLCH}$	$\overline{\text{PROG}}$ Width	1	110	$\mu\text{s}$
$t_{ELOV}$	$\overline{\text{ENABLE}}$ Low to Data Valid		1.0	$\mu\text{s}$
$t_{EFDZ}$	Data Float after $\overline{\text{ENABLE}}$	0	1.0	$\mu\text{s}$
$t_{GHBL}$	$\overline{\text{PROG}}$ High to $\overline{\text{BUSY}}$ Low		50	ns
$t_{WC}$	Byte Write Cycle Time		2.0	ms
$t_{EHH}$	$\text{RDY}/\overline{\text{BSY}}$ to Increment Clock Delay	1.0		$\mu\text{s}$
$t_{PHL}$	Increment Clock High	200		ns

Note: 1. Only used in 12-volt programming mode.

## Flash Programming and Verification Waveforms



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## Absolute Maximum Ratings\*

Operating Temperature .....	-55°C to +125°C
Storage Temperature .....	-65°C to +150°C
Voltage on Any Pin with Respect to Ground .....	-1.0V to +7.0V
Maximum Operating Voltage .....	6.6V
DC Output Current.....	25.0 mA

\*NOTICE: Stresses beyond those listed under "Absolute Maximum Ratings" may cause permanent damage to the device. This is a stress rating only and functional operation of the device at these or any other conditions beyond those indicated in the operational sections of this specification is not implied. Exposure to absolute maximum rating conditions for extended periods may affect device reliability.

## DC Characteristics

$T_A = -40^\circ\text{C}$  to  $85^\circ\text{C}$ ,  $V_{CC} = 2.0\text{V}$  to  $6.0\text{V}$  (unless otherwise noted)

Symbol	Parameter	Condition	Min	Max	Units
$V_{L}$	Input Low-voltage		-0.5	$0.2 V_{CC} - 0.1$	V
$V_{IH}$	Input High-voltage	(Except XTAL1, RST)	$0.2 V_{CC} + 0.9$	$V_{CC} + 0.5$	V
$V_{IH1}$	Input High-voltage	(XTAL1, RST)	$0.7 V_{CC}$	$V_{CC} + 0.5$	V
$V_{OL}$	Output Low-voltage <sup>(1)</sup> (Ports 1, 3)	$I_{OL} = 20\text{ mA}$ , $V_{CC} = 5\text{V}$ $I_{OL} = 10\text{ mA}$ , $V_{CC} = 2.7\text{V}$		0.5	V
$V_{OH}$	Output High-voltage (Ports 1, 3)	$I_{OH} = -80\ \mu\text{A}$ , $V_{CC} = 5\text{V} \pm 10\%$	2.4		V
		$I_{OH} = -30\ \mu\text{A}$	$0.75 V_{CC}$		V
		$I_{OH} = -12\ \mu\text{A}$	$0.9 V_{CC}$		V
$I_R$	Logical 0 Input Current (Ports 1, 3)	$V_{IN} = 0.45\text{V}$		-50	$\mu\text{A}$
$I_{TL}$	Logical 1 to 0 Transition Current (Ports 1, 3)	$V_{IN} = 2\text{V}$ , $V_{CC} = 5\text{V} \pm 10\%$		-750	$\mu\text{A}$
$I_{LI}$	Input Leakage Current (Port P1.0, P1.1)	$0 < V_{IN} < V_{CC}$		$\pm 10$	$\mu\text{A}$
$V_{OS}$	Comparator Input Offset Voltage	$V_{CC} = 5\text{V}$		20	mV
$V_{CM}$	Comparator Input Common Mode Voltage		0	$V_{CC}$	V
RRST	Reset Pull-down Resistor		50	300	$\text{K}\Omega$
$C_{IO}$	Pin Capacitance	Test Freq. = 1 MHz, $T_A = 25^\circ\text{C}$		10	pF
$I_{CC}$	Power Supply Current	Active Mode, 12 MHz, $V_{CC} = 6\text{V}/3\text{V}$		15/5.5	mA
		Idle Mode, 12 MHz, $V_{CC} = 6\text{V}/3\text{V}$ P1.0 & P1.1 = 0V or $V_{CC}$		5/1	mA
	Power-down Mode <sup>(2)</sup>	$V_{CC} = 6\text{V}$ P1.0 & P1.1 = 0V or $V_{CC}$		100	$\mu\text{A}$
		$V_{CC} = 3\text{V}$ P1.0 & P1.1 = 0V or $V_{CC}$		20	$\mu\text{A}$

Notes: 1. Under steady state (non-transient) conditions,  $I_{OL}$  must be externally limited as follows:

Maximum  $I_{OL}$  per port pin: 20 mA

Maximum total  $I_{OL}$  for all output pins: 80 mA

If  $I_{OL}$  exceeds the test condition,  $V_{OL}$  may exceed the related specification. Pins are not guaranteed to sink current greater than the listed test conditions.

2. Minimum  $V_{CC}$  for Power-down is 2V.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้