

ระบบตรวจสอบบ้านผ่านอินเทอร์เน็ต

AN INTERNET HOME MONITORING SYSTEM



ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต

สาขาวิชาอิเล็กทรอนิกส์

คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการศึกษาในปีการศึกษา 2546 ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

เลขหมู่.....  
เลขทะเบียน..... 55488

วัน,เดือน,ปี 10 พ.ค. 2548

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

ระบบตรวจสอบบ้านผ่านอินเทอร์เน็ต  
AN INTERNET HOME MONITORING SYSTEM



ปริญญาานิพนธ์สำหรับปริญญาวิศวกรรมศาสตรบัณฑิต  
สาขาวิชาอิเล็กทรอนิกส์  
คณะวิศวกรรมศาสตร์  
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง  
ปีการศึกษา 2546

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ปริญญานิพนธ์ ปีการศึกษา 2546

ภาควิชา อิเล็กทรอนิกส์

คณะ วิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

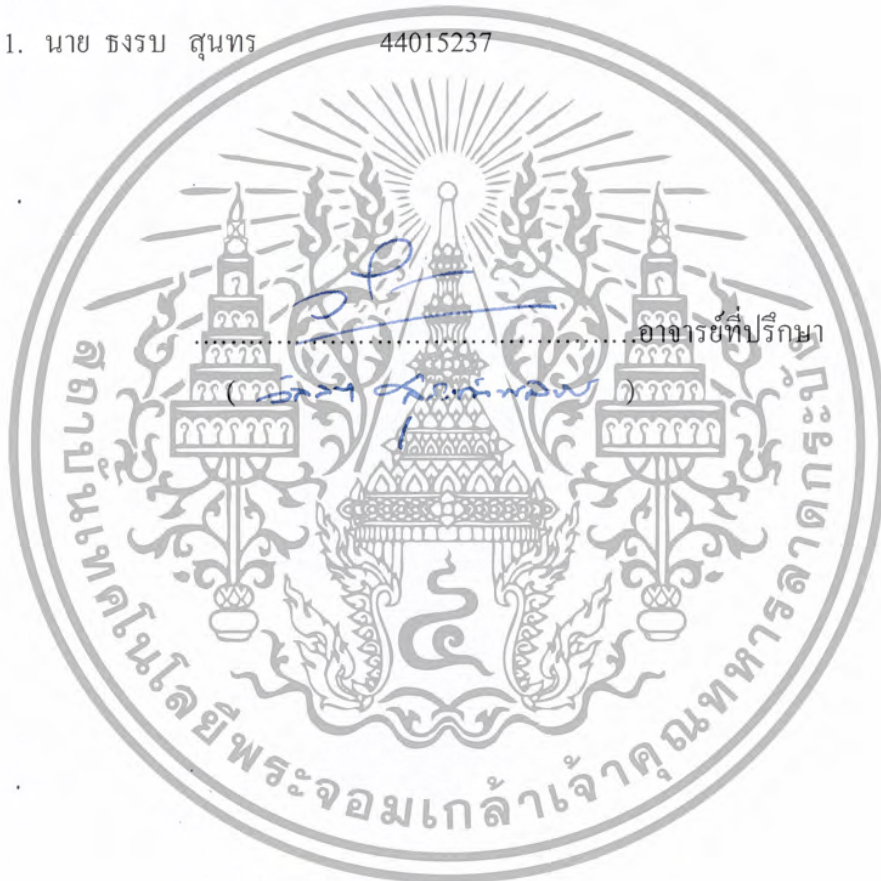
เรื่อง ระบบตรวจสอบบ้านผ่านอินเทอร์เน็ต

AN INTERNET HOME MONITORING SYSTEM

ผู้จัดทำ

1. นาย ชงรบ สุนทร

44015237



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## ระบบตรวจสอบบ้านผ่านอินเทอร์เน็ต

ธงรบ สุนทร

ศ.ดร.วัลลภ สุระคำพลธร อาจารย์ที่ปรึกษา

ปีการศึกษา 2546

### บทคัดย่อ

โครงการนี้เป็นการเขียนโปรแกรมประยุกต์ เพื่อใช้ในการสั่งงานเครื่องใช้ไฟฟ้าและรับค่าจาก เซนเซอร์ต่างๆ มาประมวลผลแล้วส่งค่าไปแสดงออกทาง คอมพิวเตอร์ โดยผ่านทางระบบอินเทอร์เน็ต ซึ่งจะแบ่งการทำงานออกเป็น 3 ส่วนทางด้านโปรแกรมคือ โปรแกรมโฮมเพจ โดยใช้ HTML ร่วมกับ เอเอสพี (ASP : Active Server Pages) โปรแกรมควบคุมอุปกรณ์ไฟฟ้าทางด้านเซิร์ฟเวอร์ โดยใช้ วินโดวส์ 2000 เซิร์ฟเวอร์ และ ไอไอเอส (IIS : Internet Information Server) ร่วมกับ วิชาลเมติก 6 โปรแกรมควบคุมไมโครคอนโทรลเลอร์ ในการพัฒนาทางด้านโปรแกรม ส่วนทางด้านฮาร์ดแวร์คือวงจรควบคุมต่างๆ โดยใช้ไมโครคอนโทรลเลอร์ MCS-51 ในการพัฒนา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

# AN INTERNET HOME MONITORING SYSTEM

THONGROB SOONTRON

Prof.Dr. WANLOB SURAKAMPONTON

Year Study 2003

## ABSTRACT

This report presents an applied software to control the electrical equipment. So receive from sensor to processing data and data send to computer display by Internet systems. The programs is divided into 3 parts, homepage program using HTML incorporating with ASP (Active Server Pages), electrical equipment controlled program at server and using Window 2000 Server and IIS (Internet Information Server) with Visual Basic 6, microcontroller controlled program. The programs are developed for using in this project. The hardware part is electrical equipment controlled circuit using MCS-51 to develop.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## สารบัญ

บทที่ 1 บทนำ	1
บทที่ 2 Visual Basic	2
2.1 ความเป็นมาของภาษา Basic	2
2.2 ขั้นตอนในการพัฒนาโปรแกรมของ Visual Basic	2
2.3 MS Winsock Control 6	3
2.3.1 ระบบเครือข่ายอินเทอร์เน็ต	3
2.3.2 Winsock Procedure	4
2.3.3 Winsock Properties & Events	5
2.4 MSComm Control	6
2.4.1 การเขียนโปรแกรมติดต่อกับ Serial Port	6
2.4.2 องค์ประกอบในการใช้ MSComm	6
2.4.3 MSComm Control Table	9
บทที่ 3 ไมโครคอนโทรลเลอร์ตระกูล MCS-51	15
3.1 คุณสมบัติของไมโครคอนโทรลเลอร์ในตระกูล MCS-51	16
3.2 โครงสร้างของไมโครคอนโทรลเลอร์ในตระกูล MCS-51	16
3.3 การจัดขาต่างๆของ MCS-51	17
3.4 ความถี่สัญญาณนาฬิกาบนชิป	19
3.5 Power Connection	20
3.6 โครงสร้างของหน่วยความจำใน MCS-51	20
3.7 การทำงานของ 8051	20
บทที่ 4 เว็บเซิร์ฟเวอร์ Web Server	22
4.1 เว็บเซิร์ฟเวอร์	22
4.1.1 เว็บเซิร์ฟเวอร์	22
4.1.2 ระบบเครือข่ายอินเทอร์เน็ต	23
4.1.3 เวิลด์ไวด์เว็บคืออะไร	24
4.1.4 โฮมเพจ	25
4.1.5 IP Address	25
4.1.6 ดีเอ็นเอส (Domain Name System : DNS)	26
4.1.7 URL	27
4.1.8 HTML	27

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## สารบัญ (ต่อ)

4.1.9 HTML Form และ CGT สคริปต์	27
4.1.10 Static และ Dynamic HTML	28
4.2 เอ เอส พี (Active Server Page : ASP)	28
4.2.1 เอ เอส พี (Active Server Page : ASP)	30
4.2.2 การเขียนสคริปต์ในรูปแบบของ เอ เอส พี	31
<b>บทที่ 5 การคำนวณและการสร้าง</b>	<b>33</b>
5.1 ส่วนฮาร์ดแวร์	33
5.1.1 วงจรเชื่อมต่อควบคุมอุปกรณ์ไฟฟ้า	33
5.1.2 วงจรตรวจจับอุณหภูมิ	34
5.1.2.1 ไอซี ตรวจจับอุณหภูมิ DS1820	34
5.1.3 วงจรควบคุมการหมุนของกัลดอง	34
5.1.4 วงจรตรวจสอบสถานะอุปกรณ์ไฟฟ้า	35
5.1.5 วงจรตรวจสอบสถานะประตู	36
5.2 ส่วนโปรแกรม	37
5.2.1 โปรแกรมควบคุมบน Server (ASP)	37
5.2.2 โปรแกรมควบคุมและแสดงผลบน Server (VB)	38
5.2.3 โปรแกรมส่วน Hardware	39
5.2.4 การเขียนโปรแกรมเพื่อใช้งาน DS 1820	40
5.2.5 โปรแกรมย่อยในส่วนต่างๆ	41
<b>บทที่ 6 สรุปผลและวิจารณ์</b>	<b>44</b>
6.1 การทดลองและผลการทดลอง	44
6.1.1 ผลการทดลองที่ Browser	44
6.1.2 ผลการทดลองที่ Server	47
6.2 ปัญหาในการทำโครงการ	50
6.3 วิธีแก้ไขปัญหา	50
6.4 แนวทางในการพัฒนาต่อไป	50
6.5 ข้อเสนอแนะเกี่ยวกับโครงการ	50

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## สารบัญ (ต่อ)

ภาคผนวก ก. โปรแกรมวิซวลเบสิก	51
ภาคผนวก ข. โปรแกรมสำหรับ MCS-51	82
ภาคผนวก ค. โปรแกรม ASP	94
ภาคผนวก ง. Data Sheet	107
กิตติกรรมประกาศ	122
หนังสืออ้างอิง	123



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## สารบัญรูป

รูปที่ 2.1 ระบบเครือข่ายคอมพิวเตอร์	4
รูปที่ 2.2 แสดงพอร์ตอนุกรม	14
รูปที่ 3.1 แสดง โครงสร้างภายในของ MCS-51	17
รูปที่ 3.2 แสดงขาต่างๆของ MCS-51	17
รูปที่ 3.3 ขาของ MCS-51 ที่ใช้ต่อกับ XTAL	19
รูปที่ 3.4 การจัดหน่วยความจำ	20
รูปที่ 4.1 แสดงภาพรวมของ เวลต์ ไวด์ เว็บ	22
รูปที่ 4.2 ขบวนการทำงานของ เอ เอส พี	29
รูปที่ 4.3 โครงสร้างของ เอ เอส พี	30
รูปที่ 5.1 แสดงวงจรควบคุมไฟ 220 โวลต์	33
รูปที่ 5.2 แสดงวงจรตรวจจับอุณหภูมิ	34
รูปที่ 5.3 DS 1820	34
รูปที่ 5.4 แสดงวงจรควบคุมมอเตอร์	35
รูปที่ 5.5 แสดงวงจรตรวจสอบสถานะอุปกรณ์ไฟฟ้า	36
รูปที่ 5.6 แสดงวงจรตรวจสอบสถานะประตู	36
รูปที่ 5.7 แสดง Flow Chart การทำงานของโปรแกรม ASP	39
รูปที่ 5.8 แสดง Flow Chart การทำงานของโปรแกรม ASP	39
รูปที่ 5.9 แสดง Flow Chart โปรแกรมควบคุมและแสดงผล VB	40
รูปที่ 5.10 แสดง Flow Chart โปรแกรมควบคุมและแสดงผล VB	41
รูปที่ 5.11 แสดง Flow Chart โปรแกรมการสั่งงาน	42
รูปที่ 5.12 แสดง Flow Chart โปรแกรมย่อย DS_1820_RD	43
รูปที่ 5.13 แสดง Flow Chart โปรแกรมย่อย DS 1820_WR	44
รูปที่ 5.14 แสดง Flow Chart โปรแกรมย่อย HE_XSEND	45
รูปที่ 6.1 แสดงหน้าแรกของโฮมเพจ	44
รูปที่ 6.2 แสดงการทำงานที่สำเร็จ	45
รูปที่ 6.3 แสดงการทำงานที่ไม่สำเร็จ	45
รูปที่ 6.4 แสดงผลอุณหภูมิและปุ่ม Refresh	46
รูปที่ 6.5 แสดงสถานะประตู	46

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## สารบัญรูป (ต่อ)

รูปที่ 6.6 แสดงปุ่มควบคุมกล้อง	46
รูปที่ 6.7 แสดงสถานะ Silent และปุ่มควบคุม	47
รูปที่ 6.8 แสดงการทำงานของโปรแกรม VB	48



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## สารบัญตาราง

ตารางที่ 2.1 Comm Event เกี่ยวกับการเกิดสถานะเมื่อเกิดการผิดพลาดในการสื่อสาร	9
ตารางที่ 2.2 Comm Event เกี่ยวกับการเกิดสถานะการเมื่อมีการสื่อสาร	10
ตารางที่ 2.3 Handshake property	10
ตารางที่ 2.4 Input mode property	10
ตารางที่ 2.5 MS Comm Control Property	11
ตารางที่ 2.6 รายละเอียดที่บ่งชี้ถึงความผิดพลาดในการใช้ MS Comm Control	12
ตารางที่ 3.1 แสดงบิตและหน้าที่ต่าง ๆ ของพอร์ท 3	18
ตารางที่ 4.1 ตัวอย่างไฟล์ เอ เอส พี	31
ตารางที่ 4.2 ตัวอย่างไฟล์ เอ เอส พี รวมกันทั้ง Jscript และ VBScript	31
ตารางที่ 6.1 แสดงผลการควบคุมกล้อง	47
ตารางที่ 6.2 แสดงผลการทดลองอุณหภูมิ	49
ตารางที่ 6.2 แสดงผลการทำงาน Motor เมื่อตั้งงาน	49



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## Visual Basic

## 2.1 ความเป็นมาของภาษา Basic

ภาษา BASIC ถูกสร้างขึ้นในปี 1963 โดย John Kemeny และ Thomas Kurtz ที่วิทยาลัย Dartmouth ในเบื้องต้นพวกเขามีจุดมุ่งหมายในการพัฒนาภาษา BASIC ขึ้น เพื่อใช้ในการสอนแนวในการเขียนโปรแกรม (Programming) โดยเน้นให้รูปแบบของภาษานั้นง่ายต่อการเข้าใจและการใช้งาน รวมทั้งทำงานในลักษณะของ Interpreter ซึ่งแตกต่างจากภาษาคอมพิวเตอร์อื่น ๆ ในยุคนั้นที่จะอาศัย Job Control Language (JCL) และขั้นตอนในการ Compile และ Link ผลก็คือภาษา BASIC ได้กลายเป็นภาษาคอมพิวเตอร์ที่นิยมใช้กันอย่างกว้างขวาง โดยเฉพาะในหมู่ผู้ใช้คอมพิวเตอร์ส่วนบุคคล จึงอาจกล่าวได้ว่า BASIC ได้รับการพัฒนาควบคู่ไปกับการพัฒนาคอมพิวเตอร์ส่วนบุคคล ในปี 1970 Microsoft ได้เริ่มผลิตตัวแปรภาษา BASIC ใน ROM ซึ่งเรียกว่า ROM-Based BASIC ขึ้นเช่น ชิพ Radio Sheek TRS-80 เป็นต้น ต่อมาได้พัฒนาเป็น GW-BASIC ซึ่งเป็น Interpreter ภาษาที่ใช้กับ MS-DOS และในปี 1982 Microsoft QuickBasic ได้รับการพัฒนาขึ้น โดยการเพิ่มความสามารถในการ Compile ให้เป็น Executed Program รวมทั้งทำให้ BASIC มีความเป็น “Structured Programming” มากขึ้น โดยการตัด Line Number ทิ้งไป เพื่อลบข้อกล่าวหาว่าเป็นภาษาคอมพิวเตอร์มีโครงสร้างในลักษณะ Spaghetti Code (Logical Flow ของภาษามาตรฐานโครงสร้าง) มาใช้รูปแบบของ Subprogram และ User Defined รวมทั้งการใช้ Structured Data Type และการพัฒนาการใช้งานด้านกราฟิกให้มีการใช้งานในระดับที่สูงขึ้น รวมทั้งมีการใช้เสียงประกอบได้ เหมือนกับภาษาคอมพิวเตอร์อื่นเช่น C หรือ Pascal

## 2.2 ขั้นตอนในการพัฒนาโปรแกรม ของ Visual Basic

ขั้นตอนในการพัฒนาโปรแกรม ประกอบด้วยขั้นตอนหลัก 2 ขั้นตอน ดังนี้  
 ขั้นตอนที่ 1 สร้างจอภาพของโปรแกรม

ในขั้นตอนนี้ จะนำ Form มาออกแบบเพื่อใช้ในการติดต่อกับผู้ใช้ หรือที่เรียกว่า การออกแบบ “User Interface” ในการพัฒนาโปรแกรมแบบเดิม ขั้นตอนนี้จะใช้เวลาและค่าใช้จ่ายสูง เนื่องจากจะต้องเขียนโปรแกรมเพื่อสร้างจอภาพต่าง ๆ จากนั้นต้อง Compile โปรแกรมนั้น แล้ว Run จึงจะเห็นจอภาพที่จัดทำขึ้น แต่สำหรับ Visual Basic ปัญหาในลักษณะนี้ได้ถูกแก้ไขโดยใช้เทคนิคของ Visualize ซึ่งเป็นความสามารถส่วนหนึ่งของ Visual Basic ขั้นตอนนี้ จึงสามารถทำได้ง่ายดาย เพียงแต่นำเอา Control ต่าง ๆ ใน Toolbox ที่ต้องการใช้งานมาวางไว้บน Form ซึ่งทำให้ประหยัดเวลาและสามารถเห็นลักษณะจอภาพที่ออกแบบได้ในขณะนั้นเลย

ขั้นตอนที่ 2 เขียนโปรแกรม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เมื่อวาง Control ต่าง ๆ ลงบน Form เป็นที่เรียบร้อยแล้ว(Control ต่าง ๆ เมื่อถูกนำมาวางไว้บน Form จะเรียกว่า"Object" )ขั้นตอนต่อมา ได้แก่ การเขียนโปรแกรมเพื่อกำหนดการทำงานให้กับแต่ละ Object ภายใต้เหตุการณ์ต่าง ๆ (Event) ที่จะเกิดขึ้นกับจอภาพนั้น ๆ

### 2.3 MS Winsock Control 6

ในการเขียน โปรแกรมให้สามารถติดต่อสื่อสารระบบเครือข่าย ถ้าเป็นเมื่อก่อนระบบปฏิบัติการ DOS นั้นคงยากทีเดียว แต่เดี๋ยวนี้เครื่องคอมพิวเตอร์อะไร ๆ ก็ Windows กันแล้วเนื่องจากมีเครื่องมือ (Tool) ที่ช่วยให้เราสามารถเขียน โปรแกรมติดต่อผ่านระบบเครือข่าย ที่เราจะกล่าวถึงก็คือ MS Winsock Control 6 เป็นเครื่องมือที่อำนวยความสะดวกสำหรับการเขียนโปรแกรมติดต่อผ่านระบบเครือข่าย โดย โปโตคอล TCP/IP

#### 2.3.1 ระบบเครือข่ายอินเทอร์เน็ต

ในปัจจุบันการสื่อสารผ่านระบบเครือข่าย (Network) รวมทั้งระบบ Internet เป็นที่แพร่หลายมากในแง่ของการใช้งาน, ใ้บริการ, อำนวยความสะดวก ในชีวิตประจำวันต่าง ๆ เนื่องจากระบบสื่อสารที่รวดเร็วและสามารถเชื่อมโยง เข้ากับหลาย ๆ ระบบได้ ทำให้สามารถนำมาประยุกต์ใช้งานผ่านระบบเครือข่ายได้หลายรูปแบบ

#### Tcp/ip

ระบบเครือข่ายเอาเป็นว่าใช้คำว่า Internet ก็แล้วกันที่ใช้โปร โดคอลมาตรฐานชื่อ TCP/IP ในการสื่อสารผ่านระบบติดต่อกับเครื่องคอมพิวเตอร์อื่น ๆ โปร โดคอล TCP/IP นั้นประกอบด้วยส่วนที่สำคัญ 2 ส่วนก็คือ

TCP(Transmission Control Protocol)

IP(Internet Protocol)

ในการติดต่อกันจริงๆแล้วเราจะไม่สามารถเห็นขั้นตอนการทำงานของระบบ ได้เพราะเป็นการทำงานของ Software&Hardware แต่เราจะอธิบายเพื่อความเข้าใจของ โปรโตคอล TCP/IP ให้ดูกันดังส่วนประกอบดังนี้

- IP Address : สำหรับการรับส่งข้อมูลในระบบ Internet จะถูกกำหนดและอ้างอิงด้วยหมายเลขประจำเครื่องนั้นก็เครื่อง IP Address ซึ่งในระบบ Internet จะมีเครื่องคอมพิวเตอร์เป็นจำนวนมากที่อยู่ในระบบที่ใช้ IP Address อาจจะไม่สะดวก จึงได้มีการเปลี่ยนมาใช้เป็น ชื่อ ในความเข้ากันก็คือ Domain name โดยทั้งหมดนี้อยู่ในระบบ Name Services ซึ่งเป็นการอ้างอิงชื่อแทนหมายเลขนั่นเอง
- Routing Configuration : ชื่อคือของโปร โดคอล TP/IP ก็คือในทางกำหนดเส้นทางสำหรับการรับส่งที่สามารถเลือกเส้นทางในการรับส่งข้อมูลได้อย่างอัตโนมัติหากถ้าเกิดเส้นทาง บางเส้นทางเสียหายระบบกลไกในการกำหนดเส้นทางสำหรับการรับส่งข้อมูลของ โปร โดคอล TP/IP ก็จะเลือกเส้นทางให้เหมาะสมถูกต้องให้สามารถรับส่งข้อมูลได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

-Protocol,Ports,Socketes: เป็นช่องทางสำหรับกำหนดทิศทางของการรับส่งข้อมูลนอกเหนือจากที่จะต้องกำหนดหลังจาก PI Address

### Server & Client

สิ่งที่เราจะพูดถึงในการเขียน โปรแกรมเพื่อใช้ในระบบเครือข่าย(Network)จุดหลักๆของระบบ จะแบ่งฝ่ายที่ต้องติดต่อรับส่งข้อมูลระหว่างกันออกเป็น 2 ส่วน คือ แม่ข่าย(Server)และลูกข่าย(Client) ซึ่งในการใช้งานจริงอาจมีส่วนประกอบอื่นๆอีก แต่เราจะขอไม่กล่าวถึงเนื่องจากต้องการให้เห็นภาพเข้าใจง่ายขึ้นจึงยกแค่ 2 ส่วนนี้มากล่าว

Server จะเป็นส่วนทำหน้าที่เสมือนกองอำนาจการ,ประชาสัมพันธ์,เมสเสจเจอร์ รวมถึงผู้จัด ให้กับระบบ, ลูกข่าย ที่จะเป็นส่วนร้องขอข้อมูลจาก Server โดย Server ในที่นี้จะเป็นส่วนที่เก็บข้อมูล,จัดการ,บริหาร,ข้อมูลหรือทรัพยากรระบบ เพื่อให้ฝ่ายลูกข่ายสามารถใช้บริการ

Server และ Client ต่างก็จะต้องมีตำแหน่งที่อยู่(IP Address),ช่องทางการติดต่อ(Port) โดยทั้งสองฝ่าย จะสามารถติดต่อกัน ได้จะต้องอยู่ในช่องทางเดียวกัน ซึ่งเราสามารถกำหนดหมายเลขของ Port ได้ ทั้งนี้ โปรแกรมที่ติดต่อกันจะต้องอ้างอิงหมายเลขของ Port ทุกครั้งเนื่องจากในระบบมีโปรแกรมมากมายที่กำลังติดต่อกันอยู่



รูปที่ 2.1 ระบบเครือข่ายคอมพิวเตอร์

### 2.3.2 Winsock Procedure

ในส่วนของ Windows Code Object-->Winsock มี Procedure สำหรับการติดต่อสื่อสารทั้งทางฝ่าย Server และ Client เราขอกล่าวอย่างนี้เพื่อความเข้าใจที่เหมือนกันดังจะกล่าวในที่นี้คือ-Close คือ เหตุการณ์เมื่อมีหยุดหรือยกเลิกการติดต่อสื่อสารของฝ่าย Server หรือ Client โดย Function Winsock.Close ซึ่งเราจะสามารถจะใช้ตรวจสอบฝ่ายตรงข้ามว่ามีการติดต่ออยู่หรือไม่ โดยอาจจะใส่ Message เตือนเป็นคั่น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

-Connect เป็นเหตุการณ์ที่ฝ่าย Client มีการส่งสัญญาณติดต่อกับมายัง Sever ส่งผลให้ Procedure นี้ได้เช่นกัน

-ConnectionRequest เป็นเหตุการณ์เมื่อฝ่าย Client ส่งสัญญาณติดต่อกับมายัง Server Procedure ส่วนนี้ก็จะทำงานพร้อมกับค่า requestID As Long ซึ่งเป็นหมายที่ Gen ขึ้นมาในระบบค่านั้นจะไม่เหมือนเดิม โดยจะให้ฝ่าย Server รับรู้ว่าใช้ ID จากคอนโทรลตัวใดเพื่อจะได้สื่อสารถูกต้อง

-DataArrival เหตุการณ์เกิดขึ้นเมื่อมีการส่งข้อมูลระหว่าง Server และ Client Porcedure นี้ก็จะทำงานขึ้นมา พร้อมกับค่าจำนวน bytesTotal As Long ที่รับเข้ามา

-Error เหตุการณ์ที่เกิดความผิดพลาดระหว่างการติดต่อสื่อสารระหว่าง Server และ Client โดยจะส่งค่า Number As Integer มาให้ว่าเป็นหมายเลขใดพร้อมทั้งรายละเอียดของการผิดพลาดในเหตุการณ์นั้นๆ คือ Description As String

-SendProgress จะเกิดขึ้นในขณะที่มีการส่งข้อมูลอยู่ เหตุการณ์นี้จะทำงานเมื่อเกิดข้อมูลหมดแล้วก็จะส่งผลทำให้เกิด Event SendComplete

-SendComplete เหตุการณ์เมื่อมีการส่งข้อมูลออกไปยังฝ่ายตรงข้ามเสร็จเรียบร้อยแล้ว

### 2.3.3 Winsock Properties & Events

-Accept (requestID) คือการตกลงกันระหว่าง Server และ Client ในการเลือกหมายเลข ID Control ให้ตรงกันเพื่อสามารถสื่อสารได้ถูกต้อง

-Close เป็นการส่งสัญญาณยกเลิกการติดต่อกัน จะเป็นฝ่าย Server หรือ Client ก็ได้ ที่จะใช้ Function นี้ จากนั้นจะทำให้ Procedure close ในฝ่ายตรงข้ามทำงาน

-Connectเป็นการส่งสัญญาณว่าตอนนี้ทำการติดต่อเรียบร้อยแล้ว ซึ่งจะส่งผลให้ Procedure ฝ่ายตรงข้ามทำงาน

-Getdata เป็นการรับข้อมูลเมื่อฝ่ายตรงข้ามส่งมา โดยประโยคคำสั่งนี้จะอยู่ในส่วนของ Procedure DataArrival เนื่องจากเป็นเหตุการณ์กระทำเมื่อฝ่ายตรงข้ามส่ง ข้อมูลเข้ามา

-listen การกระทำที่จะคอยตรวจสอบสัญญาณที่ส่งไม่ว่าฝ่ายตรงข้ามตอบรับการร้องขอการติดต่อ

-LocalHostName คำสั่งนี้จะส่งชื่อของ Computer name ของเครื่องนั้นๆ

```
Debug.Prinsock1.LocalHostName
```

-LocalIP คำสั่งนี้จะทำการส่งหมายเลข IP Address

```
Debug.Print Winsock1.LocalIP
```

-LocalPort คำสั่งจะส่งค่าของหมายเลขในการติดต่อ TCP/IP ของเครื่องนั้นๆ

```
Debug.Print Winsock1.LocalPort
```

-RemoteHost กำหนดหรือคืนค่าชื่อ Computer name ของเครื่องที่จะทำการติดต่อ

```
Winsock1.RemoteHost=MyServer
```

-RemoteHostIP กำหนดหมายเลข IP Address ของเครื่องที่จะทำการติดต่อ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Winsock1, RemoteHostIP=10.10.0.0

-RemoteHostPort กำหนดหมายเลข Port ที่จะใช้ในการติดต่อระหว่างกัน

Winsock1.RemoteHostIP=5000

-SocketHandle จะคืนค่าของช่องที่ใช้ในการติดต่อระหว่างกันซึ่งสามารถเรียกดูได้ดังนี้

Debug.Print Winsock1.SocketHandle

-State จะคืนค่าของสถานะของ Socket ขณะที่ใช้ติดต่อระหว่างอยู่ โดยอาจจะใช้ตรวจสอบสถานะ โดยค่าคงที่เหล่านี้เช่น sckClosed (มีค่า=0) Socket ปิดการใช้งานหรือ sckError(มีค่า=9)

Socket มีความผิดพลาดเกิดขึ้นเป็นต้น

## 2.4 Mscomm Control

### 2.4.1 การเขียนโปรแกรมติดต่อกับ Serial Port สามารถทำได้ 2 วิธี คือ

#### 1. การติดต่อแบบอินเทอร์รัพต์

ขบวนการอินเทอร์รัพต์ อุปกรณ์รอบข้างเกือบทุกชิ้นจะต้องปฏิบัติงานอยู่เพื่อส่งสัญญาณไปให้แก่ซีพียูเสมอ ถ้าอุปกรณ์นั้นพร้อมที่จะรับส่ง ที่เคยเจอจากการทำโครงการอุปกรณ์ จะส่งเป็นรหัสแอสกี เราจะเขียนโปรแกรมอินเทอร์รัพต์ โดยเมื่อที่ข้อมูลเข้ามาจะทำให้มี CommEvent กับ OnComm Event

#### 2. การติดต่อแบบโพลลิ่ง

ในระบบพีซี การโพลลิ่งมีบ้างที่ใช้การส่งผ่านข้อมูลระหว่าง Terminal กับ CPU กรณีข้อมูลเป็นประเภทไบนารีที่ส่งจากคีย์บอร์ด โดยวิธีการนี้จะตรวจสอบ คีย์บอร์ดว่ามีข้อมูลส่งมาหรือไม่ถ้าโดยจะตรวจสอบตลอดเวลา การทำงานกับข้อมูลที่ได้รับเข้ามาจะตรวจสอบด้วยความเร็วที่สูงกว่าอัตราความเร็วข้อมูลที่ส่งเข้ามาทาง คีย์บอร์ด กรณีที่ CPU ส่งสัญญาณออกไปตรวจสอบพบว่ามีข้อมูลที่ต้องส่งเข้ามาเรียกว่า "Wet Poll" ซึ่งจะเสียช่วงเวลา 90 เปอร์เซ็นต์ คำนวณเวลาที่เสียไปนั้น เราเลือกใช้เทคนิค การโพลแบบ "Round Robin" แทน แต่ใน VB เราจะใช้การตรวจสอบข้อมูลที่มาจาก Serial Port ตลอด โดยจะใช้ Control Timer เข้ามาช่วยในการเขียน โปรแกรมซึ่งสามารถตรวจได้ถึงระดับ 1 มิลลิวินาที หรือจะใช้ Do...Loop ก็ได้

ในตัวคอนโทรล MSComm มี Event ที่ใช้เพียง Event เดียวเท่านั้นเอง ก็คือ OnComm Event ซึ่งจะใช้ในการติดต่อแบบอินเทอร์รัพต์ การเขียนโปรแกรมติดต่อกับ Serial Port แบบธรรมดาจะใช้ comEvent เพียง comEvReceive, comEvSend ถ้าเป็นการติดต่อสื่อสารแบบ โมเด็มจะใช้หลายตัวในการตรวจสอบสัญญาณ รายละเอียดมีใน Help Visual Basic อยู่แล้ว

### 2.4.2 องค์ประกอบในการใช้ MSComm

การตั้งค่าติดต่อกับพอร์ต

-ComPort คือ เราต้องกำหนดหมายเลข Port ที่ใช้ต่อ RS-232(Com1,Com2) รายละเอียดดูในเมนู

ด้านซ้าย Serial Port

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- Setting คือ เราต้องกำหนดอัตรา Baud,Parity,Data(จำนวนบิต),Stop ตัวอย่าง 1200,n,8,1 เป็นต้น
- HandShaking คือเราจะกำหนดได้ 4 แบบ 1.comNone 2.comXXoff3. comRTS 4.comTRXonXoff

### การใช้ Buffer ในการรับส่งข้อมูล

- InBufferSize คือการกำหนด Buffer ในการรับข้อมูลเข้ามา
- OutBufferSize คือ การกำหนด Buffer ในการส่งข้อมูลออกไป
- Rthreshold คือ การที่เรากำหนดการเกิด Event-driven ในการรับข้อมูลเข้ามา
- Sthreshold คือ การที่เรากำหนดการเกิด Event-driven ในการรับข้อมูลออกไป
- Inputlen คือ จำนวนช่องข้อมูลที่อ่านไปใน Buffer รับข้อมูล
- EOFFenable คือ การที่บอกถึงสิ้นสุดของไฟล์(EOF) End of File ด้านฮาร์ดแวร์
- ParityReplace คือ ค่าของคาบเวลาที่แทนในเมื่อเกิด Parity Error
- NullDiscard คือ การกำหนดให้รับหรือไม่รับ NULL CHARACTER
- RTSEnable คือ ทำให้สัญญาณ RTS (Request To Send)
- DTSEnable คือ ทำให้สัญญาณ DTR(Data Terminal Ready)

กำหนดคุณสมบัติของ MSComm Control ให้สามารถติดต่อกับพอร์ตได้

1.Property ชื่อ CommPort คือ เลือกคอมพอร์ตที่เราจะต่อใช้งาน

ตัวอย่าง MSComm1.CommPort=1

ในที่นี้เลือกจะใช้ Com1 อยู่ที่ด้านหลังของเครื่องคอมฯ

2.Property ชื่อ Settings คือ การตั้งค่าของการรับส่งข้อมูล ซึ่งจะต้องรู้ด้วยว่าอัตราบอด ของอุปกรณ์ที่จะติดต่อด้วยเป็นเท่าไร โดยมีรายละเอียดการใส่ต่างๆดังนี้

MSComm1.Settings="Baub(อัตราการรับส่งข้อมูล),Parity(ถ้าไม่ใช่ใส่ N,จำนวนบิตข้อมูล,บิตสตอป")มาดูการเขียนโค้ดกัน

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

3.Property ชื่อ InputLen คือ กำหนดขนาดขนาดของที่มีข้อมูลเข้ามาให้ไปอ่านข้อมูลทั้งหมดที่อยู่ในบัฟเฟอร์ มาดูการเขียนโค้ดกัน

ตัวอย่าง MSComm1.Inputlen=1

4.Propertyชื่อ PortOpen คือ จะเปิดให้พอร์ตใช้งานหรือไม่ ถ้าเปิด =True ถ้าเปิด=False มาดูการเขียนโค้ดกัน

ตัวอย่าง MSComm1.PortOpen=True

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

ตัวอย่าง MSComm1.Rthreshold=1

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากรายละเอียดที่กล่าวมา เราจะมาเขียนใน โพรซีเจอร์ VB ซึ่งจะไว้ที่ Sub Form\_Load() หรือจะสร้าง Sub ขึ้นใหม่ในกรณีที่จะเรียกใช้ภายหลัง

```
Private Sub Form_Load()
```

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

```
MSComm1.CommPort=1
```

```
MSComm1.InputLen=1
```

```
MSComm1.PortOpen=True
```

```
MSComm1.Rthreshold=1
```

```
End Sub
```

### วิธีการรับส่งข้อมูลจาก Serial Port

จากวิธีเขียน โค้ดด้านบนเป็นการกำหนดค่าเบื้องต้นให้กับคอมพอร์ตและเปิดใช้การรับและส่งของพอร์ต RS-232 ดังนั้นก็สามารถรับและส่งข้อมูลทางพอร์ตได้ โดยใช้ Property ดังนี้

Output= ซึ่งจะเป็นการส่งข้อมูลไปที่พอร์ต

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

ตัวอย่าง

เช่นถ้าต้องการที่จะพิมพ์ข้อมูลส่งออกพร้อมๆกันขณะที่เราพิมพ์ไปด้วยคุณก็เพียง ไปเขียนโค้ดไว้ที่

Event KeyPress ของ Control TextBox ที่เราจะให้เป็นตัวส่งข้อมูลโดยเขียนดังนี้

```
Sub txtRXTX_KeyPress(KeyAscii As Integer)
```

```
MSComm1.Output=Chr$(KeyAscii)
```

```
End Sub
```

ส่วนการใช้ Property Input ต้องนำมาไว้ที่ EventOnComm ดังนี้

```
Private Sub MSComm1_OnComm()
```

```
Dim StrData As Variant'กำหนดชนิดตัวแปรเพราะต้องการให้เป็นอะไรก็ได้
```

```
Str=MSComm1.Input
```

```
Text1.Text=StrData
```

```
End Sub
```

### การใช้ Even ในโพรซีเจอร์ OnComm()

เราสามารถนำ Event ในตารางดังกล่าวมาใช้เขียนใน โพรซีเจอร์ OnComm() โดยการใช้เงื่อนไขแบบ

Select case ซึ่งสามารถเขียน ได้ดังนี้

```
Private Sub MSCmm1_OnComm()
```

```
Select Case MSComm1.CommEvent
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

Case comEvReceive
Dim Buffer As Variant
Buffer=MSCmm1.Input
ShowData txtRXTX,(StrConv((Buffer), vbUnicode))
Case comEvSend:'ส่วนนี้จะใส่เงื่อนไขให้ทำอะไรก็ได้ตามที่คุณวางคอนเซปไว้
Case comEvCTS
Case comEvDSR
Case comEvCD
Case comEvRingCase ComEvEOF
Case comBreak
Case comCDTO
Case comCTSTO
Case comDCB
Case comDSRTO
Case comFrame
Case comOverrun
Case comRxOver
Case comRxParity
Case comTxFull
End Select
End Sub
    
```



2.4.3 MSComm Control Table

ตารางที่ 2.1 CommEvent เกี่ยวกับการเกิดสถานะเมื่อเกิดการผิดพลาดในการสื่อสาร

ชื่อ Property	คำอธิบาย
ComEventBreak	การได้รับสัญญาณเบรก
ComEventCDTO	เมื่อเกิดไหม้เฮด ขณะที่กำลังคอยสัญญาณ CD(Carrier Detect)
ComEventCTSTO	เมื่อเกิดไหม้เฮด ขณะที่กำลังคอยสัญญาณ CTS(Carrier To Send)
ComEventDSTO	เมื่อเกิดไหม้เฮดขณะที่กำลังรอสัญญาณ DSR(Data Set Ready)
ComEventFrame	การที่เกิดความผิดพลาดทางเฟรม เป็นลักษณะที่พบบิดจบตามที่ควรจะเป็น
ComEventOverrun	การที่เกิดความผิดพลาด โอเวอร์รัน เป็นลักษณะที่รับข้อมูลไม่ทันในการประมวลผล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ComEventRxOver	บัฟเฟอร์ที่รับข้อมูลเกิด โอเวอร์โฟลล์ ก็คือรับตัวอักษรหลังจากการรับ EOF Char
ComEventRxPafty	การที่เกิดการผิดพลาดทางพาริตี เป็นลักษณะที่ตัวอักษรที่ได้รับ ได้มีพาริตีที่ไม่ถูกต้อง
ComEventTxFull	ตัวบัฟเฟอร์ที่ส่งข้อมูลเต็ม
ComEventDCB	การที่เกิดความผิดพลาดขึ้น โดยไม่ได้คาดถึง

### ตารางที่ 2.2 CommEvent เกี่ยวกับเกิดสถานะการเมื่อมีการสื่อสาร

ชื่อ Property	คำอธิบาย
ComEvCD	CD(Carriert Detect) เมื่อเปลี่ยนซึ่งคือสายของสัญญาณ Receive Line Signal Detect(RLSD)
ComEvCTS	RCTS(Carrier To Send)เมื่อมีการเปลี่ยนสถานะเกิดขึ้น
ComEvDSR	DSR(Data Set Ready) เมื่อมีการเปลี่ยนสถานะเกิดขึ้น
ComEvRing	เมื่อตรวจจับสัญญาณ Ring Indicator ได้
ComEvReceive	เมื่อได้รับข้อมูลเก็บลงใน InputBuffer
ComEvSend	เมื่อส่งข้อมูลออกจาก OutputBuffer
ComEvEof	เมื่อพบอักขระ EOF(End Of File)

### ตารางที่ 2.3 Handshake Property

ชื่อ Property	คำอธิบาย
ComNone	ไม่ใช้ให้ตรวจสอบแฮนเช็ก
ComXonZXoff	ให้มีการตรวจสอบแฮนเช็ก ในแบบ Xon/Xoff
ComRTS	ให้มีการตรวจสอบแฮนเช็ก ผ่านขา RTS และ CTS
ComRTSXOnXoff	กำหนดให้มีการตรวจสอบทั้ง 2 แบบคือ RTS-CTS และ Xon/Xo

### ตารางที่ 2.4 Input Mode Property

ชื่อ Properties	คำอธิบาย
ComInputModeText	คุณสมบัติในการรับข้อมูลเป็นแบบข้อความ ปกติจะเป็นค่านี้อยู่แล้ว
ComInputModeBinary	คุณสมบัติในการรับข้อมูลมาเป็นแบบไบนารีหรือเลขฐานสองนั่นเอง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## ตารางที่ 2.5 MSComm Control Property

ชื่อ Property	คำอธิบาย
Break	ในการที่เรากำหนดหรือเคลียร์สัญญาณเบรก
CDHolding	ตรวจสอบสัญญาณ Carrier Detect(CD)ดูว่ายังคงมีสถานะอยู่หรือเปล่า
CDTimeout	การกำหนดค่าหรือให้ค่าให้ค่าของเวลา(หน่วยmmSec) ที่รอสัญญาณ Carrier Detect
CommEvent	จะให้ผลของการเกิด Event ของ Commuication
CommID	จะให้ผลของการเสกเคิลของ Communication ที่เปิดใช้อยู่
CommPort	การกำหนดหรือว่าอ้างอิงของหมายเลขคอมพอร์ต ที่เปิดใช้อยู่เช่น Com1=1,Com2=2
CTSHolding	เป็นการตรวจสอบสัญญาณของClear To Send ว่าคงยังมีสถานะอยู่หรือเปล่า
CTSTimeout	การกำหนดค่าหรือให้ค่าของเวลา(หน่วย mmSec)ที่รอสัญญาณ Data Set Ready
DSRHolding	เป็นการตรวจสอบสัญญาณของ DataSet Ready ว่าคงมีสถานะอยู่หรือเปล่า
DSRTimeout	การกำหนดค่าหรือให้ค่าของเวลา(หน่วย mmSec)ที่รอสัญญาณ Clear To Send
DTREnable	ให้อินาเบิ้ล สายของสัญญาณ Data Terminal Ready(DTR)
Handshaking	กำหนดการแฮนเช็กทางฮาร์ดแวร์ เพื่อที่คอยตรวจสอบการรับส่งข้อมูล
InBufferCount	ให้ค่าของจำนวนข้อมูลที่อยู่ภายในบัฟเฟอร์รับข้อมูล
InBufferSize	กำหนดหรือว่าให้ค่าของขนาดในบัฟเฟอร์รับข้อมูล
Input	เป็นการให้ค่าหรือว่าเคลื่อนย้ายข้อมูลออกจากบัฟเฟอร์รับข้อมูล
InoputLen	การกำหนดหรือว่าให้ของจำนวนข้อมูลที่นำมาจากบัฟเฟอร์รับข้อมูล
Inteval	เป็นการกำหนดอัตราความเร็วของการใช้ใน โหมด โพลลิ่ง
NullDiscard	เป็นการกำหนดให้มีการรับ Null Character เก็บลงในบัฟเฟอร์รับข้อมูล
OutBufferCount	เป็นจำนวนข้อมูลที่อยู่ภายในบัฟเฟอร์ส่งข้อมูล
OutBufferSize	การกำหนดหรือว่าให้ค่าขนาดของบัฟเฟอร์ส่งข้อมูล
Output	เป็นการส่งข้อมูลให้กับบัฟเฟอร์ส่งข้อมูลเพื่อทำการส่งข้อมูลออก
ParityReplace	เป็นการกำหนดให้ส่งอักขระที่กำหนดนี้แทนหากเกิดการผิดพลาดในข้อมูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ชื่อ Property	คำอธิบาย
PortOpen	เป็นการกำหนดหรือทำให้ค่าของสถานะพอร์ตว่าเปิดหรือปิดอยู่
Rthrehold	การกำหนดหรือทำให้ค่าของจำนวนข้อมูลที่เก็บลงในบัฟเฟอร์รับข้อมูลก่อนการเกิด CommEvent ในการรับข้อมูล
RTSnable	ให้อินาเบิล สัญญาณ Request To send(RTS)
SettIngs	เป็นการกำหนดอัตราบอด พาริตี ข้อมูล บิตหยุด
Sthreshold	การกำหนดหรือทำให้ผลของจำนวนข้อมูลที่เก็บลงในของบัฟเฟอร์ส่งข้อมูลก่อนการเกิด CommEvent ในการที่ส่งข้อมูล

ตารางที่ 2.6 รายละเอียดที่บ่งชี้ถึงความผิดพลาดในการใช้ MS Comm Control

ชื่อ Property	ค่าตัว เลข	คำอธิบาย
ComInvalid Property Value	380	ตั้งค่าไม่ถูกต้อง
ComSetNot Supprted	383	กำหนดค่าที่ตั้งไว้สามารถอ่านได้อย่างเดียว เขียนหรือเปลี่ยนข้อมูลไม่ได้
ComGetNot Supported	394	กำหนดค่าที่รับไว้สามารถอ่านได้อย่างเดียว เขียนหรือเปลี่ยนข้อมูลไม่ได้
ComPortOpen	8000	จะอ่านค่าไม่ได้ในขณะที่ Port นั้นยังถูกเปิดใช้อยู่
ComPortOpen	8001	ค่าของเวลาที่หาออกมาได้ต้องมีค่ามากกว่าศูนย์
ComPortOpen	8002	กำหนดหมายเลข Port ไม่ถูกต้อง
ComPortOpen	8003	ผลลัพธ์ของข้อมูลจะเกิดในขณะที่มีการทำงาน
ComPortOpen	8004	Port นั้นจะสามารถอ่านค่าได้ในขณะที่มีการทำงานเท่านั้น
ComPort AlreadyOpen	8005	Port ได้ถูกเปิดไว้เรียบร้อยแล้ว
ComPort AlreadyOpen	8006	อุปกรณ์เกิดการผิดพลาดหรือไม่สามารถรองรับค่าได้
ComPort AlreadyOpen	8007	อุปกรณ์ไม่ยอมรับค่าที่ Baud Rate ถูกตั้งเอาไว้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ชื่อ Property	ค่าตัว เลข	คำอธิบาย
ComPort AlreadyOpe	8008	ขนาดของข้อมูลผิดพลาด
ComPort AlreadyOpen	8009	ค่าของตัวแปรที่แสดงอยู่ผิดพลาด
ComPort AlreadyOpen	8010	อุปกรณ์ภายนอก(Hardware)ยังไม่พร้อมที่จะทำงาน
ComPort AlreadyOpen	8011	ฟังก์ชันไม่สามารถกำหนดแถวข้อมูลได้
ComNoOpen	8012	Com Port ยังไม่พร้อมที่จะถูกเปิดใช้งาน
ComNoOpen	8013	Com Port พร้อมที่จะถูกเปิดใช้งาน
ComNoOpen	8014	Com Port ไม่สามารถทำงานได้
ComSetComm StateFailed	8015	ไม่สามารถตั้งค่าสถานะของPort ได้
ComSetComm StateFsiled	8016	ไม่สามารถ Set Port ตามเหตุการณ์ที่กำหนดให้ได้
ComPortNotOpen	8018	จะสามารถหาผลลัพธ์ของข้อมูลได้ก็ต่อเมื่อ Port มีการทำงานแล้วเท่านั้น
ComPortOpen	8019	Port ไม่มีที่ว่างมีข้อมูลเต็มใน Port
ComReadError	8020	เกิดความผิดพลาดขึ้นขณะที่อ่าน
ComDCBError	8021	เกิดความผิดพลาดภายในต้องไปแก้ไขที่ควบคุม Port

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

**Serial Port, RS-232 (Com1, Com2)**

- Signal Ground..... GND → 5
- Ring Indicator..... RI IN → 9
- Data Terminal Ready..... DTR OUT → 4
- Clear To Send..... CTS IN → 8
- Transmit Data..... TXD OUT → 3
- Request To Send..... RTS OUT → 7
- Receive Data..... RXD IN → 2
- Data Set Ready..... DSR IN → 6
- Carrier Detect..... DCD IN → 1

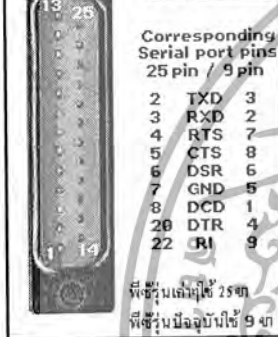
**9 ขา ตัวเมีย  
ชนิด AT**



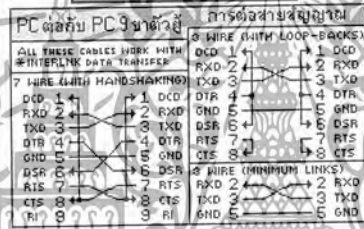
**รายละเอียดสัญญาณลักษณะสัญญาณ**

- RI "Ring Indicator" พาว์จัมสัญญาณของโมเด็มให้พีซีเตรียมพร้อม
- DTR "Data Terminal Ready" สัญญาณจากโมเด็มให้พีซีเตรียมพร้อม
- CTS "Clear To Send" ตรวจสอบสัญญาณโมเด็มว่าพร้อมจะรับข้อมูลจากพีซี
- TXD "Transmit Data" ส่งข้อมูลที่ละบิตโดยเป็นลำดับไปทีละบิต
- RTS "Request To Send" สัญญาณจากโมเด็มให้พีซีพร้อมที่จะส่งข้อมูล
- RXD "Receive Data" รับข้อมูลที่ละบิตโดยเป็นลำดับมาจากโมเด็ม
- DSR "Data Set Ready" สัญญาณบอกว่าโมเด็มพร้อมทำงานแล้ว
- DCD เมื่อเราตรวจสอบสัญญาณที่ปลายทางของสายสัญญาณ จะทำให้สายสัญญาณ Active

**25 ขา  
ชนิดตัวเมีย  
ชนิด PC**



COM1	
I/O ADDRESS	3F8H-3FFH
INTERRUPT	IRQ4
COM2	
I/O ADDRESS	2F8H-2FFH
INTERRUPT	IRQ3



**สายโมเด็ม ต่อกับ คอมพิวเตอร์ "Straight-Through"**

25 PIN MALE FOR MODEM CON-NECTION (DTE)	2-2	25 PIN FEMALE FOR PC CON-NECTION (DTE)	2-3
	3-3		3-2
	4-4		4-7
	5-5		5-8
	6-6		6-6
	7-7		7-5
	8-8		8-1
	20-20		20-4
	22-22		22-9

<http://Thaibit.hypermart.net>

รูปที่ 2.2 แสดงพอร์ตคอมพิวเตอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## ไมโครคอนโทรลเลอร์ตระกูล MCS-51

คุณสมบัติของ MCS-51 คุณสมบัติที่สำคัญๆของชิปไมโครคอนโทรลเลอร์ตระกูล MCS-51 มีดังนี้

-ต้องการแหล่งจ่ายไฟ 5 โวลต์เพียงชุดเดียว

-มีหน่วยความจำสำหรับเก็บ โปรแกรมควบคุมการทำงานอยู่ภายในชิป จำนวน 4 กิโลไบต์ (เบอร์ 8031,8032 ไม่มีหน่วยความจำส่วนนี้ ส่วนเบอร์ 8052 มีหน่วยความจำส่วนนี้ 8 กิโลไบต์และสำหรับเบอร์ 83C51FB จะมีหน่วยความจำส่วนนี้รวมทั้งสิ้น 16 กิโลไบต์)

-มีหน่วยความจำสำหรับ โปรแกรมและข้อมูล (RAM) อยู่ภายในชิป จำนวน 128 ไบต์ (ในเบอร์ 8031,8051) หรือ 256 ไบต์(ในเบอร์ 8031,8052)

-สามารถใช้หน่วยความจำสำหรับ โปรแกรมและข้อมูลที่อยู่ภายนอกชิปได้ อย่างละ 64 กิโลไบต์แยกจากกัน

-คำสั่งส่วนใหญ่ใช้การทำงานเพียง 1 ไมโครวินาที เมื่อใช้คริสตอลความถี่ 12 แมกกะเฮิรตซ์

-มีพอร์ตที่สามารถรับและส่ง ข้อมูลได้ทั้งสองทิศทางจำนวน 4 พอร์ตๆละ 8 บิต หรือสามารถใช้งานเป็นพอร์ตขนาด 1 บิตแยกจากกัน ทำให้เสมือนมีพอร์ตขนาด 1 บิต ใช้งานรวมกันทั้งสิ้น 32 พอร์ต

-รับข้อมูลและส่งข้อมูลได้ในตัว โดยสามารถกำหนดอัตราเร็วในการรับและส่งข้อมูล (Baud Rate) ได้ตั้งแต่ 300 ถึง 375 กิโลบิตต่อวินาที

-จัดลำดับความสำคัญของสัญญาณอินเทอร์รัปต์ได้ สองระดับ

-มีรีจิสเตอร์สำหรับ ใช้งานเป็น ไทม์เมอร์หรือเคาน์เตอร์ เพื่อนับสัญญาณนาฬิกาภายในชิป หรือนับเปลี่ยนสถานะของสัญญาณภายนอกขนาด 16 บิตจำนวน 2 ตัว เพื่อใช้งานสำหรับนับจำนวนพัลส์วัดความกว้างของพัลส์หรือไจค์ช่วงเวลา (ในเบอร์ 8052 จะมี 3 ตัว)

-หน่วยความจำสำหรับเก็บข้อมูลภายในบางส่วน สามารถเข้าถึงข้อมูลได้ทั้งระดับบิตเพื่อให้การออกแบบ โปรแกรมและการควบคุมระบบทำได้ง่ายขึ้น

-มีคำสั่งคูณและหารขนาด 8 บิตในตัวเอง

-สามารถประมวลผลแบบบูลีนเพื่อ ใช้งานควบคุม โดยเฉพาะ

-ใช้โปรแกรมของไมโครคอนโทรลเลอร์ตระกูล MCS-48(upwardly compatible)ได้ ไมโครคอนโทรลเลอร์ตระกูล MCS-51 เบอร์ที่จัดว่าเป็นเบอร์พื้นฐานในตระกูลนี้ คือเบอร์ 8051,8751 และ 8031 ซึ่งมีจำนวนขาภายนอก 40 ขาเหมือนกันใช้เวลาในการปฏิบัติคำสั่งแต่ละคำสั่งเท่ากันใช้แรงดันไฟเท่ากันสิ่งที่แตกต่างกันระหว่างสามเบอร์นี้คือขนาดของหน่วยความจำสำหรับเก็บในชิปมี

ไม่เท่ากัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 3.1 คุณสมบัติของไมโครคอนโทรลเลอร์ในตระกูล MCS-51

เบอร์ 8751 มีหน่วยความจำสำหรับเก็บ โปรแกรมภายในชิปเป็น EPROM ขนาด 4 กิโลไบต์ ซึ่งทำการลบข้อมูลเดิมแล้วเขียนซ้ำใหม่ได้

เบอร์ 8051 หลังการทดสอบ โปรแกรมจนไม่พบข้อผิดพลาดแล้วจะเป็นช่วงของการผลิตจริงซึ่งต้องคำนึงถึงต้นทุนเป็นอันดับแรกในการผลิตจริงจะใช้ เบอร์ 8051 ซึ่งมีหน่วยความจำภายในเป็นแบบ ROM ขนาด 4 กิโลไบต์แทนเพราะราคาต่ำกว่า แต่มีข้อจำกัดที่ไม่สามารถทำการแก้ไข โปรแกรมที่บันทึกไปแล้วได้ไม่ว่าด้วยวิธีใดก็ตาม

เบอร์ 8031 เบอร์นี้จะไม่มีหน่วยความจำสำหรับเก็บ โปรแกรมภายในชิป แต่สามารถใช้ หน่วยความจำภายนอกเพื่อเก็บ โปรแกรมได้มากถึง 64 กิโลไบต์ซึ่งอาจเป็น ROM,PROM หรือ EPROM ตามความต้องการของผู้ใช้

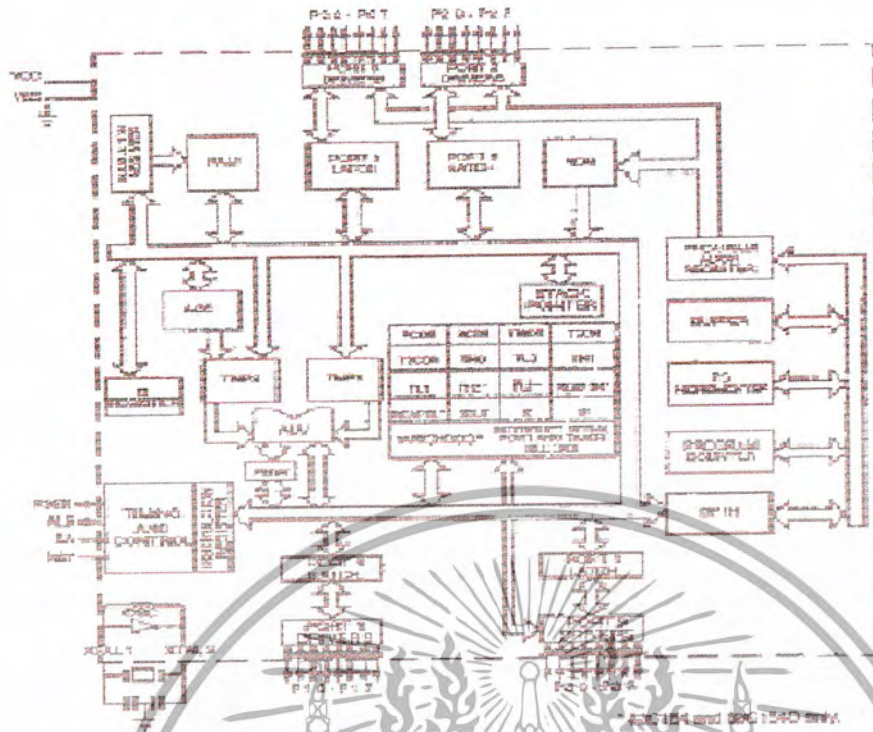
เบอร์ 8751 และ 8051 จะใช้โปรแกรมจากหน่วยความจำภายนอกได้เอง เมื่อมีโปรแกรม จากหน่วยความจำที่ยาวเกิน 4 กิโลไบต์ หรืออาจบังคับให้ไมโครคอนโทรลเลอร์ทั้งสองเบอร์ใช้ โปรแกรมจากหน่วยความจำภายนอกเพียงอย่างเดียวด้วยการต่อขา 31 (EA) ลงกราวด์ทำให้มีคุณสมบัติเหมือนกับเบอร์ 8031

### 3.2 โครงสร้างของไมโครคอนโทรลเลอร์ในตระกูล MCS-51

ไมโครคอนโทรลเลอร์ตระกูล MCS-51 มีสมาชิกในตระกูลหลายเบอร์ด้วยกัน แต่ละเบอร์จะมีคุณสมบัติพิเศษบางอย่างแตกต่างกัน เช่น มีหน่วยความจำภายในสำหรับเก็บ โปรแกรมและข้อมูล ภายในชิปเพิ่มขึ้น มีวงจรเปลี่ยนค่าสัญญาณอนาล็อกเป็นดิจิทัลในตัว สามารถรับสัญญาณอินเตอร์รัปต์ได้หลายชนิด ทำกระบวนการ DMA (Direct memory Access) ได้ในตัว มีรีจิสเตอร์สำหรับใช้ เป็นไทม์เมอร์หรือเคาน์เตอร์เพิ่มขึ้นคุณสมบัติพิเศษที่แตกต่างกันของ ไมโครคอนโทรลเลอร์แต่ละ เบอร์ในตระกูลนี้ดังแสดงในตารางที่ผ่านมา

ไมโครคอนโทรลเลอร์เบอร์นี้นับได้ว่าเป็นเบอร์พื้นฐานสำหรับตระกูล MCS-51 นี้ ได้แก่ เบอร์ 8051, 8031, 8751 โดยเบอร์ 8051 จัดเป็นสมาชิกตัวแรกในตระกูล ซึ่งมีหน่วยความจำสำหรับ เก็บโปรแกรมภายในชิปเป็น ROM ขนาด 4 กิโลไบต์ และหน่วยความจำสำหรับเก็บข้อมูลทั่วไป ภายใน MCS-51(RAM) เองจำนวน 128 ไบต์ มีพอร์ตขนาด 8 บิต 4 พอร์ต มีรีจิสเตอร์สำหรับใช้ เป็นไทม์เมอร์หรือเคาน์เตอร์ขนาด 16 บิต 2 ตัว รับสัญญาณอินเตอร์รัปต์ภายนอกได้ 2 ชนิด สามารถรับและส่งข้อมูลแบบอนุกรมผ่านพอร์ตสื่อสารข้อมูลแบบอนุกรม มีวงจรออสซิลเลเตอร์ เพื่อสร้างสัญญาณนาฬิกาควบคุมการทำงานในตัวเอง ส่วนเบอร์ 8751 จะมีคุณสมบัติเหมือนเบอร์ 8051 ทุกอย่างต่างกันเพียงชนิดของหน่วยความจำสำหรับเก็บโปรแกรมภายในชิป คือ เบอร์ 8751 จะเป็น EPROM แทนที่จะเป็น ROM ส่วนเบอร์ 8031 จะเหมือนกับเบอร์ 8051 ต่างกันเพียงในเบอร์ 8031 ไม่มีหน่วยความจำสำหรับเก็บ โปรแกรมภายในชิปเท่านั้น

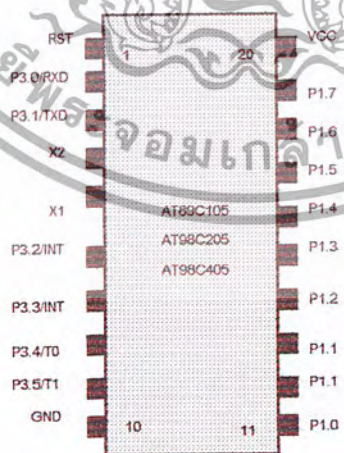
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.1 แสดงโครงสร้างภายในของ MCS-51

3.3 การจัดขาต่างๆของ MCS-51

ไมโครคอนโทรลเลอร์ตระกูล MCS-51 ทุกเบอร์จะมีตำแหน่งขาพื้นฐานที่เหมือนกัน โครงสร้าง IC เป็นแบบ DIP มีขาทั้งหมด 40 ขา โดยขาต่างๆจะใช้เป็น ขอร้ออินพุต,เอาต์พุต ขาสัญญาณควบคุม,ขาตำแหน่งหน่วยความจำ และขาข้อมูลดังรูปที่ 3.2



รูปที่ 3.2 แสดงขาต่างๆ ของ MCS-51

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ความหมายของขาต่างๆมีดังนี้

#### 1.พอร์ท 0 (PORT 0)

พอร์ท 0 ได้แก่ขาที่ 32-39 ของ MCS-51 สามารถใช้เป็นอินพุตเอาต์ ไลน์นอกจากนี้ในการติดต่อกับหน่วยความจำภายนอกยังใช้เป็นขา Address Bus และ Data Bus อีกด้วย

#### 2.พอร์ท 1 (PORT 1)

พอร์ท 1 ได้แก่ขาที่ 1-8 เป็นพอร์ท 8 บิต สามารถอ้างที่ละบิตได้ คือ P1.0,P1.1....เป็นต้น

#### 3.พอร์ท 2 (PORT 2)

พอร์ท 2 ได้แก่ขาที่ 21-28 จะใช้งาน 2 หน้าทีคือเป็นพอร์ท 8 บิตกับเป็นขาแอดเดรส 8 บิต

ในการอ้างหน่วยความจำภายนอก

#### 4.พอร์ท 3 (PORT 3)

พอร์ท 3 ได้แก่ขาที่ 10-17 จะใช้งาน 2 หน้าทีคือ เป็นพอร์ทอินพุต และเอาพุต และใช้เป็นขาควบคุมต่างๆดังตารางที่ 3.1

บิต	ชื่อ	หน้าที่พิเศษ
P3.0	RXD	ได้รับข้อมูลทางพอร์ทอนุกรม
P3.1	TXD	ส่งข้อมูลทางพอร์ทอนุกรม
P3.2	INT0	อินเตอร์รัปต์ภายนอกหมายเลข 0
P3.3	INT1	อินเตอร์รัปต์ภายนอกหมายเลข 1
P3.4	T0	ตัวจับเวลา/ตัวนับตัวที่ 0
P3.5	T1	ตัวจับเวลา/ตัวนับตัวที่ 1
P3.6	WR	สัญญาณเขียนข้อมูลหน่วยความจำภายนอก
P3.7	RD	สัญญาณอ่านข้อมูลหน่วยความจำภายนอก

ตารางที่ 3.1 แสดงบิตและหน้าที่ต่างๆของพอร์ท 3

#### 5.PSEN (Program Store Enable)

ขา PSEN เป็นขาที่ส่งสัญญาณออกคือขา 29 ขานี้จะแอกทีฟเมื่อ MCS-51 ต้องการอ่าน CODE โปรแกรมภายนอก โดยปกติถ้าหน่วยความจำภายนอกเป็น EPROM ขา PSEN จะต่อกับขา Output Enable (OE) ของ EPROM

#### 6.ALE (Address Latch Enable)

เนื่องจากพอร์ท 0 สามารถใช้เป็นขาอ้างตำแหน่ง และขาข้อมูล MCS-51 จะมีขา ALE ได้แก่ขา 30 ขานี้จะใช้ Multiplex สัญญาณ Address Bus ของ Port 0 ในการใช้งานระบบ MCS-51

ต้องการติดต่อกับหน่วยความจำภายนอก MCS-51 จะส่งสัญญาณ Address Bus ออกมาก่อนทาง

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Port 0 จากนั้นจะส่งสัญญาณ ALE มา Latch อุปกรณ์ ภายนอก ให้เก็บค่า Address Bus ของ Port 0 ไว้เพื่อใช้ Port 0 เป็น Data Bus ต่อไป

#### 7.EA (External Access)

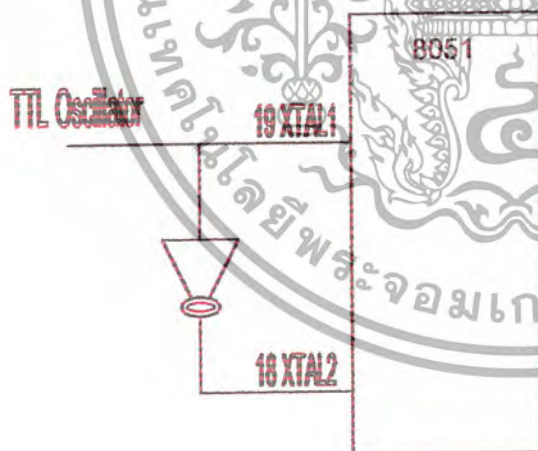
ขา EA ได้แก่ขาที่ 31 ถ้าขานี้เป็นลอจิก “1” จะใช้เบอร์ 8051/8052 เพื่อบอกว่าให้อ่านโปรแกรมจากหน่วยความจำโปรแกรมภายใน แต่ถ้าเป็นลอจิก “0” จะบอกให้ MCS-51 ทำโปรแกรมโดยอ่านจากหน่วยความจำโปรแกรมภายนอก (ถ้าขา EA เป็น “0” ขา PSEN จะแอกทีฟ) ถ้าหากเบอร์ 8031 หรือ 8032 ขา EA จะเป็น “0” เสมอเพราะว่าไม่มีหน่วยความจำโปรแกรมภายใน แต่ถ้าใช้เบอร์ 8051/8052 ซึ่งมีหน่วยความจำโปรแกรมภายในและให้ขา EA เป็น “0” ซึ่งจะ Disable ROM ภายในและจะอ่านโปรแกรมจาก EPROM ภายนอกแทน

#### 8.RST (Reset)

ขา RST ได้แก่ขา 9 จะใช้ในการรีเซ็ต MCS-51 โดยจะให้ขานี้เป็นลอจิก “1” อย่างน้อย 2 Machine Cycles จึงจะรีเซ็ตระบบได้

#### 3.4 ความถี่สัญญาณนาฬิกาบนชิป

เป็นวงจร Oscillator บนชิป ได้แก่ขา 18-19 โดยต่อ Crystal เข้ากับขานี้ โดยปกติมักจะใช้ Crystal ความถี่ 12 MHz กับตัวเก็บประจุหรืออาจใช้สัญญาณนาฬิกาจาก TTL Clock Source ต่อกับ XTAL1 และ XTAL2 ดังรูปที่ 3.3



รูปที่ 3.3 ขาของ MCS-51 ที่เชื่อมต่อกับ XTAL

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



เช่นกัน ผู้ใช้จะเขียนโปรแกรมเป็นภาษาเครื่อง ซึ่งอยู่ในรูปแบบเลขฐาน 2 เก็บไว้ในหน่วยความจำประเภท Program memory แต่ละคำสั่งของ 8051 อาจประกอบด้วย 1,2 หรือ 3 ไบต์แล้วแต่ว่าจะเป็นคำสั่งให้ทำอะไรคอมพิวเตอร์ก็เหมือนคนที่จะต้องทำงานตามคำสั่งเมื่อรับคำสั่ง แล้วก็จะไปทำตามคำสั่งเมื่อรับคำสั่งแล้วก็จะไปทำตามคำสั่งนั้นเสร็จสิ้นแล้วก็จะมารับคำสั่งต่อไป

เมื่อเริ่มป้อนไฟให้กับ 8051 ซึ่งมีวงจร Power On Set ต่ออยู่จะมีกรีเซตเกิดขึ้นการทำงานภายใน 8051 จะเริ่มจากบล็อก Program counter ซึ่งเป็นวงจรนับ (Counter Circuit) ชนิดหนึ่งส่งค่าตำแหน่งหน่วยความจำสำหรับโปรแกรมลงไปบนบัส (Bus) หมายเลข 1 บิตนี้มีขนาด 16 บิต ค่าตำแหน่งหน่วยความจำนี้จะถูกเก็บไว้ที่ Program ADDR Register ที่เป็นวงจร Latch ข้อมูลซึ่งเป็นค่าตำแหน่งหน่วยความจำจะปรากฏที่บนบัส 16 บิตหมายเลข 2 ถ้าเป็นค่าตำแหน่งหน่วยความจำแรกหลังจากกรีเซตค่าตำแหน่งหน่วยความจำจะเป็น 0000H หน่วยความจำสำหรับโปรแกรมจะเลือกได้ว่าเป็น ROM ภายในหรือภายนอก 8051 โดยการป้อนสถานะลอจิกเข้าไปที่ 8051 ทางขา EA ซึ่งต่ออยู่กับส่วน Timing and control ทำหน้าที่เป็นวงจรถอดรหัส (Decoder) แล้วสร้างสัญญาณควบคุมต่อไปถ้าป้อนสัญญาณลอจิก 0 เข้าไปที่ขา EA จะเป็นการเลือกให้ ROM ภายใน 8051 โดยที่วงจร Timing And Control จะสร้างสัญญาณไปยัง ROM ภายในให้ส่งข้อมูลเป็นคำสั่งจากตำแหน่งที่ถูกชี้ด้วยค่าตำแหน่งที่ถูกชี้มาทางบัสหมายเลข 2 ข้อมูลจาก ROM จะถูกส่งลงไปยังบัสหมายเลข 3 ที่เรียกว่า Internal Data Bas แล้วนำไปเก็บไว้ที่ Instruction Register (เป็นวงจร Latch) เพื่อส่งต่อไปให้กับวงจร Timing And Control เพื่อทำการถอดรหัสแล้วทำการควบคุมส่วนอื่นๆต่อไปแล้วแต่จะเป็นคำสั่งให้ทำงานอะไร ในกรณีที่เลือก ROM ภายนอก 8051 โดยป้อนสัญญาณลอจิก 1 เข้าไปที่ขา EA จะทำให้วงจร Timing And Control ส่งสัญญาณไปยังพอร์ต 0 และ พอร์ต 2 เพื่อส่งค่าตำแหน่งหน่วยความจำบนบัสหมายเลข 2 ออกไปยังหน่วยความจำภายนอก จากนั้นจะอ่านข้อมูลที่ เป็นคำสั่งกลับเข้ามาทางพอร์ต 0 ไปยัง Internal Data Bas แล้วไปเก็บที่ Instruction Register เพื่อทำงานต่อไปเหมือนกับขั้นตอนการอ่านจาก ROM ภายใน การทำงานในช่วงส่งค่าตำแหน่งหน่วยความจำไปยังหน่วยความจำแล้วอ่านข้อมูลที่เป็นคำสั่ง กลับเข้ามาเก็บไว้ใน Instruction Register เรียกว่าเป็นช่วงการ Fetch (Fetch Cycle) ช่วงต่อไปจะเป็นช่วงของการทำงานตามคำสั่งเรียกว่า Execute Cycle เช่นถ้าเป็นคำสั่งให้บวกข้อมูลในรีจิสเตอร์ Accumulator กับข้อมูลหน่วยความจำ Data Memory ภายใน RAM ตำแหน่ง 23H วงจร Timing And Control ก็จะส่งสัญญาณให้ Instruction Register

ส่งค่าตำแหน่งหน่วยความจำตำแหน่ง 23H ลงมายัง Internal Data Bas แล้วนำข้อมูลไปเก็บไว้ที่ RAM ADDR Register เพื่อชี้ตำแหน่งหน่วยความจำ RAM จากนั้น Timing Control จะสั่งให้ RAM ส่งข้อมูลที่เก็บในหน่วยความจำตำแหน่ง 23H ลงมายัง Internal Data Bas แล้วนำข้อมูลไปเก็บไว้ที่ TMP1 (วงจร Latch) ซึ่งจะทำหน้าที่เก็บสถานะ ผลลัพธ์ของการทำงานใน ALU เช่นผลลัพธ์การบวกมีค่าน้อยกว่า 8 บิต ก็จะทำให้บิตหนึ่งใน PSW ถูก Set เป็น 1

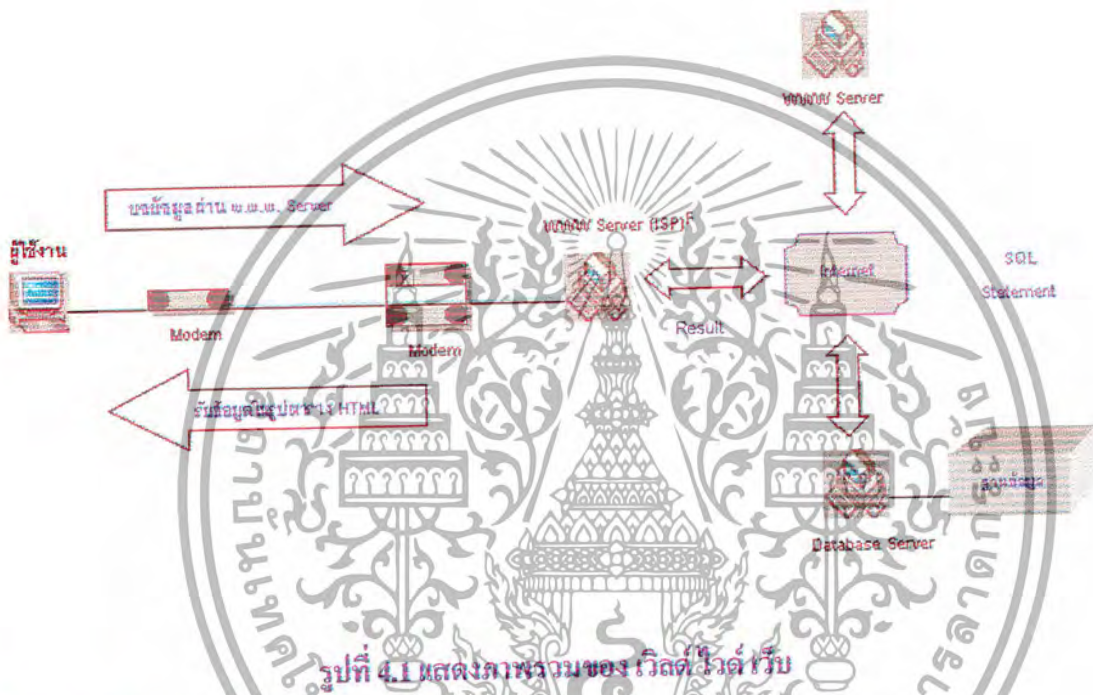
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### เว็บเซิร์ฟเวอร์ Web Server

#### 4.1 เว็บเซิร์ฟเวอร์

4.1.1 เว็บเซิร์ฟเวอร์ คือ แอปพลิเคชันทำหน้าที่รับ และประมวลผลเอกสาร ที่ถูกร้องขอจากผู้ใช้บริการอินเทอร์เน็ต จากนั้นเว็บเซิร์ฟเวอร์ก็จะส่งเอกสารกลับไปแสดงผลให้ใช้บริการผ่านบราวเซอร์ นอกจากนี้เว็บเซิร์ฟเวอร์จะถูกนำมาให้บริการในอินเทอร์เน็ตแล้ว แต่อาจมีการประยุกต์ให้นำมาใช้กับเครือข่ายภายในองค์กร หรือ อินเทอร์เน็ตได้เช่นกัน



รูปที่ 4.1 แสดงภาพรวมของ เวิลด์ ไรด์ เว็บ

ซึ่งเป็นการบริการที่ได้รับความนิยมสูงสุดของอินเทอร์เน็ต แต่เดิมนั้นเว็บเซิร์ฟเวอร์มักอยู่ในเครื่องคอมพิวเตอร์ ยูนิกซ์ ที่มีประสิทธิภาพสูง และราคาแพง ต่อมาเมื่ออินเทอร์เน็ตขยายความนิยชมานักผู้ใช้ พืช ทำให้มีการพัฒนาซอฟต์แวร์ทำหน้าที่เป็นเว็บเซิร์ฟเวอร์บนซีพี ซึ่งสามารถรันได้ทั้งวินโดวส์95/98และวินโดวส์เอ็นทีเซิร์ฟ/Workstation ตัวอย่างเช่น

- NCSA Web Server จาก NSCA เป็นเว็บเซิร์ฟเวอร์ที่สามารถดาวน์โหลดได้ฟรีจากเว็บไซต์ที่ให้ บริการ ความนี้โหลด ฟรีทั่วไป
- Net Server จาก Netscape เป็น เวิลด์ ไรด์ เว็บ (World Wide Web : www)เซิร์ฟเวอร์ (Server)ที่มีความสามารถรองรับ จาวา (JAVA) ได้อย่างเต็มรูปแบบ
- ออรากิลเว็บเซิร์ฟเวอร์ (Oracle Web Server) จากออรากิล เป็นเว็บเซิร์ฟเวอร์ที่เน้นความสามารถด้านการติดต่อกับเซิร์ฟเวอร์ฐานข้อมูลโดยเฉพาะการใช้งานร่วมกับระบบจัดการฐานข้อมูลของ ออรากิล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เกิด ปัจจุบันเราอาจจะได้ยินชื่อที่ตั้งขึ้นใหม่เป็น แอปพลิเคชัน เซิร์ฟเวอร์ (Application Server) (คือ ออราเคิล เว็บ เซิร์ฟเวอร์ ตัวเดิมที่ถูกพัฒนาขึ้นมา)

-Personal Web Server จากไมโครซอฟท์เป็นเว็บเซิร์ฟเวอร์ที่สามารถใช้งานได้กับ วินโดวส์ 95/98 หรือ วินโดวส์ เอนที เวอร์คสแตชันและรองรับการใช้งานร่วมกับ เอเอสพี (Active Server Page) โดยเราสามารถดาวน์โหลดมาใช้ได้ฟรีได้จากเว็บไซต์ของไมโครซอฟท์ แต่ถ้าใช้งาน วินโดวส์ 98 ก็ สามารถเลือกติดตั้งได้ ซึ่งมักใช้ในการทดสอบเว็บเพจ หรือแอปพลิเคชันอินเทอร์เน็ต ก่อนนำไปใช้งานจริง

-IIS (Internet Information Server) จากไมโครซอฟท์ เป็นอินเทอร์เน็ต เซิร์ฟเวอร์เวอร์ชัน 2 ที่แถมมากับ วินโดวส์ เอนที เซิร์ฟเวอร์ 4.0 มีความสามารถให้บริการได้ทั้ง เวิลด์ ไวด์ เว็บ ,เอฟทีพี (FTP:File Transfer Protocol) และ Gopher ส่วนเวอร์ชันที่ใช้งานกับเอเอสพี ได้จะเป็นเวอร์ชัน 3.0 ขึ้นไป สำหรับบทนี้จะแสดงอาการใช้งานกับเวอร์ชัน 4.0

#### 4.1.2 ระบบเครือข่ายอินเทอร์เน็ต

ในองค์กรหนึ่ง ๆ อาจมีการติดตั้งระบบเครือข่ายแบบใดก็ได้ตั้งแต่ 2 ระบบขึ้นไปแต่เป็นระบบที่อยู่ห่างไกลกันมาก เช่น อยู่คนละจังหวัด ระบบเครือข่ายแบบใดก็ได้แต่ละระบบก็สามารถถูกเชื่อมโยงเข้าด้วยกันจนกลายเป็นเครือข่ายที่เรียกว่า เครือข่ายระยะไกล (WAN: Wide Area Network) และนอกจากนี้ระบบเครือข่ายระยะไกล ที่หนึ่งก็สามารถเชื่อมกับ เครือข่ายระยะไกล ที่อยู่ห่างไกลกันออกไปมาก ๆ ได้อีกเช่น อยู่คนละประเทศหรือคนละทวีป ทำให้เกิดเครือข่ายขนาดใหญ่ที่เรียกว่า "Internetworking" ซึ่งเป็นระบบเครือข่ายขนาดใหญ่และเป็นหลักการที่กลายมาเป็นระบบเครือข่ายอินเทอร์เน็ตในที่สุด

อินเทอร์เน็ต เป็นเครือข่ายคอมพิวเตอร์ที่มีขนาดใหญ่มาก เกิดจากการเชื่อมต่อเครือข่ายคอมพิวเตอร์จำนวนมากมายังโลกเข้าด้วยกัน ไม่ว่าจะเป็เครือข่ายขนาดเล็ก เช่น ระบบเครือข่ายแบบใดก็ได้หรือระบบเครือข่ายขนาดใหญ่เช่นระบบเครือข่ายของมินิหรือเมนเฟรมคอมพิวเตอร์ซึ่งแต่ละเครือข่ายก็จะมีเครื่องคอมพิวเตอร์ที่เป็นเครื่องแม่ข่าย หรือ โฮสต์ (Host) ซึ่งมียู่หลายชนิดหลายยี่ห้อ โดยจะมีการกำหนดข้อตกลงใจการสื่อสารที่เรียกว่า โพรโตคอล (Protocol) ขึ้นมา เพื่อให้คอมพิวเตอร์แต่ละชนิดสามารถติดต่อสื่อสารกันได้ โพรโตคอลมาตรฐานที่ใช้ในการสื่อสารบนอินเทอร์เน็ตจะมีชื่อเรียกว่า

TCP/IP (Transmission Control Protocol/Internet Protocol)

ถ้าเราเปรียบให้ โพรโตคอลเหมือนภาษาที่ใช้ในการสื่อสารของมนุษย์ซึ่งภาษามนุษย์นั้นมีมากมายหลายภาษาเช่นเดียวกับ โพรโตคอลอยู่หลายแบบเช่นกัน TCP/IP นั้นจะเปรียบได้กับภาษาอังกฤษเนื่องจากเป็น โพรโตคอลที่เครื่องคอมพิวเตอร์ของทุกระบบเข้าใจและมีการใช้โปรโตคอลนี้ร่วมกันเพื่อการเชื่อมต่อเข้าสู่โลกอินเทอร์เน็ต

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 4.1.3 เวิลด์ ไรด์ เว็บ คืออะไร

เมื่อไม่กี่ปีก่อนหน้านี้มีผู้สนใจใช้งานอินเทอร์เน็ตไม่มากนัก เนื่องจากการใช้บริการอินเทอร์เน็ตไม่ว่าจะเป็นการค้นหาข่าวสารข้อมูล การรับส่งจดหมายอิเล็กทรอนิกส์ การสำเนาไฟล์ ฯลฯ จะอยู่ในรูปแบบตัวอักษร (Text Mode) เท่านั้น ไม่มีการแสดงเป็นรูปภาพ เสียง และไม่มีตัวอักษรแบบต่าง ปรากฏให้เห็นแต่อย่างใด นอกจากนี้ผู้ใช้ต้องเรียนรู้และจดจำคำสั่งของคอมพิวเตอร์มากมายเช่น ต้องเรียนรู้คำสั่งเบื้องต้นของยูนิกซ์ (UNIX) เนื่องจากเมื่อจะมีการเรียกใช้อินเทอร์เน็ต เครื่องคอมพิวเตอร์ของผู้ใช้จะถูกเปลี่ยนให้เป็นเทอร์มินัลของโฮสต์คอมพิวเตอร์ที่ให้บริการอินเทอร์เน็ต และโฮสต์ส่วนมากจะทำงานอยู่ภายใต้ระบบปฏิบัติการยูนิกซ์ ดังนั้นผู้ใช้จึงต้องเรียนรู้คำสั่งเบื้องต้นของยูนิกซ์ (เช่นเกี่ยวกับการเรียนรู้คำสั่งคอส (DOS) บนเครื่องพีซี) เพื่อทำการป้อนคำสั่งที่เป็นตัวอักษรด้วยตัวเองให้โฮสต์คอมพิวเตอร์ทำงานตามที่เราต้องการใช้

จนกระทั่งมีบริการที่เรียกว่า เวิลด์ ไรด์ เว็บ หรือเครือข่ายใยแมงมุมเกิดขึ้น ทำให้ความนิยมการใช้งานอินเทอร์เน็ตสูงขึ้นเป็นทวีคูณ เนื่องจาก เวิลด์ ไรด์ เว็บ เป็นบริการอันหนึ่งที่มีอยู่ในอินเทอร์เน็ต ทำให้การใช้งานอินเทอร์เน็ตเป็นเรื่องที่ง่ายขึ้นผู้ใช้ไม่ต้องจดจำคำสั่งของยูนิกซ์อีกต่อไป การอ่านและค้นหาข่าวสารข้อมูลทั้งได้การक्रमบนเมาส์เพียงอย่างเดียว นอกจากนี้ข่าวสาร ข้อมูล หรือเอกสารที่เก็บอยู่ในอินเทอร์เน็ตจะอยู่ในรูปสื่อผสม (Multimedia) ที่เรียกว่า "เว็บเพจ" (web Page) อันประกอบไปด้วยอักษรที่มีข้อความที่การเชื่อมโยงถึงกัน ได้เหมือนกับการใช้ Help ในโปรแกรม วินโดวส์ โดยผู้ใช้สามารถเรียกใช้เอกสารหนึ่งจากเอกสารหนึ่งได้ ซึ่งเป็นลักษณะการเชื่อมต่อที่คล้ายกับ ใยแมงมุม จึงมีการตั้งชื่อบริการนี้ว่าเป็นเครือข่ายใยแมงมุม

ส่วนประกอบของ เวิลด์ ไรด์ เว็บ

- แหล่งข้อมูลหรือเว็บไซต์
- โปรแกรมเว็บเบราว์เซอร์

#### 1. แหล่งข้อมูลหรือเว็บไซต์

เว็บไซต์ หรือ เว็บเซิร์ฟเวอร์ คือ ระบบคอมพิวเตอร์ที่เป็นแหล่งเก็บเว็บเพจที่ผู้ใช้บริการสามารถเรียกดูเว็บเพจที่เก็บอยู่ในเว็บไซต์นั้นได้ ซึ่งเครื่องคอมพิวเตอร์ที่ทำหน้าที่เป็นเว็บไซต์อาจจะใช้ระบบปฏิบัติการยูนิกซ์หรือวินโดวส์ เอนท์ ก็ได้ และจะต้องมีการติดตั้งโปรแกรมจัดการ ที่จะทำให้เครื่องคอมพิวเตอร์เครื่องนั้นทำหน้าที่เป็นเว็บไซต์ ตัวอย่างเช่นในกรณีที่เครื่องคอมพิวเตอร์นั้นใช้ระบบปฏิบัติการ วินโดวส์ เอนท์ จะมีซอฟต์แวร์เว็บเซิร์ฟเวอร์ให้เลือกใช้ เช่น

โปรแกรม Purveyor HTTP Server ของสถาบัน European Microsoft Windows NT Academic Centre(EMWAC) โปรแกรม เว็บไซต์ ของบริษัท O'Reilly and Apache เป็นต้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ผู้ที่เป็นเจ้าของเว็บไซต์จะจัดสร้างเว็บเพจของคนเก็บไว้ที่เว็บไซต์นั้นเพื่อให้ผู้ใช้อื่น ๆ ทั่วโลกสามารถเข้ามาดูเก็บไว้ในเว็บไซต์นั้นได้ ตัวอย่างเช่น เว็บเพจของสำนักข่าว CNN จะเก็บอยู่ที่เว็บไซต์ [www.cnn.com](http://www.cnn.com) เป็นต้น

## 2. โปรแกรมเว็บเบราว์เซอร์

เป็นโปรแกรมหรือซอฟต์แวร์ที่ใช้การเข้าสู่ เวิลด์ ไวด์ เว็บ จะเปิดดูเว็บเพจที่เก็บอยู่ในเว็บไซต์ ใดๆ ตัวอย่างของโปรแกรมเว็บเบราว์เซอร์ ที่มีให้ดาวน์โหลดใช้กันได้ฟรีได้แก่ โปรแกรม Netscape Navigator จากบริษัท Netscape Communications และ โปรแกรม Microsoft Internet Explorer

จากบริษัท ไมโครซอฟท์ ผู้ผลิตโปรแกรมวินโดวส์ที่ผู้คนรู้จักกันดี เป็นต้น ซึ่งโปรแกรมเว็บเบราว์เซอร์ส่วนใหญ่จะทำงานร่วมกับโปรแกรมวินโดวส์ และนอกจากจะใช้เพื่อดูเว็บเพจจากเว็บไซต์ใดๆ แล้วหลายโปรแกรมยังมีความสามารถอื่น ๆ เช่น บริการส่งจดหมายอิเล็กทรอนิกส์ การค้นหาข้อมูล การโอนถ่ายโปรแกรมด้วย FTP เป็นต้น

### 4.1.4 โฮมเพจ

โฮมเพจ (Home Page) จะหมายถึงหน้าแรกของเว็บเพจทั้งหมดที่ผู้ใช้บริการอินเทอร์เน็ตจะพบเมื่อมีการเข้าไปยังเว็บไซต์ใด ๆ โฮมเพจเปรียบเหมือนกับสารบัญ และหน้าที่เจ้าของเว็บไซต์สร้างขึ้นเพื่อจะใช้ประชาสัมพันธ์หรือกิจกรรมของตนให้บริการในสิ่งใดบ้าง นอกจากนี้ภายในโฮมเพจก็อาจมีเอกสารข้อความอื่น ๆ เป็นจำนวนมากได้

### 4.1.5 IP Address

เนื่องจากในระบบเครือข่ายอินเทอร์เน็ตจะใช้โปรโตคอล TCP/IP เป็นมาตรฐานในการสื่อสารข้อมูลซึ่งจะมีการกำหนดหมายเลขประจำตัวที่ไม่ซ้ำกัน ให้กับเครื่องคอมพิวเตอร์ทุกเครื่องที่มีการเชื่อมกันอยู่ในระบบเครือข่าย หมายเลขประจำตัวนี้จะถูกเรียกว่า IP Address หรือหมายเลข IT โดยมีรูปแบบเป็นชุดของตัวเลข 4 ชุดที่สั้นเครื่องหมายเลขจุด เช่น 202.44.192.43 ตัวเลขแต่ละชุดจะมีขนาด 8 บิตคั้งนั้นแต่ละชุดจะมีค่าได้ตั้งแต่ 0 ถึง 255 ยกเว้นถึง 255 - 1 = 254 เท่านั้นซึ่งจะสามารถกำหนดหมายเลข IP ให้กับเครื่องคอมพิวเตอร์ได้ทั้งหมด 4 พินด้าวนเลขหมายที่ไม่ซ้ำกันเลย

คั้งนั้นเมื่อมีการติดตั้งเครื่องคอมพิวเตอร์ที่เป็นโฮสต์คอมพิวเตอร์เพื่อเชื่อมต่อเข้ากับอินเทอร์เน็ตและให้บริการต่าง ๆ จำเป็นต้องขอหมายเลข IP ประจำหน่วย Internet Network Information Center (InterNIC) ขององค์กร Network Solution Incorporated (NSI) ที่รัฐเวอร์จิเนีย สหรัฐอเมริกา แต่ถ้าผู้ใช้สมัครเป็นสมาชิกกับหน่วยงานที่ให้บริการอินเทอร์เน็ต (Internet Service Provider. ISP) (ในประเทศไทยมีอยู่หลายงาน) ก็ไม่ต้องขอหมายเลข IP เนื่องจาก ISP จะเป็นผู้ส่งหมายเลข IP ให้แก่ผู้ใช้เอง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

#### 4.1.6 ดี เอ็น เอส (Domain Name System:DNS)

การใช้หมายเลข IP ซึ่งเป็นตัวเลขล้วน ๆ ในการอ้างอิงถึงเครื่องคอมพิวเตอร์แต่ละเครื่องนั้น จะมีข้อเสียคือจำยากและก่อให้เกิดความสับสนได้ง่าย จึงมีการพัฒนาวิธีการอ้างอิงถึงหมายเลข IP แบบใหม่เรียกว่า Domain Name System (DNS) ขึ้นมา

ดีเอ็นเอส เป็นเทคนิคการเปลี่ยนหมายเลข IP เป็น 202.12.97.1 ผู้ใช้บริการสามารถเขียนเป็นชื่อโดเมนคือ kku1.kku.ac.th แทน เครื่องคอมพิวเตอร์ที่เรียกว่าดีเอ็นเอส จะทำหน้าที่ในการแปลงชื่อโดเมนให้เป็นหมายเลข IP อีกทีหนึ่งรูปแบบของชื่อโดเมนจะมีดังต่อไปนี้

ชื่อ โฮสต์คอมพิวเตอร์.ชื่อเครือข่ายท้องถิ่น[.ชื่อ โดเมนย่อย].ชื่อ โดเมนระดับบนสุด

ชื่อโดเมนจะมีการแบ่งออกเป็นระดับชั้น โดยใช้เครื่องหมายจุดคั่นชื่อโดเมนที่อยู่ทางด้านขวาสุดจะเรียกว่าชื่อโดเมนระดับบนสุด ซึ่งจะแบ่งออกได้อีก 2 ประเภทคือ

1. ชื่อโดเมนระดับบนสุดที่เป็นชื่อย่อประเภทขององค์กรในสหรัฐอเมริกา เช่น

comcommercial

eda educational

gov governmental

mil military

net network services

org non-commercial organization

กลุ่มองค์กรเอกชน

กลุ่มสถาบันการศึกษา

กลุ่มองค์กรของรัฐทั่วไป

กลุ่มองค์กรทหาร

กลุ่มองค์กรบริการเครือข่าย

กลุ่มองค์กรที่ไม่แสวงหาผลกำไร

ตัวอย่างเช่น ชื่อโดเมน biz.zd.com โดเมนระดับสูงสุดคือ com ที่แสดงถึงองค์กรเอกชนที่อยู่ในประเทศสหรัฐอเมริกา ถ้ารับระดับถัดไปคือ zd จะแสดงถึงชื่อเครือข่ายของบริษัทเอกชนนั้น และ biz จะหมายถึงชื่อ โฮสต์คอมพิวเตอร์ที่ให้บริการเป็นต้น

2. ชื่อโดเมนระดับบนสุดที่เป็นชื่อย่อของประเทศต่างๆ เช่น

au ออสเตรเลีย jp ญี่ปุ่น

ca แคนาดา ,th ไทย

fr ฝรั่งเศส ,uk อังกฤษ

จะมีสับโดเมน(Subdomain)ที่แสดงถึงประเภทขององค์กรในประเทศนั้น ๆ เช่น

ac สถาบันการศึกษา ,go องค์กรรัฐบาล

co องค์กรเอกชน ,or องค์กรไม่แสวงหาผลกำไร

ตัวอย่างเช่น ชื่อโดเมน kku1.kku.ac.th มีชื่อโดเมนระดับสูงสุดคือ th ที่หมายถึงประเทศไทยระดับรองลงมาคือ โดเมนย่อยคือ ac หมายถึงสถาบันการศึกษา ระดับถัดไปคือ kku หมายถึงมหาวิทยาลัย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ขอนแก่นซึ่งโดยทั่วไปจะหมายถึง เครื่องข่ายมหาวิทยาลัยขอนแก่นสำหรับ kku1 จะเป็นชื่อโฮสต์คอมพิวเตอร์ที่ให้บริการเป็นต้น

#### 4.1.7 URL

นอกจากที่อยู่อีเมลล์แล้ว ชื่อโดเมนยังนำมาใช้เป็นส่วนหนึ่งของข้อมูล URL (Uniform Resource Locator) ซึ่งโปรแกรมเว็บเบราว์เซอร์จะใช้ข้อมูล URL นี้ในการค้นหาที่อยู่บนเว็บไซต์ที่ผู้ใช้บริการจะไปเรียกดูข้อมูล รูปแบบของ URL มีดังนี้

โปรโตคอล://ชื่อโดเมน/ไครเรกทอรีที่เก็บไฟล์ในโฮสต์/ชื่อไฟล์ในโฮสต์

โปรโตคอล หมายถึงโปรโตคอลที่ใช้เรียกบริการในอินเทอร์เน็ต เช่น http:// หมายถึงโปรโตคอลที่เรียกใช้บริการเว็ลด์ ไซด์ เว็บ ที่มีข้อมูลเป็น Hypertext

ftp:// หมายถึงโปรโตคอลที่เรียกใช้บริการ FTP เป็นต้น

ตัวอย่างเช่น ถ้าผู้ใช้ต้องการเรียกใช้บริการ เว็ลด์ ไซด์ เว็บ เพื่อเข้าไปยังโฮมเพจของบริษัทการบินไทยจะทำได้โดยการระบุ URL เป็น <http://www.thaiair.com/> ในช่องแอดเดรส ของเว็บเบราว์เซอร์ที่ใช้โดยระบุเฉพาะชื่อโดเมนอย่างเดียว

แต่ถ้าจะต้องการเรียกดูข้อมูล ตารางเที่ยวบิน ตารางเที่ยวบินของบริษัท จะสามารถระบุ URL เป็น <http://www.thaiair.com/flying/schedule.htm> หรือทำการคลิกเลือกหัวข้อตารางเที่ยวบินที่อยู่ภายในโฮมเพจนั้นได้เลย

#### 4.1.8 HTML

โฮมเพจและเว็บเพจใด ๆ โดยส่วนใหญ่จะถูกสร้างขึ้นมาจากภาษาที่เรียกว่า HTML (Hyper T extmarkup Language) หรือ “ภาษาคำหรับทำเครื่องหมาย” ประกอบด้วยคำสั่ง (Tags) ที่ใช้ในการกำหนดว่าเว็บเพจจะมีข้อความอะไร มีการแสดงรูปภาพ เสียง และภาพวีดิโอที่จัดตำแหน่งใด นอกจากนี้ ยังมีคำสั่งสำหรับการเชื่อมโยงเว็บเพจหนึ่งไปยังอีกเว็บหนึ่งและไปยังบริการอื่น ๆ ในอินเทอร์เน็ต อีกด้วย

การแสดงเว็บเพจนอกจากจะสร้างโดยการเขียน HTML บนโปรแกรม Editor ตัวใดตัวหนึ่ง เช่นโปรแกรม Notepad บน วินโดวส์ และยังสามารถใช้โปรแกรมช่วยสร้างเว็บเพจกำหนดรูปร่างหน้าตาของเว็บเพจได้โดยตรง โดยที่ผู้สร้างไม่จำเป็นต้องเขียนหรือเรียนรู้ HTML แต่อย่างใด ซึ่งในปัจจุบันจะมีโปรแกรมช่วยสร้างเว็บเพจหลายตัว เช่น โปรแกรม Microsoft FrontPage โปรแกรม Netscape Navigator Gold เป็นต้น

#### 4.1.9 HTML Form และ CGT สกริปต์

ถึงแม้ว่าเว็บเพจที่จะสร้างจะได้รับการออกแบบให้เป็นที่น่าดึงดูดความสนใจ เต็มไปด้วยสีสันรูปภาพการสร้าง Link เชื่อมโยงข้อมูล ฯลฯ แต่สิ่งหนึ่งที่จะทำให้เว็บเพจยังน่าสนใจ มากขึ้นอีกคือ การเปิดโอกาสให้มีการโต้ตอบจากผู้อ่านทั่วไป ซึ่งทำได้โดยการสร้างแบบฟอร์ม (Form) ที่มี

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ลักษณะเหมือนกับแบบสอบถามทั่ว ๆ ไปลงเว็บเพจ เพื่อให้ผู้อ่านกรอกข้อมูลลงในแบบฟอร์มแล้วส่งกับมายังเครื่องให้บริการที่เป็นเครื่องเซิร์ฟเวอร์

ในการนี้จำเป็นจะต้องมีการเขียนโปรแกรมย่อยที่เรียกว่า สคริปต์ เก็บไว้ในเครื่องเซิร์ฟเวอร์ เพื่อทำหน้าที่รวบรวมข้อมูลที่ส่งมาจากแบบฟอร์มและนำไปประมวลผลเก็บไว้ หรืออาจส่งผลลัพธ์กลับไปยังผู้อ่านอีกทีหนึ่ง

สคริปต์ ที่ถูกเขียนขึ้นนี้อาจสร้างภาษาใดภาษาหนึ่งเช่น ภาษา C,Perl,Java,Pascal หรือแม้แต่ Unix Shell

#### 4.1.10 Static และ Dynamic HTML

เอกสาร HTML โดยทั่วไปจะมีอยู่ 2 ลักษณะคือ

-Static HTML

-Dynamic HTML

##### Static HTML

จะเป็นเอกสาร HTML ที่ถูกเขียนขึ้นและเก็บอยู่ในเครื่องเซิร์ฟเวอร์ เมื่อมีการเรียกใช้งานจากผู้ใช้ผ่านทาง URL เครื่องเซิร์ฟเวอร์ก็จะส่งเอกสารนี้ไปให้กับผู้ใช้ ซึ่งเอกสารชนิดนี้จะถูกเขียนขึ้นโดยใช้ HTML tag ธรรมดาไม่จำเป็นต้องสร้างสคริปต์ใดๆ

##### Dynamic HTML

สามารถเรียกได้อีกอย่างว่า "HTML on-the-fly" เอกสารประเภทนี้จำเป็นต้องมีการเขียนสคริปต์เก็บไว้ในเครื่องเซิร์ฟเวอร์เซิร์ฟเวอร์จะทำการประมวลผล สคริปต์ จะทำการสร้างผลลัพธ์ที่เป็นเอกสาร HTML ที่มีรูปแบบของผลลัพธ์ขึ้นอยู่กับข้อมูลที่ผู้ใช้ป้อนจากแบบฟอร์ม จึงเรียกเอกสารประเภทนี้ว่า "Dynamic HTML" เพื่อให้เซิร์ฟเวอร์ส่งผลลัพธ์นั้นกลับไปยังผู้ใช้ทันที

#### 4.2 เอเอสพี (Active Server Page:ASP)

เอเอสพี เป็นคำย่อมาจาก Active Server Pages ซึ่งเป็นโปรแกรมที่พัฒนาขึ้นโดยบริษัท ไมโครซอฟท์ เพื่อใช้งานทางด้านอินเทอร์เน็ต โดย เอเอสพี จะทำหน้าที่ตีความเอกสารที่เขียนด้วยภาษาสคริปต์ เช่น VB สคริปต์ โดยที่มีเอเอสพี LAG (คือ คำสั่งที่มีเครื่องหมาย<%%>) กำกับอยู่ ซึ่งบราวเซอร์ทั่วไป เช่น Netscape Navigator หรือ Internet Explorer ไม่สามารถนำไปแสดงผล จากนั้นจึงสร้างเอกสารผลลัพธ์เป็นเอกสาร HTML อันเป็นเอกสารที่ประกอบด้วย HTML tag ต่าง ๆ (คือ คำสั่งที่มีเครื่องหมาย<>)กำกับอยู่ซึ่งบราวเซอร์ทั่วไปสามารถนำไปสร้างเป็นเว็บเพจขึ้นเพื่อใช้แสดงผลได้

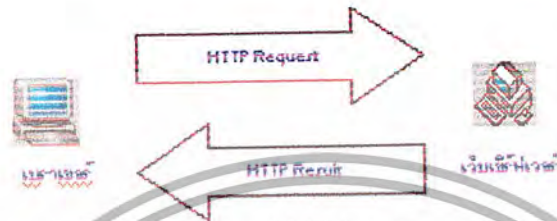
การทำงานของโปรแกรมเอเอสพี จะเกิดขึ้นเฉพาะทางฝั่งเซิร์ฟเวอร์เท่านั้น เราจึงเรียกว่าเป็นการทำงานแบบ เซิร์ฟเวอร์ไซด์ จากนั้นผลลัพธ์ที่จะได้ถูกส่งไปให้เว็บเซิร์ฟเวอร์ และเว็บเซิร์ฟเวอร์ก็จะส่งเอกสารดังกล่าวต่อไปยังบราวเซอร์อีกทีหนึ่ง เมื่อบราวเซอร์ได้รับเอกสารนั้นแล้ว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บราวเซอร์ก็จะสามารถแสดงผลได้ถูกต้องครบถ้วน โดยการทำงานของบราวเซอร์ทางฝั่งของผู้ใช้นี้ เราเรียกว่าเป็นการทำงานแบบ ไคลเอนไซด์ (client side)

เข้าใจขอบเขตการทำงานของ เอเอสพี

เพื่อจะได้ศึกษา เอเอสพี ในระดับเบื้องต้นก่อนอื่นขอให้ลองทำความเข้าใจพื้นฐาน ขอบเขตการทำงานของเอเอสพี (เอเอสพี model) จากรูป 4.2 ดังนี้



รูปที่ 4.2 ขอบเขตการทำงานของ เอเอสพี

การทำงานทั้งหมดจะเริ่มจาก บราวเซอร์ร้องขอเอกสาร HTML ไปยังเว็บเซิร์ฟเวอร์ผ่านทาง HTTP (HTTP request) โดยที่เอกสารขอไปจะเป็นแฟ้มข้อมูลที่มีนามสกุลเป็น .ASP (เช่น search.ASP ฯลฯ) เมื่อเว็บเซิร์ฟเวอร์ได้รับร้องขอดังกล่าว ก็จะส่งเอกสารนั้นไปให้ เอเอสพี ตีความ จากนั้น เอเอสพี ก็จะสร้างเอกสาร HTML ส่งกลับไม่ให้เว็บเซิร์ฟเวอร์เพื่อส่งต่อไปยังบราวเซอร์ และใช้แสดงผลในฝั่งของผู้ใช้ต่อไป (HTTP Response) ซึ่งการทำงานของเอเอสพี นี้แทบไม่แตกต่างไปจากหลักการงาน โปรแกรม CGI (Common Gateway Interface) จนอาจกล่าวได้ว่าเอเอสพี ก็เป็น โปรแกรม CGI ประเภทหนึ่งเช่นกัน

การเขียนโปรแกรมที่จะสร้างเอกสารที่จะทำงานกับเอเอสพี นั้น (ในที่นี้ขอเรียกว่า เอกสาร เอเอสพี) ไม่จำเป็นต้องอาศัยโปรแกรมเฉพาะในการเขียนเราสามารถนำ โปรแกรมประเภท text editor มาใช้งานได้ทันที เช่น โปรแกรมที่เขียนเอกสาร เอเอสพี โดยเฉพาะก็ได้เช่น Visual InterDev เป็นต้น

เอกสาร เอเอสพี แตกต่างจากเอกสาร HTML ทั่วไปตรงที่มีส่วนของคำสั่ง เอเอสพี อยู่ในเอกสารด้วย โดยทั่วไปหากเรานำเอกสาร HTML มาเปลี่ยนเป็นเอกสาร เอเอสพี เลยก็ทำได้ นั่นคือวิธีการสร้างเอกสาร เอเอสพี แบบง่ายๆ เช่น เราสามารถเปลี่ยนเอกสารที่มี HTML tag กำกับอยู่ก็จะเกิดการเปลี่ยนแปลงใดๆ ในเอกสารนั้นเลย แต่หากว่าส่วนใดมี เอเอสพี tag กำกับ เอเอสพี ก็จะเปลี่ยนเอกสารส่วนดังกล่าวไปอยู่ในรูปข้อความทั่วไป HTML tag แทน เช่น หากในเอกสารมีคำสั่ง

```
<br>%response.write("Hello"& now)%>
```

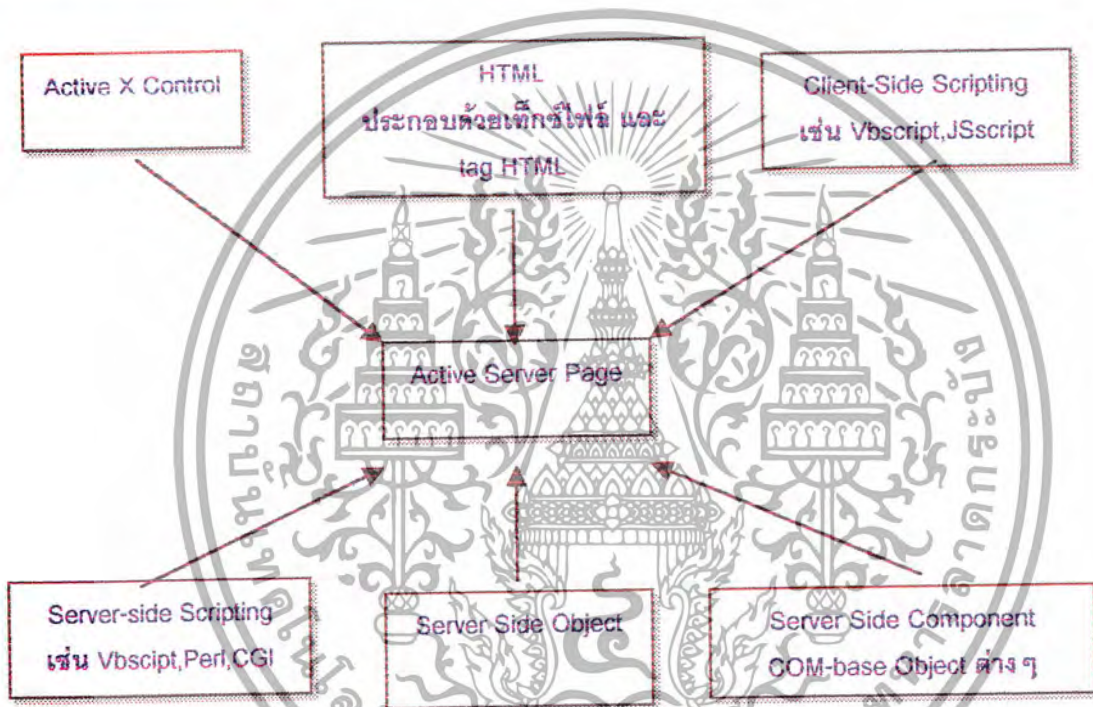
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ก็จะถูกเปลี่ยนเป็น.....

<br>Hello แล้วตามด้วยวันและเวลาปัจจุบัน

#### 4.2.1 เอเอสพี (Active Server Pages)

เอเอสพี เป็นชื่อของเทคโนโลยีในการพัฒนาแอปพลิเคชันสำหรับอินเทอร์เน็ตที่ไม่โครซอฟท์คิดขึ้นมา โดยเน้นไปที่การพัฒนา และจัดการแอปพลิเคชันที่เว็บเซิร์ฟเวอร์ โดยเราจะเรียกแอปพลิเคชันที่สร้างจากเทคโนโลยีเอเอสพี ว่า “แอปพลิเคชัน เอเอสพี”



รูปที่ 4.3 โครงสร้างทางด้านเอเอสพี

แอปพลิเคชัน เอเอสพี (ASP Application) ก็คือ ไฟล์ที่บรรจุเอาคำสั่งสคริปต์ ต่าง ๆ ผสมรวมกับเอกสาร HTML ซึ่งจะถูกเก็บไว้ที่เว็บเซิร์ฟเวอร์ เมื่อมีบราวเซอร์เรียกใช้งานก็จะถูกแปล (โดย ASP Interpreter) และถูกเอ็กซ์คิวต์ที่เว็บเซิร์ฟเวอร์ โดยผลลัพธ์ที่ได้จากเอ็กซ์คิวต์แอปพลิเคชันเอเอสพีจะเก็บในรูปแบบเอกสาร HTML แล้วถูกส่ง กลับไปให้บราวเซอร์ที่เรียกใช้แอปพลิเคชันเอเอสพี นั้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## ตารางที่ 4.1 ตัวอย่างไฟล์ เอ เอส พี

ตัวอย่างไฟล์เอเอสพี
<pre> &lt;HTML&gt; &lt;HEAD&gt;TITLE&gt;ยินดีต้อนรับเข้าสู่โลกของเอเอสพี&lt;/TITLE&gt;&lt;/HEAD&gt; &lt;BODY&gt; &lt;Script RUNAT=SERVER LANGUAGE=VBScript&gt; &lt;% Sub GreetingASPO     Response.Write("ยินดีต้อนรับสู่โลกของเอเอสพี") End Sub %&gt; &lt;% Call GreetingASPO%&gt; &lt;%= "&lt;BR&gt;ขณะนี้เวลา" &amp; Time &amp; "ขณะนี้เป็นเวลาที่เราที่เซิร์ฟเวอร์ นะครับ"%&gt; &lt;/SCRIPT&gt; &lt;/BODY&gt; &lt;/HTML&gt; </pre>

เพราะฉะนั้นจึงไม่มีการรันแอปพลิเคชัน เอเอสพี ที่เบราว์เซอร์ โดยต้องรันที่เว็บเซิร์ฟเวอร์เท่านั้น สังเกตได้จากตัวอย่างที่ Tag<Script>จะกำหนดให้แอคทริบิวต์ RUNAT=SERVER

### 4.2.2 การเขียนสคริปต์ในรูปแบบของ เอเอสพี

สำหรับการเขียนสคริปต์ในรูปแบบของเอเอสพี เราสามารถใช้ได้ตั้งแต่ โน้ตแพ็ค,เวิร์ด โพรเซสเซอร์หรือแม้กระทั่งเครื่องมือพัฒนาแอปพลิเคชันสำหรับอินเทอร์เน็ตโดยเฉพาะ เช่น Visual Interdev

สำหรับสคริปต์ที่ใช้เขียน เอเอสพี ได้นั้น ณ ขณะนี้สามารถใช้ได้ 2 ภาษา ได้แก่ VBScript กับ Jscript (เป็น จาวาสคริปต์ ในเวอร์ชันของไมโครซอฟท์)

## ตารางที่ 4.2 ตัวอย่างไฟล์เอเอสพี ร่วมกันทั้ง Jscript และ VBScript

ตัวอย่างไฟล์เอเอสพี ร่วมกันทั้ง Jscript และ VBScript
<pre> %@LANGUAGE="VBScript"% &lt;HTML&gt; &lt;HEAD&gt;&lt;TITLE&gt;ทดลอง Jscript กับ เอเอสพี&lt;/TITLE&gt;&lt;/HEAD&gt; </pre>

เอกสารนี้เป็นเอกสารสงวนลิขสิทธิ์สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

<BODY>
<Script RUNAT=Server LANGUAGE="Jscript">
function J Script Sample()
{
var DateObject=new Date()
Response.Write("<H>ทดสอบ Jscript</H1>")
Reponse.Write("<BR>ขณะนี้เวลา"+DateObject.getHours()+"."+
DateObject.getMinute())
}
</Script>
<%Call J Script Sample%>
</BODY>
</HTML>

```



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## บทที่ 5

## การคำนวณและการสร้าง

## 5.1 ส่วนฮาร์ดแวร์

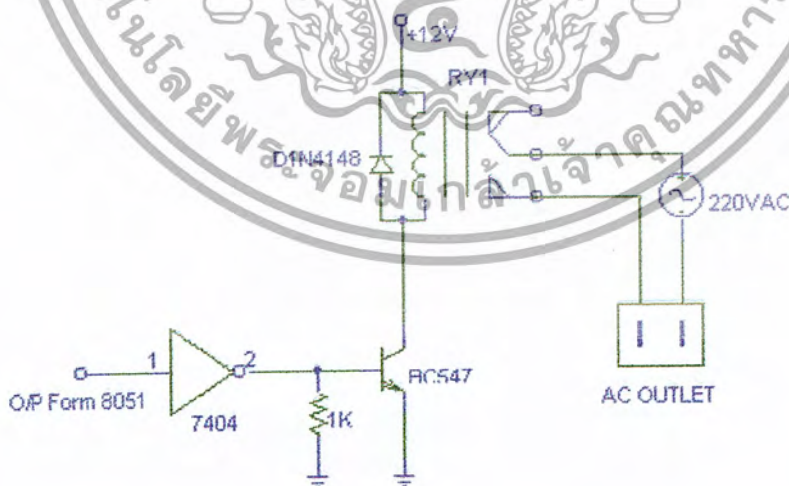
ในส่วนของวงจรซึ่งถูกควบคุมการทำงานโดย Microcontroller MCS 51 จะถูกแบ่งออกเป็น 4 วงจรดังต่อไปนี้

- 1) วงจรเชื่อมต่อเพื่อควบคุมอุปกรณ์ไฟฟ้า
- 2) วงจรตัวส่งและตัวรับสัญญาณอินฟราเรด(วงจร sensor)
- 3) วงจรตรวจวัดอุณหภูมิ
- 4) วงจรควบคุมการหมุนกลิ้ง
- 5) วงจรตรวจสอบสถานะ SW

## 5.1.1 วงจรเชื่อมต่อควบคุมอุปกรณ์ไฟฟ้า

ในการควบคุมอุปกรณ์ไฟฟ้าซึ่งใช้แรงดันไฟฟ้ากระแสสลับ 220 โวลต์ ไมโครคอนโทรลเลอร์

ไม่สามารถควบคุมได้โดยตรง จึงต้องใช้วงจร ทรานซิสเตอร์ เพื่อทำการควบคุมอุปกรณ์ไฟฟ้า โดยจะส่งสัญญาณควบคุมจากพอร์ตของไมโครคอนโทรลเลอร์ แต่เนื่องจากพอร์ตของไมโครคอนโทรลเลอร์มีกระแสไม่เพียงพอที่ จึงต้องทำการเพิ่มกระแสโดยต่อ Buffer ซึ่งในการควบคุมอุปกรณ์ไฟฟ้า ส่วนควบคุมอุปกรณ์ไฟฟ้าหรือเรียกว่า รีเลย์ (Relay) ซึ่งมีโครงสร้างประกอบด้วยส่วนต่างๆ ดังรูปที่ 5.1

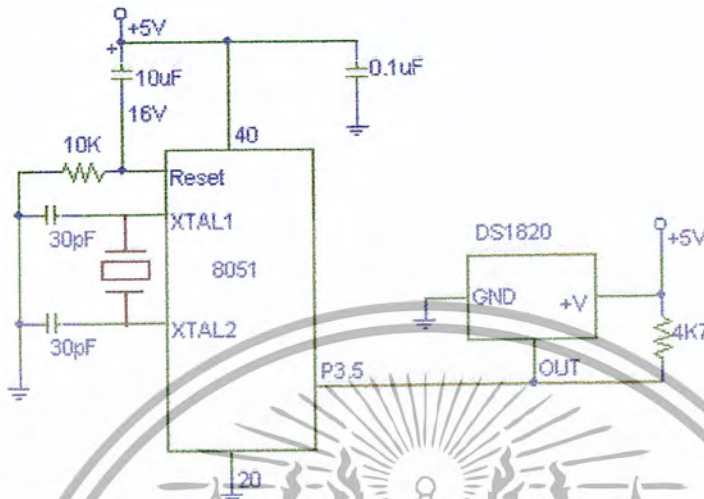


รูปที่ 5.1 แสดงวงจรควบคุมไฟ 220 โวลต์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 5.1.2 วงจรตรวจวัดอุณหภูมิ

วงจรนี้จะใช้ไอซีตรวจจับอุณหภูมิ DS1820 ที่ควบคุมการทำงานโดย MCS 51 ดังรูปที่ 5.2



รูปที่ 5.2 แสดงวงจรตรวจจับอุณหภูมิ

#### 5.1.2.1 ไอซีตรวจจับอุณหภูมิ DS 1820

เป็นไอซีตรวจจับอุณหภูมิที่ใช้การติดต่อแบบระบบบัส 1 สาย มีขาต่อการใช้งานเพียง 3 ขา คือ DQ ซึ่ง เป็นขาเชื่อมต่อกับระบบบัสขาต่อไฟเลี้ยงภายนอกและขากราวด์ดังแสดงการจัสดังรูปที่ 5.3

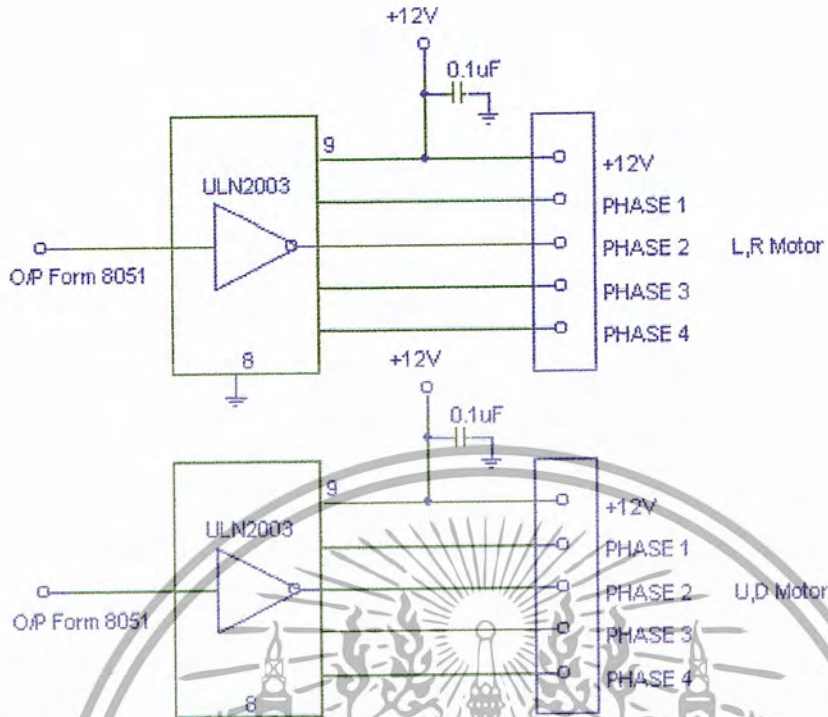


รูปที่ 5.3 DS1820

### 5.1.3 วงจรควบคุมการหมุนกลิ้ง

ส่วนของมอเตอร์นั้นจะใช้สเต็ปปีงมอเตอร์จำนวนสองตัว คือ ส่วนการหมุน ซ้ายกับขวา และ ส่วนการหมุน ขึ้นกับลง ซึ่งควบคุมจากโปรแกรมผ่านระบบอินเตอร์เน็ตและจะส่งสัญญาณการควบคุมจากไมโครคอนโทรลเลอร์มายังมอเตอร์โดยต้องผ่านอุปกรณ์ Buffer เพื่อขยายกระแสเนื่องจาก ตัวไมโครคอนโทรลเลอร์เองไม่สามารถจะจ่ายกระแสเพื่อขับมอเตอร์ได้เพียงพอ ซึ่งวงจรแสดงดังรูปที่ 5.4

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 5.4 แสดงวงจรควบคุมมอเตอร์

#### 5.1.4 วงจรตรวจสอบสถานะอุปกรณ์ไฟฟ้า

พิจารณาจากวงจรจะเห็นว่าไดโอด D1, D2 ต่ออนุกรมกับปลั๊กไฟ AC OUTLET และจะต่อกับสายไฟ 220 โวลต์ ทั้งสองเส้นในขณะที่ไม่มีโหลดที่ AC OUTLET ทราานซิสเตอร์ Q1 จะไม่ทำงานเพราะไม่มีการไบอัสที่ขาเบส ถ้ามีโหลดที่ AC OUTLET ทั้ง D1, D2 ก็จะสลับกันทำงานโดยที่ D1 จะทำงานในช่วงลบ ส่วน D2 ทำงานในช่วงบวกของไฟสลับ ขณะที่ D1 ทำงานในช่วงบวกนี้จะมีแรงดันตกคร่อม D1 ประมาณ 0.7 โวลต์ ทำให้ Q1 ทำงานและรีเลย์ RY1 ทำงานด้วยหน้าสัมผัสมาแตงที่ตำแหน่ง NO เพื่อต่อขา C กับกราวด์ส่งค่าลอจิก "0" ให้ ไมโครคอนโทรลเลอร์ซึ่งในสภาวะปกติขา C จะต่อกับขา NC ซึ่งต่ออยู่กับไฟบวก 5 โวลต์ ซึ่งจะส่งค่าลอจิกเป็น "1"

ในขณะที่ D1 ทำงานในช่วงลบทราานซิสเตอร์ Q1 จะหยุดทำงานเพราะเป็นการไบอัสกลับหน้าสัมผัสของรีเลย์ RY1 จึงดีดกลับไปทีเดิมจึงทำให้แรงดันที่จะจ่ายให้กับ โหลดขาดเป็นช่วงๆเราจึงแก้ไขโดยต่อตัวเก็บประจุ C1 ขนานกับขดลวดของ RY1 ตัวเก็บประจุจะทำการเก็บประจุขณะที่ Q1 ทำงานและในขณะที่ Q1 หยุดทำงาน ประจุใน C1 ก็จะจ่ายให้กับ RY1 จะเห็นว่าถ้ามีโหลดที่ AC OUTLET หน้าสัมผัสของ RY1 จะถูกดึงค้างไว้ที่ NO ตลอดเวลา ซึ่งวงจรแสดงดังรูปที่ 5.5

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

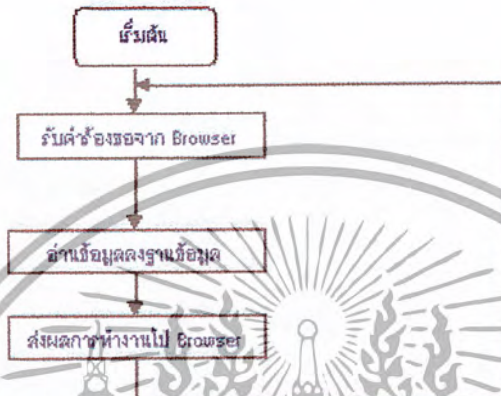


## 5.2 ส่วนโปรแกรม

### 5.2.1 โปรแกรมควบคุมบน Server (ASP)

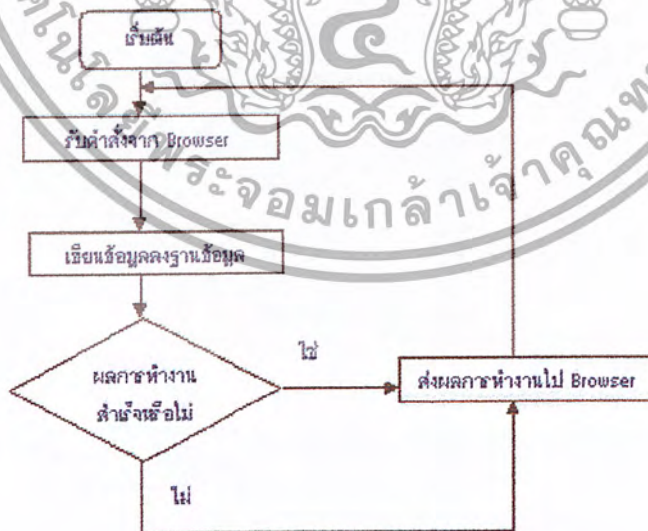
การทำงานของโปรแกรมจะแยกออกเป็นสองส่วนด้วยกัน

1. การอ่านข้อมูลจากฐานข้อมูลแล้วส่งไปแสดงผลที่ Browser ตามคำร้องขอของ Browser ซึ่งการทำงานจะเป็นไปตามรูปที่ 5.7



รูปที่ 5.7 แสดง Flow Chart การทำงานของโปรแกรม ASP

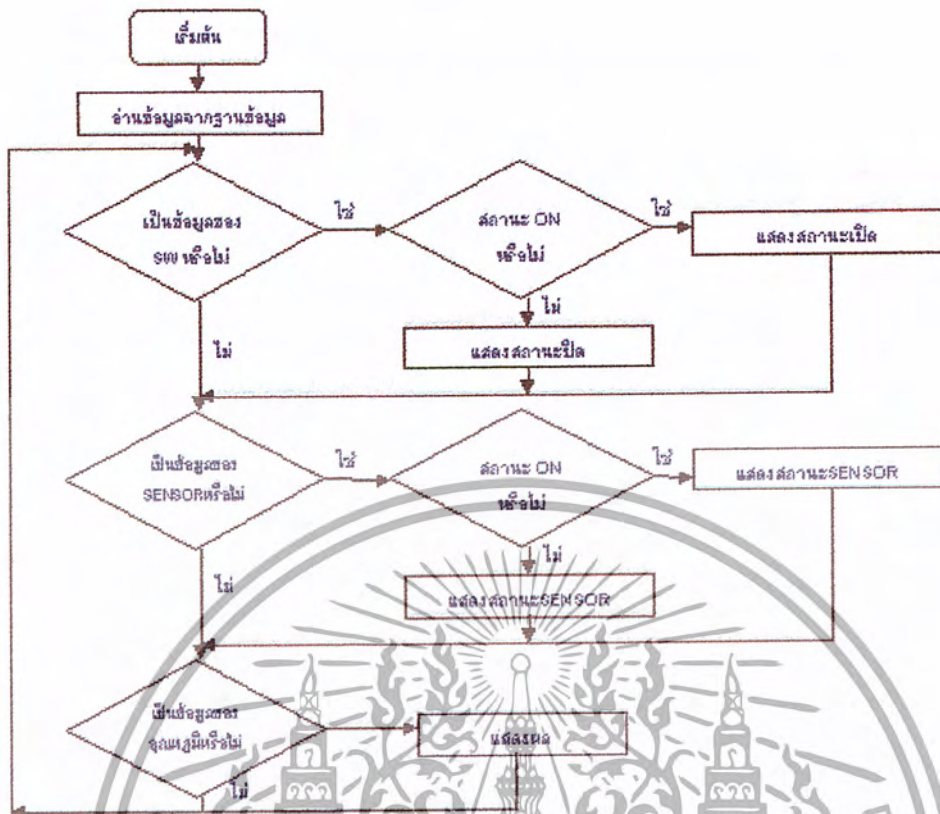
2. การเขียนข้อมูลที่ฐานข้อมูลซึ่งเป็นการส่งงาน โดย Browser ผ่านระบบ Internet ซึ่ง Server จะนำมาประมวลผลแล้วทำงานตามคำสั่ง โดยการทำงานของ Server จะเขียนข้อมูลไปที่ฐานข้อมูลเพื่อให้โปรแกรมในส่วนของ VB นำไปใช้ในการประมวลผลการทำงานอีกทีหนึ่งซึ่งในรอบการทำงานจะส่งผลการทำงานไปยัง Browser ได้ทราบด้วยการทำงานจะเป็นไปตามรูปที่ 5.8



รูปที่ 5.8 แสดง Flow Chart การทำงานของโปรแกรม ASP

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

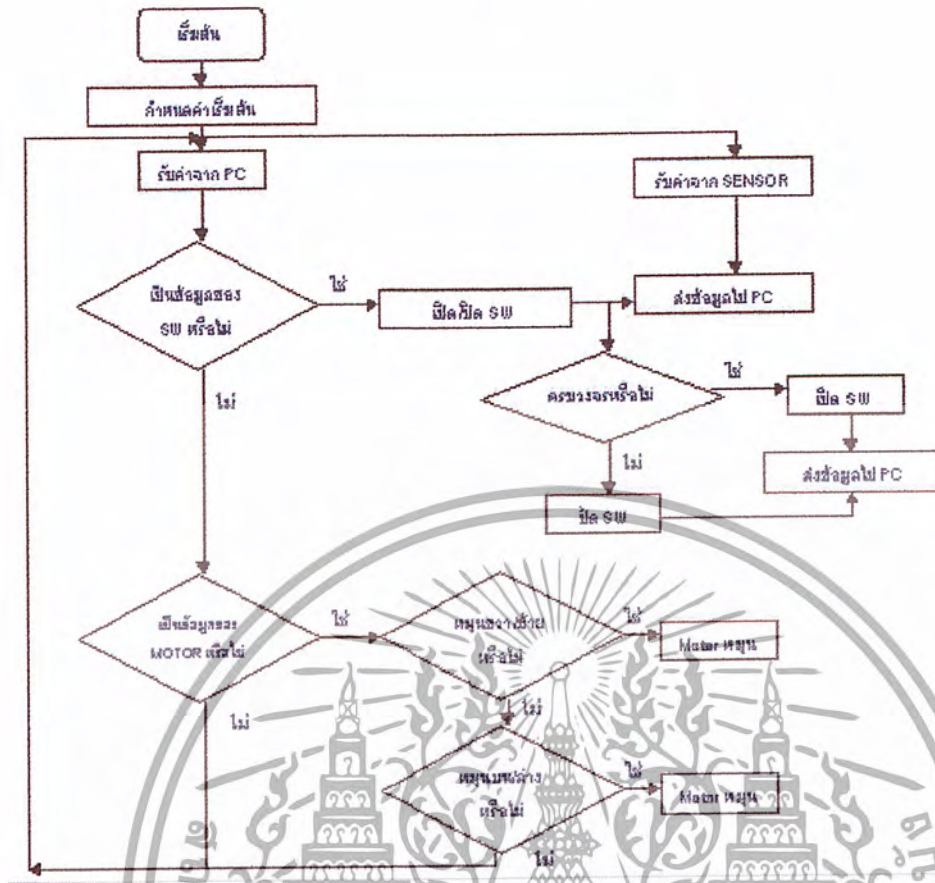




รูปที่ 5.10 แสดง Flow Chart โปรแกรมควบคุมและแสดงผล VB

### 5.2.3 โปรแกรมส่วน Hardware

จะแบ่งการทำงานออกเป็น สองส่วนใหญ่ๆคือ ส่วนของการสั่งการทำงานของ SW และการทำงานของ Motor กับส่วนการรับข้อมูลจากทาง SENSOR ซึ่งมีทั้ง SENSOR ประตูดและ SENSOR อุณหภูมิเมื่อรับข้อมูลจาก SENSOR แล้วจะประมวลผลแล้วส่งข้อมูลไปยัง PC ต่อไป ส่วนโปรแกรมการสั่งงานของ SW และ Motor แสดงดังรูปที่ 5.11



รูปที่ 5.11 แสดง Flow Chart โปรแกรมการทำงาน

ส่วนติดต่อกับ SENSOR ไอซี DS 1820 จะมีคำสั่งที่ต้องส่งให้กับไอซี DS 1820 ดังนี้เพื่อกำหนดรูปแบบการทำงาน คำสั่งที่มีมากที่สุดมีด้วยกัน 3 คำสั่งคือ

1) คำสั่งที่จะไม่ทำการติดต่อกับหน่วยความจำใน DS 1820

เนื่องจากการใช้งาน DS 1820 โดยปกติแล้วจะมี DS 1820 อยู่บนสายสัญญาณเพียงตัวเดียว จึงจำเป็นต้องใช้ข้อมูลของคำสั่งสกริปต์รวมที่ต้องส่งให้ DS 1820 คือ 0CCH

2) คำสั่งแปลงอุณหภูมิ

มีค่าเท่ากับ 44H เมื่อส่งคำสั่งนี้ให้ DS 1820 จะต้องทำการ วนลูปรอบอย่างน้อย 200 มิลลิวินาที เพื่อให้ DS 1820 ได้ใช้เวลาในการแปลงค่าอุณหภูมิเป็นข้อมูลดิจิทัลมาเก็บไว้ในสแต็คซ์แพค

3) คำสั่งอ่านข้อมูลจากสแต็คซ์แพค

มีค่าเท่ากับ 0BEH เมื่อส่งคำสั่งนี้ DS 1820 จะทยอยส่งข้อมูลค่าอุณหภูมิต่อออกมาทั้งหมด 9 ไบต์

#### 5.2.4 การเขียนโปรแกรมเพื่อการใช้งาน DS 1820

เริ่มจากการส่งสถานะด้วยโปรแกรมย่อย DS 1820\_RST จากนั้นให้รอการตอบรับจาก DS 1820 ด้วยโปรแกรมย่อย DS 1820\_PRE เมื่อมีการตอบรับแล้วทำการส่งคำสั่ง CCH (คำสั่ง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

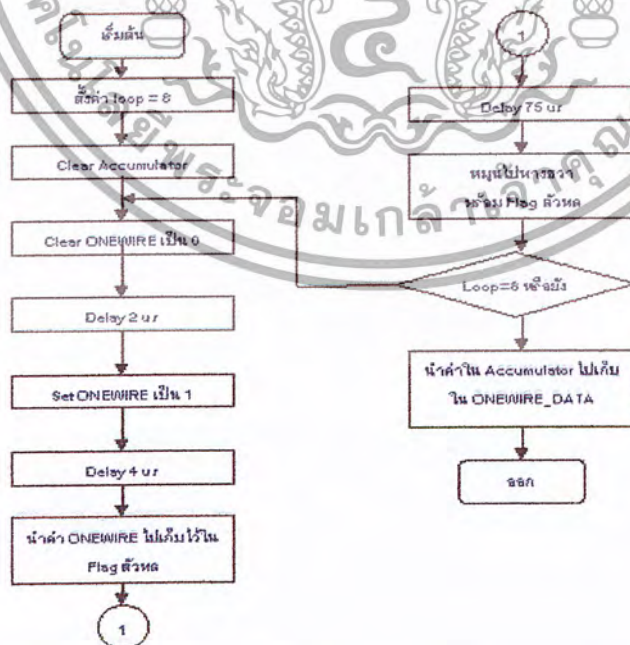
scriptrom) ไปยัง DS 1820 โดยผ่านโปรแกรมย่อย DS 1820\_WR และส่งคำสั่ง 44H (คำสั่งแปลงค่าอุณหภูมิลงในหน่วยความจำ) ไปด้วย

ในขั้นต่อมา ก็เริ่มส่งสถานะรีเซ็ตใหม่แล้วรอการตอบรับเช่นเดิม แต่จะทำการตรวจสอบสถานะบิต BUSY ว่าทำการแปลงอุณหภูมิเสร็จแล้วหรือไม่ ถ้าไม่ก็ให้โปรแกรมรอนจนกว่าจะเสร็จ จากนั้นให้ทำการส่งสถานะรีเซ็ตใหม่ แล้วรอรับการตอบรับเช่นเดิม เมื่อมีการตอบรับแล้วให้ทำการส่งคำสั่ง CCH และคำสั่ง BEH (คำสั่ง เรียกค่าจากหน่วยความจำของ DS 1820 ) จากนั้นให้อ่านค่าจาก DS 1820 ด้วยโปรแกรมย่อย DS 1820\_RD มาเก็บในรีจิสเตอร์ชื่อ TEMP อีกทีหนึ่งเพื่อนำมาใช้งานต่อไป จากนั้นให้ส่งสถานะรีเซ็ตใหม่แล้วรอการตอบรับอีกครั้งหนึ่ง

จากนั้นนำข้อมูลใน TEMP มาทำการหมุนไปทางขวาโดยใช้แฟลกร่วมด้วย โดยทำการเคลียร์ค่าของแฟลคก่อนใช้ด้วย จะได้อุณหภูมิที่เป็นจำนวนเต็มพอดี และทำการส่งค่านี้โดยเรียกจากผ่านโปรแกรมย่อย HEX\_SEND

### 5.2.5 โปรแกรมย่อยในส่วนต่างๆ

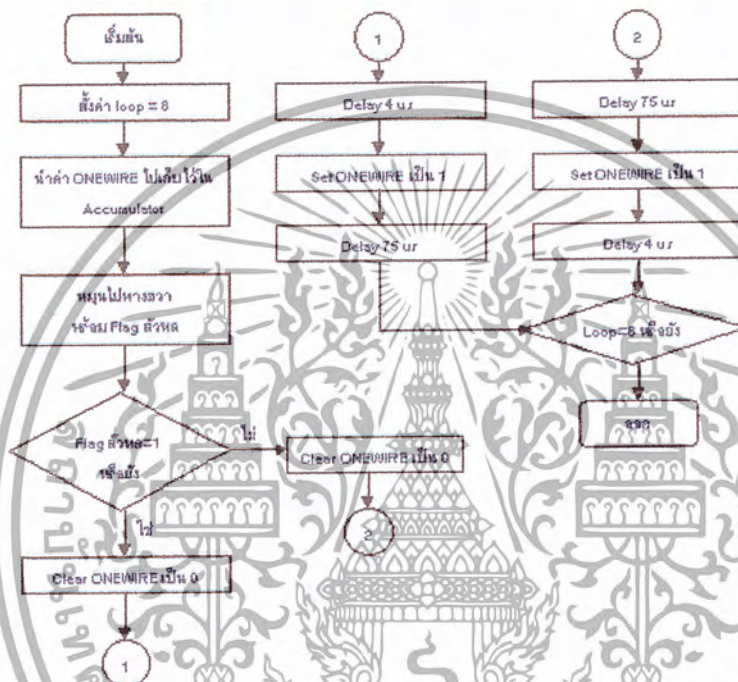
DS 1820\_RD เป็นโปรแกรมย่อยที่ใช้ในการอ่านค่าจาก DS 1820 โดยในขั้นแรกจะทำการเคลียร์สัญญาณที่บิต ONEWIRE ให้เป็น 0 จากนั้นทำการ delay เวลาไปประมาณ 2  $\mu$ sec แล้วเซตสัญญาณ ONEWIRE เป็น 1 แล้ว delay เวลาไปประมาณ 75  $\mu$ sec จึงทำการอ่านค่าสัญญาณที่บิต ONEWIRE เก็บไว้ในแฟลคตัวทศ แล้ว delay เวลาไปประมาณ 75  $\mu$ sec แล้วทำการหมุนข้อมูลไปทางขวาโดยใช้แฟลกร่วมด้วย ทำเช่นนี้ไป 8 ครั้งก็จะได้อ่านค่ามาเก็บในรีจิสเตอร์ ONEWIRE\_DATA สามารถเขียนเป็น flow chart ได้ดังนี้



รูปที่ 5.12 แสดง Flow Chart โปรแกรมย่อย DS 1820\_RD

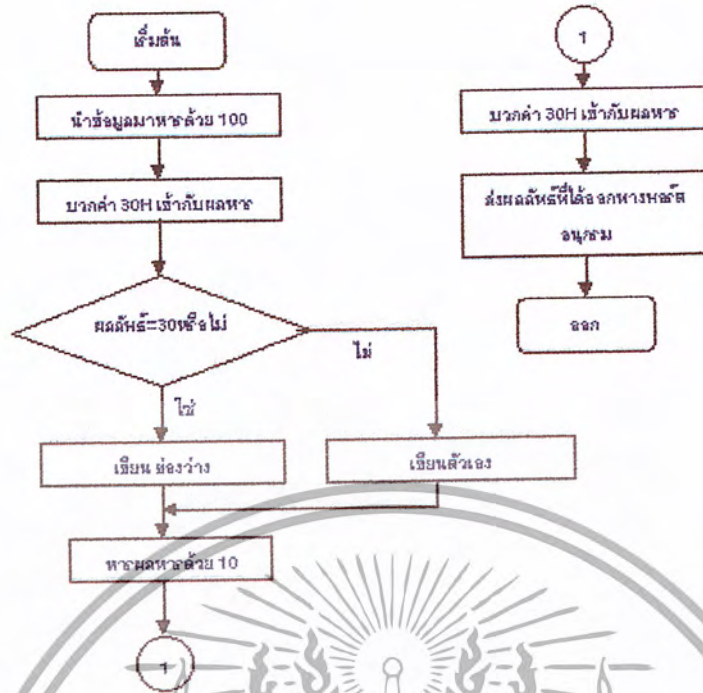
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น เมื่ออนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

DR 1820\_WR เป็นโปรแกรมย่อยที่ใช้ในการเขียนค่าจากรีจิสเตอร์ ONEWIRE\_DATA ไปยัง DS 1820 โดยขั้นแรกจะทำการหมุนข้อมูลไปทางขวาโดยใช้เฟลทกร่วมด้วย จากนั้นให้ทำการตรวจสอบเฟลททว่าถูกเซตหรือไม่ ถ้าถูกเซตก็ให้ทำการเซตบิตสัญญาณ ONEWIRE เป็น 1 เป็นเวลา 75  $\mu$ sec แต่ถ้าไม่ถูกเซตก็ให้ทำการเคลียร์บิตสัญญาณ ONEWIRE เป็น 0 เป็นเวลา 75  $\mu$ sec แล้วทำการเซตสัญญาณ ONEWIRE เป็นเวลา 4  $\mu$ sec แล้วกลับไปหมุนข้อมูลใหม่ รวมเป็น 8 ครั้ง ก็ จะทำการเขียนข้อมูลเสร็จ เขียนเป็น Flow chart ได้ดังนี้



รูปที่ 5.13 แสดง Flow Chart โปรแกรมย่อย DS 1820\_WR

HEX\_SEND เป็นโปรแกรมย่อยที่ใช้แปลงค่าในรีจิสเตอร์มาแสดงเป็นเลขฐาน 10 จำนวน 3 หลัก ที่พอร์ตอนุกรม โดยจะนำค่ามาจากรีจิสเตอร์ LCD\_DATA มาทำการหารด้วย 100 ก่อน แล้วนำค่าตัวส่วนแสดงออกที่พอร์ตอนุกรมเป็นเลขอุณหภูมิหลักแรก แต่ถ้าค่าที่ได้เป็น 0 ก็จะเขียนเป็นช่องว่างแทน ต่อมาก็จะนำเศษที่เหลือจากการหารครั้งแรก มาหารด้วย 10 แล้วนำค่าตัวส่วนแสดงออกไปที่พอร์ตอนุกรมเป็นเลขอุณหภูมิหลักถัดมาเขียนเป็น Flow chart ได้ดังนี้



รูปที่ 5.14 แสดง Flow Chart โปรแกรมย่อย HE\_XSEND

DS1820\_RST เป็นโปรแกรมย่อยที่ใช้สร้างสถานะ reset โดยจะทำการ clear สัญญาณที่บิต ONEWIRE เป็นเวลา 1 ms จากนั้นก็จะทำการ set สัญญาณที่บิต ONEWIRE ประมาณ 16 us เพื่อรอกการตอบรับจาก DS.ต่อไป

DS\_PREP เป็นโปรแกรมย่อยที่ทำหน้าที่รอรับการตอบรับจาก DS 1820 และใช้ทำหน้าที่รอกการสิ้นสุดกระบวนการแปลงอุณหภูมิด้วย โดยจะทำการตรวจสอบว่ามีการ clear สัญญาณ ONEWIRE หรือไม่ในช่วงเวลาประมาณ 8 ms โดยถ้าเกินช่วงเวลานี้ไป set แฟล็ก BUSY ให้เป็น "1" แต่ถ้าตรวจสอบได้ว่ามีการตอบรับ ก็จะรอกจนกระทั่งสัญญาณ ONEWIRE เป็น 1 อีกครั้ง แล้วหน่วงเวลาไปประมาณ 16 us ก่อนที่จะ clear แฟล็ก BUSY เป็น "0" เพื่อแสดงว่ามีการตอบรับกลับ มาจึงเข้าสู่การทำงานของโปรแกรมหลักได้

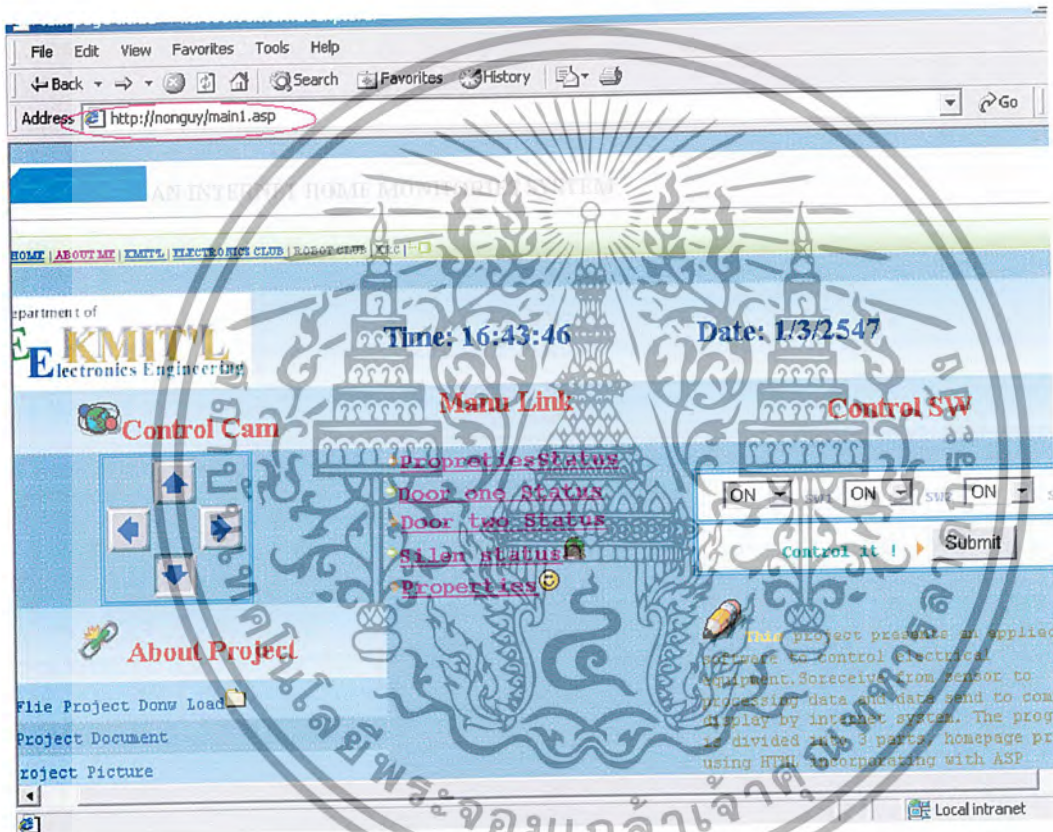
## บทที่ 6

## สรุปผลและวิจารณ์

## 6.1 การทดลองและผลการทดลอง

## 6.1.1 ผลการทดลองที่ Browser


เปิด Browser ขึ้นมาควรเป็น IE V.4 ขึ้นไป ต่อจากนั้นก็พิมพ์ชื่อ โฮมเพจลงในที่นี้คือ nonguy/main1.asp ซึ่งเป็นการจำลอง Server โดยโปรแกรม IIS (Internet Information Server) หน้าแรกจะแสดงเมนูการควบคุมต่างๆไว้ โดยสังเกตที่ช่อง Address จะเป็นนามสกุล .ASP ซึ่งจะหมายถึงมีการทำงานที่ฝั่ง Server แล้วส่งผลมาแสดงผลที่ Browser ดังรูปที่ 6.1



รูปที่ 6.1 แสดงหน้าแรกของโฮมเพจ

ผลการทดลองเมนู ปิด/เปิด SW จะต้องรอผลการทำงาน ซึ่งจะสัมพันธ์กันระหว่างโปรแกรมควบคุมของ VB และ โปรแกรมควบคุมของ ASP ซึ่งเวลาในการแสดงผลประมาณ 17 วินาที ผลการทำงานสำเร็จจะแสดงดังรูปที่ 6.2

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

AN INTERNET HOME MONITORIES SYSTEM	
<a href="#">HOME</a>   <a href="#">ABOUT ME</a>   <a href="#">KMITL</a>   <a href="#">ELECTRONICS CLUB</a>   <a href="#">ROBOT CLUB</a>   <a href="#">ERC</a>	
 Response Display	
หลอดไฟ 1	ON
หลอดไฟ 2	ON
หลอดไฟ 3	ON
<a href="#">Return to MainPage</a>	

รูปที่ 6.2 แสดงผลการทำงานที่สำเร็จ

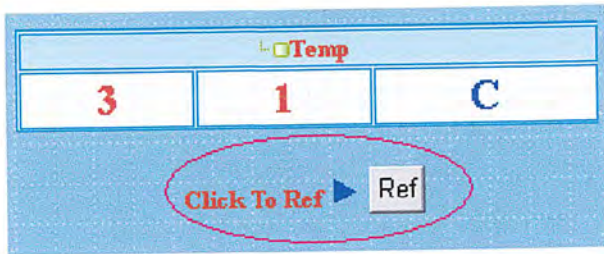
ผลการทำงานที่ไม่สำเร็จ จะแสดงดังรูปที่ 6.3



รูปที่ 6.3 แสดงผลการทำงานที่ไม่สำเร็จ

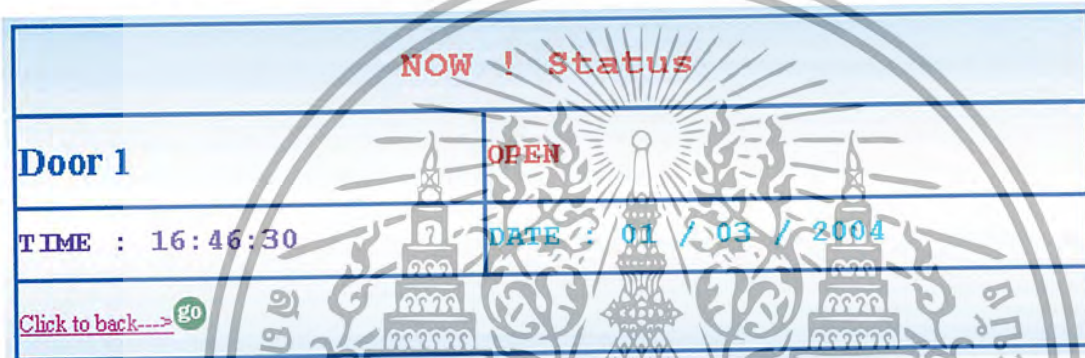
จากนั้นทำการทดสอบโดยการสั่งงานที่แตกต่างกันไปอย่างค่อเนื่องผลการทำงานเป็นไป ด้วยดีโดยไม่มีข้อผิดพลาดใดๆ เรากลับมาที่หน้าแรกอีกครั้งเพื่อดูผลการทำงานอื่นๆ ลองมาตรวจ สอบค่าของอุณหภูมิที่ Browser กับโปรแกรม VB การแสดงผลที่ VB จะแสดงผลทันทีที่มีการ เปลี่ยนแปลงอุณหภูมิ แต่เนื่องจาก โปรแกรม ASP ทำงานโดยการนำข้อมูลจากฐานข้อมูลมาแล้วจึง ค่อยส่งไปแสดงผลที่ Browser จึงต้องมีการร้องขอจาก Browser ก่อนโปรแกรม ASP จึงจะสามารถ ประมวลผลได้โดยมีปุ่ม Refresh สำหรับการทำงานนี้ซึ่งเมื่อมีการกดปุ่ม Refresh โปรแกรม ASP จะ นำข้อมูลอุณหภูมินั้นมาแสดงผลโดยทันทีโดยจะแสดงดังรูปที่ 6.4

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 6.4 แสดงผลอุณหภูมิและปุ่ม Refresh

เราสามารถที่จะดูความเป็นไปของประตูที่เราได้ติดตั้ง Sensor เอาไว้ได้โดยกดไปที่เมนู Door Status โดย โปรแกรม ASP จะตรวจสอบที่ฐานข้อมูลแล้วนำมาแสดงผลที่ Browser โดยจะแสดงสถานะขณะนั้นและรายละเอียดต่างๆเอาไว้ดังรูปที่ 6.5



รูปที่ 6.5 แสดงสถานะประตู

การทดสอบการหมุนของ Motor โดยถ่วงที่ปุ่มตั้งงาน การทำงานจะไม่สามารถตอบสนองการทำงานได้โดยทันทีเนื่องจากการทำงานต้องใช้โปรแกรมในการประมวลผลถึงสามส่วนคือ ASP ตั้งงาน ต่อจากนั้น VB จะนำข้อมูลที่ได้มาประมวลผลแล้วส่งข้อมูลนั้นไปยังโปรแกรมที่ใช้ควบคุม Hardware อีกทีโดยส่วนของปุ่มตั้งงานแสดงดังรูปที่ 6.6 และผลการทดลองแสดงไว้ที่ตาราง 6.1 ตามลำดับ



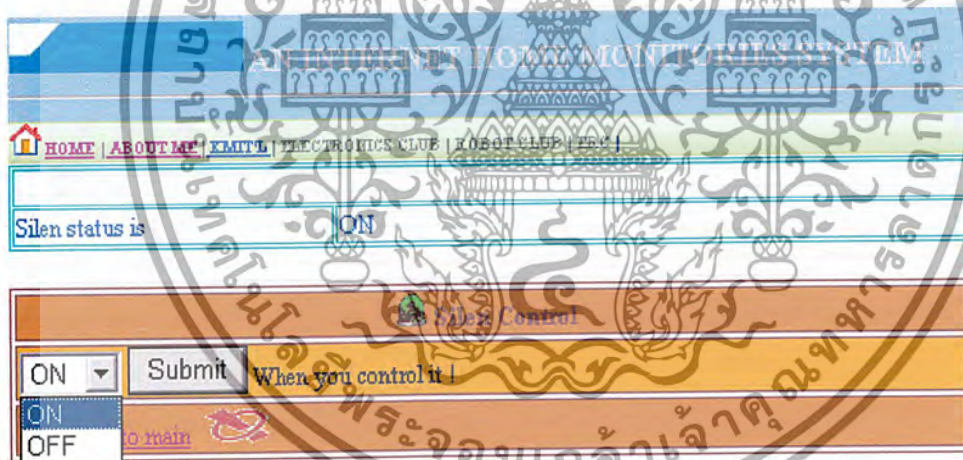
รูปที่ 6.6 แสดงปุ่มควบคุมกล้อง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปแบบการทดสอบ	ผลการทดสอบ
กดปุ่มหมุน ซ้าย/ขวา	ทำงานหลังจากกดปุ่มประมาณ 10 วินาที
กดปุ่มหมุน ขึ้น/ลง	ทำงานหลังจากกดปุ่มประมาณ 10 วินาที เกิดการ กระตุกบ้างเล็กน้อย
กดปุ่มอย่างต่อเนื่อง	ไม่สามารถตอบสนองการทำงานได้ต้องรอรอบการทำงานประมาณ 5 วินาทีโดยประมาณถึงจะทำงานได้อีก

### ตารางที่ 6.1 แสดงผลการควบคุมกล้อง

การทดสอบเมนูที่ตรวจสอบระบบเตือนภัยซึ่งยังสามารถที่จะควบคุมการทำงานของระบบเตือนภัยอีกด้วยการตรวจสอบนั้นโปรแกรม ASP จะตรวจสอบไปที่ฐานข้อมูลแล้วนำข้อมูลมาแสดงผลที่ Browser เมื่อเรามีความต้องการที่จะยกเลิกหรือปิดการทำงานก็สามารถสั่งงาน โดยหน้าเว็บเพจนี้ได้ทันที ผลการทำงานจะคล้ายการทำงานของการสั่งงาน เปิด/ปิด SW โดยใช้เวลาในการทำงานรอบละประมาณ 17 วินาที แสดงไว้ที่รูป 6.7

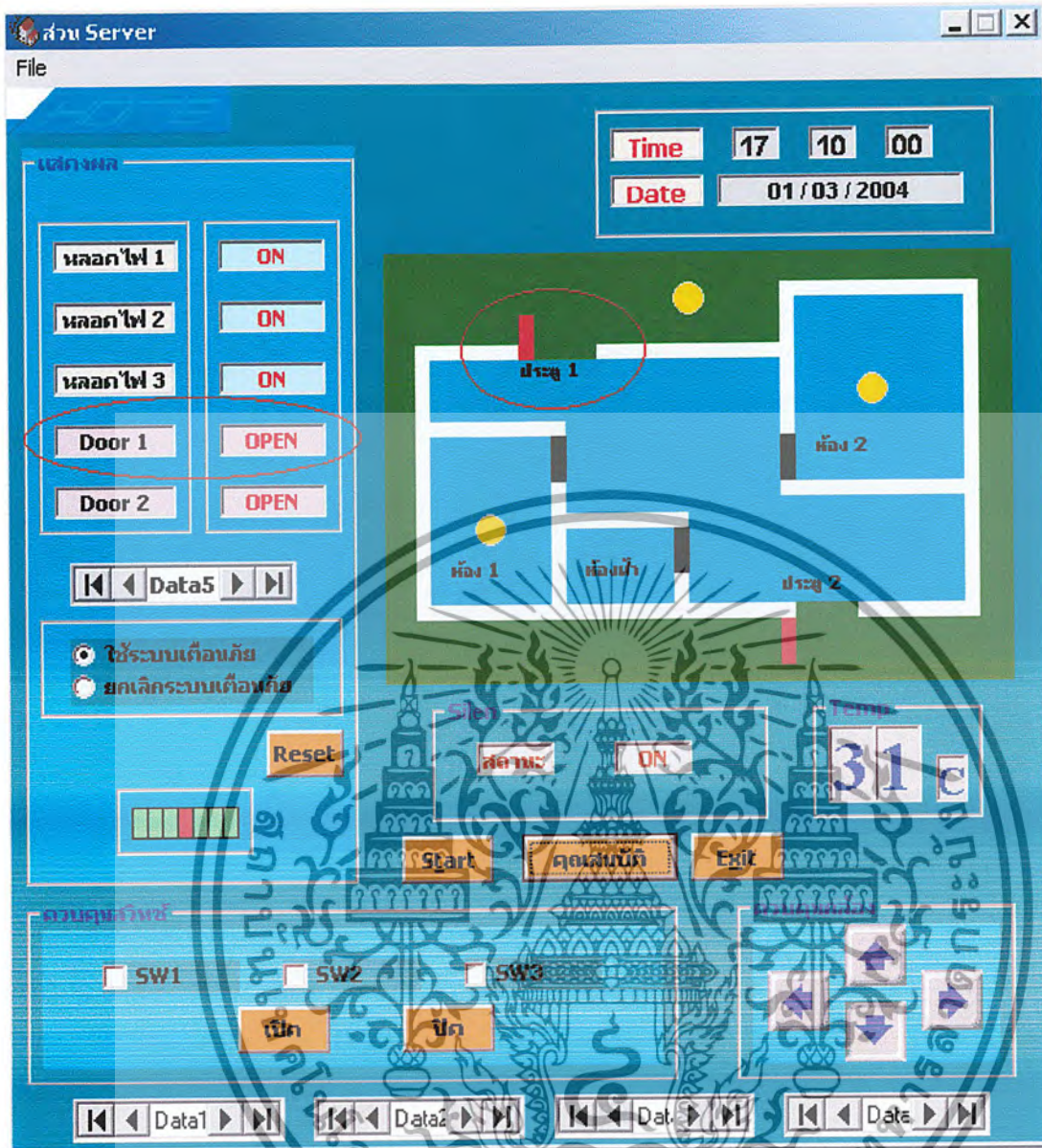


รูปที่ 6.7 แสดงสถานะ Silent และปุ่มควบคุม

#### 6.1.2 ผลการทดลองที่ Server

การทำงานของระบบจะต้องให้โปรแกรม VB ทำงานอยู่ตลอดเวลาซึ่งโปรแกรมในส่วนนี้จะทำงานร่วมกันกับ โปรแกรม ASP เมื่อใดก็ตามโปรแกรมใดโปรแกรมหนึ่งมีการประมวลผล จะส่งผลให้อีกโปรแกรมหนึ่งทำงานตาม และผลที่ได้ก็จะสอดคล้องกันด้วยเช่น ผลการแสดงผลสถานะของประตูที่ Browser ซึ่งแสดงดังรูปที่ 6.5 แล้วลองเปรียบเทียบกับการทำงานที่ Server โดยโปรแกรม VB ซึ่งจะเป็นไปในลักษณะเดียวกันดังรูปที่ 6.8

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 6.8 แสดงการทำงานของโปรแกรม VB

ทดลองการควบคุมการ เปิด/ปิด SW การทำงานจะเห็นได้อย่างชัดเจนว่าสามารถทำงานได้ไวมากเกือบจะพร้อมกับการสั่งงานเลยทีเดียซึ่งจะแตกต่างกับการควบคุมผ่าน Browser เพราะจะเป็นการสั่งงานและรับผลการทำงาน โดยตรง เมื่อลองทดสอบอย่างต่อเนื่องและในหลายๆรูปแบบ ผลการทำงานเป็นไปด้วยดีไม่มีข้อผิดพลาดใด

ทดสอบอุปกรณ์ SENSOR วัดอุณหภูมิซึ่งทดสอบโดย การใช้หัวแร้งในการทำให้ได้รับความร้อนและใช้น้ำแข็งเพื่อให้ได้รับความเย็นซึ่งทำการทดสอบดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปแบบการทดสอบ	เวลาทดสอบ	ผลการทดสอบ
ใช้น้ำแข็งทันที	10 วินาที	เปลี่ยนแปลงทุก 1.5 วินาที
ใช้หัวแรงทันที	10 วินาที	เปลี่ยนแปลงทุก 1.3 วินาที
ใช้หัวแรงขณะที่อุณหภูมิ ต่ำๆ	20 วินาที	5-7 วินาทีแรกอุณหภูมิ เปลี่ยนแปลงทุก 1 วินาที หลังจากนั้นจะเปลี่ยน แปลงทุก 1.4 วินาทีจน อุณหภูมิคงที่
ใช้น้ำแข็งขณะที่อุณหภูมิ สูงๆ	20 วินาที	5-7 วินาทีแรกอุณหภูมิ เปลี่ยนแปลงทุก 1.2 วินาที หลังจากนั้นการเปลี่ยน แปลงจะประมาณ 1.4 วินาทีจนอุณหภูมิคงที่

#### ตารางที่ 6.2 แสดงผลการทดสอบอุณหภูมิ

ทดสอบการเปิดและปิดอุปกรณ์ SW และควบคุม Motor การควบคุมเป็นไปด้วยความถูกต้อง โดยสามารถที่จะเปิดและปิดอุปกรณ์ SW ได้ในทันทีเมื่อนำการสั่งการทำงานและการแสดงผลก็สามารถที่จะแสดงผลได้โดยทันที

ส่วนการสั่งงานควบคุม Motor ทำได้ดีพอสมควรแต่จะไม่เกิดขึ้นทันทีที่จะทำงานช้ากว่าประมาณ 1/2 วินาทีและไม่สามารถจะสั่งการควบคุมได้โดยต่อเนื่องดังตารางที่ 6.3

รูปแบบการทดสอบ	ผลการทดสอบ
กดปุ่มหมุน ซ้าย/ขวา	ทำงานหลังจากกดปุ่มประมาณ 1/2 วินาที
กดปุ่มหมุน ขึ้น/ลง	ทำงานหลังจากกดปุ่มประมาณ 1/2 วินาที เกิดการ กระตุกบ้างเล็กน้อย
กดปุ่มอย่างต่อเนื่อง	ไม่สามารถตอบสนองการทำงานได้ต้องรอรอบการทำงานประมาณ 1 วินาที โดยประมาณถึงจะทำงานได้อีก

#### ตารางที่ 6.3 แสดงผลการทำงานของ Motor เมื่อสั่งงาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## 6.2 ปัญหาในการทำโครงการงาน

- 6.2.1 อุปกรณ์ในการทำโครงการงานไม่ครบถ้วน
- 6.2.2 ใช้อุปกรณ์ในการทำโครงการงานในราคาสูง
- 6.2.3 การทำงานของโปรแกรมยังไม่เป็นไปตามที่ต้องการ
- 6.2.4 โปรแกรมขาดเสถียรภาพ

## 6.3 วิธีแก้ไขปัญหา

- 6.3.1 วางแผนงานและเตรียมอุปกรณ์ให้พร้อม
- 6.3.2 ใช้อุปกรณ์ที่มีอยู่มากัดแปลงให้มากที่สุด
- 6.3.3 ทดสอบโปรแกรมในแต่ละจุดและทำการแก้ไขความบกพร่องที่เกิดขึ้น

## 6.4 แนวทางในการพัฒนาต่อไป

- 6.4.1 เพิ่มระบบตรวจสอบการทำงานของอุปกรณ์
- 6.4.2 เพิ่ม SENSOR สำหรับความสะอาดและปลอดภัยเช่น ตรวจจับควัน เป็นต้น
- 6.4.3 ออกแบบให้การแสดงผลทาง Browser เป็นภาพกราฟฟิก
- 6.4.4 ให้การทำงานและการแสดงผลมีความเร็วมากยิ่งขึ้น
- 6.4.5 ออกแบบกล่องอุปกรณ์ให้มีความสะดวกในการนำไปใช้งานจริง

## 6.5 ข้อเสนอแนะเกี่ยวกับโครงการงาน

- 6.5.1 ไมโครคอนโทรลเลอร์ควรใช้บอร์ดที่มีหน่วยความจำแบบแฟลช (Flash Memory) ซึ่งสามารถ โปรแกรมได้หลายครั้งเพื่อความสะดวกและง่ายในการทำการแก้ไขในโปรแกรมซ้ำแล้วซ้ำอีกและยังทำให้เกิดความประหยัดขึ้นอีกด้วย
- 6.5.2 ขณะทำการออกแบบหรือสร้าง โปรแกรมในส่วน PC ควรทำการออกแบบโปรแกรมในส่วนของ Hardware ไปพร้อมๆกันด้วยเพื่อเป็นการง่ายในการทดสอบและแก้ไขเมื่อเกิดความผิดพลาดขึ้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## ภาคผนวก ก.โปรแกรมวิชาการเบสิก

```
Dim data As Variant
```

```
***** ส่วนโปรแกรมเปิด สวิตช์*****
```

```
Private Sub CmdON_Click()
```

```
Dim Val As Variant, Check As Variant ' กำหนดค่าตัวแปร
```

```
Dim Answer As Integer ' กำหนดตัวแปรใช้ในการตัดสินใจ OK,Cancel
```

```
Val = MSComm1.Input
```

```
If Check1.Value = Unchecked And Check2.Value = Unchecked And Check3.Value = Unchecked
```

```
Then 'ตรวจสอบเงื่อนไขการเลือก
```

```
MsgBox "คุณยังไม่ได้เลือกอุปกรณ์", vbOKOnly + vbCritical, "Error" 'เตือนเมื่อยังไม่มีกร  
เลือก
```

```
Else
```

```
Answer = MsgBox("คุณกำลังจะเปิดอุปกรณ์", vbOKCancel + vbInformation, "SW_On") 'ยืนยัน  
การทำงาน
```

```
If Answer = vbOK Then 'ถ้าเลือก OK
```

```
*****ตรวจสอบเงื่อนไขการทำงาน*****
```

```
If Check1.Value = Checked And Check2.Value = Unchecked And Check3.Value = Unchecked
```

```
Then
```

```
MSComm1.Output = "B"
```

```
Data2.Recordset.Edit
```

```
Data2.Recordset.Fields("ch1") = "ON"
```

```
Data2.Recordset.Update
```

```
ElseIf Check1.Value = Unchecked And Check2.Value = Checked And Check3.Value =
```

```
Unchecked Then
```

```
MSComm1.Output = "C"
```

```
Data2.Recordset.Edit
```

```
Data2.Recordset.Fields("ch2") = "ON"
```

```
Data2.Recordset.Update
```

```
ElseIf Check1.Value = Unchecked And Check2.Value = Unchecked And Check3.Value =
```

```
Checked Then
```

```
MSComm1.Output = "D"
```

```
Data2.Recordset.Edit
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

Data2.Recordset.Fields("ch3") = "ON"
Data2.Recordset.Update
ElseIf Check1.Value = Checked And Check2.Value = Checked And Check3.Value = Unchecked
Then
MSComm1.Output = "B"
MSComm1.Output = "C"
Data2.Recordset.Edit
Data2.Recordset.Fields("ch1") = "ON"
Data2.Recordset.Update
Data2.Recordset.Edit
Data2.Recordset.Fields("ch2") = "ON"
Data2.Recordset.Update
ElseIf Check1.Value = Unchecked And Check2.Value = Checked And Check3.Value = Checked
Then
MSComm1.Output = "C"
MSComm1.Output = "D"
Data2.Recordset.Edit
Data2.Recordset.Fields("ch2") = "ON"
Data2.Recordset.Update
Data2.Recordset.Edit
Data2.Recordset.Fields("ch3") = "ON"
Data2.Recordset.Update
ElseIf Check1.Value = Checked And Check2.Value = Unchecked And Check3.Value = Checked
Then
MSComm1.Output = "B"
MSComm1.Output = "D"
Data2.Recordset.Edit
Data2.Recordset.Fields("ch1") = "ON"
Data2.Recordset.Update
Data2.Recordset.Edit
Data2.Recordset.Fields("ch3") = "ON"
Data2.Recordset.Update

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

ElseIf Check1.Value = Checked And Check2.Value = Checked And Check3.Value = Checked
Then
MSComm1.Output = "B"
MSComm1.Output = "C"
MSComm1.Output = "D"
Data2.Recordset.Edit
Data2.Recordset.Fields("ch1") = "ON"
Data2.Recordset.Update
Data2.Recordset.Edit
Data2.Recordset.Fields("ch2") = "ON"
Data2.Recordset.Update
Data2.Recordset.Edit
Data2.Recordset.Fields("ch3") = "ON"
Data2.Recordset.Update
End If
Else 'ถ้าหากเลือก Cancel
Exit Sub
End If
End If
End Sub
*****ส่วน โปรแกรมปิดอุปกรณ์*****
Private Sub CmdOFF_Click()
Dim Val1 As Variant ' กำหนดค่าตัวแปร
Dim Anwer1 As Integer 'กำหนดตัวแปร ใช้ในการตัดสินใจเลือก OK,Cancel
Val1 = MSComm1.Input
If Check1.Value = Unchecked And Check2.Value = Unchecked And Check3.Value = Unchecked
Then 'ตรวจสอบเงื่อนไขการเลือก
MsgBox "คุณยังไม่ได้เลือกอุปกรณ์", vbOKOnly + vbCritical, "Error" 'เตือนเมื่อยังไม่มีกร
เลือก
Else
Anwer1 = MsgBox("คุณกำลังจะปิดอุปกรณ์", vbOKCancel + vbInformation, "SW_Off")

```

#### 'ยืนยันหยุดการทำงาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

If Anwer1 = vbOK Then 'ถ้าเลือก OK

\*\*\*\*\*ตรวจสอบเงื่อนไขการทำงาน\*\*\*\*\*

```

If Check1.Value = Checked And Check2.Value = Unchecked And Check3.Value =
Unchecked Then
MSComm1.Output = "z"
Data2.Recordset.Edit
Data2.Recordset.Fields("ch1") = "OFF"
Data2.Recordset.Update
ElseIf Check1.Value = Unchecked And Check2.Value = Checked And Check3.Value =
Unchecked Then
MSComm1.Output = "x"
Data2.Recordset.Edit
Data2.Recordset.Fields("ch2") = "OFF"
Data2.Recordset.Update
ElseIf Check1.Value = Unchecked And Check2.Value = Unchecked And Check3.Value =
Checked Then
MSComm1.Output = "c"
Data2.Recordset.Edit
Data2.Recordset.Fields("ch3") = "OFF"
Data2.Recordset.Update
ElseIf Check1.Value = Checked And Check2.Value = Checked And Check3.Value = Unchecked
Then
MSComm1.Output = "z"
MSComm1.Output = "x"
Data2.Recordset.Edit
Data2.Recordset.Fields("ch1") = "OFF"
Data2.Recordset.Update
Data2.Recordset.Edit
Data2.Recordset.Fields("ch2") = "OFF"
Data2.Recordset.Update
ElseIf Check1.Value = Unchecked And Check2.Value = Checked And Check3.Value = Checked
Then

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

Data2.Recordset.Fields("ch2") = "OFF"
Data2.Recordset.Update
ElseIf Check1.Value = Unchecked And Check2.Value = Unchecked And Check3.Value =
Checked Then
MSComm1.Output = "c"
Data2.Recordset.Edit
Data2.Recordset.Fields("ch3") = "OFF"
Data2.Recordset.Update
ElseIf Check1.Value = Checked And Check2.Value = Checked And Check3.Value = Unchecked
Then
MSComm1.Output = "z"
MSComm1.Output = "x"
Data2.Recordset.Edit
Data2.Recordset.Fields("ch1") = "OFF"
Data2.Recordset.Update
Data2.Recordset.Edit
Data2.Recordset.Fields("ch2") = "OFF"
Data2.Recordset.Update
ElseIf Check1.Value = Unchecked And Check2.Value = Checked And Check3.Value = Checked
Then
MSComm1.Output = "x"
MSComm1.Output = "c"
Data2.Recordset.Edit
Data2.Recordset.Fields("ch2") = "OFF"
Data2.Recordset.Update
Data2.Recordset.Edit
Data2.Recordset.Fields("ch3") = "OFF"
Data2.Recordset.Update
ElseIf Check1.Value = Checked And Check2.Value = Unchecked And Check3.Value = Checked
Then
MSComm1.Output = "z"
MSComm1.Output = "c"

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

Data2.Recordset.Edit
Data2.Recordset.Fields("ch1") = "OFF"
Data2.Recordset.Update
Data2.Recordset.Edit
Data2.Recordset.Fields("ch3") = "OFF"
Data2.Recordset.Update
ElseIf Check1.Value = Checked And Check2.Value = Checked And Check3.Value = Checked
Then
MSComm1.Output = "z"
MSComm1.Output = "x"
MSComm1.Output = "c"
Data2.Recordset.Edit
Data2.Recordset.Fields("ch1") = "OFF"
Data2.Recordset.Update
Data2.Recordset.Edit
Data2.Recordset.Fields("ch2") = "OFF"
Data2.Recordset.Update
Data2.Recordset.Edit
Data2.Recordset.Fields("ch3") = "OFF"
Data2.Recordset.Update
End If
Else ' ถ้าเลือก Cancel
Exit Sub
End If
End If
End Sub

```

\*\*\*\*\*ยืนยันการใช้ & ไม่ใช้ระบบเตือนภัย\*\*\*\*\*

```
Private Sub Cmdchk_Click()
```

```
Dim Anwer1 As Integer
```

```
If Option1.Value = True Then
```

```
Anwer1 = MsgBox("คุณต้องการใช้ระบบเตือนภัย", vbOKCancel + vbInformation, "Silen_On")
```

```
If Anwer1 = vbOK Then
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

MSComm1.Output = "x"
MSComm1.Output = "c"
Data2.Recordset.Edit
Data2.Recordset.Fields("ch2") = "OFF"
Data2.Recordset.Update
Data2.Recordset.Edit
Data2.Recordset.Fields("ch3") = "OFF"
Data2.Recordset.Update
ElseIf Check1.Value = Checked And Check2.Value = Unchecked And Check3.Value = Checked
Then
MSComm1.Output = "z"
MSComm1.Output = "c"
Data2.Recordset.Edit
Data2.Recordset.Fields("ch1") = "OFF"
Data2.Recordset.Update
Data2.Recordset.Edit
Data2.Recordset.Fields("ch3") = "OFF"
Data2.Recordset.Update
ElseIf Check1.Value = Checked And Check2.Value = Checked And Check3.Value = Checked
Then
MSComm1.Output = "z"
MSComm1.Output = "x"
MSComm1.Output = "c"
Data2.Recordset.Edit
Data2.Recordset.Fields("ch1") = "OFF"
Data2.Recordset.Update
Data2.Recordset.Edit
Data2.Recordset.Fields("ch2") = "OFF"
Data2.Recordset.Update
Data2.Recordset.Edit
Data2.Recordset.Fields("ch3") = "OFF"
Data2.Recordset.Update

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

End If

Else ' ถ้าเลือก Cancel

Exit Sub

End If

End If

End Sub

\*\*\*\*\*ยืนยันการใช้ & ไม่ใช้ระบบเตือนภัย\*\*\*\*\*

Private Sub Cmdchk\_Click()

Dim Anwer1 As Integer

If Option1.Value = True Then

Anwer1 = MsgBox("คุณต้องการใช้ระบบเตือนภัย", vbOKCancel + vbInformation, "Silen\_On")

If Anwer1 = vbOK Then

Frmsir.Visible = True 'แสดงเฟรมของไซเรน

Else

Option1.Value = False

Option2.Value = True

End If

Else: Option2.Value = True

Anwer1 = MsgBox("คุณต้องการยกเลิกระบบเตือนภัย", vbOKCancel + vbInformation,

"Silen\_Off")

If Anwer1 = vbOK Then

Frmsir.Visible = False 'ซ่อนเฟรมของไซเรน

Timer3.Enabled = False

Else

Option1.Value = True

Option2.Value = False

End If

End If

\*\*\*\*\*

If Frmsir.Visible = True And Timer3.Enabled = True Then

Data1.Recordset.Edit

Data1.Recordset.Fields("sil") = "ON"

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

Data1.Recordset.Update
MSComm1.Output = "F"
End If
If Frmsir.Visible = False And Timer3.Enabled = False Then
Data1.Recordset.Edit
Data1.Recordset.Fields("sil") = "OFF"
Data1.Recordset.Update
MSComm1.Output = "b"
End If
*****

Cmdchk.Visible = False
End Sub
' *****เริ่มต้นตรวจสอบสถานะ *****
Private Sub Cdmstart_Click()
MSComm1.Output = "9" 'เมื่อเปิดโปรแกรมครั้งแรกให้ตรวจสอบสถานะขณะนั้น โดยส่งค่าไป
ตรวจสอบ
End Sub
' *****เมื่อเปิดโปรแกรมขึ้นมาครั้งแรก *****
Private Sub Form_Load()
MSComm1.Settings = "19200,n,8,1"
MSComm1.CommPort = "1"
MSComm1.RThreshold = "1"
MSComm1.InputLen = "1"
MSComm1.PortOpen = True
MSComm1.Output = "9" 'เมื่อเปิดโปรแกรมครั้งแรกให้ตรวจสอบสถานะขณะนั้น โดยส่งค่าไป
ตรวจสอบ
End Sub
' *****ควบคุมกล้อง *****
Private Sub Image1_Click() 'หมุนกล้องขึ้น
Dim delay As Variant
delay = 0
Do Until delay = 15
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

```

MSComm1.Output = "L"
delay = delay + 1
Loop
End Sub
Private Sub Image2_Click() 'หมุนซ้าย
Dim delay As Variant
delay = 0
Do Until delay = 15
MSComm1.Output = "U"
delay = delay + 1
Loop
End Sub
Private Sub Image3_Click() 'หมุนขวา
Dim delay As Variant
delay = 0
Do Until delay = 15
MSComm1.Output = "d"
delay = delay + 1
Loop
End Sub
Private Sub Image4_Click() 'หมุนลง
Dim delay As Variant
delay = 0
Do Until delay = 15
MSComm1.Output = "R"
delay = delay + 1
Loop
End Sub
*****Menu Exit*****
Private Sub mnuexit_Click()
Dim Answers2 As Integer

```



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
Anwers2 = MsgBox("คุณกำลังจะออกจากโปรแกรมนี้", vbOKCancel + vbInformation,
```

```
"Home_Link")
```

```
If Anwers2 = vbOK Then
```

```
End
```

```
Else: anewers2 = Cancel
```

```
Exit Sub
```

```
End If
```

```
End Sub
```

```
*****Menu Propertise*****
```

```
Private Sub mnuprop_Click()
```

```
Me.Hide
```

```
Form2.Show
```

```
End Sub
```

```
*****แสดงคุณสมบัติ*****
```

```
Private Sub cdmProb_Click()
```

```
Me.Hide
```

```
Form2.Show
```

```
End Sub
```

```
'Private Sub Surchek_Click()
```

```
'If Surchek.Value = True Then
```

```
'MSComm1.Output = "b"
```

```
'End If
```

```
'End Sub
```

```
Private Sub Reset_Click()
```

```
Timer3.Enabled = False
```

```
Data2.Recordset.Edit
```

```
Data2.Recordset.Fields("sil") = "DEFAULT"
```

```
Data2.Recordset.Update
```

```
End Sub
```

```
*****ช่วงเวลาและวันที่*****
```

```
Private Sub Timer1_Timer()
```

```
Dim REP As Variant ตัวแปรตรวจสอบการทำงานที่ส่งค่าจาก mcs_51
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

Lbl_H.Caption = Format(Now, "hh")
Lbl_M.Caption = Format(Now, "nn")
Lbl_S.Caption = Format(Now, "ss")
Labeldate.Caption = Format(Date, "dd / mm / yyyy")
End Sub

Private Sub Timerchk_Timer()
MSComm1.RThreshold = "1"
MSComm1.InputLen = "1"
MSComm1.Output = "9" 'ส่งค่าเพื่อตรวจสอบ อุณหภูมิ
Dim Val As String
Val = MSComm1.Input
***** ส่วนแสดงผลสวิตช์ 1-3 *****
    If Val = "B" Then
Label1.Caption = "ON"
Data1.Recordset.Edit
Data1.Recordset.Fields("ch1") = "ON"
Data1.Recordset.Update
        End If
    If Val = "z" Then
Label1.Caption = "OFF"
Shpsw1.BackColor = &HE0E0E0 'แสดงรูปดวงไฟดับ
Data1.Recordset.Edit
Data1.Recordset.Fields("ch1") = "OFF"
Data1.Recordset.Update
        End If
*****
    If Val = "C" Then
Shpsw2.BackColor = &HFFFF&
Label2.Caption = "ON"
Data1.Recordset.Edit
Data1.Recordset.Fields("ch2") = "ON"
Data1.Recordset.Update

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

End If
If Val = "x" Then
    Label2.Caption = "OFF"
    Data1.Recordset.Edit
Data1.Recordset.Fields("ch2") = "OFF"
Data1.Recordset.Update
    Shpsw2.BackColor = &HE0E0E0 'แสดงรูปดวงไฟ2ดับ
End If

*****

If Val = "D" Then
    Label3.Caption = "ON"
    Data1.Recordset.Edit
Data1.Recordset.Fields("ch3") = "ON"
Data1.Recordset.Update
End If
If Val = "c" Then
    Label3.Caption = "OFF"
    Shpsw3.BackColor = &HE0E0E0 'แสดงรูปดวงไฟ3ดับ
    Data1.Recordset.Edit
Data1.Recordset.Fields("ch3") = "OFF"
Data1.Recordset.Update
End If

*****SENSOR ประตู1*****

If Val = "E" Then
    Label4.Caption = "OPEN"
    CloseDoor1.Visible = False
    OpenDoor1.Visible = True
    Data1.Recordset.Edit
Data1.Recordset.Fields("send1") = "OPEN"
Data1.Recordset.Update
End If

```

If Val = "v" Then 'กำหนดการส่งค่าเพื่อหยุดการทำงาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
CloseDoor1.Visible = True
```

```
OpenDoor1.Visible = False
```

```
Stress1.Visible = False 'หยุดการเตือน
```

```
Data1.Recordset.Edit
```

```
Data1.Recordset.Fields("send1") = "CLOSE"
```

```
Data1.Recordset.Update
```

```
End If
```

```
*****SENSOR ประตู2 *****
```

```
If Val = "G" Then
```

```
Label8.Caption = "OPEN"
```

```
CloseDoor2.Visible = False
```

```
OpenDoor2.Visible = True
```

```
Data1.Recordset.Edit
```

```
Data1.Recordset.Fields("send2") = "OPEN"
```

```
Data1.Recordset.Update
```

```
Elseif Val = "n" Then 'กำหนดการส่งค่าเพื่อหยุดการทำงาน
```

```
Label8.Caption = "CLOSE"
```

```
CloseDoor2.Visible = True
```

```
OpenDoor2.Visible = False
```

```
Stress2.Visible = False 'หยุดการเตือน
```

```
Data1.Recordset.Edit
```

```
Data1.Recordset.Fields("send2") = "CLOSE"
```

```
Data1.Recordset.Update
```

```
End If
```

```
If Val = "BC" Then
```

```
Label1.Caption = "ON"
```

```
Data1.Recordset.Edit
```

```
Data1.Recordset.Fields("ch1") = "ON"
```

```
Data1.Recordset.Update
```

```
*****
```

```
Label2.Caption = "ON"
```

```
Data1.Recordset.Edit
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
Data1.Recordset.Fields("ch2") = "ON"
```

```
Data1.Recordset.Update
```

```
End If
```

```
If Val = "BD" Then
```

```
Label1.Caption = "ON"
```

```
Data1.Recordset.Edit
```

```
Data1.Recordset.Fields("ch1") = "ON"
```

```
Data1.Recordset.Update
```

```
*****
```

```
Label3.Caption = "ON"
```

```
Data1.Recordset.Edit
```

```
Data1.Recordset.Fields("ch3") = "ON"
```

```
Data1.Recordset.Update
```

```
End If
```

```
If Val = "CD" Then
```

```
Label2.Caption = "ON"
```

```
Data1.Recordset.Edit
```

```
Data1.Recordset.Fields("ch2") = "ON"
```

```
Data1.Recordset.Update
```

```
*****
```

```
Label3.Caption = "ON"
```

```
Data1.Recordset.Edit
```

```
Data1.Recordset.Fields("ch3") = "ON"
```

```
Data1.Recordset.Update
```

```
End If
```

```
If Val = "BCD" Then
```

```
Label1.Caption = "ON"
```

```
Data1.Recordset.Edit
```

```
Data1.Recordset.Fields("ch1") = "ON"
```

```
Data1.Recordset.Update
```

```
*****
```

```
Label2.Caption = "ON"
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

Data1.Recordset.Edit

Data1.Recordset.Fields("ch2") = "ON"

Data1.Recordset.Update

*****

Label3.Caption = "ON"

Data1.Recordset.Edit

Data1.Recordset.Fields("ch3") = "ON"

Data1.Recordset.Update

End If

If Val = "zx" Then

Label1.Caption = "OFF"
Shpsw1.BackColor = &HE0E0E0 'แสดงรูปดวงไฟ1 ดับ
Data1.Recordset.Edit
Data1.Recordset.Fields("ch1") = "OFF"
Data1.Recordset.Update
*****
Label2.Caption = "OFF"
Shpsw2.BackColor = &HE0E0E0 'แสดงรูปดวงไฟ2 ดับ
Data1.Recordset.Edit
Data1.Recordset.Fields("ch2") = "OFF"
Data1.Recordset.Update

End If

If Val = "zc" Then

Label1.Caption = "OFF"

Shpsw1.BackColor = &HE0E0E0 'แสดงรูปดวงไฟ1 ดับ

Data1.Recordset.Edit

Data1.Recordset.Fields("ch1") = "OFF"

Data1.Recordset.Update

*****

Label3.Caption = "OFF"

Shpsw3.BackColor = &HE0E0E0 'แสดงรูปดวงไฟ3 ดับ

Data1.Recordset.Edit

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
Data1.Recordset.Fields("ch3") = "OFF"
```

```
Data1.Recordset.Update
```

```
End If
```

```
If Val = "zv" Then
```

```
Label1.Caption = "OFF"
```

```
Data1.Recordset.Edit
```

```
Data1.Recordset.Fields("ch1") = "OFF"
```

```
Data1.Recordset.Update
```

```
*****
```

```
Label4.Caption = "CLOSE"
```

```
Data1.Recordset.Edit
```

```
Data1.Recordset.Fields("send1") = "CLOSE"
```

```
Data1.Recordset.Update
```

```
End If
```

```
If Val = "zn" Then
```

```
Label1.Caption = "OFF"
```

```
Data1.Recordset.Edit
```

```
Data1.Recordset.Fields("ch1") = "OFF"
```

```
Data1.Recordset.Update
```

```
*****
```

```
Label8.Caption = "CLOSE"
```

```
Data1.Recordset.Edit
```

```
Data1.Recordset.Fields("send2") = "CLOSE"
```

```
Data1.Recordset.Update
```

```
End If
```

```
If Val = "xc" Then
```

```
Label2.Caption = "OFF"
```

```
Shpsw2.BackColor = &HE0E0E0 'แสดงรูปดวงไฟ 2 คับ
```

```
Data1.Recordset.Edit
```

```
Data1.Recordset.Fields("ch2") = "OFF"
```

```
Data1.Recordset.Update
```

```
*****
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
Label3.Caption = "OFF"
```

```
Shpsw3.BackColor = &HE0E0E0 'แสดงรูปดวงไฟ3 ดับ
```

```
Data1.Recordset.Edit
```

```
Data1.Recordset.Fields("ch3") = "OFF"
```

```
Data1.Recordset.Update
```

```
End If
```

```
If Val = "xv" Then
```

```
Label2.Caption = "OFF"
```

```
Data1.Recordset.Edit
```

```
Data1.Recordset.Fields("ch2") = "OFF"
```

```
Data1.Recordset.Update
```

```
*****
```

```
Label4.Caption = "CLOSE"
```

```
Data1.Recordset.Edit
```

```
Data1.Recordset.Fields("send1") = "CLOSE"
```

```
Data1.Recordset.Update
```

```
End If
```

```
If Val = "xn" Then
```

```
Label2.Caption = "OFF"
```

```
Data1.Recordset.Edit
```

```
Data1.Recordset.Fields("ch2") = "OFF"
```

```
Data1.Recordset.Update
```

```
*****
```

```
Label8.Caption = "CLOSE"
```

```
Data1.Recordset.Edit
```

```
Data1.Recordset.Fields("send2") = "CLOSE"
```

```
Data1.Recordset.Update
```

```
End If
```

```
If Val = "cv" Then
```

```
Label3.Caption = "OFF"
```

```
Data1.Recordset.Edit
```

```
Data1.Recordset.Fields("ch3") = "OFF"
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
Data1.Recordset.Update
```

```
*****
```

```
Label4.Caption = "CLOSE"
```

```
Data1.Recordset.Edit
```

```
Data1.Recordset.Fields("send1") = "CLOSE"
```

```
Data1.Recordset.Update
```

```
End If
```

```
If Val = "cn" Then
```

```
Label3.Caption = "OFF"
```

```
Data1.Recordset.Edit
```

```
Data1.Recordset.Fields("ch3") = "OFF"
```

```
Data1.Recordset.Update
```

```
*****
```

```
Label8.Caption = "CLOSE"
```

```
Data1.Recordset.Edit
```

```
Data1.Recordset.Fields("send2") = "CLOSE"
```

```
Data1.Recordset.Update
```

```
End If
```

```
If Val = "vn" Then
```

```
Label4.Caption = "CLOSE"
```

```
Data1.Recordset.Edit
```

```
Data1.Recordset.Fields("send1") = "CLOSE"
```

```
Data1.Recordset.Update
```

```
*****
```

```
Label8.Caption = "CLOSE"
```

```
Data1.Recordset.Edit
```

```
Data1.Recordset.Fields("send2") = "CLOSE"
```

```
Data1.Recordset.Update
```

```
End If
```

```
If Val = "zxc" Then
```

```
Label1.Caption = "OFF"
```

```
Shpsw1.BackColor = &HEOEEOE0 'แสดงรูปดวงไฟ1ดับ
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
Data1.Recordset.Edit
```

```
Data1.Recordset.Fields("ch1") = "OFF"
```

```
Data1.Recordset.Update
```

```
*****
```

```
Label2.Caption = "OFF"
```

```
Shpsw2.BackColor = &HE0E0E0 'แสดงรูปดวงไฟ2 คับ
```

```
Data1.Recordset.Edit
```

```
Data1.Recordset.Fields("ch2") = "OFF"
```

```
Data1.Recordset.Update
```

```
*****
```

```
Label3.Caption = "OFF"
```

```
Shpsw3.BackColor = &HE0E0E0 'แสดงรูปดวงไฟ3 คับ
```

```
Data1.Recordset.Edit
```

```
Data1.Recordset.Fields("ch3") = "OFF"
```

```
Data1.Recordset.Update
```

```
End If
```

```
If Val = "zxv" Then
```

```
Label1.Caption = "OFF"
```

```
Data1.Recordset.Edit
```

```
Data1.Recordset.Fields("ch1") = "OFF"
```

```
Data1.Recordset.Update
```

```
*****
```

```
Label2.Caption = "OFF"
```

```
Data1.Recordset.Edit
```

```
Data1.Recordset.Fields("ch2") = "OFF"
```

```
Data1.Recordset.Update
```

```
*****
```

```
Label4.Caption = "CLOSE"
```

```
Data1.Recordset.Edit
```

```
Data1.Recordset.Fields("send1") = "CLOSE"
```

```
Data1.Recordset.Update
```

```
End If
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

If Val = "zxn" Then
Shape1.BackColor = &HFF0000
Label1.Caption = "OFF"
Data1.Recordset.Edit
Data1.Recordset.Fields("ch1") = "OFF"
Data1.Recordset.Update
*****

Label2.Caption = "OFF"
Data1.Recordset.Edit
Data1.Recordset.Fields("ch2") = "OFF"
Data1.Recordset.Update
*****

Label8.Caption = "CLOSE"
Data1.Recordset.Edit
Data1.Recordset.Fields("send2") = "CLOSE"
Data1.Recordset.Update
End If
If Val = "zcy" Then
Label1.Caption = "OFF"
Data1.Recordset.Edit
Data1.Recordset.Fields("ch1") = "OFF"
Data1.Recordset.Update
*****

Label3.Caption = "OFF"
Data1.Recordset.Edit
Data1.Recordset.Fields("ch3") = "OFF"
Data1.Recordset.Update
*****

Label4.Caption = "CLOSE"
Data1.Recordset.Edit
Data1.Recordset.Fields("send1") = "CLOSE"
Data1.Recordset.Update

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

End If

If Val = "zcn" Then
Label1.Caption = "OFF"
Data1.Recordset.Edit
Data1.Recordset.Fields("ch1") = "OFF"
Data1.Recordset.Update
*****

Label3.Caption = "OFF"
Data1.Recordset.Edit
Data1.Recordset.Fields("ch3") = "OFF"
Data1.Recordset.Update
*****

Label8.Caption = "CLOSE"
Data1.Recordset.Edit
Data1.Recordset.Fields("send2") = "CLOSE"
Data1.Recordset.Update
End If

If Val = "zvn" Then
Label1.Caption = "OFF"
Data1.Recordset.Edit
Data1.Recordset.Fields("ch1") = "OFF"
Data1.Recordset.Update
*****

Label4.Caption = "CLOSE"
Data1.Recordset.Edit
Data1.Recordset.Fields("send1") = "CLOSE"
Data1.Recordset.Update
*****

Label8.Caption = "CLOSE"
Data1.Recordset.Edit
Data1.Recordset.Fields("send2") = "CLOSE"
Data1.Recordset.Update

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

End If

End Sub

\*\*\*\*\*โปรแกรมย่อยแสดงผลอุณหภูมิ\*\*\*\*\*

Private Sub Tmer\_tmp\_Timer()

MSComm1.RThreshold = "0"

MSComm1.InputLen = "0"

MSComm1.Output = "A" 'ส่งค่าเพื่อตรวจสอบ อุณหภูมิ

Dim Val As String

Val = MSComm1.Input

\*\*\*\*\*display 0

If Val = "j0k0" Or Val = "j1k0" Or Val = "j2k0" Or Val = "j3k0" Or Val = "j4k0" Or Val = "j5k0"

Or Val = "j6k0" Or Val = "j7k0" Or Val = "j8k0" Or Val = "j9k0" Then

Lb\_tmp1.Caption = "0"

Data3.Recordset.Edit

Data3.Recordset.Fields("N1") = "0"

Data3.Recordset.Update

End If

\*\*\*\*\*display 1

If Val = "j0k1" Or Val = "j1k1" Or Val = "j2k1" Or Val = "j3k1" Or Val = "j4k1" Or Val = "j5k1"

Or Val = "j6k1" Or Val = "j7k1" Or Val = "j8k1" Or Val = "j9k1" Then

Lb\_tmp1 = "1"

Data3.Recordset.Edit

Data3.Recordset.Fields("N1") = "1"

Data3.Recordset.Update

End If

\*\*\*\*\*display 2

If Val = "j0k2" Or Val = "j1k2" Or Val = "j2k2" Or Val = "j3k2" Or Val = "j4k2" Or Val = "j5k2"

Or Val = "j6k2" Or Val = "j7k2" Or Val = "j8k2" Or Val = "j9k2" Then

Lb\_tmp1 = "2"

Data3.Recordset.Edit

Data3.Recordset.Fields("N1") = "2"

Data3.Recordset.Update

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

End If

\*\*\*\*\*display 3

If Val = "j0k3" Or Val = "j1k3" Or Val = "j2k3" Or Val = "j3k3" Or Val = "j4k3" Or Val = "j5k3"  
Or Val = "j6k3" Or Val = "j7k3" Or Val = "j8k3" Or Val = "j9k3" Then

Lb\_tmp1 = "3"

Data3.Recordset.Edit

Data3.Recordset.Fields("N1") = "3"

Data3.Recordset.Update

End If

\*\*\*\*\*display 4

If Val = "j0k4" Or Val = "j1k4" Or Val = "j2k4" Or Val = "j3k4" Or Val = "j4k4" Or Val = "j5k4"  
Or Val = "j6k4" Or Val = "j7k4" Or Val = "j8k4" Or Val = "j9k4" Then

Lb\_tmp1 = "4"

Data3.Recordset.Edit

Data3.Recordset.Fields("N1") = "4"

Data3.Recordset.Update

End If

\*\*\*\*\*display 5

If Val = "j0k5" Or Val = "j1k5" Or Val = "j2k5" Or Val = "j3k5" Or Val = "j4k5" Or Val = "j5k5"  
Or Val = "j6k5" Or Val = "j7k5" Or Val = "j8k5" Or Val = "j9k5" Then

Lb\_tmp1 = "5"

Data3.Recordset.Edit

Data3.Recordset.Fields("N1") = "5"

Data3.Recordset.Update

End If

\*\*\*\*\*display 6

If Val = "j0k6" Or Val = "j1k6" Or Val = "j2k6" Or Val = "j3k6" Or Val = "j4k6" Or Val = "j5k6"  
Or Val = "j6k6" Or Val = "j7k6" Or Val = "j8k6" Or Val = "j9k6" Then

Lb\_tmp1 = "6"

Data3.Recordset.Edit

Data3.Recordset.Fields("N1") = "6"

Data3.Recordset.Update

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

End If

\*\*\*\*\*display 7

If Val = "j0k7" Or Val = "j1k7" Or Val = "j2k7" Or Val = "j3k7" Or Val = "j4k7" Or Val = "j5k7"  
Or Val = "j6k7" Or Val = "j7k7" Or Val = "j8k7" Or Val = "j9k7" Then

Lb\_tmp1 = "7"

Data3.Recordset.Edit

Data3.Recordset.Fields("N1") = "7"

Data3.Recordset.Update

End If

\*\*\*\*\*display 8

If Val = "j0k8" Or Val = "j1k8" Or Val = "j2k8" Or Val = "j3k8" Or Val = "j4k8" Or Val = "j5k8"  
Or Val = "j6k8" Or Val = "j7k8" Or Val = "j8k8" Or Val = "j9k8" Then

Lb\_tmp1 = "8"

Data3.Recordset.Edit

Data3.Recordset.Fields("N1") = "8"

Data3.Recordset.Update

End If

\*\*\*\*\*display 9

If Val = "j0k9" Or Val = "j1k9" Or Val = "j2k9" Or Val = "j3k9" Or Val = "j4k9" Or Val = "j5k9"  
Or Val = "j6k9" Or Val = "j7k9" Or Val = "j8k9" Or Val = "j9k9" Then

Lb\_tmp1 = "9"

Data3.Recordset.Edit

Data3.Recordset.Fields("N1") = "9"

Data3.Recordset.Update

End If

\*\*\*\*\*0,0\*\*\*\*\*

If Val = "j0k0" Or Val = "j0k1" Or Val = "j0k2" Or Val = "j0k3" Or Val = "j0k4" Or Val = "j0k5"  
Or Val = "j0k6" Or Val = "j0k7" Or Val = "j0k8" Or Val = "j0k9" Then

Lb\_tmp2 = "0"

Data3.Recordset.Edit

Data3.Recordset.Fields("N2") = "0"

Data3.Recordset.Update

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

End If

\*\*\*\*\*0.1\*\*\*\*\*

If Val = "j1k0" Or Val = "j1k1" Or Val = "j1k2" Or Val = "j1k3" Or Val = "j1k4" Or Val = "j1k5"  
Or Val = "j1k6" Or Val = "j1k7" Or Val = "j1k8" Or Val = "j1k9" Then

Lb\_tmp2 = "1"

Data3.Recordset.Edit

Data3.Recordset.Fields("N2") = "1"

Data3.Recordset.Update

End If

\*\*\*\*\*0.2\*\*\*\*\*

If Val = "j2k0" Or Val = "j2k1" Or Val = "j2k2" Or Val = "j2k3" Or Val = "j2k4" Or Val = "j2k5"  
Or Val = "j2k6" Or Val = "j2k7" Or Val = "j2k8" Or Val = "j2k9" Then

Lb\_tmp2 = "2"

Data3.Recordset.Edit

Data3.Recordset.Fields("N2") = "2"

Data3.Recordset.Update

End If

\*\*\*\*\*0.3\*\*\*\*\*

If Val = "j3k0" Or Val = "j3k1" Or Val = "j3k2" Or Val = "j3k3" Or Val = "j3k4" Or Val = "j3k5"  
Or Val = "j3k6" Or Val = "j3k7" Or Val = "j3k8" Or Val = "j3k9" Then

Lb\_tmp2 = "3"

Data3.Recordset.Edit

Data3.Recordset.Fields("N2") = "3"

Data3.Recordset.Update

End If

\*\*\*\*\*

If Val = "j4k0" Or Val = "j4k1" Or Val = "j4k2" Or Val = "j4k3" Or Val = "j4k4" Or Val = "j4k5"  
Or Val = "j4k6" Or Val = "j4k7" Or Val = "j4k8" Or Val = "j4k9" Then

Lb\_tmp2 = "4"

Data3.Recordset.Edit

Data3.Recordset.Fields("N2") = "4"

Data3.Recordset.Update

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

End If

\*\*\*\*\*

If Val = "j5k0" Or Val = "j5k1" Or Val = "j5k2" Or Val = "j5k3" Or Val = "j5k4" Or Val = "j5k5"  
Or Val = "j5k6" Or Val = "j5k7" Or Val = "j5k8" Or Val = "j5k9" Then

Lb\_tmp2 = "5"

Data3.Recordset.Edit

Data3.Recordset.Fields("N2") = "5"

Data3.Recordset.Update

End If

\*\*\*\*\*

If Val = "j6k0" Or Val = "j6k1" Or Val = "j6k2" Or Val = "j6k3" Or Val = "j6k4" Or Val = "j6k5"  
Or Val = "j6k6" Or Val = "j6k7" Or Val = "j6k8" Or Val = "j6k9" Then

Lb\_tmp2 = "6"

Data3.Recordset.Edit

Data3.Recordset.Fields("N2") = "6"

Data3.Recordset.Update

End If

\*\*\*\*\*

If Val = "j7k0" Or Val = "j7k1" Or Val = "j7k2" Or Val = "j7k3" Or Val = "j7k4" Or Val = "j7k5"  
Or Val = "j7k6" Or Val = "j7k7" Or Val = "j7k8" Or Val = "j7k9" Then

Lb\_tmp2 = "7"

Data3.Recordset.Edit

Data3.Recordset.Fields("N2") = "7"

Data3.Recordset.Update

End If

\*\*\*\*\*

If Val = "j8k0" Or Val = "j8k1" Or Val = "j8k2" Or Val = "j8k3" Or Val = "j8k4" Or Val = "j8k5"  
Or Val = "j8k6" Or Val = "j8k7" Or Val = "j8k8" Or Val = "j8k9" Then

Lb\_tmp2 = "8"

Data3.Recordset.Edit

Data3.Recordset.Fields("N2") = "8"

Data3.Recordset.Update

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

End If

\*\*\*\*\*

If Val = "j9k0" Or Val = "j9k1" Or Val = "j9k2" Or Val = "j9k3" Or Val = "j9k4" Or Val = "j9k5"  
Or Val = "j9k6" Or Val = "j9k7" Or Val = "j9k8" Or Val = "j9k9" Then

Lb\_tmp2 = "9"

Data3.Recordset.Edit

Data3.Recordset.Fields("N2") = "9"

Data3.Recordset.Update

End If

End Sub

Private Sub TimerSrvcon\_Timer()

MSComm1.RThreshold = "0"

MSComm1.InputLen = "0"

Dim val3, delayR, delayL, delay As Variant

If Data2.Recordset.Fields("ch1") = "ON" And Data2.Recordset.Fields("ch2") = "OFF" And  
Data2.Recordset.Fields("ch3") = "OFF" Then

MSComm1.Output = "B"

MSComm1.Output = "x"

MSComm1.Output = "c"

ElseIf Data2.Recordset.Fields("ch1") = "OFF" And Data2.Recordset.Fields("ch2") = "ON" And  
Data2.Recordset.Fields("ch3") = "OFF" Then

MSComm1.Output = "C"

MSComm1.Output = "z"

MSComm1.Output = "c"

ElseIf Data2.Recordset.Fields("ch1") = "OFF" And Data2.Recordset.Fields("ch2") = "OFF" And  
Data2.Recordset.Fields("ch3") = "ON" Then

MSComm1.Output = "D"

MSComm1.Output = "z"

MSComm1.Output = "x"

ElseIf Data2.Recordset.Fields("ch1") = "ON" And Data2.Recordset.Fields("ch2") = "ON" And  
Data2.Recordset.Fields("ch3") = "OFF" Then

MSComm1.Output = "B"

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

MSComm1.Output = "C"
MSComm1.Output = "c"
ElseIf Data2.Recordset.Fields("ch1") = "OFF" And Data2.Recordset.Fields("ch2") = "ON" And
Data2.Recordset.Fields("ch3") = "ON" Then
MSComm1.Output = "C"
MSComm1.Output = "D"
MSComm1.Output = "z"
ElseIf Data2.Recordset.Fields("ch1") = "ON" And Data2.Recordset.Fields("ch2") = "OFF" And
Data2.Recordset.Fields("ch3") = "ON" Then
MSComm1.Output = "B"
MSComm1.Output = "D"
MSComm1.Output = "x"
ElseIf Data2.Recordset.Fields("ch1") = "ON" And Data2.Recordset.Fields("ch2") = "ON" And
Data2.Recordset.Fields("ch3") = "ON" Then
MSComm1.Output = "B"
MSComm1.Output = "C"
MSComm1.Output = "D"
ElseIf Data2.Recordset.Fields("ch1") = "OFF" And Data2.Recordset.Fields("ch2") = "OFF" And
Data2.Recordset.Fields("ch3") = "OFF" Then
MSComm1.Output = "z"
MSComm1.Output = "x"
MSComm1.Output = "c"
Else
End If
*****

If Data2.Recordset.Fields("sil") = "ON" Then
MSComm1.Output = "F"
Frmsir.Visible = True
Timer3.Enabled = True
Option1.Value = True
Option2.Value = False

```

**Data2.Recordset.Edit**

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
Data2.Recordset.Fields("sil") = "DEFAULT"
```

```
Data2.Recordset.Update
```

```
End If
```

```
If Data2.Recordset.Fields("sil") = "OFF" Then
```

```
MSComm1.Output = "b"
```

```
Frmsir.Visible = False
```

```
Timer3.Enabled = False
```

```
Option1.Value = False
```

```
Option2.Value = True
```

```
Data2.Recordset.Edit
```

```
Data2.Recordset.Fields("sil") = "DEFAULT"
```

```
Data2.Recordset.Update
```

```
End If
```

```
*****
```

```
If Data5.Recordset.Fields("R") = "YES" Then
```

```
Do Until delayR = 15
```

```
MSComm1.Output = "d"
```

```
delayR = delayR + 1
```

```
Loop
```

```
Data5.Recordset.Edit
```

```
Data5.Recordset.Fields("R") = "NO"
```

```
Data5.Recordset.Update
```

```
End If
```

```
If Data5.Recordset.Fields("L") = "YES" Then
```

```
Do Until delayL = 15
```

```
MSComm1.Output = "U" 'turn left
```

```
delayL = delayL + 1
```

```
Loop
```

```
Data5.Recordset.Edit
```

```
Data5.Recordset.Fields("L") = "NO"
```

```
Data5.Recordset.Update
```

```
End If
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

If Data5.Recordset.Fields("U") = "YES" Then
    Do Until delay = 15
        MSComm1.Output = "L" 'turn UP
        delay = delay + 1
    Loop
    Data5.Recordset.Edit
    Data5.Recordset.Fields("U") = "NO"
    Data5.Recordset.Update
End If

If Data5.Recordset.Fields("D") = "YES" Then
    Do Until delay = 15
        MSComm1.Output = "R" 'turn Down
        delay = delay + 1
    Loop
    Data5.Recordset.Edit
    Data5.Recordset.Fields("D") = "NO"
    Data5.Recordset.Update
End If

*****
If Frmsir.Visible = True And Timer3.Enabled = True Then
    Data1.Recordset.Edit
    Data1.Recordset.Fields("sil") = "ON"
    Data1.Recordset.Update
    MSComm1.Output = "F"
End If

If Frmsir.Visible = False And Timer3.Enabled = False Then
    Data1.Recordset.Edit
    Data1.Recordset.Fields("sil") = "OFF"
    Data1.Recordset.Update
    MSComm1.Output = "b"
End If

End Sub

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## ภาคผนวก ข. โปรแกรมสำหรับ MCS-51

```

;*****
;****          HOME NETWORK          ****
;*****

CR    EQU    0DH
LF    EQU    0AH

step  equ    34h ;address stepmotor
step2 equ    35h
stroe equ    36h

SW_1  BIT    P0.0
SW_2  BIT    P0.1
SW_3  BIT    P0.2
SW_4  BIT    P0.3
SW_5  BIT    P0.4
SW_6  BIT    P0.5
SW1   BIT    00H
SW2   BIT    01H
SW3   BIT    02H
SW4   BIT    03H
SW5   BIT    04H
SW6   BIT    05H
;*****temp*****
ONEWIRE BIT    P3.5
FLAG EQU    02FH
BUSY BIT    FLAG.0
LCD_DATA EQU 031H
ONEWIRE_DATA EQU 032H
TEMP EQU    033H
ORG    0000H
LJMP   NEW_START

;*****INT0*****
ORG    0003H
LJMP   INT_0

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
*****MAIN PROGRAME*****
```

```

ORG 0100H

NEW_START: MOV P1,#0FFH

MOV 20H,#0FFH

MOV P3,#0FFH

MOV P0,#0FFH

mov step,10000000b

MOV STEP2,00000001B

MOV SP,#59H

START: ACALLINIT

MOV DPTR,#SHOW

ACALLDISPLAY

MAIN: ACALLRECEIVE

SJMP MAIN

;*****FIG*****

INIT: MOV IE,#10000001B

MOV TMOD,#00100000B

MOV TL1,#0FDH

MOV TH1,#0FDH

MOV SCON,#01010000B

CLR IT0

SETB TRI

RET

;*****MCS TO RS232*****

SEND: MOV SBUF,A

JNB TI,$

CLR TI

RET

;*****RS232 TO MCS*****

RECEIVE: JNB RI,$

CLR RI

MOV A,SBUF

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
MOV R1,A
```

```
***** CHECK REF *****
```

```
CHECK_R: CJNE R1,#39H,ONI
```

```
***** Check Refaced *****
```

```
CHECK1: JB SW_1,REF1
```

```
JNB SW_1,REF1_1
```

```
CHECK2: JB SW_2,REF2
```

```
JNB SW_2,REF2_1
```

```
CHECK3: JB SW_3,REF3
```

```
JNB SW_3,REF3_1
```

```
CHECK4: JB SW_4,REF4
```

```
JNB SW_4,REF4_1
```

```
CHECK5: JB SW_5,REF5
```

```
JNB SW_5,REF5_1
```

```
CHECK6: JB SW_6,REF6
```

```
JNB SW_6,REF6_1
```

```
REF1: MOV A,#7AH
```

```
ACALLSEND
```

```
SJMP CHECK2
```

```
REF1_1: MOV A,#66H;B
```

```
ACALLSEND
```

```
SJMP CHECK2
```

```
REF2: MOV A,#78H
```

```
ACALLSEND
```

```
SJMP CHECK3
```

```
REF2_1: MOV A,#67H;C
```

```
ACALLSEND
```

```
SJMP CHECK3
```

```
REF3: MOV A,#63H
```

```
ACALLSEND
```

```
SJMP CHECK4
```

```
REF3_1: MOV A,#68H;D
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

ACALLSEND
SJMP CHECK4
REF4:    MOV  A,#76H
ACALLSEND
SJMP CHECK5
REF4_1:  MOV  A,#69 ;E
ACALLSEND
SJMP CHECK5
REF5:    MOV  A,#62H
ACALLSEND
SJMP CHECK6
REF5_1:  MOV  A,#70 ;F
ACALLSEND
SJMP CHECK6
REF6:    MOV  A,#6EH
ACALLSEND
LJMP RECEIVE
REF6_1:  MOV  A,#71 ;G
ACALLSEND
LJMP RECEIVE
ONI:     CJNE R1,#66,OFF1
CLR      P1.0
LJMP RECEIVE
OFF1:    CJNE R1,#7AH,ON2
SETB    P1.0
LJMP RECEIVE

***** PORT2*****
ON2:     CJNE R1,#67,OFF2
CLR      P1.1
LJMP RECEIVE
OFF2:    CJNE R1,#78H,ON3
SETB    P1.1

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## LJMP RECEIVE

```
*****PORT3*****
```

```
ON3: CJNE R1,#68,OFF3
```

```
CLR P1.2
```

```
LJMP RECEIVE
```

```
OFF3: CJNE R1,#63H,ON5
```

```
SETB P1.2
```

```
LJMP RECEIVE
```

```
*****PORT5*****
```

```
ON5: CJNE R1,#70,OFF5
```

```
CLR P1.4
```

```
;mov a,#70 ;F
```

```
;ACALL SEND
```

```
LJMP RECEIVE
```

```
OFF5: CJNE R1,#62H,MOTOR_L
```

```
SETB P1.4
```

```
;mov a,#62h
```

```
;ACALL SEND
```

```
LJMP RECEIVE
```

```
*****MOTOR_L*****
```

```
MOTOR_L: CJNE R1,#76,MOTOR_R ;L
```

```
mov a,step
```

```
swap a
```

```
clr c
```

```
rlc a
```

```
jnc next_left
```

```
mov a,#00010000b
```

```
next_left: swap a
```

```
mov step,a
```

```
mov p2,step
```

```
acall delay_100ms
```

```
ljmp receive
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

;*****MOTOR_R*****
MOTOR_R:  CJNE  R1,#82,MOTOR_UP ;R
          mov  a,step
          clr  c
          rrc  a
          jnc  next_right
          mov  a,#00001000b
next_right:  mov  step,a
          mov  p2,step
          acall delay_100ms
          ljmp receive
;*****MOTOR_UP*****
MOTOR_UP:  CJNE  R1,#85,MOTOR_DOWN ;value U_test
          mov  a,step2
          swap a
          clr  c
          rrc  a
          jnc  next_UP
          mov  a,#00001000b
next_UP:   swap a
          mov  step2,a
          mov  p2,step2
          acall delay_100ms
          ljmp receive
;*****MOTOR_DOWN*****
MOTOR_DOWN:  cjne  R1,#100,LOOP_TEMP ;value d_test
          mov  a,step2
          clr  c
          rlc  a ;*****test
          jnc  next_DOWN
          mov  a,#00010000b

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

next_DOWN:    mov    step2,a
              mov    p2,step2
              acall  delay_100ms
              ljmp   receive

```

```

;*****temp*****

```

```

LOOP_TEMP:    cjne   r1,#065,RES ;A ; value t_test
              ACALL  DS1820_RST ;LOOP_temp
              ACALL  DS1820 PRES
              MOV    ONEWIRE_DATA,#0CCH
              ACALL  DS1820_WR
              MOV    ONEWIRE_DATA,#044H
              ACALL  DS1820_WR
              SETB   BUSY
PRES_CHK_LOOP: ACALL  DS1820_RST
              ACALL  DS1820 PRES
              JB    BUSY,PRES_CHK_LOOP
              NOP
              NOP
              NOP
              NOP
              ACALL  DS1820_RST
              ACALL  DS1820 PRES
              MOV    ONEWIRE_DATA,#0CCH
              ACALL  DS1820_WR
              MOV    ONEWIRE_DATA,#0BEH
              ACALL  DS1820_WR
              ACALL  DS1820_RD
              MOV    TEMP,ONEWIRE_DATA
              ACALL  DS1820_RST
              ACALL  DS1820 PRES
              MOV    A,TEMP
              CLR   C

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

RRC  A
MOV  LCD_DATA,A
ACALL HEX2LCD
ljmp      receive

```

```

;*****RESET*****

```

```

RES:      CJNE  R1,#0DH,RECEIVE1
          LJMP  NEW_START
RECEIVE1: LJMP  RECEIVE
          RET

```

```

;*****HEX_SEND*****

```

```

HEX2LCD: PUSH  ACC
          MOV  A,LCD_DATA
          MOV  B,#100
          DIV  AB
          ADD  A,#030H
          CJNE A,#030H,HEX2_LCD_NX
          MOV  A,#' '
HEX2_LCD_NX:
          MOV  A,B
          MOV  B,#10
          DIV  AB
          ADD  A,#030H
          mov  stroc,a
          mov  a,#106      ;j_code temp
          acall  send
          mov  a,stroc
          acall  send
          MOV  A,B
          ADD  A,#030H
          mov  stroc,a
          mov  a,#107      ;k_code temp
          acall  send

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

mov    a,stroe
acall  send
POP    ACC
RET

;*****DS1820_RD AND WR*****
DS1820_RD:    MOV    R4,#8
              CLR    A
DS1820_RD_LOOP:  CLR    ONEWIRE
              NOP
              NOP
              SETB   ONEWIRE
              NOP
              NOP
              NOP
              MOV    C,ONEWIRE
              ACALL  ONEWIRE_DELAY
              RRC    A
              DJNZ   R4,DS1820_RD_LOOP
              MOV    ONEWIRE_DATA,A
              RET
DS1820_WR:    MOV    R4,#8
              MOV    A,ONEWIRE_DATA
DS1820_WR_LOOP:  RRC    A
              JNC    DS1820_WR_L
              CLR    ONEWIRE
              NOP
              NOP
              NOP
              NOP
              SETB   ONEWIRE
              ACALL  ONEWIRE_DELAY

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

AJMP DS1820_WR_NX
DS1820_WR_L: CLR ONEWIRE
              ACALL ONEWIRE_DELAY
              SETB ONEWIRE
              NOP
              NOP
              NOP
              NOP
DS1820_WR_NX: DJNZ R4,DS1820_WR_LOOP
              RET
DS1820_RST:  CLR ONEWIRE
              ACALL DELAY_1ms
              SETB ONEWIRE
              MOV R4,#8
              DJNZ R4,$
              RET
DS1820_PRES: MOV R4,#8
DS1820_PRES_1: MOV R3,#0
DS1820_PRES_2: JNB ONEWIRE,DS1820_PRES_3
              DJNZ R3,DS1820_PRES_2
              DJNZ R4,DS1820_PRES_1
              RET
DS1820_PRES_3: JNB ONEWIRE,$
              MOV R4,#8
              DJNZ R4,$
              CLR BUSY
              RET

```

\*\*\*\*\*DELAY\*\*\*\*\*

```

DELAY:      MOV R4,#05H
DELAY1:     MOV R5,#0F0H
DELAY2:     MOV R6,#0F0H
           DJNZ R6,$

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
DJNZ R5,DELAY2
```

```
DJNZ R4,DELAY1
```

```
RET
```

```
;*****DELAY_TEMP*****
```

```
ONEWIRE_DELAY: MOV 6,#012H
```

```
ONEWIRE_DELAY_1:NOP
```

```
NOP
```

```
DJNZ R6,ONEWIRE_DELAY_1
```

```
RET
```

```
DELAY_50us: MOV 6,#00CH
```

```
DELAY_50us_1: NOP
```

```
NOP
```

```
DJNZ R6,DELAY_50us_1
```

```
RET
```

```
DELAY_100us: MOV 6,#017H
```

```
DELAY_100us_1: NOP
```

```
NOP
```

```
DJNZ R6,DELAY_100us_1
```

```
RET
```

```
DELAY_1ms: MOV 6,#0E6H
```

```
DELAY_1ms_1: NOP
```

```
NOP
```

```
DJNZ R6,DELAY_1ms_1
```

```
RET
```

```
DELAY_10ms: MOV 7,#010
```

```
DELAY_10ms_1: MOV 6,#0E6H
```

```
DELAY_10ms_2: NOP
```

```
NOP
```

```
DJNZ R6,DELAY_10ms_2
```

```
DJNZ R7,DELAY_10ms_1
```

```
RET
```

```
DELAY_100ms: MOV 7,#100
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
DELAY_100ms_1: MOV 6,#0E6H
```

```
DELAY_100ms_2: NOP
```

```
    NOP
```

```
    DJNZ R6,DELAY_100ms_2
```

```
    DJNZ R7,DELAY_100ms_1
```

```
    RET
```

```
DELAY_1s: MOV 5,#100
```

```
DELAY_1s_1: ACALL DELAY_10ms
```

```
    DJNZ R5,DELAY_1s_1
```

```
    RET
```

```
*****DISPLAY*****
```

```
DISPLAY: MOV A,#00H
```

```
    MOVC A,@A+DPTR
```

```
    ACALL SEND
```

```
    JZ END_SHOW
```

```
    INC DPTR
```

```
    SJMP DISPLAY
```

```
END_SHOW: RET
```

```
*****SHOW*****
```

```
SHOW: DB CR,LF,"HOME NETWORK SWICTH",CR,LF
```

```
DB "Designed By Thongrob Soontron",CR,LF
```

```
DB "Electronics Engineering 3R/2"
```

```
DB 00H
```

```
END
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## ภาคผนวก ก. โปรแกรม ASP

## MAIN1.ASP

```

<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">
<script language="vbscript" runat=server>
</script>
<html>
<head>
<title> Main page titled </title>
<meta http-equiv="Content-Type" content="text/html; charset=iso-8859-1">
<script language="JavaScript" type="text/JavaScript">
<!--function MM_reloadPage(init) { //reloads the window if Nav4 resized
  if (init==true) with (navigator) {if ((appName=="Netscape")&&(parseInt(appVersion)==4))
  { document.MM_pgW=innerWidth; document.MM_pgH=innerHeight;
onresize=MM_reloadPage; }}
else if (innerWidth!=document.MM_pgW || innerHeight!=document.MM_pgH) location.reload();
}
MM_reloadPage(true);!-->
</script>
</head>
<body bgcolor="#FFFFFF" background="1.gif" alink="#FF0099">
<div id="Layer1" style="position:absolute; left:526px; top:243px; width:287px; height:119px; z-
index:1"> <form name="form1" method="post" action="con_ch1.asp">
  <table width="104%" height="78" border="1" bordercolor="#0099FF">
<tr background="Myweb/Thai [ Article ] _Com All Article U Looking for __files/BG2.GIF">
<td height="28" colspan="3" background="b.gif" bgcolor="#99CCFF"><div align="center">
<strong><font size="2" face="Courier New, Courier, mono">
  <select name="select">
    <option>ON</option>
    <option>OFF</option>
  </select>
  <font color="#9999CC"> SW1
  <select name="select2">

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

<option>ON</option>
<option>OFF</option>
</select>
SW2
<select name="select3">
<option>ON</option>
<option>OFF</option>
</select>
SW3</font></font></strong> </div></td> </tr>
<tr background="Myweb/Thai [ Article ] _ Com All Article U Looking
for __files/BG2.GIF">
<td width="30%" height="28" colspan="3" background="b.gif" bgcolor="#99CCFF">
<div align="center"><font color="#33CC99" face="Courier New"><strong>Control it !
</strong></font>
<input type="submit" name="Submit2" value="Submit">
</div>
<div align="center"><strong></strong></div>
<div align="center"></div>
<div align="center"></div>
<div align="center"></div></td>
</tr>
</table>
<p align="center">&nbsp;</p>
</form>
</div>
<form method="post" action="main1.asp">
<%dim objtemp,ref
set objtemp=server.CreateObject("ADODB.recordset")
objtemp.open"temp", "dsn=nonguy",1,3%>

<table width="107%" height="74" border="0" cellspacing="0"> <tr>

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

<td height="51" background="b.gif"><div align="left">
  <font color="#CCCCCC" size="+2"><strong>AN INTERNET HOME MONITORIES
SYSTEM</strong></font>
  <hr>
</div></td>
</tr>
<tr>
  <td height="21" background="Myweb/bg/hunsa_top_menu08.gif"><p><strong><font size="1">
  <font color="#CCCCCC"><a href="http://www.kmitl.ac.th">HOME</a></font></font>
  <font color="#666666" size="1"><a href="file:///Aboutme.asp"> ABOUT ME</a>
  | <a href="http:%5C%5Cwww.kmitl.ac.th">KMITL </a>| <a
href="http:%5C%5Cwww.kmitl.ac.th/electronicclub">ELECTRONICS
CLUB </a>| ROBOT CLUB | ERC<font color="#0000CC">| </font><font color="#3300FF"
size="6"></font></font></strong></p>
</td>
</tr>
</table>
<table width="107%" height="481" border="0" cellspacing="0"
background="Myweb/Thai%20%5B%20Article%20%5D%20_%20Com%20All%20Article%20
U%20Looking%20for__files/BG2.GIF">
  <tr>
    <td height="21" colspan="5" background="1.gif"><div align="left"></div></td>
  </tr>
  <tr>
    <td colspan="2" background="file:///main_ny_bg.gif"><a
href="http:%5C%5Cwww.kmitl.ac.th"></a></td>
    <td width="28%" background="file:///main_ny_bg.gif"> <div align="left"><font
color="#000000" size="5"><strong><font color="#666666" size="1"><font color="#3300FF"
size="6"><%=response.Write("Time: ")&time%></font></font></strong></font></div></td>
  </tr>

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



```

<td>&nbsp;</td>
<td><a href="file:///motD.asp"></a></td>
<td>&nbsp;</td>
</tr>
</table></td>
</tr>
</table></td>
<td background="1.gif"><blockquote>
<div align="left"><font size="5"><strong></strong></font></div>
</blockquote></td>
<td colspan="2" rowspan="6" background="1.gif">&nbsp;</td> </tr>
<tr> <td background="1.gif"><font size="5" face="Courier New"><strong><a href="file:///Com.asp" name="stat">
PropretiesStatus</a></strong></font></td> </tr>
<tr> <td background="1.gif"><strong><font size="5" face="Courier New"><a href="file:///door1.asp">Door
one Status</a></font></strong></td> </tr>
<tr> <td background="1.gif"><strong><font size="5" face="Courier New"><a href="file:///door2.asp">Door
two Status</a></font></strong></td> </tr>
<tr> <td background="1.gif"><strong><font size="5" face="Courier New"><a
href="file:///silen.asp">Silent
status</a></font></strong></td> </tr>
<tr> <td background="1.gif"><strong><font size="5" face="Courier New"><a href="file:///prop.asp">Properties</a></font></strong></td> </tr>
<tr> <td height="1" colspan="2" background="file:///main_ny_bg.gif"><div align="center">
<font size="4"><strong><font size="6">&nbsp;</font></strong><font size="6"><strong><strong><font color="#CC0000"
size="6">About Project </font></strong></strong></font></font></div></td>
<td rowspan="5" background="1.gif"><blockquote> <p><strong></strong></p>

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

</blockquote>
<blockquote> <p><strong><font size="3"> </font></strong></p>
</blockquote>
<blockquote> <p><strong></strong></p>
</blockquote>
<p>&nbsp;</p></td> <td colspan="2" rowspan="7" background="1.gif"><div align="left">
<font color="#666633" size="3" face="Courier New"> <strong><font color="#FFFF33">This </font></strong>project
presents an applied software to control electrical
equipment.Soreceive from sensor to processing data and data send to
computer display by internet system. The programs is divided into 3
parts, homepage program using HTML incorporating with ASP ( Active Server
Pages ) ,electrical equipment controlled program at server and using
Window 2000 server adn HS ( Internet Information Server ) with Visual
Basic 6 , microcontroller controlled programs are developed for suing
in this project.The hardware part is electrical equipment controlled
circuit using MCS-51 to develop.</font></div>
<p>&nbsp;</p>
<p>&nbsp;</p>
<p>&nbsp;</p></td> </tr>
<tr> <td height="2" colspan="2" bgcolor="#99CCFF"><font color="#0000CC" size="3"
face="Courier New">
Flie Project Donw Load
</font></td> </tr>
<tr> <td height="3" colspan="2" background="file:///1.gif"><font color="#0000CC"
face="Courier New"> Project Document
</font></td> </tr>
<tr> <td height="3" colspan="2" bgcolor="#99CCFF"> <font color="#0000CC" face="Courier New">Project Picture</font>
</td> </tr>
<tr> <td height="24" colspan="2" background="file:///1.gif"><table width="101%" height="52"
border="1" cellspacing="1" bordercolor="#0099FF" background="file:///b.gif">
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

```

<tr> <td colspan="3" background="main_ny_bg.gif"><div align="center"><strong><font
color="#CC0000">Temp</font>
</strong></div></td> </tr>

<tr background="file:///1.gif">
  <td><div align="center"><font color="#CC0000" size="6"><strong><%=objtemp
("N2")%></strong></font></div></td>
  <td><div align="center"><font color="#CC0000" size="6"><strong><%=objtemp
("N1")%></strong></font></div></td>
  <td><div align="center"><font color="#3300FF" size="6"><strong>C</strong></font>
</div></td> </tr> </table>

<blockquote>
  <p align="center"><strong><font color="#CC0000" size="4"> Click To Ref</font>
  
  <input type="submit" name="Submit" value="Ref">
</strong></p> </blockquote></td> </tr> <tr>
<td height="2" colspan="2" background="file:///1.gif">&nbsp;</td>
  <td background="1.gif">&nbsp;</td> </tr>
<tr><td colspan="2" background="file:///1.gif"><div align="left"> <font size="-2">&nbsp;</font>
  <font color="#3399FF"> <a
href="http://thaimail.com">nonguy91@thaimail.com</a></font>
</div></td> <td background="1.gif">&nbsp;</td> </tr>
</table>
</form>
</body>
</html>

```

## CON\_CH.ASP

```

<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">
<html>
<head>
<title>Status Show</title>

```

```

<meta http-equiv="Content-Type" content="text/html; charset=iso-8859-1">

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

</head>
<bodybackground="Myweb/Thai%20%5B%20Article%20%5D%20_%20Com%20All%20Artic
e%20U%20Looking%20for__files/BG2.GIF">
<table width="64%" height="74" border="0" cellspacing="0"> <tr>
  <td height="51" background="1.gif"><div align="left">
  <font color="#CCCCCC" size="+2"><strong>AN INTERNET HOME MONITORIES
SYSTEM</strong></font> <hr>
  </div></td> </tr>
<tr> <td height="21" background="Myweb/bg/hunsa_top_menu08.gif"><p><strong><font size="1">
  <font color="#CCCCCC"><a href="file:///main1.asp">HOME</a></font></font> <font
color="#666666" size="1">|
  <a href="file:///Aboutme.asp">ABOUT ME</a>|<a href="http://www.html.ac.th">
  KMITL </a>| ELECTRONICS CLUB | ROBOT CLUB | ERC<font color="#0000CC">
| </font></font></strong></p></td>
</tr></table>
<% dim swsat,objprop,objsrv,objsat
dim swcon2,swcon3,objcon,i
swcon=request.form("select")
swcon2=request.form("select2")
swcon3=request.form("select3")
set objprop=server.CreateObject("ADODB.recordset")
objprop.open"prop","dsn=nonguy",1,3
set objsrv=server.CreateObject("ADODB.recordset")
objsrv.open"srvdat","dsn=nonguy",1,3
set objsat=server.CreateObject("ADODB.recordset")
objsat.open"ch_nal","dsn=nonguy",1,3
objsat("ch1")=swcon
objsat("ch2")=swcon2
objsat("ch3")=swcon3

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

objsat.update
for i=0 to 9000000
i=i+1
next
if objsat("ch1")<> objsrv("ch1") or objsat("ch2")<> objsrv("ch2") or objsat("ch3")<> objsrv
("ch3") then
response.Redirect("notres.asp")
else
end if%>
<table width="64%" border="1" cellpadding="1"> <tr background="file:///BG4.GIF">
<td colspan="3"><div align="center"><strong><font color="#CC0000">  Response Display</font></strong></div></td> </tr>
<tr bgcolor="#FF9900"> <td width="54%"><font color="#0000CC" size="5"><strong><%=
objprop("p1")%></strong></font></td><td colspan="2"><font color="#0000CC" size="5">
<strong><%= objsat("ch1")%></strong>&nbsp;</font></td> </tr> <tr>
<td><font color="#0000CC" size="5"><strong><%= objprop("p2")%></strong></font></td>
<td colspan="2"><font color="#0000CC" size="5"><strong><%= objsat("ch2")%>
</strong>&nbsp;</font></td> </tr> <tr bgcolor="#FF9900"
background="Myweb/Thai%20%5B%20Article%20%5D%20 %20Com%20All%20Article%20
U%20Looking%20for ___files/BG4.GIF">
<td background="file:///BG4.GIF"><font color="#0000CC" size="5"><strong><%= objprop
("p3")%></strong></font></td>
<td colspan="2" background="file:///BG4.GIF"><font color="#0000CC" size="5"><strong>
<%= objsat("ch3")%></strong>&nbsp;</font></td> </tr> <tr>
<td colspan="3"><a href="file:///main1.asp">&lt;---Return to MainPage ! </a> </td> </tr></table></body></html>

```

## DOOR.ASP

```

<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">
<html>
<head>
<title>Status Door</title>

```

```

<meta http-equiv="Content-Type" content="text/html; charset=iso-8859-1">

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

</head>
<body background="main_ny_bg.gif">
<%dim objprop,objsrv
set objprop=server.CreateObject("ADODB.recordset")
objprop.open"prop","dsn=nonguy",1,3
set objsrv=server.CreateObject("ADODB.recordset")
objsrv.open"srvdat","dsn=nonguy",1,3 %>
<table width="75%" height="176" border="1" cellspacing="0" bordercolor="#0000CC"> <tr>
  <td colspan="2"><div align="center"><font color="#990000" size="6" face="Courier New,
Courier, mono"><strong>NOW
! Status </strong></font></div></td> </tr> <tr>
  <td width="44%"><font color="#0000FF" size="6"><strong><%=objprop("p5")%></strong>
</font></td>
  <td><strong><font color="#990000" size="5" face="Courier New"><%=objsrv("send1")%>
</font></strong></td> </tr> <tr bgcolor="#99CCCC">
  <td colspan="2"><div align="center"><font color="#990000" size="6" face="AngsanaUPC">
Last Time is Open Door !</font></div></td> </tr> <tr>
  <td><strong><font color="#9900FF" size="5" face="Courier New"><%=objsrv("time")%>
</font></strong></td>
  <td><font color="#33CCCC" size="5" face="Courier New"><strong><%=objsrv("date")%>
</strong></font></td> </tr> <tr>
  <td colspan="2"><a href="file:///main1.asp">Click to back---&gt;</a></td>
</tr></table></body>
</html>

```

## PROP.ASP

```

<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">
<html>
<head>
<title>Properties</title>
<meta http-equiv="Content-Type" content="text/html; charset=iso-8859-1">
</head>

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

<body background="file:///main_ny_bg.gif">
<% dim objprop
set objprop=server.CreateObject("ADODB.recordset")
objprop.open"prop", "dsn=nonguy", 1, 3 %>
<table width="64%" height="74" border="0" cellspacing="0"> <tr>
  <td height="51" background="1.gif"><div align="left"> <font color="#CCCCCC" size="+2"><strong>AN INTERNET
HOME MONITORIES SYSTEM</strong></font> <hr> </div></td> </tr> <tr>
  <td height="21" background="Myweb/bg/hunsa_top_mcnu08.gif"><p><strong><font size="1">
<font color="#CCCCCC"><a href="file:///main1.asp">HOME</a></font></font> <font
color="#666666" size="1">|
<a href="file:///Aboutme.asp">ABOUT ME</a> | <a href="http://www.kmitl.ac.th">
KMITL</a>
| ELECTRONICS CLUB | ROBOT CLUB | ERC<font color="#0000CC"> | </font></font></strong></p></td> </tr>
</table><form name="form1" method="post" action="propres.asp">
<table width="64%" border="1" bordercolor="#33CCFF"> <tr bgcolor="#0099FF">
  <td width="38%"><div align="center"><font face="Courier New"><strong><font
color="#CCFFFF">Old Properties</font></strong></font></div></td> <td colspan="2">
<blockquote> <p> <font color="#CCFFFF" face="Courier New"><strong>New Properties
</strong></font></p> </blockquote></td> </tr> <tr bgcolor="#66FFFF">
  <td><font color="#0099FF" face="Courier New"><strong>1.<%=objprop("p1")%></strong>
</font></td> <td colspan="2"><font color="#0099FF" face="Courier New"><strong>1.
<input type="text" name="text1"> </strong></font></td> </tr> <tr bgcolor="#66FFFF">
  <td><font color="#0099FF" face="Courier New"><strong>2.<%=objprop("p2")%></strong>
</font></td> <td colspan="2"><font color="#0099FF" face="Courier New"><strong>2.
<input type="text" name="text2"> </strong></font></td> </tr> <tr bgcolor="#66FFFF">
  <td><font color="#0099FF" face="Courier New"><strong>3.
<%=objprop("p3")%></strong></font></td> <td colspan="2"><font color="#0099FF"
face="Courier New"><strong>3. <input type="text" name="text3">

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



```

<td height="21" background="Myweb/bg/hunsa_top_menu08.gif"><p><strong><font size="1">
<font color="#CCCCCC"><a href="file:///main1.asp">HOME</a></font></font> <font
color="#666666" size="1"><a href="file:///Aboutme.asp">
ABOUT ME </a><a href="http://www.kmitl.ac.th"> KMITL </a>| ELECTRONICS
CLUB | ROBOT CLUB | ERC<font color="#0000CC"> |</font></font></strong></p></td>
</tr></table><table width="64%" border="1" cellspacing="1" bordercolor="#00FFCC">
<tr background="file:///wireless_bg.gif"><td colspan="2">&nbsp;</td> </tr> <tr>
<td width="34%"><font color="#0033FF">Silent status is</font><font
color="#0033FF">&nbsp;</font></td>
<td><font color="#0033FF"><%=objsrv("sil")%></font></td> </tr></table>
<form name="form1" method="post" action="silen1.asp">
<table width="64%" border="1" bordercolor="#663333"> <tr bgcolor="#CC6600"> <td>
<div align="center"><font color="#0033FF"></font><font color="#663333"><strong>Silent Control</strong></font></div>
</td> </tr> <tr bgcolor="#FF9900"> <td><strong><font color="#663333"></font>
</strong> <p><label></label><select name="select"><option>ON</option><option>OFF
</option></select>
<input type="submit" name="Submit" value="Submit">
When you control it !<br> </p> </td> </tr> <tr bgcolor="#CC6600">
<td> <div align="left"><a href="file:///main1.asp">Click here to main</a> </div></td>
</tr> </table></form></body></html>

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

AT89C51

Features

- Compatible with MCS-51™ Products
- 4K Bytes of In-System Reprogrammable Flash Memory
  - Endurance: 1,000 Write/Erase Cycles
- Fully Static Operation: 0 Hz to 24 MHz
- Three-Level Program Memory Lock
- 128 x 8-Bit Internal RAM
- 32 Programmable I/O Lines
- Two 16-Bit Timer/Counters
- Six Interrupt Sources
- Programmable Serial Channel
- Low Power Idle and Power Down Modes

Description

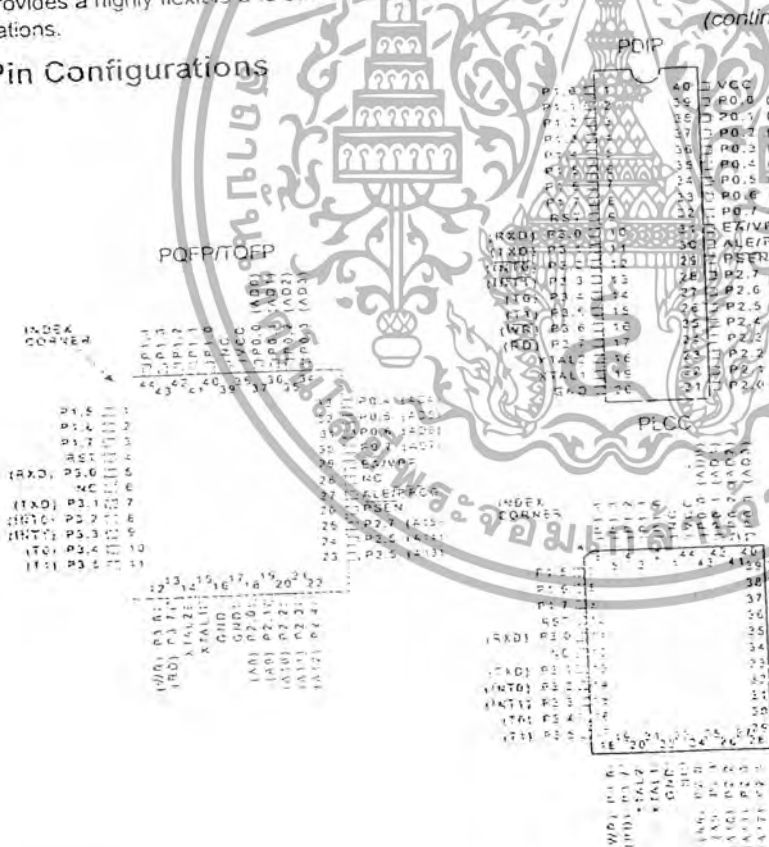
The AT89C51 is a low-power, high-performance CMOS 8-bit microcomputer with 4K bytes of Flash Programmable and Erasable Read Only Memory (PEROM). The device is manufactured using Atmel's high density nonvolatile memory technology and is compatible with the industry standard MCS-51™ instruction set and pinout. The on-chip Flash allows the program memory to be reprogrammed in-system or by a conventional nonvolatile memory programmer. By combining a versatile 8-bit CPU with Flash on a monolithic chip, the Atmel AT89C51 is a powerful microcomputer which provides a highly flexible and cost effective solution to many embedded control applications.



8-Bit  
Microcontroller  
with 4K Bytes  
Flash

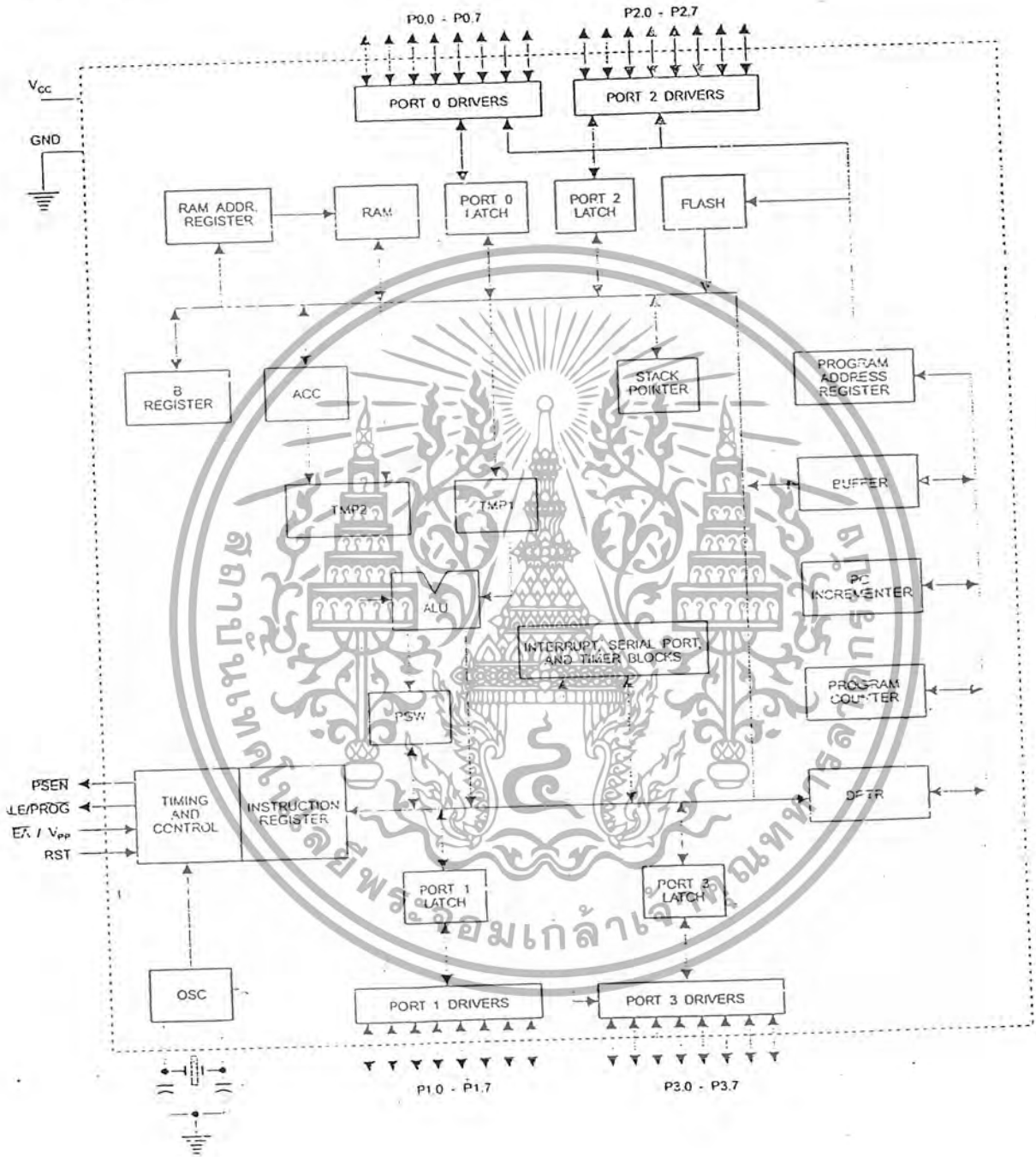
AT89C51

Pin Configurations





Diagram



AT89C51

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

The AT89C51 provides the following standard features: 4K bytes of Flash, 128 bytes of RAM, 32 I/O lines, two 16-bit timer/counters, a five vector two-level interrupt architecture, full duplex serial port, on-chip oscillator and clock circuitry. In addition, the AT89C51 is designed with static logic for operation down to zero frequency and supports two software selectable power saving modes. The Idle Mode stops the CPU while allowing the RAM, timer/counters, serial port and interrupt system to continue functioning. The Power Down Mode saves the RAM contents but freezes the oscillator disabling all other chip functions until the next hardware reset.

## Pin Description

**V<sub>CC</sub>**  
Supply voltage.

**GND**  
Ground.

**Port 0**  
Port 0 is an 8-bit open drain bidirectional I/O port. As an output port each pin can sink eight TTL inputs. When 1s are written to port 0 pins, the pins can be used as high-impedance inputs.

Port 0 may also be configured to be the multiplexed low-order address/data bus during accesses to external program and data memory. In this mode P0 has internal pullups.

Port 0 also receives the code bytes during Flash programming, and outputs the code bytes during program verification. External pullups are required during program verification.

**Port 1**  
Port 1 is an 8-bit bidirectional I/O port with internal pullups. The Port 1 output buffers can sink/source four TTL inputs. When 1s are written to Port 1 pins they are pulled high by the internal pullups and can be used as inputs. As inputs, Port 1 pins that are externally being pulled low will source current (I<sub>IL</sub>) because of the internal pullups.

Port 1 also receives the low-order address bytes during Flash programming and verification.

**Port 2**  
Port 2 is an 8-bit bidirectional I/O port with internal pullups. The Port 2 output buffers can sink/source four TTL inputs. When 1s are written to Port 2 pins they are pulled high by the internal pullups and can be used as inputs. As inputs, Port 2 pins that are externally being pulled low will source current (I<sub>IL</sub>) because of the internal pullups.

Port 2 emits the high-order address byte during fetches from external program memory and during accesses to external data memory that use 16-bit addresses (MOVX @ DPTR). In this application it uses strong internal pullups

when emitting 1s. During accesses to external data memory that use 8-bit addresses (MOVX @ RI), Port 2 emits the contents of the P2 Special Function Register.

Port 2 also receives the high-order address bits and some control signals during Flash programming and verification.

**Port 3**  
Port 3 is an 8-bit bidirectional I/O port with internal pullups. The Port 3 output buffers can sink/source four TTL inputs. When 1s are written to Port 3 pins they are pulled high by the internal pullups and can be used as inputs. As inputs, Port 3 pins that are externally being pulled low will source current (I<sub>IL</sub>) because of the pullups.

Port 3 also serves the functions of various special features of the AT89C51 as listed below:

Port Pin	Alternate Functions
P3.0	RXD (serial input port)
P3.1	TXD (serial output port)
P3.2	INT0 (external interrupt 0)
P3.3	INT1 (external interrupt 1)
P3.4	T0 (timer 0 external input)
P3.5	T1 (timer 1 external input)
P3.6	WR (external data memory write strobe)
P3.7	RD (external data memory read strobe)

Port 3 also receives some control signals for Flash programming and verification.

**RST**  
Reset input. A high on this pin for two machine cycles while the oscillator is running resets the device.

**ALE/PROG**  
Address Latch Enable output pulse for latching the low byte of the address during accesses to external memory. This pin is also the program pulse input (PROG) during Flash programming.

In normal operation ALE is emitted at a constant rate of 1/6 the oscillator frequency, and may be used for external timing or clocking purposes. Note, however, that one ALE pulse is skipped during each access to external Data Memory.

If desired, ALE operation can be disabled by setting bit 0 of SFR location 8EH. With the bit set, ALE is active only during a MOVX or MOVC instruction. Otherwise, the pin is weakly pulled high. Setting the ALE-disable bit has no effect if the microcontroller is in external execution mode.

**PSEN**  
Program Store Enable is the read strobe to external program memory.





When the AT89C51 is executing code from external program memory,  $\overline{\text{PSEN}}$  is activated twice each machine cycle, except that two  $\overline{\text{PSEN}}$  activations are skipped during each access to external data memory.

$\overline{\text{EA}}_{\text{VPP}}$

External Access Enable.  $\overline{\text{EA}}$  must be strapped to GND in order to enable the device to fetch code from external program memory locations starting at 0000H up to FFFFH. Note, however, that if lock bit 1 is programmed,  $\overline{\text{EA}}$  will be internally fetched on reset.

$\overline{\text{EA}}$  should be strapped to  $V_{\text{CC}}$  for internal program executions.

This pin also receives the 12-volt programming enable voltage ( $V_{\text{PP}}$ ) during Flash programming, for parts that require 12-volt  $V_{\text{PP}}$ .

XTAL1

Input to the inverting oscillator amplifier and input to the internal clock operating circuit.

XTAL2

Output from the inverting oscillator amplifier.

Oscillator Characteristics

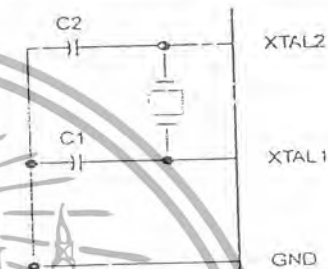
XTAL1 and XTAL2 are the input and output, respectively, of an inverting amplifier which can be configured for use as an on-chip oscillator, as shown in Figure 1. Either a quartz crystal or ceramic resonator may be used. To drive the device from an external clock source, XTAL2 should be left unconnected while XTAL1 is driven as shown in Figure 2. There are no requirements on the duty cycle of the external clock signal, since the input to the internal clocking circuitry is through a divide-by-two flip-flop, but minimum and maximum voltage high and low time specifications must be observed.

Idle Mode

In idle mode, the CPU puts itself to sleep while all the on-chip peripherals remain active. The mode is invoked by software. The content of the on-chip RAM and all the special functions registers remain unchanged during this mode. The idle mode can be terminated by any enabled interrupt or by a hardware reset.

It should be noted that when idle is terminated by a hardware reset, the device normally resumes program execution, from where it left off, up to two machine cycles before the internal reset algorithm takes control. On-chip hardware inhibits access to internal RAM in this event, but access to the port pins is not inhibited. To eliminate the possibility of an unexpected write to a port pin when Idle is terminated by reset, the instruction following the one that invokes Idle should not be one that writes to a port pin or to external memory.

Figure 1. Oscillator Connections



Note: C1, C2 = 30 pF ± 10 pF for Crystals  
= 40 pF ± 10 pF for Ceramic Resonators

Figure 2. External Clock Drive Configuration



Status of External Pins During Idle and Power Down Modes

Mode	Program Memory	ALE	$\overline{\text{PSEN}}$	PORT0	PORT1	PORT2	PORT3
Idle	Internal	1	1	Data	Data	Data	Data
Idle	External	1	1	Float	Data	Address	Data
Power Down	Internal	0	0	Data	Data	Data	Data
Power Down	External	0	0	Float	Data	Data	Data

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

**Power Down Mode**

In the power down mode the oscillator is stopped, and the instruction that invokes power down is the last instruction executed. The on-chip RAM and Special Function Registers retain their values until the power down mode is terminated. The only exit from power down is a hardware reset. Reset redefines the SFRs but does not change the on-chip RAM. The reset should not be activated before V<sub>CC</sub> is restored to its normal operating level and must be held active long enough to allow the oscillator to restart and stabilize.

**Lock Bit Protection Modes**

Program Lock Bits			Protection Type	
LB1	LB2	LB3		
1	U	U	U	No program lock features.
2	P	U	U	MOV <sub>C</sub> instructions executed from external program memory are disabled from fetching code bytes from internal memory, EA is sampled and latched on reset, and further programming of the Flash is disabled.
3	P	P	U	Same as mode 2, also verify is disabled.
4	P	P	P	Same as mode 3, also external execution is disabled.

**Programming the Flash**

The AT89C51 is normally shipped with the on-chip Flash memory array in the erased state (that is, contents = FFH) and ready to be programmed. The programming interface accepts either a high-voltage (12-volt) or a low-voltage (V<sub>CC</sub>) program enable signal. The low voltage programming mode provides a convenient way to program the AT89C51 inside the user's system, while the high-voltage programming mode is compatible with conventional third party Flash or EPROM programmers.

The AT89C51 is shipped with either the high-voltage or low-voltage programming mode enabled. The respective top-side marking and device signature codes are listed in the following table.

	V <sub>PP</sub> = 12V	V <sub>PP</sub> = 5V
Top-Side Mark	AT89C51 xxxx yyww	AT89C51 xxxx-5 yyww
Signature	(030H)=1EH (031H)=51H (032H)=FFH	(030H)=1EH (031H)=51H (032H)=05H

The AT89C51 code memory array is programmed byte-by-byte in either programming mode. To program any non-blank byte in the on-chip Flash Memory, the entire memory must be erased using the Chip Erase Mode.

**Program Memory Lock Bits**

On the chip are three lock bits which can be left unprogrammed (U) or can be programmed (P) to obtain the additional features listed in the table below:

When lock bit 1 is programmed, the logic level at the EA pin is sampled and latched during reset. If the device is powered up without a reset, the latch initializes to a random value, and holds that value until reset is activated. It is necessary that the latched value of EA be in agreement with the current logic level at that pin in order for the device to function properly.

**Programming Algorithm:** Before programming the AT89C51, the address, data and control signals should be set up according to the Flash programming mode table and Figures 3 and 4. To program the AT89C51, take the following steps:

1. Input the desired memory location on the address lines.
2. Input the appropriate data byte on the data lines.
3. Activate the correct combination of control signals.
4. Raise EA/V<sub>PP</sub> to 12V for the high-voltage programming mode.
5. Pulse ALE/PROG once to program a byte in the Flash array or the lock bits. The byte-write cycle is self-timed and typically takes no more than 1.5 ms. Repeat steps 1 through 5, changing the address and data for the entire array or until the end of the object file is reached.

**Data Polling:** The AT89C51 features Data Polling to indicate the end of a write cycle. During a write cycle, an attempted read of the last byte written will result in the complement of the written datum on P0.7. Once the write cycle has been completed, true data are valid on all outputs, and the next cycle may begin. Data Polling may begin any time after a write cycle has been initiated.

**Ready/Busy:** The progress of byte programming can also be monitored by the RDY/BSY output signal. P3.4 is pulled low after ALE goes high during programming to indicate BUSY. P3.4 is pulled high again when programming is done to indicate READY.



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



**Program Verify:** If lock bits LB1 and LB2 have not been programmed, the programmed code data can be read back on the address and data lines for verification. The lock bits cannot be verified directly. Verification of the lock bits is achieved by observing that their features are enabled.

**Chip Erase:** The entire Flash array is erased electrically by using the proper combination of control signals and by holding ALE/PROG low for 10 ms. The code array is written with all "1"s. The chip erase operation must be executed before the code memory can be re-programmed.

**Reading the Signature Bytes:** The signature bytes are read by the same procedure as a normal verification of locations 030H.

031H, and 032H, except that P3.6 and P3.7 must be pulled to a logic low. The values returned are as follows.

- (030H) = 1EH indicates manufactured by Atmel
- (031H) = 51H indicates 89C51
- (032H) = FFH indicates 12V programming
- (032H) = 05H indicates 5V programming

### Programming Interface

Every code byte in the Flash array can be written and the entire array can be erased by using the appropriate combination of control signals. The write operation cycle is self-timed and once initiated, will automatically time itself to completion.

All major programming vendors offer worldwide support for the Atmel microcontroller series. Please contact your local programming vendor for the appropriate software revision.

### Flash Programming Modes

Mode	RST	PSEN	ALE/PROG	E <sub>AVDD</sub>	P2.6	P2.7	P3.6	P3.7
Write Code Data	H	L	H	H/12V	L	H	H	H
Read Code Data	H	L	H	H/12V	H	H	H	H
Write Lock	Bit-1	H	L	H/12V	H	H	L	L
	Bit-2	H	L	H/12V	H	L	H	L
	Bit-3	H	L	H/12V	H	L	L	L
Chip Erase	H	L	H	H/12V	H	L	L	L
Read Signature Byte	H	L	H	H	L	L	L	L

Note: 1. Chip Erase requires a 10-ms PROG pulse.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

AT89C51

Figure 3. Programming the Flash

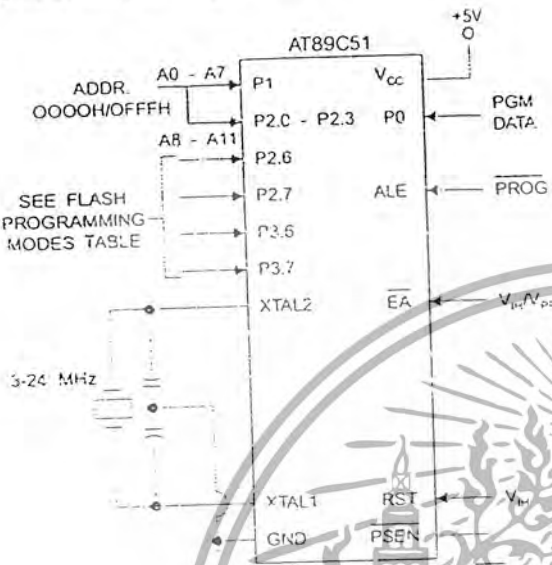
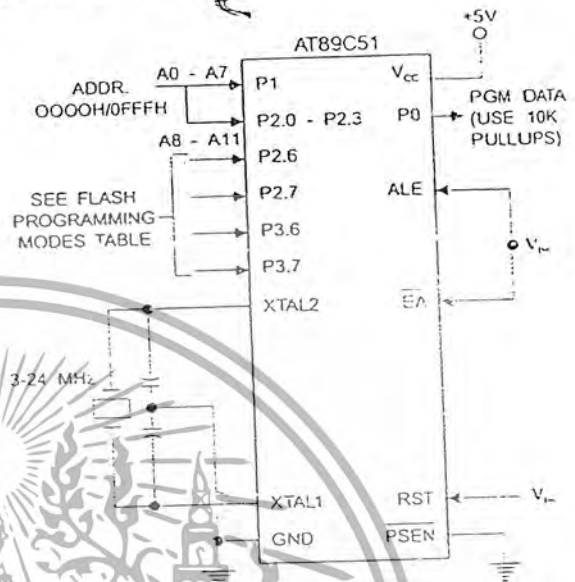


Figure 4. Verifying the Flash



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
$V_{PP}^{(1)}$	Programming Enable Voltage	11.5	12.5	V
$I_{PP}^{(1)}$	Programming Enable Current		1.0	mA
$1/t_{CLCL}$	Oscillator Frequency	3	24	MHz
$t_{AVGL}$	Address Setup to PROG Low	$48t_{CLCL}$		
$t_{GHAX}$	Address Hold After PROG	$48t_{CLCL}$		
$t_{DVGL}$	Data Setup to PROG Low	$48t_{CLCL}$		
$t_{GHDX}$	Data Hold After PROG	$48t_{CLCL}$		
$t_{EHS}$	P2.7 (ENABLE) High to $V_{PP}$	$48t_{CLCL}$		
$t_{SHGL}$	$V_{PP}$ Setup to PROG Low	10		$\mu\text{s}$
$t_{GHSL}^{(1)}$	$V_{PP}$ Hold After PROG	10		$\mu\text{s}$
$t_{CLGH}$	PROG Width	1	110	$\mu\text{s}$
$t_{AVQV}$	Address to Data Valid		$48t_{CLCL}$	
$t_{ELCV}$	ENABLE Low to Data Valid		$48t_{CLCL}$	
$t_{EHOZ}$	Data Float After ENABLE	0	$48t_{CLCL}$	
$t_{GHBL}$	PROG High to BUSY Low		1.0	$\mu\text{s}$
$t_{WC}$	Byte Write Cycle Time		2.0	ms

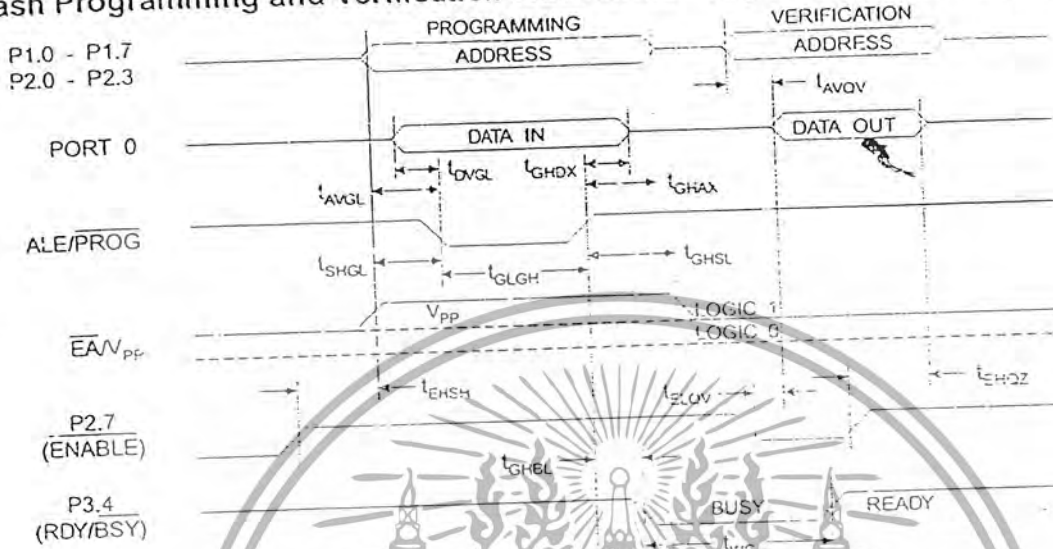
Note: 1. Only used in 12-volt programming mode.



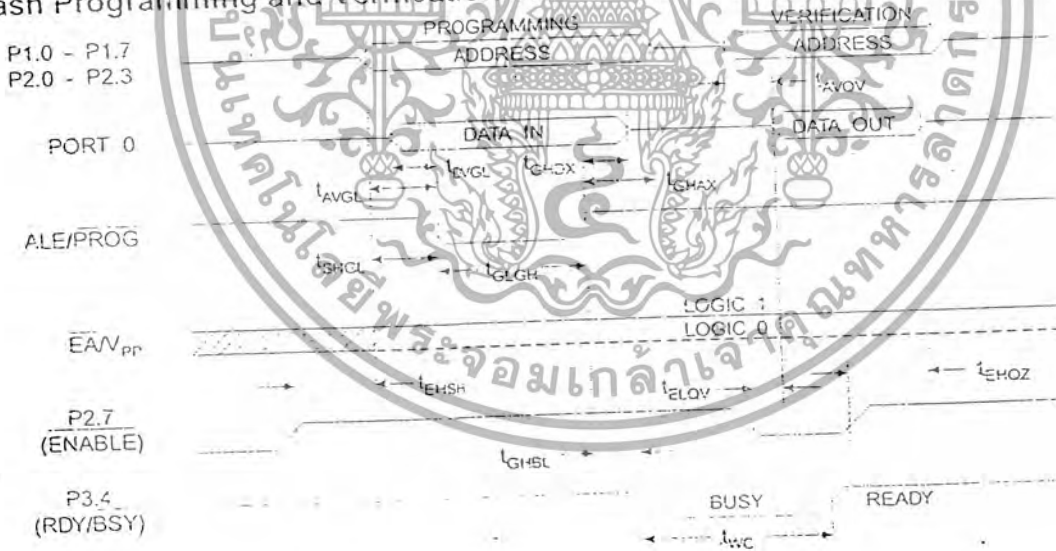
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



Flash Programming and Verification Waveforms - High Voltage Mode ( $V_{PP} = 12V$ )



Flash Programming and Verification Waveforms - Low Voltage Mode ( $V_{PP} = 5V$ )



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

**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.....	15.0 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.

**IC Characteristics**

$T_A = -40^\circ\text{C}$  to  $85^\circ\text{C}$ ,  $V_{CC} = 5.0\text{V} \pm 20\%$  (unless otherwise noted)

Symbol	Parameter	Condition	Min	Max	Units
$V_{IL}$	Input Low Voltage	(Except EA)	-0.5	$0.2 V_{CC} - 0.1$	V
$V_{IL1}$	Input Low Voltage (EA)		-0.5	$0.2 V_{CC} - 0.3$	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 1,2,3)	$I_{OL} = 1.6 \text{ mA}$		0.45	V
$V_{OL1}$	Output Low Voltage <sup>(1)</sup> (Port 0, ALE, PSEN)	$I_{OL} = 3.2 \text{ mA}$		0.45	V
$V_{OH}$	Output High Voltage (Ports 1,2,3, ALE, PSEN)	$I_{OH} = -60 \mu\text{A}$ , $V_{CC} = 5\text{V} \pm 10\%$		$0.75 V_{CC}$	V
		$I_{OH} = -25 \mu\text{A}$		$0.9 V_{CC}$	V
		$I_{OH} = -10 \mu\text{A}$		$0.9 V_{CC}$	V
$V_{OH1}$	Output High Voltage (Port 0 in External Bus Mode)	$I_{OH} = -800 \mu\text{A}$ , $V_{CC} = 5\text{V} \pm 10\%$		2.4	V
		$I_{OH} = -200 \mu\text{A}$		$0.75 V_{CC}$	V
		$I_{OH} = -80 \mu\text{A}$		$0.9 V_{CC}$	V
$I_{IL}$	Logical 0 Input Current (Ports 1,2,3)	$V_{in} = 0.45\text{V}$		-50	$\mu\text{A}$
$I_{TL}$	Logical 1 to 0 Transition Current (Ports 1,2,3)	$V_{in} = 2\text{V}$ , $V_{CC} = 5\text{V} \pm 10\%$		-650	$\mu\text{A}$
$I_{LI}$	Input Leakage Current (Port 0, EA)	$0.45 < V_{in} < V_{CC}$		$\pm 10$	$\mu\text{A}$
RRST	Reset Pull-down Resistor			300	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		20	mA
		Idle Mode, 12 MHz		5	mA
	Power Down Mode <sup>(2)</sup>	$V_{CC} = 6\text{V}$		100	$\mu\text{A}$
		$V_{CC} = 3\text{V}$		40	$\mu\text{A}$

- Notes: 1 Under steady state (non-transient) conditions,  $I_{OL}$  must be externally limited as follows:  
 Maximum  $I_{OL}$  per port pin: 10 mA  
 Maximum  $I_{OL}$  per 8-bit port: Port 0: 26 mA  
 Ports 1, 2, 3: 15 mA  
 Maximum total  $I_{OL}$  for all output pins: 71 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 อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



## C Characteristics

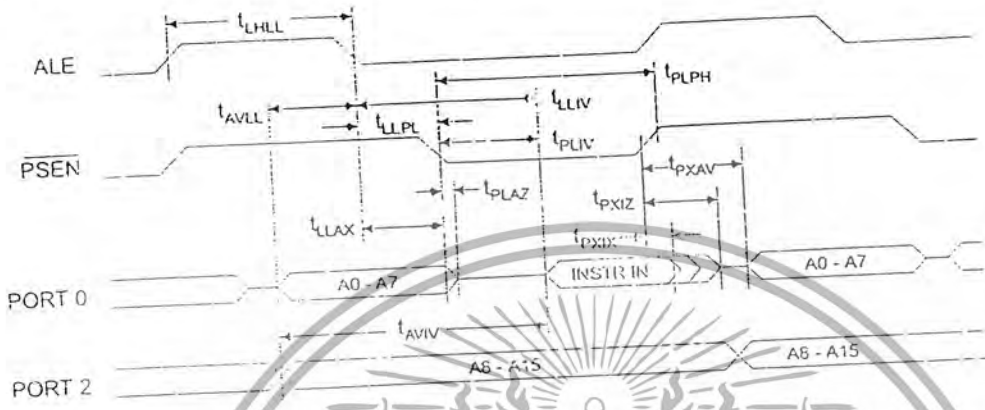
Under Operating Conditions; Load Capacitance for Port 0, ALE/PROG, and PSEN = 100 pF; Load Capacitance for all other inputs = 80 pF)

### External Program and Data Memory Characteristics

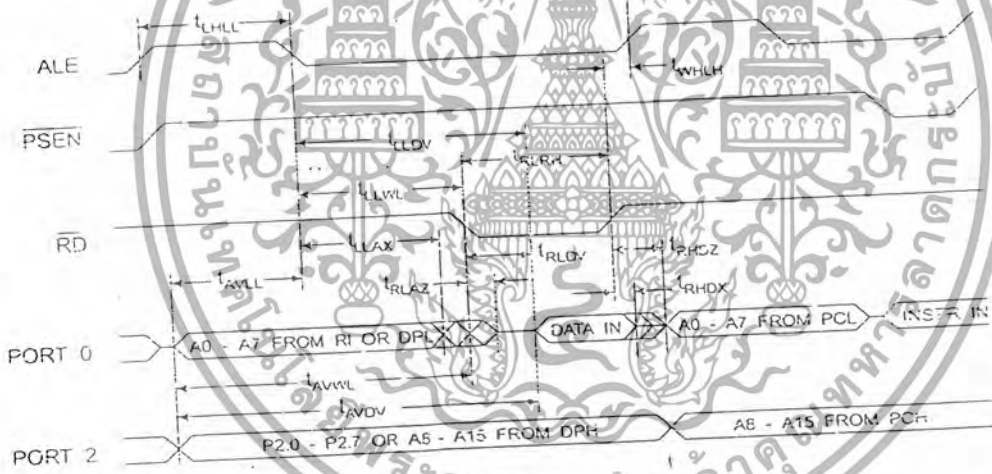
Symbol	Parameter	12 MHz Oscillator		16 to 24 MHz Oscillator		Units
		Min	Max	Min	Max	
$1/t_{CLCL}$	Oscillator Frequency			0	24	MHz
$t_{LHLL}$	ALE Pulse Width	127		$2t_{CLCL}-40$		ns
$t_{AVLL}$	Address Valid to ALE Low	43		$t_{CLCL}-13$		ns
$t_{LLAX}$	Address Hold After ALE Low	48		$t_{CLCL}-20$		ns
$t_{LLIV}$	ALE Low to Valid Instruction In		233		$4t_{CLCL}-65$	ns
$t_{LLPL}$	ALE Low to PSEN Low	43		$t_{CLCL}-13$		ns
$t_{PLPH}$	PSEN Pulse Width	205		$3t_{CLCL}-20$		ns
$t_{PLIV}$	PSEN Low to Valid Instruction In		145		$3t_{CLCL}-45$	ns
$t_{PXIX}$	Input Instruction Hold After PSEN	0		0		ns
$t_{PXIZ}$	Input Instruction Float After PSEN		59		$t_{CLCL}-10$	ns
$t_{PXAV}$	PSEN to Address Valid	75		$t_{CLCL}-8$		ns
$t_{AVIV}$	Address to Valid Instruction In		312		$5t_{CLCL}-55$	ns
$t_{PLAZ}$	PSEN Low to Address Float		10		10	ns
$t_{RLRH}$	RD Pulse Width	400		$6t_{CLCL}-100$		ns
$t_{WLWH}$	WR Pulse Width	400		$6t_{CLCL}-100$		ns
$t_{RLDX}$	RD Low to Valid Data In		252		$5t_{CLCL}-90$	ns
$t_{RHDX}$	Data Hold After RD	0		0		ns
$t_{RHDX}$	Data Float After RD		97		$2t_{CLCL}-28$	ns
$t_{LLDV}$	ALE Low to Valid Data In		517		$5t_{CLCL}-150$	ns
$t_{AVDV}$	Address to Valid Data In		585		$5t_{CLCL}-165$	ns
$t_{LLWL}$	ALE Low to RD or WR Low	200	300	$3t_{CLCL}-50$	$2t_{CLCL}+50$	ns
$t_{AVWL}$	Address to RD or WR Low	203		$4t_{CLCL}-75$		ns
$t_{QVWX}$	Data Valid to WR Transition	28		$t_{CLCL}-20$		ns
$t_{QVWH}$	Data Valid to WR High	433		$7t_{CLCL}-120$		ns
$t_{WHDX}$	Data Hold After WR	33		$t_{CLCL}-20$		ns
$t_{RLAZ}$	RD Low to Address Float		0		0	ns
$t_{WHLH}$	RD or WR High to ALE High	43	123	$t_{CLCL}-20$	$t_{CLCL}+25$	ns

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ernal Program Memory Read Cycle



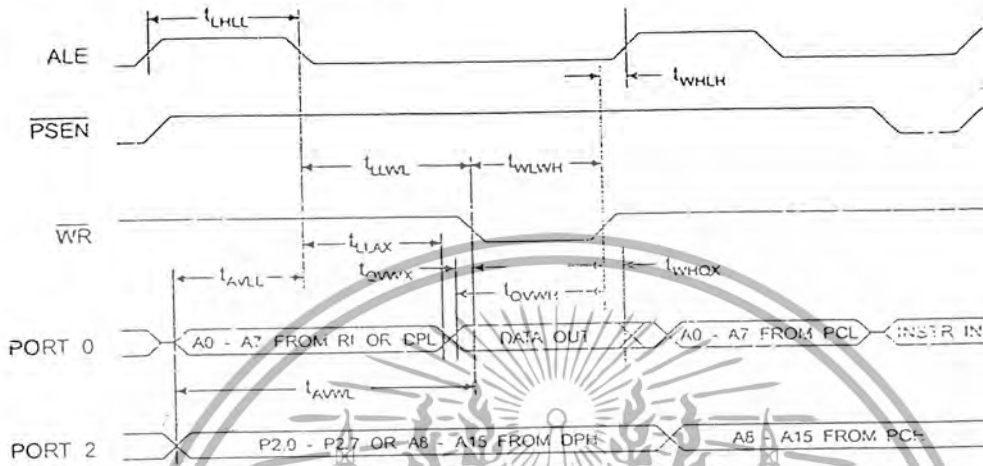
External Data Memory Read Cycle



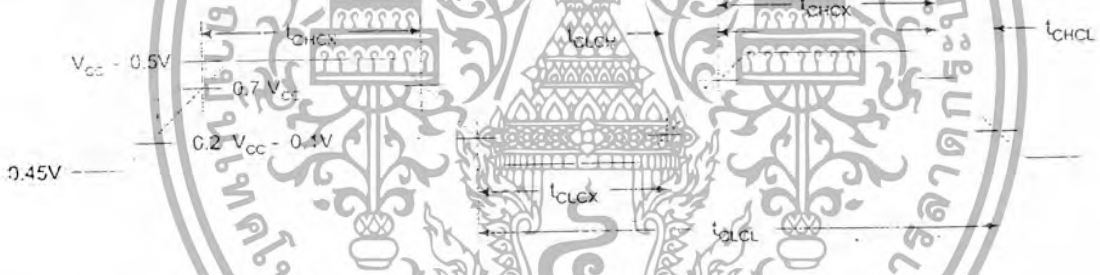
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



External Data Memory Write Cycle



External Clock Drive Waveforms



External Clock Drive

Symbol	Parameter	Min	Max	Units
$1/t_{CLCL}$	Oscillator Frequency	0	24	MHz
$t_{CLCL}$	Clock Period	41.6		ns
$t_{CHCX}$	High Time	15		ns
$t_{CLCX}$	Low Time	15		ns
$t_{CLCH}$	Rise Time		20	ns
$t_{CHCL}$	Fall Time		20	ns

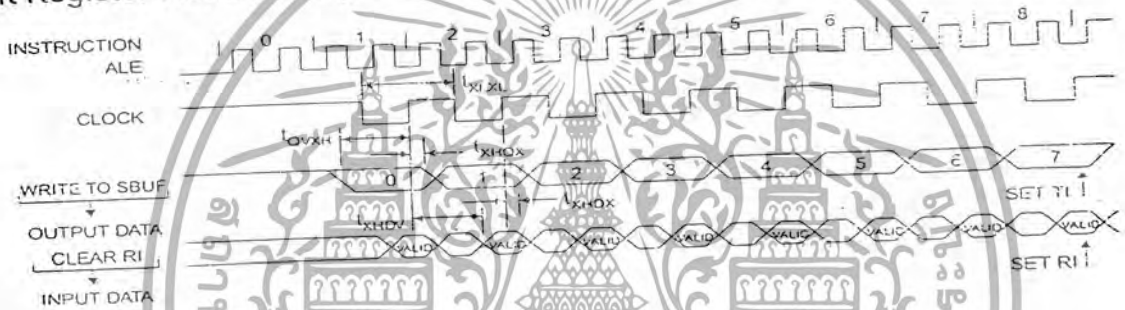
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Serial Port Timing: Shift Register Mode Test Conditions

V<sub>CC</sub> = 5.0 V ± 20%; Load Capacitance = 80 pF)

Symbol	Parameter	12 MHz Osc		Variable Oscillator		Units
		Min	Max	Min	Max	
t <sub>XLXL</sub>	Serial Port Clock Cycle Time	1.0		12t <sub>CLCL</sub>		μs
t <sub>OVXH</sub>	Output Data Setup to Clock Rising Edge	700		10t <sub>CLCL</sub> -133		ns
t <sub>XHOX</sub>	Output Data Hold After Clock Rising Edge	50		2t <sub>CLCL</sub> -117		ns
t <sub>XHDX</sub>	Input Data Hold After Clock Rising Edge	0		0		ns
t <sub>XHOV</sub>	Clock Rising Edge to Input Data Valid		700		10t <sub>CLCL</sub> -133	ns

Shift Register Mode Timing Waveforms



AC Testing Input/Output Waveforms (1) Float Waveforms (1)



- Note: 1. AC Inputs during testing are driven at V<sub>CC</sub> = 0.5V for a logic 1 and 0.45V for a logic 0. Timing measurements are made at V<sub>IH</sub> min. for a logic 1 and V<sub>IL</sub> max. for a logic 0.
- Note: 1. For timing purposes, a port pin is no longer floating when a 100 mV change from load voltage occurs. A port pin begins to float when 100 mV change from the loaded V<sub>OH</sub>/V<sub>OL</sub> level occurs.



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



## Ordering Information

Lead (z)	Power Supply	Ordering Code	Package	Operation Range
2	5V ± 20%	AT89C51-12AC	44A	Commercial (0°C to 70°C)
		AT89C51-12JC	44J	
		AT89C51-12PC	40P6	
		AT89C51-12QC	44Q	
		AT89C51-12AI	44A	Industrial (-40°C to 85°C)
		AT89C51-12JI	44J	
		AT89C51-12PI	40P6	
		AT89C51-12QI	44Q	
		AT89C51-12AA	44A	Automotive (-40°C to 105°C)
		AT89C51-12JA	44J	
		AT89C51-12PA	40P6	
		AT89C51-12QA	44Q	
16	5V ± 20%	AT89C51-16AC	44A	Commercial (0°C to 70°C)
		AT89C51-16JC	44J	
		AT89C51-16PC	40P6	
		AT89C51-16QC	44Q	
		AT89C51-16AI	44A	Industrial (-40°C to 85°C)
		AT89C51-16JI	44J	
		AT89C51-16PI	40P6	
		AT89C51-16QI	44Q	
		AT89C51-16AA	44A	Automotive (-40°C to 105°C)
		AT89C51-16JA	44J	
		AT89C51-16PA	40P6	
		AT89C51-16QA	44Q	
20	5V ± 20%	AT89C51-20AC	44A	Commercial (0°C to 70°C)
		AT89C51-20JC	44J	
		AT89C51-20PC	40P6	
		AT89C51-20QC	44Q	
		AT89C51-20AI	44A	Industrial (-40°C to 85°C)
		AT89C51-20JI	44J	
		AT89C51-20PI	40P6	
		AT89C51-20QI	44Q	

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## AT89C51

## Ordering Information

Speed (MHz)	Power Supply	Ordering Code	Package	Operation Range
24	5V $\pm$ 20%	AT89C51-24AC	44A	Commercial (0°C to 70°C)
		AT89C51-24JC	44J	
		AT89C51-24PC	44P6	
		AT89C51-24QC	44Q	
		AT89C51-24AI	44A	Industrial (-40°C to 85°C)
		AT89C51-24JI	44J	
		AT89C51-24PI	44P6	
		AT89C51-24QI	44Q	



Package Type	
44A	44 Lead, Thin Plastic Gull Wing Quad Flatpack (TQFP)
44J	44 Lead, Plastic J-Leaded Chip Carrier (PLCC)
40P6	40 Lead, 0.600" Wide, Plastic Dual Inline Package (PDIP)
44Q	44 Lead, Plastic Gull Wing Quad Flatpack (PQFP)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## กิตติกรรมประกาศ

- ศ.ดร.วัลลภ สุระกำพลธร ที่คอยให้คำปรึกษาด้านต่าง ๆ ที่เป็นประโยชน์ต่อโครงการงานและการดำเนินชีวิต
- บิดา มารดาและน้องๆ โดยเฉพาะอย่างยิ่ง คุณ รมชัย สุนทร ที่คอยช่วยเหลือในด้านการเงินมาโดยตลอดและยังคอยช่วยจัดหาตำราที่เป็นประโยชน์ให้อีกหลายเล่ม
- เพื่อนๆ ที่คอยให้กำลังใจและให้ยืมอุปกรณ์ต่างๆ

จึงแสดงความขอบคุณอย่างสูงมา ณ ที่นี้

จากผู้จัดทำ



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## หนังสืออ้างอิง

1. สัจจะ จรัสรุ่งรวีวร, “คู่มือการสร้างแอปพลิเคชันด้วย Visual Basic 6 Basic & Advance”, สำนักพิมพ์อินโฟเพรส, 2542
2. สัจจะ จรัสรุ่งรวีวร, “Internet Programming ด้วย Visual Basic 6 และ ASP”, สำนักพิมพ์ อินโฟเพรส, 2542
3. สัจจะ จรัสรุ่งรวีวร,สมพร จิวรสกุล, “Active Server Page และ แอปพลิเคชัน ฐานข้อมูลบนอินเทอร์เน็ต”, สำนักพิมพ์ อินโฟเพรส, 2541
4. ชีรวัฒน์ ประกอบผล, “การประยุกต์ใช้งาน ไมครคอนโทรลเลอร์”, สมาคมส่งเสริมเทคโนโลยี (ไทย-ญี่ปุ่น), 2542
5. อรรถผล บุญยะ โภคา, “เรียนรู้และปฏิบัติการเชื่อมต่อคอมพิวเตอร์กับอุปกรณ์ภายนอกผ่านพอร์ตอนุกรม”, บริษัท อินโนเวทีฟ เอ็กเพอริเมนต์



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้