สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

ระบบฝังตัวสำหรับการควบคุมระบบเซ็นเซอร์แบบไร้สาย

EMBEDDED SYSTEM BASED CONTROL UNIT FOR WIRELESS SENSOR NETWORK

T 061869

โดย

นายเจนณรงค์ ฉิมมัจฉา

นายศิริชัย ตระกูลรุ่งจรัส

วพ.
จ7158
2547

เลขหมู่.....
เลขทะเบียน 61869
วัน,เดือน,ปี 24 ก.ค. 2549

b 11568811
i.....

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต

ภาควิชาวิศวกรรมสารสนเทศ

คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2547

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

EMBEDDED SYSTEM BASED CONTROL UNIT FOR WIRELESS SENSOR NETWORK

BY

MR. JANENARONG CHIMMAJCHA

MR. SIRICHAI TRAGULRUNGJARAS

A THESIS SUBMITTED IN PARTIAL FULFILLMENT OF

THE REQUIREMENT FOR THE DEGREE OF

BACHELOR IN DEPARTMENT OF INFORMATION ENGINEERING

FACULTY OF ENGINEERING

KING MONGKUT'S INSTITUTE OF TECHNOLOGY LADKRABANG

2004

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หัวข้อปริญญานิพนธ์	ระบบฝังตัวสำหรับการควบคุมระบบเซ็นเซอร์แบบไร้สาย
ชื่อนักศึกษา	นายเจนณรงค์ ฉิมมัจฉา รหัสประจำตัว 45015838
	นายศิริชัย ตระกูลรุ่งจรัส รหัสประจำตัว 45015867
อาจารย์ที่ปรึกษา	อาจารย์พนารัตน์ ระวิวรรณ
	ผศ. มนต์ชัย แช่มช้อย
ระดับการศึกษา	ปริญญาตรี วิศวกรรมศาสตรบัณฑิต
	สาขาวิศวกรรมสารสนเทศ
ภาควิชา	วิศวกรรมสารสนเทศ
ปีการศึกษา	2547

ปริญญานิพนธ์ฉบับนี้ได้รับการอนุมัติเป็นส่วนหนึ่งของการศึกษาตามหลักสูตร วิศวกรรมศาสตรบัณฑิต
คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

.....

(อาจารย์พนารัตน์ ระวิวรรณ)

อาจารย์ที่ปรึกษา

.....

(ผศ. มนต์ชัย แช่มช้อย)

อาจารย์ที่ปรึกษา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หัวข้อปริญญานิพนธ์	ระบบฝังตัวสำหรับการควบคุมระบบเซ็นเซอร์แบบไร้สาย
ชื่อนักศึกษา	นายเจนณรงค์ ฉิมมัจฉา รหัสประจำตัว 45015838
	นายศิริชัย ตระกูลรุ่งจรัส รหัสประจำตัว 45015867
อาจารย์ที่ปรึกษา	อาจารย์พนารัตน์ ระวิวรรณ
	ผศ. มนต์ชัย แช่มช้อย
ระดับการศึกษา	ปริญญาตรี วิศวกรรมศาสตรบัณฑิต
	สาขาวิศวกรรมสารสนเทศ
ภาควิชา	วิศวกรรมสารสนเทศ
ปีการศึกษา	2547

บทคัดย่อ

โครงงานนี้เป็นการประยุกต์ใช้ระบบคอมพิวเตอร์ฝังตัว (Embedded System) กับงานควบคุมการจ่ายน้ำทางการเกษตร โดยตัวระบบคอมพิวเตอร์ฝังตัวสามารถรับข้อมูลได้ 2 รูปแบบ คือ 1. สั่งงานจากคอมพิวเตอร์ผ่านระบบเครือข่ายอินเตอร์เน็ทผ่านระบบคอมพิวเตอร์ฝังตัวเพื่อไปควบคุมปั๊มในการจ่ายน้ำโดยเขียนคำสั่งต่างๆเพื่อให้ระบบคอมพิวเตอร์ฝังตัวรับข้อมูลจากระบบเครือข่ายคอมพิวเตอร์แล้วทำการประมวลผลเพื่อทำการสั่งงานควบคุมการจ่ายน้ำของปั๊มได้ 2. ระบบคอมพิวเตอร์ฝังตัวสามารถรับข้อมูลจากระบบเซ็นเซอร์แบบไร้สายเพื่อไปควบคุมปั๊มน้ำ กล่าวคือทั้งสองระบบจะสามารถติดต่อถึงกันได้โดยจะทำการสร้างโปรโตคอลติดต่อระหว่างระบบคอมพิวเตอร์ฝังตัวกับระบบเซ็นเซอร์แบบไร้สายเพื่อติดต่อรับข้อมูลและสั่งงานควบคุมปั๊มน้ำ โดยที่ระบบเซ็นเซอร์แบบไร้สายนั้นจะทำการติดต่อกันภายในระบบโดยเราจะทำการสร้างโปรโตคอลให้ชุดเซ็นเซอร์ แต่ละตัวภายในระบบสามารถติดต่อกันได้อย่างมีประสิทธิภาพ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Thesis Title	Embedded System Based Control Unit for Wireless Sensor Network
Student	Mr.Janenarong Chimmajcha ID. 45015838
	Mr.Sirichai Tragulrungjaras ID. 45015867
Advisor	Mrs. Panarat Raveewan
	Asst. Prof. Monchai Chamchoy
Graduate	Bachelor Degree of Information Engineering
Department	Information Engineering
Academic Year	2004

ABSTRACT

In this project is applied Embedded System to control water pump in agriculture. The Embedded System is able to receive data by 2 ways. First, control system via internet network using Embedded System for controlling water pump by programmed Embedded System to receive data from internet network then compute and send command to control water pump. The other way Embedded System can receive data from Wireless Sensor Network System for controlling the water pump .Both of the systems can connect each other to control the water pump.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กิตติกรรมประกาศ

ปริญญานิพนธ์ฉบับนี้คงจะไม่สำเร็จได้ด้วยดี หากไม่ได้รับความร่วมมือจากอาจารย์ทุกท่าน รุ่นพี่ และเพื่อนๆทุกคน ที่ได้ให้ความช่วยเหลือทั้งด้านข้อมูล อุปกรณ์ คำแนะนำติชมต่างๆอันเป็นประโยชน์ และท่านที่ต้องขอขอบพระคุณอย่างที่สุดคือ ผู้ช่วยศาสตราจารย์ มนต์ชัย แช่มช้อย และอาจารย์ พนารัตน์ ระวิวรรณ อาจารย์ที่ปรึกษาปริญญานิพนธ์ที่คอยช่วยเหลือ ให้คำปรึกษา และให้คำแนะนำให้ได้มีความคิดและแนวทางในการแก้ไขปัญหาในการทำโครงงาน ทำให้โครงงานนี้สำเร็จไปได้ด้วยดี

นายเจนณรงค์ ฉิมมัณฉา

นายศิริชัย ตระกูลรุ่งจรัส

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ

เรื่อง	หน้า
บทคัดย่อภาษาไทย	ก
บทคัดย่อภาษาอังกฤษ	ข
กิตติกรรมประกาศ	ค
สารบัญ	ง
สารบัญรูป	ฉ
สารบัญตาราง	ซ
บทที่ 1 บทนำ	
1.1 วัตถุประสงค์	1
1.2 แนวคิดและที่มา	1
1.3 ขอบเขตของโครงงาน	3
1.4 ผลที่คาดว่าจะได้รับ	3
1.5 ประโยชน์ที่ได้	3
บทที่ 2 ทฤษฎีและหลักการ	
2.1 สถาปัตยกรรมและโครงสร้างทางฮาร์ดแวร์ PIC16F628	4
2.2 เครื่องรับ rfRXDO0420	9
2.3 ระบบคอมพิวเตอร์ฝังตัว	10
2.4 ทฤษฎีการสื่อสารอนุกรม	16
2.5 ภาษา PHP	23
2.6 ภาษา Visual Basic.Net	26
บทที่ 3 การออกแบบและการสร้าง	
3.1 โปรแกรมควบคุม	29
3.2 วงจรควบคุม	40
บทที่ 4 ผลการทดลอง	
4.1 การทดลองโปรแกรมควบคุมปั๊มและวาล์วผ่านระบบอินเตอร์เน็ต	43
4.2 การทดลองการควบคุมวาล์วและปั๊มผ่านโครงข่ายเซ็นเซอร์แบบไร้สาย	46

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ (ต่อ)

เรื่อง **หน้า**

บทที่ 5 สรุปผลการทดลองและปัญหาที่พบ

5.1 สรุปผล 49

5.2 ปัญหาที่พบ 49

5.3 ข้อเสนอแนะ 49

บรรณานุกรม

ภาคผนวก ก.วงจรที่ใช้งาน

ภาคผนวก ข. ลายพิมพ์วงจรและการจัดวางอุปกรณ์

ภาคผนวก ค. โปรแกรมที่ใช้งาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป

รูป **หน้า**

รูปที่ 1.1 สถาปัตยกรรมของระบบ 2
รูปที่ 2.1 การจัดขาของไมโครคอนโทรเลอร์ PIC16F628 4
รูปที่ 2.2 กระบวนการเฟชต์และเอ็กซิคิวต์ในรอบการทำงานของ PIC16F628 5
รูปที่ 2.3 กระบวนการไปป์ไลน์ 5
รูปที่ 2.4 ไดอะแกรมโครงสร้างการทำงานของไมโครคอนโทลเลอร์ PIC16F628 7
รูปที่ 2.5 บล็อกไดอะแกรมของ rfRXDO0420 9
รูปที่ 2.6 โครงสร้างระบบคอมพิวเตอร์ฝังตัว 12
รูปที่ 2.7 การข้อมูลตัวอย่างของการส่งแบบอนุกรม 16
รูปที่ 2.8 การส่งข้อมูลตัวอย่างของการส่งแบบขนาน 17
รูปที่ 2.9 ลักษณะสัญญาณ 17
รูปที่ 2.10 การเพิ่มบิตเริ่มเข้าไป 18
รูปที่ 2.11 การเพิ่มพาริตี้บิตเข้าไป 20
รูปที่ 2.12 รูปแบบของข้อมูลแต่ละไบท์ในการรับส่งข้อมูลแบบอนุกม 20
รูปที่ 2.13 แผนผังคอนเน็กเตอร์ของ RS-232 22
รูปที่ 2.14 บล็อกไดอะแกรมลักษณะการทำงานของ PHP 25
รูปที่ 3.1 ผังงานโปรแกรมควบคุมปั๊ม จากคอมพิวเตอร์ไปยังระบบคอมพิวเตอร์ฝังตัว 30
รูปที่ 3.2 การสร้างการติดต่อ PHP กับไฟล์ 31
รูปที่ 3.3 การสร้างการติดต่อ PHP กับฐานข้อมูล 32
รูปที่ 3.4 หน้าจอล็อกอิน ระบบควบคุมปั๊มและวาล์ว 33
รูปที่ 3.5 หน้าจอเมื่อใส่รหัสผิดพลาด 33
รูปที่ 3.6 หน้าจอหลักและสถานะของการควบคุมปั๊มผ่านทางอินเตอร์เน็ต 34
รูปที่ 3.7 หน้าจอการตั้งเวลาเปิดปิดล่วงหน้า 35
รูปที่ 3.8 ฐานข้อมูลการตั้งเวลาเปิดปิดวาล์วล่วงหน้า 35
รูปที่ 3.9 ผังงาน Visual Basic.NET ของคอมพิวเตอร์ฝังตัว 36
รูปที่ 3.10 การสร้างการติดต่อพอร์ตอนุกรมของ Visual Basic.NET 37
รูปที่ 3.11 การสร้าง Visual Basic.NET ติดต่อกับฐานข้อมูล 38

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป (ต่อ)

รูป **หน้า**

รูปที่ 3.12 ผังงานโปรแกรมภาษาซีควบคุมไมโครคอนโทรเลอร์ PIC16F628 39
รูปที่ 3.13 บล็อกไดอะแกรมการทำงานของระบบ 40
รูปที่ 3.14 ระบบคอมพิวเตอร์ฝังตัวที่ใช้ 41
รูปที่ 3.15 วงจรอินเตอร์เฟสกับระบบคอมพิวเตอร์ฝังตัว 42
รูปที่ 3.16 วงจรควบคุมวาล์วและปั๊ม 42
รูปที่ 4.1 การล็อกอินเข้าระบบควบคุมปั๊มและวาล์ว 43
รูปที่ 4.2 หน้าจอหลักของการควบคุมปั๊มผ่านทางอินเตอร์เน็ต 44
รูปที่ 4.3 ผลของการสั่งงานวาล์วตามที่ผู้ใช้ได้ทำการสั่งงาน 45
รูปที่ 4.4 การเลือกตั้งเวลาเปิดปิดวาล์วล่วงหน้า 45
รูปที่ 4.5 ฐานข้อมูลการตั้งเวลาเปิดปิดวาล์วล่วงหน้า 46
รูปที่ 4.6 การจำลองการส่งค่าของโครงข่ายเซ็นเซอร์แบบไร้สาย 47
รูปที่ 4.7 ผลที่โครงข่ายเซ็นเซอร์แบบไร้สายร้องขอ 47
รูปที่ 4.8 อุปกรณ์ทั้งหมดของโครงงาน 48

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญตาราง

ตาราง **หน้า**

ตารางที่ 2.1 ขาอินพุต/เอาต์พุตและหน้าที่ต่างๆ 10
ตารางที่ 2.2 รายละเอียดของสายสัญญาณ 22

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 1
บทนำ

1.1 วัตถุประสงค์

1. เพื่อศึกษาระบบคอมพิวเตอร์ฝังตัว (Embedded System) ที่กำลังได้รับความนิยมและน่าสนใจในปัจจุบัน
2. เพื่อพัฒนาระบบคอมพิวเตอร์ฝังตัวให้สามารถใช้งานได้หลากหลายขึ้น
3. เพื่อใช้ในการเกษตรของไทยโดยนำเอาเทคโนโลยีของระบบคอมพิวเตอร์ฝังตัวมาใช้การประยุกต์ให้มีความก้าวหน้าขึ้นไป

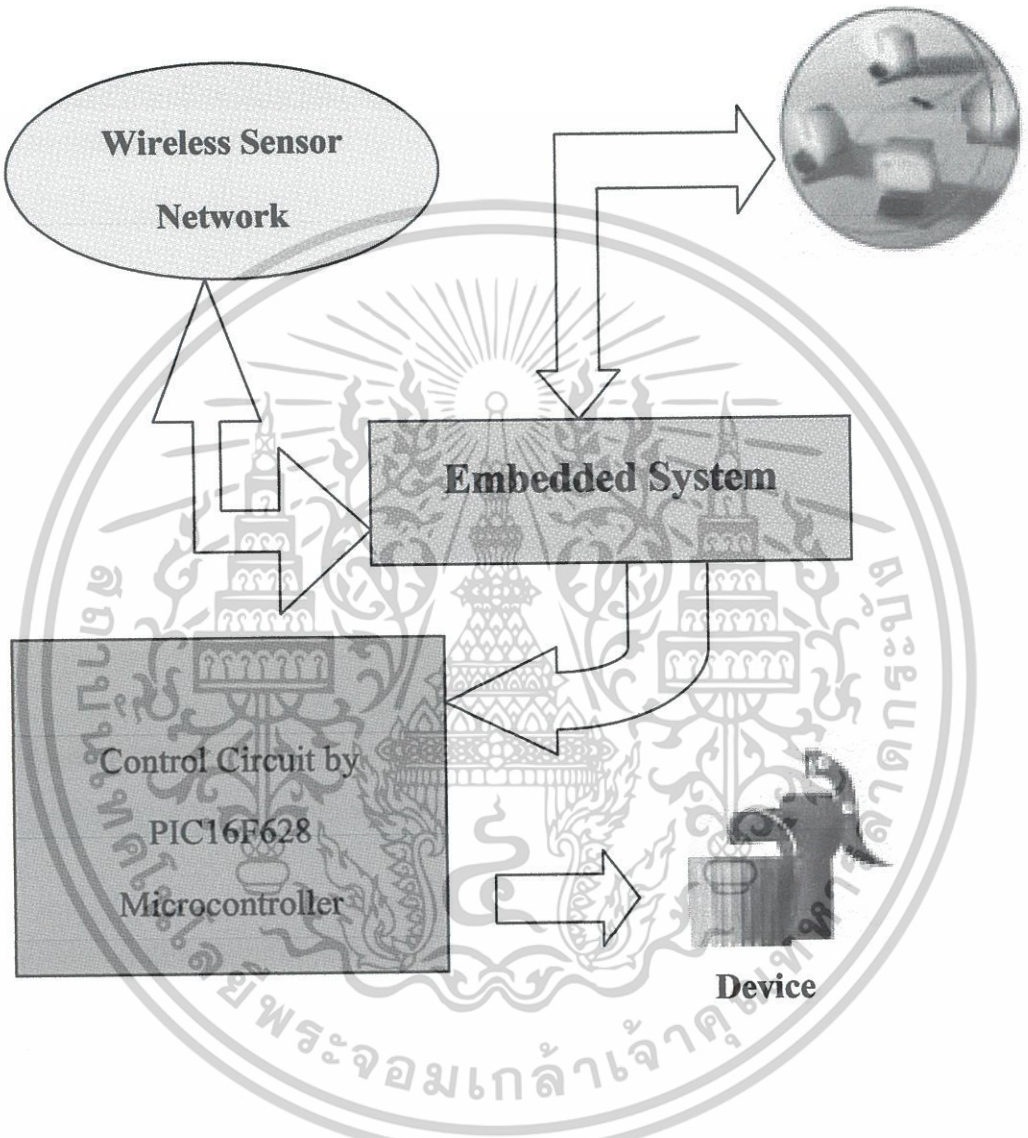
1.2 แนวความคิดและที่มา

ในอดีต การสื่อสารหรือส่งข้อมูลเพื่อไปควบคุมอุปกรณ์ไฟฟ้าหรือควบคุมอุปกรณ์ทางการเกษตรและอุปกรณ์อื่นๆนั้นยังสามารถควบคุมได้ในระยะทางใกล้ๆซึ่งมีจุดด้อยมากมายทั้งในเรื่องความสะดวกในการควบคุม ค่าบำรุงรักษาระบบ ค่าแรงคนงาน และอื่นๆอีกมากมาย และยังมีข้อบกพร่องเรื่องระยะทางอีกด้วย ดังนั้นเราจึงได้คิดหาวิธีต่างๆเพื่อบรรเทาภาระเหล่านั้นมาอย่างต่อเนื่อง และปัจจุบันได้มีอุปกรณ์ที่มนุษย์คิดค้นขึ้นคือคอมพิวเตอร์เมื่อคอมพิวเตอร์เข้ามามีบทบาทในชีวิตประจำวันของมนุษย์มากขึ้นเราจึงมีแนวคิดในการนำเอาคอมพิวเตอร์มาจัดการกับสิ่งเหล่านั้นโดยหนึ่งในนั้นคือระบบคอมพิวเตอร์ฝังตัว

โดยโครงงานนี้เราได้นำระบบคอมพิวเตอร์ฝังตัวมาประยุกต์ใช้งานและด้วยประสิทธิภาพของระบบคอมพิวเตอร์ฝังตัวซึ่งมีขนาดเล็กและใช้กำลังไฟน้อยและสามารถใช้งานเข้ากับเครือข่ายอินเตอร์เน็ทเราจึงได้นำมาใช้ประโยชน์ในการควบคุมอุปกรณ์ในระยะไกลมากๆ ทำให้ได้ประโยชน์และแนวทางการพัฒนาการควบคุมที่ซับซ้อนยิ่งขึ้นไป และลดข้อจำกัดของระบบเก่าได้ด้วย

อีกทั้งยังสามารถรองรับการใช้ระบบโครงข่ายเซ็นเซอร์แบบไร้สาย(Wireless Sensor Network) ซึ่งเป็นระบบที่เป็นโครงข่ายที่ส่งข้อมูลหรือสัญญาณโดยไม่ใช้สายนำสัญญาณเพื่อส่งข้อมูลมาควบคุมอุปกรณ์ทางการเกษตรหรืออุปกรณ์อื่นๆ และยังไม่จำกัดเรื่องระยะทางอีกด้วย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 1.1 สถาปัตยกรรมของระบบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1.3 ขอบเขตของโครงงาน

1. สามารถใช้คอมพิวเตอร์สั่งงานผ่านระบบคอมพิวเตอร์ฝังตัวเพื่อไปควบคุมปั๊มได้
2. สามารถรับข้อมูลจากระบบ เซ็นเซอร์แบบไร้สายได้
3. สามารถตั้งเวลาการสั่งจ่ายน้ำล่วงหน้าได้
4. สามารถเชื่อมต่ออินเตอร์เน็ตได้

1.4 ผลที่คาดว่าจะได้รับ

1. ทำให้ได้รับความรู้ในเรื่องระบบคอมพิวเตอร์ฝังตัวที่ศึกษา
2. ระบบคอมพิวเตอร์ฝังตัวที่พัฒนาขึ้นสามารถใช้งานได้หลากหลายและอีกทั้งยังสามารถประยุกต์ใช้งานในอนาคต
3. ระบบคอมพิวเตอร์ฝังตัวที่พัฒนาขึ้นนี้สามารถจะช่วยลดค่าใช้จ่ายในระยะยาว
4. สามารถกำจัดเรื่องระยะทางในการควบคุมอุปกรณ์ต่างๆได้

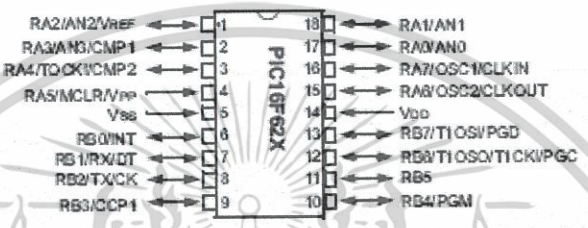
1.5 ประโยชน์ที่ได้

จากเทคโนโลยีนี้จะทำให้ได้ประโยชน์ในหลายๆด้าน อาทิเช่น ความสะดวกในการควบคุม ลดเรื่องจำนวนสายสัญญาณที่ใช้ในการเชื่อมต่อ ลดจำนวนคนในการทำงาน ลดค่าใช้จ่ายในการบำรุงรักษาระบบ ได้ใช้ระบบอินเตอร์เน็ตในการควบคุมจึงสามารถควบคุมจากที่ไหนของโลกก็ได้ ได้ศึกษาเทคโนโลยีใหม่ๆเพื่อนำมาประยุกต์ใช้กับการควบคุมอุปกรณ์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 2
ทฤษฎีและหลักการ

2.1 สถาปัตยกรรมและโครงสร้างทางฮาร์ดแวร์ของ PIC16F628

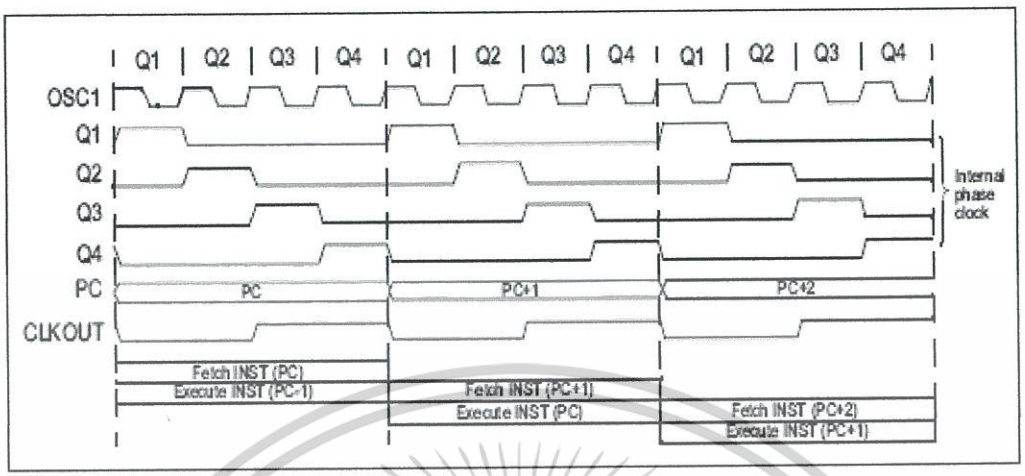


รูปที่ 2.1 การจัดขาของไมโครคอนโทรเลอร์ PIC16F628

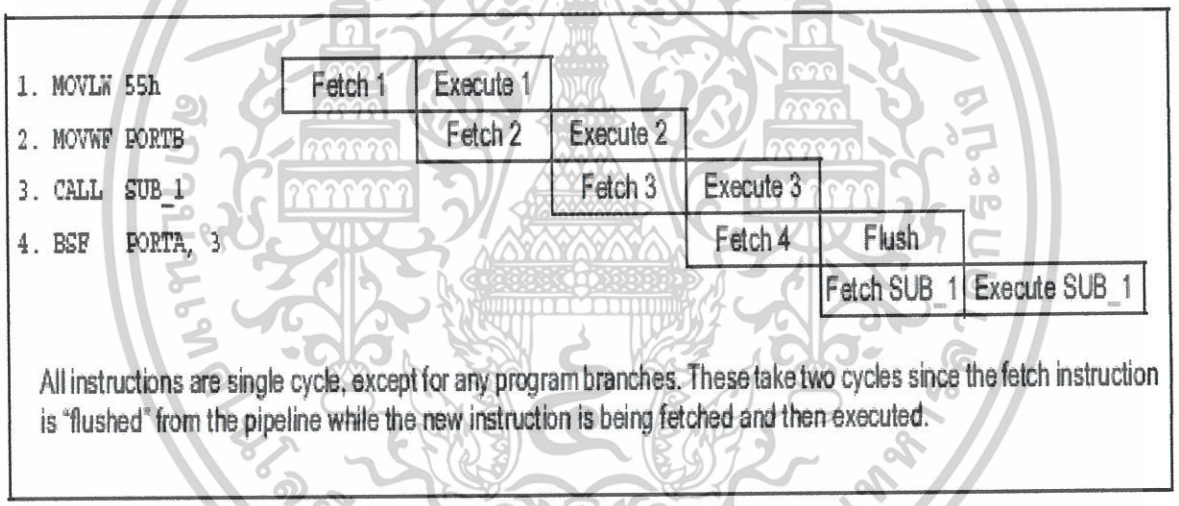
ไมโครคอนโทรลเลอร์ตระกูลPIC เป็นไมโครคอนโทรลเลอร์ที่มีสถาปัตยกรรมแบบฮาร์วาร์ด (Harvard architecture) กล่าวคือ มีการจัดแยกหน่วยความจำโปรแกรมและหน่วยความจำข้อมูลออกจากกัน มีบัสสำหรับติดต่อแยกกัน จะเห็นได้ว่าซีพียูภายในไมโครคอนโทรลเลอร์จะติดต่อกับหน่วยความจำโปรแกรมด้วยบัสของแอดเดรส 13 บิต และบัสของข้อมูลหน่วยความจำโปรแกรม 14 บิต ในขณะที่บัสติดต่อหน่วยความจำข้อมูลและรีจีสเตอร์ภายในเป็นแบบ 8 บิต

นอกจากการจัดสถาปัตยกรรมแบบนี้แล้ว การกระทำคำสั่งทำงานของไมโครคอนโทรลเลอร์ PIC ยังใช้กระบวนการที่เรียกว่า ไปป์ไลน์ (pipeline) ทำให้สามารถเฟตช์คำสั่งถัดไป ในขณะที่กำลังเอ็กซิคิวต์คำสั่งในปัจจุบัน ส่งผลให้ความเร็วในการทำงานของไมโครคอนโทรลเลอร์เพิ่มมากขึ้น นั่นจึงเป็นที่มาของความสามารถในการกระทำคำสั่ง 1 คำสั่งภายในสัญญาณนาฬิกา 1 ลูก กระบวนการเฟตช์(fetch) เป็นกระบวนการเรียกคำสั่งออกจากหน่วยความจำโปรแกรมแล้วทำการแปลคำสั่งนั้นให้เป็นรหัสเลขฐานสิบหกเพื่อให้ซีพียูเข้าใจ ส่วนกระบวนการเอ็กซิคิวต์ (execute) เป็นการกระทำคำสั่งให้เกิดผลลัพธ์ตามที่คำสั่งนั้นๆกำหนด) สำหรับกระบวนการไปป์ไลน์แสดงในรูปที่ 2.3

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.2 กระบวนการเฟชต์และเอ็กซิคิวต์ในรอบการทำงาน



รูปที่ 2.3 กระบวนการไปป์ไลน์

เมื่อเริ่มต้นกระทำคำสั่งที่ 1 ซีพียูจะเฟตช์คำสั่งจากหน่วยความจำโปรแกรมที่แอดเดรส n จากนั้นทำการเอ็กซิคิวต์ในไซเกิลต่อมา และที่ไซเกิลของการเอ็กซิคิวต์คำสั่งที่แอดเดรส n นั้น ซีพียูก็จะเริ่มเฟตช์คำสั่งจากแอดเดรส n+1 ทันที เมื่อเอ็กซิคิวต์คำสั่งที่แอดเดรส n เรียบร้อย ซีพียูก็จะสามารถเอ็กซิคิวต์คำสั่งที่แอดเดรส n+1 ต่อเนื่องกันไปได้ในทันที และในทำนองเดียวกัน ขณะที่กำลังเอ็กซิคิวต์คำสั่งแอดเดรส n+1 ซีพียูก็จะดำเนินการเฟตช์คำสั่งที่แอดเดรส n+2 ต่อไป

แต่ถ้าคำสั่งที่กระทำนั้นเป็นคำสั่งการกระโดด จะมีขั้นตอนที่เพิ่มขึ้นมา เมื่อทำการเอ็กซิคิวต์คำสั่งที่แอดเดรส n ซีพียูก็จะทำการเฟตช์คำสั่งที่แอดเดรส n+1 ปรากฏว่าคำสั่งที่แอดเดรส n+1 นั้นเป็น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

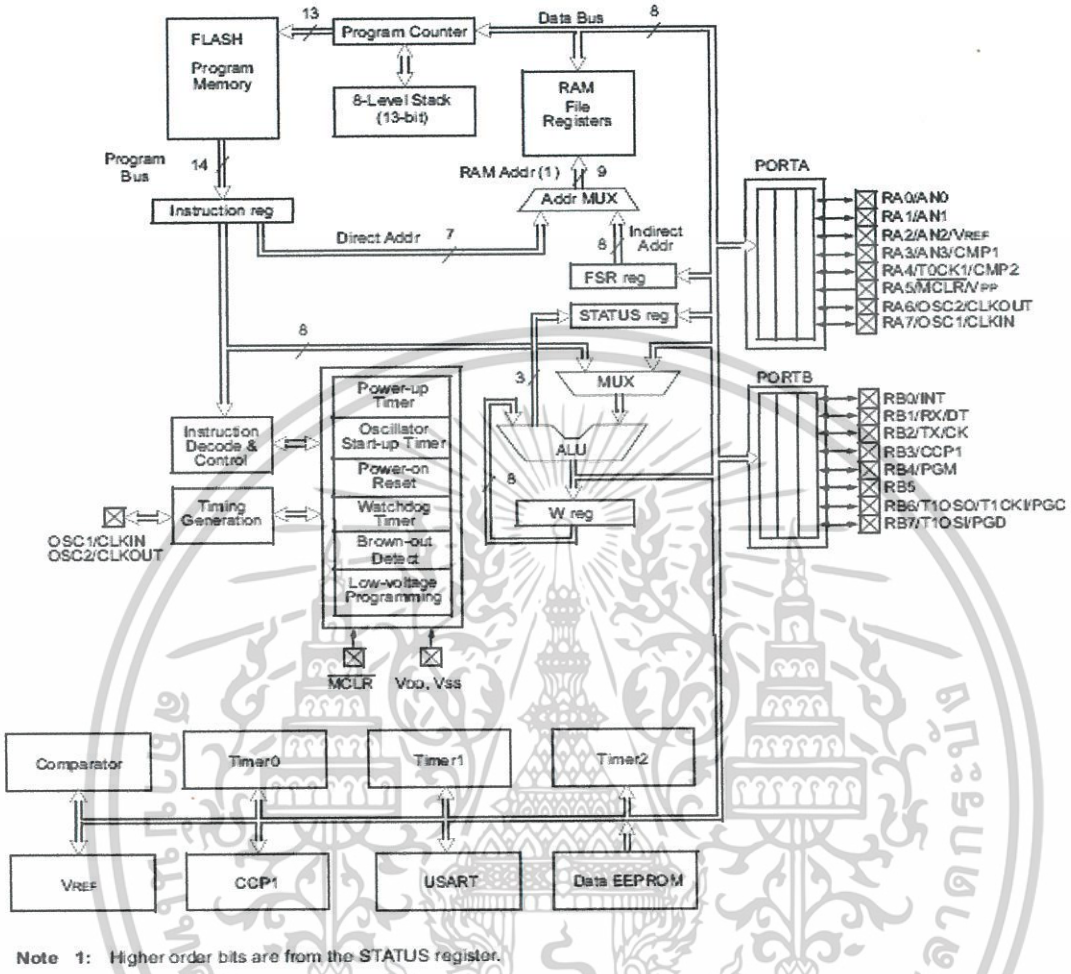
คำสั่งการกระโดด ดังนั้นในไซเกิลถัดไปจึงยังไม่เกิดการเอ็กซิคิวต์ในทันที แต่จะเป็นการเปลี่ยนแปลงค่าของโปรแกรมเคาน์เตอร์ (PC : Program Counter) ซึ่งเป็นรีจิสเตอร์ที่ใช้ในการระบุแอดเดรสถัดไปที่ซีพียูจะไปทำงาน ทำให้เกิดการกระโดดไปยังแอดเดรสใหม่ในระหว่างที่เกิดการกระโดดไปยังแอดเดรสใหม่นั้น ซีพียูจะทำการเฟตช์คำสั่งที่แอดเดรส n+2 ต่อไปตามขั้นตอนปกติแต่ทว่าเมื่อการกระโดดสิ้นสุดลง แอดเดรสของการทำงานเปลี่ยนไป จึงไม่เกิดการเอ็กซิคิวต์คำสั่งที่แอดเดรส n+2 แต่จะเกิดการเฟตช์คำสั่งที่แอดเดรสใหม่ที่กระโดดมาแทน จากนั้นจึงเข้าสู่กระบวนการทำงานตามปกติต่อไป ดังนั้นคำสั่งการกระโดดจึงต้องใช้ขนาดของหน่วยความจำมากกว่า 1 ไบต์ ส่งผลให้ต้องใช้สัญญาณนาฬิกาในการกระทำคำสั่งในกลุ่มนี้2 ลูก ซึ่งแตกต่างจากคำสั่งอื่นๆของไมโครคอนโทรลเลอร์ PIC ที่ใช้สัญญาณนาฬิกาเพียง 1 ลูก

2.1.1 โครงสร้างของไมโครคอนโทรลเลอร์ PIC16F628

แสดงดังในรูปที่ 2.4 ส่วนประกอบหลักก็จะเหมือนกับไมโครคอนโทรลเลอร์ PIC16F84 แต่จะมีส่วนที่เพิ่มเติมเข้ามามากพอควร ได้แก่ วงจรบราวเอาต์รีเซต (brown-out reset) สำหรับสร้างสัญญาณรีเซตซีพียูเมื่อไฟเลี้ยงลดต่ำลงเกินกว่าที่กำหนด, วงจรโปรแกรมข้อมูลด้วยแรงดันต่ำ (lowvoltage programming), ไทเมอร์ที่มีมากถึง3ตัว, โมดูลแรงดันเปรียบเทียบ (reference voltage module), โมดูลเปรียบเทียบแรงดันอนาลอก 2 ชุด (analog comparator), วงจรสื่อสารอนุกรม (USART :Universal Synchronous Asynchronous Receiver Transmitter), และโมดูลตรวจจับสัญญาณ-เปรียบเทียบข้อมูล-วงจรสร้างสัญญาณมอดูเลชั่นทางความกว้างของพัลส์หรือ PWM (CCP: Capture Compare Pulse-width modulation module)

นอกจากนั้นขนาดของหน่วยความจำทั้งส่วนโปรแกรม, ข้อมูล, รีจิสเตอร์ และหน่วยความจำอีอีพรอมในไมโครคอนโทรลเลอร์ PIC61F628 ก็มีเพิ่มมากขึ้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.4 ไดอะแกรมโครงสร้างการทำงานของไมโครคอนโทรเลอร์PIC16F628

2.1.2 คุณสมบัติทางเทคนิคของ PIC16F62X

- ซีพียูเป็นแบบ RISC (Reduce Instruction-Set Computer)
- ความถี่สัญญาณนาฬิกา ตั้งแต่ไฟตรงถึง 20MHz
- ขนาดหน่วยความจำโปรแกรม 2 กิโลเวิร์ด สำหรับ PIC16F628
- หน่วยความจำแรมข้อมูล 224 ไบต์
- หน่วยความจำอีอีพรอม 128 ไบต์
- ตอบสนองแหล่งกำเนิดอินเตอร์รัปต์ได้ 10 แหล่ง
- มีแสตก 8 ระดับ
- มีวงจรเพาเวอร์ออนรีเซต (POR) ,เพาเวอร์อัพไทเมอร์ (PWRT) และออสซิลเลเตอร์สตาร์ทอัพไทเมอร์ (OST)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- มีวอตช์ด็อกไทเมอร์ (WDT) ที่มีวงจรออสซิลเลเตอร์ในตัว ทำให้มีความน่าเชื่อถือในการทำงานสูง

-เลือกป้องกันข้อมูลทั้งในหน่วยความจำโปรแกรมและหน่วยความจำข้อมูล และเลือกระดับป้องกันได้

- เลือกใช้วงจรกำเนิดสัญญาณนาฬิกาได้ 6 โหมดหลัก

1. โหมด EC ใช้สัญญาณนาฬิกาจากภายนอก
2. โหมด ER ใช้ตัวต้านทานภายนอก
3. โหมด INTRC ใช้วงจร RC ภายในไมโครคอนโทรลเลอร์มี2ความถี่ให้เลือก
4. โหมด LP ใช้คริสตอลพลังงานต่ำ ความถี่สูงสุดไม่เกิน 200 kHz
5. โหมด XT ใช้คริสตอล ความถี่ตั้งแต่ 100 kHz สูงสุดไม่เกิน 4MHz
6. โหมด HS ใช้คริสตอลความถี่สูง สูงสุดไม่เกิน 20 MHz (ต้องใช้กับรุ่นที่รองรับความถี่ 20 MHz ด้วย)

- สามารถโปรแกรมโดยใช้แรงดัน +5V ได้
- ไฟเลี้ยง +3 ถึง +5.5 V
- สามารถโปรแกรมในวงจรได้
- กระแสซิงก์และซอร์สของพอร์ต 25 mA
- ขาพอร์ตปกติ 15 บิต สูงสุด 16 บิต เมื่อทำงานในโหมด INTRC และกำหนดให้ MCLR เป็นพอร์ตอินพุต
- ไทเมอร์ 3 ตัว (ไทเมอร์ 0,ไทเมอร์ 1, ไทเมอร์ 2)
- มีโมดูล CCP (Capture/Compare/PWM) 1ชุด
- มีโมดูลเปรียบเทียบแรงดันอะนาลอก 2 ชุด
- มีโมดูลสร้างแรงดันอ้างอิง
- มีโมดูลสื่อสารข้อมูลอนุกรม USART
- มีวงจรตรวจจับระดับแรงดันไฟเลี้ยงหรือบราวเอาต์ดีเท็กชั่น (Brown-out detection) เพื่อสร้างสัญญาณรีเซตซีพียูหรือเรียกว่า บราวเอาต์รีเซต (Brown-out reset :BOR)
- การใช้พลังงานไฟฟ้าในกรณีไม่ขับโหลด

น้อยกว่า 2 mA ที่ +5V และสัญญาณนาฬิกา 4MHz, 15μA ที่ +3V และสัญญาณนาฬิกา 32 kHz

น้อยกว่า 1μA ในโหมดประหยัดพลังงานหรือสแตนด์บายที่ไฟเลี้ยง +3V

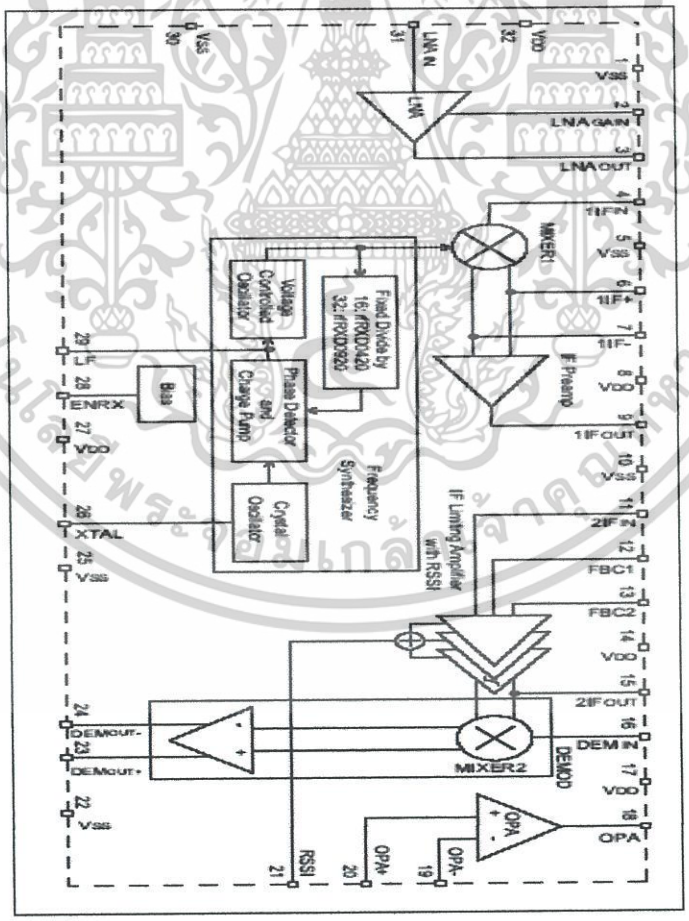
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.2 เครื่องรับ rfRXDO420 UHF ASK/FSK/FM

rfRXD0420 มีค่าใช้จ่ายต่ำ,เป็นอุปกรณ์รับคลื่นวิทยุระยะสั้นที่กะทัดรัด ต้องการอุปกรณ์ภายนอกน้อยเพื่อใช้ในการรับอย่างมีประสิทธิภาพ, rfRXD0420ครอบคลุมความถี่ในช่วง 300-400MHz สามารถกำหนดให้รองรับ Amplitude Shift Keying(ASK),Frequency Shift Keying(FSK), FM Modulation หรือRF Transmitterได้

2.2.1 คุณสมบัติที่สำคัญ

- มีความเสถียรสูงในอุณหภูมิและแรงดันแปรปรวน
- มีความผิดพลาดของสัญญาณต่ำ
- สามารถเลือกความถี่ IF ได้ในช่วง 455kHz -21.4MHzโดยใช้ เซรามิก ไอเอฟ ฟิลเตอร์
- สามารถรับดิจิตอลดาต้า ASK,FSK



รูปที่ 2.5 บล็อกไดอะแกรมของ rfRXD0420

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 2.1 ขาอินพุต/เอาต์พุตและหน้าที่ต่างๆ

Pin Name	Pin Number	Pin Type	Buffer Type	Description
LNAGAIN	2	I	CMOS	LNA gain control (with hysteresis)
LNAOUT	3	O	Analog	LNA output (open collector)
1IFIN	4	I	Analog	1st IF stage input
1IF+	6	--	Analog	MIXER1 bias (open collector)
1IF-	7	--	Analog	MIXER1 bias (open collector)
1IFOUT	9	O	Analog	1st IF stage output
2IFIN	11	I	Analog	2nd IF stage input
FBC1	12	--	Analog	Limiter IF Amplifier external feedback capacitor
FBC2	13	--	Analog	Limiter IF Amplifier external feedback capacitor
2IFOUT	15	O	Analog	2nd IF stage output
DEMIN	16	I	Analog	Demodulator input
OPA	18	O	Analog	Operational amplifier output
OPA-	19	I	Analog	Operational amplifier input (negative)
OPA+	20	I	Analog	Operational amplifier input (positive)
RSSI	21	O	Analog	Received signal strength indicator output
DEMOUT+	23	O	Analog	Demodulator output (positive)
DEMOUT-	24	O	Analog	Demodulator output (negative)
XTAL	26	I	Analog	Crystal oscillator input
ENRX	28	I	CMOS	Receiver enable input
LF	29	I	Analog	External loop filter connection. Common node of charge pump output and VCO tuning input.
LNAIN	31	I	Analog	LNA input
VDD	8, 14, 17, 27, 32	P		Positive supply
VSS	1, 5, 10, 25, 30	P		Ground reference

Legend: I = Input, O = Output, I/O = Input/Output, P = Power, CMOS = CMOS compatible input or output

2.3 ระบบคอมพิวเตอร์ฝังตัว (Embedded System)

2.3.1 คอมพิวเตอร์ล่องหน

คอมพิวเตอร์ฝังตัวหรือ Embedded System คือระบบอิเล็กทรอนิกส์ที่ประมวลผลด้วยซีพียู แต่จะต่างจากที่ใช้กับเครื่องคอมพิวเตอร์ตั้งโต๊ะ โน๊ตบุ๊ค หรือเครื่องเวิร์คสเตชั่น โดยที่คอมพิวเตอร์ฝังตัวมักจะใช้ชิพที่ออกแบบมาเฉพาะมากกว่า ที่ผ่านมาคอมพิวเตอร์ฝังตัวได้ถูกนำมาใช้กันอย่างแพร่หลายในรถยนต์ เครื่องบิน รถไฟ ยานอวกาศ กล้องถ่ายรูป อุปกรณ์เครื่องใช้ไฟฟ้าในบ้านและสำนักงาน และที่คนจำนวนมากใช้งานกันอยู่ทุกวันโดยไม่รู้ตัวก็คือ โทรศัพท์มือถือ รวมถึงพวกพีดีเอ ตลอดจนของเล่นอิเล็กทรอนิกส์ต่างๆ

คอมพิวเตอร์ฝังตัว หรือล่องหนนี้คือ อุปกรณ์ที่เป็นอิเล็กทรอนิกส์ต่างๆ ไม่ว่าจะเป็นอุปกรณ์คอมพิวเตอร์ โทรคมนาคม เครื่องใช้ไฟฟ้า ที่มีเจ้าตัวไมโครชิพที่มีการเขียนโปรแกรมใส่เข้า

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ไป ไมโครชิพที่ว่านี้มีหลายรูปแบบ เป็นทั้งแบบไม่ต้องมีโปรแกรมทำได้เลย กับแบบที่ต้องเขียนโปรแกรมเข้าไป คอมพิวเตอร์ล่องหนนี้ หมายถึงระบบที่มีไมโครชิพทำหน้าที่ควบคุมอยู่ และการควบคุมนั้น เป็นการควบคุมโดยการเขียนโปรแกรมฝังเข้าไปอุปกรณ์ที่มีระบบคอมพิวเตอร์ฝังตัวหรือล่องหนอยู่ที่เห็นได้ชัดที่สุดคือ โทรศัพท์มือถือ ภายในประกอบด้วยบอร์ดวงจร และหน่วยความจำ ที่สำคัญที่สุดคือ ไมโครโปรเซสเซอร์ ซึ่งผู้ผลิตเอสพีวีนำมาใช้เป็นไมโครชิพของ ARM เวอร์ชั่น 720 เช่นเดียวกับ พีดีเอ iPaq ซึ่งรุ่นก่อนหน้านี้ก็ใช้ไมโครชิพของ ARM รุ่น SA1110 และเพิ่งจะเปลี่ยนมาใช้โปรเซสเซอร์ Xscale ของอินเทล ซึ่งประหยัดแบตเตอรี่มากกว่าและมีความเร็วสูงกว่า โทรศัพท์มือถือทั้งหมดเป็น embedded system นั่นคือ เดิมเราเห็นคอมพิวเตอร์เป็นคอมพิวเตอร์ ถ้าจะให้ทำงานอะไร ก็จะใส่โปรแกรมเข้าไป คอมพิวเตอร์ก็จะทำงาน ถ้าเป็น Embeded System จะมองไม่เห็นการใส่โปรแกรม เวลาจะใช้ก็สามารถกดใช้งานโดยตรงได้เลย เหมือนกับการใช้เครื่องเล่นวิดีโอ นั่นคือไม่ต้องโหลด หรือเรียกโปรแกรมเวลาต้องการใช้ก็สามารถกดปุ่มสั่งให้เครื่องทำงานได้

หลายคนอาจติดภาพเครื่องคอมพิวเตอร์ที่มีจอภาพ มีฮาร์ดดิสก์ ซีพียู มีแผงหน่วยความจำ RAM มีเมาส์ ฯลฯ เป็นส่วนประกอบ แต่ในอุปกรณ์คอมพิวเตอร์ฝังตัว ซอฟต์แวร์ต่างๆ จะถูกฝังลงในหน่วยความจำชนิดอ่านได้อย่างเดียวหรือที่เรียกว่า ROM (Read Only Memory) หรือพวก flash memory chip การเรียกใช้โปรแกรมจึงทำได้อย่างรวดเร็ว ต่างจากคอมพิวเตอร์ส่วนบุคคล หรือพีซีที่โปรแกรมจะถูกเรียกมาไว้ที่หน่วยความจำ RAM (Random Access Memory) ทุกครั้งจะเห็นได้ว่าคอมพิวเตอร์ฝังตัวสามารถนำไปใช้งานได้ไม่จำกัด โดยในแต่ละปีไมโครโปรเซสเซอร์นับพันล้านจะถูกผลิตออกมาป้อนสู่ตลาดเพื่อประกอบในอุปกรณ์อิเล็กทรอนิกส์ต่างๆ เครื่องใช้ไฟฟ้าบางตัวสามารถใช้ชิพที่มีราคาไม่ถึงห้าสิบบาท โดยไม่จำเป็นต้องมีความฉลาดหาศาล หรือความเร็วในการประมวลผลระดับสูง อุปกรณ์ที่ใช้คอมพิวเตอร์ฝังตัวนี้จึงเป็นอุปกรณ์ที่ใช้ชิพขนาดเล็ก กินไฟน้อย มีระบบปฏิบัติการและโปรแกรมประยุกต์อยู่ในตัว

ไมโครชิพสามารถทำงานได้หลายอย่าง และเมื่อตัวไมโครชิพและหน่วยความจำมีขนาดเล็กลง และราคาถูกลง แถมมีความสามารถสูง จึงสามารถเอาไปใส่ในอุปกรณ์ขนาดเล็กได้ เมื่อมีความสามารถสูง มันก็สามารถเขียนโปรแกรมที่มีความซับซ้อนใส่เข้าไปได้ เพราะฉะนั้นโทรศัพท์มือถือบางเครื่องจึงมีความสามารถเพิ่มขึ้นมากกว่าแต่ก่อน ชิพที่มีขนาดเล็กลงนี้ทำให้ใช้พลังงานไฟฟ้าน้อย และสามารถใช้งานเคลื่อนที่ไปไหนมาไหนได้ โดยไม่ต้องมีสายระโยงระยาง

ข้อดีของคอมพิวเตอร์ล่องหนก็คือ สมองกลฝังตัวนี้มีย่านการใช้งานได้หลากหลายมากขึ้นอยู่กับขีดความสามารถของซีพียู ยกตัวอย่าง ไมโครโปรเซสเซอร์ที่ใช้ในคอมพิวเตอร์ทั่วไปปัจจุบันจะอยู่ที่ 32 บิต แต่สำหรับสมองกลฝังตัวนี้จะมีตั้งแต่ 4 บิต 8 บิต 16 บิต และ 32 บิต หรือ 64

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บิต ขึ้นอยู่กับลักษณะการนำไปใช้งาน ถ้าใช้ในเครื่องซักผ้าอาจจะใช้แค่ 8 บิต หรือ 4 บิต หรือถ้าใช้กับโทรศัพท์มือถือก็ต้อง 32 บิต หรือถ้าใช้ในการควบคุมเครื่องจักรอาจจะเป็น 16 หรือ 32 บิต เป็นต้น

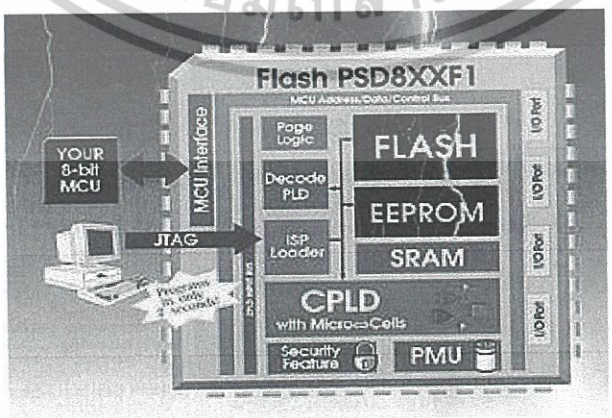
2.3.2 การใช้งาน

ในปัจจุบันเทคโนโลยีของระบบฝังตัว หรือ Embedded System นั้นกำลังเป็นที่จับตามองเป็นอย่างมาก ทั้งนี้เนื่องจากอุปกรณ์ไฟฟ้าต่างๆ ได้เริ่มมีการเพิ่มหน่วยประมวลผลเข้าไปเพื่อให้ผลิตภัณฑ์มีคุณสมบัติที่แตกต่างหรือเหนือกว่าคู่แข่ง ไม่ว่าจะเป็นโทรทัศน์ วิทยุ เครื่องซักผ้า หม้อหุงข้าว หรือแม้แต่รถยนต์เองก็มีการนำเอาหน่วยประมวลผลเข้าไปช่วยเพิ่มประสิทธิภาพในการทำงานด้วยเช่นกัน

นอกเหนือจากมุมมองจากทางด้านการใช้งานแล้ว เทคโนโลยีของหน่วยประมวลผลเองนั้นก็ได้รับการพัฒนาให้ก้าวหน้าอยู่ตลอดเวลา ในปัจจุบันเรามีไมโครโพรเซสเซอร์และไมโครคอนโทรลเลอร์ที่มีขนาดเล็ก และกินไฟน้อยออกมาให้เลือกใช้งานเป็นจำนวนมาก ดังนั้นจึงไม่น่าแปลกใจที่เราจะเห็นอุปกรณ์อิเล็กทรอนิกส์ที่มีขนาดเล็กลง แต่สามารถใช้พลังงานจากแบตเตอรี่ขนาดเล็กเพียงไม่กี่ก้อนเท่านั้น

ถ้าจะกล่าวถึงการพัฒนาในเทคโนโลยีของระบบฝังตัวแล้ว โดยส่วนใหญ่ผู้พัฒนาจะให้ความสำคัญกับไมโครคอนโทรลเลอร์ และไมโครโพรเซสเซอร์เป็นหลัก ทั้งนี้เนื่องจากเป็นส่วนที่มีความสำคัญกับการประมวลผลและคุณสมบัติของระบบฝังตัว และเช่นเดียวกันการพัฒนาเทคโนโลยีของไมโครโพรเซสเซอร์ก็มีผลต่อระบบฝังตัวที่มีอยู่ในตลาดด้วยเช่นกัน

2.3.3 การแบ่งระดับของระบบฝังตัว



รูปที่ 2.6 โครงสร้างคอมพิวเตอร์ฝังตัว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สำหรับเทคโนโลยีของระบบฝังตัว เราสามารถแบ่งออกได้เป็น 3 ระดับตามความซับซ้อนในการประมวลผลและลักษณะการใช้งาน คือ ระบบฝังตัวขนาดเล็ก เป็นระบบฝังตัวที่ใช้ไมโครคอนโทรลเลอร์สำหรับควบคุมอุปกรณ์อิเล็กทรอนิกส์พื้นฐานที่มีการทำงานไม่ซับซ้อนนัก ระบบฝังตัวขนาดกลางที่ใช้ไมโครโพรเซสเซอร์ที่มีความสามารถในการทำงานสูงขึ้นมา และระบบฝังตัวขนาดใหญ่ที่มีความสามารถสูงเหมาะสำหรับการใช้งานที่ต้องการความสามารถในการประมวลผลมากเป็นพิเศษ

สำหรับระบบฝังตัวขนาดเล็กที่ใช้ไมโครคอนโทรลเลอร์และไมโครคอนโทรลเลอร์ขนาดเล็กนั้นเหมาะสำหรับใช้ในการควบคุมอุปกรณ์อิเล็กทรอนิกส์พื้นฐานที่มีการทำงานไม่ซับซ้อนนัก โดยส่วนใหญ่มักจะเป็นไมโครคอนโทรลเลอร์ที่มีหน่วยความจำ ส่วนติดต่อกับอินพุทและเอาท์พุท รวมทั้งส่วนประกอบที่จำเป็นรวมอยู่ภายในเสร็จสรรพ ทั้งนี้ก็เพื่อความง่ายต่อการพัฒนาวงจร ไมโครคอนโทรลเลอร์เหล่านี้มักจะมีขนาดเล็ก (4 หรือ 8 บิต) มีหน่วยความจำภายในไม่มาก (ประมาณ 10-120 กิโลไบต์) มีพอร์ตอินพุทเอาท์พุทตั้งแต่ 1-4 พอร์ต สามารถติดต่อกับส่วนสื่อสารอนุกรมหรือขนานได้ทันที ตัวอย่างของไมโครคอนโทรลเลอร์เหล่านี้ได้แก่ MSC-51, PIC และ Z80 เป็นต้น

ส่วนระบบฝังตัวขนาดกลางที่ใช้ไมโครโพรเซสเซอร์ที่มีความสามารถในการทำงานสูงขึ้นมานั้นส่วนใหญ่จะถูกนำไปใช้งานกับอุปกรณ์อิเล็กทรอนิกส์ที่ต้องการคุณสมบัติพิเศษที่ไมโครคอนโทรลเลอร์หรือไมโครโพรเซสเซอร์ขนาดเล็กไม่สามารถให้ได้ ดังเช่น ต้องการหน่วยความจำที่มีขนาดมากขึ้น ต้องการความเร็วในการทำงานมากขึ้น หรือต้องการสื่อสารกับอุปกรณ์อื่น เป็นต้น ไมโครโพรเซสเซอร์เหล่านี้มักมีขนาด 16 บิต และ 32 บิต ดังเช่น ตระกูล x86 ของบริษัท Intel และ AMD ตระกูล ARM7 และ ตระกูล TMS320 เป็นต้น ไมโครโพรเซสเซอร์เหล่านี้มักมีคุณสมบัติพิเศษเฉพาะทางเช่น การสื่อสาร หรือการประมวลผลชนิดพิเศษ

และระบบฝังตัวขนาดใหญ่ที่มีความสามารถในการประมวลผลมากเป็นพิเศษ ส่วนใหญ่จะเป็นระบบฝังตัวที่ใช้ไมโครโพรเซสเซอร์ที่ได้รับการออกแบบมาสำหรับประมวลผลเป็นหลัก หรืออาจจะเป็นระบบฝังตัวที่ใช้คอมพิวเตอร์เป็นพื้นฐาน (PC-Based) ระบบฝังตัวประเภทนี้ส่วนใหญ่จะใช้ฮาร์ดแวร์และซอฟต์แวร์ที่มีใช้งานทั่วไป ดังเช่น ไมโครโพรวนใหญ่แล้วระบบฝังตัวประเภทนี้มักจะนำไปประยุกต์ใช้งานกับอุปกรณ์สำหรับระบบเครือข่าย ดังเช่น เราท์เตอร์ อินเทอร์เน็ต หรือเว็บเซิร์ฟเวอร์ เป็นต้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.3.4 การเลือกใช้งาน

เนื่องด้วยการแบ่งระดับของระบบฝังตัวตามความสามารถในการประมวลผลและการนำไปใช้งาน ทำให้ผู้พัฒนามีหลักในการเลือกได้ดียิ่งขึ้น ยกตัวอย่างเช่น ถ้านักพัฒนาต้องการพัฒนาระบบประมวลผลสำหรับการควบคุมเครื่องใช้ไฟฟ้าพื้นฐานภายในบ้าน ดังเช่น พัดลม เตาอบไมโครเวฟ หรือเครื่องซักผ้าเป็นต้น การเลือกใช้ระบบฝังตัวขนาดเล็กที่ใช้ไมโครคอนโทรลเลอร์ขนาดเล็กก็เป็นสิ่งที่สมเหตุสมผล ในด้านของอุตสาหกรรมการใช้ไมโครคอนโทรลเลอร์ขนาดเล็กเพื่อพัฒนาหุ่นยนต์พื้นฐานก็ยังเป็นสิ่งที่ทำได้

แต่สำหรับงานที่ต้องการความสามารถในการประมวลผลสูงขึ้น หรืองานที่มีความจำเป็นในการเชื่อมต่อกับอุปกรณ์ต่อพ่วงที่ซับซ้อนขึ้น การเลือกใช้ระบบฝังตัวระดับกลางเป็นสิ่งที่เหมาะสม ตัวอย่างเช่น การพัฒนาหุ่นยนต์เพื่อใช้ในงานอุตสาหกรรมซึ่งมีการทำงานที่ซับซ้อนหรือที่ต้องมีการทำงานตามเวลาจริง (Real-Time) ระบบฝังตัวที่มีความสามารถสูงขึ้น สามารถพัฒนาและแก้ไขโปรแกรมควบคุมได้ง่าย จะช่วยอำนวยความสะดวกในการพัฒนาได้มากกว่า

และสุดท้ายสำหรับการพัฒนาผลิตภัณฑ์ที่เน้นความสามารถทางด้านการประมวลผลโดยเฉพาะ ตัวอย่างเช่น อุปกรณ์สำหรับระบบเครือข่าย เราท์เตอร์ หรือเว็บเซิร์ฟเวอร์ขนาดเล็ก อุปกรณ์เหล่านี้จำเป็นต้องใช้ระบบฝังตัวที่มีความสามารถสูงในการประมวลผลเพื่อที่สามารถตอบสนองความต้องการในการทำงานของผู้ใช้ได้ หรืออาจจะเป็นคอมพิวเตอร์พีซีที่ได้รับการปรับเปลี่ยนรูปร่างหน้าตาภายนอกเพื่อให้เหมาะสมกับการใช้งานก็เป็นไปได้เช่นกัน

นอกจากการเลือกใช้ระบบฝังตัวตามลักษณะงานและความต้องการทางด้านคุณสมบัติแล้ว สิ่งที่นักพัฒนาจะต้องคำนึงถึงก่อนที่จะเลือกก็คือ เครื่องมือ และข้อมูลที่มีให้ในการพัฒนา ปัจจัยเหล่านี้นับเป็นความเสี่ยงที่มีความสำคัญต่อการเลือกระบบฝังตัวไปใช้งาน เนื่องจากการมีเครื่องมือที่ดี และการมีข้อมูลสนับสนุนที่เพียงพอสามารถช่วยให้การพัฒนาดำเนินไปได้อย่างมีประสิทธิภาพ

2.3.5 ขีดจำกัดของเทคโนโลยี

ถ้าจะกล่าวถึงไมโครคอนโทรลเลอร์และไมโครโพรเซสเซอร์สำหรับระบบฝังตัวที่มีความนิยมที่สุดในประเทศไทยนั้นก็คงจะเป็นไมโครคอนโทรลเลอร์ขนาดเล็ก นักพัฒนามีทางเลือกสำหรับไมโครคอนโทรลเลอร์ขนาดเล็กเหล่านี้อย่างหลากหลายไม่ว่าจะเป็น Atmel MCS50, Microchip PIC 16c64, Motorola 68HC11 หรือแม้แต่ไมโครโพรเซสเซอร์ขนาดเล็กอย่าง Z80 เองก็ตาม ไมโครคอนโทรลเลอร์ขนาดเล็กเหล่านี้มีการใช้งานในประเทศมานานแล้ว ทำให้มี

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เครื่องมือ ข้อมูลประกอบการพัฒนา และผู้เชี่ยวชาญจำนวนมาก เครื่องมือสำหรับการพัฒนาซอฟต์แวร์บนไมโครคอนโทรลเลอร์ขนาดเล็กเหล่านี้มีตั้งแต่ คอมไพเลอร์สำหรับภาษาระดับสูงอย่าง C และ ดีบักเกอร์ ข้อมูลก็มีหนังสือและเอกสารที่ตีพิมพ์ออกมาโดยสำนักพิมพ์ต่างๆ และสำหรับผู้เชี่ยวชาญก็นับได้ตั้งแต่อาจารย์ที่สอนอยู่ในสถาบันหรือมหาวิทยาลัย ไปจนถึงวิศวกรและช่างเทคนิคที่ใช้งานไมโครคอนเลอร์เหล่านี้อยู่

เนื่องด้วยสิ่งสนับสนุนในการใช้งานไมโครคอนโทรลเลอร์และไมโครโพรเซสเซอร์ที่มีอยู่เป็นจำนวนมากนี้ทำให้ผู้คนต่างๆในหลายๆสาขาหันมาให้ความสนใจกับไมโครคอนโทรลเลอร์ขนาดเล็กเป็นอย่างมาก แต่ปัญหาที่กำลังเกิดขึ้นก็คือ ความสามารถของผลิตภัณฑ์ที่พัฒนาขึ้นนั้นกำลังถูกจำกัดอยู่ด้วยความสามารถของไมโครคอนโทรลเลอร์หรือไมโครโพรเซสเซอร์ที่ใช้

ยกตัวอย่างเช่นในการพัฒนาหุ่นยนต์ในสถานศึกษาต่างๆ นั้นจะเห็นได้ว่าหุ่นยนต์เหล่านี้ได้รับการพัฒนาและปรับปรุงมาเป็นเวลานาน แต่สิ่งที่เกิดขึ้นก็คือ หุ่นยนต์ที่นักศึกษาได้พัฒนาขึ้นในวันนี้มีคุณสมบัติส่วนใหญ่เหมือนกับหุ่นยนต์ที่ได้รับการพัฒนาขึ้นเมื่อ 5 ถึง 10 ปีที่แล้ว แต่จะแตกต่างกันที่เวลาที่ใช้ในการพัฒนา ในอดีตการพัฒนาระบบประมวลผลสำหรับหุ่นยนต์อาจจะใช้เวลานานเป็นเดือน เนื่องด้วยข้อจำกัดของเครื่องมือและแหล่งความรู้ แต่ในวันนี้นักศึกษาสามารถพัฒนาหุ่นยนต์ที่มีคุณสมบัติเดียวกันได้ภายในเวลาไม่กี่สัปดาห์เท่านั้น

จากสิ่งที่เกิดขึ้นเมื่อได้ย้อนกลับไปพิจารณาถึงต้นเหตุของปัญหาแล้วก็จะเข้าใจได้ไม่ยากเลย เนื่องจากใช้ไมโครคอนโทรลเลอร์ขนาดเล็กซึ่งพร้อมด้วยเครื่องมือในการพัฒนาและข้อมูลประกอบ แต่ไมโครคอนโทรลเลอร์ขนาดเล็กเหล่านี้มีข้อจำกัดค่อนข้างมาก จึงไม่สามารถพัฒนาให้หุ่นยนต์ของตนมีความฉลาดหรือพัฒนาให้หุ่นยนต์สามารถทำงานได้หลากหลายมากขึ้น เนื่องจากหน่วยความจำที่ไมโครคอนโทรลเลอร์ขนาดเล็กเหล่านี้มีให้นั้นไม่เพียงพอ โดยเฉพาะอย่างยิ่งเมื่อใช้เครื่องมือในการพัฒนาด้วยภาษาระดับสูงซึ่งกินพื้นที่ในหน่วยความจำมาก ถ้าจะหันกลับไปใช้ภาษาระดับล่างอย่างแอสแซมบลี้ก็ทำให้การเขียนโปรแกรมทำได้ยาก ส่งผลให้หุ่นยนต์ไม่สามารถทำงานที่มีความซับซ้อนได้. หรือแม้แต่การใช้งานในโรงงานอุตสาหกรรมเอง เครื่องจักรที่ใช้ไมโครคอนโทรลเลอร์หรือไมโครโพรเซสเซอร์ขนาดเล็กมักจะมีคุณสมบัติและฟังก์ชันการทำงานที่ไม่สูงมาก เครื่องจักรเหล่านี้ส่วนใหญ่จะไม่มีความสามารถในการสื่อสารข้อมูลเป็นระบบเครือข่าย ซึ่งจะช่วยให้การควบคุม และการดูแลสามารถทำได้ง่ายขึ้น และถ้าวิศวกรต้องการพัฒนาให้เครื่องจักรสามารถทำงานที่มีความซับซ้อนมากขึ้น มีความฉลาดมากขึ้น หรือสามารถเชื่อมต่อกับอุปกรณ์อื่นๆ ผ่านระบบเครือข่ายอย่าง SCADA (Supervisory

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Control And Data Acquisition) แล้ว ก็มีความจำเป็นต้องใช้ระบบฝังตัวที่มีความสามารถสูงกว่าไมโครคอนโทรลเลอร์ 8051 หรือไมโครโพรเซสเซอร์ Z80

2.4 ทฤษฎีการสื่อสารอนุกรม

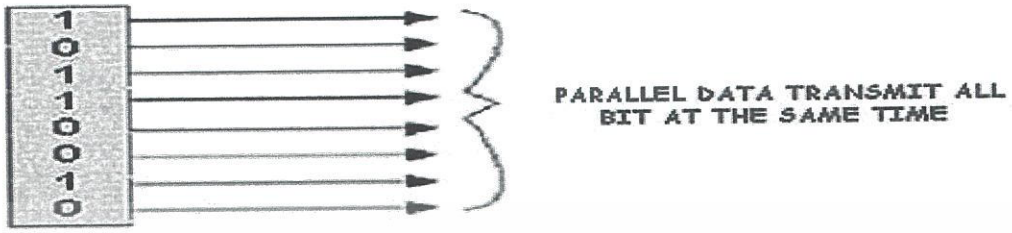
การรับส่งข้อมูลผ่านพอร์ทเครื่องพิมพ์ อุปกรณ์นี้ใช้หลักการเดียวกันเช่นการอ่านหรือเขียนข้อมูลลงในหน่วยความจำ เป็นการรับคำสั่งข้อมูลพร้อมกันทีละหลายบิต เรียกการรับส่งข้อมูลในลักษณะนี้ว่า "การรับส่งข้อมูลแบบขนาน (PARALLEL COMMUNICATION)" ในหัวข้อนี้จะกล่าวถึงการรับส่งข้อมูลในอีกรูปแบบหนึ่ง ซึ่งเป็นการรับส่งข้อมูลทีละบิต แทนที่จะทำการส่งข้อมูลพร้อมกันทุกบิตในเวลาเดียวกัน การรับส่งข้อมูลแบบนี้มีชื่อว่า "การรับส่งข้อมูลแบบอนุกรม (SERIAL COMMUNICATION)"

สำหรับการรับส่งข้อมูลแบบขนานนั้น ถึงแม้ว่าจะมีความเร็วสูงกว่าแบบอนุกรมอยู่มากก็ตามแต่ก็ต้องใช้จำนวนสายในการส่งผ่านข้อมูลเป็นจำนวนมากกว่าแบบอนุกรม ทำให้สิ้นเปลืองค่าใช้จ่ายในการวางสายไปโดยไม่จำเป็น และยังมีการลดทอนของสัญญาณมากกว่าแบบอนุกรมอีกด้วย ทำให้เกิดความผิดพลาดในการส่งผ่านข้อมูลได้ง่ายดังนั้นในการส่งผ่านข้อมูลในระยะทางไกล ๆ เรามักจะเลือกใช้การรับส่งข้อมูลแบบอนุกรม เพื่อลดจำนวนของสายส่งซึ่งจะช่วยในการลดค่าใช้จ่ายในการวางสายลงได้มากถึงแม้ว่าการรับส่งแบบนี้จะมีความยุ่งยากและช้ากว่าการรับส่งข้อมูลแบบขนานอยู่บ้างก็ตาม

รูปที่ 2.7 บิตต่าง ๆ ของข้อมูลที่จะทำการส่งแบบอนุกรม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

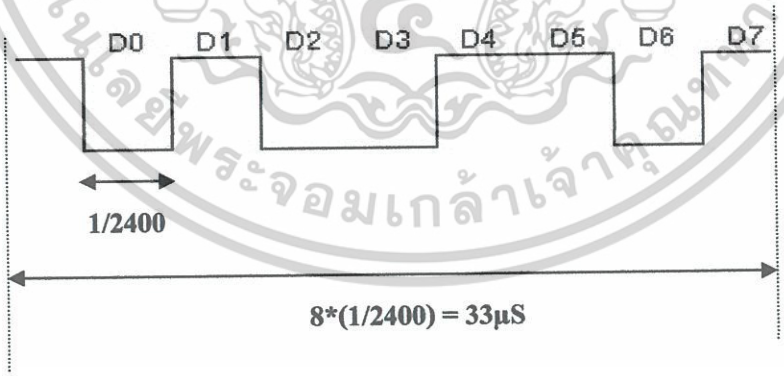
สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง



รูปที่ 2.8 บิตต่าง ๆ ของข้อมูลที่จะทำการส่งแบบขนาน

2.4.1 BAUD RATE

สิ่งที่สำคัญมากสิ่งหนึ่งในการรับส่งข้อมูลแบบอนุกรมก็คือ ความถี่ที่ใช้ในการส่งข้อมูลที่จะต้องสัมพันธ์กันระหว่างอุปกรณ์ที่ทำการรับและส่งข้อมูล และความถี่ที่ใช้นี้เรียกว่า "BAUD RATE" ซึ่งหมายถึง "อัตราการรับส่งข้อมูลเป็นจำนวนบิตใน 1 วินาที" ถ้าหากว่าเครื่องส่งใช้ BAUD RATE"ที่ไม่สัมพันธ์กับเครื่องรับแล้ว ก็จะทำให้การรับส่งข้อมูลเกิดผิดพลาด โดยทั่วไปค่าของ BAUD RATE นั้นจะใช้ค่าต่าง ๆ ดังต่อไปนี้คือ 110 , 150,300 ,1200, 2400, 4800, และ 9600 สำหรับในบทนี้จะสมมติว่าเราต้องการที่จะส่งข้อมูลแบบอนุกรมด้วยอัตราเร็ว 2400 BAUD (2400 บิต/วินาที) และข้อมูลที่ต้องการจะส่งก็คือ 0B2H หรือ 10110010Bซึ่งเราสามารถที่จะแสดงได้ในรูปของสัญญาณดังรูปที่ 2.9



รูปที่ 2.9 ลักษณะสัญญาณ

จากรูปที่ 2.9 ความกว้างของสัญญาณของแต่ละบิตจะเท่ากับ 1 /BAUD RATE วินาที ซึ่งจากBAUD RATE ที่เราต้องการจะใช้คือ 2400 BAUD นั้นจะให้ความกว้างของแต่ละบิตมีค่าเท่ากับ 1/2400 วินาที หรือเท่ากับ 416 microseconds ซึ่งจากความกว้างของแต่ละบิตที่จะส่งไปตามสายส่งนี้ ทำให้เราคำนวณเวลาที่จะต้องใช้ในการรับข้อมูลแต่ละไบท์ (8 บิต) ได้ดังนี้คือ 8x416 microseconds

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

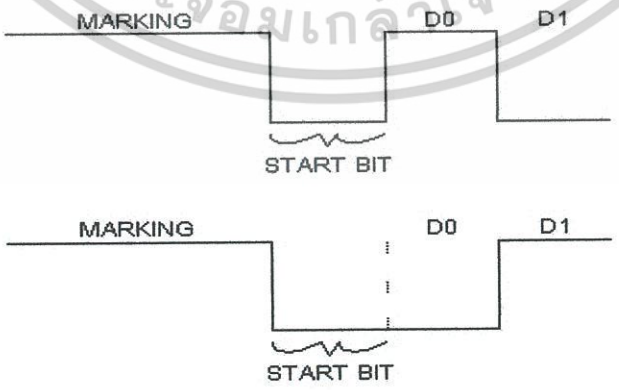
61869

หรือ 3328 microseconds อย่างไรก็ตามเพื่อป้องกันความผิดพลาดที่อาจเกิดขึ้นได้ จึงมีการเพิ่มบิตต่าง ๆ (แต่ไม่ได้หมายความว่าเมื่อเพิ่มบิตต่าง ๆ เหล่านี้เข้าไปแล้วจะทำให้การส่งผ่านข้อมูลมีความถูกต้อง 100 %) สำหรับบิตต่าง ๆ ที่เพิ่มเข้ามานี้ ก็คือ START , STOP และ PARITY BIT ซึ่งจะทำให้ข้อมูลในแต่ละไบท์ที่ส่งออกไปนี้มีมากกว่า 8 บิต และเวลาที่ใช้ในการรับส่งข้อมูลก็จะมากขึ้นตามไปด้วย

2.4.2 บิตเริ่ม (START BIT)

ในการส่งผ่านข้อมูลแบบอนุกรมนี้ จำเป็นต้องทำให้อุปกรณ์ที่จะรับส่งข้อมูลทราบว่าข้อมูลที่จะส่งมานั้นเริ่มต้นที่จุดใด ดังนั้นเราจึงจำเป็นที่จะต้องเพิ่มข้อมูล 1 บิตลงไปก่อนหน้าข้อมูลจริง (ACTUAL DATA) ที่จะทำการส่ง (การส่งอนุกรมจะส่งบิต D0 เป็นบิตแรก และ D7 เป็นบิตสุดท้าย) คือทำการเพิ่มบิตนี้ลงไปหน้าบิต D0 นั่นเอง และเรียกบิตนี้ว่า "บิตเริ่ม"

นอกจากจะใช้บิตเริ่มบอกว่าข้อมูลเริ่มต้นที่ใดแล้ว ยังทำงานร่วมกับบิตหยุด(STOP BIT) เพื่อช่วยในการแยกข้อมูลแต่ละชุดออกจากกัน และความกว้างของบิตนี้จะเท่ากับความกว้างของบิตอื่น ๆ ในข้อมูลที่จะส่ง (D0-D7) เมื่ออุปกรณ์ที่จะส่งข้อมูลยังไม่ได้ส่งข้อมูลใด ๆ ออกมา สายส่งจะอยู่ในสภาวะที่เรียกว่า "MARKING" ซึ่งเป็นสภาวะที่ไม่มีการรับส่งข้อมูลใด ๆ เกิดขึ้นในที่นี้เราสมมติให้ MARKING ของสายส่งเป็นลอจิก "1" บิตเริ่ม ที่จะเพิ่มเข้ามาจะมีลอจิกที่ตรงข้ามกับลอจิกของ MARKING ดังนั้นในกรณีนี้ บิตเริ่ม จะมีลอจิก "0" ความกว้างของบิตเริ่มเท่ากับ 1 บิตของข้อมูล เช่น ใน 1 บิตของข้อมูลมีความยาวเท่ากับ 416 ไมโครวินาทีบิตเริ่ม ก็จะมีความกว้างของสัญญาณเท่ากับ 416 ไมโครวินาที ด้วย ในรูป 2.10 จะแสดงให้เห็นถึง บิตเริ่ม ที่เพิ่มเข้าไปก่อนหน้าข้อมูล (ก่อนหน้า D0)



รูปที่ 2.10 การเพิ่ม บิตเริ่ม เข้าไปก่อนหน้าบิต D0 ในกรณีที่บิต D0 เป็น "1" และ "0" ตามลำดับ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.4.3 บิตพาริตี้ (PARITY BIT)

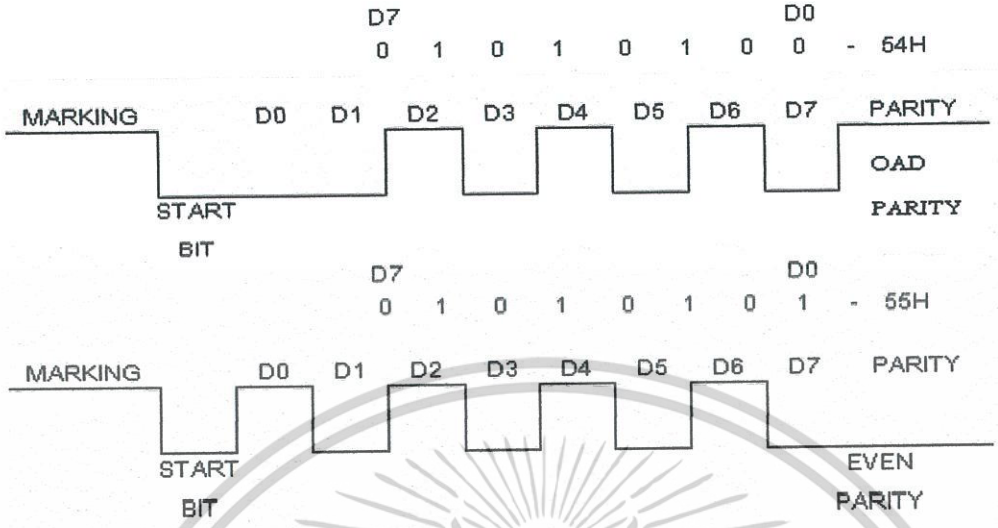
บิตพาริตี้จะทำหน้าที่ในการบอกให้ส่วนรับข้อมูลทราบว่า ข้อมูลที่ได้รับเข้ามานั้นถูกต้องเหมือนกับข้อมูลที่ถูกส่งออกมาหรือไม่ (ถึงแม้ว่าการตรวจสอบบิตนี้จะไม่พบความผิดพลาด แต่ไม่ได้หมายถึงว่าข้อมูลที่รับเข้านี้จะมีความถูกต้อง 100 %) โดยที่บิตนี้จะทำหน้าที่ในการบอกให้ส่วนรับข้อมูลทราบว่าข้อมูลที่ส่งออกมาแต่ละไบท์นั้นอยู่เป็นจำนวนคี่ หรือ จำนวนคู่ เช่น ข้อมูล 54H หรือ 01010111B จะมีจำนวนบิตที่เป็น "1" อยู่เป็นจำนวนคี่เป็นต้น สำหรับบิตที่ใช้ในการตรวจสอบนี้เรียกว่า "บิตพาริตี้"

บิตพาริตี้ นี้จะถูกส่งออกมาโดยอุปกรณ์ส่งข้อมูล ซึ่งบิตนี้จะเป็น "1" หรือ "0" นั้นขึ้นอยู่กับข้อมูลที่ส่งออกมา (D0-D7) ว่ามีจำนวนบิตที่เป็น "1" เป็นจำนวนคู่ หรือ คี่ และยังขึ้นอยู่กับอุปกรณ์รับส่งข้อมูลที่ส่งออกแบบ (โปรแกรม) ไว้ให้รับส่ง บิตพาริตี้ ในลักษณะของ บิตพาริตี้คู่ หรือ บิตพาริตี้ คี่อีกด้วย

ในกรณีที่อุปกรณ์รับส่งข้อมูลถูกออกแบบไว้ให้เป็น บิตพาริตี้ คู่ อุปกรณ์ส่งข้อมูลจะทำการส่ง บิตพาริตี้ เป็นลอจิก "1" ออกไปเมื่อจำนวนบิตที่เป็น "1" ของข้อมูลเป็นจำนวนคู่ (คือ ทำให้จำนวนบิตที่เป็น "1" ของข้อมูล (D0-D7) ร่วมกับ บิตพาริตี้ แล้เป็นจำนวนคู่นั่นเอง) สำหรับ บิตพาริตี้ คี่ก็เช่นกัน คือ บิตพาริตี้ จะเป็น "1" ในกรณีที่จำนวนบิตที่เป็น "1" ของข้อมูลเป็นจำนวนคู่และจะเป็น "0" ในกรณีที่เป็นจำนวนคี่ ในที่นี้จะสมมติว่าอุปกรณ์ถูกออกแบบไว้สำหรับ บิตพาริตี้ คู่ และเราต้องการที่จะส่งข้อมูลออกไปให้กับส่วนรับข้อมูลเป็นจำนวน 2 ไบท์คือ 54H และ 55H เมื่อเราส่งข้อมูล 54H ออกไปซึ่งมีจำนวนบิตที่เป็น "1" เป็นจำนวนคี่ ดังนั้นในกรณีนี้อุปกรณ์ส่งข้อมูลก็จะทำการส่ง บิตพาริตี้ เป็นลอจิก "1" ออกมาด้วย เพื่อจะให้จำนวนบิตที่เป็น "1" ของข้อมูล(54H0 รวมกับ บิตพาริตี้ แล้วได้เป็นจำนวนคู่ ส่วนข้อมูล 55H นั้นจำนวนบิตที่เป็น "1" นั้นเป็นจำนวนคู่อยู่แล้ว ดังนั้นอุปกรณ์ข้อมูลก็จะส่ง บิตพาริตี้ เป็น "0" ให้กับส่วนรับข้อมูล ดังในรูป 3 สำหรับส่วนรับข้อมูลนั้นเมื่อทำการรับข้อมูลเข้ามาแล้วก็จะตรวจสอบสัญญาณว่าจำนวนบิตที่เป็น "1" ของข้อมูลรวมกับ บิตพาริตี้ นั้นเป็นจำนวนคู่หรือไม่ ถ้าหากว่าเป็นจำนวนคี่ก็แสดงว่าข้อมูลที่ได้รับเข้ามานี้มีความผิดพลาดเกิดขึ้น (แต่ไม่ได้หมายความว่า ถ้าเป็นจำนวนคี่แล้ว ข้อมูลที่รับเข้ามาจะถูกต้องเสมอไป)

สิ่งสำคัญอีกสิ่งหนึ่งก็คือ ถ้าอุปกรณ์ที่ส่งข้อมูลทำการส่งในลักษณะ บิตพาริตี้ คู่หรือคี่ก็ตาม ส่วนรับข้อมูลก็จะต้องทำการรับในลักษณะ บิตพาริตี้ เดียวกับอุปกรณ์ส่งข้อมูลด้วย เช่นในกรณีที่อุปกรณ์ส่งข้อมูลทำการส่งข้อมูลในลักษณะของ บิตพาริตี้ คู่อุปกรณ์รับข้อมูลในลักษณะของ บิตพาริตี้ คู่ด้วย เป็นต้น

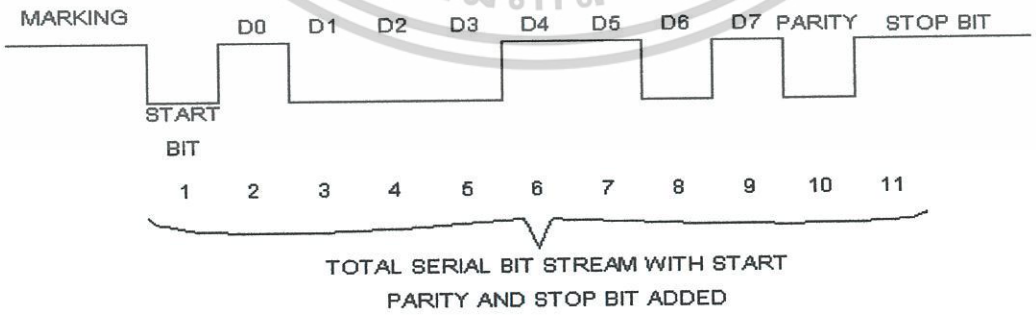
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.11 การเพิ่ม บิตพาริตี้ ลงไปในข้อมูลแต่ละไบท์

2.4.4 บิตหยุด (STOP BIT)

สำหรับบิตสุดท้ายที่เพิ่มเข้าไปนี้ จะใช้ในการตรวจสอบจุดสิ้นของข้อมูลบิตนี้ จะถูกเพิ่มเข้าไปหลัง บิตพาริตี้ ถ้าอุปกรณ์รับข้อมูลตรวจไม่พบบิตนี้ก็แสดงว่าข้อมูลที่ได้รับเข้ามานั้นมีความผิดพลาดเกิดขึ้น สำหรับ บิตหยุด นี้อาจมีจำนวนบิตเป็น 1 ,1.5 หรือ 2 บิตก็ได้ รูป 2.12 จะแสดงข้อมูลทั้ง 8 บิตที่ส่งออกรวมทั้ง บิตเริ่ม ,บิตหยุด และ บิตพาริตี้ ด้วย ซึ่งจะเห็นว่าสิ่งที่ส่งออกมาในแต่ละไบท์นั้นไม่ได้มีเพียงข้อมูล 8 บิตเท่านั้น แต่อาจจะมีได้ถึง 12 บิต (กรณีที่ส่ง บิตหยุด ออกมา 2 บิต) ดังนั้นถ้าเราทำการส่งด้วยอัตรา 2400 BAUD เราจะต้องใช้เวลาทั้งหมดเป็น 12 *416 ไมโครวินาที หรือ 4.99 มิลลิวินาที ไม่ใช่ 3328 ไมโครวินาทีดังที่ได้คำนวณไว้ในตอนต้น



รูปที่ 2.12 รูปแบบของข้อมูลแต่ละไบท์ในการรับส่งข้อมูลแบบอนุกรม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.4.5 มาตรฐานในการรับส่งข้อมูลแบบอนุกรม

สมาคมอุตสาหกรรมอิเล็กทรอนิกส์ (Electronics Industry Association) หรือ EIA ได้กำหนดมาตรฐานในการรับส่งข้อมูลแบบอนุกรมไว้ เรียกว่า Recommend Standard หรือ RS หมายเลข 232 และได้มีการปรับปรุงมาแล้ว 3 ครั้ง คือ revision A,B และ C ในปัจจุบันมาตรฐานในการสื่อสารแบบอนุกรมที่นิยมใช้กัน แพร่หลายมากคือ

1. RS-232C
2. RS-423
3. RS-422
4. RS-485

2.4.6 มาตรฐาน RS-232C

มาตรฐาน RS-232C เป็นมาตรฐานที่ได้รับการออกแบบมาเพื่อที่จะทำให้อุปกรณ์ต่อพ่วงจากผู้ผลิตต่างกันสามารถทำงานร่วมกันได้ มาตรฐานที่ได้รับความนิยมและใช้กันกว้างขวางมากที่สุดคือ มาตรฐาน RS-232C

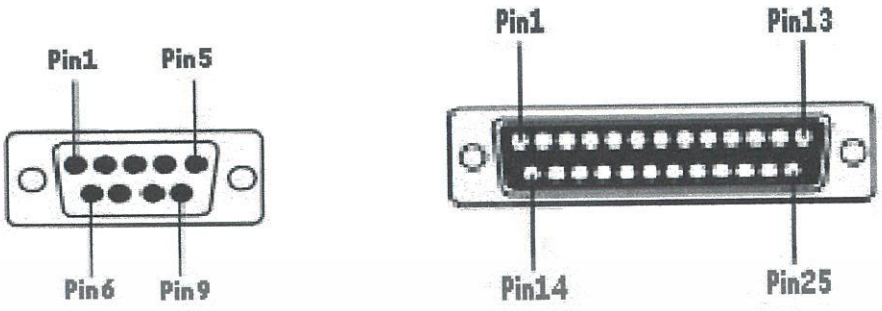
มาตรฐาน RS-232C ได้แบ่งอุปกรณ์ออกเป็น 2 ประเภท ซึ่งอุปกรณ์ทั้งสองประเภทนี้ก็คือ

1. อุปกรณ์ DTE (Data Terminal Equipment) เป็นอุปกรณ์สำหรับส่งข้อมูล (เอาต์พุต)

2. อุปกรณ์ DCE (Data Communication Equipment) เป็นอุปกรณ์สำหรับรับข้อมูล (อินพุต)

ตามมาตรฐาน RS-232C แล้วคอนเน็กเตอร์ของ DTE จะเป็นตัวผู้ ส่วนคอนเน็กเตอร์ของ DCE จะเป็นตัวเมีย ซึ่งคอนเน็กเตอร์ที่นิยมใช้กันอยู่จะเป็นชนิด D-Type แบบ 9 ขา และแบบ 25 ขา โดยจะติดตั้งอยู่หลังเครื่องคอมพิวเตอร์ ระดับแรงดันจะมีค่าระหว่าง -3V ถึง -15V สำหรับลอจิก High และลอจิก Low จะมีระดับแรงดันระหว่าง +3V ถึง +15V สามารถรับส่งข้อมูลได้ที่ความยาวของสายสัญญาณสูงสุด 50 ฟุต หรือ 150 เมตร แต่ถ้าเราต้องการสื่อสารกับอุปกรณ์อื่นที่อยู่ห่างกันมากๆ เราจำเป็นต้องใช้อุปกรณ์อื่นๆ เข้าช่วย เช่น การใช้โมเด็ม เป็นต้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.13 แผนผังคอนเน็กเตอร์ของ RS-232

ตารางที่ 2.2 รายละเอียดของสายสัญญาณ

D-Type 25 Pin	D-Type 9 Pin	สัญลักษณ์	ชื่อสัญญาณ
Pin 2	Pin 3	TD	Transmit Data
Pin 3	Pin 2	RD	Receive Data
Pin 4	Pin 7	RTS	Request To Send
Pin 5	Pin 8	CTS	Clear To Send
Pin 6	Pin 6	DSR	Data Set Ready
Pin 7	Pin 5	SG	Signal Ground
Pin 8	Pin 1	CD	Carrier Detect
Pin 20	Pin 4	DTR	Data Terminal Ready
Pin 22	Pin 9	RI	Ring Indicator

2.4.7 รายละเอียดของสายสัญญาณ

- Transmit Data : TD ใช้สำหรับส่งข้อมูลอนุกรมออกจากคอมพิวเตอร์

- Receive Data : RD ใช้สำหรับรับข้อมูลอนุกรมเข้ามายังคอมพิวเตอร์

- Request To Send : RTS ใช้สำหรับส่งข้อมูลไปยังอุปกรณ์ปลายทาง เพื่อร้องขอให้อุปกรณ์ปลายทางส่งข้อมูลกลับมา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- Clear To Send : CTS	ใช้สำหรับตรวจสอบว่าอุปกรณ์ที่เชื่อมต่อด้วยพร้อมที่จะรับข้อมูลหรือไม่
- Data Set Ready : DSR	ใช้สำหรับตรวจสอบการเชื่อมต่อกันระหว่างคอมพิวเตอร์กับอุปกรณ์ปลายทาง
- Signal Ground : SG	เป็นกราวด์ของระบบ
- Carrier Detect : CD	ขานี้จะแอ็กทีฟเมื่อมีการส่งสัญญาณแคเรี่ยจากโมเด็ม
- Data Terminal Ready : DTR	ใช้สำหรับบอกให้อุปกรณ์ปลายทางรับรู้ว่าต้องการติดต่อด้วย
- Ring Indicator : RI	ขานี้จะแอ็กทีฟเมื่อโมเด็มได้รับสัญญาณเรียกเข้าจากสายโทรศัพท์

2.5 ภาษา PHP

แต่เดิม PHP คือ Professional Home Page แต่ในปัจจุบัน PHP หมายถึง PHP Hypertext Preprocessor ซึ่งเป็นภาษาสคริปต์แบบหนึ่งที่เรียกว่า Server Side Script ที่ประมวลผลฝั่งเซิร์ฟเวอร์แล้วส่งผลลัพธ์ไปฝั่งไคลเอ็นต์ผ่านเว็บบราวเซอร์เช่นเดียวกับ ASP (Active Server Pages) ปัจจุบันได้รับความนิยมเป็นอย่างมากในการนำมาช่วยพัฒนางานบนเว็บที่เรียกว่า Web Development หรือ Web Programming เนื่องจากมีจุดเด่นหลายประการ รูปแบบของภาษา PHP มีเค้าโครงมาจากภาษา C และ Perl ที่นำมาปรับปรุงทำให้มีประสิทธิภาพสูง และทำงานได้เร็วขึ้น

2.5.1 จุดเด่นของ PHP

ถึงแม้จะรู้จักและนำมาใช้งานได้ไม่นานนัก แต่ PHP กลับได้รับความนิยมในการใช้เป็นเครื่องมือในการพัฒนาเว็บเพจ เนื่องจาก PHP มีจุดเด่นดังนี้

- Free เนื่องจากสิ่งที่ต้องการสูงสุดของโปรแกรมเมอร์ในการพัฒนาเว็บ คือของฟรี PHP ได้ตอบสนองโปรแกรมเมอร์เป็นอย่างดีเพราะเครื่องมือที่ใช้เพื่อพัฒนาทุกอย่างสามารถหาได้ฟรีๆ ไม่ว่าจะเป็นระบบปฏิบัติการ (Windows, Linux), โปรแกรมเว็บเซิร์ฟเวอร์ (IIS, PWS, Apache, OmniHTTPd), โปรแกรมระบบฐานข้อมูล (MySQL, mSQL) และ Server Site Script อย่าง PHP

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- Speed เนื่องจาก PHP นำข้อดีของภาษาสคริปต์ที่เคยมีในภาษา C , Perl และ Java รวมกับความเร็วของ CGI นำมาพัฒนาอยู่ใน PHP
- Open Source เนื่องจากการพัฒนาของ PHP ไม่ได้ยึดติดกับบุคคลหรือกลุ่มคนเล็กๆแต่เปิดโอกาสให้โปรแกรมเมอร์ทั่วไปได้เข้ามาช่วยพัฒนา ทำให้มีคนใช้งานจำนวนมากและพัฒนาได้เร็วขึ้น
- Crossable Platform เนื่องจาก PHP ใช้ได้กับหลายๆระบบปฏิบัติการไม่ว่าบน Windows, Unix, Linux หรืออื่นๆ โดยแทบจะไม่ต้องเปลี่ยนแปลงโค้ดคำสั่งเลย
- Database Access เนื่องจาก PHP สามารถติดต่อกับฐานข้อมูลอย่าง dBASE, Access, SQL Server, Oracle, Sybase, Informix, PostgreSQL, MySQL, Empress, FilePro, mSQL ได้อย่างมีประสิทธิภาพ
- Protocol Support เนื่องจาก PHP สามารถสนับสนุนโปรโตคอลหลายแบบทั้ง IMAP, SNMP, NNTP, POP3, HTTP
- Library เนื่องจาก PHP มีไลบารี่สำหรับการติดต่อกับแอพพลิเคชันได้มากมาย
- Flexible ด้วยเหตุที่ PHP มีความยืดหยุ่นตัวสูง ทำให้สามารถนำไปสร้างแอพพลิเคชันได้หลากหลายประเภท
- Easy เนื่องจาก PHP เป็นภาษาสคริปต์ภาษาหนึ่ง ทำให้สามารถแทรกตำแหน่งใดก็ได้ในแท็กของ HTML

2.5.2 PHP ทำอะไรได้บ้าง

PHP ทำทุกสิ่งที่คุณต้องการ รวมทั้งการจัดการเกี่ยวกับกราฟิกและไดนามิค HTML ด้วย ตามคู่มือของ PHP ที่กล่าวว่า "The goal of the language is to allow Web developers to write dynamically generated pages quickly" นั่นคือเป้าหมายหลักของ PHP โดยเฉพาะเรื่องไดนามิคที่สามารถเขียนได้อย่างรวดเร็ว ต่อไปนี้เป็นงานทั่วไปที่ PHP สามารถทำได้

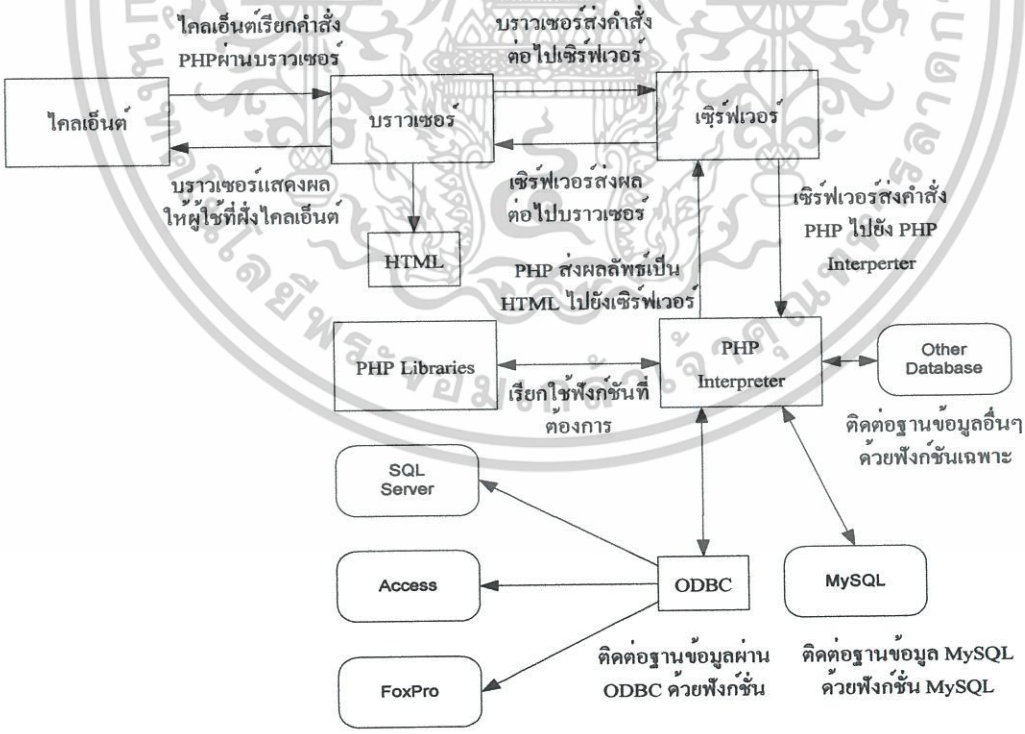
- ทำตามฟังก์ชันของระบบ ได้แก่ การสร้าง, การเปิด, อ่าน, และการปิดไฟล์ในระบบ
- เอ็กซิคิวต์คำสั่งของระบบ ได้แก่ การสร้างโฟลเดอร์ และปรับแต่งสิทธิการใช้งาน
- จัดการข้อมูลจากฟอร์ม ได้แก่ การจัดเก็บข้อมูลลงไฟล์, การส่งข้อมูลผ่านทางอีเมล, ส่งค่าข้อมูลจากการประมวลผลกลับไปยังผู้ใช้
- การติดต่อฐานข้อมูล ได้แก่ การสร้างอินเทอร์เฟซแบบเว็บเพื่อเพิ่มข้อมูล, ลบข้อมูล, การแก้ไขและอัพเดทข้อมูลในฐานข้อมูลของคุณ
- เซ็ตคุกกี้และแอ็กเซสตัวแปรคุกกี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- ใช้ PHP เพื่อรักษาความปลอดภัยของเว็บเพจ
- เข้ารหัสข้อมูล

2.5.3 หลักการทำงานของ PHP

เนื่องจาก PHP จะทำงานโดยมีตัวแปรและเอ็กซิคิวต์ที่ฝั่งเซิร์ฟเวอร์ อาจจะเรียกการทำงานว่าเป็นเซิร์ฟเวอร์ไซด์ (Server Side) ส่วนการทำงานของบราวเซอร์ของผู้ใช้เรียกว่าไคลเอ็นต์ไซด์ (Client Side) โดยการทำงานจะเริ่มต้นที่ผู้ใช้ส่งความต้องการผ่านเว็บบราวเซอร์ทาง HTTP (HTTP Request) ซึ่งอาจจะเป็นการกรอกแบบฟอร์ม หรือใส่ข้อมูลที่ต้องการ ข้อมูลเหล่านั้นจะเป็นเอกสาร PHP (เอกสารนี้จะมีส่วนขยายเป็น php หรือ php3 แล้วแต่ผู้ใช้กำหนด เช่น search.php เป็นต้น) เมื่อเอกสาร PHP เข้ามาถึงเว็บเซิร์ฟเวอร์ก็จะถูกส่งไปให้ PHP เพื่อทำหน้าที่แปลคำสั่งแล้วเอ็กซิคิวต์คำสั่งนั้น หลังจากนั้น PHP จะสร้างผลลัพธ์ในรูปแบบเอกสาร HTML ส่งกลับไปให้เว็บเซิร์ฟเวอร์เพื่อส่งต่อไปให้บราวเซอร์แสดงผลทางฝั่งผู้ใช้ต่อไป (HTTP Response) ซึ่งลักษณะการทำงานแบบนี้จะคล้ายกับการทำงานของ CGI (common Gateway Interface) หรืออาจจะกล่าวได้ว่า PHP ก็คือโปรแกรม CGI ประเภทหนึ่งก็ได้ซึ่งจะทำงานคล้ายกับ ASP นั่งเอง



รูปที่ 2.14 บล็อกไดอะแกรมลักษณะการทำงานของ PHP

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.5.4 โครงสร้างของภาษา PHP Script

การ Escape ระหว่าง HTML Codec และ PHP Code ตามมาตรฐานของ PHP สามารถกระทำได้โดยใช้ เครื่องหมาย <? และ ?> ทั้งนี้เนื่องจากเป็นวิธีการที่ใช้กันบนหลายระบบ เช่น ASP (Active Server Page ของ Microsoft) และ PHP ดังตัวอย่าง

```
< HTML> <HEAD><TITLE>
<? echo ("TEST OF PHP SCRIPT") ; ? >
</ TITLE><HEAD>
<BODY>
<? echo ("PHP Script Body") ; ?>
</BODY></HTML>
```

2.6 ภาษา Visual Basic.Net

ในบรรดาภาษาคอมพิวเตอร์ที่ใช้พัฒนาโปรแกรมที่ทำงานอยู่บนระบบปฏิบัติการวินโดวส์ คงไม่มีใครปฏิเสธได้ว่า Visual Basic เป็นภาษาที่ได้รับความนิยมมากที่สุดภาษาหนึ่ง เนื่องจากเป็นภาษาที่มีขั้นตอนในการพัฒนาที่ค่อนข้างน้อย กระทำได้ง่ายและสะดวกต่อการใช้งาน ดังนั้นจึงใช้เวลาเพียงแค่ 2-3 ชั่วโมง ก็สามารถพัฒนาโปรแกรมแรกขึ้นใช้งานได้แล้ว แต่อย่างไรก็ตาม เนื่องจาก Visual Basic เดิมยังขาดคุณสมบัติของ Object-Oriented ในบางด้านอยู่ จึงไม่สนับสนุนแนวคิดในการพัฒนาโปรแกรมแบบ Reusable Code เท่าใดนัก ด้วยเหตุนี้ VB.NET จึงได้เกิดขึ้น นอกจากนี้ ในปัจจุบัน อินเตอร์เน็ท ได้เข้ามามีบทบาทในชีวิตประจำวันของเรามากขึ้น จึงทำให้แนวโน้มของการพัฒนาโปรแกรมมิได้อยู่บนระบบปฏิบัติการวินโดวส์อีกต่อไปแต่ยังรวมถึงบนอินเตอร์เน็ทอีกด้วย ดังนั้นเพื่อให้ครอบคลุมทั้งสองจุดมุ่งหมาย VB.NET จึงได้ถูกพัฒนาให้เป็นภาษาที่สามารถนำไปพัฒนาโปรแกรมสคริปต์ที่จะทำงานร่วมกับเว็บเพ็จได้อีกด้วย

2.6.1 VB.NET คืออะไร

VB.NET ย่อมาจาก Visual Basic.NET เป็นภาษาคอมพิวเตอร์ที่นำเอา Visual Basic 6 มาพัฒนาด้วยการเพิ่มความสามารถของเทคโนโลยี.NET Framework เข้าไปเพื่อทำให้ VB.NET เป็นภาษาที่ใช้พัฒนาโปรแกรมทั้งที่ใช้งานอยู่บนเว็บและระบบปฏิบัติการวินโดวส์ตัว.NET Framework ได้แก่ เทคโนโลยีที่พัฒนาขึ้นตามแนวคิด .NET ที่บริษัทไมโครซอฟท์คิดค้นขึ้น โดยมีจุดมุ่งหมายเพื่อสนับสนุนแนวคิดแบบ Reusable Code ช่วยลดช่องว่างในความแตกต่างด้านต่างๆที่เกิดขึ้นในการพัฒนาโปรแกรม เช่น ความแตกต่างระหว่างแนวคิดในการพัฒนาโปรแกรมบนวินโดวส์กับที่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

พัฒนาบนเว็บเป็นต้น ดังนั้นไม่ว่าจะเป็นภาษาใดก็ตามที่ทำงานอยู่ภายใต้ .NET Framework จึงสามารถใช้ทรัพยากรต่างๆของ .NET Framework ได้เช่นเดียวกัน จะต่างกันเพียงแค่รูปแบบในการนำไปใช้งาน ซึ่งจะต้องเป็นไปตามไวยากรณ์ของภาษานั้นๆ

VB.NET นี้ค่อนค่างจะแตกต่างจาก Visual Basic 6 ซึ่งรายละเอียดนั้นมีมาก ที่ชัดเจนคือ

- Visual Basic 6 ไม่ได้ใช้แนวคิดแบบ Object-Oriented อย่างแท้จริง ในขณะที่ VB.NET ได้ปรับปรุงให้มีคุณสมบัติที่ใกล้เคียงกับแนวคิดแบบ Object-Oriented มากขึ้น
- Visual Basic 6 เนื่องจากขาดความสามารถบางอย่างตามแนวคิดของ Object-Oriented ไป จึงต้องชดเชยด้วยการเพิ่มวิซาร์ดไว้ให้ใช้งาน และเป็นผลให้โปรแกรมเมอร์ขาดทักษะในการเขียนโปรอกรมในลักษณะของคลาสซึ่งเป็นเทคนิคในการทำ Reusable Code และจำเป็นต่อการพัฒนาโปรแกรมขนาดใหญ่ไป ซึ่งต่างจาก VB.NET ที่ทุกอย่างนั้นจะอยู่ในรูปของคลาสทั้งหมด ดังนั้นเราจึงสามารถพัฒนาโปรแกรมในแบบ Reusable Code ได้อย่างเต็มที่ ส่งผลให้วิซาร์ดที่ช่วยในการพัฒนาโปรแกรมหลายๆตัวใน Visual Basic 6 ได้ถูกยกเลิกไป
- .NET Framework ได้นำเอา Common Language Runtime (CLR) และ MS Intermediate Language (MSIL) เข้ามาใช้ จึงทำให้ทุกภาษาภายใต้เทคโนโลยี .NET Framework เช่น VB.NET , C# , JSCRIP.NET ฯลฯ เมื่อถูกคอมไพล์จึงมี Machine Code ที่อยู่ในรูปแบบที่สอดคล้องกัน และสามารถใช้ร่วมกันได้จึงเป็นผลให้โปรแกรมหนึ่งสามารถใช้ภาษาพัฒนาได้มากกว่าหนึ่งภาษา
- เพื่อสนองแนวคิด Reusable Code ตัว .NET Framework จึงได้เตรียมฟังก์ชั่นการทำงานไว้ในรูปแบบคลาสไลบรารี่ เพื่อที่จะได้นำมาใช้กับคลาสที่เราสร้างขึ้นได้ทันที ซึ่งเป็นผลทำให้ลดภาระในการพัฒนาโปรแกรมลง
- คลาสของ Visual Basic 6 เนื่องจากไม่ได้เป็นแนวคิดแบบ Object-Oriented เท่าใดนักจึงทำให้ไม่สามารถทำ inheritance ได้อย่างแท้จริง ต้องอาศัยวิซาร์ดซึ่งมีขั้นตอนที่ยุ่งยากเข้ามาช่วย นอกจากนี้คลาสที่สร้างขึ้นยังไม่สามารถกำหนดค่าเริ่มต้นได้อีกด้วย จึงเป็นผลทำให้ไม่สามารถกำหนดสถานะของอ็อบเจ็กได้ในขณะคลาสของVB.NET ปัญหาทั้งสองนี้ได้แก้ไขแล้ว
- ใน .NET Framework ได้นำเอา Garbage Collection เข้ามาช่วยในการคืนพื้นที่ของหน่วยความจำที่ไม่ได้ใช้งานให้กับระบบ จึงจึงสนับสนุนเทคนิค Circular Reference ที่ Visual Basic 6 ไม่ได้รองรับ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- คำสั่งที่ใช้ในการตรวจหาและจัดการกับข้อผิดพลาดที่เกิดขึ้นของ VB.NET เนื่องจาก VB.NET ในรูปของบล็อก ดังนั้นจึงไม่ทำให้เกิดโครงสร้างในแบบ Spagthetti Code เช่นเดียวกับคำสั่ง On Error ของ Visual Basic 6
- VB.NET มีความเป็น Multithreading มากกว่า Visual Basic 6

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 3
การออกแบบและการสร้าง

การออกแบบและการสร้างโครงงานนี้มี 2 ส่วนคือในส่วนของซอร์ฟแวร์และฮาร์ดแวร์โดยในส่วนซอร์ฟแวร์จะมีการโปรแกรมสั่งงานจากคอมพิวเตอร์ทางอินเตอร์เน็ตไปควบคุมปั๊มน้ำและวาล์วโดยผ่านระบบคอมพิวเตอร์ฝังตัว (Embedded System) เข้าไปยังไมโครคอนโทรเลอร์ PIC16F628 เพื่อทำการสั่งงานควบคุมปั๊มให้ทำงานและอีกทางหนึ่งระบบคอมพิวเตอร์ฝังตัวจะรอรับข้อมูลจากระบบเซ็นเซอร์แบบไร้สายเพื่อสั่งงานปั๊มเช่นกันและในส่วนฮาร์ดแวร์จะมีในส่วนของระบบคอมพิวเตอร์ฝังตัวติดต่อผ่านพอร์ตอนุกรม เข้าไปยัง MAX232 เพื่อติดต่อวงจรไมโครคอนโทรเลอร์ PIC 16F628ไปควบคุมวงจรสั่งงานปั๊มให้ทำงาน

3.1 โปรแกรมควบคุม

ในส่วนโปรแกรมควบคุมของโครงงานนี้แบ่งออกเป็น 3 ส่วน ได้แก่

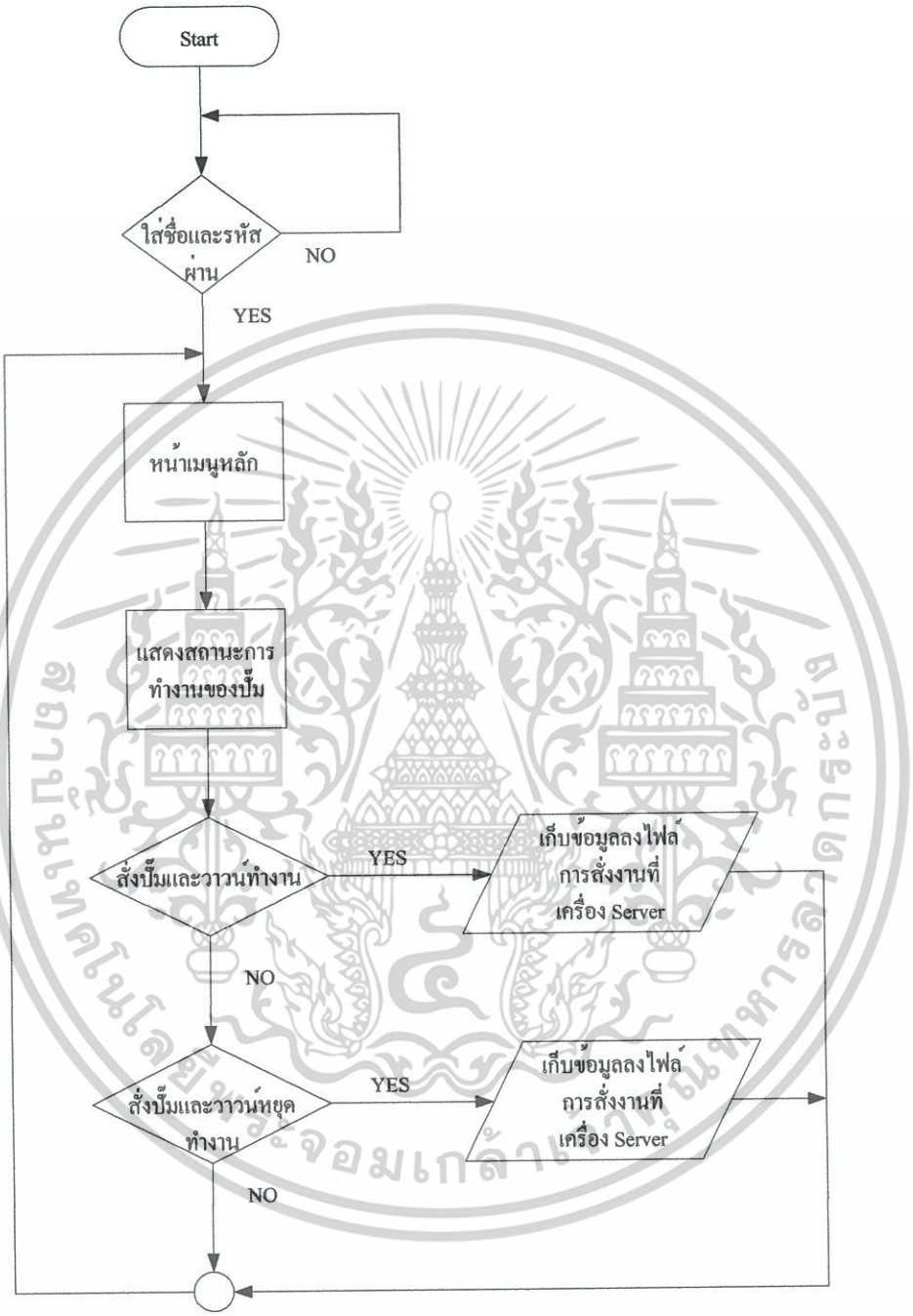
1. โปรแกรมควบคุมปั๊ม จากคอมพิวเตอร์ผานทางอินเตอร์เน็ตไปเข้าระบบคอมพิวเตอร์ฝังตัว
2. โปรแกรม Visual Basic.NET บนระบบคอมพิวเตอร์ฝังตัวจะรอรับการติดต่ออินเตอร์เฟสจากคอมพิวเตอร์ที่ผ่านทางระบบอินเตอร์เน็ตและจากระบบเซ็นเซอร์ไร้สายเพื่อทำการส่งค่าผ่านพอร์ตอนุกรมไปยังไมโครคอนโทรเลอร์ PIC16F628
3. โปรแกรมภาษาซีควบคุมไมโครคอนโทรเลอร์PIC16F628โดยจะรอรับค่าจากระบบคอมพิวเตอร์ฝังตัวผ่าน max232 เข้า PIC16F628 แล้วสั่งงานไปควบคุมให้ปั๊มทำงาน

3.1.1 โปรแกรมควบคุมปั๊ม จากคอมพิวเตอร์ไปเข้าระบบคอมพิวเตอร์ฝังตัว

ในส่วนของโปรแกรมนี้จะแสดงอยู่ที่คอมพิวเตอร์เป็นหน้าเว็บเพ็จเพื่อใช้ในการสั่งงานให้ปั๊มทำงานโดยทำการควบคุมผ่านคอมพิวเตอร์ซึ่งโปรแกรมจะส่งค่าข้อมูลออกทางพอร์ตสื่อสารอินเตอร์เน็ต โดยจะทำการติดต่อส่งค่าข้อมูลไปยังคอมพิวเตอร์ฝังตัวซึ่งทำหน้าที่เป็นเว็บเซิร์ฟเวอร์รับข้อมูลไปประมวลผลว่าควบคุมให้ปั๊มทำงานหรือหยุดทำงาน

โดยในตัวโปรแกรมจะมีฟังก์ชั่นการทำงาน คือการสั่งงานควบคุมให้ปั๊มทำงานและการสั่งให้ปั๊มนั้นหยุดทำงาน และสามารถตั้งเวลาการสั่งงานล่วงหน้าได้รวมทั้งสามารถดูฐานข้อมูลการตั้งเวลาการสั่งงานผ่านทางอินเตอร์เน็ตอีกทั้งยังแสดงสถานะ การทำงานของปั๊มว่ามีตัวใดเปิดหรือปิดการทำงาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



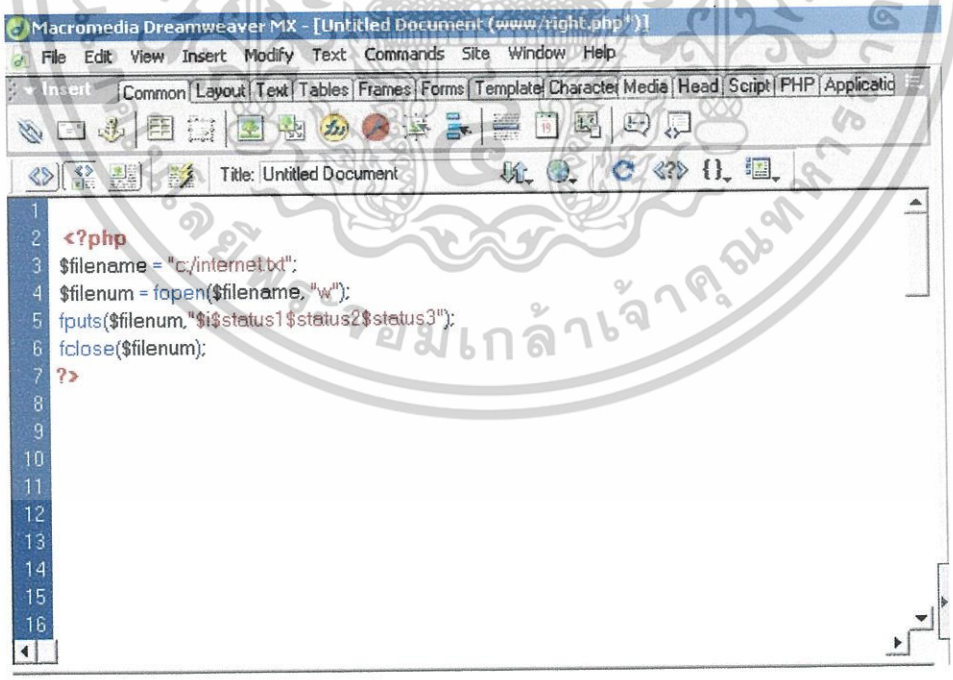
รูปที่ 3.1 ผังงานโปรแกรมควบคุมปั๊ม จากคอมพิวเตอร์ทางหน้าเว็บเพจไปยังระบบคอมพิวเตอร์ฝังตัว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.1.1.1 การสร้างการติดต่อ PHP กับไฟล์

ในการเขียน PHP ติดต่อกับไฟล์จะต้องมีฟังก์ชันในการติดต่อกับไฟล์ โดยในที่นี้จะใช้ติดต่อกับไฟล์เพื่อทำการเก็บค่าข้อมูลของการสั่งเปิดปิดวาล์วจากหน้าเว็บเพจลงไฟล์ โดยสามารถเขียนได้ดังนี้

- Fopen() เป็นการเปิดไฟล์หรือ URL
  โดยมีรูปแบบคือ Fopen(filename,mode)
  filename คือชื่อไฟล์
  mode คือโหมดกำหนดด้วย RWX คือ Read , Write , Execute
- Fputs() เขียนข้อความลงไฟล์ ถ้าไฟล์เดิมมีข้อมูลอยู่จะลบข้อมูลเดิมทั้งหมดก่อน
  โดยมีรูปแบบคือ Fputs(filename,text)
  text คือข้อความที่จะเขียนลงไฟล์
- Fclose() ปิดไฟล์ที่เปิดอยู่
  โดยมีรูปแบบคือ Fclose(file_handle)
- Fread() อ่านข้อมูลจากไฟล์ โดยสามารถระบุความยาวที่จะอ่านได้
  โดยมีรูปแบบคือ Fread(file_handle,length)



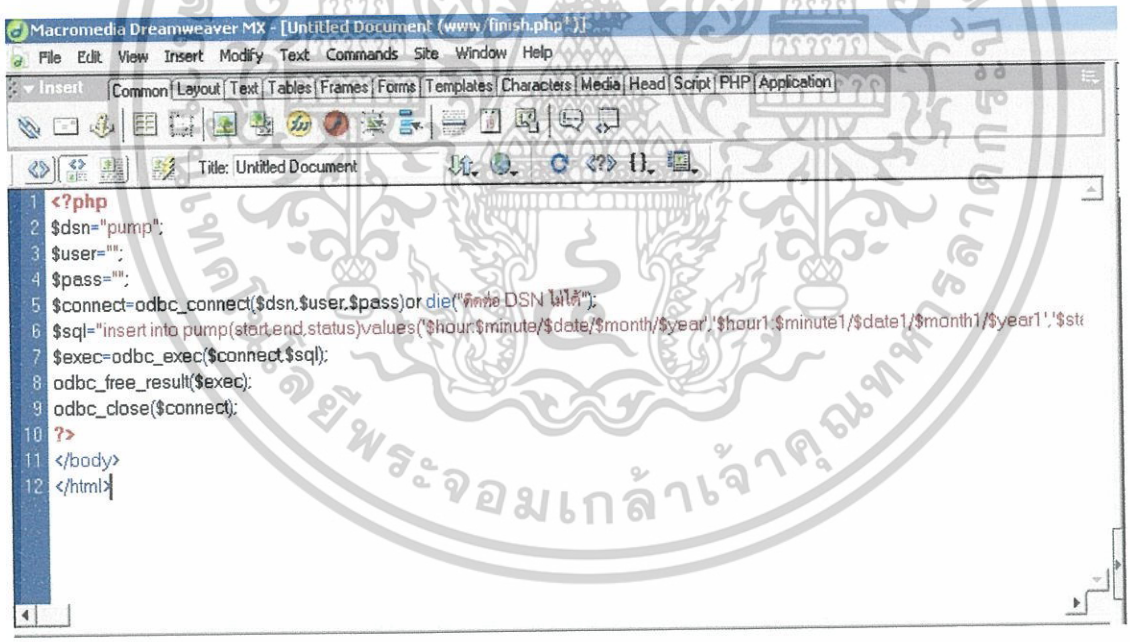
รูปที่ 3.2 การสร้างการติดต่อ PHP กับไฟล์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.1.1.2 การสร้างการติดต่อ PHP กับฐานข้อมูล

การติดต่อกับฐานข้อมูลจะใช้เพื่อเก็บข้อมูลวันและเวลาที่สั่งงานตั้งเวลาเปิดปิดวาล์วล่วงหน้าจากหน้าเว็บเพจลงฐานข้อมูลเพื่อนำไปใช้สั่งงานวาล์วในเวลาต่อไป สำหรับการติดต่อกับฐานข้อมูลและแสดงผลข้อมูล รวมทั้งปิดการเชื่อมต่อมีขั้นตอนย่อยดังนี้

- $dsn = “”; กำหนดชื่อ DSN
- $user = “”; กำหนดชื่อล็อกอิน
- $pass =””; กำหนดรหัสผ่าน
- $connect = odbc_connect($dsn , $user , $pass); เริ่มติดต่อฐานข้อมูล
- $sql = “select * from tblname”; กำหนดคำสั่ง SQL เพื่อแสดงข้อมูล
- $exec=odbc_exec($connect , $sql); เริ่มเอ็กซิคิวต์คำสั่ง SQL
- odbc_free_result($exec); ปล่อยให้รีซอร์สที่เก็บการเอ็กซิคิวต์คำสั่ง SQL เป็นอิสระ
- odbc_close($connect); ปิดการติดต่อฐานข้อมูล

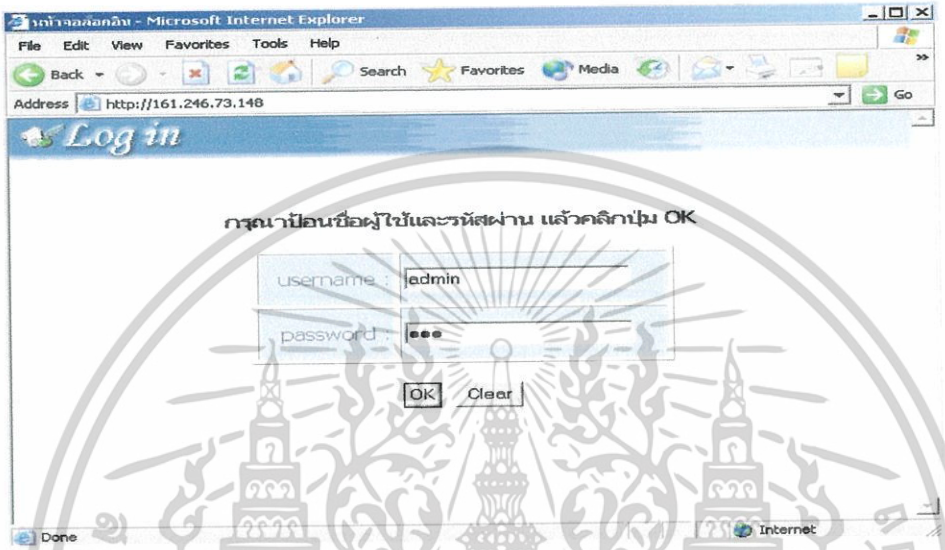


รูปที่ 3.3 การสร้างการติดต่อ PHP กับฐานข้อมูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

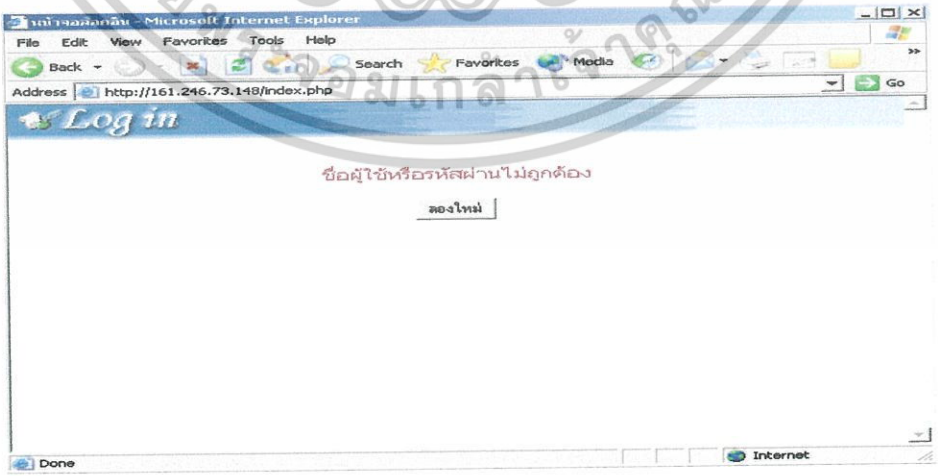
3.1.1.3 การออกแบบและการสร้างเว็บเพจ

ลำดับขั้นตอนการออกแบบและการสร้าง โดยจะเป็นการออกแบบการทำงานของโปรแกรมควบคุมปั๊มและวาล์วผ่านระบบอินเตอร์เน็ต ว่ามีผลการทำงานเป็นอย่างไร



รูปที่ 3.4 หน้าจอล็อกอินระบบควบคุมปั๊มและวาล์ว

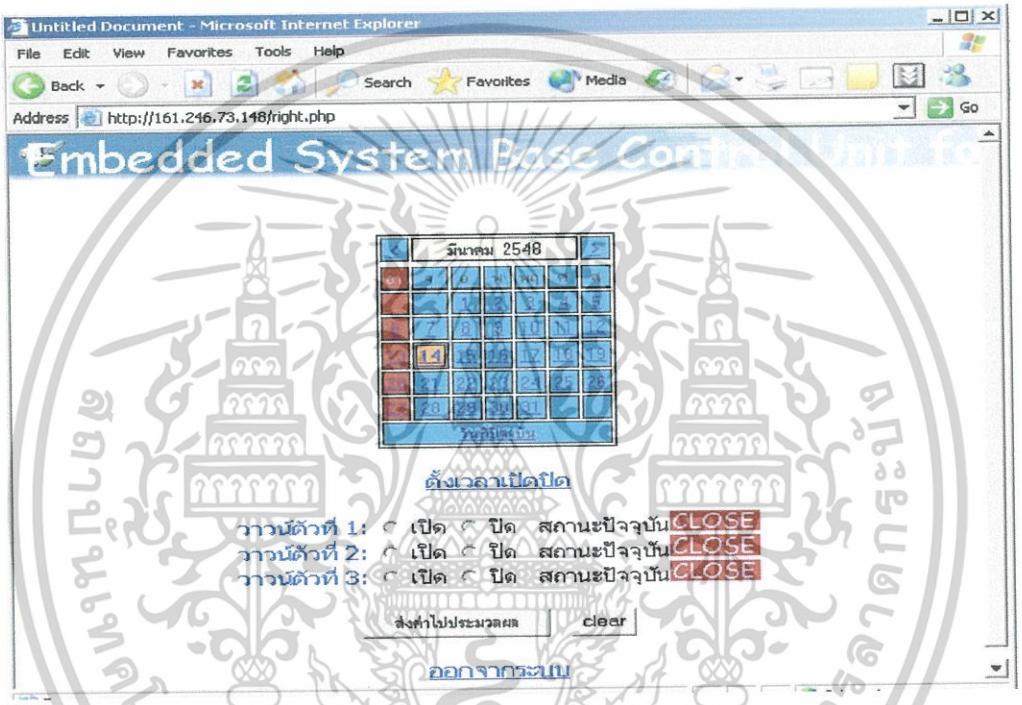
รันโปรแกรม Web Brower แล้วเรียกที่ IP ของเว็บเซิร์ฟเวอร์ดังรูปที่ 3.4 ผู้ใช้จะต้องใส่ชื่อและรหัสผ่านให้ถูกต้องก่อน ในโครงงานนี้จะใช้รหัสผ่านเป็นระบบรักษาความปลอดภัย ซึ่งถ้ารหัสไม่ถูกต้องหน้าจอจะแสดงผลดังรูปที่ 3.5



รูปที่ 3.5 หน้าจอเมื่อใส่รหัสผิดพลาด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

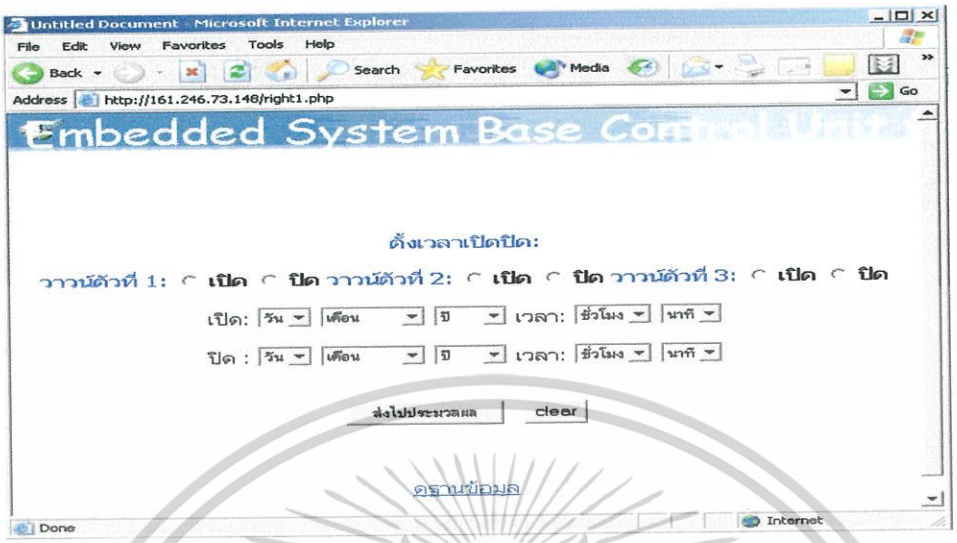
เมื่อใส่รหัสถูกต้องหน้าจอจะแสดงผลดังรูปที่ 3.6 เป็นหน้าจอเป็นหน้าจอหลักในการสั่งงานควบคุมปั๊มและวาล์วว่าให้ตัวใดทำงานหรือไม่ทำงานและสามารถสั่งงานล่วงหน้าได้โดยทำการเลือกวันทางปฏิทินหรือจะตั้งเวลาเปิดปิดแล้วแต่ผู้ใช้กำหนดว่าให้เปิดทำงานวันใดเวลาใดและจะปิดวันใดเวลาใดซึ่งข้อมูลการสั่งงานเปิดปิดล่วงหน้าจะไปเก็บอยู่ในฐานข้อมูลซึ่งจะแสดงในส่วนถัดไป และมีส่วนของการแสดงสถานะของวาล์วว่าตัวใดเปิดหรือปิดอยู่



รูปที่ 3.6 หน้าจอหลักและสถานะของการควบคุมปั๊มผ่านทางอินเตอร์เน็ต

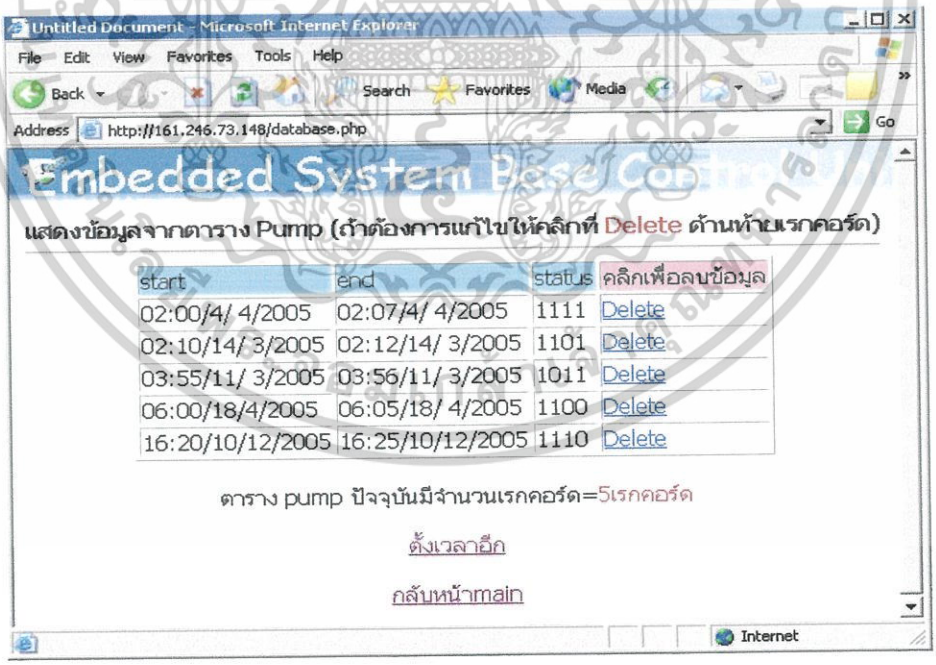
ในรูปที่ 3.7 จะเป็นการตั้งเวลาเปิดปิดวาล์วล่วงหน้าว่าให้ทำการเปิดปิดวันใดเวลาใดโดยจะมีการเลือกวันเวลาและวาล์วว่าให้ตัวใดเปิดหรือปิด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.7 หน้าจอการตั้งเวลาเปิดปิดล่วงหน้า

เมื่อเราได้ทำการตั้งเวลาเปิดปิดวาล์วล่วงหน้าแล้ว ในรูปที่ 3.8 จะเป็นข้อมูลจากฐานข้อมูลว่าได้ทำการสั่งเปิดปิดวาล์วล่วงหน้าวันและเวลาใดบ้างรวมทั้งวาล์วตัวใดเปิดหรือปิด



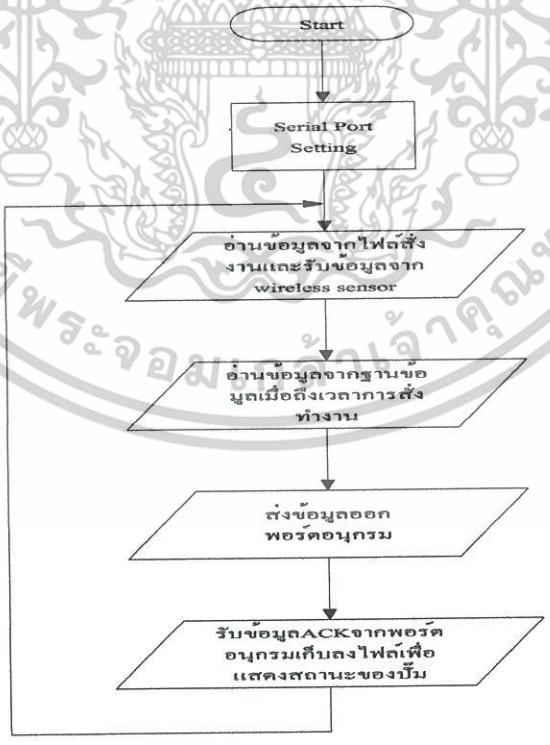
รูปที่ 3.8 ฐานข้อมูลการตั้งเวลาเปิดปิดวาล์วล่วงหน้า

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.1.2 โปรแกรม Visual Basic.NET ของคอมพิวเตอร์ฝังตัว

ในส่วนของโปรแกรมในคอมพิวเตอร์ฝังตัวที่ใช้ในการควบคุมให้ปั๊มทำงานนั้นจะใช้โปรแกรม Visual Basic.NET ในการเขียน โดยโปรแกรมจะรับค่าข้อมูลมาจากสองทางคือจากคอมพิวเตอร์ซึ่งส่งข้อมูลมาเก็บลงไฟล์ที่คอมพิวเตอร์ฝังตัวซึ่งทำหน้าที่เป็นเว็บเซิร์ฟเวอร์และจากระบบเซ็นเซอร์แบบไร้สายโดยส่งข้อมูลเข้ามาทางพอร์ตอนุกรม ซึ่งโปรแกรมจะนำค่าข้อมูลที่ได้รับมานั้นมาทำการประมวลผลว่าข้อมูลที่ได้รับนั้นเป็นการควบคุมให้ปั๊มทำงานหรือว่าหยุดการทำงานของปั๊มและอีกหน้าที่หนึ่งคือทำการตรวจสอบข้อมูลภายในฐานข้อมูลว่าได้ทำการตั้งเวลาการเปิดปิดปั๊มไว้ล่วงหน้าที่เวลาใดบ้างเมื่อถึงเวลาการเปิดปิดก็จะทำการประมวลผลส่งค่าที่ได้ไปสั่งงานปั๊มและวาล์วทำงานตามที่เราได้ตั้งเวลาไว้ล่วงหน้า

จากนั้นเมื่อโปรแกรมรับค่ามาทำการตรวจสอบแล้วว่าเป็นการควบคุมให้ปั๊มและวาล์วนั้นทำงานหรือว่าหยุดการทำงานก็จะทำการส่งค่าข้อมูลผ่านพอร์ตอนุกรมของคอมพิวเตอร์ฝังตัวไปยังไมโครคอนโทรเลอร์ PIC16F628 อีกทีหนึ่ง และอีกการทำงานหนึ่งคือทำการรับข้อมูลจากไมโครคอนโทรลเลอร์ทางพอร์ตอนุกรมที่ตอบกลับมาว่าได้ทำการสั่งงานเปิดปิดปั๊มเป็นที่เรียบร้อยแล้วโดยค่าข้อมูลที่ได้จะนำไปแสดงสถานะการทำงานของปั๊มและวาล์วต่อไปอีกทีหนึ่ง



รูปที่ 3.9 ผังงาน Visual Basic.NET ของคอมพิวเตอร์ฝังตัว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.1.2.1 การสร้างการติดต่อพอร์ตอนุกรมของ Visual Basic.NET

การเขียนโปรแกรมติดต่อ และควบคุมพอร์ตอนุกรมกับวิชวลเบสิก คอนโทรลที่สำคัญคือ MSComm โดยในที่นี้ถ้าเรากำหนดให้เขียนโปรแกรมติดต่อกับพอร์ต Com1จะเขียนเป็น

MSComm1.CommPort = 1

และกำหนดอัตราบอดเรตหรือความเร็วในการส่งข้อมูลมีหน่วยเป็นบิตต่อวินาที,พาริตี้, จำนวนของบิตข้อมูล,จำนวนของบิตปิดท้ายจะเขียนได้เป็น

MSComm1.Settings = “9600,n,8,1”

และกำหนดค่าของข้อมูลที่จะอ่านจากบัฟเฟอร์ภาครับคือ

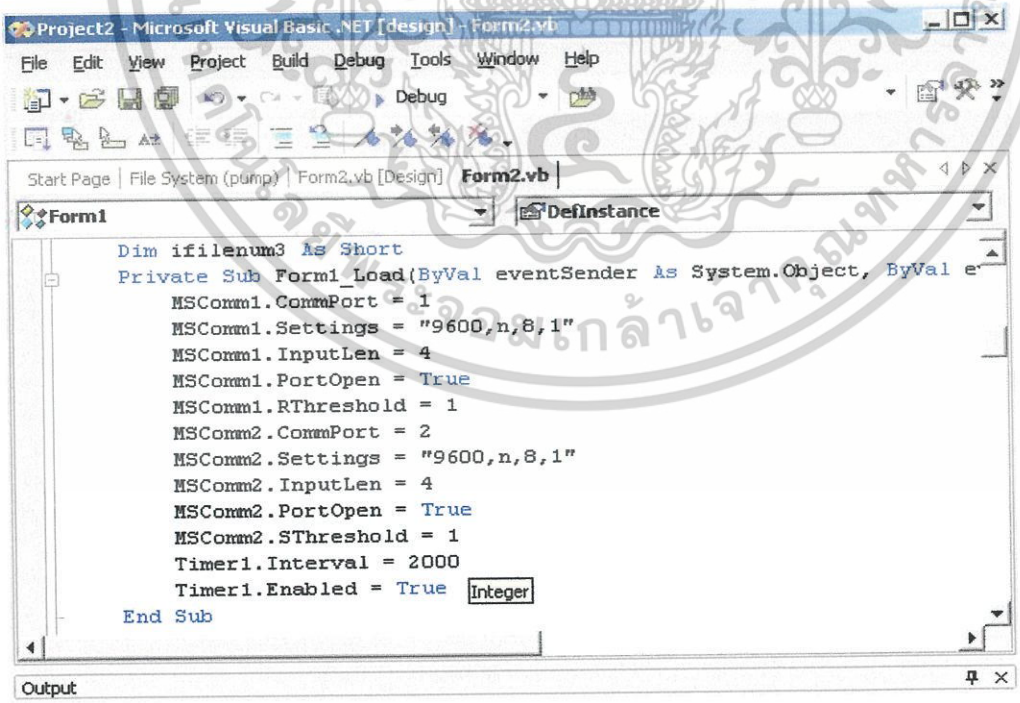
MSComm1.InputLen = 4

การเปิดการใช้งานพอร์ตอนุกรม โดยมีรูปแบบคือ

MSComm1.PortOpen = True

การปิดการใช้งานพอร์ตอนุกรม โดยมีรูปแบบคือ

MSComm1.PortOpen = False



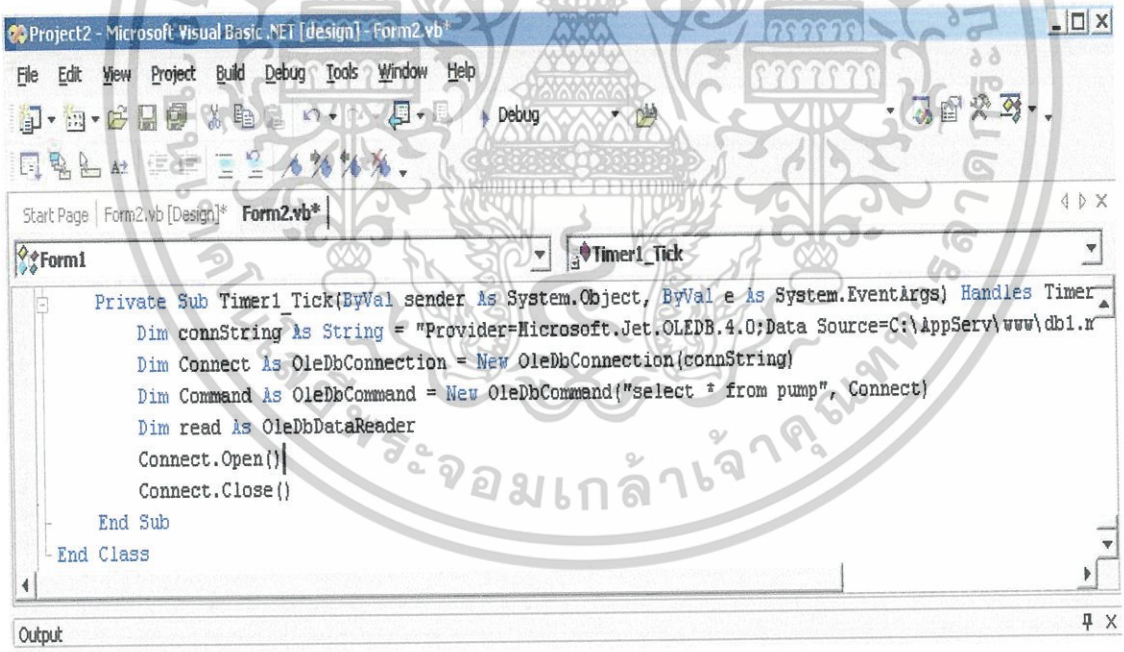
รูปที่ 3.10 การสร้างการติดต่อพอร์ตอนุกรมของ Visual Basic.NET

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.1.2.2 การสร้าง Visual Basic.NET ติดต่อกับฐานข้อมูล

การติดต่อกับฐานข้อมูลจะต้องมี Component เพื่อใช้ในการติดต่อ โดย .NET Data Provider จะได้แก่กลุ่มของ Component ที่ใช้กระทำกับข้อมูลในฐานข้อมูลได้แก่ OLE DB.NET Data Provider โดยรูปที่3.11 จะเป็นการสร้างการติดต่อกับฐานข้อมูลจะเขียนได้เป็น

- Provider ได้แก่ประเภทของ OLE DB Provider สำหรับไฟล์ฐานข้อมูลที่ใช้เป็นของ Microsoft Access ดังนั้นจึงกำหนดเป็น “Microsoft.Jet.OLEDB.4.0”
- Data Source ได้แก่ไดเร็กทรอรี่และชื่อไฟล์ฐานข้อมูลที่ต้องการติดต่อ
- Connection ใช้สร้างการติดต่อกับข้อมูล และเป็นอ็อปเจ็คแรกที่จะต้องกำหนดขึ้นใช้งานก่อนอ็อปเจ็คอื่น
- Command ใช้กระทำกับข้อมูลในแบบ Connected คล้ายกับ ADO
- DataReader ใช้เก็บผลลัพธ์ที่ได้จากการอ่านข้อมูลด้วยอ็อปเจ็ค “Command”
- DataAdapter ใช้กระทำกับข้อมูลในแบบ Disconnected ด้วยอ็อปเจ็ค “DataSet”

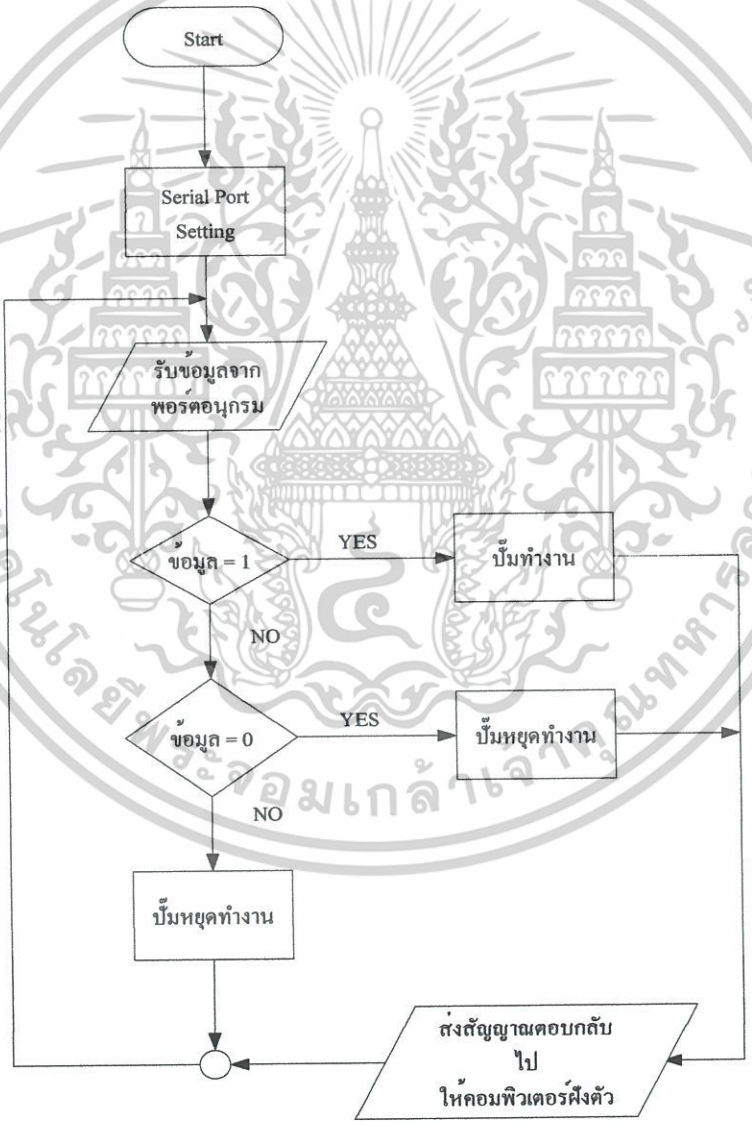


รูปที่ 3.11 การสร้าง Visual Basic.NET ติดต่อกับฐานข้อมูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.1.3 โปรแกรมภาษาซีควบคุมไมโครคอนโทรเลอร์ PIC16F628

ในส่วนของตัวไมโครคอนโทรเลอร์ PIC16F628 ภาษาที่จะใช้เขียนโปรแกรมในตัว PIC16F628จะใช้ภาษาซีในการเขียนโดยโปรแกรมนี้จะทำการรอรับค่าข้อมูลที่มาจากพอร์ตอนุกรมของระบบคอมพิวเตอร์ฝังตัวผ่านเข้ามายัง MAX232 แล้วเข้า PIC16F628แล้วนำค่าข้อมูลที่ได้มาทำการประมวลผลว่าข้อมูลนั้นเป็นข้อมูลที่สั่งให้ปั๊มทำงานหรือว่าหยุดการทำงาน จากนั้นจึงส่งออกพอร์ตไปควบคุมปั๊มตามค่าข้อมูลที่ได้รับเมื่อสั่งงานปั๊มแล้วก็จะทำการส่งสัญญาณตอบกลับไปให้คอมพิวเตอร์ฝังตัวว่าได้ทำการสั่งงานให้ปั๊มตัวใดทำงานหรือหยุดทำงานเป็นที่เรียบร้อยแล้ว

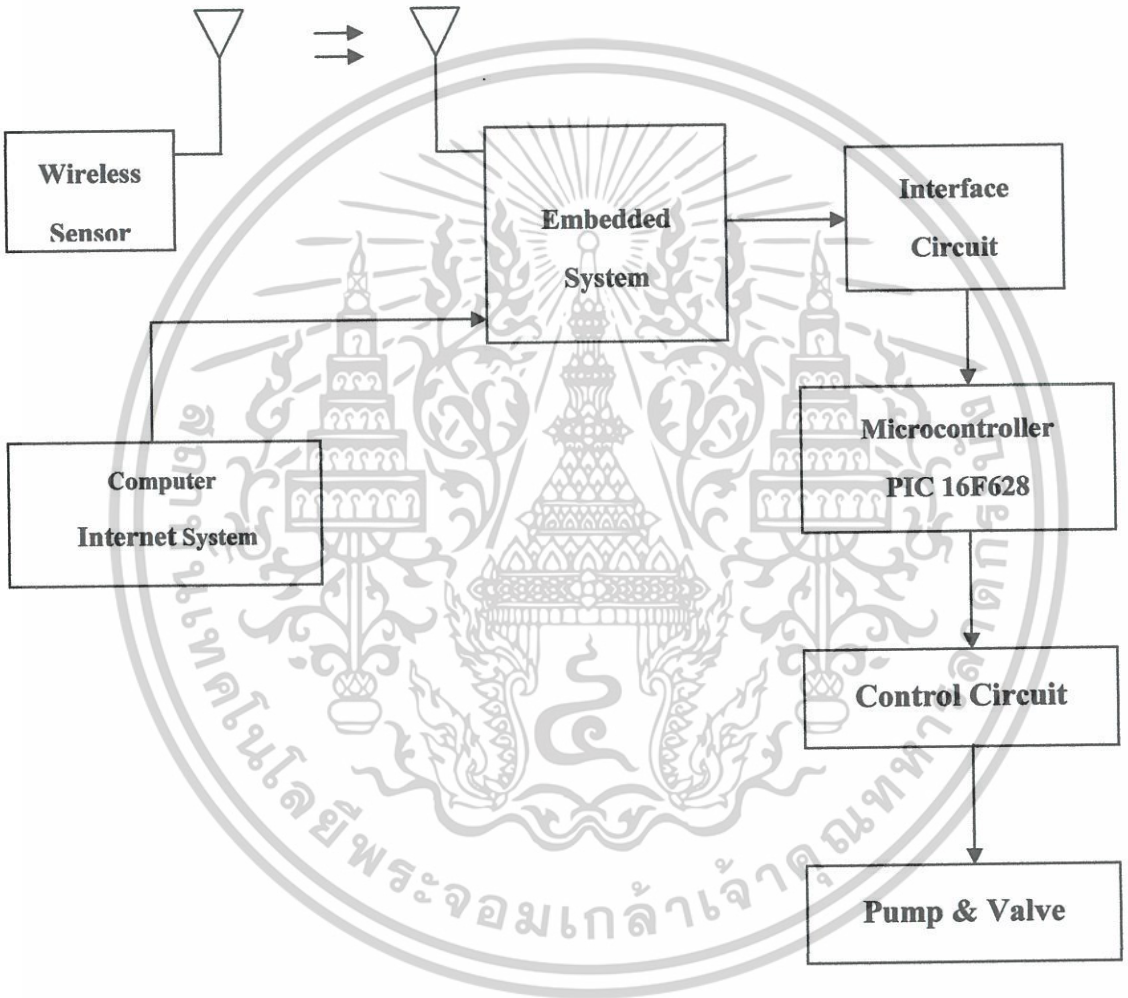


รูปที่ 3.12 ผังงานโปรแกรมภาษาซีควบคุมไมโครคอนโทรเลอร์ PIC16F628

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.2 วงจรควบคุม

ในส่วนของวงจรควบคุมของโครงงานนี้จะมีด้วยกัน 3 ส่วนคือ1.วงจรแหล่งจ่ายไฟ 2. วงจรอินเตอร์เฟสกับระบบคอมพิวเตอร์ฝังตัว 3.สุดท้ายคือวงจรควบคุมปั๊มและวาล์วโดยไมโครคอนโทรเลอร์ PIC 16F628



รูปที่ 3.13 บล็อกไดอะแกรมการทำงานของระบบ

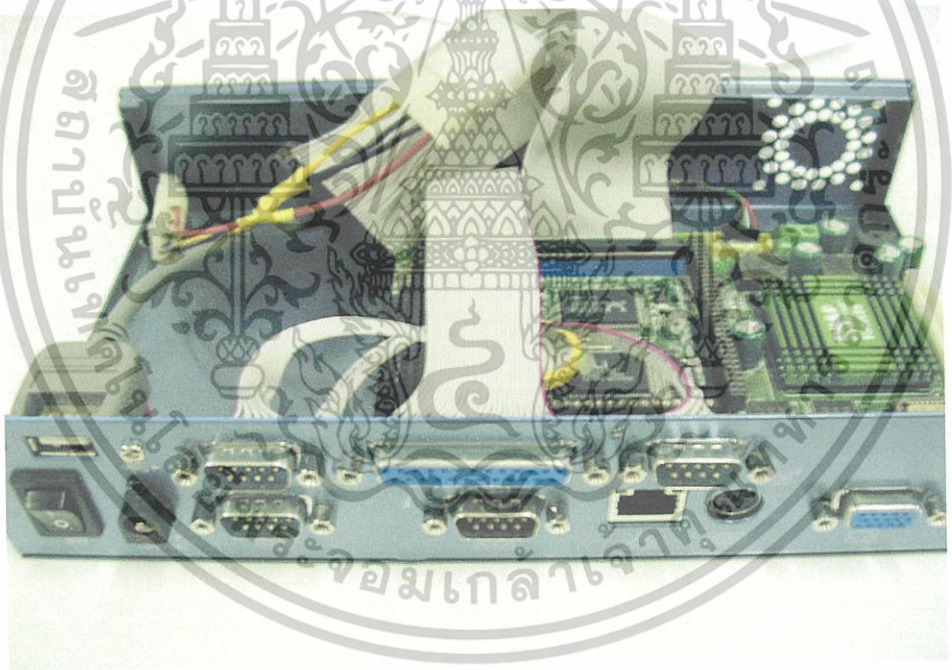
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.2.1 วงจรแหล่งจ่ายไฟ

วงจรแหล่งจ่ายไฟจะใช้ในการจ่ายไฟฟ้าให้กับทุกวงจรในระบบโดยวงจรแหล่งจ่ายไฟจะทำการแปลงไฟฟ้ากระแสสลับ 220 โวลต์ให้มาเป็นไฟฟ้ากระแสตรง 5 โวลต์และ 12 โวลต์ ไฟ 5 โวลต์จะจ่ายให้กับไมโครคอนโทลเลอร์และวงจรที่ใช้ในการติดต่อกับระบบคอมพิวเตอร์ฝังตัว และไฟ 12 โวลต์จะใช้จ่ายให้กับวงจรควบคุมวาล์วและปั๊ม

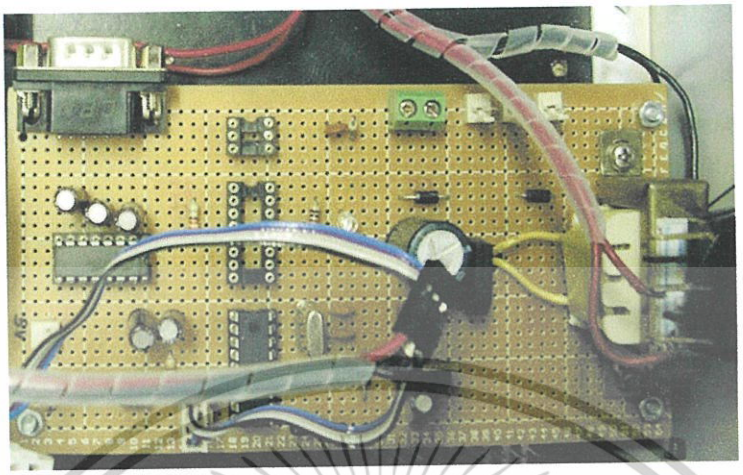
3.2.2 วงจรอินเตอร์เฟสกับระบบคอมพิวเตอร์ฝังตัว

วงจรอินเตอร์เฟสกับระบบคอมพิวเตอร์ฝังตัวจะใช้เป็นตัวกลางในการรับส่งข้อมูลระหว่างพอร์ตอนุกรมของคอมพิวเตอร์ฝังตัวกับ ไมโครคอนโทรเลอร์ PIC16F628 เพราะว่า สัญญาณที่ PIC16F628 สร้างออกมาจะเป็นแบบ 0 ถึง 5 โวลต์ ดังนั้นเราต้องให้สัญญาณดังกล่าว ผ่านไอซี MAX232 เพื่อแปลงเป็นสัญญาณแบบมาตรฐาน RS232



รูปที่ 3.14 ระบบคอมพิวเตอร์ฝังตัวที่ใช้

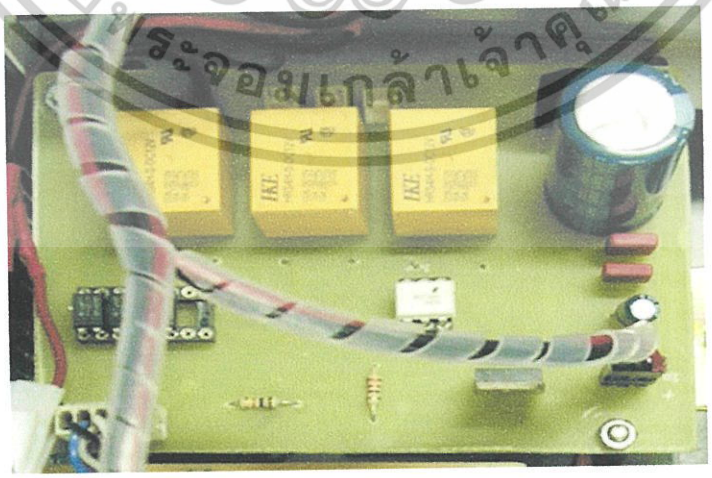
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.15 วงจรอินเตอร์เฟสกับระบบคอมพิวเตอร์ฝังตัว

3.2.3 วงจรควบคุมวาล์วและปั๊มโดยไมโครคอนโทรเลอร์ PIC16F628

วงจรควบคุมนี้จะใช้ไมโครคอนโทรเลอร์ตระกูล PIC ซึ่งจะใช้เบอร์ PIC16F628 โดยจะใช้ในการควบคุมรีเลย์และอ็อบโต้ไทรแอ็กและใช้ส่งข้อมูลไปยังระบบคอมพิวเตอร์ฝังตัวโดยผ่านวงจรอินเตอร์เฟสให้สามารถติดต่อสื่อสารรับและส่งข้อมูลจากพอร์ตอนุกรมโดยผ่านไอซี MAX232 มาเข้าที่ขารับส่งข้อมูลของไมโครคอนโทรเลอร์และต่อเอาท์พุตออกพอร์ตของไมโครคอนโทรเลอร์ โดยในการกำหนดสัญญาณความเร็วสัญญาณนาฬิกาของไมโครคอนโทรเลอร์จะใช้ คริสตอลออสซิลเลเตอร์ เป็นตัวกำเนิดสัญญาณนาฬิกาให้กับไมโครคอนโทรเลอร์ เพื่อไปทำการควบคุมอุปกรณ์ต่างๆ



รูปที่ 3.16 วงจรควบคุมวาล์วและปั๊ม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 4

ผลการทดลอง

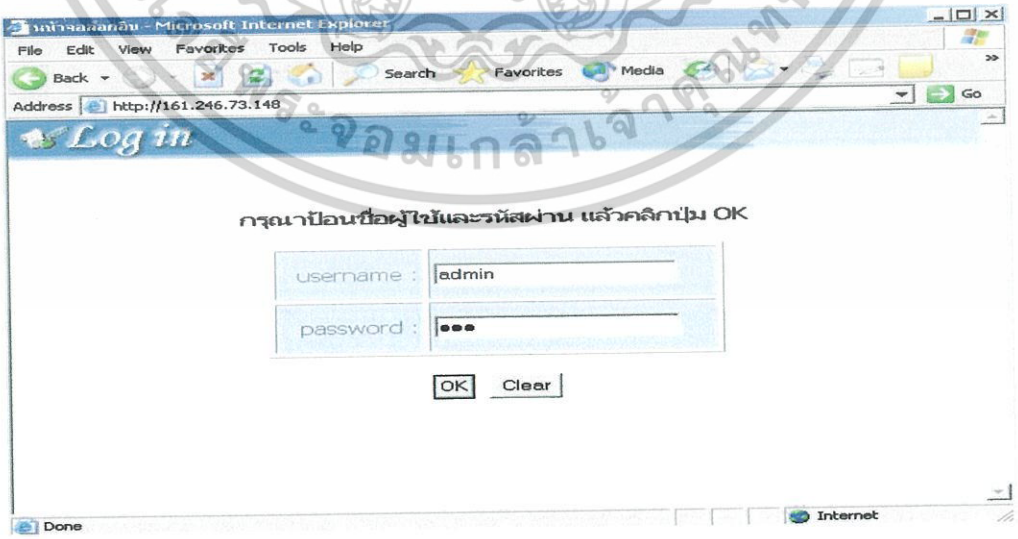
ในบทนี้จะกล่าวถึงลำดับขั้นตอนการทดลองและผลการทดลอง โดยการทดลองจะเป็น การทดลองการทำงานของโปรแกรมควบคุมปั๊มและวาล์วผ่านระบบอินเตอร์เน็ต ว่ามีผลการทำงานเป็นอย่างไร และการทำงานร่วมกันระหว่างโครงข่ายเซ็นเซอร์แบบไร้สายกับการควบคุมผ่านระบบอินเตอร์เน็ต

4.1 การทดลองการทำงานของโปรแกรมควบคุมปั๊มและวาล์วผ่านระบบอินเตอร์เน็ต

จุดประสงค์การทดลอง

1. ทดลองสั่งงานปั๊มและวาล์วให้ทำงาน
2. แสดงสถานะ การทำงานของปั๊มและวาล์ว
3. ทดลองสั่งงานปั๊มและวาล์วล่วงหน้า
4. แสดงฐานข้อมูลการตั้งเวลาสั่งงาน

ทำการรันโปรแกรม Web Brower แล้วเรียกที่ IP ของเว็บเซิร์ฟเวอร์ดังรูปที่ 4.1 เมื่อผู้ใช้ต้องการเข้าระบบจะต้องใส่ชื่อและรหัสผ่านให้ถูกต้องก่อน ในโครงงานนี้จะใช้รหัสผ่านเป็นระบบรักษาความปลอดภัย

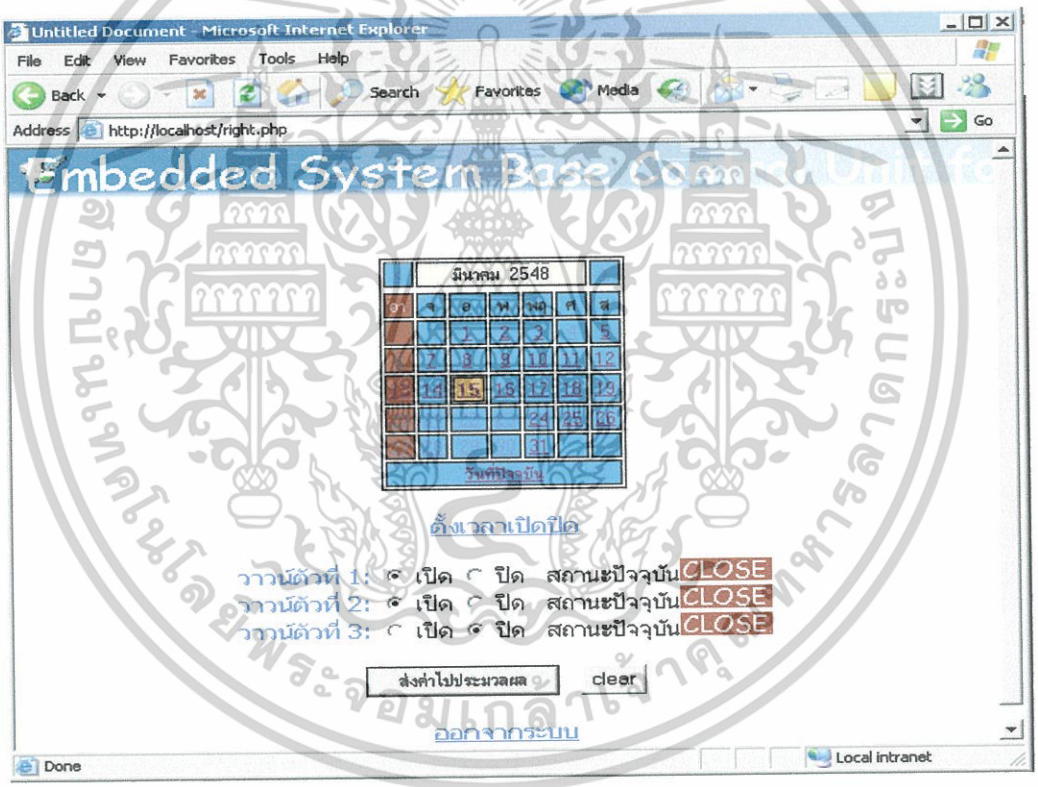


รูปที่ 4.1 การล็อกอินเข้าระบบควบคุมปั๊มและวาล์ว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เมื่อผู้ใช้ใส่ชื่อและรหัสถูกต้องจะเข้าสู่หน้าจอหลักซึ่งหน้าจอจะแสดงผลดังรูปที่ 4.2 เป็นหน้าจอหลักในการสั่งงานควบคุมปั๊มและวาล์วว่าให้ตัวใดทำงานหรือไม่ทำงานภายในช่วงเวลานั้น และสามารถสั่งงานล่วงหน้าได้โดยทำการเลือกวันทางปฏิทินหรือจะตั้งเวลาเปิดปิดแล้วแต่ผู้ใช้กำหนดว่าให้เปิดทำงานวันใดเวลาใดและจะปิดวันใดเวลาใดซึ่งข้อมูลการสั่งงานเปิดปิดล่วงหน้าจะไปเก็บอยู่ในฐานข้อมูลซึ่งจะแสดงในส่วนถัดไป และในหน้าจอนี้จะมีส่วนของการแสดงสถานะของวาล์วว่าตัวใดเปิดหรือปิดอยู่ ซึ่งจะเป็นการตรวจสอบได้ว่าค่าที่เราได้สั่งงานไปนั้นระบบได้ทำงานตามที่เราได้สั่งงานไปหรือไม่

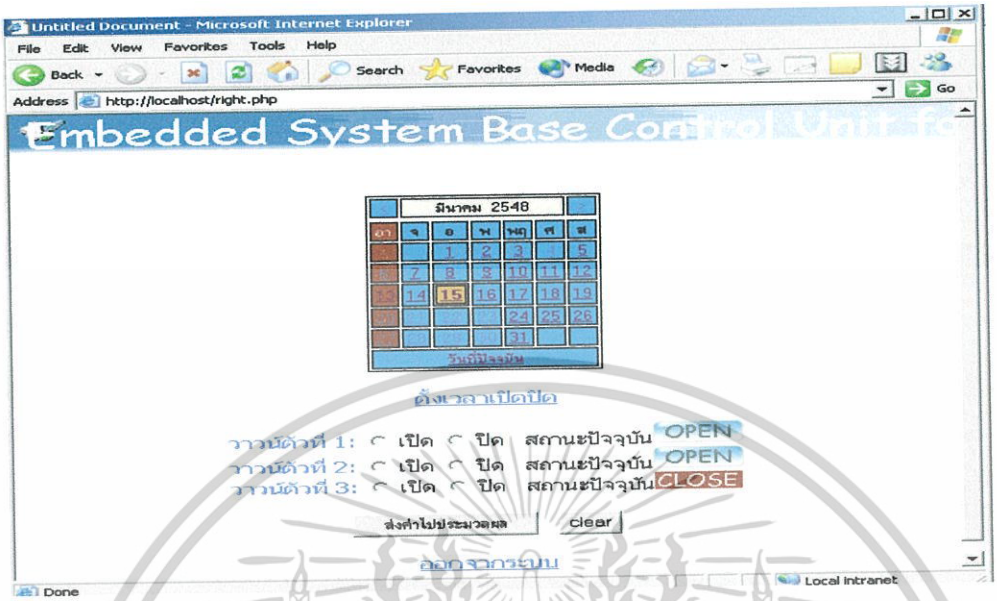
โดยในหน้านี้เราได้ทำการทดสอบเลือกให้วาล์วตัวที่ 1 ตัวที่ 2 เปิด และวาล์วตัวที่ 3 ปิดแล้วทำการกดปุ่มส่งค่าไปประมวลผล



รูปที่ 4.2 หน้าจอหลักของการควบคุมปั๊มผ่านทางอินเตอร์เน็ต

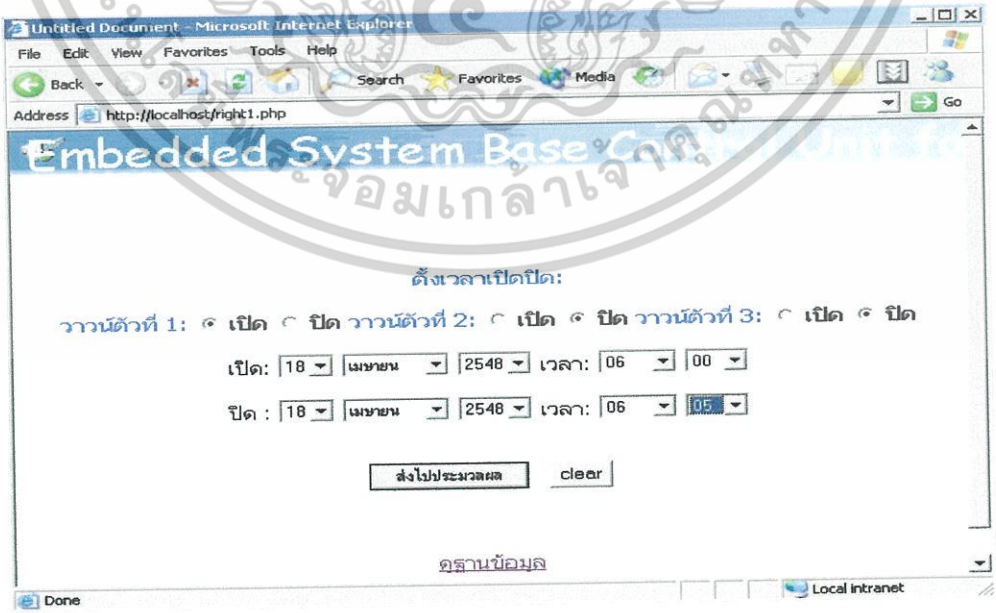
เมื่อเราได้ทำการสั่งงานให้วาล์วตัวใดทำงานหรือไม่ทำงานแล้ว วาล์วก็จะทำงานตามที่เราได้สั่งงานแล้วจะมีการแสดงผลของสถานะ การทำงานของวาล์วให้เราเห็นที่หน้าเว็บเพ็จ ว่าเราได้ทำการสั่งงานให้วาล์วตัวใดทำงานบ้าง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.3 ผลของการสั่งงานวาล์วตามที่ผู้ใช้ได้ทำการสั่งงาน

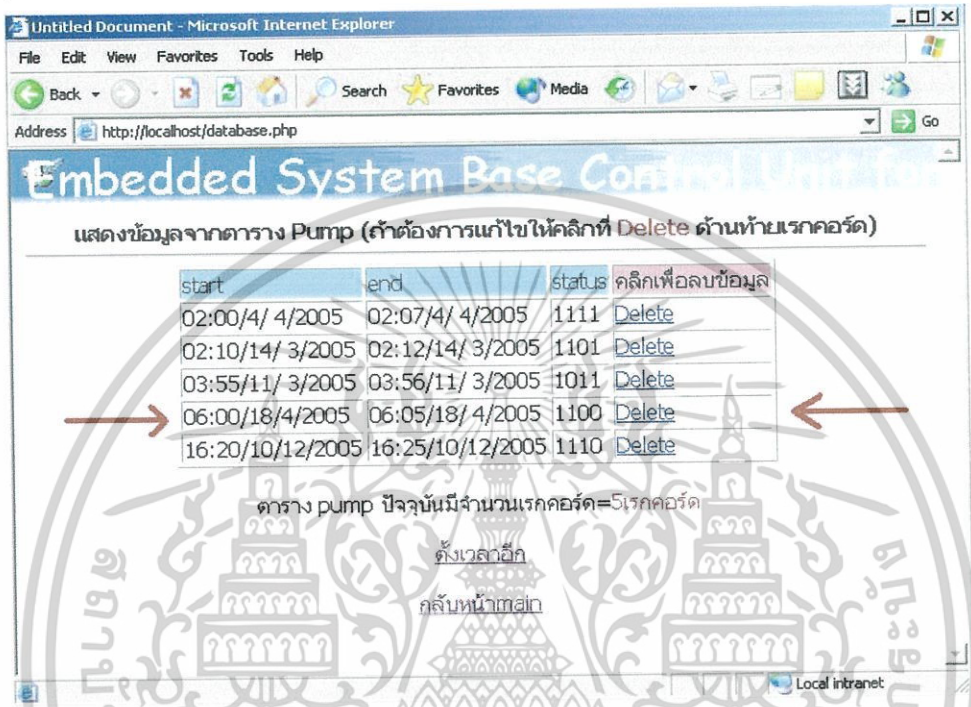
เมื่อเราได้ทำการเลือกตั้งเวลาเปิดปิดล่วงหน้าในหน้าหลัก ในรูปที่ 4.4 จะเป็นการแสดงการตั้งเวลาเปิดปิดวาล์วล่วงหน้าว่าให้ทำการเปิดปิดวันใดเวลาใด โดยจะมีการเลือกวันเวลาและวาล์วว่าให้ตัวใดเปิดหรือปิด โดยในที่นี้เราได้เลือกตั้งเวลาเปิดปิดล่วงหน้าให้วาล์วตัวที่ 1 เปิดและตัวที่ 2 และ 3 ปิด โดยตั้งเวลาให้ทำงานวันที่ 18 เมษายน 2548 และปิดทำงานวันที่ 18 เมษายน 2548



รูปที่ 4.4 การเลือกตั้งเวลาเปิดปิดวาล์วล่วงหน้า

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เมื่อเราได้ทำการตั้งเวลาเปิดปิดวาล์วล่วงหน้าแล้ว ในรูปที่ 4.5 จะแสดงข้อมูลการสั่งงานวาล์วล่วงหน้าที่เราได้สั่งงานซึ่งเก็บอยู่ในฐานข้อมูล



รูปที่ 4.5 ฐานข้อมูลการตั้งเวลาเปิดปิดวาล์วล่วงหน้า

จากที่เราได้ทำการสั่งงานตั้งเวลาเปิดปิดวาล์วล่วงหน้า ซึ่งเมื่อเวลาที่เครื่องเซิร์ฟเวอร์ตรงกับเวลาที่เราได้ตั้งเวลาไว้ วาล์วก็จะทำงานตามที่เราได้ตั้งเวลาสั่งงานไว้ล่วงหน้าว่าให้ทำงานเวลาใดและปิดเวลาใด ซึ่งข้อมูลการตั้งเวลาล่วงหน้าที่เก็บอยู่ในฐานข้อมูลจะเป็นการบันทึกการสั่งงานล่วงหน้าของผู้ใช้ว่าได้มีการตั้งเวลาสั่งงานวาล์วเมื่อใดบ้าง

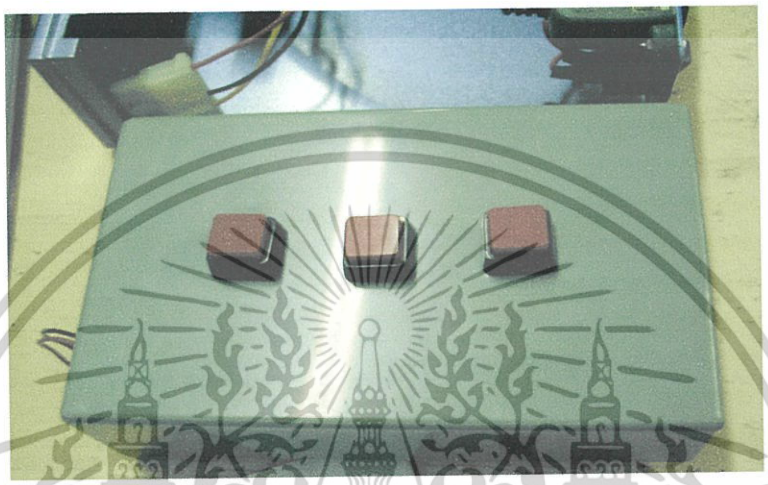
4.2 การทดลองการควบคุมวาล์วและปั๊มผ่านโครงข่ายเซ็นเซอร์แบบไร้สาย

จุดประสงค์การทดลอง

1. ทดลองการรับค่าจากโครงข่ายเซ็นเซอร์แบบไร้สายจำลอง
2. ทดลองการทำงานร่วมกันของโครงข่ายเซ็นเซอร์แบบไร้สายและการควบคุมจากระบบอินเตอร์เน็ต

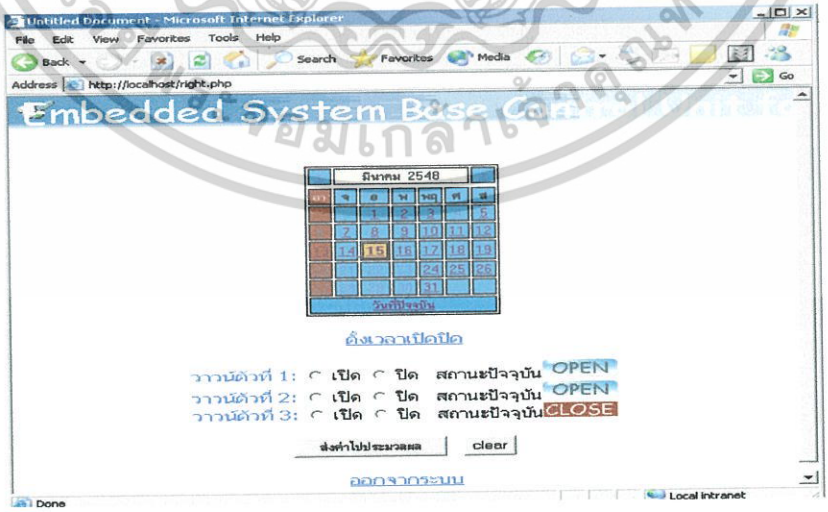
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ในส่วนการทดลองในส่วนนี้จะเป็นการจำลองการรับค่าจากโครงข่ายเซ็นเซอร์แบบไร้สายเข้าสู่ระบบ โดยโครงงานนี้ได้จำลองสร้างเซ็นเซอร์โดยใช้การสร้างโค้ดจากไมโครคอนโทรลเลอร์เพื่อรองรับโปรโตคอลของโครงข่ายเซ็นเซอร์แบบไร้สายจริง



รูปที่ 4.6 การจำลองการส่งค่าของโครงข่ายเซ็นเซอร์แบบไร้สาย

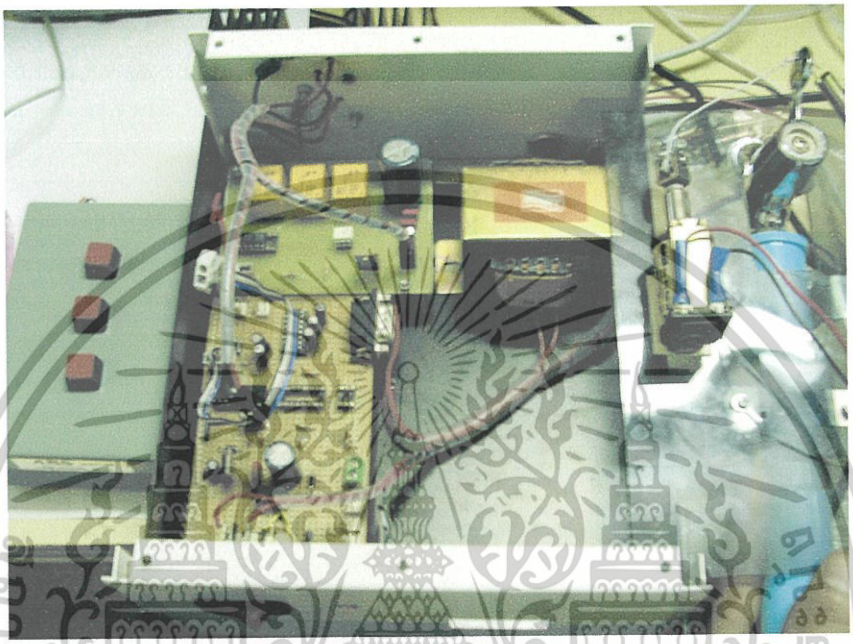
เมื่อเซ็นเซอร์ตัวที่หนึ่งและสองได้ทำการร้องขอเข้ามายังระบบคอมพิวเตอร์ฝังตัวให้วาล์วตัวที่หนึ่งและวาล์วตัวที่สองเปิด ตัวระบบคอมพิวเตอร์ฝังตัวจะทำการสั่งงานไปยังวงจรควบคุมวาล์วและปั๊ม เพื่อเปิดวาล์วตามที่เซ็นเซอร์ร้องขอมา



รูปที่ 4.7 ผลที่โครงข่ายเซ็นเซอร์แบบไร้สายร้องขอ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ในส่วนของอุปกรณ์โดยรวมทั้งหมดของโครงงานเมื่อระบบทำงานจริงโดยที่วาล์วและปั๊มจะทำการเปิด และแสดงค่าสถานะปัจจุบันผ่านทางระบบอินเตอร์เน็ตดังแสดงในรูปที่ 4.7 และ รูปที่ 4.8



รูปที่ 4.8 อุปกรณ์ทั้งหมดของโครงงาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 5
สรุปผลการทดลองและปัญหาที่พบ

5.1 สรุปผล

การที่ได้ศึกษาการทำงานของคอมพิวเตอร์ฝังตัวแล้วนั้น พบว่า การทำงานของคอมพิวเตอร์ฝังตัวนั้นมีประสิทธิภาพที่สูงเมื่อเทียบกับขนาดและผลที่ได้รับ การใช้งานเหมือนกับคอมพิวเตอร์ทั่วไปแต่ระบบปฏิบัติการนั้นค่อนข้างมีปัญหา กล่าวได้คือ จะไม่สามารถใช้งานได้ง่ายๆเหมือนคอมพิวเตอร์ธรรมดาทั่วไป ส่วนโปรแกรมที่จะสามารถใช้ได้ในคอมพิวเตอร์ฝังตัวนั้นค่อนข้างหาได้ค่อนข้างลำบาก และใช้งานยากพอสมควร

ในการใช้ระบบคอมพิวเตอร์ฝังตัวช่วยในการสื่อสารนั้น สามารถช่วยต้นทุนในระยะยาวได้ เนื่องจากมีประสิทธิภาพสูงและการดูแลรักษาค่อนข้างง่าย ลดค่าบำรุงรักษาอุปกรณ์ต่างๆ ค่าแรง ลดต้นทุนแปรได้ต่างๆ สามารถใช้งานได้หลากหลาย และมีความยืดหยุ่นในการใช้สูง มีขนาดเล็กและกินไฟน้อย

5.2 ปัญหาที่พบ

- ระบบปฏิบัติการของคอมพิวเตอร์ฝังตัวจะหาได้ยากและใช้งานยาก เพราะว่าจะไม่รองรับโปรแกรมบางตัวไม่เหมือนกับคอมพิวเตอร์
- ตัวซีพียูของคอมพิวเตอร์ฝังตัถ้าเราเขียนโปรแกรมที่ซับซ้อนมากจะทำให้ในการใช้งานจะช้า
- โหลดที่ใช้คือโซลินอยน์เมื่อต่อแล้วจะกินกระแสมากจึงต้องทำการแยกการจ่ายไฟระหว่างวงจรไมโครคอนโทรลเลอร์กับวงจรขับโหลดเพื่อมิให้เกิดการดึงกระแส เพราะว่าถ้าเกิดการดึงกระแสกันเกิดขึ้นจะทำให้เอาท์พุทของวงจรนั้นเกิดการผิดเพี้ยนไป

5.3 ข้อเสนอแนะ

การที่จะใช้ระบบคอมพิวเตอร์ฝังตัวนั้นจะต้องคำนึงถึงความจำเป็น ขนาด ของระบบที่ออกแบบและต้องศึกษาเรื่องของมาตรฐานของโปรโตคอลมามากๆ เพื่อให้ได้ประสิทธิภาพสูงสุด ควรจะศึกษาและทดลองโปรแกรมมากๆและใช้งานคอมพิวเตอร์ฝังตัวให้ได้ประโยชน์สูงสุด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บรรณานุกรม

[1] ณัฏฐพล วงศ์สุนทรชัย , ชัยวัฒน์ ลิ้มพรจิตรวิไล , “เรียนรู้และปฏิบัติการไมโครคอนโทรลเลอร์” , กรุงเทพมหานคร , บ.อินโนเวตีฟ เอ็กเพอริเมนต์ จำกัด

[2] ประจิน พลังสันติสุข , “เรียนรู้และใช้งาน CCS C คอมไพเลอร์” , กรุงเทพมหานคร , บ.อินโนเวตีฟ เอ็กเพอริเมนต์ จำกัด

[3] จำลอง ครูอุตสาหะ , “VB.NET ฉบับโปรแกรมเมอร์ ”, กรุงเทพมหานคร , บ.เคทีพี คอมพ์ แอนด์ คอนซัลท์ จำกัด

[4] ไพศาล โมลิสกุลมงคล , พัฒนา Web Database ด้วย PHP ”, กรุงเทพมหานคร , บ.ดวงกมลสมัย จำกัด

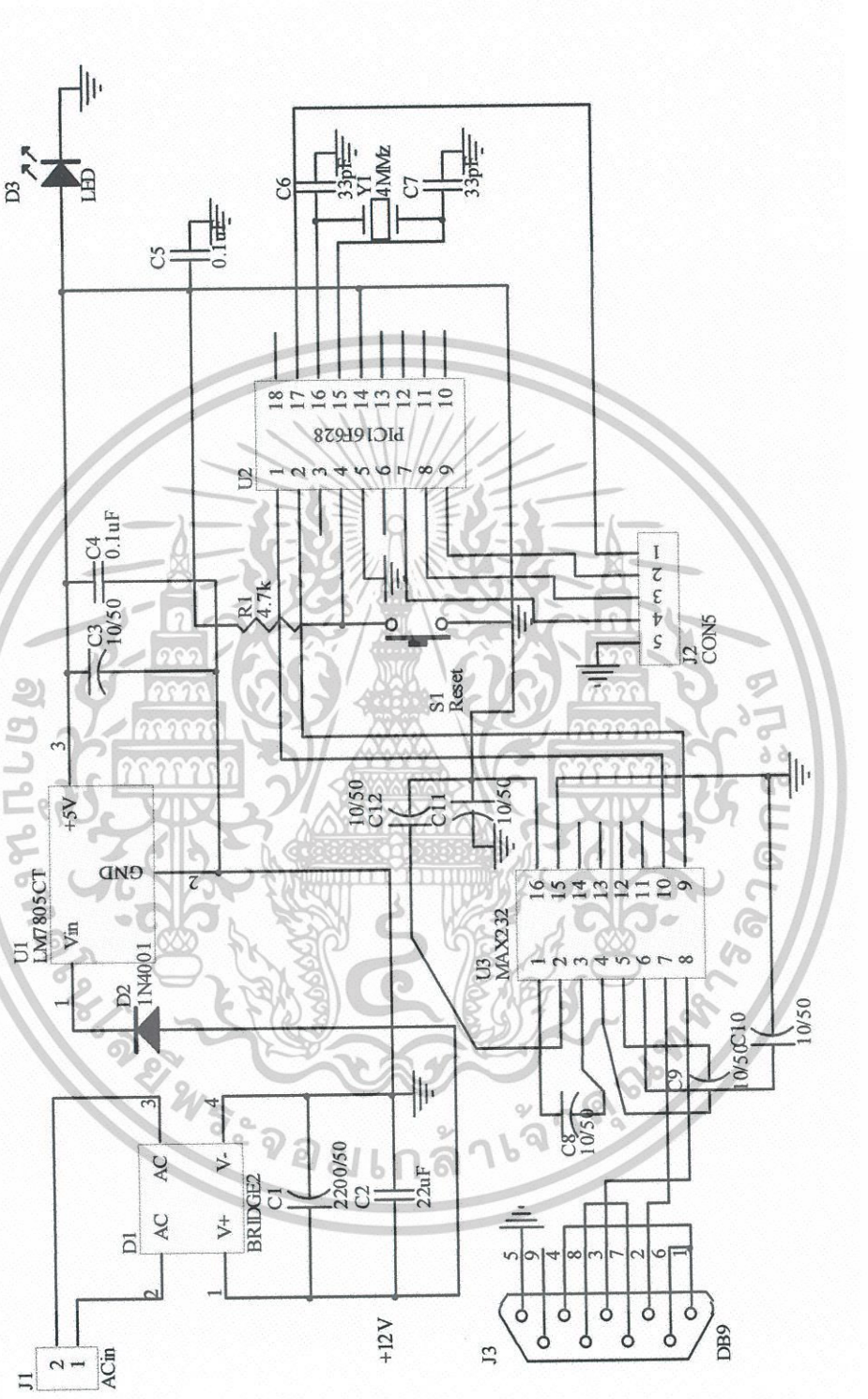
[5] การเขียนโปรแกรมติดต่อกับอุปกรณ์ภายนอก http://www.thaiio.com

[6] ระบบคอมพิวเตอร์ฝังตัว http://www.tesa.or.th

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

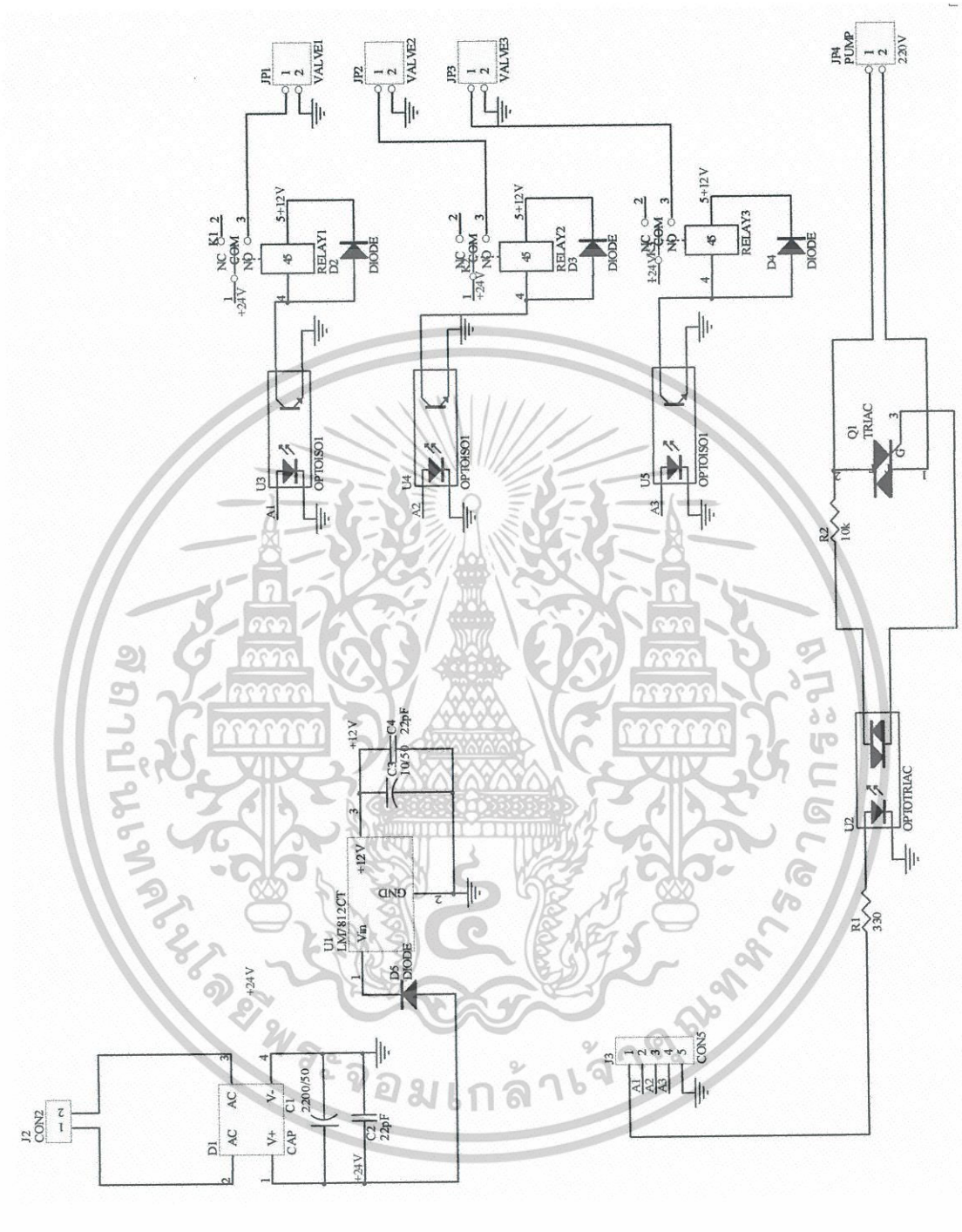
ภาคผนวก ก. วงจรที่ใช้งาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปวงจรติดต่อกับระบบคอมพิวเตอร์ฝังตัว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

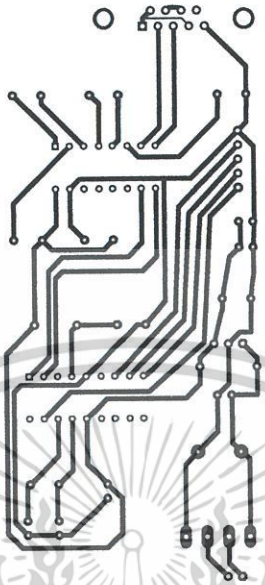


รูปวงจรควบคุมวาล์วและปั๊ม

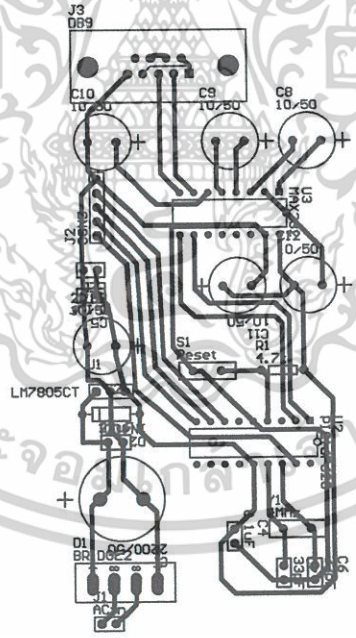
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ข. ลายพิมพ์วงจรและการจัดวางอุปกรณ์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

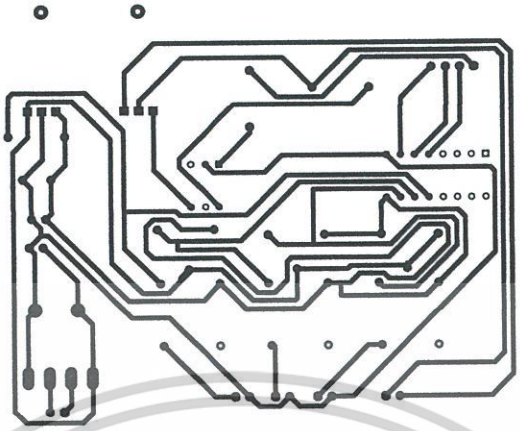


รูปลายวงจรติดต่อกับระบบคอมพิวเตอร์ฝังตัว

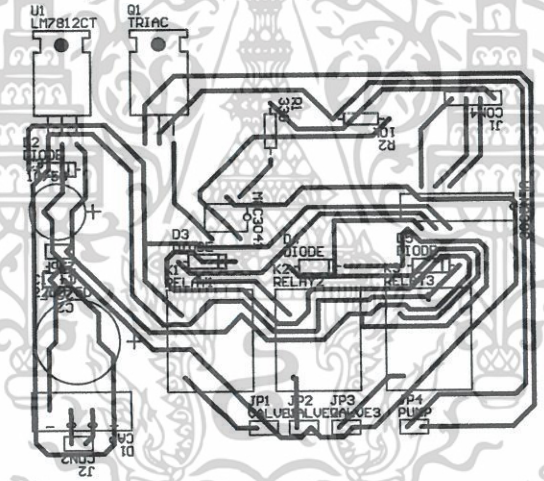


รูปการจัดวางอุปกรณ์วงจรติดต่อกับระบบคอมพิวเตอร์ฝังตัว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปลายวงจรควบคุมวาล์วและปั๊ม



รูปการวางอุปกรณ์วงจรควบคุมวาล์วและปั๊ม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ค. โปรแกรมที่ใช้งาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ซอร์สโปรแกรมหน้าเมนูหลักของเว็บเพจ

```
<?php
ob_start();
include "admin_secret.php";
if (strcmp($adpwd, $adminpage) != 0) {
header("location: index.php");
/*echo "<meta http-equiv=\"refresh\"
content=\"0;URL=index.php\">";*/
}
?>
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">
<html>
<head>
<title>Untitled Document</title>
<meta http-equiv="Content-Type" content="text/html; charset=windows-874">
<style type="text/css">
  .today  { font-family: ms sans serif; font-size: 10pt;
            font-weight: bold; background-color: #FF9933;
            color: #FFFFFF; border: 3 double #000000; }
  .sunday { font-family: ms sans serif; font-size: 10pt;
            background-color: #FF0000; color: #FFFFFF; }
  .norm   { font-family: ms sans serif; font-size: 10pt;
            background-color: #33CCFF; color: #000000; }
</style></head>
<body bgcolor="#CCCCCC" background="bg_login1.gif">
<font color="#CCCCCC"></font> <font color="#FFFFFF"></font> <font color="#FFFFFF"></font>
<font color="#FFFFFF"></font> <font color="#FFFFFF"></font>
<?php
/* $diffHour และ $diffMinute คือตัวแปรที่ใช้เก็บจำนวนชั่วโมงและจำนวนนาทีที่
   แตกต่างกันระหว่างเครื่องไคลเอนต์กับเครื่องเซิร์ฟเวอร์ ตามลำดับ เช่นถ้าเวลาของ
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
  เครื่องไคลเอ็นต์เร็วกว่าเวลาของเครื่องเซิร์ฟเวอร์ 11 ชั่วโมง 15 นาที ก็ให้กำหนด
  $diffHour เป็น 11 และกำหนด $diffMinute เป็น 15 ในที่นี้ผู้เขียนถือว่า
  เครื่องเซิร์ฟเวอร์กับเครื่องไคลเอ็นต์มีเวลาตรงกัน */
$diffHour = 0;
$diffMinute = 0;
if ($dfMonth == "") {
 /* ถ้าไม่มีการระบุให้แสดงปฏิทินของเดือนใดเดือนหนึ่ง เราจะแสดงปฏิทินของเดือน
   ปัจจุบันตามเวลาในเครื่องไคลเอ็นต์ โดยใช้ฟังก์ชั่น getdate() สร้างวันที่/เวลา
   ปัจจุบันของเครื่องไคลเอ็นต์เก็บไว้ในตัวแปร $calTime ซึ่งฟังก์ชั่นนี้จะคืนค่ากลับมา
   เป็นอาร์เรย์ */
 $calTime = getdate(date(mktime(date("H") + $diffHour,
  date("i") + $diffMinute)));
 $today = $calTime["mday"];    //วันที่
 $month = $calTime["mon"];     //เดือน
 $year = $calTime["year"];     //ปี
}
else {
 /* กรณีที่ระบุให้แสดงปฏิทินของเดือน/ปีหนึ่งๆนั้น จะมีการส่งตัวแปร $today,
   $dfMonth และ $dfYear ผ่านมาทาง query string ด้วย */
 if ($dfMonth == 0) {
  /* ถ้าตัวแปร $dfMonth เป็น 0 เราจะแสดงปฏิทินของเดือนธันวาคมของปีที่น้อย
    กว่าปีที่กำลังแสดงอยู่ */
  $dfMonth = 12;
  $dfYear = $dfYear - 1;
 }
 elseif ($dfMonth == 13) {
  /* ถ้าตัวแปร $dfMonth เป็น 13 เราจะแสดงปฏิทินของเดือนมกราคมของปีที่มาก
    กว่าปีที่กำลังแสดงอยู่ */
  $dfMonth = 1;
  $dfYear = $dfYear + 1;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
}
//สร้างวัน/เวลาของเดือนและปีที่ผู้ใช้ระบุ เก็บไว้ในตัวแปร $calTime
$calTime = getdate(date(mktime((date("H") + $diffHour),
  (date("i") + $diffMinute), 0, $dfMonth, $today, $dfYear)));
$today = $calTime["mday"];    //วันที่
$month = $calTime["mon"];     //เดือน
$year = $calTime["year"];     //ปี
}
/* เรียกฟังก์ชั่น LastDay() ซึ่งเป็นฟังก์ชั่นที่เราสร้างขึ้นมาเอง เพื่อหา "จำนวนวัน"
  ของเดือนและปีที่จะแสดงปฏิทิน โดยเก็บไว้ในตัวแปร $LDay */
$LDay = LastDay($month, $year);
//เก็บ timestamp ของวันที่ 1 ของเดือนที่จะแสดงปฏิทิน ไว้ในตัวแปร $FTime
$FTime = getdate(date(mktime(0, 0, 0, $month, 1, $year)));
//เก็บ "วันในสัปดาห์" (จันทร์, อังคาร ฯลฯ) ของวันที่ 1 ของเดือนไว้ในตัวแปร $wday
$wday = $FTime["wday"];
//สร้างตัวแปรชนิดอาร์เรย์เก็บชื่อเดือนภาษาไทย
$thmonthname = array("มกราคม", "กุมภาพันธ์", "มีนาคม", "เมษายน",
 "พฤษภาคม", "มิถุนายน", "กรกฎาคม", "สิงหาคม", "กันยายน", "ตุลาคม",
 "พฤศจิกายน", "ธันวาคม");
/* ฟังก์ชั่น LastDay() ใช้สำหรับหาวันที่สุดท้ายของเดือน/ปีที่ระบุ
  หรือกล่าวอีกนัยหนึ่งคือหาว่าเดือน/ปีที่ระบุนั้นมีกี่วัน */
function LastDay($m, $y) {
 for ($i=29; $i<=32; $i++) {
  if (checkdate($m, $i, $y) == 0) {
   return $i - 1;
  }
 }
}
?>
<p align="center"> </p>
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
<p align="center"> </p>
<div align="center">
 <table border="1" bordercolor="black" width="168">
  <tr class="norm">
   <td width="24" align="center"> <a href="<?php echo $PHP_SELF; ?>
?today=<?php echo $today; ?>
&dfMonth=<?php echo ($month - 1) ?>
&dfYear=<? echo $year; ?>">&lt;</a> </td>
    <td width="120" align="center" colspan="5" bgcolor="#F9F4DD"> <?php echo
$thmonthname[$month - 1]; ?> 
     <?php echo ($year + 543); ?> </td>
    <td width="24" align="center"> <a href="<?php echo $PHP_SELF; ?>
?today=<?php echo $today; ?>
&dfMonth=<?php echo ($month + 1); ?>
&dfYear=<?php echo $year; ?>">&gt;</a> </td>
   <tr>
   <tr>
    <td width="24" align="center" class="sunday">อา</td>
    <td width="24" align="center" class="norm">จ</td>
    <td width="24" align="center" class="norm">อ</td>
    <td width="24" align="center" class="norm">พ</td>
    <td width="24" align="center" class="norm">พฤ</td>
    <td width="24" align="center" class="norm">ศ</td>
    <td width="24" align="center" class="norm">ส</td>
   </tr>
   <?php
  $iday = 1;
  //แสดงแถวแรกของปฏิทิน
  for ($i=0; $i<=6; $i++) {
   if ($i < $wday) {  //แสดงเซลล์ว่างก่อนวันที่ 1 ของเดือน
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
 if ($i == 0) {    //กรณีที่เป็นวันอาทิตย์
  echo "<td width=\"24\" align=\"center\" class=\"sunday\"> </td>\n";
 }
 else {          //กรณีที่เป็นวันอื่นๆที่ไม่ใช่วันอาทิตย์
  echo "<td width=\"24\" align=\"center\" class=\"norm\"> </td>\n";
 }
}
else {          //แสดงวันที่ในแถวแรกของปฏิทิน
 if ($i == 0 && ($iday != $today)) {
  //กรณีที่เป็นวันอาทิตย์ และไม่ใช่วันปัจจุบัน
  echo "<td width=\"24\" align=\"center\" class=\"sunday\"><a
href=\"calendar.php?date=$iday&amp;month=$month&amp;year=$year\" >$iday</a></td>\n";
 }
 elseif ($iday == $today) { //กรณีที่เป็นวันปัจจุบัน
  echo "<td width=\"24\" align=\"center\" class=\"today\"><a
href=\"calendar.php?date=$iday&amp;month=$month&amp;year=$year\" > $iday</a></td>\n";
 }
 else {
  echo "<td width=\"24\" align=\"center\" class=\"norm\"><a
href=\"calendar.php?date=$iday&amp;month=$month&amp;year=$year\" > $iday</a></td>\n";
 }
 $iday++;
 }
}
//แสดงแถวที่เหลือของปฏิทิน (หลังจากแสดงแถวแรกไปแล้ว จะเหลืออย่างมาก 5 แถว)
for ($j=0; $j<=4; $j++) {
 if ($iday <= $Lday) {
  echo "<tr>\n";
  for ($i=0; $i<=6; $i++) {
   if ($iday <= $Lday) {
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
      if ($i == 0 && ($iday != $today)) {
        echo "<td width=\"24\" align=\"center\" class=\"sunday\"><a
href=\"calendar.php?date=$iday&amp;month=$month&amp;year=$year\" >$iday</a></td>\n";
      }
      elseif ($i == 0 && ($iday == $today)) {
        echo "<td width=\"24\" align=\"center\" class=\"today\"><a
href=\"calendar.php?date=$iday&amp;month=$month&amp;year=$year\" > $iday</a></td>\n";
      }
      elseif ($iday == $today) {
        echo "<td width=\"24\" align=\"center\" class=\"today\"><a
href=\"calendar.php?date=$iday&amp;month=$month&amp;year=$year\" > $iday</a></td>\n";
      }
      else {
        echo "<td width=\"24\" align=\"center\" class=\"norm\"><a
href=\"calendar.php?date=$iday&amp;month=$month&amp;year=$year\" > $iday</a></td>\n";
      }
      $iday++;
    }
    else {
      echo "<td width=\"24\" align=\"center\" class=\"norm\"> </td>\n";
    }
  }
  echo "</tr>\n";
 }
 else {
  break;
 }
}
?>
   <tr class="norm">
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    <td align="center" width="168" colspan="7"> <a href="<?php echo $PHP_SELF; ?>">วันที่ปัจจุบัน
</a></td>
   </tr>
  </table>
</div>
<p align="center"><a href="right1.php"><strong><font color="#0000FF">ตั้งเวลาเปิดปิดล่วงหน้า
</font></strong></a>
<p align="center">
  <?php
  $k=null;
  if($status1!=$k){
 $filename = "c:/internet.txt";
 $filenum = fopen($filename, "w");
 $i=1;
 fputs($filenum,"$i$status1$status2$status3");
 fclose($filenum);
 }
 $l=null;
  if($manual!=$l){
 $filename1 = "c:/manual.txt";
 $filenum1 = fopen($filename1, "w");
 $i=1;
 fputs($filenum1,"$manual");
 fclose($filenum1);
 }
 ?>
  <?php
 $fileread = fopen("c:/status.txt","r");
 fseek($fileread,1);
 $data1=fgets($fileread,2);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
//echo"ข้อความคือ ".$data1." <Br>";
$data2=fgets($fileread,2);
$data3=fgets($fileread,2);
fclose($fileread);
switch($data1)
{
case "0";
$a="close2.gif";
$d="close";
break;
case "1";
$a="open.gif";
$d="open";
break;
}
switch($data2)
{
case "0";
$b="close2.gif";
$e="close";
break;
case "1";
$b="open.gif";
$e="open";
break;
}
switch($data3)
{
case "0";
$c="close2.gif";
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
$f="close";
break;
case "1";
$c="open.gif";
$f="open";
break;
}
?>
<form action="right.php" method="post" >
 <div align="center"><font color="#0000FF"><strong>วาวน์ตัวที่ 1:</strong></font>
  <input name="status1" type="radio" value="1" >
  เปิด
  <input name="status1" type="radio" value="0" >
  ปิด   <?php echo"สถานะปัจจุบัน"; echo"<img src=\"$a\" alt=\"สถานะ\" border=0>";
?><br>
  <font color="#0000FF"><strong>วาวน์ตัวที่ 2:</strong> </font>
  <input type="radio" name="status2" value="1" >
  เปิด
  <input name="status2" type="radio" value="0" >
  ปิด   <?php echo"สถานะปัจจุบัน"; echo"<img src=\"$b\" alt=\"สถานะ\" border=0>";
?><br>
  <font color="#0000FF"> <strong>วาวน์ตัวที่ 3:</strong> </font>
  <input type="radio" name="status3" value="1" >
  เปิด
  <input name="status3" type="radio" value="0" >
  ปิด   <?php echo"สถานะปัจจุบัน"; echo"<img src=\"$c\" alt=\"สถานะ\" border=0>";
?><br>
  <br>
  <input name="Submit" type="submit" value="ส่งค่าไปประมวลผล">
   
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
  <input type="reset" value="clear">
  <br>
 </div>
</form>
 <?php header('refresh: 5; url=/right.php' );  ?>
</div>
<form action="right.php" method="post" >
<div align="center">
<p><font color="#0000FF"><strong>AUTO
 <input name="manual" type="radio" value="0" >
 MANUAL </strong></font>
 <input name="manual" type="radio" value="1" >
</p>
<p>
 <input name="Submit" type="submit" value="Submit">
  
 <input type="reset" value="clear">
  </div>
 </form>
 <div align="center"></p> <a href="logout.php" target="_parent"><strong><font color="#0000FF">ออก
จากระบบ</font></strong></a><br>
 </p>
 </body>
 </html>
 <?php ob_end_flush(); ?>
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ซอร์สโปรแกรมไมโครคอนโทรลเลอร์

```
#include<16F628.h>
#define CLOCK_SP 4000000
#define TxD PIN_A2
#define RxD PIN_A3
#fuses NOWDT,NOPROTECT,NOLVP
#fuses HS
#use delay(clock=CLOCK_SP)
#use rs232(baud=9600, xmit=TxD, rcv=RxD)
#byte portb=6
main(){
char a;
    while(true){
      a=getc();
          switch(a){
       case 'A':output_low(PIN_A0);
                output_low(PIN_B1);
                output_low(PIN_B2);
                output_low(PIN_B3);
                printf("1000");
                break;
        case 'B':output_high(PIN_A0);
                output_low(PIN_B1);
                output_low(PIN_B2);
                output_high(PIN_B3);
                printf("1001");
                break;
         case 'C':output_high(PIN_A0);
                output_low(PIN_B1);
                output_high(PIN_B2);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        output_low(PIN_B3);
        printf("1010");
        break;
case 'D':output_high(PIN_A0);
        output_low(PIN_B1);
        output_high(PIN_B2);
        output_high(PIN_B3);
        printf("1011");
        break;
case 'E':output_high(PIN_A0);
        output_high(PIN_B1);
        output_low(PIN_B2);
        output_low(PIN_B3);
        printf("1100");
        break;
case 'F':output_high(PIN_A0);
        output_high(PIN_B1);
        output_low(PIN_B2);
        output_high(PIN_B3);
        printf("1101");
        break;
case 'G':output_high(PIN_A0);
        output_high(PIN_B1);
        output_high(PIN_B2);
        output_low(PIN_B3);
        printf("1110");
        break;
case 'H':output_high(PIN_A0);
        output_high(PIN_B1);
        output_high(PIN_B2);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
            output_high(PIN_B3);
            printf("1111");
            break;
        default: output_low(PIN_A0);
            output_low(PIN_B1);
            output_low(PIN_B2);
            output_low(PIN_B3);
            printf("1000");
            break;
        }
    }
}
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้