

ระบบควบคุมไฟฟ้าผ่านเว็บ

Web-based electric control system for household



นายพศิน ชุมณี รหัส 46380017
นางสาวศิริพรรณ ชันธิกุล รหัส 46380043

15078519 e.2

ห้องสมุดคณะวิศวกรรมศาสตร์
วันที่รับ..... 15 ต.ค. 2550
เลขทะเบียน..... 5000071
เลขเรียกหนังสือ..... ๗๕
มหาวิทยาลัยเกษตรศาสตร์ พ.4795

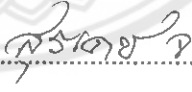
ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต
สาขาวิชาวิศวกรรมคอมพิวเตอร์ ภาควิชาวิศวกรรมไฟฟ้าและคอมพิวเตอร์
คณะวิศวกรรมศาสตร์ มหาวิทยาลัยเกษตรศาสตร์
ปีการศึกษา 2549

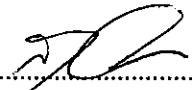


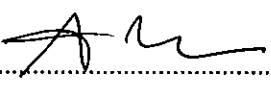
ใบรับรองโครงการวิศวกรรม

หัวข้อโครงการ	ระบบควบคุมไฟบ้านผ่านเว็บ Web-based electric control system for household
ผู้ดำเนินโครงการ	นายพศิน ชุมถี รหัส 46380017 นางสาวศิริพรรณ ขันธิกุล รหัส 46380043
อาจารย์ที่ปรึกษา	ดร. สุรเดช จิตประไพกุลศาล
สาขาวิชา	วิศวกรรมคอมพิวเตอร์
ภาควิชา	วิศวกรรมไฟฟ้าและคอมพิวเตอร์
ปีการศึกษา	2549

คณะวิศวกรรมศาสตร์ มหาวิทยาลัยบรจรัม อนุมัติให้โครงการฉบับนี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรวิศวกรรมศาสตรบัณฑิต สาขาวิชาวิศวกรรมคอมพิวเตอร์ คณะกรรมการสอบโครงการวิศวกรรม


.....ประธานกรรมการ
(ดร.สุรเดช จิตประไพกุลศาล)


.....กรรมการ
(ดร.อัครพันธ์ วงศ์กังแห)


.....กรรมการ
(ดร.สมยศ เกียรติวนิชวิไล)

หัวข้อโครงการ	ระบบควบคุมไฟบ้านผ่านเว็บ		
ผู้ดำเนินโครงการ	นายพศิน	ชุมณี	รหัส 46380017
	นางสาวศิริพรรณ	ขันธิกุล	รหัส 46380043
อาจารย์ที่ปรึกษา	คร. สุรเดช	จิตประไพกุลศาล	
สาขาวิชา	วิศวกรรมคอมพิวเตอร์		
ภาควิชา	วิศวกรรมไฟฟ้าและคอมพิวเตอร์		
ปีการศึกษา	2549		

บทคัดย่อ

โครงการนี้สร้างระบบต้นแบบสำหรับควบคุมการปิด-เปิดอุปกรณ์ไฟฟ้าผ่านเว็บ ระบบต้นแบบนี้แบ่งออกเป็น 2 ส่วนคือ Web-application และส่วนควบคุมอุปกรณ์ไฟฟ้า ผู้จัดทำได้ใช้หลักการของ minimal interface ในการออกแบบโปรแกรม กล่าวคือ โปรแกรมทั้ง 2 ส่วนติดต่อกันผ่านทาง text file ไฟล์เดียว ระบบของเราสนับสนุนการควบคุมการปิด-เปิดอุปกรณ์ไฟฟ้าทั้ง 2 ทาง ถ้าผู้ใช้ปิด-เปิดอุปกรณ์ไฟฟ้าจากทางเว็บระบบควบคุมก็จะปิด-เปิดอุปกรณ์ไฟฟ้าตาม ในทำนองเดียวกันถ้ามีการปิด-เปิดอุปกรณ์ไฟฟ้าจากระบบควบคุม ส่วน web-application ก็จะได้แสดงผลปรับเปลี่ยนตาม

Project title	Web-based electric control system for household		
Name	Mr Phasin Choomanee	ID. 46380017	
	Miss Siriphan Khantikul	ID. 46380043	
Project advisor	Dr. Suradet Jitprapaikulsarn		
Major	Computer Engineering		
Department	Electrical and Computer Engineering		
Academic year	2006		

ABSTRACT

This study demonstrates the application of the world-wide-web to control the electrical appliance. Our system divides into two components: 1) the web application and 2) the appliance control application. We take into account the principle of minimum interface during the design of the systems. Hence, these two components communicate with each other via a single text file. Our systems support bi-direction control, i.e. if a user turns on/off the virtual room on the web, the control application will turn on/off the actual appliance and vice ver sa.

กิตติกรรมประกาศ

โครงการวิศวกรรมคอมพิวเตอร์สำเร็จได้ด้วยดี เนื่องด้วยความอนุเคราะห์จากท่านอาจารย์
ที่ปรึกษา ดร.สุรเดช จิตประไพกุลศาล , ดร.อัครพันธ์ วงศ์กั้งแห, ดร.สมยศ เกียรติวนิชวิไล,
พี่เศรษฐา คั้งคำวานิช ที่กรุณาให้คำปรึกษา แนะนำวิธีการในการทำงาน ตลอดจนการตรวจสอบการ
ทำงานพร้อมทั้งเสนอแนวทางการแก้ไขตลอดระยะเวลาการทำโครงการ สุดท้ายต้อง
ขอขอบพระคุณอาจารย์ทุกท่านและเพื่อนๆที่ทุกคนที่ยังไม่ได้เอ่ยนามที่คอยสนับสนุนในการทำ
โครงการครั้งนี้



สารบัญ

หน้า

ระบบควบคุมไฟฟ้าผ่านเว็บ.....	
บทคัดย่อ.....	ก
ABSTRACT.....	ข
กิตติกรรมประกาศ.....	ค
สารบัญ.....	ง
สารบัญตาราง.....	ฉ
สารบัญรูป.....	ช

บทที่ 1 บทนำ

1.1 ความเป็นมาของโครงการ.....	1
1.2 วัตถุประสงค์ของโครงการ.....	1
1.3 ขอบข่ายของโครงข่าย.....	2
1.4 ขั้นตอนการดำเนินงาน.....	2
1.5 แผนการดำเนินงาน.....	3
1.6 ผลที่คาดว่าจะได้รับ.....	3
1.7 งบประมาณ.....	4

บทที่ 2 หลักการพื้นฐานของการสร้างระบบควบคุมไฟฟ้าผ่านเว็บ

2.1 องค์ประกอบของการสร้างระบบควบคุมไฟฟ้าผ่านเว็บ.....	5
2.2 อุปกรณ์ทางด้าน Hardware.....	5
2.3 องค์ประกอบทางด้าน Software.....	20
2.4 องค์ประกอบทางด้าน Network.....	24

บทที่ 3 วิธีการดำเนินการ

3.1 สถาปัตยกรรมของระบบ.....	28
3.2 แผนภาพการควบคุมวงจรไฟโดยควบคุมและแสดงผลผ่านทางเว็บ.....	29
3.3 การออกแบบโปรแกรมการควบคุมวงจรไฟโดยควบคุมและแสดงผลผ่านทางเว็บ.....	30
3.4 การเขียนโปรแกรมเพื่อควบคุมไฟฟ้าผ่านเว็บ.....	34
3.5 การควบคุมวงจรไฟและแสดงผลผ่านทาง โปรแกรมควบคุมทางฝั่งแม่ข่าย.....	38

สารบัญ (ต่อ)

หน้า

บทที่ 4 ผลการทดลอง

4.1 ผลการทดลองความปลอดภัยในส่วนของ Security (Secure Sockets Layer)	43
4.2 ผลการทดลองความปลอดภัยจากการ login เข้าหน้าเว็บ.....	45
4.3 ผลการทำงานของโปรแกรมในการปิด-เปิด ไฟในแต่ละห้องและแต่ละชั้น.....	47
4.4 ผลการทำงานของโปรแกรมที่ Interface กับฮาร์ดแวร์	50
4.5 ผลการทำงานการเชื่อมต่อกันระหว่างเว็บกับฮาร์ดแวร์.....	54

บทที่ 5 สรุปผล

5.1 สรุปผลการทดลอง.....	55
5.2 ปัญหาและแนวทางแก้ไข.....	56
5.3 แนวทางพัฒนาต่อ	56
5.4 ข้อเสนอแนะ.....	56

เอกสารอ้างอิง	58
---------------------	----

ภาคผนวก ก	59
-----------------	----

ประวัติผู้เขียน โครงการ	93
-------------------------------	----

สารบัญตาราง

ตารางที่	หน้า
2.1 บอครตของเครื่องคอมพิวเตอร์ PC.....	17
2.2 เปรียบเทียบบอครตเมื่อ SMOD = 0 และ SMOD = 1.....	18



สารบัญรูป

รูปที่	หน้า
2.1 บอร์ดทดลอง MCS-51.....	6
2.2 โครงสร้างภายในของ MCS-51.....	8
2.3 ขาต่างๆของ 8051.....	8
2.4 ความหมายแต่ละบิตของรีจิสเตอร์ควบคุม.....	11
2.5 บิตต่างๆของRegister SCON (Serial Port Control Register).....	12
2.6 การกำหนดบิต SM0 และ SM1 เพื่อกำหนดโหมดการทำงาน.....	12
2.7 Register PCON เพื่อกำหนดอัตราการรับส่งข้อมูล.....	14
2.8 รายละเอียดของ Com Port หรือ Serial Port.....	15
2.9 ภาพแสดงสัญญาณ RS232.....	15
2.10 RS232 LINE.....	15
2.11 บอร์ด EX-LED.....	16
2.12 การรับส่งข้อมูลแบบ simplex.....	17
3.1 สถาปัตยกรรมของระบบ (System Architecture).....	28
3.2 สถาปัตยกรรมของซอฟต์แวร์ (Software Architecture).....	29
3.3 แผนภาพแสดงการทำงานของโปรแกรมส่วนที่ติดต่อกับ Web-based.....	29
3.4 แผนภาพแสดงการทำงานของโปรแกรม SSL.....	33
3.5 แผนภาพแสดงการทำงานในส่วนของโปรแกรม web server.....	34
3.6 การทำงานของโปรแกรมในส่วนของการตรวจสอบสถานะของแต่ละห้อง.....	35
3.7 การทำงานของโปรแกรมการตรวจสอบสถานะของแต่ละห้องบนหน้าเว็บ.....	36
3.8 การทำงานของโปรแกรมการแสดงผลภาพบนเว็บ.....	36
3.9 การทำงานภาพรวมของหน้าเว็บที่ทำการติดต่อกับ Text file.....	37
3.10 โปรแกรมส่วนแม่ข่าย.....	38
3.11 กระบวนการติดต่อสื่อสาร.....	39
3.12 การกดปุ่ม ปิด-เปิด ดวงไฟแต่ละดวง.....	40
3.13 การตั้งเวลา ปิด-เปิด ดวงไฟ แต่ละชั้น.....	40
3.14 การอ่านข้อมูล text files ที่ได้จากเว็บ.....	41
3.15 การทำงานส่งค่า text file ให้บอร์ด MCS-51 โดยผ่านตัวควบคุมที่ฝั่งเว็บ.....	42
4.1 ภาพแสดงการทำงานของ SSL ก่อนเข้าหน้าแรกของเว็บ.....	43

สารบัญรูป (ต่อ)

รูปที่	หน้า
4.2 ภาพแสดงการทำงานของ SSL เมื่อเลือกการเข้ารหัส.....	44
4.3 หน้าแรกของเว็บเพจคือหน้า login	45
4.4 การทำงานของโปรแกรมหน้า login.....	46
4.5 สถานะของแต่ละห้องในบ้าน	47
4.6 การ ปิด – เปิด ไฟของแต่ละห้อง.....	48
4.7 สถานะของห้องเมื่อทำการปิดไฟในห้อง	48
4.8 สถานะเมื่อห้องถูกปิดไฟ	49
4.9 เป็นการแสดงสถานะของการเปลี่ยนแปลงเมื่อทำการ ปิด-เปิด ไฟทั้งชั้น	50
4.10 การเชื่อมต่อพอร์ตอนุกรม	50
4.11 โปรแกรมควบคุมฝั่งแม่ข่าย	51
4.12 การปิด-เปิดดวงไฟแต่ละดวง	51
4.13 สถานะของชั้น1ที่มีการเปลี่ยนแปลง.....	53
4.14 เป็นการแสดงสถานะของการตั้งเวลาทำการเปิดไฟทั้งชั้นของชั้น 1	53
4.15 ภาพการทำงานทั้งระบบของการควบคุมดวงไฟ.....	54

บทที่ 1

บทนำ

1.1 ความหมายของโครงการ

คนส่วนใหญ่ดำเนินชีวิตด้วยความรีบเร่ง เนื่องจากสภาวะภายนอกมากระทำ เช่น การรีบไปทำงาน การรีบไปต่างจังหวัด การทำงานนอกสถานที่ เป็นต้น ซึ่งบางครั้งรีบร้อนจนเปิดไฟฟ้าเครื่องใช้ไฟฟ้าทิ้งไว้ ทำให้เป็นการสิ้นเปลือง และอาจเกิดความเสียหายตามมา

เนื่องจากในปัจจุบัน การนำ Internet มาใช้ในการทำงานมีจำนวนเพิ่มมากขึ้น ไม่ว่าจะเป็นด้านต่างๆ เช่น การทำงาน การศึกษาหาความรู้เพิ่มเติม การทำธุรกิจ โดยผ่านสื่ออิเล็กทรอนิกส์ E-mail การรับส่งข้อมูลข่าวสาร เป็นต้น รวมถึงความมีประสิทธิภาพ และถูกต้องแม่นยำ

ที่ผู้พัฒนาได้เห็นความสำคัญของ Internet จึงได้พัฒนาโครงการในรูปแบบของ Web-based ขึ้น โดยเกิดแนวคิดที่จะควบคุมระบบ ไฟฟ้าภายในบ้านผ่าน Internet ขึ้น และยังช่วยอำนวยความสะดวกให้กับผู้ใช้ โดยการทำงานนั้นสามารถตรวจสอบได้ว่า มีไฟดวงไหนเปิดค้างไว้ สามารถมองเห็นรายละเอียดภายในบ้าน ได้อย่างชัดเจนว่ามีการเปิดใช้ไฟฟ้าตรงจุดไหนบ้าง และต้องการจะ ปิด - เปิด ไฟฟ้าตรงจุดไหนภายในบ้าน ได้ ก็สามารถแก้ไขได้อย่างรวดเร็ว

ในการสร้างระบบควบคุมไฟฟ้าภายในบ้าน ที่ผู้พัฒนาได้ศึกษาการทำงานของอุปกรณ์อิเล็กทรอนิกส์ที่ใช้ตรวจับการ ปิด - เปิด อุปกรณ์ไฟฟ้า, อุปกรณ์อิเล็กทรอนิกส์ที่ใช้ ปิด-เปิด อุปกรณ์ไฟฟ้าอัตโนมัติ, อุปกรณ์ที่ใช้เป็นตัวส่งสัญญาณให้กับคอมพิวเตอร์ เพื่อแสดงถึงสถานะการทำงานในขณะนั้น และการเชื่อมต่อผ่านทาง Internet

1.2 วัตถุประสงค์ของโครงการ

1. เพื่อสร้างความปลอดภัยและเพิ่มความสะดวกสบายให้แก่ผู้ใช้ไฟฟ้าได้มากขึ้น โดยผ่านทาง Internet ได้

2. นำเอาเทคโนโลยีในปัจจุบันมาประยุกต์ใช้กับชีวิตประจำวัน

3. สามารถที่จะควบคุมและจัดการระบบ ไฟฟ้าภายในบ้าน โดยผ่านทาง Internet

1.3 ขอบข่ายของโครงข่าย

โครงข่ายนี้มีเป้าหมายในการที่จะสร้างความปลอดภัยและควบคุมระบบไฟฟ้าภายในบ้านผ่านทาง Internet โดยมีหัวข้อที่จะศึกษาดังนี้

1. สร้างอุปกรณ์อินเตอร์เฟสระหว่าง Hardware กับ Internet เพื่อแสดงผลทาง Web-Based
2. ศึกษาเกี่ยวกับทฤษฎีต่างๆที่เกี่ยวข้อง ดังนี้ คือ
 - การเขียน โปรแกรมควบคุม Port input/output ของคอมพิวเตอร์
 - การเขียน โปรแกรมเพื่อติดต่อ Network และ Internet
3. จัดเตรียมอุปกรณ์และ โปรแกรมต่างๆที่ใช้ในการดำเนินงาน
4. สามารถควบคุมการปิด-เปิดของอุปกรณ์ไฟฟ้าบางชนิดภายในบ้านได้ เช่น หลอดไฟ พัดลม เป็นต้น

1.4 ขั้นตอนการดำเนินงาน

1. ศึกษาเกี่ยวกับทฤษฎีและหลักการในสิ่งต่างๆ เหล่านี้
 - หลักการและวิธีการ ใช้ฮาร์ดแวร์ MCS-51 และ LED
 - หลักการและวิธีการทำงานของ OpenSSL
 - การเขียน โปรแกรมเพื่อติดต่อกับ Text Files ที่เครื่องแม่ข่าย
 - การเขียน โปรแกรมเพื่อติดต่อกันระหว่าง โปรแกรมเว็บและ โปรแกรม VB
2. ออกแบบและพัฒนาโปรแกรม
3. ทดสอบโปรแกรม
4. ทำการปรับปรุงและแก้ไขโปรแกรม
5. วิเคราะห์การทดสอบพร้อมทั้งสรุปผล
6. จัดทำเป็นรูปเล่ม

1.5 แผนการดำเนินงาน

กิจกรรม	เดือน - ปี				
	ธ.ค. 48	ม.ค.49	ก.พ. 49	มี.ค.49	เม.ย. 49
1. ศึกษาและหาข้อมูลเกี่ยวกับทฤษฎีต่างๆที่เกี่ยวข้อง	←→				
2. จัดเตรียมอุปกรณ์และโปรแกรมที่ใช้ในการดำเนินงาน		←→			
3. ขั้นตอนการออกแบบและต่อวงจรด้าน Hardware			←→		
4. ทดสอบการทำงานของ Hardware				←→	
5. ขั้นตอนการออกแบบและเขียน โปรแกรมด้าน Software				←→	
6. ทดสอบการทำงานของ Software					←→
7. วิเคราะห์การทดลองและสรุปการทดลอง					←→
8. จัดทำรูปเล่มรายงาน					←→

1.6 ผลที่คาดว่าจะได้รับ

1. ความรู้และจัดการระบบไฟฟ้าภายในบ้าน โดยผ่าน Internet ได้อย่างมีประสิทธิภาพ
2. รู้จักการแก้ปัญหาเฉพาะหน้าจากการทำโครงงานนี้ได้
3. รู้จักการใช้โปรแกรมเพื่อควบคุมการทำงานของคอมพิวเตอร์และ ไมโครคอนโทรลเลอร์

1.7 งบประมาณ

1. ค่าเอกสาร	1,000 บาท
2. อุปกรณ์และเครื่องมือ	2,000 บาท
3. ค่าใช้อื่นๆ เช่น ค่าจัดทำรูปเล่มรายงาน	500 บาท
รวมเป็นเงินทั้งสิ้น	<u>3,500 บาท</u>



บทที่ 2

หลักการพื้นฐานของการสร้างระบบควบคุมไฟฟ้าผ่านเว็บ

ในการสร้างระบบควบคุมไฟฟ้าภายในบ้านผ่านเว็บ ทีมผู้จัดทำได้ศึกษาการทำงานของอุปกรณ์อิเล็กทรอนิกส์ และส่วนประกอบอื่น ๆ ที่จะนำมาใช้ในการควบคุมการ ปิด-เปิด ไฟฟ้าผ่านเว็บ โดยมีทฤษฎีและหลักการดังต่อไปนี้

2.1 องค์ประกอบของการสร้างระบบควบคุมไฟฟ้าผ่านเว็บ

การออกแบบระบบควบคุมไฟฟ้าผ่านเว็บ สามารถแบ่งองค์ประกอบออกเป็น 3 ส่วน ดังนี้

1. Hardware มีอุปกรณ์ที่ใช้ได้แก่ บอร์ดทดลองไมโครคอนโทรลเลอร์ MCS-51, Serial Port (RS-232), บอร์ด LED เป็นต้น
2. Software ที่ใช้ในการควบคุม Hardware จะเป็นการเขียนโปรแกรมให้สามารถติดต่อกับ Hardware ได้โดยโปรแกรมที่ใช้ ได้แก่ Visual Basic Programming และ Assembly Language
3. ภาษาที่ใช้ในการเขียน Web เพื่อควบคุมการ ปิด-เปิด ไฟฟ้าและแสดงผลออกทางจอภาพ ได้แก่ PHP และ HTML

2.2 อุปกรณ์ทางด้าน Hardware

อุปกรณ์ที่ใช้ในด้าน Hardware ได้แก่ ไมโครคอนโทรลเลอร์ MCS-51 อุปกรณ์ที่ใช้ติดต่อเข้ากับคอมพิวเตอร์ เช่น Serial Port (RS-232) รวมถึงการติดต่อสื่อสารข้อมูลระหว่างคอมพิวเตอร์กับบอร์ดไมโครคอนโทรลเลอร์ MCS-51 เป็นต้น

2.2.1 Microcontroller Single Board

Single Board “START-C51” VERSION 3.0 มาช่วยในการประมวลผลการทำงาน โดยใช้ MCS-51 ในการควบคุม การทำงานของ Hardware โดย “START-C51” VERSION 3.0 เป็น Single Board เพื่อการเรียนรู้เกี่ยวกับไมโครคอนโทรลเลอร์เบอร์ 89C51RD2 จาก Philips ในตระกูล MCS-51 ซึ่งเป็นตระกูลที่มีทางเลือกจากผู้ผลิตชิพมากที่สุดในโลก

ภาพแสดงตำแหน่งต่างๆ ของบอร์ด



รูปที่ 2.1 บอร์ดทดลอง MCS-51

คุณสมบัติทั่วไปของบอร์ด

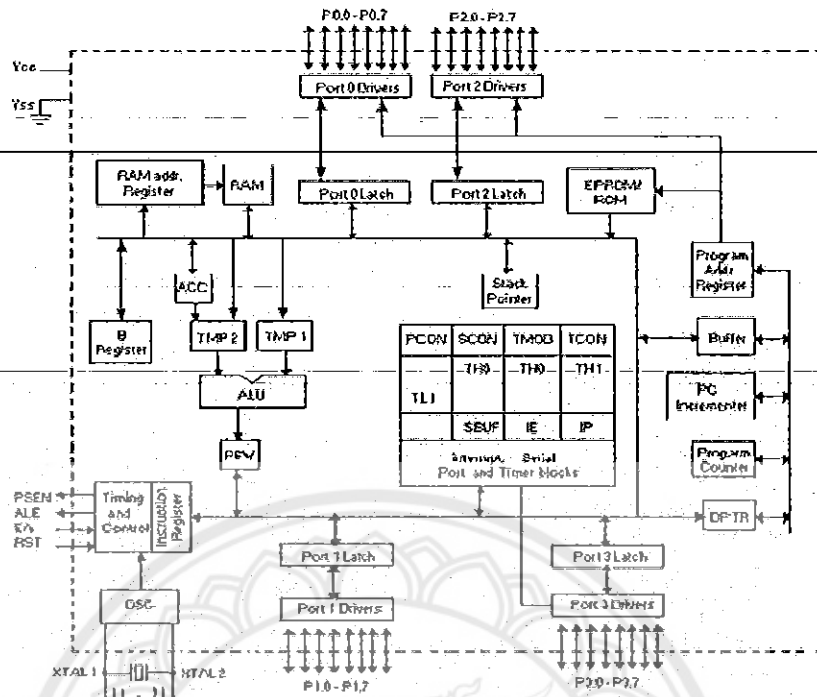
MCU	89C51RD2 MICROCONTROLLER (SREED x 2)
CLOCK	11.0592 MHz.
INTERNAL RAM	256 Bytes
PROGRAM MEMORY	64 Kbytes FLASH MEMORY 000H-FFFFH
DATA MEMORY	768 Bytes 0000H-02FFH (89C51RD2) 0/32 Kbytes 0000H-7FFFH (6264/62256) (Option)
DISPLAY	6 DIGITS 7-SEGMENT LED (RED) 8 RED LED (P1.0-P1.7) 4 YELLOW LED (P3.2, P3.3, P3.4, P3.5) 1 USER GREEN LED 1 POWER RED LED 1 GREEN LED MONITOR RX (RS232, RS485) 1 RED LED MONITOR TX (RS232, RS485) 1 RED LED MONITOR RS485 DIRECTION 1 GREEN LED MONITOR LOW BATT
KEYBOARD	20 KEYS (HEX=16, OPERATE=4) 3 KEYS HARDWARE (RESENT, INT1, INT0)
JUMPER	2-WAYS PROGRAMABLE (PROG/RUN) 2-WAYS SELECT RAM (32K/8K)
SOUND	1 BIT SPEAKER ON BOARD

RESET	DS1707 (RESET AND LOWBATT CHECK)	
ADC/DAC	PCF8591 ADC 8 BIT 4 INPUT/DAC 8 BIT 1 OUTPUT (Option)	
RTC	DS1307 (Option)	
RS232	MAX232	
RS485	75176(Option)	
CONNECTOR	16-PIN x 2	12B PORT
	16-PIN x 2	LCD PORT
	20-PIN x 2	IOE BUS PORT
	26-PIN x 1	24B PORT
	16-PIN x 1	8X8D PORT
	16-PIN x 1	4X8K PORT
	16-PIN x 1	4AD/DA PORT
	5 -PIN x 1	3B PORT
	3 -PIN x 2	SERIAL PORT(RS232)
	2 -PIN x 1	SERIAL PORT (RS485)
	2 -PIN x 1	TOOL INPUT/OUTPUT PORT
	2 -PIN x 1	5 VDC SUPPLY
	OTHER	4 BIT DIPSWITCH
	JACK 9 VDC SUPPLY	
	BATTERY 3V + BATTERY SOCKET (Option)	
	WORKING AREA PCB 2.5" x 4.3" (Option)	

2.2.2 ไมโครคอนโทรลเลอร์ตระกูล MCS-51

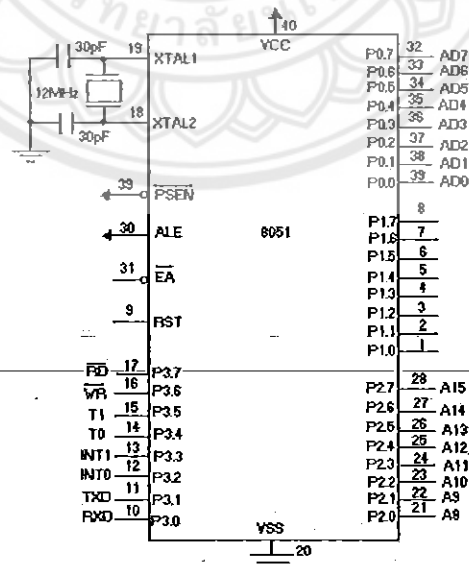
ไมโครคอนโทรลเลอร์ตระกูล MCS-51 เป็นไมโครคอนโทรลเลอร์ของบริษัท Intel ถูกผลิตขึ้นและได้มีการนำไปใช้งานอย่างแพร่หลายตั้งแต่ปี ค.ศ.1980 เป็นต้นมา และได้มีบริษัทต่างๆ หลายบริษัทได้รับลิขสิทธิ์ในการผลิตและจำหน่ายและได้รับพัฒนาเพื่อเพิ่มประสิทธิภาพ การทำงานในด้านต่างๆ มากขึ้นและหนึ่งในจำนวนนั้นก็มีบริษัท ATMEL ได้พัฒนาชิปตระกูล MCS-51 ขึ้นโดยใช้เทคโนโลยีของ Flash Memory ต่างจากของ Intel ที่ใช้ EPROM และรวมซึ่งข้อดีของ Flash Memory คือทำให้สามารถเขียนโปรแกรมและสามารถลบทิ้งลงไปได้ง่าย

ไมโครคอนโทรลเลอร์ตระกูล MCS-51 มีจำนวนมาก ขึ้นกับโครงสร้างภายใน บางเบอร์มีหน่วยความจำภายในเป็นแบบรวม บางเบอร์เป็นแบบอิพรวม บางเบอร์มีแรมภายใน 128 ไบท์



รูปที่ 2.2 แสดงโครงสร้างภายในของ MCS-51 [9]

ไอซี ไมโครคอนโทรลเลอร์ 8051 โครงสร้าง ไอซี เป็นแบบ DIP มีขาทั้งหมด 40 ขา โดยขาต่างๆ เป็นขาพอร์ทอินพุต, ขาพอร์ทเอาต์พุต, ขาสัญญาณควบคุม, ขาดำเนินงานหน่วยความจำ ข้อมูลดังรูปที่ 2.3



รูปที่ 2.3 แสดงขาต่างๆของ 8051 [9]

1. พอร์ต 0 (Port 0)

พอร์ต 0 ได้แก่ขา 32-39 สามารถเป็นพอร์ตอินพุตและเอาต์พุตได้ ซึ่งในการติดต่อกับหน่วยความจำภายนอกสามารถเป็นขาบัสดำแหน่ง และบัสดข้อมูลได้

2. พอร์ต 1 (Port 1)

พอร์ต 1 ได้แก่ขา 1-8 เป็นพอร์ต 8 บิต สามารถอ้างทีละบิตได้ เป็น P1.0, P1.1,...etc

3. พอร์ต 2 (Port 2)

พอร์ต 2 ได้แก่ขา 21-28 สามารถทำงานได้ 2 หน้าที่ เป็นพอร์ต 8 บิต และเป็นขาแอดเดรส 8 บิต ในการอ้างหน่วยความจำภายนอก

4. พอร์ต 3 (Port 3)

พอร์ต 3 ได้แก่ขา 10-17 สามารถทำงานได้ 2 หน้าที่ เป็นพอร์ตอินพุตและเอาต์พุต และเป็นขาควบคุมต่างๆ ดังตารางที่ 2.1 ตารางที่ 2.1 แสดงบิตและหน้าที่ต่างๆ ของพอร์ต

บิต	ชื่อ	หน้าที่ที่แทน
P3.0	RXD	ใช้รับข้อมูลทางพอร์ตอนุกรม
P3.1	TXD	ใช้ส่งข้อมูลทางพอร์ตอนุกรม
P3.2	INT0	อินเทอร์รัพท์ภายนอกหมายเลข 0
P3.3	INT1	อินเทอร์รัพท์ภายนอกหมายเลข 1
P3.4	T0	หัวจับเวลา / หัวนับ ตัวที่ 0
P3.5	T1	หัวจับเวลา / หัวนับ ตัวที่ 1
P3.6	\overline{WR}	สัญญาณเขียนข้อมูลหน่วยความจำภายนอก
P3.7	\overline{RD}	สัญญาณอ่านข้อมูลหน่วยความจำภายนอก

2.2.3 โครงสร้างเบื้องต้นของ 8255

8255 เป็นชิปที่มีขา 40 ขา สามารถต่อพอร์ตให้ไมโครโปรเซสเซอร์ได้ 3 พอร์ต

2.2.4 การแบ่งพอร์ตของ 8255

8255 แบ่งพอร์ตออกเป็น พอร์ต A PA7-PA0, พอร์ต B PB7-PB0, พอร์ต C ซึ่ง พอร์ต C นี้แบ่งออกเป็นพอร์ต C บน PC7-PC4 และ พอร์ต C ล่าง PC3-PC0 โดยพอร์ตทุกพอร์ตสามารถเป็นได้ทั้งพอร์ตอินพุตและพอร์ตเอาต์พุต

2.2.5 ขาต่างๆของ 8255

D0-D7 เป็นขาสำหรับให้ข้อมูลผ่าน ต่อเข้ากับบัสดของไมโครโปรเซสเซอร์ A0-A1 (สัญญาณแอดเดรส) สัญญาณจากทั้งสองขาจะถอดรหัสเป็น 4 รหัสเพื่อกำหนด รีจิสเตอร์ภายใน

PA0-PA7 เป็นสายสัญญาณที่เป็นพอร์ตของ 8255 ที่เรียกว่า พอร์ต A

PB0-PB7 เป็นสายสัญญาณที่เป็นพอร์ทของ 8255 ที่เรียกว่า พอร์ท B

PC0-PC7 เป็นสายสัญญาณที่เป็นพอร์ทของ 8255 ที่เรียกว่า พอร์ท C โดยจะแบ่งเป็น 2 กลุ่มคือ PC0-PC3, PC4-PC7

CS (สัญญาณเลือกชิป) เป็นขาที่รับสัญญาณจากภายนอกเพื่อเลือกชิป เมื่อขานี้เป็น '0' จะทำให้ 8255 ต่อเข้ากับระบบบัสของไมโครโปรเซสเซอร์

RD (สัญญาณอ่าน) เป็นสัญญาณอินพุตจากชิพยู เมื่อสัญญาณนี้และสัญญาณ CS เป็น '0' 8255 จะให้ชิพยูอ่านข้อมูลจากบัส

WR (สัญญาณเขียน) เช่นเดียวกับสัญญาณ RD แต่เป็นการเขียนข้อมูล

RESET (สัญญาณรีเซต) เป็นสัญญาณจากภายนอกที่ส่งเข้ามาเพื่อทำการรีเซต 8255 เมื่อรับสัญญาณนี้ พอร์ททุกพอร์ทจะกลายเป็นพอร์ทอินพุต โดยขา CS, RD, WR, RESET จะแอกทีฟที่ '0' ทั้งหมด

2.2.7 รีจิสเตอร์ภายในของ 8255

พอร์ทแต่ละพอร์ทของ 8255 เปรียบเสมือนรีจิสเตอร์แต่ละตัวที่สามารถ เขียนหรืออ่านได้ ซึ่งจะถูกกำหนดด้วยแอดเดรสตามตั้งไว้ โดยรีจิสเตอร์ แต่ละตัวจะได้รับการกำหนดคู่กับสัญญาณ RD และ WR เพื่อแสดงความหมาย

2.2.8 การใช้งาน 8255

การใช้งาน 8255 นั้นจะต้องส่งรหัสเข้าสู่พอร์ทควบคุมเพื่อควบคุม การทำงานของ 8255 โดยการควบคุม 8255 มีหลายโหมด แต่ละโหมดจะแตกต่างกันไป แบ่งเป็น โหมด 0, โหมด 1 และ โหมด 2 การกำหนดโหมดการทำงาน

จะกำหนดโดยการส่งข้อมูลซึ่งเป็นรหัสควบคุม ไปยังพอร์ทควบคุมเพื่อควบคุมของ 8255 โดยแต่ละบิตของรหัสควบคุมก็จะมี ความหมายของตนเอง โดยความหมาย ของแต่ละบิตมีดังนี้ D7 ถ้าบิตนี้เป็น '1' หมายถึงรหัสควบคุมนี้มีผลต่อการเปลี่ยนแปลงการเซต โหมด

D6-D5 ใช้เลือก โหมด ถ้าเป็น

00 = โหมด 0

01 = โหมด 1

1x = โหมด 2

D4 ใช้กำหนดพอร์ท A โดยถ้าเป็น '0' หมายถึง กำหนดให้พอร์ท A เป็นเอาต์พุต ถ้ากำหนดพอร์ท A โดยถ้าเป็น '1' หมายถึง กำหนดให้พอร์ท A เป็นอินพุต

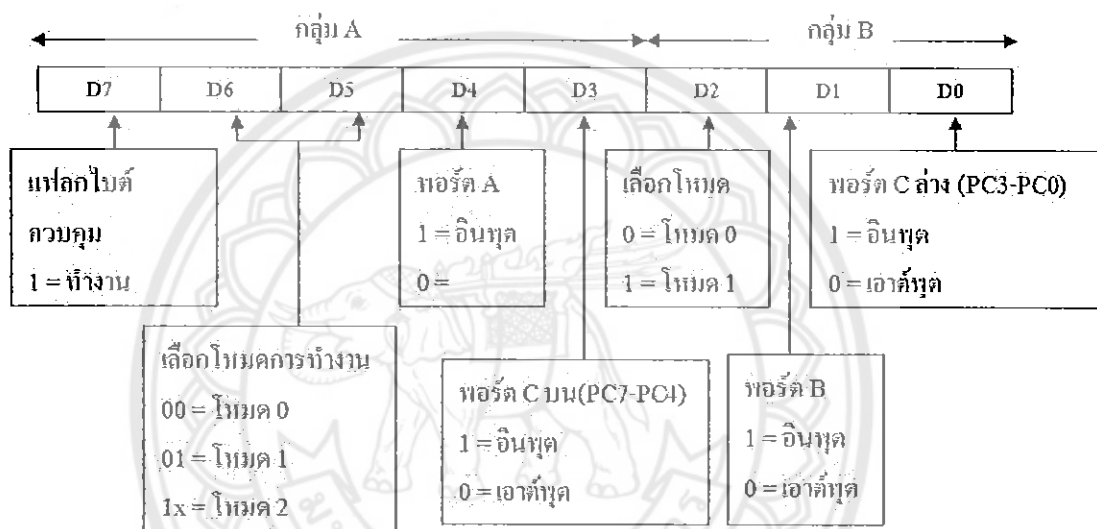
D3 ใช้กำหนดพอร์ท C บน โดยถ้าเป็น '0' หมายถึง กำหนดให้พอร์ท C บน เป็นเอาต์พุต ถ้ากำหนดพอร์ท C บน โดยถ้าเป็น '1' หมายถึง กำหนดให้พอร์ท A เป็นอินพุต

D2 เป็นการเลือก โหมด ถ้าเป็น "0" หมายถึงเป็น โหมด 0 ถ้าเป็น "1" หมายถึงเป็น โหมด 1

D1 ใช้กำหนดพอร์ต B โดยถ้าเป็น “0” หมายถึง กำหนดให้พอร์ต B เป็นเอาต์พุต
ถ้ากำหนดพอร์ต A โดยถ้าเป็น “1” หมายถึง กำหนดให้พอร์ต A เป็นอินพุต

D0 ใช้กำหนดพอร์ต C ล่าง โดยถ้าเป็น “0” หมายถึง กำหนดให้พอร์ต C ล่าง เป็นเอาต์พุต
ถ้ากำหนดพอร์ต C ล่าง โดยถ้าเป็น “1” หมายถึง กำหนดให้พอร์ต A เป็นอินพุต

ในการเชื่อมต่อ 8255 กับ MCS-51 นั้นจะใช้สัญญาณดังนี้ D0-D7, RD, WR, CS, A0 และ A1
การใช้งาน 8255 เป็นพอร์ตขนานได้นั้นจะต้องโปรแกรมการทำงานให้กับรีจิสเตอร์ควบคุม
เสียก่อนว่าต้องการใช้พอร์ตใดเป็นอินพุตพอร์ตใดเป็นเอาต์พุต สำหรับรีจิสเตอร์ควบคุมแสดงได้ดัง
รูปที่ 2.4



รูปที่ 2.4 ความหมายแต่ละบิตของรีจิสเตอร์ควบคุม

ในที่นี้จะใช้การทำงาน 8255 ในโหมดอินพุตเอาต์พุต โหมด 0 การโปรแกรมการทำงานให้กับ 8255
จะต้องส่งไบตรีจิสเตอร์ควบคุม โดยส่งไปยังแอดเดรสที่ขา A0 และ A1 เป็นลอจิก “1” ทั้งคู
จิสเตอร์ที่เกี่ยวข้องกับการทำงานของพอร์ตอนุกรมใน MCS-51

การทำงานของพอร์ตอนุกรมจะมีรีจิสเตอร์ที่เกี่ยวข้อง 2 ตัว คือ รีจิสเตอร์ SBUF และ
รีจิสเตอร์ควบคุมการทำงานของพอร์ตอนุกรม

Port สื่อสารอนุกรมของไมโครคอนโทรลเลอร์ จะมีโครงสร้างเป็นแบบฟูลดูเพล็กซ์ ซึ่งรับ
และส่งข้อมูลในเวลาเดียวกันได้ โดยจะมี Register SBUF (Serial Data Buffer) เป็น Buffer สำหรับการ
การรับส่งข้อมูลอนุกรม โดยเริ่มต้นเมื่อมีการเขียนข้อมูลเก็บไว้ใน Register SBUF หลังจากนั้น
ข้อมูลจะถูกจัดการโดยวิธีทาง ฮาร์ดแวร์ ในการเลื่อนบิต เพื่อส่งสัญญาณออกไปภายนอก หลังจาก
มีการส่งข้อมูลออกไปจนครบแล้ว จึงจะทำการเซตบิตโดยกำหนดค่าของแฟลต TI ใน Register
SCON ให้เป็นสถานะ “1” เพื่อแจ้งว่า Register SBUF ว่างแล้ว และพร้อมที่จะส่งข้อมูลไบต์ต่อไป

การรับข้อมูลจาก Port อนุกรมจะต้องเริ่มต้น โดยการกำหนดค่าของบิต REN ที่อยู่ใน Register SCON ให้มีค่าเป็นสถานะ "1" หลังจากนั้นเมื่อมีการรับข้อมูลเข้ามาจากภายนอกก็จะทำการเลื่อนข้อมูลไปโดยอัตโนมัติ และเมื่อบิตสุดท้ายถูกเลื่อนบิตเข้ามาเรียบร้อยแล้ว ข้อมูลจะถูกย้ายมาเก็บไว้ที่ Register SBUF และจะทำการเซตที่บิต RI ให้เป็นสถานะ "1" ซึ่งส่งผลให้เกิดการอินเทอร์รัพท์โปรแกรมขึ้น

2.2.10 Register ที่ใช้ในการควบคุมและรับส่งข้อมูลของPortอนุกรม

Register ที่ใช้งานในการติดต่อสื่อสารทางPort อนุกรมของไมโครคอนโทรลเลอร์ จะประกอบด้วย Register SCON ที่ทำหน้าที่ควบคุมการทำงาน Register SBUF จะใช้ในการเก็บข้อมูลที่จะรับหรือส่ง และ Register PCON ซึ่งจะใช้ ในการกำหนดอัตรารับส่ง โดย Register แต่ละตัวจะมีหน้าที่และการทำงานในแต่ละบิตดังต่อไปนี้

Register SCON (Serial Port Control Register) เป็น Register ขนาด 8 บิต อยู่ในส่วนของ Register พิเศษ (Special Function Register) ในตำแหน่งแอดเดรสที่ 98H และสามารถเข้าถึงข้อมูลแบบไบต์ และแบบบิตได้ โดยจะทำหน้าที่ควบคุมการทำงานของ Port อนุกรม การเลือกโหมดการทำงาน และเก็บข้อมูลในบิตที่ 9 (ซึ่งโดยปกติข้อมูลจะมี 8 บิต อยู่ใน Register SBUF) ของการรับข้อมูล (RB8) และส่งข้อมูล (TB8) รายละเอียดของแต่ละบิตมีดังต่อไปนี้

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
SM0	SM1	SM2	REN	TB8	RB8	TI	RI

รูปที่ 2.5 แสดงบิตต่างๆของ Register SCON (Serial Port Control Register) [4]

SM0, SM1: (Serial port mode bit 0-1) เป็นบิตที่ใช้ในการกำหนดโหมดการทำงานของ Port อนุกรมจำนวน 4 โหมด

SM0	SM1	โหมด การทำงาน อัตรารับ-ส่ง
0	0	0 Shift register $f_{osc}/12$
0	1	1 8 bit UART Variable
1	0	2 9 bit UART $f_{osc}/32$ หรือ $f_{osc}/64$
1	1	3 9 bit UART Variable

รูปที่ 2.6 แสดงการกำหนดบิต SM0 และ SM1 เพื่อกำหนดโหมดการทำงาน [4]

SM2 : เป็นบิตที่ทำหน้าที่ควบคุมการทำงานและเลือกลักษณะการเชื่อมต่อสื่อสารระหว่างไมโครคอนโทรลเลอร์แบบ Single Processor System หรือ Multi processors System โดยกำหนดให้

- SM2 = 1 เป็นการเลือกแบบ Multi processors System คือระบบการสื่อสารแบบใช้ CPU

หลายๆตัวร่วมกันทำงานจะใช้งานในโหมด2หรือโหมด3

- SM2 = 0 เป็นการเลือกแบบ Single Processor System โดยสามารถใช้ได้กับทุกโหมด (การใช้งานในโหมด0ต้องกำหนดให้SM2=0)

ในกรณีที่เลือกให้ SM2 = 1-แบบ Multi processors System ถ้าข้อมูลที่ได้รับเข้ามาบิตที่-9 (อยู่ในบิต RB8) มีค่าเป็น "1" ทำให้แฟล็ก Interruptทางด้านรับ จะถูกเซตให้เป็น 1 (RI = 1) แต่ถ้าข้อมูลในบิตที่ 9 รับเข้ามามีค่าเป็น "0" จะทำให้แฟล็ก Interruptทางด้านรับเป็น 0 (RI = 0) การทำงาน ในโหมด 1 ถ้าให้ SM2 = 1 แฟล็ก Interruptทางด้านรับ (แฟล็ก RI) จะไม่ถูกเซตหากข้อมูลที่ได้รับเข้ามาไม่มีบิตหยุด(Stopbit)

REN : (Enable Serial Reception) เป็นบิตที่ควบคุมการรับข้อมูลของPortอนุกรม กำหนดสถานะของบิตได้โดยซอฟต์แวร์

1=ให้มีการรับข้อมูล

0=ไม่ให้มีการรับข้อมูล

TB8 : (Transmit bit D8) เป็นบิตของข้อมูลบิตที่ 9 ในการส่งข้อมูลใช้งานโหมด 2 และ โหมด3กำหนดสถานะของบิตได้โดยซอฟต์แวร์

RB8 : (Receive bit D8) เป็นบิตของข้อมูลบิตที่ 9 ในการรับข้อมูล โดยใช้งานโหมด 2 และ โหมด 3 หากใช้งานในโหมด 1 ถ้ากำหนดให้ SM2 = 0 บิตนี้จะเป็ค่าของ Stop Bit ที่รับเข้ามา สำหรับโหมด 0 จะไม่ใช้งานบิตนี้

TI : (Transmit Interrupt Flag) เป็นบิตที่ใช้งานในการอินเตอร์รัพท์ด้านส่งข้อมูล และจะถูกเซตทางฮาร์ดแวร์เมื่อมีการส่งข้อมูลเสร็จสิ้นลงในบิตที่ 8 ของโหมด 0 (Shift register) หรือเมื่อเริ่มต้นส่ง Stop bit ในโหมด 1,2 หรือ 3 และจะต้องเคลียร์บิตนี้ด้วยซอฟต์แวร์ทุกครั้ง เมื่อโปรแกรมตอบสนองการอินเตอร์รัพท์ของการส่งข้อมูลเรียบร้อยแล้ว

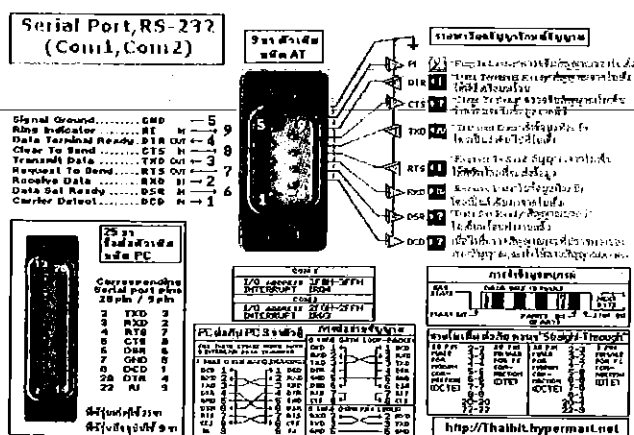
RI : (Receive Interrupt Flag) เป็นบิตที่ใช้งานในการอินเตอร์รัพท์ทางด้านรับข้อมูล จะถูกเซตทางฮาร์ดแวร์ เมื่อมีการรับข้อมูลเสร็จสิ้นลงในบิตที่ 8 ในโหมด 0 (Shift register) และจะต้องเคลียร์บิตนี้ด้วยซอฟต์แวร์ทุกครั้ง เมื่อโปรแกรมตอบสนองการอินเตอร์รัพท์ของการรับข้อมูลเรียบร้อยแล้ว หรืออาจกล่าวได้ว่าถ้าบิต RI ถูกเซต เมื่อใด หมายถึงข้อมูลได้เข้ามาเก็บไว้ที่Register SBUF จนครบทั้ง 8 บิตแล้ว สามารถที่จะอ่านข้อมูลจากRegister SBUF ได้

Register SBUF (serial data buffer register) เป็น Register ขนาด 8 บิตหรือ 1 ไบต์มีแอดเดรสอยู่ตำแหน่งที่ 99H และเข้าถึงข้อมูลแบบไบต์ได้อย่างเดียว จะทำหน้าที่รับและส่งข้อมูลออกไปยัง Port อนุกรมของไมโครคอนโทรลเลอร์ ในการอ่านค่าข้อมูลจากภายนอกที่รับเข้ามาทางPortอนุกรมจะต้องอ่านค่าจาก Register SBUF ซึ่งเป็นบัฟเฟอร์ (Buffer) ในการเก็บข้อมูลที่รับเข้ามาได้จากภายนอก และในทำนองเดียวกันขณะที่ต้องการส่งข้อมูล เราก็จะนำเอาค่าข้อมูลที่ส่งออกไปไว้ใน Register SBUF ก่อน และหลังจากนั้นจึงจะส่งออกไป โดยจะใช้คำสั่งการโอนย้ายข้อมูลแบบไบต์เช่น MOV SBUF,#20H หรือ MOV SBUF,@R1 ก็ได้ การรับข้อมูลในโหมด 0 จะเริ่มต้นรับ เมื่อค่าของบิต RI = 0 และ REN = 1 ส่วนในโหมดอื่นๆ การรับข้อมูลจะเริ่มต้นเมื่อกำหนดบิต REN = 1 และมี Start bit เข้ามาที่ขา RXD Register PCON (Power Control) เป็น Register ขนาด 1 ไบต์มีแอดเดรสอยู่ตำแหน่ง ที่ 87H เข้าถึงข้อมูลได้แบบไบต์อย่างเดียวกัน โดยจะประกอบด้วยบิตดังต่อไปนี้

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
SMOD	-	-	-	GF1	GF0	PD	IDL

รูปที่ 2.7 แสดง Register PCON เพื่อกำหนดอัตราการรับส่งข้อมูล [4]

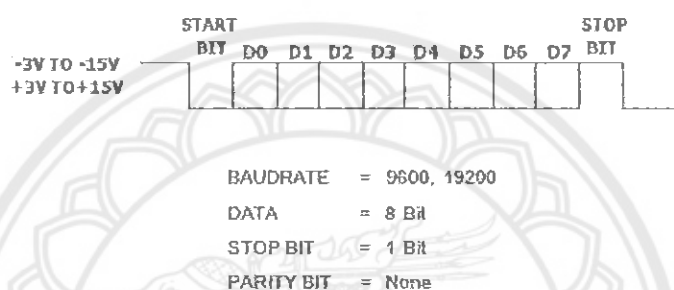
PCON.7 SMOD: ในกรณีที่ใช้ Timer1 เป็นตัวกำหนดอัตรารับส่ง (Baud rate) และหากกำหนดให้บิตนี้มีค่าเป็น "0" ในการใช้งานกับPort สื่อสารอนุกรมโหมด 1, 2 และโหมด 3 ค่าอัตรารับส่ง (Baud rate) จะเพิ่มขึ้นเป็นสองเท่า Register PCON ไม่สามารถอ้างตำแหน่งแบบบิตได้ แต่จะใช้คำสั่งทางลอจิกของการ OR เช่น ORL PCON, #80H จะเป็นการเซตบิตที่ 7 ของRegister PCON และการกำหนดให้บิตมีสถานะเป็น"0"หรือเคลียร์บิตจะใช้การANDเช่น ANL PCON, #0111111B จะเป็นการเคลียร์บิตที่ 7 ของRegister PCON



รูปที่ 2.8 รายละเอียดของ Com Port หรือ Serial Port [3]

2.2.12 การแสดงสัญญาณ RS232

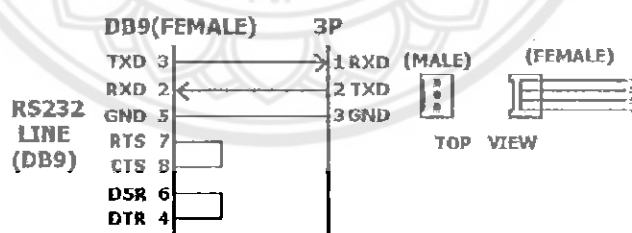
การกำหนดค่าการส่งข้อมูลระหว่างคอมพิวเตอร์กับบอร์ดทดลองเพื่อใช้ในการติดต่อสื่อสาร



รูปที่ 2.9 ภาพแสดงสัญญาณ RS232 [10]

RS232 LINE (DB9)

การเชื่อมการทำงานระหว่าง RS232 กับบอร์ดทดลอง ให้สามารถติดต่อสื่อสารกับคอมพิวเตอร์



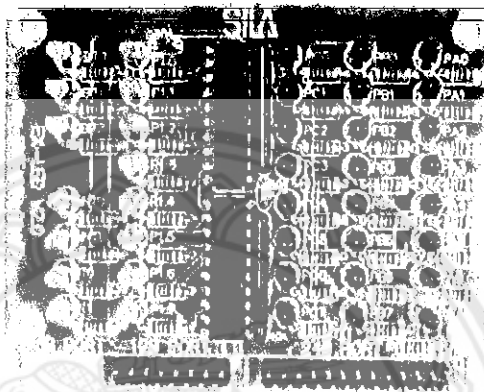
รูปที่ 2.10 RS232 LINE [10]

2.2.13 EX-LED

ผู้จัดทำโครงงานได้นำบอร์ด EX-LED มาใช้ในการแสดงผล การทำงานของโปรแกรมใน ส่วนของการควบคุม โดยแทนดวงไฟ 1 ดวง กับ หลอด LED 1 หลอด ในด้านการติดต่อกับบอร์ด

EX-LED ผู้จัดทำได้ใช้ 8255PORT (24BPORT) ในการติดต่อสื่อสาร EX-LED v2.0 LED Monitor Board

EX-LED เป็นบอร์ดแสดงผลสถานะ OUTPUT ของ 3BPORT, 12BPORT และ 8255PORT โดยมี 3LED, BIT0, BIT1, BIT2 แสดงสถานะเอาต์พุต ของ 3BPORT มี 12 LED DISPLAY สำหรับ แสดงสถานะ OUTPUT ของ 12BPORT และ 24 LED DISPLAY สำหรับ แสดงสถานะ OUTPUT ของ 8255PORT (24BPORT) ดังรูปแสดงตำแหน่งต่างๆ บนบอร์ด



รูปที่ 2.11 บอร์ด EX-LED [10]

PORT ที่ใช้ในการติดต่อสื่อสารกับ EX-LED

3B PORT

คือพอร์ตมาตรฐานคิลาพอร์ตหนึ่ง ที่กำหนดขึ้นเพื่อใช้งานกับบอร์ดหรือชิพเพิ่มขยายที่ไม่ต้องการบิตพอร์ตในการใช้งานมากๆ

12BPORT

จะประกอบไปด้วยพอร์ต 1 ซึ่งมีทั้งหมด 8 บิต คือ P1.0-P1.7 เป็น I/O PORT แบบขนาน โดยแต่ละเส้นของ I/O จะเป็นอิสระในการกำหนดให้เป็นอินพุตหรือเอาต์พุต มีโครงสร้างคล้ายพอร์ต 0

8255PORT (24BPORT)

เป็นพอร์ตของไอซี-8255 เพื่อทำหน้าที่เป็น I/O PORT เพิ่มขึ้นจากพอร์ต 1 ของไมโครคอนโทรลเลอร์ ภายในมีพอร์ตให้ใช้งานอยู่ 3 พอร์ตคือพอร์ต A, พอร์ต B, และพอร์ต C สามารถควบคุม ลักษณะการทำงานของพอร์ตแบบขนาน 8 บิตได้อย่างอิสระ โปรแกรมให้เป็นอินพุตหรือเอาต์พุตก็ได้ โดยที่พอร์ต C ยังสามารถแบ่งเป็น 4 บิตล่างและ 4 บิตบน การกำหนดให้แต่ละพอร์ตเป็น INPUT หรือ OUTPUT นั้น สามารถทำได้ด้วยการส่ง Control Word ไปก่อนโดยส่งไปที่ Control Port ของ 8255 พอร์ต A และ B ต้องกำหนดให้เป็น INPUT หรือ OUTPUT เหมือนกันทั้ง 8 บิต และพอร์ต C แยกเป็น 4 บิต 2 ส่วนอิสระจากกัน มีโหมดการทำงาน ให้เลือกอยู่ 3 โหมด แต่

โหมด 0 จะเป็นโหมดพื้นฐานที่นิยมใช้กันมากที่สุด คือเป็น การทำงาน เป็นพอร์ทอินพุท/เอาต์พุท แบบพื้นฐานไม่มีการตรวจสอบสัญญาณ (Handshake)

2.2.14 ระบบที่ใช้ติดต่อสื่อสารข้อมูลของคอมพิวเตอร์

จากระดับสัญญาณของพอร์ตอนุกรม RS-232 มีระดับตั้งแต่ -12 ถึง 12 v ในขณะที่ระดับสัญญาณของไมโครคอนโทรลเลอร์ อยู่ในระดับที่ที่แอล ดังนั้นจึงไม่สามารถเชื่อมต่อพอร์ตอนุกรมของไมโครคอนโทรลเลอร์ MCS-51 เข้ากับพอร์ตอนุกรมของคอมพิวเตอร์ได้โดยตรง จึงต้องมีการแปลงระดับสัญญาณ

การสื่อสารข้อมูลระหว่างตัวรับและตัวส่งนั้นมีหลายวิธี ถ้าหากตัวส่งทำหน้าที่ส่งอย่างเดียว และตัวรับทำหน้าที่รับอย่างเดียวจะเรียกว่าการสื่อสารแบบซิมเพิล็กซ์(Simplex)



รูปที่ 2.12 การรับส่งข้อมูลแบบ simplex

สำหรับวิธีในการส่งข้อมูลแบบอนุกรมมีสองวิธีคือการส่งแบบเข้าจังหวะเวลา (Synchronous) และ การส่งแบบไม่เข้าจังหวะเวลา (Asynchronous) จะใช้การกำหนดอัตราเร็วในการรับส่งข้อมูลให้มีค่าเท่ากันที่เรียกว่า อัตราบอด หรือบอดเรต (band rate) ซึ่งมีหน่วยเป็นบิตต่อวินาที (bit per second: bps)

การส่งข้อมูลแบบซิงโครนัส

การรับส่งข้อมูลแบบซิงโครนัสจะประกอบด้วย 4 ส่วนดังนี้

1. บิตเริ่มต้น (start bit)
2. ข้อมูลอนุกรม
3. บิตตรวจสอบความถูกต้อง
4. บิตสุดท้าย (stop bit)

บิตเริ่มต้นและบิตสุดท้าย

การส่งข้อมูลแบบอะซิงโครนัสนั้นจะส่งข้อมูลออกไปเป็นชุดเรียกว่า เฟรม ภายใน เฟรม (Framing) ภายในเฟรมจะประกอบด้วยข้อมูลหรือรหัส ASCII ก็ได้ ในแต่ละเฟรมจะเริ่มต้น (start

bit) ที่จะบอกว่าสิ่งที่ตามมาก็คือข้อมูล และจะจบข้อมูลด้วยบิตสุดท้าย (stop bit) ที่จะบอกว่าข้อมูลในเฟรมนั้นๆ สิ้นสุดลงแล้ว

อัตราการรับส่งข้อมูล

ความเร็วในการรับส่งข้อมูลแบบอนุกรมจะบอกเป็นจำนวนบิตต่อวินาที (Bits per Second: bps) ที่เรียกว่าบอดเรต (baud rate) พอร์ตอนุกรมของ MCS-51 สามารถติดต่อกับเครื่องคอมพิวเตอร์ได้แต่ต้องกำหนดอัตราเร็วให้เท่ากัน ในการสื่อสารทางพอร์ตอนุกรม RS-232 ของคอมพิวเตอร์ PC ได้กำหนดอัตราเร็วไว้หลายค่าตั้งแต่ 100 ถึง 9600 bps

การโปรแกรมการสื่อสารแบบอนุกรมของ 8051

หากคอมพิวเตอร์ ต้องการติดต่อกับระบบ 8051 จะต้องสื่อสารแบบอนุกรมผ่านทาง COM พอร์ต โดยความเร็วที่ใช้ในการสื่อสารแสดงดังตารางที่ 2.1 โดยเราสามารถเขียน โปรแกรมเลือกอัตราเร็วได้

ตารางที่ 2.1 บอดเรตของเครื่องคอมพิวเตอร์ PC

110
150
300
600
1200
2400
4800
9600
19200

2.2.15 บอดเรตของ 8051

8051 สามารถรับส่งข้อมูลได้บอดเรตหลายค่า โดยเราสามารถเลือกได้โดยโปรแกรมเข้าไปทางไทมเมอร์ 1 และความเร็วที่ได้จะสัมพันธ์กับความถี่ของ XTAL ที่ต่อกับระบบด้วย

อัตราเร็วเมื่อบิต SMOD = 0

เมื่อ SMOD มีค่าเป็น 0 ตัว 8051 จะนำค่าความถี่ของ XTAL ที่ต่ออยู่กับระบบมาหารด้วย 12 ซึ่งจะได้ค่าความถี่ของแมชชีน ไซเคิล จากนั้นนำมาหารด้วย 32 อีกครั้ง ค่าที่ได้จะถูกส่งให้ไทมเมอร์ 1 จำนวน ค่าอัตราบอดเรต

อัตราเร็วเมื่อบิต SMOD = 1

ถ้าหากระบบใช้ XTAL ที่มีความถี่ค่าเดิมและบิต SMOD ถูกเซตให้เป็นลอจิก "1" ความเร็วหรืออัตราบอดที่ได้จะเพิ่มขึ้นเท่าตัว เนื่องจากระบบจะนำค่าความถี่ของแมชชีน ไซเคิลมาหารด้วย

ตารางที่ 2.2 เปรียบเทียบบอครตเมื่อ SMOD = 0 และ SMOD = 1

TH1(Decimals)	Hex	SMOD = 0	SMOD = 1
-3	FD	9,600	19,200
-6	FA	4,800	9,600
-12	F4	2,400	4,800
-24	E8	1,200	2,400

2.2.16 การเขียนหรือส่งข้อมูลออกจากพอร์ตอนุกรม

ข้อมูลที่ต้องการส่งออกทุกค่าจะต้องนำไปเก็บไว้ในรีจิสเตอร์บัพเฟอร์ของพอร์ตอนุกรมที่ชื่อว่า SBUF ดังตัวอย่าง

MOV SBUF, #'A'

จากคำสั่งนี้จะเป็นการส่งข้อมูลของตัวอักษร A ออกไปยังพอร์ตอนุกรมของไมโครคอนโทรลเลอร์ แต่ก่อนที่จะส่งข้อมูลทุกครั้งจะต้องทำการเคลียร์บิต TI ก่อน หรือทำให้มีค่าเป็น "0" เมื่อทำการส่งข้อมูลเรียบร้อยแล้ว ก็จะเกิดการเซตบิต TI ดังตัวอย่าง

CLR TI ; เคลียร์บิต TI เพื่อเตรียมการส่งข้อมูลออก
MOV SBUF, #'A' ; ส่งข้อมูล A ส่งไปยังพอร์ตอนุกรม
JNB TI, \$; รอการเซตของบิต TI เพื่อแจ้งการส่งข้อมูล

การอ่านหรือรับข้อมูลจากพอร์ตอนุกรม

การรับข้อมูล ทำได้โดยการตรวจสอบบิต RI ว่ามีการเซตขึ้นหรือไม่ ถ้าพบว่ามีการเซตขึ้นให้ทำการอ่านค่ามาจากรีจิสเตอร์ SBUF โดยต้องทำการโอนย้ายข้อมูลผ่านทางแอดเดรสเดเคเตอร์หรือรีจิสเตอร์ A ดังตัวอย่าง

CLR RI ; เคลียร์บิต RI เพื่อเตรียมการรับข้อมูล
JNB TI, \$; รอการเซตของบิต RI เพื่อแจ้ง การรับข้อมูลเสร็จสมบูรณ์ และมีข้อมูลเกิดขึ้นที่รีจิสเตอร์ SBUF
MOVA, SBUF ; อ่านค่าจากรีจิสเตอร์ โดยการ โอนย้ายข้อมูลผ่านทางจากรีจิสเตอร์ A

CLR RI ; หลังจากอ่านข้อมูลเรียบร้อยแล้วต้องทำการเคลียร์บิต RI เสมอ

2.3 องค์ประกอบทางด้าน Software

ภาษาที่ใช้ในการเขียนโปรแกรมเพื่อติดต่อกับ Hardware ได้แก่ Language, Assembly Language และ Visual Basic Programming ส่วนภาษาที่ใช้เขียน Web-based ได้แก่

2.3.1 Visual Basic Programming Interface Hardware

ผู้จัดทำโครงการนี้ได้เลือกใช้ภาษา VISUAL BASIC เนื่องจากมีความสะดวกสบายในการออกแบบ โดยภาษา VISUAL BASIC จะมีคอนโทรลชนิดต่างๆมาช่วยในการออกแบบ และ เป็นการเขียนโปรแกรมเพื่อตอบสนองเหตุการณ์ที่เกิดขึ้น (Event-Driven Programming) ซึ่งในจุดนี้ทำให้ผู้จัดเลือกใช้ภาษา VISUAL BASIC ในการพัฒนาโปรแกรมควบคุมฝั่ง SERVER

VISUAL BASIC ถูกพัฒนาโดยไมโครซอฟท์ โดยเพิ่มพีเจอร์ต่างๆ เข้าไปมากมาย จนปัจจุบันถึงเวอร์ชัน 2005 (ปี 2549) โดยการพัฒนาแอปพลิเคชันด้วย VB จะเป็นลักษณะการนำคอนโทรลชนิดต่างๆ เช่น Textbox, Label, ComboBox เป็นต้น โดยนำมาวาด เพื่อออกแบบหน้าต่างแอปพลิเคชันที่เรียกว่า กราฟฟิคยูสเซอร์ อินเตอร์เฟส (Graphic User Interface-GUI) สามารถที่จะออกแบบอินเตอร์เฟสได้อย่างอิสระให้ตรงกับจุดประสงค์และ การนำไปใช้งานก่อน แล้วจึงเริ่มเขียนโค้ดเพื่อตอบสนองการกระทำของผู้ใช้ (ใน VB เรียกว่า เหตุการณ์ event) ซึ่งถือเป็นหลักการเขียนโปรแกรมที่เรียกว่า การเขียนโปรแกรมเพื่อตอบสนองเหตุการณ์ที่เกิดขึ้น

(Event-Driven Programming) สิ่งต่างๆ ที่นำไปใช้ร่วมกันเพื่อเป็นแอปพลิเคชันหนึ่งๆเช่น แถบเมนู, dialog box, toolbars, Textbox, ปุ่ม OK ฯลฯ เป็นต้น จะถูกมองเป็นวัตถุขึ้นหนึ่งที่เรียกว่า อ็อบเจกต์ (object model) ทุกสิ่งทุกอย่าง ในแอปพลิเคชัน VB จะมองเป็นอ็อบเจกต์ ที่สามารถควบคุมพฤติกรรม, แก๊ซ, และกระทำโดยตรงต่ออ็อบเจกต์นั้นได้ ด้วยการเขียนโค้ด หรือสามารถเปลี่ยนแปลง คุณสมบัติ (Properties) ประจำตัวของอ็อบเจกต์นั้นได้โดยตรง ในการเขียนโค้ดจะถูกแบ่งออกเป็นส่วนต่างๆ ที่เรียกว่า โพรซีเจอร์ (Procedure) และแต่ละโพรซีเจอร์จะประกอบไปด้วยโค้ดที่พิมพ์เข้าไปแล้วทำให้คอนโทรลหรืออ็อบเจกต์นั้นๆ ตอบสนองการกระทำ ของผู้ใช้ได้โดยสมบูรณ์ในตัวเอง ซึ่งเรียกว่า การเขียนโปรแกรมเชิงวัตถุ(ObjectOriented Programming-OOP) แต่ตัวภาษา VB ยังไม่ถือว่าเป็น OOP อย่างแท้จริง เนื่องจากข้อจำกัดหลายๆ อย่างที่ VB ไม่สามารถทำได้เหมือนC++

Visual Basic สามารถเขียนโปรแกรมติดต่อ I/O ผ่านทาง Port ของเครื่องคอมพิวเตอร์ ไม่ว่าจะเป็นทาง Serial Port (RS-232) หรือที่รู้จักในชื่อ Com1, Com2 และ Parallel Port หรือ Printer Port นั้นเอง หรืออาจใช้ Card I/O 8255 ซึ่งเป็นการขยาย Port I/O ของ Parallel ก็สามารถทำการติดต่อกับ Hardware ภายนอกผ่าน Port อีกทั้งสามารถติดต่อผ่านระบบ Network โดยผ่านช่องทางการติดต่ออย่าง TPC/IP จะเห็นได้ว่า VB 6 ก็สามารถทำงานด้าน Interface Hardware ได้

โดย Visual Basic จะมีตัวคอนโทรลชื่อ MSComm ที่สามารถติดต่อผ่าน RS-232 ได้ และ Winsock ที่สามารถติดต่อผ่าน TCP/IP

2.3.2 JAVA SCRIPT

ในการสร้างเว็บ ได้มีการนำ Java Script มาใช้งานส่วนหนึ่ง ซึ่ง Java Script เป็นภาษาโปรแกรม (programming language) ประเภทหนึ่ง ที่เรียกกันว่า script ซึ่งมีวิธีการทำงานในลักษณะ "แปลความและดำเนินงานไปที่ละคำสั่ง" ภาษานี้เดิมมีชื่อว่า Live Script ได้รับการพัฒนาขึ้นโดย Netscape ด้วยวัตถุประสงค์ เพื่อที่จะช่วยให้เว็บเพจสามารถแสดงเนื้อหา ที่มีการเปลี่ยนแปลงไปตามเงื่อนไขหรือสภาพแวดล้อมต่างๆกัน หรือสามารถโต้ตอบกับผู้ชมได้มากขึ้น ทั้งนี้เพราะภาษา HTML แต่เดิมนั้น เหมาะสมสำหรับใช้แสดงเอกสาร ที่มีเนื้อหาคงที่แน่นอน และไม่มีลูกเล่นอะไรมากมายนัก เนื่องจาก JavaScript ช่วยให้ผู้พัฒนา สามารถสร้างเว็บเพจได้ตรงกับความต้องการ และมีความน่าสนใจมากขึ้น ประกอบกับเป็นภาษาเปิด ที่ใครก็สามารถนำไปใช้ได้ ดังนั้นจึงได้รับความนิยมเป็นอย่างสูง มีการใช้งานอย่างกว้างขวาง รวมทั้งได้ถูกกำหนดให้เป็นมาตรฐานโดย ECMA การทำงานของ JavaScript จะต้องมีการแปลความคำสั่ง ซึ่งขั้นตอนนี้จะถูกจัดการโดยบราวเซอร์ ดังนั้น JavaScript จึงสามารถทำงานได้ เฉพาะบนบราวเซอร์ที่สนับสนุน ซึ่งปัจจุบันบราวเซอร์เกือบทั้งหมดก็สนับสนุน JavaScript แล้ว อย่างไรก็ตาม สิ่งที่ต้องระวังคือ JavaScript มีการพัฒนาเป็นเวอร์ชันใหม่ๆออกมาด้วย (ปัจจุบันคือรุ่น 1.5) ดังนั้น ถ้านำโค้ดของเวอร์ชันใหม่ ไปรันบนบราวเซอร์รุ่นเก่าที่ยังไม่สนับสนุน ก็อาจจะทำให้เกิด error ได้ การทำงานของ JavaScript เกิดขึ้นบนบราวเซอร์ (เรียกว่าเป็น Client – side script) ดังนั้นไม่ว่าจะใช้เซิร์ฟเวอร์ไหน ก็ยังคงสามารถใช้ JavaScript ในเว็บเพจได้ ต่างกับภาษาสคริปต์อื่น เช่น Perl, PHP หรือ ASP ซึ่งต้องแปลความและทำงานที่ตัวเครื่องเซิร์ฟเวอร์ (เรียกว่า server– side script) ดังนั้นจึงต้องใช้บนเซิร์ฟเวอร์ ที่สนับสนุนภาษาเหล่านี้เท่านั้น อย่างไรก็ตาม จากลักษณะดังกล่าวก็ทำให้ JavaScript มีข้อจำกัด คือไม่สามารถรับและส่งข้อมูลต่างๆ กับเซิร์ฟเวอร์โดยตรง เช่น การอ่านไฟล์จากเซิร์ฟเวอร์ เพื่อนำมาแสดงบนเว็บเพจ หรือรับข้อมูลจากผู้ชม เพื่อนำไปเก็บบนเซิร์ฟเวอร์ เป็นต้น ดังนั้นงานลักษณะนี้ จึงยังคงต้องอาศัยภาษา server-side script อยู่ (ความจริง JavaScript ที่ทำงานบนเซิร์ฟเวอร์เวอร์ก็มี ซึ่งต้องอาศัยเซิร์ฟเวอร์ที่สนับสนุน โดยเฉพาะเช่นกัน แต่ไม่เป็นที่นิยมนัก) การทำงานของ JavaScript จะมีประสิทธิภาพมาก ถ้าสามารถดัดแปลงคุณสมบัติ-ขององค์ประกอบต่างๆบนเว็บเพจ (เช่น สีหรือรูปแบบของข้อความ) และสามารถรับรู้เหตุการณ์ ที่ผู้ชมเว็บเพจโต้ตอบกับองค์ประกอบเหล่านั้น (เช่น การคลิก หรือเลื่อนเมาส์ไปวาง) ได้ ดังนั้นจากภาษา HTML เดิม ที่มีลักษณะ static ใน HTML เวอร์ชันใหม่ๆ จึงได้มีการพัฒนาให้มีคุณสมบัติบางอย่างเพิ่มขึ้น และมีลักษณะเป็นอ็อบเจกต์ "object" มากขึ้น การทำงานร่วมกันระหว่างคุณสมบัติใหม่ของ HTML ร่วมกับ JavaScript นี้เอง ทำ

ให้เกิดเป็นสิ่งที่เรียกว่า Dynamic HTML คือภาษา HTML ที่สามารถใช้สร้างเว็บเพจที่มีลักษณะพลวัต

2.3.3 Assembly Language

ภาษาแอสเซมบลีเป็นภาษาที่ใกล้เคียงกับภาษาเครื่องมากที่สุดสามารถเขียนคำสั่งเพื่อติดต่อสั่งงานกับอุปกรณ์ Hardware ได้ โดยตรงสามารถทำงานได้รวดเร็วมากแต่มีข้อจำกัดคือเขียนโปรแกรมได้ยาก เข้าใจได้ยาก

โดยมีรูปแบบการเขียนโปรแกรมติดต่ออุปกรณ์ผ่านการเขียนโปรแกรม ในอดีตนั้นการเขียนโปรแกรมติดต่อ และควบคุมฮาร์ดแวร์นั้นมีความเฉพาะตัวเป็นอย่างมากยิ่ง เป็นเรื่องยากที่จะนำเอาความรู้จากการติดต่อควบคุมอุปกรณ์ตัวหนึ่ง ไปใช้งานกับอีกตัวหนึ่งได้ ทั้งนี้เพราะการเขียนโปรแกรมเพื่อติดต่อ และควบคุมในอดีตนั้นใช้งานผ่านภาษา Assembly ซึ่งเป็นภาษาที่ผูกติดกับฮาร์ดแวร์มาก แม้จะให้ผลการทำงานดีเยี่ยม แต่กลับทำให้การเขียนโปรแกรม แก้ไขโปรแกรมทำได้ยากมาก เพราะ Assembly เป็นภาษาที่เข้าใจยากมาก และเป็นภาษาที่ไม่เป็น โครงสร้าง ต่อมาได้ใช้งานภาษาโปรแกรมยุคใหม่ๆเพื่อใช้ในการเขียนโปรแกรมไม่ว่าจะเป็นภาษา C, Pascal ซึ่งภาษา C จะได้รับความนิยมสูงสุด เพราะให้ประสิทธิภาพการทำงานใกล้เคียงกับ Assembly มาก และการเขียนโปรแกรมด้วยภาษา C นั้นมีความเข้าใจง่ายกว่า

2.3.4 PHP

PHP เป็นภาษาจําพวก scripting language คำสั่งต่างๆจะเก็บอยู่ในไฟล์ที่เรียกว่า สคริปต์ (script) และเวลาใช้งานต้องอาศัยตัวแปลชุดคำสั่ง ตัวอย่างของภาษาสคริปต์ก็เช่น JavaScript, Perl เป็นต้น ลักษณะของ PHP ที่แตกต่างจากภาษาสคริปต์แบบอื่นๆ คือ PHP ได้รับการพัฒนาและออกแบบมาเพื่อใช้งานในการสร้างเอกสารแบบ HTML โดยสามารถสอดแทรกหรือแก้ไขเนื้อหาได้โดยอัตโนมัติ ดังนั้นจึงกล่าวได้ว่า PHP เป็นภาษาที่เรียกว่า server-side หรือ HTML-embedded scripting language เป็นเครื่องมือที่สำคัญชนิดหนึ่งซึ่งช่วยให้เราสามารถสร้างเอกสารแบบ Dynamic HTML ได้อย่างมีประสิทธิภาพและมีลูกเล่นมากขึ้น

เนื่องจากว่า PHP ไม่ได้เป็นส่วนหนึ่งของตัว Web Server ดังนั้นถ้าจะใช้ PHP ก็จะต้องดูก่อนว่า Web server นั้นสามารถใช้สคริปต์ PHP ได้หรือไม่ ยกตัวอย่างเช่น PHP สามารถใช้ได้กับ Apache Web Server และ Personal Web Server (PWP) สำหรับระบบปฏิบัติการ Windows 95/98/NT ในกรณีของ Apache เราสามารถใช้ PHP ได้สองรูปแบบคือ ในลักษณะของ CGI และ Apache Module ความแตกต่างอยู่ตรงที่ว่า ถ้าใช้ PHP เป็นแบบโมดูล PHP จะเป็นส่วนหนึ่งของ Apache หรือเป็นส่วนขยายในการทำงานนั่นเอง ซึ่งจะทำงานได้เร็วกว่าแบบที่เป็น CGI เพราะว่า ถ้าเป็น CGI แล้วตัวแปลชุดคำสั่งของ PHP ถือว่าเป็นแค่โปรแกรมภายนอก ซึ่ง Apache จะต้องเรียกขึ้นมาทำงานทุก

ครั้ง ที่ต้องการใช้ PHP ดังนั้น ถ้ามองในเรื่องของประสิทธิภาพในการทำงาน การใช้ PHP แบบที่เป็น โมดูลหนึ่งของ Apache จะทำงาน ได้มีประสิทธิภาพมากกว่า

2.3.5 HTML

HTML (Hyper Text Markup Language) เป็นภาษาหลักของ World Wide Web (WWW เป็นบริการหนึ่งที่ใช้ในอินเทอร์เน็ต) ปัจจุบันมาตรฐาน HTML ถูกดูแลภายใต้องค์กร W3C ซึ่งมาตรฐานเวอร์ชันล่าสุดคือ HTML-4.01 แต่เดิม HTML ถูกใช้สำหรับสร้างทุกส่วนประกอบของเว็บเพจ ตั้งแต่การกำหนดสี สร้างเฟรม สร้างตาราง สร้างฟอร์ม กำหนดขนาดอักษร ฯลฯ แต่ปัจจุบันมาตรฐาน HTML ถูกปรับแต่งใหม่ให้ทำหน้าที่อย่างจำกัดมากยิ่งขึ้น โดยมุ่งเฉพาะการกำหนดโครงสร้างของเว็บเพจเท่านั้น เช่น การกำหนดส่วนหัว HEAD และ BODY หรือ การสร้างตัวเชื่อม (นิยมเรียกกันว่าลิงค์) เป็นต้น ส่วนหน้าที่อื่นๆ ได้ถูกโอนไปเป็นมาตรฐานหรือภาษาตัวใหม่แทน เช่น CSS และ XForms

คำสั่ง HTML แต่ละตัวนิยมเรียกกันว่า "แท็ก" ซึ่งแท็กทุกตัวจะมีเครื่องหมาย < และ > ปิดท้าย ตัวอย่างเช่น
, <P>, <HEAD> ฯลฯ ลักษณะการทำงานของคำสั่ง HTML ส่วนใหญ่ จะต้องมีการกำหนดจุดเริ่มต้นและจุดลงท้าย เช่น หากต้องการทำให้อักษรแสดงผลเป็นตัวหนาเราต้องใช้คำสั่ง แล้วพิมพ์ข้อความที่ต้องการทำให้เป็นตัวหนา จากนั้นปิดคำสั่งด้วย เช่น Web-based electric control system for household เป็นต้น ปัจจุบันคำสั่ง HTML มีอยู่เป็นจำนวนมาก แต่เบราว์เซอร์หลักเช่น Netscape และ Internet Explorer ยังไม่สามารถแสดงคำสั่งได้ทั้งหมด นอกจากนี้ในบางคำสั่งเบราว์เซอร์ทั้งสองตัวยังแสดงผลแตกต่างกันไปอีกด้วย

HTML ถือได้ว่าเป็นภาษาที่ง่ายที่สุดบนเว็บ โดยที่คำสั่งจะจดจำง่ายและสื่อความหมายได้ในตัวเองเป็นส่วนใหญ่ ที่สำคัญปัจจุบันมีเครื่องมือทั้งที่ต้องเสียเงินและแจกฟรีบนอินเทอร์เน็ตมากมายให้ได้ทดลองใช้ ซอฟต์แวร์เหล่านี้ช่วยให้การเขียนเว็บเพจด้วย HTML เป็นเรื่องง่ายยิ่งขึ้น และไม่ต้องไปวุ่นวายกับโค้ด HTML เลย เพียงดึงวัตถุที่ต้องการไปวางตำแหน่งที่ต้องการ พิมพ์ข้อความลงไป ซอฟต์แวร์เหล่านี้ก็จะสร้างโค้ด HTML ให้อัตโนมัติ สำหรับส่วนขยายไฟล์ HTML นั้นนิยมใช้กันทั้ง .htm และ .html

2.4 องค์ประกอบทางด้าน Network

ในการแสดงผลออกทาง Web-based นั้นต้องติดต่อผ่านทาง Network จึงได้ศึกษาในส่วน
ของ Network ดังนี้

2.4.1 ความรู้เบื้องต้นเกี่ยวกับเครือข่าย (Understand Basic Networking)

เครือข่าย (Network) หมายถึง กลุ่มสิ่งที่เราสนใจ (Entity หรือ Object) ที่มีปฏิสัมพันธ์
โต้ตอบกับสิ่งแวดล้อมรอบข้าง มีความเกี่ยวเนื่องซึ่งกันและกัน สามารถแลกเปลี่ยนข้อมูลข่าวสาร
หรือโต้ตอบกันไปมาเพื่อให้บรรลุวัตถุประสงค์ หรือเป้าหมายอย่างใดอย่างหนึ่งร่วมกันได้
เครือข่ายคอมพิวเตอร์ (Computer Network) หมายถึง การรวมกลุ่มของเครื่องคอมพิวเตอร์ เพื่อ
แลกเปลี่ยนข้อมูลข่าวสารระหว่างกัน โดยผ่านสื่อ (Media หรือ Channel) ที่ใช้ในการเชื่อมต่อ ข้อมูล
ที่ส่งผ่านระหว่างเครือข่ายคอมพิวเตอร์นั้นเราจะเรียกว่า “Data Communication Network” ซึ่งไม่ว่า
จะนั่งทำงานที่ใดก็สามารถติดต่อ สื่อสาร แลกเปลี่ยนข้อมูลระหว่างกัน โดยไม่ขีดจำกัดทางด้าน
สถานที่ เวลา และระยะทาง โพรโตคอล (Protocol) คือ กฎ ระเบียบ หรือข้อปฏิบัติที่ใช้เป็น
มาตรฐานในการกำหนดบทบาท และหน้าที่ ต่าง ๆ ที่เกี่ยวกับการสื่อสารข้อมูลโดยใช้ระบบ
เครือข่ายคอมพิวเตอร์

2.4.2 การจัดแบ่งประเภทของเครือข่ายคอมพิวเตอร์

เครือข่ายคอมพิวเตอร์นั้นสามารถแบ่งตามขนาดออกได้เป็น 3 กลุ่มดังนี้

1. ระยะทางระหว่างเครื่องคอมพิวเตอร์กับอุปกรณ์เครือข่ายคอมพิวเตอร์ หรือระหว่างเครื่อง
คอมพิวเตอร์กับเครื่องคอมพิวเตอร์ที่มีระยะทางระหว่าง 10 เมตร น้อยกว่าหรือเท่ากับ
1,000 เมตร เราจะจัดประเภทของเครือข่ายประเภทนี้ว่า เครือข่ายเฉพาะพื้นที่ (Local Area
Network : LAN)
2. ระยะทางระหว่างเครื่องคอมพิวเตอร์กับอุปกรณ์เครือข่ายคอมพิวเตอร์ หรือระหว่างเครื่อง
คอมพิวเตอร์กับเครื่องคอมพิวเตอร์ที่มีระยะทางระหว่าง 1,000 เมตรขึ้นไป น้อยกว่าหรือ
เท่ากับ 10,000 เมตร เราจะจัดประเภทของเครือข่ายประเภทนี้ว่า เครือข่ายระดับเมือง
(Metropolitan Area Network : MAN)
3. ระยะทางระหว่างเครื่องคอมพิวเตอร์กับอุปกรณ์เครือข่ายคอมพิวเตอร์ หรือระหว่างเครื่อง
คอมพิวเตอร์กับเครื่องคอมพิวเตอร์ที่มีระยะทางระหว่าง 100,000 เมตร น้อยกว่าหรือ
เท่ากับ 100,000,000 เมตร เราจะจัดประเภทของเครือข่ายประเภทนี้ว่า เครือข่ายขนาดใหญ่
(Wide Area Network: LAN)

2.4.3 ระบบเครือข่ายเฉพาะพื้นที่ (Local Area Network)

5000071

เครือข่ายเฉพาะพื้นที่เป็นเครื่องที่นำเอาเครื่องคอมพิวเตอร์หลาย ๆ เครื่องที่อยู่ ใกล้ ๆ กัน หรืออยู่ในบริเวณเดียวกันเข้ามาต่อเชื่อมกัน โดยผ่านสื่อกลางในการสื่อสารข้อมูลและ NIC ดังนั้น

LANs จึงถูกออกแบบมาให้สามารถทำงาน ได้ดังนี้

รศ.

พ4715

- ปฏิบัติงานในลักษณะพื้นที่ที่จำกัดได้ หรือพื้นที่ในการปฏิบัติงานไม่ใหญ่มากนัก 2549.
- อนุญาตให้มีการใช้สื่อกลางในการส่งผ่านข้อมูลได้หลากหลาย (Multi-access) และใช้ความเร็วสูงในการส่งผ่านข้อมูล
- สามารถควบคุม บริหารงาน จัดการเครือข่ายโดยผู้บริหาร โครงข่ายได้
- สามารถใช้งาน ได้ตลอดเวลา อนุญาตให้ทำการเชื่อมต่อ ได้ตลอด 24 ชั่วโมง
- ทำการเชื่อมต่อกับอุปกรณ์เครือข่าย ได้ตามความเหมาะสม

อุปกรณ์เครือข่ายคอมพิวเตอร์เฉพาะพื้นที่ ได้แก่ Bridge, Ethernet Switch, Hub, ATM Switch, and Router เป็นต้น

2.4.4 ระบบเครือข่ายขนาดใหญ่ (Wide Area Network)

เครือข่ายขนาดใหญ่เป็นเครือข่ายที่นำเอาเครือข่ายเฉพาะพื้นที่หลาย ๆ เครือข่าย เข้ามา ต่อเชื่อมกัน โดยไม่มีการจำกัดรูปแบบที่แน่นอนตายตัว ทำให้ขนาดของเครือข่ายที่ใช้งานมีการ ขยายตัวเพิ่มมากขึ้นอย่างไม่มีขีดจำกัด เทคโนโลยีของเครือข่ายขนาดใหญ่ได้แก่

- MODEM (Modulation and Demodulation)
- ISDN (Integrated Services Digital Network)
- DSL (Digital Subscriber Loop)
- Frame Relay
- ATM (Asynchronous Transfer Mode)
- T-Carrier Series
- SONET (Synchronous Optical Network)

ดังนั้นเครือข่ายขนาดใหญ่ (WANs) จึงเป็นเครือข่ายที่ถูกออกแบบมาให้สามารถทำงาน ได้ดังนี้

- ปฏิบัติงานในพื้นที่ที่มีขนาดใหญ่ได้ หรือไม่จำกัดพื้นที่ในการปฏิบัติงาน

- อนุญาตให้มีการใช้สื่อกลางในการส่งผ่านข้อมูลได้เฉพาะที่สามารถส่งผ่านข้อมูลได้แบบอนุกรม (Serial Interface) เท่านั้น สามารถส่งผ่านข้อมูลได้ด้วยความเร็วต่ำ
- สามารถต่อเชื่อม / ให้บริการ ได้ทั้งแบบตลอดเวลา และนอกเหนือเวลาทำการ
- สามารถทำการต่อเชื่อมกับอุปกรณ์เครือข่ายที่ใช้กับเครือข่ายขนาดใหญ่เท่านั้น

อุปกรณ์เครือข่ายคอมพิวเตอร์ขนาดใหญ่ ได้แก่ Router, WAN Bandwidth Switch, MODEM CSU/DSU, TA/NT1 and Communication Server เป็นต้น



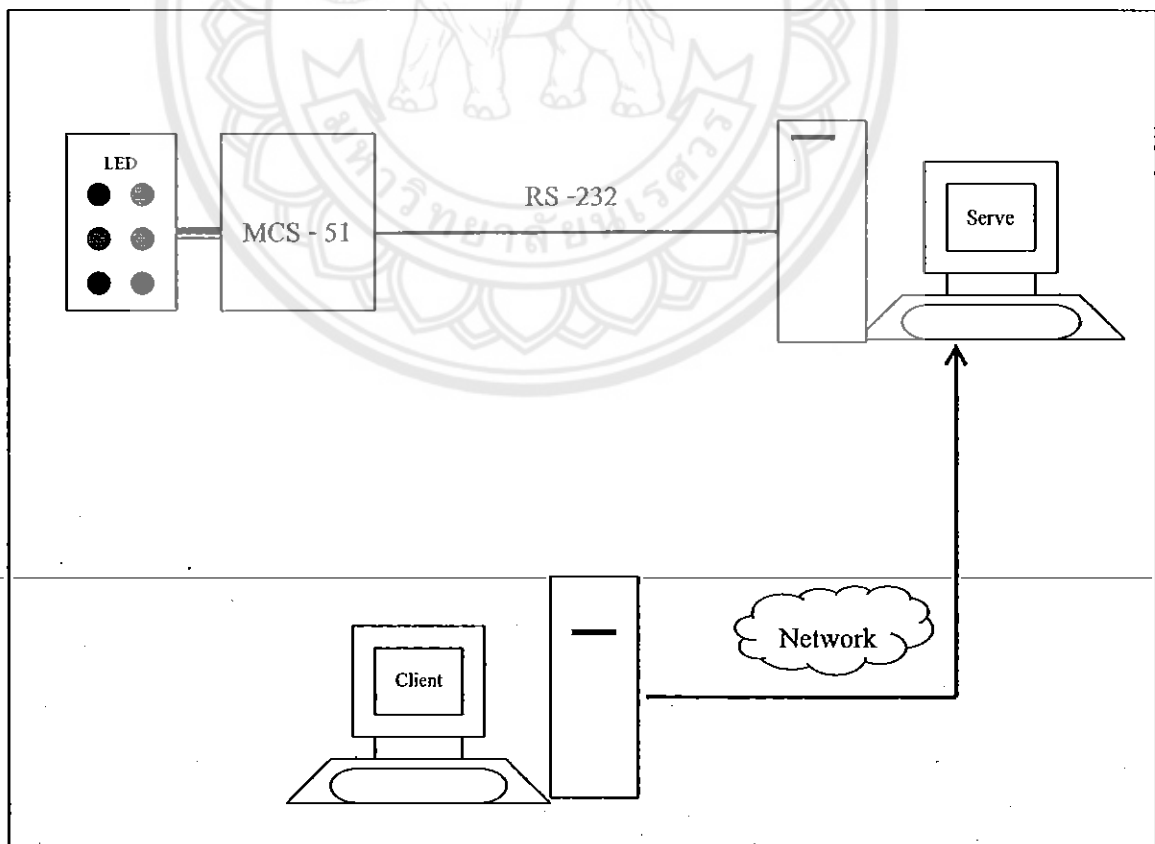
บทที่ 3

วิธีการดำเนินการ

ในการดำเนินงานสร้างระบบควบคุมไฟฟ้าภายในบ้านผ่านเว็บ ทีมผู้จัดทำได้ทำการแยกระบบการทำงานออกเป็น 2 ส่วน คือ ส่วนโปรแกรมที่ติดต่อกับ Web-based และ โปรแกรมที่ติดต่อกับ Hardware ซึ่งทั้งสองโปรแกรมนี้ถูกเชื่อมต่อกันโดย text files โดยมีวิธีการดำเนินงานดังต่อไปนี้

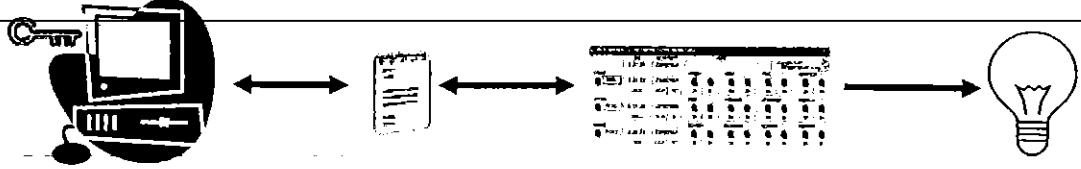
3.1 สถาปัตยกรรมของระบบ

การควบคุมการ ปิด-เปิด ดวงไฟสามารถควบคุมผ่านทางเว็บ โดยเมื่อเครื่องลูกข่ายทำการร้องขอการปิด-เปิด ไฟไปยังเครื่องแม่ข่าย เครื่องแม่ข่ายก็จะทำการเขียนสถานะ (ปิด-เปิด) ของห้องที่ต้องการจะ ปิด-เปิด ลงไฟล์ จากนั้น โปรแกรมที่ทำการควบคุมเครื่อง MCS-51 ก็จะทำการอ่านไฟล์ที่เครื่องแม่ข่ายเขียนแล้วส่งค่าไปยัง เครื่อง MCS-51 เครื่อง MCS-51 ก็จะแสดงผลออกมาทางดวงไฟ LED ตามที่เครื่องลูกข่ายร้องขอ ดังแสดงในรูป 3.1



รูปที่ 3.1 สถาปัตยกรรมของระบบ (System Architecture)

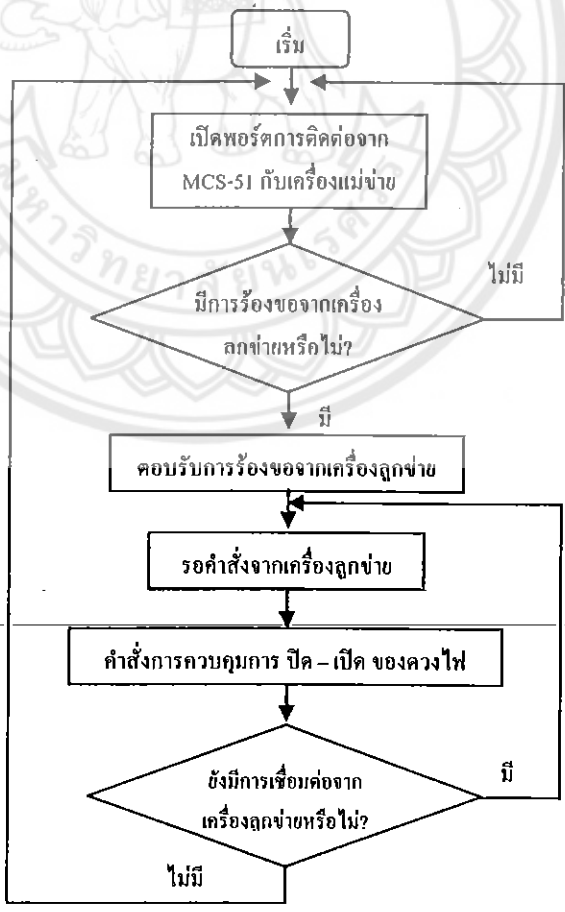
ในการทำงานของเครื่องแม่ข่ายจะประกอบไปด้วย 2 โปรแกรม คือ โปรแกรมที่ทำหน้าที่แสดงผลทางหน้าเว็บและโปรแกรมที่ควบคุมฮาร์ดแวร์ ซึ่งทั้งสองโปรแกรมนี้นี้ติดต่อกันผ่าน Text files ดังแสดงในรูปที่ 3.2



รูปที่ 3.2 สถาปัตยกรรมของซอฟต์แวร์ (Software Architecture)

3.2 แผนภาพการควบคุมดวงไฟโดยควบคุมและแสดงผลผ่านทางเว็บ

ในส่วนของโปรแกรมในการควบคุมดวงไฟนั้นจะแบ่งออกเป็น 2 ส่วน คือ โปรแกรมที่ทำงานติดต่อกับ Web-based โดยจะเป็นส่วนที่ทำหน้าที่ในการแสดงผลและรับค่าจากเครื่องลูกข่ายแล้วทำการติดต่อไปยังเครื่อง MCS-51 โปรแกรมส่วนที่ทำหน้าที่เป็นตัวควบคุม อุปกรณ์ที่ใช้ในการแสดงผล โดยโปรแกรมนี้อาจทำหน้าที่ให้ MCS-51 ทำงานรวมทั้งทำให้ LED ติด-ดับ โดยการควบคุมจากเครื่องลูกข่าย ดังจะได้แสดงการทำงานดังรูปที่ 3.3



รูปที่ 3.3 แผนภาพแสดงการทำงานของโปรแกรมส่วนที่ติดต่อกับ Web-based

3.3 การออกแบบโปรแกรมการควบคุมดวงไฟโดยควบคุมและแสดงผลผ่านทางเว็บ

3.3.1 Security ความปลอดภัยในการส่งข้อมูลผ่าน Internet

ในระหว่างการพัฒนาเครือข่าย Internet ในระยะเริ่มแรกนั้น ไม่ได้มีการเน้นในการพัฒนา ด้านความปลอดภัยในการส่งข้อมูลบนเครือข่าย เนื่องจากในระยะนั้นเครือข่าย Internet นี้ถูกใช้ในการติดต่อสื่อสารระหว่างกลุ่มนักวิจัยในมหาวิทยาลัยและสถาบันต่างๆ ไม่กี่กลุ่ม ซึ่งมีความรู้จักคุ้นเคยกันและมีความเชื่อถือต่อกันและกัน ดังนั้นข้อมูลที่ถูกส่งไปบนเครือข่าย Internet จึงเป็นลักษณะของข้อมูลที่ไม่ได้เข้ารหัสลับใดๆ หรือที่เรียกกันว่า clear text จนทุกวันนี้การส่งข้อมูลส่วนใหญ่บนเครือข่าย Internet ก็ยังคงลักษณะนี้อยู่

ในปัจจุบัน ผู้ใช้เครือข่าย Internet มีจำนวนเพิ่มขึ้นอย่างมาก และเครือข่ายนี้ถูกใช้งานในรูปแบบต่างๆ มากมายหลายรูปแบบ โดยเฉพาะอย่างยิ่งในการพาณิชย์อิเล็กทรอนิกส์ (e-commerce) ซึ่งผู้ซื้อและผู้ขายจะต้องส่งข้อมูลที่เป็นความลับถึงกันและกัน เช่น ผู้ซื้อส่งหมายเลขบัตรเครดิต หรือที่อยู่และเบอร์โทรศัพท์โดยส่งผ่าน ไปบนเครือข่าย Internet หากข้อมูลเหล่านี้ถูกส่งไปแบบธรรมดา ก็จะเป็นการค่อนข้างง่ายที่ผู้ไม่หวังดีจะสามารถดักจับข้อมูลเหล่านี้ (sniffing) แล้วนำไปใช้ได้ เนื่องจากข้อมูลเหล่านี้อยู่ในรูปของ clear text ผู้ดักจับข้อมูลก็จะสามารถนำข้อมูลนั้นไปใช้ได้ทันที

ในอีกกรณีหนึ่ง บนเครือข่าย Internet นี้ ผู้ใช้สามารถเชื่อมต่อและใช้งานเครื่องคอมพิวเตอร์ใดๆ ก็ได้หากผู้ใช้นั้น ได้รับอนุญาตและสามารถพิสูจน์ตนเองโดยใช้ username และ password ที่ถูกต้องบนเครื่องนั้นๆ โดยจะมีโปรแกรมที่ใช้ช่วยในการเชื่อมต่อและใช้งานนั้น เช่น telnet, rsh, rlogin, rcp, และ ftp เป็นต้น โปรแกรมเหล่านี้ส่งข้อมูลตามแบบมาตรฐานดั้งเดิมของเครือข่าย Internet กล่าวคือ ส่งข้อมูลทุกอย่าง (รวมทั้ง username และ password) ในรูปของ clear text ดังนั้นหากมีผู้ดักจับข้อมูลเกี่ยวกับ username และ password ได้ ผู้นั้นก็จะสามารถนำเอา username และ password นี้ไปใช้ในการเชื่อมต่อและใช้งานเครื่องคอมพิวเตอร์เครื่องนั้นได้ต่อไป

เนื่องจากปัญหาที่กล่าวมานี้เป็นปัญหาที่ค่อนข้างใหญ่เพราะการ Sniffing นั้นสามารถกระทำได้อย่างค่อนข้างง่าย จึงได้มีการคิดแก้ไขปัญหานี้ขึ้น โดย Netscape ได้คิดค้น protocol ใหม่ขึ้นมาคือ Secure Socket Layer Protocol (SSL) ซึ่ง SSL จะเข้ารหัสลับข้อมูลใดๆ ก่อนที่ข้อมูลนั้นจะถูกส่งไปบนเครือข่าย Internet ดังนั้น หากผู้ไม่หวังดีสามารถดักจับข้อมูลนั้นไปได้ ผู้นั้นก็ไม่สามารถที่จะนำข้อมูลนั้นไปใช้ได้เพราะเขาไม่สามารถตีความข้อมูลนั้นได้

SSL นั้นได้รับการยอมรับอย่างกว้างขวางบน World Wide Web ในการใช้สำหรับตรวจสอบและเข้ารหัสลับการติดต่อสื่อสารระหว่าง client และ server หน้าที่ของ SSL จะแบ่งออกเป็น 3 ส่วนใหญ่ๆ คือ

1. การตรวจสอบ Server ว่าเป็นตัวจริง: ตัวโปรแกรม Client ที่มีขีดความสามารถในการสื่อสารแบบ SSL จะสามารถตรวจสอบเครื่อง Server ที่คนกำลังจะไปเชื่อมต่อได้ว่า Server นั้นเป็น Server ตัวจริงหรือไม่ โดยใช้เทคนิคการเข้ารหัสแบบ public key ในการตรวจสอบใบรับรอง (certificate) และ public ID ของ server นั้น (โดยที่เมื่อครั้งที่ client เชื่อมถือเป็นผู้ออกใบรับรองและ public ID ให้แก่ server นั้น) หน้าที่นี้ของ SSL เป็นหน้าที่ที่สำคัญ โดยเฉพาะอย่างยิ่งในกรณีที่ Client ต้องการที่จะส่งข้อมูลที่เป็นความลับ (เช่น หมายเลข credit card) ให้กับ server ซึ่ง client จะต้องตรวจสอบก่อนว่า Server เป็นตัวจริงหรือไม่

2. การตรวจสอบว่า Client เป็นตัวจริง Server ที่มีขีดความสามารถในการสื่อสารแบบ SSL จะใช้เทคนิคเช่นเดียวกับในหัวข้อที่แล้วในการตรวจสอบ Client หรือผู้ใช้ว่าเป็นตัวจริงหรือไม่ โดยจะตรวจสอบใบรับรองและ public ID (ที่เมื่อครั้งที่ server เชื่อมถือเป็นผู้ออกให้) ของ Client หรือผู้ใช้นั้น

หน้าที่นี้ของ SSL จะมีประโยชน์ในกรณีเช่น ธนาคารต้องการที่จะส่งข้อมูลกลับทางการเงินให้แก่ลูกค้าของตนผ่านทางเครือข่าย Internet (Server ก็จะต้องตรวจสอบ Client ก่อนว่าเป็น Client นั้นจริง)

3. การเข้ารหัสลับการเชื่อมต่อ: ในกรณีนี้ ข้อมูลทั้งหมดที่ถูกส่งระหว่าง Client และ Server จะถูกเข้ารหัสลับ โดยโปรแกรมที่ส่งข้อมูลเป็นผู้เข้ารหัสและโปรแกรมที่รับข้อมูลเป็นผู้ถอดรหัส (โดยใช้วิธี public key) นอกจากการเข้ารหัสลับในลักษณะนี้แล้ว SSL ยังสามารถปกป้องความถูกต้องสมบูรณ์ของข้อมูลได้อีกด้วย กล่าวคือ ตัวโปรแกรมรับข้อมูลจะทราบได้หากข้อมูลถูกเปลี่ยนแปลงไปในขณะกำลังเดินทางจากผู้ส่งไปยังผู้รับ

3.3.2 Server ที่ใช้เพื่อให้มันเป็นเครื่องแม่ข่าย

IIS: Internet Information Service เป็นเครื่องมือที่ไว้ให้เครื่องแม่ข่ายทำงานร่วมกับ PHP Compiler และ SSL certificate ซึ่งให้บริการบน Transport Layer ของ Microsoft Windows XP² Professional ในงานนี้ใช้ IIS V5.1 แสดงผลเว็บด้วย Transmission Control Protocol: port 80 ในการใช้ IIS นี้จะช่วยให้ง่ายต่อการจัดการ Web sites และ Web Server บริการ IIS นี้จะต้องติดตั้งเพิ่มเอง แต่ IIS version นี้ออกแบบให้เหมาะสมกับใช้งานในบ้านหรือสำนักงาน จึงให้บริการลูกข่ายได้เพียง 10 connections เท่านั้น

ใช้โปรโตคอล HTTPS ซึ่งเป็นโปรโตคอลที่ปรับปรุงขึ้นโดยรองรับการเข้ารหัส ทำให้มีความเชื่อมั่นสูงในการรับส่งข้อมูลระหว่าง บราวเซอร์ของผู้ใช้ และเซิร์ฟเวอร์ ผ่านตัวกลางที่ไม่ทราบว่าจะมีการดักจับข้อมูลหรือไม่ เว็ปเซิร์ฟเวอร์ปัจจุบันได้รองรับการทำงานในโปรโตคอลนี้อยู่แล้ว และการใช้งานก็เพียงแค่ตั้งค่าเพียงเล็กน้อยเท่านั้น มีประเด็นสำคัญอันหนึ่งก็คือ certificate โดยจะใช้ในการรับรองความถูกต้องที่มาของเว็บไซต์ และใช้ในกระบวนการ

เข้ารหัสนั้น โดยปกติแล้วจะมีบริษัทที่รับดูแลในเรื่องนี้อยู่ และเราสามารถใช้บริการจากบริษัทเหล่านั้นได้ ในทางปฏิบัติ หลายเว็บไซต์มิได้ทำการติดต่อขอรับบริการดังกล่าว แต่อาศัยการสร้าง certificate นี้ขึ้นมาเอง โดยเมื่อเบราว์เซอร์เข้าเรียกใช้เว็บของเราในโปรโตคอล HTTPS ในครั้งแรก จะมีการเตือนให้ผู้ใช้ทราบว่า certificate นี้มิได้รับรองโดยบริษัทที่ดูแลเรื่องนี้โดยตรง ซึ่งหากเป็นเว็บเซิร์ฟเวอร์ที่ใช้กันภายในองค์กร ก็สามารถส่งไฟล์ certificate นี้ไปติดตั้งยังเครื่องต่างๆ ได้ โดยผู้ใช้ติดตั้งเอง หรืออาจจะบอกข้อมูลเบื้องต้นที่มีอยู่ใน certificate นั้นให้ผู้ใช้ตรวจสอบเองได้ ก่อนทำการติดตั้ง และเมื่อผู้ใช้ติดตั้ง certificate ลงไปแล้ว ก็จะสามารถใช้งานเว็บไซต์เดิมในภายหลังได้ โดยไม่มีการร้องเตือนเรื่อง certificate อีก

ก่อนอื่น ทำการสร้าง Certificate สำหรับเว็บเซิร์ฟเวอร์ คอมพิวเตอร์ที่จะใช้สร้าง certificate จะต้องมี OpenSSL โดยขั้นตอนการใช้งานเพื่อสร้าง self-signed certificate มีดังนี้

1. สร้าง RSA private key ในที่นี้สามารถกำหนดการเข้ารหัสแบบ Triple-DES และคีย์มีขนาด 1024 บิต ใน PEM format โดยจะต้องกำหนดชื่อเซิร์ฟเวอร์ เป็นชื่อเซิร์ฟเวอร์ของเว็บไซต์ เช่น เว็บนี้คือ ecpe.nu.ac.th ชื่อจะต้องเป็น ecpe.nu.ac.th.key ซึ่งจะได้ไฟล์ในชื่อเว็บไซต์เป็น .key

2. สร้างไฟล์ CSR (Certificate Signing Request) จากนั้นจึงสร้างไฟล์โดยจะแทรกรายละเอียดต่างๆ เกี่ยวกับเว็บไซต์ลงไป เพื่อในเวลาที่ผู้ใช้เปิดเว็บของเรา จะสามารถตรวจสอบรายละเอียดเบื้องต้นเกี่ยวกับเว็บไซต์ได้

3. สร้าง Self-signed certificate เพื่อไว้ให้เบราว์เซอร์มาอ่านไป โปรแกรมจะทำการสร้างไฟล์ .CRT ขึ้นมา

การสร้าง IIS Certificate Request

1. เปิด Internet Service Manager
2. เลือกไปยังเว็บไซต์ที่ต้องการติดตั้ง SSL
3. คลิกที่ tab Directory security
4. ภายในส่วน Secure communication คลิกที่ Server Certificate จะเข้าสู่ Server Certificate Request Wizard
5. คลิก Next เลือกที่ Create a new certificate แล้วคลิก Next
6. เลือกที่ Prepare a new request but send it later แล้วคลิก Next
7. ใส่ชื่อเว็บไซต์เลือก bit length เป็น 1024 แล้วคลิก Next
8. ใส่ชื่อองค์กรและแผนกแล้วคลิก Next
9. ใส่ชื่อเว็บไซต์แล้วคลิก Next
10. เลือกประเทศ, จังหวัด, อำเภอ/เขต แล้วคลิก Next

11. ใส่ข้อมูลผู้ดูแลระบบ หรือผู้ดูแลเว็บไซต์
12. เลือกให้เซฟไปที่ C:\ssl\requests\certreq.txt แล้วคลิก Next
13. จะมีหน้าจอแสดงข้อมูลที่กรอกไปแล้วคลิก Next เราก็จะได้ไฟล์ certreq.txt

ทำการติดตั้ง Certificate เข้ากับ IIS

1. เปิด Internet Service Manager
2. คลิกขวาที่เว็บไซต์ที่ต้องการติดตั้ง
3. เลือกที่ tab Directory security คลิกที่ Server Certificate จะมี Dialog ของ Website Certificate Wizard คลิก Next
4. เลือกที่ Process the pending request and install the certificate คลิก Next
5. ทำการเลือกไปยัง C:\ssl\certs\iisx509.cer คลิก Next
6. จะมีหน้าต่างแสดงรายละเอียดของ certificate คลิก Next ไปจนจบ

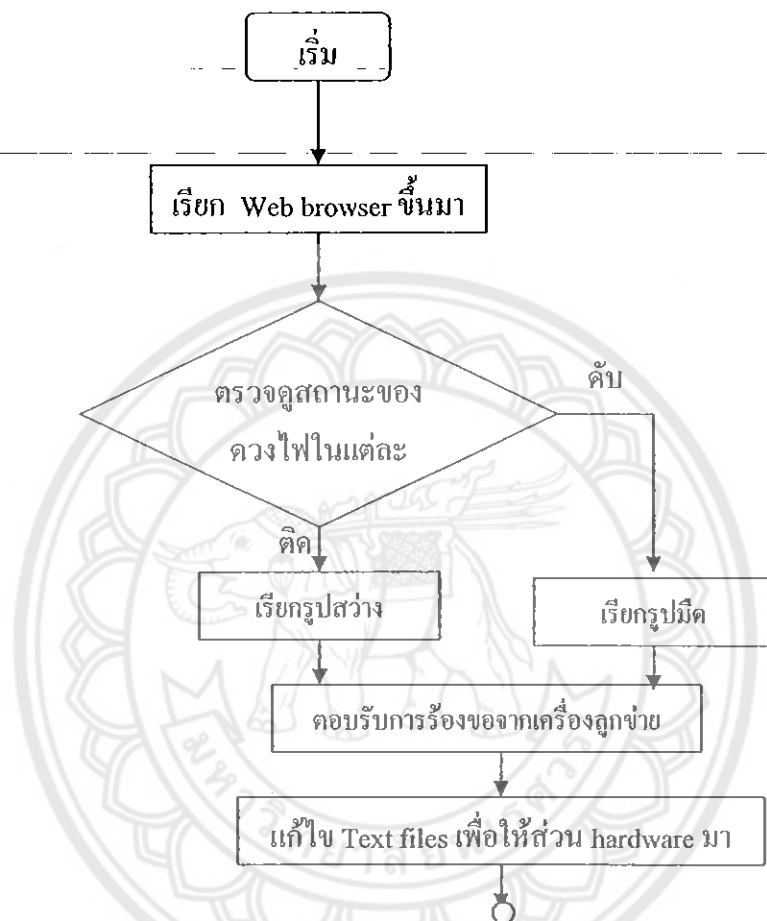
การแสดงผล SSL (Secure Socket Layer)



รูปที่ 3.4 แผนภาพแสดงการทำงานของโปรแกรม SSL

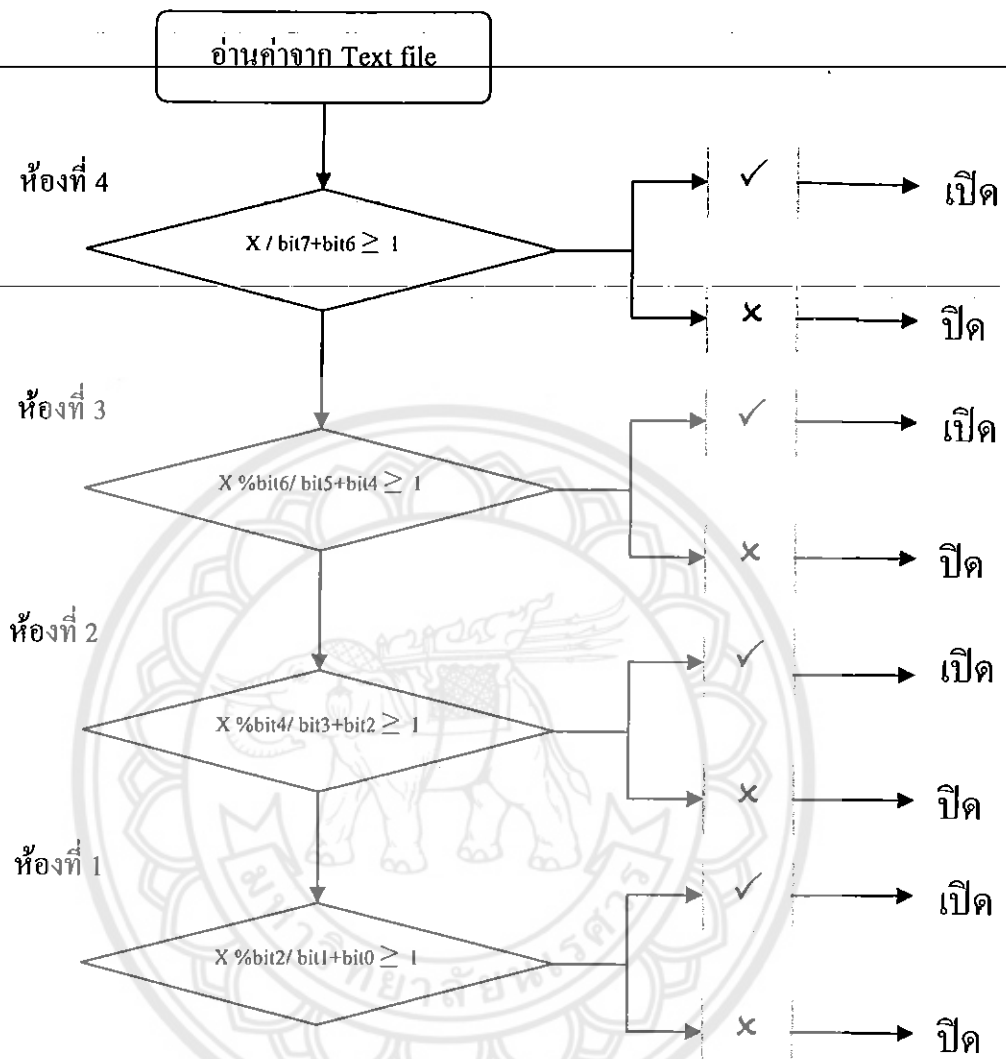
3.4 การเขียนโปรแกรมเพื่อควบคุมไฟบ้านผ่านเว็บ

1. โปรแกรมในส่วน web server



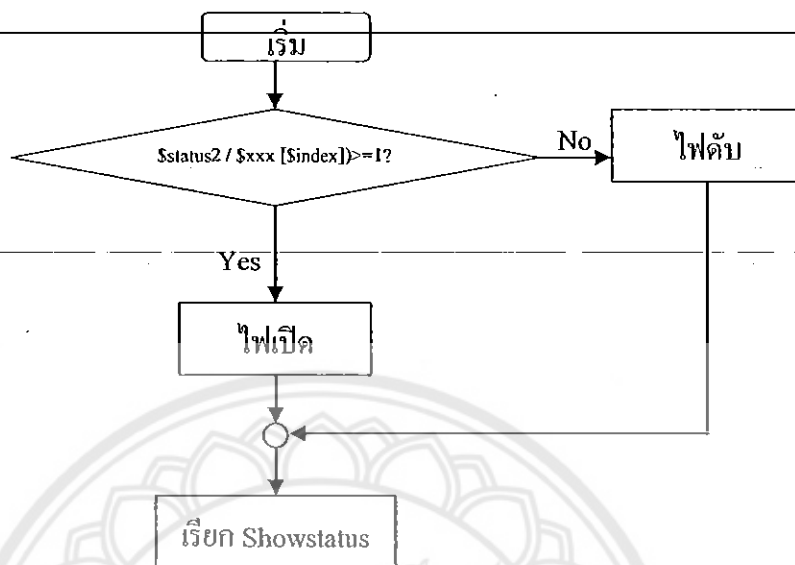
รูปที่ 3.5 แผนภาพแสดงการทำงานในส่วนของโปรแกรม web server

2. โปรแกรมในส่วนของการตรวจสอบสถานะและการอ่านค่าจาก Text file



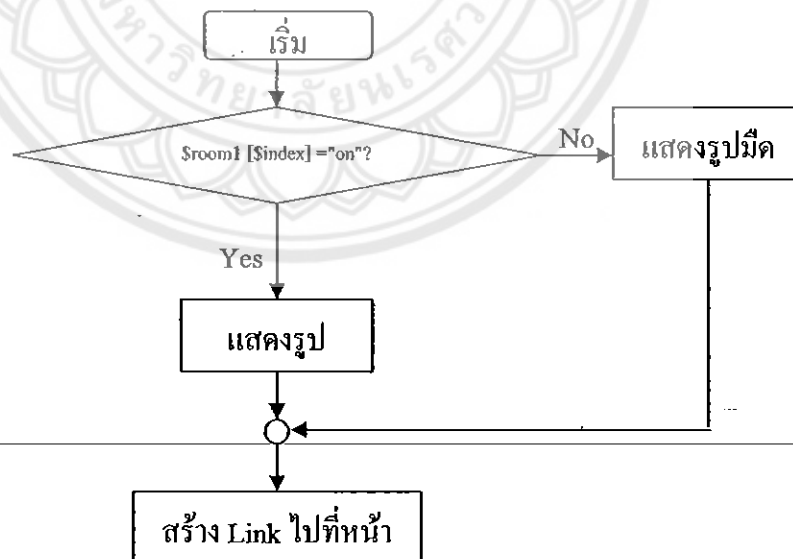
รูปที่ 3.6 แสดงการทำงานของโปรแกรมในส่วนของการตรวจสอบสถานะของแต่ละห้อง

3. โปรแกรมในส่วนของการ check สถานะของแต่ละห้องบนหน้าเว็บเพจ



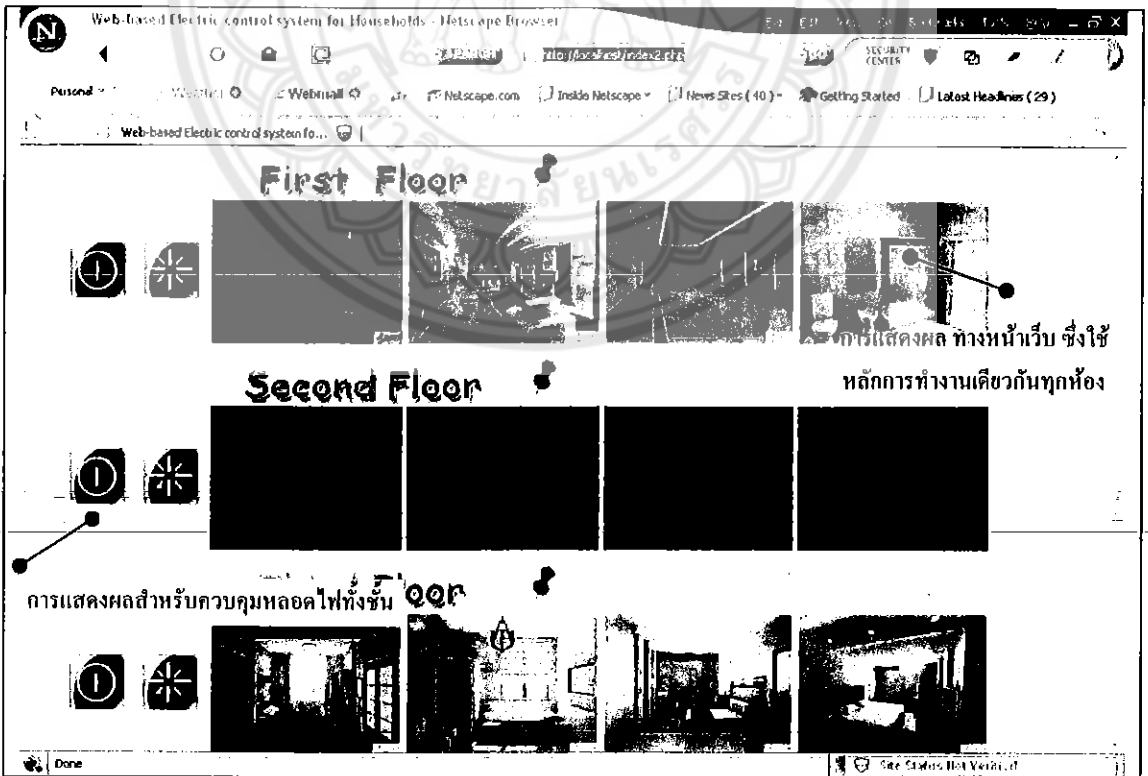
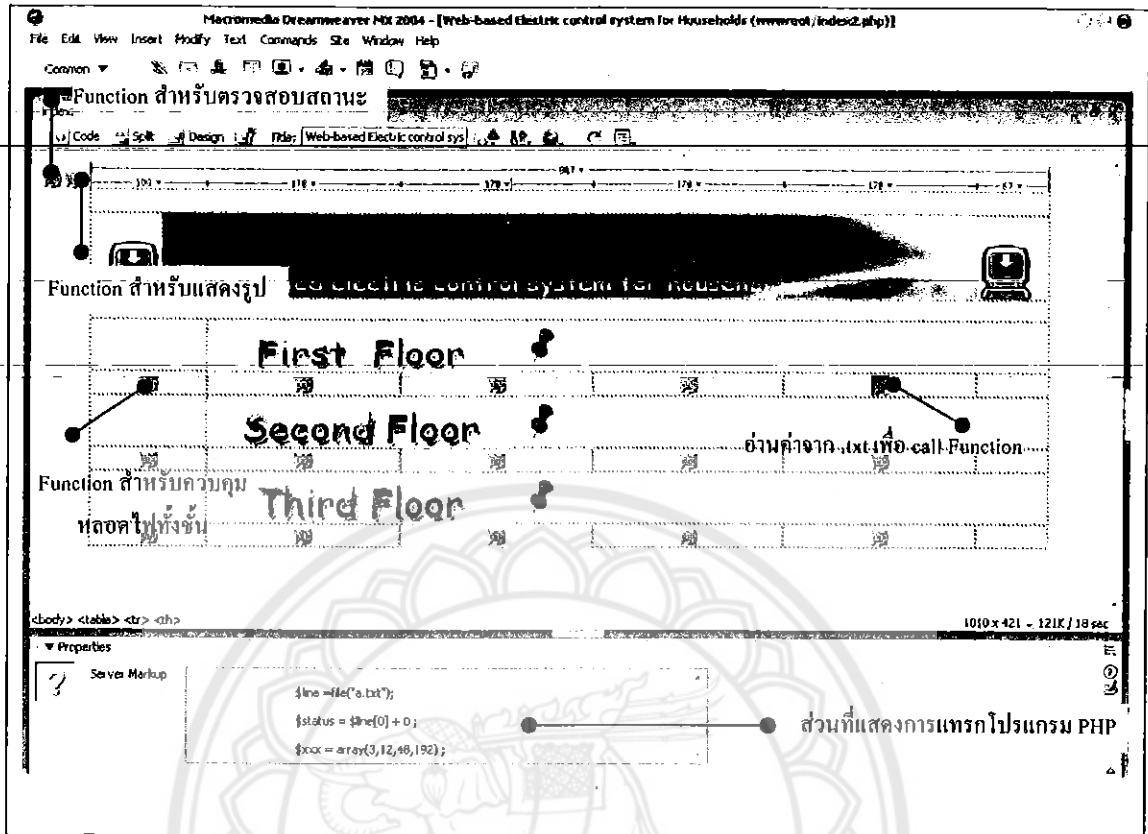
รูปที่ 3.7 แสดงการทำงานของโปรแกรมการตรวจสอบสถานะของแต่ละห้องบนหน้าเว็บ

4. โปรแกรมในส่วนของการแสดงผลบนเว็บ



รูปที่ 3.8 แสดงการทำงานของโปรแกรมการแสดงผลภาพบนเว็บ

5. การสร้างเว็บ ติดต่อกับ Text file

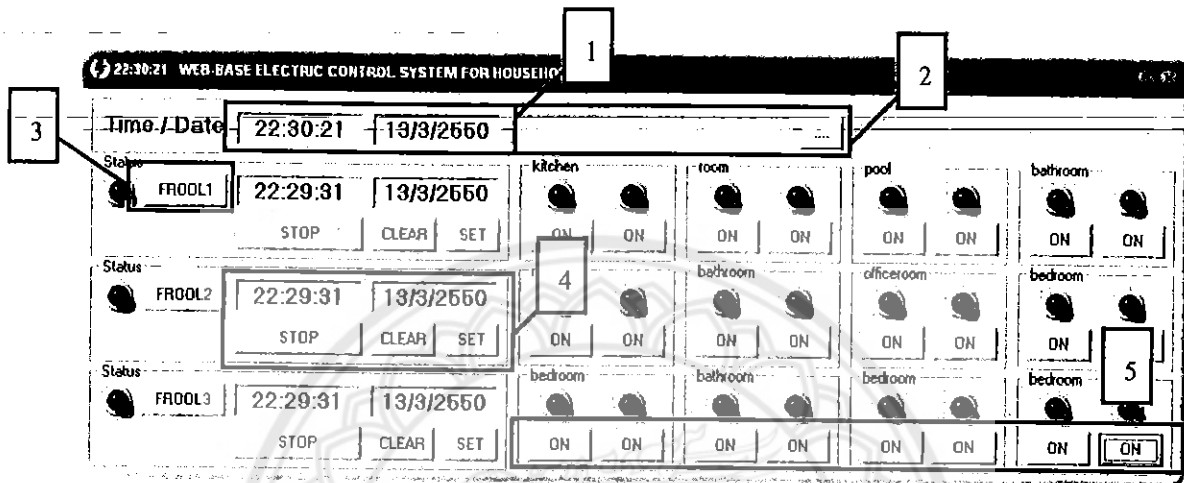


รูปที่ 3.9 แสดงการทำงานภาพรวมของหน้าเว็บที่ทำการติดต่อกับ Text file

3.5 การควบคุมดวงไฟและแสดงผลผ่านทาง โปรแกรมควบคุมทางฝั่งแม่ข่าย

1. การออกแบบรูปร่างหน้าตาโปรแกรม

โปรแกรมส่วนนี้จะเป็น โปรแกรมที่ทำงานบนเครื่องแม่ข่ายโดยจะมีหน้าที่ ควบคุมการ ปิด-เปิดดวงไฟ ที่ได้รับคำสั่งจากผู้ใช้โดยตรง หรือรับคำสั่งจากลูกข่าย โดยสร้างจากภาษา VISUAL BASIC 6.0 ที่มีลักษณะเป็น GUI เพื่อตอบสนองการกระทำของผู้ใช้ ดังแสดงในรูปที่ 3.10



รูปที่ 3.10 โปรแกรมส่วนแม่ข่าย

ส่วนประกอบต่างๆของหน้าตาโปรแกรม มีดังนี้

1. ช่องแสดงวัน เวลา ปัจจุบัน ของเครื่องแม่ข่าย
2. ช่องสำหรับใส่ file เพลง เพื่อ ใช้แสดงผล เมื่อการตั้งเวลา ปิด-เปิด ได้ตรงกับวันเวลา ของเครื่องแม่ข่าย
3. ปุ่มสำหรับสั่ง ปิด-เปิด ดวงไฟทั้งชั้น
4. ส่วนของการตั้งเวลา ปิด-เปิด ดวงไฟ แต่ละชั้น
5. ปุ่มสำหรับสั่ง ปิด-เปิด ดวงไฟ เพื่อ ปิด-เปิด ดวงไฟในแต่ละห้อง

2. การออกแบบการติดต่อสื่อสาร

การติดต่อจะติดต่อสื่อสารผ่านทาง-Text-file-โดยมีการออกแบบดังนี้

การออกแบบ Input

โดยการเขียนข้อมูลลง Text files เพื่อใช้ในการติดต่อ Web-Based และ ด้วยบอร์ดทดลอง MCS-51 โดย text file จะมีข้อมูลเป็น Char โดยต้องทำการแปลงข้อมูลที่ได้นำไปใช้งาน เช่น เปลี่ยนจาก char เป็น integer ก่อนเพื่อใช้ในการคำนวณ

การออกแบบ Output

จะมี 3 ฝั่งคือ ฝั่ง Server, เว็บบและฝั่งลูกข่าย

จะแสดง Output 3 ที่คือ แสดงการเปลี่ยนแปลงที่เกิดขึ้นผ่าน โปรแกรมควบคุมการทำงานฝั่ง Server, web-based และแสดงผลบนบอร์ด LED โดยการส่ง text file จะเป็นในลักษณะของข้อมูลเป็น integer โดยต้องทำการแปลงข้อมูลที่ได้อีก่อนนำไปใช้งานเช่น เปลี่ยนจากinteger เป็นchar ก่อนส่งให้บอร์ดรับค่า char แล้วบอร์ดจะมองเห็นเป็นรหัส ASCII เพื่อใช้ไปทำงานต่อ

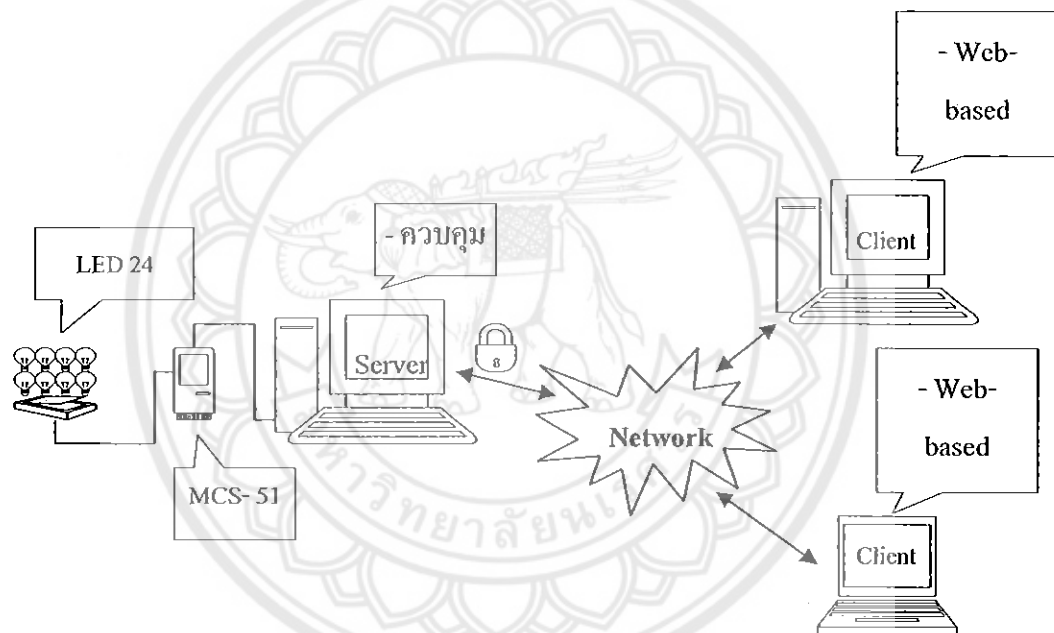
ตำแหน่งที่เก็บ Text file

จะถูกเก็บไว้ใน C:\inetpub\wwwroot\ ของเครื่องserver แบ่งออกเป็น 3 file คือ a.txt, b.txt และ c.txt โดยแต่ละ file จะควบคุมแต่ละชั้นดังนี้

a.txt จะใช้ในการควบคุมการปิด-เปิด ดวงไฟชั้นที่ 1

b.txt จะใช้ในการควบคุมการปิด-เปิด ดวงไฟชั้นที่ 2

c.txt จะใช้ในการควบคุมการปิด-เปิด ดวงไฟชั้นที่ 3

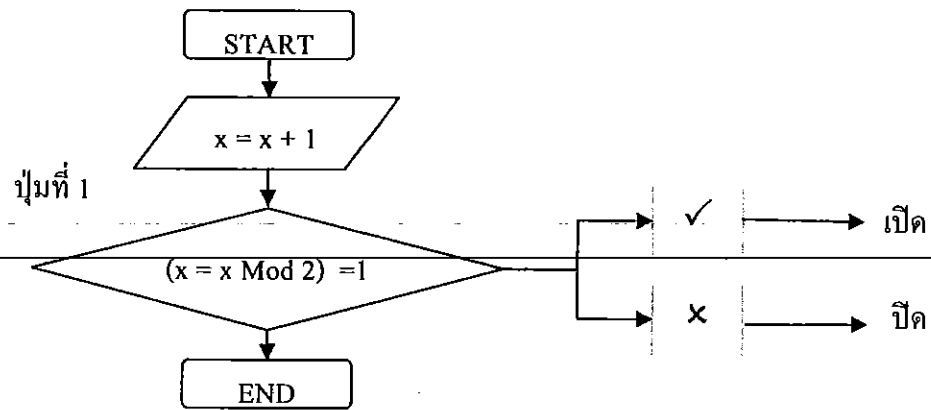


รูปที่ 3.11 กระบวนการติดต่อสื่อสาร

3. การออกแบบการทำงานของโปรแกรมควบคุม

1. การกดปุ่มดวงไฟ (การเขียน Text File)

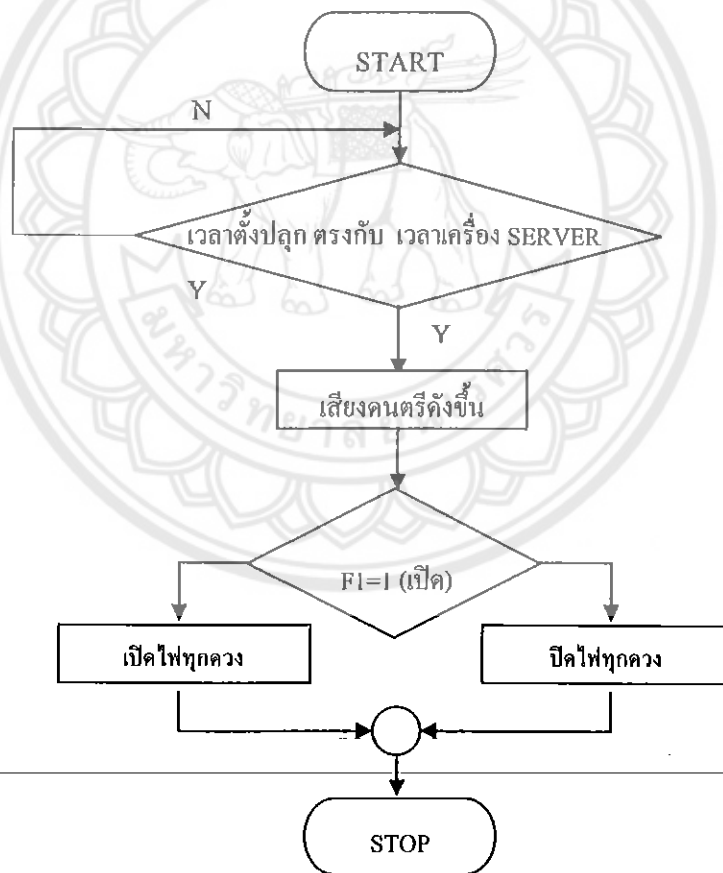
เนื่องจากการทำงานจะเหมือนกันทั้ง 3 ชั้น ดังนั้นผู้จัดทำจะขอยกตัวอย่างการทำงานของชั้นที่ 1 เพื่อใช้ในการออกแบบการทำงาน “การกดปุ่ม ปิด-เปิด ดวงไฟ”



รูปที่ 3.12 การกดปั๊ม ปิด-เปิด ดวงไฟแต่ละดวง

2.การตั้งเวลาแต่ละชั้น

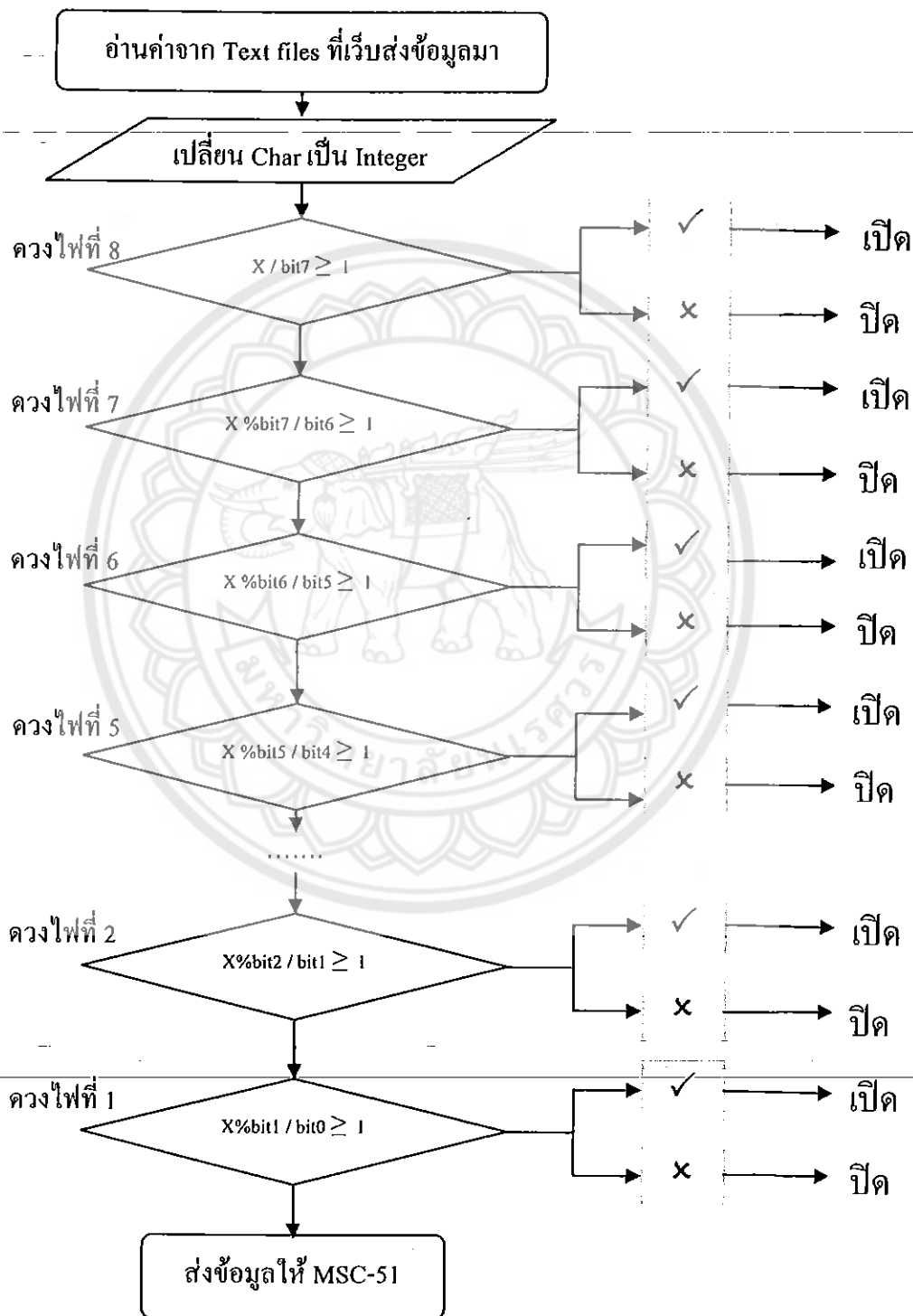
เป็นการควบคุมที่ Server เท่านั้น สามารถตั้งเวลา ปิด-เปิด ดวงไฟแต่ละชั้นหลักการทำงาน “การตั้งเวลา ปิด-เปิด ดวงไฟ แต่ละชั้น”



รูปที่ 3.13 การตั้งเวลา ปิด-เปิด ดวงไฟ แต่ละชั้น

3. การอ่าน Text File

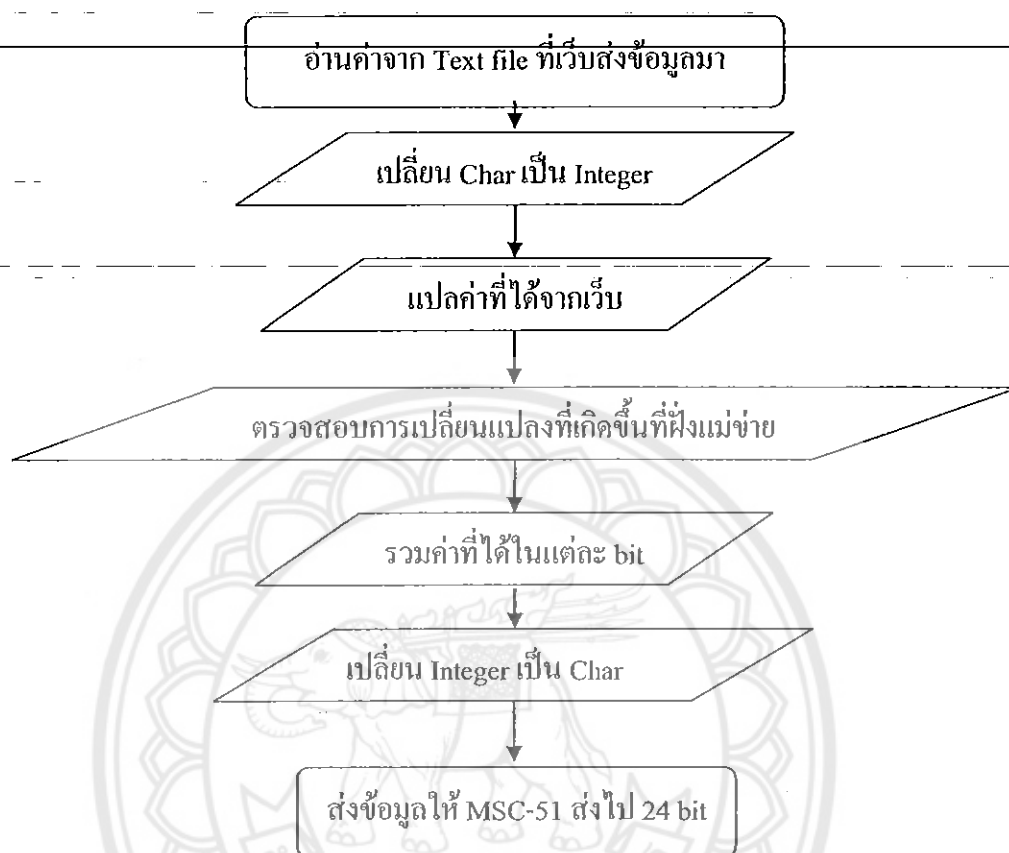
การอ่าน Text files จะอ่าน File a.txt ที่ Path "C:\inetpub\wwwroot" ซึ่งเป็น files เดียวกับที่เขียน โดยการอ่านข้อมูลใน text files โดยตัวโปรแกรมควบคุมได้ทำการตั้งเวลา ให้ทำการอ่านทุกๆ 1 วินาที "การอ่านข้อมูล text files"



รูปที่ 3.14 การอ่านข้อมูล text files ที่ได้จากเว็บ

4. การออกแบบการติดต่อสื่อสารระหว่าง Web-Based กับ โปรแกรมควบคุม

“กระบวนการทำงานส่งค่าที่อยู่ใน text file ให้บอร์ด MCS-51 โดยผ่านตัวควบคุมที่ฝั่งแม่ข่าย”



รูปที่ 3.15 การทำงานส่งค่า text file ให้บอร์ด MCS-51 โดยผ่านตัวควบคุมที่ฝั่งเว็บ

บทที่ 4

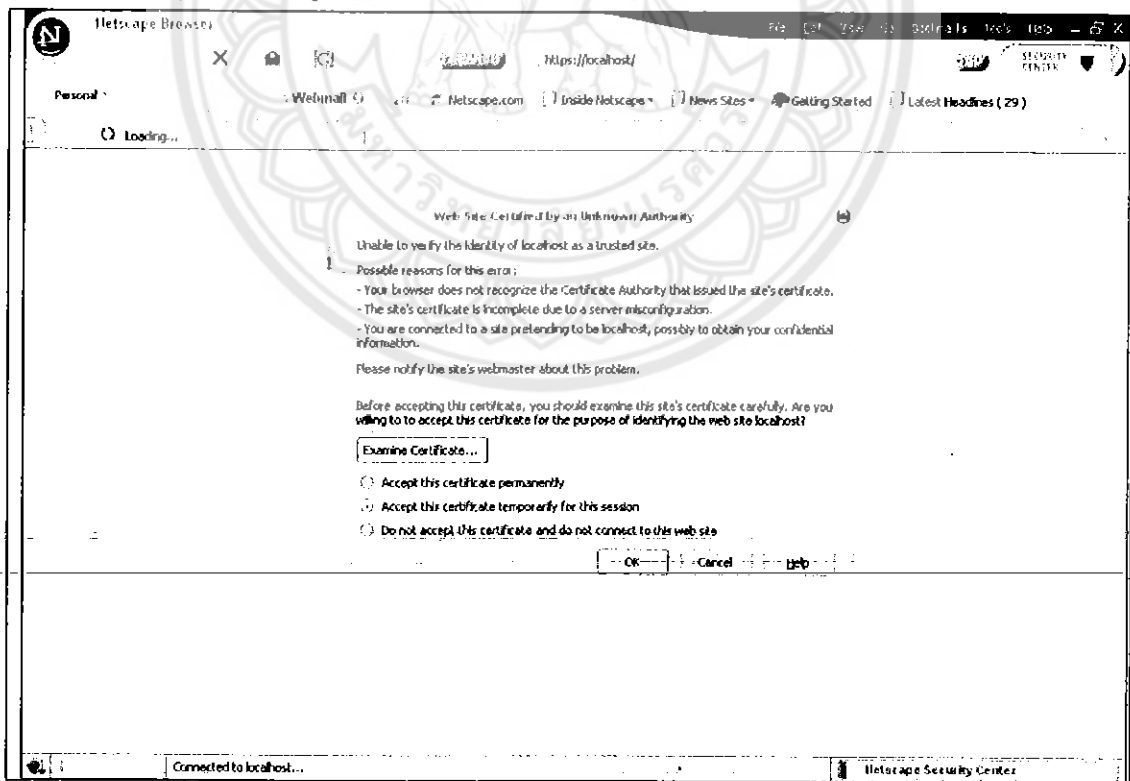
ผลการทดลอง

การทดลองของโครงการนี้ได้แบ่งผลการทดลองออกเป็น 5 ส่วน คือ

1. Security (Secure Sockets Layer)
2. ความปลอดภัยจากการ login เข้าหน้าเว็บ
3. การทำงานของโปรแกรมในการปิด-เปิดไฟในแต่ละห้องและแต่ละชั้น
4. การ Interface ของฮาร์ดแวร์
5. การติดต่อของโปรแกรมจากเว็บไปยังฮาร์ดแวร์และจากฮาร์ดแวร์ไปยังเว็บ

4.1 ผลการทดลองความปลอดภัยในส่วนของ Security (Secure Sockets Layer)

SSL เป็นส่วนแรกเมื่อทำการเปิดหน้าเว็บ Web-based electric control system for household โดยที่เมื่อทำการพิมพ์ `https://localhost` (สำหรับเครื่องแม่ข่าย) และ พิมพ์ `https://` แล้วตามด้วย address (สำหรับเครื่องลูกข่าย) ของเครื่องแม่ข่ายก็จะเป็นการติดต่อเข้าไปยัง Port ของ SSL จะปรากฏบนหน้าดังรูปที่ 4.1

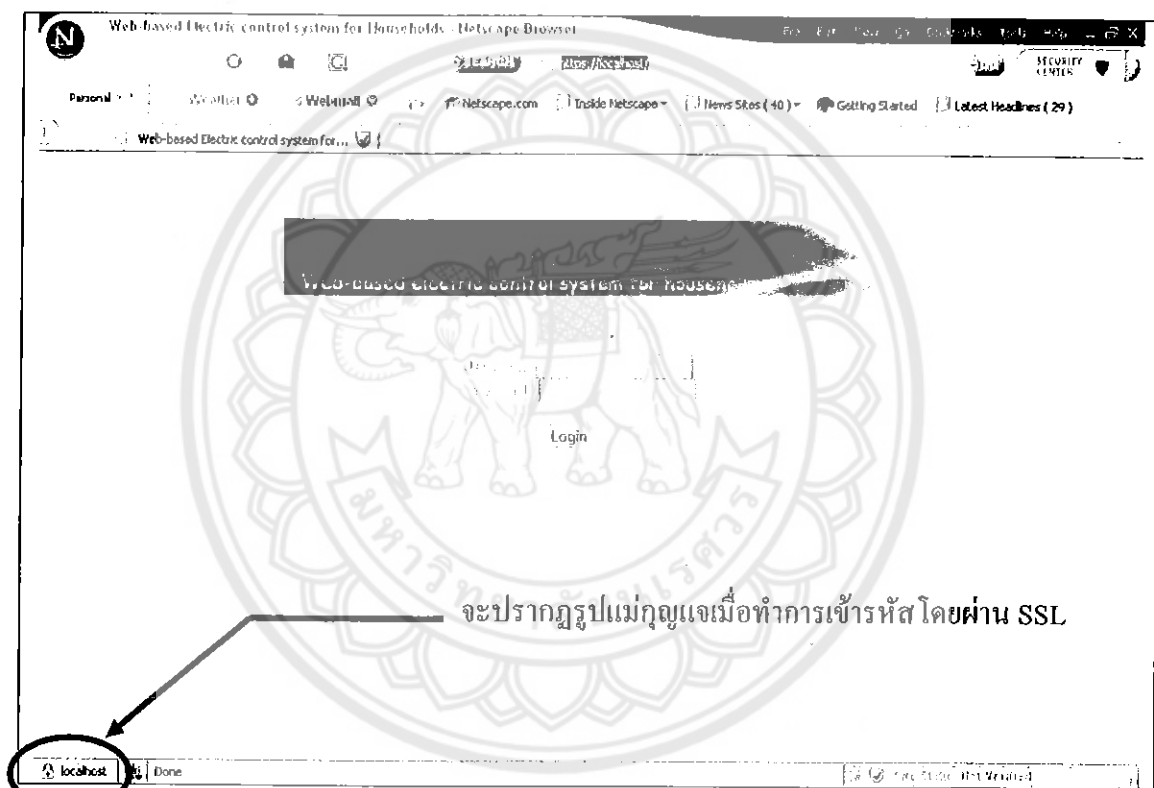


รูปที่ 4.1 ภาพแสดงการทำงานของ SSL ก่อนเข้าหน้าแรกของเว็บ

จากภาพเป็นการแสดงการทำงานของ SSL เพื่อให้ผู้ใช้ได้เลือกเพื่อทำการเข้ารหัสข้อมูลระหว่างทาง โดยผู้ใช้สามารถเลือกได้ 3 กรณีคือ

1. Accept this certificate permanently (เข้ารหัสถาวรจะเป็นการส่ง certificate มาติดตั้งยังเครื่องลูกข่าย)
2. Accept this certificate temporarily for this session (เข้ารหัสสำหรับครั้งนี้)
3. Do not accept this certificate and do not connect to this web site (ไม่เข้ารหัสและไม่เชื่อมต่อกับ web site นี้)

เมื่อทำการเลือกในกรณีที่ 1 หรือ 2 จะปรากฏหน้า Login ของโปรแกรมดังรูปที่ 4.2



รูปที่ 4.2 ภาพแสดงการทำงานของ SSL เมื่อเลือกการเข้ารหัส

4.2 ผลการทดลองความปลอดภัยจากการ login เข้าหน้าเว็บ

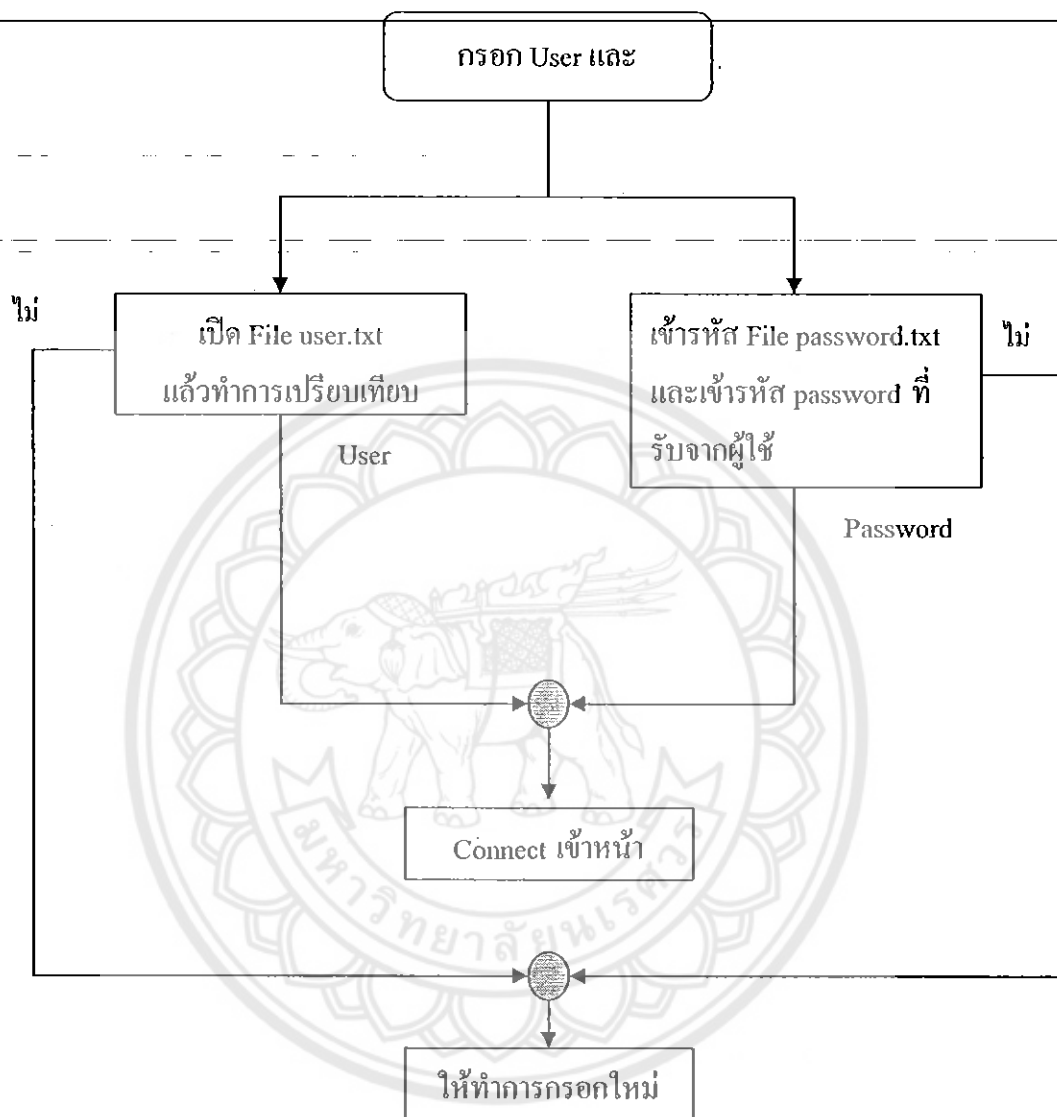
เมื่อผู้ใช้ได้ทำการผ่านกระบวนการ SSL ผู้ใช้จะเห็นหน้าแรกของเว็บเพจก็คือหน้า login

ดังรูปที่ 4.3



รูปที่ 4.3 หน้าแรกของเว็บเพจคือหน้า login

ในการทำงานของส่วน Login นี้จะเป็นการเข้ารหัส password และแยก user กับ password ออกจากตัวโปรแกรม เพื่อเป็นการป้องกันการ hack ซึ่งในการทำงานของการ login จะเป็นดัง Flowchart



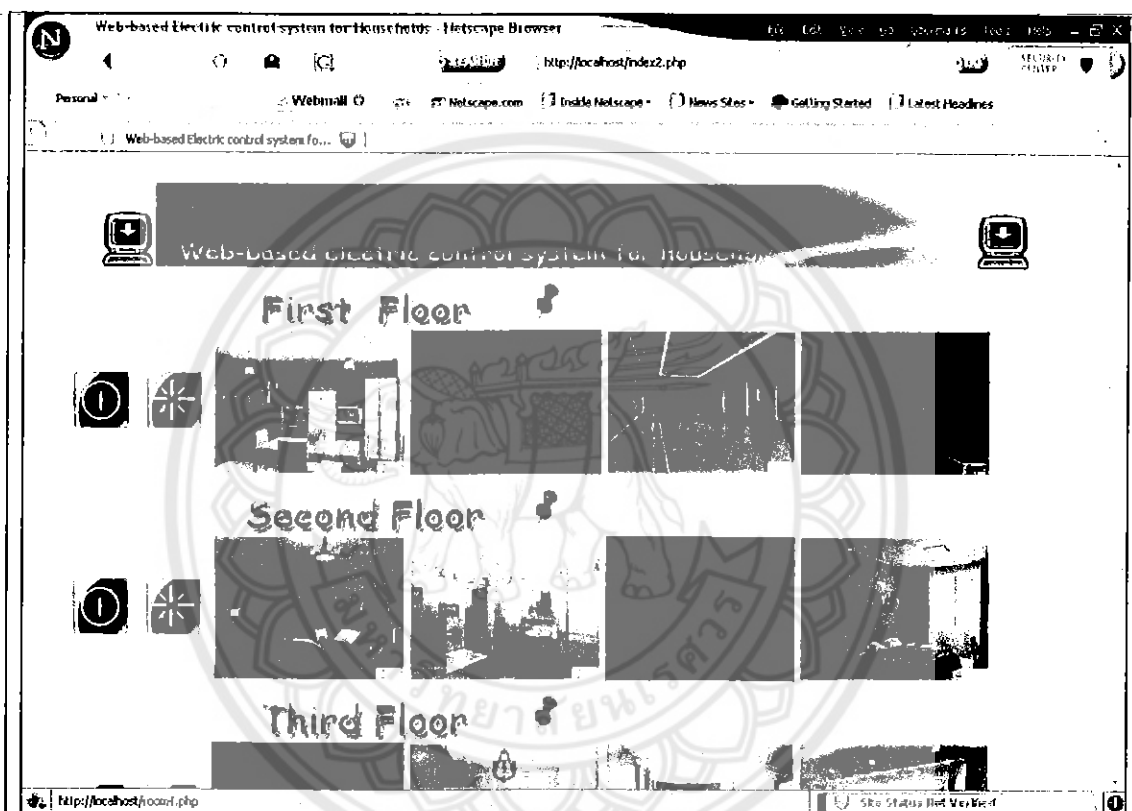
รูปที่ 4.4 แสดงการทำงานของโปรแกรมหน้า login

4.3 ผลการทำงานของโปรแกรมในการ ปิด-เปิด ไฟในแต่ละห้องและแต่ละชั้น

ในส่วนของเว็บแอปพลิเคชันผู้ใช้สามารถมองเห็นห้องต่างๆและสามารถทราบได้ว่าห้องไหนปิดไฟหรือเปิดไฟอยู่โดยแสดงจากภาพของห้องต่างๆ โดยแสดงดังนี้ -

1. ห้องเปิดไฟอยู่จะแสดงภาพของห้องที่สว่าง
2. ห้องถูกปิดไฟจะแสดงภาพของห้องที่มืด

โดยสถานะของห้องสามารถแสดงดังรูปที่ 4.5 ดังนี้ -



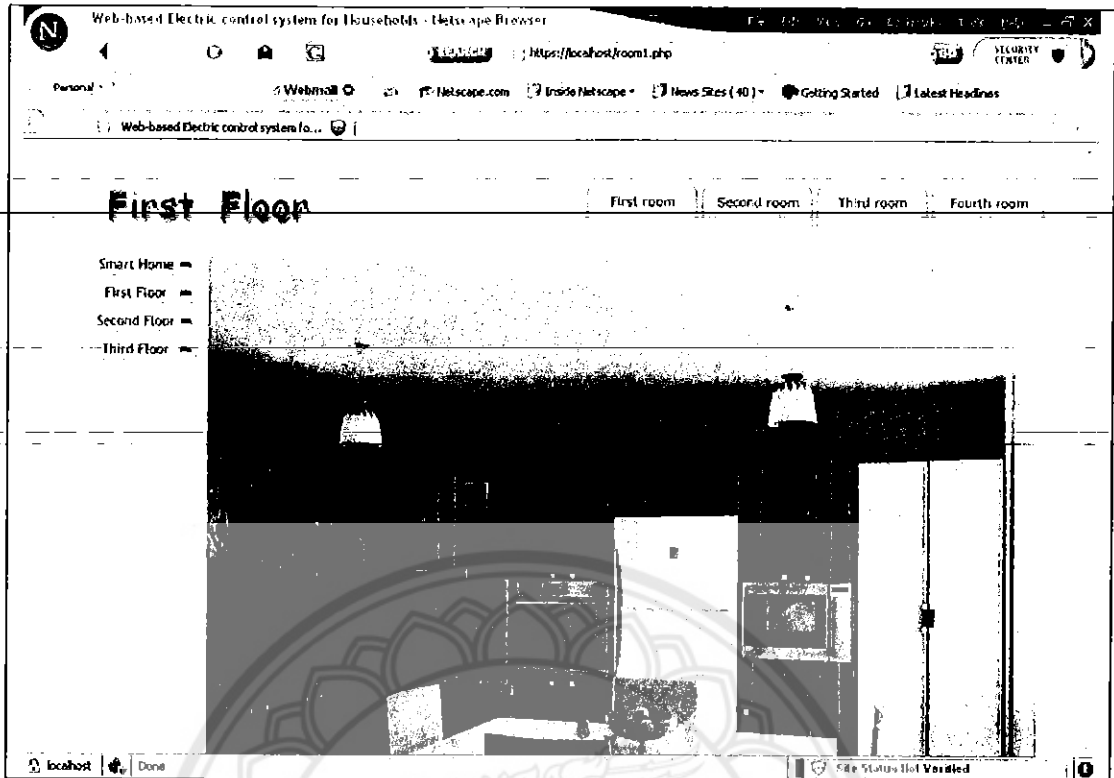
รูปที่ 4.5 แสดงสถานะของแต่ละห้องในบ้าน

ในการที่จะ ปิด - เปิด ไฟในแต่ละส่วนผู้จัดทำได้แบ่งออกเป็น 2 ส่วน ดังนี้

1. ปิด - เปิด ได้ทีละห้อง
2. ปิด - เปิด ได้ทั้งชั้น

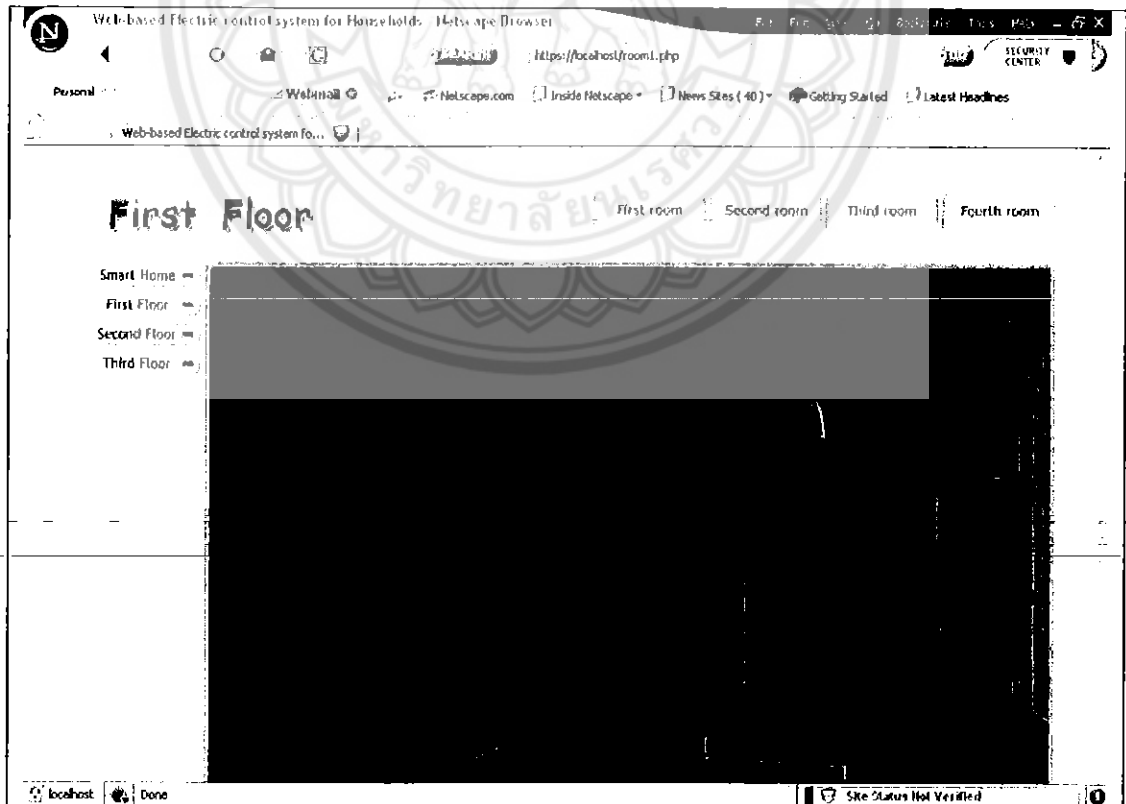
กรณีที่ 1 ปิด - เปิด ในแต่ละห้อง

ผู้ใช้สามารถทำการ ปิด - เปิด ไฟของแต่ละห้องได้โดยการ Click ไปที่ห้องที่ต้องการเข้าไป control ได้เลย ดังรูปที่ 4.6



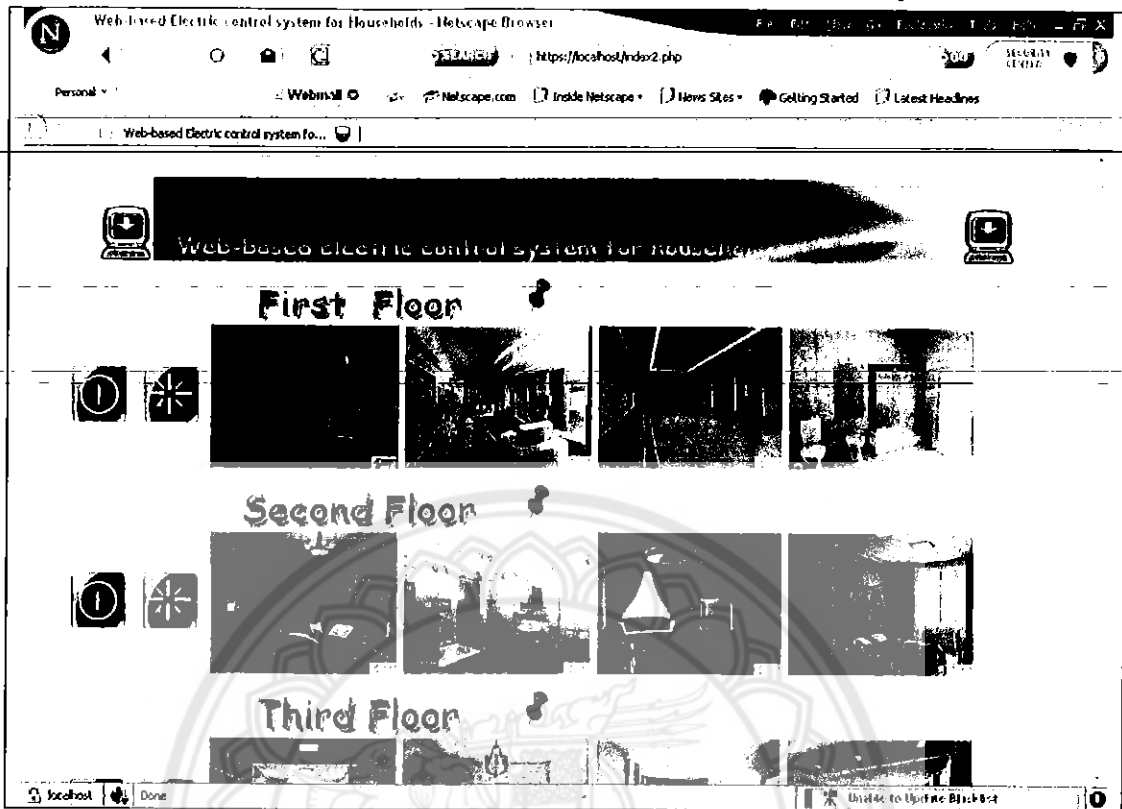
รูปที่ 4.6 แสดงการ ปิด - เปิด ไฟของแต่ละห้อง

เมื่อทำการกดปุ่มเพื่อทำการปิดไฟจะปรากฏสถานะของห้องเป็นภาพมืดดังรูปที่ 4.7



รูปที่ 4.7 แสดงสถานะของห้องเมื่อทำการปิดไฟในห้อง

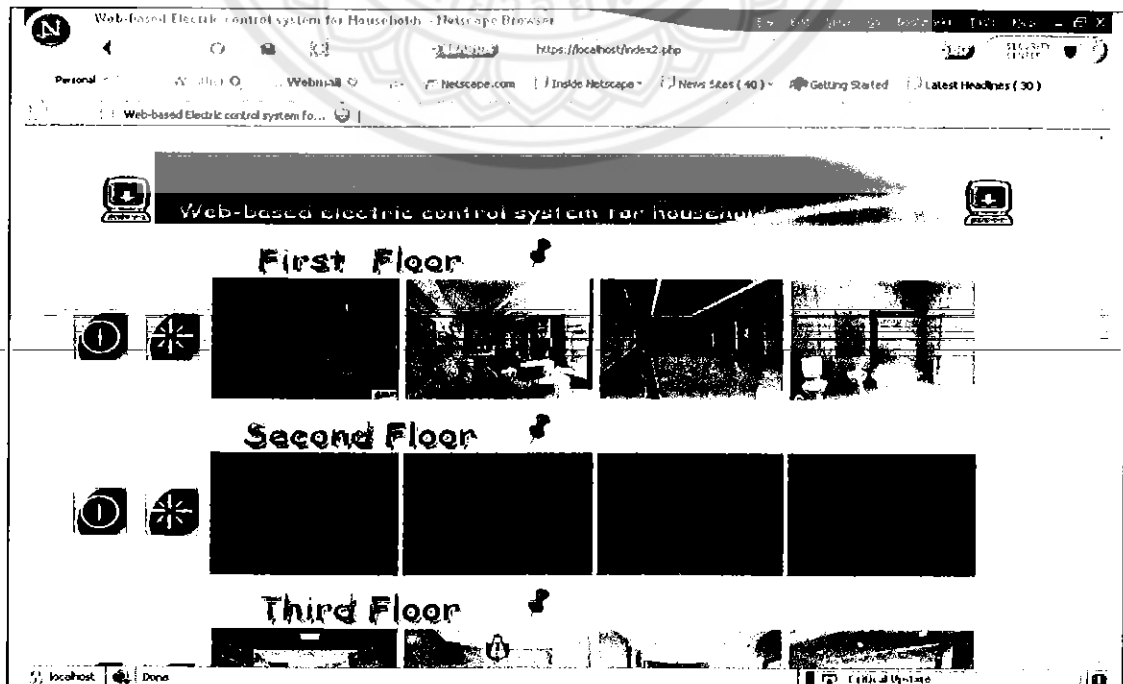
เมื่อทำการกดที่ปุ่ม Smart Home ก็จะแสดงสถานะที่เปลี่ยนไปของห้องต่างๆดังรูปที่ 4.8



รูปที่ 4.8 แสดงสถานะเมื่อห้องถูกปิดไฟ

กรณีที่ 2 ปิด – เปิด ไฟทั้งชั้น

ผู้ใช้สามารถทำการ ปิด – เปิด ไฟได้ทั้งชั้นโดยการ Click ที่ปุ่มที่หน้า Smart Home ได้เลย โดยเมื่อทำการกดเปิดหรือปิดก็จะมีการเปลี่ยนแปลงสถานะดังรูปที่ 4.9



รูปที่ 4.9 เป็นการแสดงสถานะของการเปลี่ยนแปลงเมื่อทำการ ปิด-เปิด ไฟทั้งชั้น

4.4 ผลการทำงานของโปรแกรมที่ Interface กับฮาร์ดแวร์

การ Interface ควบคุมฮาร์ดแวร์นั้นจะเริ่มจากการ อ่าน-เขียน Text file โดยโปรแกรมที่ฝั่ง server อ่าน text file ที่ได้จากเว็บแอปพลิเคชันจากนั้นจะส่งเปลี่ยน Text file เป็น char แล้วส่งให้บอร์ดทดลอง โดย Output จะแสดงบนหน้าเว็บแอปพลิเคชันและอุปกรณ์แสดงผล (ดวงไฟ LED บนบอร์ด)

แต่ก่อนที่จะทำการส่งต้องทำการคิดค้ระหว่างเครื่องคอมพิวเตอร์กับพอร์ตอนุกรมถึงจะส่งข้อมูลให้ MSC-51 โดยการทำการเชื่อมผ่านทางพอร์ตอนุกรม โดยพิมพ์ mode com1:19200, n, 8, 1 ในช่วง RUN เพื่อเชื่อมต่อทำการดำเนินงานดังรูป 4.10

```
C:\>mode com1:19200,n,8,1
Status for device COM1:
-----
Baud:                19200
Parity:              None
Data Bits:           8
Stop Bits:           1
Timeout:             ON
XON/XOFF:            OFF
CIS handshaking:    OFF
DSR handshaking:    OFF
DSR sensitivity:    OFF
DIR circuit:         ON
RTS circuit:         ON
C:\>
```

รูปที่ 4.10 แสดงการเชื่อมต่อพอร์ตอนุกรม

จากนั้นจะใช้โปรแกรมของฝั่งServerในการควบคุมSwitch ห้องต่างๆภายในบ้าน โดยจะติดต่อกับเว็บแอปพลิเคชันโดยแบ่งการทำงานออกเป็น 2 ส่วน ได้แก่

1. การรับ Input จากเครื่องลูกข่าย (website)
2. การรับ Input จากเครื่องแม่ข่าย

เครื่องลูกข่ายสามารถควบคุมการ ปิด-เปิดดวงไฟได้ 2 แบบคือ

- 1.1 ควบคุมการปิด-เปิดดวงไฟแต่ห้องภายในบ้าน (ดวงไฟ LED บนบอร์ด)
- 1.2 ควบคุมการปิด-เปิดดวงไฟแต่ชั้นภายในบ้าน (ดวงไฟ LED บนบอร์ด)

เครื่องแม่ข่ายสามารถควบคุมการปิด-เปิดดวงไฟได้ 3 แบบคือ

- 1.3 ควบคุมการปิด-เปิดดวงไฟแต่ละดวงภายในบ้าน (ดวงไฟ LED บนบอร์ด)
- 1.4 ควบคุมการปิด-เปิดดวงไฟแต่ชั้นภายในบ้าน (ดวงไฟ LED บนบอร์ด)
- 1.5 ควบคุมการตั้งเวลาปิด-เปิดดวงไฟภายในบ้านแต่ละชั้น (ดวงไฟ LED บนบอร์ด)

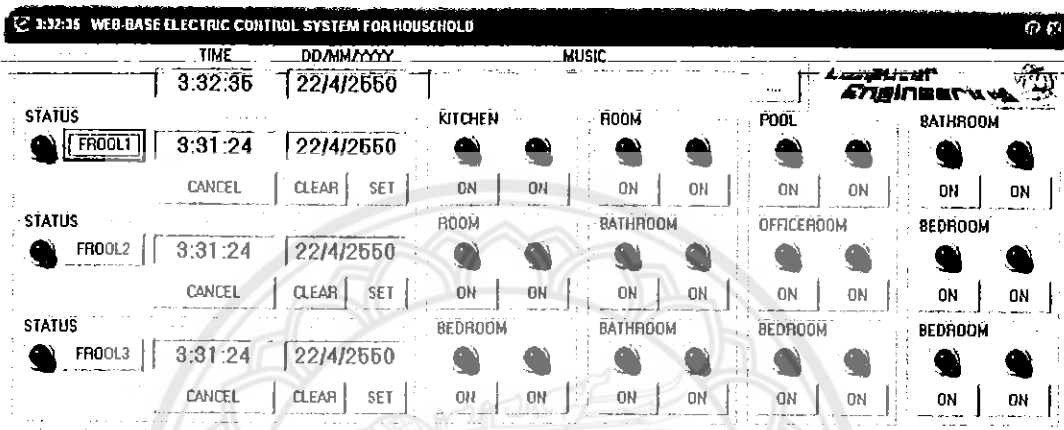
4.4.1 การใช้งานโปรแกรมโดยควบคุมที่แม่ข่าย

เริ่มจากการทำการติดต่อสื่อสารได้โดยการรันโปรแกรมควบคุมฝั่งแม่ข่าย จะปรากฏดังรูป 4.11

ในส่วนนี้ผู้ใช้สามารถมองเห็นปุ่ม Switch ON ของแต่ละห้องและสามารถทราบได้

ว่าห้องไหนปิดไฟหรือเปิดไฟอยู่โดยแสดงจากภาพแสดงได้ดังนี้

- 1 ห้องที่เปิดไฟอยู่จะแสดงดวงไฟสีแดงและที่ switch แสดงข้อความ "OFF"
- 2 ห้องที่ปิดไฟอยู่จะแสดงดวงไฟสีดำและที่ switch แสดงข้อความ "ON"



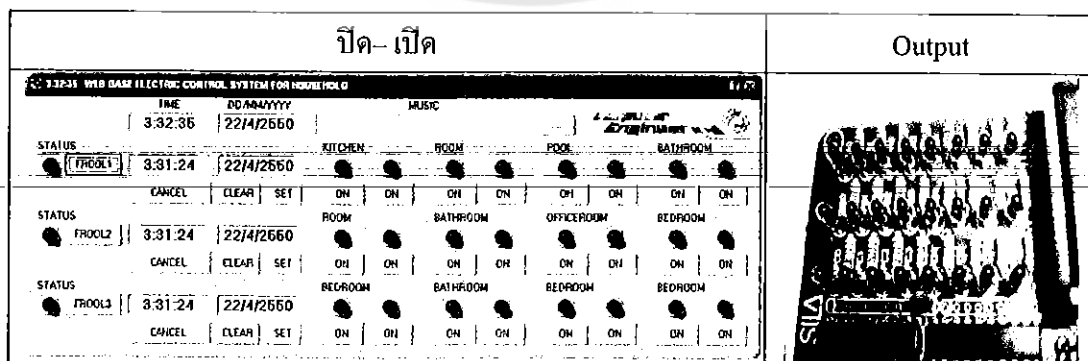
รูปที่ 4.11 โปรแกรมควบคุมฝั่งแม่ข่าย

การที่จะ ปิด-เปิดไฟในแต่ละส่วน ผู้จัดทำได้แบ่งออกเป็น 3 ส่วน ต่อไปนี้

- 1 ปิด-เปิดแต่ละดวง
- 2 ปิด-เปิดได้ทั้งชั้น
- 3 ตั้งเวลา ปิด-เปิดทั้งชั้น

กรณีที่ 1 ปิด-เปิดแต่ละดวง

โดยการ Click ที่ปุ่ม OFF-ON จะได้ดังรูปที่ 4.12

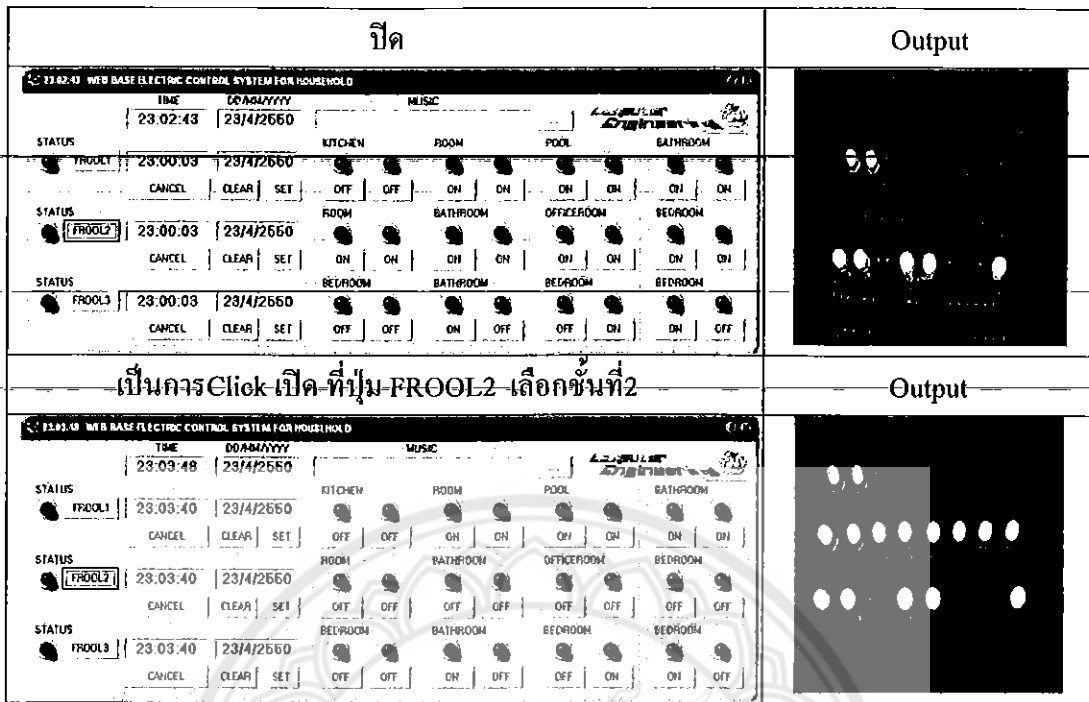


รูปที่ 4.12 การปิด-เปิดดวงไฟแต่ละดวง

กรณีที่ 2 ปิด-เปิดได้ทั้งชั้น

โดย Click เลือกชั้นที่ต้องการควบคุมดังรูปที่ 4.13



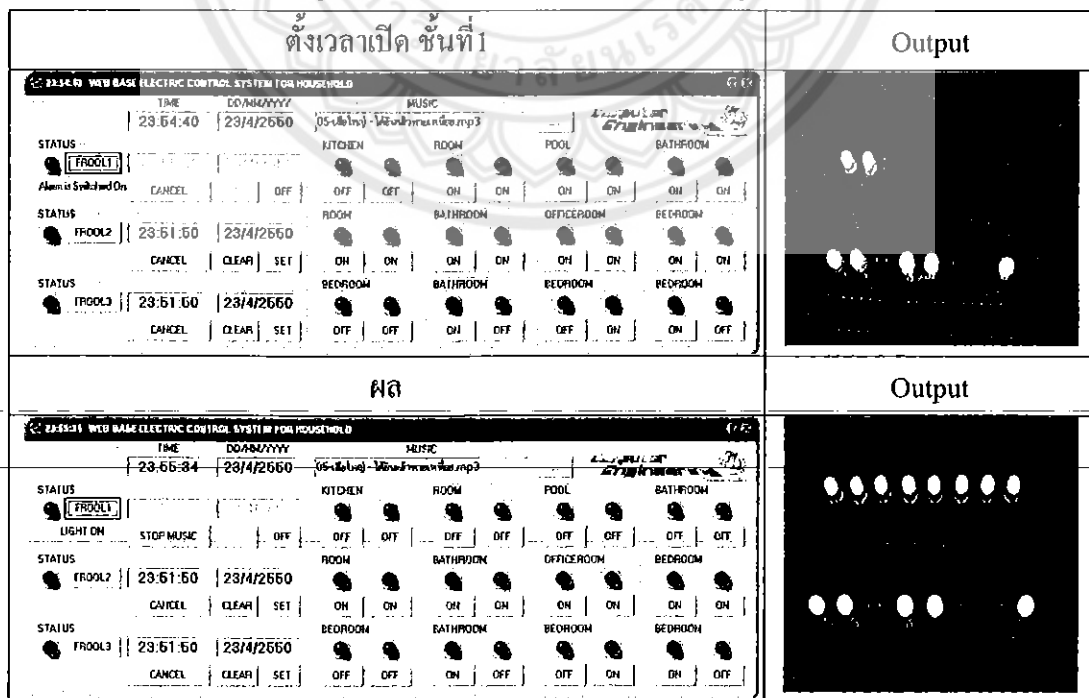


รูปที่ 4.13 แสดงสถานะของชั้น1ที่มีการเปลี่ยนแปลง

กรณีที่ 3 ตั้งเวลา ปิด-เปิดไฟทั้งชั้น

โดยการ Click ที่ปุ่มที่ SET | โดยทำการตั้งเวลาและสามารถปลุกเป็นเสียงเพลงตามที่เรา

ต้องการได้ การตั้งเวลาจะอยู่ในส่วนนี้ CANCEL | CLEAR | SET | ดังรูปที่ 4.14

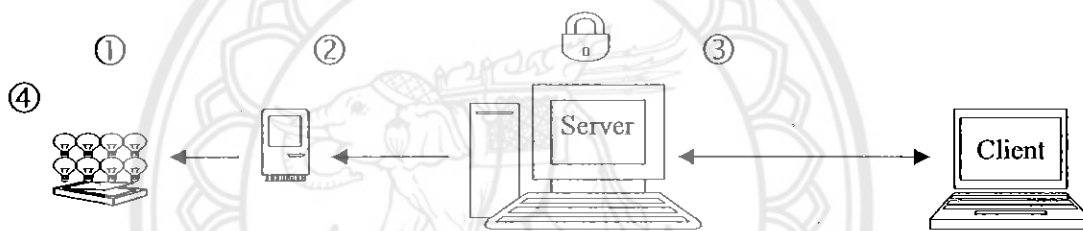


รูปที่ 4.14 เป็นการแสดงสถานะของการตั้งเวลาทำการเปิดไฟทั้งชั้นของชั้น 1

สามารถทำการปิด-เปิด ไฟ ทั้ง 3 ชั้น เราสามารถหยุดเสียงเพลง และตั้งเวลาทำงานใหม่
หลังจากกดปุ่ม **STOP MUSIC**

4.5 ผลการทำงานการเชื่อมต่อกันระหว่างเว็บกับฮาร์ดแวร์

ในการเชื่อมต่อกันระหว่างเว็บกับฮาร์ดแวร์เป็นการเชื่อมต่อกันระหว่าง Text file ซึ่งโปรแกรมที่ทำงานอยู่ที่เว็บจะทำหน้าที่ในการเปิด Text files เพื่ออ่านค่าที่ได้แล้วนำไปคำนวณและนำค่าที่ได้ไปเปลี่ยนแปลงสถานะของการแสดงผลที่หน้าเว็บไซต์ จากนั้นเมื่อมีการกดปุ่มจากหน้าเว็บเพื่อทำการปิดหรือเปิดไฟในแต่ละห้องหรือทั้งชั้น โปรแกรมก็จะทำการคำนวณและเขียนค่าที่ได้ลงไป ใน Text file เดิมเพื่อให้โปรแกรมที่ทำการควบคุมฮาร์ดแวร์อ่านค่าและแสดงผลทาง LED ซึ่งการทำงานของโปรแกรมทั้งสองจะแสดงดังรูปที่ 4.15 ดังนี้



รูปที่ 4.15 ภาพการทำงานทั้งระบบของการควบคุมดวงไฟ

- ① LED ทำงานตามคำสั่งจากเครื่องแม่ข่าย
- ② ฮาร์ดแวร์(MCS-51) จะทำการรับคำสั่งจากโปรแกรม VB เพื่อแสดงผลคิด-ค้น
- ③ เครื่องแม่ข่ายที่ทำการอ่านค่าจาก Text files และนำค่าไปคำนวณเพื่อนำค่าที่ได้ไปแสดงผลที่หน้าเว็บและรับค่าจากการกดปุ่มไปคำนวณและเขียนค่าที่ได้ลง Text files
- ④ เครื่องลูกข่าย ทำการร้องขอไปยังเครื่องแม่ข่าย เพื่อสั่งการ ปิด-เปิด ดวงไฟ

บทที่ 5

สรุปผล

โครงการนี้แสดงให้เห็นถึงประสิทธิภาพในการควบคุมการทำงานของอุปกรณ์ฮาร์ดแวร์ โดยผ่านทางเว็บ โดยเป็นการทำงานออกเป็น 2 ส่วน คือส่วนที่ติดต่อกับอุปกรณ์ทางฮาร์ดแวร์ซึ่งได้ทำการจำลองการทำงานของเครื่องใช้ไฟฟ้าจาก อุปกรณ์ Hardware ที่เรียกว่า MCS-51 ผ่านทาง Serial port RS-232 แล้วแสดง Output บนบอร์ด LED และส่วนของโปรแกรมที่ทำงานติดต่อกับหน้าเว็บเพื่อแสดงผลออกมาตามที่ LED แสดงผล

ภาษาคอมพิวเตอร์ที่นำมาใช้พัฒนาการเขียนโปรแกรมในโครงการนี้ได้เลือกใช้ภาษา Visual Basic 6 มาพัฒนาโปรแกรมในส่วนที่ใช้เพื่อการติดต่อกับอุปกรณ์ฮาร์ดแวร์ ภาษาคอมพิวเตอร์ที่นำมาพัฒนาเว็บแอปพลิเคชันได้เลือกใช้ภาษา PHP มาพัฒนาโปรแกรม เนื่องจากเป็นภาษาที่สามารถใช้งานได้ง่ายและตรงต่อวัตถุประสงค์ที่ต้องการ อีกทั้งยังสามารถแทรก code ของ PHP ไปใน HTML ได้อีกด้วย

5.1 สรุปผลการทดลอง

จากการจัดทำโครงการการควบคุมไฟบ้านผ่านเว็บ ผลที่ได้รับของโครงการนี้ คือสามารถเขียนโปรแกรมใช้ในการสั่งงานเพื่อ ปิด-เปิด ดวงไฟบ้านผ่านที่เว็บได้ โปรแกรมที่พัฒนาขึ้นจะสามารถนำมาใช้งานในการสั่งงานเพื่อ ปิด-เปิด ไฟบ้านมีการทำงาน 2 ส่วน คือ โปรแกรมส่วนที่แสดงผลทางเว็บแอปพลิเคชัน โดยทำการติดต่อกับ โปรแกรมที่ทำหน้าที่ควบคุม Hardware โดยอ่าน-เขียน ลง Text File และในขณะเดียวกัน โปรแกรมที่ทำหน้าที่ควบคุม Hardware จะทำการอ่าน-เขียน ลง Text File เดียวกันซึ่งเป็นการติดต่อกันระหว่างการส่งและรับค่าจากส่วนเว็บแอปพลิเคชัน ไปยัง Hardware โดยที่เครื่องที่ทำหน้าที่เก็บ โปรแกรมทั้งหมดคือเครื่องแม่ข่ายส่วนเครื่องที่เป็นลูกข่ายสามารถทำการเชื่อมต่อกับเครื่องแม่ข่ายเพื่อทำการ ปิด-เปิด ดวงไฟได้โดยผ่านทางเว็บได้ เมื่อทำการสั่งงานผ่านทางเครื่องลูกข่ายจะสามารถ ปิด-เปิด ดวงไฟได้แบบ Real time ทุกๆ 1 วินาที ส่วนการแสดงผลทางหน้าเว็บแอปพลิเคชันนั้นค่อนข้างมี delay time มากกว่า โดยประมาณ 5-10 วินาที

จากการทดลองที่ได้สามารถสรุปได้ว่า โปรแกรมที่พัฒนาขึ้นในส่วนของเว็บแอปพลิเคชันสามารถทำงานได้เป็นอย่างดี แม้ว่าส่วนหนึ่งส่วนใดของระบบเสียหายไประบบก็ยังสามารถทำงานได้ เนื่องจากโปรแกรมใช้ในการสั่งงานเพื่อ ปิด-เปิด ดวงไฟและ โปรแกรมที่ทำหน้าที่ควบคุม Hardware แยกกันอยู่ โดยมีส่วนที่เชื่อมต่อกันอยู่ตรงกลางคือ text files ที่สามารถแก้ไขได้จากทั้งสองส่วน ซึ่งในผลการทดลองสามารถบอกได้ 3 กรณีถึงการควบคุมดวงไฟดังนี้

1. สามารถควบคุมการ ปิด-เปิด ดวงไฟจากการสั่งงานที่หน้าเว็บ โดยในการปิด-เปิด ไฟที่หน้าเว็บจะส่งค่าไปยัง โปรแกรมที่ควบคุมอุปกรณ์ฮาร์ดแวร์ซึ่งจะแสดงผลและสั่งให้ฮาร์ดแวร์ปิด-เปิดตามที่สั่งจากเว็บ

2. สามารถควบคุมการ ปิด-เปิด ดวงไฟจาก โปรแกรม VB ที่ทำหน้าที่ควบคุมฮาร์ดแวร์ ได้โดยตรงจากเครื่องแม่ข่าย เมื่อมีการเปลี่ยนแปลงสถานะภาพของฮาร์ดแวร์ โดยโปรแกรมที่ควบคุมอุปกรณ์ฮาร์ดแวร์ ส่วนเว็บก็จะแสดงผลตามสถานะภาพที่เปลี่ยนไป

กล่าวโดยสรุป ระบบที่สร้างขึ้นมานี้สามารถควบคุมการปิด-เปิดผ่านเว็บ ได้ทั้ง 2 ทิศทาง จากเว็บสู่ส่วนควบคุมและจากส่วนควบคุมสู่เว็บ

5.2 ปัญหาและแนว

แสดงผลการเปลี่ยนแปลงของห้องต่างๆที่แสดงบนเว็บแอปพลิเคชันนั้นในการแสดงผลการทำงาน จะช้ากว่าการทำงานจริงของ Hardware อาจเป็นเพราะขนาดของภาพที่ส่งมา ใช้วิธีการส่งที่ไม่ดีพอ หรือการใช้งานเครือข่ายมีสภาพที่แออัด ทำให้ข้อมูลภาพที่ส่งมาแสดงผลช้า การแก้ไขโดยการบีบอัดรูปภาพให้เล็กลงหรือใช้วิธีการบีบอัดรูปภาพแบบอื่น ในกรณีที่เกิดจากเครือข่าย

แก้ไขได้โดยการเชื่อมต่อเครื่องแม่ข่ายและเครื่องลูกข่ายโดยใช้รูปแบบการเชื่อมต่อโดยผ่านเครือข่ายความเร็วสูงมาใช้งาน ผู้ออกแบบต้องคำนึงถึงสิ่งเหล่านี้ด้วยหากต้องการให้การแสดงผลภาพที่ละเอียดระบบจะช้า

5.3 แนวทางพัฒนาต่อ

โครงการนี้เป็นเพียงการจำลองการควบคุมการ ปิด-เปิด ไฟฟ้าภายในบ้าน โดยการนำอุปกรณ์ Hardware คือ Micro board MCS – 51 มาทำการทดลอง ซึ่งมีแนวทางพัฒนาต่อโดยใช้เครื่องใช้ไฟฟ้าภายในบ้านมาทำการเชื่อมต่อเพื่อทำการควบคุมผ่านทางอินเทอร์เน็ตได้จริง ซึ่งในการพัฒนาจะต้องทำการเชื่อมต่อกับอุปกรณ์อื่นอีกหลายชนิด เช่น sensor วัดแสง, sensor วัดอุณหภูมิ เป็นต้น

5.4 ข้อเสนอแนะ

เนื่องจากการแสดงผลใน โครงการนี้ใช้ หลอด LED เป็นตัวแสดงการทำงาน แทนหลอดไฟจริง ดังนั้นหากต้องการตรวจสอบว่าดวงไฟเสียหรือไม่ โครงการนี้ยังไม่สามารถตรวจสอบได้

หากต้องการตรวจสอบว่าดวงไฟ เสียหรือไม่ สามารถทำได้โดยใช้ ชิพ 8255 อีกตัวมาช่วยในการตรวจสอบดวงไฟได้

เอกสารอ้างอิง

[1] ~~ประภาภรณ์ ช่างไม้. DREAMWEAVER MX 2004. กรุงเทพมหานคร: ไอดี ซีๆ, 2548.~~

[2] บริษัท ศิลาเรีเสิร์ช จำกัด. **START-C51 VERSION 3.0 Microcontroller Single Board.**
กรุงเทพมหานคร

[3] ไม่ทราบชื่อผู้แต่ง. **Visual Basic Programming Interface Hardware.**
[http://www.thaiio.com/prog-cgi/programing.cgi?0001,2000.](http://www.thaiio.com/prog-cgi/programing.cgi?0001,2000)

[4] อติศักดิ์ ชิมะวงศ์. การคิดต่อทางพอร์ตอนุกรมของไมโครคอนโทรลเลอร์.
[http://202.8.85.164/~adisak51/page14.html,2002.](http://202.8.85.164/~adisak51/page14.html,2002)

[5] ไม่ทราบชื่อผู้แต่ง. การอินเตอร์เฟสกับคอมพิวเตอร์หรือไมโครโปรเซสเซอร์.
[http://www.thaiio.com/Hardware-cgi/hardware.cgi?0007,2000.](http://www.thaiio.com/Hardware-cgi/hardware.cgi?0007,2000)

[6] ไม่ทราบชื่อผู้แต่ง. ความรู้เบื้องต้นเกี่ยวกับเครือข่าย.
[http://teacher.stjohn.ac.th/bachompui/Lec_data1.htm,2003.](http://teacher.stjohn.ac.th/bachompui/Lec_data1.htm,2003)

[7] สิทธิชัย กลิ่นประยงค์. การพัฒนาโปรแกรมควบคุมหุ่นยนต์โดยใช้คอมพิวเตอร์.
[http://student.chandra.ac.th/~control_robot/index.htm,2004.](http://student.chandra.ac.th/~control_robot/index.htm,2004)

[8] อภิชาติ ภู่วลัย. รวมคอนโทรลฟรี สำหรับ VB6. นนทบุรี : ไอดีซีๆ, 2547.

[9] ไม่ทราบชื่อผู้แต่ง. ไมโครคอนโทรลเลอร์ MCS-51.
<http://www.geocities.com/sutadet/mcs.html>

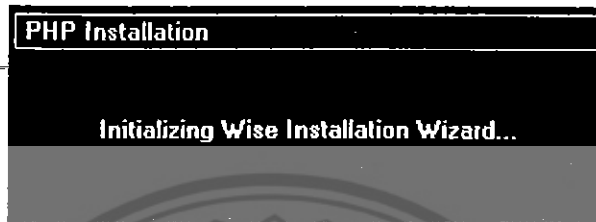
[10] ไม่ทราบชื่อผู้แต่ง. EX-LED v2.0. http://silaresearch.com/product/p_ex-led.pdf

ภาคผนวก ก

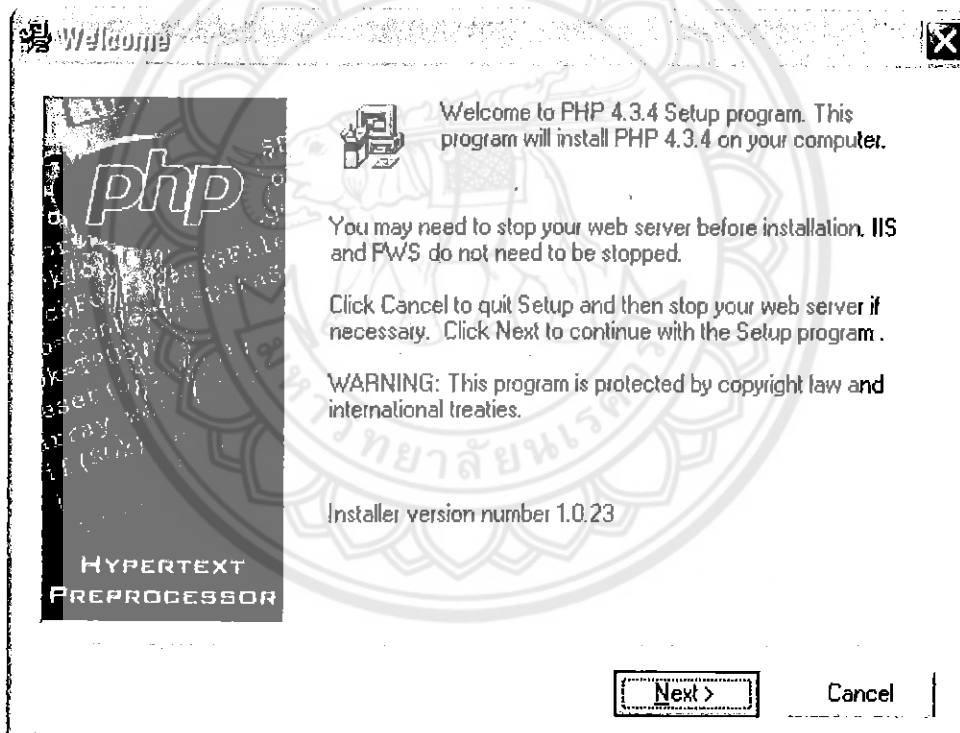
โปรแกรมที่ใช้งาน

การติดตั้ง PHP

หลังจากที่ Download ไฟล์ .exe แล้ว ทำการดับเบิลคลิกไฟล์เพื่อเข้าสู่โหมดของการติดตั้ง



กด Next จากนั้นกด I Agree



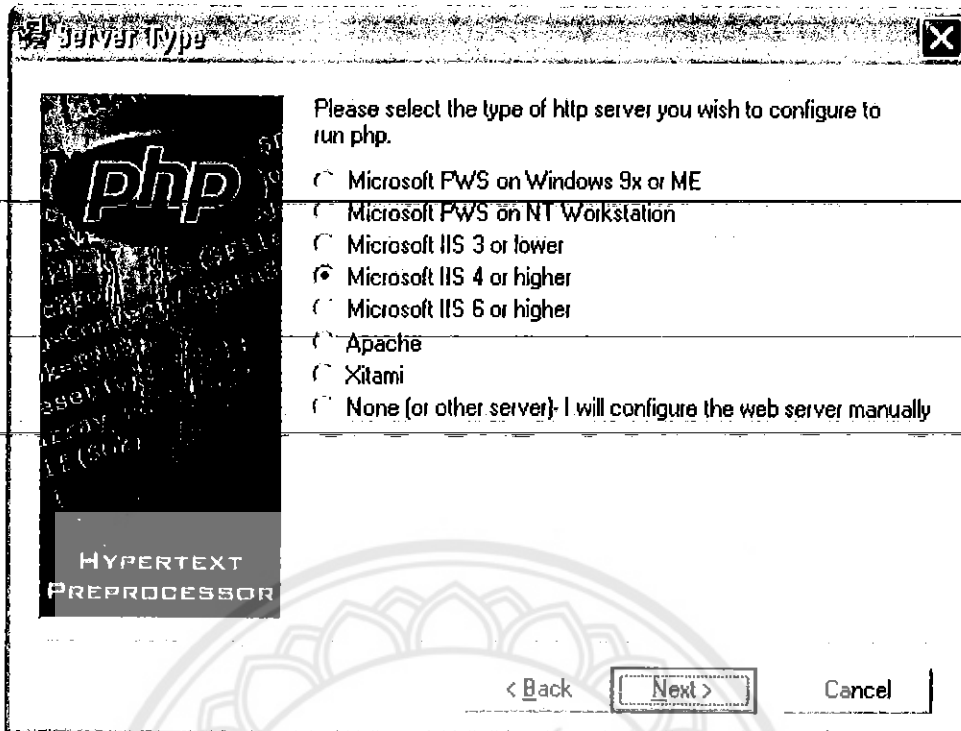
เลือกรูปแบบการติดตั้งตามต้องการกด Next



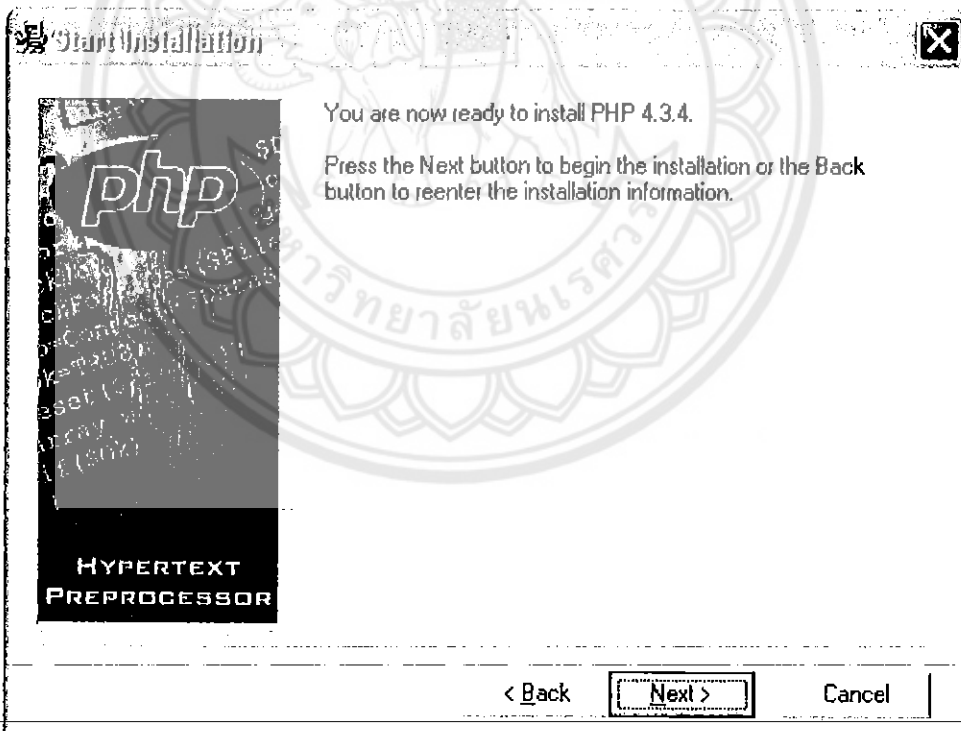
กำหนดชื่อ SMTP สำหรับใช้ในการทดสอบการส่งเมลล์ผ่านเครื่อง และ ใส่ชื่อ Email



ระบุโปรแกรม Web Server ที่ใช้อยู่ แต่ถ้าหากว่ายังไม่ได้ลงหรือว่าไม่มีชื่อโปรแกรม Web Server ที่ใช้ก็เลือกเป็น None (or the server)



กดปุ่ม Next โปรแกรมจะทำการติดตั้ง



เมื่อโปรแกรมทำการปรับแต่งค่าในโปรแกรม Web Server ที่เราเลือกให้แล้วจะกดปุ่ม OK

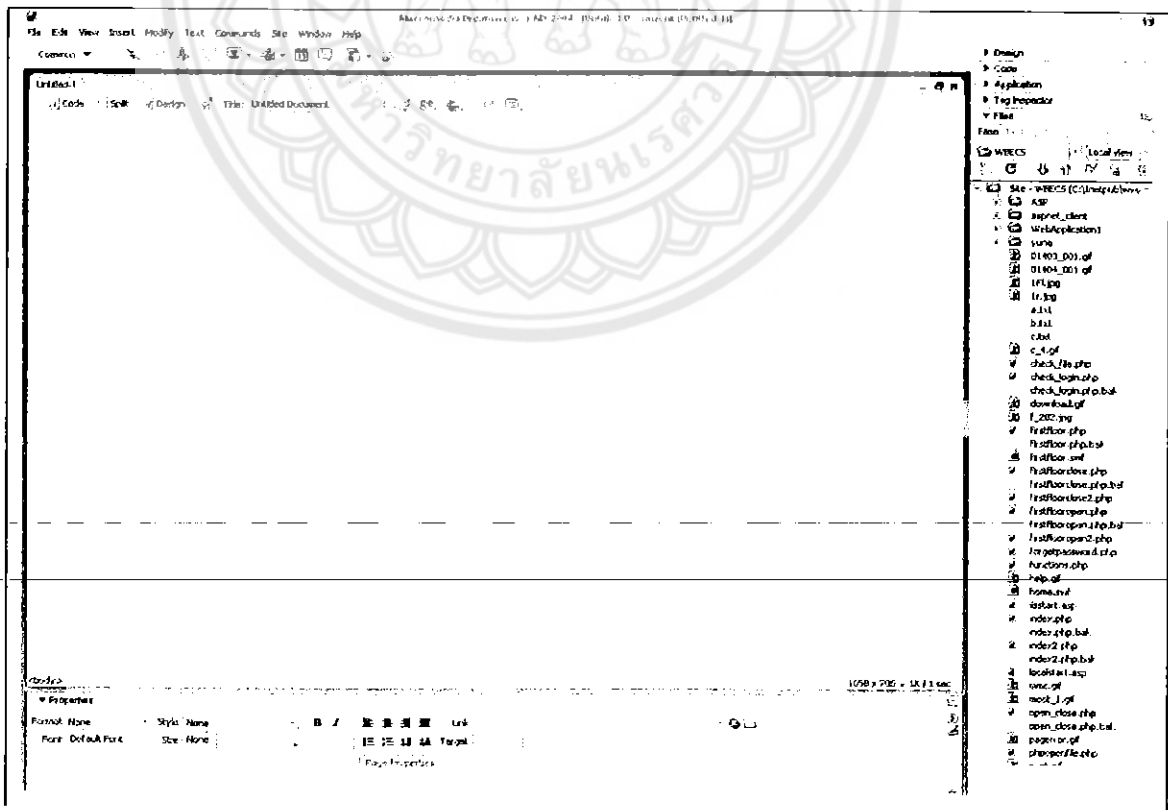
1.การใช้งาน DREAMWEAVER MX 2004

โปรแกรมที่ทำหน้าที่แสดงผลให้เครื่องลูกข่าย(Web - based) เขียนโดยใช้โปรแกรม Dreamweaver MX 2004 เนื่องจากเป็น โปรแกรมที่ทำให้การสร้างเว็บไซต์เป็นเรื่องง่าย นอกจากนี้โปรแกรมยังสามารถสร้าง interface และยังสามารถโค้ดเพื่อควบคุมการทำงาน หรือใส่ลูกเล่นอื่นๆที่น่าสนใจให้กับเว็บเพจได้ โดยแยกคุณสมบัติที่เพิ่มขึ้นของ Dreamweaver MX 2004 ได้ดังนี้

1. สนับสนุนความปลอดภัยในการส่งข้อมูลผ่าน FTP
2. มีการตรวจสอบคำสั่งแท็ก และของการใช้คำสั่งผ่านทางบราวเซอร์ได้
3. เขียนโค้ดได้รวดเร็วขึ้น เพราะมีเครื่องมือช่วยคลิกเมาส์ขวา
4. สามารถเพิ่มการปฏิสัมพันธ์ระหว่างหน้าเว็บเพจด้วย Macromedia Flash ภายใน Dreamweaver ได้

ทั้งนี้โปรแกรมที่ผู้ใช้จัดทำได้เลือกใช้โปรแกรม PHP มาพัฒนาเว็บเพจ เนื่องจากภาษาสคริปต์ PHP จะทำงานร่วมกับเอกสาร HTML โดยการแทรกโค้ดระหว่าง Tag HTML และสร้างไฟล์ที่มีนามสกุลเป็น .php, php3 หรือ php4 ซึ่งไวยากรณ์ที่ใช้ในสคริปต์ PHP เป็นการนำรูปแบบของภาษาต่างๆมารวมกันได้แก่ C, Perl และ JAVA ทำให้ผู้ใช้ที่มีพื้นฐานภาษาเหล่านี้อยู่แล้วสามารถศึกษาและใช้งานภาษา

สคริปต์ PHP ได้ไม่ยาก หน้าตาของ โปรแกรมที่ใช้พัฒนาเว็บเพจมีหน้าตาดังรูปที่ 3.3

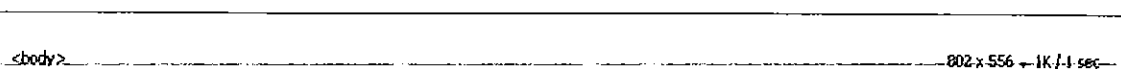


รูปที่ 6.1 หน้าตาโปรแกรมที่ใช้พัฒนาเว็บเพจ

จากรูปการทำงานของโปรแกรมมีส่วนการทำงานดังรูป

ส่วนที่ 1 แถบสถานะ

แถบสถานะของ Dreamweaver จะแสดงข้อมูลเพิ่มเติมเกี่ยวกับงานที่เรากำลังทำ โดยมี ส่วนประกอบต่างๆดังนี้



ส่วนใช้เลือกแท็ก HTML (Tag Selector)

จะแสดงโค้ด HTML ของเว็บเพจในส่วนที่เคอร์เซอร์อยู่เหนือ สามารถ Click ที่แท็ก HTML ที่แสดงเพื่อเลือกทำงานกับส่วน โค้ดที่ต้องการได้

- ส่วนแสดงขนาดของหน้าต่าง (Window Size)

เป็นส่วนที่แสดงขนาดของหน้าต่างเว็บเพจที่ได้กำหนดไว้

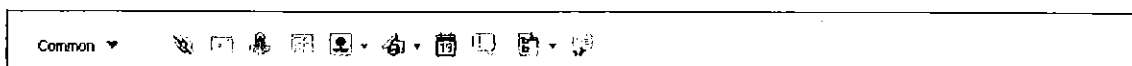
- ขนาดของเอกสารและเวลาที่ใช้ในการโหลด

เป็นส่วนที่แสดงเวลาที่ใช้ในการโหลดหน้าเว็บเพจ โดยถ้าหน้าเว็บนั้นมีขนาดใหญ่เพราะมีข้อความและรูปภาพจำนวนมาก ก็ต้องใช้เวลาโหลดนาน นอกจากนั้นความเร็วในการโหลดหน้าเว็บขึ้นกับความเร็วในการ โอนย้ายความเร็ว 28.8 (สามารถเปลี่ยนแปลงค่าได้) ข้อมูลของผู้ใช้ที่เชื่อมต่อกับเครือข่ายอินเทอร์เน็ตด้วย โดยเริ่มแรก Dreamweaver จะแสดงเวลานี้โดยอ้างอิงกับโมเด็ม

ส่วนที่ 2 แท็บ Insert (Insert Bar)

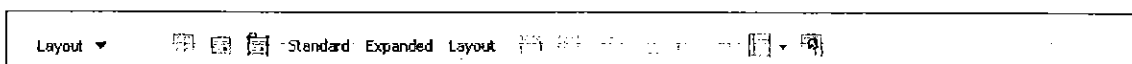
เป็นกลุ่มเครื่องมือที่ใช้สร้างองค์ประกอบต่างๆ ที่จะนำมาสร้างเป็นหน้าเว็บเพจ โดยมี เครื่องมือต่างๆทั้งหมด 12 กลุ่มโดยสามารถเลือกจากรายการ ดังนี้

- Common



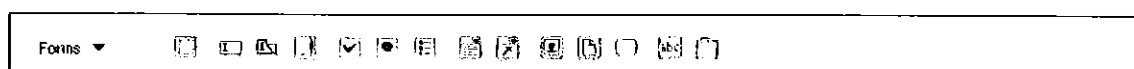
เป็นส่วนที่รวบรวมอ็อบเจกต์ต่างๆ ที่เรียกใช้บ่อย ได้แก่ การแทรกภาพ การแทรกตาราง เป็นต้น

- Layout



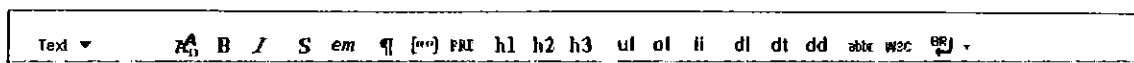
เป็นเครื่องมือที่ใช้เลือกมุมมองในการสร้างเว็บเพจ เช่น มุมมองปกติ หรือมุมมองแบบขยายเพื่อให้ เห็นอ็อบเจกต์ต่างๆ และองค์ประกอบในการสร้างเฟรมได้ชัดเจน

- Forms



เป็นเครื่องมือที่ใช้ในการสร้างฟอร์มรับข้อมูล เช่น ช่องรับข้อความ ปุ่มตัวเลือกต่างๆ

- Text



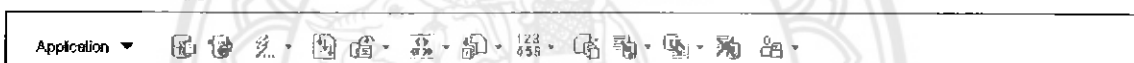
เป็นเครื่องมือในการปรับแต่งรูปแบบของตัวอักษรและข้อความ เช่น สร้างตัวหนา ตัวเอียง จัดย่อหน้า จักหัวข้อ (Head) และแทรกอักขระพิเศษ

- HTML



เป็นส่วนที่ใช้แทรกภาษา HTML ที่สำคัญเข้าไปในหน้าเว็บ

- Application



เป็นกลุ่มเครื่องมือที่ใช้งานด้านการเขียนสคริปต์ เช่น ASP, PHP, JSP มักใช้ร่วมกับฐานข้อมูล

- Flash elements



เป็นเครื่องมือที่ใช้ในการแทรกไฟล์ Flash ลงในหน้าเว็บ

ส่วนที่ 3 พื้นที่สร้างงาน (Document area)

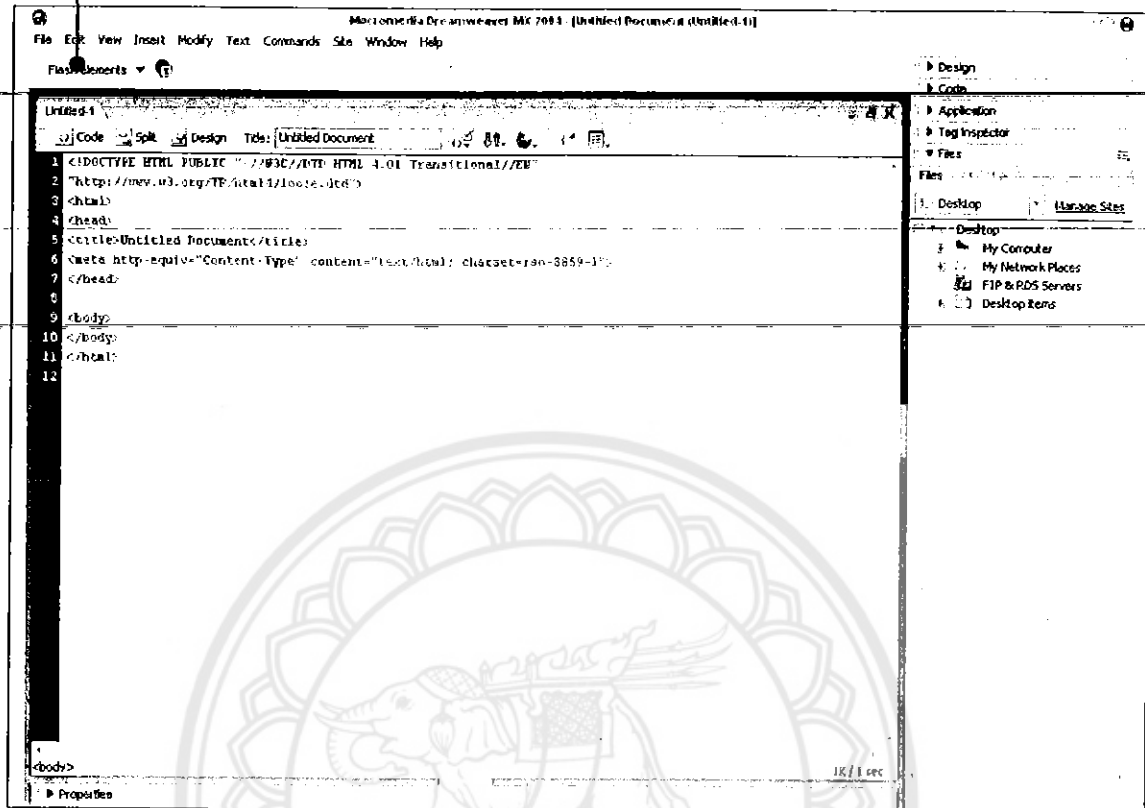
พื้นที่สร้างงานหรือ Document area ถือเป็นพื้นที่หลักของโปรแกรม เป็นพื้นที่ส่วนที่ใช้ในการสร้างหน้าเว็บเพจ โดยนำองค์ประกอบต่างๆ ที่ต้องการเข้ามาวางลงในบริเวณนี้

ตามปกติแล้วจะอยู่ที่มุมมองออกแบบ (Design) ซึ่งเป็นมุมมองที่ใช้ทำงานในการดึง Object เข้ามาวางเพื่อประกอบเป็นเว็บเพจ

เปิดมุมมองแสดงคำสั่ง

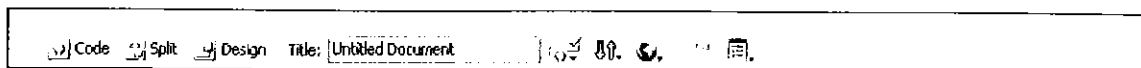
การเปิดดูคำสั่ง HTML ของเว็บเพจหน้าที่กำลังออกแบบอยู่ ให้เปิดเว็บเพจในมุมมอง Code เพื่อแสดงคำสั่งขึ้นมา นอกจากนี้ยังสามารถใช้พื้นที่สร้างงานในมุมมอง Code เพื่อเขียนคำสั่ง HTML หรือสคริปต์ภาษาอื่นๆ เช่น ASP, PHP หรือ JSP

คลิกเพื่อเลือกมุมมองแสดงคำสั่ง



ส่วนที่ 4 แถบ Toolbar

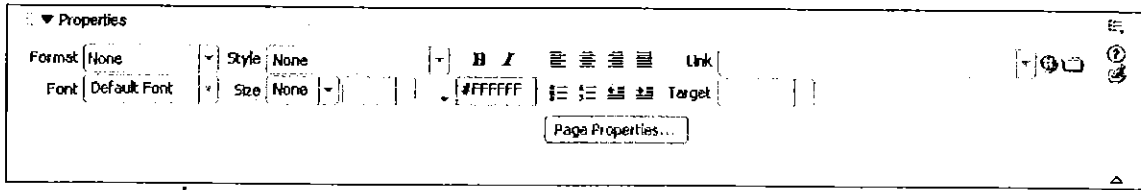
แถบ Toolbar เป็นส่วนที่อยู่ด้านบนของพื้นที่สร้างงาน ภายในประกอบด้วยซอร์ตคัตเพื่อเลือกเปิดเว็บเพจในมุมมองต่างๆ ไม่ว่าจะเป็นมุมมองออกแบบ มุมมองคำสั่ง หรือมุมมองแบบผสม นอกจากนี้ภายในแถบ Toolbar ยังประกอบด้วยส่วนของการกำหนดชื่อเว็บเพจ และซอร์ตคัตสำหรับเปิดดูการแสดงผลของเว็บเพจผ่านเบราว์เซอร์อีกด้วย



ส่วนที่ 5 หน้าต่าง Properties Inspector

หน้าต่าง Properties Inspector เป็นหน้าต่างเล็กๆ ที่อยู่ด้านล่างสุดของหน้าจอ โปรแกรม แต่กลับเป็นส่วนที่มีความสำคัญเป็นอย่างมาก เนื่องจากเป็นส่วนที่ทำหน้าที่ปรับแต่งคุณสมบัติของส่วนประกอบต่างๆ ในเว็บเพจ เช่น ข้อความ รูป ตาราง หรือลิงค์ เป็นต้น

โดยหน้าต่าง Properties Inspector จะมีหน้าต่างไม่เหมือนกันแล้วแต่ผู้ใช้เลือกทำอะไรอยู่ เช่น ถ้าในพื้นที่สร้างงานกำลังเลือกที่ข้อความ หน้าต่าง Properties Inspector จะแสดงคุณสมบัติเฉพาะของข้อความขึ้นมาให้ปรับแต่ง



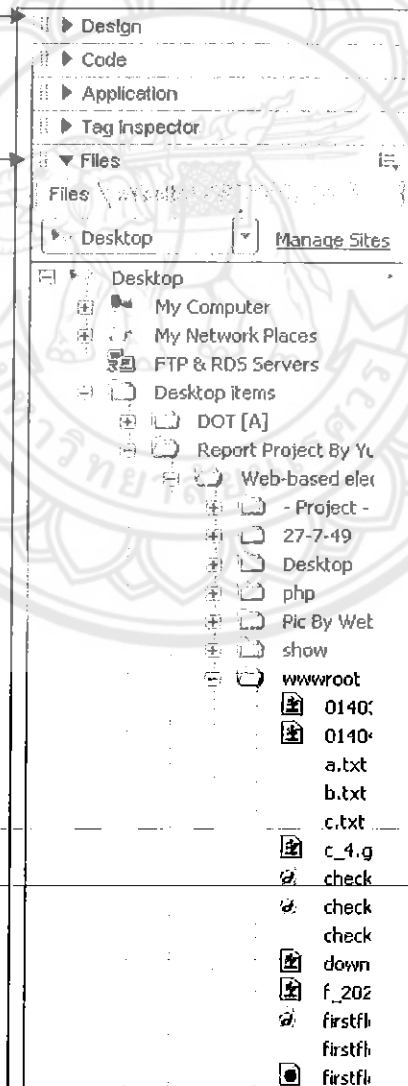
ส่วนที่ 6 กลุ่มหน้าต่างแผง (Panels)

ทางด้านขวาของ โปรแกรมจะมีกลุ่มหน้าต่างเล็กๆ อยู่จำนวนหนึ่งซึ่งหน้าต่างเล็กๆ เหล่านี้ เรียกว่า แผง. แต่ละแผงมีหน้าที่แตกต่างกันไป อย่างเช่น แผงสำหรับจัดการกับไฟล์ แผงสำหรับสร้าง CSS และแผงสำหรับสร้าง Behavior

ประโยชน์ที่สำคัญของแผงคือ ช่วยให้การทำงานได้สะดวกมากขึ้น เนื่องจากแต่ละแผงจะรวบรวมคำสั่งและการใช้งานในเรื่องนั้นๆ มาไว้ที่เดียวกัน อย่างเช่น ในแผง Files ก็จะมีแต่คำสั่งและซอร์คคัตสำหรับจัดการกับไฟล์และโฟลเดอร์รวมไว้ทั้งหมด

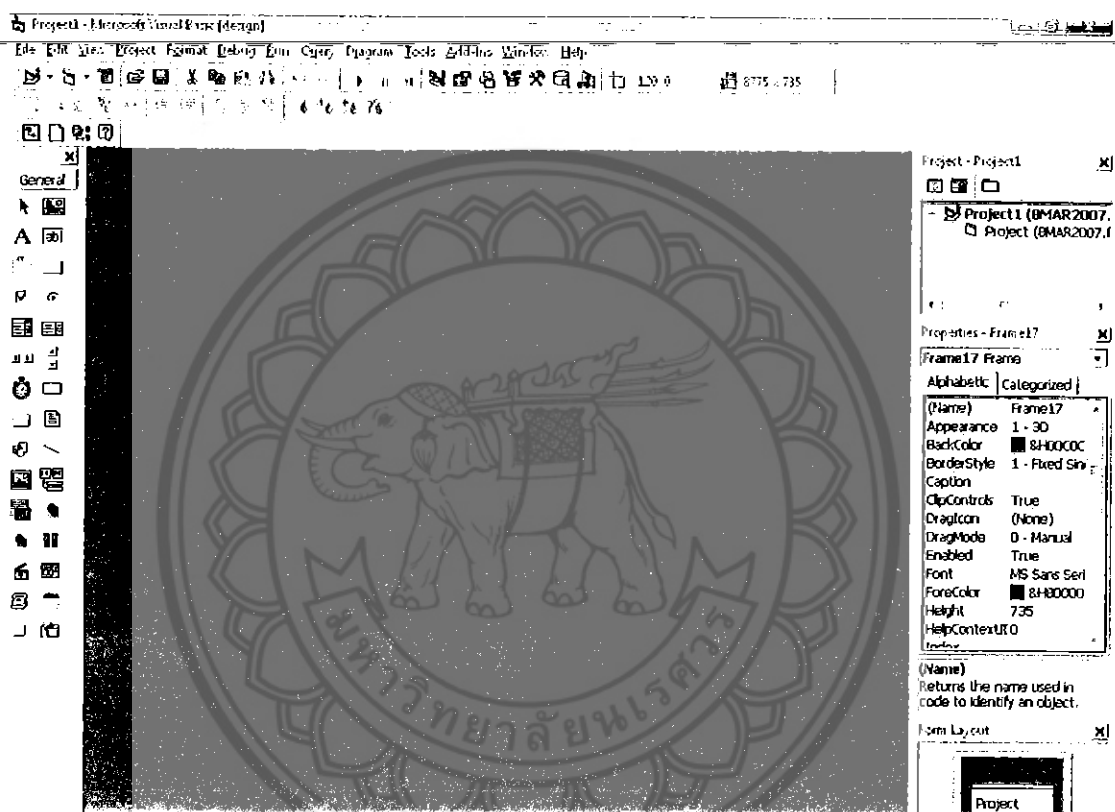
คลิกเพื่อขยายแผง

คลิกเพื่อย่อแผง



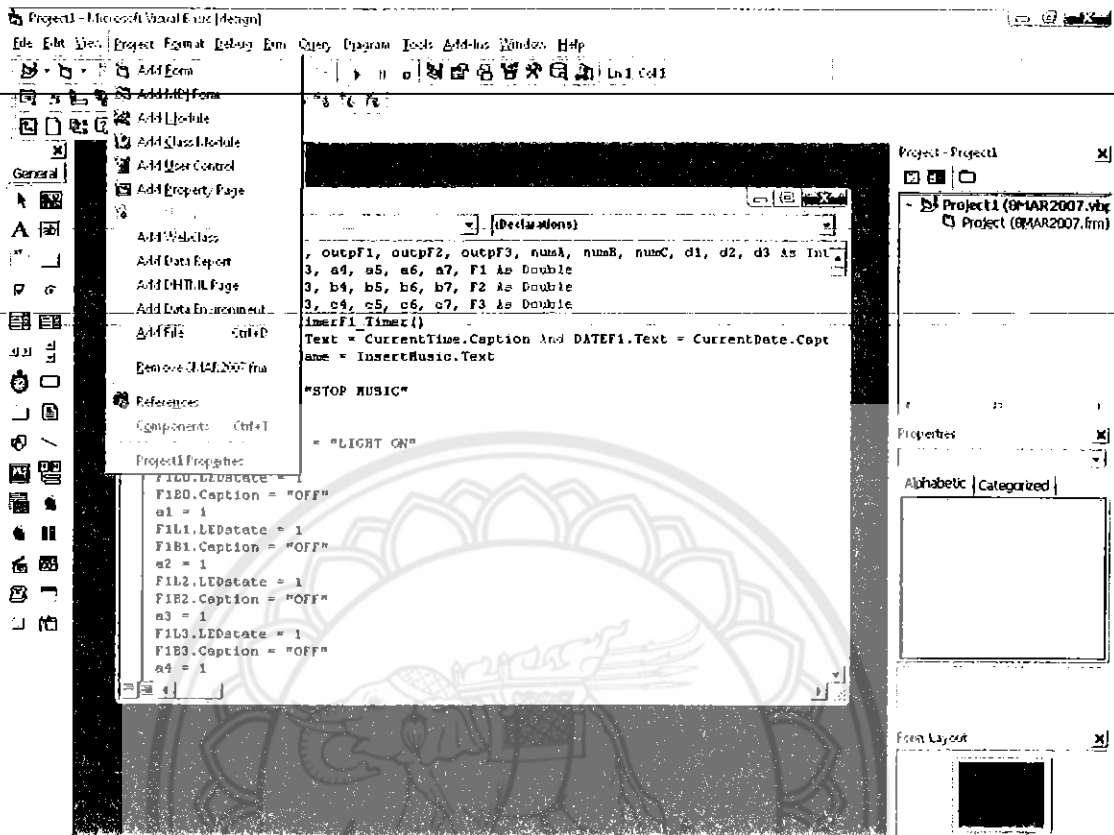
2. Visual Basic Programming Interface Hardware

การใช้ VB เขียนโปรแกรมติดต่อ I/O ผ่านทาง Port ของเครื่องคอมพิวเตอร์ จะผ่านเป็นทาง Serial Port (RS-232) หรือที่รู้จักในชื่อ Com1 สามารถทำการติดต่อกับ Hardware ภายนอกผ่าน Port ได้ และสามารถติดต่อผ่านระบบ Network โดยผ่านช่องทางการติดต่ออย่าง TCP/IP VB สามารถทำงานด้าน Interface Hardware และสะดวกรวดเร็ว ต่อการประยุกต์ใช้งาน



รูปที่ 6.2 หน้าตาโปรแกรม Visual Basic 6.0

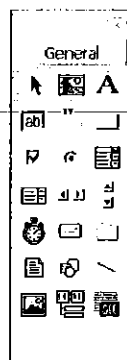
1. ขั้นตอนที่แรก เลือกที่เมนูบาร์ด้านบนของโปรแกรม Visual Basic ดังรูปด้านล่าง



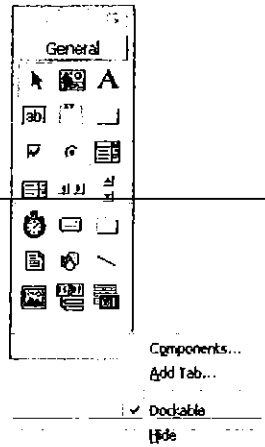
หรือ การเพิ่มคอนโทรลเพิ่มเติมโดยการ คลิกขวาที่ Toolbox

คอนโทรลเพิ่มเติมที่ Visual Basic เตรียมไว้ให้ นั้นไม่ได้ปรากฏอยู่ใน Toolbox เวลาจะใช้ จะต้องทำการเรียกมาใช้ ซึ่งมีวิธีดังนี้

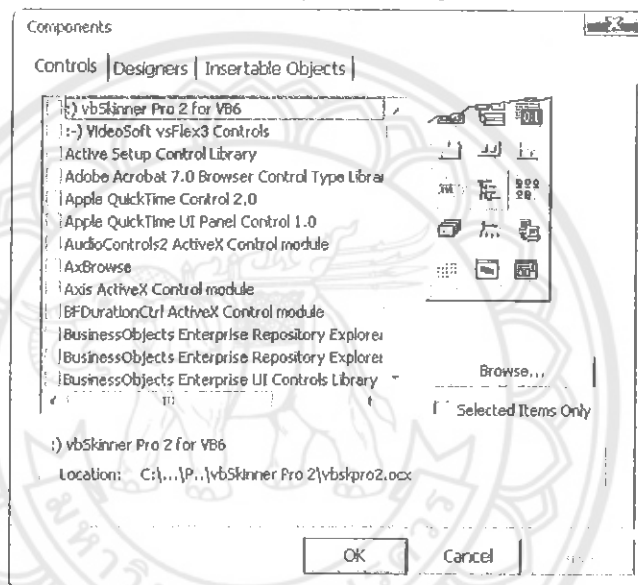
1. Toolbox เมื่อเปิดใช้งานครั้งแรก



2. คลิกขวาที่ Toolbox แล้วคลิกที่เมนู Components...

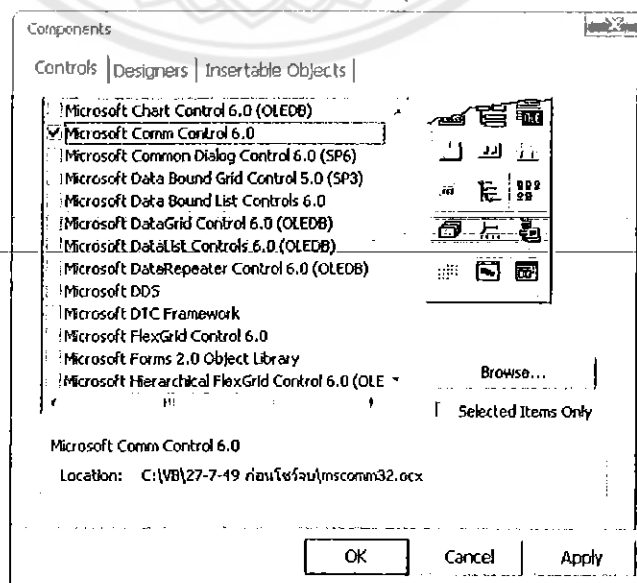


3. เมื่อกดคลิกที่เมนู Components... จะปรากฏหน้าต่างดังรูป



4. เลือกคอนโทรลจากแท็บที่ต้องการ

4.1 เลือกคอนโทรลที่เราต้องการ แล้วคลิกปุ่ม



4.2 การเพิ่มคอนโทรลที่โหลดเพิ่มเติม

เราสามารถหาคำว่าโหลดเพิ่มเติมได้ เช่น คอนโทรล ScottLED สามารถดาวน์โหลดได้จาก <http://members.tripod.com/forjsoft> ซึ่งคอนโทรลที่จะใช้เราสนใจ คอนโทรลที่ชื่อ

LED Control

Free ActiveX Controls (OCXs) By Scott Hather

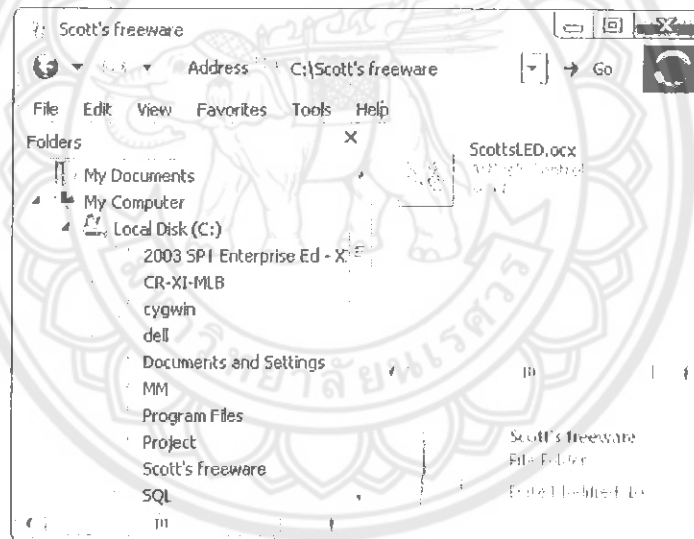
All these ActiveX Controls are intended for Programmers to simply drop into their application, and all should work in VB, VC++, Delphi or any Development program that supports ActiveX Controls...even web pages! There's no help, but very easy to use, descriptions of each method, property and event exposed are embedded in the OCX, so you can use a "object browser" or similar feature of your dev system to see how to use the control. Eg. In Visual Basic press F2

LED Control

Download
ScottsLED AX
Control (OCX)

This is a simple Red or Green LED control, two properties LEDstate is set to 0=off 1=on and 2=flashing, the flashrate is changed using LEDflashrate property.

เมื่อดาวน์โหลดเสร็จแล้ว

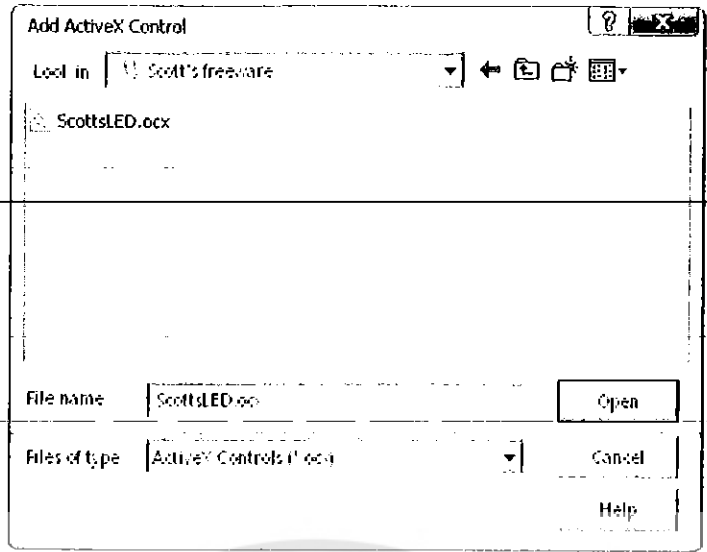


จากนั้นจะนำไปใช้งานโดยคลิกที่ปุ่ม


Browse...

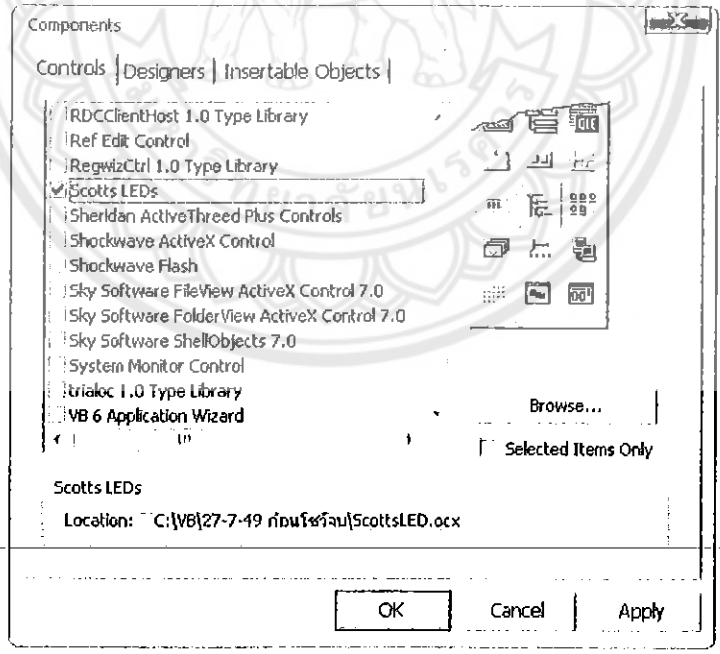
จะปรากฏหน้าต่าง

Add ActiveX Control ขึ้นมา ให้เลือกไฟล์จากโฟลเดอร์ที่ได้ดาวน์โหลดเก็บไว้

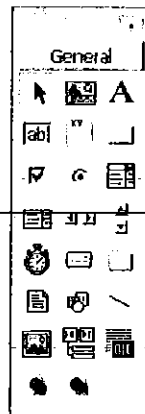


คลิกที่ชื่อไฟล์แล้วคลิก 

กลับมายังไดอะล็อกบ็อกซ์ Components จะปรากฏชื่อคอนโทรลที่เพิ่ม แล้วให้
เลือก คอนโทรล Scotts LEDs จากนั้นคลิกปุ่ม 

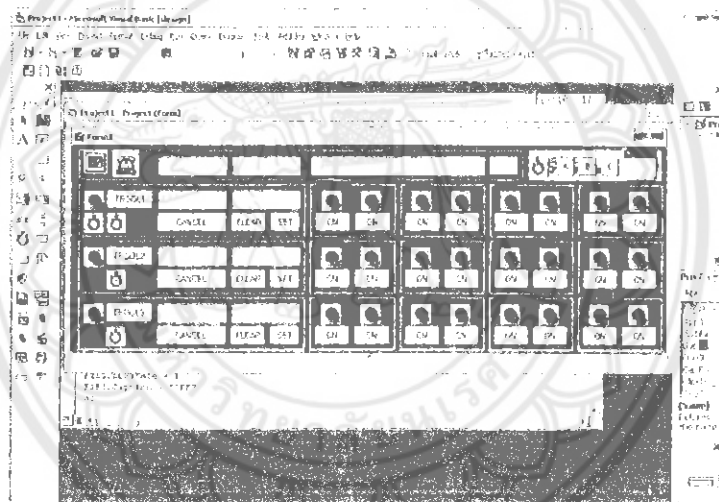


จะปรากฏคอนโทรลขึ้นที่ Toolbox ดังรูป



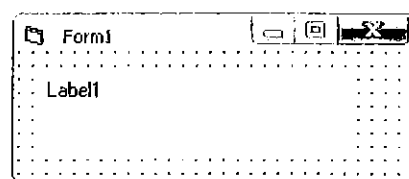
คอนโทรล VbSkinner สามารถดาวน์โหลดได้จาก <http://www.visual-basic.com.ar>

การออกแบบรูปร่างหน้าต่างของโปรแกรม



การออกแบบ ตัวโปรแกรมโดยใช้ คอนโทรลต่างๆ ดังนี้

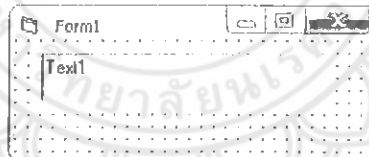
คอนโทรล Label มีหน้าที่สำหรับแสดงข้อความต่างๆ ลงบนตัวคอนโทรล ซึ่งผู้ใช้ไม่สามารถแก้ไขข้อความดังกล่าวได้ โดยการคลิกเลือกที่ **A** บนทูลบ็อกซ์ รูปแบบของ คอนโทรล Label เมื่อปรากฏบนฟอร์ม ดังรูป



แสดงคอนโทรล Label ที่อยู่บนฟอร์ม

0:25:25	29/3/2550																																																																																																																				
<p>Properties - CurrentTime</p> <p>CurrentTime Label</p> <p>Alphabetic Categorized </p> <table border="1"> <thead> <tr> <th>(Name)</th> <th>CurrentTime</th> </tr> </thead> <tbody> <tr><td>Alignment</td><td>2 - Center</td></tr> <tr><td>Appearance</td><td>1 - 3D</td></tr> <tr><td>AutoSize</td><td>False</td></tr> <tr><td>BackColor</td><td><input type="checkbox"/> &#x00FFFFFF&</td></tr> <tr><td>BackStyle</td><td>1 - Opaque</td></tr> <tr><td>BorderStyle</td><td>1 - Fixed Single</td></tr> <tr><td>Caption</td><td></td></tr> <tr><td>DataField</td><td></td></tr> <tr><td>DataFormat</td><td></td></tr> <tr><td>DataMember</td><td></td></tr> <tr><td>DataSource</td><td></td></tr> <tr><td>DragIcon</td><td>(None)</td></tr> <tr><td>DragMode</td><td>0 - Manual</td></tr> <tr><td>Enabled</td><td>True</td></tr> <tr><td>Font</td><td>MS Sans Serif</td></tr> <tr><td>ForeColor</td><td><input checked="" type="checkbox"/> &#x0000012&</td></tr> <tr><td>Height</td><td>375</td></tr> <tr><td>Index</td><td></td></tr> <tr><td>Left</td><td>1690</td></tr> <tr><td>LinkItem</td><td></td></tr> <tr><td>LinkMode</td><td>0 - None</td></tr> <tr><td>LinkTimeout</td><td>50</td></tr> <tr><td>LinkTopic</td><td></td></tr> <tr><td>MouseIcon</td><td>(None)</td></tr> <tr><td>MousePointer</td><td>0 - Default</td></tr> <tr><td>OLEDropMode</td><td>0 - None</td></tr> <tr><td>RightToLeft</td><td>False</td></tr> <tr><td>TabIndex</td><td>1</td></tr> </tbody> </table> <p>(Name) Returns the name used in code to identify an object.</p>	(Name)	CurrentTime	Alignment	2 - Center	Appearance	1 - 3D	AutoSize	False	BackColor	<input type="checkbox"/> �&	BackStyle	1 - Opaque	BorderStyle	1 - Fixed Single	Caption		DataField		DataFormat		DataMember		DataSource		DragIcon	(None)	DragMode	0 - Manual	Enabled	True	Font	MS Sans Serif	ForeColor	<input checked="" type="checkbox"/> &	Height	375	Index		Left	1690	LinkItem		LinkMode	0 - None	LinkTimeout	50	LinkTopic		MouseIcon	(None)	MousePointer	0 - Default	OLEDropMode	0 - None	RightToLeft	False	TabIndex	1	<p>Properties - CurrentDate</p> <p>CurrentDate Label</p> <p>Alphabetic Categorized </p> <table border="1"> <thead> <tr> <th>(Name)</th> <th>CurrentDate</th> </tr> </thead> <tbody> <tr><td>Alignment</td><td>2 - Center</td></tr> <tr><td>Appearance</td><td>1 - 3D</td></tr> <tr><td>AutoSize</td><td>False</td></tr> <tr><td>BackColor</td><td><input type="checkbox"/> &#x00FFFFFF&</td></tr> <tr><td>BackStyle</td><td>1 - Opaque</td></tr> <tr><td>BorderStyle</td><td>1 - Fixed Single</td></tr> <tr><td>Caption</td><td></td></tr> <tr><td>DataField</td><td></td></tr> <tr><td>DataFormat</td><td></td></tr> <tr><td>DataMember</td><td></td></tr> <tr><td>DataSource</td><td></td></tr> <tr><td>DragIcon</td><td>(None)</td></tr> <tr><td>DragMode</td><td>0 - Manual</td></tr> <tr><td>Enabled</td><td>True</td></tr> <tr><td>Font</td><td>MS Sans Serif</td></tr> <tr><td>ForeColor</td><td><input checked="" type="checkbox"/> &#x0000012&</td></tr> <tr><td>Height</td><td>375</td></tr> <tr><td>Index</td><td></td></tr> <tr><td>Left</td><td>3240</td></tr> <tr><td>LinkItem</td><td></td></tr> <tr><td>LinkMode</td><td>0 - None</td></tr> <tr><td>LinkTimeout</td><td>50</td></tr> <tr><td>LinkTopic</td><td></td></tr> <tr><td>MouseIcon</td><td>(None)</td></tr> <tr><td>MousePointer</td><td>0 - Default</td></tr> <tr><td>OLEDropMode</td><td>0 - None</td></tr> <tr><td>RightToLeft</td><td>False</td></tr> <tr><td>TabIndex</td><td>74</td></tr> </tbody> </table> <p>(Name) Returns the name used in code to identify an object.</p>	(Name)	CurrentDate	Alignment	2 - Center	Appearance	1 - 3D	AutoSize	False	BackColor	<input type="checkbox"/> �&	BackStyle	1 - Opaque	BorderStyle	1 - Fixed Single	Caption		DataField		DataFormat		DataMember		DataSource		DragIcon	(None)	DragMode	0 - Manual	Enabled	True	Font	MS Sans Serif	ForeColor	<input checked="" type="checkbox"/> &	Height	375	Index		Left	3240	LinkItem		LinkMode	0 - None	LinkTimeout	50	LinkTopic		MouseIcon	(None)	MousePointer	0 - Default	OLEDropMode	0 - None	RightToLeft	False	TabIndex	74
(Name)	CurrentTime																																																																																																																				
Alignment	2 - Center																																																																																																																				
Appearance	1 - 3D																																																																																																																				
AutoSize	False																																																																																																																				
BackColor	<input type="checkbox"/> �&																																																																																																																				
BackStyle	1 - Opaque																																																																																																																				
BorderStyle	1 - Fixed Single																																																																																																																				
Caption																																																																																																																					
DataField																																																																																																																					
DataFormat																																																																																																																					
DataMember																																																																																																																					
DataSource																																																																																																																					
DragIcon	(None)																																																																																																																				
DragMode	0 - Manual																																																																																																																				
Enabled	True																																																																																																																				
Font	MS Sans Serif																																																																																																																				
ForeColor	<input checked="" type="checkbox"/> &																																																																																																																				
Height	375																																																																																																																				
Index																																																																																																																					
Left	1690																																																																																																																				
LinkItem																																																																																																																					
LinkMode	0 - None																																																																																																																				
LinkTimeout	50																																																																																																																				
LinkTopic																																																																																																																					
MouseIcon	(None)																																																																																																																				
MousePointer	0 - Default																																																																																																																				
OLEDropMode	0 - None																																																																																																																				
RightToLeft	False																																																																																																																				
TabIndex	1																																																																																																																				
(Name)	CurrentDate																																																																																																																				
Alignment	2 - Center																																																																																																																				
Appearance	1 - 3D																																																																																																																				
AutoSize	False																																																																																																																				
BackColor	<input type="checkbox"/> �&																																																																																																																				
BackStyle	1 - Opaque																																																																																																																				
BorderStyle	1 - Fixed Single																																																																																																																				
Caption																																																																																																																					
DataField																																																																																																																					
DataFormat																																																																																																																					
DataMember																																																																																																																					
DataSource																																																																																																																					
DragIcon	(None)																																																																																																																				
DragMode	0 - Manual																																																																																																																				
Enabled	True																																																																																																																				
Font	MS Sans Serif																																																																																																																				
ForeColor	<input checked="" type="checkbox"/> &																																																																																																																				
Height	375																																																																																																																				
Index																																																																																																																					
Left	3240																																																																																																																				
LinkItem																																																																																																																					
LinkMode	0 - None																																																																																																																				
LinkTimeout	50																																																																																																																				
LinkTopic																																																																																																																					
MouseIcon	(None)																																																																																																																				
MousePointer	0 - Default																																																																																																																				
OLEDropMode	0 - None																																																																																																																				
RightToLeft	False																																																																																																																				
TabIndex	74																																																																																																																				
แสดงเวลาตามเครื่อง server	แสดงวันตามเครื่อง server																																																																																																																				

คอนโทรล Textbox จะมีหน้าที่สำหรับใช้แสดง หรือรับข้อมูลจากผู้ใช้โดยยอมให้ผู้ใช้สามารถแก้ไขข้อมูลดังกล่าวได้ด้วย โดยการคลิกเลือกที่ `label` บนทูลบ็อกซ์ รูปแบบของ คอนโทรล Label เมื่อปรากฏบนฟอร์ม ดังรูป



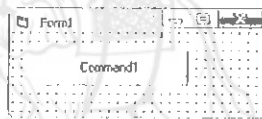
แสดงคอนโทรล Textbox ที่อยู่บนฟอร์ม

29/3/2550

รักเรณ Living In C Major · ได้ ดีกดีสิทธ์ เวชสุภ

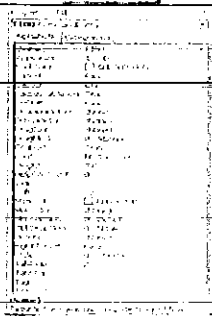
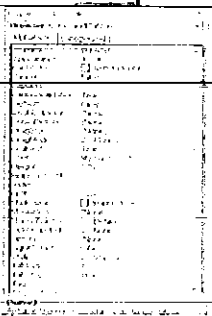
<p>Properties - DATEF1</p> <p>DATEF1 TextBox</p> <p>Alphabetic Categorized</p> <p>(Name) DATEF1 Alignment 2 - Center Appearance 1 - 3D BackColor <input type="checkbox"/> &H00000058 BorderStyle 1 - Fixed Single CausesValidation True DataField DataFormat Date DataMember DataSource DragIcon (None) DragMode 0 - Manual Enabled True Font MS Sans Serif ForeColor <input checked="" type="checkbox"/> &H00000088 Height 375 HelpContextID 0 HideSelection True</p> <p>(Name) Returns the name used in code to identify an object.</p>	<p>Properties - DisplayTimeF1</p> <p>DisplayTimeF1 TextBox</p> <p>Alphabetic Categorized</p> <p>(Name) DisplayTimeF1 Alignment 2 - Center Appearance 1 - 3D BackColor <input type="checkbox"/> &H00000058 BorderStyle 1 - Fixed Single CausesValidation True DataField DataFormat Time DataMember DataSource DragIcon (None) DragMode 0 - Manual Enabled True Font MS Sans Serif ForeColor <input checked="" type="checkbox"/> &H00000088 Height 375 HelpContextID 0 HideSelection True</p> <p>(Name) Returns the name used in code to identify an object.</p>
<p>แสดงแก้ไข วันตามเวลาที่ต้องปลูก</p>	<p>แสดงชื่อเพลงที่ได้ทำการเลือก</p>

คอนโทรล CommandButton จะมีหน้าที่สำหรับสร้างปุ่มกด หรือปุ่มตอบรับ โดยการคลิก
 เลือกที่ — บนทูลบ็อกซ์ รูปแบบของ คอนโทรล Label เมื่อปรากฏบนฟอร์ม ดังรูป




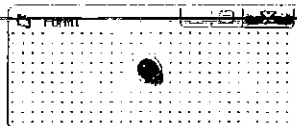
แสดงคอนโทรล CommandButton ที่อยู่บนฟอร์ม

<p>FR00L1</p>	<p>ON</p>
<p>สร้างปุ่มควบคุมแต่ละชั้น</p>	<p>ปุ่มเปิดไฟทั้งชั้น</p>
<p>CLEAR</p>	<p>SET</p>

สร้างปุ่ม clear	สร้างปุ่ม set
ON
	
สร้างปุ่มเปิด ปิดแต่ละดวง	สร้างปุ่ม browser



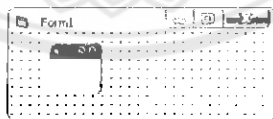
คอนโทรล ScottLED จะมีหน้าที่แสดงสถานะของ LED โดยการคลิกเลือกที่  บนทูล บั
ทูลบ็อกซ์ รูปแบบของ คอนโทรล ScottLED เมื่อปรากฏบนฟอร์ม ดังรูป



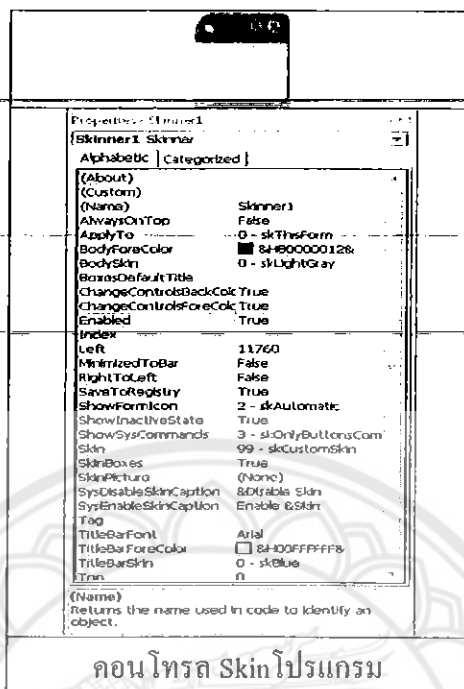
แสดงคอนโทรล ScottLED ที่อยู่บนฟอร์ม



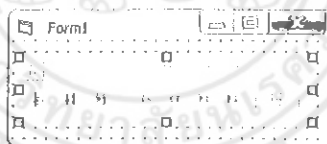
คอนโทรล VbSkinner จะใช้สำหรับเปลี่ยน Skin ของฟอร์ม โดยการคลิกเลือกที่  บน
ทูลบ็อกซ์ รูปแบบของ คอนโทรล VbSkinner เมื่อปรากฏบนฟอร์ม ดังรูป



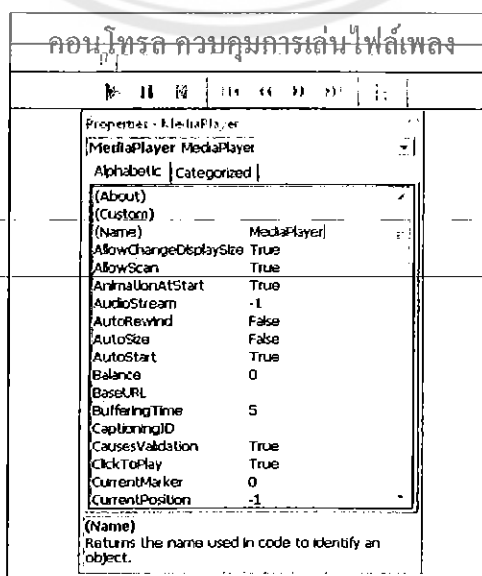
แสดงคอนโทรล VbSkinner ที่อยู่บนฟอร์ม




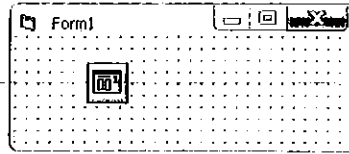
คอนโทรลMediaPlayer จะมีหน้าที่สำหรับเล่น file .mp3 .wav โดยการคลิกเลือกที่  บน
ทูลบ็อกซ์ รูปแบบของ คอนโทรล MediaPlayer เมื่อปรากฏบนฟอร์ม ดังรูป



แสดงคอนโทรล MediaPlayer ที่อยู่บนฟอร์ม




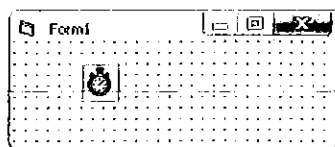
คอนโทรล CommonDialog จะมีหน้าที่สำหรับ Browser หา file โดยการคลิกเลือกที่  บน
ทูลบ็อกซ์ รูปแบบของ คอนโทรล CommonDialog เมื่อปรากฏบนฟอร์ม ดังรูป






แสดงคอนโทรล CommonDialog ที่อยู่บนฟอร์ม


คอนโทรล ควบคุมการ BROWSER
ไปยังตำแหน่ง folder ที่ต้องการ

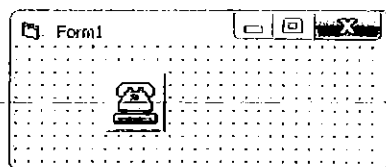
คอนโทรล Timer จะมีหน้าที่สำหรับเป็นตัวจับเวลาในการประมวลผล มีลักษณะเมื่อนำไปใช้งานแล้วจะไม่ปรากฏขึ้นมาทันที มันจะทำงานอยู่เบื้องหลัง โดยการคลิกเลือกที่  บนทูลบ็อกซ์ รูปแบบของ คอนโทรล Timer เมื่อปรากฏบนฟอร์ม ดังรูป



แสดงคอนโทรล Timer ที่อยู่บนฟอร์ม

 <p>Properties - AlarmTimerF1</p> <p>AlarmTimerF1 Timer</p> <p>Alphabetic Categorized</p> <table border="1"> <tr><td>(Name)</td><td>AlarmTimerF1</td></tr> <tr><td>Enabled</td><td>False</td></tr> <tr><td>Index</td><td></td></tr> <tr><td>Interval</td><td>1</td></tr> <tr><td>Left</td><td>600</td></tr> <tr><td>Tag</td><td></td></tr> <tr><td>Top</td><td>720</td></tr> </table> <p>(Name) Returns the name used in code to identify an object.</p>	(Name)	AlarmTimerF1	Enabled	False	Index		Interval	1	Left	600	Tag		Top	720	 <p>Properties - Timerread</p> <p>Timerread Timer</p> <p>Alphabetic Categorized</p> <table border="1"> <tr><td>(Name)</td><td>Timerread</td></tr> <tr><td>Enabled</td><td>True</td></tr> <tr><td>Index</td><td></td></tr> <tr><td>Interval</td><td>1000</td></tr> <tr><td>Left</td><td>9000</td></tr> <tr><td>Tag</td><td></td></tr> <tr><td>Top</td><td>240</td></tr> </table> <p>(Name) Returns the name used in code to identify an object.</p>	(Name)	Timerread	Enabled	True	Index		Interval	1000	Left	9000	Tag		Top	240
(Name)	AlarmTimerF1																												
Enabled	False																												
Index																													
Interval	1																												
Left	600																												
Tag																													
Top	720																												
(Name)	Timerread																												
Enabled	True																												
Index																													
Interval	1000																												
Left	9000																												
Tag																													
Top	240																												
<p>การกำหนดเวลาปลูก แต่ละชั้น</p>	<p>การกำหนดเวลาอ่าน Text file</p>																												
																													
<p>Properties - TimerF1</p> <p>TimerF1 Timer</p> <p>Alphabetic Categorized</p> <table border="1"> <tr><td>(Name)</td><td>TimerF1</td></tr> <tr><td>Enabled</td><td>False</td></tr> <tr><td>Index</td><td></td></tr> <tr><td>Interval</td><td>60</td></tr> <tr><td>Left</td><td>120</td></tr> <tr><td>Tag</td><td></td></tr> <tr><td>Top</td><td>720</td></tr> </table> <p>(Name) Returns the name used in code to identify an object.</p>		(Name)	TimerF1	Enabled	False	Index		Interval	60	Left	120	Tag		Top	720														
(Name)	TimerF1																												
Enabled	False																												
Index																													
Interval	60																												
Left	120																												
Tag																													
Top	720																												
<p>การติดต่อเวลาของเครื่อง server กับ โปรแกรม</p>																													

คอนโทรล MSComm จะมีหน้าที่สำหรับเป็นตัวจับเชื่อมต่อพอร์ท คอมพิวเตอร์ โดยการคลิกเลือกที่  บนทูลบ็อกซ์ รูปแบบของ คอนโทรล Mscomm เมื่อปรากฏบนฟอร์ม ดังรูป



แสดงคอนโทรล Mscomm ที่อยู่บนฟอร์ม

Properties - MSComm1

MSComm1 MSComm

Alphabetic | Categorized |

(About)
 (Custom)
 (Name) MSComm1
 CommPort 1
 DTREnable True
 EOFEnable False
 Handshaking 0 - comNone
 InBufferSize 1024
 Index
 InputLen 0
 InputMode 0 - comInputModeText
 Left 720
 NullDiscard False
 OutBufferSize 512
 ParityReplace ?
 RThreshold 0
 RTSEnable False
 Settings 9600,n,8,1
 SThreshold 0
 Tag
 Top 120

(Name)
 Returns the name used in code to identify an object.

การเชื่อมต่อคอมพิวเตอร์กับRS232

Property ชื่อ PortOpen คือ จะเปิดให้พอร์ทใช้งานหรือไม่ ถ้าเปิด =True ถ้าปิด =False มาดูการเขียนโค้ดกัน ตัวอย่าง MSComm1.PortOpen = True

ตารางกำหนดคุณสมบัติให้กับคอนโทรลต่างๆ ดังตาราง

คอนโทรล	คุณสมบัติ	ค่าที่กำหนด
ForgeLEDb	Name	F1L0
ForgeLEDb	Name	F1L1
ForgeLEDb	Name	F1L2
ForgeLEDb	Name	F1L3
ForgeLEDb	Name	F1L4
ForgeLEDb	Name	F1L5
ForgeLEDb	Name	F1L6
ForgeLEDb	Name	F1L7
ForgeLEDb	Name	F2L0
ForgeLEDb	Name	F2L1
ForgeLEDb	Name	F2L2
ForgeLEDb	Name	F2L3
ForgeLEDb	Name	F2L4
ForgeLEDb	Name	F2L5
ForgeLEDb	Name	F2L6
ForgeLEDb	Name	F2L7
ForgeLEDb	Name	F3L0
ForgeLEDb	Name	F3L1
ForgeLEDb	Name	F3L2
ForgeLEDb	Name	F3L3
ForgeLEDb	Name	F3L4
ForgeLEDb	Name	F3L5
ForgeLEDb	Name	F3L6
ForgeLEDb	Name	F3L7
ForgeLEDb	Name	SF1
ForgeLEDb	Name	SF2
ForgeLEDb	Name	SF3
Timer	Name	TimerF1

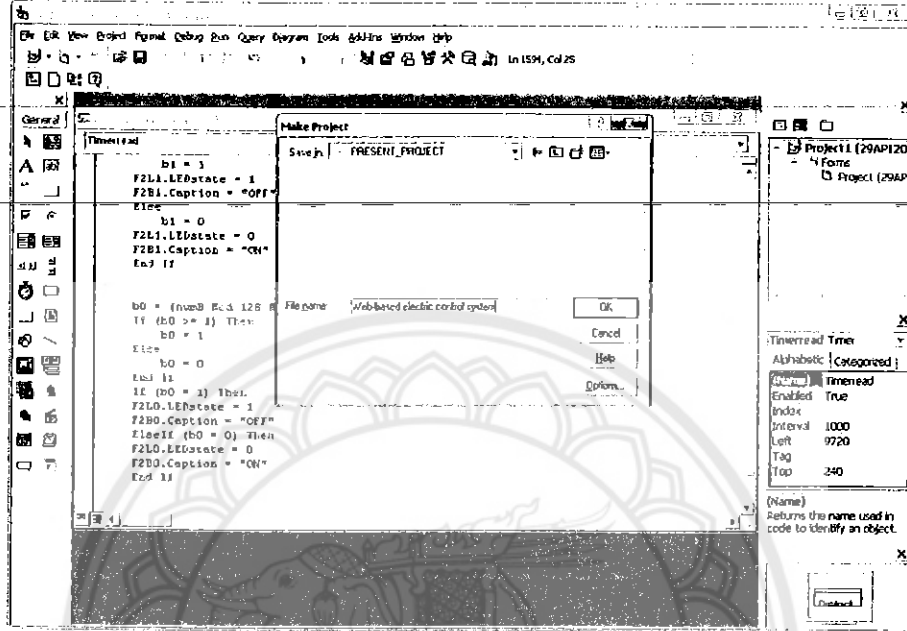
	Interval	60
Timer	Name	AlamTimerF1
	Interval	1
Timer	Name	AlamTimerF2
	Interval	1
Timer	Name	AlamTimerF3
	Interval	1
Timer	Name	Timerread
	Interval	1000
CommonDialog	Name	CommonDialog1
MSComm1	Name	MSComm
TextBox	Name	DisplayTimeF1
TextBox	Name	DisplayTimeF2
TextBox	Name	DisplayTimeF3
TextBox	Name	DATEF1
TextBox	Name	DATEF2
TextBox	Name	DATEF3
TextBox	Name	InsertMusic
Label	Name	CurrentTime
Label	Name	CurrentDate
Label	Name	Showtext1
Label	Name	showtext2
Label	Name	Showtext3
CommandButton	Name	F1B0
CommandButton	Name	F1B1
CommandButton	Name	F1B2
CommandButton	Name	F1B3
CommandButton	Name	F1B4
CommandButton	Name	F1B5
CommandButton	Name	F1B6
CommandButton	Name	F1B7

CommandButton	Name	F2B0
CommandButton	Name	F2B1
CommandButton	Name	F2B2
CommandButton	Name	F2B3
CommandButton	Name	F2B4
CommandButton	Name	F2B5
CommandButton	Name	F2B6
CommandButton	Name	F2B7
CommandButton	Name	F3B0
CommandButton	Name	F3B1
CommandButton	Name	F3B2
CommandButton	Name	F3B3
CommandButton	Name	F3B4
CommandButton	Name	F3B5
CommandButton	Name	F3B6
CommandButton	Name	F3B7
CommandButton	Name	FROOL1
CommandButton	Name	FROOL2
CommandButton	Name	FROOL3
CommandButton	Name	setswF1
CommandButton	Name	setswF2
CommandButton	Name	setswF3
CommandButton	Name	ClearF1
CommandButton	Name	ClearF2
CommandButton	Name	ClearF3
CommandButton	Name	SETF1
CommandButton	Name	SETF2
CommandButton	Name	SETF3
CommandButton	Name	writel
CommandButton	Name	Write2
CommandButton	Name	Write3

CommandButton	Name	Browser
Skinner	Name	Skinner1
MediaPlayer	Name	MediaPlayer
Frame	Name	Frame1
Frame	Name	Frame2
Frame	Name	Frame3
Frame	Name	Frame4
Frame	Name	Frame5
Frame	Name	Frame6
Frame	Name	Frame7
Frame	Name	Frame8
Frame	Name	Frame9
Frame	Name	Frame10
Frame	Name	Frame11
Frame	Name	Frame12
Frame	Name	Frame13
Frame	Name	Frame14
Frame	Name	Frame15
Frame	Name	Frame16
Frame	Name	Frame17

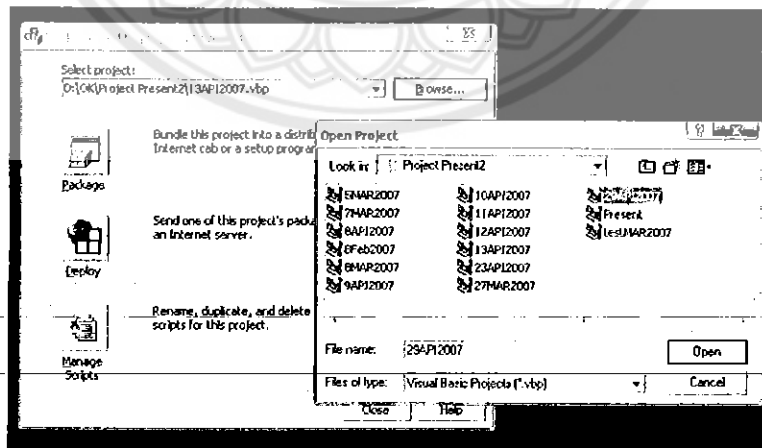
การ Compile project ให้เป็น .exe

หลังจากที่ทำโปรแกรมเสร็จขั้นตอนต่อไป จะทำเป็น.exe เพื่อให้ง่ายต่อการใช้งาน เริ่มจาก click ที่ File เลือก Make Project.exe จากนั้นจะแสดงดังรูป เพื่อทำการตั้งชื่อโปรเจก และหาที่เก็บ project



การสร้างชุดติดตั้ง

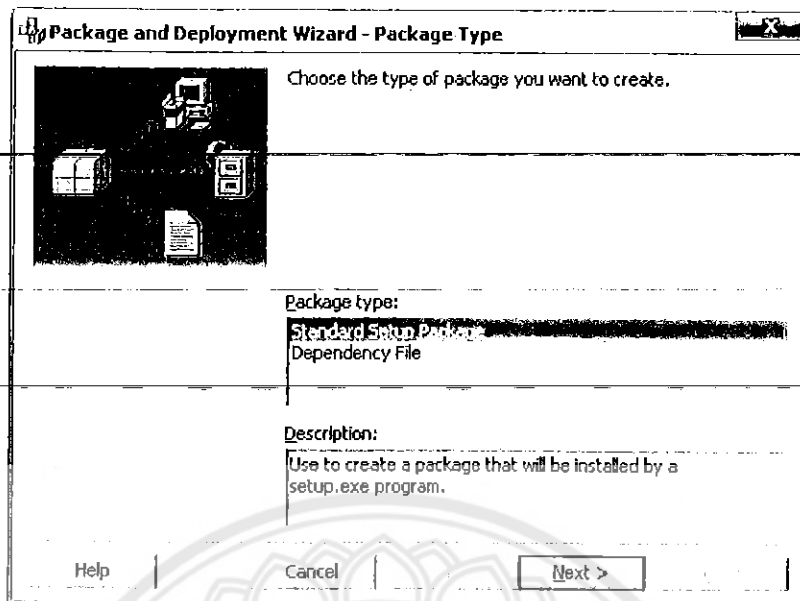
เมื่อได้ File.exe จากนั้น ให้เลือกคำสั่ง Start/All Programs/Microsoft Visual Studio 6.0/Microsoft Visual Studio 6.0 tools/ Package & Deployment Wizard ให้ทำการ browse... เลือกโปรแกรมที่ต้องการสร้างตัวติดตั้ง



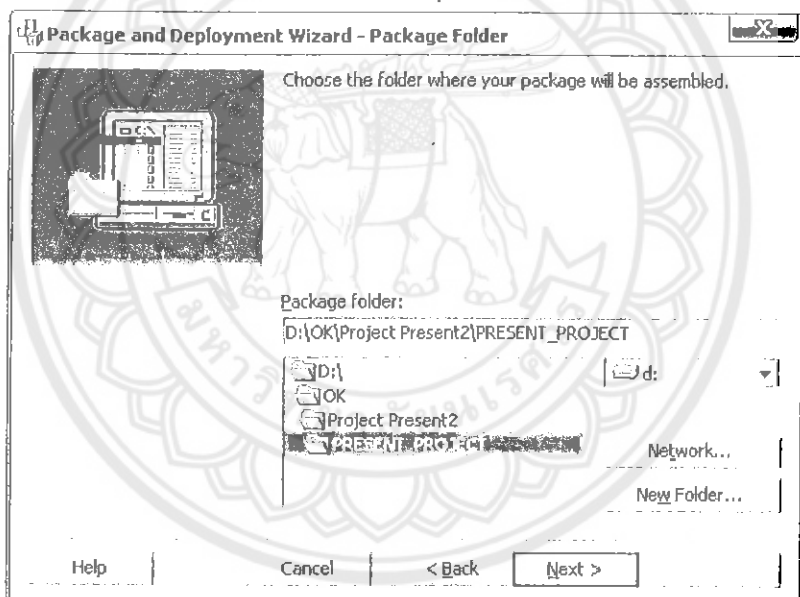
จากนั้นกด  Package

ทำการเลือกแบบ Standard Setup Package ซึ่งหมายถึงการสร้างชุดติดตั้งแบบทั่วไป

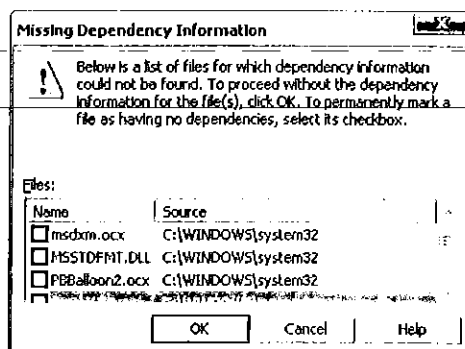
จากนั้น Click ปุ่ม Next >



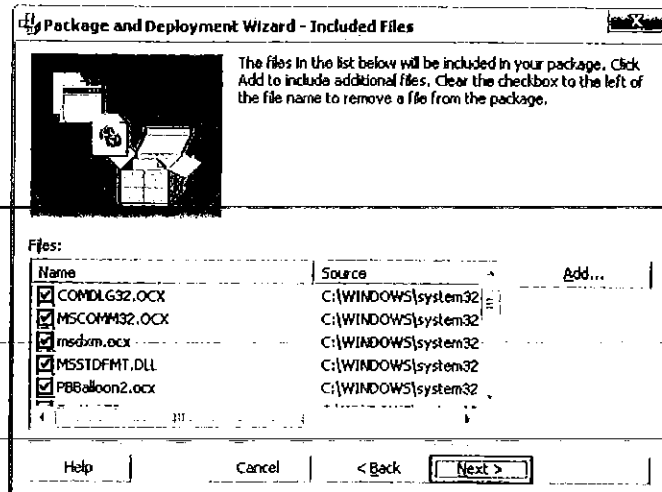
เลือก Folder ที่ต้องการเก็บ file ติดตั้งจากนั้น click ปุ่ม Next >



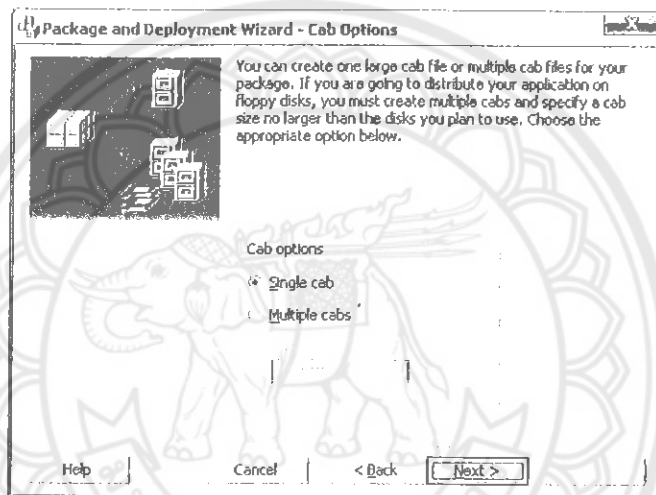
จะนั้นให้ Click ปุ่ม OK



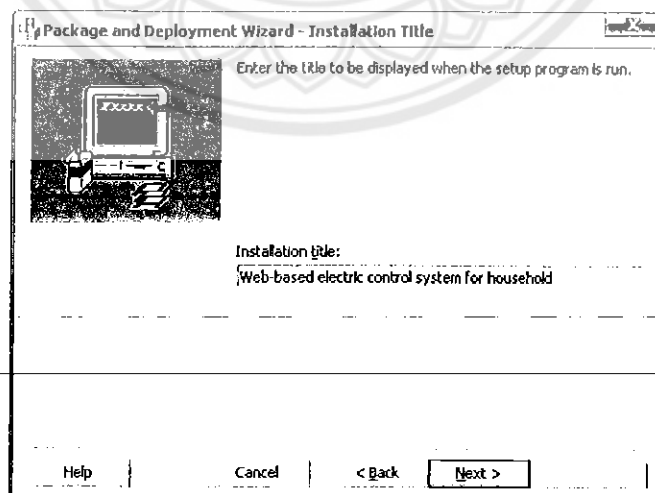
จากนั้นจะแสดงรายละเอียดของ File ที่จะถูกสร้างขึ้นมา ให้ทำการ click ปุ่ม Next >



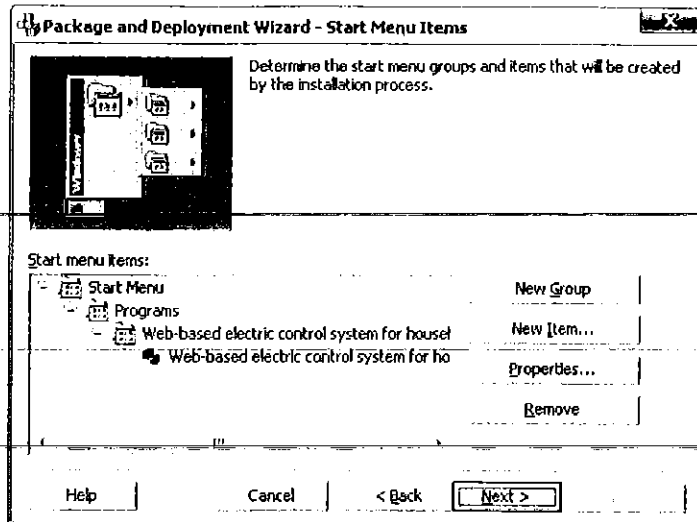
Clickปุ่ม Next >



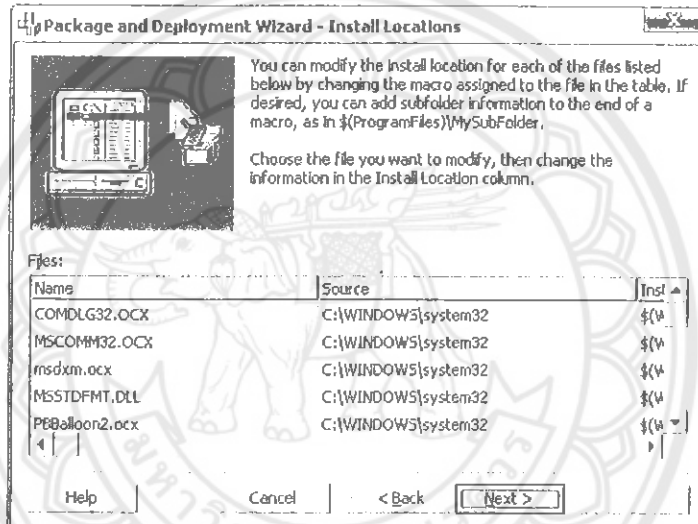
ช่องป้อนข้อความระหว่างขั้นตอนการติดตั้งจะแสดงข้อความดังกล่าว จากนั้น clickปุ่ม Next >



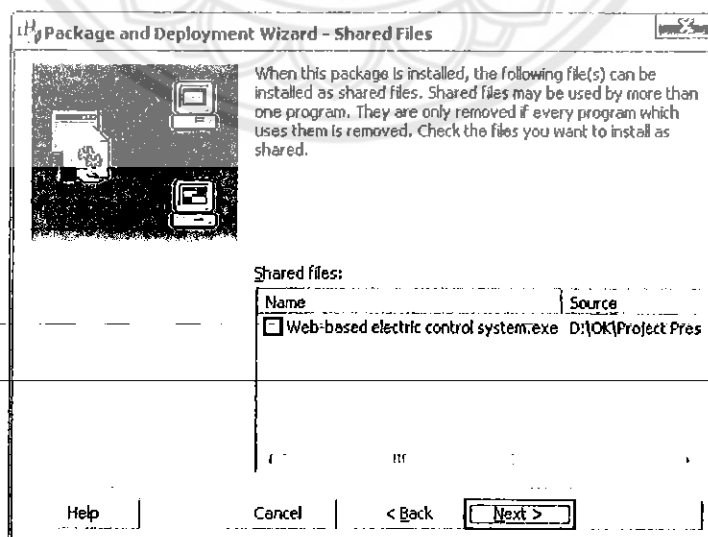
เป็นการเลือก Group ในการติดตั้งและสามารถทำการเปลี่ยนชื่อ Project ได้โดยเลือก Properties...
จากนั้น clickปุ่ม Next >



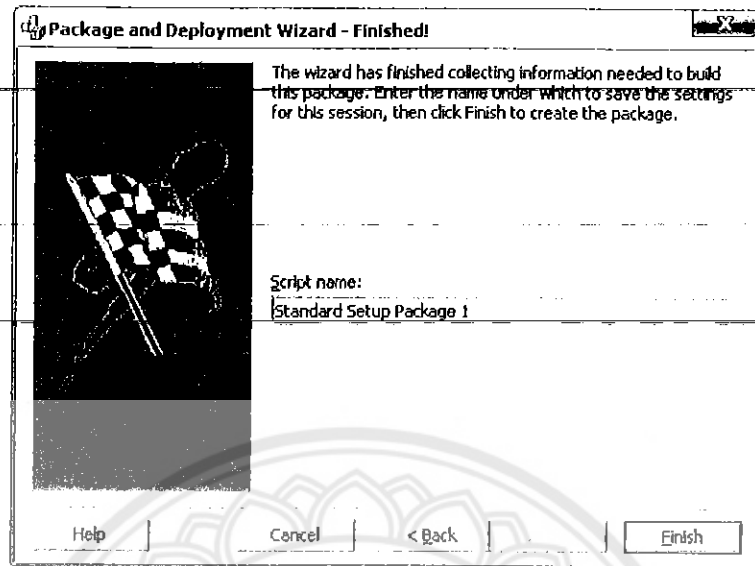
จากนั้น Click ปุ่ม Next >



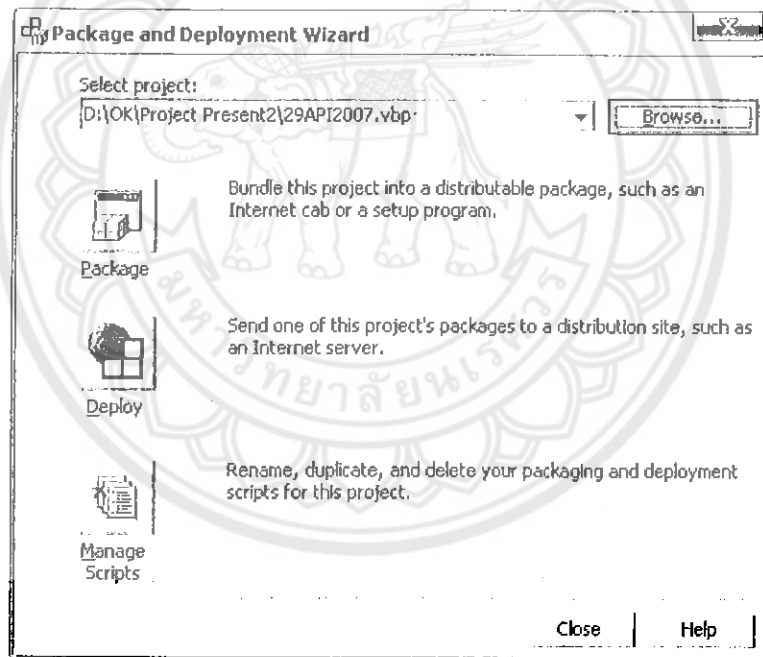
จากนั้น Click ปุ่ม Next >



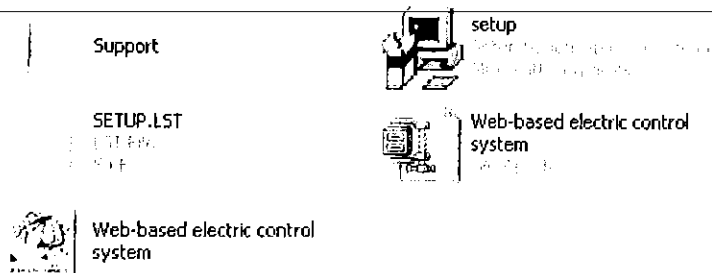
จากนั้น Click ปุ่ม Finish




จากนั้น Click ปุ่ม Close

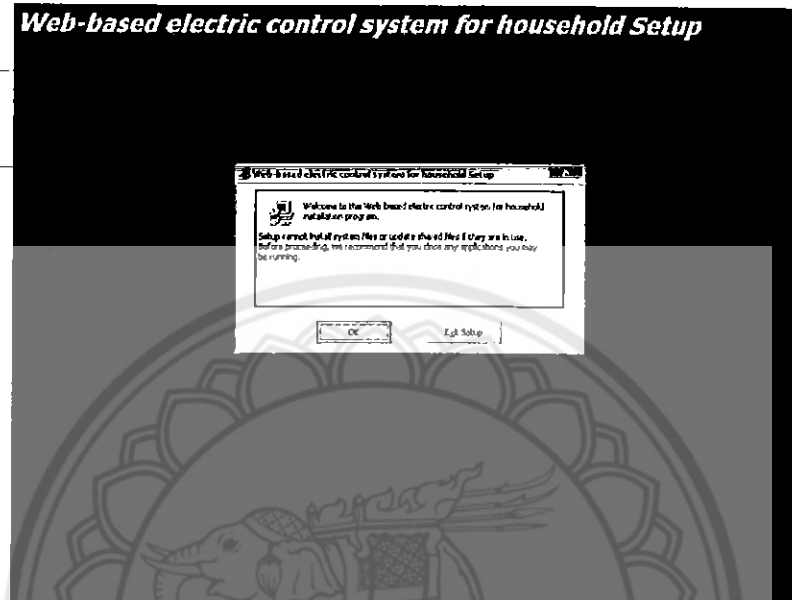


เมื่อเข้าไปยัง Folder ที่เก็บ File ติดตั้ง จะแสดงดังรูป

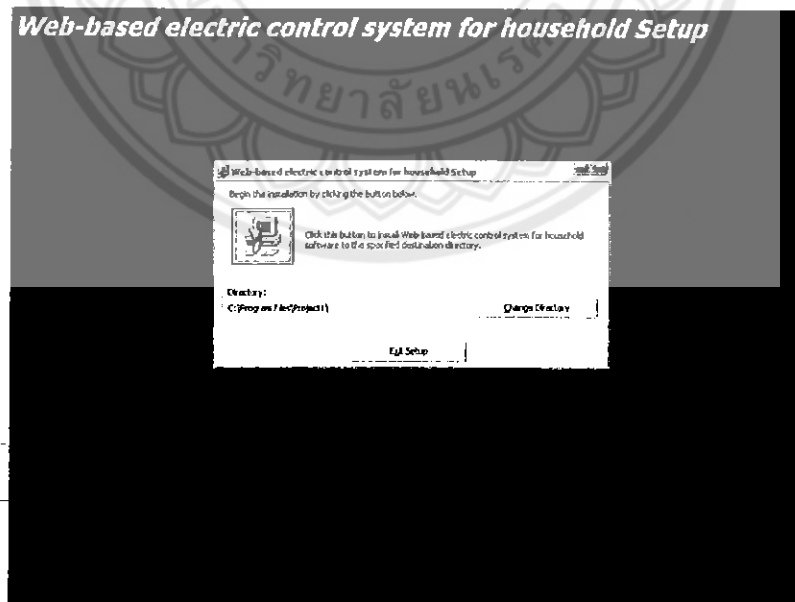


การติดตั้งโปรแกรม

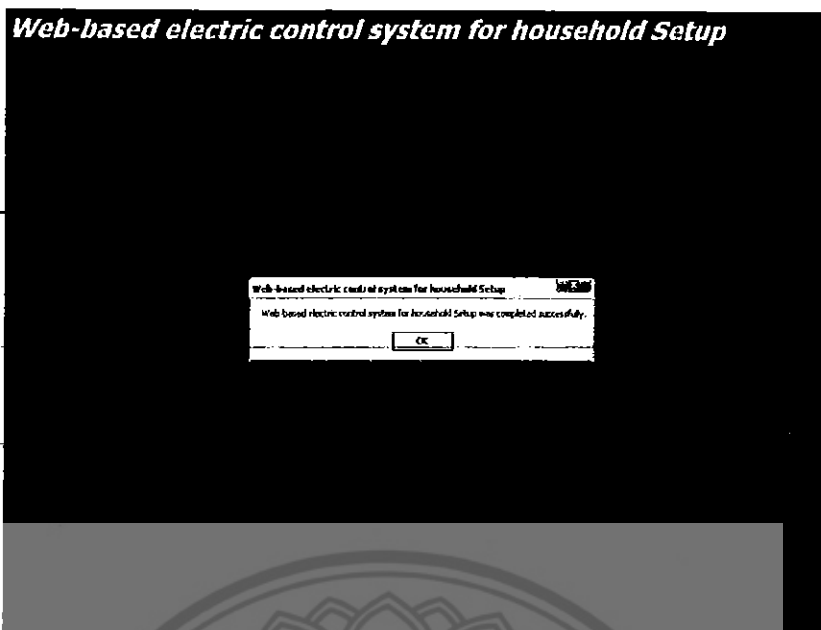
Double click ที่  จะแสดงดังรูป จากนั้น click ปุ่ม OK



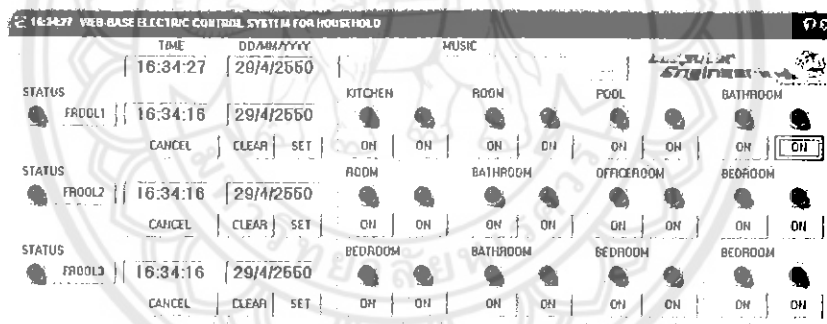
จากนั้นทำการ Click ที่  เพื่อดำเนินการติดตั้ง



แสดงการติดตั้งสมบูรณ์ จากนั้น Click ปุ่ม OK



การ Run โปรแกรมสามารถทำได้โดยการ Start/All Programs/Web-based electric control system for household/Web-based electric control system for household



การพัฒนาด้วยภาษา Assembly บนเครื่องคอมพิวเตอร์

1. Copy โปรแกรม SXA51 ลงเครื่องคอมพิวเตอร์ก่อน
2. เขียนโปรแกรม ลง notepad T55.txt จากนั้น save เป็น T55.asm
3. เข้า command prompt ไปยัง path ที่เก็บตัว SXA51 ในที่นี้ผมเก็บไว้ที่ C:\

```
C:\>SXA51
```

```
C:\>SXA51 T55.asm
```

โดยโปรแกรมจะทำการแปลให้ตามต้องการ โดยจะแสดงข้อแตกต่างๆ ให้เราทราบดังนี้

```
No errors detected
```

```
Object file size: 481 Bytes
```

```
Program entry address: 8000(Hex)
```

4. ในกรณีที่ต้องการดูผลการแปลที่เกิดขึ้น เทียบบรรทัดต่อบรรทัดตามโปรแกรมภาษา Assembly โดยการใส่ -L ดังนี้

```
C:\>SXA51 -L T55.asm
```

ซึ่งตัวแปลจะสร้างไฟล์ T55.LST ขึ้นมาซึ่งเป็น Text File ที่เราสามารถดูได้ตามต้องการ

5. สุดท้ายก็คือการนำไฟล์ T55.HEX ที่แปลได้ มาโหลดลงบอร์ด START-C51 โดยที่เครื่องคอมพิวเตอร์ จะต้องเตรียมให้พร้อมดังนี้

```
C:\>MODE COM1:19200, N, 8, 1
```

เป็นการกำหนดคุณสมบัติของพอร์ตอนุกรม COM1 ให้มีความเร็ว = 19200 No Parity มี

Date = 8 Bit และ Stop Bit = 1 การกำหนดด้วยคำสั่ง MODE นี้จะคงอยู่ตลอดไปจนกว่าจะปิด window ของการทำงานนี้จากนั้นก็ให้ใช้คำสั่ง Copy เพื่อส่งไฟล์ออกทางพอร์ตดังนี้

```
C:\>Copy T55.HEX COM1
```

ให้คีย์คำสั่งนี้โดยยังไม่กด Enter โดยเราจะไปเตรียมความพร้อมบนบอร์ด START-C51 ก่อน

ประวัติผู้เขียนโครงการ



ชื่อ นายพศิน ชุมณี

ภูมิลำเนา 110/1 ถนนประสาท ตำบลเวียง อำเภอเมือง จังหวัดพะเยา
ประวัติการศึกษา

- จบมัธยมศึกษาจาก โรงเรียนพะเยาพิทยาคม จังหวัดพะเยา
- ปัจจุบันกำลังศึกษาอยู่ระดับปริญญาตรีชั้นปีที่ 4

สาขาวิชาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์
มหาวิทยาลัยนเรศวร

E-mail: chokun_tum@hotmail.com



ชื่อ นางสาวศิริพรรณ ชันธิกุล

ภูมิลำเนา 151 หมู่ 5 ตำบลบ่อแก้ว อำเภอเมือง จังหวัดลำปาง
ประวัติการศึกษา

- จบมัธยมศึกษาจาก โรงเรียนเขลางค์นคร จังหวัดลำปาง
- ปัจจุบันกำลังศึกษาอยู่ระดับปริญญาตรีชั้นปีที่ 4

สาขาวิชาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์
มหาวิทยาลัยนเรศวร

E-mail: yuna_khwan@hotmail.com