



ภาควิชาวิศวกรรมศาสตร์วิศวกรรม
 คณะครุศาสตร์อุตสาหกรรม
 สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
 ใบรับรองปริญญาโท

ชื่อหัวข้อ ระบบเปิด-ปิดและรายงานสถานะสวิตช์ผ่านเครือข่ายอินเทอร์เน็ต
 On-Off Switch System and Status Report Via Internet

ชื่อนักศึกษา 1. นายโชคชัย สุสา รหัสประจำตัว 48035489
 2. นายศักดิ์ชัย รอดคง รหัสประจำตัว 48035517
 3. นายอภิญา แซ่ลิ้ม รหัสประจำตัว 48035528

หลักสูตร ครุศาสตร์อุตสาหกรรมบัณฑิต
 สาขาวิชา วิศวกรรมอิเล็กทรอนิกส์
 อาจารย์ที่ปรึกษา ผศ.กิตติพงศ์ มะโน
 อาจารย์ที่ปรึกษาร่วม อ.อำพล ทองระอา

คณะกรรมการสอบปริญญาโท	ลายมือชื่อ
1. อ.พิชญ์สินี มะโน	
2. ผศ.กิตติพงศ์ มะโน	
3. อ.อำพล ทองระอา	
4. อ.สุระชัย พิมพ์สาส์	
5. อ.ปิยะ ศุภวาราสุวัฒน์	

วัน/เดือน/ปีที่สอบ วันอาทิตย์ที่ 29 เดือนเมษายน พ.ศ. 2550 เวลา 10.00 น.

สถานที่สอบ ห้อง ค.310 คณะครุศาสตร์อุตสาหกรรม สจล.

ภาควิชารับรองแล้ว

ลงนาม.....

(รศ.สุรสิทธิ์ รัตรี)

หัวหน้าภาควิชาวิศวกรรมศาสตร์วิศวกรรม
 วันที่ 4 เดือน ม.ค. พ.ศ. 50



<BT492332>

ระบบเปิด-ปิดและรายงานสถานะสวิตช์ผ่านเครือข่ายอินเทอร์เน็ต การศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
 ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

ปริญญาานิพนธ์

ระบบเปิด - ปิดและรายงานสถานะสวิตช์ผ่านเครือข่ายอินเทอร์เน็ต

ON-OFF SWITCH SYSTEM AND STATUS REPORT VIA INTERNET



โชคชัย สุสา
ศักดิ์ชัย รอดคง
อภิญา แซ่ลิ้ม

21/11/2549

เลขหมู่.....
เลขทะเบียน **75169**
วัน,เดือน,ปี **24 ต.ค. 2550**

b. 4491310
i.

ปริญญาานิพนธ์ฉบับนี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรครุศาสตร์อุตสาหกรรมบัณฑิต
สาขาวิศวกรรมอิเล็กทรอนิกส์
ภาควิชาครุศาสตร์วิศวกรรม คณะครุศาสตร์อุตสาหกรรม
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
ปีการศึกษา 2549

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ปริญญาานิพนธ์

เรื่อง ระบบเปิด - ปิดและรายงานสถานะสวิทช์ผ่านเครือข่ายอินเทอร์เน็ต
On-Off Switch System and Status Report via Internet

วัตถุประสงค์

1. เพื่อศึกษาการติดต่อสื่อสารควบคุมผ่านระบบอินเทอร์เน็ต
2. เพื่อออกแบบระบบให้สามารถควบคุมผ่านระบบอินเทอร์เน็ต
3. เพื่อสร้างระบบ เปิด - ปิด และรายงานสถานะสวิทช์ผ่านระบบอินเทอร์เน็ต
4. เพื่อทดสอบระบบ เปิด - ปิด และรายงานสถานะสวิทช์ผ่านระบบอินเทอร์เน็ต
5. เพื่อประยุกต์ใช้งานระบบ เปิด - ปิด และรายงานสถานะสวิทช์ผ่านระบบอินเทอร์เน็ต

ประโยชน์ที่คาดว่าจะได้รับ

1. ได้ศึกษาหาความรู้ในการควบคุมผ่านระบบอินเทอร์เน็ต
2. ได้ออกแบบระบบเปิด - ปิดและรายงานสถานะของสวิทช์ผ่านระบบอินเทอร์เน็ต
3. ได้สร้างระบบเปิด - ปิดและรายงานสถานะของสวิทช์ผ่านระบบอินเทอร์เน็ต
4. ได้ทดสอบระบบเปิด - ปิดและรายงานสถานะของสวิทช์ผ่านระบบอินเทอร์เน็ต
5. ได้นำไปประยุกต์ใช้งานระบบเปิด - ปิดและรายงานสถานะของสวิทช์ผ่านระบบอินเทอร์เน็ต

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ชื่อหัวข้อ	ระบบเปิด - ปิดและรายงานสถานะสวิตช์ผ่านเครือข่ายอินเทอร์เน็ต	
นักศึกษา	นายโชคชัย	สุสา
	นายศักดิ์ชัย	รอดคง
	นายอภิญา	เชษฐ์
อาจารย์ที่ปรึกษา	ผศ.กิติพงษ์	มะโน
อาจารย์ที่ปรึกษาร่วม	อาจารย์อำพล	ทองระอา
หลักสูตร	ครุศาสตร์อุตสาหกรรมบัณฑิต	
สาขาวิชา	วิศวกรรมอิเล็กทรอนิกส์	
ปีการศึกษา	2549	

บทคัดย่อ

ปริญญาานิพนธ์ฉบับนี้นำเสนอการสร้างระบบเปิด - ปิด และรายงานสถานะสวิตช์ผ่านเครือข่ายอินเทอร์เน็ต เพื่อทดสอบการควบคุมอุปกรณ์ไฟฟ้า และกล้อง Web Cam ผ่านระบบอินเทอร์เน็ตซึ่งระบบประกอบด้วยบอร์ดไมโครคอนโทรลเลอร์ร่วมกับ MCS51AC2 และกล้อง Web Cam โดยสามารถควบคุมการเปิด - ปิดอุปกรณ์ไฟฟ้า 4 จุด จุดละไม่เกิน 1200 วัตต์ และสามารถแสดงภาพผ่านระบบอินเทอร์เน็ตได้จากผลการทดสอบการทำงานของระบบปรากฏว่าระบบสามารถควบคุมการเปิด - ปิดอุปกรณ์ไฟฟ้าได้ และสามารถแสดงผลของกล้อง Web Cam ผ่านระบบนี้ได้ตามวัตถุประสงค์ที่วางไว้

Thesis Title	On - Off Switch System and Status Report via Internet	
Student	Mr. Chokchai	Susa
	Mr. Sakchai	Rodkong
	Mr. Apinya	Saelim
Advisor	Assist.Prof. kitipong	Mano
Co-Advisor	Mr. Amphon	Thongra-ar
Education Level	Bachelor of Science in Industrial Education	
Program in	Electronics Engineering	
Academic Year	2006	

ABSTRACT

This thesis presents the system for On-Off Switch System and Status Report via Internet to test electric equipment controlling and camera web cam through internet that contains with electronic board module together with MCS51AC2 and web cam. This can control close & open system for electric equipment for 4 point, each point not more than 1200 watts and able to show picture on internet. From the result of this test, this system can control electronic switch and present the result from Web cam through this system as objective.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กิตติกรรมประกาศ

ปริญาานิพนธ์ฉบับนี้สามารถสำเร็จลุล่วงได้ดั่งนั้น เนื่องมาจากความร่วมมือร่วมใจของสมาชิกภายในกลุ่มทุกท่าน คณะผู้จัดทำขอขอบพระคุณผู้ช่วยศาสตราจารย์กิติพงศ์ มะโน อาจารย์อำพล ทองระอา และอาจารย์ประจำภาควิชาครุศาสตร์วิศวกรรมทุกท่านเป็นอย่างมากที่ได้กรุณาให้คำแนะนำในการแก้ไขปัญหาต่างๆ ตลอดจนจนถึงข้อมูลและอุปกรณ์ที่เป็นประโยชน์ต่อการทดลองโครงการ และในการจัดทำปริญาานิพนธ์ฉบับนี้ ขอขอบคุณห้องสมุดคณะครุศาสตร์อุตสาหกรรม ห้องสมุดคณะวิศวกรรมศาสตร์ และสำนักหอสมุดกลางที่ช่วยอำนวยความสะดวกและเอื้อเฟื้อสถานที่ในการค้นคว้าข้อมูล

ขอกราบขอบพระคุณบิดา มารดา และผู้มีพระคุณสำหรับพวกเราที่ได้ให้การสนับสนุนทุกสิ่งทุกอย่างทางด้านการศึกษาตลอดมาจนถึงปัจจุบัน และสุดท้ายต้องขอขอบพระคุณเพื่อนๆ ที่เป็นกำลังใจให้เสมอมา



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ

เรื่อง	หน้า
บทคัดย่อภาษาไทย	I
บทคัดย่อภาษาอังกฤษ	II
กิตติกรรมประกาศ	III
สารบัญ	IV
สารบัญตาราง	VIII
สารบัญรูป	IX
บทที่ 1 บทนำ	1
1.1 ความเป็นมาและความสำคัญของปัญหา	1
1.2 จุดมุ่งหมายของโครงการ	1
1.3 สมมติฐานของการจัดทำโครงการ	2
1.4 ขีดความสามารถของโครงการ	2
1.5 ขั้นตอนการทำโครงการ	2
1.6 เนื้อหาโดยสังเขป	2
บทที่ 2 ทฤษฎีและหลักการ	4
2.1 อินเทอร์เน็ต (Internet)	4
2.2 เวิลด์ไวด์เว็บ (World Wide Web)	5
2.2.1 ส่วนประกอบของ เวิลด์ไวด์เว็บ	5
2.3 Web Server และ Web Browser	6
2.4 โฮมเพจ	6
2.5 การรับส่งข้อมูลในอินเทอร์เน็ต	7
2.6 IP Address	8
2.7 ดีเอ็นเอส (Domain name System: DNS)	8
2.8 HTML (Hypertext Markup Language)	8
2.8.1 คำสั่งเริ่มต้นที่ใช้ในการสร้างโฮมเพจ	9
2.9 PHP (Personal Home Page)	11
2.9.1 ภาษา PHP คืออะไร	11
2.9.2 ความสามารถของภาษา PHP	11

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ(ต่อ)

เรื่อง	หน้า
2.9.3 การติดตั้งภาษา PHP	12
2.9.4 ความรู้เบื้องต้นเกี่ยวกับภาษา PHP	12
2.9.5 PHP กับการอัปโหลดไฟล์	14
2.10 สถาปัตยกรรมโปรโตคอล TCP/IP	19
2.11 การทำงานในแต่ละเลเยอร์ใน TCP/IP	20
2.11.1 Network Interface Layer	20
2.11.2 Internet Protocol (IP)	21
2.11.3 Transport Layer	21
2.11.4 Application Layer	23
2.12 พอร์ตอนุกรม (RS-232)	23
2.12.1 จังหวะเวลาของการสื่อสารข้อมูลอนุกรม	26
2.12.2 รูปแบบของการสื่อสารข้อมูลอนุกรม	27
2.12.3 การเชื่อมต่อพอร์ตอนุกรมมาตรฐาน RS-232	28
2.13 ไมโครคอนโทรลเลอร์ MCS-51	29
2.13.1 ไมโครคอนโทรลเลอร์ MCS-51	29
2.13.2 โครงสร้างของ MCS-51	29
2.13.3 การจัดการหน่วยความจำและการเชื่อมต่อ	30
2.13.4 การอ้างตำแหน่งข้อมูล	31
2.14 ไอซีเบอร์ RTL8019AS	31
2.14.1 คุณสมบัติของ RTL8019AS	31
2.14.2 รายละเอียดทั่วไปของไอซีเบอร์ RTL8019AS	32
บทที่ 3 การออกแบบ การสร้าง และการทำงาน	34
3.1 กล่าวนำ	34
3.2 การออกแบบและการสร้างแผนวงจรชุดควบคุม	34
3.3 การออกแบบและการสร้างระบบควบคุม	35
3.3.1 การเขียนโค้ดด้วยโปรแกรม Keil	35
3.3.2 การสร้าง Website	37
3.3.3 การสร้างระบบล็อกอิน	39

เอกสารนี้เป็นเอกสารลิขสิทธิ์ของมหาวิทยาลัยเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง เพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ(ต่อ)

เรื่อง	หน้า
3.3.4 การใช้งานโปรแกรม Windows Media Encoder	41
บทที่ 4 การทดลองและผลการทดลอง	47
4.1 กล่าวนำ	47
4.2 การควบคุมอุปกรณ์ผ่านระบบเน็ตเวิร์ก	47
4.2.1 การกำหนดค่าและการใช้งานของ Ethernet	47
4.2.2 อินพุตและเอาต์พุตของบอร์ดคอนโทรลเลอร์	48
4.2.3 สิ่งที่ต้องการในการพัฒนาบอร์ดคอนโทรลเลอร์	48
4.2.4 การเชื่อมต่อของไมโครคอนโทรลเลอร์กับไอซีอินเตอร์เน็ต	49
4.2.5 การกำหนดค่าทางอินเตอร์เน็ต	49
4.2.6 การกำหนดโปรโตคอล DATA PTOTOCOL	49
4.2.7 คำสั่งในการควบคุมการทำงานของ RELAY OUTPUTS	49
4.3 การทดลองการทำงานผ่านระบบอินเตอร์เน็ต	49
4.3.1 การควบคุมบอร์ด SAT5102	50
4.3.2 การติดต่อกับกล้อง Web Camera	51
4.3.3 การทดลองผ่านระบบอินเตอร์เน็ต	54
4.3.4 การทดสอบไฟล์ผ่านโปรแกรม Ws_ftp	56
บทที่ 5 บทสรุป	61
5.1 สรุป	61
5.2 ปัญหาและวิธีการแก้ไข	61
5.3 แนวทางการพัฒนา	61
บรรณานุกรม	63
ภาคผนวก ก เครื่องต้นแบบ	64
ภาคผนวก ข วงจรและแผ่นวงจรพิมพ์	67
ภาคผนวก ค รายการอุปกรณ์	71
ภาคผนวก ง รายละเอียดและคุณสมบัติของอุปกรณ์	74
ภาคผนวก จ ผังงาน	89
ภาคผนวก ฉ รหัสต้นฉบับของโปรแกรม	92
ภาคผนวก ช คู่มือการใช้งาน	107

สารบัญ(ต่อ)

เรื่อง
ประวัติผู้แต่ง

หน้า
112



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญตาราง

ตารางที่	หน้า
2.1 แสดงไดอะล็อกกำหนดคุณสมบัติของ Form Object ชนิด File Field	16
2.2 แสดงไดอะล็อกกำหนดคุณสมบัติของ Form Object ชนิด Button	16
2.3 การเปรียบเทียบของเลเยอร์ OSI และ TCP/IP	20
2.4 แสดงการจัดการขาของคอนเน็กเตอร์แบบ DB-9	25
ค.1 รายการของอุปกรณ์ของวงจรควบคุมระบบเปิด - ปิดและรายงานสถานะสวิตช์	72
ค.1 (ต่อ) รายการของอุปกรณ์ของวงจรควบคุมระบบเปิด - ปิดและรายงานสถานะสวิตช์	73
ค.3 รายการอุปกรณ์ของ Ethernet Module	73
ช.1 การแก้ปัญหาเบื้องต้น	110
ช.2 ข้อมูลจำเพาะ	111

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป

รูปที่	หน้า
2.1 ภาพรวมของเว็ลด์ไวด์เว็บ	5
2.2 แบบโครงสร้างพื้นฐานของไฟล์เอชทีเอ็มแอล	9
2.3 ส่วนเนื้อหาของโปรแกรม	10
2.4 การแทรกคำสั่งภาษา PHP	12
2.5 การวางคำสั่งในภาษา PHP ไว้ภายในเอกสาร HTML	13
2.6 ผลลัพธ์วางคำสั่งในภาษา PHP ไว้ภายในเอกสาร HTML	13
2.7 Comments ในภาษา PHP	13
2.8 แสดงการใช้คำสั่ง Variables	14
2.9 ผลลัพธ์จากการใช้คำสั่ง Variables	14
2.10 แสดงไดอะล็อกกำหนดคุณสมบัติของแท็ก <form>	15
2.11 แสดงไดอะล็อกกำหนดคุณสมบัติของ Form Object ชนิด File Field	16
2.12 แสดงไดอะล็อกกำหนดคุณสมบัติของ Form Object ชนิด Button	16
2.13 แสดงโค้ดของไฟล์ fileupload.php	17
2.14 แสดงการเขียนสคริปต์ PHP สำหรับการอัปโหลดไฟล์	18
2.15 แสดงการเขียนสคริปต์ PHP สำหรับการอัปโหลดไฟล์	18
2.16 แสดงการเขียนสคริปต์ PHP สำหรับการอัปโหลดไฟล์	18
2.17 แสดงการเขียนสคริปต์ PHP สำหรับการอัปโหลดไฟล์	19
2.18 แสดงการเขียนสคริปต์ PHP สำหรับการอัปโหลดไฟล์	19
2.19 แสดงคอนเน็กเตอร์อนุกรม 9 ขา หรือแบบ DB-9	24
2.20 การเชื่อมต่ออุปกรณ์ภายนอกกับพอร์ตอนุกรมของคอมพิวเตอร์ในลักษณะต่างๆ	25
2.21 แสดงการส่งข้อมูลแบบอนุกรมด้วยความเร็ว 9600 บิตต่อวินาที	27
2.22 แสดงการส่งข้อมูลขนาด 8 บิตแบบอนุกรมพร้อมด้วย บิตเริ่มต้น, บิตพาริตี, บิตหยุดด้วยความเร็ว 9600 บิตต่อวินาที	28
2.23 แสดงระดับแรงดันสัญญาณของพอร์ตอนุกรม RS-232 กับ TTL ในสถานะลอจิก "1" และ "0"	29
2.24 ไอซีเบอร์ RTL8019AS	33
3.1 บล็อกไดอะแกรมของระบบโดยรวม	34
3.2 แสดงรูปหน้าจอของโปรแกรม Keil	36

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป(ต่อ)

รูปที่	หน้า
3.3 แสดงการเขียนโปรแกรมคำสั่งที่ใช้งาน	37
3.4 แสดงการเขียนคำสั่งด้วยภาษา HTML	37
3.5 แสดงไฟล์เอกสาร HTML ผ่าน Web Browser	38
3.6 แสดงหน้าต่าง Output ควบคุมการทำงานผ่าน Web Browser	39
3.7 แสดง Flow chat ของระบบล็อกอิน	40
3.8 แสดงการกำหนด Properties ของฟอร์ม	40
3.9 แสดงการกำหนดค่าของโต๊ะล็อก Table	41
3.10 แสดงเว็บเพจไฟล์ login.php	41
3.11 รูปแสดงการเปิดโปรแกรม Windows Media Encoder	42
3.12 ทำการเลือก Broadcast a live event	43
3.13 แสดงการเลือกกล่อง	43
3.14 แสดงการเลือก Pull from the encoder	44
3.15 แสดงการกำหนด Port	44
3.16 แสดงการเลือกอัตราความเร็วว่าจะเอากี่ Kbps	45
3.17 แสดงการเก็บข้อมูลที่ต้องการบันทึก	45
3.18 แสดงหน้าจอการใช้งานของโปรแกรม	46
4.1 แสดงการเชื่อมต่อใช้งานของระบบโดยรวม	47
4.2 การเชื่อมต่ออุปกรณ์ในการทดลองการควบคุมบอร์ด SAT5102	50
4.3 วิธีการตั้งค่า IP ให้กับคอมพิวเตอร์	50
4.4 ใช้คำสั่ง ping เพื่อทดสอบการเชื่อมต่อบอร์ดและคอมพิวเตอร์ผ่านสาย LAN	51
4.5 แสดงการตั้งค่าของโปรแกรม Windows Media Encoder	51
4.6 แสดงการตั้งค่าโปรแกรม Windows Media Encoder	52
4.7 แสดงการตั้งค่าวีดีโอ	52
4.8 แสดงการตั้งค่าวีดีโอ	52
4.9 แสดงการเชื่อมต่อผ่านพอร์ตที่วางอยู่	53
4.10 แสดงการตั้งค่า Compression	53
4.11 แสดงการทดสอบการทำงานของกล่อง	53
4.12 การสร้างหน้าเว็บเพจด้วยโปรแกรม Macromedia Dreamweaver MX	54

สารบัญรูป(ต่อ)

รูปที่	หน้า
4.13 การล็อกอินเข้าสู่หน้าควบคุมการทำงาน	55
4.14 หน้าต่างแสดงการทำงานของ TCP/IP	55
4.15 หน้าต่างการควบคุมเอาต์พุต	55
4.16 การใช้งานโปรแกรม Ws_ftp	56
4.17 การใส่ชื่อผู้ใช้งาน	56
4.18 การใส่ชื่อใน Server Address	57
4.19 การ User Name และ Password	57
4.20 แสดง User Name และ Password เพื่อตรวจสอบความถูกต้อง	58
4.21 การเข้าสู่โปรแกรม Ws_ftp สำหรับการอัปโหลดไฟล์	58
4.22 แสดงหน้าต่างการล็อกอินบนหน้าเว็บเบราว์เซอร์	59
4.23 หน้าต่างควบคุมการสั่งงานบนหน้าเว็บเบราว์เซอร์	60
ก.1 แสดงภาพเครื่องต้นฉบับ	65
ก.2 แสดงภาพเครื่องต้นฉบับ	65
ก.3 แสดงภาพการประกอบอุปกรณ์ภายในเครื่องต้นฉบับ	66
ข.1 วงจรควบคุมระบบเปิด-ปิดและรายงานสถานะสวิทช์	68
ข.2 แผ่นวงจรพิมพ์ส่วนบนของระบบเปิด - ปิดและรายงานสถานะสวิทช์	69
ข.3 แผ่นวงจรพิมพ์ส่วนล่างของระบบเปิด - ปิดและรายงานสถานะสวิทช์	69
ข.4 ตำแหน่งการวางอุปกรณ์แผ่นพิมพ์ของระบบเปิด - ปิดและรายงานสถานะสวิทช์	70

บทที่ 1

บทนำ

1.1 ความเป็นมาและความสำคัญของปัญหา

เนื่องจากเจ้าของบ้านมีบ้านอยู่แต่ไม่มีเวลาที่จะได้ดูแลอยู่กับบ้านได้ตลอดเวลาบางครั้งออกไปข้างนอกออกไปค้างแรมที่อื่นหรือกลับมาดึกก็ไม่สามารถกลับมาดูแลความเรียบร้อยภายในบ้านได้อย่างเช่นในเวลาค่าคืน ภายในบ้านจะต้องมีการเปิดไฟเพื่อให้ตัวบ้านนั้นมองแล้วให้น่าอยู่น่าเกรงขามจากบุคคลภายนอกหรือจากมิจฉาชีพต่างๆ และเมื่อเปิดในช่วงค่ำแล้วก็สามารถสั่งปิดได้ในช่วงเวลากลางวันจึงได้มีการนำระบบเปิด - ปิดไฟผ่านระบบอินเทอร์เน็ตระบบนี้มาใช้ในชีวิตประจำวัน

ระบบเปิด - ปิดและรายงานสถานะสวิตช์ผ่านระบบอินเทอร์เน็ต ระบบจะทำงานแบบเครือข่าย (Network) จะเห็นได้ว่าระบบเครือข่ายซึ่งเป็นทางผ่านของข้อมูลระหว่างศูนย์ควบคุมหลักกับวงจรเป็นส่วนที่มีความสำคัญต่อระบบโดยรวมเป็นอย่างยิ่งถ้าหากระบบเครือข่ายขาดความปลอดภัยของข้อมูล หรือขาดการระบบจัดการข้อมูลที่ระบบโดยรวมก็ถือว่าไม่มีประสิทธิภาพเช่นกัน โครงการระบบเปิด - ปิดและรายงานสถานะสวิตช์ผ่านระบบอินเทอร์เน็ต เป็นโครงการเพื่อพัฒนาระบบเครือข่ายสำหรับการควบคุมและตรวจสอบข้อมูลสำหรับเปิด - ปิด และรายงานสถานะของสวิตช์ผ่านเครือข่ายอินเทอร์เน็ต ซึ่งเป็นการนำไมโครคอมพิวเตอร์และไมโครคอนโทรลเลอร์มาทำหน้าที่ในการรับส่งข้อมูลผ่านเครือข่ายแบบอนุกรมอ้างอิงมาตรฐาน RS-232 โดยเป็นระบบเครือข่ายที่มีคุณสมบัติคือมีความรวดเร็วและมีประสิทธิภาพในการรับส่งข้อมูล

1.2 จุดมุ่งหมายของโครงการ

เนื่องจากในปัจจุบันมีการนำไมโครคอนโทรลเลอร์มาใช้ในงานควบคุมและรับส่งข้อมูลจากอุปกรณ์ต่างๆ เป็นจำนวนมากซึ่งส่วนใหญ่ระบบเครือข่ายและกระบวนการรับส่งข้อมูลจะถูกออกแบบโดยปราศจากการรักษาความปลอดภัยของข้อมูลบนเครือข่าย และขาดการออกแบบกระบวนการรับ-ส่งข้อมูล (Protocol) ที่ดี ดังนั้นวัตถุประสงค์ของการศึกษาเพื่อ

1. เพื่อศึกษาการติดต่อสื่อสารควบคุมผ่านระบบอินเทอร์เน็ต
2. เพื่อออกแบบระบบให้สามารถควบคุมผ่านระบบอินเทอร์เน็ต
3. เพื่อสร้างระบบ เปิด - ปิด และรายงานสถานะสวิตช์ผ่านระบบอินเทอร์เน็ต
4. เพื่อทดสอบระบบ เปิด - ปิด และรายงานสถานะสวิตช์ผ่านระบบอินเทอร์เน็ต
5. เพื่อประยุกต์ใช้งานระบบ เปิด - ปิด และรายงานสถานะสวิตช์ผ่านระบบอินเทอร์เน็ต

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1.3 สมมุติฐานของการจัดทำโครงการ

เมื่อผ่านการศึกษาในโครงการนี้แล้วผู้ศึกษาสามารถมีความรู้เกี่ยวกับการติดต่อสื่อสารผ่านอินเทอร์เน็ต ที่นำไปใช้ควบคุมอุปกรณ์ภายนอกแบบต่างๆ เช่นในโครงการนี้ได้ใช้ควบคุมสวิทช์และรายงานสถานะของสวิทช์ เพื่อเปิด - ปิดหลอดไฟฟ้าจนสามารถนำไปใช้งานจริงได้โดยโครงการนี้จะผ่านการประเมินจากผู้ทรงคุณวุฒิในระดับการประเมินดีขึ้น

1.4 ขีดความสามารถของโครงการ

โครงการนี้มีขีดความสามารถดังนี้

1. เป็นโปรแกรมประยุกต์ที่ทำงานผ่าน Web-based application
2. สามารถเปิด - ปิดสวิทช์ 4 จุดผ่านระบบอินเทอร์เน็ต
3. สามารถเปิด - ปิดหลอดไฟฟ้า 40 วัตต์ 4 จุด ผ่านระบบอินเทอร์เน็ต
4. สามารถแสดงการทำงานของอุปกรณ์ปลายทางได้ 1 ช่อง
5. สามารถแสดงภาพการทำงานของหลอดไฟภายในบ้านโดยผ่านกล้อง Webcam ได้

1.5 ขั้นตอนการทำโครงการ

งานวิจัยนี้เริ่มต้นด้วยการศึกษาองค์ประกอบต่างๆ ของระบบเปิด - ปิดและรายงานสถานะสวิทช์ผ่านระบบอินเทอร์เน็ต ได้แก่การต่อวงจรฮาร์ดแวร์การศึกษากระบวนการทำงานของโปรแกรมที่เขียนไว้รวมถึงกระบวนการรับส่งข้อมูล และรูปแบบการทำงานด้วย

เมื่อได้โครงสร้างของอุปกรณ์ฮาร์ดแวร์แล้วต่อไปจึงเป็นการศึกษาการเขียนโปรแกรมสำหรับไมโครคอนโทรลเลอร์ MCS - 51 โดยใช้ภาษา C ซึ่งทำให้ง่ายต่อการพัฒนาโปรแกรมมีการเขียนควบคุมการรับ - ส่งข้อมูลจะทำการทดลองบนมาตรฐานแบบ RS232 เมื่อมีการรับส่งข้อมูลเป็นไปตามที่ออกแบบไว้แล้วจึงทำการเพิ่มกระบวนการทำงานในส่วนของการแสดงภาพการทำงานโดยผ่านกล้อง Webcam

1.6 เนื้อหาโดยสังเขป

เนื้อหาในปฏิญานีพจน์ฉบับนี้แบ่งออกเป็นบทต่างๆ เพื่อสะดวกต่อการศึกษา และทำความเข้าใจในแต่ละบทจะประกอบด้วยเนื้อหาดังต่อไปนี้

บทที่ 1 กล่าวถึงความเป็นมาและความสำคัญของปฏิญานีพจน์ ขีดความสามารถของโครงการ และเนื้อหาในบทต่างๆ โดยสังเขป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 2 ประกอบด้วยทฤษฎีต่างๆ เกี่ยวกับโครงสร้างของระบบเปิด - ปิดและรายงานสถานะของ สวิตช์ผ่านเครือข่ายอินเทอร์เน็ต การรับส่งข้อมูลในอินเทอร์เน็ต สถาปัตยกรรมของ TCP/IP พื้นฐานของ ASP HTML พอร์ตอนุกรม (RS232) และไมโครคอลโทรลเลอร์

บทที่ 3 กล่าวถึงเนื้อหาเกี่ยวกับแผนผังการทำงานของโครงการ ผังวงจรต่างๆ ที่ใช้ในโครงการ ตลอดจนการออกแบบและการสร้างส่วนประกอบต่างๆ เช่นวงจรการทำงานของกการเปิด - ปิดสวิตช์ไฟ การแสดงภาพของกล้องวิดีโอ การทำงานด้าน Server และการทำงานในส่วนของเว็บเบราว์เซอร์

บทที่ 4 ประกอบด้วยการทดลองและผลการทดลองของระบบเปิด - ปิดสวิตช์ผ่านเครือข่าย อินเทอร์เน็ต การแสดงภาพการทำงานผ่านกล้องวิดีโอ และการทำงานในส่วนของเว็บเบราว์เซอร์

บทที่ 5 เป็นการสรุปผลการจัดทำโครงการ ปัญหาที่เกิดขึ้นและแนวทางในการแก้ไข รวมทั้งแนวทาง ในการพัฒนา

ภาคผนวก ก เครื่องต้นแบบ

ภาคผนวก ข วงจรและแผ่นวงจรพิมพ์

ภาคผนวก ค รายการอุปกรณ์

ภาคผนวก ง รายละเอียดและคุณสมบัติของอุปกรณ์

ภาคผนวก จ ผังงาน

ภาคผนวก ฉ รหัสต้นฉบับของโปรแกรม

ภาคผนวก ช คู่มือการใช้งาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 2

ทฤษฎีและหลักการ

2.1 อินเทอร์เน็ต (Internet)

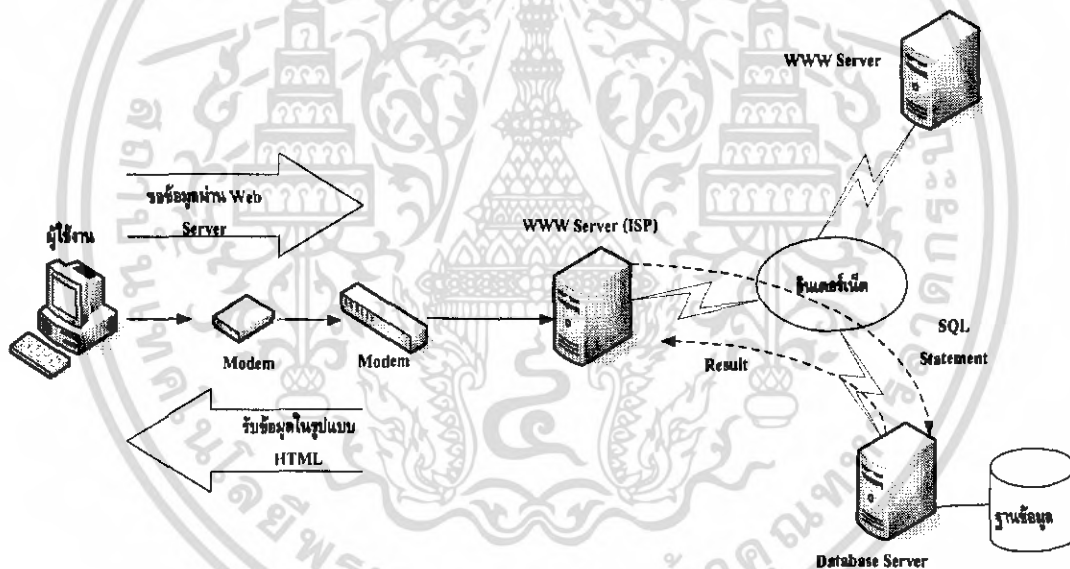
อินเทอร์เน็ต คือ เครือข่ายของเครื่องคอมพิวเตอร์ระบบต่างๆ ที่เชื่อมโยงกันมาจากคำว่า Inter Connection Network อินเทอร์เน็ตเป็นระบบเครือข่ายคอมพิวเตอร์ที่มีขนาดใหญ่ เครื่องคอมพิวเตอร์ทุกเครื่องทั่วโลกสามารถเชื่อมต่อถึงกันได้โดยใช้มาตรฐานในการรับส่งข้อมูลที่เป็นหนึ่งเดียว หรือที่เรียกว่า โพรโทคอล (Protocol) ซึ่งโพรโทคอลที่ใช้ในระบบอินเทอร์เน็ตมีชื่อเรียกว่า ทีซีพี/ไอพี (TCP/IP: Transmission Control Protocol/Internet Protocol) ลักษณะของระบบอินเทอร์เน็ตเป็นเสมือนใยแมงมุมที่ครอบคลุมทั่วโลกในแต่ละจุดที่เชื่อมต่ออินเทอร์เน็ตนั้นสามารถสื่อสารกันได้หลายเส้นทางตามความต้องการ โดยไม่กำหนดตายตัวและไม่จำเป็นต้องไปตามเส้นทางโดยตรงอาจจะผ่านจุดอื่นๆ หรือเลือกไปเส้นทางอื่นได้หลายๆ เส้นทาง การติดต่อสื่อสารผ่านระบบเครือข่ายอินเทอร์เน็ตนั้นอาจเรียกว่า การติดต่อสื่อสารแบบไร้มิติ หรือ Cyberspace

ในการใช้งานเครือข่ายคอมพิวเตอร์ทั่วไปหรือในเครือข่ายอินเทอร์เน็ตก็ตามจะมีการส่งผ่านข้อมูลไปมาระหว่างเครื่องคอมพิวเตอร์ภายในเครือข่ายหรือข้ามเครือข่ายออกไป ระบบคอมพิวเตอร์ที่เชื่อมต่ออยู่ในเครือข่ายแต่ละเครือข่ายอาจจะใช้ฮาร์ดแวร์และซอฟต์แวร์ที่เหมือนกันหรือแตกต่างกัน ดังนั้นการจะส่งผ่านข้อมูลถึงกันและตีความได้อย่างถูกต้องจะต้องทำข้อกำหนดร่วมกันในการสื่อสารเสียก่อนเรียกว่าจะต้องกำหนดระเบียบวิธีในการติดต่อให้ตรงกันเปรียบเสมือนการติดต่อสื่อสารกันของมนุษย์เรา ถ้าเราต้องติดต่อกับผู้คนที่ต่างเชื้อชาติต่างภาษาให้อย่างเข้าใจได้ถูกต้องตรงกันก็จะต้องตกลงกันเสียก่อนว่าจะติดต่อสื่อสารกันได้อย่างไรด้วยภาษาใดถึงจะเข้าใจกันได้เช่น ปัจจุบันภาษาอังกฤษถูกใช้เป็นภาษากลางในการติดต่อสื่อสารกันมาก ทำให้เราพูดได้ว่าภาษาอังกฤษเป็นโพรโทคอลการติดต่อสื่อสารของมนุษย์ที่ใช้งานอย่างแพร่หลายเช่นเดียวกันกับโพรโทคอล TCP/IP เป็นโพรโทคอลหลักที่ใช้ในการสื่อสารข้อมูลในเครือข่ายอินเทอร์เน็ต

TCP/IP (Transmission Control Protocol/Internet Protocol) เป็นชุดของโพรโทคอลที่ถูกใช้ในการสื่อสารผ่านเครือข่ายอินเทอร์เน็ตโดยมีวัตถุประสงค์เพื่อให้สามารถใช้สื่อสารจากต้นทางข้ามเครือข่ายไปยังปลายทางได้ และสามารถหาเส้นทางที่จะส่งข้อมูลไปได้เองโดยอัตโนมัติถึงแม้ว่าในระหว่างทางอาจจะผ่านเครือข่ายที่มีปัญหา โพรโทคอลก็ยังคงหาเส้นทางอื่นในการส่งผ่านข้อมูลไปให้ถึงปลายทางได้ ชุดโพรโทคอลนี้ได้รับการพัฒนามาตั้งแต่ปี 1960 ซึ่งถูกใช้เป็นครั้งแรกในเครือข่าย ARPANET ซึ่งต่อมาได้ขยายการเชื่อมต่อไปทั่วโลกเป็นเครือข่ายอินเทอร์เน็ต ทำให้ TCP/IP เป็นที่ยอมรับอย่างกว้างขวางจนถึงปัจจุบัน

2.2 เวิลด์ไวด์เว็บ (World Wide Web)

เป็นการบริการรูปแบบหนึ่งในอินเทอร์เน็ตที่เชื่อแน่ว่าเราทุกคนต้องได้เคยสัมผัส หรือที่เราเรียกกันสั้นๆ ว่าเว็บ (Web) นั่นเอง การให้บริการในรูปแบบนี้คือการเรียก Browser เช่น IE, Netscape จากเครื่องของเราและทำการระบุ URL เพื่อใช้ในการอ้างที่อยู่ที่เกิดเว็บโดยชื่อ URL ที่เราป้อนจะถูกส่งไปที่ DNS (Domain Name Server) ซึ่งเป็นเซิร์ฟเวอร์ที่ใช้ในการเปลี่ยนชื่อ URL (เช่น www.yahoo.com เป็นต้น) ให้กลายเป็นชื่อแบบตัวเลขหรือไอพีแอดเดรส (IP Address) โดยหมายเลขไอพีนั้นจะถูกใช้ในการอ้างถึงตำแหน่งเครื่องคอมพิวเตอร์ในอินเทอร์เน็ต โดยเครื่องทุกเครื่องที่อยู่ในอินเทอร์เน็ตจะมีหมายเลขไอพีไม่เหมือนกันทำให้เราสามารถระบุที่อยู่ของเครื่องที่เก็บข้อมูลเว็บที่เราต้องการเปิดดูได้ โดยเราอาจจะเปรียบเทียบหมายเลขไอพีเหมือนกับเลขที่บ้านของเราในการส่งจดหมายก็ได้ จากนั้นเมื่อเราได้เลขไอพีที่ต้องการแล้วคำร้องขอเว็บของเราก็จะถูกส่งไปยังเครื่องเซิร์ฟเวอร์มาแสดงผลที่เครื่องเราเท่านี้เราก็สามารถเปิดดูเว็บได้แล้ว



รูปที่ 2.1 ภาพรวมของเวิลด์ไวด์เว็บ

2.2.1 ส่วนประกอบของ เวิลด์ ไวด์ เว็บ

การจะใช้บริการ เวิลด์ ไวด์ เว็บ ได้นั้นจำเป็นต้องมีส่วนประกอบดังนี้

2.2.1.1 แหล่งข้อมูลหรือเว็บไซต์

เว็บไซต์หรือเว็บเซิร์ฟเวอร์ คือระบบคอมพิวเตอร์ที่เป็นแหล่งเก็บเว็บเพจที่ผู้ใช้บริการสามารถเรียกดูเว็บเพจที่เก็บอยู่ในเว็บไซต์นั้นได้ ซึ่งเครื่องคอมพิวเตอร์ที่ทำหน้าที่เป็นเว็บไซต์อาจจะใช้ระบบปฏิบัติการยูนิกซ์ หรือวินโดวส์เอ็นที (Windows NT) ก็ได้ และจะต้องมีการจัดการติดตั้งโปรแกรมจัดการที่จะทำให้เครื่องเอกสารเป็นเอกสารทสงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

คอมพิวเตอร์เครื่องนั้นทำหน้าที่เป็นเว็บไซต์ตัวอย่างเช่น ในกรณีที่เครื่องคอมพิวเตอร์นั้นใช้ระบบปฏิบัติการวินโดวส์เอ็นทีจะมีซอฟต์แวร์เว็บเซิร์ฟเวอร์ให้เลือกใช้เช่น โปรแกรม Purveyou HTTP Server ของสถาบัน European Microsoft Windows NT Academic Center (EMWAC) โปรแกรมเว็บไซต์ของบริษัท O'Reilly and Associate หรือถ้าใช้ระบบปฏิบัติการยูนิกซ์ก็อาจใช้เว็บเซิร์ฟเวอร์เป็น NSCA http, Apache เป็นต้น

ผู้ที่เป็นเจ้าของเว็บไซต์จะจัดสร้างเว็บเพจของตนเก็บไว้ที่เว็บไซต์นั้น เพื่อให้ผู้ใช้คนอื่นๆ ทั่วโลกสามารถเข้ามาดูข้อมูลที่เก็บไว้ในเว็บไซต์นั้นได้ตัวอย่างเช่น เว็บเพจของสำนักข่าว CNN จะเก็บอยู่ที่เว็บไซต์ www.cnn.com เป็นต้น

2.2.1.2 โปรแกรมเว็บเบราว์เซอร์

เป็นโปรแกรมหรือซอฟต์แวร์ที่ใช้ในการเข้าสู่ เวิลด์ ไวด์ เว็บ จะเปิดดูเว็บเพจที่เก็บอยู่ในเว็บไซต์ใดๆ ตัวอย่างของโปรแกรมเว็บเบราว์เซอร์ ที่มีให้ดาวน์โหลดใช้กันได้ฟรี ได้แก่ Netscape Navigator จากบริษัท Netscape Communication และ โปรแกรม Microsoft Internet Explorer จากบริษัท ไมโครซอฟต์ เป็นต้น ซึ่งโปรแกรมเว็บเบราว์เซอร์ส่วนใหญ่จะทำงานร่วมกับโปรแกรมวินโดวส์ และนอกจากจะใช้เพื่อดูเว็บเพจจากเว็บไซต์ใดๆ แล้ว หลายโปรแกรมยังมีความสามารถอื่นๆ เช่น บริการส่งจดหมายอิเล็กทรอนิกส์ การค้นหาข้อมูล การโอนถ่ายโปรแกรมด้วย FTP เป็นต้น

2.3 Web Server และ Web Browser

ศัพท์อีก 2 คำที่เราควรทำความเข้าใจกันในตอนนี้คือ Web Server และ Web Browser

Web Server เป็นโปรแกรมที่อยู่ในเครื่องเซิร์ฟเวอร์ทำหน้าที่ในการรับคำร้องขอ และทำการประมวลผลแล้วส่งไปที่เครื่องฝั่งไคลเอนต์ หรือจะเรียกว่าเป็นโปรแกรมที่ใช้ในการให้บริการเว็บนั่นเอง

Web Browser เป็นโปรแกรมที่อยู่เครื่องฝั่งไคลเอนต์ มีหน้าที่ในการส่งข้อมูลในการร้องขอเว็บ และจะนำเสนอข้อมูลเว็บโดยตัว เว็บเบราว์เซอร์จะมีความเข้าใจในภาษามาตรฐานของเว็บก็คือ ภาษา HTML และจะสามารถแปลงภาษา HTML ให้เป็นหน้าเว็บหรือหน้าเอกสารให้เราได้ใช้งาน

2.4 โฮมเพจ

โฮมเพจ (Home Page) จะหมายถึงหน้าแรกของเว็บเพจทั้งหมดที่มีผู้ใช้บริการอินเทอร์เน็ตจะพบเมื่อมีการเข้าไปยังเว็บไซต์ใดๆ โฮมเพจเปรียบเสมือนสารบัญ และคำนำที่เจ้าของเว็บไซต์สร้างขึ้นเพื่อจะใช้ประชาสัมพันธ์องค์กรของตนว่าให้บริการในสิ่งใดบ้าง นอกจากนี้แล้วภายในโฮมเพจหนึ่งๆ ก็อาจมีการเชื่อมกับเว็บเพจอื่นๆ เป็นจำนวนมากๆ ได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.5 การรับส่งข้อมูลในอินเทอร์เน็ต

ตัวอย่างต่อไปนี้ แสดงรายละเอียดการรับและการส่งข้อมูลระหว่างไคลเอนต์กับเซิร์ฟเวอร์ เมื่อไคลเอนต์พิมพ์ <http://www.test.com> ก็จะเกิดการทํางานในขั้นตอนต่างๆ ดังนี้

1. เมื่อไคลเอนต์พิมพ์ที่อยู่ URL ในที่นี้คือ <http://www.test.com> บราวเซอร์ก็จะทํากาการ request HTTP พร้อมกับผ่านค่าชื่อที่อยู่ปลายทางไปยังซอฟต์แวร์ TCP
2. ซอฟต์แวร์ TCP ก็จะแปลงชื่อปลายทางไปยังที่อยู่ไอพี (IP address) โดนส่งการคิวรี (Query) ไปยังตัวเซิร์ฟเวอร์ที่แกชื่อ URL ซึ่งเป็นเซิร์ฟเวอร์ DNS
3. จะเริ่มสร้างการติดต่อแบบเสมือนกับเซิร์ฟเวอร์ปลายทาง โดยมีการตรวจสอบค่าระยะเวลาที่กำหนดและการรับทราบ Handshaking ในแต่ละแพ็กเกจเพื่อให้แน่ใจได้ว่าการโอนย้ายข้อมูลระหว่างไคลเอนต์กับเซิร์ฟเวอร์มีความเชื่อถือได้ในกรณีที่มีข้อมูลใหญ่มาก TCP จะแบ่งข้อมูลออกเป็นแพ็กเกจย่อยๆ แล้วเพิ่มส่วนหัวที่มีหมายเลขพอร์ตและอื่นๆ ให้แก่แพ็กเกจย่อยๆ เหล่านั้น เช่น ที่อยู่ปลายทางและข้อมูลสัญญาณนาฬิกาเพื่อ synchronize กับข้อมูล
4. TCP จะส่งแพ็กเกจไปยังเลเยอร์ไอพีในเลเยอร์จะเพิ่มส่วนหัวของไอพีซึ่งจะเป็นที่อยู่ของไอพีทั้งไคลเอนต์และเซิร์ฟเวอร์และอื่นๆ ให้กับข้อมูลและส่งต่อไปยัง Ethernet
5. ไดรเวอร์ Ethernet จะเพิ่มส่วนอื่นๆ อีกและส่งแพ็กเกจไปยัง router โดยขึ้นอยู่กับตารางเส้นทางภายในจากนั้นตัว router จะหาเส้นทางหรือเน็ตเวิร์คที่มีประสิทธิภาพสูงสุด เพื่อส่งแพ็กเกจออกไปยังเซิร์ฟเวอร์ปลายทาง
6. เมื่อถึงเซิร์ฟเวอร์แล้ว เลเยอร์ TCP ของเซิร์ฟเวอร์ปลายทางจะส่ง Acknowledgment ไปยังไคลเอนต์ว่าได้รับแพ็กเกจแล้ว ถ้าเซิร์ฟเวอร์ปลายทางไม่ได้รับในระยะเวลาที่กำหนดไว้ในไคลเอนต์จะเกิดขึ้น มีผลให้ไคลเอนต์ต้องส่งไปอีกรอบเลเยอร์ TCP/IP ของเซิร์ฟเวอร์ปลายทาง จะต่อส่วนหัว TCP/IP จากแต่ละแพ็กเกจให้เป็นข้อมูล ซึ่งจะเหมือนกับไคลเอนต์
7. TCP จะทํากาการส่ง Request ไปยัง HTTP เซิร์ฟเวอร์ ซึ่งจะกระจายค่าออกมา เมื่อกระจายค่าเรียบร้อยแล้วเซิร์ฟเวอร์จะส่งไฟล์ HTML พร้อมกับผ่านไฟล์ข้อมูลไปยังเลเยอร์ TCP/IP แล้วแพ็กส่งไปยังปลายทางกลับไปยังไคลเอนต์ (โดยใช้วิธีเดียวกับที่ไคลเอนต์ส่งไปยังเซิร์ฟเวอร์)
8. ท้ายสุดแล้วแพ็กเกจข้อมูลทั้งหมดจะถูกต่อใหม่เมื่อถึงฝั่งไคลเอนต์ และบราวเซอร์ก็จะกระจายข้อมูลแล้วแสดงบนหน้าจอไคลเอนต์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.6 IP Address

เนื่องจากในระบบเครือข่ายอินเทอร์เน็ตจะใช้โปรโตคอล TCP/IP เป็นมาตรฐานในการสื่อสารข้อมูล ซึ่งจะมีการกำหนดหมายเลขประจำตัวที่ไม่ซ้ำกันให้กับคอมพิวเตอร์ทุกเครื่องที่มีการเชื่อมกันอยู่ในระบบ เครือข่ายหมายเลขประจำตัวนี้จะถูกเรียกว่า IP Address หรือหมายเลข IP โดยมีรูปแบบเป็นชุดของตัวเลข 4 ชุดที่คั่นด้วยเครื่องหมายจุด เช่น 202.44.192.43 ตัวเลขในแต่ละชุดจะมีขนาด 8 บิตดังนั้นในแต่ละชุดจะมีค่าได้ตั้งแต่ 0 ถึง $2^8 - 1 = 255$ เท่านั้น ซึ่งจะทำให้สามารถกำหนดหมายเลข IP ให้กับเครื่องคอมพิวเตอร์ได้ทั้งหมดถึงสี่พันล้านเลขหมายที่ไม่ซ้ำกันเลย

ดังนั้นเมื่อมีการติดตั้งเครื่องคอมพิวเตอร์ที่เป็นโฮสต์คอมพิวเตอร์เพื่อเชื่อมต่อเข้ากับอินเทอร์เน็ต และให้บริการต่างๆ จำเป็นต้องขอหมายเลข IP ประจำหน่วยงาน Internet Network Information Center (InterNIC) ขององค์กร Network Solution Incorporated (NSI) ที่รัฐเวอร์จิเนีย สหรัฐอเมริกา แต่ถ้าผู้ใช้สมัครเป็นสมาชิกกับหน่วยงานที่ทำหน้าที่เป็นผู้ให้บริการอินเทอร์เน็ต (Internet Service Provider: ISP) ในประเทศไทยมีอยู่หลายหน่วยงาน ก็ไม่ต้องขอหมายเลข IP ให้แก่ผู้ใช้เอง

2.7 ดีเอ็นเอส (Domain Name System: DNS)

การใช้หมายเลข IP ซึ่งเป็นตัวเลขล้วนๆ ในการอ้างอิงถึงคอมพิวเตอร์แต่ละเครื่องนั้น จะมีข้อเสียคือ จำได้ยาก และก่อให้เกิดความสับสนได้ง่ายจึงมีการพัฒนาวิธีการอ้างอิงถึงหมายเลข IP แบบใหม่ที่เรียกว่า Domain Name System (DNS) ขึ้นมา

ดีเอ็นเอส เป็นเทคนิคการเปลี่ยนหมายเลข IP ที่เป็นตัวเลขให้เป็นตัวอักษรแทนหมายเลข IP เป็น 202.12.97.1 ผู้ให้บริการสามารถเขียนเป็นชื่อโดเมนคือ kku.ac.th แทนเครื่องคอมพิวเตอร์ที่เรียกว่า ดีเอ็นเอส จะทำหน้าที่ในการแปลงจากชื่อโดเมนเนมให้เป็นหมายเลข IP อีกทีหนึ่ง

2.8 HTML (Hypertext Markup Language)

HTML เป็นรูปแบบมาตรฐานในการสร้างหน้าเว็บโดยที่ภาษา HTML เป็นภาษาที่ง่ายต่อการเรียนรู้ และเข้าใจ โดยเราสามารถใช้เวลาสั้นๆ ในการสร้างเว็บเพจโดยใช้ HTML ได้โดยมีพื้นฐานอยู่ในรูปแท็ก <...> โดยมีลักษณะการใช้แท็ก 2 แบบดังนี้

1. แท็กมีจุดเริ่มต้นและสิ้นสุด
2. แท็กเดี่ยวๆ

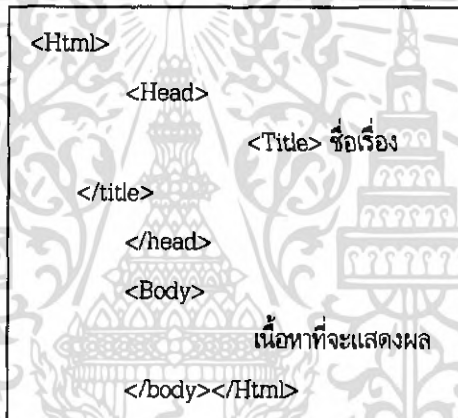
ตัวอย่างแท็ก HTML ที่จะต้องมีจุดเริ่มต้นและจุดสิ้นสุด เช่น และ ซึ่งเป็นการทำให้ตัวอักษรเป็นตัวหนา ถ้าเราเขียน ไว้ต้นเอกสาร และไม่ได้ปิดด้วย แล้วทั้งเอกสารจะกลายเป็นตัวหนาทั้งหมดเพราะไม่มี ปิดเพื่อให้รู้ว่าตรงไหนคือการสิ้นสุดการแสดงผลตัวหนานั้นเอง และโค้ดเอกสารนี้เป็นเอกสารที่ส่งวนเวียนสำหรับการใช้งานเพื่อการศึกษาเท่านั้น เมื่ออนุญาตให้นำไปใช้โดยไม่ว่ากรรมใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

HTML ที่เป็นคำสั่งเดียวๆ ก็คือ เขียนโค้ดไปแล้ว ไม่ต้องเขียนจุดสิ้นสุด เช่น `<hr>` จะเป็นการสร้างเส้นกันแนวนอน หรือ `<p>` จะเป็นการขึ้นย่อหน้าใหม่เป็นต้น

ในการเขียน HTML มีหลักการดังนี้

1. เริ่มต้นไฟล์จะต้องเขียนแท็ก `<html>` และท้ายไฟล์จะต้องปิดด้วย `</html>`
2. ต่อจาก `<html>` จะต้องเป็น `<head>...</head>` ซึ่งภายในแท็ก `<head>` จะใส่แท็กที่เป็นชื่อเรื่อง ซึ่งเป็นคำสั่งย่อยอีก 1 คำสั่งคือ `<title>...</title>`
3. ต่อจาก `<head>...</head>` แล้วจะเป็นส่วนของ `<body>...</body>` เราจะเขียนสิ่งที่เราต้องการจะนำเสนอลงไประหว่างแท็ก `<body>...</body>` นี้

รูปแบบโครงสร้างพื้นฐานของไฟล์เอชทีเอ็มแอล



รูปที่ 2.2 แบบโครงสร้างพื้นฐานของไฟล์เอชทีเอ็มแอล

2.8.1 คำสั่งเริ่มต้นที่ใช้ในการสร้างโฮมเพจ

คำสั่ง (Tags) ที่ใช้ในภาษาเอชทีเอ็มแอลจะประกอบไปด้วยเครื่องหมายน้อยกว่า "<" ตามด้วยชื่อคำสั่ง และปิดท้ายด้วยเครื่องหมายมากกว่า ">" เป็นส่วนที่ทำหน้าที่แสดงผลข้อมูลต่างๆ พื้นฐานหลักโดยทั่วไปคำสั่งของเอชทีเอ็มแอลส่วนใหญ่จะเป็นคู่ จะมีแค่บางคำสั่งที่มีรูปแบบอยู่เพียงตัวเดียว โดยตัวเปิดของคำสั่งจะอยู่ในเครื่องหมาย "<>" และตัวปิดคำสั่งจะอยู่ในเครื่องหมาย "</>" ตัวอย่างคำสั่ง

1. หมายเหตุ

รูปแบบคำสั่ง `<!>`

การใช้คำสั่งหมายเหตุในโปรแกรมภาษาเอชทีเอ็มแอล มีประโยชน์สำหรับแจ้งให้ผู้อ่านได้ทราบว่าเรากำลังทำอะไรอยู่ หรือรายละเอียดต่างๆ ที่เราต้องการจะบอกเกี่ยวกับโปรแกรมของเราซึ่งข้อความที่เขียนจะปรากฏอยู่แต่เฉพาะภายในโปรแกรมเอชทีเอ็มแอลจะไม่ปรากฏบนจอภาพโปรแกรมเว็บเบราว์เซอร์

เอกสารนี้เป็นเอกสารสงวนลิขสิทธิ์สำหรับการศึกษานานาชาติเท่านั้น เมื่อผู้เยี่ยมชมเว็บไซต์หรือใช้เอกสารนี้โดยไม่ได้รับอนุญาตจากเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้ ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2. คำสั่งเริ่มต้น

รูปแบบคำสั่ง `<Html>...</html>`

คำสั่ง `<Html>` เป็นคำสั่งที่เริ่มต้นในการเขียนโปรแกรมเอชทีเอ็มแอล และคำสั่ง `</html>` จะเป็นคำสั่งแสดงว่าสิ้นสุดโปรแกรมเอชทีเอ็มแอล

3. ส่วนหัวของโปรแกรม

รูปแบบคำสั่ง `<Head>...</head>`

คำสั่ง `<Head>` เป็นคำสั่งที่ใช้กำหนดข้อความในส่วนที่เป็นชื่อเรื่องของไฟล์เอชทีเอ็มแอล และภายในคำสั่งจะมีคำสั่งย่อยอีกคำสั่งหนึ่งคือ คำสั่ง `<title>` คำสั่งปิดคือ `</title>` และ `</head>`

4. กำหนดข้อความในส่วนหัวของโปรแกรมหรือไตเติลบาร์

รูปแบบคำสั่ง `<Title>...</title>`

คำสั่งข้างต้นจะเป็นส่วนที่แสดงชื่อของเอกสาร หรือชื่อของไฟล์เอชทีเอ็มแอล ซึ่งข้อความภายในคำสั่งจะปรากฏหรือแสดงผลในส่วนของไตเติลบาร์ (Title Bar) ของโปรแกรมเว็บเบราว์เซอร์ แต่จะไม่แสดงในส่วนของการแสดงผลของโปรแกรมเว็บเบราว์เซอร์

5. ส่วนเนื้อหาของโปรแกรม

รูปแบบคำสั่ง `<Body>...</body>`

คำสั่งในส่วนเนื้อหาของโปรแกรมนี้เป็นคำสั่งที่สำคัญในส่วนการแสดงผลในเว็บเบราว์เซอร์ ซึ่งอาจจะประกอบไปด้วยตัวอักษรและรูปภาพการฟิกต่างๆ ที่อยู่ในเว็บเพจของเรา ตัวอย่างโปรแกรม

```
<Html>
  <! Comment>
    <Head>
      <Title>test</title>
    </head>
    <Body>
      Create the
      example homepage
    </body>
</html>
```

รูปที่ 2.3 ส่วนเนื้อหาของโปรแกรม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.9. PHP (Personal Home Page)

2.9.1. ภาษา PHP คืออะไร

PHP คือภาษา script อย่างหนึ่งที่เป็น server-side script ซึ่งจะทำงานในฝั่ง server แล้วส่งการแสดงผลมายัง browser ของตัว Client และนอกจากนี้ มันยังเป็น script ที่ embedded บน HTML อีกด้วย

PHP เป็นภาษาคำพวก script language คำสั่งต่างๆ จะเก็บอยู่ในไฟล์ที่เรียกว่า สคริปต์ (script) และเวลาใช้งานต้องอาศัยตัวแปรชุดคำสั่ง ตัวอย่างของภาษาสคริปต์เช่น JavaScript, Perl เป็นต้น ลักษณะของ PHP ที่แตกต่างจากภาษาสคริปต์แบบอื่นๆ คือ PHP ได้รับการพัฒนาและออกแบบมา เพื่อใช้งานในการสร้างเอกสารแบบ HTML โดยสามารถสอดแทรกหรือแก้ไขเนื้อหาโดยอัตโนมัติ ดังนั้นจึงกล่าวว่า PHP เป็นภาษาที่เรียกว่า server-side หรือ HTML-embedded Scripting language เป็นเครื่องมือที่สำคัญชนิดหนึ่งซึ่งช่วยให้เราสามารถสร้างเอกสารแบบ Dynamic HTML ได้อย่างมีประสิทธิภาพมากขึ้น

PHP ได้รับการเผยแพร่เป็นครั้งแรกในปี ค.ศ. 1994 โดยคุณ Rasmus Lerdorf ต่อมาได้มีโปรแกรมเมอร์เข้ามาช่วยในการพัฒนาต่อมาตามลำดับ เป็นเวอร์ชันต่างๆ จนกระทั่งถึงเวอร์ชันล่าสุดซึ่งเป็นเวอร์ชัน 5 นักพัฒนาสำคัญของเวอร์ชัน 4 และ เวอร์ชัน 5 คือคุณ Zeev Suraski และคุณ Andi Gutmans ในขณะนี้เว็บไซต์เวอร์ประมาณ 16 ล้านโดเมน (domains) ที่ใช้ PHP เราสามารถตรวจสอบจำนวนของ domains ที่ใช้ PHP ได้ที่ <http://www.php.net/usage.php>

ในตอนแรก PHP ย่อมาจาก Personal Home Page แต่ต่อมาได้เปลี่ยนชื่อให้ตรงตามกฎเกณฑ์ของ GNU ชื่อในปัจจุบันของ PHP นั้นย่อมาจาก Hypertext Preprocessor รายละเอียดต่างๆ ของ PHP เราสามารถเข้าไปค้นหาเพิ่มเติมได้ที่เว็บไซต์ของ PHP ซึ่งคือ <http://www.php.net>

2.9.2 ความสามารถของภาษา PHP

1. ภาษาอื่นที่ทำหน้าที่คล้ายๆ กับภาษา PHP คือ Perl, Microsoft Active Server Pages (ASP), Java Server Page (JSP), และ Allaire ColdFusion
2. ถ้าเปรียบเทียบภาษา PHP กับภาษาอื่นๆ เหล่านี้เราจะพบว่าภาษา PHP มีข้อได้เปรียบหลายอย่างดังต่อไปนี้
3. มีสมรรถนะสูงสามารถรองรับการใช้หลายล้าน hits ในแต่ละวัน
4. สามารถติดต่อกับหลายประเภทของฐานข้อมูลอย่างเช่น MySQL, PostgreSQL, mSQL, Oracle, Informix, Sybase และสามารถใช้อื่นๆ Open Database Connectivity Standard (ODBC) เพื่อติดต่อกับผลิตภัณฑ์ฐานข้อมูลของ Microsoft
5. ไม่ต้องเสียค่าใช้จ่ายในการใช้เราสามารถดาวน์โหลด PHP ได้จาก <http://www.php.net> โดยไม่ต้องเสียค่าใช้จ่ายใดๆ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

6. เรียนรู้และใช้งาน โดยเฉพาะถ้าเรารู้ภาษา C, C++, Perl, และ Java อยู่แล้ว
7. สามารถใช้ PHP ได้บนหลายระบบปฏิบัติการโดยไม่ต้องเปลี่ยนโปรแกรม

2.9.3 การติดตั้งภาษา PHP

ในที่นี้การทดลองทำบนเครื่อง Windows XP Professional

1. Download AppServ2.5.2 จาก <http://gear.kku.ac.th/krunapon/178375/tools.html>
2. ทำการติดตั้ง AppServ โดยการกดปุ่ม Next ไปเรื่อยๆ จนกระทั่งการติดตั้งเสร็จเรียบร้อย
3. ทดสอบการติดตั้งโดยการเปิดเว็บเบราว์เซอร์และให้ระบุ URL ที่ <http://localhost> เราควรจะได้หน้าเว็บเพจดังแสดงในรูปที่ 2.7

หลังจากที่เราติดตั้งเสร็จเรียบร้อยแล้วจะพบว่ามีไดเรกทอรี AppServ ซึ่งเป็นไดเรกทอรีที่มีไดเรกทอรี www อยู่ข้างในไฟล์ PHP ที่เราสร้างเราควรเก็บไว้ได้ไดเรกทอรี c:/AppServ/www

2.9.4 ความรู้เบื้องต้นเกี่ยวกับภาษา PHP

2.9.4.1 การแทรกคำสั่งภาษา PHP ในเอกสาร HTML

เราสามารถแทรกคำสั่งภาษา PHP ได้โดยการใช้ PHP tags ซึ่งมีไว้เพื่อเป็นการบ่งบอกให้รู้ส่วนที่เป็นคำสั่งของภาษา PHP ที่อยู่ในเอกสาร HTML การใช้ PHP tags นั้นสามารถทำได้หลายรูปแบบดังต่อไปนี้

```
<? Php ...?> (XML style)
<? ... ?> (short style)
<script language='php'>...</script> (Script style)
<% ... %> (ASP style)
```

รูปที่ 2.4 การแทรกคำสั่งภาษา PHP

แบบที่ควรใช้คือ XML style เนื่องจากสามารถทำงานได้กับทุกเซิร์ฟเวอร์ (Server) อีกทั้งสอดคล้องกับไวยากรณ์ของภาษา PHP

เราสามารถวางคำสั่งในภาษา PHP ไว้ภายในเอกสาร HTML ตามที่ต้องการได้โดยที่ PHP tags อาจจะถูกวางอยู่สลับกับ HTML tags ดังตัวอย่างเช่น

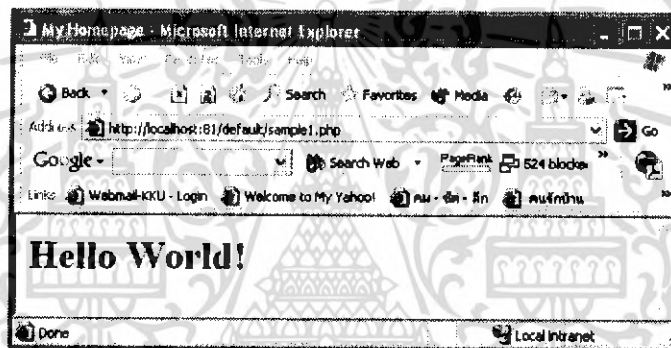
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

<html>
<head>
<title>My Homepage</title>
</head>
<body>
<h1><?php echo "Hello World!"; ?></h1>
</body>
</html>

```

รูปที่ 2.5 การวางคำสั่งในภาษา PHP ไว้ภายในเอกสาร HTML



รูปที่ 2.6 ผลลัพธ์วางคำสั่งในภาษา PHP ไว้ภายในเอกสาร HTML

2.9.4.2 รูปแบบภาษา PHP และคำสั่ง

2.9.4.2.1 Comments

Comments ในภาษา PHP เอารูปแบบมาจาก comments ในภาษา C++ และภาษา XML Comments มีไว้เพื่อเป็นบันทึกช่วยให้ผู้เขียนหรือผู้อ่านโปรแกรมเข้าใจโปรแกรม ตัวอย่างของ comments

```

<! -- File: hello.php (comments in xml style) -->

// this style of comment is suitable for a short comment that is not over one line
/*
* to write comments more than one line, you may want to use this style
*/

```

รูปที่ 2.7 Comments ในภาษา PHP

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

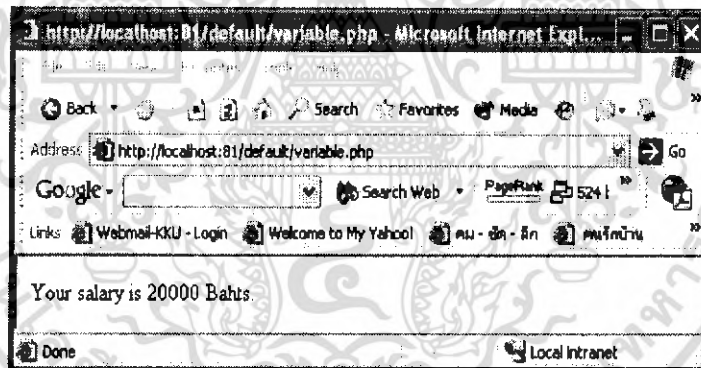
2.9.4.2.2 Variables (ตัวแปร)

Variables หรือ ตัวแปร คือ สัญลักษณ์ที่ใช้แทนข้อมูล Literals คือข้อมูลซึ่งสามารถเป็นค่าของตัวแปร \$salary เป็นตัวแปร 20,000 เป็น Literal วิธีการกำหนดตัวแปรในภาษา PHP จะต้องขึ้นต้นด้วยเครื่องหมาย \$ แล้วตามด้วยชื่อของตัวแปร ตัวอย่างการใช้ตัวแปร

```
Example 2: variable.php
<?php
    $salary = 20000;
    echo "Your salary is $salary Bahts.";
?>
```

ผลลัพธ์ที่ได้จากการใช้โปรแกรมข้างล่างนี้

รูปที่ 2.8 แสดงการใช้คำสั่ง Variables



รูปที่ 2.9 ผลลัพธ์จากการใช้คำสั่ง Variables

ถ้าหากว่าคุณไปที่ "View" และ คลิกที่ "Source" คุณจะพบว่าเนื้อหาของ file "variable.php" เป็น "Your salary is 20000 Bahts" ไม่มีคำสั่งของภาษา PHP ซึ่งหมายความว่า Server ได้รับและแปลคำสั่งของคุณเรียบร้อยแล้วไม่ว่า Browsers ใดก็จะเห็นเนื้อหาของ file นี้เหมือนกันหมด

2.9.5 PHP กับการอัปโหลดไฟล์

การอัปโหลดไฟล์ก็คือ การส่งไฟล์จากเครื่องคอมพิวเตอร์ Client ไปยังเครื่องคอมพิวเตอร์ Server สำหรับภาษา PHP นั้นสามารถเขียนโค้ดอัปโหลดได้เลยโดยเราไม่ต้องติดตั้งโมดูลใดๆ เพิ่มเติมเราสามารถสร้างเว็บเพจสำหรับการอัปโหลดไฟล์ได้อย่างง่ายดาย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การสร้างเว็บเพจเพื่ออัปโหลดไฟล์แบ่งการทำงานได้เป็นขั้นตอนดังต่อไปนี้

- การสร้างฟอร์มสำหรับอัปโหลดไฟล์
- การเขียนสคริปต์ PHP สำหรับการอัปโหลดไฟล์
- การทดสอบระบบดาวน์โหลด

2.9.5.1 การสร้างฟอร์มสำหรับอัปโหลดไฟล์

การสร้างระบบอัปโหลดนั้น Dreamweaver MX จะช่วยได้ในการสร้างฟอร์มสำหรับอัปโหลดส่วนได้สำหรับจัดการไฟล์จำเป็นจะต้องสวิตช์ไปที่โหมดโค้ด (Code View) เพื่อเขียนโค้ดเอง


ในขั้นตอนแรกจะเป็นการใช้ Dreamweaver MX แทรกฟอร์มลงในเว็บเพจสำหรับให้ผู้ใช้สามารถคลิกปุ่ม Browse เลือกไฟล์ที่ต้องการอัปโหลดนอกจากนั้นก็มีการแทรกปุ่ม Upload สำหรับคลิกเพื่อเริ่มต้นการอัปโหลดไฟล์

2.9.5.2 ขั้นตอนการสร้างฟอร์มสำหรับอัปโหลดไฟล์

1. เปิดโปรแกรม Dreamweaver MX คลิกเมนู File > New จะปรากฏไดอะล็อก New Document ขึ้นมาให้คลิกแท็บ General เลือก Dynamic Page > PHP หลังจากนั้นคลิกปุ่ม Create บันทึกเป็นไฟล์ fileupload.php
2. แทรกฟอร์มลงในเว็บเพจโดยคลิกที่เมนู Insert > Form
3. กำหนดคุณสมบัติของฟอร์มโดยการคลิกเมาส์วางเคอร์เซอร์ให้อยู่ภายในฟอร์มซึ่งแสดงขอบเขตด้วยเส้นประสีแดง หลังจากนั้นคลิกเลือกแท็บ <Form> ที่ Tag selector บริเวณขอบล่างซ้ายของหน้าจอคลิกที่เมนู Window > Properties เมื่อปรากฏไดอะล็อก Properties ขึ้นมาให้กำหนดคุณสมบัติของฟอร์มดังต่อไปนี้



รูปที่ 2.10 แสดงไดอะล็อกกำหนดคุณสมบัติของแท็ก <form>

หลังจากการแทรกฟอร์มและกำหนดค่าต่างๆ เสร็จแล้วให้คลิกที่ปุ่ม Show Code View  แล้วสองสังเกตดูโค้ดที่ Dreamweaver MX สร้างขึ้นมาจะเห็นว่าในแท็บ <Form> จะปรากฏค่า attribute enctype="multipart/form-data" ซึ่งจะต้องใช้เสมอในฟอร์มที่มีการเรียกใช้ Form Object ชนิด File Field

4. แทรก Form Object ชนิด File Field ลงภายในฟอร์มหลังจากนั้นให้กำหนดคุณสมบัติของ File Field โดยการคลิกที่เมนู Window>Properties เมื่อปรากฏไดอะล็อก Properties ขึ้นมาให้กำหนดชื่อในช่อง File Field Name เป็น "File" ดังรายละเอียดด้านล่างนี้

ตารางที่ 2.1 แสดงไดอะล็อกกำหนดคุณสมบัติของ Form Object ชนิด File Field

ค่าของไดอะล็อก	ค่าที่กำหนด
File Field Name	File
CharWidth	50

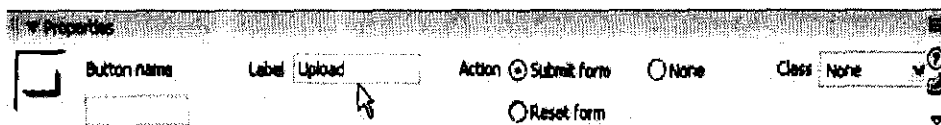


รูปที่ 2.11 แสดงไดอะล็อกกำหนดคุณสมบัติของ Form Object ชนิด File Field

5. แทรก Form Object ชนิด Button ลงภายในฟอร์มหลังจากนั้นให้กำหนดคุณสมบัติของ Button โดยการคลิกที่เมนู Window>Properties เมื่อปรากฏไดอะล็อก Properties ขึ้นมาให้กำหนดคุณสมบัติของ Button ดังต่อไปนี้

ตารางที่ 2.2 แสดงไดอะล็อกกำหนดคุณสมบัติของ Form Object ชนิด Button

ค่าของไดอะล็อก	ค่าที่กำหนด
Label	Upload
Action	Submit form



รูปที่ 2.12 แสดงไดอะล็อกกำหนดคุณสมบัติของ Form Object ชนิด Button

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

6. คลิกที่ปุ่ม Show Code View แล้วลองสังเกตโค้ดที่ Dreamweaver MX สร้างขึ้นมา เห็นว่ามีโค้ดการแทรก Form Object ชนิด File Field และ Button ลงในฟอร์มดังรูปด้านล่าง

```
<html>
<head>
<title></ title>
<meta http-equiv="Content-Type" content="text/html; charset=windows-874">
</head>
<body>
<form action="upload.php" method="POST" enctype="multipart/ form-data" name="fmupload">
<input name="File" type="file" value="" size="50">
<input name="" type="submit" value="upload">
</ form>
</body>
</html>
```


รูปที่ 2.13 แสดงโค้ดของไฟล์ fileupload.php

7. บันทึกไฟล์ fileupload.php

2.9.5.3 การเขียนสคริปต์ PHP สำหรับการอัปโหลดไฟล์

ขั้นตอนต่อไปนี้จะเป็นการเขียนโค้ด PHP สำหรับจัดการไฟล์ที่ถูกอัปโหลดขึ้นมาบนเซิร์ฟเวอร์เข้าไปในไดเรกทอรีหรือโฟลเดอร์ที่กำหนดไว้

ขั้นตอนการเขียนสคริปต์ PHP สำหรับการอัปโหลดไฟล์

1. สร้างโฟลเดอร์สำหรับจัดเก็บไฟล์โดยคลิกที่เมนู Site > Site Files จะปรากฏแถบ Files ที่ด้านขวาของหน้าจอหลังจากนั้นชี้เมาส์ที่ตำแหน่งบรรทัดแรกของแถบ Files คลิกเมาส์ปุ่มขวา จะปรากฏเมนูให้คลิกเลือก New Folder และกำหนดชื่อโฟลเดอร์เป็น "Upload Files"
2. เปิดโปรแกรม Dreamweaver MX คลิกเมนู File > New จะปรากฏไดอะล็อก New Document ขึ้นมาให้คลิก แท็บ General เลือก Dynamic Page > PHP หลังจากนั้นคลิกปุ่ม Create บันทึกเป็นไฟล์ upload.php
3. คลิกที่ปุ่ม Show Code View  หลังจากนั้นให้พิมพ์โค้ดต่อไปนี้ลงในเว็บเพจ

```
<?php
// upload full path (Detect path automatically)
If (!defined('UPLOADDIR')) define('UPLOADDIR',(dirname(__FILE__).'/ Upload Files'));
```

อธิบาย

เปิดแท็ก: <?php กำหนดชื่อโฟลเดอร์สำหรับจัดเก็บไฟล์สำหรับจัดเก็บไฟล์ที่อัปโหลดขึ้นมาในที่นี้ กำหนดให้เป็นโฟลเดอร์ชื่อ "Upload Files" โฟลเดอร์นี้ขึ้นมาภายใต้โฟลเดอร์ของระบบไฟล์ของเซิร์ฟเวอร์และต้องมี permission ที่สามารถเขียนไฟล์ลงได้เช่น หากเป็นระบบปฏิบัติการ linux ให้กำหนด permission เป็น 777 ในที่นี้จะจัดเก็บชื่อโฟลเดอร์ไว้ในค่าตัวแปรค่าคงที่ที่ "UPLOADDIR"

รูปที่ 2.14 แสดงการเขียนสคริปต์ PHP สำหรับการอัปโหลดไฟล์

```
<?php
//checks if file uploaded
If (is_uploaded_file($_FILES['File']['tmp_name']))
```

อธิบาย

เรียกใช้ฟังก์ชัน is_uploaded_file() ของPHP สำหรับตรวจสอบว่ามีการอัปโหลดไฟล์หรือไม่โดยตรวจสอบจากไฟล์ที่ถูกเก็บไว้ชั่วคราว \$_FILES['File']['tmp_name'] ฟังก์ชันนี้จะถูกใช้ไ้ในการตรวจสอบข้อผิดพลาดด้วยกล่าวคือหากมีการอัปโหลดไฟล์จะส่งสถานะกลับเป็น TRUE

รูปที่ 2.15 แสดงการเขียนสคริปต์ PHP สำหรับการอัปโหลดไฟล์

```
$File_tmpname = $_FILES["File"]["tmp_name"];
$File_name = $_FILES["File"]["name"];
$File_type = $_FILES["File"] ["type"];
$File_extension = substr($File_type,(strpos($File_type,"")+1));
$File_size = $_FILES["File"]["size"];
```

อธิบาย

\$_FILES["File"]["tmp_name"] ตัวแปรแบบที่เกิดจากการอัปโหลดเก็บชื่อไฟล์ชั่วคราวบนเซิร์ฟเวอร์

\$_FILES["File"]["name"] เป็นตัวแปรที่เกิดจากการอัปโหลดเก็บชื่อไฟล์จริงที่อัปโหลดขึ้นมา

\$_FILES["File"] ["type"] ตัวแปรแบบที่เกิดจากการอัปโหลดเก็บชนิดของไฟล์ที่อัปโหลดขึ้นมา

เอกสารนี้เป็นเอกสารที่สง **รูปที่ 2.16** แสดงการเขียนสคริปต์ PHP สำหรับการอัปโหลดไฟล์ ำใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
//checks if file moved successful
If ( move_uploaded_file($File_tmpname, (UPLOADDIR . "/" . $File_tmpname)) ){
    echo "<br> File uploading successful.";
}
else{
    echo "<br> File uploading failure.";
}
```

อธิบาย:

หากมีการอัปโหลดไฟล์จะย้ายไฟล์จากไฟล์ชั่วคราว (Temporary) ไปที่โฟลเดอร์ที่จัดเก็บไฟล์โดยเรียกใช้ฟังก์ชัน `move_uploaded_file()`

เมื่อการย้ายไฟล์สำเร็จจะแสดงข้อความ **"File uploading successful."**

เมื่อย้ายไฟล์ด้วยฟังก์ชัน `move_uploaded_file()` ไม่สำเร็จจะแสดงข้อความ **"File uploading failure."**

รูปที่ 2.17 แสดงการเขียนสคริปต์ PHP สำหรับการอัปโหลดไฟล์

```
}
else { // else checks if file uploaded
    echo "<br> No File selected for uploading.";
} //end checks if file uploaded
```

อธิบาย:

หากไม่มีการอัปโหลดไฟล์จะแสดงข้อความ No File for uploading."

รวมทั้งการปิดเครื่องหมายปีกกาเพื่อเป็นการจบ else และ if

รูปที่ 2.18 แสดงการเขียนสคริปต์ PHP สำหรับการอัปโหลดไฟล์

4. บันทึกไฟล์ upload.php

2.10 สถาปัตยกรรมโปรโตคอล TCP/IP

โปรโตคอล (Protocol) คือ ระเบียบ กฎ และกำหนดต่างๆ รวมถึงมาตรฐานที่ใช้ เพื่อให้ตัวรับและตัวส่งสามารถดำเนินการติดต่อสื่อสารได้สำเร็จ

TCP/IP เป็นชุดโปรโตคอลที่ออกแบบมาสำหรับการเชื่อมโยงระหว่างเน็ตเวิร์กขนาดใหญ่เชื่อมต่อกับเครือข่าย WAN (Wide Area Network) TCP/IP พัฒนาขึ้นในปี ค.ศ.1969 โดยหน่วยงานของรัฐบาล
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.11.2 Internet Protocol (IP)

มีหน้าที่รับผิดชอบการระบุที่อยู่ การทำแพ็คเกจ และฟังก์ชันต่างๆ โดยมีโปรโตคอลแกนหลักของ Internet Layer ดังนี้

1. Internet Protocol (IP) ทำหน้าที่ให้บริการส่งผ่านข้อมูลที่มาจาก Host-to-Host Layer เพื่อส่งข้ามไปยังเครือข่ายใดๆ ได้อย่างถูกต้องแม้ว่าจะมีเครือข่ายเชื่อมต่อกันอยู่ในอินเทอร์เน็ตเป็นล้านๆ เครือข่ายก็ตามเนื่องจากโปรโตคอลไอพีมีข้อมูลตำแหน่งไอพีปลายทางที่จะส่งข้อมูลไปให้โดยทำงานร่วมกับอุปกรณ์ Router เพื่อส่งข้อมูลข้ามเครือข่ายออกไปได้ ตัวโปรโตคอลไอพีจะทำงานแบบแพ็คเกจ Switching คือมีการส่งข้อมูลผ่านสวิตช์ไปยังปลายทาง ตัววงจรผ่านหรือ Switch นี้อาจเป็นเกตเวย์ หรือ Router ในระบบเครือข่ายก็ได้ซึ่งในข้อมูลของโปรโตคอลไอพีจะมีข้อมูลของหมายเลขไอพีปลายทางที่จะส่งข้อมูลไปและเมื่อถึงเครือข่ายปลายทางแล้ว จะมีกลไกแปลงหมายเลขไอพีให้เป็นหมายเลขฮาร์ดแวร์ประจำเครื่องที่ถูกต้องอีกทีหนึ่งด้วยโปรโตคอล
2. Address Resolution Protocol (ARP) รับผิดชอบการแก้ปัญหาที่อยู่ IP
3. Internet Control Message Protocol (ICMP) หน้าที่หลักของโปรโตคอล ICMP คือการแจ้ง หรือแสดงข้อความจากระบบ เพื่อบอกให้ผู้ใช้ทราบว่าจะเกิดอะไรขึ้นในการส่งผ่านข้อมูลนั้น ซึ่งปัญหาที่พบส่วนมากคือข้อมูลส่งไปไม่ได้ หรือปลายทางรับข้อมูลไม่ได้ เป็นต้น นอกจากนี้โปรโตคอล ICMP ยังถูกเรียกใช้งานจากจากเครื่องเซิร์ฟเวอร์ และ Router อีกด้วย เพื่อแลกเปลี่ยนข้อมูลที่ใช้ควบคุม ส่วนรูปแบบการทำงานของโปรโตคอล ICMP นั้นจะทำความคู่กับโปรโตคอลไอพีในระดับเดียวกัน และข้อความต่างๆ ที่แจ้งให้ทราบ จะถูกผนึกอยู่ภายในข้อมูลของไอพีอีกทีหนึ่ง โปรโตคอล TCP และ UDP อาศัยโปรโตคอลที่อยู่ชั้นล่างเพื่อส่งข้อมูลระหว่างเครื่อง และในชั้น Internet work Protocol ยังมีโปรโตคอล ICMP ทำหน้าที่ส่งข้อความแจ้งเตือนและโปรโตคอล ARP ทำหน้าที่แปลงเลขหมายไอพี ไปเป็นเลขหมายฮาร์ดแวร์จริง
4. Internet Group Management Protocol (IGMP) รับผิดชอบการจัดการของกลุ่มลิตีคลาส IP

2.11.3 Transport Layer

หรือเรียกอีกอย่างว่า Host-to-Host Transport Layer มีหน้าที่รับผิดชอบสำหรับการให้บริการที่มีความเชื่อถือได้ และมีประสิทธิภาพ โดยมีโปรโตคอลที่เป็นแกนหลัก คือ โปรโตคอล TCP และ UDP

TCP (Transmission Control Protocol) โปรโตคอลที่ซีพีเป็นโปรโตคอลที่มีการรับส่งข้อมูลแบบ Stream Oriented Protocol หมายความว่า การรับส่งข้อมูลจะไม่คำนึงถึงปริมาณข้อมูลที่จะส่งไป แต่จะแบ่ง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ข้อมูลเป็นส่วนย่อยๆ ก่อน แล้วจึงส่งไปยังปลายทางต่อเนื่องเป็นลำดับข้อมูล ในกรณีที่มีข้อมูลส่วนใดส่วนหนึ่งสูญหายไปก็จะส่งข้อมูลส่วนนั้นใหม่อีกครั้ง สำหรับปลายทาง ก็จะทำหน้าที่จัดเรียงส่วนของข้อมูลหน่วยใหม่ให้ต่อเนื่องกันและประกอบกลับเป็นข้อมูลทั้งหมดได้ซึ่งแยกข้อมูลที่ไม่ถูกต้องออก ดังนั้นแอปพลิเคชัน หรือโปรเซสใดที่อาศัยการส่งผ่านข้อมูลด้วยโปรโตคอลทีซีพี จะต้องให้หน่วยความจำ และช่องสัญญาณ (Bandwidth) มากกว่ายูดีพี

การติดต่อระหว่างกันจะต้องเป็นแบบ Connection - Oriented คือต้องมีการสร้างการติดต่อกันเป็น Session ทั้ง 2 ด้านเสียก่อน แล้วจึงจะรับส่งข้อมูลไปได้พร้อมกัน (Full duplex) เหมือนกับการใช้โทรศัพท์ติดต่อกัน เมื่อผู้ติดต่อต้นทางเรียกให้ตรงข้ามรับสายแล้วจึงเริ่มสนทนา เช่น พูดคำว่า "สวัสดี" หรือ "ฮัลโล" กันก่อนเพื่อให้แน่ใจว่า ฝ่ายตรงข้ามพร้อมจะติดต่อด้วย จากนั้นจึงเริ่มต้นติดต่อกัน และเมื่อต้องการจะเลิกติดต่อ ก็จะมีการพูดว่า "สวัสดี" ให้ฝ่ายตรงข้ามทราบว่าจะเลิกติดต่อ และวางสายไปซึ่งในระหว่างการติดต่อกันนั้น แม้ว่าฝ่ายใดฝ่ายหนึ่งหรือทั้งสองฝ่ายจะเงียบไป คือ ไม่พูดอะไรเป็นเวลานานๆ แต่การเชื่อมโยงระหว่างทั้งสองด้าน ยังคงมีอยู่ไม่ขาดไปจนฝ่ายหนึ่งจะวางสาย เช่นเดียวกับการติดต่อด้วยกลไกโปรโตคอล ทีซีพี เมื่อแอปพลิเคชันต้องการส่งผ่านข้อมูล จะใช้โปรโตคอลที่เหมาะสมในชั้น Protocol Layer ติดต่อไป และมีการสร้างช่องส่งข้อมูลผ่าน Port ที่กำหนด เพื่อส่งข้อมูลไปยังโปรโตคอล ทีซีพี

UDP (User Datagram Protocol) ใน Host-to-Host-Layer นอกจากมีโปรโตคอลทีซีพีทำงานแล้ว ก็ยังมีโปรโตคอลยูดีพี ที่มีคุณสมบัติแตกต่างกันอยู่ด้วย ในการรับส่งข้อมูลผ่านโปรโตคอลยูดีพี จะเป็นแบบที่ทั้งสองนั้นไม่จำเป็นต้องอาศัย การสร้างช่องทางเชื่อมต่อกันระหว่างเครื่องเซิร์ฟเวอร์ให้บริการ กับเครื่องที่ขอใช้บริการ โดยไม่ต้องแจ้งให้ฝ่ายรับข้อมูลเหมือนโปรโตคอลทีซีพี และไม่มีการตรวจสอบความถูกต้องครบถ้วนในการรับส่งข้อมูลนั้นๆ ด้วย เนื่องจากโปรโตคอลยูดีพีไม่มีสัญญาณตรวจทานข้อมูลในการส่งข้อมูลแต่ละครั้ง และไม่มีการส่งข้อมูลใหม่อีกในกรณีที่เกิดความผิดพลาดของการส่งข้อมูล เมื่อเป็นเช่นนั้น แอปพลิเคชัน หรือโปรเซสที่ต้องอาศัยโปรโตคอลยูดีพี ในการส่งผ่านข้อมูล ก็อาจจะสร้างขบวนการตรวจสอบข้อมูลขึ้นเอง

จะเห็นว่าโปรโตคอลชั้นบนขึ้นไป ที่ใช้การส่งผ่านข้อมูลโดยโปรโตคอลยูดีพี เช่น โปรโตคอล SNMP (ใช้ควบคุม และจัดการอุปกรณ์ในเครือข่าย) หรือโปรโตคอล DHCP (ใช้ส่งข้อมูลพารามิเตอร์ของเครือข่ายให้กับเครื่องลูกข่ายได้ใช้งาน) การส่งข้อมูลเหล่านั้น ไม่ต้องรับทราบหรือตรวจสอบว่าข้อมูลไปถึงปลายทางถูกหรือไม่ แต่กลไกการตรวจสอบข้อมูลที่มีการรับส่ง จะไปทำในขั้นตอนของโปรโตคอลชั้นที่สูงกว่าแทน

ตัวอย่างขั้นตอนกลไกการทำงานโดยใช้โปรโตคอลยูดีพี มีดังต่อไปนี้

1. ในขั้นตอนของ Process layer เมื่อโปรโตคอล ควบคุมอุปกรณ์เครือข่ายเช่น โปรแกรม Network Management ต้องการส่งข้อมูลไปยังอุปกรณ์ที่ต้องการแอปพลิเคชันนั้นจะติดต่อด้วยโปรโตคอล SNMP ในชั้น Process Layer

2. โพรโทคอล SNMP จะติดต่อกับโพรโทคอลยูดีพีในชั้นถัดไป เพื่อขอติดต่อผ่าน Port ที่กำหนด
3. โพรโทคอล SNMP เตรียมข้อมูลที่จะส่งรวมทั้งที่อยู่ปลายทาง
4. โพรโทคอล SNMP ส่งผ่านข้อมูลให้โพรโทคอลยูดีพีที่อยู่ในชั้น Host-to-Host layer
5. โพรโทคอลยูดีพีทำหน้าที่ผนึกข้อมูลหรือ ผนวງข้อมูลนั้นไปกับโพรโทคอลไอพีในชั้นถัดลงไป เพื่อส่งข้อมูลออกจากเครื่อง

ซึ่งจะเห็นว่ามึกลไกต่างจากการส่งข้อมูลด้วยโพรโทคอลที่ซีพีซึ่งจะต้องมีการติดต่อกันก่อนและทั้งสองฝ่ายรับทราบการรับส่งข้อมูลของช่องทางการส่งข้อมูลนั้น ซึ่งรูปแบบของยูดีพีแพ็กเกจ จะมีฟิลด์ข้อมูลส่วนข้อมูลนำทางน้อยมาก และไม่มีข้อมูลส่วนการตรวจสอบข้อมูล ทำให้ยูดีพีแพ็กเกจมีขนาดเล็ก และใช้หน่วยความจำหรือทรัพยากรของระบบน้อย

2.11.4 Application Layer

จะทำหน้าที่ให้ Application มีความสามารถเข้าถึงการบริการของเลเยอร์อื่นๆ และกำหนดโพรโทคอลที่ Application เพื่อแลกเปลี่ยนข้อมูลมีหลายๆ โพรโทคอล Application Layer และโพรโทคอลใหม่ที่กำลังอยู่ในระหว่างการพัฒนาโพรโทคอล Application Layer ที่รู้จักกันมาซึ่งถูกใช้ในการแลกเปลี่ยนข้อมูลดังนี้

1. Hypertext Transfer Protocol (HTTP) ใช้สำหรับการโอนย้ายไฟล์ที่ทำขึ้นบนเว็บเพจ
2. File Transfer Protocol (FTP) ใช้สำหรับการโอนย้ายไฟล์แบบโต้ตอบ (Interactive)
3. Simple Mail Transfer Protocol (SMTP) ใช้สำหรับการโอนย้ายเมลเสจเมลล์ และไฟล์ที่แนบมา
4. Simple Network Management Protocol (SNMP) ใช้ระหว่างคอนโซลการจัดการเน็ตเวิร์คกับเน็ตเวิร์คดีไวซ์ต่างๆ เช่น bridges, router และ hub เพื่อรวบรวมและแลกเปลี่ยนข้อมูลการจัดการเน็ตเวิร์ค

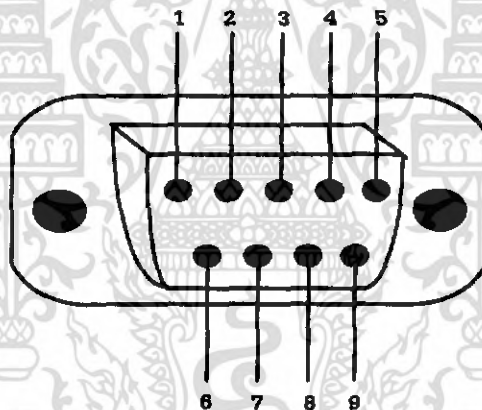
2.12 พอร์ตอนุกรม (RS-232)

มาตรฐานการเชื่อมต่อแบบอนุกรม RS-232 เป็นมาตรฐานอุตสาหกรรมที่ออกแบบมาเพื่อใช้ในการส่งข้อมูลอนุกรมแบบอะซิงโครนัส 2 ทิศทาง โดยมาตรฐาน RS-232 ในอดีตนั้นถูกออกแบบมาเพื่อการส่งข้อมูลจากคอมพิวเตอร์ไปยังโมเด็มเพียงอย่างเดียวเพื่อที่จะนำข้อมูลจากโมเด็มนี้สื่อสารผ่านสายโทรศัพท์ไปยังคอมพิวเตอร์อีกชุดซึ่งอยู่ห่างไกลกัน โดยคณะกรรมการที่ เรียกว่า อุตสาหกรรมอิเล็กทรอนิกส์ (Electronic Industries Association : EIA) ได้วางมาตรฐานที่มีชื่อเรียกกันว่า EIA RS-232 มาตรฐานนี้ในช่วงแรกจะ

ใช้คอนเน็กเตอร์แบบ DB-25 โดยกำหนดความยาวสูงสุดของสายสัญญาณไว้ที่ 50 ฟุต มีระดับสัญญาณตั้งแต่ -3 ถึง -12 V แสดงว่ามีข้อมูล (Mark) และ +3 ถึง +12 V แสดงว่าเป็นช่องว่าง (Space)

มาตรฐาน RS-232 ได้กำหนดรูปแบบของอุปกรณ์เชื่อมต่อข้อมูล (Data Terminal Equipment: DTE) ไว้ว่า อุปกรณ์ DTE จะต้องเป็นอุปกรณ์ที่มีการประมวลผลในตัวเช่น ไมโครคอนโทรลเลอร์หรือไมโครคอมพิวเตอร์ ซึ่งมีความสามารถในการสร้างบิตข้อมูลแบบอนุกรมได้ ส่วนอุปกรณ์ DCE จะทำหน้าที่เป็นเพียงตัวรับข้อมูลที่ส่งมาจาก DTE เท่านั้น โดยการรับส่งข้อมูลระหว่างอุปกรณ์ทั้งสองจะกระทำผ่านมาตรฐาน RS-232

คอนเน็กเตอร์สำหรับการเชื่อมต่อแบบ RS-232 จะใช้คอนเน็กเตอร์แบบ DB-25 ตัวผู้หรือ DB-9 ตัวผู้ซึ่งคอนเน็กเตอร์แบบ DB-25 จะมีขาต่อใช้งานเพียง 9 เส้นเช่นเดียวกับคอนเน็กเตอร์แบบ DB-9 เนื่องจากขาอื่นๆ ที่เคยใช้งานในอดีต ปัจจุบันมีการใช้งานไม่มากนัก จึงถูกยกเลิกไป โดยแสดงรูปร่างในตำแหน่งขาในรูปที่ 2.20

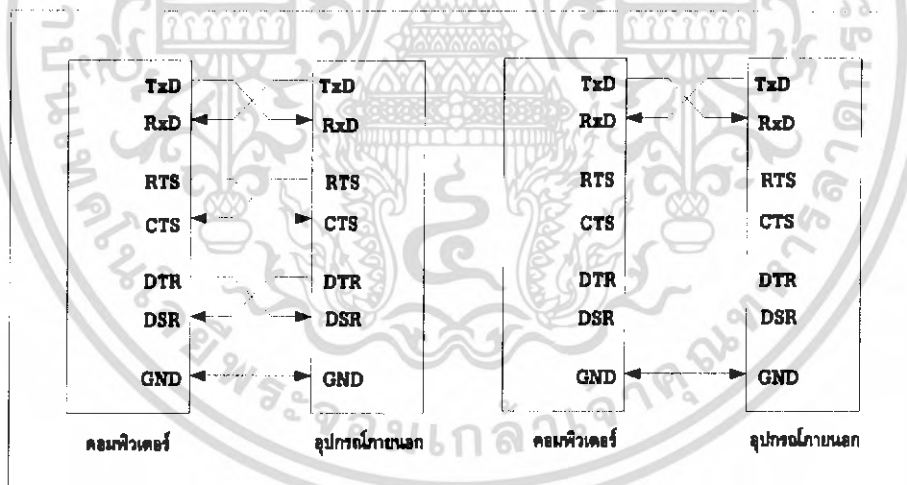


รูปที่ 2.19 แสดงคอนเน็กเตอร์อนุกรม 9 ขา หรือแบบ DB-9

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 2.4 แสดงการจัดขาของคอนเน็กเตอร์แบบ DB-9

ขาที่	ชื่อสายสัญญาณ	ชนิดของสายสัญญาณ
1	Data Carrier Detect : DCD	อินพุต
2	Received Data : RxD	อินพุต
3	Transmitted Data : TxD	เอาต์พุต
4	Data Terminal Ready : DTR	เอาต์พุต
5	Signal Ground : GND	-
6	Data Set Ready : DSR	อินพุต
7	Request To Send : RTS	เอาต์พุต
8	Clear To Send : CTS	อินพุต
9	Ring Indicator : RI	อินพุต



รูปที่ 2.20 การเชื่อมต่ออุปกรณ์ภายนอกกับพอร์ตอนุกรมของคอมพิวเตอร์ในลักษณะต่างๆ

สำหรับการเชื่อมต่อคอมพิวเตอร์กับอุปกรณ์ภายนอกแสดงในรูปที่ 2.21 เป็นการเชื่อมต่อแบบ Null modem หรือการเชื่อมต่อโดยตรงไม่ต้องผ่านโมเด็ม โดยมีการตรวจสอบหรือแฮนด์เช็กเต็มรูปแบบ ส่วนในรูปที่ 1-4 เป็นการเชื่อมต่อแบบ Null modem ในลักษณะที่ใช้สายสัญญาณเพียง 3 เส้นโดยเส้นหนึ่งสำหรับข้อมูล อีกเส้นสำหรับรับข้อมูล และเส้นสุดท้ายเป็นกราวด์ สำหรับรายละเอียดหน้าที่การทำงานในแต่ละขาของพอร์ตอนุกรม RS-232 มีดังนี้

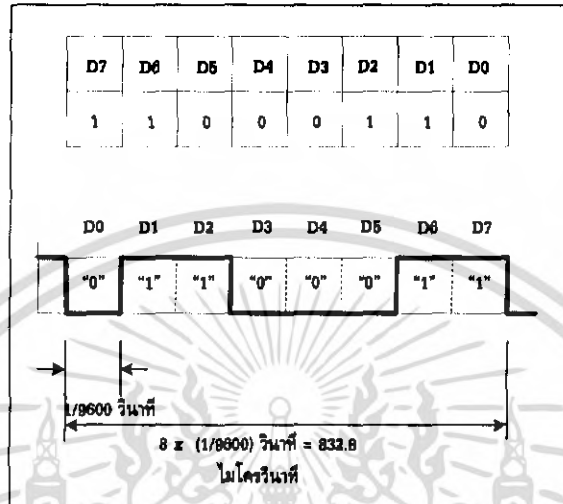
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1. Data Carrier Detect: DCD หรืออาจเรียกว่า Carrier Detect: CD ขานี้จะแอกทีฟเมื่อมีการส่งสัญญาณพาห์จากอุปกรณ์สื่อสารข้อมูลเช่น โมเด็ม
2. Receive Data: RD หรือ RXD ขานี้ใช้เพื่อรับสัญญาณอนุกรมเข้ามายังคอมพิวเตอร์โดยนำข้อมูลที่อ่านได้เก็บไว้ในรีจิสเตอร์ บัฟเฟอร์
3. Transmitted Data: TD หรือ TXD ขานี้ใช้เพื่อส่งข้อมูลออกจากคอมพิวเตอร์โดยนำข้อมูลที่เก็บไว้ในบัฟเฟอร์สำหรับส่งข้อมูลส่งออกไป
4. Data Terminal Ready: DTR เป็นขาสัญญาณที่ส่งออกจากคอมพิวเตอร์เพื่อให้อุปกรณ์ปลายทางรับรู้ว่าการติดต่อด้วย โดยขา DTR นี้จะต้องเชื่อมต่อกับขา DSR ของอุปกรณ์ปลายทาง และขา DSR ของอุปกรณ์ปลายทางจะต้องเชื่อมต่อกับขา DSR คอมพิวเตอร์ ถ้าใช้การเชื่อมต่อเป็นแบบ Null Modem ซึ่งใช้สายในการเชื่อมต่อเพียง 3 เส้นจะต้องต่อขา DTR และ DSR ของตัวมันเองเข้าด้วยกันและต้องต่อกับขา DCD ด้วยในกรณีที่โปรแกรมสื่อสารที่ใช้มีการตรวจจับสัญญาณพาห์
5. Signal Ground: GND ขากราวด์ของระบบ
6. Data Set Ready: DSR ขานี้จะใช้คู่กับขา DTR เพื่อตรวจสอบการเชื่อมต่อกันระหว่างคอมพิวเตอร์กับอุปกรณ์ปลายทาง ซึ่งขา DSR นี้จะเป็นขาสำหรับรับข้อมูลจากภายนอกซึ่งถูกส่งมาจากขา DTR
7. Request To Send: RTS เป็นขาสำหรับร้องขอให้ทางอุปกรณ์ปลายทางส่งข้อมูลกลับมายังคอมพิวเตอร์ โดยขาที่รับสัญญาณ RTS ก็คือขา CTS ในกรณีที่ใช้การเชื่อมต่อแบบ Null modem 3 สาย จะต้องเชื่อมต่อกับขา RTS และ CTS ของตัวมันเองเข้าด้วยกันเพื่อจะให้การรับและส่งข้อมูลสามารถเกิดขึ้นได้ตลอดเวลา
8. Clear To Send: CTS ขานี้จะคอยรับสัญญาณจากขา RTS เมื่อรับสัญญาณได้ ข้อมูลที่ขา TXD จะถูกส่งออกไป ดังนั้นขานี้จึงถูกตรวจสอบอุปกรณ์ต่อพ่วงว่าพร้อมหรือไม่
9. Ring Indicator: RI ใช้แสดงสถานะสัญญาณเรียกจากสายโทรศัพท์ ปกติในการสื่อสารโดยทั่วไปสายนี้จะไม่ถูกใช้งานจะใช้งานต่อเมื่อมีการเชื่อมต่อกับโมเด็มและโปรแกรมมีการตรวจสอบสัญญาณนั้นเท่านั้น

2.12.1 จังหวะเวลาของการสื่อสารข้อมูลอนุกรม

ในการสื่อสารข้อมูลแบบอนุกรม เพื่อรับหรือส่งข้อมูล จะเป็นลักษณะของกลุ่มข้อมูล ดังนั้นอัตราความเร็วจะต้องมีค่าเท่ากันระหว่างการรับและการส่งโดยทั่วไปเราจะระบุความเร็วของจำนวนบิตในการรับและส่งข้อมูล เป็นจำนวนของบิตที่จะส่งใน 1 วินาที โดยเรียกความเร็วในการส่งข้อมูลว่า อัตราบอด (Baud Rate) ซึ่งมีหน่วยเป็นบิตต่อวินาที เช่น 300, 1, 200, 2,400, 4, 800 และ 9,600 บิตต่อวินาที ในรูป 12 ถ้าหากมีการ

ส่งข้อมูลด้วยความเร็ว 9600 บิตต่อวินาที จะใช้เวลาในการรับส่งข้อมูลหนึ่งบิตมีค่าเท่ากับ $1/9600$ หรือ 104.1 ไมโครวินาที และเวลาในการรับส่งข้อมูลทั้ง 8 บิตจะมีค่าเท่ากับ 8×104.1 หรือ 832.8 ไมโครวินาที



รูป 2.21 แสดงการส่งข้อมูลแบบอนุกรมด้วยความเร็ว 9600 บิตต่อวินาที

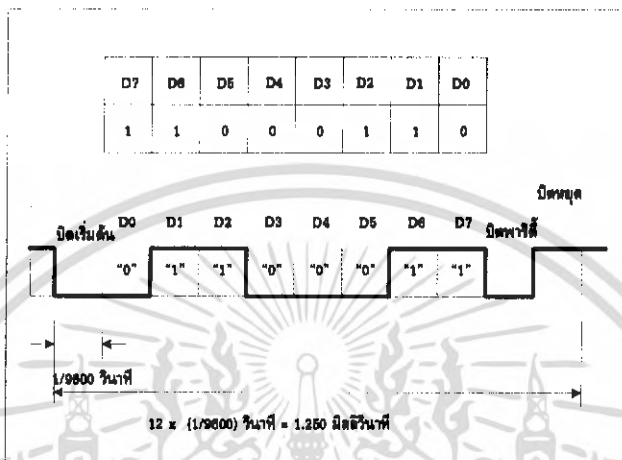
2.12.2 รูปแบบของการสื่อสารข้อมูลอนุกรม

การสื่อสารข้อมูลอนุกรมแบบอะซิงโครนัส เป็นวิธีการรับและส่งข้อมูลโดยไม่ต้องอาศัยสัญญาณนาฬิกาส่งร่วมไปด้วย แต่จะใช้อัตราความเร็วของจำนวนข้อมูลต่อวินาที และจะทำการเพิ่มบิตข้อมูลบางอย่างร่วมไปกับการส่งข้อมูลจริง เพื่อจะได้ทำการตรวจสอบข้อมูลได้อย่างถูกต้องมากยิ่งขึ้นแสดงดังรูปที่ 13 ซึ่งประกอบด้วยกัน 4 ส่วนคือ

1. บิตเริ่มต้น (Start bit) จะมีขนาด 1 บิต จะเป็นระดับลอจิกตรงกันข้ามกับระดับลอจิกของสถานะการสื่อสารขณะที่ยังไม่มีการส่งข้อมูล
2. บิตข้อมูล (Data bit) จะเริ่มจากบิตที่มีนัยสำคัญต่ำสุดก่อนหรือ บิต LSB ก่อน โดยข้อมูลที่จะส่งอาจจะมีความยาว 5, 6, 7 หรือ 8 บิตก็ได้
3. บิตแสดงสถานะเลขคู่หรือเลขคี่ (Parity bit) มีขนาด 1 บิตโดยบิตนี้จะนำไปต่อท้ายกับบิตข้อมูล ค่าของบิตนี้ขึ้นอยู่กับจำนวนค่าของข้อมูลที่เป็น "1" โดยเลือกการส่งข้อมูลเป็นแบบ พาริตีคู่หรือ พาริตีคี่ ตัวอย่าง ถ้ากำหนดให้มีการส่งข้อมูลแบบพาริตีคู่ แต่ข้อมูลมีเลข 1 เป็นจำนวนคี่ก็จะให้บิตพาริตีนี้เป็น "1" เพื่อจะได้จำนวนเลข "1" เป็นคี่นั่นเองทำนองเดียวกันทางด้านรับเองก็ต้องมีการตรวจสอบจำนวนข้อมูลที่ได้รับเข้ามาเป็น "1" รวมทั้งบิตพาริตี 1 บิต ถ้ามีค่า "1" เป็นจำนวนคู่แสดงว่าข้อมูลที่รับเข้ามาถูกต้องสามารถกำหนดการรับและส่งข้อมูลเป็นแบบ NONE โดยไม่ต้องมีการตรวจสอบพาริตีบิตก็ได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับใช้เพื่อการเรียนการสอนเท่านั้น เมื่ออนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4. บิตสุดท้ายหรือบิตหยุด (Stop bit) เป็นการระบุถึงขอบเขตของการสิ้นสุดข้อมูล โดยจะทำให้หาข้อมูลมีสถานะ ลอจิกเป็น "1" ซึ่งอาจมีจำนวนมากกว่าหนึ่งบิตก็ได้ เช่น 1 บิต 1.5 บิต หรือ 2 บิต

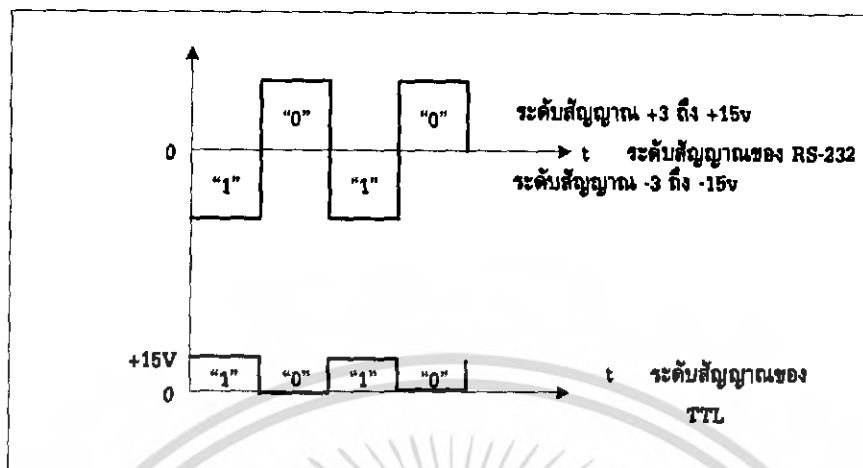


รูป 2.22 แสดงการส่งข้อมูลขนาด 8 บิตแบบอนุกรมพร้อมด้วย บิตเริ่มต้น, บิตพาริตี, บิตหยุด ด้วยความเร็ว 9600 บิตต่อวินาที

2.12.3 การเชื่อมต่อพอร์ตอนุกรมมาตรฐาน RS-232

การกำหนดมาตรฐานการเชื่อมต่อแบบอนุกรม EIA RS-232 (x) เป็นมาตรฐานอุตสาหกรรม โดยคณะกรรมการสมาคมอุตสาหกรรมอิเล็กทรอนิกส์ (Electronic Industries Association) ออกแบบมาเพื่อใช้ในการส่งข้อมูลอนุกรมแบบ อะซิงโครนัส 2 ทิศทางเพื่อให้มีการใช้งานในการเชื่อมต่อที่สอดคล้องกันระหว่างอุปกรณ์คอมพิวเตอร์ต่างๆ การรับส่งสัญญาณจะกำหนดความยาวสูงสุดไว้ที่ไม่เกิน 50 ฟุตโดยมีระดับ สัญญาณตั้งแต่ 3 โวลต์ จนถึง 15 โวลต์ สำหรับลอจิก "0" และมีระดับแรงดันที่ -3 โวลต์ จนถึง -15 โวลต์ สำหรับลอจิก "1" ดังแสดงในรูป 2.24

ดังนั้นสังเกตได้ว่าจะมีระดับแรงดันที่ใช้ในสถานะลอจิก "0" และ ลอจิก "1" แตกต่างออกไปจากระบบไอซีดิจิทัลทั่วๆ ไป การต่อใช้งานจึงต้องมีอุปกรณ์ที่ทำหน้าที่เปลี่ยนระดับแรงดันจาก 0 - 5 โวลต์ จากไมโครคอนโทรลเลอร์ให้เป็นระดับแรงดันที่สูงกว่า +3 หรือต่ำกว่า - 3 โดยจะมีไอซีสำเร็จรูปพร้อมใช้งานหรืออาจจะต่อวงจรจากทรานซิสเตอร์ได้



รูปที่ 2.23 แสดงระดับแรงดันสัญญาณของพอร์ตอนุกรม RS-232 กับ TTL ในสถานะลอจิก "1" และ "0"

2.13 ไมโครคอนโทรลเลอร์ MCS-51

2.13.1 ไมโครคอนโทรลเลอร์ MCS-51

ปัจจุบันไมโครคอนโทรลเลอร์ได้เข้ามามีบทบาทมากกับการควบคุมในงานอุตสาหกรรม การพัฒนาอุปกรณ์สารกึ่งตัวนำอย่างต่อเนื่องผนวกกับเทคโนโลยีการผลิตที่วิวัฒนาการมาเป็นหน่วยไมโครนำไปสู่การสร้าง CHIP และมากด้วยคุณภาพ

ไมโครคอนโทรลเลอร์ 8051 เป็น CHIP ที่นิยมมากตัวหนึ่งที่ใช้ในงานควบคุมเนื่องจากความสามารถที่สูง และง่ายต่อการใช้งานทั้งนี้ต้องขึ้นอยู่กับผู้ใช้งานที่จะมีวิธีการอย่างไรในการดึงความสามารถของ CHIP มาใช้ให้เต็มที่เพื่อให้เกิดประสิทธิภาพสูงสุดต่อไป

2.13.2 โครงสร้างของ MCS-51

ไมโครคอนโทรลเลอร์ตระกูล MCS - 51 มีด้วยกันหลายเบอร์ขึ้นกับโครงสร้างภายใน บางเบอร์จะมีหน่วยความจำภายในเป็นแบบ รอม บางเบอร์เป็นแบบ อีพรอม บางเบอร์มี แรม ภายใน 128 ไบต์ บางเบอร์มี 256 ไบต์ เป็นต้น คุณสมบัติที่สำคัญของ MCS - 51 มีดังนี้

1. มีหน่วยความจำ รอม 4 กิโลไบต์
2. มีหน่วยความจำ แรม 128 ไบต์
3. มีพอร์ตอินพุต เอาต์พุต ขนาด 8 บิต 4 พอร์ต
4. มีไทม์เมอร์ 16 บิต 2 ตัว
5. สามารถอินเทอร์รัพท์ได้ 5 แหล่ง
6. มีวงจรรอสซิลเลเตอร์และวงจรมหาพีคานชิพ
7. มีพอร์ตอนุกรมที่สามารถรับส่งข้อมูลแบบฟูลดูเพล็กซ์ (Full Duplex) ความเร็วสูง

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์สำหรับการใช้งานเพื่อการศึกษาเท่านั้น เมื่ออนุญาตให้นำไปใช้ประโยชน์ด้านการศึกษา
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

8. อ่างหน่วยความจำโปรแกรมภายนอกได้ 64 กิโลไบต์
9. อ่างหน่วยความจำข้อมูลภายนอกได้ 64 กิโลไบต์
10. สามารถประมวลผลที่ละบิตได้
11. สามารถอ่างหน่วยความจำแบบบิตได้ 210 ตำแหน่ง
12. หนึ่งวัฏจักรคำสั่งกินเวลาประมาณ 1 ไมโครวินาที ขณะทำงานด้วยสัญญาณนาฬิกา 12 เมกกะเฮิรตซ์

2.13.3 การจัดการหน่วยความจำและการเชื่อมต่อ

หน่วยความจำของไมโครคอนโทรลเลอร์ 8051 แบ่งเป็น 2 ประเภท คือ

1. หน่วยความจำโปรแกรม (Program Memory) หน่วยความจำประเภทนี้ คือ ROM ใช้เก็บโปรแกรมที่ใช้
2. ในการควบคุมระบบ ซึ่งเป็นหน่วยความจำประเภท non-volatile หน่วยความจำข้อมูล (Data Memory) หน่วยความจำประเภทนี้ได้แก่ RAM

2.13.3.1 หน่วยความจำโปรแกรม

ใน 8051 จะแบ่งหน่วยความจำประเภทนี้เป็นอีก 2 ประเภท หน่วยความจำโปรแกรมภายนอก (external memory) กล่าวคือ ROM ที่มาต่อภายนอกตัว 8051 ส่วนอีกประเภทหนึ่ง คือ หน่วยความจำภายใน (internal memory) ได้แก่ ROM ภายในตัวไมโครคอนโทรลเลอร์เอง การเลือกใช้หน่วยความจำโปรแกรมภายนอกหรือหน่วยความจำโปรแกรมภายในทำได้โดยการให้สัญญาณทางไฟฟ้าที่ขา /EA โดย

1. สัญญาณทางไฟฟ้าที่ขา /EA เป็นลอจิก 0 หมายถึง หน่วยความจำโปรแกรมภายนอก
2. สัญญาณทางไฟฟ้าที่ขา /EA เป็นลอจิก 1 หมายถึง หน่วยความจำโปรแกรมภายใน

2.13.3.2 หน่วยความจำข้อมูล

หน่วยความจำข้อมูลมีหน้าที่สำหรับเก็บข้อมูล หรือตัวแปรที่เกิดขึ้นในขณะที่กำลังประมวลผลโปรแกรมไว้เป็นการชั่วคราว โดยพื้นฐานแล้วหน่วยความจำข้อมูลจัดเป็นหน่วยความจำ RAM แบบสแตติก ดังนั้นเมื่อไม่มีการ จ่ายไฟฟ้าให้กับระบบ ก็จะมีผลทำให้ข้อมูลที่จัดเก็บไว้ภายในหน่วยความจำนี้สูญหายไป พื้นที่ของหน่วยความจำข้อมูลของ 8051 สามารถมีได้สูงสุดไม่เกิน 64 กิโลไบต์ และแยกประเภทออกเป็นสองลักษณะตามตำแหน่งที่ตั้งของหน่วยความจำนั้น

1. หน่วยความจำโปรแกรมภายใน (Internal Data Memory) ซึ่ง เป็น RAM ที่อยู่ภายในตัวของไอซีไมโครคอนโทรลเลอร์
2. หน่วยความจำข้อมูลภายนอก (External Data Memory) ซึ่งเป็นการใช้ไอซีหน่วยความจำ RAM มาเพิ่มเติมเข้าไปในวงจร ลักษณะเดียวกับการนำไอซี EPROM มาใช้งานเป็นหน่วยความจำโปรแกรมนั่นเอง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.13.4 การอ้างตำแหน่งข้อมูล

การอ้างตำแหน่งข้อมูล (Addressing Mode) เป็นการกำหนดสถานที่อยู่ของข้อมูลที่จะนำมาใช้งานคำสั่งต่างๆ ของไมโครคอนโทรลเลอร์ MCS-51 มีการทำงานกับข้อมูลที่แตกต่างกันอยู่หลายคำสั่ง ซึ่งแต่ละคำสั่งจะมีความยากง่ายและสลับซับซ้อนแตกต่างกันไป คำสั่งแต่ละคำสั่งจะมีความยากของรหัสคำสั่งแตกต่างกันตั้งแต่ 1-3 ไบต์ โดยมีหลายคำสั่งที่ทำงานในลักษณะเดียวกัน แต่มีความยาวของคำสั่งแตกต่างกัน คำสั่งการโอนย้ายข้อมูลจากหน่วยความจำภายนอกหรือหน่วยความจำภายใน เราสามารถกำหนดตำแหน่งที่อยู่ของข้อมูลได้หลายวิธี เช่นการอ้างตำแหน่งที่อยู่โดยตรง หรือโดยทางอ้อมซึ่งใช้รีจิสเตอร์ชี้ตำแหน่งก็ได้ ดังนั้นการทำความเข้าใจกับวิธีการอ้างตำแหน่งที่อยู่ของข้อมูลจึงเป็นสิ่งที่ต้องเรียนรู้ และทำความเข้าใจก่อนที่จะเรียนรู้การทำงานของคำสั่งต่างๆ ต่อไปซึ่งจะทำให้การเขียนโปรแกรมมีประสิทธิภาพมากขึ้น การอ้างถึงตำแหน่งของข้อมูล MCS-51 สามารถแบ่งออกเป็นกลุ่มต่างๆ ดังนี้

1. การอ้างตำแหน่งโดยตรง (Direct addressing)
2. การอ้างตำแหน่งทางอ้อม (Indirect addressing)
3. การอ้างตำแหน่งแบบรีจิสเตอร์ (Register addressing)
4. การอ้างตำแหน่งแบบรีจิสเตอร์เฉพาะ (Register-specific instruction)
5. การอ้างตำแหน่งแบบค่าทันที (Immediate addressing)
6. การอ้างตำแหน่งแบบอินเด็กซ์ (Index addressing)
7. การอ้างตำแหน่งแบบบิต (Bit addressing)

2.14 ไอซีเบอร์ RTL8019AS

2.14.1 คุณสมบัติของ RTL8019AS

คุณสมบัติของ RTL8019AS มีรายละเอียดดังต่อไปนี้

1. เป็น POFB ชนิด 100 pin
2. ใช้ได้กับ software RTL8019
3. รองรับโหมด PnP auto detect (เฉพาะรุ่น RTL8019AS)
4. ใช้ได้กับ Ethernet II และ IEEE802.3 10Base5, 10Base2, 10BaseT
5. Software เข้ากับ NE2000 ได้ ทั้งชนิด slot 8 บิต และ 16 บิต
6. รองรับได้ทั้งโหมด jumper และ jumperless
7. โหมด jumperless รองรับการทำงานแบบ Plug and Play ของ Microsoft
8. สามารถรองรับฟังก์ชัน Full-Duplex Ethernet เพื่อใช้ช่อง bandwidth คู่รองรับโหมด power down ได้ 3 ระดับ คือ

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์ระดับ sleep การใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- Power down ขณะนาฬิกาภายในเดินอยู่
 - Power down พร้อมหยุด (halt) นาฬิกาภายใน
9. มีฟังก์ชันการรับข้อมูลล่วงหน้า (built-in data prefetch function) เพื่อการทำงานที่มีประสิทธิภาพยิ่งขึ้น
 10. รองรับ UTP, AUI & BNC auto-detect (เฉพาะรุ่น RTL8019AS)
 11. รองรับการจับขั้วอัตโนมัติ (auto polarity correction) เมื่อใช้กับ 10BaseT
 12. รองรับ 8 IRQ lines
 13. รองรับ 16 I/O base address และโหมด extra I/O address fully decode (เฉพาะรุ่น RTL8019AS)
 14. รองรับ 16K, 32K, 64K และโหมดการเข้าถึง BROM แบบ 16K-page (สามารถเพิ่มได้ถึง 256 Pages ในอัตรา 16 K byte/page)
 15. รองรับคำสั่ง BROM disable command เพื่อปลดหน่วยความจำหลังการ boot รีโมด
 16. รองรับหน่วยความจำ flash memory อ่าน/เขียน (เฉพาะรุ่น RTL8019AS)
 17. มี SRAM 16 k byte ติดตั้งในตัว (เฉพาะรุ่น RTL8019AS)
 18. ใช้ 9346 (EEPROM 64*16-bit) เก็บข้อมูล resource configuration และ ID parameter
 19. สามารถโปรแกรม blank 9346 แบบ on board เพื่อให้เหมาะกับการใช้งานได้
 20. รองรับ 4 diagnostic LED pins, สามารถโปรแกรม outputs ได้

2.14.2 รายละเอียดทั่วไปของไอซีเบอร์ RTL8019AS

RTL8019AS เป็น Ethernet Controller ภายนอกที่รองรับการปฏิบัติการ Plug and Play NE2000 สามารถปรับใช้ได้ทั้งฟังก์ชัน full-duplex และ power down และควบคุมระดับการ power down ได้ถึง 3 ระดับ RTL8019AS ออกแบบมาเพื่อใช้กับอุปกรณ์ network ระบบ GREEN PC โดยเฉพาะในฟังก์ชัน full-duplex สามารถรับและส่งกับ full-duplex Ethernet switching hub พร้อมกันแบบ twisted-pair ซึ่งไม่เพียงเพิ่มช่อง bandwidth จาก 10 Mbps เป็น 20 Mbps ได้เท่านั้น แต่ยังคงปัญหาอันเนื่องมาจากการชนกัน (contention) ของ Ethernet CSMA/CD protocol ได้อีกด้วย มีฟังก์ชัน Plug and Play ของ Microsoft ซึ่งทำให้ผู้ใช้ไม่ต้องกังวลถึง adapter's resource configuration เช่น IRQ, I/O และ memory address ฯลฯ แต่ไม่สามารถใช้ Plug and Play ในงานพิเศษบางอย่างได้ นอกจากนี้ RTL8019AS ยังรองรับออปชั่น jumper และ jumperless ได้อีกด้วย

RTL8019AS ออกแบบมาให้สามารถตรวจพบ 10BaseT transceiver, BNC และ AUI interface ได้โดยอัตโนมัติ (auto-detect) 10BaseT transceiver สามารถแก้ไขขั้วผิด (polarity error) บน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

receiving pair ได้แบบอัตโนมัติ นอกจากนี้ยังมีออฟชั่น 8 IRQ lines และ 16 I/O base address สำหรับการเปลี่ยน resource configuration จำนวนมากๆ อีกด้วย

RTL8019AS รองรับได้ทั้ง BROM 16k, 32k & 64k byte และ flash memory interface มีฟังก์ชัน page mode ซึ่งสามารถเพิ่มหน่วยความจำจากระบบเดิม 16k-byte เป็นหน่วยความจำ BROM 4M-byte ได้ โดยมีคำสั่ง BROM disable command สำหรับปลดหน่วยความจำ BROM ให้พร้อมสำหรับระบบอื่นๆ (เช่น EMM386, ฯลฯ) หลังจากที่โหลดโปรแกรม BROM เสร็จเรียบร้อยแล้วอีกด้วย

RTL8019AS มีชิพ SRAM เดียวขนาด 16 k-byte ติดตั้งแบบในตัวจึงเหมาะกับการใช้งานของฟังก์ชันมากกว่า และลดการ sourcing และ inventory ของ SRAM ได้อีกด้วย

รูปที่ 2.24 ไอซีเบอร์ RTL8019AS

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

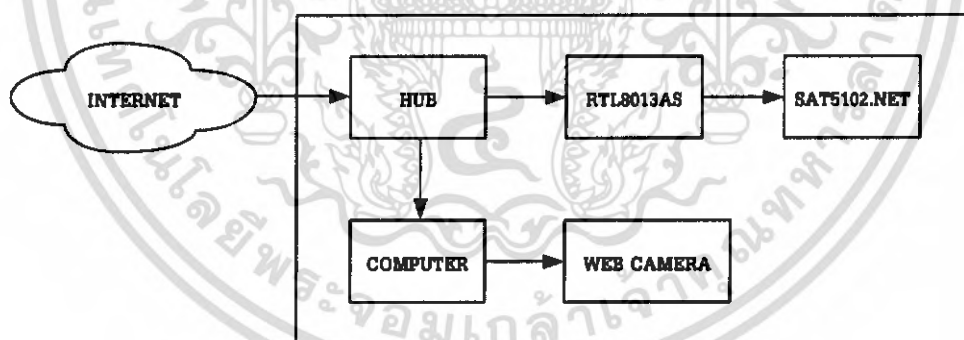
บทที่ 3

การออกแบบ การสร้างและการทำงาน

3.1 กล่าวนำ

การออกแบบและการสร้างระบบเปิด - ปิดและรายงานสถานะสวิตช์ผ่านระบบอินเทอร์เน็ต ได้แบ่งออกเป็น 2 ส่วน คือ ชุดของวงจรควบคุม และส่วนของโปรแกรม ในชุดของวงจรควบคุมจะประกอบไปด้วย 2 ส่วนคือ ในส่วนของแผงวงจรควบคุมสวิตช์ และส่วนของกล้อง Web Camera

โดยการทำงานจะทำการสั่งงานสวิตช์ผ่านหน้า Web Page ของชุดคำสั่งที่ได้เตรียมไว้โดยติดต่อผ่านระบบอินเทอร์เน็ตเชื่อมโยงเข้ากับ HUB ซึ่งเป็นตัวกลางในการเชื่อมต่อข้อมูลเข้ากับบอร์ดใช้งานซึ่งมีไอซีเบอร์ RTL8019AS ซึ่งเป็น IC ที่ออกแบบมาเพื่อใช้กับอุปกรณ์เน็ตเวิร์กเชื่อมต่อเข้ากับบอร์ด SAT5102.NET เพื่อสนับสนุนการทำงานซึ่งกันและกัน เมื่อมีการสั่งให้สวิตช์ ON ค่าที่ส่งออกไปจะสั่งให้สวิตช์ทำงาน และในส่วนของภาคสวิตช์จะส่งค่ากลับมาเพื่อบอกสถานะการทำงานในส่วนของกล้อง Web Camera จะมีการเก็บภาพในบริเวณที่กำหนดจากนั้นจะมีการส่งภาพกลับมายังหน้า Web Page ที่ได้ทำการสร้างไว้แล้วในโปรแกรมบนคอมพิวเตอร์



รูปที่ 3.1 บล็อกไดอะแกรมของระบบโดยรวม

3.2 การออกแบบและการสร้างแผงวงจรชุดควบคุม

การออกแบบและการสร้างแผงวงจรระบบเปิด - ปิดและรายงานสถานะสวิตช์ผ่านระบบอินเทอร์เน็ต ได้คำนึงถึงความสะดวกในการออกแบบคือ

1. เพื่อใช้ในควบคุมอุปกรณ์ไฟฟ้าผ่านระบบอินเทอร์เน็ตได้ 4 จุด
2. เพื่อให้แผงควบคุมการรายงานสถานะของสวิตช์ทั้ง 4 จุด ผ่านระบบอินเทอร์เน็ตได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3. เพื่อให้แผงวงจรสามารถประยุกต์ใช้งานควบคุมสวิตช์อุปกรณ์ไฟฟ้าในระบบ LAN ได้
4. เพื่อให้ในส่วน Web Page ของชุดคำสั่งสามารถรองรับการทำงานของกล้อง Web Camera ได้

การออกแบบต้องกำหนดขีดความสามารถของวงจรควบคุม ระบบเปิด - ปิดและรายงานสถานะสวิตช์ผ่านระบบอินเทอร์เน็ตให้เหมาะกับโครงการนี้โดย

1. การติดต่อกับคอมพิวเตอร์สามารถติดต่อสื่อสารผ่านพอร์ตอนุกรมโดยใช้มาตรฐานการติดต่อแบบ Rs-232 โดยใช้ไอซีเบอร์ MAX232 เป็นตัวปรับแรงดันระหว่างคอมพิวเตอร์และแผงวงจรไมโครคอนโทรลเลอร์ MCS-51
2. การใช้หน่วยประมวลผลกลางใช้ไอซีตระกูล MCS-51 เบอร์ AT89C51AC2
3. การเชื่อมต่อกับระบบอินเทอร์เน็ตผ่านไอซีเบอร์ RTL8019AS ซึ่งเป็นไอซีที่ออกแบบมาเพื่อใช้ในการติดต่อสื่อสารกับระบบเน็ตเวิร์ก
4. การติดต่อกับระบบอินเทอร์เน็ตสามารถติดต่อโดยการสื่อสารผ่านระบบ LAN ผ่านสาย RJ-45
5. แหล่งจ่ายไฟออกแบบให้มีไฟเลี้ยงวงจรโดยใช้ไอซีเบอร์ LM2575HV ที่ทำหน้าที่รักษาระดับแรงดันให้คงที่ เพื่อจ่ายให้กับแผงวงจรควบคุมโดยที่แหล่งจ่ายนี้จะมีไดโอดบริดจ์ 1 ตัว ทำหน้าที่ป้องกันการจ่ายไฟผิวดันที่จะทำให้เกิดความเสียหายต่อแผงวงจรไมโครคอนโทรลเลอร์

3.3 การออกแบบและการสร้างระบบควบคุม

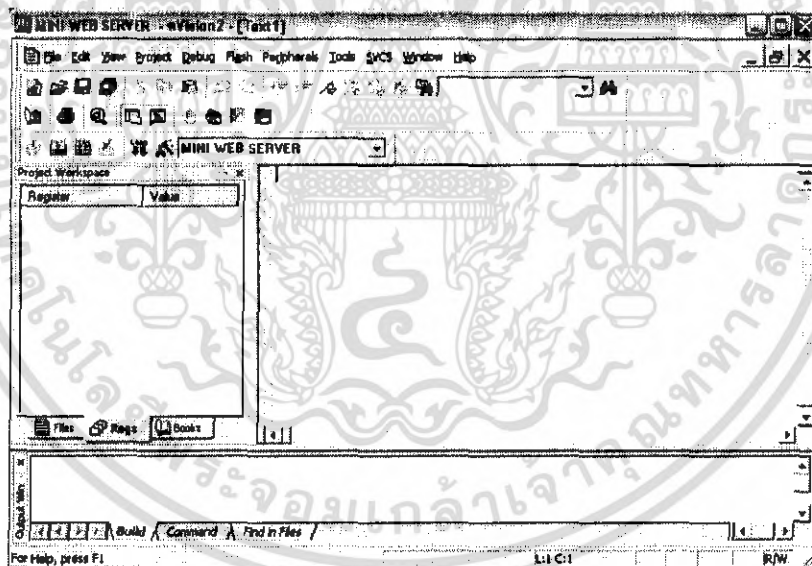
3.3.1 การเขียนคำสั่งด้วยโปรแกรม Keil

เมื่อทำการติดตั้งเรียบร้อยแล้ว ขั้นตอนต่อไปคือการเริ่มต้นใช้งานสำหรับโปรแกรม Keil การเขียนโปรแกรมด้วย Kiel นั้นต้องทำภายใต้โปรเจ็ค หมายความว่า เราต้องสร้างโปรเจ็คขึ้นมาก่อนแล้วนำไฟล์ภาษาซีที่เขียนขึ้น เพิ่มเข้ามาในโปรเจ็ค ก่อนที่จะทำการคอมไพล์ และซิมูเลต ซึ่งขั้นตอนต่างๆ มีดังนี้

1. เปิดโปรแกรม Keil โดยคลิกที่ Start -> All Program -> Keil uVision ดังรูป
2. เริ่มสร้างโปรเจ็คใหม่โดยการ คลิกที่ Project -> New Project หรืออาจใช้คีย์ลัดโดยการกด Alt+P แล้วตามด้วย N
3. ทำการเลือกตำแหน่งที่จะเซฟโปรเจ็ค และตั้งชื่อโปรเจ็คตามความต้องการ เมื่อเรียบร้อยแล้วคลิกที่ Save
4. เมื่อทำการบันทึกโปรเจ็คเรียบร้อยแล้ว จะได้หน้าต่าง Select Device for Target เพื่อให้ผู้ใช้งานเลือกว่าต้องการที่จะเขียนโปรแกรมเพื่อนำไปใช้กับไมโครคอนโทรลเลอร์ตัวไหน ในที่นี่เราจะใช้ AT89C51AC2 ซึ่งเป็นของ Atmel

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

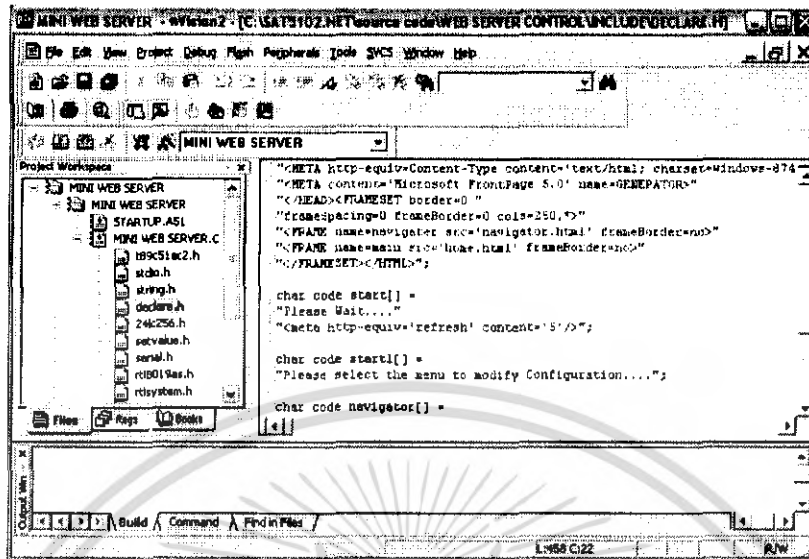
5. ให้ทำการเลื่อน Scroll bar ลงมาจนถึง Atmel แล้วคลิกที่ ⊕ เพื่อทำการแยกเบอร์ต่างๆ ออกมา หลังจากนั้นให้เลื่อน Scroll bar ลงมาจนถึงเบอร์ที่ต้องการในที่นี้คือเบอร์ AT89C51AC2 หลังจากนั้นคลิก OK
6. ทำการเพิ่มไฟล์เข้าไปยังโปรเจกต์ โดยไฟล์ที่จะเพิ่มต้องเป็นไฟล์ .c ถ้าหากยังไม่มีไฟล์ดังนั้น ให้ทำการสร้างไฟล์ .c ก่อนโดยคลิกเลือกที่ File -> New ถึงขั้นนี้แล้วเราจะเห็นพื้นที่สำหรับใช้ในการเขียนโปรแกรม
7. ทำการเซฟเป็นไฟล์ใหม่เป็นไฟล์ .c โดยคลิกที่ File -> Save as...
8. ถึงขั้นนี้เราได้ทำการเพิ่มไฟล์เข้าไปยังโปรเจกต์เรียบร้อยแล้ว ซึ่งสามารถสังเกตได้จากหน้าต่างของโปรเจกต์ดังรูป
9. จากนั้นก็เป็นขั้นตอนของการเขียนโปรแกรมตามกระบวนการสั่งงานที่ต้องการ
10. ทำการเซฟไฟล์โปรเจกต์ที่เขียนขึ้น เพื่อทำการ Build Target สังเกตที่หน้าต่าง Output Window จะแสดงการรายงานข้อผิดพลาดออกมา



รูปที่ 3.2 แสดงรูปหน้าจอของโปรแกรม Keil

หากไม่มีข้อผิดพลาดเกิดขึ้นแสดงว่าโปรแกรมที่เขียนขึ้นมีความถูกต้องตามรูปแบบของข้อกำหนดต่างๆ ของการเขียนโปรแกรมแล้ว แต่ไม่ได้หมายความว่าโปรแกรมที่เขียนขึ้นนี้จะมีการทำงานถูกต้องตามที่ต้องการเสมอไปเพราะ Compiler จะทำการตรวจสอบเฉพาะรูปแบบของการเขียนโปรแกรมเท่านั้น แต่อย่างไรก็ตามเราสามารถจะตรวจสอบผลการทำงานของโปรแกรมได้โดยการใช้ความสามารถของ Keil ในโหมด Debug ต่อไป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

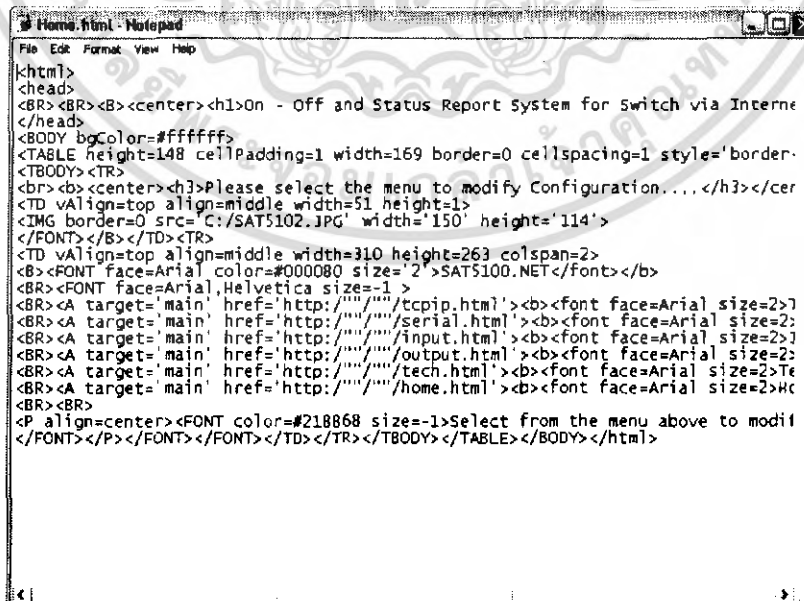


รูปที่ 3.3 แสดงการเขียนโปรแกรมคำสั่งที่ใช้งาน

3.3.2 การสร้าง Website

ขั้นตอนการสร้างเอกสาร HTML ด้วย Notepad

1. เลือก Start > Program > Accessories > Notepad
2. จะปรากฏหน้าต่างต่างของโปรแกรม Notepad
3. ให้คีย์เอกสาร HTML ดังภาพต่อไปนี้



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับรูปที่ 3.4 แสดงการเขียนคำสั่งด้วยภาษา HTML ต่อกำหนดให้หน้าไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4. เลือกเมนู File > save
5. จะปรากฏหน้าต่าง Save As ให้เลือกตำแหน่ง Save in
6. เลือกชนิดของไฟล์ที่ Save as type เป็น All Files
7. ตั้งชื่อไฟล์ HTML ที่ Filename โดยจะต้องให้มีนามสกุลเป็น .html หรือ .htm เท่านั้น
8. คลิก Save
9. เข้าไปดูใน Folder ที่เลือกเป็นตำแหน่งเก็บไฟล์จะปรากฏไฟล์ HTML ที่ได้บันทึกไว้ (สังเกต จะมีสัญลักษณ์ IE อยู่บนหน้าจอ)

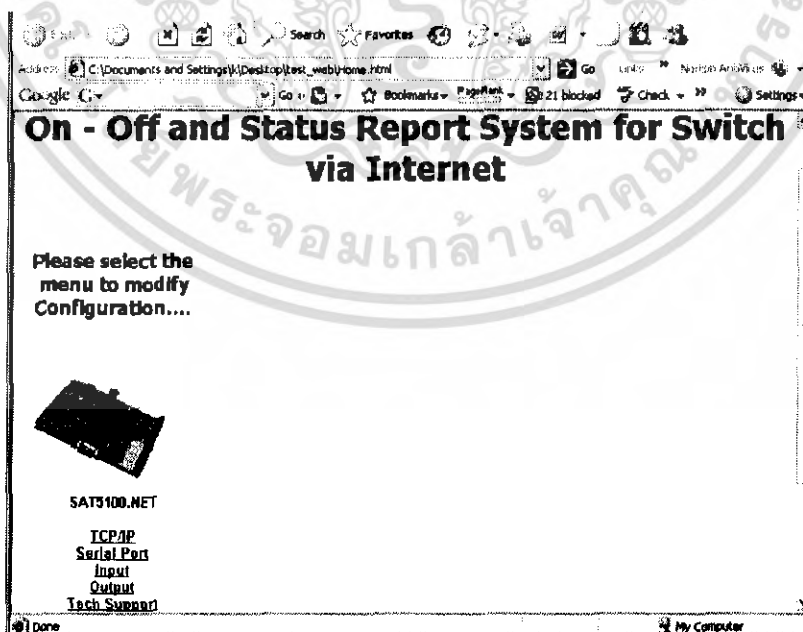
การแสดงผลเอกสาร HTML มี 2 วิธี คือ

วิธีที่ 1. การแสดงผลเอกสาร HTML โดยเปิดผ่าน IE โดยมีขั้นตอนดังนี้

1. ดับเบิลคลิกที่ IE แล้วเลือกเมนู File > Open
2. คลิก Browse
3. เลือกชื่อไฟล์ที่ต้องการเปิด คลิก Open
4. คลิก OK
5. จะปรากฏผลลัพธ์ของไฟล์เอกสาร HTML ผ่าน Web Browser

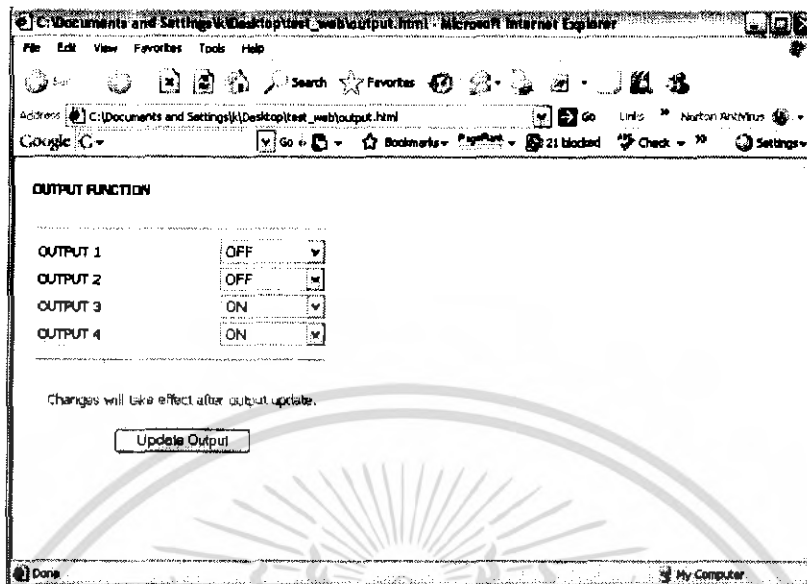
วิธีที่ 2. การแสดงผลเอกสารโดยตรง มีขั้นตอนดังนี้

1. ดับเบิลคลิกที่ชื่อไฟล์เอกสารนั้น
2. จะปรากฏผลลัพธ์ของไฟล์เอกสาร HTML ผ่าน Web Browser ดังรูป



รูปที่ 3.5 แสดงไฟล์เอกสาร HTML ผ่าน Web Browser

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไมอนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.6 แสดงหน้าต่าง Output ควบคุมการทำงานผ่าน Web Browser

3.3.3 การสร้างระบบล็อกอิน

ระบบล็อกอิน คือ ระบบสำหรับตรวจสอบสิทธิ์การเข้าใช้งานเว็บแอปพลิเคชัน หากเว็บแอปพลิเคชันที่เราพัฒนาขึ้นมาเป็นระบบเฉพาะของสมาชิก ผู้เข้าใช้งานเว็บแอปพลิเคชันต้องระบุ Username และ password ก่อนที่จะเข้าไปใช้งานรูปแบบต่างๆ เช่น แสดงข้อมูล เพิ่มข้อมูล แก้ไขข้อมูล หรือลบข้อมูล เป็นต้น ทั้งนี้ขึ้นอยู่กับลักษณะการนำไปประยุกต์ หรือการออกแบบของผู้พัฒนาเว็บ

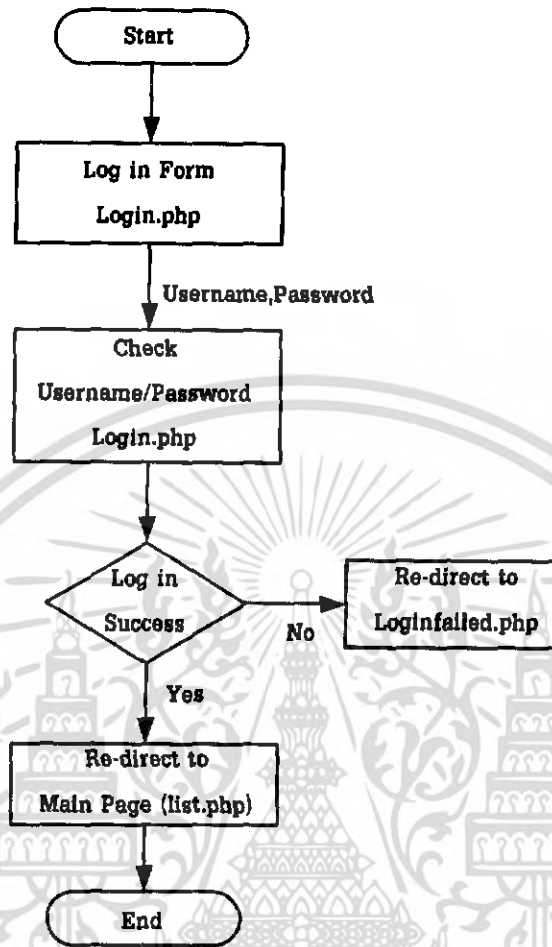
3.3.3.1 การสร้างฟอร์มเพื่อรับค่าการล็อกอิน

ขั้นตอนนี้จะเป็นการสร้างฟอร์มสำหรับการล็อกอิน โดยผู้ใช้ต้องระบุ Username และ password เพื่อตรวจสอบสิทธิ์ในการเข้าใช้งาน โดยจะเป็นการแทรกตารางเพื่อช่วยในการจัดรูปแบบของหน้าเว็บเพจ และแทรกฟอร์มลงไป

3.3.3.2 ขั้นตอนการสร้างฟอร์มเพื่อรับค่าการล็อกอิน

1. เปิดโปรแกรม Dreamweaver MX คลิกเมนู File > New จะปรากฏไดอะล็อก New Document ขึ้นมา ให้คลิกแท็บ General เลือก Dynamic Page > PHP หลังจากนั้นคลิกปุ่ม Create บันทึกเป็นไฟล์ login.php
2. แทรกฟอร์มลงในเว็บเพจ โดยคลิกที่เมนู insert > Form > Form กำหนดชื่อของฟอร์มเป็น frmlogin โดยให้คลิกเลือกแท็ก <form> ที่ Tag Selector บริเวณขอบล่างซ้ายของหน้าจอ หลังจากนั้นให้ไปที่เมนู Window > Properties เมื่อปรากฏไดอะล็อก Properties ให้พิมพ์ชื่อฟอร์มในช่อง Form name

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



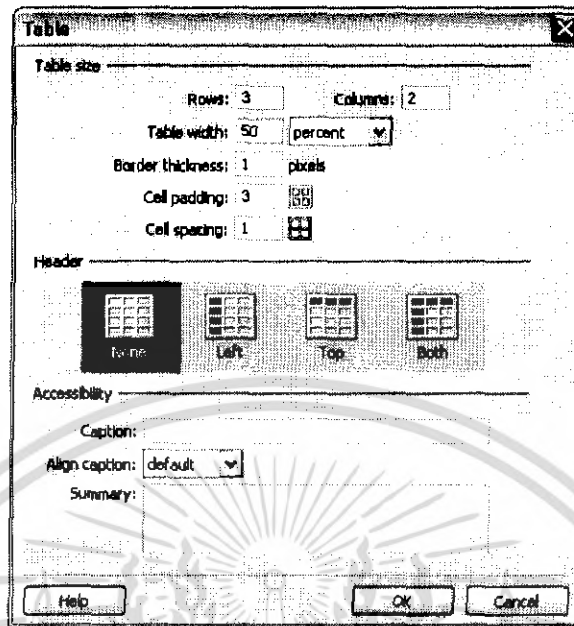
รูปที่ 3.7 แสดง Flow chat ของระบบล็อกอิน



รูปที่ 3.8 แสดงการกำหนด Properties ของฟอร์ม

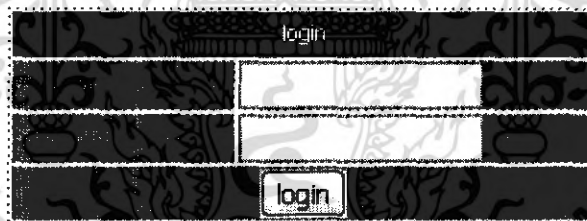
3. แทรกตารางภายใต้เว็บเพจ โดยคลิกที่เมนู Insert > Table และให้กำหนดค่าดังต่อไปนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.9 แสดงการกำหนดค่าของไดอะล็อก Table

4. เมื่อดำเนินการตามขั้นตอนต่างๆ เสร็จแล้วจะได้เว็บเพจต่างๆ ดังรูป



รูปที่ 3.10 แสดงเว็บเพจไฟล์ login.php

3.3.4 การใช้งานโปรแกรม Windows Media Encoder

ก่อนอื่นต้องทำความเข้าใจเกี่ยวกับ Streaming Video เสียก่อนว่ามีไว้เพื่ออะไรถ้าท่านที่เคยใช้ Internet จะเห็นว่าการเปิด Video จาก Internet นั้นเป็นเรื่องที่ยากมากเพราะกว่าจะ Download เอา File ที่เป็น Video ทั้งหมดมาลงเครื่องคอมพิวเตอร์แล้วจึงจะเปิดชมได้ซึ่งความเป็นไปได้ในการเผยแพร่ Video ผ่าน Internet แทบจะเป็นไปไม่ได้เลยสาเหตุที่เป็นเช่นนี้เพราะข้อจำกัดของระบบการติดต่อสื่อสารเช่นการเชื่อมโยง Internet ผ่านสายโทรศัพท์ ที่สามารถส่งข้อมูลได้ที่ประมาณ 28.8 kbps ขณะที่มาตรฐานของ VCD มีอัตราการส่งข้อมูลที่ 1150 kbps ดังนั้นวิธีการที่จะทำได้คือจะต้องส่ง Video ที่ความเร็วต่ำกว่านี้ซึ่งปัจจุบัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ได้สามารถที่จะทำได้ ด้วยเทคโนโลยีที่เรียกว่า Streaming ดังนั้นปัจจุบันเราจึงสามารถดู TV ผ่านทาง Internet ได้หรือฟังวิทยุผ่าน Internet ได้

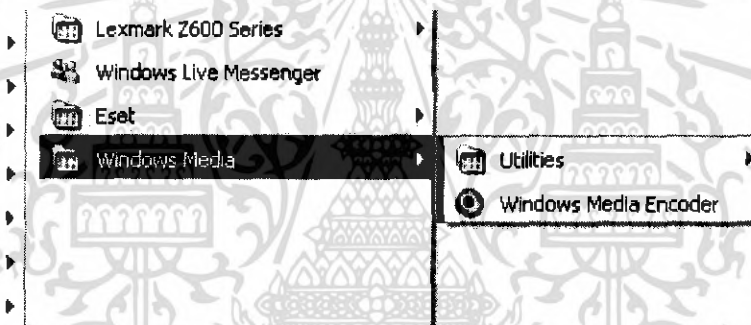
Video ที่เราสร้าง และอยู่ใน Format ของ avi หรือ mpg นี้ไม่สามารถส่งในรูปแบบของ streaming ได้จึงต้องมีการแปลงเสียก่อนซึ่งในการแปลงนั้นมีโปรแกรมหลายโปรแกรมที่ช่วยแปลงแต่ในที่นี้จะใช้โปรแกรมของบริษัท Microsoft คือโปรแกรม Windows Media Encoder ซึ่งมีวิธีการทำดังนี้

3.3.4.1 เตรียม Video File

โดยเตรียมจาก Video ที่เราตัดต่อไว้แล้วแต่ควรเป็น Video ที่มีความยาวไม่มากนักควรทำเป็นตอนสั้นๆเพื่อสะดวกในการเผยแพร่ผ่าน Internet

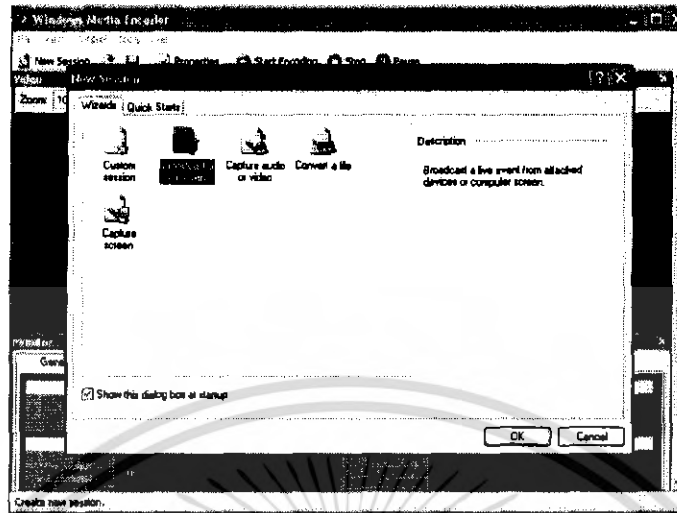
3.3.4.2 เปิดโปรแกรม Windows Media Encoder

เมื่อติดตั้งโปรแกรมเสร็จแล้วเราสามารถเปิดโปรแกรมขึ้นมาใช้งานได้ดังขั้นตอนการเปิดโปรแกรม

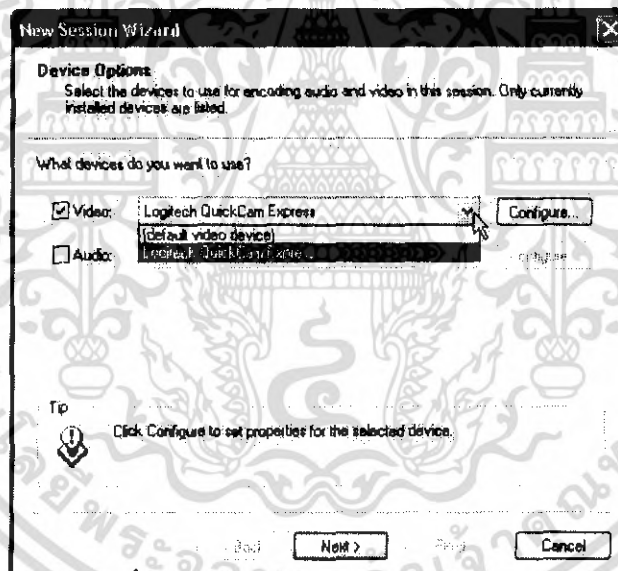


รูปที่ 3.11 รูปแสดงการเปิดโปรแกรม Windows Media Encoder

1. คลิกที่ปุ่ม Start >> Program >> Windows Media จะพบแถบคำสั่ง Windows Media Encoder ดังภาพให้คลิกที่แถบคำสั่ง Windows Media Encoding
2. แสดงการเปิดโปรแกรม Windows Media Encoder 9 แล้วเลือก broadcast a live event



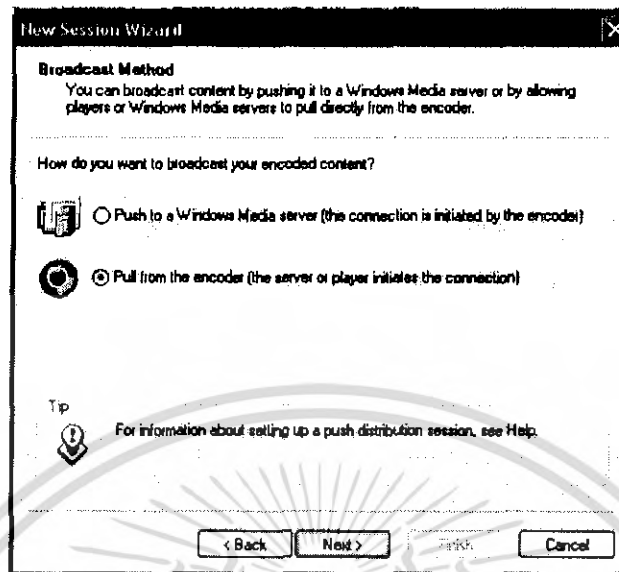
รูปที่ 3.12 ทำการเลือก Broadcast a live event



รูปที่ 3.13 แสดงการเลือกกล้อง

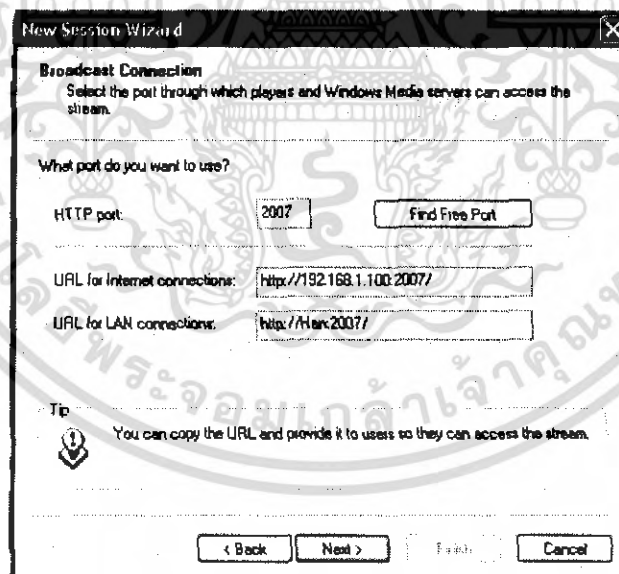
3. เลือกกล้องที่คุณใช้ในการเปิดโปรแกรม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.14 แสดงการเลือก Pull from the encoder

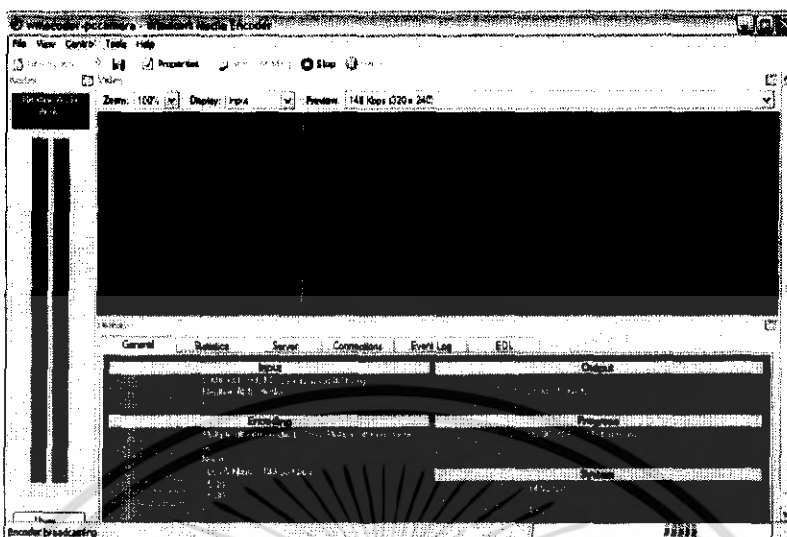
4. เลือก Pull from the encoder



รูปที่ 3.15 แสดงการกำหนด Port

5. กำหนดเลข Port ด้วยตนเองตามตัวอย่างกำหนดเป็น 2007 หรือจะปล่อยให้โปรแกรมช่วยหาให้ก็ได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.18 แสดงหน้าจอการใช้งานของโปรแกรม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

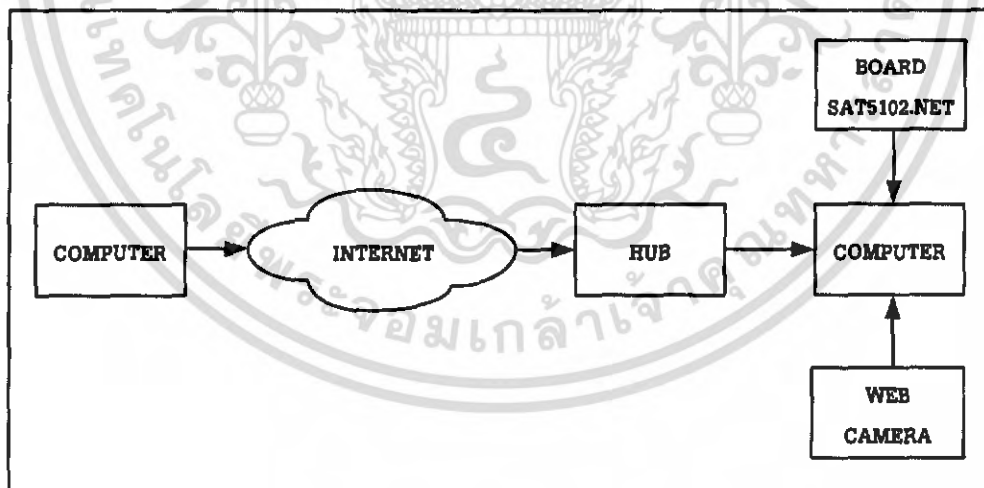
บทที่ 4

การทดลองและผลการทดลอง

4.1 กล่าวนำ

ในส่วนของบทนี้จะกล่าวถึงการทดลองและผลการทดลองของชุดทดลองระบบเปิด - ปิด และรายงานสถานะสวิตช์ผ่านเครือข่ายอินเทอร์เน็ต ไม่ว่าจะเป็นในส่วนของวงจรควบคุม การควบคุมผ่านอินเทอร์เน็ต การแสดงผลของภาพผ่านอินเทอร์เน็ต และในส่วนของวงจรประยุกต์ต่างๆ ซึ่งจะใช้โปรแกรมเป็นตัวทดลองแผงวงจรต่างๆ

จากรูปเป็นลักษณะการเชื่อมต่อโดยรวมของระบบเปิด - ปิด และรายงานสถานะสวิตช์ผ่านเครือข่ายอินเทอร์เน็ต ซึ่งการเรียกใช้งานสามารถกระทำได้ดังนี้ จากเครื่องคอมพิวเตอร์เราสามารถเรียกใช้งานโดยผ่านโปรแกรมอินเทอร์เน็ตซึ่งจะมี HUB เป็นอุปกรณ์ในการเชื่อมต่อเข้ากับระบบอินเทอร์เน็ต และเชื่อมต่อเข้ากับคอมพิวเตอร์อีกเครื่องที่ใช้ในการควบคุม ซึ่งชุดของคอมพิวเตอร์ตัวนี้จะมีบอร์ด SAT5102.NET และ WEB CAMERA เชื่อมต่ออยู่ให้เราสามารถสั่งการควบคุมการเปิด - ปิด เครื่องใช้ไฟฟ้า และดูภาพการใช้งานในสถานที่ปัจจุบันที่เราสั่งใช้งานได้



รูปที่ 4.1 แสดงการเชื่อมต่อใช้งานของระบบโดยรวม

4.2 การควบคุมอุปกรณ์ผ่านระบบเน็ตเวิร์ก

4.2.1 การกำหนดค่าและการใช้งานของ Ethernet

เอกสารนี้เป็น มี 2 ส่วน คือ ICMP, ARP, IP และ MAC ศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ในการกำหนด IP และ MAC ADDRESS นั้นสามารถทำได้ครั้งเดียวในการโปรแกรมให้กับ CPU จะสามารถแก้ไขได้ก็ต่อเมื่อมีการแก้ไขในส่วนของ SOURCE CODE ก่อนทำการโปรแกรมลงบนตัว CPU และจะใช้ IP ที่กำหนดนี้ทำการคอนเน็คผ่านระบบ LAN

ลักษณะของ IP จะมีด้วยกัน 8 BYTE [FF-FF-FF-FF] ถ้าเรามองเป็นเลขฐานสิบจะได้ [255.255.255.255] ตัวอย่างเรากำหนด IP: 192.168.1.2 ไว้ที่เครื่อง PC แล้ว เราก็กำหนดที่บอร์ดคอนโทรลเลอร์ให้เป็น IP: 192.168.1.3 ก็ได้แต่ไม่ซ้ำกับ IP ของเครื่องอื่น

ลักษณะของ MAC ADDRESS จะมีด้วยกัน 16 BYTE [FF-FF-FF-FF-FF-FF] จะเป็นค่าเฉพาะของฮาร์ดแวร์แต่ละตัวจะไม่ซ้ำกันในส่วนของเครื่องขายนั่นเช่น กำหนด MAC: 00-00-43-4E-4F-45 เครื่องอื่นก็ต้องไม่ใช่ค่านี้

ICMP หรือ PING เป็น PACKET ของ IP ADDRESS ตัวหนึ่ง ที่ส่วนใหญ่เรานำมาเช็คค่าต่างๆ ในการติดต่อการตอบกลับของบอร์ดคอนโทรลเลอร์ว่าสามารถเชื่อมต่อได้หรือไม่ เวลาในการติดต่อจำนวนครั้งในการรับ - ส่งข้อมูล เป็นต้น

ARP เป็น PACKET ของ IP ADDRESS และ MAC ADDRESS เหมือนกัน ใช้ในการตรวจสอบ IP, MAC, และ TYPE ของเครื่องและบอร์ดคอนโทรลเลอร์ และสามารถแก้ไขการตรวจสอบค่า MAC โดยการกำหนดจากเครื่อง PC ได้ แต่เราต้องแก้ไขกลับมาเหมือนเดิมเมื่อเราทดสอบเสร็จไม่เช่นนั้นเครื่องจะจำค่าตรวจสอบนั้นไว้

การต่อสาย LAN ของเครื่อง PC กับบอร์ดคอนโทรลเลอร์ถ้าต่อโดยไม่ผ่าน HUB หรือ SWITCH ให้ใช้สายแบบไขว้ และถ้าผ่าน HUB หรือ SWITCH ให้ใช้สายปกติ

4.2.2 อินพุตและเอาต์พุตของบอร์ดคอนโทรลเลอร์

อินพุต 4 พอร์ต (OPTO INPUTS) และมี LED แสดงสถานะของอินพุตบนบอร์ดคอนโทรลเลอร์ ให้ลอจิก 0 เมื่อ OPTO ทำงาน (ACTIVE LOW)

เอาต์พุต 4 พอร์ต (RELAY OUTPUTS) และมี LED แสดงสถานะของอินพุตบนบอร์ดคอนโทรลเลอร์ RELAY จะทำงานเมื่อป้อนลอจิก 1 ไม่ทำงานเมื่อเป็นลอจิก 0

4.2.3 สิ่งที่ต้องการในการพัฒนาบอร์ดคอนโทรลเลอร์

1. ต้องการไฟกระแสตรง 12 โวลต์ 300 มิลลิแอมป์
2. พอร์ตอนุกรม (RS-232) ใช้สายมาตรฐาน D-SUB 9 PIN แบบตัวผู้ตัวเมีย
3. การเชื่อมต่อทางพอร์ตอินเตอร์เน็ตใช้สาย UTP แบบไขว้ หัวเป็นแบบ RJ-45
4. การโปรแกรม CPU ผ่าน PORT RS-232 เพื่อสะดวกในการพัฒนา
5. คอมไพเลอร์ในการเขียนโปรแกรมใช้ภาษาซี
6. คอมพิวเตอร์เพื่อใช้ในการเขียนโปรแกรม และทดสอบการใช้งาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.2.4 การเชื่อมต่อของไมโครคอนโทรลเลอร์กับไอซีอินเตอร์เน็ต

1. DATA INPUT/OUTPUT 8 บิต
2. ADDRESS 5 บิต
3. LED แสดงสถานะในตัวชิพอินเตอร์เน็ต 3 ดวง (LINK, TXD, RXD)
4. การเชื่อมต่อผ่าน LAN แบบ FULL DUPLEX 10 BASE-T

4.2.5 การกำหนดค่าทางอินเตอร์เน็ต

1. IP ADDRESS ให้ใช้ IP: 192.168.1.2
2. MAC ADDRESS ให้ใช้ MAC: 255.255.255.0
3. กำหนด PORT เท่ากับ PORT: 23
4. การสั่งงานเอาต์พุตจากเครื่อง PC ให้เป็นแบบเวลาจริง
5. การอ่านค่าอินพุตให้ PC ถามทุกๆ 10 วินาที

4.2.6 การกำหนดโปรโตคอล DATA PROTOCOL

เพื่อให้เป็นข้อกำหนดในการติดต่อสั่งงานระหว่างคอมพิวเตอร์กับบอร์ดไมโครคอนโทรลเลอร์ ใช้โปรโตคอล TCP ETHERNET PROTOCOL

ให้ RP เป็นการถามสถานะของโปรแกรม

ให้ RxxxxPxxxx เป็นการตอบของคอนโทรลเลอร์มีการสั่งงานหรือการถามทุกครั้งเพื่อนำค่า R และ P มาบอกสถานะบนจอคอมพิวเตอร์

4.2.7 คำสั่งในการควบคุมการทำงานของ RELAY OUTPUTS

A1 = Relay 1 ON, B1 = Relay 1 OFF

A2 = Relay 2 ON, B2 = Relay 2 OFF

A3 = Relay 3 ON, B3 = Relay 3 OFF

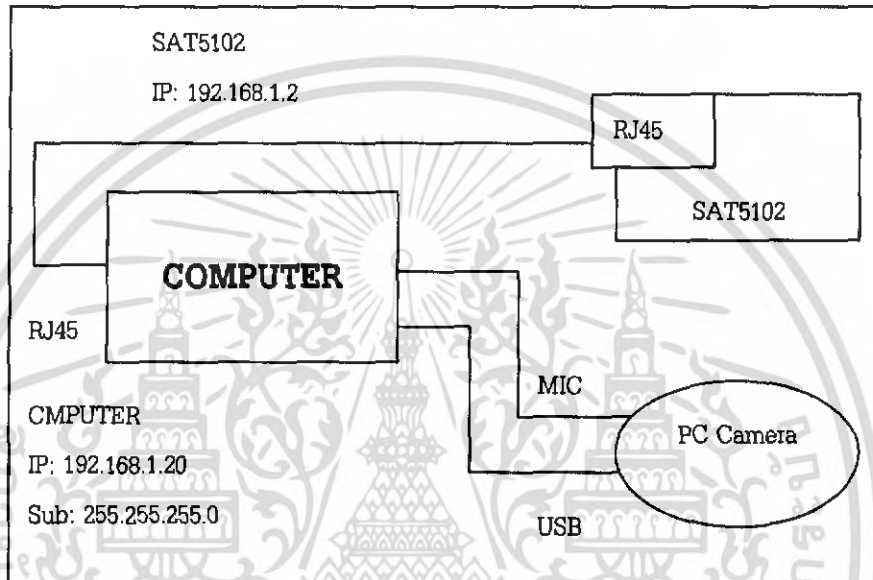
A4 = Relay 4 ON, B4 = Relay 4 OFF

4.3 การทดลองการทำงานผ่านระบบอินเตอร์เน็ต

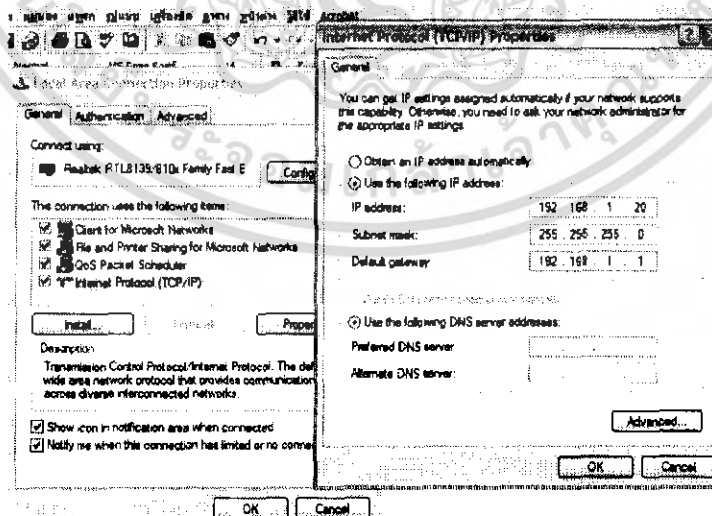
การทำงานของระบบเปิด - ปิดและรายงานสถานะสวิทช์ผ่านระบบอินเตอร์เน็ตจะสามารถมีการทำงานได้นั้นจะต้องทำการสร้างโฮมเพจขึ้นมาก่อนเพื่อใช้ในการควบคุมการเปิด - ปิดของเครื่องใช้ไฟฟ้าให้สามารถใช้งานตามที่เราต้องการ ดังนั้นในการทดลองการทำงานผ่านระบบอินเตอร์เน็ตสามารถทดลองได้ดังนี้

4.3.1 การควบคุมบอร์ด SAT5102.NET

1. ทำการต่อเชื่อมอุปกรณ์ดังรูปที่ 1 เซต ให้บอร์ด SAT5102 มีค่า IP เท่ากับ 192.168.1.2/24 จากนั้นทำการเซตให้คอมพิวเตอร์มีค่า IP:192.168.1.20/24 จากนั้นทำการต่อเชื่อมสายแบบไขว้ระหว่างบอร์ดกับคอมพิวเตอร์



รูปที่ 4.2 การเชื่อมต่ออุปกรณ์ในการทดลองการควบคุมบอร์ด SAT5102



รูปที่ 4.3 การตั้งค่า IP ให้กับคอมพิวเตอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2. ทำการทดสอบโดยใช้คำสั่ง ping

```

Microsoft Windows [C:\Documents and Settings\Administrator\My Documents]
C:\Documents and Settings\Administrator>ping 192.168.1.2

Pinging 192.168.1.2 with 32 bytes of data:

Reply from 192.168.1.2: byte = 32 time = 20 ms TTL = 128
Reply from 192.168.1.2: byte = 32 time = 20 ms TTL = 128
Reply from 192.168.1.2: byte = 32 time = 20 ms TTL = 128
Reply from 192.168.1.2: byte = 32 time = 20 ms TTL = 128

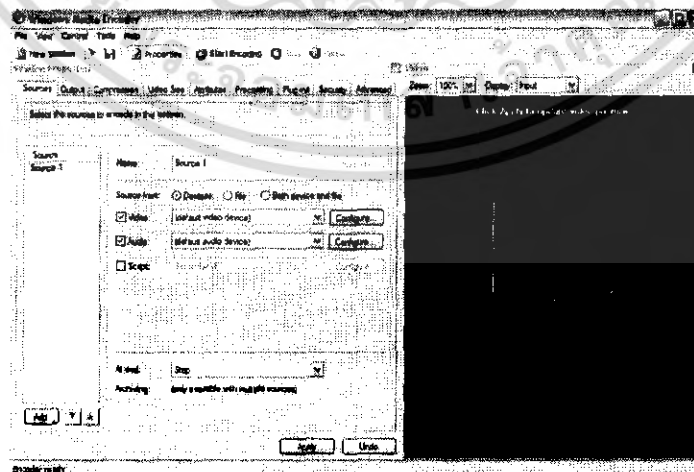
Ping statistics for 192.168.1.2:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milliseconds:
        Minimum = 20ms, Maximum = 20ms, Average = 20ms

C:\Documents and Settings\Administrator>
  
```

รูปที่ 4.4 ผลการใช้คำสั่ง ping เพื่อทดสอบการเชื่อมต่อบอร์ดและคอมพิวเตอร์ผ่านสาย LAN

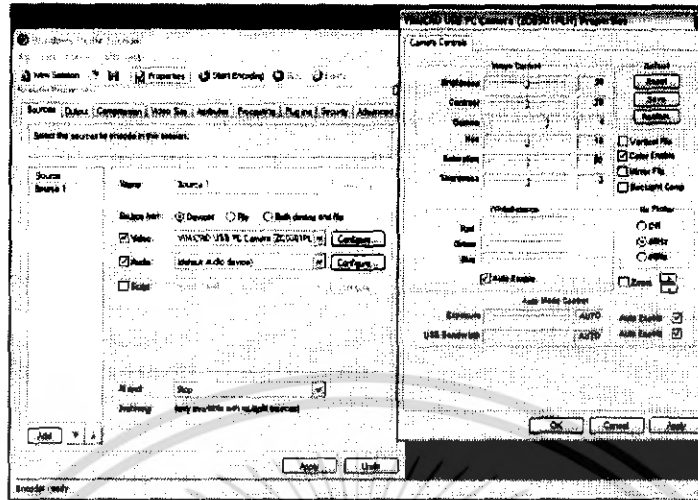
4.3.2 การติดต่อกับกล้อง Web Camera

1. ทำการติดตั้งกล้องตามรูปที่ 4.1
2. จากนั้นทำการติดตั้งโปรแกรม Windows Media Encoder
3. ทดสอบการทำงานของโปรแกรม windows media encoder
4. เลือกการตั้งค่าเอง
5. ตั้งค่า Video โดยเลือกชนิดของกล้องให้ถูกต้อง
6. ตั้งค่าการเชื่อมต่อโดยเลือก PORT ที่วางอยู่ในที่นี้เลือกที่ค่า 2500
7. ตั้งค่า Compression ไว้ที่ 141 Kbps และ 58 Kbps ซึ่งเหมาะสมโดยที่ 141 Kbps เหมาะกับผู้ใช้ต่อผ่านระบบ LAN และ 58 Kbps เหมาะกับการเชื่อมต่อกับ Modem

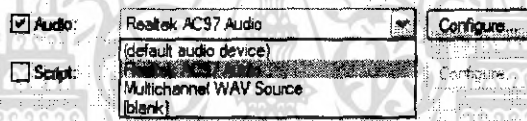


รูปที่ 4.5 การตั้งค่าของโปรแกรม Windows Media Encoder

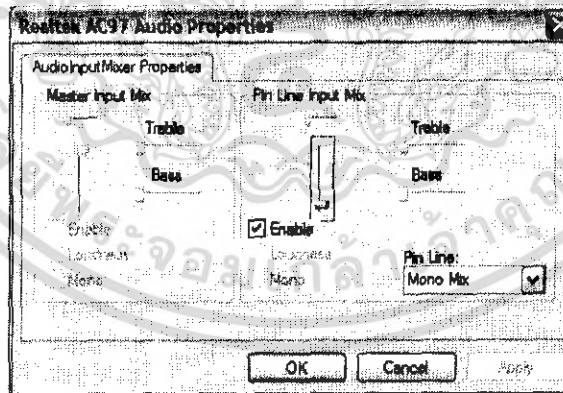
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่นอนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.6 การตั้งค่าโปรแกรม Windows Media Encoder

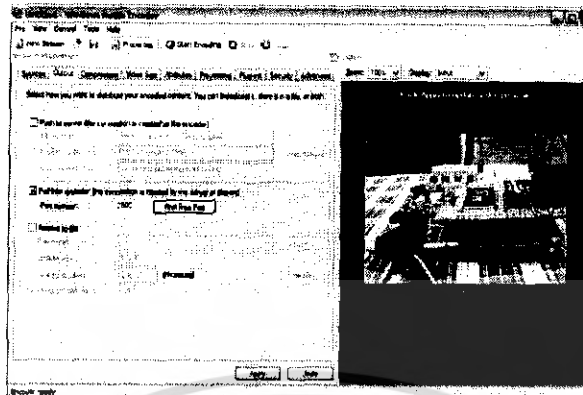


รูปที่ 4.7 การตั้งค่าวิดีโอ

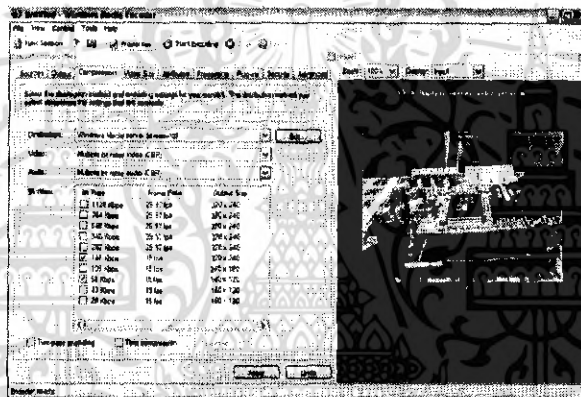


รูปที่ 4.8 การตั้งค่าวิดีโอ

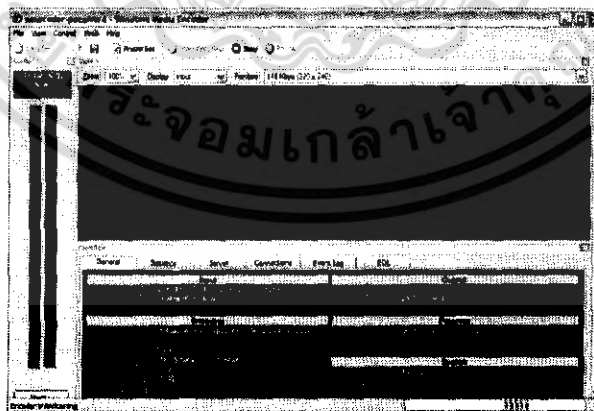
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.9 การเชื่อมต่อผ่านพอร์ตที่วางอยู่



รูปที่ 4.10 การตั้งค่า Compression

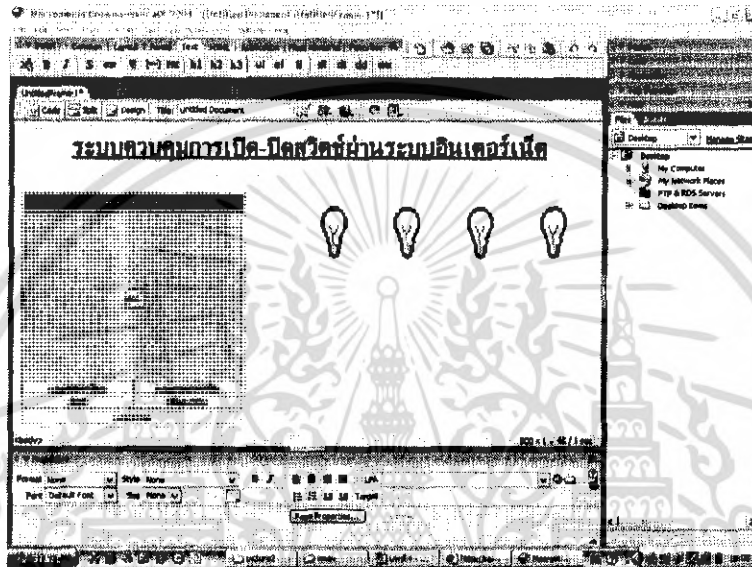


รูปที่ 4.11 การทดสอบการทำงานของกล้อง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.3.3 การทดลองผ่านระบบอินเทอร์เน็ต

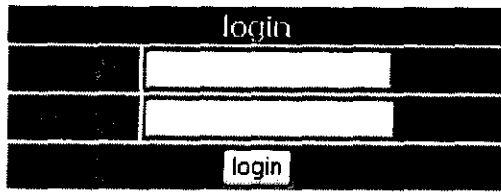
จากการศึกษาภาษา HTML ซึ่งเป็นภาษาที่สร้าง Web Page ที่มีขั้นตอนการสร้างไม่ยากและใช้โปรแกรมในการสร้างเอกสาร HTML ได้ซึ่งการสร้างดังกล่าวได้ใช้โปรแกรม Macromedia Dreamweaver MX ซึ่งเมื่อทำการออกแบบเสร็จก็ทำการแสดงเอกสารบนหน้า Web Browsers ได้ทันที



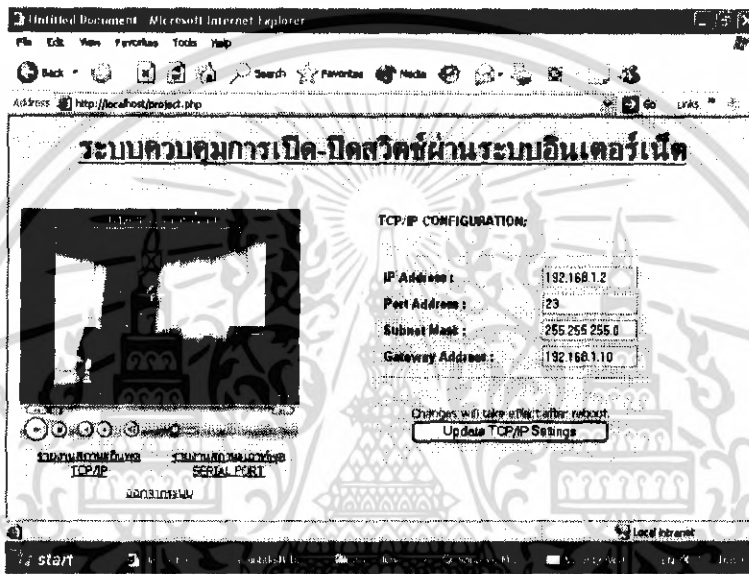
รูปที่ 4.12 การสร้างหน้าเว็บเพจด้วยโปรแกรม Macromedia Dreamweaver MX

1. เปิดโปรแกรม Internet Explorer ใส่ URL: <http://localhost/Project.php>
2. จะเข้าสู่หน้าแรกของ Web Page จากนั้นให้เราทำการควบคุมผ่านหน้า Web Browsers นี้ได้โดย
3. เมื่อเข้าสู่หน้าแรกจะเป็นหน้าการล็อกอินเข้าสู่ระบบการควบคุมการทำงานบนหน้าเว็บแอปพลิเคชัน
4. เมื่อเข้าสู่การล็อกอินแล้วก็เป็นส่วนของหน้าต่างของการปรับแต่ง IP Address ให้ตรงกับค่าที่ใช้งาน ซึ่งในที่นี้ไม่ต้องปรับแต่งใดๆ ทั้งสิ้น
5. ในส่วนของเมนูอื่นๆ ที่อยู่ด้านขวามือเราสามารถคลิกเพื่อเชื่อมต่อไปยังส่วนต่างๆ ได้ไม่ว่าจะเป็นส่วนของการดูภาพการทำงานของเครื่องใช้ไฟฟ้า และส่วนของการควบคุมการเปิด - ปิดของเครื่องใช้ไฟฟ้า

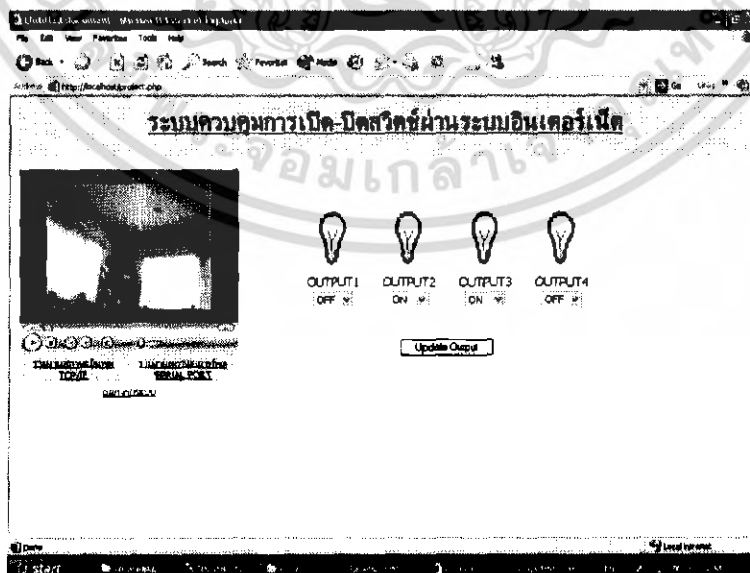
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.13 การล็อกอินเข้าสู่หน้าควบคุมการทำงาน



รูปที่ 4.14 หน้าต่างแสดงการทำงานของ TCP/IP



รูปที่ 4.15 หน้าต่างการควบคุมเอาต์พุต

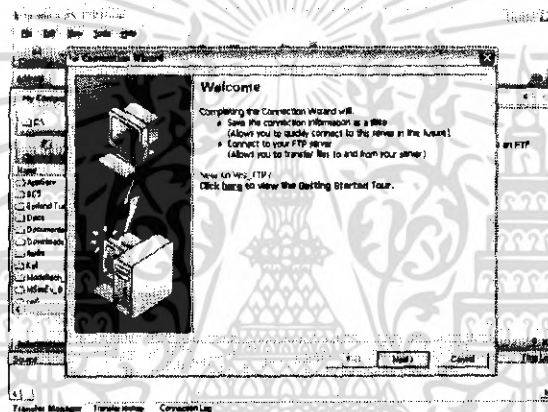
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับ... อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า... ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.3.4 การทดสอบไฟล์ผ่านโปรแกรม Ws_ftp

นอกจากการจัดการไฟล์ผ่านทาง File Manager ที่ไม่สะดวกเท่าไรนักในการจัดการโอนถ่ายไฟล์ที่มีข้อจำกัดในการส่งไฟล์ได้เพียงครั้งละ 3 ไฟล์ แล้วก็ยังมียกวิธีหนึ่งซึ่งสะดวกรวดเร็วกว่านั่นก็คือการจัดการไฟล์ทางโปรแกรม Ws_ftp นั่นเองครับ

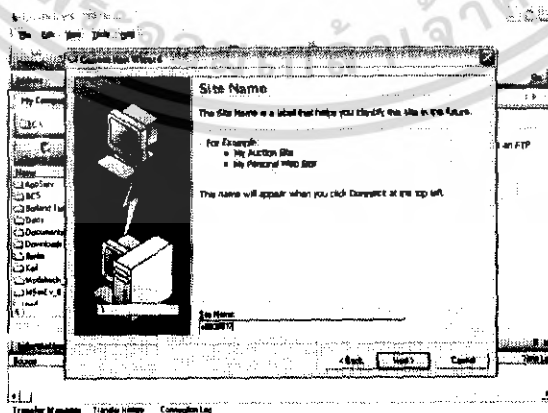
การใช้งานโปรแกรม Ws_ftp เพื่อจัดการไฟล์ข้อมูลระหว่างเครื่องของท่านกับ Hosting มีขั้นตอนการใช้งานดังนี้

1. ติดตั้งโปรแกรม Ws_ftp ให้เรียบร้อย แล้วเปิดโปรแกรมขึ้นมา
2. ตั้งค่าการเชื่อมต่อโดยคลิกที่ Connect จากนั้นคลิกที่ Create Site



รูปที่ 4.16 การใช้งานโปรแกรม Ws_ftp

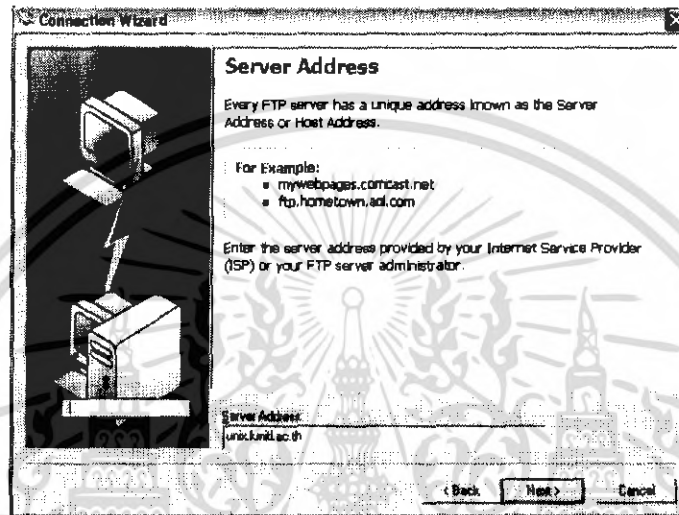
3. ในช่อง Site Name ตั้งชื่อการเชื่อมต่อจากนั้นกด Next



รูปที่ 4.17 การใส่ชื่อผู้ใช้งาน

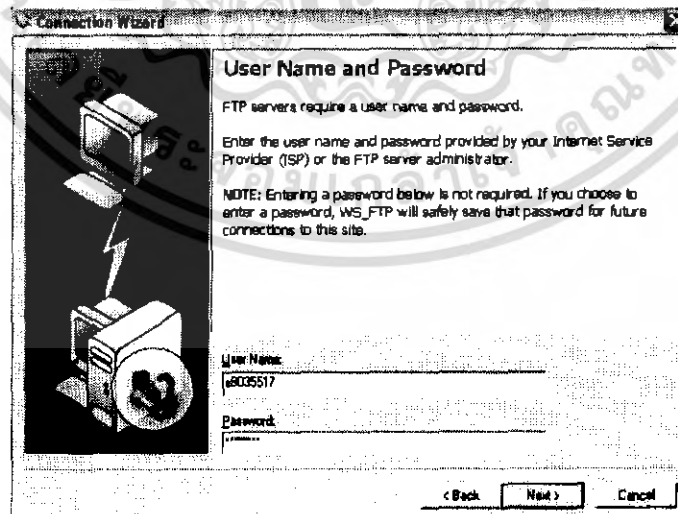
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4. ใส่ชื่อผู้ใช้งาน และรหัสผ่าน การเข้าสู่ ftp (เบื้องต้นใช้ค่าที่ท่านได้จากเจ้าหน้าที่ดูแลระบบ) จากนั้นกด Next
5. ต่อมาในช่อง Server Address ใส่ Host Address โดยกำหนดค่าเป็น unix.kmitl.ac.th จากนั้นกด Next



รูปที่ 4.18 การใส่ชื่อใน Server Address

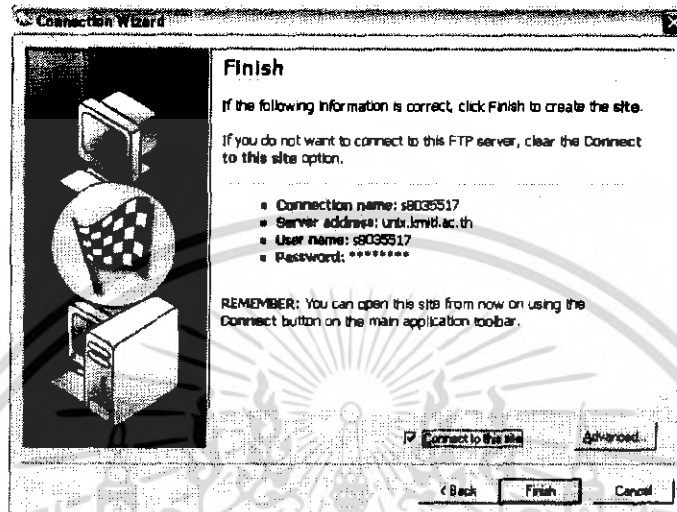
6. ต่อมาในช่อง Connection Wizard ให้ใส่ชื่อผู้ใช้และรหัสผ่านของท่าน



รูปที่ 4.19 การ User Name และ Password

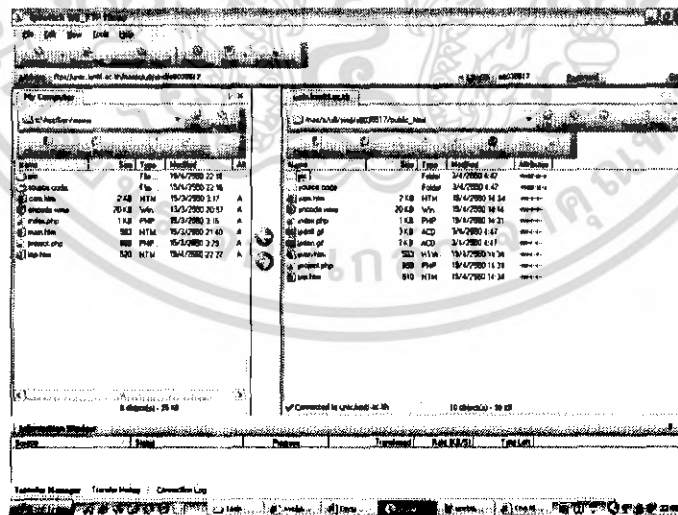
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

7. โปรแกรมจะขึ้นหน้าสรุปรวมค่าที่ตั้งมาทั้งหมดให้ จะมีช่อง Checkbox Connect to this site ถ้าติ๊กถูกเมื่อกด Finish จะเชื่อมต่อกับ Hosting



รูปที่ 4.20 แสดง User Name และ Password เพื่อตรวจสอบความถูกต้อง

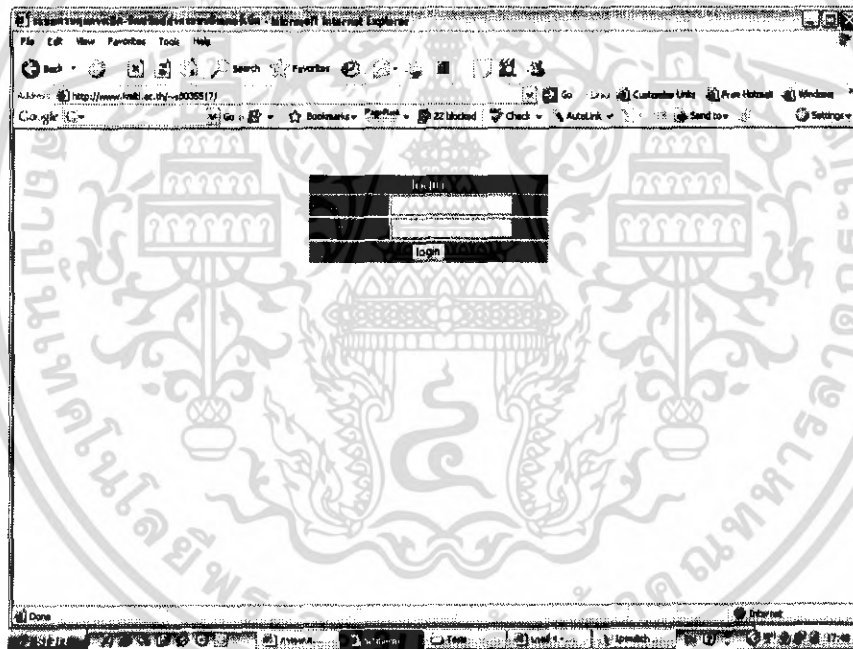
8. ทันที คลิกที่ Finish เพื่อจบการตั้งค่าเสร็จแล้วคลิกที่เครื่องหมายลูกศรลงที่ Connect ก็จะมีข้อการเชื่อมต่อที่เราได้ตั้งไว้ คลิกเลือกเลยครบเมื่อโปรแกรมเชื่อมต่อสำเร็จก็จะเป็นดังนี้



รูปที่ 4.21 การเข้าสู่โปรแกรม Ws_ftp สำหรับการอัปโหลดไฟล์

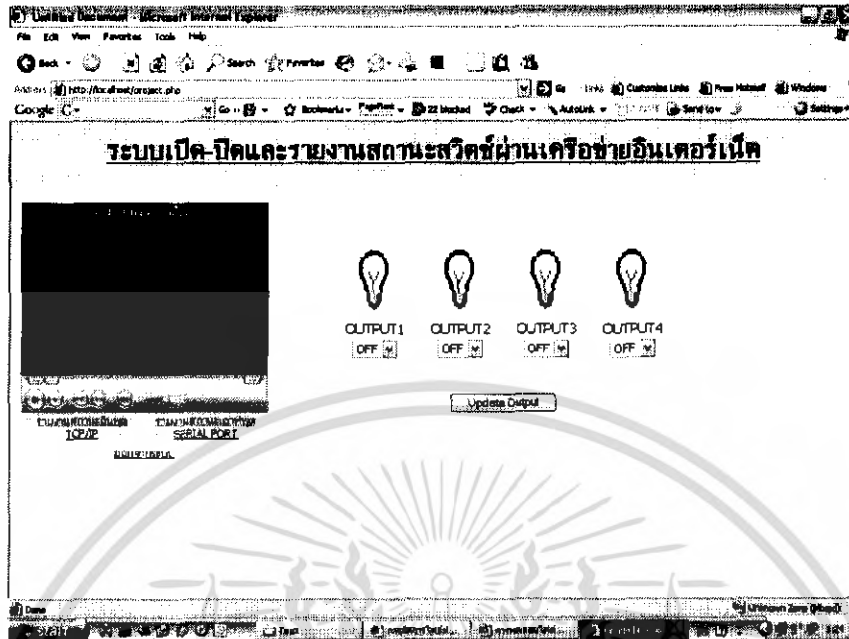
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

10. ทางด้านซ้ายคือข้อมูลในเครื่องคอมพิวเตอร์ของเรา ส่วนทางด้านขวาคือฝั่ง Server ให้นำไฟล์ทั้งหมดไปไว้ที่โฟลเดอร์ public_html ครับ
11. ให้สังเกตที่ตรงกลางจะมีลูกศรที่ชี้ไปทั้งสองฝั่ง ถ้าต้องการอัปโหลดไฟล์จากเครื่องคอมพิวเตอร์ของคุณก็เลือกไฟล์หรือโฟลเดอร์และกำหนดที่ฝั่ง Server ว่าจะ Upload ไฟล์หรือโฟลเดอร์ไปไว้ที่ใด แล้วกดลูกศรที่หันไปทางฝั่ง Server ครับ กรณีที่ต้องการ Download ไฟล์จากฝั่ง Server มาเก็บไว้ที่เครื่องคอมพิวเตอร์ของคุณก็ให้ทำกลับกันครับ คือเลือกไฟล์หรือโฟลเดอร์ทางฝั่ง Server แล้วกำหนดที่ฝั่งเครื่องคอมพิวเตอร์ของคุณว่าจะเก็บข้อมูลไว้ที่ใด แล้วกดลูกศรที่หันไปทางฝั่งเครื่องคอมพิวเตอร์ของคุณ
12. จากนั้นทำการทดสอบโดยเข้าโปรแกรมอินเทอร์เน็ตเพื่อพิมพ์ URL ให้เข้าสู่หน้าต่างการควบคุมผลงาน



รูปที่ 4.22 แสดงหน้าต่างการล็อกอินบนหน้าเว็บเบราว์เซอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.23 หน้าต่างควบคุมการสั่งงานบนหน้าเว็บเบราว์เซอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 5

บทสรุป

5.1 สรุป

ระบบเปิด - ปิดและรายงานสถานะสวิตช์ผ่านระบบอินเทอร์เน็ตเป็นระบบที่ทำการควบคุมสวิตช์ อุปกรณ์ไฟฟ้า 4 จุด ผ่านระบบอินเทอร์เน็ตแล้วทำการรายงานสถานะของทำงานสวิตช์กลับมายังวงจรรายงานสถานะและสามารถตรวจสอบภาพการทำงานของสวิตช์ได้ด้วยกล้อง Web cam ผ่านระบบอินเทอร์เน็ต

5.2 ปัญหาและวิธีการแก้ไข

จากการดำเนินการสร้างและทดสอบโครงงานปรากฏว่าปัญหาที่เกิดขึ้นสามารถสรุปได้คือ

1. ความล่าช้าในการออกแบบ ลายวงจรเนื่องจาก CPU ที่ใช้เป็นเบอร์ T89C51AC2 จึงมีขนาดของเส้นของลายทองแดงที่มีต่อการใช้งานกับขา CPU ที่ขนาด 8 mil จึงต้องใช้ความละเอียดมากในการออกแบบและในการนำเนินแผ่น PCB ต้องส่งมอบให้กับบริษัทที่มีเครื่องมือที่ทันสมัยสามารถรองรับการทำเส้นวงจรทองแดงที่มีขนาดเล็กได้
2. ในการส่งผ่านสวิตช์ค่าที่ส่งออกไปยัง Output ให้สวิตช์ทำงานนั้นค่านั้นต้องเป็น 1 เพื่อให้สวิตช์ทำงานและต้องทำการส่งค่า 1 เข้ามายังวงจรรายงานสถานะเมื่อให้ทำการตรวจสอบค่าการทำงานดังกล่าวว่าเป็นจริงหรือไม่ในส่วนของวงจรการตรวจสอบค่าในที่นี้ได้มีการใช้งานอุปกรณ์ OPB17 ในการตรวจสอบค่าดังกล่าว ซึ่งวงจรในการตรวจสอบค่าที่ได้มีการออกแบบไว้ในตอนต้นไม่สามารถตรวจสอบค่าได้ มีการต่อใช้งานผ่านอุปกรณ์ที่ใช้ไฟเลี้ยงกระแสสลับ จึงต้องมีการสลับลายวงจรที่ทำการป้องกันและทำการแปลงค่าส่งกลับเข้ามายังวงจรรายงานสถานะและในส่วนของอุปกรณ์จึงต้องมีความสัมพันธ์กับชุดของวงจรรายงานสถานะดังกล่าวเข้าด้วยกัน

5.3 แนวทางการพัฒนา

1. ระบบเปิด - ปิดและรายงานสถานะสวิตช์ผ่านระบบอินเทอร์เน็ตในส่วนของวงจรควบคุมได้มีการรองรับการควบคุมการเปิด - ปิดสวิตช์และรายงานสถานะสวิตช์ไว้ 4 จุด หากต้องการเพิ่มจุดการควบคุมการทำงานของสวิตช์ให้ได้มากยิ่งขึ้นก็ต้องทำการขยายอุปกรณ์การทำงานในวงจรในส่วนของภาค Output และภาคการรับสถานะเพื่อเป็นการเพิ่มขีดความสามารถรองรับการทำงานได้มากยิ่งขึ้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2. การตรวจสอบสถานะของสวิตช์ผ่านกล้อง Webcam หากจะนำไปใช้งานให้มีความคล่องตัว และสะดวกให้มีประสิทธิภาพควบคุมการทำงานให้ได้มากยิ่งขึ้นควรเปลี่ยนมาเป็นกล้องแบบ Wire less และทำการสร้างฐานข้อมูลเพื่อที่จะสามารถตรวจสอบการทำงานย้อนหลังได้



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

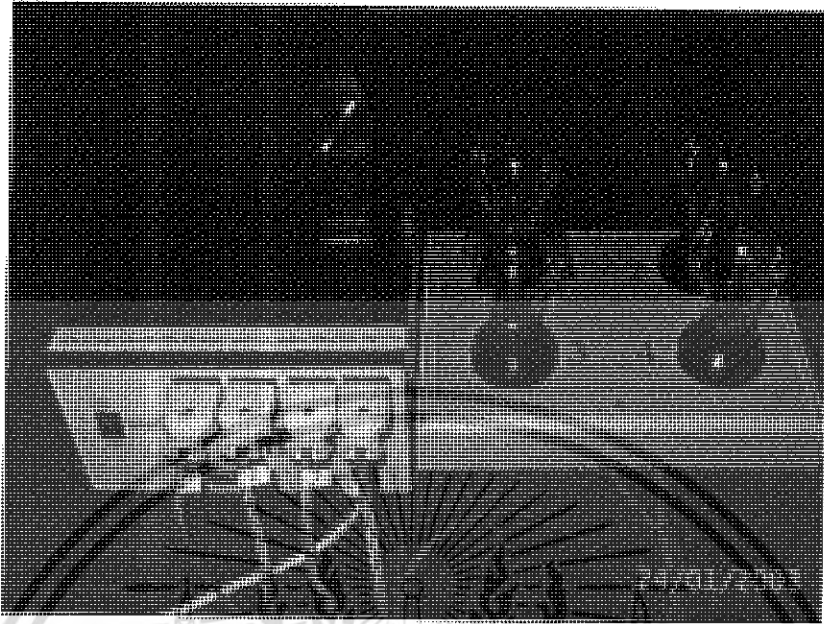
บรรณานุกรม

- คมสันต์ ประพันธ์เทวา และจตุเดช วิวัฒน์สมวงศ์. 2545. "ระบบเครือข่ายสำหรับระบบรักษาความปลอดภัยในหมู่บ้าน." ปรินิพนธ์วิศวกรรมศาสตรบัณฑิต สาขาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์, สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง.
- ฉัททวุฒิ พิษผล,พิชิต สันติกุลานนท์ และพร้อมเลิศ หล่อวิจิตร. 2547. **Visual Basic 6**. กรุงเทพฯ: ไพรวิชั่น.
- ชาญวัฒน์ ขาญเวชช์,ณัฐ จุฑะวนิช และวิมุกต์ เศรษฐนันท์. 2546. "ระบบกล้องโทรทัศน์วงจรปิดรักษาความปลอดภัยผ่านอินเทอร์เน็ต." ปรินิพนธ์วิศวกรรมศาสตรบัณฑิต สาขาวิศวกรรมโทรคมนาคม คณะวิศวกรรมศาสตร์, สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง.
- ชนัย ชัดดีนันทน์,ธีระ ศิริเจริญ และธีระชัย สาดราวาหะ. 2547. "ระบบกล้องรักษาความปลอดภัยผ่านอินเทอร์เน็ต." ปรินิพนธ์วิศวกรรมศาสตรบัณฑิต สาขาวิศวกรรมโทรคมนาคม คณะวิศวกรรมศาสตร์, สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง.
- นิรุช อำนวยศิลป์. 2546. **คู่มือเขียนโปรแกรมด้วยภาษาซี**. กรุงเทพฯ : ไพรวิชั่น.
- ประภาพร ช่างไม้. 2548. **สร้างเว็บสวยด้วย Dreamweaver MX 2004**. นนทบุรี : ไอดีซี.
- พงษ์ศักดิ์ จันทรพิทักษ์,ศิริพร คิดการดี และสลิพงษ์ กิมจำปา. 2547. "ระบบกล้องโทรทัศน์วงจรปิดรักษาความปลอดภัยผ่านอินเทอร์เน็ต." ปรินิพนธ์วิศวกรรมศาสตรบัณฑิต สาขาวิศวกรรมโทรคมนาคม คณะวิศวกรรมศาสตร์, สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง.
- ภิญญา กำเนิดหล่ม. 2546. **เขียน Webpage ด้วย HTML**. กรุงเทพฯ : บุ๊คเซ็นเตอร์.
- วุฒิมณ ไชยศรีชลธาร,ศุภชัย ปิยะพรสกุล และศุภชาญ รุ่งอรุณแสงชัย. 2544. "ระบบรักษาความปลอดภัยและควบคุมเครื่องใช้ไฟฟ้าผ่านโครงข่ายอินเทอร์เน็ต." ปรินิพนธ์วิศวกรรมศาสตรบัณฑิต สาขาวิศวกรรมโทรคมนาคม คณะวิศวกรรมศาสตร์, สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง.
- สันติ นุราช และอุกฤษฏ์ ดันทสุทธานนท์. [ม.ป.ป.]. **เรียนรู้ไมโครคอนโทรลเลอร์ MCS-51**. ปทุมธานี. [ม.ป.พ.].
- โอบาส เอี่ยมสิริวงศ์. 2549. **เครือข่ายคอมพิวเตอร์และการสื่อสาร**. กรุงเทพฯ : เอชเอ็นกรุ๊ป.

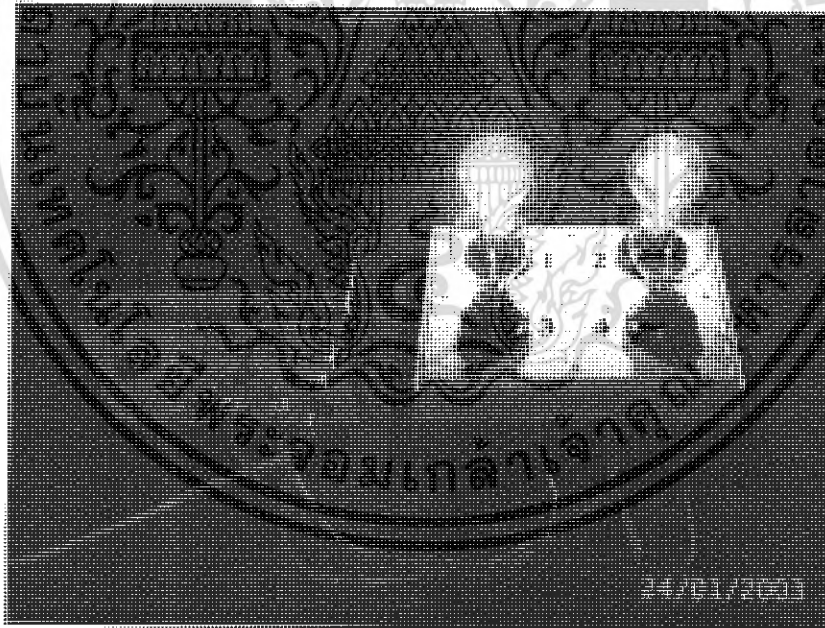


ภาคผนวก ก
เครื่องต้นแบบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

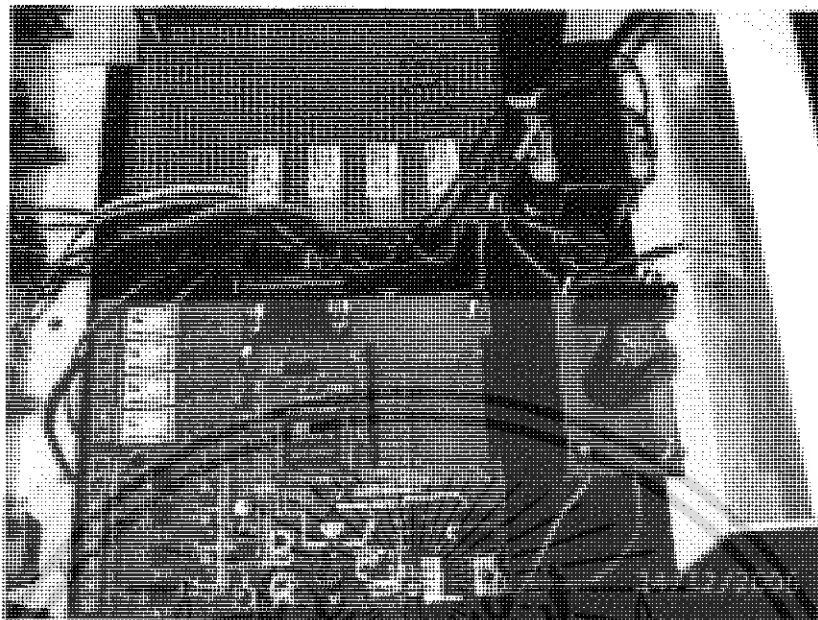


รูปที่ ก.1 แสดงภาพเครื่องต้นฉบับ



รูปที่ ก.2 แสดงภาพเครื่องต้นฉบับในสภาวะการทำงาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



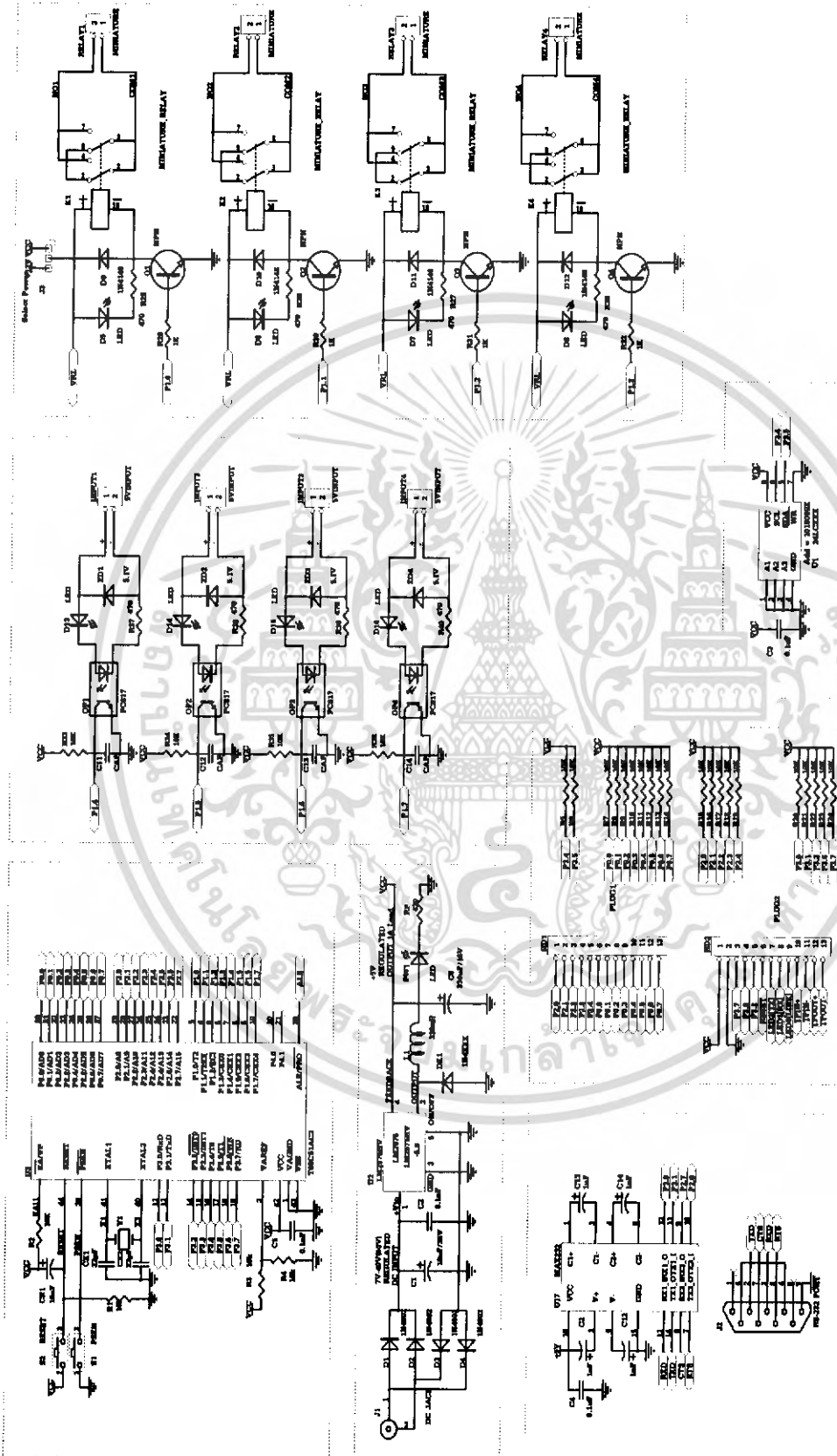
รูปที่ ก.3 แสดงภาพการประกอบอุปกรณ์ภายในเครื่องต้นฉบับ



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

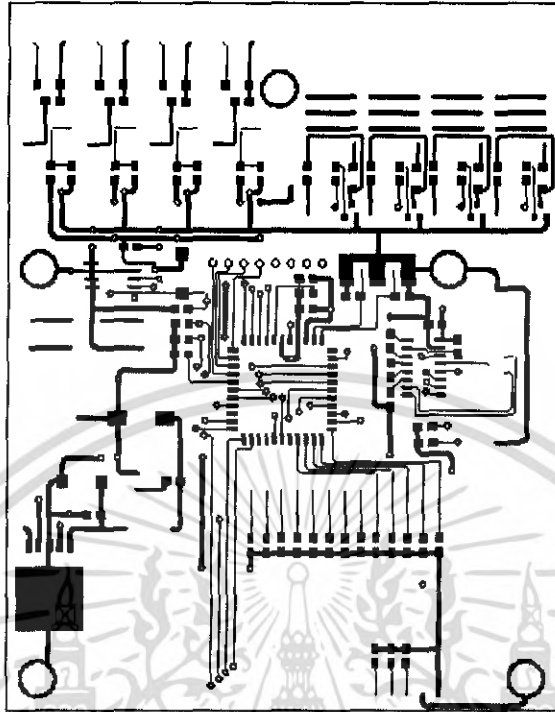


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

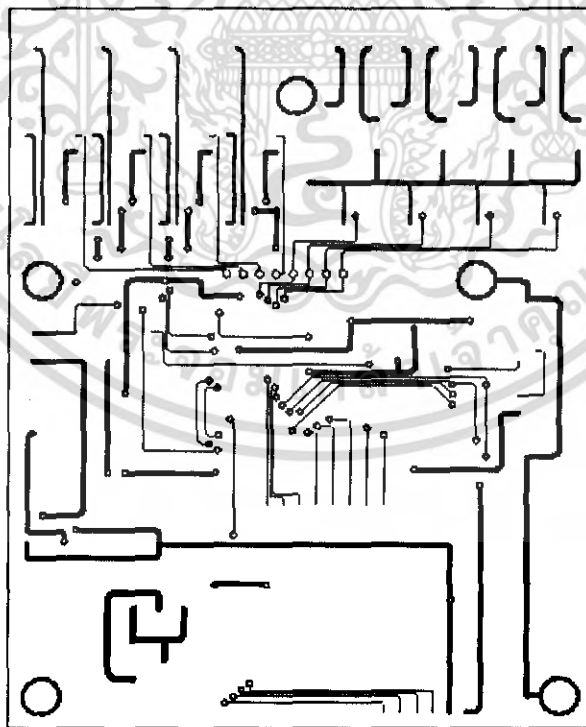


รูปที่ ข.1 วงจรควบคุมระบบเปิด - ปิดและรายงานสถานะลิวิตซ์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

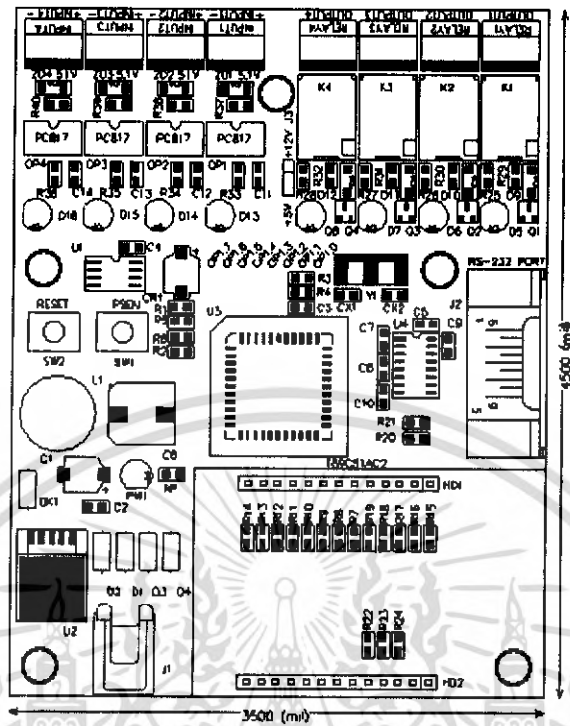


รูปที่ ๒.2 แผนวงจรพิมพ์ส่วนบนของระบบเปิด - ปิดและรายงานสถานะสวิตช์



รูปที่ ๒.3 แผนวงจรพิมพ์ส่วนล่างของระบบเปิด - ปิดและรายงานสถานะสวิตช์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ ข.4 ตำแหน่งการวางอุปกรณ์แผ่นพิมพ์ของระบบเปิด - ปิดและรายงานสถานะสวิทช์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ ค.1 รายการของอุปกรณ์ของวงจรควบคุมระบบเปิด - ปิดและรายงานสถานะสวิทช์

ชื่ออุปกรณ์	รายละเอียด	จำนวน
วงจรรวม		
IC1	T89C51AC2	1 ตัว
IC2	MAX-232	1 ตัว
IC3	LM2575	1 ตัว
IC4	24LCXXX	1 ตัว
อุปกรณ์สารกึ่งตัวนำ		
Q1-Q4	9013	4 ตัว
D1-D4	1N4002	4 ตัว
D5-D8	LED	4 ตัว
D9-D12	1N4148	4 ตัว
D13-D16	LED	4 ตัว
ZD1-ZD4	5.1V	1 ตัว
OP1-OP4	PC817	1 ตัว
PW1	LED	1 ตัว
DK1	1N4001	1 ตัว
ตัวเก็บประจุ		
C1	10 uF/35v	1 ตัว
C2,C4	0.1 uF/16v	3 ตัว
C5	0.1 uF/16v	1 ตัว
C6	330 uF/16v	4 ตัว
C11-C14	0.1 uF/16v	1 ตัว
CR1	10 uF /16v	12 ตัว
CX1,CX2	22 uF /16v	2 ตัว
ตัวความต้านทาน		
R1-R24	10K Ω 1/4W %	24 ตัว
R25-R28	470 Ω 1/4W %	4 ตัว
R29-R32	1 K Ω 1/4W %	4 ตัว
R33-R36	10K Ω 1/4W %	4 ตัว
R37-R40	470 Ω 1/4W %	4 ตัว
RP	470 Ω 1/4W %	1 ตัว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ ค.1 (ต่อ) รายการของอุปกรณ์ของวงจรควบคุมระบบเปิด - ปิดและรายงานสถานะสวิทช์

อุปกรณ์อื่นๆ		
K1-K4	Relay 6v CD/250v AC/5A	4 ตัว
L1	330uH	1 ตัว
SW1,SW2	สวิทช์	1 ตัว
J1	DC JACK	1 ตัว
J2	Rs-232	1 ตัว
Y1	Crystal	1 ตัว

ตารางที่ ค.2 รายการอุปกรณ์ของ Ethernet Module

ชื่ออุปกรณ์	รายละเอียด	จำนวน
วงจรรวม		
IC1	RTL8019AS	1 ตัว
อุปกรณ์สารกึ่งตัวนำ		
D1-D3	LED	3 ตัว
ตัวเก็บประจุ		
C1-C10	0.1uF/16v	10 ตัว
ตัวต้านทาน		
R1	4.7k Ω	1 ตัว
R2	10 k Ω	1 ตัว
R3	200 Ω	1 ตัว
R4-R6	470 Ω	3 ตัว
อุปกรณ์อื่น		
Y1	20MHz	1 ตัว
FC1	20F001N	1 ตัว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



ภาคผนวก ง
รายละเอียดและคุณสมบัติของอุปกรณ์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



SPECIFICATION
RTL8019AS

RTL8019AS

Realtek Full-Duplex Ethernet Controller with Plug and Play Function (RealPNP)

SPECIFICATION

REALTEK SEMICONDUCTOR CORP.

HEAD OFFICE

NO. 2, INDUSTRY E. RD. IX, SCIENCE-BASED
INDUSTRIAL PARK, HSINCHU 300, TAIWAN, R.O.C.
TEL: 886-3-5780211 FAX: 886-3-5776047

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



1. FEATURES

- 100-pin PQFP
- RTL8019 software compatible
- Supports PnP auto detect mode (RTL8019AS only)
- Compliant to Ethernet II and IEEE802.3 10Base5, 10Base2, 10BaseT
- Software compatible with NE2000 on both 8 and 16-bit slots
- Supports both jumper and jumperless modes
- Supports Microsoft's Plug and Play configuration for jumperless mode
- Supports Full-Duplex Ethernet function to double channel bandwidth
- Supports three level power down modes:
 - Sleep
 - Power down with internal clock running
 - Power down with internal clock halted
- Built-in data prefetch function to improve performance
- Supports UTP, AUI & BNC auto-detect (RTL8019AS only)
- Supports auto polarity correction for 10BaseT
- Support 8 IRQ lines
- Supports 16 I/O base address options
 - and extra I/O address fully decode mode (RTL8019AS only)
- Supports 16K, 32K, 64K and 16K-page mode access to BROM (up to 256 pages with 16K bytes/page)
- Supports BROM disable command to release memory after remote boot
- Supports flash memory read/write (RTL8019AS only)
- 16k byte SRAM built in (RTL8019AS only)
- Use 9346 (64*16-bit EEPROM) to store resource configurations and ID parameters
- Capable of programming blank 9346 on board for manufacturing convenience
- Support 4 diagnostic LED pins with programmable outputs

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



2. General Description

The RTL8019AS is a highly integrated Ethernet Controller which offers a simple solution to implement a Plug and Play NE2000 compatible adapter with full-duplex and power down features. With the three level power down control features, the RTL8019AS is made to be an ideal choice of the network device for a GREEN PC system. The full-duplex function enables simultaneously transmission and reception on the twisted-pair link to a full-duplex Ethernet switching hub. This feature not only increases the channel bandwidth from 10 to 20 Mbps but also avoids the performance degrading problem due to the channel contention characteristics of the Ethernet CSMA/CD protocol. The Microsoft's Plug and Play function can relieve the users from pains of taking care the adapter's resource configurations such as IRQ, I/O, and memory address, etc. However, for special applications not to be used as a Plug and Play compatible device, the RTL8019AS also supports the jumper and proprietary jumperless options.

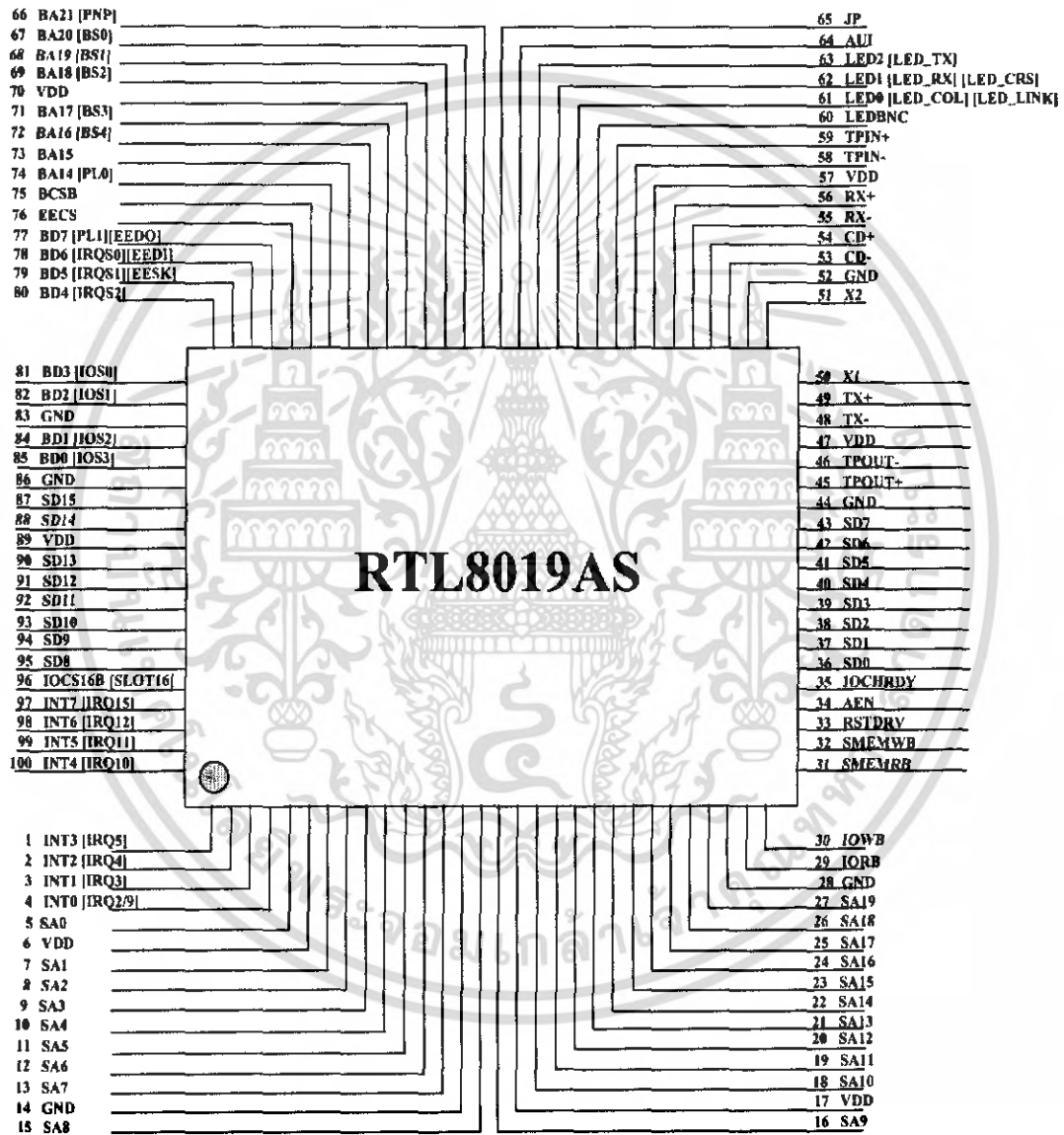
To offer a fully *plug and play* solution, the RTL8019AS provides the auto-detect capability between the integrated 10BaseT transceiver, BNC and AUI interface. Besides, the 10BaseT transceiver can automatically correct the polarity error on its receiving pair. Furthermore, 8 IRQ lines and 16 I/O base address options are provided for grand resource configuration flexibility.

The RTL8019AS supports 16k, 32k & 64k byte BROM and flash memory interface. It also offers the page mode function which can support up to 4M-byte BROM within only 16k-byte system memory space. Besides, the BROM disable command is provided to release the BROM memory space for other system usage (e.g. EMM386, etc.) after the BROM program is loaded.

The RTL8019AS is built in with 16K-byte SRAM in a single chip. It is designed not only to provide more friendly functions but also to save the effort of SRAM sourcing and inventory.



3. PIN CONFIGURATION



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



4. PIN DESCRIPTIONS

4.1. Power Pins

No.	Name	Type	Description
6, 17, 47, 57, 70, 89	VDD	P	+5V DC power
14, 28, 44, 52, 83, 86	GND	P	Ground

4.2. ISA Bus Interface Pins

No.	Name	Type	Descriptions
34	AEN	I	Address Enable. This ISA signal must be low for a valid I/O command.
97-100, 1-4	INT7-0	O	Interrupt request lines which are mapped to IRQ15, IRQ12, IRQ11, IRQ10, IRQ5, IRQ4, IRQ3, IRQ2/9 respectively. Only one line is selected to reflect the interrupt requests at one time. All other lines are tri-stated. The RTL8019AS also uses these pins as inputs to monitor the actual state of the corresponding interrupt lines on ISA bus. The result is recorded in the INTR register, which may be used by software to detect interrupt conflict.
35	IOCHRDY	O	This ISA signal is driven low to insert wait cycles to current host read/write command.
96	IOCS16B [SLOT16]	O	Upon power-on reset, this pin acts as an input named SLOT16 to detect whether a 16-bit or 8-bit slot is in use. To do this, it is connected to a pull-down resistor (about 27KW) externally. At the falling edge of RSTDRV, the RTL8019AS senses this pin's state. If it is sensed high, the adapter is thought to be placed on a 16-bit slot where this pin is connected to the host's IOCS16B pin, which is typically pulled up by a 300W resistor on the mother board. If it is sensed low, the adapter is thought to be placed on an 8-bit slot where this pin is merely pulled low by the 27KW resistor. After having latched the input state, this pin is switched as the IOCS16B signal which is an open-drain output and is driven low during a 16-bit host data transfer. It is decoded from AEN and SA9-0.
29	IORB	I	Host I/O read command.
30	IOWB	I	Host I/O write command.
33	RSTDRV	I	High active hardware reset signal from the ISA bus. Pulses with high level less than 800ns are ignored.
27-18, 16-15, 13-7, 5	SA19-0	I	Host address bus. SA10 is added to implement the fully decode of PnP ports, address 279h and A79h. In RTL8019, SA10 is not decoded. In RTL8019AS, SA10 should be 0 for a valid access to PnP ports.
87-88, 90-95, 43-36	SD15-0	I/O	Host data bus.
31	SMEMRB	I	Host memory read command.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



32	SMEMWB	I	Host memory write command. This pin is added to decode the write command of a flash memory.
----	--------	---	---

4.3. Memory Interface Pins (including BROM, EEPROM)

No.	Name	Type	Description
75	BCSB	O	BROM chip select. Active low signal, asserted when BROM is read. RTL8019AS drives this pin low when SA19-14 matches the selected BROM memory base address and either of the 2 conditions below meets: (1) SMEMRB is low (2) SMEMWB is low and RTL8019AS's flash memory write function is enabled.
76	EECS	O	9346 chip select. Active high signal, asserted when 9346 is read/write.
66-69, 71-74	BA21-14	O	BROM address. *
77-82, 84-85	BD7-0	I/O	BROM data bus.
[79]	[EESK]	O	9346 serial data clock
[78]	[EEDI]	O	9346 serial data input
[77]	[EEDO]	I	9346 serial data output
[66]	[PNP]	I	The following pins are defined for jumper options. Their states are latched at the falling edge of RSTDRV, then they are changed to serve as the SRAM bus. Each of them is internally pulled down by a 100KΩ resistor. Therefore, the input will be low when left open and high when pulled up by a 10K resistor externally. When it is high in jumperless mode (i.e. JP=low), the RTL8019AS is forced into Plug and Play mode regardless of the contents of 9346. The following pins are don't care in jumperless mode (JP=low).
[72-71, 69-67]	[BS4-0]	I	Select BROM size and base address.
[85-84, 82-81]	[IOS3-0]	I	Select I/O base address.
[77, 74]	[PL1-0]	I	Select network medium type.
[80-78]	[IRQS2-0]	I	Select one interrupt line among INT7-0.
65	JP	I	When high, this pin selects jumper mode. When low, it selects jumperless modes (including RT jumperless and Plug and Play).

* After RTL8019AS latches all jumper status upon power on reset, these pins *always* reflect the value of BPAGE register directly in BROM page mode. In normal mode, BA16-21 are not used and BA14-15 act as:

BROM Size	BA14	BA15
16K	high	high
32K	SA14	high
64K	SA14	SA15

*Note: RTL8019AS doesn't drive BA14-21 until the SMEMRB goes from high to low.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



4.4. Medium Interface Pins

No.	Name	Type	Description
64	AUI	I	This input is used to detect the usage of an external MAU on the AUI interface. The input should be driven low for embedded BNC and high for external MAU. When the input is high, RTL8019AS sets the AUI bit (bit5) in CONFIG0 and drives LEDBNC low to disable the BNC. If this pin is not used, it should be connected to GND such that RTL8019AS acts like RTL8019. Please refer to section 5.1.2.2. CONFIG0 for more details.
54,53	CD+,CD-	I	This AUI collision input pair carries the differential collision input signal from the MAU.
56,55	RX+,RX-	I	This AUI receive input pair carries the differential receive input signal from the MAU.
49,48	TX+,TX-	O	This AUI transmit output pair contains differential line drivers which send Manchester encoded data to the MAU. These outputs are source followers and require 270 ohm pull-down resistors to GND.
59,58	TPIN+, TPIN-	I	This TP input pair receives the 10 Mbits/s differential Manchester encoded data from the twisted-pair wire.
45,46	TPOUT+, TPOUT-	O	This pair carries the differential TP transmit output. The output Manchester encoded signals have been pre-distorted to prevent overcharge on the twisted-pair media and thus reduce jitter.
50	X1	I	20Mhz crystal or external oscillator input.
51	X2	O	Crystal feedback output. This output is used in crystal connection only. It must be left open when X1 is driven with an external oscillator.

4.5. LED Output Pins

No.	Name	Type	Description
60	LEDBNC	O	This pin goes high when RTL8019AS's medium type is set to 10Base2 mode or auto-detect mode with link test failure. Otherwise, this pin is low. This pin can be used to control the power of the DC convertor for CX MAU and connected to an LED to indicate the used medium type.
61	LED0	O	When LEDS0 bit (in CONFIG3 register of RTL8019AS Page3) is 0, this pin acts as LED_COL. When LEDS0=1, it acts as LED_LINK.
62,63	LED1,LED2	O	When LEDS1 bit (in CONFIG3 register of RTL8019AS Page3) is 0, these 2 pins act as LED_RX & LED_TX respectively. When LEDS1=1, these pins act as LED_CRS & MCSB. Please refer to section 6.5 for details of the lightning behavior of all LEDs.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่นิยมนำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



5. Register Descriptions

The registers in RTL8019AS can be roughly divided into two groups by their address and functions -- one for NE2000, the other for Plug and Play (PnP).

5.1. Group 1: NE2000 Registers

This group includes 4 pages of registers which are selected by bit PS0 & PS1 in the CR register. Each page contains 16 registers. Besides those registers compatible with NE2000, the RTL8019AS defines some registers for software configuration and feature enhancement.

5.1.1. Register Table

No (Hex)	Page0		Page1	Page2	Page3	
	[R]	[W]	[R/W]	[R]	[R]	[W]
00	CR	CR	CR	CR	CR	CR
01	CLDA0	PSTART	PAR0	PSTART	9346CR	9346CR
02	CLDA1	PSTOP	PAR1	PSTOP	BPAGE	BPAGE
03	BNRY	BNRY	PAR2	-	CONFIG0	-
04	TSR	TPSR	PAR3	TPSR	CONFIG1	CONFIG1
05	NCR	TBCR0	PAR4	-	CONFIG2	CONFIG2
06	FIFO	TBCR1	PAR5	-	CONFIG3	CONFIG3
07	ISR	ISR	CURR	-	-	TEST
08	CRDA0	RSAR0	MAR0	-	CSNSAV	-
09	CRDA1	RSAR1	MAR1	-	-	HLTCLK
0A	8019ID0	RBCR0	MAR2	-	-	-
0B	8019ID1	RBCR1	MAR3	-	INTR	-
0C	RSR	RCR	MAR4	RCR	-	FMWP
0D	CNTR0	TCR	MAR5	TCR	CONFIG4	-
0E	CNTR1	DCR	MAR6	DCR	-	-
0F	CNTR2	IMR	MAR7	IMR	-	-
10-17	Remote DMA Port					
18-1F	Reset Port					

Notes: "-" denotes reserved. Registers with names typed in **bold italic** format are RTL8019AS defined registers and are not supported in a standard NE2000 adapter.


Page 0 (PS1=0, PS0=0)

No.	Name	Type	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
00H	CR	R/W	PS1	PS0	RD2	RD1	RD0	TXP	STA	STP
01H	CLDA0	R	A7	A6	A5	A4	A3	A2	A1	A0
	PSTART	W	A15	A14	A13	A12	A11	A10	A9	A8
02H	CLDA1	R	A15	A14	A13	A12	A11	A10	A9	A8
	PSTOP	W	A15	A14	A13	A12	A11	A10	A9	A8
03H	BNRY	R/W	A15	A14	A13	A12	A11	A10	A9	A8
04H	TSR	R	OWC	CDH	0	CRS	ABT	COL	-	PTX
	TPSR	W	A15	A14	A13	A12	A11	A10	A9	A8
05H	NCR	R	0	0	0	0	NC3	NC2	NC1	NC0
	TBCR0	W	TBC7	TBC6	TBC5	TBC4	TBC3	TBC2	TBC1	TBC0
06H	FIFO	R	D7	D6	D5	D4	D3	D2	D1	D0
	TBCR1	W	TBC15	TBC14	TBC13	TBC12	TBC11	TBC10	TBC9	TBC8
07H	ISR	R/W	RST	RDC	CNT	OVW	TXE	RXE	PTX	PRX
08H	CRDA0	R	A7	A6	A5	A4	A3	A2	A1	A0
	RSAR0	W	A7	A6	A5	A4	A3	A2	A1	A0
09H	CRDA1	R	A15	A14	A13	A12	A11	A10	A9	A8
	RSAR1	W	A15	A14	A13	A12	A11	A10	A9	A8
0AH	8019ID0	R	0	1	0	1	0	0	0	0
	RBCR0	W	RBC7	RBC6	RBC5	RBC4	RBC3	RBC2	RBC1	RBC0
0BH	8019ID1	R	0	1	1	1	0	0	0	0
	RBCR1	W	RBC15	RBC14	RBC13	RBC12	RBC11	RBC10	RBC9	RBC8
0CH	RSR	R	DFR	DIS	PHY	MPA	0	FAE	CRC	PRX
	RCR	W	-	-	MON	PRO	AM	AB	AR	SEP
0DH	CNTR0	R	CNT7	CNT6	CNT5	CNT4	CNT3	CNT2	CNT1	CNT0
	TCR	W	-	-	-	OFST	ATD	LB1	LB0	CRC
0EH	CNTR1	R	CNT7	CNT6	CNT5	CNT4	CNT3	CNT2	CNT1	CNT0
	DCR	W	-	FT1	FT0	ARM	LS	LAS	BOS	WTS
0FH	CNTR2	R	CNT7	CNT6	CNT5	CNT4	CNT3	CNT2	CNT1	CNT0
	IMR	W	-	RDCE	CNTE	OVWE	TXEE	RXEE	PTXE	PRXE

Page 1 (PS1=0, PS0=1)

No.	Name	Type	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
00H	CR	R/W	PS1	PS0	RD2	RD1	RD0	TXP	STA	STP
01H	PAR0	R/W	DA7	DA6	DA5	DA4	DA3	DA2	DA1	DA0
02H	PAR1	R/W	DA15	DA14	DA13	DA12	DA11	DA10	DA9	DA8
03H	PAR2	R/W	DA23	DA22	DA21	DA20	DA19	DA18	DA17	DA16
04H	PAR3	R/W	DA31	DA30	DA29	DA28	DA27	DA26	DA25	DA24
05H	PAR4	R/W	DA39	DA38	DA37	DA36	DA35	DA34	DA33	DA32
06H	PAR5	R/W	DA47	DA46	DA45	DA44	DA43	DA42	DA41	DA40
07H	CURR	R/W	A15	A14	A13	A12	A11	A10	A9	A8
08H	MAR0	R/W	FB7	FB6	FB5	FB4	FB3	FB2	FB1	FB0
09H	MAR1	R/W	FB15	FB14	FB13	FB12	FB11	FB10	FB9	FB8
0AH	MAR2	R/W	FB23	FB22	FB21	FB20	FB19	FB18	FB17	FB16
0BH	MAR3	R/W	FB31	FB30	FB29	FB28	FB27	FB26	FB25	FB24
0CH	MAR4	R/W	FB39	FB38	FB37	FB36	FB35	FB34	FB33	FB32
0DH	MAR5	R/W	FB47	FB46	FB45	FB44	FB43	FB42	FB41	FB40
0EH	MAR6	R/W	FB55	FB54	FB53	FB52	FB51	FB50	FB49	FB48
0FH	MAR7	R/W	FB63	FB62	FB61	FB60	FB59	FB58	FB57	FB56

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่นอนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Page 2(PS1=1, PS0=0)

No.	Name	Type	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
00H	CR	R/W	PS1	PS0	RD2	RD1	RD0	TXP	STA	STP
01H	PSTART	R	A15	A14	A13	A12	A11	A10	A9	A8
02H	PSTOP	R	A15	A14	A13	A12	A11	A10	A9	A8
03H	-									
04H	TPSR	R	A15	A14	A13	A12	A11	A10	A9	A8
05H 0BH	-									
0CH	RCR	R	-	-	MON	PRO	AM	AB	AR	SEP
0DH	TCR	R	-	-	-	OFST	ATD	LB1	LB0	CRC
0EH	DCR	R	-	FT1	FT0	ARM	LS	LAS	BOS	WTS
0FH	IMR	R	-	RDCE	CNTE	OVWE	TXEE	RXEE	PTXE	PRXE

Page 3(PS1=1, PS0=1)

No.	Name	Type	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
00H	CR	R/W	PS1	PS0	RD2	RD1	RD0	TXP	STA	STP
01H	9346CR	R	EEM1	EEM0	-	-	EECS	EESK	EEDI	EEDO
		W	EEM1	EEM0	-	-	EECS	EESK	EEDI	-
02H	BPAGE	R/W	BP7	BP6	BP5	BP4	BP3	BP2	BP1	BP0
03H	CONFIG0	R	VerID1	VerID0	AUI	PNPJP	JP	BNC	0	0
04H	CONFIG1	R	IRQEN	IRQS2	IRQS1	IRQS0	IOS3	IOS2	IOS1	IOS0
		W*	IRQEN	-	-	-	-	-	-	-
05H	CONFIG2	R	PL1	PL0	BSELB	BS4	BS3	BS2	BS1	BS0
		W*	PL1	PL0	BSELB	-	-	-	-	-
06H	CONFIG3	R	PNP	FUDUP	LEDS1	LEDS0	-	SLEEP	PWRDN	ACTIVEB
		W*	-	-	-	-	-	SLEEP	PWRDN	-
07H	TEST	R/W	Reserved, Do not write							
08H	CSNSAV	R	CSN7	CSN6	CSN5	CSN4	CSN3	CSN2	CSN1	CNS0
09H	HLTCLK	W	HLT7	HLT6	HLT5	HLT4	HLT3	HLT2	HLT1	HLT0
0AH	-	-	Reserved							
0BH	INTR	R	INT7	INT6	INT5	INT4	INT3	INT2	INT1	INT0
0CH	FMWP	W*	Flash Memory Write Protect							
0DH	CONFIG4	R	-	-	-	-	-	-	-	IOMS
0EH 0FH	-	-	Reserved							

Note: The registers marked with type='W*' can be written only if bits EEM1=EEM0=1.

5.1.2. Register Functions
5.1.2.1. NE2000 Compatible Registers

CR: Command Register (00H; Type=R/W)

This register is used to select register pages, enable or disable remote DMA operation and issue commands.



Bit	Symbol	Description																								
7, 6	PS1, PS0	<table border="1"> <thead> <tr> <th>PS1</th> <th>PS0</th> <th>Register Page</th> <th>Remark</th> </tr> </thead> <tbody> <tr> <td>0</td> <td>0</td> <td>0</td> <td>NE2000 compatible</td> </tr> <tr> <td>0</td> <td>1</td> <td>1</td> <td>NE2000 compatible</td> </tr> <tr> <td>1</td> <td>0</td> <td>2</td> <td>NE2000 compatible</td> </tr> <tr> <td>1</td> <td>1</td> <td>3</td> <td>RTL8019AS Configuration</td> </tr> </tbody> </table>	PS1	PS0	Register Page	Remark	0	0	0	NE2000 compatible	0	1	1	NE2000 compatible	1	0	2	NE2000 compatible	1	1	3	RTL8019AS Configuration				
PS1	PS0	Register Page	Remark																							
0	0	0	NE2000 compatible																							
0	1	1	NE2000 compatible																							
1	0	2	NE2000 compatible																							
1	1	3	RTL8019AS Configuration																							
5-3	RD2-0	<table border="1"> <thead> <tr> <th>RD2</th> <th>RD1</th> <th>RD0</th> <th>Function</th> </tr> </thead> <tbody> <tr> <td>0</td> <td>0</td> <td>0</td> <td>Not allowed</td> </tr> <tr> <td>0</td> <td>0</td> <td>1</td> <td>Remote Read</td> </tr> <tr> <td>0</td> <td>1</td> <td>0</td> <td>Remote Write</td> </tr> <tr> <td>0</td> <td>1</td> <td>1</td> <td>Send Packet</td> </tr> <tr> <td>1</td> <td>*</td> <td>*</td> <td>Abort/Complete remote DMA</td> </tr> </tbody> </table>	RD2	RD1	RD0	Function	0	0	0	Not allowed	0	0	1	Remote Read	0	1	0	Remote Write	0	1	1	Send Packet	1	*	*	Abort/Complete remote DMA
RD2	RD1	RD0	Function																							
0	0	0	Not allowed																							
0	0	1	Remote Read																							
0	1	0	Remote Write																							
0	1	1	Send Packet																							
1	*	*	Abort/Complete remote DMA																							
2	TXP	This bit must be set to transmit a packet. It is internally reset either after the transmission is completed or aborted. Writing a 0 has no effect.																								
1	STA	The STA bit controls nothing. It only reflects the value written to this bit. POWER UP=0.																								
0	STP	This bit is the STOP command. When it is set, no packets will be received or transmitted. POWER UP=1. <table border="1"> <thead> <tr> <th>STA</th> <th>STP</th> <th>Function</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>0</td> <td>Start Command</td> </tr> <tr> <td>0</td> <td>1</td> <td>Stop Command</td> </tr> </tbody> </table>	STA	STP	Function	1	0	Start Command	0	1	Stop Command															
STA	STP	Function																								
1	0	Start Command																								
0	1	Stop Command																								

ISR: Interrupt Status Register (07H; Type=R/W in Page0)

This register reflects the NIC status. The host reads it to determine the cause of an interrupt. Individual bits are cleared by writing a "1" into the corresponding bit. It must be cleared after power up.

Bit	Symbol	Description
7	RST	This bit is set when NIC enters reset state and is cleared when a start command is issued to the CR. It is also set when receive buffer overflows and is cleared when one or more packets have been read from the buffer.
6	RDC	Set when remote DMA operation has been completed.
5	CNT	Set when MSB of one or more of the network tally counters has been set.
4	OVW	This bit is set when the receive buffer has been exhausted.
3	TXE	Transmit error bit is set when a packet transmission is aborted due to excessive collisions.
2	RXE	This bit is set when a packet received with one or more of the following errors: - CRC error - Frame alignment error - Missed packet
1	PTX	This bit indicates packet transmitted with no errors.
0	PRX	This bit indicates packet received with no errors.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่นอนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



IMR: Interrupt Mask Register (0FH; Type=W in Page0, Type=R in Page2)

All bits correspond to the bits in the ISR register. POWER UP=all 0s. Setting individual bits will enable the corresponding interrupts.

DCR: Data Configuration Register (0EH; Type=W in Page0, Type=R in Page2)

Bit	Symbol	Description
7	-	Always 1
6, 5	FT1, FT0	FIFO threshold select bit 1 and 0.
4	ARM	Auto-initialize Remote 0: Send Packet Command not executed. 1: Send Packet Command executed.
3	LS	Loopback Select 0: Loopback mode selected. Bits 1 and 2 of the TCR must also be programmed for Loopback operation. 1: Normal Operation
2	LAS	This bit must be set to zero. NIC only supports dual 16-bit DMA mode. POWER UP =1
1	BOS	Byte Order Select (Not implement) 0: MS byte placed on MD15-8 and LS byte on MD7-0. (32xxx,80x86) 1: MS byte placed on MD7-0 and LS byte on MD15-8. (680x0)
0	WTS	Word Transfer Select 0: byte-wide DMA transfer 1: word-wide DMA transfer

TCR: Transmit Configuration Register (0DH; Type=W in Page0, Type=R in Page2)

Bit	Symbol	Description																								
7	-	Always 1																								
6	-	Always 1																								
5	-	Always 1																								
4	OFST	Collision Offset Enable.																								
3	ATD	Auto Transmit Disable. 0: normal operation 1: reception of multicast address hashing to bit 62 disables transmitter, reception of multicast address hashing to bit 63 enables transmitter.																								
2, 1	LB1, LB0	<table border="1"> <thead> <tr> <th>LB1</th> <th>LB0</th> <th>Mode</th> <th>Remark</th> </tr> </thead> <tbody> <tr> <td>0</td> <td>0</td> <td>0</td> <td>Normal Operation</td> </tr> <tr> <td>0</td> <td>1</td> <td>1</td> <td>Internal Lookback</td> </tr> <tr> <td>1</td> <td>0</td> <td>2</td> <td>External Lookback</td> </tr> <tr> <td>1</td> <td>1</td> <td>3</td> <td>External Lookback</td> </tr> </tbody> </table>	LB1	LB0	Mode	Remark	0	0	0	Normal Operation	0	1	1	Internal Lookback	1	0	2	External Lookback	1	1	3	External Lookback				
LB1	LB0	Mode	Remark																							
0	0	0	Normal Operation																							
0	1	1	Internal Lookback																							
1	0	2	External Lookback																							
1	1	3	External Lookback																							
0	CRC	<p>The NIC CRC logic comprises a CRC generator for transmitter and a CRC checker for receiver. This bit controls the activity of the CRC logic. If this bit set, CRC is inhibited by transmitter. Otherwise CRC is appended by transmitter.</p> <table border="1"> <thead> <tr> <th colspan="2">Conditions</th> <th colspan="2">CRC Logic Activities</th> </tr> <tr> <th>CRC Bit</th> <th>Mode</th> <th>CRC Generator</th> <th>CRC Checker</th> </tr> </thead> <tbody> <tr> <td>0</td> <td>normal</td> <td>enabled</td> <td>enabled</td> </tr> <tr> <td>1</td> <td>normal</td> <td>disabled</td> <td>enabled</td> </tr> <tr> <td>0</td> <td>loopback</td> <td>enabled</td> <td>disabled</td> </tr> <tr> <td>1</td> <td>loopback</td> <td>disabled</td> <td>enabled</td> </tr> </tbody> </table>	Conditions		CRC Logic Activities		CRC Bit	Mode	CRC Generator	CRC Checker	0	normal	enabled	enabled	1	normal	disabled	enabled	0	loopback	enabled	disabled	1	loopback	disabled	enabled
Conditions		CRC Logic Activities																								
CRC Bit	Mode	CRC Generator	CRC Checker																							
0	normal	enabled	enabled																							
1	normal	disabled	enabled																							
0	loopback	enabled	disabled																							
1	loopback	disabled	enabled																							

TSR: Transmit Status Register (04H; Type=R in Page0)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



This register indicates the status of a packet transmission.

Bit	Symbol	Description
7	OWC	Out of Window Collision. It is set when a collision is detected after a slot time (51.2us). Transmissions are rescheduled as in normal collisions.
6	CDH	CD Heartbeat. The NIC watches for a collision signal (i.e. CD Heartbeat signal) during the first 6.4us of the interframe gap following a transmission. This bit is set if the transceiver fails to send this signal.
5	-	Always 1.
4	CRS	Carrier Sense lost bit is set when the carrier is lost during transmitting a packet.
3	ABT	It indicates the NIC aborted the transmission because of excessive collisions.
2	COL	It indicates the transmission collided with some other station on the network.
1	-	Always 1
0	PTX	This bit indicates the transmission completes with no errors.

RCR: Receive Configuration Register (0CH; Type=W in Page0, Type=R in Page2)

Bit	Symbol	Description
7	-	Always 1
6	-	Always 1
5	MON	When monitor mode bit is set, received packets are checked for address match, good CRC and frame alignment but not buffered to memory. Otherwise, packets will be buffered to memory.
4	PRO	If PRO=1, all packets with physical destination address accepted. If PRO=0, physical destination address must match the node address programmed in PAR0-5.
3	AM	If AM=1, packets with multicast destination address are accepted. If AM=0, packets with multicast destination address are rejected.
2	AB	If AB=1, packets with broadcast destination address are accepted. If AB=0, packets with broadcast destination address are rejected.
1	AR	If AR=1, packets with length fewer than 64 bytes are accepted. If AR=0, packets with length fewer than 64 bytes are rejected.
0	SEP	If SEP=1, packets with receive errors are accepted. If SEP=0, packets with receive errors are rejected.

RSR: Receive Status Register (0CH; Type=R in Page0)

Bit	Symbol	Description
7	DFR	Deffering. Set when a carrier or a collision is detected.
6	DIS	Receiver Disabled. When the NIC enters the monitor mode, this bit is set and receiver is disabled. Reset when receiver is enabled after leaving the monitor mode.
5	PHY	PHY bit is set when the received packet has a multicast or broadcast destination address. It is reset when the received packet has a physical destination address.
4	MPA	Missed Packet bit is set when the incoming packet can not be accepted by NIC because of a lack of receive buffer or if NIC is in monitor mode. Increment CNTR2 tally counter.
3	-	Always 1.
2	FAE	Frame Alignment Error bit reflects the incoming packet didn't end on a byte boundary and CRC did not match at last byte boundary. Increment CNTR0 tally counter.
1	CRC	CRC error bit reflects packet received with CRC error. This bit will also be set for FAE errors. Increment CNTR1 tally counter.
0	PRX	This bit indicates packet received with no errors.

CLDA0, 1: Current Local DMA Registers (01H & 02H; Type=R in Page0)

These two registers can be read to get the current local DMA address.

PSTART: Page Start Register (01H; Type=W in Page0, Type=R in Page 2)

The Page Start register sets the start page address of the receive buffer ring.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่นิยมนำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

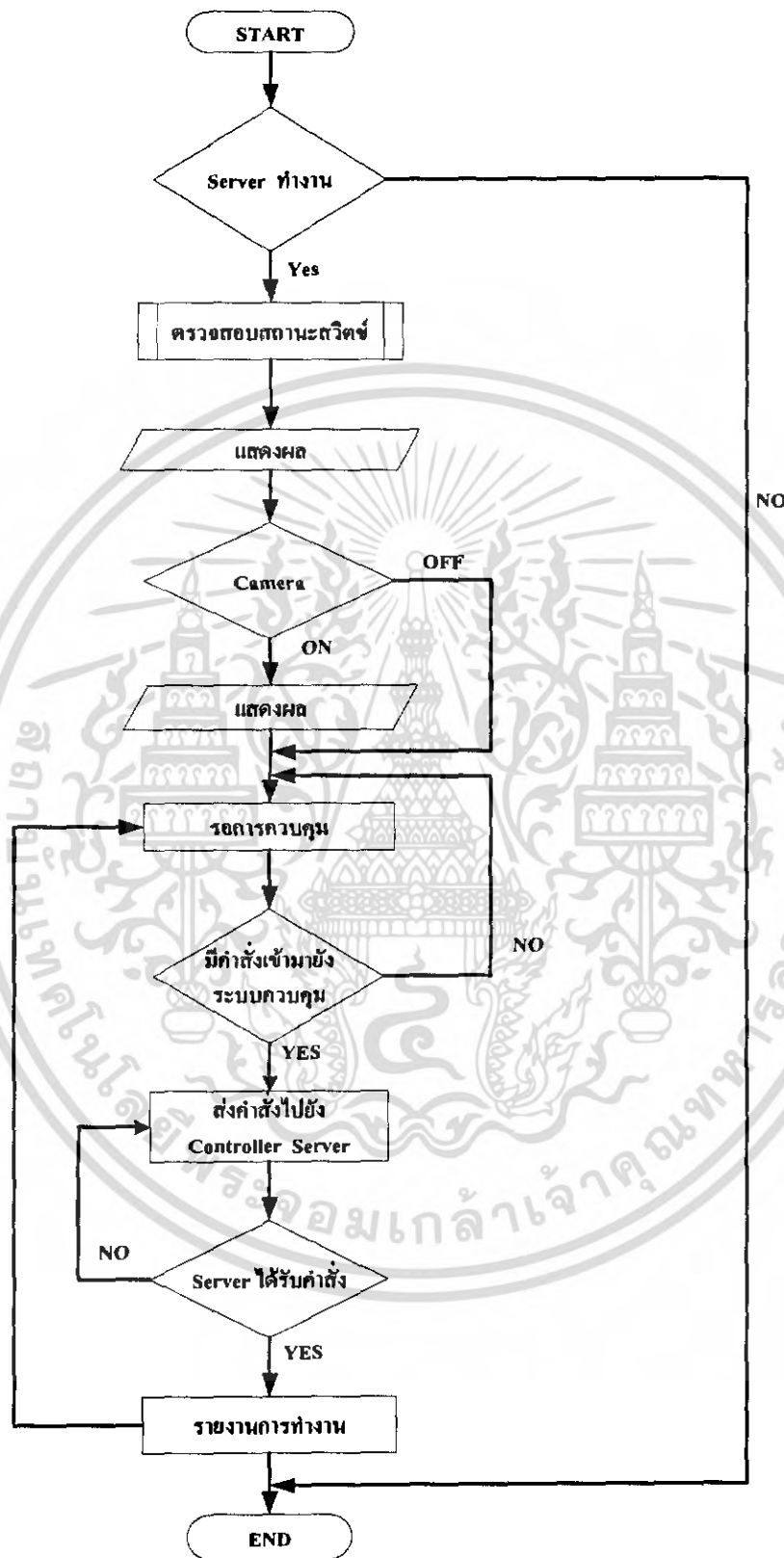


- PSTOP:** Page Stop Register (02H; Type=W in Page0, Type=R in Page2)
The Page Stop register sets the stop page address of the receive buffer ring. In 8 bit mode the PSTOP register should not exceed to 0x60, in 16 bit mode the PSTOP register should not exceed to 0x80.
- BNRY:** Boundary Register (03H; Type=R/W in Page0)
This register is used to prevent overwrite of the receive buffer ring. It is typically used as a pointer indicating the last receive buffer page the host has read.
- TPSR:** Transmit Page Start Register (04H; Type=W in Page0)
This register sets the start page address of the packet to be transmitted.
- TBCR0,1:** Transmit Byte Count Registers (05H & 06H; Type=W in Page0)
These two registers set the byte counts of the packet to be transmitted.
- NCR:** Number of Collisions Register (05H; Type=R in Page0)
The register records the number of collisions a node experiences during a packet transmission.
- FIFO:** First In First Out Register (06H; Type=R in Page0)
This register allows the host to examine the contents of the FIFO after loopback.
- CRDA0, 1:** Current Remote DMA Address registers (08H & 09H; Type=R in Page0)
These two registers contain the current address of remote DMA.
- RSAR0,1:** Remote Start Address Registers (08H & 09H; Type=W in Page0)
These two registers set the start address of remote DMA.
- RBCR0,1:** Remote Byte Count Registers (0AH & 0BH; Type=W in Page0)
These two registers set the data byte counts of remote DMA.
- CNTR0:** Frame Alignment Error Tally Counter Register (0DH; Type=R in Page0)
- CNTR1:** CRC Error Tally Counter Register (0EH; Type=R in Page0)
- CNTR2:** Missed Packet Tally Counter Register (0FH; Type=R in Page0)
- PAR0-5:** Physical Address Registers (01H - 06H; Type=R/W in Page1)
These registers contain my Ethernet node address and are used to compare the destination address of incoming packets for acceptance or rejection.
- CURR:** Current Page Register (07H; Type=R/W in Page1)
This register points to the page address of the first receive buffer page to be used for a packet reception.
- MAR0-7:** Multicast Address Register (08H - 0FH; Type=R/W in Page1)
These registers provide filtering bits of multicast addresses hashed by the CRC logic.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่นิยมนำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

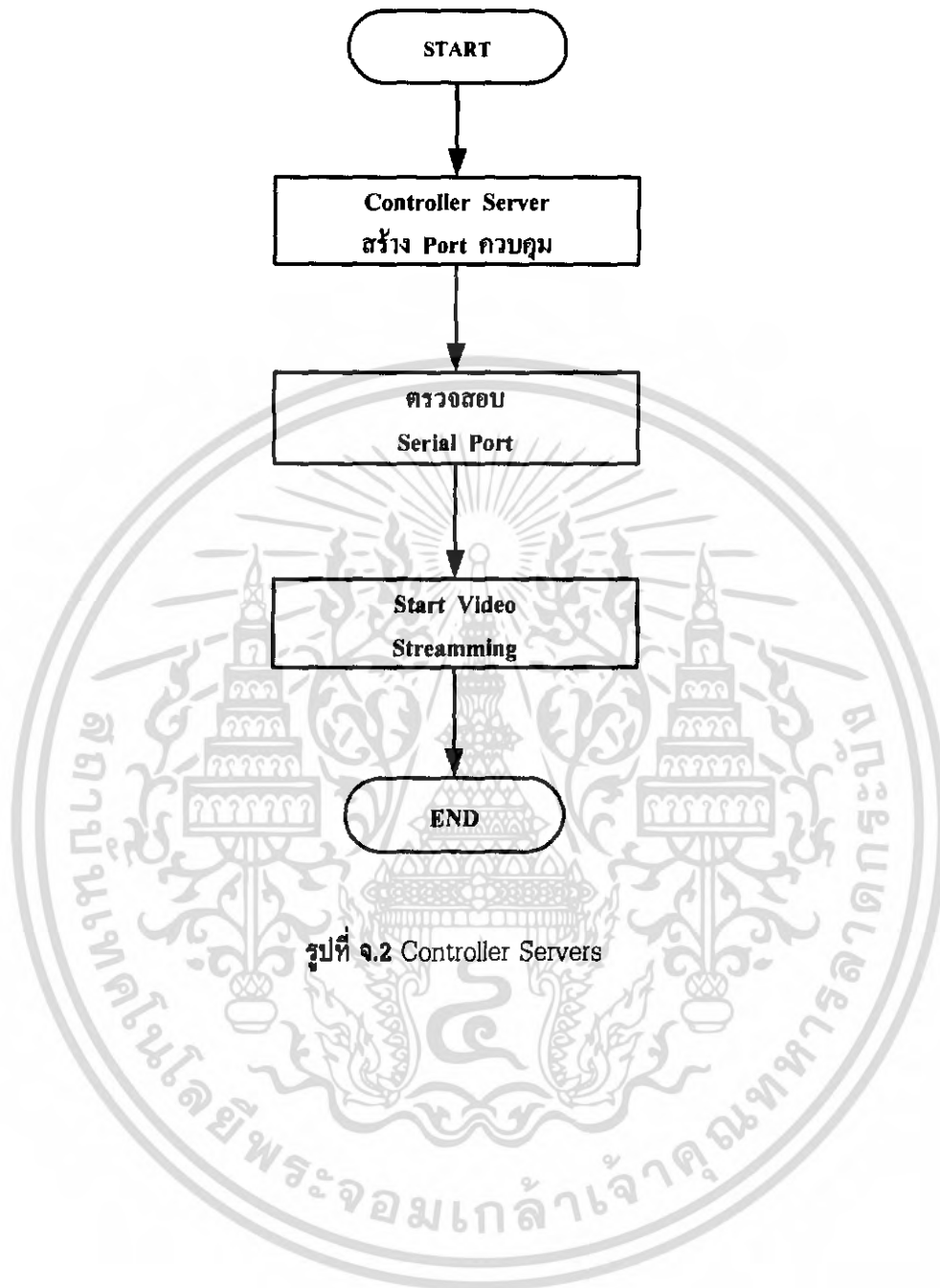


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ ๑.1 การควบคุมผ่านอินเทอร์เน็ต

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ ๑.๒ Controller Servers

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



ภาคผนวก จ
รหัสต้นฉบับของโปรแกรม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โปรแกรมควบคุมการทำงานผ่านอินเทอร์เน็ต

```

/**(-----)**/
/**(-          HTML FILE CODE WEB-PAGE          -)**/
/**(-----)**/

unsigned char code logo1[5] = {
0X47,0X49,0X46,0X38,0X39
};

char code Html_code[] =
"<HTML><HEAD><TITLE>Sign-Advancetech Web Manager</TITLE>"
"<META http-equiv=Content-Type content= 'text/html; charset=windows-874'>"
"<META content='Microsoft FrontPage 5.0' name=GENERATOR>"
"</HEAD><FRAMESET border=0 "
"frameSpacing=0 frameBorder=0 cols=250,*>"
"<FRAME name=navigator src='navigator.html' frameBorder=no>"
"<FRAME name=main src='home.html' frameBorder=no>"
"</FRAMESET></HTML>";

char code start[] =
"Please Wait..."
"<meta http-equiv='refresh' content='5/'>";

char code start1[] =
"Please select the menu to modify Configuration....";

char code navigator[] =
"<BODY bgColor=#ffffff>\n\r"
"<TABLE height=148 cellPadding=1 width=169 border=0 cellspacing=1 style='border-collapse:
collapse' bordercolor=#111111>\n\r"
"<TBODY><TR>\n\r"
"<TD vAlign=top align=middle width=51 height=1>\n\r"
"<IMG border=0 src='C:/SAT5100.JPG' width='150' height='114'>\n\r"
"</FONT></B></TD><TR>\n\r"
"<TD vAlign=top align=middle width=310 height=263 colspan=2>\n\r"
"<B><FONT face=Arial color=#000080 size='2'>SAT5100.NET</font></b>\n\r"

```

เอกสารนี้เป็นลิขสิทธิ์ของ บริษัท เทคโนโลยี คอมพิวเตอร์ จำกัด กรุณาแจ้งเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

"<BR><FONT face=Arial,Helvetica size=-1 >\n\r"
"<BR><A target='main' href='http://"/"/"/tcpip.html'><b><font face=Arial size=2>TCP/IP</A>\n\r"
"<BR><A target='main' href='http://"/"/"/serial.html'><b><font face=Arial size=2>Serial Port</A>\n\r"
"<BR><A target='main' href='http://"/"/"/input.html'><b><font face=Arial size=2>Input</A>\n\r"
"<BR><A target='main' href='http://"/"/"/output.html'><b><font face=Arial size=2>Output</A>\n\r"
//"<BR><A target='main' href='http://"/"/"/tech.html'><b><font face=Arial size=2>Tech
Support</A>\n\r"
//"<BR><A target='main' href='http://"/"/"/home.html'><b><font face=Arial size=2>Home</A>\n\r"
"<BR><BR>\n\r"
"<P align=center><FONT color=#218868 size=-1>Select from the menu above to modify
Configuration.\n\r"
"</FONT></P></FONT></FONT></TD></TR></TBODY></TABLE></BODY>\n\r";

char code tcpip[] =
"<BODY bgColor=#ffffff>\n\r"
"<TABLE cellPadding=5 border=0 height=1 width=300>\n\r"
"<TD width=501 bgColor=#d4d4d4 height=1>\n\r"
"<form method='post'>"
"<P align=left><font color=#000080 face=Arial size=2><B>TCP/IP
CONFIGURATION:</B></font></P>\n\r"
"<TABLE height='1'><TBODY>\n\r"
"<TR><TD width=275' height='1'>\n\r"
"<HR SIZE=1>\n\r"
"<TABLE><TBODY>\n\r"
"<TR><TD width='175'><b><font face=Arial size=2>IP Address :</TD>\n\r"
"<TD width='100'><INPUT type='text' name=T4 value='' size='13'></TD>\n\r"
"<TR><TD width='175'><b><font face=Arial size=2>Port Address :</TD>\n\r"
"<TD width='100'><INPUT type='text' name=T5 value='' size='13'></TD>\n\r"
"<TR><TD width='175'><b><font face=Arial size=2>Subnet Mask :</TD>\n\r"
"<TD width='100'><INPUT type='text' name=T6 value='' size='13'></TD>\n\r"
"<TR><TD width='175'><b><font face=Arial size=2>Gateway Address :</TD>\n\r"
"<TD width='100'><INPUT type='text' name=T7 value='' size='13'></TD>\n\r"
"</TBODY></TABLE>\n\r"
"<HR SIZE=1>\n\r"
"</TD></TBODY></TABLE>\n\r"

```

```

"<P align=center><FONT color=#218868 size=-1>Changes will take effect after reboot."
"</FONT><BR><INPUT name=post type=submit value='Update TCP/IP Settings'></P>\n\r"
"</FORM></TD></TR></TABLE></BODY>\n\r";

char code serial[] =
"<BODY bgColor=#ffffff>\n\r"
"<TABLE cellPadding=5 border=0 height=1 width=300>\n\r"
"<TD width=501 bgColor=#ddddff height=1>\n\r"
"<form method='post'>\n\r"
"<P align=left><font color=#000080 face=Arial size=2><B>Port Properties:</B></font></P>\n\r"
"<TABLE height='1'><TBODY>\n\r"
"<TR><TD width='275' height='1'>\n\r"
"<HR SIZE=1>\n\r"
"<TABLE><TBODY>\n\r"
"<TR><TD width=175><b><font size=2>Bits per second :</TD>\n\r"
"<TD width=100><b><select size=1 name=S1>\n\r"
"<option>38400</option>\n\r"
"<option>19200</option>\n\r"
"<option>9600</option>\n\r"
"<option>4800</option>\n\r"
"<option>2400</option>\n\r"
"<option>1200</option>\n\r"
"</select></TD>\n\r"
"<TR><TD width=175><b><font size=2>Data bits :</TD>\n\r"
"<TD width=100><b><select size=1 name=S2>"
"<option>8</option>\n\r"
"<option>7</option>\n\r"
"</select></TD>\n\r"
"<TR><TD width=175><b><font size=2>Parity :</TD>\n\r"
"<TD width=100><b><select size=1 name=S3>"
"<option>Ood</option>\n\r"
"<option>Event</option>\n\r"
"<option>None</option>\n\r"
"</select></TD>\n\r"
"<TR><TD width=175><b><font size=2>Stop bits :</TD>\n\r"

```

```
"<TD width=100><b><select size=1 name=S4>"
"<option>1</option>\n\r"
"<option>2</option>\n\r"
"</select></TD>\n\r"
"</TBODY></TABLE>\n\r"
"<HR SIZE=1>\n\r"
"</TD></TBODY></TABLE>\n\r"
"<P align=center><FONT color=#218868 size=-1>Changes will take effect after reboot."
"</FONT><BR><INPUT name=post type=submit value='Update Port Settings'></P>\n\r"
"</FORM></TD></TR></TABLE></BODY>\n\r";
```

```
char code input[] =
```

```
"<BODY bgColor=#ffffff>\n\r"
"<TABLE cellPadding=5 border=0 height=1 width=300>\n\r"
"<TD width=501 bgColor=#d4d4d4 height=1>\n\r"
"<FORM><P align=left><font color=#000080 size=2><B>Monitor mode:</B></font></P>\n\r"
"<TABLE height='1'><TBODY>\n\r"
"<TR><TD width='275' height='1'>\n\r"
"<HR SIZE=1>\n\r"
"<TABLE><TBODY>\n\r"
"<TD width=229><font size=2>INPUT 1</TD>\n\r"
"<TD><select size=1 multiple style='width:100' name=I1></select></TD><TR>\n\r"
"<TD width=229><font size=2>INPUT 2</TD>\n\r"
"<TD><select size=1 multiple style='width:100' name=I2></select></TD><TR>\n\r"
"<TD width=229><font size=2>INPUT 3</TD>\n\r"
"<TD><select size=1 multiple style='width:100' name=I3></select></TD><TR>\n\r"
"<TD width=229><font size=2>INPUT 4</TD>\n\r"
"<TD><select size=1 multiple style='width:100' name=I4></select></TD><TR>\n\r"
/*<TD width=229><font size=2>INPUT 5</TD>\n\r"
"<TD><select size=1 multiple style='width:100' name=I5></select></TD><TR>\n\r"
"<TD width=229><font size=2>INPUT 6</TD>\n\r"
"<TD><select size=1 multiple style='width:100' name=I6></select></TD><TR>\n\r"
"<TD width=229><font size=2>INPUT 7</TD>\n\r"
"<TD><select size=1 multiple style='width:100' name=I7></select></TD><TR>\n\r"
"<TD width=229><font size=2>INPUT 8</TD>\n\r"
```

```
"<TD><select size=1 multiple style='width:100' name=i8></select></TD><TR>\n\r" */
"</TBODY></TABLE>\n\r"
"<HR SIZE=1></TD>\n\r"
"</TBODY></TABLE>\n\r"
"<P align=center><FONT color=#218868 size=-1>Show effect after input update."
"</FONT><BR><BR><INPUT name=post type=submit value='Update Input'></P>\n\r"
"</FORM></TD></TR></TABLE>\n\r"
"<meta http-equiv='refresh' content='1'>"
"</BODY>\n\r";
```

```
char code output[] =
```

```
"<BODY bgColor=#ffffff>\n\r"
"<TABLE cellPadding=5 border=0 height=1 width=300>\n\r"
"<TD width=501 bgColor=#d4d4d4 height=1>\n\r"
"<FORM method='post'><P align=left><font color=#000080 size=2><B>OUTPUT
FUNCTION</B></font></P>\n\r"
"<TABLE height='1'><TBODY>\n\r"
"<TR><TD width='275' height='1'>\n\r"
"<HR SIZE=1>\n\r"
"<TABLE><TBODY>\n\r"
"<TD width=229><font size=2>OUTPUT 1</TD>\n\r"
"<TD><select size=1 style='width:100' name=O1></select></TD><TR>\n\r"
"<TD width=229><font size=2>OUTPUT 2</TD>\n\r"
"<TD><select size=1 style='width:100' name=O2></select></TD><TR>\n\r"
"<TD width=229><font size=2>OUTPUT 3</TD>\n\r"
"<TD><select size=1 style='width:100' name=O3></select></TD><TR>\n\r"
"<TD width=229><font size=2>OUTPUT 4</TD>\n\r"
"<TD><select size=1 style='width:100' name=O4></select></TD><TR>\n\r"
/*<TD width=229><font size=2>OUTPUT 5</TD>\n\r"
"<TD><select size=1 style='width:100' name=O5></select></TD><TR>\n\r"
"<TD width=229><font size=2>OUTPUT 6</TD>\n\r"
"<TD><select size=1 style='width:100' name=O6></select></TD><TR>\n\r"
"<TD width=229><font size=2>OUTPUT 7</TD>\n\r"
"<TD><select size=1 style='width:100' name=O7></select></TD><TR>\n\r"
"<TD width=229><font size=2>OUTPUT 8</TD>\n\r"
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

"<TD><select size=1 style='width:100' name=O8></select></TD><TR>\n\r"*/
"</TBODY></TABLE>\n\r"
"<HR SIZE=1></TD>\n\r"
"</TBODY></TABLE>\n\r"
"<P align=center><FONT color=#218868 size=-1>Changes will take effect after output update.\n\r"
"</FONT><BR><BR><INPUT name=post type=submit value='Update Output'></P>\n\r"
"</FORM></TD></TR></TABLE></BODY>\n\r";

char code home[] =
"<HTML><HEAD>"
"<META http-equiv=Content-Type content='text/html; charset=windows-874'>"
"<META content='MSHTML 6.00.2900.2180' name=GENERATOR></HEAD>"
"<BODY>"
"<TABLE cellPadding=5 width=498 border=0>"
"<TBODY><TR>"
"<th width=484 height=334 align='center' valign='top' nowrap f<TABLE width='435'>"
"<TBODY>"
"<TR >"
"<TD colSpan=2><B><FONT size='2' face=Arial>DEVICE INFORMATION:</FONT></B></TD>"
"<TR><TD width='129' height='20'><font size='2' face='Arial'>Firmware Version :</font></TD>"
"<TD width='245' height='20'><font size='2' face='Arial'>V1.00B02T02.REV.01062006 </font></TD>"
"</TR></TBODY>"
"</TABLE><TABLE width='300'>"
"<TR>"
"<TD colSpan=3><B><FONT size='2' face=Arial>TCP/IP</FONT></B></TD>"
"<TBODY>"
"<TD colSpan='2'><FONT face=Arial size=2>IP Address :</FONT></TD>"
"<TD ><font size='2' face='Arial' name='T4'></font></TD>"
"</TR><TR>"
"<TD colspan='2'><FONT face=Arial size=2>Port :</FONT></TD>"
"<TD><font size='2' face='Arial' name='T5'></font></TD>"
"</TR><TR>"
"<TD colspan='2'><FONT face=Arial size=2>Subnet Mask :</FONT></TD>"
"<TD><font size='2' face='Arial' name='T6'></font></TD>"
"</TR><TR>"

```

```

"<TD colspan='2'><FONT face=Arial size=2>Gateway Address :</FONT></TD>"
"<TD><font size='2' face='Arial' name='T7'></font></TD>"
"</TR>"
"<TR>"
"<TD colspan=3><B><FONT size='2' face=Arial>Serial Port</FONT></B></TD>"
"<TR><TD width='158'><FONT face=Arial size=2>Serial status:</FONT></TD>"
"<TD width='54'><font size='2' face='Arial'>Status</font></TD>"
"<TD><font size='2' face='Arial'></font></TD>"
"</TR><TR>"
"<TD colspan='2'><FONT face=Arial size=2>Bits per second :</FONT></TD>"
"<TD><font size='2' face='Arial'>115200</font></TD>"
"</TR><TR>"
"<TD colspan='2'><FONT face=Arial size=2>Data Bits :</FONT></TD>"
"<TD><font size='2' face='Arial'>8</font></TD>"
"</TR><TR>"
"<TD colspan='2'><FONT face=Arial size=2>Parity :</FONT></TD>"
"<TD><font size='2' face='Arial'>None</font></TD>"
"</TR><TR>"
"<TD colspan='2'><FONT face=Arial size=2>Stop Bits :</FONT></TD>"
"<TD><font size='2' face='Arial'>1</font></TD>"
"</TR><TR>"
"</TBODY></TABLE></th>"
"</TR>"
"</TBODY></TABLE>"
"</BODY></HTML>";

```

```
#endif
```

```

/**(-----)**/
/**(-          END          -)**/
/**(-----)**/

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โปรแกรมควบคุมในส่วนของ Network

```

/**(-----)**/
/**(-          NETWORK-CONTROL.C          -)**/
/**(-----)**/
/**(-          INCLUDE FILE          -)**/
/**(-----)**/

#include "T89C51AC2.H" // INCLUDE FILE AT89C51ED2.H FOR CPU CLASS DESIGN
#include "STDIO.H"     // INCLUDE FILE STDIO.H
#include "STRING.H"   // INCLUDE FILE STRING.H COMPILER APPLICATION
#include "DECLARE.H"  // INCLUDE FILE DECLARE.H
#include "24LC256.H"  // INCLUDE FILE SETVALUE.H
// #include "24LC16.H" // INCLUDE FILE SETVALUE.H
#include "SETVALUE.H" // INCLUDE FILE SETVALUE.H
#include "SERIAL.H"   // INCLUDE FILE SERIAL.H
#include "RTL8019AS.H" // INCLUDE FILE RTL8019AS.H FOR CHIP INTERFACE DESIGN
#include "RTLSYSTEM.H" // INCLUDE FILE GETPACKET.H
#include "ARP.H"      // INCLUDE FILE ARP.H
#include "TCP.H"      // INCLUDE FILE TCP.H
#include "UDP.H"      // INCLUDE FILE TCP.H
#include "HTTP.H"     // INCLUDE FILE HTTP.H
#include "ICMP.H"     // INCLUDE FILE ICMP.H
/**(-----)**/
/**(-          START MAIN PROGRAM          -)**/
/**(-----)**/

void main(void) {
    AUXR = 0x0C;      // T89C51AC2 CONFIG SELECT ERAM.
//  AUXR = 0x18;     // AT89C51ED2 CONFIG SELECT ERAM.
    P1 = 0xFF;       // SET PORT1 IS INPUT AND OUTPUT
// |-
    Init_serial(); // RETURN SERIAL CONFIGURATION TO SET
// |-
    init_rtl8019as(); // ETHERNET SETTING.
// |-

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

Flag_Start1 = 1;

Bytes_write = 0;  // SET BYTE WRITE OF ETHERNET DATA TO 0.
Bytes_read  = 0;  // SET BYTE READ OF ETHERNET DATA TO 0.
Delay(10);      // DELAY.

Y = 0;

while (1) {      // LOOP ALWAYS.
    if (Bytes_write > 0) { // CHECK BYTE WRITE > 0 FROM DATA SERIAL.

        Send_packet(); // SEND DATA AT RECEIVE FROM SERIAL TO ETHERNET
        TCP/IP.
    }
    if (Bytes_read > 0) { // CHECK BYTE READ > 0 FROM DATA ETHERNET TCP/IP.
        for (Y = 0; Y < Bytes_read; Y++) { // I-
            printf("%C", Packet[55+Y]); // SEND DATA AT RECEIVE FROM ETHERNET TCP/IP
            TO SERIAL.
        } // I-
        Bytes_read = 0; // CLEAR BYTE READ OF ETHERNET TCP/IP TO 0.
    }
    Nicint(); // CHECK NIC OF ETHERNET PACKET.
}
}

/**(-----)**/
/**(-          DELAY OF MAIN PROGRAM          -)**/
/**(-----)**/

void Delay(unsigned int Time_count) {
    unsigned int j;
    do {
        for(j=0;j<100;j++) {} Time_count--;
    } while(Time_count > 0);
return;
}

/**(-----)**/
/**(-          END          -)**/
/**(-----)**/

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โปรแกรมควบคุมการทำงานของ RS-232

```

/**(-----)*/
/**(- FILE NAME      : SERIAL.H                -)*/
/**(- LAST UPDATE   : --/02/2007              -)*/
/**(-----)*/
/**(-          DEFINE EXTERN MAIN PROGRAM      -)*/
/**(-----)*/
extern void Init_serial();

/**(-----)*/
/**(-          DEFINE INTERNAL DATA OF SERIAL COMMUNICATION  -)*/
/**(-----)*/
unsigned int  TimeDataSpeed;
unsigned int  DataSpeed;
unsigned char DataBits;
unsigned char Parity;
unsigned char StopBits;
unsigned char SerialSet;

char code DataSpeedDefault[] = "9600";
char code DbitsDefault[]    = "8";
char code ParityDefault[]   = "None";
char code StopBitsDefault[] = "1";

/**(-----)*/
/**(-          INITIAL FUNCTION SERIAL              -)*/
/**(-----)*/
void Init_serial() {
    /* READ DATA SERIAL COMMUNICATION CONFIG FROM EEPROM'
    Y = 0;
    RD_Eeprom(0);
    RD_Eeprom(1);
    for (I = AddEepDataSpeed; I < (AddEepDataSpeed+17); I++) { /* I-
        BufArray[Y++] = RD_Eeprom(I); /* READ DATA SERIAL SPEED

```

```

    if (RD_Eeprom(I) == ' ') {BufArray[Y] = RD_Eeprom(I); break;} // CHECK END READ DATA
SERIAL SPEED FROM EEPROM.

    Delay(500);
} // ←

BufString = BufArray; // MOVE DATA ARRAY TO DATA
STRING

I = 0;
while (1) {if (*BufString == ' ') {Data[I] = *BufString; break;} Data[I++] = *BufString++;} // .
DataSpeed = Val16(Data[0],Data[1],Data[2],Data[3],Data[4],I); // SET STRING DATA TO VALUE
INTERGER.

DataBits = RD_Eeprom(AddEepDbits); // READ DATA SERIAL DBITS FROM
EEPROM.

Delay(100);
Parity = RD_Eeprom(AddEepParity); // READ DATA SERIAL PARITY BITS
FROM EEPROM.

Delay(100);
StopBits = RD_Eeprom(AddEepStopBits); // READ STOP BITS FROM EEPROM.

Delay(100);
// SET TH1 AND PCON AT FEQUENCY. = 18.432 MHz
if (DataSpeed == 38400) {TimeDataSpeed = 1; PCON |= 0x80; TH1 = 251;} // CHECK
BOUD RATE 38400.
else if (DataSpeed == 19200){TimeDataSpeed = 1; PCON |= 0x80; TH1 = 246;} // CHECK
BOUD RATE 19200.
else if (DataSpeed == 9600) {TimeDataSpeed = 1; PCON |= 0x80; TH1 = 236;} // CHECK
BOUD RATE 9600.
else if (DataSpeed == 4800) {TimeDataSpeed = 100; PCON |= 0x80; TH1 = 216;} // CHECK
BOUD RATE 4800.
else if (DataSpeed == 2400) {TimeDataSpeed = 250; PCON |= 0x80; TH1 = 176;} // CHECK
BOUD RATE 2400.
else if (DataSpeed == 1200) {TimeDataSpeed = 500; PCON |= 0x80; TH1 = 96;} // CHECK
BOUD RATE 1200.
else if (DataSpeed == 600) {TimeDataSpeed = 750; PCON = 0x00; TH1 = 96;} // CHECK
BOUD RATE 600.

// SET SCON AND TMOD

if ((DataBits == '7') && (Parity == 'O')) {SCON = 0x50; TMOD = 0x20; SerialSet = 0x00;}

```

```

else if ((DataBits == '7') && (Parity == 'E')) {SCON = 0x50; TMOD = 0x20; SerialSet = 0x01;}
else if ((DataBits == '7') && (Parity == 'N')) {SCON = 0x50; TMOD = 0x20; SerialSet = 0x02;}
else if ((DataBits == '8') && (Parity == 'O')) {SCON = 0xD0; TMOD = 0x20; SerialSet = 0x03;}
else if ((DataBits == '8') && (Parity == 'E')) {SCON = 0xD0; TMOD = 0x20; SerialSet = 0x04;}
else if ((DataBits == '8') && (Parity == 'N')) {SCON = 0x50; TMOD = 0x20; SerialSet = 0x05;}

// SET SCON OF STOPBITS

if ((DataBits == '7') && (Parity == 'N') && (StopBits == '2')) {SCON = 0x50; TMOD = 0x20;
SerialSet = 0x06;}

TR1= 1; // TIMER 1 RUN.
TI = 0; // SET TI SEND FIRST CHAR OF UART.

return;
}

/**(-----)*/
/**(- APPLICATIONS SEND DATA -)*/
/**(-----)*/

char putchar(unsigned char c) {
    switch (SerialSet) {
        case 0x00: // 7,O
            ACC = c;
            if (P) c &= 0x7F;
            else c |= 0x80;
            break;

        case 0x01: // 7,E
            ACC = c;
            if (P) c |= 0x80;
            else c &= 0x7F;
            break;

        case 0x02: // 7,N
            c |= 0x80;
            break;

        case 0x03: // 8,O
            ACC = c;
            if (P == 1) TB8 = 0;
            else TB8 = 1;

```

```

break;

    case 0x04:  //' 8,E
ACC = c;
if (P) TB8 = 1;
else TB8 = 0;
break;

    case 0x05:  //' 8,N  //- Defualt
//' ACC = c;
break;
case 0x06:  //' 7,N,2
c |= 0x80;
break;
}
SBUF = c;
while(!TI);
TI = 0;  //' CLEAR BIT FLAG TI_0
return (c);  //' MOV DATA TO SERIAL
}

/**(-----)*/
/**(- APPLICATIONS RECEIVE DATA -)*/
/**(-----)*/

void RxdReceive () interrupt 4 {
    unsigned char c;
    unsigned int TimeOut;
    TimeOut = 0;
    if (RI == 1) { // Check Interrupt Rxd receive Data
        while (TimeOut++ <= TimeDataSpeed) {
            if (RI == 1) {
                switch (SerialSet) {
                    case 0x00:  //' 7,O
                        ACC = SBUF;
                        if (P) c = SBUF & 0x7F;
                        else c = SBUF | 0x80;
                        break;

```

```

case 0x01:  //' 7,E
    ACC = SBUF;
    if (P) c = SBUF | 0x80;
    else c = SBUF & 0x7F;
    break;
case 0x02:  //' 7,N,1
    c = SBUF & 0x7F;
    break;
case 0x03:  //' 8,O
    c = SBUF;
    break;
case 0x04:  //' 8,E
    c = SBUF;
    break;
case 0x05:  //' 8,N
    c = SBUF;
    break;
case 0x06:  //' 7,N,2
    c = SBUF & 0x7F;
    break;
}
AuxData[Bytes_write++] = c; //
RI = 0;
TimeOut = 0;
}
}
}
return;
}

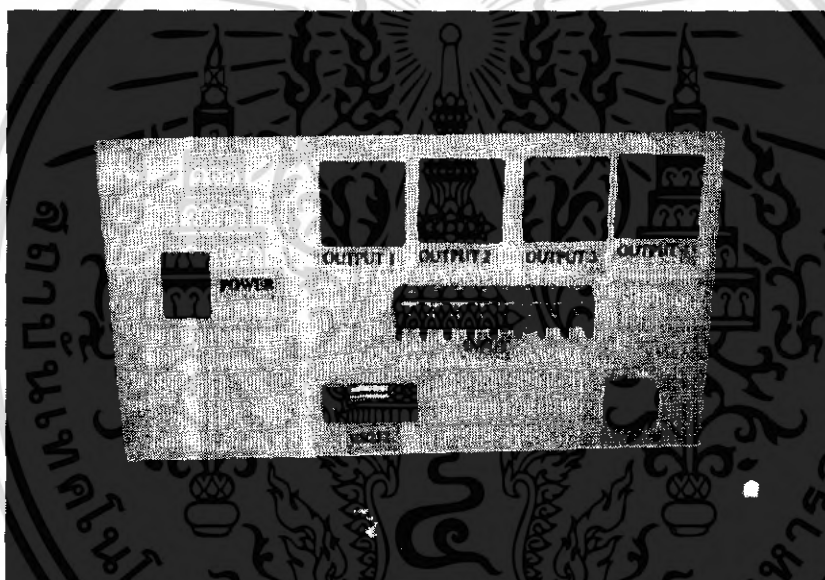
/**(-----)*/
/**(-          END          -)*/
/**(-----)*/

```



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

คู่มือการใช้งาน
เครื่องควบคุมระบบเปิด - ปิดและรายงานสถานะสวิทช์ผ่านเครือข่าย
อินเทอร์เน็ต



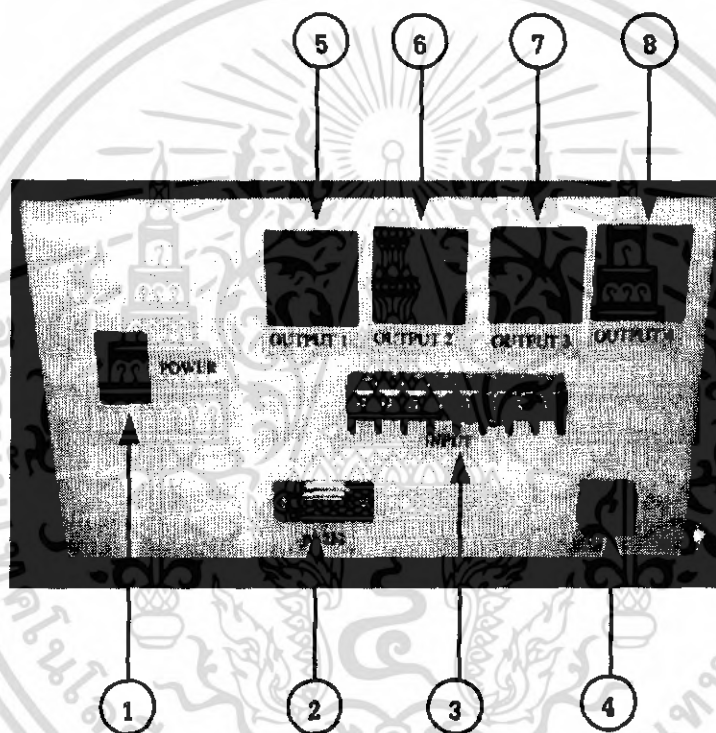
ภาควิชาครุศาสตร์วิศวกรรม
คณะครุศาสตร์อุตสาหกรรม
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
ปีการศึกษา 2549

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1. คำแนะนำเบื้องต้น

ก่อนที่จะลงมือใช้งานเครื่องยืมหนังสือด้วยตนเอง ควรทำการศึกษาการใช้งานจากคู่มือให้เข้าใจ เพื่อการบริการที่ถูกต้อง และเพื่อป้องกันการเสียหายที่เกิดขึ้นกับเครื่องควบคุมระบบเปิด - ปิดและรายงานสถานะสวิตช์ผ่านเครือข่ายอินเทอร์เน็ต

2. ส่วนประกอบและปุ่มควบคุม



1. ปุ่ม Power ใช้สำหรับการเปิด-ปิด เครื่องควบคุม
2. ปุ่ม DB-9 หรือ RS-232 ใช้สำหรับส่งควบคุมโปรแกรม
3. ปุ่มควบคุมการทำงานทางด้านอินพุต (แต่ในที่นี้ ไม่ได้ใช้งาน)
4. ช่องเสียบ UTP สำหรับเชื่อมต่อกับสาย LAN
5. ปุ่มควบคุมการทำงานด้านเอาต์พุตของเครื่องใช้ไฟฟ้าเครื่องที่ 1
6. ปุ่มควบคุมการทำงานด้านเอาต์พุตของเครื่องใช้ไฟฟ้าเครื่องที่ 2
7. ปุ่มควบคุมการทำงานด้านเอาต์พุตของเครื่องใช้ไฟฟ้าเครื่องที่ 3
8. ปุ่มควบคุมการทำงานด้านเอาต์พุตของเครื่องใช้ไฟฟ้าเครื่องที่ 4

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3. การติดตั้งและการใช้งาน

- 3.1 เสียบปลั๊กไฟฟ้ากระแสสลับ 220 โวลต์
- 3.2 เปิดสวิตซ์การทำงานของเครื่อง
- 3.3 ต่อสาย LAN แบบไขว้ เข้ากับเครื่องควบคุมและเครื่องคอมพิวเตอร์
- 3.4 ทำการ Start Video Encoders ที่กำหนดไว้
- 3.5 เปิดโปรแกรมอินเทอร์เน็ต
- 3.6 พิมพ์ URL: localhost
- 3.7 ใส่ USER NAME และ Password ที่กำหนดไว้
- 3.8 ควบคุมการทำงานผ่านโปรแกรมอินเทอร์เน็ต
- 3.9 รอดูภาพการทำงานผ่านกล้อง Web Cam

4. การแก้ปัญหาเบื้องต้น

เมื่อท่านประสบปัญหาในการใช้งานเครื่องระบบเปิด - ปิดและรายงานสถานะสวิตซ์ผ่านเครือข่ายอินเทอร์เน็ต สามารถตรวจสอบแนวทางแก้ไขปัญหาเบื้องต้นได้จากตารางข้างล่างนี้

ตารางที่ ข.1 การแก้ปัญหาเบื้องต้น

อาการ	สาเหตุและวิธีแก้ไข
เครื่องควบคุมระบบไม่ทำงาน	ตรวจสอบสาย LAN ที่ช่องเสียบ UTP ว่าต่ออยู่หรือไม่
ไม่สามารถควบคุมการสั่งงานผ่านอินเทอร์เน็ตได้	ตรวจสอบหมายเลข IP Address ว่ามีการตั้งค่าไว้ถูกต้องหรือไม่ ถ้าไม่ให้ทำการตั้งค่า IP เสียก่อน
กล้อง Web Cam ไม่สามารถดูภาพได้	ตรวจสอบปุ่ม Start Encoding บนโปรแกรม Window Media Encoder
หลอดไฟฟ้าไม่สว่าง	ตรวจสอบว่าใส่หลอดขาดหรือไม่ ถ้าไม่ให้ตรวจสอบวงจรภายใน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5. การดูแลรักษาและข้อควรระวัง

5.1 การดูแลรักษา

1. เช็ดทำความสะอาดตัวเครื่องด้วยผ้านุ่ม อย่าใช้สารใดๆ ที่เป็นตัวทำลายเพราะอาจทำให้ตัวเครื่องเป็นรอยเสียหาย

2. ตรวจสอบขั้วต่อสายไฟของวงจรภายในเครื่องให้อยู่ในสภาพพร้อมใช้งานเสมอ

3. ควรมีการซ่อมบำรุงตัวเครื่องเป็นระยะเพื่อป้องกัน และลดอัตราการเสื่อมสภาพของตัวเครื่อง เพื่อให้การใช้งานของตัวเครื่องเป็นไปอย่างมีประสิทธิภาพ

5.2 ข้อควรระวัง

1. ควรศึกษาคู่มือการใช้งานของเครื่องก่อนการใช้งานตัวเครื่อง

2. ควรวางตัวเครื่องใกล้กับเครื่องคอมพิวเตอร์เพื่อความสะดวกในการใช้งาน

3. การเคลื่อนย้ายควรระวังอย่าให้มีการกระแทกเพื่อป้องกันความเสียหายของวงจรภายในต่างๆ ของเครื่อง

6. ข้อมูลจำเพาะ

ตารางที่ ๕.2 ข้อมูลจำเพาะ

คุณสมบัติ	รายละเอียด
เครื่องควบคุมระบบเปิด - ปิดและรายงานสถานะสวิตช์ผ่านเครือข่ายอินเทอร์เน็ต	ใช้ควบคุมระบบเปิด - ปิดหลอดไฟฟ้าภายในบ้านพักอาศัยหรือสถานที่อื่นๆ
การเชื่อมต่อกับระบบอินเทอร์เน็ต	เชื่อมต่อกับ HUB ผ่านระบบอินเทอร์เน็ต
การเชื่อมต่อกับกล้อง Web Cam	กล้อง Web Cam เชื่อมต่อกับเครื่องคอมพิวเตอร์ผ่าน USB Port
การเปิด - ปิดการทำงาน	สามารถเปิด - ปิดการทำงานของหลอดไฟ 4 หลอด
ความเร็วในการทำงาน	เปิด - ปิดหลอดไฟ 1 ดวงใช้เวลา 1 วินาที
แหล่งจ่ายพลังงาน	ไฟฟ้ากระแสสลับ 220 โวลต์ ความถี่ 50-60 เฮิรตซ์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ประวัติผู้แต่ง



ชื่อ-สกุล	นายโชคชัย สุธา
วัน เดือน ปีเกิด	16 มกราคม พ.ศ. 2527
ภูมิลำเนา	77/1 หมู่ 8 ตำบลทุ่งกระเบื้อง อำเภอยานตาขาว จังหวัดตรัง 92140 โทรศัพท์ 083-7161551
ประวัติการศึกษา	
ประถมศึกษา	โรงเรียนวัดควนอินทนิพนาม
มัธยมศึกษาตอนต้น	ศูนย์การศึกษาออกโรงเรียน
ประกาศนียบัตรวิชาชีพ	วิทยาลัยการอาชีพตรัง
ประกาศนียบัตรวิชาชีพชั้นสูง	วิทยาลัยเทคนิคตรัง
ปริญญาตรี	สาขาวิชาวิศวกรรมอิเล็กทรอนิกส์ ภาควิชาวิศวกรรมอิเล็กทรอนิกส์ คณะวิศวกรรมศาสตร์อุตสาหกรรม สจล.
คติพจน์	ชื่อสัตย์ สุจริต จะเป็นดีเลิศ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ประวัติผู้แต่ง



ชื่อ-สกุล	นายศักดิ์ชัย รอดคง
วัน เดือน ปีเกิด	3 มีนาคม พ.ศ. 2527
ภูมิลำเนา	49/1 หมู่ 2 ตำบลน้ำขาว อำเภอจะนะ จังหวัดสงขลา 90130 โทรศัพท์ 081-5993093
ประวัติการศึกษา	
ประถมศึกษา	โรงเรียนวัดน้ำขาวใน
มัธยมศึกษาตอนต้น	โรงเรียนจะนะชนูปถัมภ์
ประกาศนียบัตรวิชาชีพ	วิทยาลัยการอาชีพนาทวี
ประกาศนียบัตรวิชาชีพชั้นสูง	สถาบันเทคโนโลยีราชมงคล วิทยาเขตภาคใต้ สงขลา
ปริญญาตรี	สาขาวิศวกรรมอิเล็กทรอนิกส์ ภาควิชาครุศาสตร์วิศวกรรม คณะครุศาสตร์อุตสาหกรรม สจล.
คติพจน์	คนที่ไม่เคยทำผิดพลาด นั่นคือคนที่ไม่เคยทำอะไรเลย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ประวัติผู้แต่ง



ชื่อ-สกุล	นายอภิญา แซ่ลิ่ม
วัน เดือน ปีเกิด	22 สิงหาคม พ.ศ.2525
ภูมิลำเนา	487/31 ซอย 35 ถนนเพชรเกษม ตำบลควนลัง อำเภอหาดใหญ่จังหวัดสงขลา 90110 โทรศัพท์ 081-5417422
ประวัติการศึกษา	
ประถมศึกษา	โรงเรียนกิติวิทย์
มัธยมศึกษาตอนต้น	โรงเรียนศรีนคร
ประกาศนียบัตรวิชาชีพ	วิทยาลัยเทคนิคหาดใหญ่
ประกาศนียบัตรวิชาชีพชั้นสูง	สถาบันเทคโนโลยีราชมงคล วิทยาเขตภาคใต้ สงขลา
ปริญญาตรี	สาขาวิศวกรรมอิเล็กทรอนิกส์ ภาควิชาวิศวกรรมอิเล็กทรอนิกส์ คณะวิศวกรรมศาสตร์อุตสาหกรรม สจล.
คดีพจนี	ย้อนมองส่องตน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้