สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

เว็บเซิร์ฟเวอร์โดยใช้ไมโครคอนโทรลเลอร์

Web Server by Microcontroller

โดย

นายเทวา เชิดสูงเนิน

นายเทียนชัย สุริยาอมรชัย

เลขหมู่.....
เลขทะเบียน.....62904
วัน,เดือน,ปี.....23 ส.ค. 2549

b.....11633427
i.....

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต

สาขาวิศวกรรมแมคคาทรอนิกส์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา พ.ศ.2548

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ปริญญานิพนธ์ปีการศึกษา 2548

สาขาวิศวกรรมแมคคาทรอนิกส์

คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เรื่อง เว็บเซิร์ฟเวอร์โดยใช้ไมโครคอนโทรลเลอร์

Web Server by Microcontroller

ผู้จัดทำ

1. นายเทวา เชิดธูงเนิน รหัสนักศึกษา 45010308

2. นายเทียนชัย ธุริยาอมรชัย รหัสนักศึกษา 45010309

.....อาจารย์ที่ปรึกษา

(รศ.ดร. วันชัย ริ้วรุจา)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เว็บเซิร์ฟเวอร์โดยใช้ไมโครคอนโทรลเลอร์

Web Server by Microcontroller

โดย

นายเทวา เชิดสูงเนิน 45010308

นายเทียนชัย สุริยาอมรชัย 45010309

อาจารย์ที่ปรึกษา

รศ.ดร. วันชัย ริ้วรุจา

อาจารย์ที่ปรึกษาร่วม

อาจารย์ ธวัชชัย คำศรี

บทคัดย่อ

โดยทั่วไปคอมพิวเตอร์ที่ทำหน้าที่เป็นเว็บเซิร์ฟเวอร์ จะต้องมีส่วนประกอบหลัก 3 ส่วนด้วยกัน คือ หน่วยประมวลผลกลางหรือซีพียูซึ่งทำหน้าที่จัดการประมวลผลคำสั่งต่างๆ เน็ตเวิร์คการ์ดซึ่งทำหน้าที่เชื่อมต่อระหว่างเครื่องเซิร์ฟเวอร์กับระบบอินเตอร์เน็ต และฮาร์ดดิสก์ซึ่งทำหน้าที่เก็บข้อมูลเว็บเพจ สำหรับการสร้างเว็บเซิร์ฟเวอร์ขนาดเล็กการใช้คอมพิวเตอร์มาสร้างนั้นจะทำให้สิ้นเปลืองค่าใช้จ่ายเกินไป ดังนั้นปริญญานิพนธ์ฉบับนี้จึงนำเสนอวิธีการสร้างเว็บเซิร์ฟเวอร์ โดยการนำไมโครคอนโทรลเลอร์ PIC16F877 มาประยุกต์ใช้แทนในส่วนของหน่วยประมวลผลกลาง ส่วนการเชื่อมต่อกับระบบอินเตอร์เน็ตจะใช้ไอซี RTL8019AS แทนส่วนของเน็ตเวิร์คการ์ด และส่วนข้อมูลเว็บเพจจะถูกเก็บไว้ในหน่วยความจำข้อมูลภายนอกแบบอีอีพรอมแทนส่วนฮาร์ดดิสก์ ซึ่งเราจะได้เว็บเซิร์ฟเวอร์ที่มีขนาดเล็ก ราคาถูกและสะดวกในการเคลื่อนย้าย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เว็บเซิร์ฟเวอร์โดยใช้ไมโครคอนโทรลเลอร์

Web Server by Microcontroller

By

Mr. Tewa Cherdsoongnern 45010308

Mr. Tianchai Suriyaamornchai 45010309

Advisor

Assoc.Prof.Dr. Vanchai Riewruja

Co-Advisor

Mr. Thawatchai Kamsri

ABSTRACT

Typically a computer which works as a web server must has three main components. The central processing unit or CPU, the network interface card, and the storage device. The implement of a small web server using personal computer (PC) will take much cost. So this thesis posses another method to implement web server by using a microcontroller "PIC16F877" instead of the PC and IC "RTL8019AS" instead of the network interface card and a peripheral memory for containing webpage data. Finally we got a small web server, which has low cost and comfortable to move everywhere.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กิตติกรรมประกาศ

ปริญญานิพนธ์ฉบับนี้สามารถสำเร็จลุล่วงไปได้ด้วยดี ด้วยคำแนะนำ และคำปรึกษาจาก รศ.ดร. วันชัย ริ้วรุจา ซึ่งเป็นอาจารย์ผู้ควบคุมวิทยานิพนธ์ และ อาจารย์ธวัชชัย คำศรี ที่ให้ความกรุณาเอื้อเฟื้ออุปกรณ์ที่จำเป็น และความช่วยเหลืออื่นๆ แก่คณะผู้จัดทำ คณะผู้จัดทำรู้สึกซาบซึ้งในความอนุเคราะห์จากท่านอาจารย์ทั้งสองท่าน และขอขอบพระคุณเป็นอย่างสูง

ขอกราบขอบพระคุณคณาจารย์ภาควิชาวิศวกรรมระบบควบคุม คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบังทุก ๆ ท่านที่ได้ประสิทธิ์ประสาทวิชาแก่คณะผู้จัดทำ

ขอขอบคุณเพื่อนๆ พี่ๆ น้องๆ ในภาควิชาวิศวกรรมระบบควบคุม สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบังทุกคนที่ให้คำแนะนำต่างๆ และคอยให้กำลังใจเสมอมา

ขอขอบคุณบริษัทไมโครชิปที่สนับสนุนบอร์ดทดลอง *PICDEM.net* เพื่อใช้ในการทำการทดลอง ในการศึกษาการเชื่อมต่อระบบอินเตอร์เน็ต

สุดท้ายนี้ข้าพเจ้าขอกราบขอบพระคุณ บิดา มารดา และครอบครัวของคณะผู้จัดทำที่เป็นกำลังใจ และให้การสนับสนุนในทุกเรื่องๆ ทำให้ข้าพเจ้าสามารถทำปริญญานิพนธ์ฉบับนี้สำเร็จลุล่วงด้วยดี

คุณค่าและประโยชน์อันพึงมาจากปริญญานิพนธ์ฉบับนี้ ข้าพเจ้าขอมอบแด่ผู้มีพระคุณทุกท่าน

คณะผู้จัดทำ

นายเทวา เชิดสูงเนิน

นายเทียนชัย สุริยาอมรชัย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ

	หน้า
บทคัดย่อภาษาไทย	I
บทคัดย่อภาษาอังกฤษ	II
กิตติกรรมประกาศ	III
สารบัญ	IV
สารบัญภาพ	VII
สารบัญตาราง	IX
บทที่ 1 บทนำ	1
1.1 วัตถุประสงค์ของโครงงาน	1
1.2 ขอบเขตของโครงงาน	2
1.3 วิธีการดำเนินการ	2
1.4 ประโยชน์ที่คาดว่าจะได้รับ	3
1.5 ส่วนประกอบของปริญญานิพนธ์	3
บทที่ 2 ทฤษฎีพื้นฐานที่ใช้ในโครงงาน	4
2.1 ความหมายของเครือข่ายคอมพิวเตอร์	4
2.2 เน็ตเวิร์คการ์ด	5
2.3 สายสัญญาณ	6
2.4 อุปกรณ์ในระบบเครือข่าย	6
2.4.1 เราท์เตอร์(Router)	6
2.4.2 ฮับ (Hub)	7
2.4.3 สวิตช์(Switch)	7
2.5 เครือข่ายแบบท้องถิ่น(Local Area Network)	8
2.5.1อีเธอร์เน็ต	9
2.6 ความหมายของเซิร์ฟเวอร์(Server)	10
2.7 เครือข่ายแบบไคลเอนท์เซิร์ฟเวอร์(Client/Server Network)	10
2.7.1 ชนิดของเซิร์ฟเวอร์ที่ให้บริการในรูปแบบต่างๆ	11
2.8 อินเตอร์เน็ต(Internet)	12
2.9 โปรโตคอล คืออะไร ?	13
2.10 โปรโตคอล TCP/IP	14

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ (ต่อ)

	หน้า
2.11 โปรโตคอลที่ทำงานในชั้นอินเตอร์เน็ต (Internet Layer)	15
2.11.1 Internet Protocol (IP)	15
2.11.2 Address Resolution Protocol (ARP)	17
2.11.3 Internet Control Message Protocol (ICMP)	17
2.11.4 กลไกการทำงานของโปรโตคอล IP	18
2.11.4.1 หมายเลขไอพี (IP Address) หรือ URL (Uniform Resource Locator)	18
2.11.4.2 Domain Name System (DNS)	19
2.11.4.3 Mac address (Media Access Control address)	20
2.12 โปรโตคอลที่ทำงานในชั้นโฮสต์ทูโฮสต์ (Host to Host Layer)	20
2.12.1 Transmission Control Protocol (TCP)	20
2.12.1.1 การจัดการเกี่ยวกับเซสชั่น	20
2.12.1.2 การควบคุมการไหลและการกู้ข้อมูลคืน	21
2.12.2 การทำงานของ TCP/IP Stack	24
2.13 แอพพลิเคชั่นเลเยอร์(Application layer)	28
2.13.1 เวิร์ลไวด์เว็บ (World Wide Web): HTTP (Hyper Text Transfer Protocol)	28
2.13.1.1 กลไกการทำงานของโปรโตคอล HTTP	28
2.13.1.2 HTML (Hyper Text Markup Language)	30
2.13.2 การทดลองเขียนเว็บเพจอย่างง่ายๆ	32
บทที่ 3 บอร์ดทดลอง PICDEM.net	35
3.1 อุปกรณ์ที่มีมากับบอร์ดทดลอง	35
3.1.1 อุปกรณ์ที่มีบนบอร์ดทดลอง	36
3.2 อุปกรณ์หลักบนบอร์ดทดลองที่ใช้ในโครงงานนี้	37
3.3 ความรู้เบื้องต้นของไมโครคอนโทรลเลอร์ PIC16F877	37
3.3.1 โครสร้างของไมโครคอนโทรเลอร์ PIC16F877	38
3.3.2 คุณสมบัติทางเทคนิคของ PIC16F877	38
3.3.3 การจัดการขาของไมโครคอนโทรลเลอร์ PIC16F877	39

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ (ต่อ)

	หน้า
3.3.4 การจัดหน่วยความจำของไมโครคอนโทรลเลอร์ PIC 16F877	40
3.4 ความรู้เบื้องต้นเกี่ยวกับไอซี RTL8019AS	42
3.4.1 คุณสมบัติของไอซี REALTEK RTL8019AS	42
3.5 ความรู้เบื้องต้นเกี่ยวกับหน่วยความจำภายนอกแบบอีอีพรอม	43
3.6 การทำงานของบอร์ดทดลอง PICDEM.net	44
บทที่ 4 เริ่มการใช้บอร์ดทดลอง	47
4.1 การเขียนโปรแกรมติดต่อกับระบบอินเตอร์เน็ต	47
4.1.1 โปรแกรมส่วนอีเธอร์เน็ตไดรฟ์เวอร์	47
4.1.2 โปรแกรมการรับแพ็กเก็ต	50
4.1.3 โปรแกรมการวิเคราะห์แพ็กเก็ต	52
4.1.4 โปรแกรมการส่งแพ็กเก็ต	53
4.1.5 โปรแกรมการติดต่อกับจอแสดงผล	54
4.1.6 โปรโตคอล ARP	56
4.1.7 โปรโตคอล IP	57
4.1.8 โปรโตคอล ICMP	58
4.1.9 โปรโตคอล TCP	60
4.2 เขียนเว็บเพ็จเพื่อเก็บไว้ในหน่วยความจำภายนอกอีอีพรอม	63
4.3 กำหนดการทำงานของบอร์ดทดลอง	65
4.4 การใช้งานบอร์ดเป็นเว็บเซิร์ฟเวอร์	70
บทที่ 5 บทวิจารณ์และสรุป	73
5.1 บทสรุป	73
5.2 วิจารณ์สิ่งที่ได้จากโครงงาน	73
5.3 ปัญหาอุปสรรคและแนวทางในการแก้ไข	73
5.4 แนวทางการพัฒนาต่อ	74
ภาคผนวก ก	75
บรรณานุกรม	79

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญภาพ

รูปที่	หน้า
1.1 การติดต่อระหว่างคอมพิวเตอร์กับเซิร์ฟเวอร์	1
2.1 องค์ประกอบพื้นฐานของระบบเครือข่าย	5
2.2 เน็ตเวิร์คการ์ด	5
2.3 สายสัญญาณคู่บิดเกลียวแบบ UTP ที่มีหัว RJ-45 เชื่อมต่ออยู่	6
2.4 เราท์เตอร์	7
2.5 ฮับ (HUB)	7
2.6 สวิทช์ Switch	8
2.7 Local Area Network (LAN)	8
2.8 การเรียกชื่ออีเธอร์เน็ต	9
2.9 เครือข่ายแบบไคลเอนท์เซิร์ฟเวอร์	11
2.10 เครือข่ายที่ทำการเชื่อต่อกับระบบอินเตอร์เน็ต	13
2.11 การทำงานของโปรโตคอล TCP	24
2.12 การทำงานของโปรโตคอล IP	25
2.13 การทำงานของเราท์เตอร์	26
2.14 การทำงานของโปรโตคอล IP เมื่อมีการรับแพ็กเก็ต	27
2.15 การทำงานของโปรโตคอล HTTP	28
2.16 แสดงหน้าต่าง Microsoft Internet Explorer	29
2.17 แสดงหน้าต่างเว็บเพ็จ	30
2.18 แสดงหน้าต่างโปรแกรม Notepad	32
2.19 แสดงการบันทึกไฟล์	33
2.20 แสดงลักษณะของไฟล์เว็บเพ็จ	33
2.21 แสดงเว็บเพ็จที่ได้ทำการเขียนเอาไว้	34
3.1 บอร์ดทดลอง PICDEM.net	35
3.2 ไมโครคอนโทรเลอร์ PIC16F877	37
3.3 แสดงหน่วยความจำโปรแกรมและหน่วยความจำข้อมูล	37
3.4 ขาของไมโครคอนโทรลเลอร์	40
3.5 ไอซีติดต่อระบบอินเตอร์เน็ต RTL8019AS	43
3.6 การติดต่อกันระหว่างไมโครคอนโทรลเลอร์กับ RTL8019AS	43

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญภาพ (ต่อ)

รูปที่	หน้า
3.7 อีอีพรอม	44
3.8 บล็อกไดอะแกรมองค์ประกอบของบอร์ดทดลอง PICDEM.net ที่ใช้สร้างเว็บเซิร์ฟเวอร์	44
3.9 โฟลว์ชาร์ตการทำงานของไมโครคอนโทรลเลอร์	45
4.1 โครงสร้างของ Network Interface Controller	47
4.2 อีเธอร์เน็ตเฟลม	50
4.3 แสดงที่อยู่ของไฟล์ตัวอย่างที่จะนำมาแปลงให้เป็น .ROM	64
4.4 แสดงตัวอย่างการรันโปรแกรม WEBROM.EXE	64
4.5 การเชื่อมต่อบอร์ดทดลองกับเครื่องคอมพิวเตอร์	66
4.6 หน้าต่างโปรแกรม Hyper Terminal	66
4.7 หน้าต่างถามเลขอนุกรมของบอร์ดทดลอง	67
4.8 หน้าต่างถามเลขไอพีของบอร์ดทดลอง	67
4.9 แสดงหน้าต่างการรอโหลดโปรแกรมลงบอร์ดทดลอง	68
4.10 แสดงการโหลดเว็บเพ็จ	68
4.11 แสดงการเลือกเว็บเพ็จที่ต้องการจะโหลดลงบอร์ดทดลอง	69
4.12 แสดงหน้าต่างเมื่อโหลดเว็บเพ็จเสร็จ	69
4.13 การเชื่อมต่อบอร์ดทดลองเพื่อใช้งานเป็นเซิร์ฟเวอร์	70
4.14 แสดงการ Enable LAN	71
4.15 แสดงหน้าต่างของ Internet Explorer Browser	71
4.16 แสดงเว็บเพ็จที่ได้ถูกเก็บไว้ในอีอีพรอมของบอร์ดทดลอง	72
ก.1 รูปวงจรการเชื่อมต่อไมโครคอนโทรลเลอร์กับอุปกรณ์ต่างๆ	67
ก.2 รูปวงจรการเชื่อมต่อของไอซี MAX 232 ACSE สำหรับต่อกับพอร์ตอนุกรม	67
ก.3 รูปวงจรการเชื่อมต่อของไอซี 24LC256	68
ก.4 รูปวงจรการเชื่อมต่อจอแสดงผล	68
ก.5 รูปวงจรการเชื่อมต่อไอซีปรับแรงดัน	68
ก.6 รูปวงจรการเชื่อมต่อของไอซี RTL8019AS	69

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญตาราง

ตารางที่	หน้า
2.1 แสดงความสัมพันธ์ระหว่างโปรโตคอลต่างๆของ TCP/IP	14
2.2 โปรโตคอลต่างๆของ TCP/IP ที่นำมาใช้สำหรับในการรับส่งเว็บเพ็จทางอินเตอร์เน็ต	14
2.3 แสดงฟอร์แมตของแพ็กเก็ต IP	16
2.4 ฟอร์แมตข้อมูลของแพ็กเก็ต TCP	22

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 1

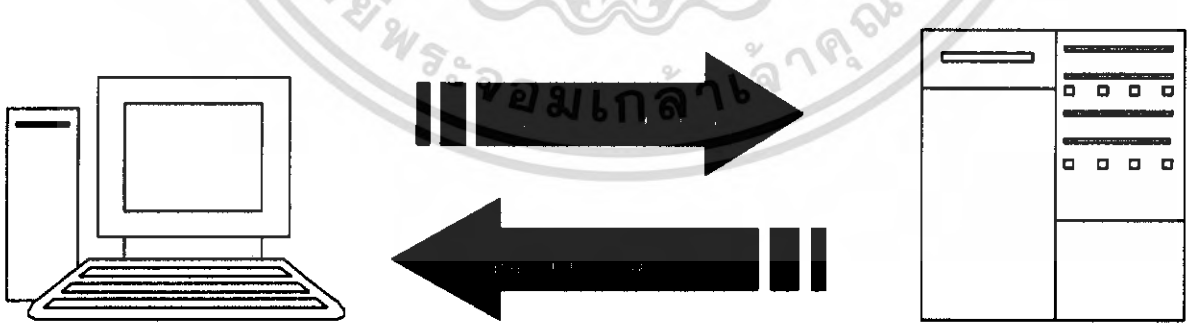
บทนำ

ปัจจุบันเทคโนโลยีทางด้านอินเตอร์เน็ต (Internet) ได้เข้ามามีบทบาทสำคัญอย่างมากในชีวิตประจำวัน การติดต่อสื่อสารหรือการสั่งงานต่างๆสามารถเชื่อมโยงกันได้ทั่วโลกด้วยระบบอินเตอร์เน็ต เราสามารถนำระบบอินเตอร์เน็ตไปประยุกต์ใช้ในการสั่งงานต่างๆ ผ่านเครือข่ายอินเตอร์เน็ต หรือทำการเรียกดูข้อมูลที่ทำหน้าที่ช่วยตรวจสอบสถานะปัจจุบันของแหล่งข้อมูลที่อยู่ห่างไกลออกไปได้ ซึ่งความสามารถนี้ช่วยเพิ่มความสะดวกรวดเร็วในการเข้าถึงข้อมูลได้อย่างมีประสิทธิภาพ ดังนั้นการศึกษาถึงการใช้ระบบอินเตอร์เน็ตในการส่งผ่านข้อมูลจึงมีประโยชน์และมีความสำคัญมากในปัจจุบัน

1.1 วัตถุประสงค์ของโครงงาน

ปริญญานิพนธ์ฉบับนี้มุ่งหวังเพื่อศึกษาระบบการส่งผ่านข้อมูลผ่านทางระบบอินเตอร์เน็ต ซึ่งเป็นเทคนิคหนึ่งที่จะถูกนำไปใช้ในการสื่อสารอย่างแพร่หลายในอนาคตอันใกล้ ซึ่งเป็นการติดต่อสื่อสารข้อมูลที่สะดวกรวดเร็วและแม่นยำ

สำหรับการติดต่อสื่อสารกันผ่านทางระบบอินเตอร์เน็ตนั้น จำเป็นจะต้องมีคอมพิวเตอร์อย่างน้อยสองเครื่อง เพื่อทำหน้าที่เป็นเครื่องเซิร์ฟเวอร์ (ผู้ให้บริการ) และเครื่องไคลเอ็นท์ (ผู้ใช้บริการ) โดยเครื่องไคลเอ็นท์จะทำการส่งสัญญาณร้องขอเรียกดูข้อมูลไปยังเครื่องเซิร์ฟเวอร์ จากนั้นเครื่องเซิร์ฟเวอร์จะทำการประมวลผลและส่งข้อมูลที่เครื่องไคลเอ็นท์ต้องการกลับไป ดังแสดงไว้ในรูปที่ 1.1



รูปที่ 1.1 การติดต่อระหว่างคอมพิวเตอร์กับเซิร์ฟเวอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การรับส่งข้อมูลที่มีขนาดเล็กและไม่สลับซับซ้อนผ่านทางระบบอินเตอร์เน็ตควรจะมีอุปกรณ์ที่สามารถใช้งานแทนเครื่องคอมพิวเตอร์ที่ทำหน้าที่เป็นเครื่องเซิร์ฟเวอร์ที่มีราคาแพงได้ ทำให้เกิดการประยุกต์ใช้งานไมโครคอนโทรเลอร์แทนการใช้เครื่องคอมพิวเตอร์ที่ทำหน้าที่เป็นเครื่องเซิร์ฟเวอร์ เพื่อทำให้ต้นทุนในการติดต่อสื่อสารข้อมูลผ่านทางระบบอินเตอร์เน็ตลดลง และเป็นการประหยัดค่าใช้จ่ายในการบำรุงรักษา ทั้งยังสามารถเคลื่อนย้ายได้อย่างสะดวกอีกด้วย

ภายในคอมพิวเตอร์ส่วนที่เป็นเซิร์ฟเวอร์นั้นจะประกอบไปด้วยส่วนสำคัญ 3 อย่าง คือ หน่วยประมวลผลกลาง(CPU) ซึ่งทำหน้าที่ประมวลผลและคำนวณคำสั่งต่างของคอมพิวเตอร์, เน็ตเวิร์คการ์ด หรือบางทีก็เรียกว่า "LAN การ์ด" เป็นการ์ดที่เสียบเข้ากับช่องบนเมนบอร์ดของคอมพิวเตอร์ ซึ่งเป็นจุดเชื่อมต่อระหว่างคอมพิวเตอร์และเครือข่าย และส่วนสุดท้ายคือฮาร์ดดิสก์ (Hard disk) ที่ทำหน้าที่เก็บข้อมูลต่างๆของระบบคอมพิวเตอร์

ดังนั้นในปริญญานิพนธ์เล่มนี้จึงแสดงให้เห็นถึงการนำไมโครคอนโทรลเลอร์ PIC16F877มาประยุกต์แทนที่ในส่วนของหน่วยประมวลผลกลาง ใช้ ไอซี(IC) RTL8019AS มาแทนในส่วนของเน็ตเวิร์คการ์ด และใช้อีอีพรอม (EEPROM) มาเก็บข้อมูลของแทนในส่วนของฮาร์ดดิสก์ รวมทั้งศึกษาถึงการส่งผ่านข้อมูลผ่านระบบเน็ตเวิร์ค (Network) ด้วย

1.2 ขอบเขตของโครงงาน

ในปริญญานิพนธ์ฉบับนี้ได้นำเสนอวิธีการส่งผ่านข้อมูลผ่านทางระบบอินเตอร์เน็ต โดยใช้ไมโครคอนโทรลเลอร์ทำหน้าที่เป็นเซิร์ฟเวอร์ในการส่งผ่านข้อมูลที่เป็นเว็บเพ็จไปยังเครื่องคอมพิวเตอร์ที่ทำหน้าที่เป็นเครื่องไคลเอ็นท์

1.3 วิธีการดำเนินการ

1. ทำการศึกษาค้นคว้าการรับส่งข้อมูลผ่านทางอินเตอร์เน็ต
2. ศึกษาการใช้งานไมโครคอนโทรลเลอร์ PIC16F877 เพื่อนำไปประยุกต์ใช้งานในการทำ หน้าที่เป็นเซิร์ฟเวอร์ที่ทำหน้าที่ประมวลผล รับส่งข้อมูล และเก็บเว็บเพ็จ
3. ศึกษาการใช้งานไอซี RTL8019AS ที่ทำหน้าที่ในการเชื่อมต่อไมโครคอนโทรลเลอร์กับระบบเครือข่ายอินเตอร์เน็ต
4. ศึกษาการใช้งานหน่วยความจำแบบอีอีพรอม (EEPROM) ซึ่งทำหน้าที่ในการเก็บเว็บเพ็จ
5. ศึกษาการใช้งานและโครงสร้างวงจรของบอร์ดทดลอง PICDEM.net (ได้รับการอนุเคราะห์บอร์ดทดลองจากบบริษัทไมโครชิป)
6. ศึกษาการเขียนโปรแกรมภาษาซี
7. ศึกษาการใช้งานโปรแกรม CCS Compiler เพื่อใช้ในการแปลงไฟล์ภาษาซีให้เป็นไฟล์.hex

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

8. ศึกษาการใช้งานโปรแกรม MPLAB เพื่อใช้ในการนำไฟล์.hex อัดลงไปที่บอร์ดทดลอง
9. ศึกษาการเขียนเว็บเพ็จด้วยภาษา html

1.4 ประโยชน์ที่คาดว่าจะได้รับ

1. ได้รับความรู้ทางด้านระบบเครือข่ายอินเตอร์เน็ต
2. ได้รับความรู้เกี่ยวกับการใช้งานของโปรแกรมต่างๆ
3. สามารถนำไปประยุกต์ใช้งานเพื่อลดต้นทุนของการติดต่อสื่อสารผ่านทางระบบอินเตอร์เน็ต

1.5 ส่วนประกอบของปริญญานิพนธ์

ปริญญานิพนธ์ฉบับนี้ได้แบ่งเนื้อหาออกเป็น 5 บทด้วยกันคือ

บทที่ 1 กล่าวถึงความสำคัญและที่มาของโครงงาน วัตถุประสงค์ของโครงงานขอบเขตของโครงงาน วิธีการดำเนินการ ประโยชน์ที่คาดว่าจะได้รับ และส่วนประกอบของปริญญานิพนธ์

บทที่ 2 กล่าวถึงทฤษฎีพื้นฐานที่ใช้ในโครงงาน เช่นทฤษฎีการเชื่อมต่อกันเป็นระบบอินเตอร์เน็ต อุปกรณ์ที่ใช้ในการเชื่อมต่อ ภาษาที่ใช้สื่อสารในระบบอินเตอร์เน็ต และการเขียนเว็บเพ็จ

บทที่ 3 กล่าวถึงชิ้นงานของโครงงาน การที่จะทำให้ไมโครคอนโทรลเลอร์สามารถติดต่อกับระบบอินเตอร์เน็ตได้ต้องใช้อุปกรณ์อะไรบ้าง และการติดต่อระหว่างไมโครคอนโทรลเลอร์กับอุปกรณ์ต่างๆต้องทำอย่างไร

บทที่ 4 กล่าวถึงการทดลอง และผลการทดลอง อธิบายการใช้งานบอร์ดทดลองทั้งการกำหนดค่าเริ่มต้นให้กับบอร์ดทดลองและการใช้งานบอร์ดทดลองในรูปแบบต่างๆ

บทที่ 5 กล่าวถึงบทสรุปของโครงงาน บทวิจารณ์ สิ่งที่ได้รับจากโครงงาน และข้อเสนอแนะสำหรับเป็นแนวทางในการพัฒนาต่อซึ่งจะเป็นประโยชน์ต่อผู้ที่ต้องการส่งผ่านข้อมูลปัจจุบันผ่านทางระบบอินเตอร์เน็ต

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 2

ทฤษฎีพื้นฐานที่ใช้ในโครงงาน

ในหัวข้อนี้จะกล่าวถึงทฤษฎีพื้นฐานต่างๆ ที่เกี่ยวข้องในการศึกษาการรับส่งข้อมูลผ่านทางระบบอินเตอร์เน็ต เช่นพื้นฐานของระบบเครือข่ายคอมพิวเตอร์ อุปกรณ์ต่างๆที่มีบทบาทสำคัญในระบบเครือข่าย ชุดโปรโตคอลที่ทำหน้าที่จัดเรียงข้อมูลที่ต้องการส่งผ่านเครือข่าย เว็บเซิร์ฟเวอร์(Web Server) คืออะไร ซึ่งเนื้อหาทั้งหมดนี้จำเป็นสำหรับการศึกษา เพื่อนำไปประยุกต์ใช้กับโครงงานได้อย่างมีประสิทธิภาพ

2.1 ความหมายของเครือข่ายคอมพิวเตอร์

เครือข่ายคอมพิวเตอร์(Computer network) คือระบบที่มีคอมพิวเตอร์อย่างน้อยสองเครื่องต่อกันโดยใช้สื่อกลาง และสามารถสื่อสารกันได้อย่างมีประสิทธิภาพ ซึ่งทำให้ผู้ใช้คอมพิวเตอร์สามารถแลกเปลี่ยนข้อมูลซึ่งกันและกันได้ และนอกจากนี้ยังสามารถใช้ข้อมูลและอุปกรณ์ ที่มีอยู่ในเครือข่ายร่วมกันได้ เช่น เครื่องพิมพ์ ซีดีรอม ฮาร์ดดิสก์ เป็นต้น เมื่อเชื่อมต่อกับเครือข่ายอื่นๆที่อยู่ห่างไกล เช่น ระบบอินเตอร์เน็ต ซึ่งเป็นเครือข่ายที่เชื่อมต่อคอมพิวเตอร์ทั่วโลก ก็จะทำให้สามารถแลกเปลี่ยนข้อมูลและติดต่อกับคนทั่วโลกได้หลายทาง เช่น เว็บ อีเมล์ เป็นต้น

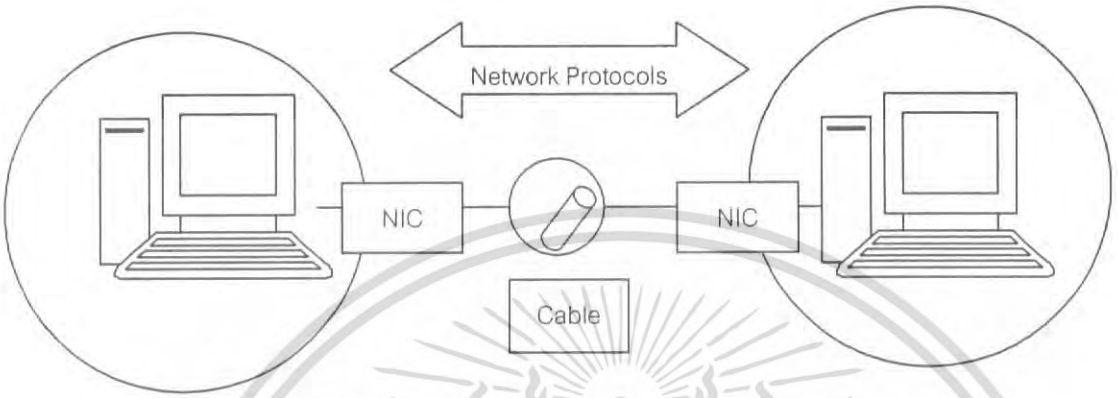
องค์ประกอบพื้นฐานที่สำคัญของระบบเครือข่าย จะมีองค์ประกอบดังรูปที่ 2.1

การที่คอมพิวเตอร์จะเชื่อมต่อกันเป็นเครือข่ายได้ เราจะต้องมีองค์ประกอบพื้นฐานดังต่อไปนี้

1. คอมพิวเตอร์ อย่างน้อย 2 เครื่อง
2. เน็ตเวิร์คการ์ด หรือ NIC (Network Interface Card) เป็นการ์ดที่เสียบเข้ากับช่องบนเมนบอร์ดของคอมพิวเตอร์ ซึ่งเป็นจุดเชื่อมต่อระหว่างคอมพิวเตอร์และเครือข่าย
3. สื่อกลางอุปกรณ์สำหรับการรับส่งข้อมูล เช่น สายสัญญาณ สายสัญญาณที่นิยมในเครือข่าย เช่น สายโคแอ็กเชียล สายคู่บิดเกลียว และสายใยแก้วนำแสง เป็นต้น ส่วนอุปกรณ์ในระบบเครือข่าย เช่น ฮับ สวิตช์ เราท์เตอร์ เกตเวย์ เป็นต้น
4. โปรโตคอล (Protocol) โปรโตคอลเป็นภาษาที่คอมพิวเตอร์ใช้สื่อสารกันผ่านเครือข่ายคอมพิวเตอร์จะสามารถสื่อสารกันได้นั้นจำเป็นที่จะต้องใช้ “ภาษา” หรือโปรโตคอลเดียวกัน เช่น OSI, TCP/IP, IPX/SPX เป็นต้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

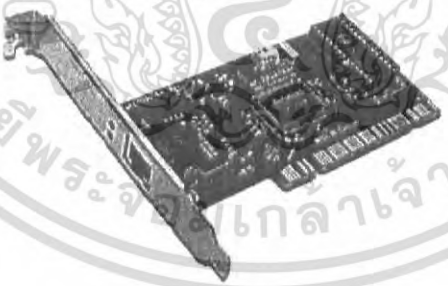
5. ระบบปฏิบัติการเครือข่าย หรือ NOS (Network Operating System) จะมีหน้าที่เป็นตัวคอยจัดการเกี่ยวกับการใช้งานเครือข่ายของผู้ใช้แต่ละคนระบบปฏิบัติการเครือข่ายที่เป็นที่นิยม เช่น Windows Server 2003, Novell NetWare และ Red Hat Linux เป็นต้น



รูปที่ 2.1 องค์ประกอบพื้นฐานของระบบเครือข่าย

2.2 เน็ตเวิร์คการ์ด

เน็ตเวิร์คการ์ดมีหน้าที่เชื่อมต่อระหว่างคอมพิวเตอร์และระบบเครือข่าย ส่วนใหญ่จะเรียกว่า NIC (Network Interface Card) หรือเรียกว่า "LAN การ์ด" (LAN Card) อุปกรณ์นี้จะทำหน้าที่แปลงข้อมูลเป็นสัญญาณที่สามารถส่งไปตามสายสัญญาณหรือสื่อแบบอื่นๆ ได้



รูปที่ 2.2 เน็ตเวิร์คการ์ด

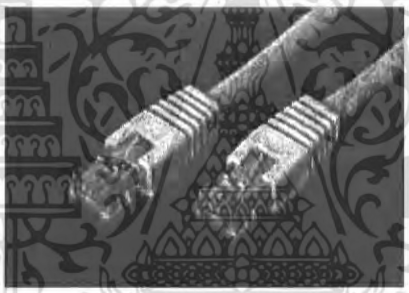
เน็ตเวิร์คการ์ดมีหลายประเภท แต่ละประเภทก็ออกแบบให้ใช้กับเครือข่ายประเภทต่างๆกัน เช่น อีเธอร์เน็ตการ์ด เป็นต้น การ์ดแต่ล่ะประเภทอาจใช้ได้กับสายสัญญาณบางชนิดเท่านั้น เช่น รูปที่ 2.2 เป็นอีเธอร์เน็ตการ์ดที่ใช้กับสายสัญญาณแบบคู่เกลียวบิด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

อัตราการส่งข้อมูลแบ่งได้เป็นหลายระดับ เช่น 10 Mbps, 100 Mbps หรือ 1,000 Mbps การ์ดบางประเภทอาจทำงานได้ที่ความเร็วเดียว บางการ์ดอาจทำงานได้หลายระดับความเร็ว การเลือกอัตราเร็วข้อมูลขึ้นอยู่กับฮับและสวิตซ์ที่คอมพิวเตอร์เครื่องนี้เชื่อมต่ออยู่

2.3 สายสัญญาณ

โครงงานนี้จะใช้สายสัญญาณแบบคู่บิดเกลียว ซึ่งเป็นสายสัญญาณที่นิยมใช้มากที่สุดในปัจจุบัน ตัวสายจะประกอบด้วยสายทองแดงที่ห่อหุ้มด้วยฉนวน 2 เส้นแล้วบิดเป็นเกลียว เพื่อลดการถูกรบกวนของสัญญาณ และสามารถแบ่งได้อีกตามคุณภาพของสาย เช่นสายโทรศัพท์ จะมีสายทองแดง 2 คู่ ซึ่งจะเรียกส่วนหัวที่ใช้ต่อสายนี้ว่า RJ-11 ส่วนสายที่นิยมใช้ในระบบเครือข่ายแบบอีเธอร์เน็ต คือสาย UTP (Unshielded Twisted Pair) ซึ่งจะคล้ายกับสายโทรศัพท์ แต่มีคุณภาพดีกว่า โดยมีสายทองแดงทั้งหมด 4 คู่ ส่วนหัวเชื่อมต่อจะเรียกว่า หัว RJ-45 ดังแสดงดังรูปที่ 2.3 ซึ่งโครงงานของเราจะเป็นการเชื่อมต่อกับระบบอีเธอร์เน็ตเราจึงเลือกใช้สายแบบ UTP ในการเชื่อมต่อสัญญาณ



รูปที่ 2.3 สายสัญญาณคู่บิดเกลียวแบบ UTP ที่มีหัว RJ-45 เชื่อมต่ออยู่

2.4 อุปกรณ์ในระบบเครือข่าย

2.4.1 เราท์เตอร์ (Router)

รูปที่ 2.4 แสดงถึงเราท์เตอร์ซึ่งเป็นอุปกรณ์ในเครือข่ายซึ่งทำหน้าที่อ่านที่อยู่ (Address) ของปลายทางที่ส่วนหัว (Header) ของแพ็กเก็จข้อมูล เพื่อกำหนดหรือเลือกเส้นทางที่จะส่งแพ็กเก็จนั้นต่อไป ในเราท์เตอร์จะมีข้อมูลเกี่ยวกับการจัดเส้นทางให้แพ็กเก็จที่เรียกว่า " เราท์ติ้งเทเบิ้ล (Routing Table) " หรือตารางการจัดเส้นทาง ข้อมูลในตารางนี้เป็นข้อมูลที่เราท์เตอร์เลือกเส้นทางที่ดีที่สุดไปยังปลายทาง ถ้าเส้นทางหลักเกิดขัดข้อง เราท์เตอร์ก็จะทำหน้าที่ในการจัดหาเส้นทางใหม่ในการส่งข้อมูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 2.4 เราท์เตอร์

2.4.2 ฮับ (Hub)

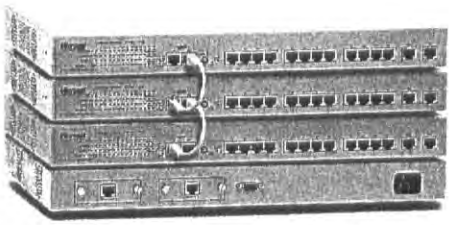
รูปที่ 2.5 แสดงฮับซึ่งเป็นอุปกรณ์ที่มีหน้าที่เชื่อมต่อกลุ่มของคอมพิวเตอร์เข้าด้วยกัน ฮับมีหน้าที่รับส่งเฟรมข้อมูลจากพอร์ตใดพอร์ตหนึ่งไปยังทุกๆพอร์ตที่เหลือ คอมพิวเตอร์ที่เชื่อมต่อเข้ากับฮับจะแชร์แบนด์วิธหรืออัตราข้อมูลของเครือข่าย ฉะนั้นยิ่งมีคอมพิวเตอร์มาเชื่อมต่อเข้ากับฮับมาก ยิ่งทำให้อัตราแบนด์วิธของคอมพิวเตอร์แต่ละเครื่องลดลง

รูปที่ 2.5 ฮับ

2.4.3 สวิตช์(Switch)

รูปที่ 2.6 แสดงสวิตช์ซึ่งจะมีความสามารถมากกว่าฮับ คือสามารถที่จะส่งข้อมูลจากพอร์ตใดพอร์ตหนึ่ง ๆไปยังพอร์ตปลายทางที่ต้องการเท่านั้น ทำให้คอมพิวเตอร์ที่เหลือสามารถส่งข้อมูลถึงกันและกันได้ในเวลาเดียวกัน คอมพิวเตอร์ทุกเครื่องจะมีแบนด์วิธเท่ากับแบนด์วิธของสวิตช์ ทำให้ไม่ขึ้นอยู่กับจำนวนเครื่องของคอมพิวเตอร์ที่มาต่อ ดังนั้นเครือข่ายแบบใหม่ๆนิยมใช้สวิตช์มากกว่าฮับ เพราะจะไม่มีปัญหาที่เกิดจากการชนกันของข้อมูลในระบบเครือข่าย

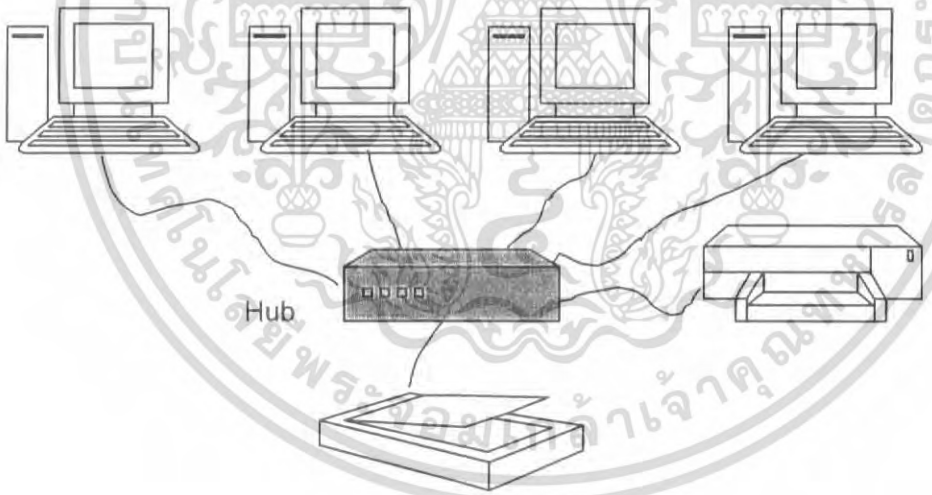
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.6 สวิทช์

2.5 เครือข่ายแบบท้องถิ่น(Local Area Network)

รูปที่ 2.7 แสดงแผนผังของเครือข่ายแบบท้องถิ่นซึ่งเป็นส่วนประกอบพื้นฐานของเครือข่ายคอมพิวเตอร์ทั่วไป ในทุกๆเครือข่ายต้องมีระบบ LAN เป็นองค์ประกอบ เครือข่ายแบบ LAN เองอาจเป็นได้ทั้งเครือข่ายแบบง่ายๆ เช่น มีคอมพิวเตอร์สองเครื่องเชื่อมต่อกันด้วยสายสัญญาณ ไปจนถึงเครือข่ายที่ซับซ้อน เช่น มีคอมพิวเตอร์เป็นร้อยๆเครื่องและมีอุปกรณ์เครือข่ายอื่นๆ อีกมาก แต่เครือข่ายแบบท้องถิ่นจะมีข้อเสีย คือเครือข่ายจะนั้นครอบคลุมพื้นที่ได้ค่อนข้างจำกัด



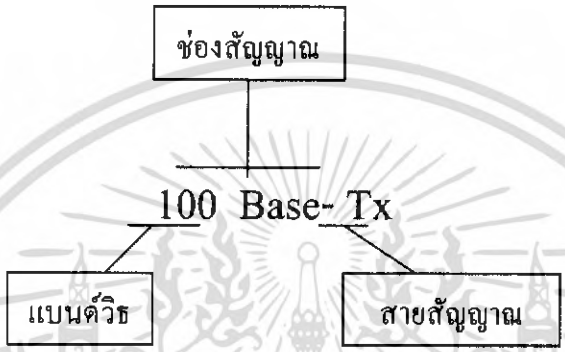
รูปที่ 2.7 Local Area Network (LAN)

เทคโนโลยี LAN มีหลายประเภท เช่น Ethernet, Token Ring เป็นต้น แต่ที่นิยมมากที่สุดในปัจจุบันคือ อีเธอร์เน็ต (Ethernet)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.5.1 อีเธอร์เน็ต

อีเธอร์เน็ต เป็นเทคโนโลยีเครือข่ายที่เป็นรากฐานที่สำคัญของเทคโนโลยีสารสนเทศทั้งหมด เนื่องจากเป็นเครือข่ายแบบท้องถิ่นที่นิยมมากที่สุด และได้มีการพัฒนาอย่างต่อเนื่อง จึงเป็นการยากที่จะพัฒนาเทคโนโลยีใหม่มาแทนที่ได้ เทคโนโลยีนี้ได้ถูกปรับปรุงภายใต้ความดูแลและรับผิดชอบของสถาบัน IEEE (Institute of Electrical Engineer) โดยสิ่งสำคัญอย่างหนึ่งในการเปลี่ยนแปลงปรับปรุงคือการเพิ่มความเร็วในการรับส่งข้อมูลหรือแบนด์วิธ และมีการกำหนดชื่อเรียกอีเธอร์เน็ตประเภทต่างๆไว้ดังนี้



รูปที่ 2.8 การเรียกชื่ออีเธอร์เน็ต

จากรูปที่ 2.8 ส่วนแรกจะบอกถึงแบนด์วิธของเครือข่าย ซึ่งเลขที่เป็นไปได้คือ 10, 100, 1000 และ 10G ซึ่งหมายถึง 10 Mbps, 100 Mbps, 1,000 Mbps และ 10 Gbps ตามลำดับ

ส่วนที่สองบอกถึงลักษณะของช่องสัญญาณ เช่น Base มาจาก Base band หมายถึงช่องสัญญาณที่ใช้รับส่งข้อมูลแบบเด็มช่อง หรือสัญญาณที่ใช้จะเป็นสัญญาณดิจิตอล หรือแสควร์เวฟ (Square Wave) ตรงข้ามคือ บรอดแบนด์ (Broadband) ซึ่งช่องสัญญาณที่ใช้จะกว้างกว่าแบบเบสแบนด์ และเป็นการส่งข้อมูลแบบแบ่งช่องสัญญาณออกเป็นสัญญาณย่อยแล้วส่งข้อมูลแต่ละช่องได้ การส่งข้อมูลแบบบรอดแบนด์ แต่เครื่องที่ใช้ส่งจะซับซ้อนและยากต่อการผลิตมากกว่าเครื่องที่ใช้กับเบสแบนด์ ดังนั้นมาตรฐานอีเธอร์เน็ตส่วนใหญ่จะเป็นแบบเบสแบนด์

ส่วนที่สามจะบอกถึงประเภทของสายสัญญาณที่ใช้ เช่น Tx เป็นตัวย่อของ Twisted Pair หมายความว่าเครือข่ายประเภทนั้นใช้สายสัญญาณแบบสายคู่บิดเกลียว ส่วน Fx เป็นตัวย่อของ Fiber Optic หรือก็คือใยแก้วนำแสงนั่นเอง จะมีกรณีพิเศษอยู่ 2 กรณีคือถ้าเป็นตัวเลข 5 จะเป็นอีเธอร์เน็ตที่ใช้สายโคแอกซ์แบบหนา ซึ่งเลขห้านี้มาจากการที่ว่าสายสัญญาณประเภทนี้สามารถรับส่งสัญญาณได้ไกลสุด 500 เมตร ส่วนเลข 2 หมายถึงเป็นอีเธอร์เน็ตที่ใช้สายโคแอกซ์แบบเล็กกว่าหรือแบบบาง ซึ่งรับส่งข้อมูลได้ไกลสุด 185 เมตร แต่เพื่อความสะดวกในการเรียกชื่อจึงปัดเป็น 200 เมตร แต่เป็นที่เข้าใจว่าส่งสูงสุดได้ 185 เมตร

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.6 ความหมายของ เซิร์ฟเวอร์ (Server)

เมื่อคอมพิวเตอร์หลายๆเครื่องถูกเชื่อมต่อเข้ากับระบบเครือข่ายแล้วจะมีคอมพิวเตอร์เครื่องหนึ่งทำหน้าที่เป็น "เซิร์ฟเวอร์" ซึ่งคอยให้บริการต่างๆ กับคอมพิวเตอร์เครื่องอื่นๆ เช่น เป็นศูนย์กลางในการจัดเก็บ ข้อมูล ไฟล์ หรือโปรกรมต่างๆ นอกจากนี้ยังทำหน้าที่จัดการเกี่ยวกับอุปกรณ์ที่เชื่อมต่ออยู่ในระบบเครือข่าย เช่น เครื่องพิมพ์และซีดีรอม (CD-ROM) เป็นต้น ซึ่งการใช้อุปกรณ์เหล่านี้ร่วมกันทำให้ประหยัดค่าใช้จ่ายในส่วนของอุปกรณ์เหล่านี้ได้ ส่วนคอมพิวเตอร์ที่ทำการเชื่อมต่อกับเซิร์ฟเวอร์เพื่อที่จะใช้บริการดังกล่าวเราจะเรียกว่า "ไคลเอนท์"

2.7 เครือข่ายแบบไคลเอนท์เซิร์ฟเวอร์(Client/Server Network)

เมื่อเครือข่ายมีขนาดใหญ่ขึ้นและผู้ใช้มากขึ้น การดูแลจัดการระบบก็จะมีความซับซ้อนมากขึ้น ดังนั้นเครือข่ายจำเป็นที่ต้องมีเซิร์ฟเวอร์ทำหน้าที่คอยจัดการเรื่องต่างๆ และให้บริการอื่นๆ เครื่องเซิร์ฟเวอร์ที่นำใช้จะต้องมีประสิทธิภาพสูงและสามารถให้บริการกับผู้ใช้หลายๆคนในเวลาเดียวกัน และขนาดเดียวกันก็ทำหน้าที่ในการดูแลรักษาความปลอดภัยในการเข้ามาใช้บริการทรัพยากรต่างๆของผู้ใช้ เครือข่ายแบบระบบไคลเอนท์เซิร์ฟเวอร์เป็นระบบที่ได้รับการยอมรับว่าเป็นมาตรฐานของการสร้างระบบเครือข่ายในปัจจุบัน

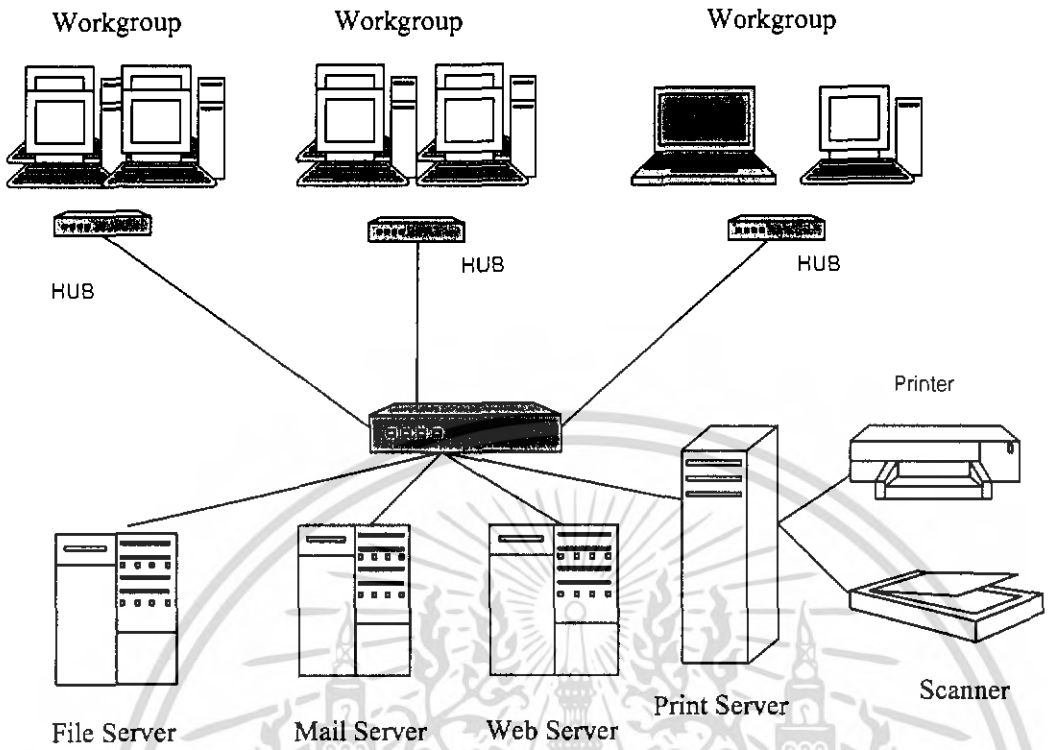
การติดตั้งระบบแบบไคลเอนท์เซิร์ฟเวอร์นั้น การกำหนดค่าต่างๆ และการดูแลจัดการเครือข่ายค่อนข้างจะยุ่งยาก แต่ก็มีข้อดีหลายอย่าง เช่น

ขณะที่เซิร์ฟเวอร์ทำหน้าที่ให้บริการด้านต่างๆแก่ผู้ใช้ ในเวลาเดียวกันก็ควบคุมและรักษาความปลอดภัยของข้อมูลด้วย เครือข่ายแบบไคลเอนท์เซิร์ฟเวอร์จะเป็นศูนย์รวมการดูแลและจัดการเครือข่าย เนื่องจากว่าทรัพยากรเหล่านี้ถูกเก็บรวบรวมไว้ที่เครื่องเซิร์ฟเวอร์จึงทำให้ง่ายต่อการค้นหาและจัดการทรัพยากรที่เก็บไว้กระจัดกระจายตามเครื่องไคลเอนท์ต่างๆ

ความปลอดภัยของข้อมูลอาจเป็นหนึ่งในจุดประสงค์หลักที่ทำให้ต้องเลือกเครือข่ายแบบไคลเอนท์เซิร์ฟเวอร์ เพราะในสภาวะแวดล้อมอย่างนี้ผู้ดูแลระบบสามารถกำหนดนโยบายรักษาความปลอดภัยและบังคับใช้กับผู้ใช้ทุกคนในเครือข่ายได้ ทำให้การรักษาความปลอดภัยง่ายขึ้น

เนื่องจากเซิร์ฟเวอร์มีโปรแกรมอเนกประสงค์ที่ใช้ในการจัดการเครือข่ายหลายอย่าง จึงทำให้เครือข่ายแบบไคลเอนท์เซิร์ฟเวอร์สามารถรองรับผู้ใช้ได้เป็นพันๆคน ในรูปที่ 2.9 จะเป็นการแสดงรูปแบบเครือข่ายแบบไคลเอนท์เซิร์ฟเวอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.9 เครือข่ายแบบไคลเอนท์เซิร์ฟเวอร์

2.7.1 ชนิดของเซิร์ฟเวอร์ที่ให้บริการในรูปแบบต่างๆ

เมื่อเครือข่ายมีการขยายตัวหรือจำนวนผู้ใช้มีจำนวนมากขึ้นนั้น ระยะห่างระหว่างคอมพิวเตอร์และปริมาณข้อมูลที่ถูกส่งผ่านเครือข่ายมีปริมาณเพิ่มมากขึ้น อาจจะต้องมีจำนวนเซิร์ฟเวอร์เพิ่มมากขึ้นด้วย การกระจายหน้าที่ของเซิร์ฟเวอร์ไปหลายๆเครื่อง เพื่อให้เซิร์ฟเวอร์แต่ละเครื่อง ทำหน้าที่เฉพาะอย่าง ซึ่งจะมีประสิทธิภาพมากกว่าการใช้เซิร์ฟเวอร์เพียงตัวเดียว แต่ให้บริการหลายๆอย่าง

เซิร์ฟเวอร์ต้องสามารถที่จะทำหน้าที่ที่ซับซ้อนและมีการเปลี่ยนแปลงอยู่เสมอ เซิร์ฟเวอร์ของเครือข่ายขนาดใหญ่ถูกกำหนดให้ทำหน้าที่เฉพาะอย่างเพื่อรับรองความต้องการของผู้ใช้ที่เพิ่มขึ้น ต่อไปนี้เป็นตัวอย่างเซิร์ฟเวอร์ชนิดต่างๆ ที่ส่วนใหญ่มีในเครือข่ายขนาดใหญ่ๆทั่วไป

ไฟล์และพรินต์เซิร์ฟเวอร์(File and Print Server) ไฟล์เซิร์ฟเวอร์จะให้บริการเกี่ยวกับพื้นที่เก็บไฟล์ต่างๆ ซึ่งเซิร์ฟเวอร์ประเภทนี้จะมีฮาร์ดดิสก์ที่สามารถบรรจุข้อมูลได้อย่างเพียงพอกับความต้องการของผู้ใช้ ส่วนพรินต์เซิร์ฟเวอร์ (Printer Server) ทำหน้าที่เกี่ยวกับการใช้เครื่องพิมพ์ที่พ่วงต่อเข้ากับเครือข่าย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

แอพพลิเคชั่นเซิร์ฟเวอร์(Application Server) ทำหน้าที่เกี่ยวกับโปรแกรมและข้อมูลที่เกี่ยวกับโปรแกรมนั้นๆ ตัวอย่างเช่น เซิร์ฟเวอร์ฐานข้อมูล ซึ่งจะทำหน้าที่ในการจัดเก็บข้อมูลขนาดใหญ่ให้ง่ายต่อการเรียกดูข้อมูลของผู้ใช้ เซิร์ฟเวอร์ชนิดนี้จะแตกต่างจากไฟล์เซิร์ฟเวอร์ตรงที่ข้อมูลจะถูกเก็บไว้ที่ทางด้านเซิร์ฟเวอร์ตลอดเวลา ในขณะที่เป็นไฟล์เซิร์ฟเวอร์แล้วไคลเอนท์ต้องดาวโหลดไฟล์ไปทำการเปลี่ยนแปลงที่ทางฝั่งไคลเอนท์ แล้วค่อยนำกลับมาเก็บที่ทางฝั่งเซิร์ฟเวอร์อีกทีหนึ่ง

ไคลเอนท์ของแอพพลิเคชั่นจะรันโปรแกรมบนไคลเอนท์ แต่จะดึงข้อมูลจากทางฝั่งเซิร์ฟเวอร์ เช่นการค้นหาข้อมูลของลูกค้าจากเซิร์ฟเวอร์ฐานข้อมูล เฉพาะข้อมูลที่ผู้ใช้ต้องการเท่านั้นที่จะถูกส่งมาแทนที่จะเป็นข้อมูลทั้งฐานข้อมูล

อินเตอร์เน็ตเซิร์ฟเวอร์(Internet Server) ปัจจุบันอินเตอร์เน็ตมีผลกระทบต่อระบบเครือข่ายในระบบปัจจุบันมาก อินเตอร์เน็ตเป็นเครือข่ายที่มีขนาดใหญ่และมีผู้ใช้งานมากที่สุดในโลกเทคโนโลยีที่ทำให้อินเตอร์เน็ตเป็นที่นิยมคือ เว็บและอีเมล์ เพราะทั้งสองแอพพลิเคชันทำให้ผู้ใช้สามารถแลกเปลี่ยนข้อมูลและสื่อสารกันได้ง่ายและรวดเร็ว

เว็บเซิร์ฟเวอร์คือเซิร์ฟเวอร์ที่ให้บริการข้อมูลในรูปแบบ HTML (Hyper Text Markup Language) ซึ่งไฟล์เหล่านี้เปิดอ่านโดยใช้เว็บบราวเซอร์ (Web Browser) อย่างเช่น Internet Explorer เป็นต้น ปัจจุบันทุกองค์กรจะมีเว็บเซิร์ฟเวอร์เพื่อให้บริการข้อมูลต่อพนักงาน และผู้ใช้ทั่วๆไป

เมลเซิร์ฟเวอร์(Mail Server) คือเซิร์ฟเวอร์ที่ให้บริการในการรับส่ง จัดเก็บ และจัดการเกี่ยวกับอีเมล์ของผู้ใช้ ซึ่งอาจเป็นอีเมล์ที่ใช้ได้เฉพาะภายในองค์กร หรือเชื่อมต่อเข้ากับอินเตอร์เน็ต ขึ้นอยู่กับนโยบายใช้งานของเครือข่าย

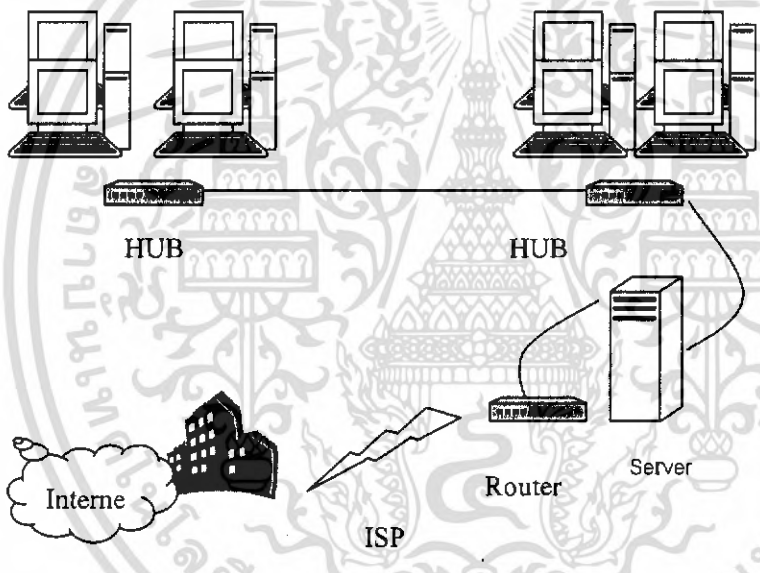
2.8 อินเตอร์เน็ต(Internet)

อินเตอร์เน็ตเป็นเครือข่ายที่ครอบคลุมไปทั่วโลก มีคอมพิวเตอร์นับล้านๆ เครื่องเชื่อมต่อเข้ากับระบบและยังขยายตัวขึ้นเรื่อยๆในทุกๆปี อินเตอร์เน็ตมีผู้ใช้หลายร้อยล้านคนทั่วโลก และผู้ใช้เหล่านี้สามารถแลกเปลี่ยนข้อมูลข่าวสารกันได้อย่างอิสระ โดยระยะทางและเวลาไม่เป็นอุปสรรค นอกจากนี้ผู้ใช้ยังสามารถเข้าดูข้อมูลต่างๆที่มีการเผยแผ่ในอินเตอร์เน็ตได้ อินเตอร์เน็ตได้เชื่อมแหล่งข้อมูลเหล่านี้เข้าไว้ด้วยกัน ไม่ว่าจะเป็นองค์กรธุรกิจหลายองค์กรได้ใช้อินเตอร์เน็ตเป็นตัวช่วยในการทำการค้า เช่น การติดต่อซื้อขายผ่านอินเตอร์เน็ตหรืออีคอมเมิร์ซ(E-Commerce) ซึ่งเป็นอีกช่องทางหนึ่งสำหรับการทำธุรกิจที่กำลังเป็นที่นิยม เนื่องจากมีต้นทุนที่ถูกกว่าและมีฐานลูกค้าที่ใหญ่มาก ส่วนข้อเสียของอินเตอร์เน็ตคือ ความปลอดภัยของข้อมูล เนื่องจากทุกคนสามารถเข้าถึงข้อมูลทุกอย่างที่แลกเปลี่ยนผ่านอินเตอร์เน็ตได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

อินเตอร์เน็ตมีการเรียกใช้โปรโตคอลที่เรียกว่า "TCP/IP" (Transport Connection Protocol/Internet Protocol) ในการสื่อสารข้อมูลผ่านเครือข่าย ซึ่งโปรโตคอลนี้เป็นผลจากโครงงานหนึ่งของกระทรวงกลาโหมสหรัฐอเมริกา จุดประสงค์เพื่อเชื่อมต่อคอมพิวเตอร์ที่อยู่ห่างไกลกัน และภายหลังจึงกำหนดให้เป็นโปรโตคอลมาตรฐานในเครือข่ายอินเตอร์เน็ต

ปัจจุบันนี้อินเตอร์เน็ตได้กลายเป็นเครือข่ายสาธารณะ ซึ่งไม่มีองค์กรใดองค์กรหนึ่งเป็นเจ้าของอย่างแท้จริง ในการเชื่อมต่อเข้ากับระบบอินเตอร์เน็ตต้องเชื่อมต่อผ่านองค์องค์ที่เรียกว่า "ISP (Internet Service Provider)" ซึ่งจะทำหน้าที่ให้บริการในการเชื่อมต่อเข้ากับระบบอินเตอร์เน็ต เนื่องจากอินเตอร์เน็ตเป็นเครือข่ายสาธารณะจึงไม่มีหลักประกันความปลอดภัยของข้อมูลที่ส่งผ่านในระบบ นั่นคือ ข้อมูลทุกอย่างที่ส่งผ่านเครือข่าย ทุกคนสามารถที่จะดูได้ นอกเสียจากจะมีการเข้ารหัสลับซึ่งผู้ใช้ต้องทำเอง



รูปที่ 2.10 เครือข่ายที่ทำการเชื่อต่อกับระบบอินเตอร์เน็ต

2.9 โปรโตคอล คืออะไร ?

โปรโตคอล (Protocol) เป็นมาตรฐานในการสื่อสารข้อมูลของคอมพิวเตอร์ หรืออาจเรียกได้ว่าโปรโตคอลเป็นภาษาที่คอมพิวเตอร์ใช้สื่อสารกัน ดังนั้นคอมพิวเตอร์ที่ต้องการสื่อสารกันจำเป็นต้องใช้ "ภาษา" เดียวกันหรือโปรโตคอลเดียวกันเพราะไม่เช่นนั้นก็จะสื่อสารกันไม่ได้ เช่น คอมพิวเตอร์ที่เชื่อมต่อเข้ากับระบบอินเตอร์จะใช้ "ภาษา" หรือโปรโตคอล TCP/IP ส่วนคอมพิวเตอร์ที่ใช้ระบบปฏิบัติการเครือข่ายเน็ตแวร์ (Netware) ก็จะใช้ "ภาษา" หรือโปรโตคอล IPX/SPX ในการสื่อสารกัน เป็นต้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.10 โปรโตคอล TCP/IP

โปรโตคอล TCP/IP (Transmission Control Protocol/Internet Protocol) เป็นโปรโตคอลที่นิยมใช้ในเครือข่ายมากที่สุดในปัจจุบัน เนื่องจากหลายบริษัทที่ผลิตอุปกรณ์หรือซอร์ฟแวร์ของเครือข่ายได้นำมาใช้เป็นมาตรฐานในการผลิต และอีกอย่าง TCP/IP เป็นโปรโตคอลพื้นฐานของเครือข่ายอินเตอร์เน็ต ซึ่งเป็นเครือข่ายที่ใหญ่ที่สุดในโลก

โปรโตคอล TCP/IP มีการจัดแบ่งกลไกการทำงานออกเป็นชั้นๆ หรือ layer ซึ่งในแต่ล่ะกลไกของโปรโตคอล TCP/IP ก็จะมีโปรโตคอลอื่นๆในชุดของ TCP/IP ร่วมทำงานอยู่ด้วย จึงทำให้เป็นที่มาของชื่อเรียก TCP/IP Stack เนื่องจากมีโปรโตคอลซ้อนทับกันอยู่เพื่อช่วยกันทำงานดังตารางที่ 2.1

ตารางที่ 2.1 แสดงความสัมพันธ์ระหว่างโปรโตคอลต่างๆของ TCP/IP

<table>
<tr><td colspan="3">TCP/IP</td></tr>
<tr><td>Application</td><td colspan="2">FTP , Telnet , HTTP , SMTP,
SNMP , DNS , etc.</td></tr>
<tr><td>Host-to-Host</td><td>TCP</td><td>UDP</td></tr>
<tr><td rowspan="2">Internet</td><td>ICMP , IGMP</td><td>ARP , RARP</td></tr>
<tr><td colspan="2">IP</td></tr>
<tr><td>Network Access</td><td colspan="2">Not Specified</td></tr>
</table>

ซึ่งในโครงงานที่จัดทำนี้จะเลือกใช้โปรโตคอลต่างๆ เพื่อทำการติดต่อกับระบบอินเตอร์เน็ตดังรูปตารางที่ 2.2

ตารางที่ 2.2 โปรโตคอลต่างๆของ TCP/IP ที่นำมาใช้สำหรับในการรับส่งเว็บเพ็จทางอินเตอร์เน็ต

<table>
<tr><td>Application</td><td>HTTP</td></tr>
<tr><td>Host-to-Host</td><td>TCP</td></tr>
<tr><td rowspan="2">Internet</td><td>ICMP,ARP</td></tr>
<tr><td>IP</td></tr>
<tr><td>Network Access</td><td>Ethernet</td></tr>
</table>

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.11 โปรโตคอลที่ทำงานในชั้นอินเตอร์เน็ต (Internet Layer)

โปรโตคอลที่ทำงานในเลเยอร์อินเตอร์เน็ตคือ IP, ARP และ ICMP ซึ่งจะอธิบายหลักการทำงานได้ดังต่อไปนี้

2.11.1 Internet Protocol (IP)

โปรโตคอล IP ทำหน้าที่เสมือนกับที่ทำการไปรษณีย์ กล่าวคือทำหน้าที่ในการจัดส่งแพ็กเก็ตหรือบางทีเรียกว่า "ดาต้าแกรม" คือหน่วยของข้อมูลที่รับมาจากโปรโตคอลที่อยู่เลเยอร์ที่สูงกว่า เช่น TCP ถ้าโฮสต์ปลายทางอยู่คนละเครือข่ายกับโฮสต์ที่ส่งข้อมูล IP จะทำหน้าที่รับผิดชอบในการจัดเส้นทาง (Routing) ให้แพ็กเก็ตส่งไปยังเครือข่ายที่โฮสต์นั่นอยู่ ซึ่งในการจัดส่งแพ็กเก็ตข้ามเครือข่ายนั้น โปรโตคอล IP จะใช้เราท์เตอร์ในการเชื่อมต่อเครือข่ายเหล่านั้น โดยทั่วไปแล้วอุปกรณ์รับส่งข้อมูลระหว่างเครือข่ายจะเรียกว่าเราท์เตอร์ แต่บางทีอุปกรณ์ตัวนี้ก็จะเรียกว่า "เกตเวย์ (Gateway)" ซึ่งทำหน้าที่เสมือนประตูไปยังเครือข่ายอื่นๆ

โปรโตคอล IP เป็นโปรโตคอลที่ให้บริการแบบคอนเน็กชันเลส (Connectionless) มีความเชื่อถือได้น้อย เนื่องจากไม่มีการสร้างการเชื่อมต่อก่อนที่จะทำการรับส่งข้อมูล กล่าวคือ ในการส่งข้อมูลแต่ละครั้งโฮสต์ส่งจะไม่ทำการติดต่อโฮสต์ปลายทางเพื่อตกลงเกี่ยวกับการรับส่งข้อมูลก่อน แต่โฮสต์ที่ต้องการส่งข้อมูลจะส่งออกไปทันที โดยที่คาดหวังว่าโฮสต์ปลายทางจะได้รับแพ็กเก็ตนั้นในที่สุด ดังนั้นความเชื่อถือในการส่งข้อมูลจึงมีน้อยเพราะแพ็กเก็ตอาจสูญหายระหว่างทางก็ได้ หรืออาจเดินทางมาถึงปลายทางไม่เป็นลำดับ หรือมีการส่งแพ็กเก็ตซ้ำกัน หรือส่งล่าช้า การแก้ปัญหานี้จะปล่อยให้โปรโตคอลที่อยู่สูงกว่ารับผิดชอบ

ตารางที่ 2.3 แสดงถึงฟอร์แมตของแพ็กเก็ต IP ซึ่งประกอบด้วยหลายฟิลด์ดังนี้

- **Version (4 บิต):** ข้อมูล 4 บิตแรกจะเป็นข้อมูลที่บอกถึงเวอร์ชันของโปรโตคอล IP ที่ใช้ ซึ่งในปัจจุบันจะใช้เวอร์ชัน 4
- **Internet Header Length หรือ IHL (4บิต):** เป็นตัวเลขบอกความยาวของข้อมูลในส่วนหัว (Header)
- **Type of Service (8บิต):** ในแต่ละบิตของส่วนข้อมูลนี้จะเป็นธงหรือแฟลก (Flag) ที่แสดงถึงลำดับความสำคัญ (precedence) ความล่าช้า (Delay) อัตราส่งผ่าน (Throughput) และค่ากำหนดความเชื่อถือได้ของแพ็กเก็ตนี้อยู่
- **Total Length (16บิต):** ข้อมูลนี้จะบอกถึงความยาวของแพ็กเก็ตทั้งหมดซึ่งมีหน่วยเป็นไบต์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

o **Identifier (16 บิต):** ถ้าดาต้าแกรมประกอบด้วยหลายแพ็กเก็ต หมายเลขนี้จะถูกกำหนดให้กับแต่ละแพ็กเก็ตย่อย ซึ่งจะมีหมายเลขไม่ซ้ำกัน

o **Flag (3 บิต):** เป็นฟิลด์ที่ใช้ในการจัดการเกี่ยวกับการแบ่งข้อมูลเป็นแพ็กเก็ตย่อย

o **Flagment Offset (13 บิต):** เป็นค่าที่บอกจุดเริ่มต้นในส่วนของข้อมูลย่อย (Fragmented Content) ซึ่งเป็นตัวเลขที่บอกว่าแพ็กเก็ตย่อยนี้อยู่ห่างจากจุดเริ่มต้นของดาต้าแกรมทั้งหมดเท่าใด โดยจำนวนนี้มีหน่วยวัดเป็น 64 บิต

o **Time to live หรือ TTL (8 บิต):** แพ็กเก็ตจะไหลเวียนอยู่ในเครือข่ายได้ช่วงเวลาหนึ่ง จะกำหนดค่าโดยการคิดจำนวนครั้งที่ผ่านเราท์เตอร์ ทุกครั้งที่ผ่านค่า TTL จะลดลงทีละหนึ่ง เมื่อเป็นศูนย์แพ็กเก็ตนี้ก็จะถูกละทิ้งไป

o **Protocol (8บิต):** เป็นข้อมูลที่บอกโปรโตคอลของชั้นที่เหนือกว่า เช่น TCP, UDP เป็นต้น

o **Header Checksum (16บิต):** เป็นข้อมูลส่วนที่ใช้ในการตรวจสอบข้อผิดพลาดในส่วนเฮดเดอร์ของแพ็กเก็ต ซึ่งเมื่อผ่านอุปกรณ์เครือข่ายแต่ล่ะครั้งก็จะทำการเช็คแต่ล่ะครั้งไป

o **Source IP Address :(32 บิต):** หมายเลข IP ของเครื่องที่ส่งข้อมูล

o **Destination IP Address :(32 บิต):** หมายเลย IP ของเครื่องปลายทาง

o **Padding:** เป็นเลข 0 ที่เพิ่มให้กับส่วนหัวของแพ็กเก็ตเพื่อให้ส่วนหัวมีความยาวที่หารด้วย 32 บิตลงตัว

o **Data:** ข้อมูลของโปรโตคอลที่อยู่สูงกว่าความยาวไม่คงที่

ตารางที่ 2.3 แสดงฟอร์แมตของแพ็กเก็ต IP

<table>
<tr><td>Version</td><td>IHL</td><td>Type-of-Service</td><td colspan="2">Total Length</td></tr>
<tr><td colspan="3">Identification</td><td>Flag</td><td>Flagment Offset</td></tr>
<tr><td colspan="2">Time –to-Live</td><td>Protocol</td><td colspan="2">Header Checksum</td></tr>
<tr><td colspan="5">Source Address</td></tr>
<tr><td colspan="5">Destination Address</td></tr>
<tr><td colspan="5">Options+Padding</td></tr>
<tr><td colspan="5">Data</td></tr>
</table>

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.11.2 Address Resolution Protocol (ARP)

การที่คอมพิวเตอร์ที่อยู่ในเครือข่ายเดียวกันต้องการจะสื่อสารกันจำเป็นที่ต้องทราบหมายเลขเน็ตเวิร์คการ์ด หรือแม็กแอดเดรส(MAC Address) ของกันและกัน แพ็กเก็ตระดับไอพีจะถูกห่อหุ้มด้วยเฟรมในระดับของดาต้าลิ้งค์ ซึ่งแม็กแอดเดรสของเครื่องส่งและเครื่องรับจะต้องถูกใส่ไปด้วย ปัญหาก็คือเครื่องส่งอาจไม่ทราบหมายเลขแม็กแอดเดรสของเครื่องรับ

โปรโตคอล ARP จะทำหน้าที่ค้นหาหมายเลขแม็กแอดเดรสของเครื่องที่มีหมายเลขไอพีที่ต้องการ หลักการทำงานคือ โฮสต์ที่ต้องการทราบหมายเลขแม็กแอดเดรสของเครื่องที่มีเครื่องหมายเลขไอพีนั้น จะทำการบรอดคาสต์แพ็กเก็ตไปยังคอมพิวเตอร์ทุกเครื่องที่อยู่ในเครือข่ายเดียวกัน ถ้ามีเครื่องที่มีหมายเลขไอพีดังกล่าว เครื่องนั้นก็จะตอบกลับมาพร้อมหมายเลขแม็กแอดเดรสของ หลังจากนั้นเครื่องส่งก็จะสามารถสื่อสารกับเครื่องปลายทางได้โดยตรง โดยใช้แม็กแอดเดรสที่ส่งมาพร้อมกับแพ็กเก็ตตอบกลับ

2.11.3 Internet Control Message Protocol (ICMP)

โปรโตคอล ICMP ทำหน้าที่รายงานข้อผิดพลาดต่างๆ ที่เกิดขึ้นในระหว่างส่งแพ็กเก็ตในเครือข่าย ICMP ใช้การส่งแบบคอนเน็กชันเลส ซึ่งหมายถึงการส่งข้อมูลที่ฝ่ายรับฝ่ายส่งไม่ได้ประสานกันก่อน กล่าวคือฝ่ายรับจะไม่ทราบว่าจะมีแพ็กเก็ตส่งมาหาตัวเองดังนั้นโอกาสที่แพ็กเก็ตจะไม่ถึงปลายทางจึงเป็นไปได้สูง

โปรโตคอล ICMP จะทำหน้าที่รายงานข้อผิดพลาดบางอย่างที่เกิดขึ้นในระหว่างการส่งข้อมูล ฟังก์ชันที่สำคัญมีดังนี้

o ประกาศข้อผิดพลาดของเครือข่าย เช่นโฮสต์หรือบางส่วนของเครือข่ายติดต่อไม่ได้ เนื่องจากสาเหตุบางอย่าง โปรโตคอล TCP ที่จะส่งไปยังโฮสต์นั้นก็จะถูกรายงานโดย ICMP เช่นกัน

o ประกาศความคับคั่งของเครือข่าย เช่น เราท์เตอร์ได้รับแพ็กเก็ตมาเกินที่จะรับได้ ก็จะแจ้ง ICMP Source Quench ให้สถานีส่งทราบ

o ช่วยในการค้นหาข้อผิดพลาด โดย ICMP จะรองรับฟังก์ชัน Echo ซึ่งเป็นการส่งแพ็กเก็ตไปกลับระหว่างสองโฮสต์ใดๆ Ping เป็นเครื่องมือที่ใช้ฟังก์ชันนี้ โดย Ping จะส่งแพ็กเก็ตหลายๆแพ็กเก็ตไปยังโฮสต์ปลายทางและวันเวลาที่แพ็กเก็ตเดินทางไปกลับ และหาค่าเฉลี่ยและอัตราสูญหายของแพ็กเก็ต

o ประกาศการหมดเวลา กล่าวคือเมื่อแพ็กเก็ตไหลเวียนอยู่ในเครือข่ายนานเกินไป ค่า TTL จะกลายเป็นศูนย์ในที่สุด เราท์เตอร์ก็จะทิ้งแพ็กเก็ตนี้และรายงานความผิดผ่าน ICMP

ฟังก์ชันต่างๆ ของโปรโตคอล ICMP มีประโยชน์มากในการใช้วิเคราะห์และค้นหาจุดเสียของระบบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

62904

2.11.4 กลไกการทำงานของโปรโตคอล IP

ในการส่งผ่านข้อมูลไปยังเครือข่ายอินเตอร์เน็ตนั้น โปรโตคอล IP จะทำหน้าที่พิจารณาว่าปลายทางในการส่งข้อมูลนั้นจะเป็นในเครือข่ายของตนเองหรือว่าจะต้องส่งข้อมูลข้ามเครือข่ายไปอีก

โดยการพิจารณานี้โปรโตคอล IP จะตรวจสอบค่า "หมายเลขไอพี (IPAddress)" ปลายทางว่าส่วนที่เป็นค่าหมายเลขเครือข่าย (Network Address) จะเหมือนกับค่าหมายเลขเครือข่ายของ IP Address ต้นทางหรือไม่ ถ้าค่าตรงกันแสดงว่าการส่งข้อมูลอยู่ในเครือข่ายเดียวกัน แต่ถ้าต่างกัน แสดงว่าต้องส่งข้อมูลไปยังปลายทางที่อยู่คนละเครือข่ายกัน

การส่งข้อมูล ภายในเครือข่ายเดียวกัน มีกลไกดังนี้

1. โปรโตคอล IP จะเรียกใช้บิการโปรโตคอล ARP เพื่อแปลงหมายเลขไอพีปลายทางให้เป็นค่าหมายเลขฮาร์ดแวร์ เช่น Mac address
2. เมื่อโปรโตคอล IP ได้รับค่าหมายเลขฮาร์ดแวร์แล้ว ก็จะส่งข้อมูลไปยังฮาร์ดแวร์ที่ระบุไว้

การส่งข้อมูลข้ามเครือข่าย มีกลไกดังนี้

1. โปรโตคอล IP ตรวจสอบพบว่าหมายเลขไอพี ปลายทางอยู่คนล่ะเครือข่ายกัน โดยโปรโตคอล IP จะอ่านค่าหมายเลขไอพีของเราท์เตอร์เพื่อเตรียมส่งข้อมูลไปที่เราท์เตอร์แทน ซึ่งในที่นี้จะเป็นการกำหนดเป็น default Router
2. โปรโตคอล IP จะเรียกใช้บริการโปรโตคอล ARP เพื่อแปลงค่าหมายเลขไอพีของเราท์เตอร์ให้เป็นค่าหมายเลขฮาร์ดแวร์
3. โปรโตคอล IP ส่งข้อมูลไปยังเราท์เตอร์ที่กำหนดไว้จากนั้นเราท์เตอร์จะส่งข้อมูลข้ามเครือข่ายไปตามขั้นตอน

โปรโตคอล IP จะรู้ได้อย่างไรว่าเครือข่ายดังกล่าวมีการเชื่อมต่อเราท์เตอร์อยู่และมีค่าหมายเลขไอพีอะไร ผู้ใช้จะต้องกำหนดค่าที่เรียกว่า default Router หรือ default Gateway เสียก่อนว่ามีค่าหมายเลขไอพีอะไร โดยสามารถสอบถามได้จากผู้ดูแลระบบ

2.11.4.1 หมายเลขไอพี (IP Address) หรือ URL (Uniform Resource Locator)

หมายเลขไอพีจะเป็นเลขที่อยู่เฉพาะของโฮสต์ที่อยู่ในเครือข่าย รวมถึงคอมพิวเตอร์และเราท์เตอร์ หมายเลขไอพีจะมีขนาด32 บิตอยู่ในรูปเลขฐานสอง เนื่องจากเลขฐานสองที่เป็นตัวเลขที่มีขนาด 32 บิตเป็นตัวเลขที่ยาวและยากต่อการจดจำ ดังนั้นเพื่อเป็นการง่าย หมายเลขไอพีจึงนิยมเขียนให้อยู่ในรูปเลขฐานสิบ ซึ่งจะทำได้โดยจัดกลุ่มเลขฐานสองเป็น 4 กลุ่มๆล่ะ 8 บิตหลังจากนั้นแปลงเลขฐานสองแต่ล่ะกลุ่มให้อยู่ในเลขฐานสิบ จากนั้นเมื่อแปลงเสร็จให้นำเอาเอาเลขทั้งสี่มารวมกันโดยใช้จุดเป็นตัวเชื่อม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตัวอย่างการแปลงเลขฐานสองเป็นฐานสิบ

10101100 00010100 000000001 00011000

172 . 20 . 1 . 24

จากตัวอย่างจะเห็นได้ว่าหมายเลขไอพีที่อยู่ในดูของเลขฐานสิบจะง่ายต่อการจดจำมากกว่า เนื่องจากหมายเลขไอพีที่เป็นเลขฐานสิบนี้เป็นการแปลงมาจากเลขฐานสอง 8 บิต ดังนั้นเลขฐานสิบแต่ละตัวจะมีค่าอยู่ระหว่าง 0 ถึง 255 เพราะฉะนั้นหมายเลขไอพีที่ถูกต้องจะอยู่ระหว่าง 0.0.0.0 ถึง 255.255.255.255

แต่อินเตอร์เน็ตมีการเจริญเติบโตที่เร็วมากทำให้การจดจำหมายเลขไอพีของเครื่องหลายๆ เป็นไปได้ยาก จึงมีผู้คิดค้นพัฒนาระบบ ที่ชื่อว่า **Domain Name System** หรือ **DNS** ขึ้น

2.11.4.2 Domain Name System (DNS)

DNS จะเป็นระบบที่ทำให้หมายเลยไอพีจดจำง่ายขึ้นด้วยการตั้งชื่อให้มัน กล่าวคือ DNS จะทำหน้าที่เหมือนสมุดโทรศัพท์ เมื่อมีคนต้องการที่จะโทรศัพท์หาใคร ก็จะทำการเปิดดูสมุดโทรศัพท์เพื่อทำการค้นหาหมายเลขโทรศัพท์ของคนที่จะทำการติดต่อ คอมพิวเตอร์ก็เช่นกันเมื่อต้องการสื่อสารกับคอมพิวเตอร์เครื่องอื่น เครื่องนั้นก็จะทำการสอบถามหมายเลขไอพีของเครื่องที่ต้องการสื่อสารด้วยกับ DNS เซิร์ฟเวอร์ ซึ่งจะการค้นหาหมายเลขดังกล่าวในฐานข้อมูลแล้วแจ้งข้อมูลให้โฮสต์ดังกล่าวทราบ

ตัวอย่างเช่น หมายเลขไอพี 172.20.1.24 เมื่อมีการเข้าระบบ DNS อาจตั้งชื่อได้เป็น www.liverpoolthailand.com เมื่อเราพิมพ์ชื่อนี้ในช่องแอดเดรสของเว็บบราวเซอร์ DNS เซิร์ฟเวอร์ก็จะทำการค้นหาหมายเลขไอพีที่ตรงกับชื่อแล้วแจ้งให้โฮสต์ดังกล่าวรับทราบว่ามีการร้องขอเรียกดูข้อมูล การตั้งชื่อเราจะตั้งอะไรก็ได้ แต่ระบบจะมีการตรวจดูว่าชื่อที่ตั้งนั้นมีการใช้แล้วรึเปล่า ถ้ามีจะไม่สามารถใช้ชื่อนั้นๆ ได้อีกจะกว่ามีผู้ขอใช้รายเก่าจะทำการยกเลิกชื่อนั้นไป

การตั้งชื่อนั้นผู้ร้องขอจะต้องทำการลงทะเบียนกับผู้ให้บริการ DNS Server เพื่อทำการตั้งชื่อให้กับหมายเลขไอพีแอดเดรสที่เราต้องการ ซึ่งในวิทยานิพนธ์ฉบับนี้ขอที่จะไม่กล่าวถึงวิธีการลงทะเบียนเพื่อขอโดเมนเนม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.11.4.3 Mac address (Media Access Control address)

เป็นหมายเลขประจำตัวของอุปกรณ์ที่ต่ออยู่ในเครือข่าย ค่า Mac address จะถูกกำหนดจากบริษัทผู้ผลิตอุปกรณ์ตั้งแต่เริ่มผลิต เช่น อุปกรณ์ NIC (Network Interface Card) จะมีค่า Mac address ประจำตัวที่ไม่ซ้ำกันและไม่สามารถแก้ไขได้ ค่า Mac address เป็นการระบุค่าอ้างอิงของอุปกรณ์ฮาร์ดแวร์ในระดับล่างสุด (Physical layer) ของกลไกการรับส่งข้อมูลภายในเครือข่าย ถ้าใช้หมายเลข Mac address สำหรับการอ้างอิงในเครือข่ายแล้วจะเกิดปัญหามาก เมื่อมีการเปลี่ยนหรือย้ายเครื่องต้องมีการกำหนดระบบเครือข่ายใหม่ นอกจากนี้ยังจดจำได้ยากกว่า ตัวอย่างหมายเลข Mac address คือ 08:0a:0e:12:b5:05 การที่ IP Address ถูกใช้อ้างอิงในการติดต่อกันด้วยโปรโตคอล TCP/IP เพราะการใช้ IP Address จะยืดหยุ่นและคล่องตัวกว่า

2.12 โปรโตคอลที่ทำงานในชั้นโฮสต์ทูโฮสต์เลเยอร์ (Host to Host Layer)

2.12.1 Transmission Control Protocol (TCP)

โปรโตคอล TCP เป็นโปรโตคอลที่ให้บริการแบบคอนเน็กชันโอเรียลเต็ด (Connection-Oriented) ซึ่งเป็นการรับส่งข้อมูลที่เชื่อถือได้ TCP จะส่งข้อมูลทั้งหมดจนสำเร็จ ซึ่งถ้าข้อมูลมีขนาดใหญ่ก็จะถูกแบ่งย่อยเป็นหลายแพ็กเก็ต โปรโตคอล TCP จะทำการควบคุมการรับส่งแพ็กเกจข้อมูลย่อยเหล่านี้

กลไกในการควบคุมการไหลของข้อมูลมีรายละเอียดดังนี้

2.12.1.1 การจัดการเกี่ยวกับเซสชั่น

เนื่องจาก TCP เป็นโปรโตคอลที่ให้บริการแบบโอเรียนเต็ดคอนเน็กชัน ดังนั้นก่อนที่จะทำการส่งข้อมูลจำเป็นที่ต้องสร้างเซสชั่นเพื่อเชื่อมต่อกับโฮสต์ปลายทางก่อน เซสชั่นเป็นการสร้างการสนทนาอย่างเป็นรูปแบบระหว่างทั้งสองโฮสต์เพื่อใช้สำหรับการกู้คืนข้อมูลเมื่อเกิดข้อผิดพลาดระหว่างการรับส่งข้อมูล ขั้นตอนในการสร้างเซสชั่นมีอยู่ 3 ขั้นตอนบางที่ก็เรียกว่า "ทรีเวย์แฮนด์เชค(Three-Way Handshake)"

1. โฮสต์ที่ต้องการส่งข้อมูลจะส่งแพ็กเก็ตไปยังโฮสต์ปลายทางเพื่อแจ้งให้ทราบว่าต้องการที่จะส่งข้อมูล
2. โฮสต์ปลายทางก็จะตอบตกลงกลับมาพร้อมทั้งรหัสที่จะให้ในการรับข้อมูล
3. โฮสต์ต้นทางก็จะส่งแพ็กเก็ตพร้อมรหัสที่ได้รับ เพื่อเป็นการยืนยันการเชื่อมต่อ

หลังจากที่ได้มีการสร้างเซสชั่นสำเร็จแล้วถึงมีการเริ่มขบวนการรับส่งข้อมูลจริงๆ ซึ่งการรับส่งข้อมูลแต่ละครั้งก็จะมีการยืนยันการยืนยันการรับข้อมูลจากโฮสต์ปลายทางทุกครั้ง เมื่อส่งเสร็จก็ทำการยกเลิกเซสชั่นเป็นขั้น เหมือนการสร้างการเชื่อมต่อ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.12.1.2 การควบคุมการไหลและการกู้ข้อมูลคืน

ในแต่ล่ะเซสชั่นโฮสต์ฝ่ายรับต้องตอบกลับทุกๆแพ็กเก็ตภายในเวลาที่กำหนด เพื่อเป็นการยืนยันการรับข้อมูลทุกๆ แพ็กเก็ตที่ส่ง ฝ่ายรับจะทำการเช็คความถูกต้องของข้อมูลแพ็กเก็ตทุกครั้ง และแจ้งให้ทราบถึงผลการตรวจสอบนั้น ถ้าฝ่ายส่งไม่ได้รับการตอบรับจากฝ่ายรับภายในเวลาที่กำหนด ฝ่ายส่งก็จะคาดเดาเอาว่าแพ็กเก็ตสูญหายระหว่างทาง ฝ่ายส่งก็จะทำการส่งแพ็กเก็ตนั้นใหม่อีกครั้ง เพื่อให้มั่นใจว่าข้อมูลทุกๆแพ็กเก็ตจะส่งถึงปลายทางได้อย่างสมบูรณ์

นอกจากนี้การแบ่งข้อมูลออกเป็นแพ็กเก็ตย่อยๆ TCP ก็จะกำหนดหมายเลขลำดับ (Sequence Number) ให้แต่ล่ะแพ็กเก็ต เพื่อใช้สำหรับการจัดรวบรวมแพ็กเก็ตย่อยๆเหล่านั้นให้เป็นข้อมูลเดิม นอกจากนี้หมายเลขลำดับยังใช้ในการตรวจสอบว่าข้อมูลส่งถึงปลายทางทุกแพ็กเก็ตหรือไม่

กลไกการตอบกลับมีอยู่ 2 ประเภท คือ

1. PAR (Positive Acknowledgment and Retransmission) กลไกการทำงาน คือเมื่อฝ่ายส่งทำการส่งแพ็กเก็ตหนึ่งก็จะรอการดอบกลับของฝ่ายรับ แล้วจึงส่งแพ็กเก็ตต่อไป ถ้าไม่ได้รับการตอบกลับในเวลาที่กำหนดก็จะส่งแพ็กเก็ตนั้นไปอีกครั้ง ปัญหาก็คือถ้าข้อมูลมีแพ็กเก็ตมากๆ ฝ่ายรับต้องส่งแพ็กเก็ตตอบรับทุกๆครั้งทำให้เป็นการสิ้นเปลืองแบนด์วิธ และเป็นกระบวนการที่ไร้ประสิทธิภาพเนื่องจากรอข้อมูลมากว่าการส่ง

2. สไลดิ้งวินโดว์ (Sliding Window) กลไกการทำงานคือ ฝ่ายรับสามารถยืนยันการรับแพ็กเก็ตโดยส่งแพ็กเก็ตเดียวสำหรับการรับหลายแพ็กเก็ต วิธีนี้จะลดจำนวนแพ็กเก็ตที่ไหลเวียนในเครือข่าย ฝ่ายส่งสามารถส่งทีล่ะหลายแพ็กเก็ตก่อนที่จะรอตอบรับ ทำให้มีความรวดเร็วยิ่งขึ้น

เมื่อสร้างเซสชั่นเสร็จ ขั้นต่อไปคือการต่อรองเกี่ยวกับขนาดวินโดว์(Window Size) ขนาดของวินโดว์คือ จำนวนไบต์ที่ฝ่ายรับได้รับก่อนที่จะทำการตอบกลับ หรือจำนวนไบต์ที่ฝ่ายส่งสามารถส่งได้ก่อนที่จะรอการตอบกลับ การทำงานของสไลดิ่งวินโดว์มีขั้นตอนดังนี้

1. เมื่อโฮสต์ต้องการที่จะส่งข้อมูล TCP จะย้ายข้อมูลไปไว้ที่บัฟเฟอร์ที่จะใช้ส่งข้อมูล ซึ่งข้อมูลส่วนนี้จะเรียกว่า "เซกเมนต์ (Segment)" ซึ่งแต่ละเซกเมนต์อาจจะถูกแบ่งย่อยเป็นหลายๆ แพ็กเก็ต ซึ่งแต่ละแพ็กเก็ตก็จะถูกกำหนดหมายเลขลำดับ
2. ทุกๆแพ็กเก็ตในเซกเมนต์จะถูกส่งต่อไปให้โปรโตคอล IP เพื่อทำการส่งไปยังโฮสต์ปลายทาง
3. เซ็กเมนต์ข้อมูลจะยังคงถูกเก็บไว้ที่บัพเฟอร์จนกว่าจะได้รับการดอบรับจากโฮสต์ฝ่ายรับก่อนและโฮสต์ฝ่ายส่งจะตั้งเวลาเพื่อรอการตอบกลับถ้าโฮสต์ฝ่ายรับไม่ตอบกลับภายในเวลาที่กำหนด ข้อมูลที่อยู่ในบัฟเฟอร์ก็จะถูกส่งใหม่อีกครั้ง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4. เมื่อแพ็กเก็ตเดินทางมาถึงฝ่ายรับ โฮสต์ฝ่ายรับก็จะใช้หมายเลขลำดับในการเรียบเรียงแพ็กเก็ตให้ได้เป็นเซกเมนต์ดังเดิม
5. เมื่อโฮสต์ฝ่ายรับได้รับแพ็กเก็ตครบและตรวจสอบแล้วไม่มีข้อผิดพลาดใดๆ ก็จะส่งแพ็กเก็ตตอบกลับไปยังโฮสต์ฝ่ายรับว่าได้รับข้อมูลครบหมดแล้ว
6. เมื่อโฮสต์ฝ่ายรับได้รับการตอบกลับ เซ็กเมนต์ในบัฟเฟอร์ก็จะถูกลบทิ้งไปแล้วทำการส่งเซ็กเมนต์ถัดไปถ้ามี จนกว่าข้อมูลจะถูกส่งทั้งหมด

ขบวนการส่งข้อมูลแบบนี้ทำให้มั่นใจได้ว่าข้อมูลจะส่งถึงปลายทางอย่างแน่นอนและถูกต้อง ซึ่งการให้บริการแบบนี้จะเรียกว่า "คอนเน็กชันโอเรียนเต็ด (Connection-Oriented)" นั่นเอง

ตารางที่ 2.4 ฟอร์แมตข้อมูลของแพ็กเก็ต TCP

32 บิต

Source Port			Destination Port
Sequence Number			
Acknowledgement Number			
Data Offset	Reversed	Flag	Window Size
Checksum			Urgent Pointer
Options+ Padding			
Data			

ข้อมูลในส่วนหัวของโปรโตคอล TCP จะประกอบด้วยข้อมูลมากที่สุด 20 ไบต์ และประกอบด้วยส่วนต่างๆ ดังแสดงในตารางที่ 2.14 ซึ่งแต่ล่ะฟิลด์มีความหมายดังต่อไปนี้

o **TCP Source Port (16 บิต):** ส่วนนี้จะเป็นหมายเลขพอร์ตที่เป็นจุดเริ่มการสื่อสาร หมายเลขพอร์ตเมื่อรวมกับหมายเลข IP จะเป็นที่อยู่ของการรับส่งข้อมูลกลับ

o **TCP Destination Port (16 บิต):** เป็นหมายเลขพอร์ตของเครื่องรับ ซึ่งพอร์ตนี้จะเป็นพอร์ตที่ใช้เชื่อมต่อกับแอพลิเคชันที่จะนำข้อมูลที่ส่งไปให้นี้ไปโพรเซสต่อไป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

o **TCP Sequence Number (32 บิต)**: เป็นหมายเลขที่บอกลำดับแพ็กเก็ตที่จะใช้ โดยฝั่งเครื่องรับในการเรียงข้อมูลให้อยู่ในรูปแบบเดิม ในการส่งข้อมูลผ่านเครือข่ายที่สลับซับซ้อนนั้นแพ็กเก็ตแต่ละชุดอาจถูกส่งไปบนเส้นทางที่ต่างกัน ดังนั้นจึงเป็นไปได้ที่แพ็กเก็ตจะเดินทางมาถึงปลายทางไม่เป็นไปตามลำดับที่ส่ง หมายเลขนี้จะใช้ในการจัดเรียงแพ็กเก็ตเหล่านี้ ให้อยู่ในลำดับเดิม

o **TCP Acknowledgement Number (32 บิต)**: เป็นหมายเลขลำดับแพ็กเก็ตถัดไปที่ทางฝั่งรับคาดหวัง ซึ่งเป็นการบอกเป็นนัยว่าแพ็กเก็ตที่มีหมายเลยลำดับก่อนหน้านี้ได้รับหมดแล้วนั่นเอง

o**Data Offset (4 บิต)**: เป็นตัวเลขที่บอกขนาดของข้อมูลส่วนหัว (TCP Header) ซึ่งมีหน่วยเป็น 32 บิต หรือ word

o**Reversed (6 บิต) :** ส่วนนี้จะถูกกำหนดให้เป็นศูนย์ตลอด ซึ่งข้อมูลส่วนนี้ไม่มีความหมายอะไรเพียงแต่เป็นการสงวนไว้ใช้ในอนาคตเมื่อมีการปรับปรุงโปรโตคอล

o**Flags (6 บิต)** : เป็นข้อมูลที่ใช้สำหรับควบคุมการรับส่งแพ็กเก็ต เช่น บิต SYN และ ACK ใช้สำหรับสร้างการเชื่อมต่อ ส่วนบิต FIN เป็นการแจ้งการยกเลิกการเชื่อมต่อ

o**Window Size (16 บิต)** : เป็นตัวเลขที่เครื่องปลายทางบอกให้เครื่องต้นทางทราบขนาดของวินโดว์ที่เครื่องปลายทางสามารถรับข้อมูลได้

o**Checksum (16 บิต)** : เป็นข้อมูลที่ใช้ในการตรวจสอบข้อผิดพลาดของข้อมูลในส่วนหัว เมื่อเครื่องปลายทางได้รับข้อมูลก็จะทำการคำนวณ เช็คซัม (Checksum) ของข้อมูลส่วนหัว เมื่อเครื่องปลายทางได้รับข้อมูลก็จะทำการคำนวณเช็คซัมด้วยวิธีเดียวกัน แล้วทำการเปรียบเทียบค่าของข้อมูลที่คำนวณได้กับค่าที่อยู่ในฟิลด์นี้ ถ้าเหมือนกันแสดงว่าไม่มีข้อผิดพลาดเกิดขึ้นในข้อมูลที่ได้รับ

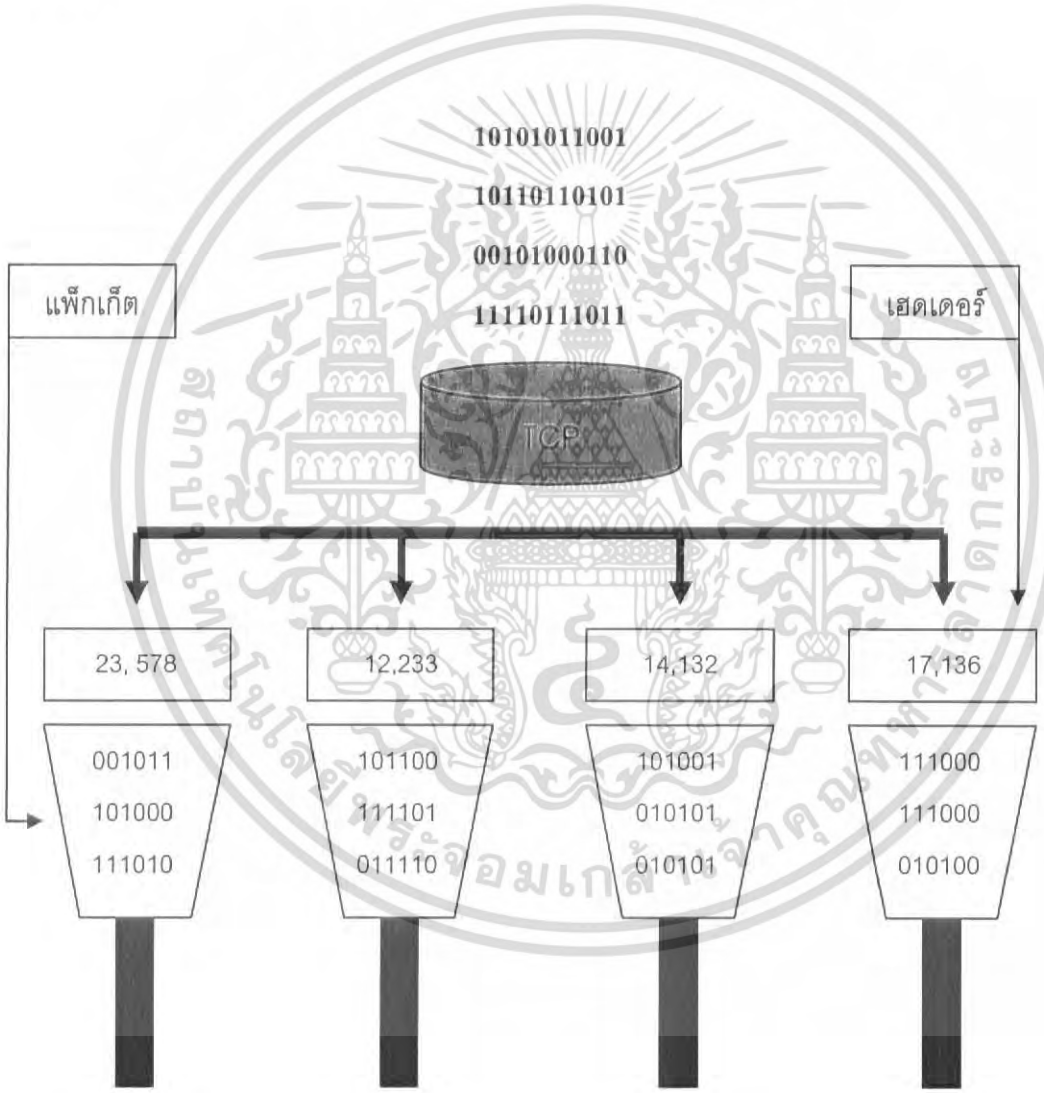
o**Padding :** เป็นข้อมูลที่เพิ่มขึ้นเพื่อให้ข้อมูลส่วนหัวมีจำนวนบิตที่หารด้วย 32 ลงตัว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.12.2 การทำงานของ TCP/IP Stack

แบ่งเป็นขั้นตอนหลักๆดังนี้

ขั้นแรก ข้อมูลที่จะส่งผ่านทางอินเตอร์เน็ต จะถูกแบ่งออกเป็นแพ็กเก็ตย่อยๆ ที่มีขนาดเล็กกว่า 1500 ตัวอักษรหรือ 1500 ไบต์ โดย TCP จะทำการแตกข้อมูลออกเป็นแพ็กเก็ต แต่ละแพ็กเก็ตจะมีส่วนหัวหรือเฮดเดอร์ (header) ซึ่งจะเก็บข้อมูลหลายอย่างไว้ เช่นลำดับที่ใช้ในการประกอบข้อมูลกลับตามเดิมและขณะที่ทำการแยกข้อมูลออกเป็นแพ็กเก็ตก็จะมีการคำนวณค่าผลรวมสำหรับตรวจสอบหรือ checksum ขึ้นจากลักษณะและปริมาณของข้อมูลในแพ็กเก็ตแล้วใส่ค่านี้ลงไปในเฮดเดอร์ด้วย ดังแสดงในรูปที่ 2.11

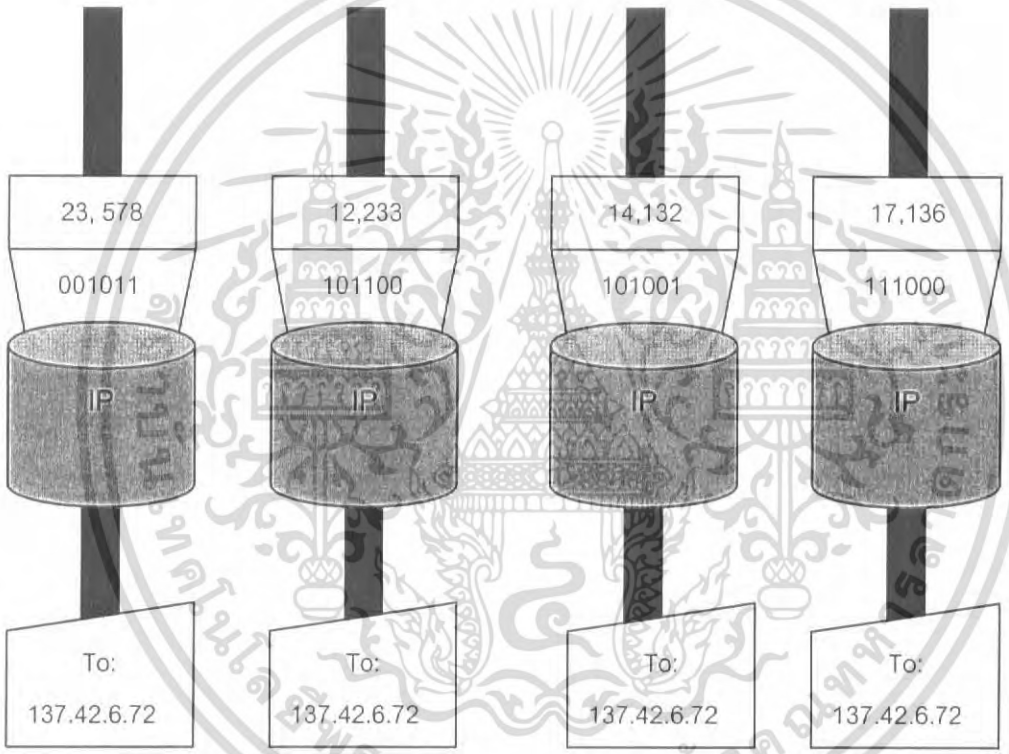


รูปที่ 2.11 การทำงานของโปรโตรคอล TCP

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ขั้นที่ 2

แต่ละแพ็กเก็ตจะถูกใส่ไปใน "ซอง" IP ที่แยกจากกัน ซองเหล่านี้จะบรรจุข้อมูลแอดเดรสที่บอกว่าให้ส่งข้อมูลไปที่ไหน ซองทั้งหมดสำหรับข้อมูลชุดหนึ่งจะมีแอดเดรสที่เหมือนกัน เพื่อจะได้ส่งไปที่เดียวกันแล้วประกอบกลับอย่างเดิมได้ แต่ละ "ซอง" IP จะต้องมีเฮดเดอร์ซึ่งบรรจุข้อมูลต่างๆ เช่น แอดเดรสของผู้ส่ง แอดเดรสของผู้รับ ช่วงเวลาหรืออายุที่แพ็กเก็ตจะถูกเก็บไว้ก่อนที่จะถูกทิ้งไปเพราะเก่าเกิน (เช่นในกรณีแพ็กเก็ตที่มีข้อมูลขึ้นกับเวลาจริงหรือ real-time เช่นส่วนของภาพเคลื่อนไหวหรือเสียง ถูกส่งไปผิดทางหรือไม่ถึงที่หมายสักทีจนช้าเกินกว่าจะไปทันแพ็กเก็ตอื่นๆและหมดความจำเป็นต้องส่งไปอีก ก็จะทิ้งไปได้เลย) และข้อมูลอื่นๆอีกมาก ซึ่งจะแสดงขั้นตอนได้ดังรูปที่ 2.12



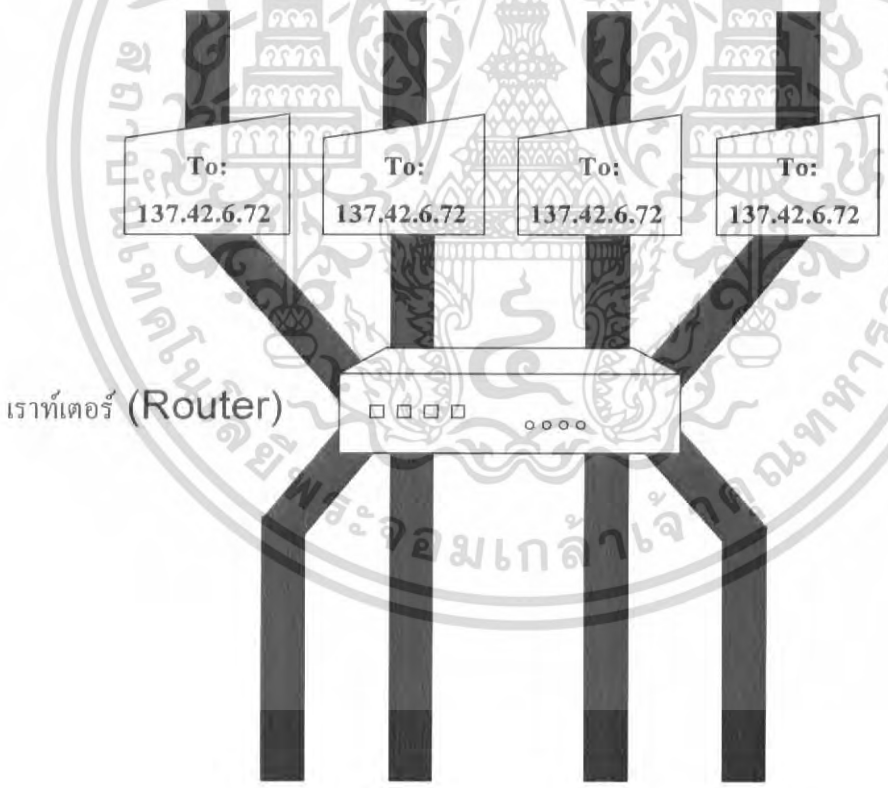
รูปที่ 2.12 การทำงานของโปรโตคอล IP

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ขั้นที่ 3

ขณะที่แพ็กเก็ต ถูกส่งข้ามอินเตอร์เน็ต เราท์เตอร์ที่อยู่ตามทางจะตรวจสอบซอง IP โดยที่แอดเดรสของมัน และจะหาเส้นทางที่ดีที่สุดสำหรับส่งแต่ละแพ็กเก็ตไปยังเราท์เตอร์ตัวที่อยู่ใกล้ปลายทางของแพ็กเก็ตนั้นๆ ที่สุด หลังจากเดินทางผ่านเราท์เตอร์หลายๆตัวแล้ว แพ็กเก็ตก็จะมาถึงปลายทาง แต่ด้วยเหตุที่การจราจรบนอินเตอร์เน็ตมีการเปลี่ยนแปลงตลอดเวลา แพ็กเก็ตอาจถูกส่งไปยังเส้นทางต่างๆ กันและมาถึงปลายทางโดยไม่เรียงลำดับกันก็ได้ ซึ่งจะแสดงขั้นตอนได้ดังรูปที่ 2.13

เราท์เตอร์จะอ่านที่อยู่ (Address) ของสถานีปลายทางที่ส่วนหัว (Header) ของแพ็กเก็ตข้อมูลเพื่อใช้ในการกำหนดหรือเลือกเส้นทางที่จะส่งแพ็กเก็ตนั้นต่อไป ในเราท์เตอร์จะมีข้อมูลเกี่ยวกับการจัดการกับแพ็กเก็ตที่เรียกว่า "เราท์ติ้งเทเบิ้ล (Routing Table)" หรือตารางการจัดเส้นทาง ข้อมูลในตารางนี้จะเป็นข้อมูลที่เราท์เตอร์จะใช้ในการเลือกเส้นทางที่ดีที่สุดยังปลายทางถ้าเส้นทางหลักเกิดขัดข้อง เราท์เตอร์ก็สามารถสร้างเส้นทางใหม่ได้ และเราท์เตอร์ยังสามารถเชื่อมต่อเครือข่ายเข้ากับเครือข่ายอื่นๆ ที่ใหญ่กว่าได้ เช่น เครือข่ายอินเตอร์เน็ต เป็นต้น

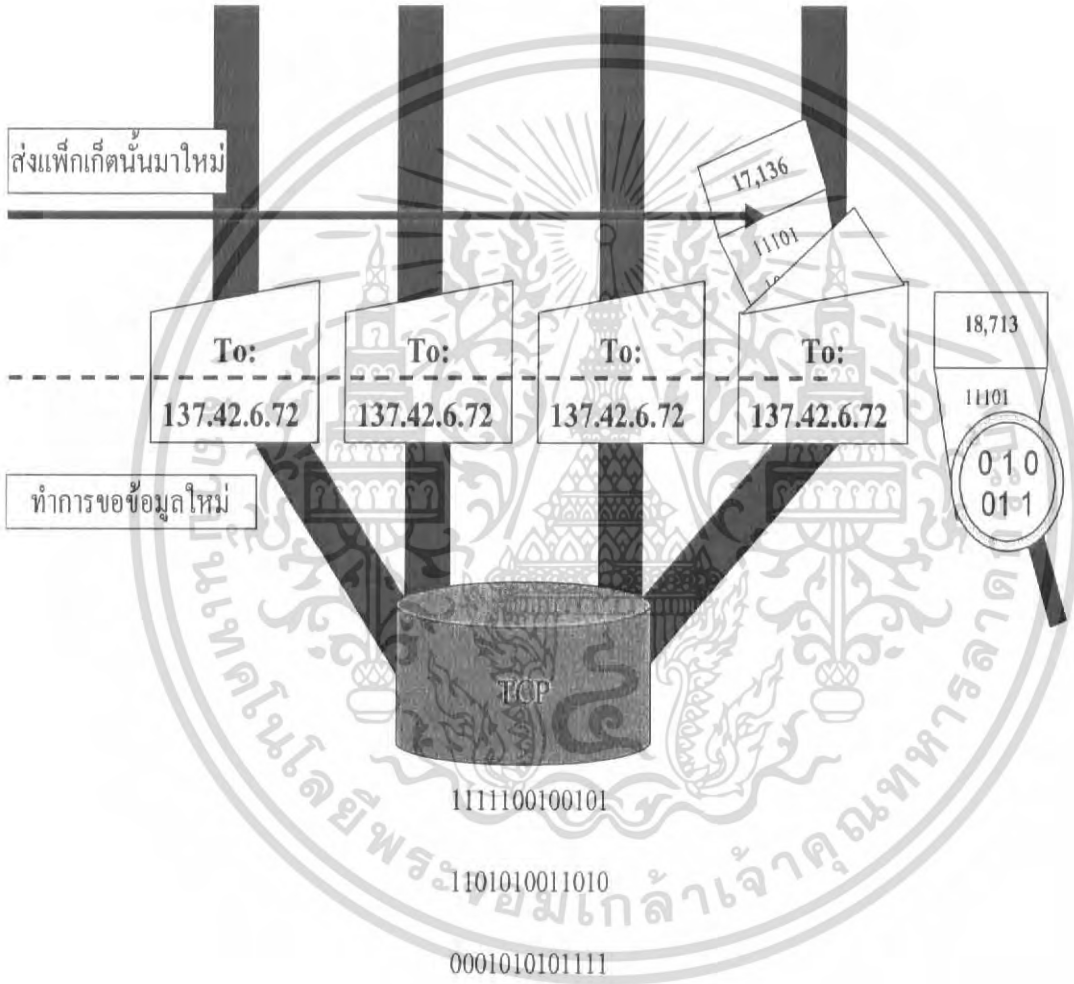


รูปที่ 2.13 การทำงานของเราท์เตอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ขั้นสุดท้าย

รูปที่ 2.14 จะแสดงให้เห็นว่าเมื่อแพ็กเก็ตมาถึงปลายทาง TCP จะคำนวณค่า checksum ของแต่ละแพ็กเก็ตใหม่แล้วเปรียบเทียบกับค่า checksum ที่ส่งมาในแพ็กเก็ตนั้นถ้าไม่เท่ากัน TCP บนเครื่องปลายทางก็จะรู้ว่าข้อมูลไม่สมบูรณ์คือเกิดการผิดพลาดขึ้นในระหว่างการส่ง และจะทิ้งแพ็กเก็ตนั้นไปแล้วร้องขอให้ TCP บนเครื่องต้นทางส่งแพ็กเก็ตนั้นมาใหม่เมื่อได้รับแพ็กเก็ตที่สมบูรณ์ครบทั้งหมดแล้ว TCP ก็จะทำการประกอบข้อมูลนั้นกลับขึ้นมาเป็นรูปแบบเดิม



รูปที่ 2.14 การทำงานของโปรโตคอล IP เมื่อมีการรับแพ็กเก็ต

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.13 โปรโตคอลที่ทำงานในชั้นแอพพลิเคชั่นเลเยอร์(Application layer)

การทำงานของโปรโตคอลในชั้นนี้จะเป็นการเข้าถึงข้อมูลที่มีระยะไกล และโปรโตคอลที่จัดอยู่ในชั้นนี้คือโปรโตคอล HTTP (Hyper Text Transfer Protocol)

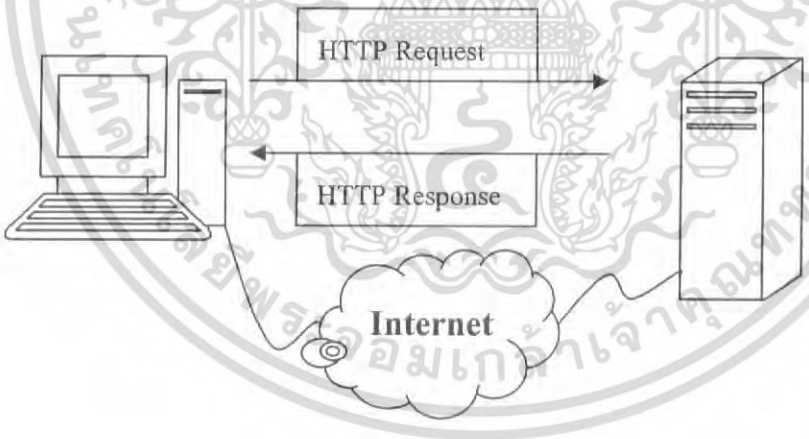
2.13.1 เวิร์ดไวด์เว็บ (World Wide Web): HTTP (Hyper Text Transfer Protocol)

เวิร์ลไวด์เว็บ หรือเรียกสั้นๆว่าเว็บ(Web) หรือ WWW เป็นแอพพลิเคชันชนิดหนึ่งที่ทำให้อินเตอร์เน็ตเป็นที่นิยมมากในปัจจุบัน WWW ใช้โปรโตคอล HTTP ซึ่งเป็นโปรโตคอลที่ใช้รับส่งไฟล์ HTML (Hyper Text Markup Language) นั้นเป็นภาษาที่ใช้อธิบายการแสดงเว็บเพจนั่นเอง WWW เป็นแอพพลิเคชันที่ทำงานแบบไคลเอนท์เซิร์ฟเวอร์ กล่าวคือ WWW จะมีโฮสต์เครื่องหนึ่งที่ทำหน้าที่เป็นเซิร์ฟเวอร์ เรียกว่า "เว็บเซิร์ฟเวอร์" ซึ่งจะทำหน้าที่ให้บริการเอกสาร HTML

ส่วนเครื่องไคลเอนท์นั้นใช้โปรแกรมเว็บบราวเซอร์ (Web Browser) เช่น อินเตอร์เน็ตเอ็กพลอเรอร์ (IE) ซึ่งจะทำการร้องขอไฟล์ HTML จากเว็บเซิร์ฟเวอร์และแสดงผลให้ผู้ใช้ดู

2.13.1.1 กลไกการทำงานของโปรโตคอล HTTP

โปรโตคอล HTTP จะเป็นตัวที่กำหนดรูปแบบการร้องขอไฟล์ของไคลเอนท์ (เว็บบราวเซอร์) จากเว็บเซิร์ฟเวอร์ และรูปแบบการถ่ายโอนไฟล์จากเว็บเซิร์ฟเวอร์ไปยังไคลเอนท์ แสดงโดยขั้นตอนคร่าวๆได้ดังรูปที่ 2.15



รูปที่ 2.15 การทำงานของโปรโตคอล HTTP

กระบวนการนั้นจะเริ่มขึ้นที่ฝั่งไคลเอนท์ โดยผู้ใช้ที่คลิกลิงค์ในเว็บเพ็จ หรือพิมพ์ URL (Uniform Resource Locator) ในช่องที่อยู่ของเว็บบราวเซอร์ หลังจากนั้นเว็บบราวเซอร์จะทำการส่งการร้องขอ (HTTP Request) ผ่านเครือข่ายไปยังเว็บเซิร์ฟเวอร์ เมื่อเซิร์ฟเวอร์ได้รับการร้องขอก็จะทำการค้นหาไฟล์ที่กำหนดใน URL ซึ่งถ้าพบก็จะตอบกลับ (HTTP Response) พร้อมกับไฟล์ กลับไปยังฝั่งไคลเอนท์ เว็บบราวเซอร์เมื่อได้รับการตอบกลับก็จะแสดงไฟล์นั้นให้ผู้ใช้ดู โปรโตคอล HTTP นั้น

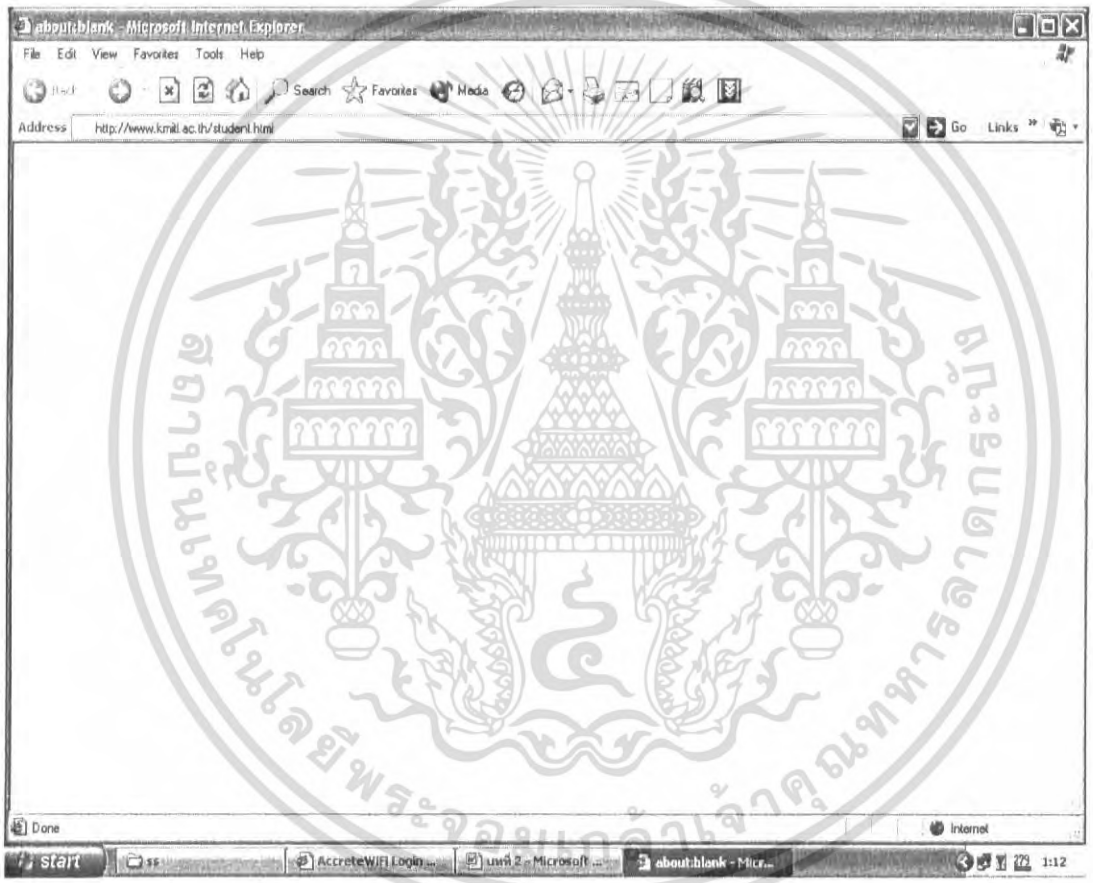
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ไม่ได้กำหนดรูปแบบการแสดงผลให้ผู้ใช้ดู ซึ่งหน้าที่นี้เป็นของเว็บบราวเซอร์ ดังนั้นเว็บบราวเซอร์ที่ต่างกันอาจแสดงผลไม่เหมือนกันก็ได้

สมมุติว่าผู้ใช้ต้องการดูเว็บเพ็จโดยมี URL ด้านล่าง ขั้นตอนการทำงานอย่างละเอียดของโปรโตคอล HTTP สามารถสรุปได้ดังนี้

http://www.kmitl.ac.th/student.html

1. ผู้ใช้จะต้องทำการเปิดเว็บบราวเซอร์ขึ้นมา ตัวอย่างที่ใช้จะเป็น เว็บบราวเซอร์ที่ชื่อ Internet Explorer จากนั้นทำการพิมพ์ URL ลงในช่อง Address ดังรูปที่ 2.16

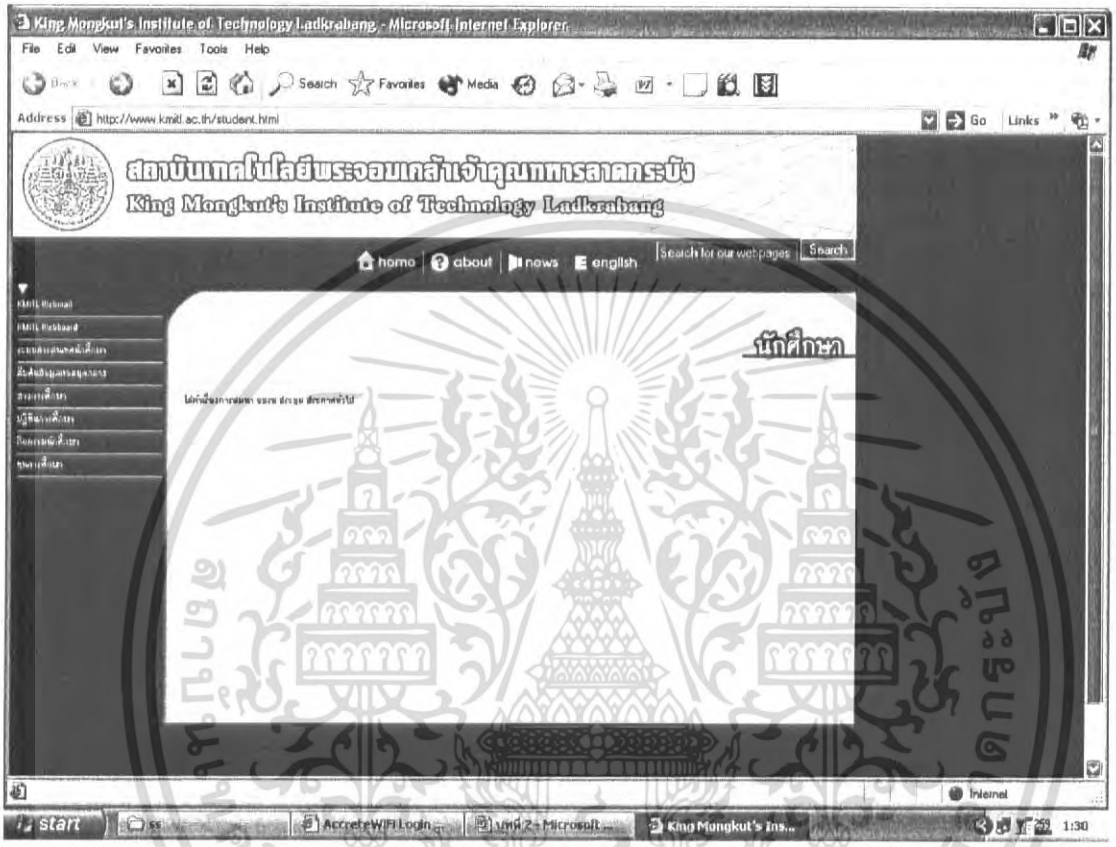


รูปที่ 2.16 แสดงหน้าต่าง Microsoft Internet Explorer

2. เมื่อกดปุ่ม Enter ที่คีย์บอร์ด (Key Board) เว็บบราวเซอร์จะเริ่มสร้างการเชื่อมต่อ (TCP Connection) ไปยังเว็บเซิร์ฟเวอร์ www.kmitl.ac.th ในกรณีที่ไม่ได้กำหนดพอร์ตใน URL นั้น พอร์ต 80 จะถูกใช้เป็นดีฟอลต์พอร์ตที่เว็บเซิร์ฟเวอร์จะคอยฟังเพื่อรับการร้องขอ
3. เว็บบราวเซอร์จะทำการส่งข้อความการร้องขอ (HTTP Request) ผ่านพอร์ตที่สร้างผ่านการเชื่อมต่อ TCP ในข้อที่ 2 โดยข้อความที่ส่งไปนั้นจะรวมถึงชื่อไฟล์ /student.html ด้วย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4. เมื่อเว็บเซิร์ฟเวอร์ได้รับการร้องขอผ่านทางพอร์ตที่ได้สร้างในข้อที่ 2 ก็จะค้นหาไฟล์ student.html แล้วส่งไปในแพ็กเก็ตการตอบกลับ (HTTP Response) กลับไปยังฝั่งไคล-เอนท์
5. เมื่อไคลเอนท์ได้รับข้อความตอบกลับ (HTTP Response) จากเซิร์ฟเวอร์ ก็จะอ่านไฟล์ student.html ซึ่งจะอยู่ในรูปแบบของ HTML และแสดงผลออกมาเป็นหน้าเว็บเพ็จดังรูปที่ 2.17



รูปที่ 2.17 แสดงหน้าต่างเว็บเพ็จ

2.13.1.2 HTML (Hyper Text Markup Language)

HTML เป็นสิ่งที่ผู้พัฒนาเว็บเพ็จคุ้นเคยกันดีเพราะเป็นภาษาที่ออกแบบมาให้เว็บบราวเซอร์สามารถเข้าใจ ซึ่งผู้สร้างเว็บจะใช้ภาษา HTML นี้สร้างเว็บเพ็จแล้วเก็บไว้ในเว็บเซิร์ฟเวอร์ เมื่อมีผู้ใช้งานติดต่อผ่านทางเว็บบราวเซอร์ที่เครื่องไคลเอนท์ โดยระบุ URL ของเว็บเซิร์ฟเวอร์นั้นๆ ไฟล์ html ที่ถูกเก็บไว้ในเว็บเซิร์ฟเวอร์นั้นก็จะถูกส่งไปโดยใช้โปรโตคอล HTTP และแสดงผลให้ผู้ใช้งานเห็นผ่านทางเว็บบราวเซอร์ ภาษา HTML ปัจจุบันได้ถูกพัฒนาขึ้นมาเป็นเวอร์ชัน 4.0 ซึ่งสามารถใช้งานร่วมกับเว็บบราวเซอร์ทั่วไปได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

แท็ก (Tag) เป็นส่วนสำคัญในการเรียนรู้ภาษา HTML โดยแท็กจะทำหน้าที่กำหนดขอบเขต หรือแบ่งแยกการสั่งงานต่างๆในไฟล์ HTML ให้ตัวแปลภาษาสามารถเข้าใจได้ โดยการเริ่มต้นแท็กจะใช้เป็นตัวอักษรอยู่ภายในสัญลักษณ์ “<” และ “>” และปิดท้ายด้วยตัวอักษรที่อยู่ระหว่างสัญลักษณ์ “</” และ “>”

โครงสร้างทั่วไปของไฟล์ HTML

```
**********************************************************
<HTML>
<HEAD>
<TITLE>
< ---------- รายละเอียดของ Title ---------- >
</TITLE>
< ---------- Header อื่นๆ ---------- >
</HEAD>
<BODY>
< ---------- ส่วนที่เป็นเนื้อหาของเอกสาร ---------- >
</BODY>
</HTML>
**********************************************************
```

ในการใช้คำสั่ง HTML นั้นจะมีแท็กบางประเภทที่จำเป็น โดยในไฟล์จะต้องเริ่มต้นด้วยแท็ก <HTML> และผิดท้ายด้วย</HTML> เสมอ ซึ่งใช้เป็นตัวกำหนดขอบเขตของไฟล์ htmlทั้งหมด สำหรับรายละเอียดภายในประกอบด้วย <HEAD> </HEAD> เป็นแท็กที่กำหนดเฮดเดอร์ ซึ่งภายในเฮดเดอร์นี้จะประกอบด้วยแท็ก<TITLE> </TITLE> ซึ่งกำหนดขอบเขตที่เป็นแท็กซ์ และใช้แสดงหัวเรื่องของเว็บเพ็จที่ด้านบน รายละเอียดของแท็กประเภทต่างๆ หาอ่านได้อย่างละเอียดตามคู่มือการเขียนเว็บเพ็จโดยใช้ภาษา HTML ทั่วไป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.13.2 การทดลองเขียนเว็บเพจอย่างง่ายๆ

มีขั้นตอนดังนี้

1. ทำการเปิดโปรแกรม Notepad ที่อยู่ในวินโดว์ขึ้นมาดังรูปที่ 2.18

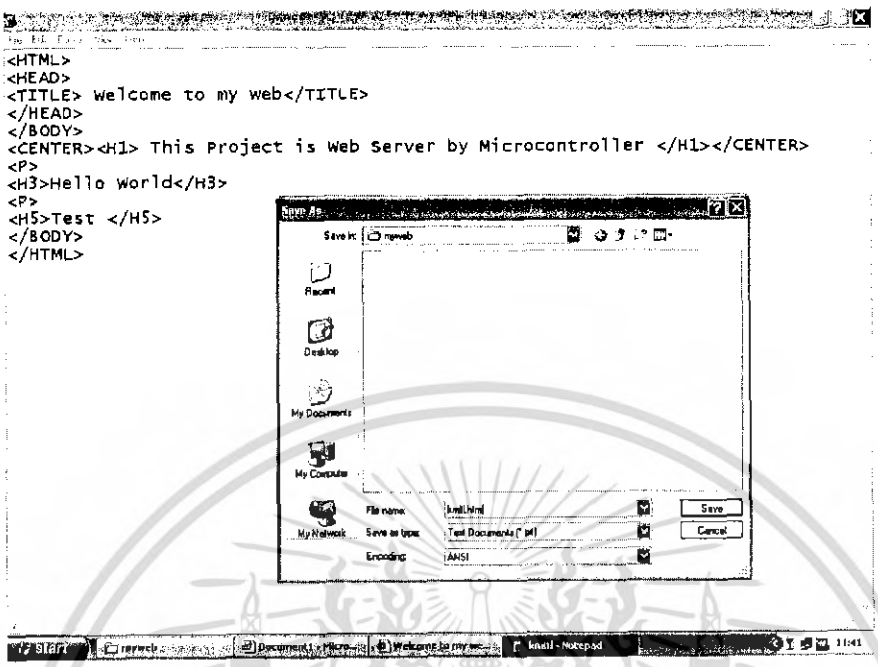
รูปที่ 2.18 แสดงหน้าต่างโปรแกรม Notepad

2. ทำการพิมพ์ข้อมูลตามด้านล่างลงใน Notepad

```
<HTML>
<HEAD>
<TITLE> Welcome to my web</TITLE>
</HEAD>
</BODY>
<CENTER><H1> This Project is Web Server by Microcontroller </H1></CENTER>
<P>
<H3>Hello World</H3>
<P>
<H5>Test </H5>
</BODY>
</HTML>
```

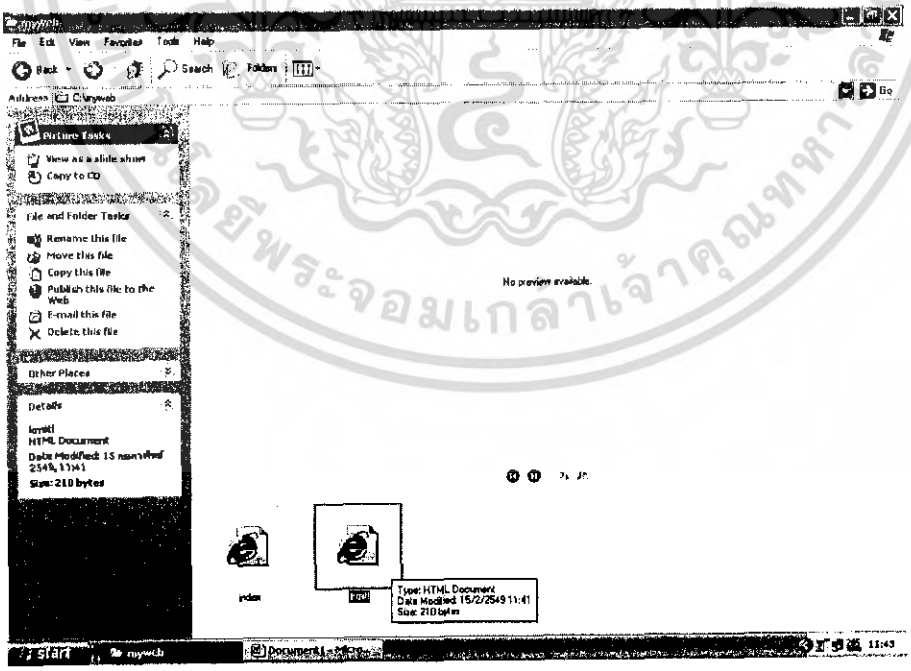
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3. ทำการเซฟไฟล์ดังกล่าว สมมุติให้ชื่อเป็น kmitl.html (หลัง "." จะต้องเป็น html หรือ htm) ดังรูปที่ 2.19



รูปที่ 2.19 แสดงการบันทึกไฟล์

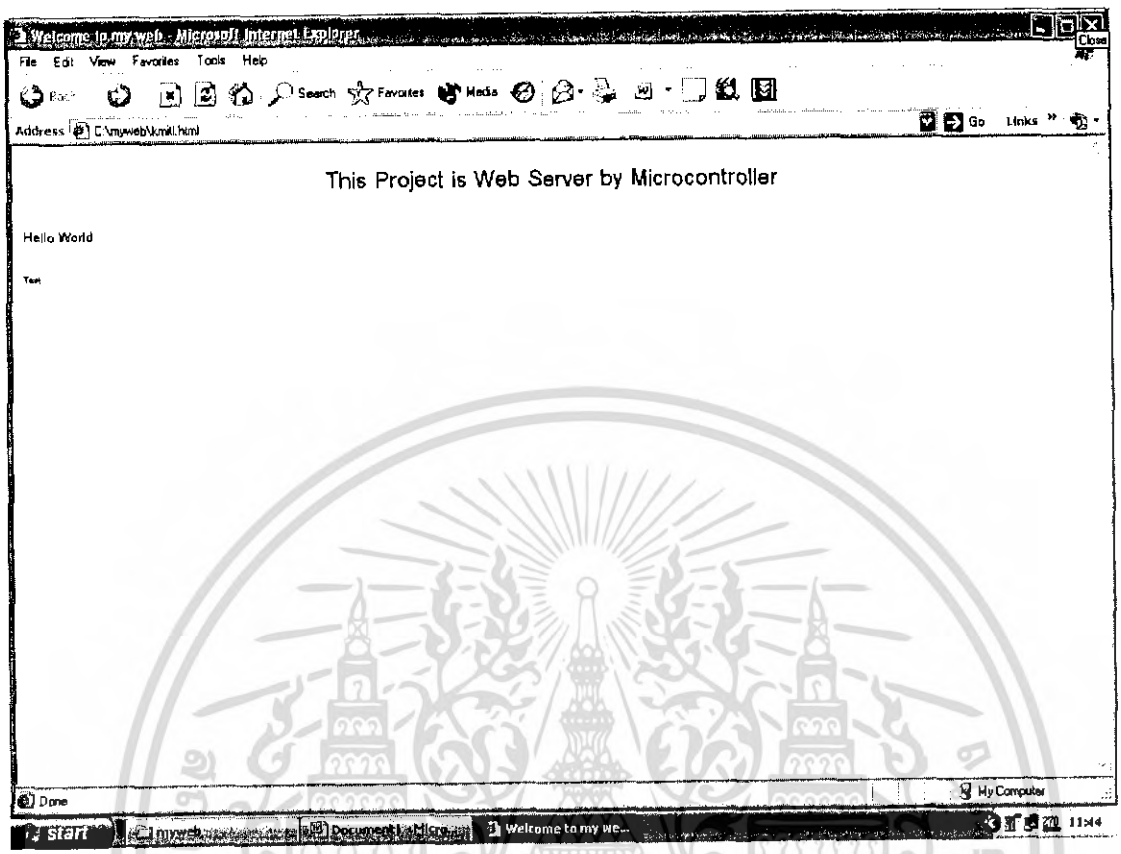
จากรูปที่ 2.19 เราจะได้ไฟล์ที่ชื่อ Kmitl.html มาหนึ่งไฟล์ที่อยู่ในรูปเว็บเพ็จที่พร้อมใช้งานแล้ว



รูปที่ 2.20 แสดงลักษณะของไฟล์เว็บเพ็จ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เมื่อทำการดับเบิ้ลคลิกไฟล์ก็จะได้ออกมาดังรูปที่ 2.21



รูปที่ 2.21 แสดงเว็บเพ็จที่ได้ทำการเขียนเอาไว้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

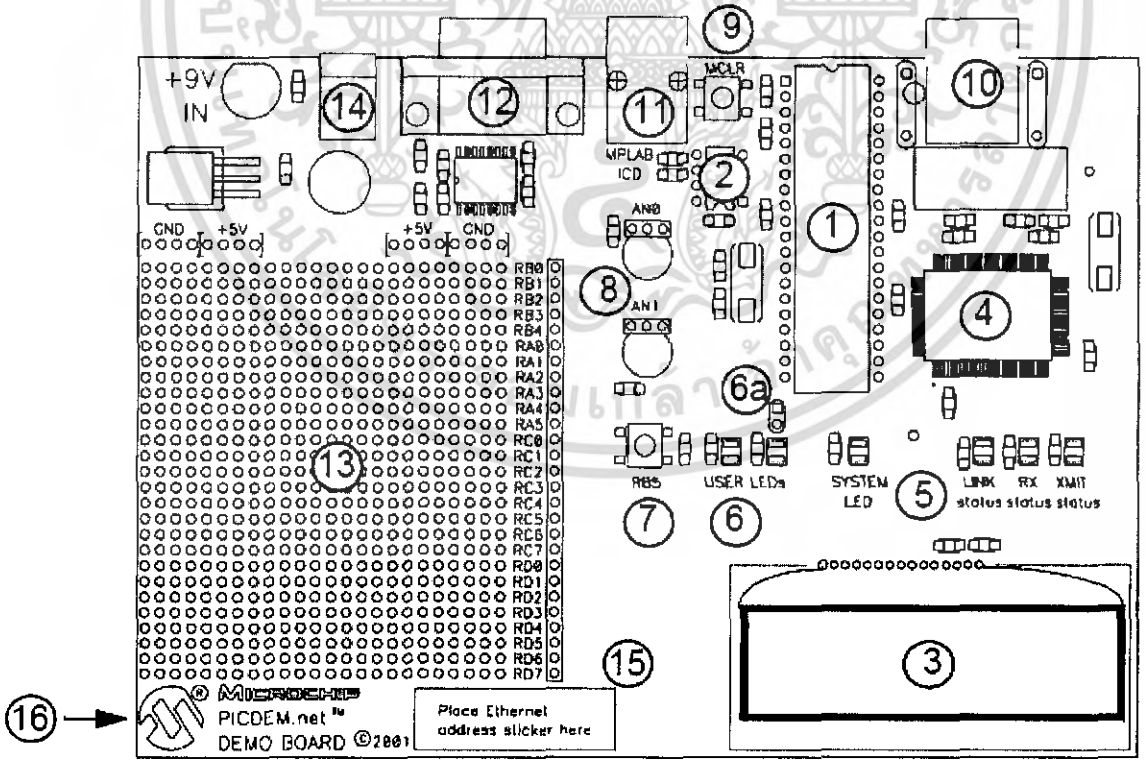
บทที่ 3

บอร์ดทดลอง PICDEM.net

บอร์ดทดลอง PICDEM.net ได้ถูกเตรียมมาเพื่อใช้สำหรับติดต่อกับระบบอินเตอร์เน็ต เพื่อใช้ในการแสดงเว็บเพ็จ และยังมีความสามารถในการควบคุมอุปกรณ์ระยะไกลผ่านทางเครือข่ายอินเตอร์เน็ต โดยบอร์ดทดลองนี้มีที่ไว้ให้สำหรับประยุกต์ต่อวงจรควบคุมอุปกรณ์เพื่อความสะดวกในการใช้งานอีกด้วย

3.1 อุปกรณ์ที่มีมากับบอร์ดทดลอง

1. บอร์ดทดลอง PICDEM.net
2. หม้อแปลงสำหรับบอร์ดทดลอง (แปลงจาก 110-230 โวลต์เป็น 9 โวลต์)
3. สายพอร์ตอนุกรมสำหรับติดต่อกับเครื่องคอมพิวเตอร์ (DB9, M/F connectors)
4. สายสำหรับต่อเครือข่าย (RJ45 connectors)
5. ซีดีรอม คู่มือการใช้บอร์ดทดลอง



รูปที่ 3.1 บอร์ดทดลอง PICDEM.net

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.1.1 อุปกรณ์ที่มีบนบอร์ดทดลอง

1. **ช่องสำหรับเสียบไมโครคอนโทรลเลอร์ :** เป็นช่องสำหรับเสียบไมโครคอนโทรลเลอร์ขนาด 40 ขา บอร์ดทดลองนี้รองรับไมโครคอนโทรลเลอร์ตระกูล PIC16 และ PIC18
2. **หน่วยความจำ :** เป็นแบบอีอีพรอม เบอร์ 24LC256 ใช้สำหรับเก็บเว็บเพ็จมีขนาด 256 กิโลบิต
3. **จอแสดงผล LCD :** เป็นแบบสองบรรทัด 16 ตัวอักษร
4. **ไอซีควบคุมระบบอีเธอร์เน็ต :** ที่มีมากับบอร์ดทดลองเป็นเบอร์ RTL8019AS ใช้สำหรับให้ไมโครคอนโทรลเลอร์สามารถติดต่อกับระบบอินเตอร์เน็ตได้
5. **ไฟแสดงสถานะ :** มีทั้งหมด 4 ดวง
   - **ไฟระบบ (SYSTEM) :** แสดงเมื่อเปิดบอร์ดทดลองและเมื่อการติดต่อสมบูรณ์ไฟระบบจะกระพริบ
   - **ไฟแสดงการเชื่อมต่อ (LINK) :** แสดงเมื่อมีการเชื่อมต่อกับระบบเครือข่าย
   - **ไฟแสดงการรับส่งข้อมูล (XMIT และ RX) :** ใช้แสดงเมื่อมีการรับส่งข้อมูล โดยเมื่อมีการรับข้อมูลไฟ RX จะกระพริบ และเมื่อมีการส่งข้อมูลไฟ XMIT จะกระพริบ
6. **ไฟสำหรับผู้ใช้ (USER LEDs) :** เตรียมไว้ให้สำหรับผู้ใช้ทำการทดลองเขียนโปรแกรมติดต่อกับบอร์ดทดลอง โดยหลอด LED จะต่ออยู่กับขาของไมโครคอนโทรลเลอร์
7. **ปุ่มสำหรับผู้ใช้ (RB5) :** เตรียมไว้ให้สำหรับผู้ใช้ทำการทดลองเขียนโปรแกรมติดต่อกับบอร์ดทดลอง โดยปุ่ม RB5 จะต่ออยู่กับขาของไมโครคอนโทรลเลอร์
8. **ตัวความต้านทานปรับค่าได้สำหรับผู้ใช้ (AN0 และAN1) :** ตัวความต้านทานปรับค่าได้สำหรับผู้ใช้มีค่าความต้านทานเท่ากับ 10 กิโลโอห์ม เตรียมไว้สำหรับผู้ใช้ทำการทดลองเขียนโปรแกรมติดต่อกับบอร์ดทดลองโดยตัวต้านทานปรับค่าได้จะต่ออยู่กับขาของไมโครคอนโทรลเลอร์
9. **ปุ่ม RESET (MCLR) :** ใช้สำหรับ RESET บอร์ดทดลอง
10. **ช่องสำหรับเสียบสายสำหรับต่อกับเครือข่าย (RJ45)**
11. **ช่องสำหรับเสียบสายสำหรับต่อกับตัวอัดโปรแกรม (RJ11)**
12. **ช่องสำหรับเสียบสายอนุกรมสำหรับต่อกับเครื่องคอมพิวเตอร์ (DB9M)**
13. **ช่องสำหรับเสียบอุปกรณ์ต่อเพิ่มเติม**
14. **ไฟที่บอร์ดทดลองเตรียมไว้ให้ 5 โวลต์ที่ 1A**
15. **รหัสอีเธอร์เน็ต :** เป็นรหัสของ MAC Address
16. **ชนิดของบอร์ดทดลองและรุ่นของบอร์ดทดลอง**

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.2 อุปกรณ์หลักบนบอร์ดทดลองที่ใช้ในโครงงานนี้

1. ไมโครคอนโทรลเลอร์ PIC16F877 : ใช้ในการประมวลผลและจัดเก็บโปรแกรมประมวลผล
2. ไอซี RTL8019AS : ใช้ในการแปลงข้อมูลในระบบอินเตอร์เน็ตให้ไมโครคอนโทรลเลอร์สามารถประมวลผลได้
3. หน่วยความจำแบบอีอีพรอม : ใช้ในการเก็บข้อมูลที่เป็นเว็บเพ็จ
4. จอแสดงผลแบบ LCD : ใช้ในการแสดงสถานะโหมดการทำงานของบอร์ดทดลอง
5. ช่องสำหรับเสียบสายอนุกรม
6. ช่องสำหรับเสียบสายแลนด์

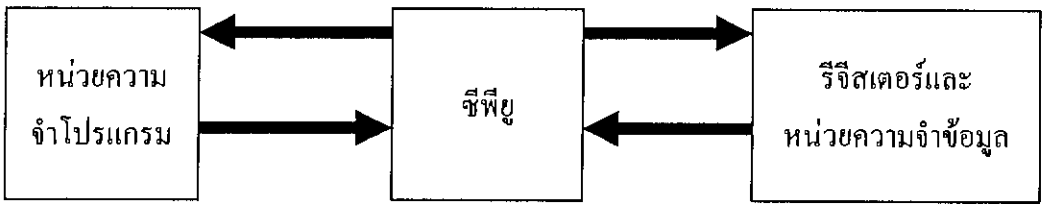
เหตุผลที่เลือกบอร์ดทดลอง PICDEM.net มาประยุกต์ใช้งาน

1. มีอุปกรณ์ที่ต้องการในโครงงานนี้อย่างครบครัน
2. ไอซี RTL8019AS เป็นแบบเซอร์เฟซเมาส์ ไม่สามารถทำการบักกรีได้

3.3 ความรู้เบื้องต้นของไมโครคอนโทรลเลอร์ PIC16F877

รูปที่ 3.2 ไมโครคอนโทรเลอร์ PIC16F877

ไมโครคอนโทรเลอร์ตระกูล PIC มีสถาปัตยกรรมแบบฮาร์วาร์ด (Harvard architecture) กล่าวคือ มีการแยกหน่วยความจำโปรแกรมออกจากหน่วยความจำข้อมูล โดยมีบัสสำหรับติดต่อแยกออกจากกันด้วย ดังแสดงในรูปที่ 3.3



รูปที่ 3.3 แสดงหน่วยความจำโปรแกรมและหน่วยความจำข้อมูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

นอกจากการแยกหน่วยความจำโปรแกรมออกจากหน่วยความจำข้อมูลแล้ว ยังมีการทำงานแบบที่เรียกว่า **ไปป์ไลน์ (Pipeline)** ทำให้สามารถเฟตช์ (fetch) คำสั่งถัดไปในขณะที่ทำการเอ็กซิคิวต์ (execute) คำสั่งปัจจุบันอยู่ ทำให้การทำงานของไมโครคอนโทรลเลอร์มีความไวมากขึ้น ซึ่งเกิดจากการที่สามารถกระทำคำสั่งหนึ่งคำสั่งด้วยสัญญาณนาฬิกาเพียงหนึ่งลูกเท่านั้น (กระบวนการที่เรียกว่าเฟตช์คำสั่งนั้น หมายถึง การที่ไมโครคอนโทรลเลอร์เรียกคำสั่งออกจากหน่วยความจำโปรแกรมแล้วทำการแปลงคำสั่งนั้นให้อยู่ในรูปของเลขฐานสิบหกเพื่อเตรียมที่จะนำไปทำการเอ็กซิคิวต์ ส่วนกระบวนการเอ็กซิคิวต์นั้นเป็นกระบวนการที่กระทำตามคำสั่งนั้นๆให้เกิดผลลัพธ์ตามคำสั่ง)

ซึ่งการกระทำแบบไปป์ไลน์สามารถอธิบายได้ดังนี้

เมื่อเริ่มการกระทำคำสั่งที่ 1 ไมโครคอนโทรลเลอร์จะเฟตช์คำสั่งออกจากหน่วยความจำโปรแกรมที่แอดเดรสที่ n จากนั้นไมโครคอนโทรลเลอร์จะทำการเอ็กซิคิวต์ในไซเคิลต่อมา และที่ไซเคิลของการเอ็กซิคิวต์คำสั่งที่แอดเดรสที่ n นี้ไมโครคอนโทรเลอร์จะเริ่มต้นเฟสช์คำสั่งออกจากแอดเดรสที่ n+1 ทันที เมื่อการเอ็กซิคิวต์คำสั่งที่แอดเดรสที่ n เรียบร้อยแล้ว ไมโครคอนโทรเลอร์ก็จะสามารถเอ็กซิคิวต์คำสั่งที่แอดเดรสที่ n+1 ต่อไปได้เลย โดยไมโครคอนโทรเลอร์ก็จะทำการเฟสช์คำสั่งแอดเดรสที่ n+2 ต่อไป

แต่ถ้าเป็นคำสั่งการกระโดดนั้นจะไม่สามารถทำได้ในหนึ่งไซเคิล แต่ต้องใช้เวลาสองไซเคิล

3.3.1 โครสร้างของไมโครคอนโทรเลอร์ PIC16F877

มีส่วนประกอบหลักๆดังนี้

-ซีพียู
-หน่วยความจำโปรแกรม
-หน่วยความจำข้อมูล
-ส่วนติดต่อพอร์ต
-ส่วนจัดการสัญญาณนาฬิกาของระบบ
-วงจรรีเซตหลัก
-ส่วนควบคุมการตอบสนองอินเตอร์รัปต์
-ส่วนจัดการด้านไฟเลี้ยง

3.3.2 คุณสมบัติทางเทคนิคของ PIC16F877

- ซีพียูเป็นแบบ RISC (Reduced Instruction-Set Computer) มีคำสั่งใช้งาน 35 คำสั่ง
- สามารถกระทำคำสั่งได้เพียงหนึ่งลูกสัญญาณ ยกเว้นคำสั่งกระโดด
- ความถี่สัญญาณนาฬิกาตั้งแต่ไฟตรงถึง 20 MHz
- หน่วยความจำโปรแกรม 8 กิโลเวิร์ด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- หน่วยความจำข้อมูล 368 ไบต์
- หน่วยความจำข้อมูลอีอีพรอม 256 ไบต์
- มีสแต็ก 8 ระดับ
- ตอบสนองต่อแหล่งกำเนิดอินเตอร์รัปต์ 15 แหล่ง
- มีโหมดประหยัดพลังงาน
- สามารถโปรแกรมโดยใช้แรงดัน 5 โวลล์
- มีไทเมอร์ 3 ตัว
- มีวงจรแปลงสัญญาณอะนาลอกเป็นสัญญาณดิจิตอล

3.3.3 การจัดการขาของไมโครคอนโทรลเลอร์ PIC16F877

ไมโครคอนโทรลเลอร์ PIC16F877 สามารถแบ่งขาใช้งานได้เป็น 4 กลุ่มดังรูปที่ 3.4 ซึ่งอธิบายได้ดังนี้

1. กลุ่มขาสัญญาณนาฬิกา

   มี 2 ขา คือ OSC1/CLKIN (ขา 13) และ OSC2/CLKOUT (ขา 14)

2. กลุ่มขาควบคุม

   มี 1 ขา คือ MCLR (ขา 1)

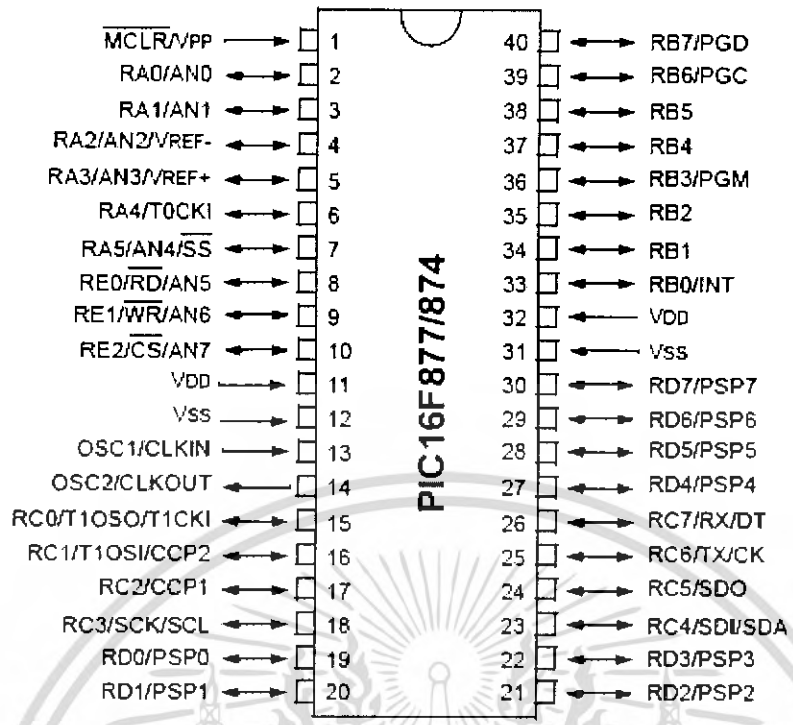
3. กลุ่มขาพอร์ตอินพุตเอาต์พุต

   มีทั้งหมด 33 ขาแบ่งออกเป็นพอร์ต A 6 ขา ได้แก่ ขา RA0-RA5 (ขา 2 ถึง ขา 7) พอร์ต B 8 ขา ได้แก่ ขา RB0-RB7 (ขา 33 ถึงขา 40) พอร์ต C 8 ขา ได้แก่ ขา RC0-RC7 (ขา 15 ถึงขา 26) พอร์ต D 8 ขา ได้แก่ ขา RD0-RD3 (ขา 19 ถึงขา 22) และขา RD4-RD7 (ขา 27 ถึงขา 30) พอร์ต E 3 ขา ได้แก่ ขา RE0-RE7 (ขา 8 ถึงขา 10)

4. กลุ่มขาไฟเลี้ยง

   มี 4 ขาคือ ขาต่อกราวด์ (V_{SS}) (ขา 12 และขา 31) และขาไฟเลี้ยง (V_{DD}) (ขา 11 และขา 32)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.4 ขาของไมโครคอนโทรลเลอร์

3.3.4 การจัดหน่วยความจำของไมโครคอนโทรลเลอร์ PIC 16F877

หน่วยความจำของไมโครคอนโทรลเลอร์ PIC16F877 แบ่งออกเป็น 3 ส่วน

1. หน่วยความจำโปรแกรม

หน่วยความจำโปรแกรมของไมโครคอนโทรลเลอร์ PIC16F877 เป็นแบบแฟลช ซึ่งสามารถที่จะโปรแกรมลงในหน่วยความจำนี้ได้ทั้งในโหมดโปรแกรมและในขณะทำงานอยู่ในโหมดปกติ ไมโครคอนโทรลเลอร์ PIC16F877 มีโปรแกรมเคาน์เตอร์ขนาด 13 บิต ที่สามารถอ้างตำแหน่งหน่วยความจำได้ 8 K ขนาด 14 บิต โดยมีตำแหน่งรีเซตเวคเตอร์อยู่ที่ตำแหน่ง 0000h และที่ตำแหน่งอินเตอร์รัปต์เวคเตอร์อยู่ที่ตำแหน่ง 0004h และมีความลึกของสแต็ก 8 ระดับ

2. หน่วยความจำข้อมูล

หน่วยความจำข้อมูลได้ถูกแบ่งออกเป็น 2 ส่วน คือเป็นรีจีสเตอร์ฟังก์ชันพิเศษ และรีจีสเตอร์ที่ใช้งานทั่งไป การจัดสรรหน่วยความจำข้อมูลทั้งแบบ SFR และGPR จะถูกแบ่งออกเป็น 4 แบงก์ ซึ่งเป็นที่อยู่ของรีจีสเตอร์ทั่วไป และรีจีสเตอร์พิเศษทั้งนี้สามารถทำการเปลี่ยนแปลงแบงก์ได้โดยการเลือกที่บิต RP0 และ RP1

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รีจีสเตอร์ควบคุมของไมโครคอนโทรลเลอร์ PIC16F877

รีจีสเตอร์ STATUS เป็นรีจีสเตอร์ที่ใช้ในการแสดงสถานะทางคณิตศาสตร์ของหน่วยประมวลผล และใช้เป็นตัวกำหนดเลือกแบงก์ของหน่วยความจำข้อมูล การเข้าถึงรีจีสเตอร์ STATUS เพื่ออ่านและเขียนข้อมูลสามารถกระทำได้ด้วยวิธีการเดียวกันกับการอ่านและเขียนรีจีสเตอร์ตัวอื่นๆ

รีจีสเตอร์ OPTION เป็นรีจีสเตอร์ที่ใช้ในการอินเตอร์รัปต์จากสัญญาณภายนอก

รีจีสเตอร์ INTCON เป็นรีจีสเตอร์ที่ใช้ในการเก็บค่าบิตของการอีนาเบิลสัญญาณอินเตอร์รัปต์ที่มีแอดเดรสอยู่ที่ 0X0B มีบทบาทสำคัญมากในการอินเตอร์รัปต์

รีจีสเตอร์โปรแกรมเคาน์เตอร์ PCL และ PCLATH โปรแกรมเคาน์เตอร์เป็นรีจีสเตอร์ที่มีหน้าที่ชี้ตำแหน่งต่อไปของหน่วยความจำโปรแกรมที่ซีพียูจะต้องไปทำ

สแต็ก ในไมโครคอนโทรลเลอร์ PIC16F877ได้จัดสรรสแต็กเพื่อใช้ในการเก็บค่าโปรแกรมเคาน์เตอร์ชั่งคราวไว้ 8 ระดับ หากมีอินเตอร์รัปต์เป็นครั้งที่ 9 ซีพียูจะนำค่าของ PC ในครั้งที่ 9 นี้ไปเก็บไว้ในสแต็กที่เก็บค่า PC ครั้งที่ 1

รีจีสเตอร์ W ในไมโครคอนโทรลเลอร์ PIC16F877 มีรีจีสเตอร์ที่ใช้ในการทำงานหลัก คือ รีจีสเตอร์ W หากเปรียบเทียบกับไมโครคอนโทรลเลอร์เบอร์อื่นๆ รีจีสเตอร์ W เทียบได้กับแอ็กคิวมูเลเตอร์

รีจีสเตอร์ไฟล์ เนื่องจากไมโครคอนโทรลเลอร์ PIC 16F877 มีรีจีสเตอร์ที่เกี่ยวข้องอยู่หลายตัว จึงต้องมีการรวมรีจีสเตอร์ที่ต้องการเข้าถึงไว้ในลักษณะแฟ้มข้อมูล ซึ่งแฟ้มนั้นก็คือ รีจีสเตอร์ไฟล์

การใช้งานอินพุต/เอาต์พุตพอร์ต

ไมโครคอนโทรลเลอร์ PIC16F877 มีพอร์ตสำหรับติดต่อกับอุปกรณ์ภายนอก 3 พอร์ต คือ พอร์ต A มีขนาด 6 บิต พอร์ต B มีขนาด 8 บิต และพอร์ต C มีขนาด 8 บิตขาแต่ละขาของ PIC16F877 สามารถจ่ายกระแสออกได้สูงสุด 25 มิลลิแอมป์ และสามารถรับกระแสสูงสุดต่อขาได้ 20 มิลลิแอมป์ เมื่อใช้ไฟเลี้ยง 5 โวลต์ แต่ถ้าจะนำไปขับ LED จะต้องจำกัดกระแสโดยการต่อตัวต้านทานเข้าไป แต่ถ้าใช้ไฟเลี้ยง 3 โวลต์จะสามารถขับ LED ได้โดยตรง

1. พอร์ต A และรีจีสเตอร์ TRIS A

พอร์ต A เป็นพอร์ตแบบสองทิศทาง มีขนาด 6 บิต มีรีจีสเตอร์กำหนดทิศทางการถ่ายทอดข้อมูลพอร์ต A คือ TRISA เมื่อต้องการใช้งานเป็นพอร์ตอินพุตต้องเขียนข้อมูล 1 ไปยังบิตที่ต้องการ แต่ถ้าต้องการให้เป็นพอร์ตเอาต์พุต ต้องเขียนข้อมูลเป็น 0 สำหรับขาที่ 5 ของพอร์ต A หรือ RA4 / TOCKI มีความแตกต่างจากขาอื่นๆ คือจะมีวงจรชมิดทริกเกอร์ (Schmitt Trigger) ต่อเข้าที่ขาอินพุตไว้สำหรับใช้รับสัญญาณนาฬิกาภายนอก

2. พอร์ต B และรีจีสเตอร์ TRIS B

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

พอร์ต B เป็นพอร์ตแบบสองทิศทาง มีขนาด 8 บิต มีรีจีสเตอร์กำหนดทิศทางการถ่ายทอดข้อมูลของพอร์ต B คือ TRISB เมื่อต้องการใช้งานเป็นพอร์ตอินพุตต้องเขียนข้อมูล 1 ไปยังบิตที่ต้องการ แต่ถ้าต้องการให้เป็นพอร์ตเอาต์พุต ต้องเขียนข้อมูลเป็น 0 แต่ละขาของพอร์ต B สามารถเลือกให้พูลอัพได้โดยทำการเคลียร์บิต RBPU ในรีจีสเตอร์ OPTION และการพูลอัพจะถูกยกเลิกโดยอัตโนมัติเมื่อทำการกำหนดให้พอร์ต B เป็นเอาต์พุต หรือเกิดเพาเวอร์ออนรีเซต

3. พอร์ต C และรีจีสเตอร์ TRIS C

พอร์ต C เป็นพอร์ตแบบสองทิศทาง มีขนาด 8 บิต มีรีจีสเตอร์กำหนดทิศทางการถ่ายทอดข้อมูลพอร์ต C คือ TRISC เมื่อต้องการใช้งานเป็นพอร์ตอินพุตต้องเขียนข้อมูล 1 ไปยังบิตที่ต้องการ แต่ถ้าต้องการให้เป็นพอร์ตเอาต์พุต ต้องเขียนข้อมูลเป็น 0 แต่ละขาของพอร์ด C จะมัลติเพล็กซ์กับฟังก์ชันอุปกรณ์เสริมมากมาย และพอร์ต C นี้ยังมีบัฟเฟอร์ทางอินพุตแบบชมิตทริกเกอร์ เมื่อทำการอีนาเบิลฟังก์ชันอุปกรณ์เสริม ควรระวังในการกำหนดค่าของบิตอินพุตและบิตเอาต์พุต

4. พอร์ต D และพอร์ต E

สำหรับพอร์ต D และพอร์ต E จะไม่อยู่ในตระกูล PIC ที่มีขนาดขา 28 ขา สำหรับพอร์ต D นั้นมีขนาด 8 บิต ซึ่งจะมีตัวกันสัญญาณรบกวนและบัฟเฟอร์อยู่ในตัว โดยที่เราสามารถกำหนดค่าแต่ละบิตของพอร์ตให้เป็นอินพุตหรือเอาต์พุตได้โดยอิสระจากกัน พอร์ต D สามารถทำเป็นพอร์ตขนานได้ด้วย โดยทำการเซต SPI Mode ซึ่งในบัฟเฟอร์ภายในจะกลายเป็นแบบ TTL

สำหรับพอร์ต E นั้นใช้งานเป็นพอร์ตอินพุตเอาต์พุตปกติ และยังเป็นอินพุตอนาลอกของโมดูลแปลงอนาลอกเป็นดิจิตอลและควบคุมอุปกรณ์ภายนอกแบบ PSP ด้วย

3.4 ความรู้เบื้องต้นเกี่ยวกับไอซี RTL8019AS

เป็นไอซีที่ใช้ในการเชื่อมต่อกับระบบอินเตอร์เน็ต มีทั้งหมด 100 ขา แต่สำหรับโครงงานนี้จะใช้เพียงแค่ 16 ขา ในการกำหนดค่าและการรับส่งข้อมูล

3.4.1 คุณสมบัติของไอซี REALTEK RTL8019AS

-มีฟังค์ชัน Plug and play auto detect mode

-สามารถใช้ได้กับ Ethernet II และ IEE802.3 10Base5, 10Base2, 10BaseT

-สามารถรับส่งแบบ Full-Duplex Ethernet

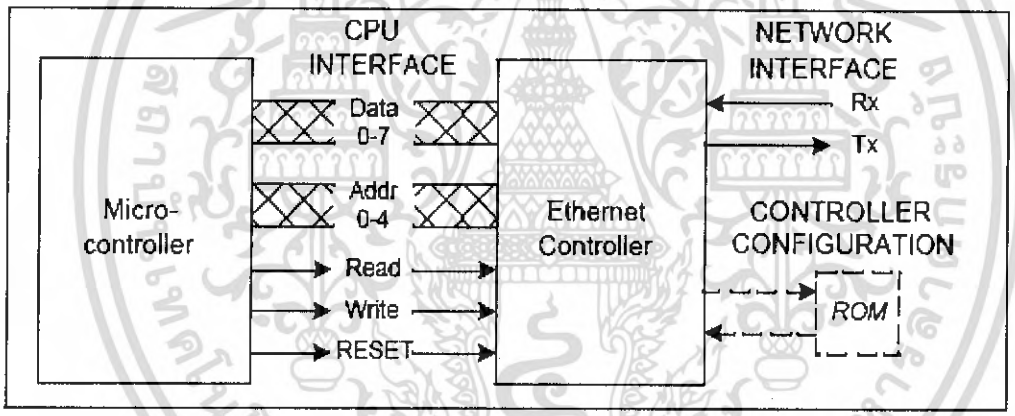
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

-มีโหมดประหยัดพลังงานให้เลือกใช้ 3 โหมด

-มีขาสำหรับต่อหลอดไฟ LED เพื่อแสดงผลการทำงาน

-มีหน่วยความจำภายใน (SRAM) 16 กิโลไบท์

รูปที่ 3.5 ไอซีติดต่อระบบอินเตอร์เน็ต RTL8019AS



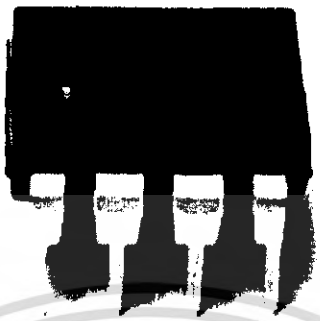
รูปที่ 3.6 การติดต่อกันระหว่างไมโครคอนโทรลเลอร์กับ RTL8019AS

3.5 ความรู้เบื้องต้นเกี่ยวกับหน่วยความจำภายนอกแบบอีอีพรอม

อีอีพรอม หรือ Electrically Erasable Programmable Read-Only Memory หรือ อีเอ็ปรอมคือหน่วยความจำรอม (ROM) ที่ผู้ใช้สามารถลบหรือแก้ไขหรือเขียนซ้ำข้อมูลที่บรรจุอยู่ภายในได้ และสามารถกระทำซ้ำได้หลายครั้ง โดยอาศัยแอพพลิเคชั่นที่ใช้กำลังไฟฟ้าสูงกว่าปกติอีอีพรอมจะต่างจากอีพีรอมตรงที่ไม่จำเป็นต้องถอดออกจากเครื่องคอมพิวเตอร์เพื่อทำการแก้ไขข้อมูล การลบข้อมูลในอีอีพรอมจะเป็นการลบข้อมูลทั้งหมด ไม่สามารถเลือกลบเฉพาะบางส่วนได้ อย่างไรก็ตามมันมีอายุการใช้งานจำกัด ขึ้นอยู่กับจำนวนครั้งในการลบหรือแก้ไขข้อมูล เช่น 10 ครั้งหรือ 100 ครั้ง รูปแบบพิเศษ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ของอีอีพรอม คือหน่วยความจำแบบแฟลช (Flash Memory) ซึ่งใช้ระดับไฟปกติในเครื่องพีซีสำหรับการลบหรือเขียนข้อมูล



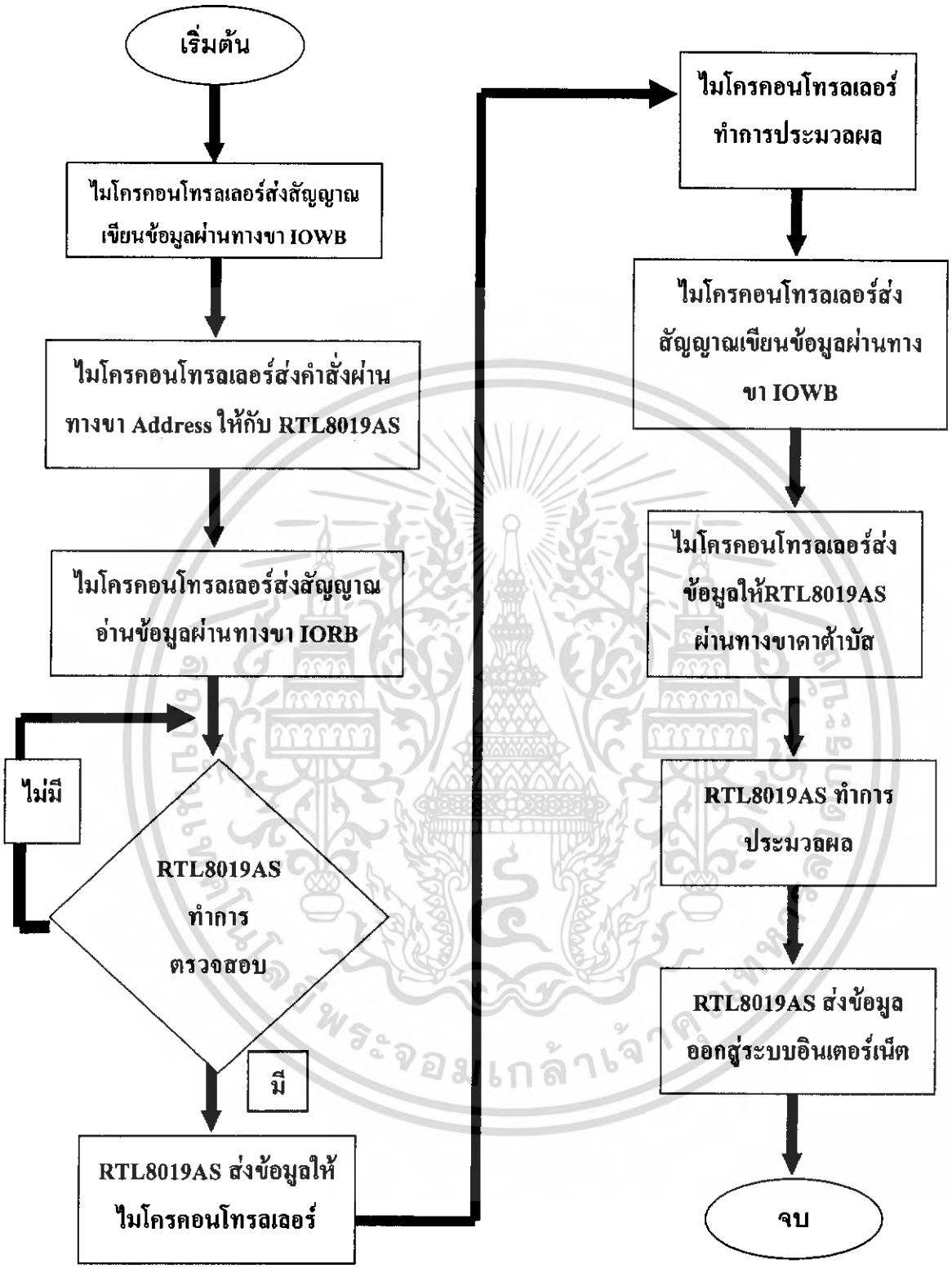
รูปที่ 3.7 EEPROM

3.6 การทำงานของบอร์ดทดลอง PICDEM.net

รูปที่ 3.8 แสดงองค์ประกอบสำคัญของบอร์ดทดลอง PICDEM.net ที่ใช้สร้างเว็บเซิร์ฟเวอร์ ซึ่งจะต้องเขียนโปรแกรมสำหรับไมโครคอนโทรลเลอร์ตามโฟลวชาร์ตในรูปที่ 3.9 ซึ่งอธิบายได้ดังนี้

รูปที่ 3.8 บล็อกไดอะแกรมองค์ประกอบของบอร์ดทดลอง PICDEM.net ที่ใช้สร้างเว็บเซิร์ฟเวอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.9 โฟลว์ชาร์ตการทำงานของไมโครคอนโทรลเลอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เริ่มต้นไมโครคอนโทรลเลอร์จะทำการกำหนดการทำงานให้กับ RTL8019AS โดยส่งผ่านคำสั่งผ่านทางขา Address bus ด้วยการส่งสัญญาณการเขียนข้อมูลผ่านทางขา IOWB ให้กับ RTL8019AS ก่อนแล้วจึงส่งคำสั่งออกไป เมื่อRTL8019AS ถูกกำหนดการทำงานแล้ว ไมโครคอนโทรลเลอร์จะคอยส่งสัญญาณอ่านข้อมูลไปยัง RTL8019AS เพื่อทำการกำหนดค่าการทำงานของ RTL8019AS ถ้า RTL8019AS ไม่ได้รับการติดต่อจากระบบภายนอกมันจะไม่ส่งสัญญาณกลับไป แต่ถ้ามีการเรียกดูเว็บเพ็จเข้ามา ทาง RTL8019AS จะทำการส่งสัญญาณอินเตอร์รัปต์ให้กับไมโครคอนโทรลเลอร์ ทำให้ทราบว่ามีการเรียกดูเว็บเพ็จ จากนั้นไมโครคอนโทรลเลอร์จะส่งสัญญาณขออ่านข้อมูลไปยัง RTL8019AS ผ่านทางขา IORB เมื่อ RTL8019AS ได้รับสัญญาณขออ่านข้อมูล ข้อมูลก็จะถูกส่งผ่านทางขา Data bus ไปยังไมโครคอนโทรลเลอร์ จากนั้นไมโครคอนโทรลเลอร์จะทำการประมวลผลและดึงข้อมูลที่ต้องการส่งมาจากหน่วยความจำภายนอกอีอีพรอม จากนั้นจะส่งสัญญาณขอส่งข้อมูลให้กับ RTL8019AS ผ่านทางขา IOWB และส่งข้อมูลผ่านทาง Data bus ไปยัง RTL8019AS ข้อมูลก็จะถูกแปลงเป็นข้อมูลที่สามารถส่งผ่านทางระบบอินเตอร์เน็ตและถูกส่งผ่านทางสายสัญญาณออกสู่ระบบอินเตอร์เน็ตต่อไป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 4
การทำเว็บเซิร์ฟเวอร์

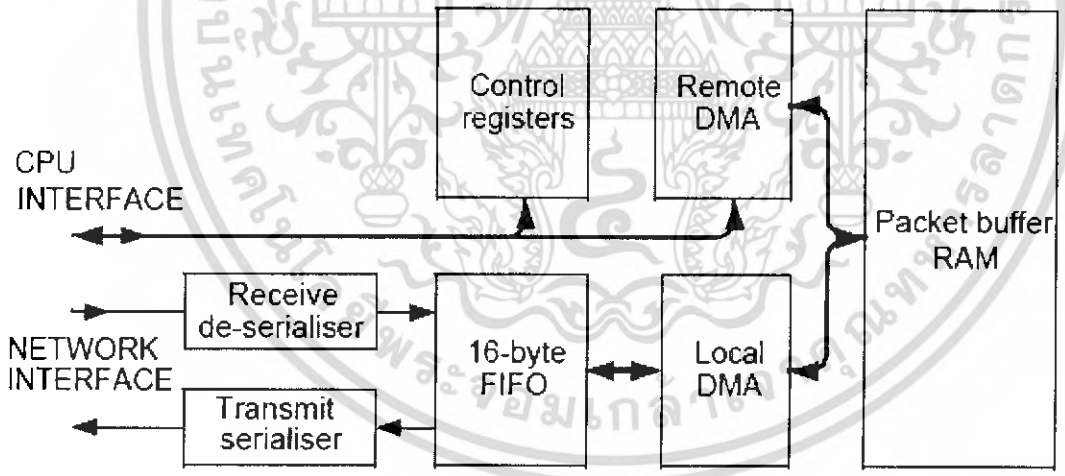
การที่จะนำบอร์ดทดลอง PICDEM.net มาใช้งานเป็นเว็บเซิร์ฟเวอร์ มีขั้นตอนดังนี้

1. เขียนโปรแกรมสำหรับไมโครคอนโทรลเลอร์เพื่อควบคุมและอ่านข้อมูลจาก RTL8019A ซึ่งมีรูปวงจรการเชื่อมต่อ ดังรูปในภาคผนวก ก
2. เขียนเว็บเพ็จเพื่อเก็บไว้ในหน่วยความจำภายนอกอีอีพรอม
3. กำหนดการทำงานของบอร์ดทดลอง

4.1 การเขียนโปรแกรมติดต่อกับระบบอินเตอร์เน็ต

4.1.1 โปรแกรมส่วนอีเธอร์เน็ตไดรฟ์เวอร์

ปกติการเชื่อมต่อกันในระบบอินเตอร์เน็ตเป็นการรับส่งข้อมูลแบบรับหรือส่งอย่างเดียวในเวลาหนึ่ง คือไม่สามารถส่งและรับในเวลาเดียวกันได้ ซึ่งเป็นรูปแบบที่เรียกว่า Network Interface Controller (NIC)



รูปที่ 4.1 โครงสร้างของ Network Interface Controller

โปรแกรมการกำหนดค่าเบื้องต้นของ **Network Interface Controller**

```
#BIT        NIC_IOW_    =PORTE.1
#BIT        NIC_IOR_    =PORTE.0
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
#BYTE      NIC_DATA     =PORTE.1
#define    DATA_TO_NIC      set_tris_d (ALL_OUT) ;
#define    DATA_FROM_NIC    set_tris_d (ALL_IN) ;
outnic (CMDR,  0x21) ;
delay_ms (2) ;
outnic (DCR,  DCRVAL) ;
outnic (RBCR0,  0) ;
outnic (RBCR1,  0) ;
outnic (RCR,  0x20) ;
outnic (TCR,  0x02) ;

/*ฟังก์ชันในการรับค่าเป็นไบท์จาก NIC รีจีสเตอร์*/
BYTE innic (int reg)
{
        BYTE  b ;
        DATA_FROM_NIC ;
        NIC_ADDR  =  reg ;
        NIC_IOR_   =  0 ;
        delay_cycles (1) ;
        b  =  NIC_DATA ;
        NIC_IOR_  =  1 ;
        return (b) ;
}
/*ฟังก์ชันในการส่งค่าเป็นไบท์ไป NIC รีจีสเตอร์*/
void  outnic  (int reg,  int  b)
{
        NIC_ADDR  =  reg ;
        DATA_DATA  =  b ;
        DATA_TO_NIC ;
        NIC_IOW_   =  0 ;
        delay_cycles (1) ;
        DATA_FROM_NIC ; }
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โดยปกติ CPU จะส่งข้อมูลที่เป็นอินเตอร์เน็ตแพ็กเก็ตไปที่แรมก่อนจะประมวลผลมัน แต่ 1 แพ็กเก็ตมีขนาดใหญ่มากเมื่อเทียบกับแรมของไมโครคอนโทรลเลอร์ ดังนั้นจำเป็นจะต้องแบ่งแพ็กเก็ตเป็นส่วนย่อยๆก่อนโดยใช้ remote DMA ด้วยการกำหนดค่าของแพ็กเก็ตบัฟเฟอร์แอดเดรสและไบท์เคาร์ท

```
void setnic_addr (WORD addr)
{
        outnic (ISR, 0x40) ;
        outnic (RSAR0 , (BYTE) addr) ;
        outnic (RSAR1 , addr >> 8) ;
}
void getnic_data (BYTE * data, int len)
{
        outnic (RBCR0 , len) ;
        outnic (RBCR1 , 0) ;
        outnic (CMDR , 0x0a) ;
        while (len--) ;
        {
            b = innic (DATAPORT) ;
            * data++ = b ;
}
void putnic_data (BYTE *data , int len)
{
        outnic (ISR, 0x40) ;
        outnic (RBCR0 , len) ;
        outnic (RBCR1 , 0) ;
        outnic (CMDR , 0x12) ;
        while (lin--)
            outnic (DATAPORT, *data++) ;
        len = 255 ;
        while (len && (innic (ISR) & 0x40 ) = = 0 )
            len --
}
```

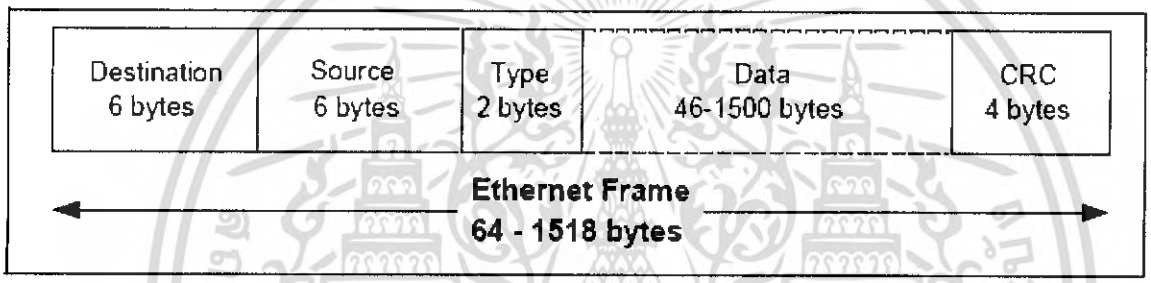
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.1.2 โปรแกรมการรับแพ็กเก็ต

การจะรับแพ็กเก็ตเข้ามาต้องทำสิ่งเหล่านี้

1. ตรวจสอบ over-run ของแพ็กเก็ตบัฟเฟอร์
2. ตรวจสอบจำนวนแพ็กเก็ตที่เข้ามา
3. ตรวจสอบสถานะผิดพลาดของแพ็กเก็ต
4. กำหนดแอดเดรสของแพ็กเก็ตในบัฟเฟอร์
5. กำหนดความยาวของแพ็กเก็ต
6. ทำพื้นที่ของแรมให้ว่างพอสำหรับการรับแพ็กเก็ต

การจะรับข้อมูลแพ็กเก็ตเข้ามานั้นจำเป็นต้องทำการวิเคราะห์แพ็กเก็ตที่เข้ามาก่อน โดยทุกๆ แพ็กเก็ตต้องมีส่วนหัวของแพ็กเก็ตที่เรียกว่า Media Access and Control (MAC) address ดังรูปที่ 4.2



รูปที่ 4.2 อีเธอร์เน็ตเฟลม

เราจึงต้องกำหนดโครงสร้างของอีเทอร์เน็ดเฮดเดอร์เพื่อที่จะสามารถเข้าถึง Media Access and Control ได้สะดวก

```
#define  MACLEN 6
typedef struct  {
        BYTE  dest [MACLEN] ;
        BYTE  srce [MACLEN] ;
        WORD  pcol ;
} ETHERHEADER ;
```

ขนาดทั้งหมดของอีเธอร์เน็ตเฟรมอยู่ระหว่าง 64-1518 ไบท์ ดังนั้นค่าของข้อมูลที่แท้จริงอยู่ระหว่าง 46-1500 ไบท์ แต่ถ้ามีข้อมูลน้อยกว่า 64 ไบท์ ข้อมูลต้องถูกขยายทำให้พอดี และเมื่อรับแพ็กเก็ตเข้ามา NIC จะทำการเพิ่มส่วนหัวให้กับอีเธอร์เน็ดเฟรมอีก เรียกว่า NICHEADER ซึ่งในส่วนของ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

NICHEADER นี้จะบอกถึงความยาวของแพ็กเก็ตและค่าความผิดพลาดของแพ็กเก็ต ซึ่งโครงสร้างที่ใช้อ้างถึง NICHEADER เขียนได้ดังนี้

```
typedef struct {
        BYTE stat ;
        BYTE next ;
        WORD len ;
} NICHEADER ;
```

และเพื่อให้ง่ายต่อการเข้าถึงเฮดเดอร์ ต้องมีโปรแกรมคัดลองเฮดเดอร์ไปที่ local RAM

```
typedef struct {
        NICHEADER nic ;
        ETHERHEAD eth ;
} NICETHERHEADER ;
.....
NICETHERHEADER nicin ;
WORD get_ether ()
{
        WORD len = 0 , curr ;
        .....
        getnic_data ((BYTE * ) & nicin , sizeof (nicin)) ;
        len = nicin.nic.len ;
        .....
        len -= MACLEN + MACLEN + 2 + CRCLEN ;
        .....
        return (len) ;
}
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.1.3 โปรแกรมการวิเคราะห์แพ็กเก็ต

แพ็กเก็ตที่เข้ามาจะต้องถูกวิเคราะห์โดยเอามาเก็บไว้ที่บัฟเฟอร์แรม ซึ่งที่นี้แพ็กเก็ตจะถูกดึงออกมาวิเคราะห์ทีละไบท์ และตรวจสอบความถูกต้องโดยโปรโตคอลไอพี

โปรแกรมที่ดึงแพ็กเก็ตออกมาทีละไบท์

```
BYTE getch_net (void)
{
        BYTE b = 0 ;
        atend  = rxout >= rxin;
        if (! attend )
        {
                b = getnic_byte () ;
                rxout ++ ;
                check_byte (b) ;
        }
        return (b) ;
}
BYTE ungot_byte ;
BOOL ungot ;

BOOL get_byte (BYTE &b)
{
        if (ungot)
                b = ungot_byte ;
        eles
                b = getch_net () ;
        ungot = 0 ;
        return (!atend) ;
}
void unget_byte (BYTE &b)
{
        ungot_byte = b ;
        ungot = 1 ; }
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

แพ็กเก็ตจะถูกวิเคราะห์โดยการใช้งานคำสั่ง get_ , match_ และ skip_

```
BOOL get_word (WORD &w)
{
        BYTE hi , lo ;
        hi = getch_net () ;
        lo = getch_net () ;
        w = ((WORD) hi << 8) | (WORD) lo ;
        return (!atend) ;
}
BOOL match_word (WORD w)
{
        WORD inw ;
        return (get_word (inw) && inw = = w) ;
}
BOOL skip_word (void)
{
        getch_net () ;
        getch_net () ;
        return (!atend) ;
}
```

4.1.4 โปรแกรมการส่งแพ็กเก็ต

การส่งแพ็กเก็ตสามารถทำได้ง่ายมากกว่าการรับมากโดยมีขั้นตอนดังนี้

1. เขียนอีเธอร์เน็ตเฮดเดอร์ (แอดเดรสปลายทางและแอดเดรสต้นทาง และโปรโตคอล) ลงไปในแพ็กเก็ตบัฟเฟอร์
2. เขียนแพ็กเก็ตข้อมูลลงในบัฟเฟอร์
3. กำหนดความยาวของแพ็กเก็ตลงใน NIC รีจีสเตอร์ และต้องไม่น้อยกว่า 64 ไบท์
4. เริ่มการทำงานของ NIC

เมื่อการทำงานของ NIC มันจะทำการส่งโดยอัตโนมัติ ถ้าเกิดการชนของข้อมูล NIC จะทำการส่งใหม่อีกครั้งโดยอัตโนมัติ แต่การส่งจะทำไม่ได้ถ้ามีการส่งหนาแน่นมากเกินไป

โปรแกรมการเติมแพ็กเก็ตข้อมูลให้ไม่น้อยกว่า 64 ไบท์

```
#define TXBUFFLEN 64
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
BYTE txbuff [TXBYFFLEN] ;
int txin , txout ;
void putch_net (BYTE b )
{
        if (txin < TXBYFFLEN)
        txbuff [txin ++ ] = b ;
        check_byte (b) ;
}
```

โปรแกรมการการส่งแพ็กเก็ต

```
void put_ether (void *data, WORD dlen )
{
        outnic (ISR, 0x0a) ;
        setnic_addr (TXSTART << 8) ;
        putnic_data (nicin.eth.srce, MACLEN) ;
        putmic_data (myeth , MACLEN) ;
        swapw (nicin.eth.pcol) ;
        putnic_data (&nicin.eth.pcol , 2 ) ;
        putnic_data (data , dlen) ;
}
```

4.1.5 โปรแกรมการติดต่อกับจอแสดงผล

อันดับแรกต้องกำหนดพอร์ตการติดต่อระหว่างไมโครคอนโทรลเลอร์และจอแสดงผล

```
struct {
        BYTE data : 4 ;
        BYTE regsel : 1 ;
        BYTE read : 1 ;
} LCD_PORT ;
#BYTE LCD_PORT          = 8
#BIT      LCD_E          = PORTA.5
#define LCD_RD          LCD_PORT.read
#define LCD_RS          LCD_PORT.regsel
#define LCD_DATA        LCD_PORT.data
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ทำการกำหนดการทำงานเบื้องต้นให้จอแสดงผล โดยให้เป็นแบบ 16 ตัวอักษร 2 บรรทัด

```
void init_lcd (void)
{
        int i ;
        LCD_E = 0 ;
        DATA_FROM_LCD ;
        LCD_RD = LCD_RS = 0 ;
        delay_ms (15) ;
        for (i = 0 ; i < 4 ; i ++)
        {       lcd_nybble (0x03) ;
                delay_ms (5) ;
        }
        lcd_cmd (0x28) ;
        lcd_cmd (0x06) ;
        lcd_cmd (0x0e) ;
        lcd_cmd (0x01) ;
}
void lcd_byte (BYTE &b)
{
        lcd_nybble (b >> 4) ;
        lcd_nybble (b) ;
        DATA_FROM_LCD ;
        delay_us (40) ;
}
void lcd_nybble (BYTE b)
{
        DATA_TO_LCD ;
        LCD_E = 1 ;
        LCD_DATA = b ;
        LCD_E = 0 ;
}
void lcd_cmd (BYTE b)
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
{
        DATA_TO_LCD ;
        LCD_RD = LCD_RS =0 ;
        lcd_byte (b) ;
        if ((b & 0xfc) = = 0)
                delay_ms (2) ;
void lcd_gotoxy (BYTE x , BYTE y)
{
        if (y ! = 1) ;
            x += LCD_LINE2 ;
        lcd_cmd (LCD_SETPOS - 1 + x) ;
}
void lcd_char (BYTE b)
{
        DATA_TO_LCD ;
        LCD_RD = 0 ;
        LCD_RS = 1 ;
        lcd_byte (b) ;
}
```

4.1.6 โปรโตคอล ARP

ทุกๆข้อความของอีเธอร์เน็ตแพ็กเก็ตจะมี 6 ไบท์ที่เป็น MAC address และโดยปกติการส่ง MAC address จะส่งเป็นเลขไอพีที่มีจุดขั้นกลางเช่น 212.34.68.2 เครื่องที่ใช้บริการต้องใช้โปรโตคอล ARP ในการแปลงหมายเลขไอพีให้เป็นค่า MAC address การวิเคราะห์ค่าไอพีที่เข้ามาจะใช้ชุดคำสั่ง match_ และ skip_ จากนั้นเมื่อแปลงข้อมูลเสร็จแล้วจะใช้ชุดคำสั่ง put_ ในการส่งข้อมูลไปที่ NIC แพ็กเก็ตบัฟเฟอร์ด้วยคำสั่ง xmit_ether ()

```
BOOL arp_recv (void)
{
        BOOL ret = 0 ;
        if (match_byte (0x00) && match_byte (0x01) &&
            match_byte (0x08) && match_byte (0x00) &&
            match_byte (6) && match_byte (4) &&
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
            match_word (ARPREQ) &&
            skip_lword () && skip_word () &&
            get_lword (remip.l) &&
            skip_lword () && skip_word () &&
            match_lword (myip.l))
        {
            ret = 1 ;
            txin = 0 ;
            put_word (0x0001) ;
            put_word (0x0800) ;
            put_byte (6) ;
            put_byte (4) ;
            put_word (ARPRESP) ;
            put_data (myeth, MACLEN) ;
            put_lword (myip. l) ;
            put_data (nicin.eth.srce, MACLEN) ;
            put_lword (remip. l) ;
            put_ether (txbuff, txin) ;
            xmit_ether (txin) ;
        }
        return (ret) ;
}
```

4.1.7 โปรโตคอล IP

อินเตอร์เน็ตโปรโตคอลไอพีใช้ในการตรวจสอบข้อมูลที่เข้ามาว่าเป็นโปรโตคอลชนิดอะไร ในโครงงานนี้ใช้โปรโตคอลอยู่ 2 ชนิด คือ ICMP และ TCP ถ้าเป็นชนิดอื่นจะไม่สามารถรับข้อมูลได้

```
BOOL ip_recv (viod)
{
        BYTE b , hi , lo ;
        int n = 0 ;
        BOOL ret = 1 ;
        checkflag = 0 ;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        checkhi = checklo = 0 ;
        if (match_byte (0x45) && skip_byte () &&
            get_word (iplen) && skip_word () &&
            skip_word () && skip_byte() &&
            get_byte (ipcol) && skip) word () &&
            get_lword (remip. l) && match_lword (myip.l) &&
            checkhi = = 0xff && checklo = = 0xff)
        {
            if (ipcol = = PICMP)
                icmp_recv () ;
            else if (ipcol = = PTCP)
                tcp_recv () ;
            else
                discard_data () ;
        }
        else
            discard_data () ;
        return (ret) ;
}
```

4.1.8 โปรโตคอล ICMP

โปรโตคอล ICMP เป็นเครื่องมือที่สำคัญที่ใช้ในการตรวจสอบสถานะของการรับส่งข้อมูลหรือที่รู้จักกันในชื่อ ping ในการใช้โปรโตคอล ICMP เครื่องที่เป็นเซิร์ฟเวอร์ต้องตอบกลับมาด้วยข้อมูลที่เหมือนเดิม

```
BOOL icmp_recv (void)
{
        BOOL ret = 0 ;
        WORD csum ;
        rpdlen = 0 ;
        if (match_byte (8) && match_byte (0) && get_word (csum))
        {
                while (skip_byte ())
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                rpdlen++ ;
                ret = (checkhi = = 0xff) && (checklo = = 0xff) ;
                if (ret && rpdlen < = MAXPING_LEN)
                {
                        DEBUG_PUTC ('>') ;
                        checkhi = chccklo = 0 ;
                        put_ip () ;
                        put_word (0) ;
                        csum += 0x0800;
                        if (csum < 0x800)
                                csum ++ ;
                        put_word (csum) ;
                        put_ether (txbuff , txin) ;
                        copy_rx_tx (txin, IPHDR_LEN + ICMPHDR_LEN, rpdlen) ;
                        xmit_ether (IPHDR_LEN + ICMPHDR_LEN + rpdlen) ;
                }
        }
        return (ret) ;
}
void copy_rx_tx (BYTE dest , BYTE srce , BYTE len)
{
        BYTE b ;
        outnic (ISR , 0x40) ;
        dest + = sizeof (ETHERHEADER) ;
        srce + = sizeof (NICETHERHEADER) ;
        while (len--)
        {
                outnic (RSAR0 , srce) ;
                outnic (RSAR1 , curr_rx_page) ;
                b = getnic_byte () ;
                outnic (RSAR0 , dest) ;
                outnic (RSAR1 , TXSTART) ;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                putnic_byte (b) ;
                srce ++ ;
                dest ++ ;
            }
        }
```

4.1.9 โปรโตคอล TCP

เมื่อโปรโตคอล IP วิเคราะห์ข้อมูลแพ็กเก็ตที่เข้ามาว่าเป็นแบบโปรโตคอล TCP แล้วโปรโตคอล TCP จะรับข้อมูลโดยคำสั่ง tcp_recv เมื่อรับข้อมูลมาเสร็จแล้วจะดำเนินการแบบโปรโตคอล TCP ด้วยคำสั่ง tcp_handler (void)

```
BOOL  tcp_recv (viod)
{
        int  hlen ,  n ;
        BOOL  ret = 0 ;
        WORD  addr ;
        checkhi  =  checklo  =  0 ;
        if  (get_word (remport)  &&  get_word (locport)  &&
          get_lword (rseq . l)  &&  get_lword (rack . l)  &&
          get_byte (hlen)  &&  get_byte (rflags)  &&
          skip_word ()  &&  skip_lword ())
        {
          iplen  - =  IPHDR_LEN ;
          check_word (iplen) ;
          check_lword (myip . l) ;
          check_lword (remip . l) ;
          check_byte (0) ;
          check_byte (PTCP) ;
          rxout  =  (hlen >> 2)  +  IPHDR_LEN ;
          rpdlen  =  iplen  -  rxout  +  IPHDR_LEN ;
          addr  =  getnic_addr () ;
          check_rxbytes (IPHDR_LEN + TCPHDR_LEN ,  iplen – TCPHDR_LEN) ;
          setnic_addr (addr) ;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        ret = (checkhi = = 0xff) && (checklo = = 0xff) ;
        if (ret)
            tcp)handler () ;
    }
    return (ret) ;
}
void tcp_handler (void)
{
    BOOL tx = 1 ;
    tpdlen = tpxdlen = 0 ;
    d_checkhi = d_checklo = 0 ;
    checkflag = 0 ;
    tflags = TACK ;
    if (rflags & TRST)
        tx = 0 ;
    else if (rflags & TSYN)
    {
      add_lword (rseq . l , l) ;
      if (locport = = DAYPORT || locport = = HTTPORT)
      {
        rack . w [0] = 0xffff ;
        rack . w [1] = concount ++ ;
        tflags = TSYN + TACK ;
      }
      else
        tflags = TRST + TACK ;
    }
    else if (rflags & TFIN)
        add_lword (rseq . l , rpdlen + l ) ;
    else if (rflags & TACK)
    {
        if (rpdlen)
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
            add_lword (rseq . l , rpdlen) ;
        else
            tx = 0 ;
        if (locport == HTTPORT && rpdlen)
        {
            http_recv () ;
            tx = 0 ;
        }
        else if (locport == DAYPORT && rack . w [0] == 0 )
        {
            daytime_handler () ;
            tx = 1 ;
        }
    }
    if (tx)
        tcp_xmit () ;
}
/* การส่งค่าข้อมูลแพ็กเก็ตที่เป็นโปรโตคอลแบบ TCP*/
void put_tcp (void)
{
    WORD len ;
    checkflag = 0 ;
    checkhi = d_checkhi ;
    checklo = d_checklo ;
    put_word (locport) ;
    put_word (remport) ;
    put_lword (rack . l) ;
    put_lword (rseq . l) ;
    put_byte (tflags & TSYN ? TCPSYN_LEN*4 : TCPHDR_LEN*4) ;
    put_byte (tflags) ;
    put_byte (0x0b) ;
    put_byte (0xb8) ;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

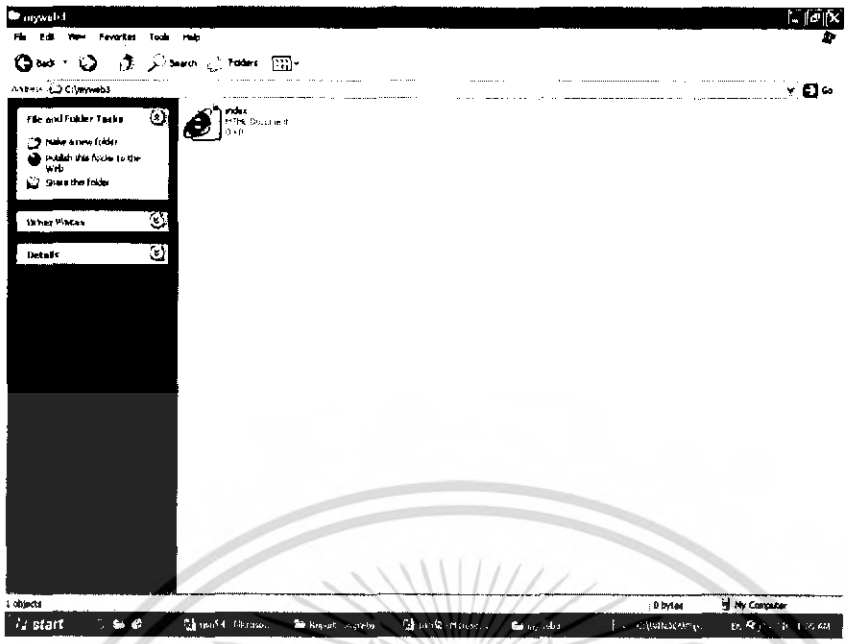
```
        len = tpdlen + tpxdlen +TCPHDR_LEN ;
        if (tflags & TSYN)
        {
                txin += 4 ;
                len += TCPOPT_LEN ;
                put_byte (2) ;
                put_byte (4) ;
                put_word (TCP_MSS) ;
                txin - = TCPOPT_LEN + 4 ;
        }
        check_lword (myip . l) ;
        check_lword (remip . l) ;
        check_byte (0) ;
        check_byte (PTCP) ;
        check_word (len) ;
        put_byte (~checkhi) ;
        put_byte (~checklo) ;
        put_word (0) ;
        if (tflags & TSYN)
          txin + = TCPOPT_LEN ;
}
```

4.2 เขียนเว็บเพ็จเพื่อเก็บไว้ในหน่วยความจำภายนอกอีอีพรอม

นำไฟล์เว็บเพจที่เขียนขึ้นในหัวข้อที่ 2.13.2 มาทำการแปลงโดยใช้โปรแกรมที่ชื่อว่า WEBROM.EXE ให้เป็นไฟล์ที่มีนามสกุลที่เป็น .ROM ซึ่งจะแสดงขั้นตอนได้ดังนี้

1. นำไฟล์ที่ได้มาเปลี่ยนชื่อให้เป็น index.htm ซึ่งเป็นรูปแบบพื้นฐานของโปรแกรม WEBROM.EXE
2. นำไฟล์และรูปที่เกี่ยวข้องกับไฟล์นั้นไปเก็บไว้ใน Folder เดียวกัน เช่น c:\myweb3 เป็นต้น แสดงได้ดังรูปที่ 4.3

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

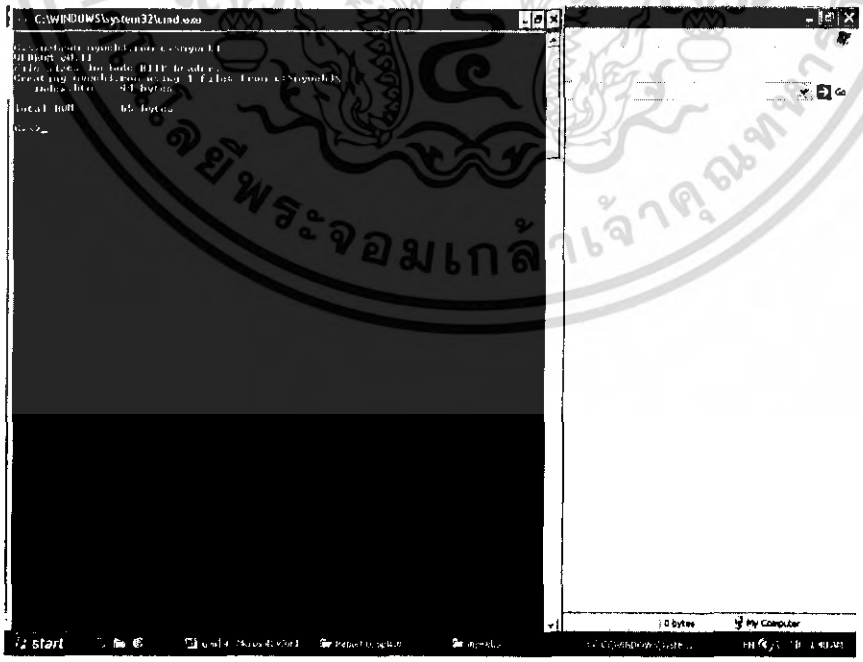


รูปที่ 4.3 แสดงที่อยู่ของไฟล์ตัวอย่างที่จะนำมาแปลงให้เป็น .ROM

3. ทำการรันโปรแกรมตามรูปแบบดังรูปด้านล่าง โดยมีรูปแบบคือ

   webrom xxxx.rom <ที่อยู่ของไฟล์ index.htm >

   เมื่อพิมพ์เสร็จแล้วก็กดเอนเตอร์(Enter) โปรแกรมก็จะทำการแปลงไฟล์นั้นให้กลายเป็น .ROM



รูปที่ 4.4 แสดงตัวอย่างการรันโปรแกรม WEBROM.EXE

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

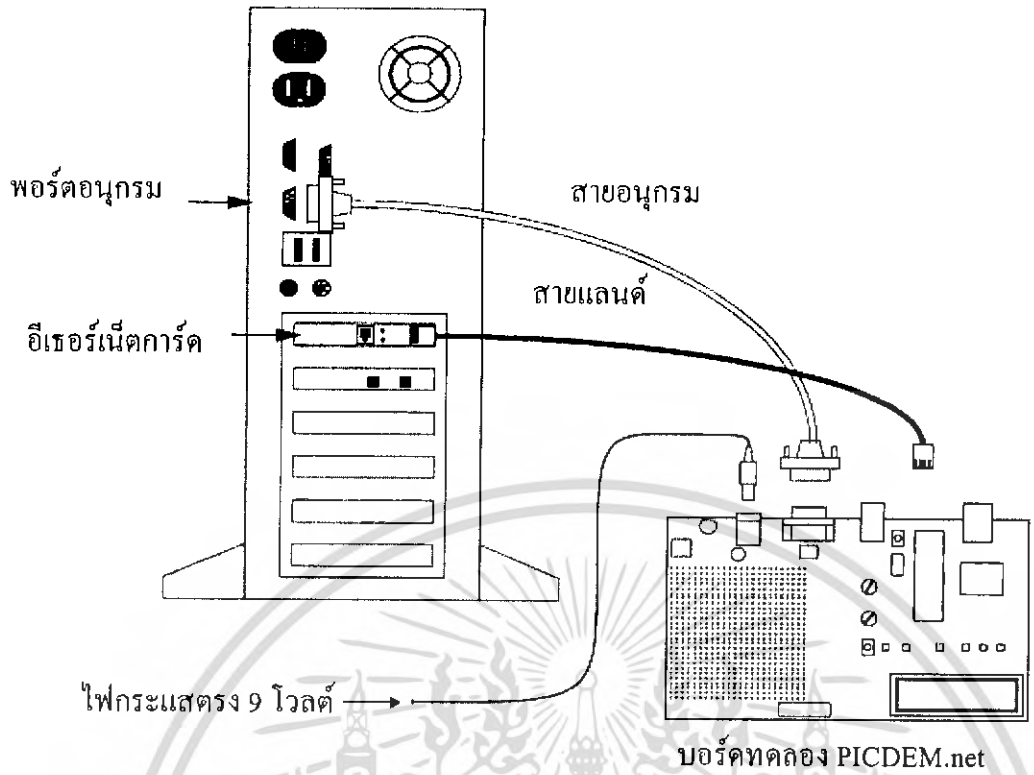
4. เราจะได้ไฟล์ที่ชื่อว่า myweb3.rom ที่พร้อมจะนำไปใช้แล้ว

4.3 กำหนดการทำงานของบอร์ดทดลอง มีขั้นตอนดังนี้

1. ทำการต่อบอร์ดทดลองตามรูปที่ 4.3
2. ใช้เครื่องคอมพิวเตอร์ที่ทำหน้าที่เป็นไคลเอ็นท์เปิดโปรแกรม Hyper Terminal
3. ที่บอร์ดทดลองให้กดปุ่ม RB5 ค้างไว้ แล้วกดปุ่ม MCLR จากนั้นให้ปล่อยปุ่ม MCLR และปล่อยปุ่ม RB5 ตามลำดับ
4. ที่จอ LCD จะแสดงคำว่า

5. ที่โปรแกรม Hyper Terminal จะให้ผู้ใช้งานใส่รหัสอนุกรมของบอร์ดทดลอง โดยขึ้นคำว่า Put Serial โดยรหัสอนุกรมของบอร์ดทดลองจะติดอยู่ที่บอร์ดทดลอง
6. ทำการใส่รหัส Serial number แล้วกดปุ่ม ENTER
7. ที่โปรแกรม Hyper Terminal จะแสดงคำว่า Put IP
8. ทำการใส่รหัส IP Address แล้วกดปุ่ม ENTER (หมายเลขไอพีผู้ใช้ต้องทำการกำหนดขึ้นเอง โดย 3 ตำแหน่งแรกเป็นไอพีของวง LAN ส่วนตำแหน่งสุดท้ายผู้ใช้ต้องกำหนดขึ้นเองโดยไม่ให้ซ้ำกันในวง LAN)
9. จากนั้นที่จอ LCD จะแสดงเลข IP Address
10. ที่โปรแกรม Hyper Terminal จะแสดงคำว่า xmodem (ถ้าต้องการจะโหลดเว็บเพ็จลงในอีอีพรอมให้ทำในขั้นตอนที่ 11 แต่ถ้าไม่ต้องการโหลดเว็บเพ็จให้กดปุ่ม MCLR)
11. ให้เข้าไปที่ Transfer > Send file
12. ทำการเลือกเว็บเพ็จที่ต้องการ จากนั้นคลิกที่ปุ่ม Send

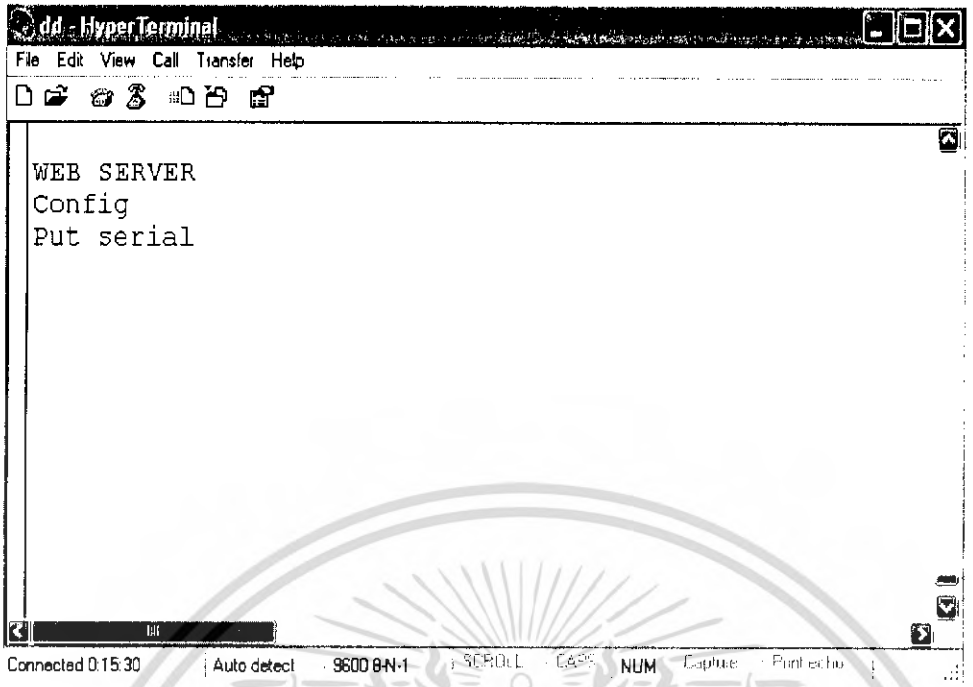
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



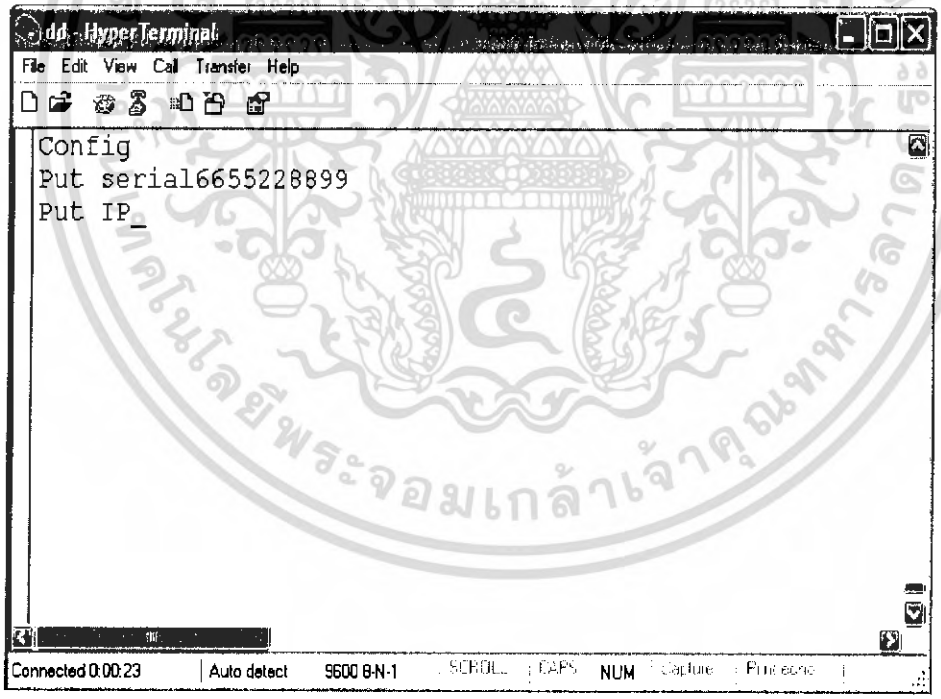
รูปที่ 4.5 การเชื่อมต่อบอร์ดทดลองกับเครื่องคอมพิวเตอร์

รูปที่ 4.6 หน้าต่างโปรแกรม Hyper Terminal

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

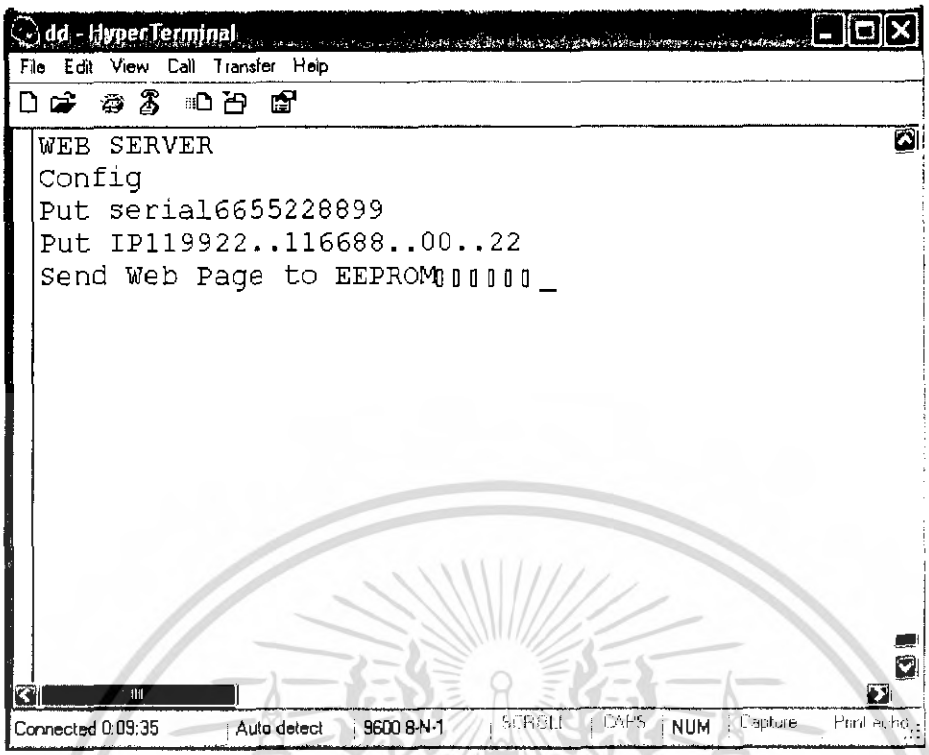


รูปที่ 4.7 หน้าต่างถามเลขอนุกรมของบอร์ดทดลอง

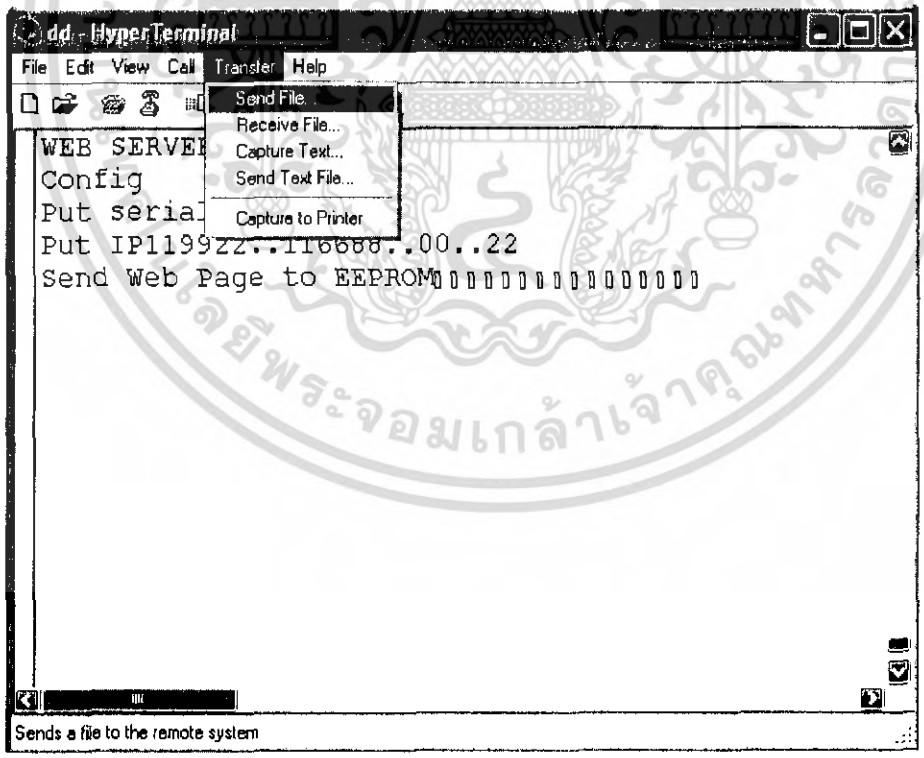


รูปที่ 4.8 หน้าต่างถามเลขไอพีของบอร์ดทดลอง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

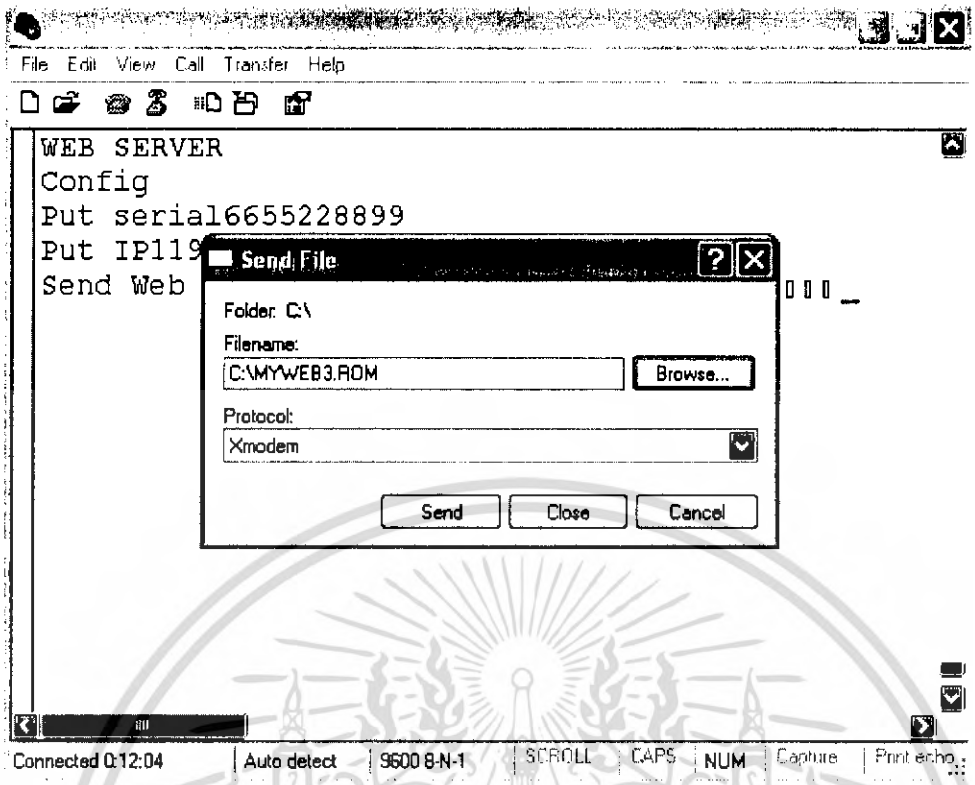


รูปที่ 4.9 แสดงหน้าต่างการรอโหลดโปรแกรมลงบอร์ดทดลอง

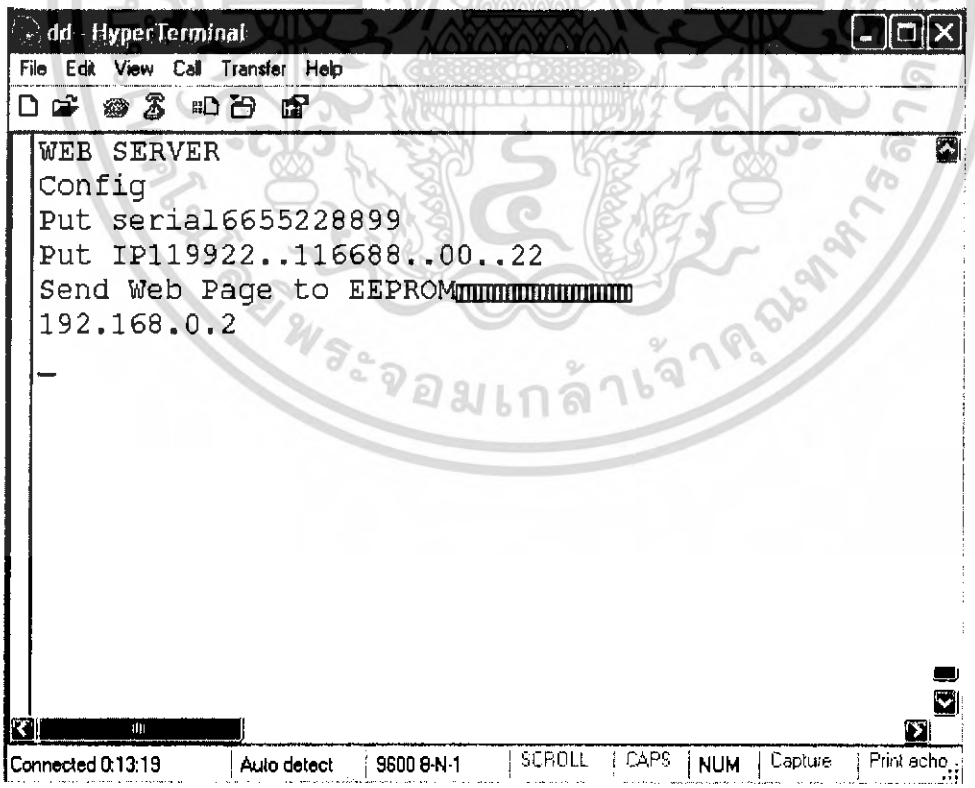


รูปที่ 4.10 แสดงการโหลดเว็บเพ็จ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.11 แสดงการเลือกเว็บเพ็จที่ต้องการจะโหลดลงบอร์ดทดลอง

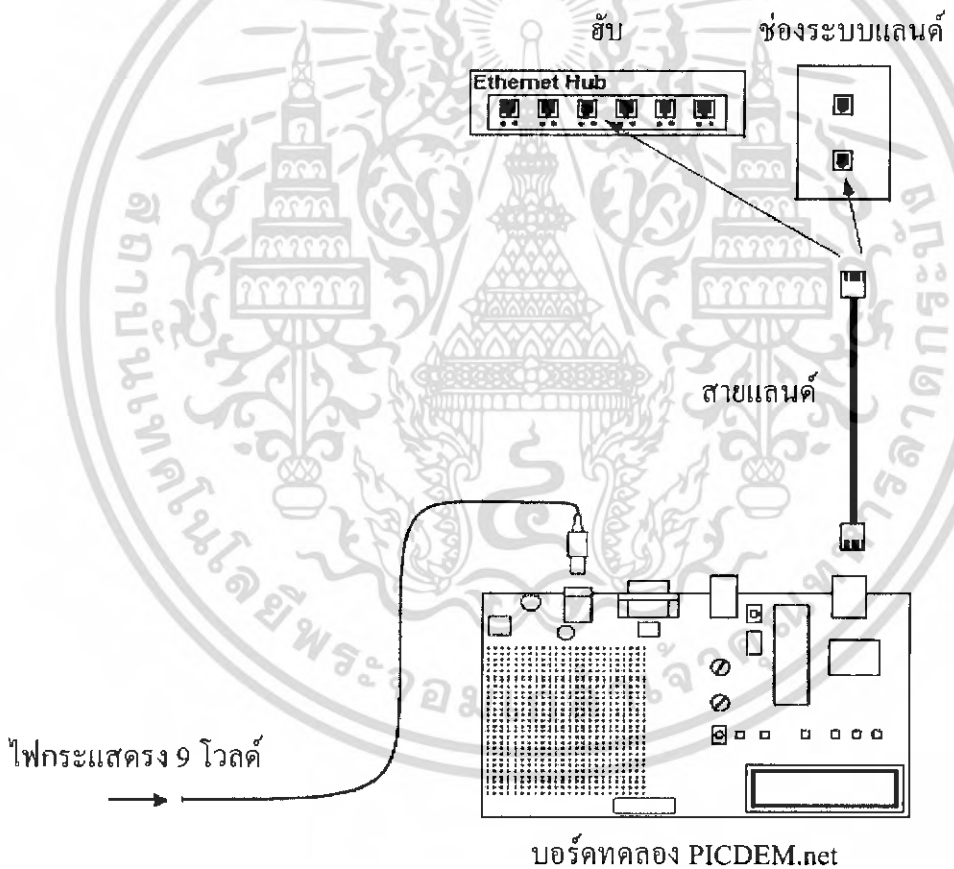


รูปที่ 4.12 แสดงหน้าต่างเมื่อโหลดเว็บเพ็จเสร็จ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

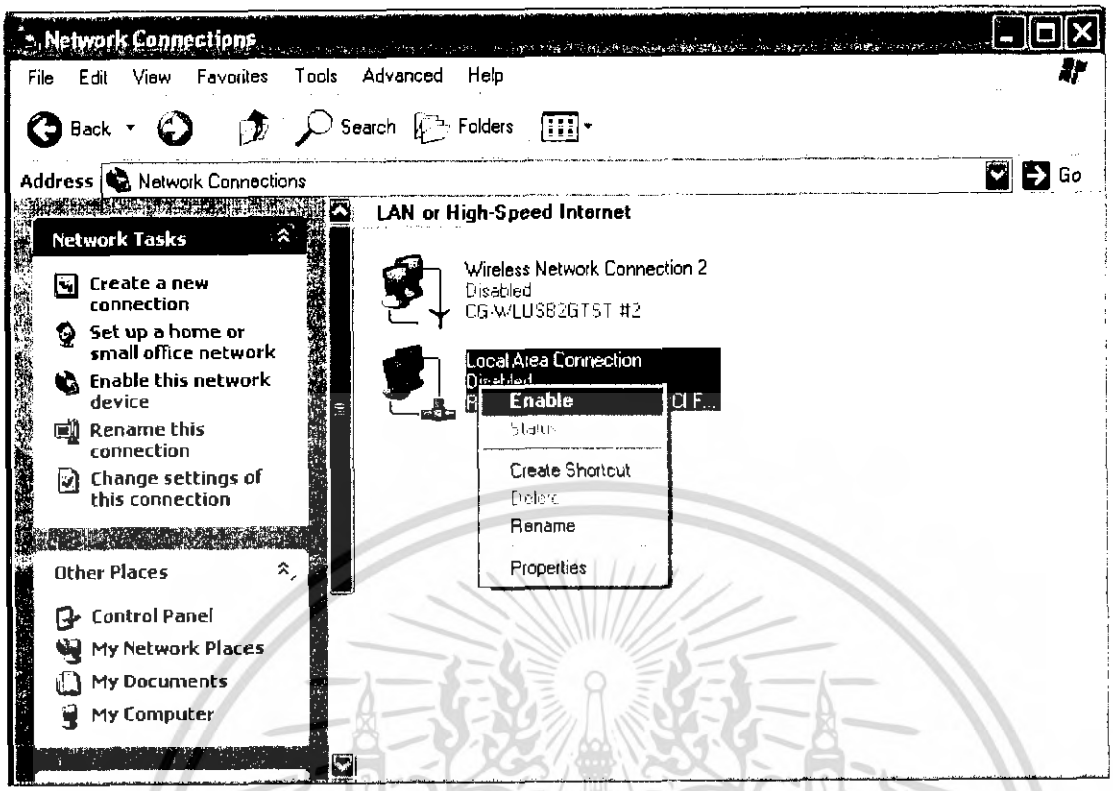
4.4 การใช้งานบอร์ดเป็นเว็บเซิร์ฟเวอร์ มีขั้นตอนดังนี้

1. ทำการต่อบอร์ดทดลองตามรูปที่ 4.4
2. ให้คอมพิวเตอร์ทำการเชื่อมต่อระบบ LAN โดยเข้าไปที่ My Network Places > View Network Connection
3. ทำการ Enable Network
4. เปิดโปรแกรม Internet Explorer Browser
5. ใส่ค่าไอพีแอดเดรสที่ได้ทำการกำหนดให้กับบอร์ดทดลองลงในช่องแอดเดรสของ Internet Explorer Browser
6. กดปุ่ม ENTER
7. เว็บเพ็จที่เก็บไว้ในบอร์ดทดลองจะปรากฏขึ้น

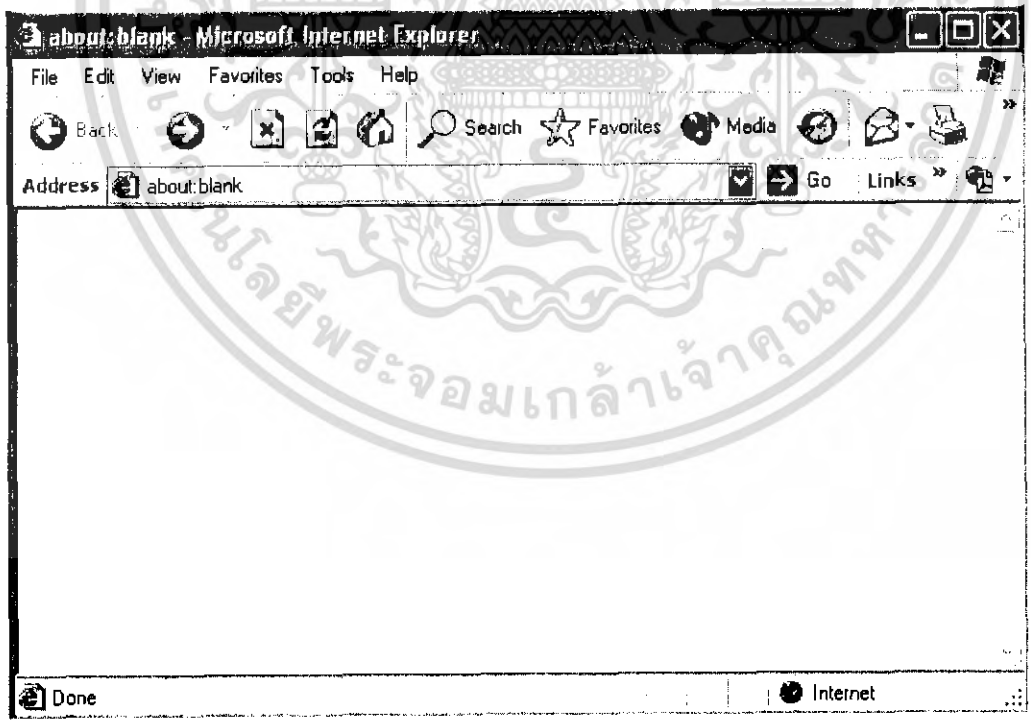


รูปที่ 4.13 การเชื่อมต่อบอร์ดทดลองเพื่อใช้งานเป็นเซิร์ฟเวอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

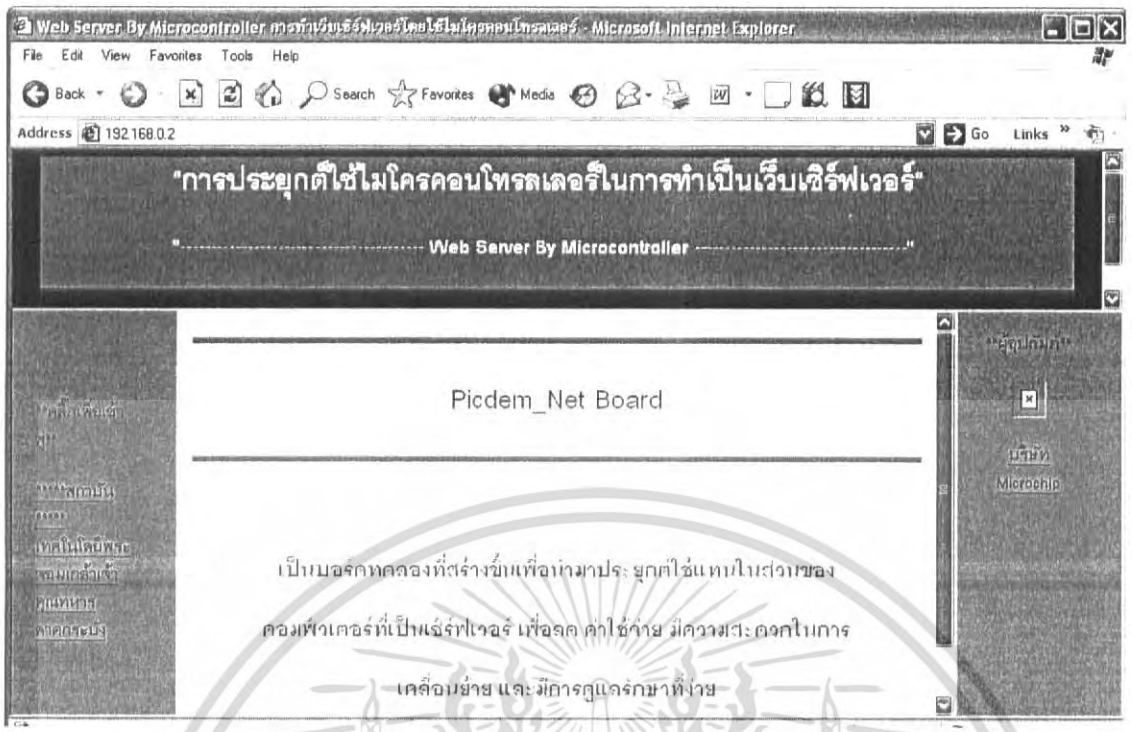


รูปที่ 4.14 แสดงการ Enable LAN



รูปที่ 4.15 แสดงหน้าต่างของ Internet Explorer Browser

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.16 แสดงเว็บเพจที่ได้ถูกเก็บไว้ในอีอีพรอมของบอร์ดทดลอง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 5

บทวิจารณ์และสรุป

5.1 บทสรุป

โครงงานนี้ได้นำเอาเทคโนโลยีไมโครคอนโทรลเลอร์ที่สามารถประมวลผลและจัดเก็บข้อมูลมาใช้งานเป็นเว็บเซิร์ฟเวอร์แทนเครื่องคอมพิวเตอร์ เพื่อใช้ในการแสดงเว็บเพ็จผ่านทางระบบอินเตอร์เน็ต โดยการใช้อุปกรณ์ติดต่อสื่อสารกับระบบอินเตอร์เน็ต คือ ไอซี RTL8019AS ซึ่งสามารถแปลงข้อมูลของไมโครคอนโทรลเลอร์ไปยังระบบอินเตอร์เน็ตและสามารถแปลงข้อมูลจากระบบอินเตอร์เน็ตเป็นข้อมูลที่ไมโครคอนโทรลเลอร์สามารถเข้าใจได้ มีการใช้หน่วยความจำแบบอีอีพรอมสำหรับเก็บข้อมูลที่เป็นเว็บเพ็จ มีหน่วยแสดงผลแบบจอ LCD เพื่อใช้แสดงผลโหมดการทำงานของไมโครคอนโทรลเลอร์ โดยได้เขียนโปรแกรมควบคุมอุปกรณ์ต่างๆไว้ในไมโครคอนโทรลเลอร์ PIC16F877 ทำให้สามารถใช้งานไมโครคอนโทรลเลอร์เป็นเว็บเซิร์ฟเวอร์แทนเครื่องคอมพิวเตอร์ได้

5.2 วิจารณ์สิ่งที่ได้จากโครงงาน

ข้อดีที่เกิดจากการใช้ไมโครคอนโทรลเลอร์เป็นเว็บเซิร์ฟเวอร์มีมากมาย เช่น การประหยัดงบประมาณ ความสะดวกในการพกพา และดูแลรักษาง่าย แต่การใช้ไมโครคอนโทรลเลอร์เป็นเว็บเซิร์ฟเวอร์ก็มีข้อเสียด้วยเช่นกัน เช่น ไม่สามารถส่งข้อมูลที่มีขนาดมากๆได้ มีเสถียรภาพน้อยกว่าเครื่องเซิร์ฟเวอร์มาก และรับส่งข้อมูลได้ช้ากว่ามาก ดังนั้น โครงงานนี้จึงเหมาะสำหรับเป็นทางเลือกของการรับส่งข้อมูลแบบธรรมดาไม่มีภาพกราฟิกสลับซับซ้อน

5.3 ปัญหาอุปสรรคและแนวทางในการแก้ไข

ปัญหาที่เกิดขึ้น

1. ไมโครคอนโทรลเลอร์ PIC 16F877A มีข้อจำกัดในด้านความเร็วในการรับส่งข้อมูลและการประมวลผลจึงทำให้การรับส่งเว็บเพ็จที่มีความสลับซับซ้อนไม่สามารถรับส่งได้

2. หน่วยความจำภายนอกมีขนาดไม่ใหญ่พอที่จะทำการเก็บข้อมูลที่มีขนาดใหญ่ได้

แนวทางแก้ไข

1. ใช้ไมโครคอนโทรเลอร์ที่มีประสิทธิภาพในการประมวลผลและส่งผ่านข้อมูลที่ดีขึ้นเพื่อแก้ไขปัญหาการรับส่งเว็บเพ็จที่สลับซับซ้อน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2. เพิ่มประสิทธิภาพในการเก็บข้อมูลที่เป็นเว็บเพจโดยการใช้หน่วยความจำที่มีขนาดใหญ่ขึ้น หรือใช้หน่วยความจำจำนวนมากขึ้น

5.4 แนวทางการพัฒนาต่อ

1. พัฒนาในส่วนของโปรแกรมให้มีเสถียรภาพมากยิ่งขึ้นเพื่อลดข้อผิดพลาดที่อาจจะเกิดขึ้น
2. พัฒนาในส่วนของอุปกรณ์ฮาร์ดแวร์ที่มาติดต่อกับไมโครคอนโทรลเลอร์เพื่อใช้ในการควบคุมอุปกรณ์ระยะไกลผ่านทางระบบอินเตอร์เน็ต เช่นอาจเพิ่มวงจรขับมอเตอร์เพื่อขับโหลดที่อยู่อีกที่หนึ่งผ่านทางระบบอินเตอร์เน็ต
3. เพิ่มการรับส่งข้อมูลแบบเรียลไทม์ (Real Time) เช่นการรับส่งค่าข้อมูลที่เป็นอุณหภูมิที่มีการเปลี่ยนแปลงอยู่ตลอดเวลาผ่านทางระบบอินเตอร์เน็ตเพื่อใช้ในการตรวจสอบสถานะของสภาพแวดล้อมที่ไม่เหมาะสมสำหรับการใช้มนุษย์ในการเข้าไปตรวจสอบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บรรณานุกรม

[1] สุวัฒน์ ปุณณชัยยะ, ต้น ตัณฑ์สุทธิวงศ์, สุพจน์ปุณณชัยยะ.**เปิดโลก TCP/IP และ โปรโตคอลขอ อินเตอร์เน็ต**.พิมพ์ครั้งที่ 1: สำนักพิมพ์ Provision.2545.

[2] จตุชัย แพงจันทร์, อนุโชต วุฒิพรพงษ์.**เจาะระบบ Network ฉบับสมบูรณ์** .พิมพ์ครั้งที่ 1: สำนักพิมพ์ Infopress.2546

[3] ณัฐพล วงศ์สุนทรชัย, ชัยวัตร ลิ้มพรจิตรวิไล.**เรียนรู้และปฏิบัติการไมโครคอนโทรลเลอร์ PIC16F877**.พระราชบัญญัติลิขสิทธิ์ 2521.สำนักพิมพ์ อินโนเวตีฟ เอ็กเพอริเมนต์จำกัด

[4] สวัสดิ ไกรคุ้ม.**การออกแบบเว็บกราฟฟิกด้วย HTML 4.0**.พิมพ์ครั้งที่ 1: สำนักพิมพ์ เดอะไลบรารี่ พับลิชชิ่ง.2541

[5] ธวัชชัย ศรีสุเทพ.**คัมภีร์ WEB DESIGN**.พิมพ์ครั้งที่ 3: สำนักพิมพ์ Provision.2544

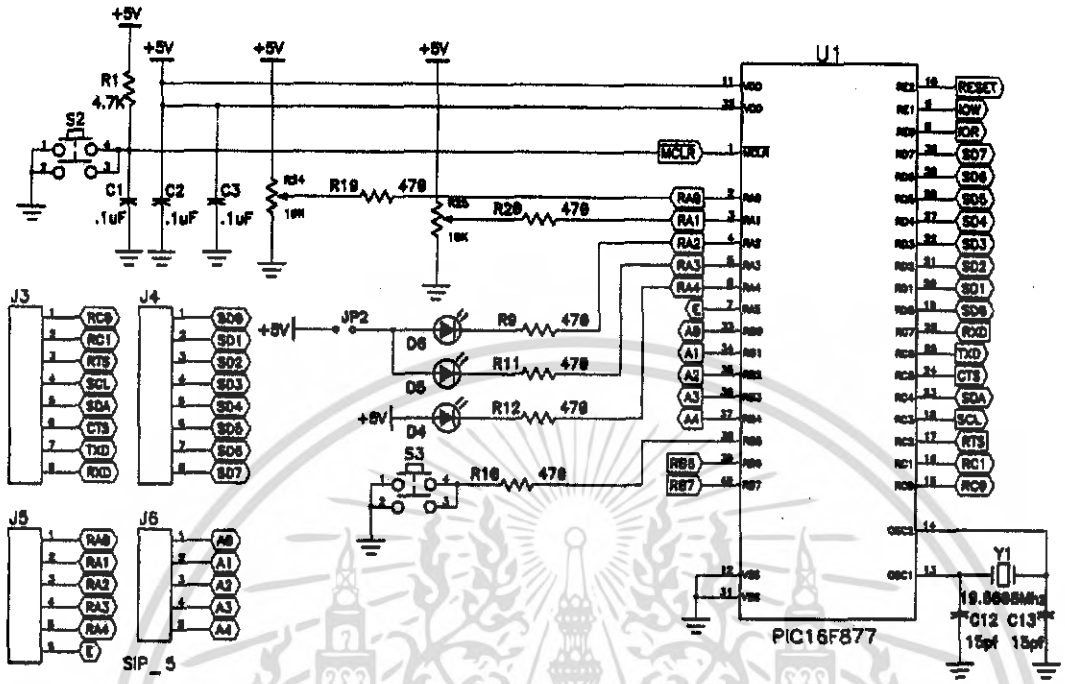
[6] Shelly Cashman Woods.**HTML Complete Concept and Techniques Course Technology** .Inc 2000

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

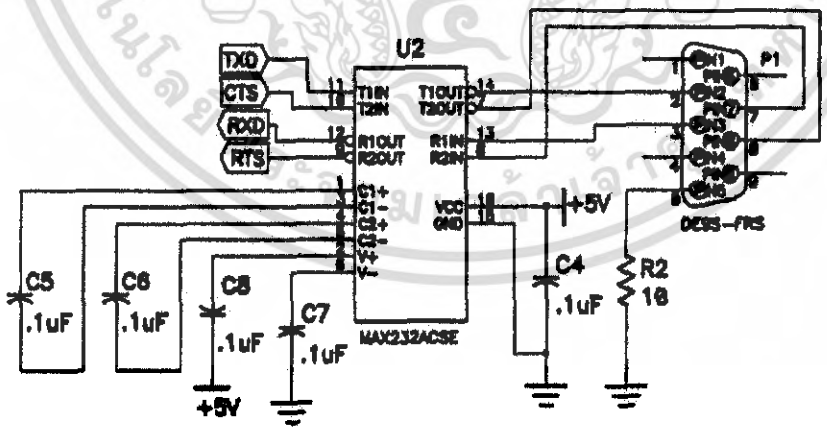
ภาคผนวก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ก

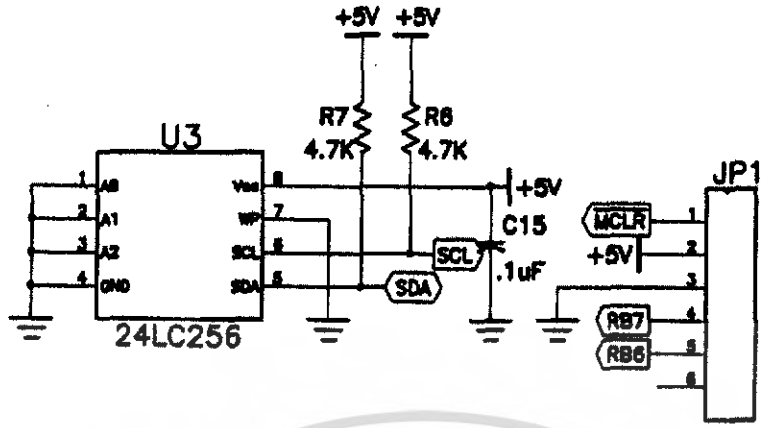


รูปที่ ก.1 รูปวงจรการเชื่อมต่อไมโครคอนโทรลเลอร์กับอุปกรณ์ต่างๆ

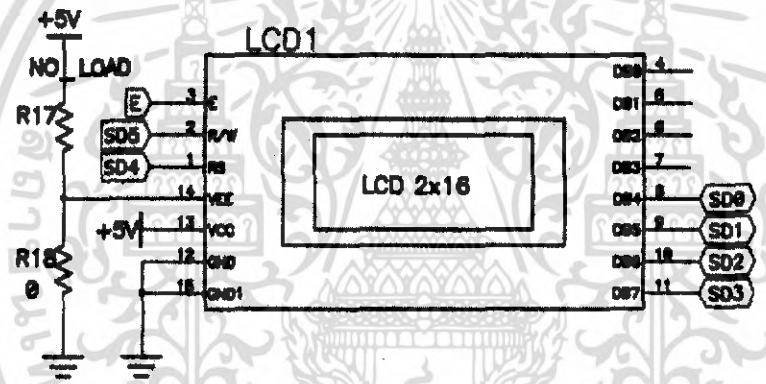


รูปที่ ก.2 รูปวงจรการเชื่อมต่อของไอซี MAX 232 ACSE สำหรับต่อกับพอร์ตอนุกรม

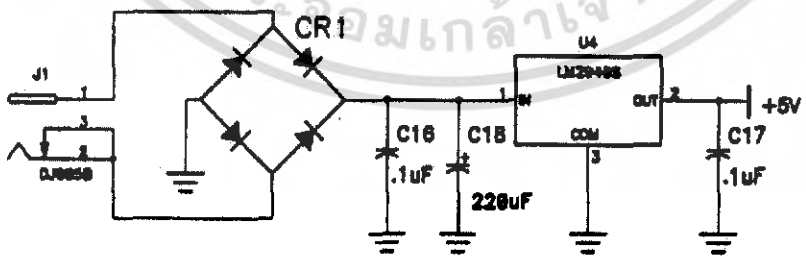
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ ก.3 รูปวงจรการเชื่อมต่อของไอซี 24LC256

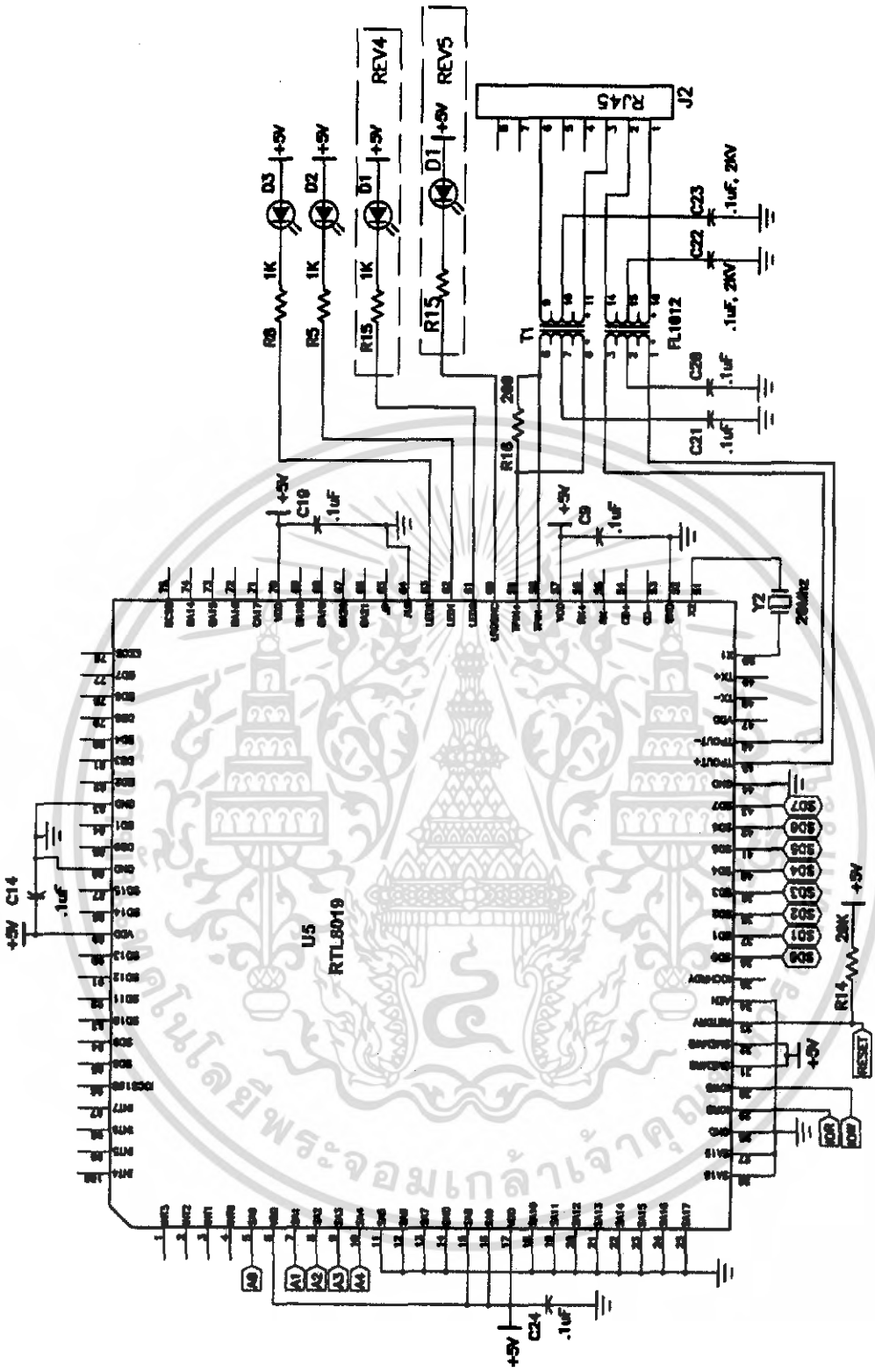


รูปที่ ก.4 รูปวงจรการเชื่อมต่อจอแสดงผล



รูปที่ ก.5 รูปวงจรการเชื่อมต่อไอซีปรับแรงดัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ ก.6 รูปวงจรการเชื่อมต่อไอซี RTL8019AS

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้