



ภาควิชาครุศาสตร์วิศวกรรม  
คณะครุศาสตร์อุตสาหกรรม  
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง  
ใบรับรองปริญญาโท

ชื่อหัวข้อ การทดลองเครือข่ายท้องถิ่นแบบเสมือนบนระบบอินเทอร์เน็ต  
VLAN Laboratory on Internet

ชื่อนักศึกษา 1. นายโกสินทร์ สังข์นุ่น รหัสประจำตัว 47035558  
2. นางสาวทัศนีย์ จัตุรภัทร รหัสประจำตัว 47035565  
3. ว่าที่ร้อยตรีชนภัทร เต็มเปี่ยม รหัสประจำตัว 47035567

หลักสูตร ครุศาสตร์อุตสาหกรรมบัณฑิต สาขาวิชา คอมพิวเตอร์  
อาจารย์ที่ปรึกษา อ.อำพล ทองระอา  
อาจารย์ที่ปรึกษาร่วม อ.ไพบุลย์ พวงวงศ์ตระกูล

คณะกรรมการสอบปริญญาโท	ลายมือชื่อ
1. อ.พิชญ์สินี มะโน	
2. อ.อำพล ทองระอา	
3. อ.สุระชัย พิมพ์สวัสดิ์	
4. อ.สุชิน อัจฉาญ	
5. อ.สมชาย หมั่นสายญาติ	

วัน/เดือน/ปีที่สอบ วันพุธที่ 10 เดือนพฤษภาคม พ.ศ. 2549 เวลา 16.00 น.

สถานที่สอบ ห้อง ค.310 คณะครุศาสตร์อุตสาหกรรม สจล.

ภาควิชารับรองแล้ว

ลงนาม.....

(ผศ.สุรสิทธิ์ รัตรี)

หัวหน้าภาควิชาครุศาสตร์วิศวกรรม  
วันที่ 1 เดือน พ.ค. พ.ศ. 2549



<BT481512>

การทดลองเครือข่ายท้องถิ่นแบบเสมือนบนระบบอินเทอร์เน็ต

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## ปริญญาบัตร

การทดลองเครือข่ายท้องถิ่นแบบเสมือนบนระบบอินเทอร์เน็ต

VLAN LABORATORY ON INTERNET



เลขที่.....  
เลขที่..... **66678**  
วันที่มอบ..... - ๐ พ.ย. 2549

11660242

ปริญญาบัตรฉบับนี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรครุศาสตรบัณฑิต

สาขาวิชาคอมพิวเตอร์

ภาควิชาครุศาสตร์วิศวกรรม คณะครุศาสตร์อุตสาหกรรม

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2548

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## ปริญญาโท

เรื่อง การทดลองเครือข่ายท้องถิ่นแบบเสมือนบนระบบอินเทอร์เน็ต  
VLAN Laboratory on Internet

### วัตถุประสงค์

1. เพื่อศึกษาภาษา Java และการตั้งค่าอุปกรณ์ Switch เชื่อมต่อเครือข่าย
2. เพื่อออกแบบระบบการทดลองเครือข่ายท้องถิ่นแบบเสมือนบนเครือข่ายอินเทอร์เน็ต
3. เพื่อสร้างระบบการทดลองเครือข่ายท้องถิ่นแบบเสมือนบนเครือข่ายอินเทอร์เน็ต
4. เพื่อทดสอบระบบการทดลองเครือข่ายท้องถิ่นแบบเสมือนบนเครือข่ายอินเทอร์เน็ต
5. เพื่อนำระบบการทดลองเครือข่ายท้องถิ่นแบบเสมือนบนเครือข่ายอินเทอร์เน็ตไปใช้งานได้

### ประโยชน์ที่คาดว่าจะได้รับ

1. ได้เรียนรู้ภาษา Java และการตั้งค่าอุปกรณ์ Switch เชื่อมต่อเครือข่าย
2. ได้รูปแบบระบบการทดลองเครือข่ายท้องถิ่นแบบเสมือนบนเครือข่ายอินเทอร์เน็ต
3. ได้ระบบการทดลองเครือข่ายท้องถิ่นแบบเสมือนบนเครือข่ายอินเทอร์เน็ต
4. ได้ผลการทดสอบระบบการทดลองเครือข่ายท้องถิ่นแบบเสมือนบนเครือข่ายอินเทอร์เน็ต
5. ได้ระบบการทดลองเครือข่ายท้องถิ่นแบบเสมือนบนเครือข่ายอินเทอร์เน็ตไปใช้งานได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

<b>ชื่อหัวข้อ</b>	การทดลองเครือข่ายท้องถิ่นแบบเสมือนบนระบบอินเทอร์เน็ต	
<b>นักศึกษา</b>	นายโกสินทร์	สังข์หนู่น
	นางสาวทัศนีย์	จักรภัทร
	ว่าที่ร้อยตรีชนภัทร	เต็มเปี่ยม
<b>อาจารย์ที่ปรึกษา</b>	อาจารย์อำพล	ทองระอา
<b>อาจารย์ที่ปรึกษาร่วม</b>	อาจารย์ไพบุลย์	พวงวงศ์ตระกูล
<b>หลักสูตร</b>	ครุศาสตร์อุตสาหกรรมบัณฑิต	
<b>สาขาวิชา</b>	คอมพิวเตอร์	
<b>ปีการศึกษา</b>	2548	

#### บทคัดย่อ

บริยุณานิพนธ์ฉบับนี้นำเสนอโปรแกรมทดลองการตั้งค่าให้กับอุปกรณ์สวิตช์เชื่อมต่อเครือข่าย ผ่านทางระบบเครือข่ายอินเทอร์เน็ต โดยมีการทดลองเกี่ยวกับเครือข่ายท้องถิ่นแบบเสมือน (VLAN) ในหัวข้อการทดลองการแบ่งพื้นที่การใช้งาน (Work Group) บนระบบเครือข่าย ซึ่งผู้ที่ต้องการศึกษาเกี่ยวกับการตั้งค่าอุปกรณ์สวิตช์เชื่อมต่อเครือข่าย สามารถทดลองตั้งค่าต่าง ๆ ได้โดยที่ไม่ต้องต่อกับอุปกรณ์จริง และสามารถทำการทดลองผ่านระบบอินเทอร์เน็ตได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

<b>Thesis</b>	VLAN Laboratory on Internet	
<b>Students</b>	Mr. Kosin	Sungnun
	Miss. Touchsanee	Jaturaphat
	Acting Sub Lt. Thanaphat	Tempiam
<b>Advisor</b>	Mr. Amphon	Thongra-ar
<b>Co-Advisor</b>	Mr. Paipoon	Pongwongtragull
<b>Education Level</b>	Bachelor of Science in Industrial Education	
<b>Program in</b>	Computer	
<b>Academic Year</b>	2005	

### ABSTRACT

This thesis presents construction of Laboratory Switch configuration on Internet. The Laboratory related with Virtual Local Area Network (VLAN). Experiment item workgroup provides on Internet which the person wants study Switch configuration. The person can use the configuration and the variable doesn't join to real accessories. This Laboratory can experiment on Internet.

## กิตติกรรมประกาศ

ปริญญาานิพนธ์ฉบับนี้สามารถสำเร็จลุล่วงได้ดีนั้น เนื่องมาจากความร่วมมือร่วมใจของสมาชิกภายในกลุ่มทุกท่าน คณะผู้จัดทำขอขอบพระคุณท่านอาจารย์อำพล ทองระอา อาจารย์ไพบุลย์ พวงวงศ์ตระกูล และอาจารย์ประจำภาควิชาครุศาสตร์วิศวกรรมทุกท่านเป็นอย่างมากที่ได้กรุณาให้คำปรึกษาในการแก้ไขปัญหาต่างๆ ตลอดจนจนถึงข้อมูลและอุปกรณ์ที่เป็นประโยชน์ต่อการทดลองของโครงการ และในการจัดทำปริญญาานิพนธ์ฉบับนี้ ขอขอบคุณห้องสมุดคณะครุศาสตร์ สำนักหอสมุดกลางและสำนักวิจัยคอมพิวเตอร์ที่ช่วยอำนวยความสะดวกและเอื้อเฟื้อสถานที่ในการค้นคว้าข้อมูลและทดลองโครงการ

ขอกราบขอบพระคุณบิดา มารดา และผู้มีพระคุณสำหรับพวกเราที่ได้ให้การสนับสนุนทุกสิ่งทุกอย่าง ทางด้านการศึกษาตลอดมาจนถึงปัจจุบัน และสุดท้ายต้องขอบคุณเพื่อนๆ ที่เป็นกำลังใจให้เสมอมา



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## สารบัญ

เรื่อง	หน้า
บทคัดย่อภาษาไทย	I
บทคัดย่อภาษาอังกฤษ	II
กิตติกรรมประกาศ	III
สารบัญ	IV
สารบัญตาราง	VII
สารบัญรูป	VIII
บทที่ 1 บทนำ	1
1.1 ความเป็นมาและความสำคัญของปัญหา	1
1.2 จุดมุ่งหมายของโครงการ	1
1.3 สมมุติฐานของการจัดทำโครงการ	1
1.4 ขีดความสามารถของโครงการ	1
1.5 ขั้นตอนการทำโครงการ	2
1.6 เนื้อหาโดยสังเขป	2
บทที่ 2 ทฤษฎีและหลักการ	3
2.1 กล่าวนำ	3
2.2 ระบบเครือข่าย	3
2.2.1 รูปแบบการเชื่อมต่อ (Topology)	3
2.2.2 ขนาดของระบบเครือข่าย	4
2.3 ประโยชน์ของระบบเครือข่าย	5
2.4 อุปกรณ์ในระบบเครือข่าย	6
2.4.1 ฮับหรือรีพีทเตอร์ (Hub, Repeater)	6
2.4.2 สวิตช์ (Switch)	7
2.4.3 บริดจ์ (Bridge)	8
2.4.4 เราเตอร์ (Router)	8
2.4.5 เกทเวย์ (Gateway)	9
2.5 ระบบเครือข่ายท้องถิ่นเสมือน (วีแลน)	9
2.5.1 สถาปัตยกรรมเครือข่ายที่ใช้สวิตช์และใช้วีแลน	10
2.5.2 ชนิดของวีแลน	13

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## สารบัญ (ต่อ)

เรื่อง	หน้า
2.5.3 มาตรฐานของวีแลน	15
2.5.4 ช่องโหว่ของการใช้วีแลน	16
2.5.5 ประโยชน์ของวีแลน	17
2.5.6 ข้อเสียและปัญหาที่พบของการใช้วีแลน	17
2.6 Spanning Tree	17
2.6.1 หลักการทำงานของ Spanning Tree	18
2.7 Java Servlet	19
2.7.1 จุดเด่นของ Servlet	20
2.7.2 สถาปัตยกรรมของ Servlet	20
2.7.3 การใช้ Java Servlet กับการพัฒนาระบบงานบนเครือข่ายอินเทอร์เน็ต Servlet API	21
บทที่ 3 การออกแบบ การสร้าง และการทำงาน	30
3.1 กล่าวนำ	30
3.2 การออกแบบโปรแกรม	30
3.2.1 ส่วนของโปรแกรมโปรแกรมที่อยู่ในเครื่องเซิร์ฟเวอร์	30
3.2.2 ส่วนของเครื่องผู้ใช้งาน	32
3.3 วิธีการสร้างโปรแกรม	34
3.4 การทำงานของส่วนต่างๆ	40
3.4.1 ส่วนของโปรแกรมโปรแกรมที่อยู่ในเครื่องเซิร์ฟเวอร์	40
3.4.2 ส่วนของเครื่องผู้ใช้งาน	40
บทที่ 4 การทดลองและผลการทดลอง	41
4.1 กล่าวนำ	41
4.2 การทดลองเครือข่ายท้องถิ่นแบบเสมือน	41
4.2.1 การต่อวงจรการทดลองเครือข่ายท้องถิ่นแบบเสมือน	41
4.2.2 การทดลองโปรแกรมระหว่างเครื่องผู้ใช้งานกับเครื่องเซิร์ฟเวอร์	42
4.2.3 การทดลองโปรแกรมระหว่างเครื่องเซิร์ฟเวอร์กับอุปกรณ์สวิตช์	43
บทที่ 5 บทสรุป	46
5.1 สรุป	46

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## สารบัญ (ต่อ)

เรื่อง	หน้า
5.2 ปัญหาและวิธีการแก้ไข	46
5.3 แนวทางการพัฒนา	47
บรรณานุกรม	48
ภาคผนวก ก รายละเอียดและคุณสมบัติของอุปกรณ์	49
ภาคผนวก ข รหัสต้นฉบับของโปรแกรม	58
ภาคผนวก ค คู่มือการใช้งาน	67
ประวัติผู้แต่ง	71



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## สารบัญตาราง

ตารางที่	หน้า
2.1 การกำหนดพอร์ตให้กับวีแลน	13
2.2 การกำหนด MAC Address ให้กับวีแลนต่างๆ	14
2.3 การแบ่งวีแลนโดยใช้ชนิดของโปรโตคอลกำหนด	14
2.4 การแบ่งวีแลนโดยใช้ IP Subnet	14
2.5 คำอธิบายส่วนต่างๆ ของมาตรฐาน 802.3	16
ค.1 การแก้ปัญหาเบื้องต้น	70



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## สารบัญรูป

รูปที่	หน้า
2.1 เครือข่ายแบบดาว (Star)	3
2.2 เครือข่ายแบบวงแหวน (Ring)	4
2.3 เครือข่ายแบบบัส (Bus)	4
2.4 การเชื่อมโยงในระบบเครือข่าย	6
2.5 อุปกรณ์กระจายสัญญาณ (Hub)	7
2.6 สวิตช์หรือบริดจ์	7
2.7 เราเตอร์	8
2.8 ใช้สวิตช์ในเครือข่าย	9
2.9 การกำหนดวงเวียนในเครือข่าย ATM	10
2.10 โครงสร้างการทำงานของเครือข่าย	11
2.11 การเชื่อมโยงระหว่างอุปกรณ์ต่างๆ	11
2.12 เป็นโครงสร้างการจัดกลุ่มอุปกรณ์ให้เข้าเป็นเครือข่ายวงเวียนภายใต้เครือข่ายแบบสวิตช์	12
2.13 วิธีการติดต่อระหว่างเครื่องลูกข่าย	13
2.14 รูปแบบของเฟรม 802.3 ก่อนที่จะทำวงเวียน Tagging	15
2.15 รูปแบบของเฟรม 802.3 ที่มีการ tagging 802.1Q แล้ว	15
2.16 การเชื่อมค่า Switching Hub จำนวน 2 ตัวโดยใช้สาย UTP	18
2.17 การตรวจสอบ Root Bridge	18
2.18 การทำงานของ Servlet	20
3.1 การเชื่อมต่อระหว่างเครื่องเซิร์ฟเวอร์ เครื่องของผู้ใช้งาน และสวิตช์	31
3.2 ผังงานการทำงานของเครื่องเซิร์ฟเวอร์	32
3.3 ผังงานการทำงานของเครื่องผู้ใช้งาน	33
3.4 โปรแกรม NetBeans IDE 4.1	34
3.5 ขั้นตอนการเขียนโปรแกรม ขั้นที่ 1	35
3.6 ขั้นตอนการเขียนโปรแกรม ขั้นที่ 2	35
3.7 ขั้นตอนการเขียนโปรแกรม ขั้นที่ 3	36
3.8 ขั้นตอนการเขียนโปรแกรม ขั้นที่ 4	36
3.9 ขั้นตอนการเขียนโปรแกรม ขั้นที่ 5	37
3.10 ขั้นตอนการเขียนโปรแกรม ขั้นที่ 6	37

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## สารบัญรูป (ต่อ)

รูปที่	หน้า
3.11 หน้าจอหลักของโปรแกรมในเครื่องเซิร์ฟเวอร์	38
3.12 หน้าจอหลักของโปรแกรมของเครื่องผู้ใช้งาน	39
4.1 การเชื่อมต่อระหว่างอุปกรณ์สวิตช์เชื่อมต่อเครือข่ายกับเครื่องคอมพิวเตอร์ลูกข่าย	41
4.2 ทดสอบการเชื่อมต่อกับอุปกรณ์สวิตช์	42
4.3 การติดต่อกับเครื่องเซิร์ฟเวอร์เป็นผลสำเร็จ	43
4.4 การแสดงผลตอบกลับจากเครื่องเซิร์ฟเวอร์	44
4.5 การแสดงผลตอบกลับจากอุปกรณ์สวิตช์	45
ค.1 หน้าจอหลักของโปรแกรมของเครื่องผู้ใช้งาน	69

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

# บทที่ 1

## บทนำ

### 1.1 ความเป็นมาและความสำคัญของปัญหา

เนื่องจากการเรียนวิชาทดลองปฏิบัติการฯ นั้น นักศึกษามีเวลาน้อยในการลงปฏิบัติ ทำให้ไม่สามารถเข้าใจเรื่องที่ปฏิบัติได้ถ่องแท้ และเมื่อนักศึกษาผ่านการเรียนและลงปฏิบัติไปแล้วไม่สามารถทบทวนการปฏิบัติได้สะดวกนัก นอกจากนี้ในการลงปฏิบัติอุปกรณ์มีจำนวนจำกัดทำให้การลงปฏิบัติทำได้ไม่ทั่วถึง

### 1.2 จุดมุ่งหมายของโครงการ

สร้างระบบการเรียนรู้อุปกรณ์สวิตช์เชื่อมต่อเครือข่าย โดยผู้ใช้สามารถใช้งานผ่านทางระบบอินเทอร์เน็ต ซึ่งระบบการทดลอง VLAN บนเครือข่ายอินเทอร์เน็ตนี้ โดยผู้ใช้สามารถใช้งานได้ทั้งที่มีระบบอินเทอร์เน็ต เป็นระบบการทำงานเสมือนว่าผู้ใช้ได้ทำการตั้งค่ากับอุปกรณ์ สวิตช์เชื่อมต่อเครือข่ายจริงๆ และสามารถทดลองได้หลายๆ ครั้งเพื่อให้เกิดความเข้าใจที่ดียิ่งขึ้นเรื่อง Switch Configuration

### 1.3 สมมุติฐานของการจัดทำโครงการ

เมื่อผ่านการปฏิบัติการทดลองตามโครงการนี้แล้ว ผู้เรียนจะมีความรู้ความสามารถเกี่ยวกับการตั้งค่าต่างๆ ของอุปกรณ์สวิตช์เชื่อมต่อเครือข่าย ซึ่งผู้ใช้งานสามารถทดลองผ่านระบบอินเทอร์เน็ต โดยระบบการทดลองนี้จะผ่านการประเมินจากผู้ทรงคุณวุฒิในระดับผลการประเมิน ดี ขึ้นไป

### 1.4 ขีดความสามารถของโครงการ

โครงการนี้มีขีดความสามารถดังนี้

1. สามารถทำการตั้งค่าอุปกรณ์สวิตช์เชื่อมต่อเครือข่ายผ่านระบบอินเทอร์เน็ต
2. สามารถแสดงผลการตั้งค่าอุปกรณ์สวิตช์เชื่อมต่อเครือข่ายที่เครื่องของผู้ใช้ได้
3. ควบคุมการทำงานระบบการทดลองด้วยภาษา Java
4. ใช้อุปกรณ์สวิตช์เชื่อมต่อเครือข่ายเป็นตัวตั้งค่ารูปแบบการเชื่อมต่อ
5. มีเครื่องแม่ข่ายสำหรับจัดการตามรูปแบบการทดลอง
6. มีการทดลองครอบคลุมเนื้อหาเรื่องเครือข่ายท้องถิ่นเสมือนดังนี้

6.1) การทดลองเครือข่ายท้องถิ่นแบบเสมือนใช้อุปกรณ์สวิตช์เชื่อมต่อเครือข่ายหนึ่งตัว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## 1.5 ขั้นตอนของการทำโครงการ

โครงการนี้ประกอบด้วยส่วนของอุปกรณ์สวิตช์เชื่อมต่อเครือข่ายซึ่งเป็นฮาร์ดแวร์ และส่วนของชุดคำสั่งการติดต่อซึ่งเป็นซอฟต์แวร์ การทำงานในระยะแรกจะเริ่มต้นจากการสร้างรูปแบบการเชื่อมต่อระหว่างฮาร์ดแวร์ จึงเริ่มเขียนคำสั่งเพื่อใช้ในการติดต่อระหว่างฮาร์ดแวร์

## 1.6 เนื้อหาโดยสังเขป

เนื้อหาในปฏิญานีพจนานุกรมฉบับนี้แบ่งออกเป็นบทต่างๆ เพื่อสะดวกต่อการศึกษาและทำความเข้าใจ ในแต่ละบทประกอบด้วยเนื้อหาดังต่อไปนี้

บทที่ 1 กล่าวถึงความเป็นมาความสำคัญของปฏิญานีพจนานุกรม ชี้ความสามารถของโครงการ และเนื้อหาของโครงการโดยสังเขป

บทที่ 2 ประกอบด้วย ทฤษฎีและหลักการต่างๆ ประกอบด้วยเนื้อหา ดังนี้ คือ ภาษาจาวา วิแลน (VLAN) อุปกรณ์สวิตช์เชื่อมต่อเครือข่าย

บทที่ 3 กล่าวถึงเกี่ยวกับการออกแบบ การสร้าง และการทำงาน กล่าวถึงการสร้างที่มีลำดับขั้น การศึกษาโปรแกรมต่างๆ ขั้นตอนการทำงานและขั้นตอนการออกแบบโดยใช้ภาษา Java

บทที่ 4 ประกอบด้วย การทดลองและผลการทดลอง กล่าวถึงขั้นตอนการทดลองและ การทดสอบประสิทธิภาพในการทำงานของโครงการทดลองวิแลนบนเครือข่ายอินเทอร์เน็ต เพื่อตรวจสอบว่าโครงการนี้สามารถทำงานได้ตามวัตถุประสงค์หรือไม่

บทที่ 5 เป็นการสรุปผลในการจัดทำโครงการ ปัญหาที่เกิดขึ้น และได้เสนอแนวทางในการแก้ไขปัญหา รวมทั้งแนวทางการพัฒนาให้มีประสิทธิภาพมากยิ่งขึ้น

ภาคผนวก ก รายละเอียดและคุณสมบัติของอุปกรณ์สวิตช์

ภาคผนวก ข รหัสต้นฉบับของโปรแกรม

ภาคผนวก ค คู่มือการใช้งาน

## บทที่ 2

### ทฤษฎีและหลักการ

#### 2.1 กล่าวนำ

เนื้อหาของปริญญาโทฉบับนี้เป็นทฤษฎีและหลักการที่จะนำมาใช้ประกอบการสร้างโครงงานโดยประกอบด้วย เนื้อหาระบบเครือข่าย อุปกรณ์ที่ใช้ในระบบเครือข่าย อุปกรณ์สวิตช์เชื่อมต่อเครือข่าย วีแลน (VLAN) Spanning Tree และ Java Servlet

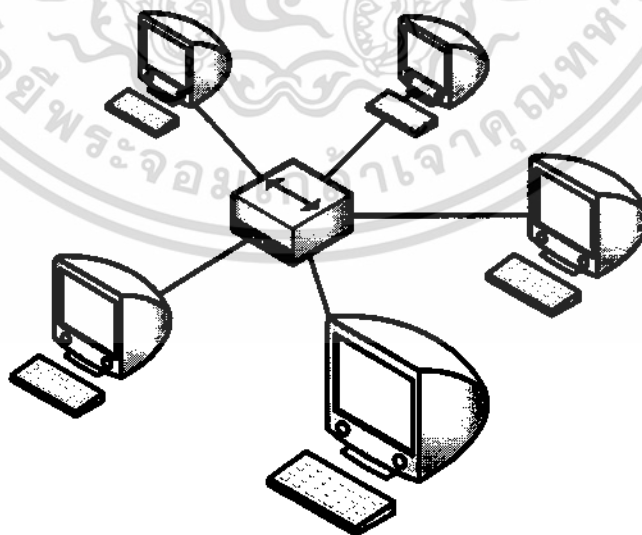
#### 2.2 ระบบเครือข่าย

ระบบเครือข่ายหรือเน็ตเวิร์ก (Network) คือ ระบบที่มีคอมพิวเตอร์ ตั้งแต่ 2 เครื่องขึ้นไปเชื่อมต่อกันอยู่หรือโหนด (Node) ที่มีการเชื่อมต่อกันด้วยเส้นทางการสื่อสารอย่างใดอย่างหนึ่ง และระบบเครือข่ายใดๆ สามารถที่จะมีระบบเครือข่ายย่อยๆ ซ้อนอยู่ในตัวมัน

การจำแนกระบบเครือข่ายตามวิธีที่นิยมกัน 3 วิธีคือ รูปแบบการเชื่อมต่อโทโปโลยี รูปแบบการสื่อสาร (Protocol) และสถาปัตยกรรมเครือข่าย (Architecture) รูปแบบการเชื่อมต่อบอกถึงรูปแบบที่ทำการเชื่อมต่ออุปกรณ์ในเครือข่ายเข้าด้วยกัน ซึ่งมีรูปแบบที่นิยมกัน 3 วิธี

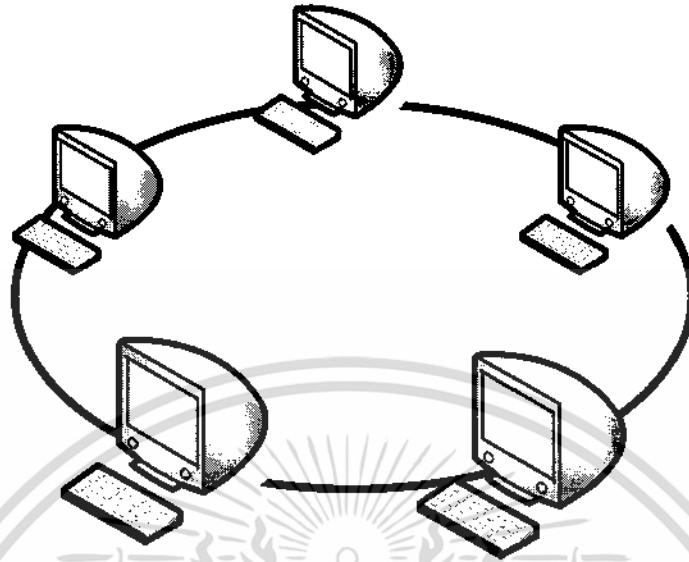
##### 2.2.1 รูปแบบการเชื่อมต่อโทโปโลยี

รูปแบบที่นิยมใช้กันมีอยู่ 3 วิธีคือแบบดาว (Star) แบบวงแหวน (Ring) และแบบบัส (Bus)

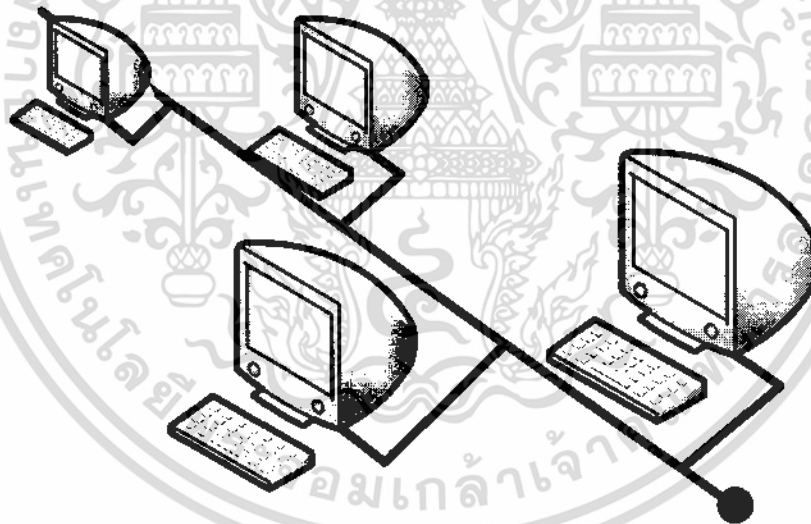


รูปที่ 2.1 เครือข่ายแบบดาว (Star)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.2 เครือข่ายแบบวงแหวน (Ring)



รูปที่ 2.3 เครือข่ายแบบบัส (Bus)

## 2.2.2 ขนาดของระบบเครือข่าย

### 2.2.2.1 เครือข่ายท้องถิ่น (LAN : Local Area Network)

เป็นระบบเครือข่ายขนาดเล็ก ที่ใช้ในการเชื่อมโยงคอมพิวเตอร์ในบริเวณใกล้เคียงเข้าด้วยกัน เช่น ภายในห้องเดียวกันหรือในสำนักงานเดียวกัน โดยทั่วไป ระบบเครือข่ายแบบเครือข่ายท้องถิ่นใช้เพื่อ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

วัตถุประสงค์ ในการเชื่อมต่อเครื่องคอมพิวเตอร์ กับอุปกรณ์ต่างๆ ในอาคารเพื่อแบ่งปันการใช้ข้อมูลและอุปกรณ์ร่วมกันเช่น เครื่องพิมพ์ ไฟล์ข้อมูล เป็นต้น

#### 2.2.2.2 แมน (MAN : Metropolitan Area Network)

เป็นเครือข่ายที่มีขนาดใหญ่กว่าเครือข่ายท้องถิ่น ทำการเชื่อมต่อกัน ในระดับเมืองหรืออาคารหลายๆ หลังเข้าด้วยกันโดยใช้สายสัญญาณที่มีความเร็วสูงเป็นสายสัญญาณหลัก ที่ใช้ในการรับส่งข้อมูล ทำให้ระบบเครือข่ายในอาคารต่างๆ ทำงานที่ระดับความเร็วสูงเช่นกัน ตัวอย่างของการใช้งาน ระบบเครือข่ายในลักษณะนี้ เช่น ในมหาวิทยาลัยโดยทำการเชื่อมต่อเครือข่ายเครือข่ายท้องถิ่นของคณะหรือภาควิชาต่างๆ เข้าด้วยกัน เรียกว่า แคมปัสเน็ตเวิร์ก (Campus Network)

#### 2.2.2.3 ระบบเครือข่ายระยะไกล (WAN : Wide Area Network)

เป็นระบบเครือข่ายที่ครอบคลุมพื้นที่กว้างขวางเป็นระยะทางหลายๆ กิโลเมตร จนถึงระดับประเทศและทวีป การเชื่อมต่อในระบบเครือข่ายระยะไกลส่วนใหญ่ใช้สายสัญญาณต่างๆ ไป เช่น โทรศัพท์หรือการเช่าสายลีสไลน์ (Leased Line) การใช้ดาวเทียมการใช้สัญญาณไมโครเวฟ และสัญญาณอื่นๆ ที่สามารถส่งข้อมูลผ่านระยะทางไกลๆ ได้ ระบบงานที่ใช้ในเครือข่ายระยะไกล ได้แก่ ระบบฝากถอนผ่านธนาคาร ระบบบัตรเครดิต และระบบโอนเงินต่างประเทศ เป็นต้น

### 2.3 ประโยชน์ของระบบเครือข่าย

ผู้ใช้ทั่วไปที่ใช้คอมพิวเตอร์ส่วนบุคคล แบบคิดป็นเดี่ยวหรือภาษาทางเทคนิค เรียกกันว่า "Stand Alone System" นั้น อาจคุ้นเคยกับการใช้งานเพียงคนเดียวมาจึงไม่มีความจำเป็นต้องหันมาใช้งานเครือข่ายเน็ตเวิร์กหรือระบบคอมพิวเตอร์เครือข่ายนั้นให้ประโยชน์อะไรได้บ้าง

1. ทำให้ใช้ทรัพยากร ของเครื่องคอมพิวเตอร์ร่วมกันได้ (Resources Sharing) ซึ่งเป็นการช่วยประหยัดค่าใช้จ่ายและเพิ่มความสะดวกในเวลากการทำงาน เช่น การใช้พื้นที่บนฮาร์ดดิสก์และเครื่องพิมพ์ร่วมกัน สามารถบริหารจัดการการทำงานของคอมพิวเตอร์ทุกเครื่องได้จากศูนย์กลาง (Centralized Management) เช่น สร้างเวิร์กกรุปกำหนดสิทธิ์ในการเข้าถึงข้อมูลและสามารถทำการสำรองข้อมูลของแต่ละเครื่องได้
2. การสื่อสารภายในเครือข่าย (Communication) มีด้วยกันหลายแบบ เช่น อีเมล, การพูดคุย (Chat), การประชุมทางไกล (Teleconference) และการประชุมทางไกลซึ่งสามารถมองเห็นภาพได้ (Video Conference)
3. มีระบบรักษาความปลอดภัย ของข้อมูลบนเครือข่าย (Network Security) เช่น สามารถระบุผู้ที่มีสิทธิ์เข้าถึงข้อมูลในระดับต่างๆ ป้องกันผู้ไม่ได้รับการอนุญาตให้เข้าถึงข้อมูลและให้การคุ้มครองข้อมูลที่สำคัญ
4. ให้ความบันเทิงไม่รู้จัก (Entertainment) เช่น สามารถสนุกกับการเล่นเกมแบบที่ว่ามีผู้เล่น

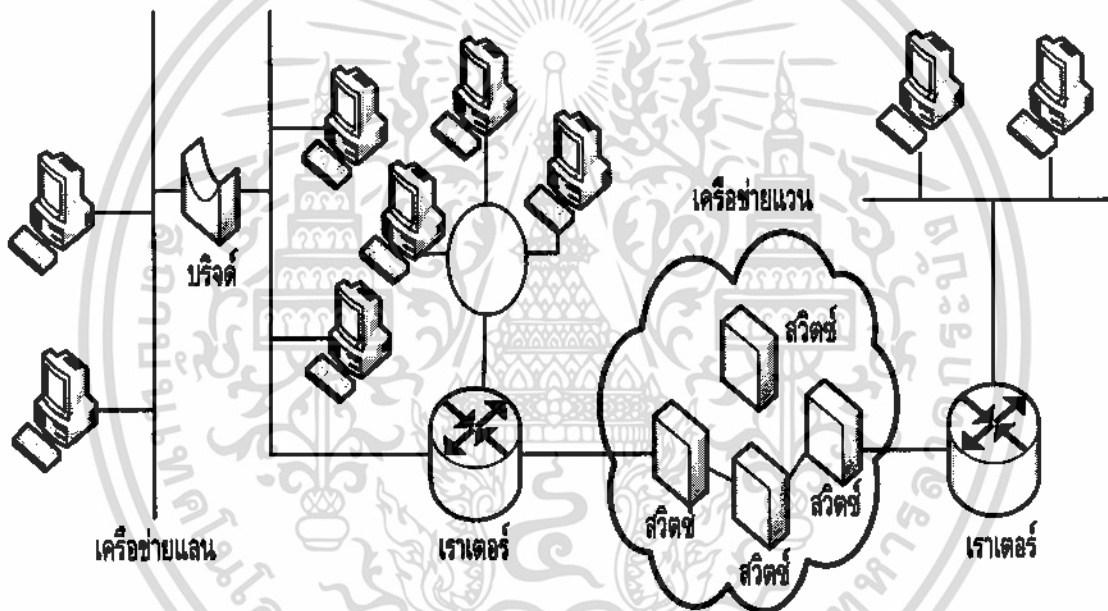
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หลายคน หรือที่เรียกว่า มัลติเพลเยอร์ (Multi Player)

- ใช้งานอินเทอร์เน็ตร่วมกัน (Internet Sharing) เพียงเชื่อมเข้าอินเทอร์เน็ตจากเครื่องหนึ่งในเครือข่ายโดยมีแอดเคาท์เพียงหนึ่งแอดเคาท์ก็ทำให้ผู้ใช้อีกหลายคนที่อยู่ในเครือข่ายเดียวกันสามารถใช้งานอินเทอร์เน็ตได้เสมือนกับมีหลายแอดเคาท์

## 2.4 อุปกรณ์ในระบบเครือข่าย

อุปกรณ์ที่ใช้ในการเชื่อมโยงเครือข่ายและทำหน้าที่ในการรับส่งข้อมูลระหว่างเครือข่าย มีหลายประเภทด้วยกัน อุปกรณ์แต่ละชนิดมีขีดความสามารถแตกต่างกันออกไป

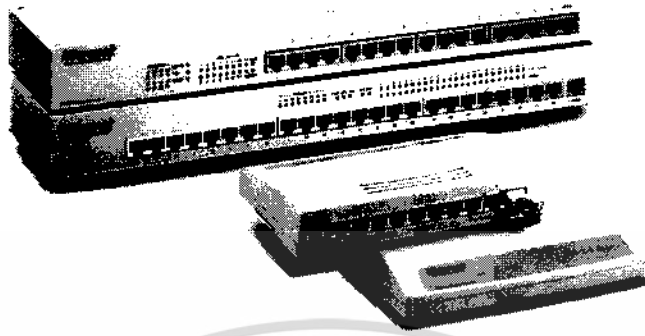


รูปที่ 2.4 การเชื่อมโยงในระบบเครือข่าย

### 2.4.1 ฮับหรือรีพีทเตอร์ (Hub, Repeater)

ฮับหรือรีพีทเตอร์ (Hub, Repeater) เป็นอุปกรณ์ที่ทวนและขยายสัญญาณ เพื่อส่งต่อไปยังอุปกรณ์อื่น ให้ได้ระยะทางที่ยาวไกลขึ้น ไม่มีการเปลี่ยนแปลงข้อมูลก่อนและหลังการรับ-ส่ง และไม่มีการใช้ซอฟต์แวร์ใดๆ มาเกี่ยวข้องกับอุปกรณ์ชนิดนี้ การติดตั้งจึงทำได้ง่าย ข้อเสียคือ ความเร็วในการส่งข้อมูลจะเฉลี่ยลดลงเท่ากันทุกเครื่องเมื่อมีคอมพิวเตอร์มาเชื่อมต่อมากขึ้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.5 อุปกรณ์กระจายสัญญาณ (Hub)

#### 2.4.2 สวิตช์ (Switch)

สวิตช์ เป็นอุปกรณ์สำหรับเชื่อมต่อเครือข่ายท้องถิ่นประเภทเดียวกัน ใช้โปรโตคอลเดียวกันสองวงเข้าด้วยกัน เช่น ใช้เชื่อมต่ออีเทอร์เน็ตแลน (Ethernet LAN) หรือโทเคนริงแลน (Token Ring LAN) ทั้งนี้สวิตช์จะมีความสามารถในการเชื่อมต่อฮาร์ดแวร์และตรวจสอบข้อผิดพลาดของการส่งข้อมูลได้ด้วยความเร็วในการส่งข้อมูลที่ไม่ลดลงและติดตั้งง่าย

สวิตช์มีหลายแบบหากแบ่งกลุ่มข้อมูลเป็นแพ็คเกจเล็กๆ และเรียกใหม่ว่า เซล ก็กลายเป็น เซลสวิตช์ หรือที่รู้จักกันในนาม เอทีเอ็มสวิตช์ ถ้าสวิตช์ข้อมูลในระดับเฟรมของอีเทอร์เน็ต ก็เรียกว่า อีเทอร์เน็ตสวิตช์ และถ้าสวิตช์ตามมาตรฐานเฟรมข้อมูลที่เป็นกลางและสามารถนำข้อมูลอื่นมาประกอบภายในได้ก็เรียกว่า เฟรมรีเลย์ อุปกรณ์สวิตซ์จึงเป็นอุปกรณ์ที่ใช้เทคโนโลยีใหม่และมีแนวโน้มที่จะพัฒนาให้ใช้กับความเร็วของการรับส่งข้อมูลจำนวนมาก เช่น เฟรมรีเลย์และเอทีเอ็มสวิตซ์สามารถสวิตซ์ข้อมูลขนาดหลายร้อยล้านบิตต่อวินาทีได้เทคโนโลยีนี้จึงเป็นเทคโนโลยีที่กำลังได้รับความนิยม



รูปที่ 2.6 สวิตช์หรือบริดจ์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 2.4.3 บริดจ์ (Bridge)

บริดจ์ เป็นอุปกรณ์เชื่อมโยงเครือข่ายสองเครือข่ายที่แยกจากกัน แต่ก่อนนั้นบริดจ์ได้รับการออกแบบมาให้ใช้กับเครือข่ายประเภทเดียวกัน เช่น ใช้เชื่อมโยงระหว่างอีเทอร์เน็ตกับอีเทอร์เน็ต บริดจ์เหมือนสะพานเชื่อมระหว่างสองเครือข่ายการติดต่อภายในเครือข่ายเดียวกันมีลักษณะการส่งข้อมูลแบบกระจาย (Broadcasting) ดังนั้นจึงกระจายได้เฉพาะเครือข่ายเดียวกันเท่านั้น การรับส่งภายในเครือข่ายมีข้อกำหนดให้แพ็คเกจที่ส่งกระจายไปยังตัวรับได้ทุกตัว แต่ถ้ามีการส่งมาที่แอดเดรสต่างเครือข่าย บริดจ์จะนำข้อมูลเฉพาะแพ็คเกจนั้นส่งให้ บริดจ์จะเป็นตัวแบ่งแยกข้อมูลระหว่างเครือข่ายให้มีการสื่อสารภายในเครือข่ายไม่ปะปนไปยังอีกเครือข่ายหนึ่งเพื่อลดปัญหาปริมาณข้อมูลกระจายในสายสื่อสารมากเกินไป

### 2.4.4 เราเตอร์ (Router)

เป็นอุปกรณ์ที่ทำงานคล้ายสวิตช์แต่จะสามารถเชื่อมต่อระบบที่ใช้สื่อหรือสายสัญญาณต่างชนิดกันได้ เช่น เชื่อมต่ออีเทอร์เน็ตแลนที่ส่งข้อมูลแบบยูทีพี (UTP : Unshield Twisted Pair) เข้ากับอีเทอร์เน็ตอีกเครือข่ายแต่ใช้สายแบบโคแอกเซียล (Coaxial Cable) ได้นอกจากนี้ยังช่วยเลือกหรือกำหนดเส้นทางที่จะส่งข้อมูลผ่านและแปลงข้อมูลให้เหมาะสมกับการนำส่งแน่นอนว่าการติดตั้งจะยุ่งยากมากขึ้น

เราเตอร์จะรับข้อมูลเป็นแพ็คเกจเข้ามาตรวจสอบแอดเดรสปลายทาง จากนั้นนำมาเปรียบเทียบกับตารางเส้นทางที่ได้รับการโปรแกรมไว้เพื่อหาเส้นทางที่ส่งต่อ หากเส้นทางที่ส่งต่อมีมาตรฐานทางเครือข่ายที่แตกต่างออกไป ก็จะแปลงให้เข้ากับมาตรฐานใหม่ เช่น รับข้อมูลมาจากอีเทอร์เน็ตและส่งต่อทางพอร์ตแวนที่เป็นแบบจุดไปจุด ก็จะมีการปรับปรุงรูปแบบสัญญาณให้เข้ากับมาตรฐานใหม่ เพื่อส่งไปยังเครือข่ายระยะไกลได้ อุปกรณ์เราเตอร์ได้รับการพัฒนาไปมาก ทำให้การใช้งานเราเตอร์มีประสิทธิภาพ โดยเฉพาะเมื่อเชื่อมอุปกรณ์เราเตอร์หลายๆ ตัวเข้าด้วยกันเป็นเครือข่ายขนาดใหญ่ เราเตอร์สามารถทำงานอย่างมีประสิทธิภาพโดยการหาเส้นทางเดินที่สั้นที่สุด เลือกตามความเหมาะสมและแก้ปัญหาที่เกิดขึ้นในระบบเครือข่ายได้



รูปที่ 2.7 เราเตอร์

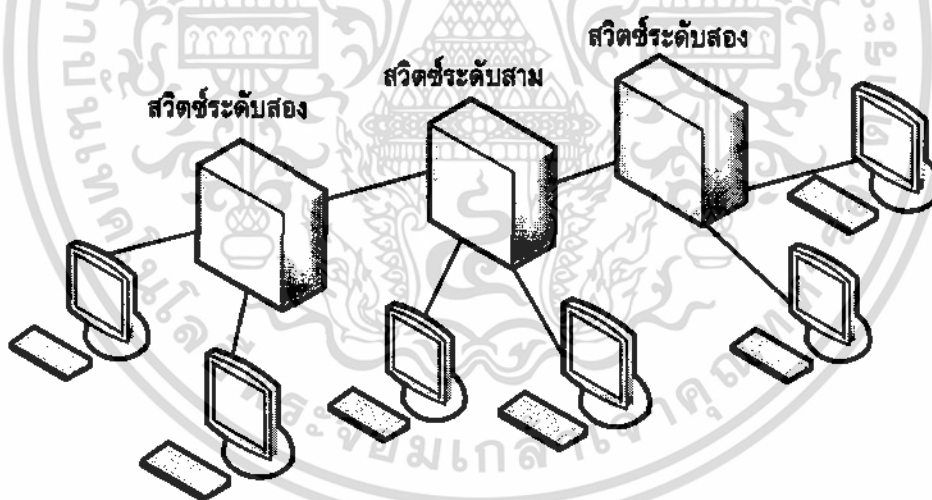
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 2.4.5 เกตเวย์ (Gateway)

เกตเวย์ เป็นอุปกรณ์ที่มีความสามารถสูงที่สุดในการเชื่อมต่อเครือข่ายต่างๆ เข้าด้วยกัน โดยไม่มีขีดจำกัด ทั้งระหว่างเครือข่ายต่างระบบหรือแม้กระทั่งโปรโตคอลจะแตกต่างกันออกไปเกตเวย์จะแปลงโปรโตคอลให้เหมาะสมกับอุปกรณ์ที่ต่างชนิดกันจัดเป็นอุปกรณ์ที่มีราคาแพงและติดตั้งใช้งานยุ่งยาก เกตเวย์บางตัวจะรวมคุณสมบัติในการเป็น เราเตอร์ด้วยในตัวหรือแม้กระทั่ง อาจรวมเอาฟังก์ชันการทำงาน ด้านการรักษาความปลอดภัย ที่เรียกว่า ไฟร์วอลล์ (Firewall) เข้าไว้ด้วย

### 2.5 ระบบเครือข่ายท้องถิ่นเสมือน (VLAN)

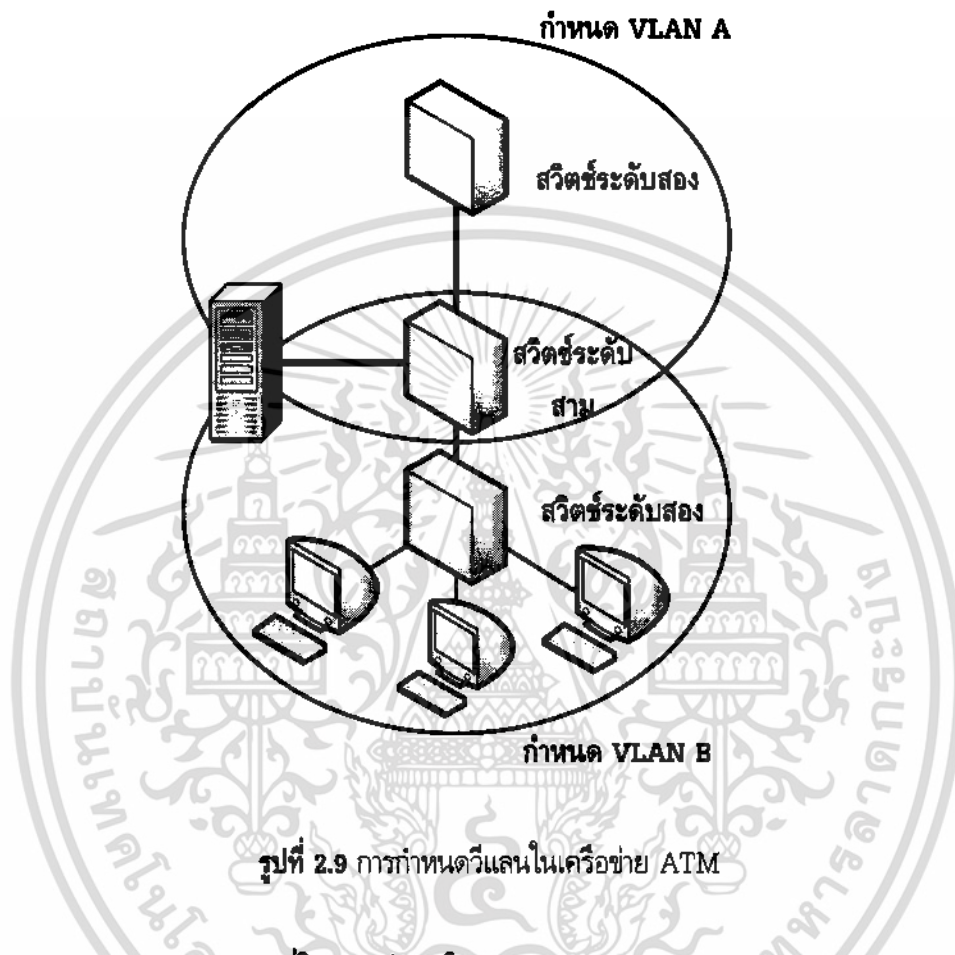
VLAN ย่อมาจาก Virtual Local Area Network หมายถึงสถานการณ์สร้างระบบเครือข่ายท้องถิ่นในเครือข่ายสวิตชิงหรือเอทีเอ็ม เพื่อให้กลุ่มของคอมพิวเตอร์และอุปกรณ์สามารถเชื่อมโยงกันเป็นกลุ่มแบบลอจิกคอล โดยมีสถานการณ์การทำงานที่เหมือนระบบเครือข่ายท้องถิ่น ในกลุ่มวิแลนเดียวกันนั้นสามารถที่จะกระจายข่าวสารระหว่างกันสถานการณ์การทำงานโดยรวม จะทำให้เหมือนหรือคล้ายสถานการณ์ระบบเครือข่ายท้องถิ่นที่ใช้ในการเชื่อมโยงแบบเราเตอร์



รูปที่ 2.8 ใช้สวิตช์ในเครือข่าย

เห็นได้ว่าวิแลนก็คือการสร้างเครือข่ายสวิตชิงให้แบ่งแยกเป็นเครือข่ายย่อยตามสถานะ การทำงานโดยกำหนดให้เป็นเสมือนเครือข่ายท้องถิ่นที่ต่อกันเป็นกลุ่มๆ ดังที่เคยเป็นในระบบที่ใช้เราเตอร์ การกำหนดกลุ่มวิแลนแต่ละกลุ่ม ก็เพื่อให้การทำงานเหมือนเป็นกลุ่มเครือข่ายย่อยหนึ่งเครือข่ายและหากต้องการส่งข้อมูลข้ามเครือข่ายวิแลน ก็จะได้เสมือนการมีฟังก์ชันของการกำหนดเส้นทาง เช่น จากรูปที่ 2.9 เราได้ทำการกำหนดวิแลนสองเครือข่ายคือ วิแลน A และ วิแลน B สังเกตว่าอุปกรณ์บางตัวเรากำหนดคาบเกี่ยวกันได้ คือให้เป็นเอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ทั้งเครือข่าย A และ B หากมีการส่งข้อมูลข้ามวีแลน อุปกรณ์สวิตช์ซึ่งจะดำเนินการให้เหมือนสถานะคล้ายเราเตอร์คือให้ข้อมูลข้ามวีแลนกันได้



### 2.5.1 สถาปัตยกรรมเครือข่ายที่ใช้สวิตช์และใช้วีแลน

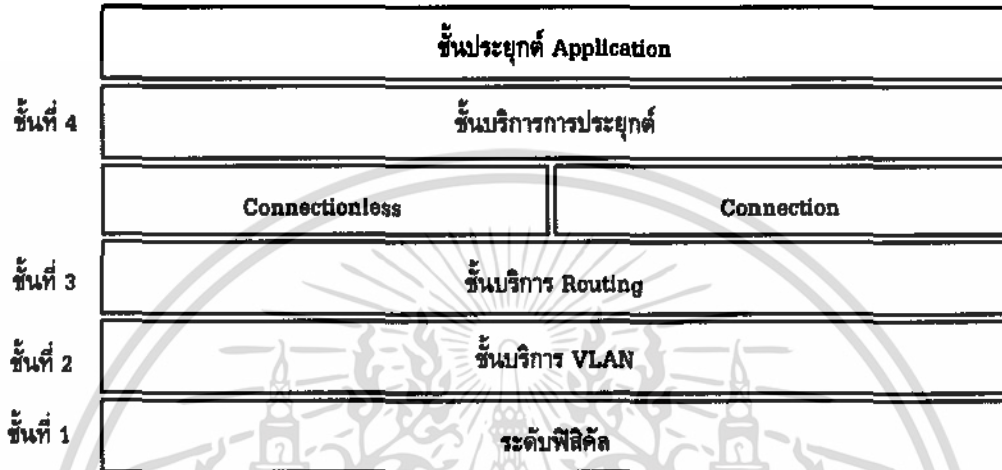
การทำงานของระบบเครือข่ายจะประสบผลสำเร็จได้ด้วยซอฟต์แวร์ที่ใช้ในการจัดการ การที่สวิตช์สามารถทำการแยกแพ็คเกจข้อมูลได้รวดเร็ว มีได้หมายความว่าสวิตช์จะประสบผลสำเร็จได้ เพราะการสวิตช์อย่างเดียวไม่เพียงพอ จำเป็นต้องมีการควบคุมการสวิตช์เพื่อส่งผ่านข้อมูลให้ถูกช่องทางดำเนินการในระดับช่วยจัดการจึงอยู่ที่ซอฟต์แวร์จัดการในระดับที่อยู่เหนือจากสวิตช์ขึ้นไป ซอฟต์แวร์จัดการเหล่านี้ทำให้สามารถเชื่อมโยงเครือข่ายระยะไกลและเครือข่ายท้องถิ่นได้ ช่วยในการปรับเปลี่ยนโครงสร้างเครือข่ายตามความต้องการการช่วยให้มีการแบ่งแถบกว้างสัญญาณแต่ละช่องตามความต้องการอย่างเหมาะสม และสนองความต้องการรวมถึงการจัดการเรื่องความปลอดภัยของข้อมูลและทำให้ลดข้อยุ่งยากในการจัดการเครือข่าย

โครงสร้างการจัดการเหล่านี้จึงต้องเป็นไปตามโครงสร้างระบบการแบ่งชั้นของเครือข่ายตามมาตรฐาน OSI ที่แบ่งระดับชั้นออกเป็น 7 ชั้น สำหรับเอทีเอ็มมีโครงสร้างการแบ่งระดับชั้นดังรูปที่ 2.10

เริ่มจากระดับชั้นแรก เป็นวิธีการเชื่อมต่อและการสวิตช์แพ็คเกจที่เป็นแพ็คเกจขนาดเล็ก

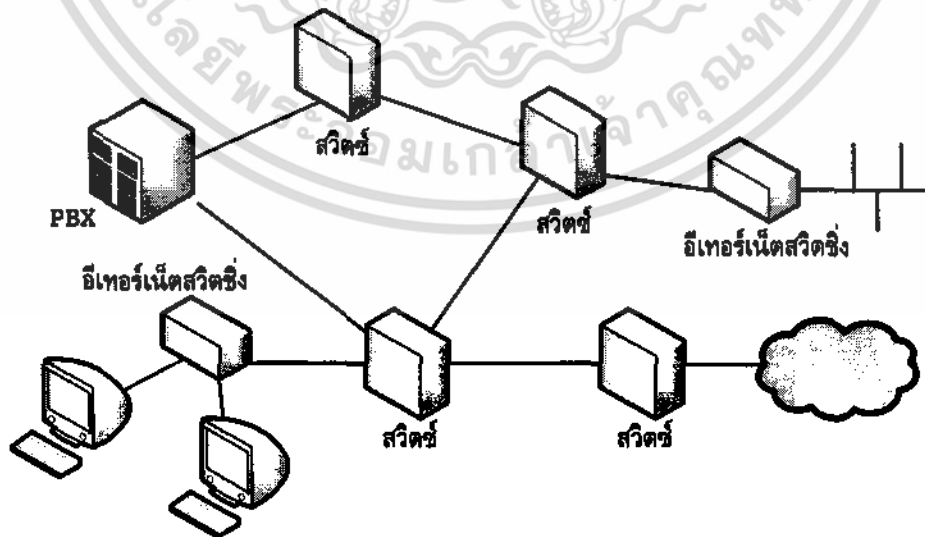
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ระดับที่ 2 เป็นระดับซอฟต์แวร์ที่ดำเนินการจัดการกระทำเครือข่ายท้องถิ่นแบบเสมือน ชั้นนี้จึงเป็นชั้นสำคัญที่ทำให้เครือข่ายมีลักษณะเหมือนเครือข่ายท้องถิ่น และใช้งานได้เหมือนใช้เครือข่ายท้องถิ่น นอกจากนี้ยังทำให้เครือข่ายเชื่อมโยงกับเราเตอร์แบบเดิม เพื่อเชื่อมต่อในระดับที่ 3 ต่อไป



รูปที่ 2.10 โครงสร้างการทำงานของเครือข่าย

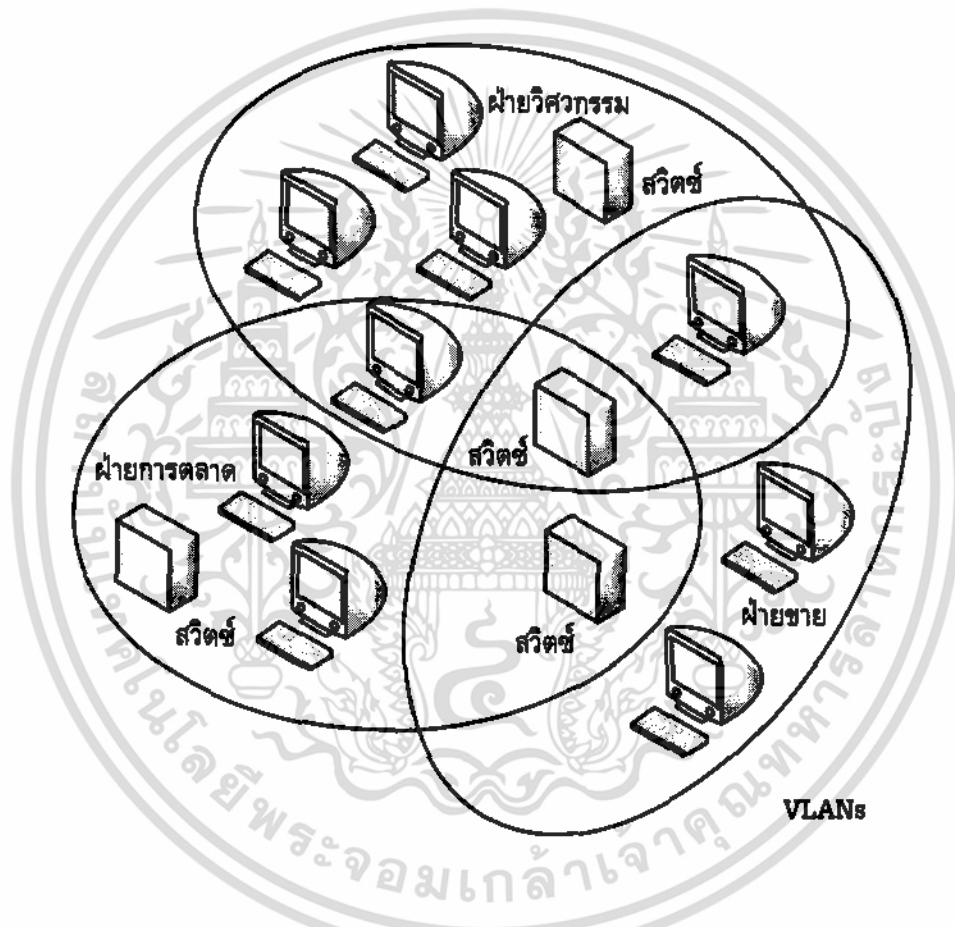
ระดับที่ 3 เป็นระดับเครือข่ายเป็นระดับที่เชื่อมโยงระหว่างวิแลนกับวิแลนหรือระบบเครือข่ายท้องถิ่นอื่นๆ ระดับนี้จึงใช้ซอฟต์แวร์ทำหน้าที่กำหนดเส้นทางฟังก์ชันการทำงานจึงเหมือนกับเครือข่ายท้องถิ่นที่เชื่อมโยงกันด้วยเราเตอร์



รูปที่ 2.11 การเชื่อมโยงระหว่างอุปกรณ์ต่างๆ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

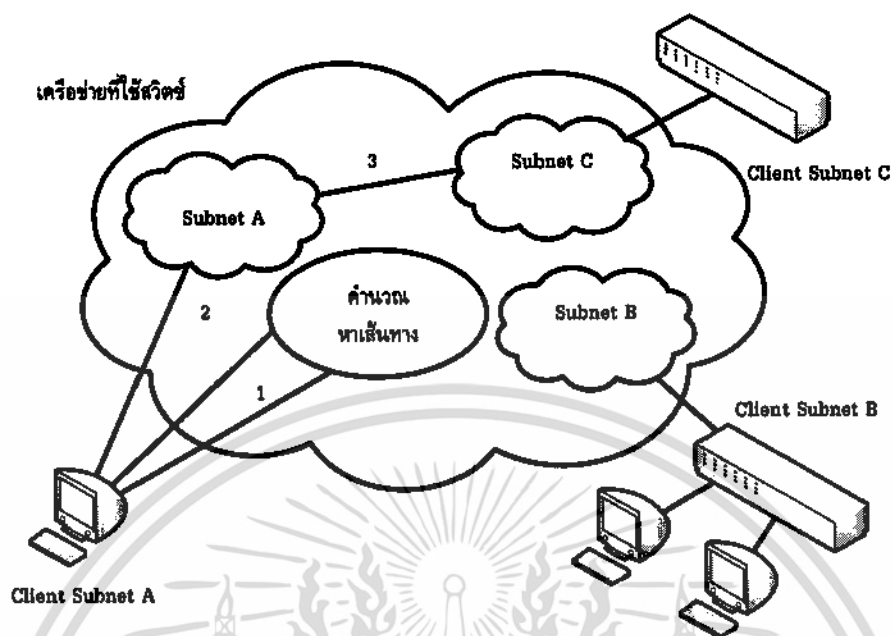
เมื่อสร้างวิแลนในระดับที่สองจะมีการจัดกลุ่มเครือข่ายเข้าด้วยกัน โดยแต่ละกลุ่มทำหน้าที่เหมือนเป็นเครือข่ายท้องถิ่นหนึ่งเครือข่าย สังเกตว่าอุปกรณ์บางตัวอยู่ในเครือข่ายย่อยวิแลน ได้หลายเครือข่ายหากพิจารณาเครือข่ายวิแลน ตามรูปแบบ TCP/IP นั้นหมายความว่า แต่ละอุปกรณ์มีตำแหน่งแอดเดรสของตนเอง ระบบซอฟต์แวร์จะทำการกำหนดตำแหน่งต่างๆ ของอุปกรณ์เข้าเป็นเครือข่ายวิแลนสามารถปรับเปลี่ยนและดูแลเครือข่ายโดยรวมได้ วิแลนจึงเป็นซอฟต์แวร์ที่ทำให้ส่วนการทำงานเหมือนชั้น LLC ในระดับเครือข่ายท้องถิ่นเดิมและสามารถสนับสนุนให้เป็นระบบเครือข่ายท้องถิ่นได้หลายรูปแบบ



รูปที่ 2.12 เป็นโครงสร้างการจัดกลุ่มอุปกรณ์ให้เข้าเป็นเครือข่ายวิแลนภายใต้เครือข่ายแบบสวิตช์

การทำงานกำหนดเส้นทางเป็นซอฟต์แวร์ในระดับที่ 3 ที่มีความสำคัญยิ่ง เพราะในระดับที่ 3 เป็นการเชื่อมวิแลนเข้าด้วยกันเป็นเครือข่าย ดังนั้นจึงต้องมีการกำหนดเส้นทางด้วยอุปกรณ์ที่จะกำหนดเส้นทางได้ถูกต้อง วิธีการหาเส้นทางนั้นจะขึ้นอยู่กับอุปกรณ์ที่จะเป็นตัวบอกให้มีการสวิตช์ตามเส้นทางใด การหาเส้นทางจึงต้องมีการสอบถามและให้ข้อมูลเพื่อดำเนินการสร้างการเชื่อมต่อของข้อมูลผ่านวิแลนได้ถูกต้อง ดังรูปที่ 2.13 เป็นวิธีการในการที่เครื่องลูกข่าย A ต้องการติดต่อกับเครื่องลูกข่าย C มีวิธีการติดต่อโดยการให้การกำหนดเส้นทางผ่านอุปกรณ์ที่ใช้ในการกำหนดเส้นทางซึ่งเป็นซอฟต์แวร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.13 วิธีการติดต่อระหว่างเครื่องลูกข่าย

ระดับที่ 4 เมื่อระดับการกำหนดเส้นทางมีความชัดเจนและทำงานได้ตามฟังก์ชันของเราเตอร์แล้ว การเชื่อมโยงแพ็คเก็ตจกระดับ IP ก็เกิดขึ้นได้ แพ็คเก็ตในระดับ IP วิ่งผ่านเข้าไปยังวิแลน และถูกตัดแบ่งเป็น เซล เมื่อสวิตช์ไปตามเส้นทางที่กำหนดได้ระดับชั้นที่เชื่อมต่อกับระดับ 3 คือ ระดับ Transport ที่จะเชื่อมโยง ต่อเข้าสู่ระดับการประยุกต์การทำงานในระดับนี้จึงเข้าไปใกล้กับการประยุกต์ใช้งานต่างๆ

## 2.5.2 ชนิดของวิแลน

### 2.5.2.1 Layer 1 VLAN : Membership by Ports

ในการแบ่งวิแลน จะใช้พอร์ตบอกว่าเป็นของวิแลนใด เช่น สมมุติว่าในสวิตช์ที่มี 4 พอร์ต กำหนดให้ พอร์ต 1, 2 และ 4 เป็นของวิแลน เบอร์ 1 และพอร์ตที่ 3 เป็นของวิแลน เบอร์ 2 ดังตารางที่ 1

ตารางที่ 2.1 การกำหนดพอร์ตให้กับวิแลน

Port	VLAN
1	1
2	1
3	2
4	1

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 2.5.2.2 Layer 2 VLAN : Membership by MAC Address

ให้ MAC Address ในการแบ่งวีแลน โดยให้สวิตช์ตรวจหา MAC Address จากแต่ละวีแลน ดังตารางที่ 2

ตารางที่ 2.2 การกำหนด MAC Address ให้กับวีแลนต่างๆ

MAC Address	VLAN
1212354145121	1
2389234873743	2
3045834758445	2
5483573475843	1

### 2.5.2.3 Layer 2 VLAN : Membership by Protocol Types

แบ่งวีแลนโดยใช้ชนิดของโปรโตคอลที่ปรากฏอยู่ในส่วนของ Layer 2 Header ดังตารางที่ 3

ตารางที่ 2.3 การแบ่งวีแลนโดยใช้ชนิดของโปรโตคอลกำหนด

Protocol	VLAN
IP	1
IPX	2

### 2.5.2.4 Layer 3 VLAN : Membership by IP Subnet Address

แบ่งวีแลนโดยใช้ Layer 3 Header นั่นก็คือใช้ IP Subnet เป็นตัวแบ่ง ดังตารางที่ 4

ตารางที่ 2.4 การแบ่งวีแลนโดยใช้ IP Subnet

IP Subnet	VLAN
23.2.24	1
26.21.35	2

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

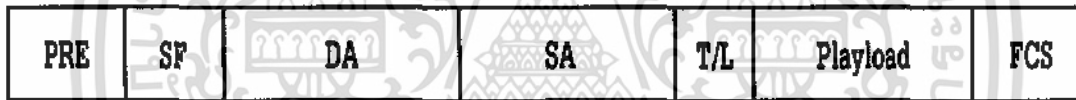
**2.5.2.5 Higher Layer VLAN's**

วีแลนทำได้โดยใช้โปรแกรมประยุกต์แบ่งวีแลน เช่น การใช้โปรแกรม FTP สามารถใช้ได้ ใน วีแลน 1 เท่านั้น และถ้าจะใช้ Telnet สามารถเรียกใช้ได้ ในวีแลน 2 เท่านั้น

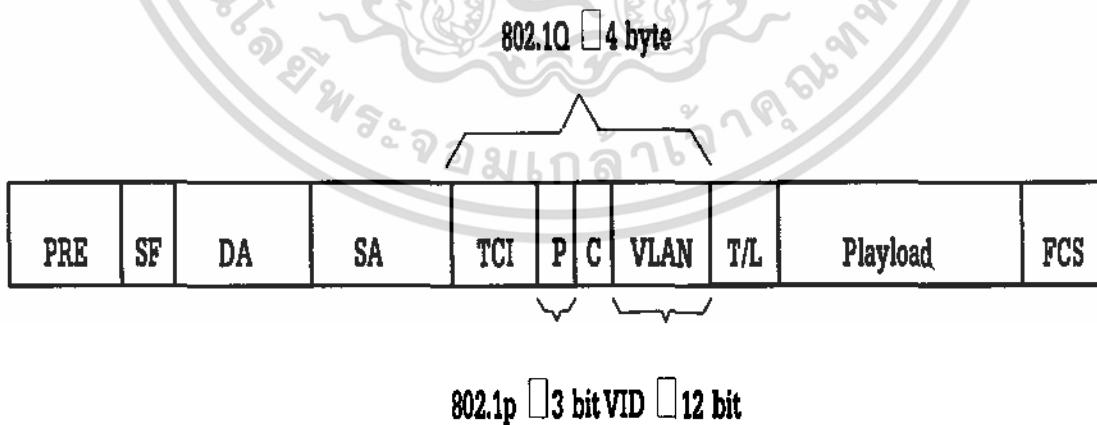
**2.5.3 มาตรฐานของวีแลน**

มาตรฐานของวีแลน คือ 802.1Q มีลักษณะเป็นมาตรฐานในการนำข้อมูลของวีแลน Membership ใส่เข้าไปใน Ethernet Frame หรือที่เรียกว่าการ Tagging และโปรโตคอล 802.1Q นี้ถูกพัฒนาเพื่อแก้ปัญหาเรื่องการบริหารจัดการด้านเครือข่ายที่เพิ่มขึ้น เช่น การกระจายเครือข่ายใหญ่ๆ ให้เป็นส่วนย่อยๆ ทำให้ไม่สูญเสียแบนวิธให้กับการ Broadcast และ Multicast มากเกินไปและยังเป็นการรักษาความปลอดภัยระหว่างส่วนย่อยต่างๆ ภายในเครือข่ายให้สูงขึ้นอีกด้วย

การต่อเติมเฟรม (Tagging Frame) ด้วยมาตรฐาน 802.1Q นั้นจะทำในระดับ Data-Link layer และการทำวีแลน Tagging นั้นจะเป็นการเปลี่ยนรูปแบบของ Ethernet Frame มาตรฐาน 802.3 ให้เป็นรูปแบบใหม่ที่เป็นมาตรฐาน 802.3 ac ซึ่งมีไคอะแกรมของเฟรมมาตรฐาน 802.3 ดังรูปที่ 2.14 และไคอะแกรมของมาตรฐาน 802.3 ac ดังรูปที่ 2.15



รูปที่ 2.14 รูปแบบของเฟรม 802.3 ก่อนที่จะทำวีแลน Tagging



รูปที่ 2.15 รูปแบบของเฟรม 802.3 ที่มีการ Tagging 802.1Q แล้ว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## ตารางที่ 2.5 คำอธิบายส่วนต่างๆ ของมาตรฐาน 802.3

Label	Field Name	Size	Description
PRE	Preamble	7 bytes	Used to synchronize traffic between nodes
SF	Start Frame Delimiter	1 bytes	Marks the beginning of the header
DA	Destination Address	6 bytes	The MAC address of the next/final hop
SA	Source Address	6 bytes	The MAC address of the source
TCI	Tag Control Info	2 bytes	When set to "8100", indicates this frame uses 802.1p and Q tags
P	Priority	3 bits	Indicates 802.1p priority level 0-7
C	Canonical Indicator	1 bit	Indicates if the MAC addresses are in canonical format Ethernet uses "0"
VLAN	VLAN Identifier (VID)	12 bits	Indicates which VLAN this frame belongs to (0-4095)
T/L	Type/Length Field	2 bytes	Ethernet II "type" or 802.3 "length" information
Payload	Payload	≤1500 bytes	User data or higher layer protocol information
FCS	Frame Check Sequence	4 bytes	Error checking on the frame's contents-also known as "CRC" (Cyclical Redundancy Check)

### 2.5.4 ช่องโหว่ของการใช้วีแลน

โดยปกติแล้วจะไม่สามารถส่งข้อมูลข้ามวีแลนได้ถ้าไม่ใช่ เราเตอร์ สวิตช์เลเยอร์สามหรือตัวกลางที่ช่วยค้นหาเส้นทางอื่นๆ แต่มีช่องโหว่ที่ทำให้ผู้ใช้สามารถส่งข้อมูลข้ามวีแลนได้โดยไม่ต้องอาศัยตัวกลาง เรียกว่า "การเบรกวีแลน" ซึ่งช่องโหว่นี้เกิดจาก Trunking Protocol ของสวิตช์บางรุ่น และวิธีการทดสอบคือ

ทำการส่งข้อมูลตัวอย่างจากวีแลนหนึ่งไปยังวีแลนอื่นที่อยู่บนสวิตช์คนละตัว และข้อมูลที่ส่งนั้นให้ทำการสร้างอีเทอร์เน็ตเฟรมที่มี Tag 802.1Q และเปลี่ยนค่าของหมายเลขวีแลน ให้เป็นค่าของหมายเลขวีแลนปลายทางที่ต้องการเบรกเฟรมที่ถูกสร้างขึ้นใหม่นั้นจะมีลักษณะดังรูปที่ 2.15 และค่าของ Tag 802.1Q จะมีรูปแบบ "81 00 0n nn" โดยที่ nnn คือหมายเลขของวีแลนซึ่งผลจากการทดสอบดังกล่าวสามารถที่จะเบรกวีแลนได้

สถานการณ์ที่จะทำให้เกิดช่องโหว่ของวีแลน เมื่อผู้บุกรุกสามารถที่จะเข้าถึงพอร์ตในสวิตช์ที่เป็นวีแลนเดียวกันกับวีแลนของ Trunk Port เครื่องเป้าหมายอยู่บนสวิตช์ต่างกันแต่มีกลุ่ม Trunk เดียวกัน ผู้บุกรุกทราบถึงเม็คแอดเดรสของเครื่องเป้าหมาย Layer 3 Device สามารถสร้าง Connection จากวีแลนเป้าหมายกลับไปยังวีแลนที่เป็นต้นทางได้

## 2.5.5 ประโยชน์ของวีแลน

### 2.5.5.1 เพิ่มประสิทธิภาพของเครือข่าย

ในระบบเครือข่ายทั่วไปจะมีการส่งข้อมูล Broadcast จำนวนมากทำให้เกิดความคับคั่ง และวีแลนมีความสามารถช่วยเพิ่มประสิทธิภาพของเครือข่ายได้เนื่องจากวีแลน จะจำกัดให้ส่งข้อมูล Broadcast ไปยังผู้ที่อยู่ในวีแลนเดียวกันเท่านั้น

### 2.5.5.2 ง่ายต่อการบริหารการใช้งาน

วีแลนอำนวยความสะดวกในการบริหารจัดการโครงสร้างของระบบเครือข่ายให้ง่าย มีความยืดหยุ่น และเสียค่าใช้จ่ายน้อย โดยเพียงเปลี่ยนโครงสร้างทางตรรกะเท่านั้น ไม่จำเป็นต้องเปลี่ยนโครงสร้างทางกายภาพ กล่าวคือ ถ้าต้องการเปลี่ยนโครงสร้างของวีแลนก็ทำโดยการคอนฟิกที่อุปกรณ์เครือข่ายใหม่ ไม่จำเป็นต้องเปลี่ยนรูปแบบทางกายภาพของการเชื่อมต่อเครือข่ายที่มีอยู่เดิม

### 2.5.5.3 เพิ่มการรักษาความปลอดภัยมากขึ้น

เนื่องจากการติดต่อระหว่างอุปกรณ์เครือข่ายจะสามารถทำได้ภายในวีแลนเดียวกันเท่านั้น ถ้าต้องการที่จะติดต่อข้ามวีแลนต้องติดต่อผ่านอุปกรณ์ค้นหาเส้นทางหรือสวิตช์เลเยอร์สาม

## 2.5.6 ข้อเสียและปัญหาที่พบของการใช้วีแลน

### 2.5.6.1 การแบ่งวีแลนแบบ Port-Based

การแบ่งวีแลนแบบ Port-Based นั้นจะมีข้อเสียเมื่อมีการเปลี่ยนพอร์ตนั้นอาจจะต้องทำการคอนฟิกวีแลนใหม่

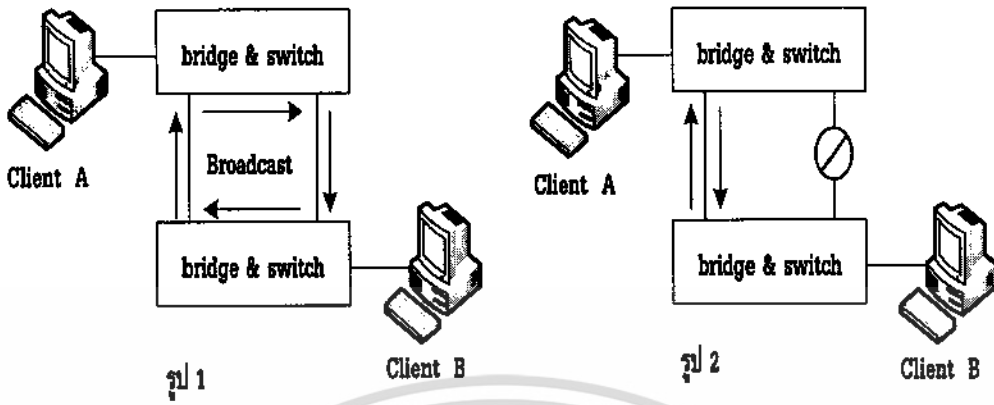
### 2.5.6.2 การแบ่งวีแลนในรูปแบบของ MAC - Based

การแบ่งวีแลนในรูปแบบของ MAC-Based นั้นจะต้องให้ค่าเริ่มต้นของวีแลน Membership ก่อน และปัญหาที่เกิดขึ้นคือในระบบเครือข่ายที่ใหญ่มาก จำนวนเครื่องนับพันเครื่อง นอกจากนี้ถ้ามีการใช้เครื่องโน้ตบุ๊กด้วย ซึ่งก็จะมีค่า MAC และเมื่อทำการเปลี่ยนพอร์ตที่ต่อก็ต้องทำการ คอนฟิกวีแลนใหม่

## 2.6 Spanning Tree

มาตรฐาน IEEE 802.1d เป็นมาตรฐานที่ว่าด้วยข้อตกลงในการสื่อสารที่มีความสามารถในการทำงานเกี่ยวกับ Spanning Tree โดยมีการบรรจุฟังก์ชันนี้ไว้ในสวิตช์ซึ่งยับยั้งที่สนับสนุนการทำงานของ IEEE 802.1d ฟังก์ชันนี้ไม่ถูกใช้งานเป็นที่แพร่หลาย เพราะว่ายังมีผู้ใช้งานจำนวนมากที่อาจยังไม่เข้าใจ

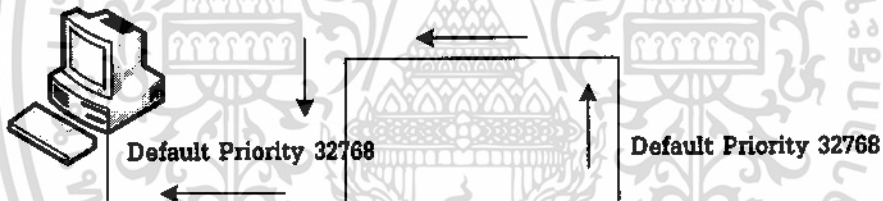
จากรูปที่ 2.16 เป็นการเชื่อมต่อสวิตช์ซึ่งยับยั้ง จำนวน 2 ตัวโดยใช้สาย UTP (CAT 5) 2 เส้น ในรูปที่ 2.16 พบว่าเครื่องคอมพิวเตอร์ที่เชื่อมต่อภายในระบบเครือข่ายนี้จะไม่สามารถใช้งานได้ เนื่องจากมี Broadcast ที่เกิดขึ้นจำนวนมากภายในระบบ เหตุการณ์นี้ถ้าสวิตช์ซึ่งยับยั้งมีการสนับสนุนการทำงานของ Spanning Tree จากรูปที่ 2.17 ระบบเครือข่ายจะสามารถทำงานได้เป็นปกติ



รูปที่ 2.16 การเชื่อมค่า Switching Hub จำนวน 2 ตัวโดยใช้สาย UTP

### 2.6.1 หลักการทำงานของ Spanning Tree

#### 2.6.1.1 สวิตชิงฮับ



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ระบบเน็ตเวิร์กแต่ละจะไม่ส่งดาต้าเฟรมเข้าไประบบ Networks Forwarding ส่งและรับ ดาต้าเฟรมทุกพอร์ตของสวิตชิง

1. ขั้นตอนในการตรวจสอบว่าสวิตชิงฮับตัวไหนเป็น Root Bridge ดูจากรูปที่ 2.18 ประกอบ Mac Address Switching Hub ค่าที่น้อยจะได้เป็น Root Bridge เสมอโดยค่า Priority จะต้องเท่ากันดูได้จากค่าของ Priority = 32768 ในกรณีที่ต้องการให้สวิตชิงฮับนั้นมีค่าเป็น Mac Address มากกว่าโดยให้เป็น Root Bridge ทำได้โดยกำหนด Priority ให้มีค่าน้อยกว่า เช่น ถ้าต้องการให้สวิตชิงฮับตัวที่ 2 เป็น Root Bridge กำหนดค่าของ Priority ให้มีค่า 32767 สวิตชิงฮับตัวที่ 2 จะมีค่าเป็น Root Bridge
2. สวิตชิงฮับที่ว่าจะเป็น Root Bridge จะให้ความสำคัญของค่า Priority ที่มีค่าน้อยเป็น Root Bridge ก่อน
3. ถ้า Priority เท่ากันจะให้ความสำคัญค่า Mac Address มีค่าน้อยเป็น Root Bridge ก่อน ประโยชน์ Spanning Tree เป็นการสำรองเส้นทางกรณีที่สายเคเบิลขาดพอร์ตสถานะที่คือ B (Blocking) เปลี่ยนเป็น Forwarding ทำให้ระบบเน็ตเวิร์กยังสามารถจะทำงานได้อยู่ถ้าอุปกรณ์สวิตชิงฮับมีปัญหาสวิตชิงฮับอีกตัวจะสามารถทำงานได้ตาม

#### 2.6.1.2 หมายเหตุ

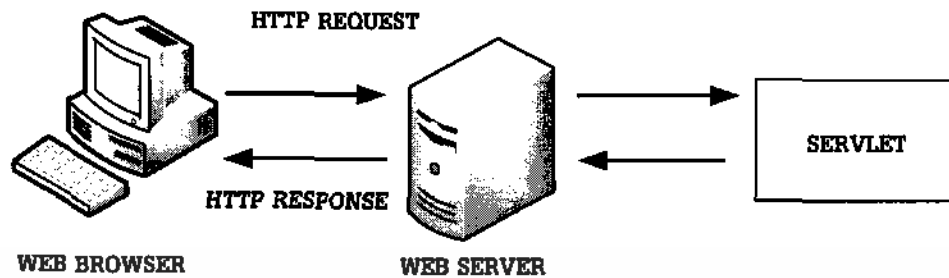
สวิตชิงฮับแต่ละพอร์ตจะมีการเปลี่ยนแปลงค่า B = Blocking, Li = Listening, Le = Learning และ F = Forwarding ค่า Priority สามารถเปลี่ยนแปลงได้ ส่วน Mac Address ไม่สามารถเปลี่ยนแปลงจากรูปที่ 2.19

## 2.7 Java Servlet

Java Servlet คือ โปรแกรมภาษาจาวาที่ถูกสร้างมาให้ทำงานได้บน HTTP Server โดยการทำงานของ Servlet มีลักษณะเป็น Server-Side ซึ่งต่างจาก Java Applet โดยทั่วไปที่ทำงานในลักษณะของ Client-Side โดย Servlet เป็น Modules ที่ทำงานอยู่ภายในกระบวนการ Request/Response Oriented Server และ Servlet ยังสามารถตอบสนองในการส่งข้อมูลโดยให้อยู่ในรูปของภาษา HTML โดย Servlet ยังสามารถนำมาประยุกต์ใช้ในการพัฒนาฐานข้อมูลต่างๆ ได้อีกด้วย

ประโยชน์ของ Servlet นั้นช่วยให้เราเขียนโปรแกรมที่สั่งให้ HTTP Server ทำงานหรือโต้ตอบกับโปรแกรมของเครื่องลูกข่ายเหมือนที่ CGI (Common Gateway Interface) ทำได้ซึ่ง Servlet ถูกนำมาใช้แทน CGI Scripts ได้อย่างมีประสิทธิภาพและให้ความสะดวกได้อย่างมากคือสามารถเขียนได้ง่ายกว่าและทำการประมวลผลได้เร็วกว่า CGI และ Servlet ยังได้มีการจัดการในส่วนของโปรแกรมทางด้าน Server-Side ไว้โดยเฉพาะ Servlet ได้ถูกพัฒนาโดยใช้ Java Servlet API ซึ่งเป็นส่วนเพิ่มเติมมาตรฐานของ Java

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.18 การทำงานของ Servlet

### 2.7.1 จุดเด่นของ Servlet

1. Servlet สามารถที่จะฝังตัวทำงานอยู่บนเซิร์ฟเวอร์เมื่อทำการให้บริการ Client เสร็จสิ้นแล้ว จะคอยรับ Request ใหม่ที่อาจจะมาจากเครื่องลูกข่ายอีกโดยไม่ต้องถูกสร้างขึ้นใหม่จึงทำให้ Servlet ใช้ทรัพยากรของเซิร์ฟเวอร์น้อยมาก
2. Servlet สามารถจัดการงานประเภท Multiple Connection ได้ดีโดยอาศัยความสามารถของ Thread ที่มีอยู่ในตัวจาวา
3. Servlet สามารถติดต่อกับ Applet บนเครื่องลูกข่ายอย่างต่อเนื่องเป็นผลให้การรับส่งข้อมูลระหว่างการดำเนินเป็นไปอย่างต่อเนื่อง
4. Servlet สามารถถูก Upload จากเครื่องลูกข่ายเพื่อส่งให้ไปทำงานบนเซิร์ฟเวอร์ใดๆ บนเน็ตเวิร์กได้ประโยชน์คือเราสามารถเขียนโปรแกรมเพื่อค้นหาข้อมูลใน Web Host ใดๆ โดยไม่ต้องส่งข้อมูลมาประมวลผลที่เครื่องต้นทาง สามารถทำได้ที่ตัว Host นั้นๆ เลยเสร็จแล้วค่อยส่งผลกลับคืนจึงสามารถลดเวลาและทรัพยากรบนระบบเน็ตเวิร์กลงอย่างมาก

### 2.7.2 สถาปัตยกรรมของ Servlet

ใจความสำคัญของ Servlet API คือ การสนับสนุนการ Interface ของ Servlet ทั้งแบบทางตรงหรือทางอ้อมโดยประกอบไปด้วยคลาสที่สนับสนุนการ Interface เช่น Http Servlet ซึ่ง Servlet Interface ประกอบด้วย Method ที่ใช้เป็นตัวจัดการ Servlet และติดต่อสื่อสารกับเครื่องลูกข่ายผู้เขียน Servlet จะกำหนด Method นี้ทั้งหมดหรือบางส่วนเมื่อต้องการพัฒนา Servlet เมื่อ Servlet ได้รับการร้องเรียกจากเครื่องลูกข่ายจะมีการรับ Object อยู่ 2 Object ซึ่งอันแรกคือ Servlet Request และอีกอันคือ Servlet Response โดยคลาส Servlet Request นั้นจะเป็นตัวติดต่อสื่อสารจากเครื่องลูกข่ายไปยังเซิร์ฟเวอร์ในขณะที่คลาส Servlet Response เป็นตัวติดต่อสื่อสารจาก Servlet กลับไปยังเครื่องลูกข่าย

Servlet Request Interface อนุญาตให้ Servlet สามารถเข้าถึงข้อมูลได้ เช่น ชื่อของพารามิเตอร์ที่ผ่านเข้ามาโดยเครื่องลูกข่ายโปรโตคอล และชื่อ Host ของเครื่องลูกข่ายที่เป็นตัวส่ง Request และของเอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น เมื่ออนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เซิร์ฟเวอร์ที่เป็นตัวรับ Request ด้วยรวมไปถึงการกำหนดให้ Servlet เข้าถึง Input Stream ได้ด้วยโดย Servlet Input Stream โดย Servlet ทำหน้าที่รับข้อมูลจากเครื่องลูกข่ายโดยผ่านทางโปรโตคอล เช่น HTTP, POST และ PUT Method โดยคลาสย่อยของคลาส Servlet Request จะอนุญาตให้ Servlet สามารถรับข้อมูลที่เป็นลักษณะเฉพาะของโปรโตคอลกลับมาได้

Servlet Response Interface ประกอบด้วย Method สำหรับการตอบกลับไปยังเครื่องลูกข่ายโดยอนุญาตให้ Servlet กำหนดขนาด และ Mime Type ในการตอบกลับและเป็นตัวจัดเตรียม Output Stream ด้วยโดยผู้เขียน Servlet สามารถใช้ Servlet Output Stream ในการส่งข้อมูลกลับได้ Subclass ของ Servlet Response ทำให้ Servlet มีโปรโตคอลที่แตกต่างกันได้หลายชนิด ตัวอย่างเช่น Http Servlet Response ประกอบด้วย Method ที่อนุญาตให้ Servlet สามารถถ่ายเทข้อความจาก HTTP-Specific Header ได้

### 2.7.3 การใช้ Java Servlets กับการพัฒนาระบบงานบนเครือข่ายอินเทอร์เน็ต Servlet API

#### 2.7.3.1 Package : java.servlet

Java Servlet เป็นแพ็คเกจที่กำหนดการทำงานของ Servlet จากสัณฐานของ Protocol Independent ในระดับนี้โปรโตคอลที่ใช้ในการเข้าถึง Servlet ได้แก่ HTTP, โปรโตคอลของระบบไฟล์ เครือข่ายหรือโปรโตคอลที่ใช้ในการขนส่งอื่นๆ Servlet ถูกออกแบบให้ทำงานบน Server-Side โดยใช้คลาสในแพ็คเกจนี้ซึ่งประกอบไปด้วย Interfaces Class 4 คลาส และ Implementation Class 3 คลาสดังนี้

1. Servlet Interface Class ประกอบไปด้วย Method พื้นฐาน 4 Method ดังนี้คือ
  - 1.1 init(ServletStub) ถูกเรียกโดยอัตโนมัติเมื่อระบบต้องการหาค่าเริ่มต้นของ Servlet โดยกำหนด ตัวแปรเริ่มต้น Servlet ในไฟล์ servlet.properties ได้โดยการเรียก getniParameter()หรือgetInitParameters()
  - 1.2 destroy() ถูกเรียกโดยระบบเพื่อคืนทรัพยากรให้ระบบเมื่อ Servlet ถูกยกเลิกการใช้งาน Service (ServletRequest, Servlet Response) ถูกเรียกมาใช้ในการบวนการรับ Request ที่เข้ามาและสร้าง Response ตอบกลับไป
  - 1.3 getServletInfo() ถูกเรียกเพื่อรับข้อมูล (Information) ที่เป็นสตริงซึ่งบอกลักษณะของ Servlet
2. Servlet Context Interface Class เป็นการกำหนด Method เพื่อให้ Servlet สามารถจะทำงานแบบสลับฉาก (Interact) ได้โดย Method ต่างๆ มีดังนี้
  - 2.1 getMime Type(String filename) ส่งคืน MIME Type ของแต่ละไฟล์โดยใส่ชื่อของไฟล์ (filename) ลงใน mime.properties file

- 2.2 `getRealPath (String Path)` การประยุกต์ Alias Rules ใน `alias.properties` file เพื่อเป็นการกำหนด Virtual Path และตอบสนองด้วยการส่ง Real Path หากการส่งข้อมูลเกิดการผิดพลาดค่าที่ส่งกลับคืน Null
- 2.3 `getServerInfo()` ส่งคืนค่าที่เป็นสตริงที่บ่งบอกชื่อ Version และฐานของ Server Software
- 2.4 `getServlet (String)` ส่งคืน Servlet ที่ตรงกับชื่อที่เรียกมา
- 2.5 `getServlets()` ส่งคืน Servlets ทั้งหมดที่ถูกโหลดในเซิร์ฟเวอร์โดยรวมไปถึง Servlet ที่ถูกเรียกด้วย
- 2.6 `log(Servlet, String)` เขียนข้อความลงใน Servlet Log ของ Servlet ที่ถูกกำหนด
3. Servlet Request Interface Class ประกอบไปด้วย Method ที่ใช้สำหรับค้นหาข้อมูลที่เกี่ยวข้องกับ Servlet Request ซึ่งเป็นอิสระจากโปรโตคอลที่ใช้อยู่จริงในการส่งผ่านข้อมูลระหว่างเครื่องลูกข่ายและเซิร์ฟเวอร์โดย Request จะถูกนำพาโดย IP พื้นฐานของโปรโตคอลเช่น HTTP ซึ่งการส่งผ่านข้อมูลของ ServletRequest โดยอาศัยโปรโตคอลเช่น HTTP นั้นจะต้องมี Method ที่มีความสัมพันธ์กับโปรโตคอลดังนี้
  - 3.1 `getRemoteAddr()` ส่งคืนค่า IP Address ของเครื่องลูกข่ายที่ส่ง Request
  - 3.2 `getRemoteHost()` ส่งคืน Host Name ของตัวแทนที่ส่ง Request
  - 3.3 `getServerName()` ส่งคืน Host Name ของเซิร์ฟเวอร์ที่รับ Request
  - 3.4 `getServerPort()` ส่งคืนค่า Port Number ที่รับ Request
  - 3.5 `getProtocol()` ส่งคืนค่าที่เป็น String ซึ่งแสดง Protocol และ Version
4. Request ประกอบไปด้วยพารามิเตอร์และข้อมูลต่างๆ โดย Servlet สามารถหาขนาดความยาวของ Content ได้โดย `getContentLength()` และสามารถหาชนิดของ Content จากที่ `getcontentType()` จากการ Content Servlet สามารถอ่านได้โดยใช้ `HttpInputStream` ซึ่งได้จาก `getInputStream()` โดย Method ที่มีความสัมพันธ์กับ Content มีดังนี้
  - 4.1 `getParameter(String)` ส่งคืนพารามิเตอร์ที่กำหนดมาสำหรับ Request
  - 4.2 `getParameters()` ส่งคืนพารามิเตอร์ทั้งหมดของ Request
  - 4.3 `getContentType()` ส่งคืน MIME Type ของข้อมูลที่รวมมากับ Request หรือค่า Null ถ้าไม่มีกำหนด MIME Type มากับ Request
  - 4.4 `getInputStream()` ส่งคืน Input Stream เพื่ออ่านข้อมูลที่มากับ Request
5. Servlet Reaponse Interface Class คือตัวกำหนดการสร้าง Response สำหรับตอบสนอง Request โดยทั่วไปแล้วนั้น Response จะประกอบไปด้วยส่วนที่บอกลักษณะของ MIME และขนาดของ Content โดยการเขียน Content เข้าไปใน Response นั้นโดย Servlet จะ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เรียก `getOutputStream()` เพื่อทำหน้าที่รับ `ServletOutputStream` จากนั้นเรียก `print()` เพื่อส่ง `Response` โดย Method ที่ใช้ในการสร้าง `Response` มีดังนี้

5.1 `setContentType(String)` เซ็ท MIME Type ของ `Response`

5.2 `setContentLength(int)` เซ็ทขนาดความยาวของ `Response`

5.3 `getOutputStream()` การส่งคืน `ServletOutputStream` เพื่อจะทำการเขียนข้อมูล `Response` ให้กลับไปยังเครื่องลูกข่าย

6. `GenericServlet Class` เป็นคลาสพื้นฐานที่สนับสนุนการส่งผ่านข้อมูลของ `Servlet` และใช้สำหรับการปลุก `Servlet` จาก `Server-Side Include` โดยประกอบด้วย Method ดังนี้

6.1 `init()` ถูกเรียกหลังจาก `Servlet` ได้รับการโหลดให้เป็นชั้นเริ่มต้น โดยจะข้าม Method นี้ไปได้เมื่อมีการรับพารามิเตอร์เริ่มต้นต่างๆ หรือมีการทำงานที่ฟังก์ชันอื่นๆ ซึ่งในแต่ละครั้งเมื่อ `Servlet` ถูกโหลด `Servlet` จะมีค่าอยู่ที่ค่าเริ่มต้นหนึ่งครั้งเท่านั้น

6.2 `getInitParameter (String Name)` รับค่าพารามิเตอร์เริ่มต้นของ `Servlet` ทั้งแบบ `servet.properties file` และอยู่ในรูปของ `<Servlet> Tag` สำหรับส่วน `Server - Side Includes Servlets`

6.3 `GetInitParameters()` ส่งคืนชื่อและค่าพารามิเตอร์เริ่มต้นของ `<FOCUS01>`

6.4 `GetServletContext()` ส่งคืน `Servlet Context`

6.5 `Log(String)log` ข้อความลงใน `Servlet Log File`

7. `ServletInputStream` เป็น subclass ของ `InputStream` ที่กำหนด `reasLine(bte|l, int offset, int count)` Method ซึ่งทำการอ่านไบต์ลงใน `Byte Array` จนกระทั่งอาร์เรย์เต็ม หรือมีการอ่านบรรทัดใหม่เข้ามา

8. `ServletOutputStream Abstract Class` เป็น Subclass ของ `OutputStream` ซึ่งคล้ายกับ `PrintStream` แต่เป็นเพียงการกำหนด Method ที่ใช้ Print ข้อมูลโดยทั่วไปชนิดของข้อมูลจะใช้ `Ints Longs` และ `Strings` การทำงานของ `ServletOutputStream` จะทำการเปลี่ยนโค้ดที่ใช้ในโปรแกรม (Internal) ให้เป็น `US-ASCII` ซึ่งเป็น Code ที่ใช้กันทั่วไปในระบบ `Internet Protocol` โดยประกอบด้วย Method ดังนี้

8.1 `print(int), println (int)`

8.2 `Print(long), println (long)`

8.3 `Println (String), println (String)`

8.4 `Println()`

8.5 โดยสามารถแปลงข้อมูลชนิดอื่นก่อนการสั่ง Print ออกทาง `ServletOutputStream` ได้โดยใช้ `toString()`

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 2.7.3.2 Package : javax.servlet.http

Java.servlet.http เป็นแพ็คเกจที่สนับสนุน Servlet ในการใช้ HTTP โดยเฉพาะแพ็คเกจนี้ ประกอบไปด้วย Interface Class 2 คลาส Implementation Class 2 คลาส และคลาสสนับสนุนที่เป็น Subclass ดังนี้

1. HttpServlet Abstract Class เป็นเครื่องมือที่ใช้ในการค้นหาและแปลงข้อมูลให้อยู่ระหว่าง ServletRequest และ ServletResponse โดยประกอบด้วย Method ดังนี้
  - 1.1 service(ServletRequest, ServletResponse) ทำหน้าที่ค้นหาและแปลงข้อมูลทุกอย่างไประหว่าง HTTP-specific Request กับ response โดยทั่วไปแล้วจะไม่มีกรเขียนทับ Method นี้
  - 1.2 service(HttpServletRequest, HttpServletResponse) สามารถที่จะให้เขียนทับ abstract Method เพื่อรับ HTTP Service Request
2. HttpServletRequest Interface Class เป็น Subclass ของ ServletRequest ซึ่งเป็นตัวจัดการข้อมูลของ Extra Request และแสดงลักษณะของ Content โดยทำการแบ่งประเภทของข้อมูลที่ได้จาก HttpServletRequest เป็น 4 ลำดับดังนี้
  - 2.1 General Request Information ประกอบด้วย Method ที่ให้ข้อมูลทั่วไปเกี่ยวกับ Request รวมไปถึงชื่อของ Method ใช้ในการเข้าถึงข้อมูลต่างๆ มีดังนี้คือ
    - 2.1.1 getMethod() ส่งคืนค่าสตริงที่บรรจุ Method ใน Request อย่างเช่น "GET", "HEAD" หรือ "POST"
    - 2.1.2 getQueryString() ส่งคืน Query String ซึ่งเป็นส่วนหนึ่งของ URI หรือถ้าไม่มีจะส่งค่า Null โดย Query String จะตามหลังเครื่องหมาย? ใน URL
3. URL Information ประกอบด้วย Method ที่ให้ข้อมูลเกี่ยวกับการร้องขอ URL ดังนี้ คือ
  - 3.1 getRequestURL() ส่งคืน Request URL ส่วนที่อ้างอิงถึงการปลุก Servlet
    - 3.1.1 getServletPath() ส่งคืน Request URL ส่วนที่อ้างอิงถึงการปลุก Servlet
    - 3.1.2 getPathInfo() ส่งคืนข้อมูลของ Extra Path ที่ตามหลัง Servlet Path ซึ่งนำหน้า Query String
    - 3.1.3 getPathTranslated() ส่งข้อมูลคืน Extra Path ที่จะส่งผ่านไปยัง Real Path โดยใช้ alias.properties rule
    - 3.1.4 getRequestPath() ส่งคืน Request URL ส่วนที่ตอบสนอง Servlet Path ที่เพิ่มเติมด้วยข้อมูลของ Extra Path
4. Authentication-Related Information จะเป็นการเข้าถึงข้อมูลส่วนของชื่อและที่อยู่ของเครื่องลูกข่าย Servlet สามารถที่จะทำได้โดยใช้ getHostName() และ getHostAddress()

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Method แต่สำหรับชนิดของ HTTP Level Authentication ที่จะสามารถแสดงได้โดยใช้ `getAuthType()` และจะติดต่อบน Basic Mode โดยจะมีการส่งคืนชื่อของผู้ใช้โดยการใช้ `getRemoteUser()` Method

4.1 `getAuthType()` ส่งคืนรูปแบบของการรับรอง (Authentication) ของ Request ในกรณีที่หาไม่พบค่าที่ส่งคืนคือ Null

4.2 `getRemoteUser()` ส่งคืนชื่อของผู้ใช้ที่สร้าง Request หรือถ้าไม่มีชื่อมากับ Request นั้นจะให้ค่าเป็น Null

5. Access To HTTP Request Header Fields สำหรับ `HttpServletRequest` เพื่อที่จะเปิดช่องทางให้มีการเข้าถึง HTTP Request Header Fields โดยตรงเพื่อจะได้รับข้อมูลจากการสนับสนุนเล็กน้อย `HttpServletRequest` Method เช่น Include User Agent, Referrer, Preferred Languages และ Cookie (สถานะของเครื่องลูกข่าย) ซึ่งว่าสิ่งเหล่านี้มี Header Fields ที่ไม่แน่นอน

6. Method 2 Method ที่ใช้ในการเข้าถึง Header เริ่มต้นเพื่อที่จะใส่ค่าดัชนีให้กับ Header นั้นโดยค่าดัชนีที่เป็น 0 จะส่งคืน Header Field อันดับแรกและถ้าค่าดัชนีมีค่ามากกว่าจำนวนของฟิลด์ที่เป็นไปได้ค่าที่ส่งคืนคือ Null

6.1 `getHeader(int)` ส่งคืนจำนวน Header Field (n) ถ้ามีค่าน้อยกว่า Header Field (n) ค่าที่ส่งคืนคือ Null

6.2 `getHeaderName(int)` ส่งคืนชื่อของจำนวน (n) ถ้าค่าน้อยกว่า (n) ค่าที่ส่งกลับคือ Null

7. ในกรณีต่อไปเป็นการค้นหา Request Header ตรงกันกับชื่อของ Header Field และการส่งคืนค่าของ Header Field นั้นกลับไป โดยเลือกใช้ Method ที่กำหนดให้ต่อไปนี้ที่จะขึ้นอยู่กับรูปแบบประเภทของข้อมูลที่ต้องการจะได้กลับคืนมา

7.1 `getHeader(String fieldname)` ส่งคืนค่าของ Header Field ถ้าไม่มีการกำหนดใน Method นี้ค่าที่ส่งคืนคือ Null

7.2 `getIntHeader(String fieldname, int default)` ส่งคืนค่าที่เป็น Integer เป็นค่าของ Header Field หลังจากมีการแปลงข้อมูลที่เป็น String ให้เป็น Int ถ้าไม่พบฟิลด์นี้ค่าที่ส่งคืนจะไม่ได้ถูกกำหนดไว้

7.3 `getDateHeader(String fieldname, long default)` ส่งคืนค่าวันเดือนปีของ Header Field หลังจากแปลงข้อมูลที่เป็น String ให้เป็นค่าแล้วถ้าไม่พบฟิลด์นี้จะไม่มีการกำหนดค่าที่ส่งกลับ

8. HttpServletResponse Interface Class เป็น Subclass ของ ServletResponse และจะเป็นเครื่องมือในการสร้าง HTTP Response Header แบ่งตามการนำไปใช้ประโยชน์ได้ 4 แบบ
- 8.1 Setting The Response Status เป็นการเจ็ดสถานะของ Response โดยค่าที่ส่งคืนคือ 200 หรือ OK ถ้ามี Error ส่งคืนหรือ OK อยู่ในตำแหน่งที่ไม่ถูกต้อง Servlet สามารถระบุแยกรายละเอียดของทั้ง Error Code และข้อความที่บอกถึงลักษณะของ Response ได้เองถ้ามีเพียง Error เกิดขึ้นเท่านั้น Servlet จะแปลงโค้ดให้อยู่ในรูปข้อความที่เป็น String แทนซึ่ง Method ต่างๆ มีดังนี้
- 8.2 setStatus(status Code) เจ็ด Status Code ในการส่งคืน HTTP Header โดยการ Status Code ถูกแปลงให้เป็นข้อความที่เป็น String
- 8.3 setStatus(statusCode, messageString) เจ็ด Status Code ในการส่งคืน HTTP Header
9. Specifying Non-Standard Header Fields การรวม Field ต่างๆ ที่เข้ากับ Response Header นั้น Servlet กำหนดให้ใช้ setHeader() Method แต่ถ้าสำหรับ On = Standard Headerfield ซึ่งไม่ได้รับการสนับสนุนจาก HttpServletResponse ซึ่งกำหนดให้ใช้ชื่อของ headerfield (ไม่มีเครื่องหมาย ;) อยู่ที่ argument แรก และข้อมูลของ Argument จะอยู่ที่ 2 setHeader ที่ต่างกันนั้นแยกด้วยชนิดของพารามิเตอร์เป็น String Int และ Date โดยประกอบด้วย Method ดังนี้
- 9.1 setHeader(String fieldname, String) เจ็ดค่าของชื่อ Header Field เพื่อจะแสดงค่าออกมา
- 9.2 setIntHeader(String fieldname,int) เจ็ดค่าของชื่อ Header Field เป็น Integer หลังจากมีการแปลงค่าให้เป็น String
- 9.3 setDateHeader(String fieldname, long) เจ็ดค่าของชื่อ Date Field หลังจากที่ได้ทำการแปลง Long เป็น Internet-standard Date-Time String
- 9.4 unsetHeader(Stringfieldname) ย้ายชื่อของ Header Field ออกจาก Response
10. Returning Error Response การสร้างและส่ง Standard Error Response เพื่อการส่งคืน Error นั้นมี Method m ทำหน้าที่อยู่ภายใต้ HttpServletResponse คือ sendError() Method ซึ่งประกอบด้วย Statuscod และข้อความที่อธิบายถึงรายละเอียด ของ Error ซึ่งการส่งค่าคืนนี้ที่ส่งไปบน Page โดยประกอบด้วย Header ซึ่งแสดงค่าของ Status Code และข้อความมาตรฐานของ Status Code และในขณะที่ข้อความที่อธิบายรายละเอียดของ Error นั้นจะอยู่บนส่วน Body ของ Page ที่ส่งคืนตัวอย่างเช่น เมื่อมีการเรียก sendError

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

(404, "Couldn't find it!") จะมีการสร้าง Page ซึ่งประกอบด้วย Header "404 Not Found" ข้อความคือ "Couldn't Find It!" โดย Method ที่กล่าวถึงมีดังนี้

10.1 `sendError(int)` ส่ง Error Response ไปที่เครื่องลูกข่ายโดยระบุแต่ Status Code และไม่มีข้อความเพิ่มเติม

10.2 `sendError(int, String)` ส่ง Error Response ไปที่โดยระบุทั้ง Status Code และข้อความที่อธิบายรายละเอียดของ Error

11. Returning Redirect Response การสร้างและส่ง Standard Redirect Response นั้น `HttpServletResponse` ได้กำหนดให้ใช้ `sendRedirect()` Method โดยจะมีการสร้างและส่งค่า "302" เป็น Response ค่าที่ส่งไปจะเป็น String ซึ่งเป็นตัวแสดงตำแหน่งใหม่ของ Page โดย String ที่กล่าวถึงก็คือ URL นั้นเองเนื่องจาก URL คือที่อยู่ที่ต้องการส่ง Page ไป เหมือนกับการเชื่อมโยงนั่นเอง `sendRedirect()` ไม่สามารถตรวจสอบได้ว่าค่า String ที่ เข้ามานั้นตรงกับ URL ที่ถูกต้องหรือไม่ถ้าผู้เขียนมี URL ที่ถูกต้องแล้วให้แปลง URL นั้นเป็น String โดยใช้ `toString()` ก่อนที่จะเรียก `sendRedirect()` โดยการเรียก Method นี้ จะอยู่ในรูป `sendRedirect(String)` ซึ่งจะส่ง Redirect Response ไปยังเครื่องลูกข่าย โดยการระบุที่อยู่ของ URL ที่ต้องการมาด้วย

12. `FormServlet Class` เป็นคลาสของ Servlet ที่ถูกระบุให้เป็นส่วนหนึ่งของ Servlet API และด้วยเหตุนี้มันจึงสามารถอยู่บนเว็บเซิร์ฟเวอร์ทุกชนิดที่สามารถจะให้สนับสนุน Servlet โดยทำหน้าที่สร้างแบบฟอร์มการทำงานของ Servlet โดยที่ Subclass ของ `FormServlet` และเขียนทับบน `sendResponse()` Method ซึ่ง Method นี้จะคอยทำหน้าที่ควบคุมแบบฟอร์มของข้อความและสร้าง Response ที่เป็นการ Subclass ของ `HttpServlet` โดยที่ว่า Method ต่อไปนี้ถูกกำหนดไว้ใน `FormServlet Class`

12.1 `getServletInfo()` เขียนทับ Method เพื่อส่งคืน String บรรจุนอยู่ในข้อความของ Servlet เช่น ชื่อผู้เขียนหรือเวอร์ชัน

12.2 `service(HttpServletRequest, HttpServletResponse)` ผู้เขียนสามารถเขียนทับ Method นี้เมื่อมีความต้องการ SpecialRequest Processing `sendResponse(HttpServletRequest, Hashtable)` เขียนทับ Method นี้เมื่อต้องการแปลง Code เพื่อจัดรูปแบบและสร้าง Response

12.3 `HttpUtils Class` เป็นคลาสที่ทำหน้าที่กำหนด Static Utility Routine ในการที่จะสร้าง Servlet

### 2.7.3.3 Package : java.servlet.html

แพ็คเกจลำดับที่ 3 ซึ่งกำหนดใน Servlet API คือ java.servlet.html โดยแพ็คเกจนี้ประกอบด้วย คลาสที่สนับสนุน HTML 13 คลาสซึ่งสนับสนุนการปฏิบัติการของ HTML เช่น ตาราง, การแทรกรูปภาพ และฟอร์มผู้เขียนจึงสามารถสร้าง HTML Page โดยใช้ ภาษา HTML จากนั้นจึงส่ง HTML Page นั้นเขียน ลง ServletOutputStream โดยคลาสต่างๆ มีดังนี้

1. HtmlElement สนับสนุนการกำหนด Basic Interface ประกอบด้วย 2 Method ดังนี้
  - 1.1 wrap() สำหรับรวม HTML Tag Pairs (เช่น <h1> และ </h1>) เข้ากับ Element โดย Tag ถูกระบุให้เป็น Argument ใน wrap()
  - 1.2 write() เขียน HTML ไปที่ OutputStream
2. HtmlText สนับสนุนข้อความ (Text) ที่อยู่ภายใน Tag ประกอบด้วย 2 Method ดังนี้
  - 2.1 add(String), add(String tag) เพิ่มข้อความ (Text) ซึ่งรวมไปถึงข้อความอยู่ใน Tag
  - 2.2 addTag(String), addTag (String, String) เพิ่ม Tag เดี่ยวที่ไม่เป็น Tag คู่ให้อยู่ใน (Non-Paired Tag) เช่น <p>
2. HtmlContainer Class สนับสนุน List ของ HtmlElement ที่จะอยู่ภายในของ Tags ซึ่ง HtmlContainer ประกอบด้วย Beginning Tags, List ของ HtmlElement และ Ending Tags Htmlcontainer สนับสนุน wrap(), write() และชนิดต่างๆ ของ add() และของที่ addTag() Method โดย HtmlContainer ประกอบด้วย Method ต่างๆ ดังนี้
  - 3.1 addImg (String) เพิ่ม <IMG> Tag โดยที่ String คือ URL ที่บรรจุรูปที่ต้องการ
  - 3.2 addLink (String text, String URL) เพิ่มการเชื่อมโยงเมื่อ "Text" อยู่ระหว่าง Tags และ URL โดยเป็น URL ที่ต้องการจะเชื่อมโยงไป
4. HtmlApplet Class จะสนับสนุน Applet HTML Tag กับรายละเอียดของความกว้างและความสูง
5. HtmlChoicd Class สนับสนุน HTML Tag ที่ถูกเลือก
6. HtmlDefinitionList Class สนับสนุน dl dt และdd HTML Tag
7. HtmlForm Class สนับสนุนการสร้าง HTML Form ที่ประกอบด้วย Input Fields, Check Boxes, Radio Buttons, Test Area, Selections และ Submit Buttons
8. HtmlFramest Class สนับสนุน Client-Side Image Map HTML Tag
9. Html ImageMap Class สนับสนุน Client-Side Image Map HTML Tags
10. HtmlList Class สนับสนุน ul และ ol HTML List Tag
11. HtmlPages Class สนับสนุนการสร้าง Web Page ที่ประกอบด้วย Header และ Body Areas

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

12. HTMLTable Class สนับสนุนการสร้างตารางจาก Row Element

13. HtmlRow Class สนับสนุน HTML เพื่อกำหนดแถวของตาราง



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## บทที่ 3

### การออกแบบ การสร้าง และการทำงาน

#### 3.1 กล่าวนำ

การออกแบบและการสร้างโปรแกรมทดลองเครือข่ายท้องถิ่นแบบเสมือนบนเครือข่ายอินเทอร์เน็ต แบ่งออกเป็น 2 ส่วนคือ ส่วนของโปรแกรมที่อยู่ในเครื่องเซิร์ฟเวอร์และส่วนของเครื่องคอมพิวเตอร์ที่เป็นเครื่องของผู้ใช้งาน ในส่วนของโปรแกรมที่อยู่ในเครื่องเซิร์ฟเวอร์นั้นจะเก็บโปรแกรมที่คอยตอบสนองเวลาที่เครื่องของผู้ใช้งานติดต่อมาเพื่อทำการตั้งค่าให้กับสวิตช์ โดยเป็นตัวจัดการรับ-ส่งค่าหรือข้อความต่างๆ ที่จะกระทำกับสวิตช์ให้กับเครื่องของผู้ใช้งานเสมือนว่าได้เข้าตั้งค่าต่างๆ กับสวิตช์จริง ในส่วนนี้จะใช้โปรแกรม NetBeans IDE 4.1 ที่อยู่ในโปรแกรมนี้ทำหน้าที่ควบคุมเซิร์ฟเวอร์ที่สร้างไว้ ในส่วนของเครื่องคอมพิวเตอร์ที่เป็นเครื่องของผู้ใช้งานนั้น จะมีโปรแกรม Internet Explorer ติดตั้งอยู่เพื่อทำการติดต่อกับเครื่องที่เป็นเซิร์ฟเวอร์ โดยจะรับ-ส่งการตั้งค่าต่างๆ ที่ผู้ใช้ต้องการผ่านทางโปรแกรมนี้ การสร้างโดยะแกรมการทดลองจะมีความสัมพันธ์กับโปรแกรมทดลองจนคำสั่งการใช้งานต่างๆ การทดลองได้ออกแบบให้มีเนื้อหาเกี่ยวกับการแบ่งพื้นที่การใช้งานในตัวอุปกรณ์สวิตช์เชื่อมต่อเครือข่าย ซึ่งเนื้อหาปริญญาณิพนธ์ในบทนี้จะได้กล่าวถึงการออกแบบของทั้ง 2 ส่วน

#### 3.2 การออกแบบโปรแกรม

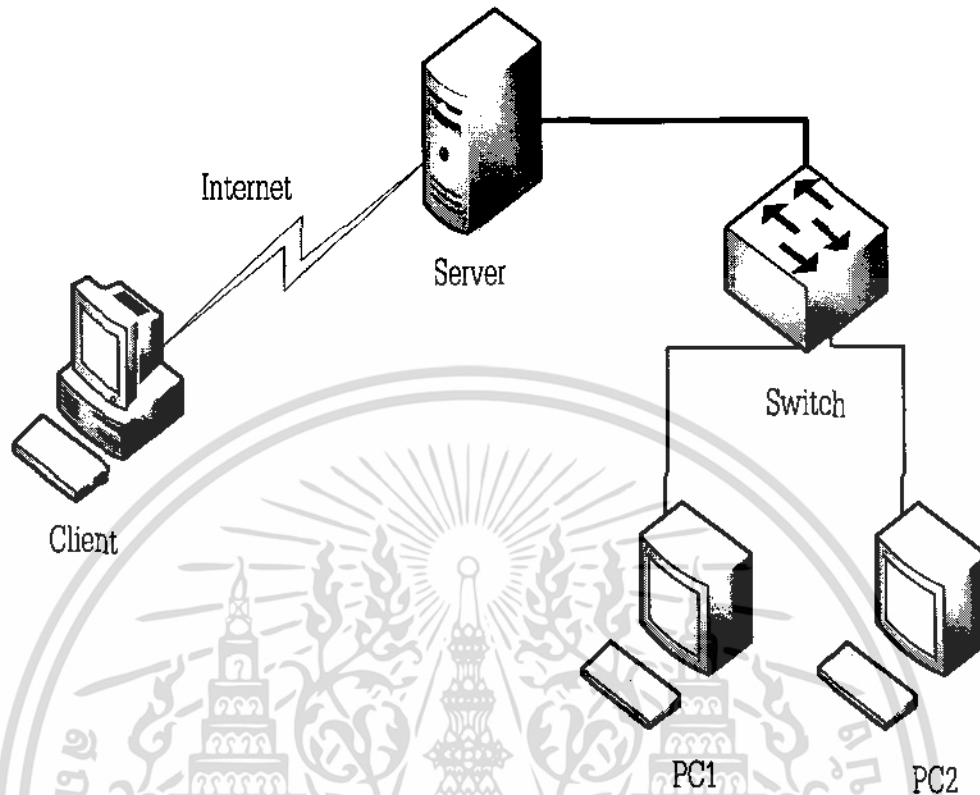
การออกแบบโปรแกรมทดลองเครือข่ายท้องถิ่นแบบเสมือนบนเครือข่ายอินเทอร์เน็ต โปรแกรมนี้ ออกแบบโดยใช้การเขียนโปรแกรมเชิงวัตถุ (Object Oriented Programming) เพื่อให้การเขียนโปรแกรม เป็นไปได้อย่างต่อการเพิ่มและเปลี่ยนแปลงโปรแกรม ดังได้กล่าวแล้วข้างต้นว่าแบ่งออกเป็น 2 ส่วน ดังนั้นจะได้กล่าวถึงการออกแบบทีละส่วน ดังนี้

##### 3.2.1 ส่วนของโปรแกรมที่อยู่ในเครื่องเซิร์ฟเวอร์

ส่วนของโปรแกรมที่อยู่ในเครื่องเซิร์ฟเวอร์นี้ หลักการออกแบบนั้นสามารถอธิบายได้ดังต่อไปนี้

1. โปรแกรมที่อยู่ในเครื่องเซิร์ฟเวอร์นั้นได้อาศัยการเขียนเป็นโปรแกรมภาษาจาวาแอปพลิเคชัน เพราะว่าสามารถที่จะรันได้ทุกแพลตฟอร์ม และยังสามารถแก้ไขเปลี่ยนแปลงได้ง่าย ทั้งยังเป็นภาษาที่นิยมของโปรแกรมเมอร์ในขณะนี้ ยังมีความกระชับมากกว่าเดิมแต่ มีคำสั่งพื้นฐานเหมือนภาษาซีพลัสพลัสที่นิยมในสมัยก่อนด้วย ซึ่งโครงการนี้จะใช้เขียนโปรแกรมภาษาจาวาติดต่อกันระหว่างเครื่องผู้ใช้งาน เครื่องเซิร์ฟเวอร์ และอุปกรณ์สวิตช์ การเชื่อมต่อของเครื่องผู้ใช้งาน เครื่องเซิร์ฟเวอร์ และอุปกรณ์สวิตช์ แสดงได้ดังรูปที่ 3.1

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.1 การเชื่อมต่อระหว่างเครื่องเซิร์ฟเวอร์ เครื่องของผู้ใช้งาน และสวิตช์

ในรูปที่ 3.1 การทดลองได้ออกแบบลักษณะการต่อใช้งานอุปกรณ์สวิตช์เชื่อมต่อเครือข่ายร่วมกับเครื่องลูกข่ายจำนวน 2 เครื่องโดยเครื่องลูกข่ายทั้ง 2 จะกำหนดให้ต่อเชื่อมกับอุปกรณ์สวิตช์แบบถาวรเพื่อให้สะดวกในการทดลองและยังจะช่วยลดปัญหาในการเชื่อมต่อพอร์ตของอุปกรณ์สวิตช์อีกด้วย การออกแบบการทดลองนี้สังเกตได้ว่าจะมีรูปวงจรถอดลงที่เหมือนกับโครงสร้างการเชื่อมต่อของโครงงาน ที่ตั้งรูปการทดลองแบบนี้เพื่อทดลองว่าระบบนี้สามารถตั้งค่าต่างๆ ออกมาได้ใช้งานได้

จากรูปที่ 3.1 สามารถอธิบายถึงอุปกรณ์ต่างๆ ได้ดังนี้

Client เป็นเครื่องของผู้ใช้ที่สามารถใช้งานผ่านระบบอินเทอร์เน็ต

Server เป็นเครื่องบริการความเสถียรต่างๆ ในการใช้งานระบบการทดลอง

Switch เป็นอุปกรณ์หลักในการทดลองโดยจะรับการตั้งค่าต่างๆ จากผู้ใช้งานระบบ

PC1 และ PC2 เป็นเครื่องคอมพิวเตอร์ลูกข่ายที่ใช้ในการทดลองโดยกำหนดให้มีการตั้งค่า IP

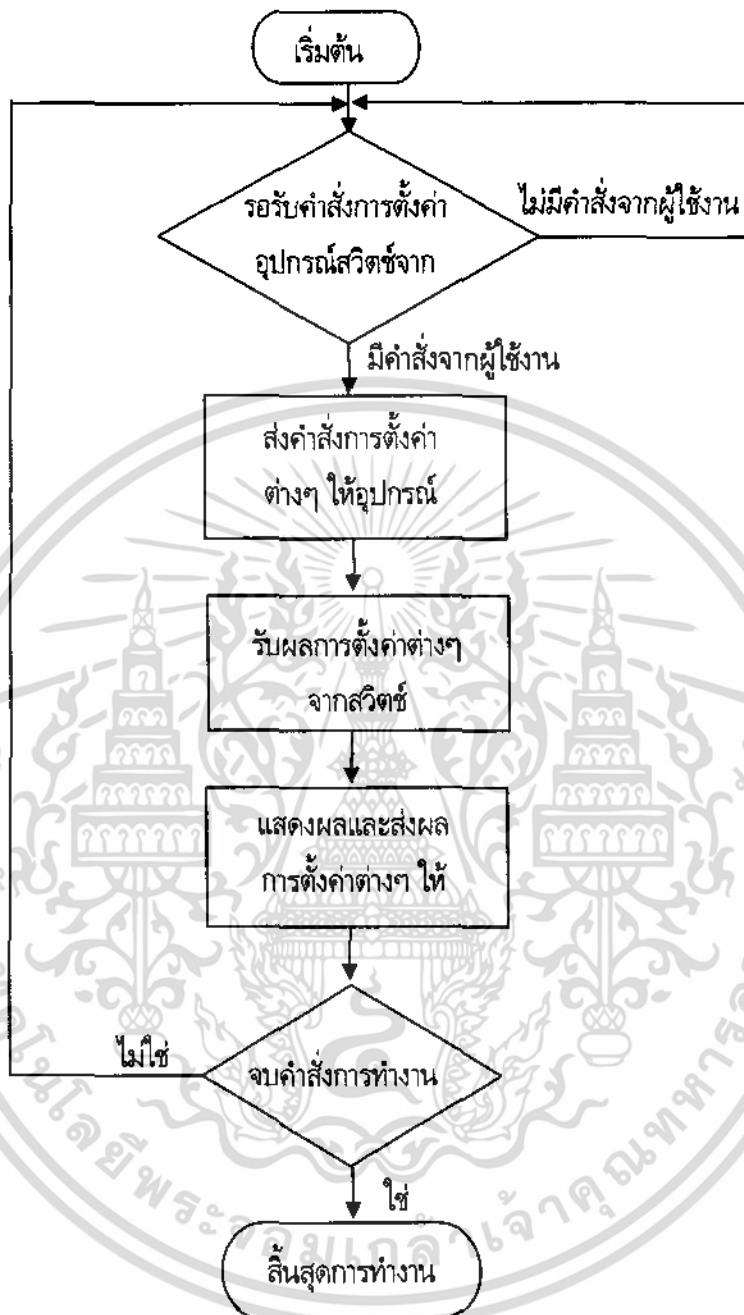
Address ไว้คงที่และต่อกับพอร์ตของอุปกรณ์สวิตช์ไว้คงที่เช่นกัน

การออกแบบในส่วนของเซิร์ฟเวอร์นั้นสามารถอธิบายหลักการออกแบบได้จากผังงานการทำงานใน

ส่วนของเซิร์ฟเวอร์ได้ดังรูปที่ 3.2

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



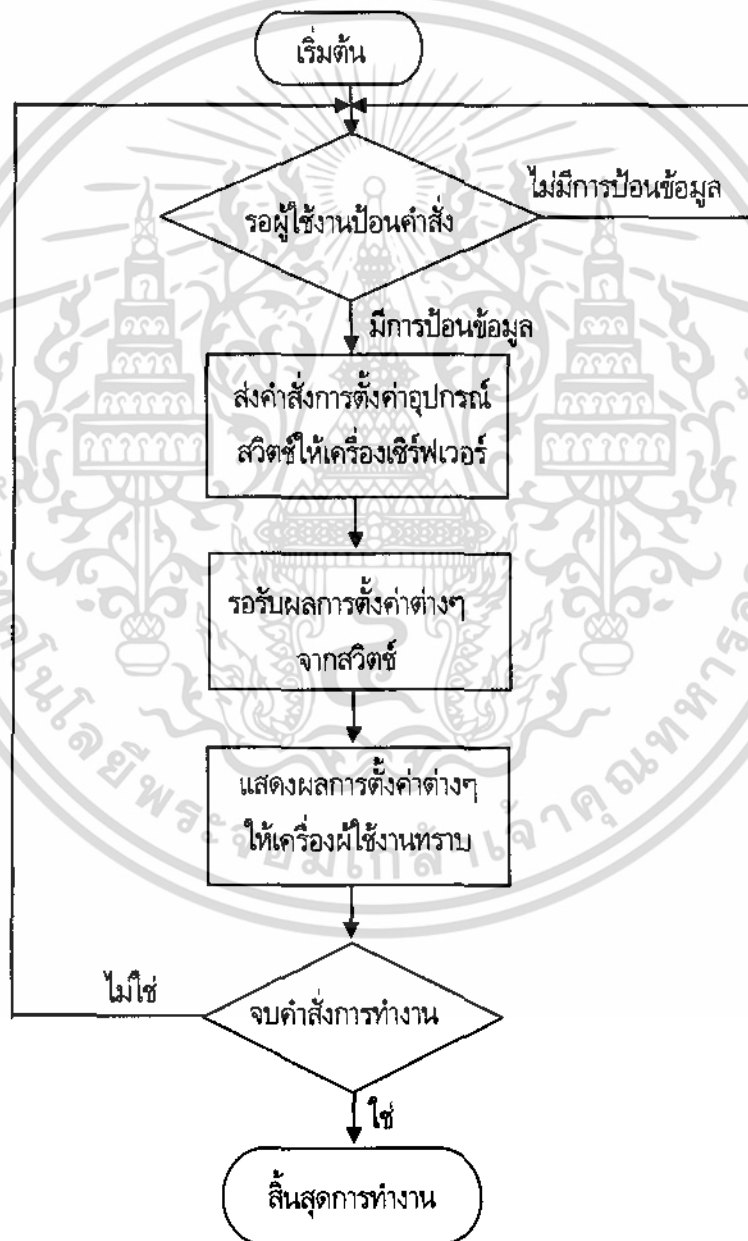
รูปที่ 3.2 ผังงานการทำงานของเครื่องเซิร์ฟเวอร์

### 3.2.2 ส่วนของเครื่องผู้ใช้งาน

ส่วนของเครื่องผู้ใช้งานนั้นหลักการออกแบบ จากขอบเขตของโครงการที่กล่าวมาแล้วว่าจะต้องทำการทดลองบนเครือข่ายอินเทอร์เน็ต ฉะนั้นบนเครื่องของผู้ใช้งานนั้นจะต้องทำการติดตั้งโปรแกรม Internet Explorer ซึ่งส่วนมากจะมีอยู่ทุกเครื่องที่ทำการลง Windows XP อยู่แล้ว เพื่อที่ผู้ใช้งานจะได้ทำการทดลองเอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

แบบเสมือนจริงว่าได้ตั้งค่าต่างๆ บนอุปกรณ์สวิตช์จริงๆ และสามารถที่จะทำการทดลองจากที่ไหนก็ได้ที่เครื่องนั้นจะต้องต่ออินเทอร์เน็ตอยู่ที่เท่านั้นเอง ซึ่งก็จะต้องรู้ URL ที่เข้าไปใช้งานโปรแกรมนี้ ซึ่งจะทำให้ผู้ใช้งานได้เรียนรู้ศึกษาเพิ่มเติมจากที่ทำการทดลองในห้องหรือเมื่อมีอุปกรณ์สวิตช์ไม่เพียงพอกับจำนวนนักศึกษาที่ทำการทดลอง

หลักการออกแบบของเครื่องผู้ใช้งานนั้นสามารถจะอธิบายได้จากผังงานการทำงานของเครื่องผู้ใช้งานได้ดังรูปที่ 3.3



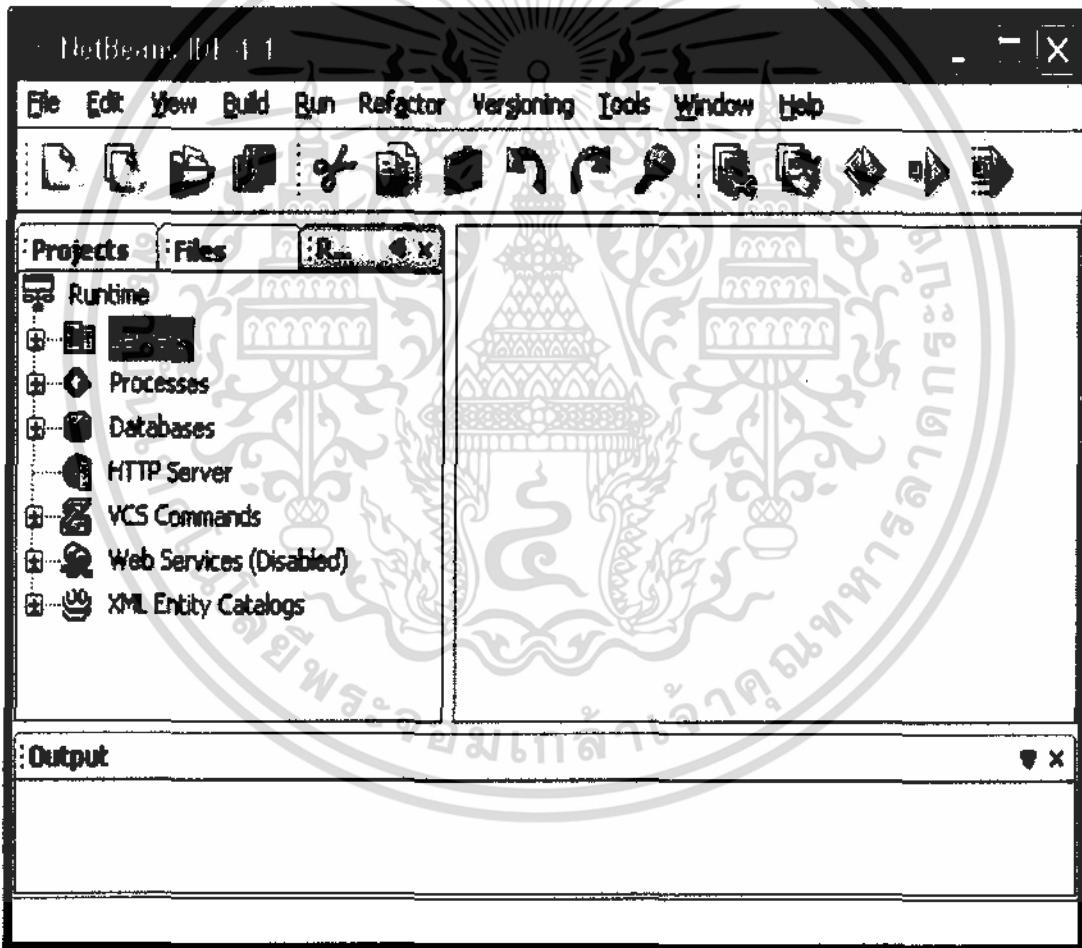
**รูปที่ 3.3** ผังงานการทำงานของเครื่องผู้ใช้งาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น เมื่ออนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 3.3 วิธีการสร้างโปรแกรม

ในการสร้างโปรแกรมที่อยู่ในเครื่องเซิร์ฟเวอร์ใช้โปรแกรม NetBeans IDE 4.1 เป็นเครื่องมือในการช่วยเขียนโปรแกรม ซึ่งเราจะเขียนด้วยใช้ภาษาจาวาแอปพลิเคชัน ใช้เป็นเว็บเซิร์ฟเวอร์ ส่วนโปรแกรมที่ใช้เขียนจะเป็น Java เพราะว่าเขียนใช้วิธีแบบออบเจกต์โอเรียนเตดโปรแกรมมิง (Object Oriented Programming) เพื่อให้การเขียนโปรแกรมเป็นไปได้อย่างต่อการเพิ่มและเปลี่ยนแปลงโปรแกรม การสร้างเว็บเซิร์ฟเวอร์โดยใช้ NetBeans IDE 4.1 สามารถสร้างขึ้นได้ดังนี้

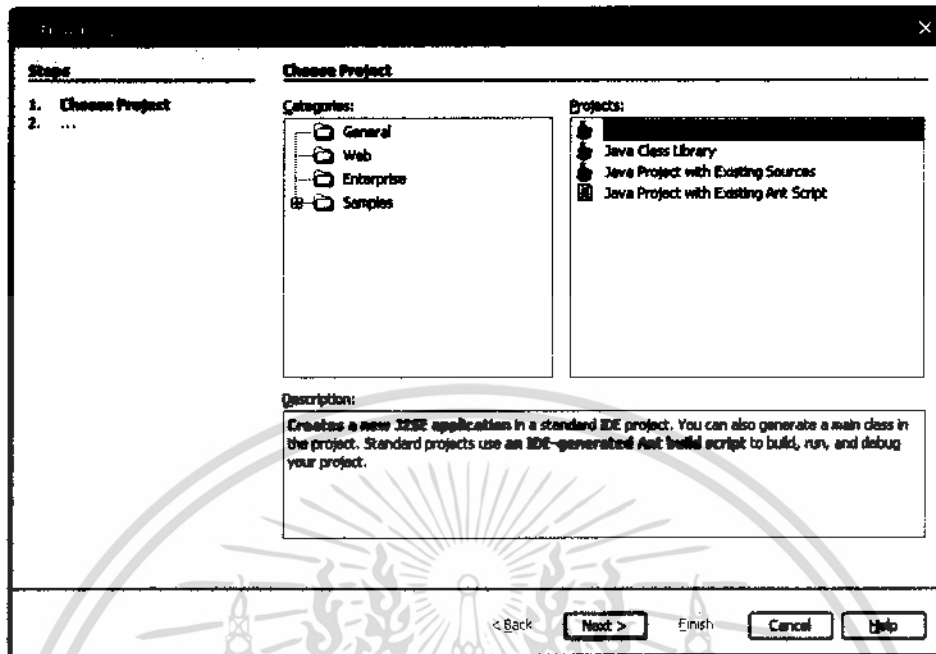
1. เปิดโปรแกรม NetBeans IDE 4.1 ที่ได้ติดตั้งไว้ อาจจะเปิดได้จากหน้าจอคอมพิวเตอร์หรือจากไฟล์ที่เราติดตั้งและเก็บไว้ผลจะปรากฏดังรูปที่ 3.4



รูปที่ 3.4 โปรแกรม NetBeans IDE 4.1

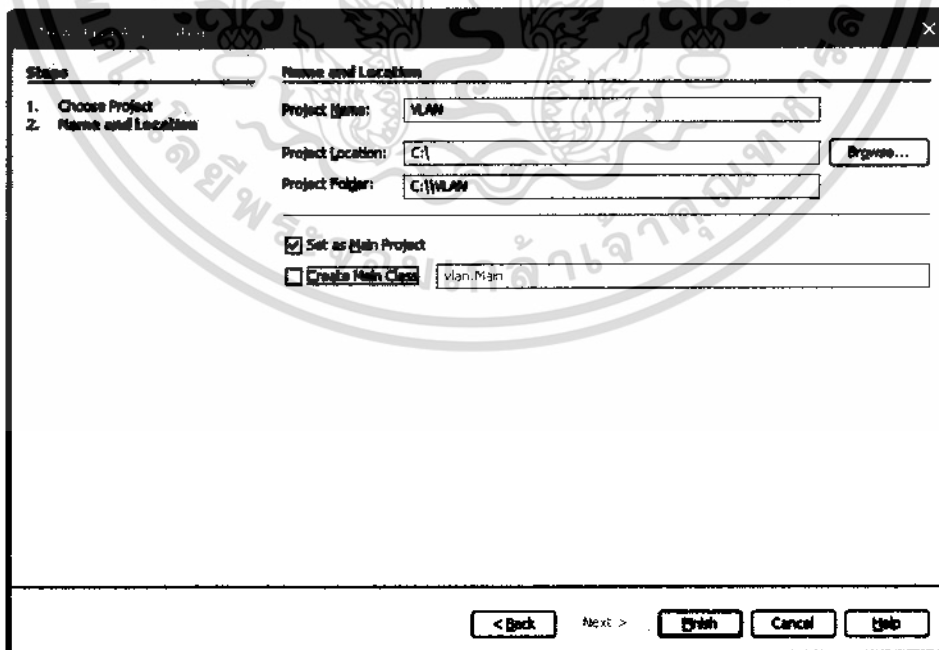
2. เมื่อจะเขียนโปรแกรมให้เลือกไปที่ File -> New Project แล้วเลือก General ที่อยู่ท้ายมือ และเลือก Java Application อยู่ขวามือ แล้วกดปุ่ม Next จะแสดงผลดังรูปที่ 3.5

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.5 ขั้นตอนการเขียนโปรแกรม ขั้นที่ 1

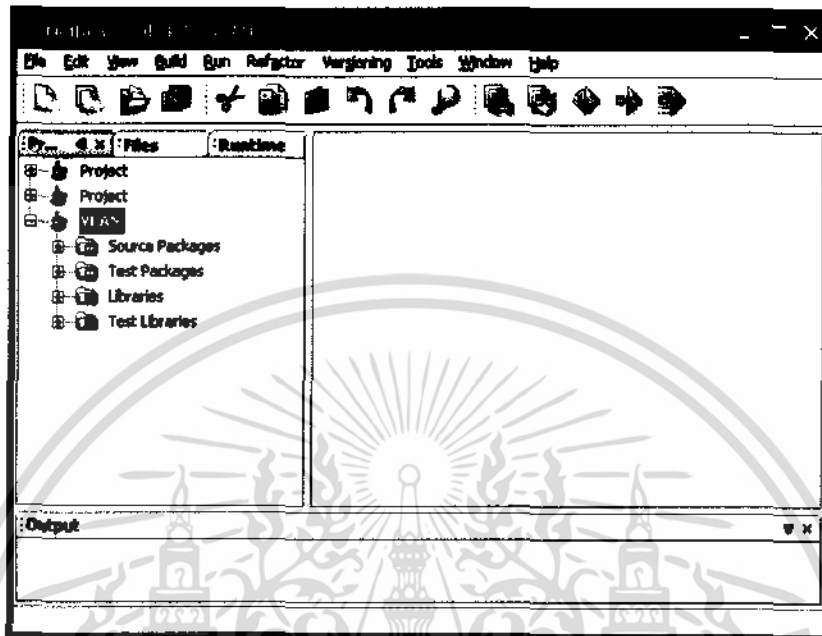
3. ตั้งชื่อที่ชื่อ Project Name ว่า VLAN แล้วก็เลือกที่อยู่ให้อยู่ที่ไดร์ฟ c:\ สุดท้ายก็กดที่ปุ่ม Finish ผลจะปรากฏดังรูปที่ 3.6



รูปที่ 3.6 ขั้นตอนการเขียนโปรแกรม ขั้นที่ 2

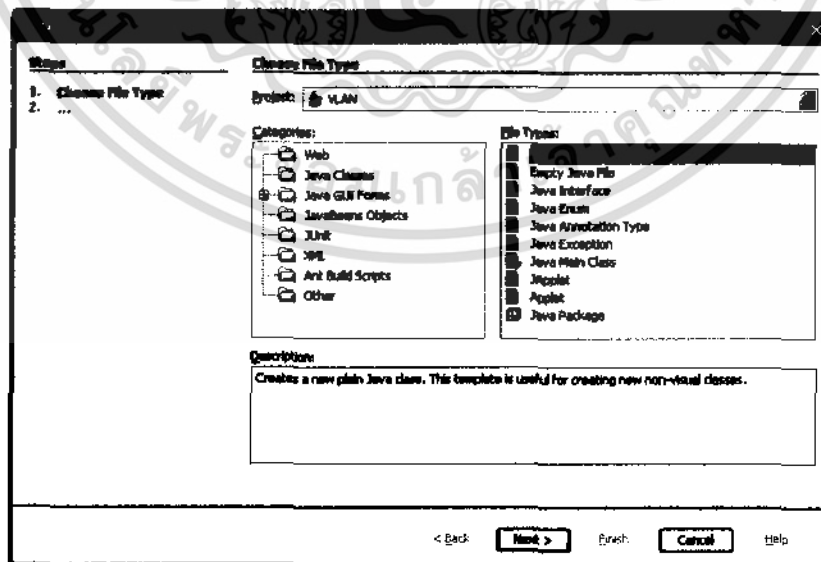
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4. เมื่อเลือกดั่งที่ได้กล่าวไปแล้วในข้อ 3 ปรากฏผลดังรูปที่ 3.7 ซึ่งเป็นการสร้าง Project ใหม่



รูปที่ 3.7 ขั้นตอนการเขียนโปรแกรม ขั้นที่ 3

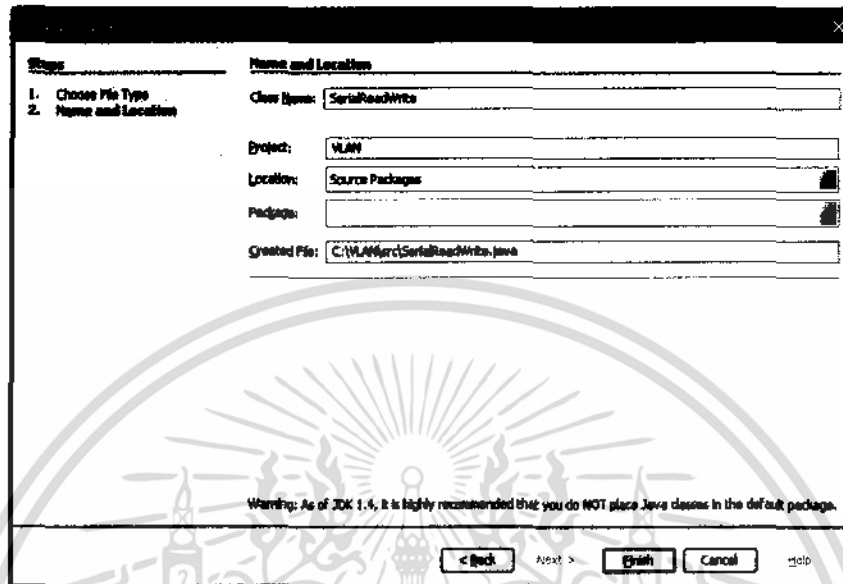
5. และเมื่อต้องการเขียนโปรแกรมก็ไปที่ File -> New File จะปรากฏผลดังรูปที่ 3.8 ให้เลือกเลือก Java Classes ที่อยู่ทางด้านซ้ายมือ และเลือก Java Class ที่อยู่ขวามือ กด Next



รูปที่ 3.8 ขั้นตอนการเขียนโปรแกรม ขั้นที่ 4

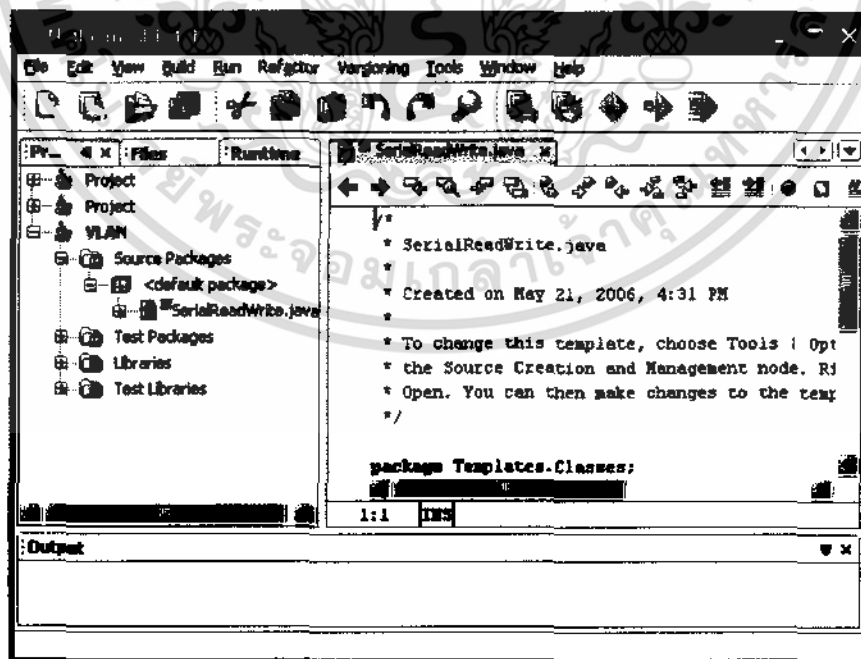
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

6. เมื่อปรากฏผลดังรูปที่ 3.9 ก็ให้ใส่ชื่อของคลาสว่า SerialReadWrite แล้วกด Finish



รูปที่ 3.9 ขั้นตอนการเขียนโปรแกรม ขั้นที่ 5

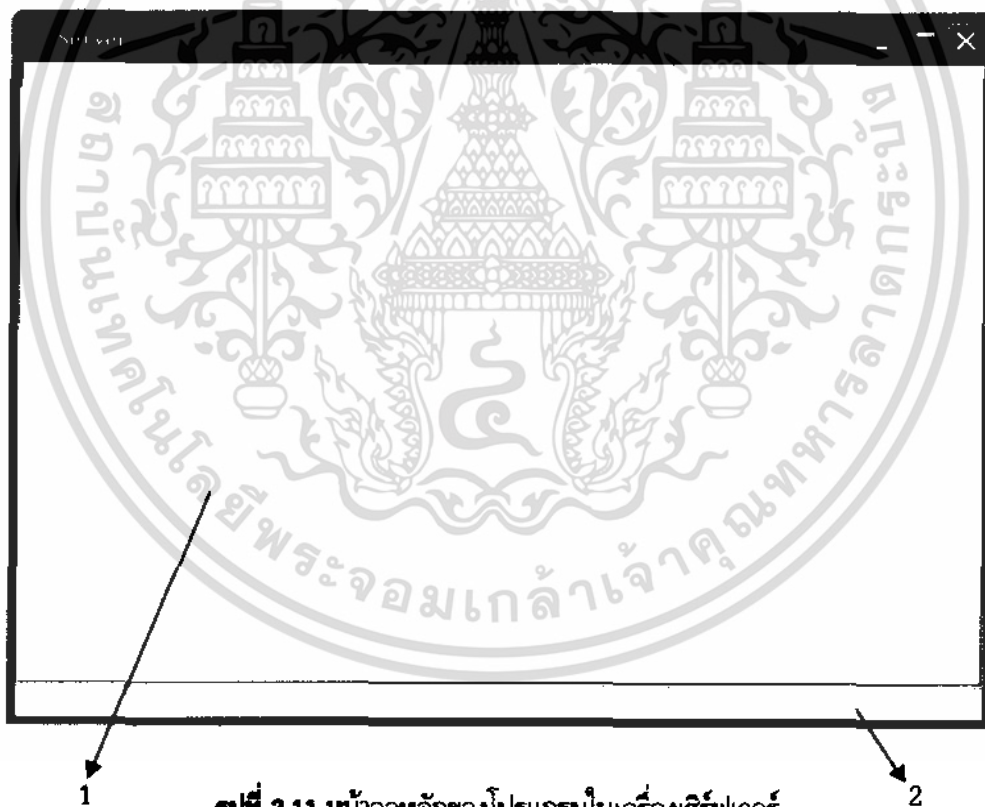
7. เมื่อเลือกตามข้อ 6 ก็จะปรากฏชื่อคลาสที่ตั้งไว้แล้วก็เขียนโปรแกรมเพื่อติดต่อกับสวิทช์



รูปที่ 3.10 ขั้นตอนการเขียนโปรแกรม ขั้นที่ 6

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

8. โปรแกรม SerialReadWrite.java นี้จะเป็นการเขียนเพื่อที่ให้เครื่องเซิร์ฟเวอร์ได้ติดต่อกับอุปกรณ์สวิตช์เพื่อให้เซิร์ฟเวอร์ส่งการตั้งค่าต่างๆ ที่ผู้ใช้งานส่งมาให้กับอุปกรณ์สวิตช์และเครื่องเซิร์ฟเวอร์จะคอยรับการแสดงผลจากอุปกรณ์สวิตช์และส่งไปยังเครื่องของผู้ใช้งาน
9. เมื่อเขียนโปรแกรม SerialReadWrite.java เสร็จแล้วก็สร้างโปรแกรม Server.java เพื่อจะให้เซิร์ฟเวอร์ทำหน้าที่เป็นตัวกลางในการเชื่อมต่อระหว่างเครื่องผู้ใช้งานกับอุปกรณ์สวิตช์ให้ทำงานได้อย่างสมบูรณ์ ในการเขียนจะใช้คลาส Serversocket เพื่อเขียนติดต่อกับเครื่องของผู้ใช้งาน โดยจะใช้หมายเลขพอร์ตในการเชื่อมต่อระหว่างเครื่องเซิร์ฟเวอร์และเครื่องผู้ใช้งานที่ตรงกัน ในโครงงานนี้ใช้ หมายเลขพอร์ต 5001 ซึ่งทางเครื่องผู้ใช้งานก็ต้องเป็นใช้หมายเลขพอร์ต 5001 ส่วนการเชื่อมต่อกับอุปกรณ์สวิตช์ก็เชื่อมการติดต่อกันโดยทาง Serial Port วิธีการเขียนโปรแกรมก็ทำเหมือนกับข้างต้นที่ได้กล่าวไปแล้ว ซึ่งหน้าจอหลักของโปรแกรมที่อยู่ในเครื่องเซิร์ฟเวอร์นั้นแสดงได้ดังรูปที่ 3.8

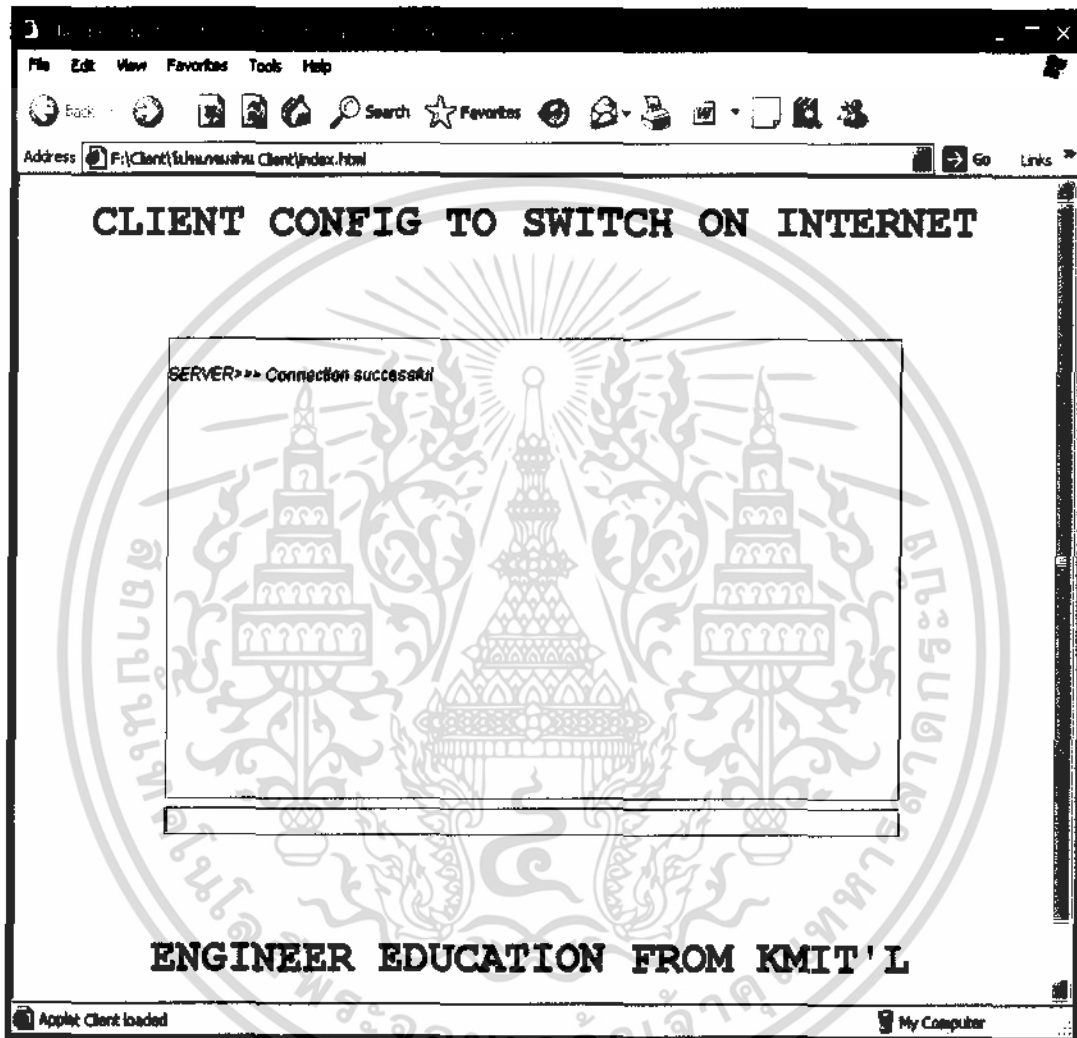


รูปที่ 3.11 หน้าจอหลักของโปรแกรมในเครื่องเซิร์ฟเวอร์

จากรูปหน้าจอหลักของโปรแกรมที่อยู่ในเครื่องเซิร์ฟเวอร์นั้น หมายเลข 1 คือ เป็นหน้าจอแสดงผลที่จะรับ-ส่งการตั้งค่าต่างๆ ระหว่างเครื่องผู้ใช้งานกับอุปกรณ์สวิตช์ หมายเลข 2 คือ เป็นส่วนที่เก็บไว้เพื่อทดสอบการติดต่อกับอุปกรณ์สวิตช์ และเครื่องผู้ใช้งานทดสอบดูว่าสามารถเชื่อมต่อกันได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

10. โปรแกรมสุดท้ายคือ โปรแกรม *Client.java* ที่เป็นตัวเชื่อมต่อกับเครื่องเซิร์ฟเวอร์ให้ส่งการแสดงผลการตั้งค่าต่างๆ ให้อุปกรณ์เสิร์ช ซึ่งตัวเซิร์ฟเวอร์จะเป็นผู้ส่งไป แล้วจึงตั้งค่าต่างๆ ผ่านหน้าจอนี้ และหน้าจอหลักของโปรแกรมปรากฏดังรูปที่ 3.12



รูปที่ 3.12 หน้าจอหลักของโปรแกรมของเครื่องผู้ใช้งาน

จากโปรแกรมที่เขียนให้เครื่องของผู้ใช้งานสามารถตั้งค่าต่างๆ จะใช้วิธีการเขียนโปรแกรมโดยใช้คลาส *Socket* (`InetAddress.getByName ("161.246.14.142", 5001)`; ซึ่ง 161.246.14.142 คือหมายเลขไอพีของเซิร์ฟเวอร์ และ 5001 หมายถึงพอร์ตที่เอาไว้เชื่อมการติดต่อกับเครื่องเซิร์ฟเวอร์ โดยจะติดต่อกันผ่านทางพอร์ตนี้ ด้วยการเขียนเป็น Java Applet และใช้ตัวภาษา HTML ในการดึงไฟล์ที่เป็น Java Applet มาแสดงโดยจะทำการติดตั้งตัวไฟล์ HTML ไว้ที่เครื่องของผู้ใช้งาน เพื่อที่จะทำการตั้งค่าต่างๆ ได้โดยสามารถเชื่อมต่อกับ Internet Explorer ได้ ซึ่งภาษา HTML เขียนได้ดังนี้ ผลลัพธ์แสดงได้จากรูปที่ 3.12

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 3.4 การทำงานของส่วนต่างๆ

การทำงานของโครงการทดลองเครือข่ายท้องถิ่นแบบเสมือนบนเครือข่ายอินเทอร์เน็ตดังได้กล่าวแล้วข้างต้นว่าแบ่งออกเป็น 2 ส่วน คือ ส่วนของโปรแกรมที่อยู่ในเครื่องเซิร์ฟเวอร์ และส่วนของเครื่องผู้ใช้งาน ดังนั้นจะได้กล่าวถึงการทำงานที่ละส่วนได้ดังนี้

#### 3.4.1 ส่วนของโปรแกรมที่อยู่ในเครื่องเซิร์ฟเวอร์

เป็นโปรแกรมติดตั้งไว้ในเครื่องเซิร์ฟเวอร์เพื่อคอยบริการให้กับ การติดต่อของผู้ใช้งาน คอยจัดการ และตรวจสอบคำสั่งการใช้งานจากผู้ใช้และส่งข้อมูลนั้นไปให้กับอุปกรณ์สวิตช์เชื่อมต่อเครือข่ายรวมถึงรับค่าต่างๆ เมื่อเกิดการเปลี่ยนแปลงที่อุปกรณ์สวิตช์เชื่อมต่อเครือข่ายเพื่อจะได้นำส่งกลับไปแสดงผลที่ส่วนของโปรแกรมที่ติดต่อกับผู้ใช้ การติดต่อกันระหว่างเครื่องเซิร์ฟเวอร์กับเครื่องผู้ใช้งานจะใช้พอร์ต 5001 เป็นตัวคอยเชื่อมการติดต่อ ส่วนเครื่องเซิร์ฟเวอร์กับอุปกรณ์สวิตช์ใช้การติดต่อผ่าน Serial Port

#### 3.4.2 ส่วนของเครื่องผู้ใช้งาน

ส่วนของเครื่องผู้ใช้งานจะทำหน้าที่รับ-ส่งคำสั่งระหว่างผู้ใช้กับเครื่องเซิร์ฟเวอร์ โดยติดต่อกันผ่านทางโปรแกรม Internet Explorer ที่ทำการดึงไฟล์ข้อมูลที่ถูกฝังตัวอยู่ที่เครื่องเซิร์ฟเวอร์ออกมาใช้ ให้เครื่องเซิร์ฟเวอร์ส่งการตั้งค่าต่างๆ ให้กับอุปกรณ์สวิตช์แล้วเครื่องเซิร์ฟเวอร์ก็ส่งค่ามาแสดงผลที่ผู้ใช้งานได้ถูกต้อง โดยเชื่อมการสื่อสารกันด้วยพอร์ต 5001 ซึ่งพอร์ตนี้จะเหมือนกับเครื่องเซิร์ฟเวอร์ เครื่องเซิร์ฟเวอร์ใช้พอร์ตไหนเครื่องผู้ใช้งานก็ใช้พอร์ตตัวเดียวกัน

## บทที่ 4

### การทดลองและผลการทดลอง

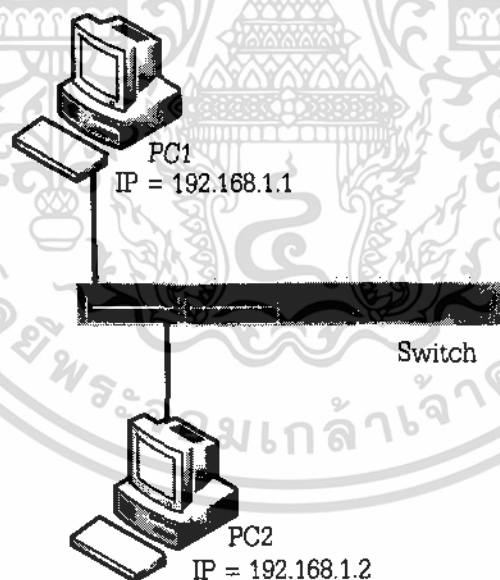
#### 4.1 กล่าวนำ

ในส่วนของบทนี้จะกล่าวถึงการทดลองและผลการทดลองของโปรแกรมทดลองเครือข่ายท้องถิ่นแบบเสมือนบนเครือข่ายอินเทอร์เน็ต ในส่วนของโปรแกรมการทดลองเมื่อผนวกกับการใช้งานร่วมกับวงจรการทดลองและเมื่อใช้งานผ่านระบบอินเทอร์เน็ต

#### 4.2 การทดลองเครือข่ายท้องถิ่นแบบเสมือน

##### 4.2.1 การต่อวงจรการทดลองเครือข่ายท้องถิ่นแบบเสมือน

เป็นรูปแบบการเชื่อมต่อระหว่างอุปกรณ์สวิตช์เชื่อมต่อเครือข่ายกับเครื่องคอมพิวเตอร์ลูกข่าย โดยระบุค่าที่ใช้ตามการออกแบบดังนี้



**รูปที่ 4.1** การเชื่อมต่อระหว่างอุปกรณ์สวิตช์เชื่อมต่อเครือข่ายกับเครื่องคอมพิวเตอร์ลูกข่าย

การเชื่อมต่อระหว่างเครื่องคอมพิวเตอร์กับอุปกรณ์สวิตช์จะใช้สาย UPT category 5 แบบตรง และกำหนดหมายเลข IP Address ไว้คงที่ คือ PC1 = 192.168.1.1 ต่อไว้ที่พอร์ต 1 ของอุปกรณ์สวิตช์และ PC2 = 192.168.1.2 ต่อไว้ที่พอร์ต 18 ของอุปกรณ์สวิตช์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

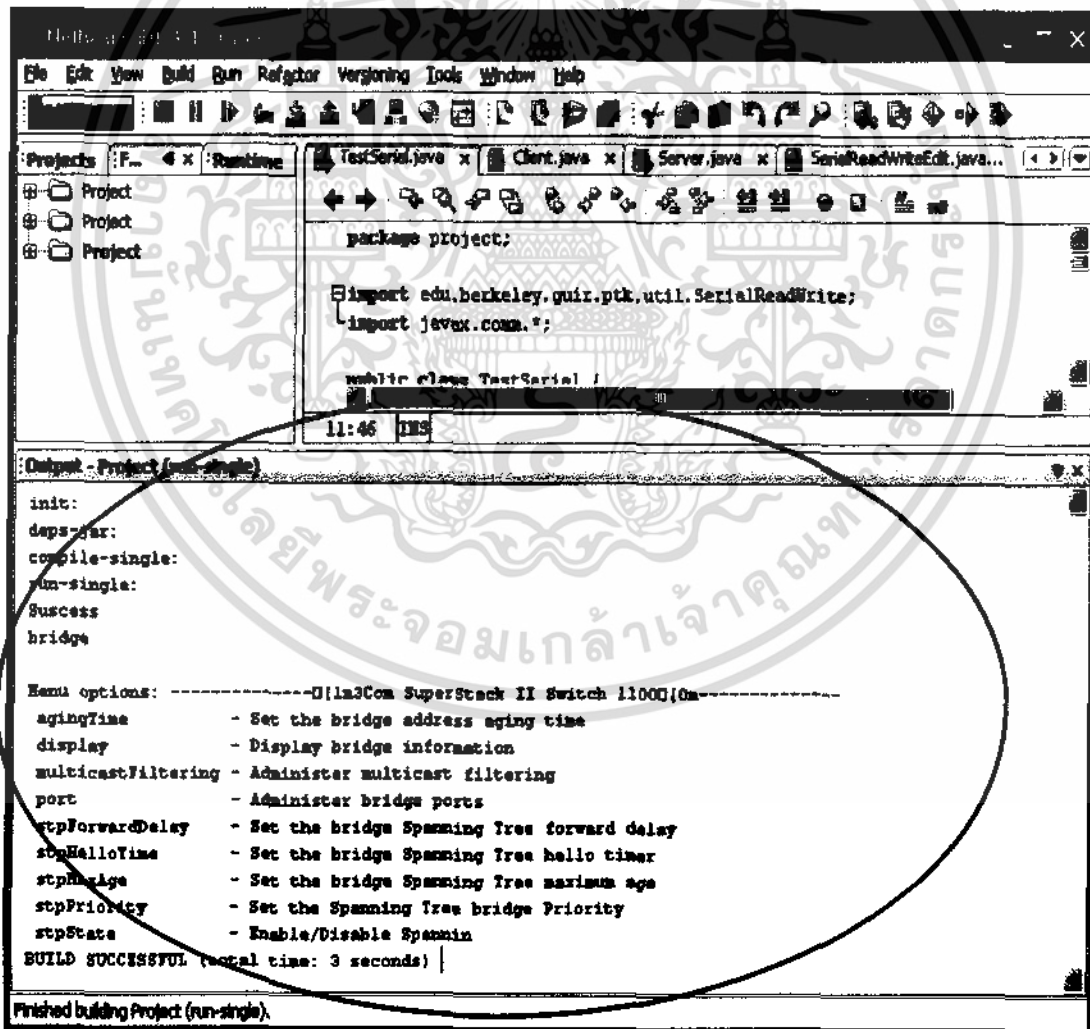
## 4.2.2 การทดลองโปรแกรมระหว่างเครื่องเซิร์ฟเวอร์กับอุปกรณ์สวีตช์

### 4.2.2.1 ขั้นตอนการทดลอง

1. เขียนโปรแกรม TestSerial.java เพื่อทดสอบการติดต่อระหว่างเครื่องเซิร์ฟเวอร์กับอุปกรณ์สวีตช์ว่าสามารถเชื่อมการติดต่อถึงกันได้ โดยการทดสอบการพิมพ์คำสั่ง Bridge เข้าไปในอุปกรณ์สวีตช์
2. ทำการคอมไพล์ และรันไฟล์โปรแกรม TestSerial.java สังเกตผลการทดลอง

### 4.2.2.2 ผลการทดลอง

1. ถ้าหากว่าโปรแกรมที่เขียนทดสอบนั้น ซึ่งสามารถที่จะส่งข้อมูลติดต่อกับอุปกรณ์สวีตช์ได้เป็นผลสำเร็จ โดยการแสดงผลของเมนูจะเข้าสู่คำสั่งเมนู Bridge และส่งการแสดงผลจากอุปกรณ์สวีตช์มาปรากฏที่เอาต์พุตของโปรแกรม TestSerial.java ซึ่งปรากฏได้ดัง รูปที่ 4.5



```

package project;

import edu.berkeley.gui.ptk.util.SerialReadWrite;
import javax.comm.*;

public class TestSerial {
    ...
}
11:46 AM

:Output - Project (run-single)
init:
deps-jar:
compile-single:
run-single:
Success
bridge

Menu options: -----D:\lab3Com SuperStack II Switch 1100010a-----
agingTime      - Set the bridge address aging time
display         - Display bridge information
multicastFiltering - Administer multicast filtering
port           - Administer bridge ports
stpForwardDelay - Set the bridge Spanning Tree forward delay
stpHelloTime   - Set the bridge Spanning Tree hello timer
stpMaxAge      - Set the bridge Spanning Tree maximum age
stpPriority     - Set the Spanning Tree bridge Priority
stpState       - Enable/Disable Spanning
BUILD SUCCESSFUL (total time: 3 seconds)

Finished building Project (run-single).
  
```

รูปที่ 4.2 ผลลัพธ์การรันโปรแกรม TestSerial.java

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

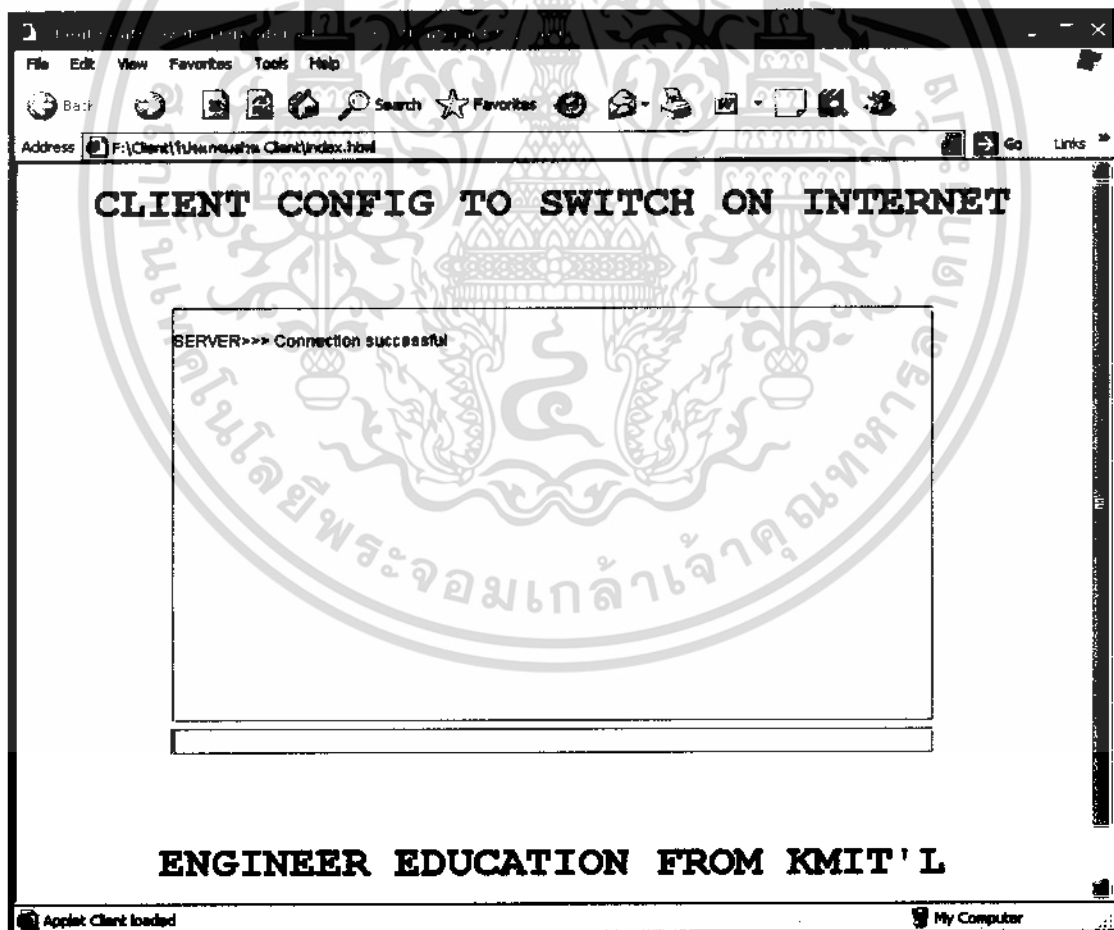
## 4.2.3 การทดลองโปรแกรมระหว่างเครื่องผู้ใช้งานกับเครื่องเซิร์ฟเวอร์

### 4.2.3.1 ขั้นตอนการทดลอง

1. เครื่องของผู้ใช้งานให้เปิดโปรแกรม Internet Explorer ซึ่งได้ทำการติดตั้งไว้ที่หน้าจอของคอมพิวเตอร์ชื่อ Index.html
2. เมื่อเข้าสู่หน้าต่างการใช้งานก็สามารถทดลองการตั้งค่าต่างๆ ได้ โดยเริ่มจากใช้คำสั่ง Bridge แล้วกด Enter เพื่อส่งไปติดต่อกับเครื่องเซิร์ฟเวอร์และรอการแสดงผลตอบกลับจากเครื่องเซิร์ฟเวอร์ เมื่อเครื่องเซิร์ฟเวอร์ได้รับการตอบกลับจากอุปกรณ์สวิตช์แล้ว

### 4.2.3.2 ผลการทดลอง

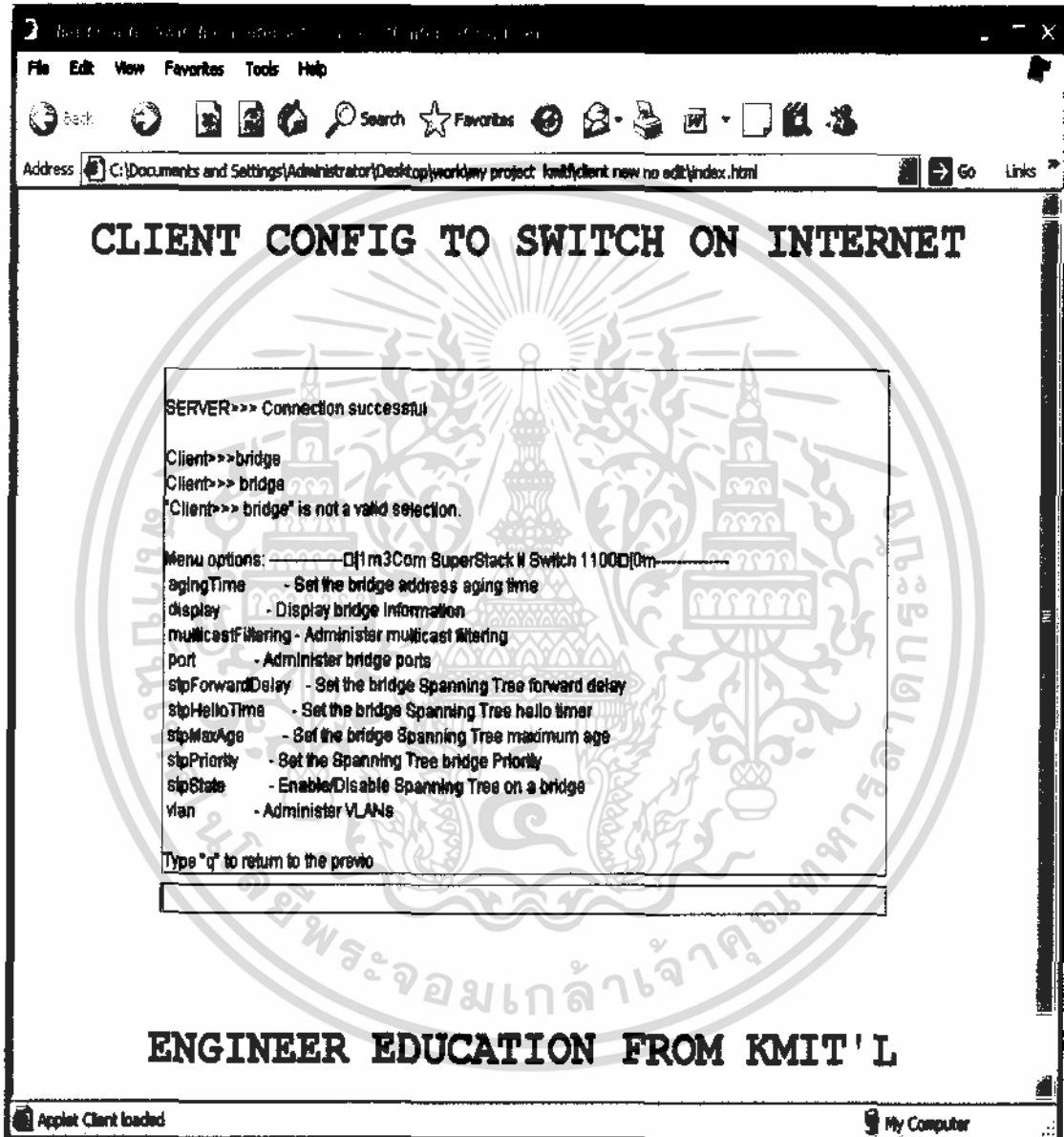
1. ถ้าหากโปรแกรมของเครื่องผู้ใช้งานสามารถที่จะติดต่อกับเครื่องเซิร์ฟเวอร์ได้ ก็จะสามารถทำการเชื่อมต่อกับอุปกรณ์สวิตช์ได้ โดยจะมีข้อความแสดงการเชื่อมการติดต่อได้ว่า Server>>>Connection successful แสดงดังรูปที่ 4.2



รูปที่ 4.3 การติดต่อกับเครื่องเซิร์ฟเวอร์เป็นผลสำเร็จ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

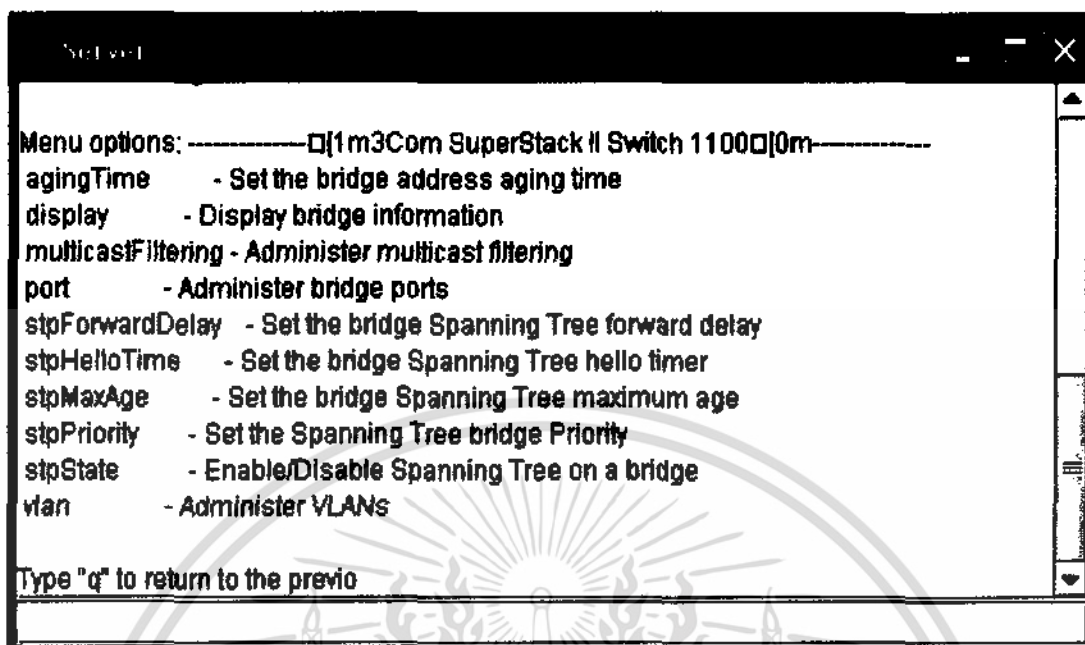
- เริ่มการทดลองด้วยการสร้าง VLAN ใหม่ด้วยการเริ่มพิมพ์คำสั่งเข้าเมนู Bridge เมื่อเข้าสู่คำสั่งเมนูนี้แล้ว จะปรากฏผลดังรูปที่ 4.3 ซึ่งหมายถึงผู้ใช้งานนั้นสามารถที่จะตั้งค่าต่างๆ ให้แก่อุปกรณ์สวิทช์ได้แล้ว



รูปที่ 4.4 การแสดงผลตอบกลับจากเครื่องเซิร์ฟเวอร์

- เมื่อเครื่องเซิร์ฟเวอร์เมื่อได้รับคำสั่งจากเครื่องผู้ใช้งานก็จะส่งไปให้อุปกรณ์สวิทช์และอุปกรณ์สวิทช์ก็ส่งการแสดงผลมายังเครื่องเซิร์ฟเวอร์ และเครื่องเซิร์ฟเวอร์จะส่งมายังเครื่องผู้ใช้งานต่อไป การแสดงผลจากอุปกรณ์สวิทช์ให้เครื่องเซิร์ฟเวอร์ปรากฏผลดังรูปที่ 4.4

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.5 การแสดงผลตอบกลับจากอุปกรณ์สวิตช์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## บทที่ 5

### บทสรุป

#### 5.1 สรุป

การทดลองเครือข่ายท้องถิ่นแบบเสมือนบนเครือข่ายอินเทอร์เน็ต เป็นโปรแกรมทดลองการตั้งค่าต่าง ๆ ให้กับอุปกรณ์สวิตช์เชื่อมต่อกับเครือข่ายซึ่งประกอบด้วย เครื่องเซิร์ฟเวอร์ เครื่องผู้ใช้งาน สวิตช์ โปรแกรมสำหรับเครื่องเซิร์ฟเวอร์ และโปรแกรมการทดลอง

โปรแกรมทดลอง จะแบ่งได้เป็น 2 ส่วน คือ โปรแกรมสำหรับเครื่องเซิร์ฟเวอร์ จะทำการรับส่งค่าต่างๆ ระหว่างเครื่องไคลเอ็นต์กับเครื่องเซิร์ฟเวอร์ และเครื่องเซิร์ฟเวอร์ไปยังตัวอุปกรณ์สวิตช์ เมื่อมีการตั้งค่าจากเครื่องผู้ใช้งานมายังตัวอุปกรณ์สวิตช์จะส่งผลการตั้งค่าต่างๆ มาที่เครื่องเซิร์ฟเวอร์ และเครื่องเซิร์ฟเวอร์จะส่งการตั้งค่าต่างๆ ไปให้กับอุปกรณ์สวิตช์ และจะส่งการแสดงผลมายังผู้ใช้งานให้ผู้ใช้ทราบ ส่วนที่ 2 คือ โปรแกรมผู้ใช้งานคือโปรแกรมที่เขียนขึ้นมาโดย ภาษา HTML เป็นตัวรันไฟล์ Java Applet ที่เครื่องผู้ใช้งานจะทำการตั้งไฟล์ที่เป็นหน้าตาของการติดต่อสามารถตั้งค่าต่างๆ ผ่านโปรแกรมนี้ ไปให้เครื่องเซิร์ฟเวอร์จัดการ จากนั้นเครื่องเซิร์ฟเวอร์ก็จะส่งข้อมูลการตั้งค่าต่างๆ ไปที่ตัวอุปกรณ์สวิตช์เพื่อที่จะตั้งค่าตามที่ผู้ใช้งานต้องการ

ผลการทดลองนั้นผู้ใช้งานสามารถที่จะเข้าโปรแกรมการตั้งค่าต่างๆ ให้กับอุปกรณ์สวิตช์ได้ และมีการตอบกลับด้วยการแสดงผลจากตัวอุปกรณ์สวิตช์มายังเครื่องผู้ใช้งานได้อย่างสมบูรณ์

#### 5.2 ปัญหาและวิธีการแก้ไข

จากการดำเนินการสร้างและทดสอบโครงงานปรากฏว่ามีปัญหาที่เกิดขึ้นหลายประการ ซึ่งสามารถสรุปได้ดังนี้

1. ความล่าช้าในการเขียนโปรแกรม เนื่องจากใช้โปรแกรมภาษาจาวา โดยต้องใช้ฟังก์ชันที่เกี่ยวข้องกับการทำงานผ่านระบบเครือข่าย และหนังสือแต่ละเล่มนั้นให้ข้อมูลที่แตกต่างกัน โดยหนังสือภาษาไทยยังมีจำนวนน้อย รวมทั้งการทดสอบจะต้องกระทำผ่านระบบเครือข่าย
2. การเขียนโปรแกรมในส่วนที่ติดต่อกับ Serial Port ใช้โปรแกรม NetBeans IDE 4.1 นั้นทำงานไม่สมบูรณ์ เพราะไม่รองรับกับ Java เวอร์ชัน 1.5 ซึ่งเป็นเวอร์ชันที่ใหม่

**วิธีการแก้ไข** จะต้องลงโปรแกรม Java เวอร์ชัน 1.4 ที่โปรแกรมสามารถรองรับการทำงานได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปเผยแพร่โดยไม่ได้รับอนุญาต

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3. การติดต่อระหว่างเครื่องผู้ใช้งานกับอุปกรณ์สวิตช์นั้นไม่สามารถเขียนติดต่อในส่วนแรกที่มีการล็อกอินและใส่พาสเวิร์ดได้

**วิธีการแก้ไข** ที่เครื่องเซิร์ฟเวอร์จะต้องเปิดโปรแกรม Hyper Terminal แล้วทำการล็อกอินและพาสเวิร์ดให้เรียบร้อยแล้วทำการปิดโปรแกรมหรือเลิกการเชื่อมต่อ

### 5.3 แนวทางการพัฒนา

1. โปรแกรมชุดนี้สามารถให้บริการของเครื่องของผู้ใช้งานได้เพียงเครื่องเดียวเท่านั้น จึงควรพัฒนาให้สามารถมีจำนวนผู้ใช้งานได้มากขึ้น
2. การเพิ่มจำนวนสวิตช์ให้มากขึ้นเพื่อให้ครอบคลุมลักษณะการใช้งานของระบบเครือข่ายที่ใช้งานจริง
3. ควรพัฒนาให้สามารถทดลองกับสวิตช์เชื่อมต่อเครือข่ายได้หลายยี่ห้อ
4. ควรพัฒนาให้โปรแกรมสามารถทดลองได้กับอุปกรณ์ที่ใช้งานจริงในระบบเครือข่าย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## บรรณานุกรม

- กิตติ ภัคดีวัฒนะกุล. 2546. **คัมภีร์ Java เล่ม1**. กรุงเทพฯ : เคทีพี คอมพ์ แอนด์ คอนซัลท์.
- จตุชัย แพงจันทร์ และอนุชิต วุฒิพรพงษ์. 2546. **เจาะระบบ Network**. กรุงเทพฯ : ไอทีซี อินโฟ ดิสทริบิวเตอร์ เซ็นเตอร์.
- เจนวิทย์ เหลืองอร่าม,พ.อ. และปิยวิทย์ เหลืองอร่าม. 2545. **การพัฒนา Web Applications ด้วย JavaServer และ Servlet,Java Beans,XML**. กรุงเทพฯ : ชรรวมสาร.
- รุ่งโรจน์ โพนคำ. มปป. **เกาะเปลือกจาวา**. กรุงเทพฯ : จีรวัฒน์ เอ็กซ์เพรส.
- Herbert Schildt. 2001. **Java2 A Beginner's Guide**. California : McGraw-Hill.
- H.M. Deitel and P.J. Deite. 1999. **Java How to Program**.Third Edition. New Jersey: Prentice-Hall.
- H.M. Deitel and P.J. Deitel. 2005. **Java How to Program**.Sixth Edition. New Jersey: Prentice-Hall.
- Lude Cassady and Dorion,ET AL. 1997. **Industrial Strength Java for Serious Java Programmers**. Indianapolis : New Riders Publishing.
- Naba Barkakati,Ph.D. 1999. **Java Annotated Archives**. California : McGraw-Hill.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

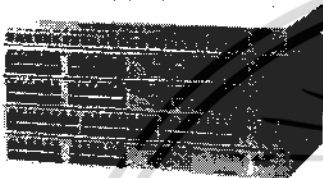


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



## 3Com® SuperStack® 3 Switch 4400 Family

### DATA SHEET



### Key Benefits

#### Performance

Enterprise-class managed Layer 2/4 10/100 switching for workgroups, with a choice of Gigabit downlinks and stacking. The 3Com® SuperStack® 3 Switch 4400 family is available in 24 and 48-port products covering important network interface types including Power over Ethernet, 100BASE-FX, and 100BASE-LX10, with two Gigabit module slots on every unit.

#### Security

Advanced security for network data and management traffic controlling a network.

User network login with IEEE 802.1X and RADIUS, combined with RADIUS Authenticated Device Access (RADA), provides secure flexible network access control at the edge of the network. Local authentication for up to 50 IEEE 802.1X users and RADA devices offers added flexibility in the absence of a centralized RADIUS server, and supplies a backup solution in case of server failure or connectivity issues.

Authenticated users can be placed automatically into a specific VLAN, restricting access further. The Private Ports feature can isolate edge ports to ensure user privacy. Access Control Lists (ACLs) allow or deny traffic based on specific subnet, or individual IP address. SNMP v3 secure encrypted management, Secure Shell (SSH), management VLAN, and management station "trusted IP address" lists help protect the network from rogue management threats.

#### Resiliency

Hot-swappable stacking of up to eight units, or 384 10/100 ports, allows continuous operation of the network. Stack-wide link aggregation (LACP) enables high-performance connectivity of up to 4Gbps to the core of the network, with up to four ports from different units aggregated together across the stack to maximize network uptime.

#### Network Control

Network management is available through an embedded web and command line interface (CLI), or via an SNMP management station. Supported by the range of 3Com management offerings including 3Com Network Supervisor.

#### Multilayer Traffic Shaping

Advanced Multilayer packet classification enables prioritization of mission critical traffic and real-time applications, while freeing bandwidth by eliminating unwanted protocols and applications from the network. Through network login profiles, classification can be dynamically assigned on a per-user basis. Port based rate-limiting enables the maximum bandwidth on each port to be limited, allowing maximum control of network resources while maintaining the full benefit of the powerful traffic prioritization features.

#### Ease of Use

The switches automatically select the optimal speed and duplex mode of connected cables to prevent mis-configuration of the network. They also detect and adjust to cross-over or straight-through cable connections—a feature called auto MDI/MDIX—eliminating the need for specific cables.

#### Warranty

Limited Lifetime Warranty, with Advance Hardware Replacement. See [www.3com.com/warranty](http://www.3com.com/warranty) for details.

#### Service

3Com products are backed by 3Com Global Services and authorized partners with demonstrated expertise in network assessment, implementation, and maintenance. Ask about 3Com's Network Health Check, installation services, and maintenance service packages available in your area.

- Stackable 10/100 Layer 2/4 switches
- 24 and 48-port models, each with two Gigabit module slots
- Network Login combined with RADA for security with flexibility
- Access control lists
- Private Ports facility
- Advanced traffic control and prioritization

3COM® SUPERSTACK® 3 SWITCH 4400 FAMILY DATA SHEET



Tolly Certified: Layer 2/4 Fast Ethernet Switch Performance Evaluation and Layer 2 Interoperability.

For details, see [www.3com.com/other/pdfs/products/en\\_US/ss4400\\_tolly.pdf](http://www.3com.com/other/pdfs/products/en_US/ss4400_tolly.pdf) and [www.3com.com/other/pdfs/products/en\\_US/tolly\\_test\\_4400.pdf](http://www.3com.com/other/pdfs/products/en_US/tolly_test_4400.pdf), respectively.



Tolly Verified for Switch Interoperability.

For details, see [www.tolly.com/TVDetail.aspx?ProductID=93](http://www.tolly.com/TVDetail.aspx?ProductID=93)

3Com SuperStack 3 Switch 4400 family switches provide wirespeed 10/100 Fast Ethernet switching specifically designed for Enterprise workgroup environments. This award-winning switch family boasts advanced Quality of Service features that can reduce the latency of packets through the network, with specific traffic prioritization improving the delivery of converged data such as voice and high quality video over IP.

The switch family has great deployment flexibility, with dual expansion slots to support a full mix of Fast Ethernet fiber and Gigabit Ethernet downlink modules and stacking support of up to eight stacked units. Stacking reduces network management complexity by allowing up to 384 Fast Ethernet ports to be managed as a single entity. The Switch 4400's hot-swappable stacking, with the ability to implement stack-wide aggregated links, provides advanced resiliency for mission critical networks.

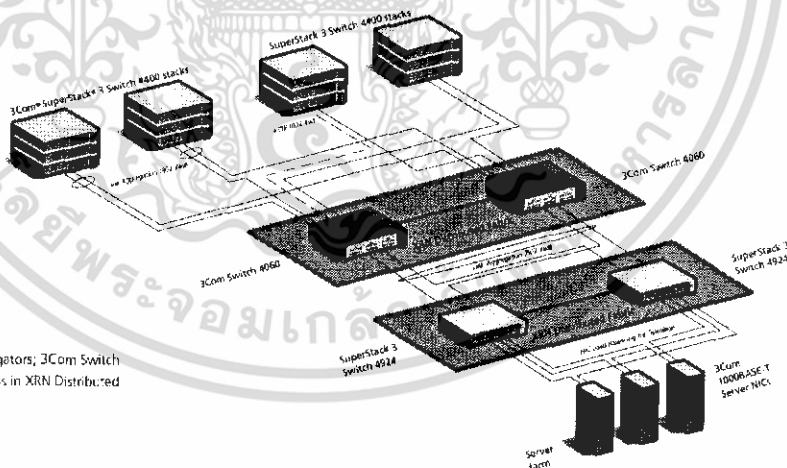
Security is of utmost importance for a network, and the SuperStack 3 Switch 4400 family is a strong contributor to this for workgroups. IEEE 802.1X network login ensures all users are authorized before being granted access to any network resource. User authentication is carried out using any standards-based RADIUS server, avoiding any proprietary authentication mechanisms.

3Com backs these products with more than 25 years of networking experience. They come with a Limited Lifetime product hardware warranty.

The SuperStack 3 Switch 4400 SE 24 Port is an economical alternative if advanced Layer 4 features are not needed initially, a switch that preserves the option of getting these features in the future.

The Switch 4400 SE lacks the traffic classification and prioritization features of the other Switch 4400 family models, while sharing the family's performance, security, stackability (with other Switch 4400 SEs only), and available uplink modules. Enable the advanced features later with the purchase of the SuperStack 3 Switch 4400 SE Enhanced Software Upgrade.

SuperStack 3 Switch 4400s in a 3Com XRN™ Technology-based Network



SuperStack 3 Switch 4400s as aggregators; 3Com Switch 4060s and SuperStack 3 Switch 4924s in XRN Distributed Fabrics.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## 3COM® SUPERSTACK® 3 SWITCH 4400 FAMILY DATA SHEET

### Features

PERFORMANCE	
Switching Capacity	24-port models, 8.8 Gbps, 48-port models, 13.6 Gbps
Forwarding Rate	Forwarding rate: 24-port models, 6.6 Mpps; 48-port models, 10.1 Mpps
Store-and-Forward Switching	Latency <2.6 μs
LAYER 2 SWITCHING	
MAC Address	8K MAC addresses Secure MAC addresses (256 addresses)
VLAN	64 VLANs (IEEE 802.1Q)
Link Aggregation	IEEE 802.3ad (LACP) Four trunk groups (up to four ports in each) Link Aggregation across stack
Auto-negotiation	Auto-negotiation of port speed, duplex, and connection (MDI/MDIX)
Traffic Control	IEEE 802.3x full-duplex flow control Back pressure flow control for half-duplex Broadcast Storm Suppression (3,000 pps threshold)
Spanning Tree	IEEE 802.1D Spanning Tree Protocol (STP) IEEE 802.1w Rapid Spanning Tree Protocol (RSTP) <ul style="list-style-type: none"> <li>Fast-start mode</li> <li>Spanning tree enable/disable per port</li> </ul>
Multicast Snooping	IGMP v1, v2, and v3 snooping IGMP Quener Filtering for 128 multicast groups
STACKING	
Stacking	Up to 384 ports Single IP address for stack management Resilient loop back Hot-swappable
CONVERGENCE	
Priority Queues	Four hardware queues per port Strict priority queuing Weighted Round Robin queuing
Traffic Prioritization*	CoS Marking / Remarking IEEE 802.1p to DSCP mapping Auto classification of 3Com NBX® telephony traffic Priority based on: <ul style="list-style-type: none"> <li>IEEE 802.1p CoS</li> <li>DSCP (DiffServ Code Point)</li> <li>TCP/UDP source or destination port number</li> <li>Default port priority</li> <li>IP Address / Protocol</li> </ul>
Traffic Shaping*	Egress rate limiting, port-based: <ul style="list-style-type: none"> <li>1 Mbps increments (10/100 ports)</li> <li>8 Mbps increments (Gigabit ports)</li> </ul> Application and protocol blocking
SECURITY	
Network Access and User Security	IEEE 802.1X user authentication <ul style="list-style-type: none"> <li>RADIUS authentication</li> <li>Advanced security by locking a port to the MAC address of the authenticated user</li> <li>Automatically assign VLANs and QoS profile to a port based on user</li> <li>Guest VLAN option</li> </ul> RADIUS Authenticated Device Access (RADA) <ul style="list-style-type: none"> <li>Authenticate devices based on MAC address against a RADIUS server</li> <li>Authenticate multiple devices per port</li> <li>Automatically assign VLANs and QoS to a port specific to the devices attached</li> <li>Operates alongside IEEE 802.1X authentication to ensure user and device are allowed access</li> <li>Local authentication for up to fifty IEEE 802.1X users and RADA devices in absence of a centralized RADIUS server</li> <li>RADA "whitelist" support enhances flexibility for IEEE 802.1X/RADA deployment</li> <li>Option to allow default VLAN access when RADIUS server is unavailable</li> </ul> Disconnect Unknown Device (DUD) Access Control Lists (ACLs) allow or deny traffic based on specific subnet or individual IP address Private Ports feature isolates edge ports at layer 2 to ensure user privacy.
Switch Management	Local or RADIUS management of switch passwords Trusted IP Management Addresses Syslog Telnet <ul style="list-style-type: none"> <li>SSH v2 (56 bit or 168 bit DES)</li> </ul>

\* Switch 4400 SE requires 3Com SuperStack 3 Switch 4400 Enhanced Software Upgrade, 3C17207, to perform these functions.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

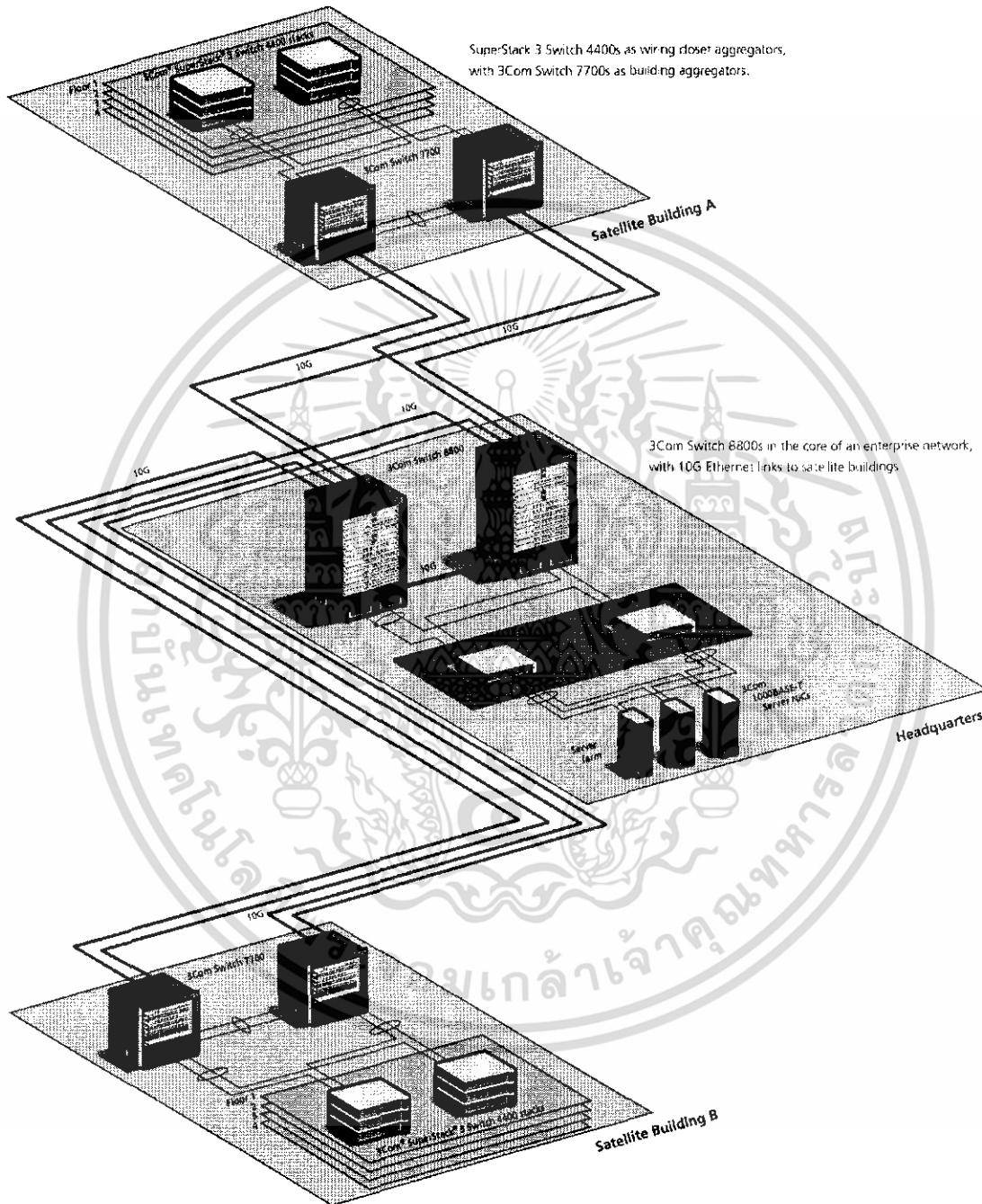
## 3COM® SUPERSTACK® 3 SWITCH 4400 FAMILY DATA SHEET

## Features (continued)

RESILIENCY	
	Support for 3Com Advanced Redundant Power Supply; provides backup power to the switch Backup and restore of switch settings
MANAGEMENT	
Remote Management	SNMP v1, SNMP v2, SNMP v3 secure encrypted management User-selectable management VLAN
Software	Backup and restore TFTP configuration: upload/download TFTP agent upload
Configuration	Command line Serial (9-pin, D-type connector) Telnet Web-based SNMP
Mirror Port / RAP (Roving Analysis Port)	One-to-one
RMON (Remote Monitoring)	Four groups: statistics, history, alarm, and events
IP Address Allocation	DHCP BootP Manual Auto IP
Network Time	Simple Network Time Protocol (SNTP) for time stamp of events captured via Syslog
Management Software	3Com Network Supervisor (copy provided with product) <ul style="list-style-type: none"> <li>• Topology discovery</li> <li>• Change management reporting</li> <li>• Capacity planning</li> <li>• Event logging</li> <li>• Fault identification and troubleshooting</li> <li>• Utilization monitoring</li> </ul>
OPTIONAL SERVICE AND SUPPORT	
Network Health Check	An activity-auditing service focused on improving network performance and productivity Includes traffic monitoring, utilization analysis, problem identification, and asset deployment recommendations Extensive report provides blueprint for action
Network Design	Includes review of business plan, extensive inventory of requirements, and complete design document specifying implementation details
Network Installation	Experts set up and configure equipment and integrate technologies to maximize functionality and minimize business disruption Service may include physical site survey and network design and engineering based on evaluation of business objectives
Project Management	Provides extra focus and resources that special projects demand 3Com engineer(s) manage entire process from initial specifications to post-project review Using structured methodology, requirements are identified, projects planned, and progress of implementation activities tracked
3Com Guardian™ Maintenance Service	Provides comprehensive onsite support, including advance hardware replacement, telephone technical support, and software upgrades: <ul style="list-style-type: none"> <li>• Telephone support backed by powerful call-tracking database and replication laboratory</li> <li>• Software upgrades ensure access to pertinent patches</li> </ul>
3Com Express™ Maintenance Service	Benefits customers who prefer to maintain own hardware Bolsters in-house resources with convenient and speedy access to 3Com hardware replacements, software upgrades, and telephone support

3COM® SUPERSTACK® 3 SWITCH 4400 FAMILY DATA SHEET

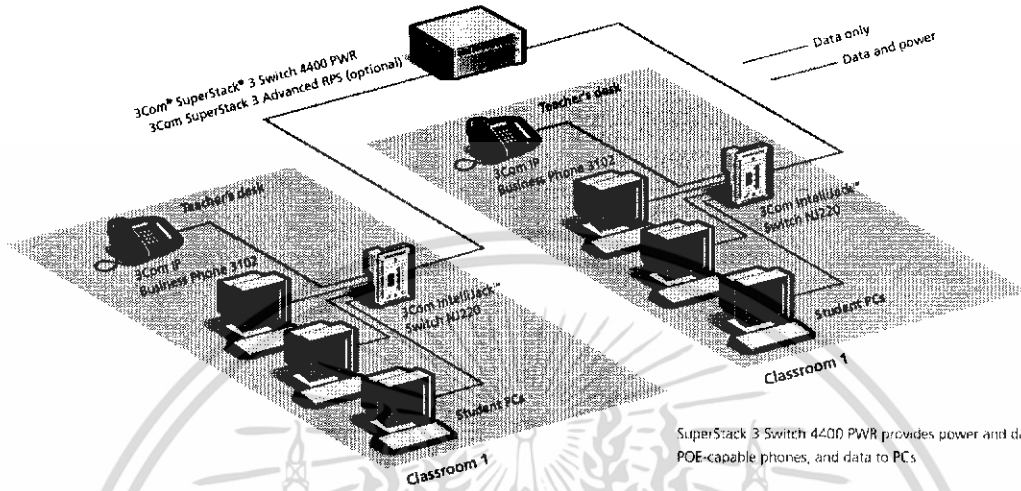
SuperStack 3 Switch 4400s in an Enterprise 10G Network



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

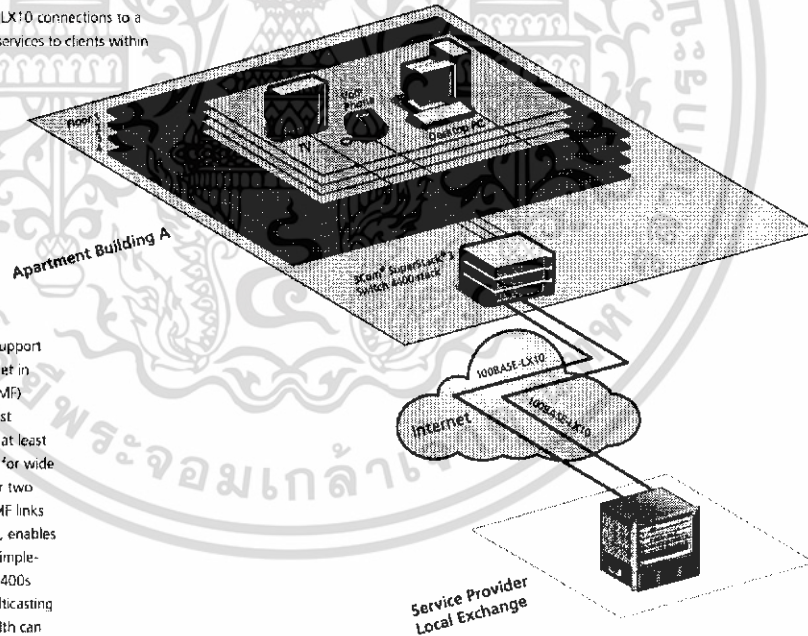
3COM® SUPERSTACK® 3 SWITCH 4400 FAMILY DATA SHEET

SuperStack 3 Switch 4400 PWR and Power over Ethernet



SuperStack 3 Switch 4400 PWR provides power and data to POE-capable phones, and data to PCs

SuperStack 3 Switch 4400, with 100BASE-LX10 connections to a local service provider, provides converged services to clients within a 10km radius.



The SuperStack 3 Switch 4400 adds support for the emerging IEEE 802.3ah Ethernet in First Mile over Point to Point Fiber (EFMF) standard. This enables the reach of Fast Ethernet over fiber to be extended to at least 10km, giving a cost effective solution for wide area Ethernet distribution. Support for two EFMF links per switch, or multiple EFMF links aggregated across a stack of switches, enables a resilient and scalable solution to be implemented. Combined with the Switch 4400's powerful QoS, Traffic Shaping and Multicasting features, the available uplink bandwidth can be intelligently shared between users.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## 3COM® SUPERSTACK® 3 SWITCH 4400 FAMILY DATA SHEET

### Specifications

#### Connectors

24<sup>1</sup> or 48 auto-negotiating 10BASE-T/100BASE-TX ports configured as Auto MDI/MDIX

24 auto-negotiating In-Line Power 10BASE-T/100BASE-FX ports configured as Auto MDI/MDIX (24-port PWR only)

24 100BASE-FX ports (24-port FX only)

2 module slots accommodating media modules or stacking modules

All fiber connectors are MT-RJ (LC connector on SuperStack 3 Switch 4400 100BASE-LX10 Module, 3C17229)

Advanced Redundant Power System Type 2A connector

Advanced Redundant Power System Type 3 connector (24-port PWR only)

R5232 console port

#### Security

RADIUS (RFC 2865, RFC 2869)

Session accounting (RFC 2866)

SSH v2 (SSH Secure Shell, "PuTTY", OpenSSH)

IEEE 802.1X network login

#### Stacking

Up to 384 10/100 front panel ports

Mix and match Switch 4400 24-port (3C17203), 4400 48-port (3C17204), 4400 PWR (3C17205), and 4400 FX (3C17210)

stack Switch 4400 SE (3C17206) only with other like units<sup>1</sup>

#### Performance

##### 24-port

8.8 Gbps forwarding bandwidth

6.6 million packets per second

8,000 MAC addresses supported

##### 48-port

13.6 Gbps forwarding bandwidth

10.1 million packets per second

8,000 MAC addresses supported

#### Reliability

(MTBF @ 40°C)

24-port: 29 years (254,000 hours)

24-port PWR: 26 years (231,000 hours)

24-port FX: 26 years (224,000 hours)

48-port: 20 years (179,000 hours)

#### Dimensions

Height: 43.6 mm (1.7 in or 1U)

Width: 440 mm (17.3 in)

##### Depth:

24-port PWR: 304 mm (12.0 in)

24-port FX: 270 mm (10.6 in)

all others: 274 mm (10.8 in)

##### Weight:

24-port: 2.8 kg (6.2 lb)

24-port PWR: 4.6 kg (10.1 lb)

24-port FX: 2.7 kg (6.0 lb)

48-port: 3.2 kg (7.1 lb)

#### Environmental Requirements

##### Operating Temperature:

0° to 40°C (32° to 104°F)

##### Storage Temperature:

-40° to +70°C (-40° to +158°F)

##### Operating Humidity:

10% to 90% relative humidity noncondensing

Standards: EN60068 (IEC68)

#### Industry Standards Supported

IEEE 802.1D (STP)

IEEE 802.1p (CoS)

IEEE 802.1Q (VLANs)

IEEE 802.1w (RSTP)

IEEE 802.1X (Security)

IEEE 802.3ab (1000BASE-T)

IEEE 802.3ad (Link Aggregation)

IEEE 802.3af (Power over Ethernet)

IEEE 802.3ah (Ethernet in First Mile over Point to Point Fiber - FEMF)

IEEE 802.3i (10BASE-T)

IEEE 802.3u (Fast Ethernet)

IEEE 802.3x (Flow Control)

IEEE 802.3z (Gigabit Ethernet)

#### Safety Agency Certifications

24-port: UL1950, EN60950, CSA22.2 No. 950, IEC 60950

24-port PWR, 24-port FX, and 48-port: UL60950,

EN60950, CSA2.2 No. 60950, IEC 60950

#### Emissions

EN55022 Class A, FCC Part 15 Subpart B Class A,

ICES-003 Class A, VCCI Class A, AS/NZS 3548 Class A,

CNS 13438 Class A

CISPR 22 CLASS A, EN61000-3-2, EN61000-3-3 (24-

port PWR only)

Immunity: EN55024

#### Heat Dissipation:

24-port: 100 W maximum (341 BTU/hr maximum)

24-port PWR: 275 W maximum (938 BTU/hr maximum)

24-port FX: 105 W maximum (358 BTU/hr maximum)

48-port: 120 W maximum (410 BTU/hr maximum)

#### Power Supply

AC Line Frequency 50/60 Hz

Input Voltage 90-240 VAC

Current Rating:

24-port: 2.3A maximum

48-port: 2.8A maximum

#### SNMP Standards

SNMP Protocol (RFC 1157)

MIB-II (RFC 1213)

Bridge MIB (RFC 1493)

RMON MIB II (RFC 2021)

Remote Monitoring MIB (RFC 1757)

Interface MIB (RFC 2233)

Remote Poll MIB (RFC 2925)

MAU MIB (RFC 2668)

SNTP (RFC 2030)

PoE MIB (RFC 484) (24-port PWR only)

#### Management

Web interface

Command line interface

SNMP compatibility

Management through 3Com management applications

- 3Com Network Supervisor

- 3Com Network Director

- 3Com Network Administrator

- 3Com Enterprise Management Suite

#### Warranty

Limited Lifetime Hardware Warranty, Limited

Software Warranty for ninety (90) days. See

[www.3com.com/warranty](http://www.3com.com/warranty) for details.

#### Other Benefits

Other Service Benefits: Advance hardware replacement with same day shipment for North America and Western Europe. Next day shipment for rest of world. Calls must be received by published cut-off times.

90 days of telephone technical support. Limited software updates.

See [www.3com.com/warranty](http://www.3com.com/warranty) for more detail.

Register products at <http://eSupport.3com.com>.

#### Service

Americas:

[www.3com.com/products/en\\_US/global\\_services](http://www.3com.com/products/en_US/global_services)

International:

<http://emea.3com.com/globalservices>

(1) 24-port: specifications refer to Switch 4400 24-Port (3C17203), Switch 4400 24-Port PWR (3C17205), Switch 4400 24-Port FX (3C17210), and Switch 4400 SE 24-Port (3C17206), unless otherwise noted.

(2) With installation of optional SuperStack 3 Switch 4400 Enhanced Software Upgrade, 3C17207, the Switch 4400 SE will stack with the Switch 4400 24-port, 4400 48-port, 4400 PWR, and 4400 FX.

## 3COM® SUPERSTACK® 3 SWITCH 4400 FAMILY DATA SHEET

## Ordering Information

PRODUCT DESCRIPTION	3COM SKU	
3Com SuperStack 3 Switch 4400 24-port	3C17203	
3Com SuperStack 3 Switch 4400 48-port	3C17204	
3Com SuperStack 3 Switch 4400 PWR	3C17205	
3Com SuperStack 3 Switch 4400 SE 24-port	3C17206	
3Com SuperStack 3 Switch 4400 FX	3C17210	
<b>Optional Modules and Accessories</b>		
3Com SuperStack 3 Switch 4400 Enhanced Software Upgrade	3C17207	
3Com SuperStack 3 Switch 4400 1000BASE-T Module	3C17220	
3Com SuperStack 3 Switch 4400 1000BASE-SX Module	3C17221	
3Com SuperStack 3 Switch 4400 100BASE-FX Module	3C17222	
3Com SuperStack 3 Switch 4400 100BASE-LX Module	3C17223	
3Com SuperStack 3 Switch 4400 100BASE-LX10 Module*	3C17229	
3Com SuperStack 3 Switch 4400 Stacking Kit (Two stacking modules and cable to stack two Switch 4400s—see stacking specifications above for restrictions)	3C17227	
3Com SuperStack 3 Switch 4400 Stack Extender Kit (One stacking module, one cascade module and one cable for adding more Switch 4400s to an existing Switch 4400 stack—see stacking specifications above for restrictions)	3C17228	
3Com SuperStack 3 Advanced Redundant Power System	3C160718	
3Com SuperStack 3 Advanced RPS Type 2A module	3C16074A	
3Com SuperStack 3 Advanced RPS Type 3 module	3C16075	
<b>Express™ Service Plans†</b>		
	<b>Americas</b>	<b>EMEA / AP‡</b>
3Com SuperStack 3 Switch 4400 24-port	3CS-EXP-05E	3CS-EXP-X305
3Com SuperStack 3 Switch 4400 48-port	3CS-EXP-08E	3CS-EXP-X309
3Com SuperStack 3 Switch 4400 PWR	3CS-EXP-06E	3CS-EXP-X307
3Com SuperStack 3 Switch 4400 FX	3CS-EXP-09E	3CS-EXP-X310
3Com SuperStack 3 Switch 4400 SE	3CS-EXP-03E	3CS-EXP-X303

\* Switch 4400 software version 5.1, minimum, required for operation.

† Includes telephone support, advance hardware replacement, and software upgrades.

‡ Europe, Middle East, Africa and Asia Pacific.



3Com Corporation, Corporate Headquarters, 350 Camous Drive, Marlborough, MA 01752-3064

To learn more about 3Com solutions, visit [www.3com.com](http://www.3com.com). 3Com is publicly traded on NASDAQ under the symbol COMS.

Copyright © 2005 3Com Corporation. All rights reserved. 3Com, the 3Com logo, NBE, and SuperStack are registered trademarks of 3Com Corporation. XRN is a trademark of 3Com Corporation. All other company and product names may be trademarks of their respective companies. While every effort is made to ensure the information given is accurate, 3Com does not accept liability for any errors or omissions which may arise. Specifications and other information in this document may be subject to change without notice. 400674-016 06/05

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



ภาคผนวก ข  
รหัสต้นฉบับของโปรแกรม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## โปรแกรมที่ติดต่อกับสวิตช์

```

package project;

import javax.comm.*;
import java.io.*;
import java.util.*;

public class SerialReadWrite{
    private static CommPortIdentifier    portId ;
    private static String                messageString = "B";
    private static Enumeration           portList;
    private DataInputStream              inputStream;
    private PrintStream                  outputStream;
    private SerialPort                   serialPort;
    private Thread                       readThread;
    private String                       theMessage;
    private String                       defaultPort;
    private boolean                      portOpen = false;

    public SerialReadWrite() throws PortInUseException,
PortNotFoundException {
        boolean    portFound = false;
        defaultPort = "COM1";
        portList = CommPortIdentifier.getPortIdentifiers();
        while (portList.hasMoreElements()) {
            portId = (CommPortIdentifier) portList.nextElement();
            if (portId.getPortType() ==
CommPortIdentifier.PORT_SERIAL) {
                if (portId.getName().equals(defaultPort)) {
                    portFound = true;
                }
            }
        }

        if (!portFound) {
            throw new PortNotFoundException();
        }
        serialPort = (SerialPort) portId.open("SimpleReadApp", 2000);

    try {
        inputStream = new
DataInputStream(serialPort.getInputStream());
        outputStream = new
PrintStream(serialPort.getOutputStream(), true);
    } catch (IOException e) {
    }

    try {
        serialPort.setSerialPortParams(9600,
SerialPort.DATABITS_8,
SerialPort.STOPBITS_1,
SerialPort.PARITY_NONE);
    } catch (UnsupportedCommOperationException e)

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

{
    }
    portOpen = true;
}

public SerialReadWrite(String port) throws PortInUseException,
PortNotFoundException {
    boolean    portFound = false;
    defaultPort = port;
    portList = CommPortIdentifier.getPortIdentifiers();
    while (portList.hasMoreElements()) {
        portId = (CommPortIdentifier) portList.nextElement();
        if (portId.getPortType() ==
CommPortIdentifier.PORT_SERIAL) {
            if (portId.getName().equals(defaultPort)) {
                portFound = true;
            }
        }
    }

    if (!portFound) {
        throw new PortNotFoundException();
    }
    serialPort = (SerialPort) portId.open("SimpleReadApp", 2000);
    try {
        inputStream = new
DataInputStream(serialPort.getInputStream());
        outputStream = new
PrintStream(serialPort.getOutputStream(), true);
    } catch (IOException e) {
    }

    try {
        serialPort.setSerialPortParams(9600,
SerialPort.DATABITS_8,
        SerialPort.STOPBITS_1,
        SerialPort.PARITY_NONE);
    } catch (UnsupportedCommOperationException e) {
    }
    portOpen = true;
}

public byte[] read() throws IOException {
    byte [] nums = new byte[2];
    inputStream.read(nums);
    return nums;
}

public void write(int output) {
    outputStream.print(output);
}

public void write(String output) {
    outputStream.print(output);
}
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

public String writeOut() {
    int turnflag = 0;
    outputStream.print(turnflag);
    outputStream.print("Z");
    return "Sent 0 then Z.";
}

public void stop() throws IOException{
    inputStream.close();
    outputStream.close();
    serialPort.close();
}

public boolean wasPortOpened() {
    return portOpen;
}
}

```

### โปรแกรมในส่วนของผู้ใช้งานที่เป็น Applet

```

import java.io.*;
import java.net.*;
import java.awt.*;
import java.awt.event.*;
import javax.swing.*;
import java.applet.*;

public class Client extends Applet
private JTextField enter;
private JTextArea display;
ObjectOutputStream output;
ObjectInputStream input;
String message = "";

public void init()
{
    display = new JTextArea ("",20,50);
    enter = new JTextField("",50);
    enter.setEnabled(false);
    enter.addActionListener(
        new ActionListener() {

            public void actionPerformed (ActionEvent e)
            {
                sendData (e.getActionCommand());
            }
        }
    );

    add(display);
    add (new JScrollPane (display),BorderLayout.CENTER);
    add (enter, BorderLayout.SOUTH);
    show();
}

Socket client;

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

    try
    {
        client = new Socket(
InetAddress.getByName("161.246.14.142"),5001);
        output = new ObjectOutputStream(client.getOutputStream()
);
        output.flush();
        input = new ObjectInputStream(client.getInputStream() );
        enter.setEnabled(true);

        do
        {
            try
            {
                message = (String) input.readObject ();
                display.append ("\n"+ message );
                display.setCaretPosition (
                    display.getText().length() );
            }
            catch (ClassNotFoundException cnfex)
            {
                display.append("\nUnknown object type
received");
            }
        }
        while (!message.equals("SERVER>>> TERMINATE" ));
        display.append("Close connection.\n");
        input.close();
        output.close();
        client.close();
    }
    catch (EOFException eof)
    {
        System.out.println("Server terminated connection" );
    }
    catch (IOException e){
        e.printStackTrace();
    }
}

private void sendData (String s)
{
    try
    {
        output.writeObject("Client>>> " + s);
        output.flush();
        enter.setText("");
        display.append("\nClient>>>"+s);
    }
    catch (IOException cnfex)
    {
        display.append( "\nError writing object" );
    }
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## โปรแกรมในส่วนของเครื่องผู้ใช้งานที่เป็น HTML

```

}
}
}

<HTML>
<HEAD>
<TITLE> Client Config Switch On Internet </TITLE>
<META NAME="Generator" CONTENT="EditPlus">
<META NAME="Author" CONTENT="">
<META NAME="Keywords" CONTENT="">
<META NAME="Description" CONTENT="">
<style type="text/css">
<!--
.style1 {
    font-family: "Courier New", Courier, mono;
    color: #00FFFF;
}
.style2 {color: #FF0000}
.style8 {color: #0000FF}
.style9 {font-family: "Courier New", Courier, mono}
-->
</style>
</HEAD>

<BODY bgcolor="#FFFFCC"><center>

    <h1 class="style1 style2 style9">CLIENT CONFIG TO SWITCH ON
INTERNET </h1>
    <p>&nbsp;</p>
    <p>
        <applet code="Client.class" height=355 width=556>
        </applet>
    </p>
    <p>&nbsp;</p>
    <h1 class="style8"> <span class="style9">ENGINEER EDUCATION FROM
KMIT'L</span> </h1>
    <p>&nbsp;</p>
</center>
</BODY>
</HTML>

```

## โปรแกรมในส่วนของเครื่องเซิร์ฟเวอร์

```

package project;

import java.io.*;
import java.net.*;
import java.awt.*;
import java.awt.event.*;
import javax.swing.*;
import java.lang.*;
import java.util.*;
import edu.berkeley.guir.ptk.util.*;

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

public class Server extends JFrame
{

    private JTextField enter;
    private JTextArea display;
    ObjectOutputStream output;
    ObjectInputStream input;
    SerialReadWrite rw;
    String ms,message;

    public Server()
    {
        super ("Server");

        Container c = getContentPane();
        enter = new JTextField();
        enter.setEnabled(false);
        enter.addActionListener(
            new ActionListener() {
                public void actionPerformed (ActionEvent e)
                {
                    sendData (e.getActionCommand());
                }
            }
        );
        c.add (enter, BorderLayout.SOUTH);
        display = new JTextArea ();
        c.add (new JScrollPane (display),BorderLayout.CENTER);

        setSize (500,250);
        show();
    }
    public void runServer()
    {
        ServerSocket server;
        Socket connection;
        int counter = 1;

        try
        {
            server = new ServerSocket (5001,100);

            while (true)
            {
                connection = server.accept();

                output = new
ObjectOutputStream(connection.getOutputStream());
                output.flush();
                input = new
ObjectInputStream(connection.getInputStream());
                message = "SERVER>>> Connection successful\n";
                output.writeObject (message);
                output.flush();
                enter.setEnabled(true);
            }
        }
    }
}

```

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์ไว้สำหรับใช้เพื่อการศึกษาเท่านั้น ไม่อนุญาตให้เผยแพร่ไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

do
{
    rw = new SerialReadWrite(750);

    try
    {
        message = (String) input.readObject();
        System.out.println(message);

        rw.openSource();
        rw.writePacket(message+"\n\r");
        display.append ("\n"+message);
        display.setCaretPosition(
            display.getText().length());

        byte[] mesR = new byte[20];
        mesR = rw.readPacket();
        String ms = new String(mesR);
        sendData(ms);
        display.append ("\n"+ms);
        display.setCaretPosition(
            display.getText().length());
    }
    catch (ClassNotFoundException cnfex)
    {
        display.append("\nUnknow object type
received");
    }
}

while (!message.equals("CLIENT>>> TERMINATE"));
display.append("\nUser terminated connection");
enter.setEnabled(false);
output.close();
input.close();
connection.close();
rw.closeSource();
++counter;
}
}
catch (EOFException eof)
{
    System.out.println("Client terminate connection");
}
catch (IOException io){
io.printStackTrace();
}
}

private void sendData (String s)
{
    try
    {
        output.writeObject(s);
        output.flush();
        enter.setText("");
        display.append(s);
    }
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้拿去ใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

        catch (IOException cnfex)
        {
            display.append( "\nError writing object" );
        }
    }

public static void main (String args[])
{
    Server app = new Server();

    app.addWindowListener(
        new WindowAdapter() {
            public void windowClosing (WindowEvent e)
            {
                System.exit(0);
            }
        }
    );
    app.runServer();
}

```

### โปรแกรมส่วนที่ติดต่อผ่าน Serial Port

```

package project;

import edu.berkeley.guir.ptk.util.SerialReadWrite;
import javax.comm.*;

public class TestSerial {

    public static void main(String args[]) {
        SerialReadWrite rw = new SerialReadWrite(600);
        rw.openSource();
        rw.writePacket("bridge\n\r");
        byte[] data = new byte[790];
        data = rw.readPacket();
        String ms = new String(data);
        System.out.println("Suscess");
        rw.closeSource();
        System.out.print(ms);
    }
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

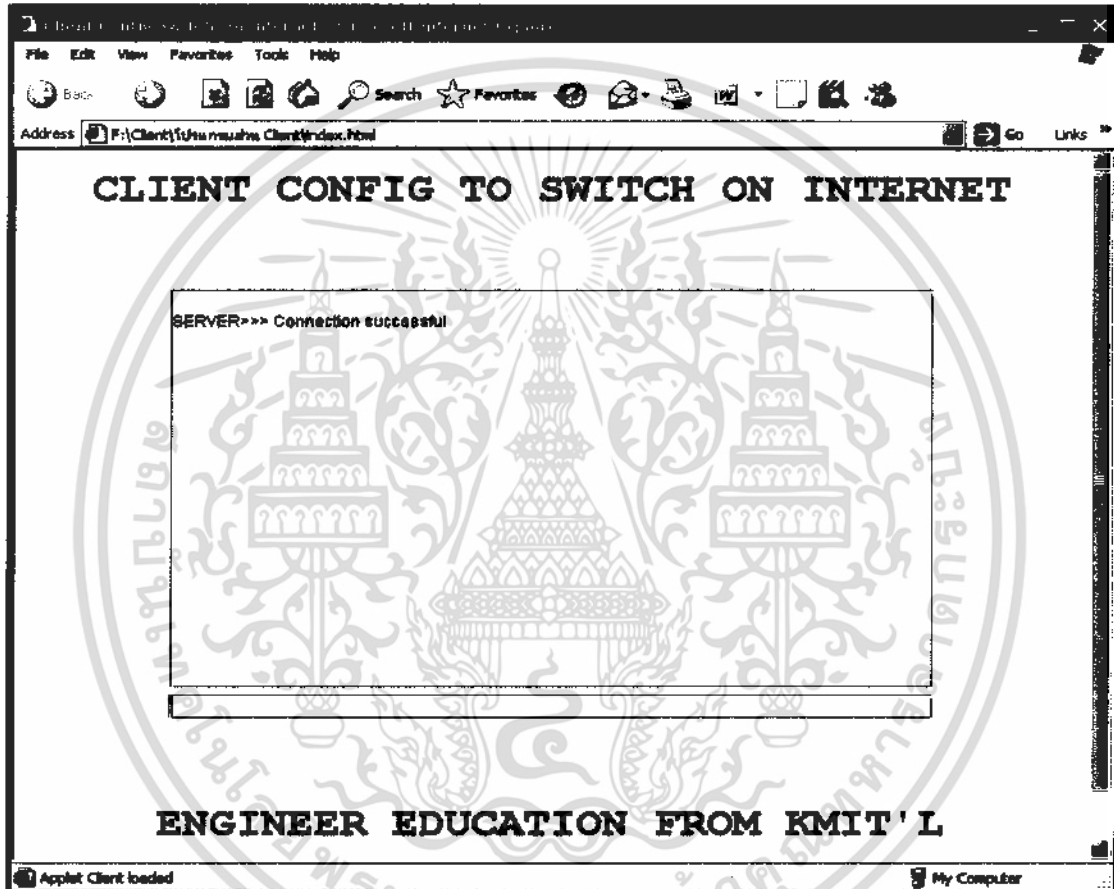


**ภาคผนวก ค**  
**คู่มือการใช้งาน**

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## คู่มือการใช้งาน

### โปรแกรมการทดลองเครือข่ายท้องถิ่นแบบเสมือนบนระบบอินเทอร์เน็ต



ภาควิชาครุศาสตร์วิศวกรรม

คณะครุศาสตร์อุตสาหกรรม

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2548

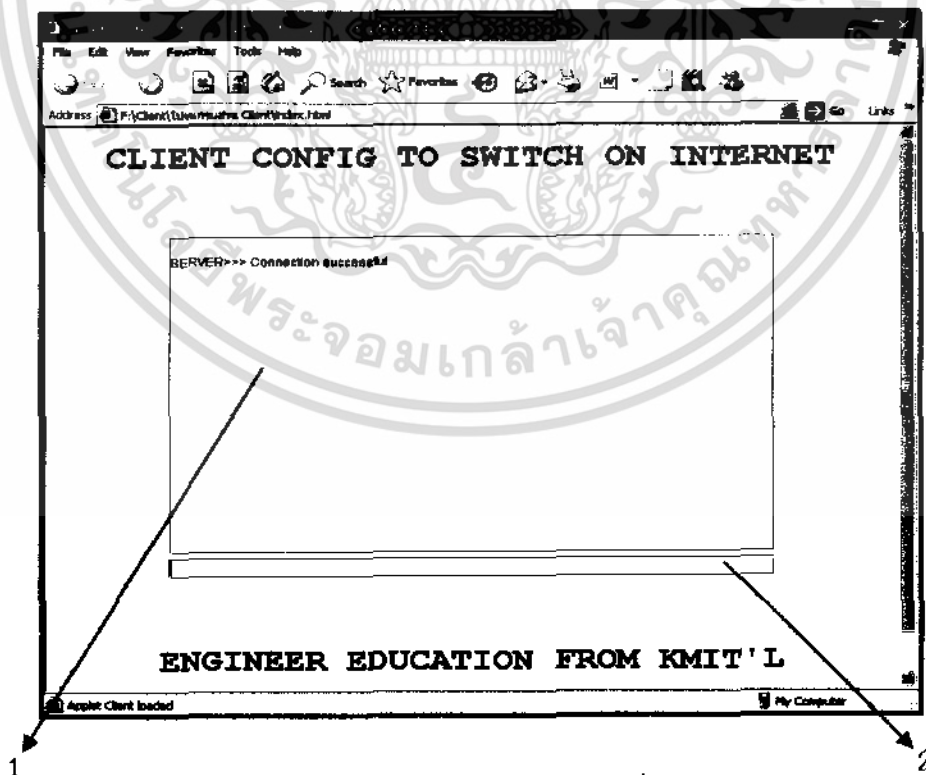
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## 1. คำแนะนำเบื้องต้น

ก่อนที่จะลงมือใช้งานโปรแกรมการทดลองเครือข่ายท้องถิ่นแบบเสมือนบนระบบอินเทอร์เน็ต ควรทำการศึกษาการใช้งานจากคู่มือให้เข้าใจ เพื่อการใช้งานที่ถูกต้อง และเพื่อป้องกันปัญหาที่จะเกิดขึ้นมาในภายหลังได้

## 2. วิธีใช้งาน

1. ที่เครื่องเซิร์ฟเวอร์จะต้องเปิดเครื่องเซิร์ฟเวอร์ และ Start เซิร์ฟเวอร์ก่อน
  2. เปิด Hyper Terminal ที่เครื่องเซิร์ฟเวอร์และทำการล็อกอิน ว่า Admin ส่วนพาสเวิร์ดก็กด Enter ได้เลย เสร็จแล้วก็ทำการ Disconnect หรือปิดโปรแกรมเลยก็ได้
  3. แล้วทำการรันไฟล์ Server.java ไว้ที่เครื่องเซิร์ฟเวอร์
  4. ที่เครื่องผู้ใช้งานทำการเปิดโปรแกรม Internet Explorer ที่ได้ติดตั้งไว้ที่เครื่องผู้ใช้งานแล้ว
  5. คลิกเปิดที่ Index.html บนหน้าจอบราวเซอร์ และบราวเซอร์ที่อยู่บนจอภาพก็จะปรากฏหน้าจอหลักของโปรแกรมของเครื่องผู้ใช้งาน
3. เมื่อเครื่องผู้ใช้งานเชื่อมการติดต่อกับเครื่องเซิร์ฟเวอร์ได้ก็จะแสดงคำว่า SERVER>>> Connection Successful ดังรูปที่ ค.1



รูปที่ ค.1 หน้าจอหลักของโปรแกรมของเครื่องผู้ใช้งาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2. เมื่อเปิดโปรแกรมแล้วก็เริ่มทำการทดลอง ได้โดยตั้งค่าต่างๆ ให้แก่อุปกรณ์สวิตซ์ซึ่งสามารถอธิบายส่วนการใช้งานดังนี้

2.1 หมายเลข 1 หมายถึง ส่วนแสดงผลที่ทำการตั้งค่าต่างๆ ไปและเป็นส่วนที่ใช้แสดงผลของอุปกรณ์สวิตซ์ที่ตอบกลับมา

2.2 หมายเลข 2 หมายถึง ส่วนที่ใช้พิมพ์คำสั่งการตั้งค่าต่างๆ ออกไปและเมื่อพิมพ์คำสั่งเสร็จก็กด Enter

## 2. การแก้ปัญหาเบื้องต้น

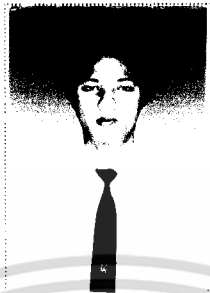
เมื่อผู้ใช้งานประสบกับปัญหาในการใช้โปรแกรมการทดลองเครือข่ายท้องถิ่นแบบเสมือนบนระบบอินเทอร์เน็ต ผู้ใช้งานสามารถแก้ไขปัญหาเบื้องต้นได้ดังนี้

### ตารางที่ ค.1 การแก้ปัญหาเบื้องต้น

อาการ	สาเหตุและวิธีแก้ไข
โปรแกรมรันไม่ได้	ตรวจสอบสายเชื่อมต่ออินเทอร์เน็ตต่ออยู่หรือไม่
โปรแกรมติดต่อกับเครื่องเซิร์ฟเวอร์ไม่ได้	ดูที่เครื่องเซิร์ฟเวอร์ว่าโปรแกรมทำงานอยู่หรือไม่ หรือดูว่าเซิร์ฟเวอร์ Start หรือไม่
โปรแกรมตั้งค่าต่างๆ ไปยังอุปกรณ์สวิตซ์ไม่ได้	ตรวจสอบสายที่เชื่อมต่อกับอุปกรณ์สวิตซ์ว่าหลวมหรือไม่ เปิดอุปกรณ์สวิตซ์หรือยัง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## ประวัติผู้แต่ง



ชื่อ-สกุล

นายโกสินทร์ สังข์นุ่น

วัน เดือน ปีเกิด

๘ เมษายน พ.ศ. ๒๕๒๕

ภูมิลำเนา

บ้านเลขที่ ๘/๑๙ หมู่ ๑ ตำบลท่าข้าม  
อำเภอสามพราน จังหวัดนครปฐม  
รหัสไปรษณีย์ ๗๓๑๑๐

ประวัติการศึกษา

ประถมศึกษา

โรงเรียนนักบุญเปโตร จังหวัดนครปฐม

มัธยมศึกษาตอนต้น

โรงเรียนนักบุญเปโตร จังหวัดนครปฐม

ประกาศนียบัตรวิชาชีพ

วิทยาลัยเทคนิคนครปฐม จังหวัดนครปฐม

ประกาศนียบัตรวิชาชีพชั้นสูง

สถาบันเทคโนโลยีราชมงคลวิทยาเขตพระนครเหนือ

ปริญญาตรี

สาขาวิชา คอมพิวเตอร์

ภาควิชา วิศวกรรมศาสตร์วิศวกรรม

คณะวิศวกรรมศาสตร์อุตสาหกรรม

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ผลที่ได้รับรางวัล

-

ทุนการศึกษา

-

ความสนใจพิเศษ

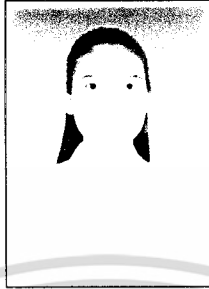
คอมพิวเตอร์, ระบบเครือข่าย

คติพจน์

-

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## ประวัติผู้แต่ง



ชื่อ-สกุล	นางสาวทัศนีย์ จัตุมัทธ
วัน เดือน ปีเกิด	๒๔ พฤศจิกายน พ.ศ. ๒๕๒๕
ภูมิลำเนา	บ้านเลขที่ ๑๖๗ หมู่ ๔ ตำบลสวาย อำเภอเมือง จังหวัดสุรินทร์ รหัสไปรษณีย์ ๓๒๐๐๐
ประวัติการศึกษา	โรงเรียนบ้านสวาย จังหวัดสุรินทร์ โรงเรียนวิวัฒน์โยธิน จังหวัดสุรินทร์ โรงเรียนสุรวิทยาคาร จังหวัดสุรินทร์ วิทยาลัยเทคนิคสุรินทร์ จังหวัดสุรินทร์ สาขาวิชา คอมพิวเตอร์ ภาควิชา ครุศาสตร์วิศวกรรม คณะครุศาสตร์อุตสาหกรรม สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
ผลที่ได้รับรางวัล	-
ทุนการศึกษา	ทุนทำงาน 50 ชั่วโมง คณะครุศาสตร์อุตสาหกรรม
ความสนใจพิเศษ	ระบบเครือข่าย
คติพจน์	ฝันได้ แต่ต้องไปให้ถึงฝัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## ประวัติผู้แต่ง



ชื่อ-สกุล	ว่าที่ร้อยตรี ธนภัทร เต็มเปี่ยม
วัน เดือน ปีเกิด	๗ พฤศจิกายน พ.ศ. ๒๕๒๖
ภูมิลำเนา	บ้านเลขที่ ๑๕๙ หมู่ ๒ ตำบลนางตะเคียน อำเภอเมือง จังหวัดสมุทรสงคราม รหัสไปรษณีย์ ๗๕๐๐๐
ประวัติการศึกษา	
ประถมศึกษา	โรงเรียนวัดลาดเป้ง จังหวัดสมุทรสงคราม
มัธยมศึกษาตอนต้น	โรงเรียนวัดลาดเป้ง จังหวัดสมุทรสงคราม
ประกาศนียบัตรวิชาชีพ	วิทยาลัยเทคนิคสมุทรสงคราม จังหวัดสมุทรสงคราม
ประกาศนียบัตรวิชาชีพชั้นสูง	วิทยาลัยเทคนิคสมุทรสงคราม จังหวัดสมุทรสงคราม
ปริญญาตรี	สาขาวิชา คอมพิวเตอร์ ภาควิชา วิศวกรรมศาสตร์วิศวกรรม คณะครุศาสตร์อุตสาหกรรม สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
ผลที่ได้รับรางวัล	รางวัลรองชนะเลิศอันดับสองการประกวดเรียงความเรื่องการ อนุรักษ์พลังงาน สจล.
ทุนการศึกษา	ทุนทำงาน 50 ชั่วโมง คณะครุศาสตร์อุตสาหกรรม
ความสนใจพิเศษ	ด้านระบบเครือข่าย โปรแกรมเมอร์
คติพจน์	ทำวันนี้ให้ดีที่สุด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้