

สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

โปรแกรมกำหนดค่าการทำงานอุปกรณ์เครือข่าย

Network Device Configuration Software

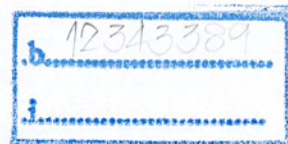


T117394



บังฉิมนุช สุกาญจน์ไพโร  
สมยาพร ยิงเจริญ  
สุกฤษณ์ ชื่นอิม

เลขทะเบียน 117394  
วันเดือนปี 1 ต.ค. 2554



ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต

สาขาวิชาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2553

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ปริญญาานิพนธ์ปีการศึกษา 2553

สาขาวิชาวิศวกรรมคอมพิวเตอร์

คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เรื่อง โปรแกรมกำหนดค่าการทำงานอุปกรณ์เครือข่าย

NETWORK DEVICE CONFIGURATION SOFTWARE

ผู้จัดทำ

1. นางสาวปัจฉิมนุช สุกาญจนไพโร รหัสนักศึกษา 50010944

2. นายสมชายพร ยิ่งเจริญ รหัสนักศึกษา 50011630

3. นายสุกฤษณ์ ชื่นอ้อม รหัสนักศึกษา 50011692





(อาจารย์จรัสศักดิ์ สิทธิกร)

อาจารย์ที่ปรึกษา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

# โปรแกรมกำหนดค่าการทำงานอุปกรณ์เครือข่าย

นางสาวปัจฉิมนุช สุกาญจนไพโร 50010944

นายสมยาพร ยิ่งเจริญ 50011630

นายสุกฤษฎ์ ชื่นอ้อม 50011692

อาจารย์จรัสศักดิ์ สิทธิกร อาจารย์ที่ปรึกษา  
ปีการศึกษา 2553

## บทคัดย่อ

ในปัจจุบันการกำหนดค่าการทำงานอุปกรณ์ในเครือข่ายจะต้องทำการส่งคำสั่งผ่านทาง Console Port โดยใช้ Command Line Interface ซึ่งการกำหนดค่าการทำงานสำหรับอุปกรณ์นั้นจะต้องทำการศึกษาคำสั่งที่จะต้องใช้กับอุปกรณ์แต่ละชนิด ทำให้ผู้ที่ต้องการใช้งานอุปกรณ์เครือข่ายนอกจากต้องมีความรู้พื้นฐานทางด้านเครือข่ายแล้วจะต้องศึกษาการใช้งานคำสั่งต่างๆ เพื่อกำหนดค่าการทำงาน ปริมาณนี้ยังมีแนวคิดที่จะปรับปรุงรูปแบบการกำหนดค่าการทำงานอุปกรณ์ในระบบเครือข่ายโดยผ่านทางโปรแกรมที่พัฒนาขึ้นมา ซึ่งทำให้ผู้ใช้มีความสะดวกและง่ายในการใช้งานมากกว่าการกำหนดค่าการทำงานโดยใช้ Command Line Interface และสามารถแสดงผลการเชื่อมต่อออกมาเป็นรูปแบบกราฟฟิก จำลองระบบออกมาให้ผู้ใช้ได้เห็นการเชื่อมต่อ ทำให้ง่ายต่อการศึกษาระบบเครือข่าย นอกจากนี้ยังมีส่วนแจ้งเตือนความผิดพลาดในกรณีที่ผู้ใช้กำหนดค่าการทำงานผิดพลาดซึ่งอาจจะทำให้ระบบไม่สามารถเชื่อมต่อกันได้

# Network Device Configuration Software

Ms. Padchimmuch Sukarnjanaprai 50010944

Mr. Somyaporn Yingcharoen 50011630

Mr. Sukrit Chuen-im 50011692

Mr. Jirasak Sittikorn Advisor

Academic Year 2010

## ABSTRACT

Currently, configuring network devices will need to send commands through the Console Port using the Command Line Interface which is configured to work for each device type will have to study the command to be used with each device. The user who want to work with network equipment also must have basic knowledge of the network shall use the commands. To determine the value of work. This project is an idea to improve the form of configuration devices on the network through the application program, which gives users a convenient and easy to use, rather than configuration using the Command Line Interface and can show a connection out in the graphic simulation. It easier to study Network. There is also the fault in a user-defined error value, which may be made. The system can't connect to each other.

## กิตติกรรมประกาศ

ปริญญานิพนธ์ฉบับนี้สำเร็จได้อย่างดีด้วยคำแนะนำจากอาจารย์จรัสศักดิ์ สิทธิกร และสาขาวิชาวิศวกรรมคอมพิวเตอร์ ที่ให้โอกาสเข้าไปทดสอบกับอุปกรณ์จริงในห้องทดลอง เพื่อพัฒนาโปรแกรมกำหนดค่าการทำงานอุปกรณ์เครือข่าย

ขอกราบขอบพระคุณท่านคณาจารย์สาขาวิชาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง และพี่ๆ เพื่อนๆ น้องๆ ณ ห้องวิจัยเน็ตเวิร์กแห่งนี้ ทุกคนที่ให้คำแนะนำต่างๆ และคอยให้กำลังใจเสมอมา

สุดท้ายนี้ขอกราบขอบพระคุณ บิดา มารดา และครอบครัวที่เป็นกำลังใจให้ และให้การสนับสนุนในทุกเรื่อง ทำให้สามารถทำปริญญานิพนธ์ฉบับนี้สำเร็จลุล่วงด้วยดี คุณค่าและประโยชน์อันพึงมาจากปริญญานิพนธ์ฉบับนี้ ขอมอบแด่ผู้มีพระคุณทุกๆ ท่าน

ปัจฉิมบท สุกัญญาไพโร  
สมยาพร ยิ่งเจริญ  
สุกฤษณ์ ชื่นอิม

# สารบัญ

	หน้า
บทคัดย่อภาษาไทย.....	I
บทคัดย่อภาษาอังกฤษ.....	II
กิตติกรรมประกาศ.....	III
สารบัญ.....	IV
สารบัญรูป.....	VI
บทที่ 1 บทนำ.....	1
1.1 ความเป็นมาของปัญหา.....	1
1.2 หลักการและเหตุผล.....	1
1.3 วัตถุประสงค์.....	1
1.4 ผลที่จะได้รับจากการทำโครงการ.....	2
1.5 ขอบเขตของโครงการ.....	2
1.6 ข้อยกเว้นของโปรแกรมที่พัฒนา.....	2
1.7 ส่วนประกอบของโครงการ.....	2
บทที่ 2 ทฤษฎีและหลักการที่เกี่ยวข้อง.....	4
2.1 การทำงานของเราเตอร์.....	4
2.2 คำสั่งที่ใช้ในการทำงานของเราเตอร์.....	6
2.3 หลักการและโปรโตคอลที่เกี่ยวข้อง.....	12
2.4 การค้นหาแบบกว้างก่อน (Breadth first search).....	14
บทที่ 3 การออกแบบและโครงสร้างของโปรแกรม.....	16
3.1 ภาษาและเครื่องมือที่ใช้ในการพัฒนา.....	16
3.2 อินพุตและเอาพุตที่เข้าและออกจากระบบ.....	16

## สารบัญ (ต่อ)

	หน้า
3.3 หลักการทำงานของโปรแกรม .....	17
3.4 การวิเคราะห์ความต้องการของระบบ .....	19
<b>บทที่ 4 ผลการทดลองการใช้โปรแกรม.....</b>	<b>27</b>
4.1 ตัวอย่างหน้าจอโปรแกรม .....	27
4.2 การรับส่งข้อมูลผ่านพอร์ตอนุกรม.....	52
4.3 การนำข้อมูลที่รับส่งผ่านพอร์ตอนุกรมเขียนลงไฟล์.....	54
<b>บทที่ 5 บทสรุป .....</b>	<b>55</b>
5.1 บทสรุป.....	55
5.2 วิจารณ์สิ่งที่ได้จากโครงการ.....	55
5.3 ปัญหา อุปสรรค และแนวทางการแก้ไข .....	56
5.4 แนวทางการพัฒนาต่อ.....	56
<b>บรรณานุกรม.....</b>	<b>57</b>

# สารบัญรูป

รูป	หน้า
2.1 เราท์เตอร์ .....	4
2.2 การแบ่งแยกประเภทของ Dynamic Routing Protocols .....	9
2.3 ลำดับการเดินทางบนโหนดของการค้นหาแบบกว้างก่อนบนโครงสร้างต้นไม้ .....	14
2.4 โครงสร้างข้อมูลแบบกราฟ .....	14
2.5 ลำดับการค้นหาแบบกว้างก่อน .....	15
3.1 การออกแบบระบบเครือข่าย .....	17
3.2 การตั้งค่าเริ่มต้นให้กับเราท์เตอร์ .....	18
3.3 การตั้งค่าให้ระบบเครือข่าย .....	19
3.4 แผนภาพ UsecaseDiagram ของ Network Device Configuration Software .....	20
3.5 การแบ่งแพ็คเกจ .....	23
3.6 คลาสในแพ็คเกจ Topology .....	24
3.7 คลาสในแพ็คเกจ Initial_configuration .....	24
3.8 คลาสในแพ็คเกจ Basic_configuration .....	24
3.9 คลาสในแพ็คเกจ Routing .....	25
3.10 คลาสในแพ็คเกจ Device_configuration .....	25
3.11 คลาสในแพ็คเกจ Monitoring .....	25
3.12 แผนภาพ Class ของ โปรแกรม Network Device Configuration .....	26
4.1 หน้าต่างโปรแกรมโดยรวม .....	27
4.2 หน้าต่างขั้นตอนการออกแบบเครือข่าย .....	28
4.3 หน้าต่างขั้นตอนการลากวางอุปกรณ์ .....	29
4.4 การลบอุปกรณ์เครือข่าย .....	29

## สารบัญรูป (ต่อ)

รูป	หน้า
4.5 การเลือกพอร์ตของอุปกรณ์เพื่อใช้ในการเชื่อมต่อ .....	30
4.6 การออกแบบเครือข่ายเสร็จสิ้นตามต้องการ .....	30
4.7 กดปุ่ม Complete เพื่อยืนยันการสร้างเครือข่ายเสร็จสิ้น .....	31
4.8 การสร้างเครือข่ายใหม่.....	31
4.9 การลบแถบพื้นที่การออกแบบเครือข่าย .....	32
4.10 การออกแบบเครือข่ายโดยผู้ใช้โปรแกรมเสร็จสิ้น .....	33
4.11 การเลือกอุปกรณ์เครือข่ายที่ต้องการตั้งค่าเริ่มต้น .....	33
4.12 โปรแกรมกำลังดำเนินการตั้งค่าเริ่มต้น.....	34
4.13 ไดอะล็อกแสดงการตั้งค่าเริ่มต้นสำเร็จ.....	34
4.14 ไดอะล็อกแสดงการเลือกอุปกรณ์ที่ผู้ใช้ต้องการให้เป็นตัวกลางในการส่งคำสั่ง .....	35
4.15 ผู้ใช้โปรแกรมทำการเลือกอุปกรณ์ที่ใช้เป็นตัวกลางในการส่งคำสั่ง.....	35
4.16 ไดอะล็อกแสดงการยืนยันการบันทึกค่าการทำงานเริ่มต้นลงใน startup-config.....	36
4.17 การเลือกตั้งค่าพื้นฐานให้กับอุปกรณ์ชนิดเราท์เตอร์ .....	37
4.18 ไดอะล็อกแสดงการตั้งค่าพื้นฐานระหว่างเราท์เตอร์กับเราท์เตอร์.....	38
4.19 การออกแบบเครือข่ายแบบกึ่ง .....	39
4.20 การแจ้งเตือนให้ผู้ใช้ทำการเปลี่ยนการเชื่อมต่อไปยังอุปกรณ์ตัวอื่นเพื่อส่งคำสั่ง .....	39
4.21 การแจ้งเตือนเพื่อให้ผู้ใช้ทำการเลือกอุปกรณ์ตัวกลางใหม่ในกรณีแบบกึ่ง .....	40
4.22 ไดอะล็อกแสดงโปรโตคอลแบบ Dynamic Route .....	41
4.23 ผู้ใช้ทำการบันทึกข้อมูลที่ตั้งค่าไว้ลงใน startup-config.....	41
4.24 การบันทึกการตั้งค่าลงใน startup-config สำเร็จ.....	42
4.25 การทดสอบการเชื่อมต่อกันของเครือข่ายโดยการ ping .....	42
4.26 ไฟล์ข้อมูลที่พบการส่งข้อมูลผิดพลาดจากเราท์เตอร์.....	43
4.27 หน้าต่างแสดงสถานะของการทำงานของอุปกรณ์เราท์เตอร์ .....	44

## สารบัญรูป (ต่อ)

รูป	หน้า
4.28 หน้าต่างแสดงสถานะเมื่อทำการเราท์ติ้งเสร็จสิ้น .....	44
4.29 ไฟล์ข้อมูลที่ได้จากการรับส่งข้อมูลผ่านพอร์ตอนุกรมในระหว่างโปรแกรมทำงาน .....	45
4.30 หน้าต่างแสดงการบันทึกการออกแบบและการตั้งค่าเครือข่าย .....	45
4.31 ไฟล์ที่ได้จากการบันทึกโดยผู้ใช้โปรแกรม .....	46
4.32 การโหลดไฟล์ที่ได้บันทึกการออกแบบและการตั้งค่าเครือข่าย.....	46
4.33 โปรแกรมสอบถามว่าต้องการใช้งานกับอุปกรณ์ที่มีการตั้งค่าอยู่ก่อนหรือไม่ .....	47
4.34 โปรแกรมทำการสอบถามผู้ใช้งานว่าต้องการแก้ไขการออกแบบหรือไม่.....	47
4.35 โปรแกรมสอบถามผู้ใช้งานว่าต้องการตั้งค่าการทำงานเริ่มต้นด้วยค่าที่โหลดมาหรือไม่.....	48
4.36 ผู้ใช้ทำการโหลดไฟล์เพื่อใช้กับอุปกรณ์เดิมและไม่แก้ไขเครือข่ายเดิม .....	48
4.37 การแจ้งเตือนเมื่อผู้ใช้งานไม่ได้ทำการเลือกอุปกรณ์ในการตั้งค่าเริ่มต้น .....	49
4.38 การแจ้งเตือนเมื่อขณะการตั้งค่าเริ่มต้นผู้ใช้งานไม่ได้เสียบสายคอนโซล.....	49
4.39 การแจ้งเตือนเมื่อขณะการตั้งค่าเริ่มต้นผู้ใช้งานเลือกตั้งค่าที่อุปกรณ์ตัวเดิมซ้ำ .....	50
4.40 การแจ้งเตือนเมื่อขณะการตั้งค่าเริ่มต้นอุปกรณ์มีการตั้งค่าพาสเวิร์ดไว้ก่อนแล้ว .....	50
4.41 การแจ้งเตือนเมื่อขณะการเลือกอุปกรณ์ตัวกลางไม่สัมพันธ์กับอุปกรณ์ที่เชื่อมต่อจริง.....	51
4.42 การแจ้งเตือนขณะทำการตั้งค่าพื้นฐานในเครือข่ายแบบกึ่งเพื่อให้การตั้งค่าพื้นฐานสมบูรณ์ ..	51
4.43 การทดลองรับค่าจากพอร์ตอนุกรม .....	52
4.44 การทดลองส่งค่าผ่านพอร์ตอนุกรม .....	53
4.45 การทดลองส่งคำสั่งผ่านพอร์ตอนุกรม .....	53
4.46 การทดลองนำข้อมูลที่รับส่งผ่านพอร์ตอนุกรมเขียนลงไฟล์ .....	54
5.1 ลักษณะการใช้งานโปรแกรมในการพัฒนาต่อ .....	56

# บทที่ 1

## บทนำ

### 1.1 ความเป็นมาของปัญหา

ในปัจจุบันการกำหนดค่าการทำงานอุปกรณ์ในเครือข่ายจะต้องทำการส่งคำสั่งผ่านทาง Console Port โดยใช้ Command Line Interface ซึ่งการกำหนดค่าการทำงานสำหรับอุปกรณ์แต่ละชนิดนั้นจะต้องทำการศึกษาคำสั่งที่จะต้องใช้กับอุปกรณ์แต่ละชนิด ทำให้ผู้ที่ต้องการใช้งานอุปกรณ์เครือข่ายนอกจากต้องมีความรู้พื้นฐานทางด้านเครือข่ายแล้วจะต้องศึกษาการใช้งานคำสั่งต่างๆ เพื่อกำหนดค่าการทำงาน

### 1.2 หลักการและเหตุผล

ในการใช้งานอุปกรณ์เครือข่ายเพื่อให้ทำงานในการส่งผ่านข้อมูลจากเครือข่ายหนึ่งไปยังอีกเครือข่ายหนึ่งนั้น ต้องมีการกำหนดคำสั่งการทำงานให้กับอุปกรณ์เครือข่ายผ่าน CLI (Command Line Interface) โดยใช้คำสั่งเฉพาะ สำหรับการกำหนดค่าการทำงานให้กับอุปกรณ์เครือข่ายทำงานตามความต้องการของผู้ใช้ และจำเป็นที่จะต้องเชื่อมต่อกับอุปกรณ์เครือข่ายทุกตัวที่ต้องการจะกำหนดค่าการทำงาน

ดังนั้นทางทีมพัฒนาโปรแกรมได้เล็งเห็นถึงปัญหาดังที่ได้กล่าวมานี้ จึงได้มีการพัฒนาโปรแกรมที่ช่วยให้ผู้ใช้ที่มีความรู้ทางด้านเครือข่ายคอมพิวเตอร์เบื้องต้น สามารถที่จะกำหนดค่าการทำงานอุปกรณ์เครือข่ายได้ตามที่ผู้ใช้ต้องการ โดยผ่านตัวโปรแกรมที่เข้าใจและใช้งานได้ง่าย ทำให้ผู้ใช้งานเบื้องต้นรวมไปถึงผู้ใช้งานที่มีความรู้อยู่แล้วสามารถใช้งานโปรแกรมนี้เพื่อกำหนดค่าการทำงานให้กับอุปกรณ์เครือข่ายแต่ละตัวในระบบ โดยผ่านอุปกรณ์เครือข่ายที่เชื่อมต่ออยู่ได้ ซึ่งจะทำให้เกิดความสะดวกมากขึ้น

### 1.3 วัตถุประสงค์

- 1) ศึกษาวิธีการแปลความหมายของข้อมูลที่ได้รับจากอุปกรณ์เครือข่ายผ่านทาง Serial Port
- 2) สร้างโปรแกรมที่สามารถส่งคำสั่งไปยังอุปกรณ์เครือข่าย และสามารถตรวจสอบการทำงานของอุปกรณ์ได้
- 3) สร้างโปรแกรมที่สามารถตรวจสอบการเชื่อมต่อระหว่างอุปกรณ์ในเครือข่าย
- 4) สร้างโปรแกรมแสดงผลของรูปแบบการเชื่อมต่อของอุปกรณ์ในเครือข่าย
- 5) สร้างโปรแกรมที่สามารถกำหนดค่าการทำงานของอุปกรณ์หลายตัวผ่านอุปกรณ์ที่เชื่อมต่อ

อยู่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- 6) สร้างโปรแกรมที่สามารถตรวจสอบสถานะของการทำงานของอุปกรณ์เครือข่ายแต่ละตัวผ่านอุปกรณ์ที่เชื่อมต่ออยู่

#### 1.4 ผลที่จะได้รับจากการทำโครงการ

- 1) ความรู้และความเข้าใจเกี่ยวกับการรับส่งของข้อมูลของอุปกรณ์เครือข่ายผ่าน Serial Port
- 2) การตัดและแปลความหมายของข้อมูลที่ได้จากอุปกรณ์เครือข่าย
- 3) โปรแกรมที่ช่วยในการกำหนดค่าการทำงานของอุปกรณ์เครือข่ายที่สามารถใช้งานได้สะดวกและเข้าใจง่าย
- 4) โปรแกรมที่ช่วยในการกำหนดค่าการทำงานของอุปกรณ์เครือข่ายผ่านอุปกรณ์เครือข่ายที่เชื่อมต่ออยู่เพียงเครื่องเดียว
- 5) โปรแกรมที่ช่วยในการตรวจสอบสถานะของการเชื่อมต่อของอุปกรณ์เครือข่ายแต่ละตัวได้อย่างสะดวก

#### 1.5 ขอบเขตของโครงการ

- 1) ผู้ใช้ออกแบบโครงข่ายของอุปกรณ์เครือข่ายที่ตนมีอยู่ได้ตามต้องการได้
- 2) โปรแกรมสามารถกำหนดค่าการทำงานของอุปกรณ์เครือข่ายให้สามารถเชื่อมต่อกันเบื้องต้นได้
- 3) ผู้ใช้จะต้องสามารถกำหนดค่าการทำงานทั่วไปของอุปกรณ์เครือข่ายผ่านอุปกรณ์ที่เชื่อมต่ออยู่เพียงตัวเดียวได้
- 4) โปรแกรมสามารถแสดงค่าการทำงานที่ได้ทำการกำหนดค่าไว้ที่อุปกรณ์เครือข่ายแต่ละตัวได้
- 5) โปรแกรมสามารถแสดงสถานะของการเชื่อมต่อของอุปกรณ์เครือข่ายแต่ละตัวได้

#### 1.6 ข้อจำกัดของโปรแกรมที่พัฒนา

ใช้ได้เฉพาะกับเราเตอร์ของ Cisco รุ่น 1700 Series โมเดล 1760 IOS Software Release 12.2(11)YV และ เราเตอร์ Cisco 2600 Series IOS Software Releases 12.4 Mainline และ 12.4T

#### 1.7 ส่วนประกอบของโครงการ

รายงานฉบับนี้ประกอบด้วยเนื้อหา 5 บท ซึ่งแต่ละบทกล่าวถึงเรื่องดังต่อไปนี้

บทที่ 1 กล่าวถึงความเป็นมาของปัญหา วัตถุประสงค์ของโครงการ ประโยชน์ที่คาดว่าจะได้รับ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ขอบเขตของโครงการ ขั้นตอนการดำเนินงาน และส่วนประกอบของรายงาน

บทที่ 2 กล่าวถึงทฤษฎีและหลักการที่เกี่ยวข้อง

บทที่ 3 กล่าวถึงการออกแบบและโครงสร้างของโปรแกรม

บทที่ 4 กล่าวถึงผลการทดลองการใช้โปรแกรม

บทที่ 5 กล่าวถึงบทสรุป



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## บทที่ 2

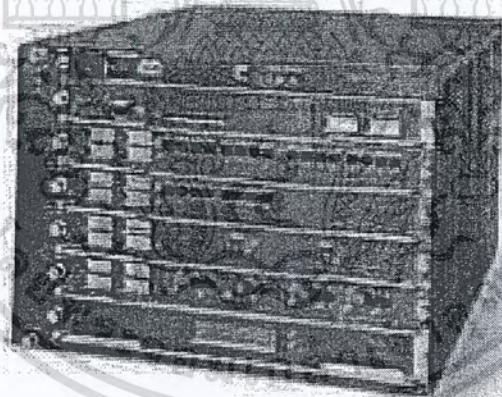
# ทฤษฎีและหลักการที่เกี่ยวข้อง

### 2.1 การทำงานของเราเตอร์

#### 2.1.1 เราเตอร์(Router)

เราเตอร์เป็นอุปกรณ์ที่ซับซ้อนกว่าสวิตช์ โดยทำงานเสมือนเป็นเครื่องหรือ Node หนึ่งใน LAN ซึ่งจะทำหน้าที่รับข้อมูลเข้ามาแล้วส่งต่อไปยังปลายทาง โดยอาจส่งในรูปแบบของแพคเกจที่ต่างออกไป เพื่อไปผ่านสายสัญญาณแบบอื่นๆ เช่น สายโทรศัพท์ที่ต่อผ่านโมเด็มก็ได้ ดังนั้นจึงอาจใช้เราเตอร์ในการเชื่อมต่อ LAN หลายแบบเข้าด้วยกันผ่าน WAN ได้ และจากการที่ เราเตอร์ทำตัวเสมือนเป็น Node หนึ่งใน LAN ยังทำให้สามารถทำงานอื่นๆ ได้อีก เช่น รวบรวมข้อมูลเพื่อหาเส้นทางที่ดีที่สุดในการส่งข้อมูลต่อหรือตรวจสอบข้อมูลที่เข้ามานั้นมาจากไหน ควรจะให้ผ่านหรือไม่ เพื่อช่วยในเรื่องการรักษาความปลอดภัยด้วย ตัวอย่างของเราเตอร์ดูได้จากรูป

2.1



รูป 2.1 เราเตอร์

#### 2.1.2 ข้อแตกต่างระหว่างเราเตอร์กับสวิตช์

ข้อที่แตกต่างกันระหว่างสวิตช์กับเราเตอร์คือ สวิตช์จะทำงานในระดับชั้น Data Link Layer ก็จะใช้ข้อมูล MAC Address ในการทำงานส่งข้อมูลไปยังที่ใดๆ ซึ่งหมายเลข MAC Address นี้มีการกำหนดมาจากฮาร์ดแวร์หรือที่ส่วนของ Network Interface Card (NIC) และถูกกำหนดมาเฉพาะตัวจากโรงงานไม่ให้ซ้ำกัน ถ้ามีการเปลี่ยน Network Interface Card (NIC) นี้ไป ก็จะทำให้ MAC Address เปลี่ยนไปด้วย ส่วน Network Layer Address ในการส่งผ่านข้อมูล เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โปรโตคอลของเครือข่ายชนิดต่างๆ ไม่ว่าจะเป็น IPX, TCP/IP หรือ AppleTalk ซึ่งจะเป็นโปรโตคอลที่ทำงานใน Network Layer การกำหนด Network address ทำได้โดยผู้ดูแลระบบเครือข่ายนั้น ทำให้สามารถแก้ไขเปลี่ยนแปลงได้ง่าย และสามารถใช้อุปกรณ์เราท์เตอร์เชื่อมโยงเครือข่ายที่แยกจากกันให้สามารถส่งผ่านข้อมูลร่วมกันได้และทำให้เครือข่ายขยายออกไปได้เรื่อยๆ

### 2.1.3 หน้าที่หลักของเราท์เตอร์

เราท์เตอร์เป็นอุปกรณ์ที่ถูกนำมาใช้เพื่อการเชื่อมต่อระหว่างเครือข่าย เช่น การเชื่อมต่อระหว่างเครือข่าย ที่มีไอพีแอดเดรส 192.168.20.0 กับเครือข่ายที่มีไอพีแอดเดรส 192.168.30.0 เป็นต้น รวมทั้งการเชื่อมต่อเครือข่ายย่อย (เครือข่ายเดียวกันแต่ถูกแบ่งออกเป็นส่วนย่อยๆ เช่น แบ่งเครือข่ายที่มีหมายเลขไอพีแอดเดรส 192.168.30.0 ออกเป็นเครือข่ายย่อยๆ (Subnet) จำนวน 6 เครือข่าย) จากนั้นนำมาเชื่อมต่อกัน เพื่อการสื่อสารกันด้วยเราท์เตอร์

เราท์เตอร์เป็นอุปกรณ์เชื่อมต่อเครือข่ายทำงานในระดับชั้น Network ตามมาตรฐานของ OSI Model หน้าที่หลักของเราท์เตอร์ได้แก่ การอ้างอิงไอพีแอดเดรสระหว่างเครื่องลูกข่ายที่อยู่กันคนละเครือข่าย รวมทั้งการเลือกและจัดเส้นทางที่ดีที่สุด เพื่อนำข้อมูลข่าวสาร ในรูปแบบของแพ็กเกจจากเครื่องลูกข่ายต้นทางบนเครือข่ายที่ตนดูแลอยู่ไปยังเครื่องลูกข่ายที่อยู่กันคนละเครือข่าย เราท์เตอร์ ที่ใช้เพื่อการเชื่อมต่อระหว่างเครือข่ายผ่านทาง WAN หรือโครงข่ายสาธารณะ อย่างเช่น ผ่านทางเฟรมรีเลย์ หรือ ISDN เราเรียกว่า WAN Router ส่วนเราท์เตอร์ที่ใช้เชื่อมต่อระหว่างเครือข่ายชนิดติดตั้งบนแลนและเชื่อมต่อระหว่างเครือข่ายด้วยสายสัญญาณของระบบแลน เรียกว่า Local Router หรือบางครั้งจะถูกเรียกว่า Internal Router ซึ่งเราท์เตอร์ประเภทนี้อาจเป็นเราท์เตอร์ในรูปแบบผลิตภัณฑ์ของเราท์เตอร์เต็มตัว หรือแบบที่มีการติดตั้งการ์ดแลนหลายชุดบนเครื่องคอมพิวเตอร์ที่ติดตั้งเซิร์ฟเวอร์เป็นต้น

หน้าที่หลักของเราท์เตอร์คือการหาเส้นทางในการส่งผ่านข้อมูลที่ดีที่สุด และเป็นตัวกลางในการส่งต่อข้อมูลไปยังเครือข่ายอื่น ทั้งนี้เราท์เตอร์สามารถเชื่อมโยงเครือข่ายที่ใช้สื่อสัญญาณหลายแบบแตกต่างกันได้ไม่ว่าจะเป็น Ethernet, Token Ring หรือ FDDI ซึ่งในแต่ละระบบจะมีแพ็คเกจเป็นรูปแบบแตกต่างกัน โดยโปรโตคอลที่ทำงานในระดับบนหรือ Layer 3 ขึ้นไปเช่น IP, IPX หรือ AppleTalk เมื่อมีการส่งข้อมูลก็จะบรรจุข้อมูลนั้นเป็นแพ็คเกจในรูปแบบของ Layer 2 คือ Data Link Layer เมื่อเราท์เตอร์ได้รับข้อมูลมาก็จะตรวจสอบในแพ็คเกจ เพื่อจะทราบว่าใช้โปรโตคอลแบบใด จากนั้นก็จะตรวจสอบเส้นทางส่งข้อมูลจากตาราง Routing Table ว่าจะต้องส่งข้อมูลนี้ไปยังเครือข่ายใดจึงจะต่อไปถึงปลายทางได้ แล้วจึงบรรจุข้อมูลลงเป็นแพ็คเกจของ Data Link Layer ที่ถูกต้องอีกครั้ง เพื่อส่งต่อไปยังเครือข่ายปลายทาง

## 2.2 คำสั่งที่ใช้ในการทำงานของเราเตอร์

เมื่อทำการติดต่อกับเราเตอร์ผ่านพอร์ตอนุกรมแล้ว ผู้ใช้จะทำการส่งคำสั่งไปทำการกำหนดค่าการทำงานให้กับเราเตอร์ สำหรับคำสั่งในการกำหนดค่าการทำงานเข้าไปมีดังนี้

### 2.2.1 คำสั่งกำหนดค่าการทำงานของเราเตอร์

#### 2.2.1.1 คำสั่งที่ใช้ในการเปลี่ยนชื่อเราเตอร์

สามารถทำการเปลี่ยนชื่อเราเตอร์แต่ละตัวได้ด้วยคำสั่งดังตัวอย่าง 2.1

#### ตัวอย่าง 2.1 คำสั่งที่ใช้ในการเปลี่ยนชื่อเราเตอร์

```
Router> enable
Router# configure terminal
Router(config)# hostname Cisco
Cisco(config)#
```

จะเห็นว่าชื่อของเราเตอร์จะเปลี่ยนจาก Router เป็น Cisco

#### 2.2.1.2 คำสั่งที่ใช้ในการกำหนดค่ารหัสผ่านของเราเตอร์

สามารถทำการกำหนดค่า Console Password บนเราเตอร์ได้ด้วยคำสั่งดังตัวอย่าง

## 2.2

#### ตัวอย่าง 2.2 คำสั่งที่ใช้ในการกำหนดค่ารหัสผ่านของเราเตอร์

```
Router(config)# line console 0
Router(config-line)# password ccna
Router(config-line)# login
Router(config-line)# exit
Router(config)#
```

จากคำสั่งด้านบนจะทำให้การเข้าใช้งานผ่าน Console จะต้องใส่รหัสผ่านคำว่า ccna ก่อนถึงเข้าใช้งานได้ดังตัวอย่าง 2.3

#### ตัวอย่าง 2.3 การเข้าใช้งาน Console เมื่อมีการตั้งรหัสผ่าน

```
Router>
Router>enable
Password:
Router#
```

### 2.2.1.3 คำสั่งที่ใช้ในการกำหนดคำรหัสผ่านในส่วนของ Virtual Terminal Lines ของเราเตอร์

สามารถทำการกำหนดคำรหัสผ่านในส่วนของ Virtual Terminal Lines บนเราเตอร์ได้ด้วยคำสั่งดังตัวอย่าง 2.4

#### ตัวอย่าง 2.4 คำสั่งที่ใช้ในการกำหนดคำรหัสผ่านในส่วนของ Virtual Terminal Lines ของเราเตอร์

```
Router(config)# line vty 0
Router(config-line)# password ccna
Router(config-line)# login
Router(config-line)# exit
Router(config)#
```

จากคำสั่งด้านบนจะทำให้การเข้าใช้งานผ่านการ Telnet จะต้องใส่รหัสผ่านคำว่า ccna ก่อนถึงเข้าใช้งานได้ดังตัวอย่าง 2.5

#### ตัวอย่าง 2.5 การเข้าใช้งาน Telnet เข้าไปที่ Router\_first เมื่อมีการตั้งรหัสผ่าน

```
Router>
Router>telnet 192.168.1.1
Password:
Router_first>
```

### 2.2.1.4 คำสั่งที่ใช้กำหนดคำรหัสผ่านในการเข้าใช้ Privileged Exec Mode ของเราเตอร์

สามารถทำการกำหนดคำรหัสผ่านสำหรับการเข้าใช้งาน Privileged Exec Mode ได้ด้วยคำสั่งดังตัวอย่าง 2.6

#### ตัวอย่าง 2.6 คำสั่งที่ใช้กำหนดคำรหัสผ่านในการเข้าใช้ Privileged Exec Mode ของเราเตอร์

```
Router(config)# enable password ccna
Router(config)# exit
```

จากคำสั่งด้านบนจะทำให้การเข้าใช้งาน Privileged Exec Mode จะต้องใส่รหัสผ่านคำว่า ccna ก่อนถึงเข้าใช้งานได้แต่ถ้าต้องการให้รหัสผ่านนั้นมีการ Encryption ด้วยจะต้องใช้คำสั่งดังตัวอย่าง 2.7 แทน

### ตัวอย่าง 2.7 การกำหนดให้รหัสผ่านมีการ Encryption

```
Router(config)# enable secret ccna
Router(config)# exit
```

#### 2.2.1.5 คำสั่งที่ใช้ในการกำหนดค่าให้กับอินเทอร์เฟซ

สามารถทำการกำหนดค่า IP Address, Subnet Mark, Clock rate, และการเปิดใช้งานอินเทอร์เฟซให้แต่ละอินเทอร์เฟซได้ด้วยคำสั่งดังตัวอย่าง 2.8

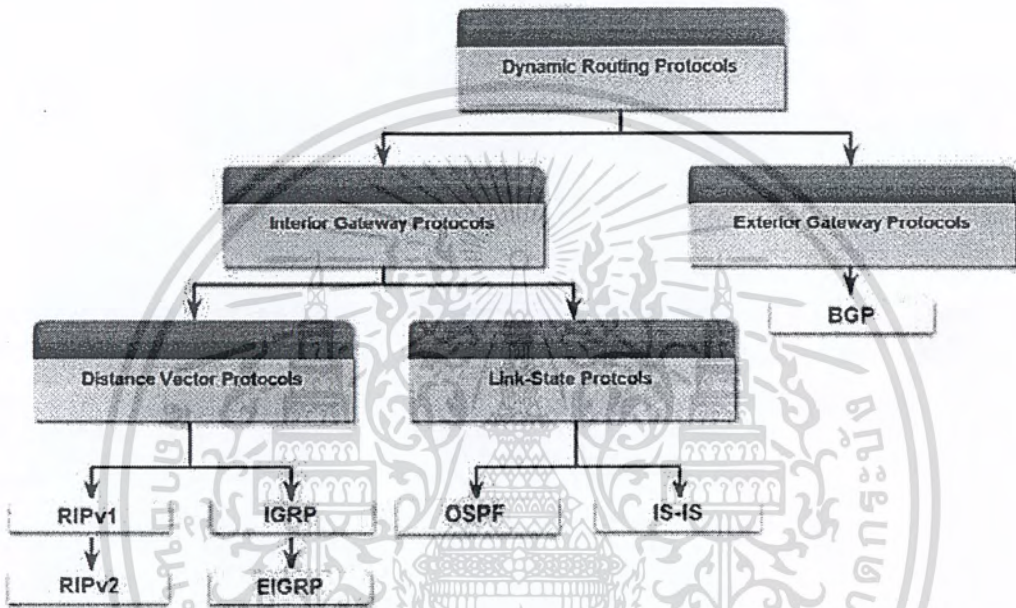
#### ตัวอย่าง 2.8 คำสั่งที่ใช้ในการกำหนดค่าให้กับอินเทอร์เฟซ

```
Router(config)# interface serial 0/0
Router(config-if)# ip address 192.168.1.1 255.255.255.0
Router(config-if)# clock rate 56000
//only DCE Interface Type
Router(config-if)# no shutdown
Router(config)# exit
```

จากคำสั่งด้านบนเป็นการกำหนดค่าให้กับอินเทอร์เฟซ serial 0/0 ซึ่งกำหนดให้ IP Address เท่ากับ 192.168.1.1, Subnet Mark เท่ากับ 255.255.255.0 และ Clock rate เท่ากับ 56000 (สำหรับอินเทอร์เฟซที่มีชนิดเป็น DCE เท่านั้น) จากนั้นก็ทำการเปิดใช้งานอินเทอร์เฟซนี้ด้วยคำสั่ง no shutdown

### 2.2.1.6 คำสั่งที่ใช้ในการกำหนดค่าการส่งต่อแพคเกจของเราเตอร์

เนื่องจากเราเตอร์เป็นอุปกรณ์เครือข่ายที่ทำงานที่ลำดับชั้นที่สาม ซึ่งมีความสามารถในการส่งผ่านแพคเกจข้ามเน็ตเวิร์กที่ต่างกันได้ โดยจะมีรูปแบบในการส่งผ่านแพคเกจอยู่สองรูปแบบคือ Static Routing และ Dynamic Routing โดย Dynamic Routing จะอาศัยโปรโตคอลต่าง ๆ ซึ่งแยกตามประเภทได้ดังรูป 2.3



รูป 2.2 การแบ่งแยกประเภทของ Dynamic Routing Protocols

การกำหนดค่าการส่งผ่านแพคเกจในระบบเครือข่ายสามารถทำได้โดยใช้คำสั่งต่างๆ แยกตามประเภทดังตัวอย่างต่อไปนี้

#### ตัวอย่าง 2.9 คำสั่งที่ใช้ในการกำหนด Static Route

```
//Static Routing
Router(config)# ip route 192.168.2.0 255.255.255.0
serial 0/0
OR
Router(config)# ip route 192.168.2.0 255.255.255.0
192.168.1.1
```

## ตัวอย่าง 2.10 คำสั่งที่ใช้ในการกำหนด RIPv1

```
//RIPv1
Router(config)# router rip
Router(config-router)# network 192.168.1.0
Router(config-router)# network 192.168.2.0

Router(config-router)# passive-interface FastEthernet
0/0
Router(config-router)# end
```

## ตัวอย่าง 2.11 คำสั่งที่ใช้ในการกำหนด RIPv2

```
//RIPv2
Router(config)# router rip
Router(config-router)# version 2
Router(config-router)# network 192.168.1.0
Router(config-router)# network 192.168.2.0

Router(config-router)# exit
Router(config)# exit
```

## ตัวอย่าง 2.12 คำสั่งที่ใช้ในการกำหนด EIGRP

```
//EIGRP
Router(config)# router eigrp 123
Router(config-router)# network 192.168.1.0
Router(config-router)# network 192.168.2.0

Router(config-router)# no auto-summary
Router(config-router)# exit
Router(config)# exit
```

ตัวอย่าง 2.13 คำสั่งที่ใช้ในการกำหนด OSPF

```
//OSPF
Router(config)# ip routing
Router(config)# router ospf 456
Router(config-router)# network 192.168.1.1 0.0.0.255
area 0
Router(config-router)# network 192.168.1.2 0.0.0.255
area 0

Router(config-router)# log-adjacency-changes
Router(config-router)# exit
Router(config)# exit
```

คำสั่งต่างๆ เหล่านี้เป็นคำสั่งที่ใช้ในการกำหนดค่าการทำงานให้เราเตอร์สามารถส่งผ่านแพคเกจข้ามเน็ตเวิร์กที่ต่างกันได้ โดยแต่ละโปรโตคอลจะมีรูปแบบการกำหนดค่าที่ต่างกันไป



## 2.3 หลักการและโปรโตคอลที่เกี่ยวข้อง

### 2.3.1 โหมดในการใช้งาน CLI

โดยปกติเมื่อเปิดโปรแกรมจำลองเทอร์มินอลอย่างเช่น HyperTerminal หรือ SecureCRT และต่อพอร์ตคอนโซลเข้าไปยังเราท์เตอร์ หลังจากกด enter แล้ว โดยปกติโหมดแรกที่จะพบก็คือ User EXEC Mode (ในการเข้าถึง CLI จะมี 3 วิธี ได้แก่ การต่อพอร์ตคอนโซล, การเทลเน็ตผ่านเนตเวิร์ก และการต่อพอร์ต AUX) ปกติแล้วโหมดของการใช้งาน CLI โหมดแรกที่จะพบก็คือโหมด User EXEC เสมอ โดยที่โหมด User EXEC เป็นโหมดที่เปิดโอกาสให้มีการใช้คำสั่งพื้นฐานที่จำเป็นต่อการดูแลสถานะของการทำงานบางอย่างของเราท์เตอร์ โดยจำกัดคำสั่งที่สามารถใช้ได้ไว้ข้อสังเกตก็คือ ภายในโหมดนี้จะมีเครื่องหมายพรอมต์ (prompt) รอรับคำสั่งที่ได้รับจะเป็นเครื่องหมาย ">"

หากต้องการเรียกใช้คำสั่งอื่นๆทั้งหมด เราจำเป็นต้องเปลี่ยนจากโหมด User EXEC เข้าสู่โหมดหนึ่งที่เราเรียกว่า Privileged EXEC Mode ด้วยการใช้คำสั่ง enable (โหมดนี้นิยมเรียกอีกอย่างว่า ENABLE Mode) โหมดนี้เปิดโอกาสให้เราใช้งานได้ทุกคำสั่งที่เกี่ยวข้องกับการสำรวจสถานะของการทำงานและความเป็นไปต่างๆของเราท์เตอร์ ข้อสังเกตก็คือ ภายในโหมดนี้จะมีเครื่องหมายพรอมต์(prompt) รอรับคำสั่งที่ได้รับจะเป็นเครื่องหมาย "#" ภายใต้โหมด Privileged EXEC นี้ เราทำได้แค่เพียงการดูสถานะของการทำงานต่างๆของเราท์เตอร์ทั้งหมดเท่านั้น แต่ยังไม่สามารถทำการกำหนดตั้งค่าต่างๆให้กับเราท์เตอร์ได้ จนกว่าจะได้เปลี่ยนจากโหมด Privileged EXEC (หรือที่เรียกว่า Enable Mode) เข้าสู่โหมดที่เราเรียกว่า Global Configuration Mode (โกลบอลคอนฟิกูเรชันโหมด) ด้วยคำสั่ง configure terminal ข้อสังเกตคือ ภายในโหมดนี้ เครื่องหมายพรอมต์(prompt) รอรับคำสั่งที่ได้รับจะเป็น (config)#

หลังจากเข้าสู่โหมด Global Configuration Mode แล้ว จะสามารถกำหนดตั้งค่าต่างๆได้ทั้งหมด ภายใต้โหมด Global Configuration นี้ ผู้ใช้สามารถที่จะตั้งค่าคำสั่งต่างๆลงไปเพื่อกำหนดตั้งค่าต่างๆ และเปิดพีเจอรต์ต่างๆของเราท์เตอร์ขึ้นมา คำสั่งต่างๆที่คีย์ได้ภายใต้โหมดนี้ โดยตรงจะถูกเรียกว่าเป็นคำสั่งที่ทำงานภายใต้ Global Configuration Mode คำสั่งที่ตั้งค่าได้ภายใต้ Global Configuration Mode มักเป็นคำสั่งที่มีผลต่อการทำงานของเราท์เตอร์ทั้งตัว ตัวอย่างเช่น คำสั่ง hostname, คำสั่ง ip domain-lookup เป็นต้น ภายใต้ Global Configuration Mode นี้ยังมีการแบ่งย่อยออกเป็นโหมด Sub-Configuration สำหรับการกำหนดตั้งค่าให้กับองค์ประกอบหรือพีเจอรต์ต่างๆของเราท์เตอร์อีก โดยคำสั่งที่คีย์ลงไปเพื่อกำหนดตั้งค่าต่างๆของเราท์เตอร์จะถูกนำไปสร้างเป็นสิ่งที่เรียกว่า "running-configuration" ซึ่ง "แอกทีฟ (active)" อยู่ในหน่วยความจำ (RAM) ของเราท์เตอร์และ IOS ของเราท์เตอร์ ซึ่งเป็นระบบปฏิบัติการที่มีหน้าที่ควบคุมเราท์เตอร์ก็จะทำงานต่างๆ ตาม running-configuration ที่แอกทีฟอยู่ในหน่วยความจำ RAM หากต้องการดูว่าขณะนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

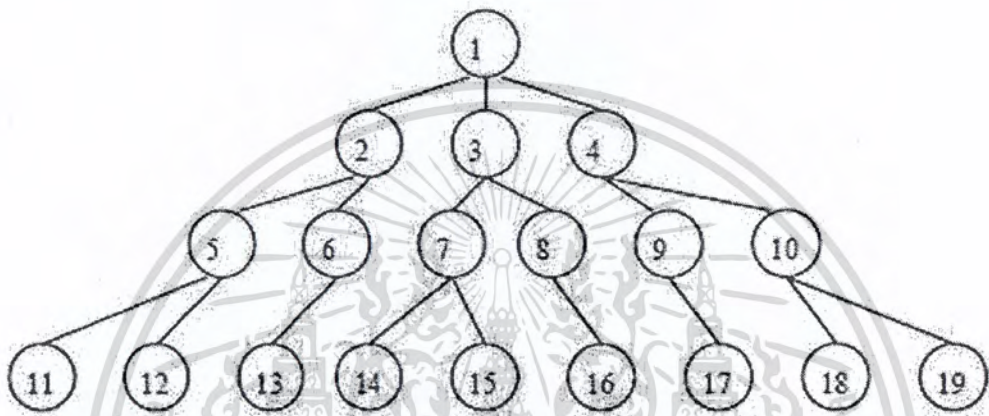
running-configuration ที่แอกทีฟอยู่มีอะไรบ้าง คำสั่งที่ใช้คือ show running-configuration หรือย่อๆ ว่า show run

### 2.3.2 การทำให้สามารถเทลเน็ต (Telnet) มายังเราเตอร์ได้

โดยปกติเราเตอร์จะยังไม่เปิดโอกาสให้เราได้เทลเน็ตผ่านเน็ตเวิร์กเข้ามายังตัวมันได้ จนกว่าเราจะได้ทำการต่อพอร์ตคอนโซลเข้ามาเพื่อสั่งอินเทอร์เฟซที่เป็น LAN และตั้งค่าหมายเลข IP Address ให้กับมัน และทำการตั้งค่านามบน LINE VTY ด้วยคำสั่ง password และกำหนดให้มีการล็อกอินด้วยคำสั่ง login หรือไม่มีอีกทางเลือกหนึ่งคือ การใส่คำสั่ง no login เข้าไปภายใต้ Line VTY คำสั่ง no login จะเป็นการสั่งให้เราเตอร์ไม่ต้องตรวจสอบการล็อกอินของผู้ใช้ที่เข้ามาทาง Line VTY ผู้ใช้ที่เทลเน็ตเข้ามาทาง Line VTY สามารถเข้าถึง CLI ในโหมด User EXEC ได้ทันทีโดยไม่ต้องระบุรหัสผ่านใดๆ (แต่แบบนี้ไม่ปลอดภัย) การกำหนดรหัสผ่านก่อนเข้าสู่ Privileged Mode (หรือที่เรียกว่า Enable Mode) ด้วยคำสั่ง enable secret หรือ enable password ทั้งนี้ขึ้นกับว่าต้องการให้ผู้ที่เข้าถึงผ่านทางเทลเน็ตสามารถเข้าสู่โหมด Privileged EXEC ได้หรือไม่ หากต้องการ เราจำเป็นต้องตั้งค่านามด้วยคำสั่ง enable secret หรือ enable password (ควรใช้ enable secret)

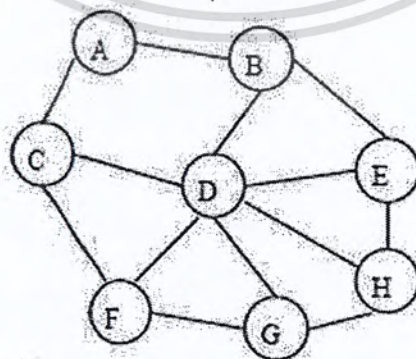
## 2.4 การค้นหาแบบกว้างก่อน (Breadth First Search)

การค้นหาแบบกว้างก่อนเป็นการค้นหาที่กำหนดทิศทางการค้นหาแบบทีละระดับของโครงสร้างต้นไม้โดยเริ่มจากโหนดราก (ระดับที่ 0) แล้วลงมาระดับที่ 1 จากซ้ายไปขวา เมื่อเสร็จระดับที่ 1 ไประดับที่ 2 จากซ้ายไปขวา เช่นกัน ทำเช่นนี้ไปเรื่อยๆ จนพบโหนดที่ต้องการตามรูป 2.6 โดยลำดับการเดินทางของโหนดเป็นไปตามหมายเลขที่กำกับไว้บนโหนด



รูป 2.3 ลำดับการเดินทางบนโหนดของการค้นหาแบบกว้างก่อนบนโครงสร้างต้นไม้

สำหรับการค้นหาแบบกว้างก่อน (Breadth First Search) บนโครงสร้างต้นไม้ จะอาศัยโครงสร้างข้อมูลแบบคิว (Queue) มาช่วย และด้วยวิธีการเช่นเดียวกับการค้นหาแบบลึกก่อนคือ ให้เริ่มต้นสำรวจที่โหนดเริ่มต้น แล้วนำโหนดข้างเคียงเก็บไว้ในคิว เมื่อสำรวจโหนดเริ่มต้นเสร็จให้นำข้อมูลในคิวออกมาสำรวจ แล้วนำโหนดข้างเคียงที่ยังไม่ได้สำรวจและไม่ได้อยู่ในคิวใส่คิวไว้ ทำเช่นนี้ไปเรื่อยๆ จนพบโหนดที่ต้องการ หรือเมื่อสำรวจครบทุกโหนด



รูป 2.4 โครงสร้างข้อมูลแบบกราฟ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การสำรวจเริ่มต้นที่ A นำโหนดข้างเคียง B C ไว้ในคิว เมื่อสำรวจ A เสร็จ นำข้อมูลในคิว คือ B ออกมาสำรวจ แล้วนำข้อมูลข้างเคียงคือ D E ใส่คิว ตอนนี้คิวจะมี B D E อยู่แล้วนำ B ออกมาสำรวจทำเช่นนี้เรื่อยๆ จะได้ลำดับการสำรวจข้อมูลคือ (A B C D E F G H) ตามรูป 2.8

โหนดที่สำรวจ	คิว
A	B C
B	C D E
C	D E F
D	E F G H
E	F G H
F	G H
G	H
H	

รูป 2.5 ลำดับการค้นหาแบบกว้างก่อน

ในการค้นหาข้อมูลแบบนี้บนโครงสร้างของกราฟ มีข้อสังเกตคือ โหนดที่เริ่มต้นการสำรวจ จะต้องมีการกำหนดมาให้ก่อนว่าโหนดใดที่จะเป็นโหนดเริ่มต้น และข้อสังเกตอีกประการหนึ่งคือ วิธีการค้นหาแบบกว้างก่อนที่ใช้สำหรับ โครงสร้างข้อมูลแบบกราฟ สามารถใช้กับ โครงสร้างข้อมูลแบบต้นไม้ได้ด้วย

## บทที่ 3

# การออกแบบและโครงสร้างของโปรแกรม

### 3.1 ภาษาและเครื่องมือที่ใช้ในการพัฒนา

#### 3.1.1 ภาษาที่ใช้ในการพัฒนา

- 1) ภาษา C#
- 2) ภาษา ActionScript 3.0

#### 3.1.2 เครื่องมือที่ใช้ในการพัฒนา

- 1) โปรแกรม Microsoft Visual Studio 2008
- 2) โปรแกรม Adobe Flash CS4
- 3) ระบบปฏิบัติการ Windows
- 4) .NET Framework version 3.5

### 3.2 อินพุตและเอาต์พุตที่เข้าและออกจากระบบ

#### 3.2.1 ข้อมูลเข้าของระบบ (System Input)

ส่วนของการออกแบบ โครงสร้างของระบบเครือข่าย

- 1) การเลือกชนิดของอุปกรณ์เครือข่ายในการวางระบบ
- 2) การวางรูปแบบ โครงข่ายของระบบเครือข่าย

ส่วนของคำสั่งที่ใช้ในการกำหนดค่าการทำงาน

คำสั่งในการกำหนดค่าการทำงานของเราเตอร์

- 1) คำสั่งที่ใช้ในการเปลี่ยนชื่อเราเตอร์
- 2) คำสั่งที่ใช้ในการกำหนดค่ารหัสผ่านของเราเตอร์
- 3) คำสั่งที่ใช้ในการกำหนดค่ารหัสผ่านในส่วนของ Virtual Terminal Lines ของเราเตอร์
- 4) คำสั่งที่ใช้กำหนดค่ารหัสผ่านในการเข้าใช้ Privileged Exec Mode ของเราเตอร์
- 5) คำสั่งที่ใช้ในการกำหนดค่าให้กับอินเตอร์เฟซ
- 6) คำสั่งที่ใช้ในการกำหนดค่าการส่งต่อแพคเกจของเราเตอร์

3.2.2 ข้อมูลออกที่ได้จากระบบ (System Output)

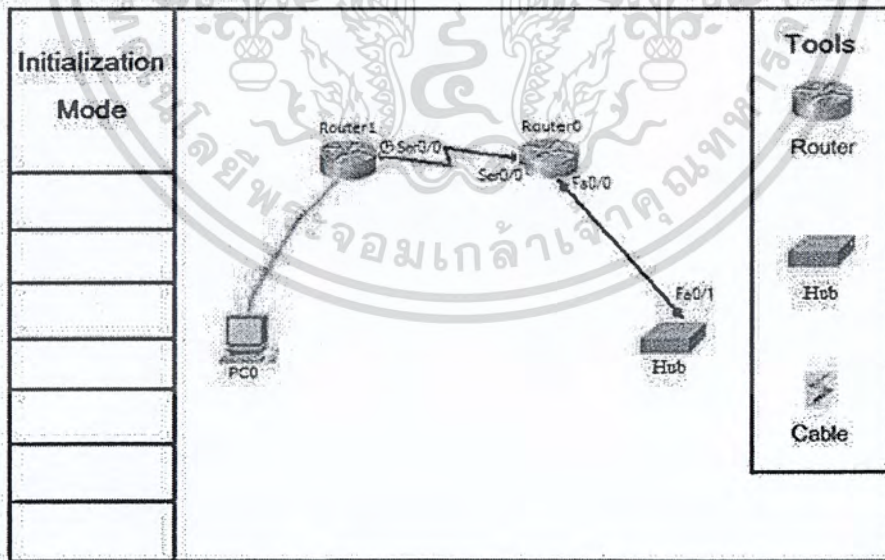
- 1) Dialog Box แสดงการทำการตั้งค่าเริ่มต้น
- 2) เราทเตอร์แต่ละตัวมีค่าที่เกิดจากการตั้งค่าเริ่มต้น
- 3) Dialog box แสดงการตรวจสอบการทำงานของผู้ใช้งานระบบเมื่อมีสิ่งผิดพลาดเกิดขึ้น
- 4) อุปกรณ์เครือข่ายทำงานตามสิ่งที่ผู้ใช้งานระบบกำหนดไว้ในโปรแกรม

3.3 หลักการทำงานของโปรแกรม

เมื่อผู้ใช้งานระบบมีอุปกรณ์เครือข่ายที่ต้องการจะทำการตั้งค่าการทำงานซึ่งเป็นรุ่นที่ตรงตามความสามารถของโปรแกรม สามารถทำการตั้งค่าหรือแสดงค่าการทำงานโดยผ่านอุปกรณ์เครือข่ายเพียงตัวเดียวได้โดยอาศัยโปรแกรมนี้ ซึ่งมีหลักการและขั้นตอนการทำงานดังนี้

3.3.1 การออกแบบโครงสร้างของระบบเครือข่าย

เมื่อทำการเปิดใช้งาน โปรแกรมจะเข้าสู่โหมดการตั้งค่าเริ่มต้นของอุปกรณ์เครือข่าย ซึ่งโปรแกรมจะมีพื้นที่สำหรับให้ผู้ใช้งานทำการออกแบบการเชื่อมต่ออุปกรณ์เครือข่ายแต่ละตัวตามที่ต้องการ ผู้ใช้งานโปรแกรมสามารถออกแบบระบบเครือข่ายได้โดยการเลือกวางอุปกรณ์ที่มีให้จากแถบเครื่องมือ



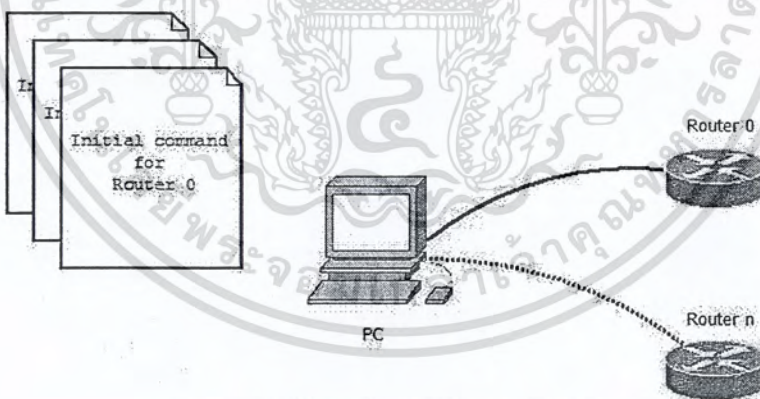
รูป 3.1 การออกแบบระบบเครือข่าย

หลังจากผู้ใช้งานโปรแกรมทำการออกแบบเครือข่ายเรียบร้อยแล้ว ผู้ใช้งานโปรแกรมจะต้องทำการเชื่อมต่ออุปกรณ์จริงให้มีลักษณะเหมือนกับที่ผู้ใช้งานออกแบบไว้ในระบบ ซึ่งตัวเอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาของเอกสารนี้อ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โปรแกรมจะมีส่วนที่ใช้ในการกำหนดค่าเริ่มต้นให้กับอุปกรณ์ที่ผู้ใช้ได้ออกแบบไว้ โดยที่ผู้ใช้งาน จะทำการเชื่อมต่อพอร์ตอนุกรมเข้ากับสายคอนโซล หลังจากนั้นจึงนำไปเชื่อมต่อกับอุปกรณ์ และ ผู้ใช้งานทำการเลือกตัวอุปกรณ์ที่จะทำการตั้งค่าเริ่มต้น ซึ่งขั้นตอนนี้ผู้ใช้งานจะต้องเชื่อมต่อตัว โปรแกรมเข้ากับอุปกรณ์จริงให้ตรงกับตัวอุปกรณ์ที่ได้ทำการเลือกไว้ในตัวโปรแกรม หลังจากทำ การตั้งค่าเริ่มต้นให้กับอุปกรณ์เรียบร้อยแล้วโปรแกรมจะทำการตรวจสอบการเชื่อมต่อของอุปกรณ์ ต่างๆ ซึ่งถ้าเกิดความผิดพลาดในการตั้งค่าเริ่มต้นกับอุปกรณ์จริง

### 3.3.2 การตั้งค่าเริ่มต้นให้กับอุปกรณ์

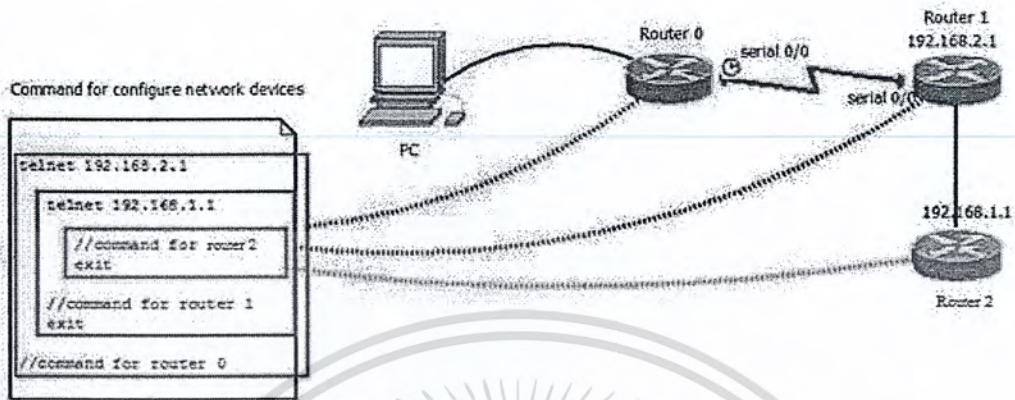
เมื่อผู้ใช้งาน โปรแกรมได้ทำการเชื่อมต่อตัวโปรแกรมและอุปกรณ์จริงเข้าด้วยกัน หลังจากทำการออกแบบเครือข่ายเรียบร้อยแล้ว โปรแกรมจะทำการตั้งค่าเริ่มต้นให้กับอุปกรณ์ โดย จะมีการสร้างชุดคำสั่งขึ้นมาเพื่อทำการตั้งค่าเริ่มต้นให้กับตัวของอุปกรณ์ต่างๆ เช่น ค่า IP Address ของอินเทอร์เฟซที่มีการใช้งานตามแบบเครือข่ายที่ได้ออกแบบไว้แล้ว ชื่อของอุปกรณ์ และ รหัสผ่าน เป็นต้น เมื่อโปรแกรมทำการตั้งค่าเริ่มต้นได้สำเร็จโปรแกรมจะมีการแจ้งแก่ผู้ใช้งาน หรือ เมื่อทำการติดตั้งค่าเริ่มต้นให้กับอุปกรณ์ไม่สำเร็จโปรแกรมจะมีการแจ้งเตือนสิ่งที่จะต้องตรวจสอบ เช่น การตรวจสอบสายและพอร์ตที่ใช้ในการออกแบบ และ/หรือ แจ้งให้ผู้ใช้งานเริ่มต้นการตั้งค่า เริ่มต้นใหม่



รูป 3.2 การตั้งค่าเริ่มต้นให้กับเราเตอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 3.3.3 การตั้งค่าอุปกรณ์



รูป 3.3 การตั้งค่าให้ระบบเครือข่าย

เมื่อผู้ใช้โปรแกรมทำการตั้งค่าเริ่มต้นให้กับอุปกรณ์ ผู้ใช้งานยังสามารถที่จะทำการตั้งค่าอื่นๆ หรือเปลี่ยนแปลงค่าที่ได้ตั้งไว้โดยโปรแกรมได้ ซึ่งค่าต่างๆ ที่ผู้ใช้งานทำการตั้งค่าเพิ่มเติม หรือ เปลี่ยนแปลงค่าต่างๆ ที่ได้ตั้งไว้แล้วนั้น ทางตัวโปรแกรมจะเป็นผู้นำไปตั้งค่าที่อุปกรณ์ต่างๆ ให้ โดยใช้อัลกอริทึมของการค้นหาแบบกว้างก่อน (Breadth first search) ในการสร้างการจำลองทางเดินของเครือข่ายภายในโปรแกรม เป็นอัลกอริทึมที่มีความเหมาะสมในการใช้งาน และประหยัดเวลาในหาเส้นทางในการติดต่อระหว่างอุปกรณ์แล้วจึงทำการกำหนดค่าอุปกรณ์ตัวนั้นๆ เหตุผลที่ต้องทำการกำหนดค่าในลักษณะนี้เป็นเพราะว่า หากทำการกำหนดค่าของอุปกรณ์ที่บริเวณใดก็ได้ของระบบ จะพบปัญหาคือระบบไม่สามารถที่จะติดต่อกันได้เป็นปกติ ซึ่งถ้าเกิดปัญหานี้ขึ้นระบบจะไม่สามารถตั้งค่าที่ได้เพิ่มเติมลงในตัวอุปกรณ์หรือทำการเปลี่ยนแปลงค่าต่างๆ ให้กับตัวอุปกรณ์ที่โปรแกรมไม่สามารถติดต่อไปถึงได้ จึงทำให้การตั้งค่านั้นล้มเหลวและระบบใช้งานไม่ได้ อย่างที่ผู้ใช้งานโปรแกรมต้องการดังรูป 3.4

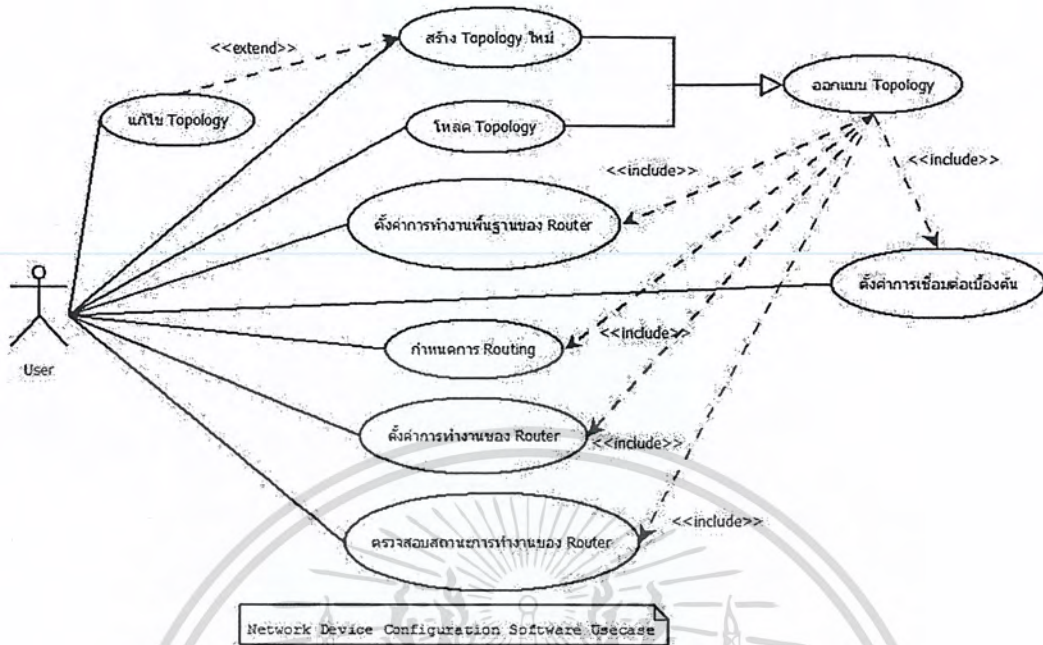
### 3.4 การวิเคราะห์ความต้องการของระบบ

จากการวิเคราะห์ความต้องการของระบบ โดยยึดถือตามขอบเขตของระบบที่จะใช้งาน โดยผู้ใช้งานโปรแกรม สามารถสรุปความต้องการของระบบในรูปแบบภาพ UML โดยแบ่งเป็นแผนภาพ Usecase และแผนภาพ Class ดังต่อไปนี้

#### 3.4.1 แผนภาพ Usecase

- แผนภาพ Usecase ใช้แสดงถึงความสัมพันธ์ระหว่างผู้ใช้และบริการที่ระบบมีให้ใช้งาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูป 3.4 แผนภาพ Usecase diagram ของ Network Device Configuration Software

### 3.4.2 ผู้ใช้งานระบบ

User คือผู้ที่ต้องการใช้งาน โปรแกรม (หรือ Actor ของระบบ ซึ่งมีสัญลักษณ์เป็นรูปคน) โดยมีความต้องการที่จะทำการตั้งค่าเริ่มต้นของอุปกรณ์เครือข่ายให้สามารถเชื่อมต่อกันเบื้องต้นก่อน และต้องการที่จะทำการตั้งค่าการทำงานหรือแสดงค่าการทำงานของอุปกรณ์ที่อยู่ในเครือข่ายผ่านอุปกรณ์เครือข่ายเพียงตัวเดียว

### 3.4.3 ความสัมพันธ์ระหว่างผู้ใช้งานระบบกับ Usecase

ความสัมพันธ์ระหว่างผู้ใช้งานระบบกับ Usecase แทนด้วยเส้นลูกศร ซึ่งแสดงให้เห็นว่าผู้ใช้งานระบบสามารถใช้งานบริการใดของระบบได้บ้าง จากรูปสามารถแจกแจงได้ดังนี้

ผู้ใช้งานระบบสามารถเรียกใช้บริการ

- 1) สร้าง Topology ใหม่
- 2) แก้ไข Topology
- 3) โหลด Topology
- 4) ตั้งค่าการเชื่อมต่อเบื้องต้น
- 5) ตั้งค่าการทำงานพื้นฐานของเราเตอร์
- 6) กำหนดการ เราท์ติ้ง
- 7) ตั้งค่าการทำงานของเราเตอร์
- 8) ตรวจสอบสถานะการทำงานของเราเตอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 3.4.4 ความหมายของแต่ละ Usecase

รูปร่างในแผนภาพ Usecase ใช้แทนแต่ละ Usecase ซึ่งเป็นบริการที่ระบบมีไว้ให้บริการกับผู้ใช้งานระบบ บริการของระบบที่มีให้ใช้แสดงในรูปมีดังนี้

#### 3.4.4.1 บริการสร้าง Topology ใหม่

เป็นบริการหลักที่โปรแกรมจำเป็นต้องมี โดยจะมีพื้นที่ให้ผู้ใช้งานระบบใช้ในการออกแบบโครงสร้างของระบบเครือข่ายตามที่ผู้ใช้งานต้องการ ซึ่งจะมีอุปกรณ์เครือข่ายให้เลือกใช้ในการออกแบบอยู่สองประเภทคือเราท์เตอร์และสับ และมีสายที่เอาไว้ใช้ในการเชื่อมต่ออุปกรณ์เครือข่ายแต่ละตัวเข้าด้วยกัน โดยในบริการนี้ผู้ใช้งานระบบสามารถสร้าง Topology ได้ตามจำนวนที่ต้องการ

#### 3.4.4.2 บริการแก้ไข Topology

เป็นบริการที่มีความสามารถให้ทำการแก้ไขเพิ่มเติมโครงสร้างของระบบเครือข่ายได้ โดยยังมีอุปกรณ์และเครื่องมือให้ใช้ในการแก้ไขเหมือนกับบริการสร้าง Topology ใหม่ โดยในบริการนี้สามารถทำได้กับทั้ง Topology ที่สร้างขึ้นใหม่หรือ Topology ที่ทำการโหลดมา

#### 3.4.4.3 บริการโหลด Topology

เป็นบริการที่มีความสามารถให้ผู้ใช้งานโปรแกรมสามารถเรียกใช้งาน Topology ที่เคยสร้างไว้กลับมาใช้ใหม่ได้ ซึ่ง Topology เหล่านั้นจะต้องเคยถูกบันทึกจากโปรแกรมก่อน

#### 3.4.4.4 บริการตั้งค่าการเชื่อมต่อเบื้องต้น

เป็นบริการที่จะทำการกำหนดค่าการทำงานของอุปกรณ์เครือข่ายทุกตัวให้โดยอัตโนมัติ เพื่อให้อุปกรณ์เครือข่ายแต่ละตัวสามารถเชื่อมต่อกันได้ โดยที่ผู้ใช้งานไม่ต้องทำการกำหนดค่าใดๆ แต่จะต้องทำการเชื่อมต่อกับอุปกรณ์เครือข่ายผ่านพอร์ตอนุกรมเพื่อส่งค่าการทำงานเบื้องต้นไปยังอุปกรณ์

#### 3.4.4.5 บริการตั้งค่าการทำงานพื้นฐานของเราท์เตอร์

เป็นบริการที่มีให้สำหรับผู้ใช้งานระบบใช้ในการกำหนดค่าการทำงานพื้นฐานของเราท์เตอร์โดยง่ายได้ ซึ่งจะประกอบไปด้วย

- 1) ชื่อของเราท์เตอร์
- 2) รหัสผ่านที่ใช้ในการเข้าใช้งานผ่าน Console
- 3) รหัสผ่านที่ใช้ในการเข้าใช้งานผ่าน Virtual Terminal Lines
- 4) รหัสผ่านที่ใช้ในการเข้าใช้งาน Privileged โหมด
- 5) เนตเวิร์กแอดเดรส
- 6) สับเนตมาร์ก

โดยในบริการส่วนนี้ผู้ใช้งานจะทำการกำหนดค่าแค่บางส่วนที่ต้องการ หรือใช้ค่า

การทำงานที่ถูกกำหนดมาแต่ต้นก็ได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

#### 3.4.4.6 บริการกำหนดการเราท์ติ้ง

เป็นบริการที่ทำให้ผู้ใช้งานระบบสามารถเลือกรูปแบบในการส่งผ่านแพคเกจข้ามเน็ตเวิร์กที่ต่างกัน ได้ ซึ่งมีทั้งหมด 5 รูปแบบดังนี้

- 1) Static routing
- 2) RIP
- 3) RIPv2
- 4) EIGRP
- 5) OSPF

ซึ่งบริการเราท์ติ้งนี้จะทำการสร้างและส่งคำสั่งที่ใช้ในการเราท์ติ้งไปยังอุปกรณ์เครือข่ายทุกตัวที่ทำการเลือกโดยอัตโนมัติ

#### 3.4.4.7 บริการตั้งค่าการทำงานของเราท์เตอร์

เป็นบริการที่มีให้สำหรับผู้ใช้งานระบบทำการกำหนดค่าการทำงานพื้นฐาน รวมไปถึงการทำงานอื่นๆของเราท์เตอร์ได้ ซึ่งประกอบไปด้วย

- 1) ชื่อของเราท์เตอร์
- 2) รหัสผ่านที่ใช้ในการเข้าใช้งานผ่าน Console
- 3) รหัสผ่านที่ใช้ในการเข้าใช้งานผ่าน Virtual Terminal Lines
- 4) รหัสผ่านที่ใช้ในการเข้าใช้งาน Privileged โหมด
- 5) อินเตอร์เฟส
- 6) การเราท์ติ้ง

ซึ่งบริการตั้งค่าการทำงานของเราท์เตอร์นี้จะทำการสร้างและส่งคำสั่งที่ใช้ในการกำหนดค่าการทำงานไปยังเราท์เตอร์แต่ละตัวเพื่อทำการตั้งค่าโดยอัตโนมัติ

#### 3.4.4.8 บริการตรวจสอบสถานะของการทำงานของเราท์เตอร์

เป็นบริการที่จะทำการรวบรวมข้อมูลการทำงานทั้งหมดของเราท์เตอร์แต่ละตัว ตัวอย่างเช่น สถานะของอินเตอร์เฟส เป็นต้น มาแสดงให้ผู้ใช้งานเห็นในรูปแบบที่เข้าใจง่าย

#### 3.4.5 แผนภาพการวิเคราะห์ในเชิงวัตถุ

แผนภาพการวิเคราะห์ในเชิงวัตถุ หรือแผนภาพ Class ใช้แสดงความสัมพันธ์ของ Class ต่างๆ ในระบบ ตามมุมมองของวิธีการวิเคราะห์ในเชิงวัตถุ

เริ่มต้นด้วยแผนภาพ Component ซึ่งแสดงให้เห็นถึงการจัดเก็บกลุ่มของ Class ตามหมวดหมู่ที่สัมพันธ์กัน กลุ่มของ Class จะถูกจัดเก็บไว้ด้วยกันใน “แพคเกจ” แทนด้วยสัญลักษณ์โฟลเดอร์ ดังรูป 3.6

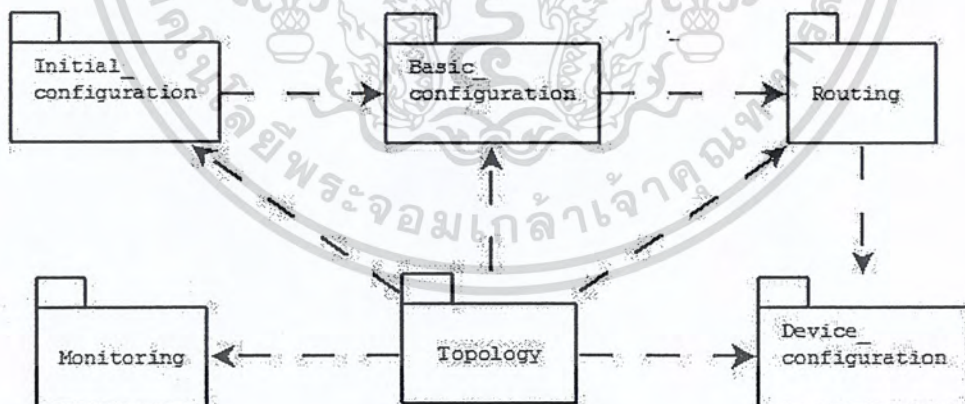
จากการวิเคราะห์ระบบ เราได้แบ่งการจัดเก็บ Class ออกเป็นหลายแพคเกจ ดังนี้

- 1) แพคเกจ Topology ใช้เก็บเหล่าคลาสที่เกี่ยวข้องกับการออกแบบโครงสร้างระบบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่นิยมนำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เครือข่ายซึ่งมีการอ้างอิงหรือมีความสัมพันธ์ถึงคลาสในแพ็คเกจ Initial\_Configuration แพ็คเกจ Basic\_Configuration แพ็คเกจ เราต์ติ้ง แพ็คเกจ Device\_Configuration และ แพ็คเกจ Monitoring (แสดงด้วยเส้นประ ในรูป 3.6)

- 2) แพ็คเกจ Initial\_Configuration ใช้เก็บเหล่าคลาสที่ใช้ในการตั้งค่าการทำงานเริ่มต้นของอุปกรณ์เครือข่าย ซึ่งมีการอ้างอิงหรือมีความสัมพันธ์ถึง คลาสในแพ็คเกจ Basic\_Configuration (แสดงด้วยเส้นประ ในรูป 3.6)
- 3) แพ็คเกจ Basic\_Configuration ใช้เก็บเหล่าคลาสที่ใช้ในการตั้งค่าการทำงานพื้นฐานของอุปกรณ์เครือข่าย ซึ่งมีการอ้างอิงหรือมีความสัมพันธ์ถึง คลาสในแพ็คเกจ เราต์ติ้ง (แสดงด้วยเส้นประ ในรูป 3.6)
- 4) แพ็คเกจ Routing ใช้เก็บเหล่าคลาสที่ใช้ในการกำหนดค่าการส่งผ่านแพ็คเกจข้ามเน็ตเวิร์ก ซึ่งมีการอ้างอิงหรือมีความสัมพันธ์ถึง คลาสในแพ็คเกจ Device\_Configuration (แสดงด้วยเส้นประ ในรูป 3.6)
- 5) แพ็คเกจ Device\_Configuration ใช้เก็บเหล่าคลาสที่ใช้ในการตั้งค่าการทำงานต่างๆ ของอุปกรณ์เครือข่าย
- 6) แพ็คเกจ Monitoring ใช้เก็บเหล่าคลาสที่ใช้ในการตรวจสอบสถานะ และค่าที่ทำการตั้งค่าไว้ของอุปกรณ์เครือข่ายทุกตัวในระบบ



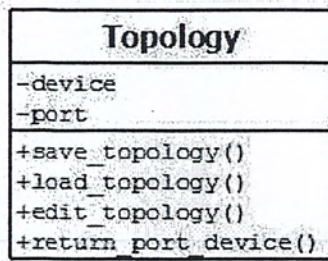
รูป 3.5 การแบ่งแพ็คเกจ

ต่อไปจะกล่าวถึงรายละเอียดภายในแต่ละแพ็คเกจ ซึ่งมีรายละเอียดดังต่อไปนี้

#### 3.4.5.1 แพ็คเกจ Topology

ภายในจะประกอบด้วยคลาส Topology

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูป 3.6 คลาสในแพ็คเกจ Topology

### 3.4.5.2 แพคเกจ Initial\_Configuration

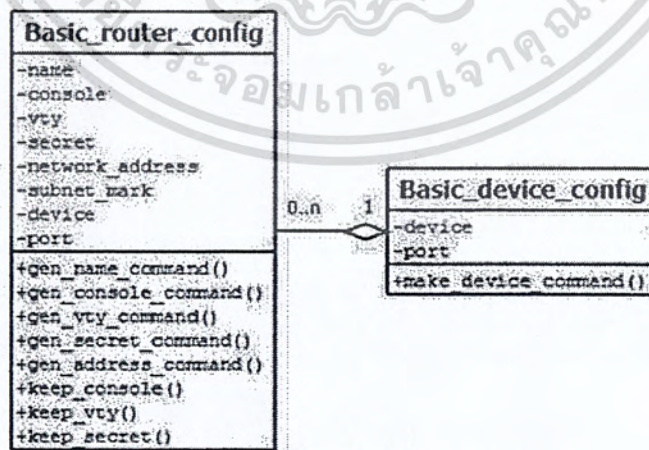
ภายในประกอบไปด้วยคลาส Initial



รูป 3.7 คลาสในแพคเกจ Initial\_Configuration

### 3.4.5.3 แพคเกจ Basic\_Configuration

ภายในประกอบไปด้วยคลาส Basic\_device\_config ซึ่งเป็นคอนเทนเนอร์ สำหรับ  
คลาส Basic\_router\_config

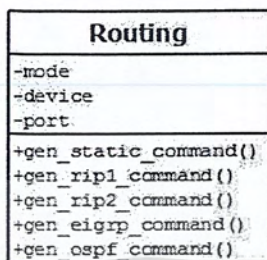


รูป 3.8 คลาสในแพคเกจ Basic\_configuration

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 3.4.5.4 แพคเกจ Routing

ภายในประกอบไปด้วยคลาส Routing

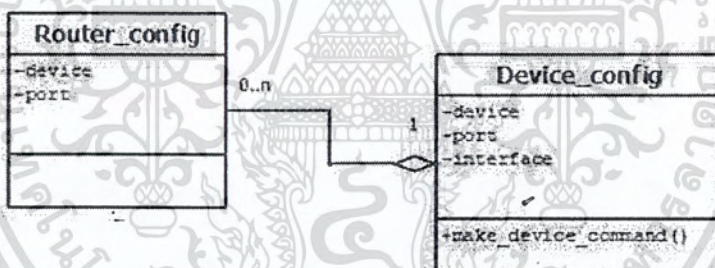


รูป 3.9 คลาสในแพคเกจ Routing

### 3.4.5.5 แพคเกจ Device\_Configuration

ภายในประกอบไปด้วยคลาส Device\_config ซึ่งเป็นคอนเทนเนอร์สำหรับคลาส

Router\_config

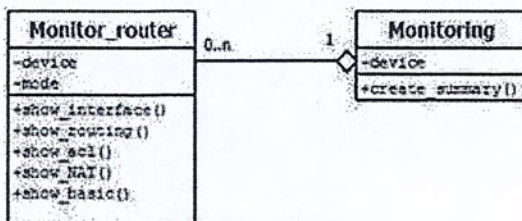


รูป 3.10 คลาสในแพคเกจ Device\_configuration

### 3.4.5.6 แพคเกจ Monitoring

ภายในประกอบไปด้วยคลาส Monitoring ซึ่งเป็นคอนเทนเนอร์สำหรับคลาส

Monitor\_router

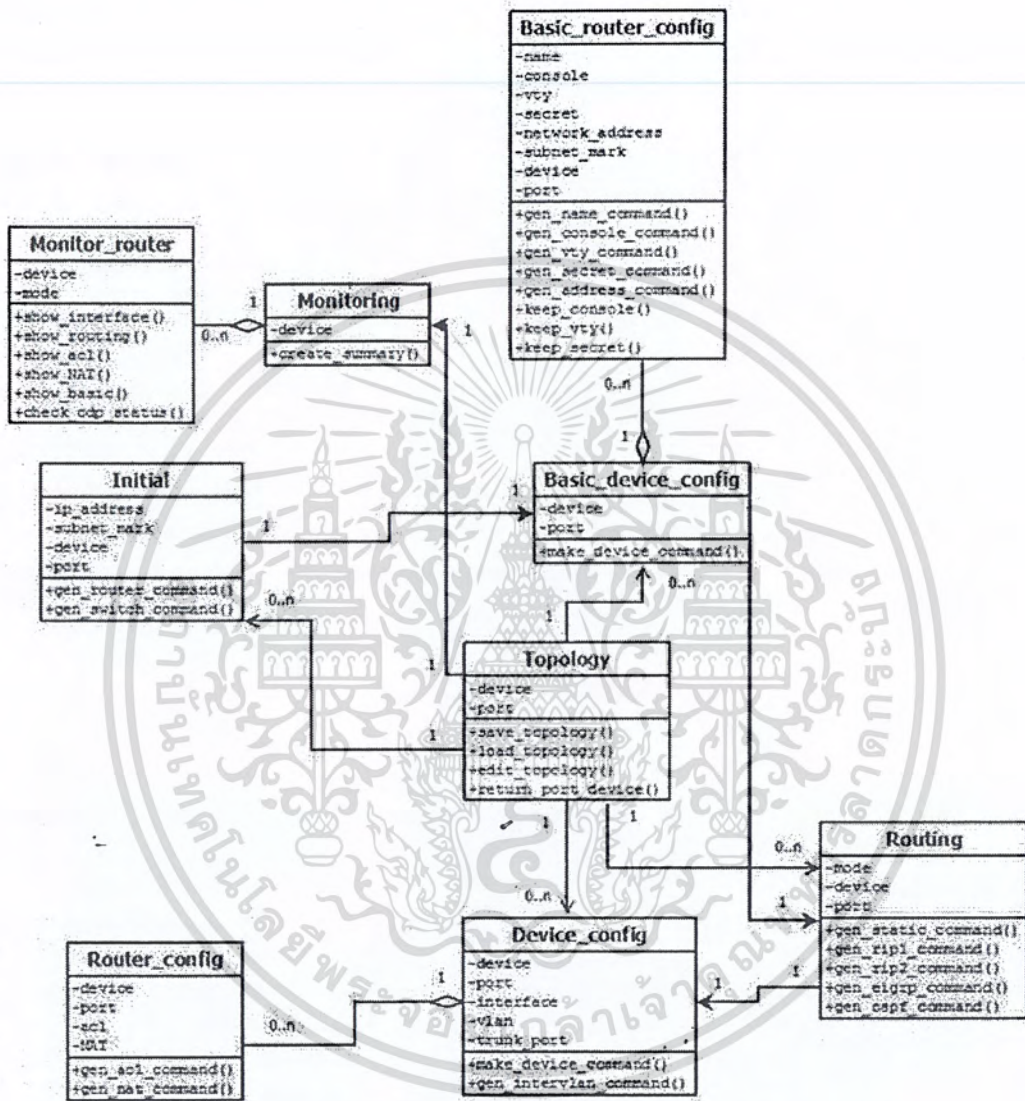


รูป 3.11 คลาสในแพคเกจ Monitoring

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ซึ่งเมื่อนำมาเชื่อมโยงความสัมพันธ์ระหว่างแต่ละคลาสแล้วจะได้เป็นแผนภาพ

Class ดังนี้



รูป 3.12 แผนภาพ Class ของโปรแกรม Network Device Configuration

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

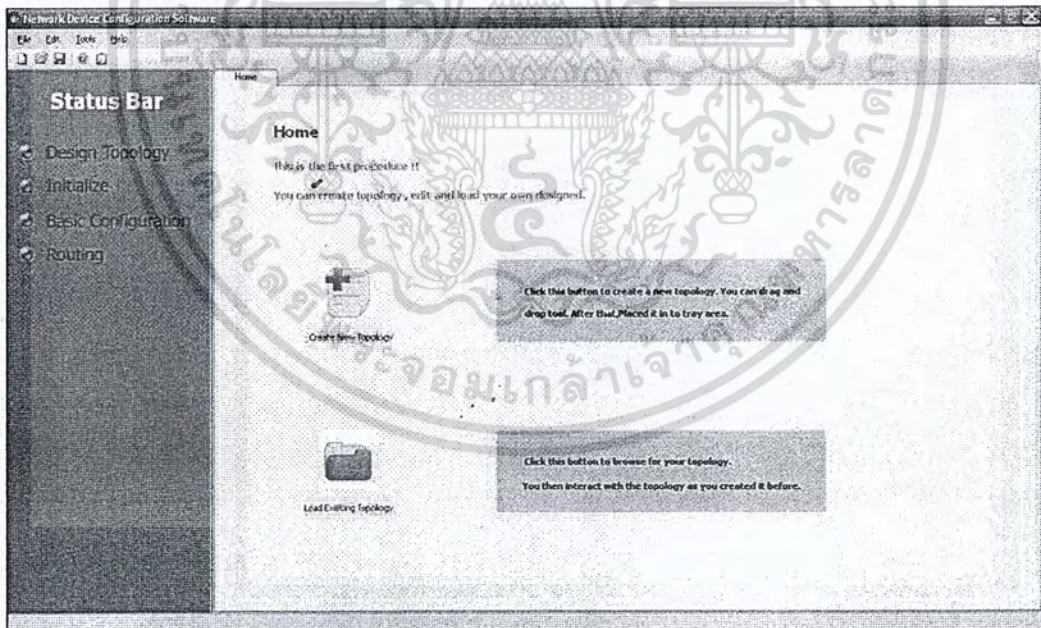
## บทที่ 4

# ผลการทดลองการใช้โปรแกรม

### 4.1 ตัวอย่างหน้าจอโปรแกรม

#### 4.1.1 รูปแบบหน้าต่างหลักโปรแกรม

หน้าต่างที่เกิดขึ้นหลังจากทำการเปิดโปรแกรมขึ้นมาดังรูป 4.1 ซึ่งเมื่อเปิดโปรแกรมขึ้นมา โปรแกรมจะเริ่มการทำงานจากส่วนของการออกแบบ Topology โดยทางแถบแสดงสถานะของการทำงานทางด้านซ้ายจะแสดงขั้นตอนการทำงานโดยรวมของโปรแกรม และในพื้นที่ทำงานที่ใช้ในการออกแบบเครือข่ายจะแสดงดังรูป 4.1 ซึ่งจะมีรายละเอียดและปุ่มให้เลือกการทำงาน และจะมีโหมดการทำงานให้เลือกอยู่ 2 โหมด คือ การสร้างเครือข่ายใหม่ และการเปิดไฟล์เครือข่ายที่มีอยู่แล้วที่ผู้ใช้ได้ทำการบันทึกไว้ในระหว่างการทำงานในโปรแกรม เพื่อให้เกิดความสะดวกสบายต่อผู้ใช้ในกรณีที่ผู้ใช้ต้องการแก้ไข หรือเปลี่ยนแปลงข้อมูลบางอย่างในการออกแบบที่ได้สร้างไว้ก่อนหน้า



รูป 4.1 หน้าต่างโปรแกรมโดยรวม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

#### 4.1.2 รูปแบบหน้าต่างขั้นตอนการออกแบบเครือข่าย

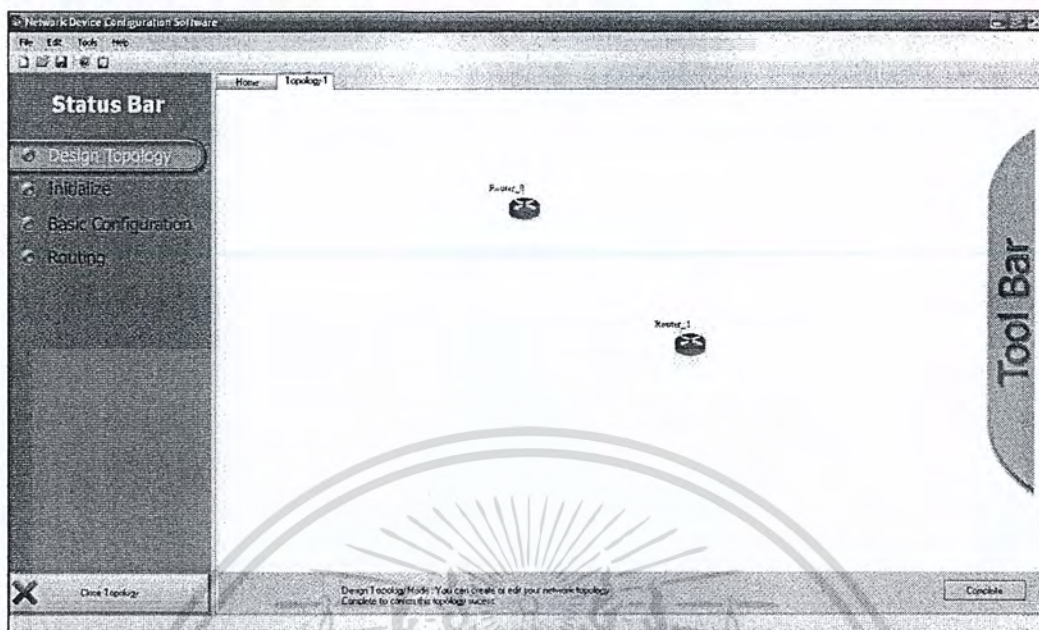
โปรแกรมจะเริ่มทำงานได้เมื่อ ผู้ใช้ทำการสร้างพื้นที่ทำงาน โดยเลือกในส่วนหน้าหลักของโปรแกรมหรือไปที่แถบเมนูหลัก File -> New หรือกดคีย์บอร์ด Ctrl+N จะมีพื้นที่การทำงานให้สามารถออกแบบ หรือสร้างเครือข่ายขึ้นมาใหม่ดังรูป 4.2



รูป 4.2 หน้าต่างขั้นตอนการออกแบบเครือข่าย

เมื่อผู้ใช้งานเลือกทำการสร้างเครือข่ายใหม่ พื้นที่การทำงานจะเป็นพื้นที่ว่างบริเวณตรงกลางของหน้าต่าง ซึ่งจะมีแถบเครื่องมือหรืออุปกรณ์เครือข่ายอยู่ทางด้านขวา ให้ผู้ใช้งานเลือกใช้การทำงานของอุปกรณ์บนแถบเครื่องมือทางด้านขวา จะมีลักษณะเป็นสัญลักษณ์อุปกรณ์เครือข่าย ซึ่งผู้ใช้งานโปรแกรมสามารถลากอุปกรณ์วางลงบนพื้นที่การทำงานและเชื่อมต่ออุปกรณ์ต่างๆ ได้ตามต้องการ ผู้ใช้สามารถลบอุปกรณ์ได้ดังรูป 4.4

ซึ่งการทำงานของแถบเครื่องมือทางด้านขวาจะเป็นการทำงานแบบลากวางตามรูป 4.3 ซึ่งจะเห็นว่าผู้ใช้งานโปรแกรมสามารถเลือกอุปกรณ์และลากมาวางลงบนพื้นที่การทำงานได้ อีกทั้งเมื่อขณะเลือกการเชื่อมต่อระหว่างอุปกรณ์เครือข่ายจะแสดงพอร์ตที่อุปกรณ์เครือข่ายตัวนั้นมีให้ผู้ใช้ทำการเลือกเชื่อมต่อ โดยตัวโปรแกรมจะทำการแสดงจำนวนพอร์ตที่ใช้ในการเชื่อมต่อทั้งสิ้น 4 พอร์ต คือ Serial0/0, Serial0/1, FastEthernet0/0, FastEthernet0/1 ดังรูป 4.5

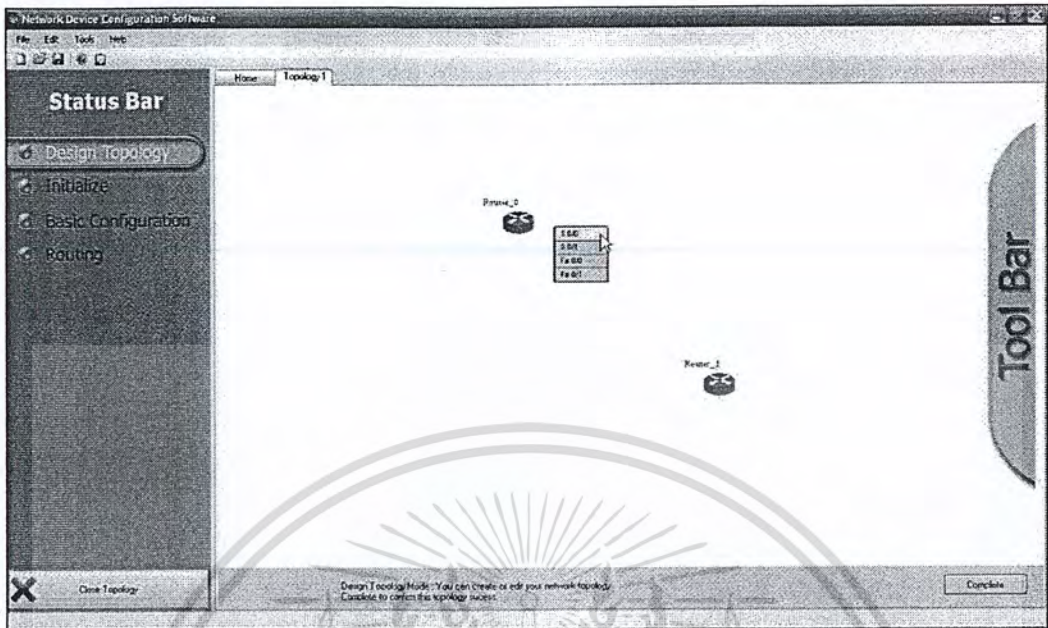


รูป 4.3 หน้าต่างขั้นตอนการลากวางอุปกรณ์



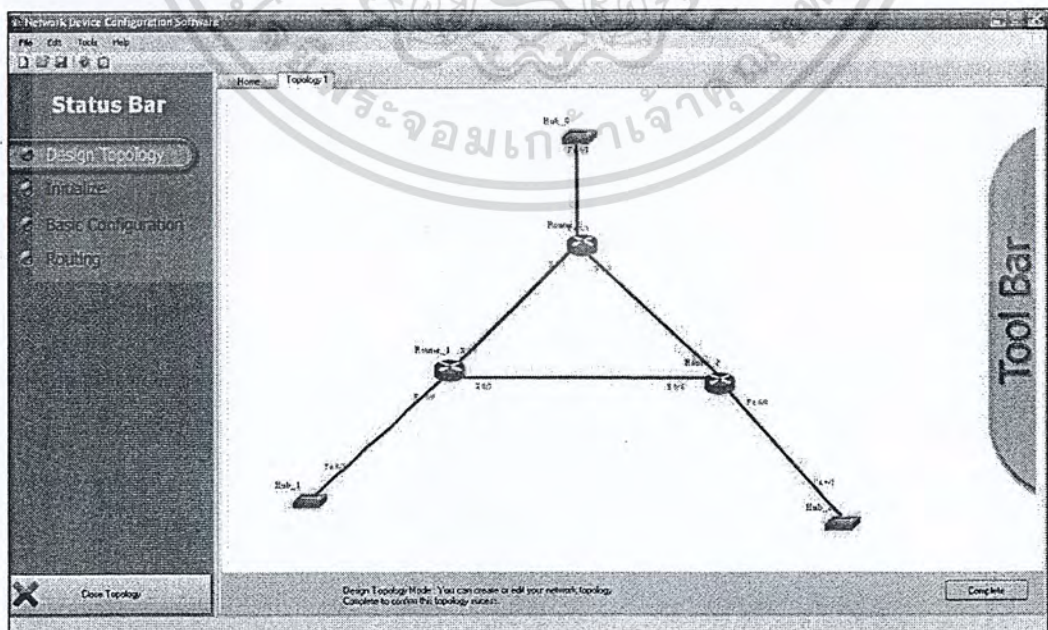
รูป 4.4 การลบอุปกรณ์เครือข่าย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



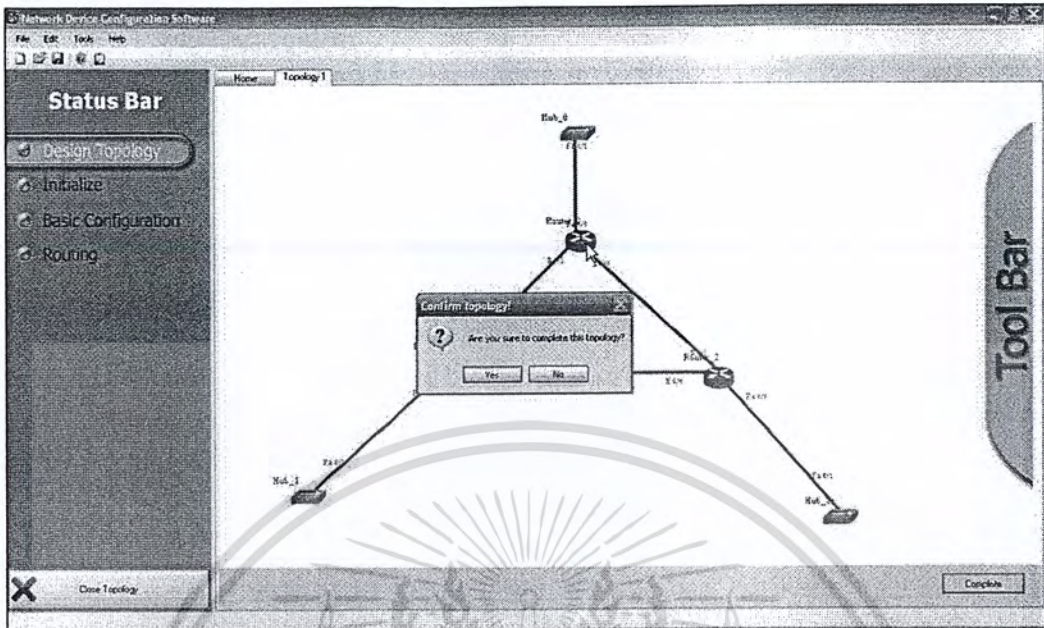
รูป 4.5 การเลือกพอร์ตของอุปกรณ์เพื่อใช้ในการเชื่อมต่อ

เมื่อผู้ใช้งาน โปรแกรมทำการสร้างเครือข่ายเรียบร้อยแล้ว จะต้องทำการยืนยันการทำงานเสร็จสิ้นในขั้นตอนการออกแบบเครือข่าย โดยผู้ใช้งานจะต้องทำการคลิกปุ่มยืนยันการทำงานที่ปุ่มตรงมุมล่างขวาดังรูป 4.6 และ 4.7 เพื่อ ไปทำงานในขั้นตอนการตั้งค่าเริ่มต้นให้กับอุปกรณ์เครือข่าย ถ้าผู้ใช้งาน โปรแกรมไม่ทำการยืนยัน โปรแกรมจะถือว่ายังอยู่ในขั้นตอนการออกแบบและแก้ไขเครือข่าย



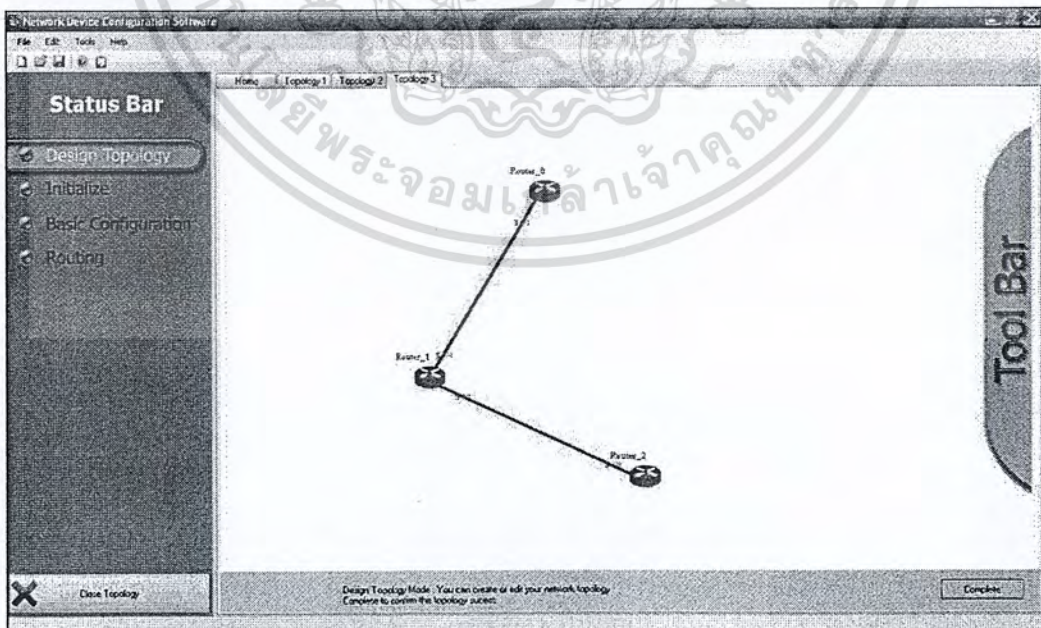
รูป 4.6 การออกแบบเครือข่ายเสร็จสิ้นตามต้องการ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่นิยมนำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



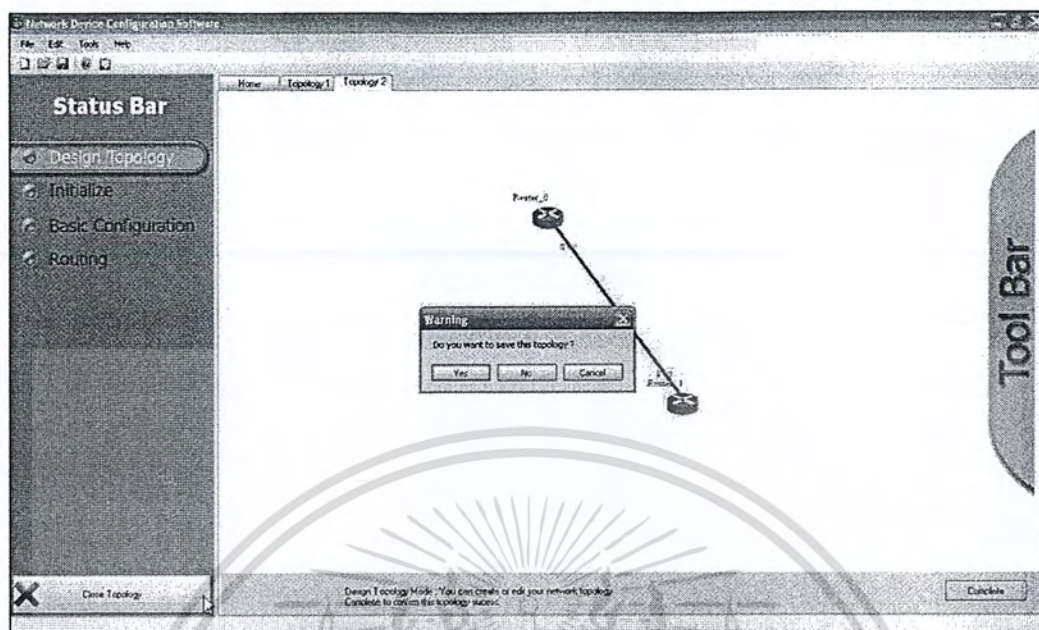
รูป 4.7 กดปุ่ม Complete เพื่อยืนยันการสร้างเครือข่ายเสร็จสิ้น

เมื่อผู้ใช้งานทำการสร้างเครือข่ายใหม่โดยการเลือก Ctrl+N หรือ Create New Topology หน้าต่างโปรแกรมจะมีแถบพื้นที่การทำงานเพิ่มขึ้นเพื่อให้ผู้ใช้สามารถทำการออกแบบเครือข่ายอีกเครือข่ายหนึ่งได้ตามต้องการ ดังรูป 4.8 และสามารถลบเครือข่ายที่สร้างได้ เมื่อต้องการลบจะมีไอคอนเพื่อสอบถามความแน่ใจของผู้ใช้ดังรูป 4.9



รูป 4.8 การสร้างเครือข่ายใหม่

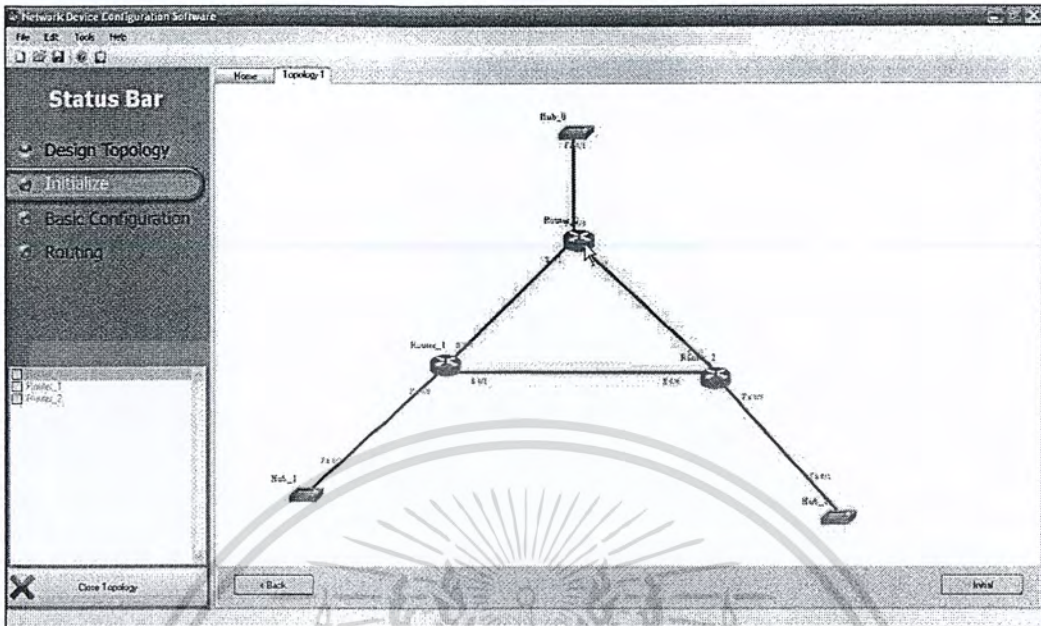
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



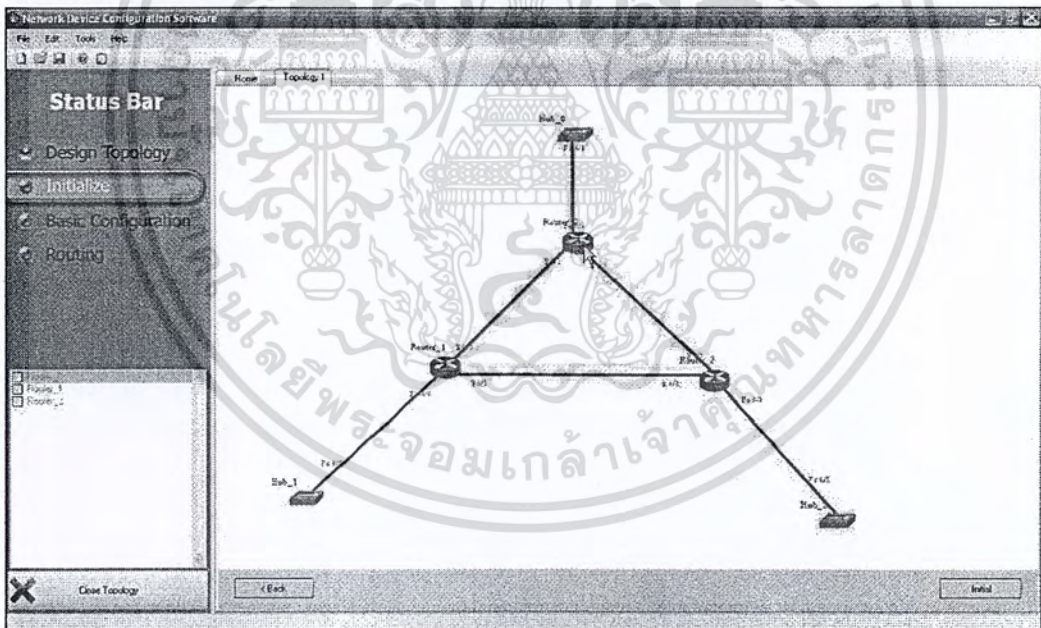
รูป 4.9 การลบแถบพื้นที่การออกแบบเครือข่าย

#### 4.1.3 รูปแบบหน้าต่างขั้นตอนการตั้งค่าเริ่มต้นให้กับเครือข่าย

เมื่อได้ทำการออกแบบเครือข่ายเสร็จสิ้นแล้วดังรูป 4.10 จากนั้นจะเป็นการตั้งค่าเริ่มต้นให้กับเครือข่ายนั้น ผู้ใช้โปรแกรมจะต้องทำการจัดเรียงอุปกรณ์จริงและทำการเชื่อมต่ออุปกรณ์ให้มีลักษณะเหมือนกับที่ได้ทำการออกแบบไว้ในโปรแกรม ซึ่งทางตัวโปรแกรมจะสร้างชุดคำสั่งขึ้นมา เพื่อทำการตั้งค่าเริ่มต้นให้กับอุปกรณ์ทุกตัวเอง เพื่อให้อุปกรณ์เครือข่ายได้มีการตั้งค่าเบื้องต้นไว้ใช้ในการอ้างอิงตัวโดยใช้บริการเทคโนโลยีในการติดต่อไปยังอุปกรณ์ตัวอื่นๆ เพื่อส่งคำสั่งการทำงานที่ผู้ใช้ได้ทำการเปลี่ยนแปลงการตั้งค่าพื้นฐาน โดยผู้ใช้โปรแกรมจะต้องเลือกอุปกรณ์เครือข่ายที่ต้องการทำการตั้งค่าเริ่มต้นโดยการดับเบิลคลิกเลือกที่ตัวอุปกรณ์ที่ต้องการดังรูป 4.11 แล้วจึงกดปุ่มทางมุมล่างขวาของหน้าต่างเพื่อดำเนินการตั้งค่าเริ่มต้นให้กับอุปกรณ์



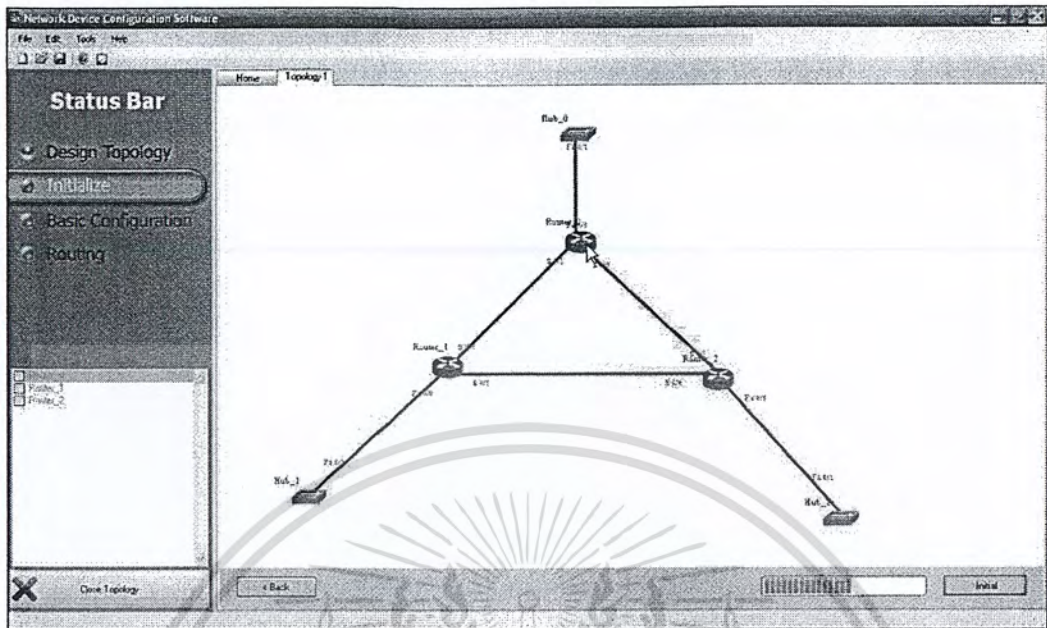
รูป 4.10 การออกแบบเครือข่ายโดยผู้ใช้โปรแกรมเสร็จสิ้น



รูป 4.11 การเลือกอุปกรณ์เครือข่ายที่ต้องการตั้งค่าเริ่มต้น

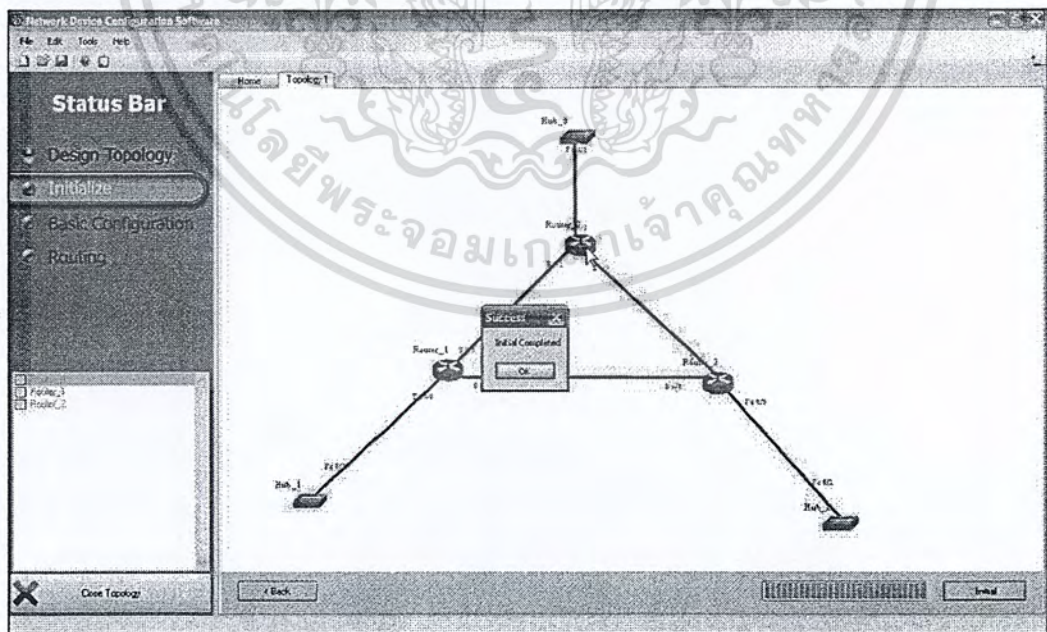
จากรูป 4.12 เมื่อผู้ใช้ทำการเลือกอุปกรณ์เครือข่ายที่ต้องการตั้งค่าเริ่มต้น แล้วจากนั้นให้ผู้ใช้กดปุ่ม Initial เพื่อให้โปรแกรมทำการส่งคำสั่งการตั้งค่าเริ่มต้นไปยังอุปกรณ์เครือข่ายตัวนั้น เมื่อกดปุ่ม Initial แล้วโปรแกรมจะส่งคำสั่งและแสดงสถานะของการทำงานให้ผู้ใช้ทราบดังรูป 4.13 โดยผู้ใช้จะต้องทำการตั้งค่าเริ่มต้นให้กับอุปกรณ์เครือข่ายทุกตัวที่ได้ออกแบบไว้ในโปรแกรม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูป 4.12 โปรแกรมกำลังดำเนินการตั้งค่าเริ่มต้น

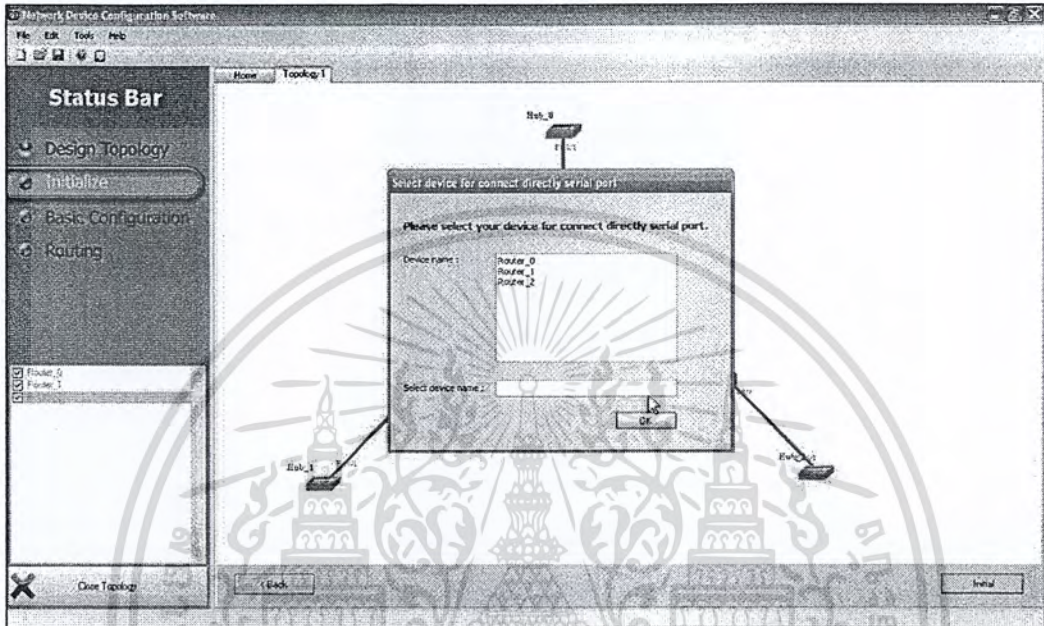
เมื่อการตั้งค่าเริ่มต้นให้กับอุปกรณ์เครือข่ายสำเร็จเรียบร้อยแล้ว โปรแกรมจะมีไอคอนแสดงขึ้นมาเพื่อแจ้งแก่ผู้ใช้ว่าการตั้งค่าเริ่มต้นให้กับอุปกรณ์เครือข่ายตัวที่ผู้ใช้เลือกสำเร็จเรียบร้อยแล้วดังรูป 4.13



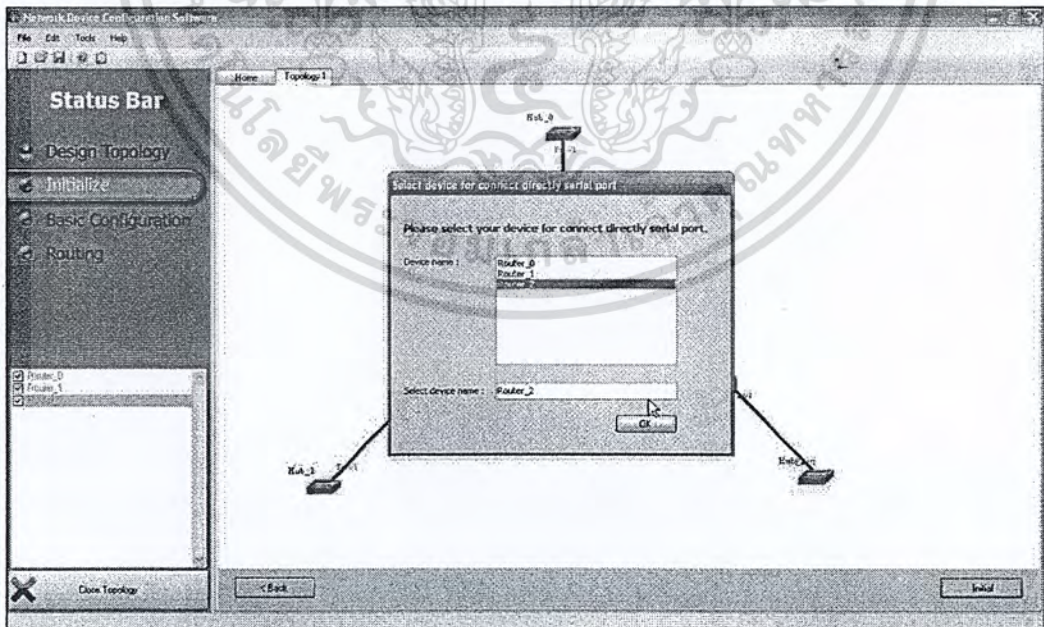
รูป 4.13 ไอคอนแสดงการตั้งค่าเริ่มต้นสำเร็จ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เมื่อทำการตั้งค่าเริ่มต้นให้กับอุปกรณ์เครือข่ายทั้งหมดแล้ว ผู้ใช้ต้องทำการเลือกอุปกรณ์ที่เชื่อมต่อโดยตรงกับคอนโซล เพื่อให้อุปกรณ์เครือข่ายตัวนั้น เป็นตัวกลางที่ใช้ในการส่งคำสั่งไปยังอุปกรณ์เครือข่ายตัวอื่นๆ ดังรูป 4.14 และ 4.15



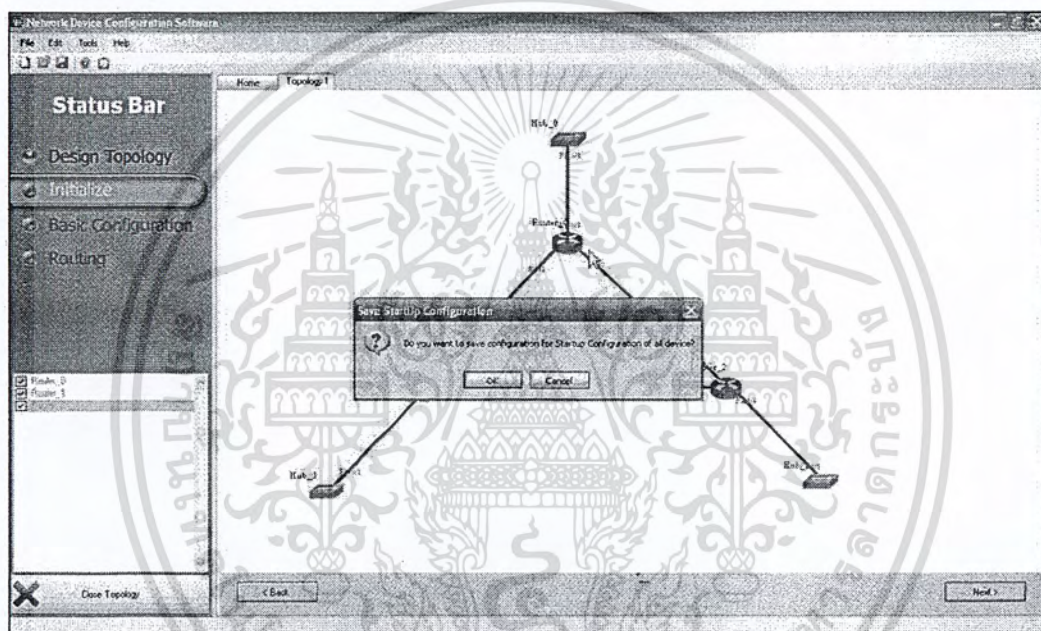
รูป 4.14 ใ้กดแสดงการเลือกอุปกรณ์ที่ผู้ใช้งานต้องการให้เป็นตัวกลางในการส่งคำสั่ง



รูป 4.15 ผู้ใช้โปรแกรมทำการเลือกอุปกรณ์ที่ใช้เป็นตัวกลางในการส่งคำสั่ง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หลังจากโปรแกรมทำการตั้งค่าเริ่มต้นให้กับอุปกรณ์เครือข่ายเสร็จเรียบร้อยแล้ว ผู้ใช้ต้องทำการกดปุ่ม Next แล้วจะมีไดอะล็อกแสดงขึ้นมาสอบถามผู้ใช้ว่าต้องการบันทึกการตั้งค่าเริ่มต้นนี้ลงไป ใน startup-config หรือไม่ เพื่อให้ผู้ใช้ไม่ต้องทำการตั้งค่าเริ่มต้นใหม่อีกครั้ง ในกรณีที่ผู้ใช้กลับมาใช้อุปกรณ์เครือข่ายชุดเก่าใหม่อีกครั้ง โปรแกรมจะทำการบันทึกการตั้งค่าเริ่มต้นไปไว้ใน startup-config ในตัวของอุปกรณ์เราเตอร์ เมื่อผู้ใช้เปิดเครื่องขึ้นมาใหม่การตั้งค่าต่างๆของเครื่องจะเป็นการตั้งค่าที่ได้ตั้งค่าเริ่มต้นไว้แล้วทันที ทำให้สะดวกในการใช้งานยิ่งขึ้น โดยผู้ใช้ไม่จำเป็นต้องทำการตั้งค่าเริ่มต้นให้กับอุปกรณ์ใหม่อีกครั้งดังรูป 4.16

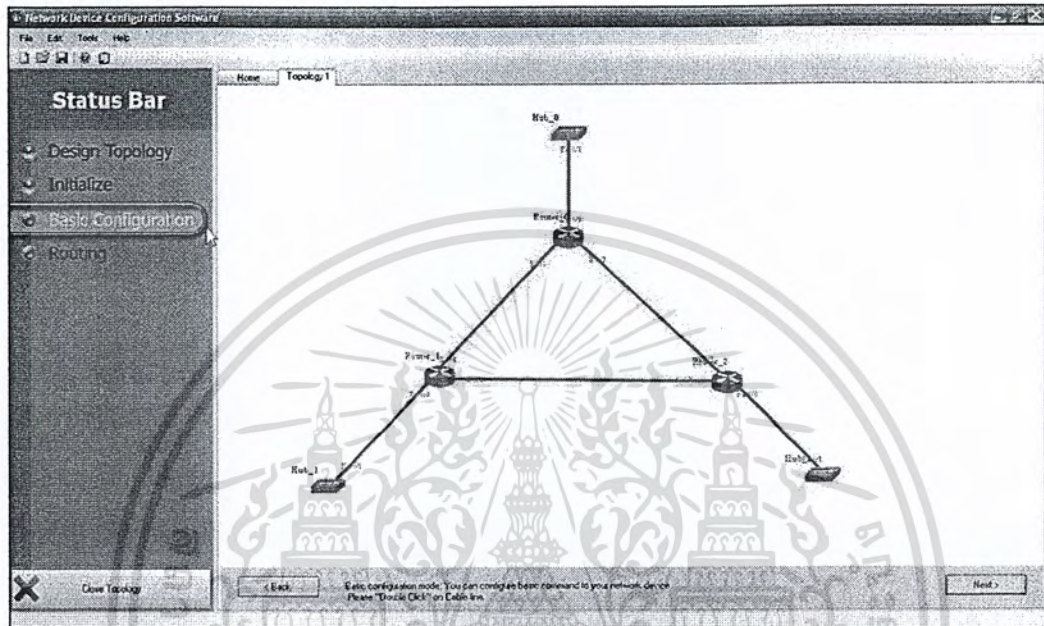


รูป 4.16 ไดอะล็อกแสดงการยืนยันการบันทึกค่าการทำงานเริ่มต้นลงใน startup-config

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

#### 4.1.4 รูปแบบหน้าต่างขั้นตอนการตั้งค่าพื้นฐานให้อุปกรณ์ในเครือข่าย

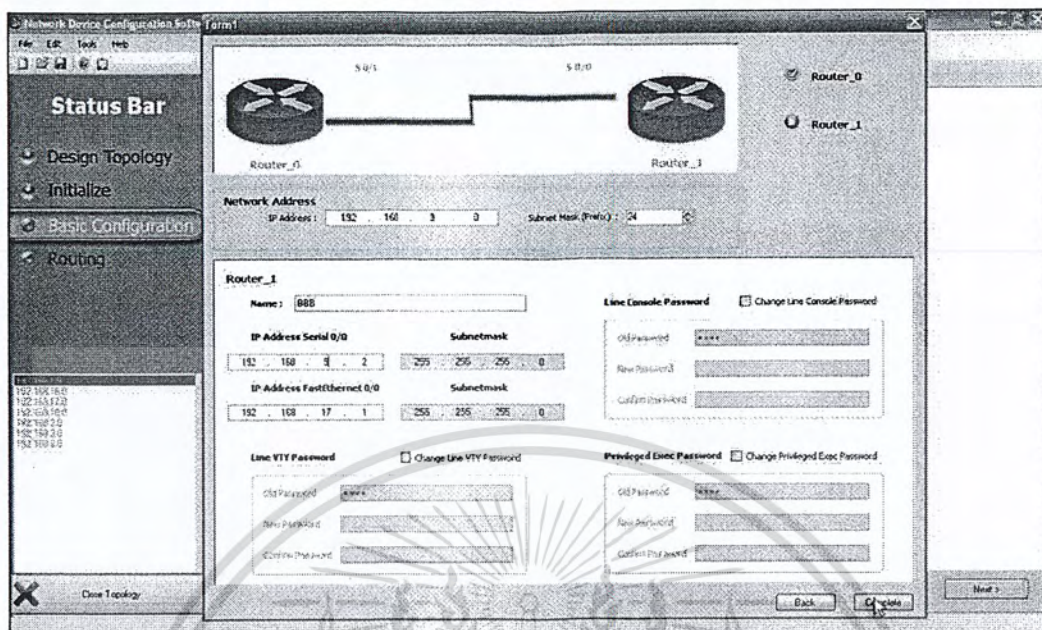
การตั้งค่าพื้นฐานให้อุปกรณ์ในเครือข่ายนั้น จะทำได้โดยผู้ใช้ทำการดับเบิลคลิกเลือกที่บริเวณสายที่เชื่อมต่อระหว่างตัวของอุปกรณ์ ดังรูป 4.17



รูป 4.17 การเลือกตั้งค่าพื้นฐานให้อุปกรณ์ชนิดเราเตอร์

หลังจากผู้ใช้เลือกส่วนที่ต้องการทำการตั้งค่าพื้นฐาน โปรแกรมจะมีไดอะล็อกที่ใช้ในการตั้งค่าพื้นฐานต่างๆ แสดงขึ้นมา ซึ่งจะแสดงข้อมูลที่โปรแกรมตั้งค่าพื้นฐานของอุปกรณ์ที่เชื่อมต่ออยู่กับสายที่ทำการเลือกนั้นดังรูป 4.18 ซึ่งแสดงไดอะล็อกของการตั้งค่าการเชื่อมต่อระหว่างเราเตอร์กับเราเตอร์

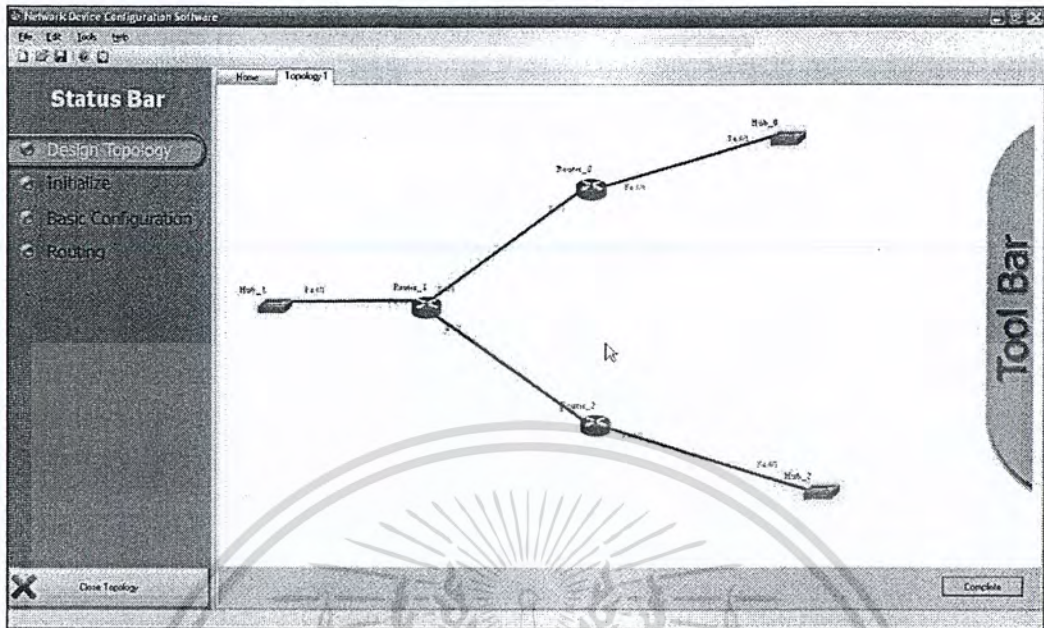
และในไดอะล็อกการตั้งค่าพื้นฐานระหว่างเราเตอร์กับเราเตอร์จะแบ่งการตั้งค่าของอุปกรณ์ออกเป็น 3 ส่วน ซึ่งจะแบ่งออกเป็นการตั้งค่าเครือข่ายระหว่างอุปกรณ์เราเตอร์ทั้งสองตัว การตั้งค่าไอพีของแต่ละพอร์ตของอุปกรณ์เครือข่าย และการตั้งพาสเวิร์ดทั้งสามชนิด คือ Line Console Password, Privileged Password, Line VTY Password



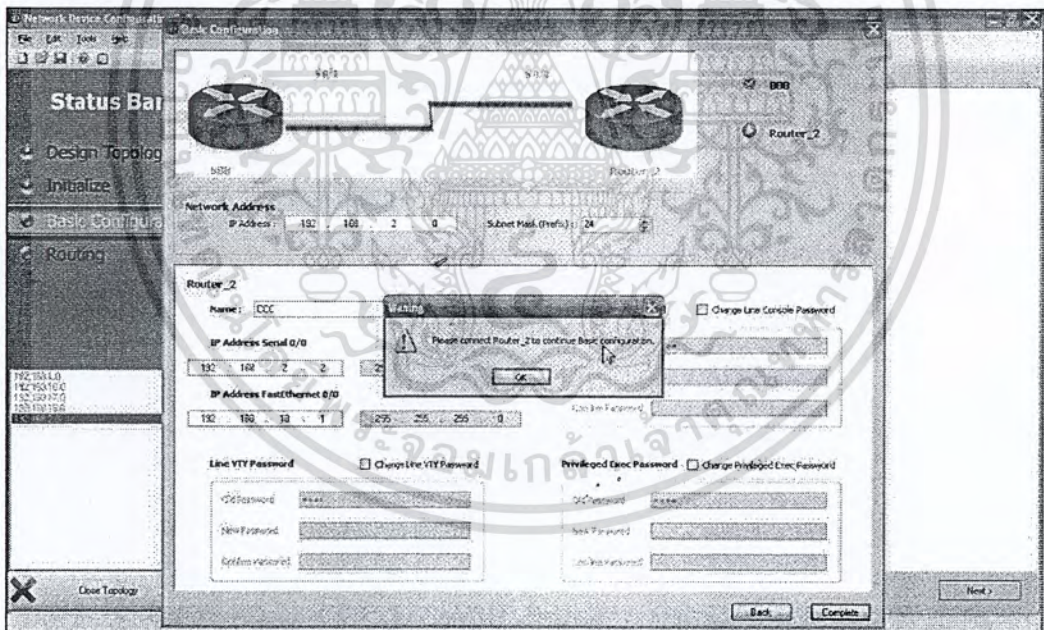
รูป 4.18 ไดอะแกรมแสดงการตั้งค่าพื้นฐานระหว่างเราเตอร์กับเราเตอร์

ในการออกแบบเครือข่ายนั้น ผู้ใช้สามารถออกแบบเครือข่ายให้เป็นแบบกึ่งได้ดังรูป 4.19 การออกแบบเครือข่ายแบบกึ่งนั้นจะมีข้อพิเศษในขั้นตอนการทำ Basic Configuration โดยผู้ใช้ต้องทำการเชื่อมต่อสายคอนโซลไปยังอุปกรณ์ที่โปรแกรมแจ้งเตือนเพื่อให้การตั้งค่าออกมาสมบูรณ์ เพราะโปรแกรม ไม่สามารถใช้บริการเทลเน็ตผ่านไปยังอุปกรณ์ที่โปรแกรมแจ้งเตือนได้โดยตรง เนื่องจากเมื่อผู้ใช้ทำการเปลี่ยนค่าไอพีแอดเดรสจะทำให้การเชื่อมต่อผ่านเส้นทางนั้นถูกตัดขาดโดยทันที และไม่สามารถกลับมายังอุปกรณ์ตัวกลางที่ใช้ในการส่งคำสั่งได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



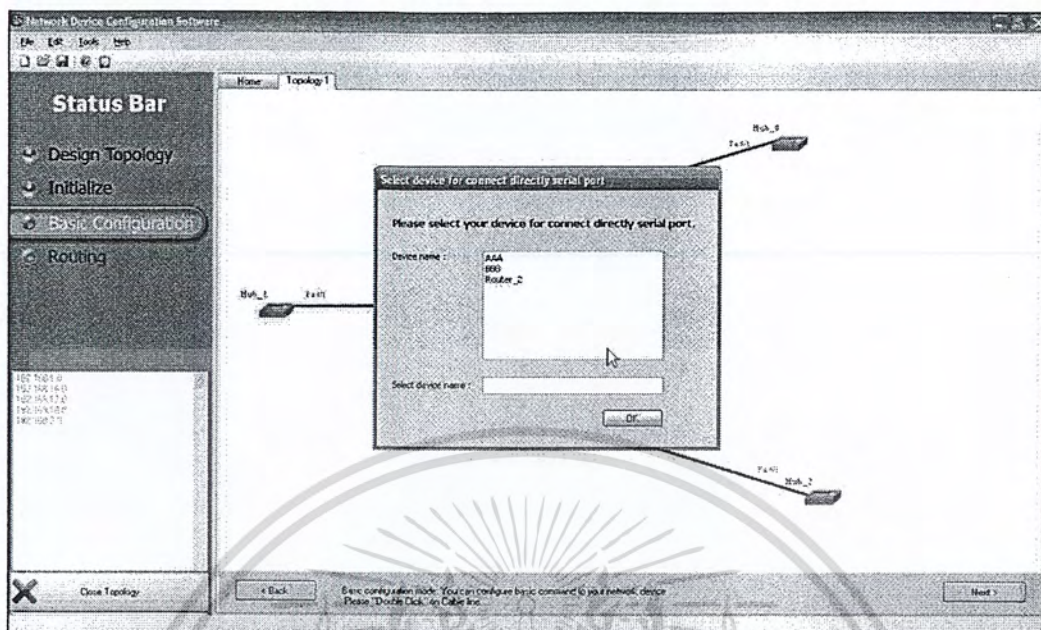
รูป 4.19 การออกแบบเครือข่ายแบบกึ่ง



รูป 4.20 การแจ้งเตือนเพื่อให้ผู้ใช้ทำการเปลี่ยนการเชื่อมต่อไปยังอุปกรณ์ตัวอื่นเพื่อส่งคำสั่ง

การเชื่อมต่อแบบกึ่งนั้น ผู้ใช้ต้องทำการเลือกอุปกรณ์ตัวกลางในการส่งข้อมูลใหม่ เพราะมีการเปลี่ยนการเชื่อมต่อของโปรแกรมกับอุปกรณ์ในการตั้งค่าดังรูป 4.20 ดังนั้นผู้ใช้ต้องทำการเลือกอุปกรณ์ที่จะเป็นตัวกลางในการส่งคำสั่งใหม่ดังรูป 4.21

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



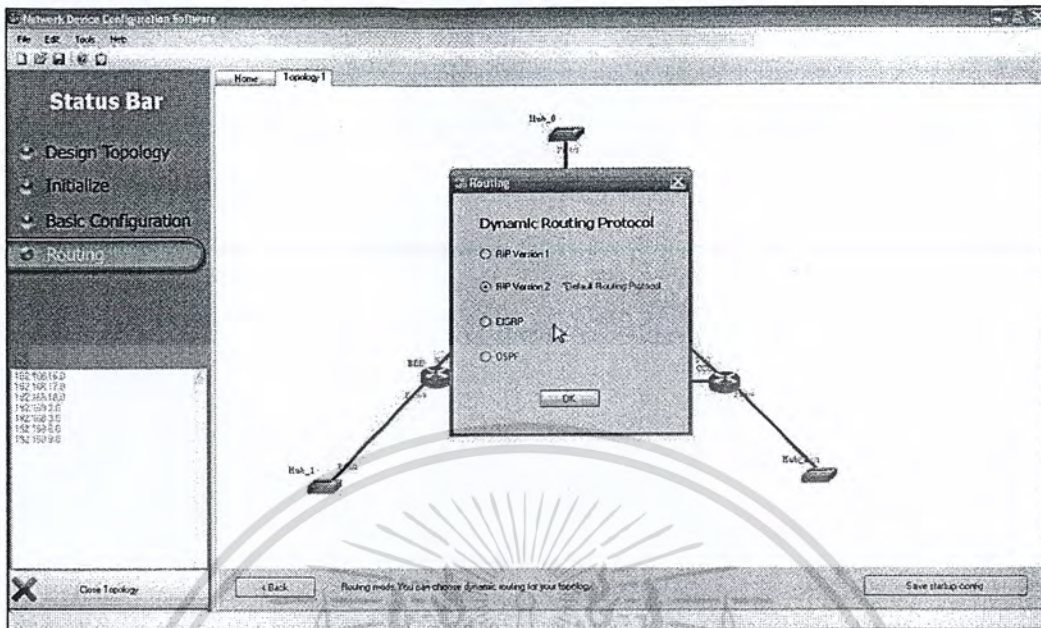
รูป 4.21 การแจ้งเตือนเพื่อให้ผู้ใช้ทำการเลือกอุปกรณ์ตัวกลางใหม่ในกรณีแบบกึ่ง

#### 4.1.5 รูปแบบหน้าต่างขั้นตอนการเราท์ติ้งที่ในเครือข่าย

จากนั้นผู้ใช้ต้องทำการเราท์ติ้งที่ในเครือข่าย เพื่อให้เครือข่ายแต่ละเครือข่ายสามารถติดต่อสื่อสารกันได้ โดยผู้ใช้จะต้องทำการเลือกโปรโตคอลในการเราท์ติ้ง โดยโปรแกรมจะมีให้เลือกการทำเราท์ติ้งแบบ Dynamic Route เมื่อผู้ใช้โปรแกรมทำการเลือกโปรโตคอลในการเราท์ติ้งเสร็จสิ้นแล้ว จะต้องกดปุ่มยืนยันเพื่อให้โปรแกรมทำการส่งคำสั่งไปยังอุปกรณ์เครือข่ายทุกตัว

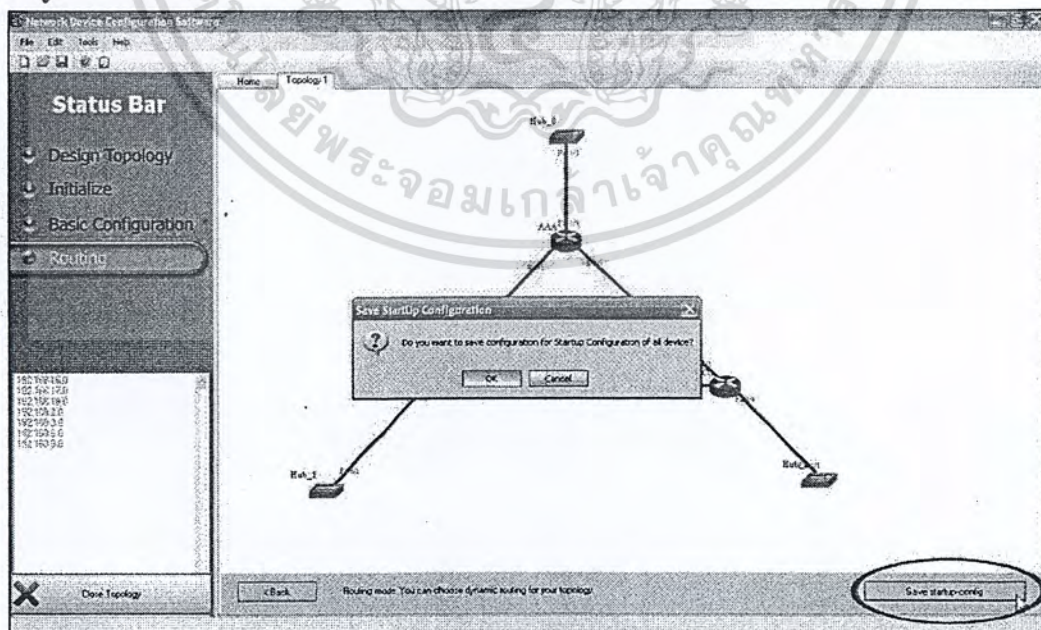
ดังรูป 4.22

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



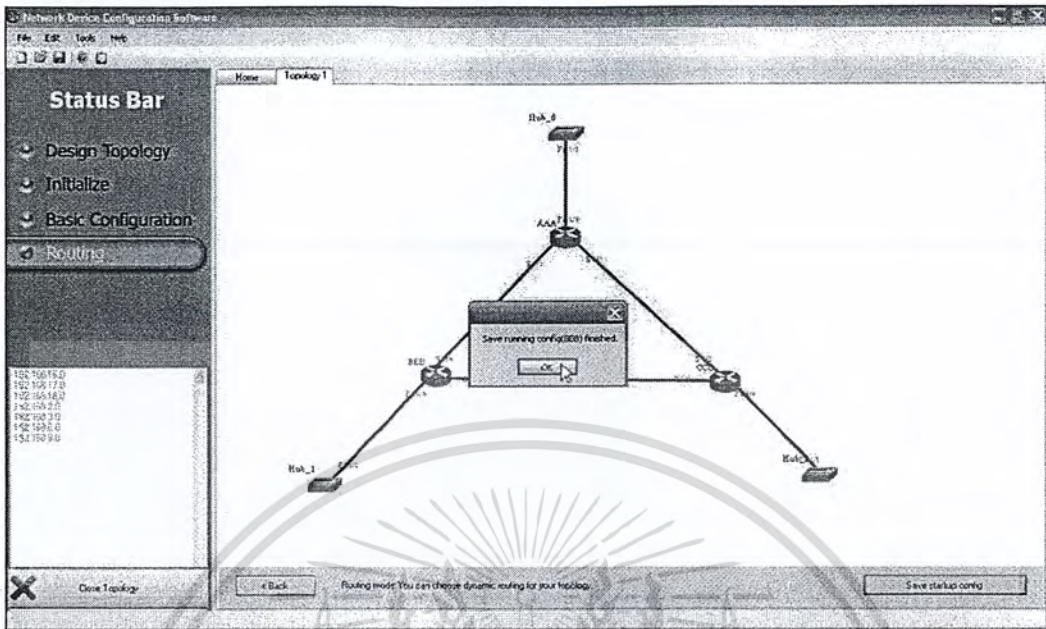
รูป 4.22 โดอะลือกแสดงโปรโตคอลแบบ Dynamic Route

เมื่อผู้ใช้ทำการเราท์ติ้งที่เสร็จเรียบร้อยแล้ว จากนั้นผู้ใช้สามารถทำการบันทึกการตั้งค่าทั้งหมดลงไปใน startup-config ให้การตั้งค่าต่างๆ ถูกบันทึกลงไปในอุปกรณ์เครือข่าย เพื่อให้เมื่อผู้ใช้กลับมาใช้งานใหม่อีกครั้ง ผู้ใช้ไม่ต้องทำการตั้งค่าการทำงานให้กับอุปกรณ์เครือข่ายอีกครั้ง ดังรูป 4.23 และ 4.24



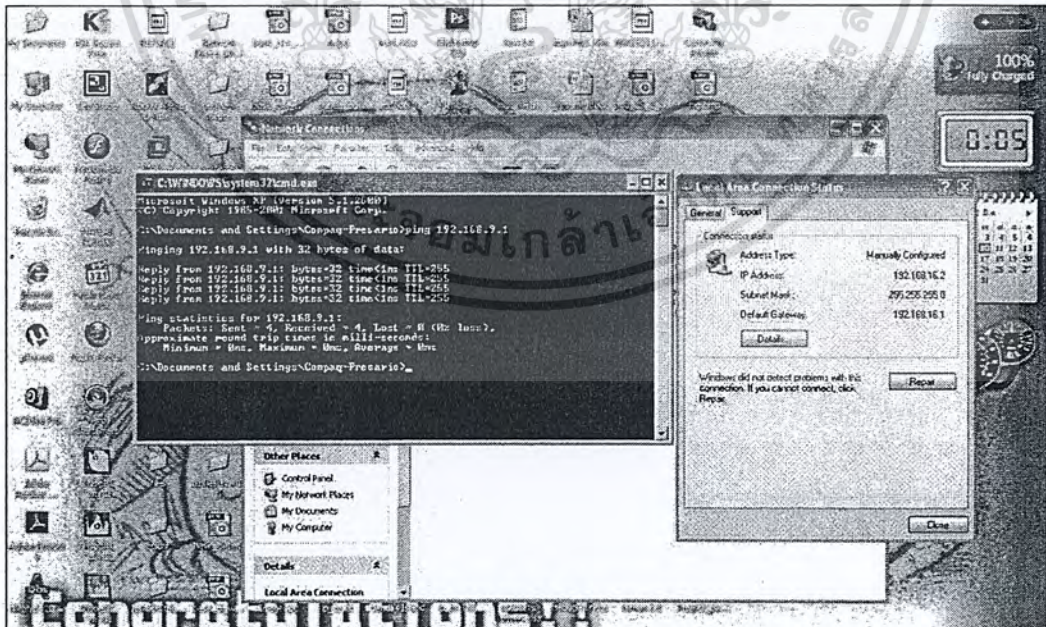
รูป 4.23 ผู้ใช้ทำการบันทึกข้อมูลที่ตั้งค่าไว้ลงใน startup-config

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูป 4.24 การบันทึกการตั้งค่าลงใน startup-config สำเร็จ

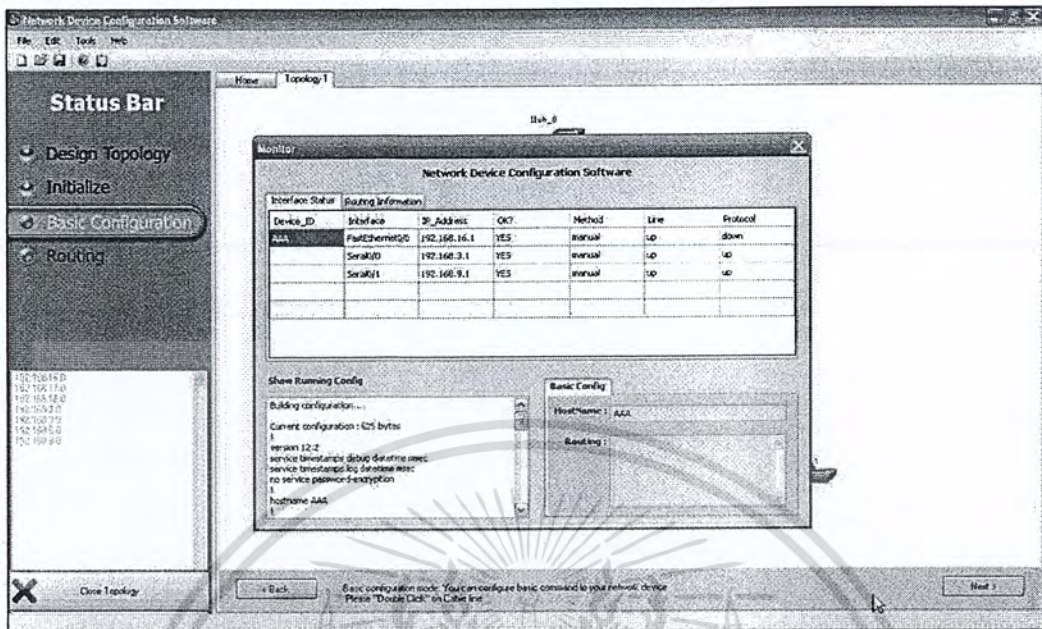
4.1.6 รูปแบบหน้าต่างทดสอบการติดต่อสื่อสารกันระหว่างสองเครือข่าย  
เมื่อผู้ใช้โปรแกรมได้ทำการเลือกโปรโตคอลในการเราท์ติ้งแล้ว จะทำให้อุปกรณ์  
เครือข่ายสามารถติดต่อสื่อสารกันได้ ซึ่งทดสอบโดยการ ping หากันเพื่อทดสอบดังรูป 4.25



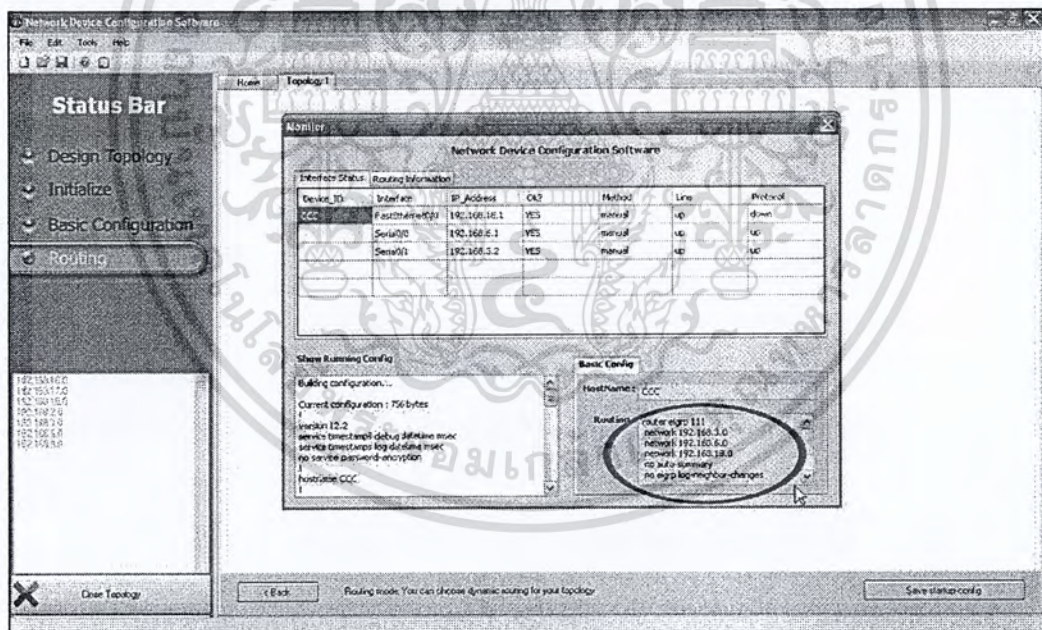
รูป 4.25 การทดสอบการเชื่อมต่อกันของเครือข่ายโดยการ ping

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่นิยญาติให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้





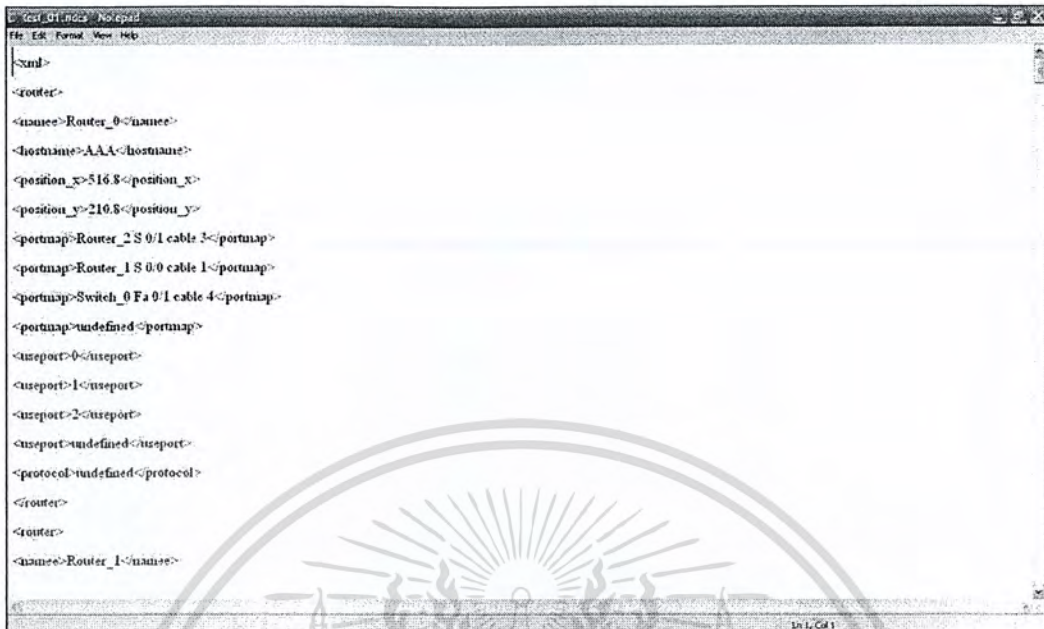
รูป 4.27 หน้าต่างแสดงสถานะของการทำงานของอุปกรณ์เราเตอร์



รูป 4.28 หน้าต่างแสดงสถานะเมื่อทำการเราเตอร์ที่ติดตั้งเสร็จสิ้น

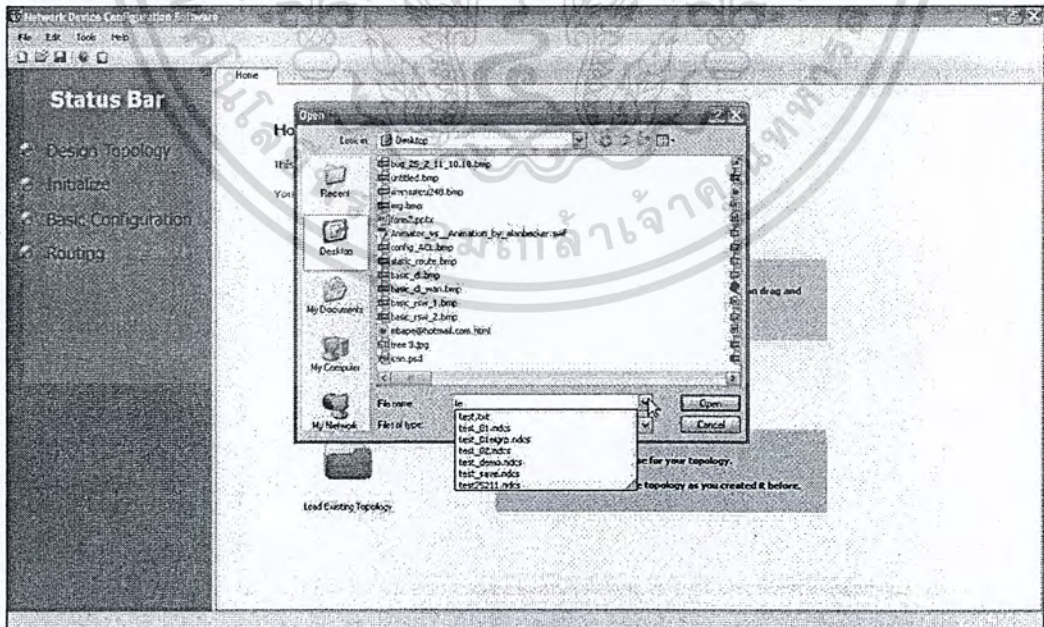
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้





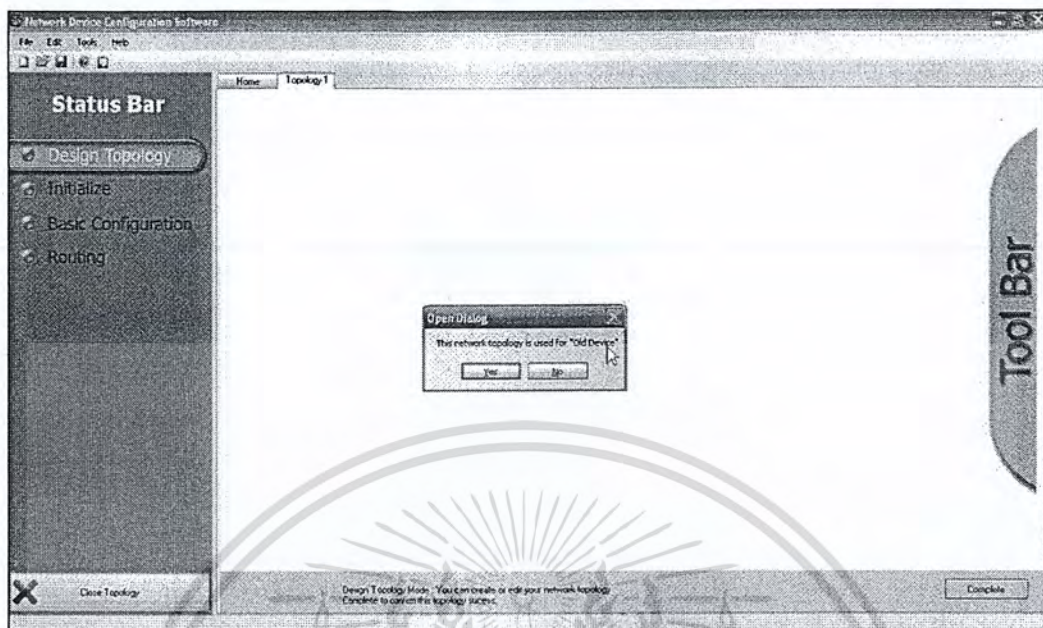
รูป 4.31 ไฟล์ที่ได้จากการบันทึกโดยผู้ใช้โปรแกรม

ผู้ใช้สามารถโหลดการตั้งค่าเครือข่ายขึ้นมาใช้ใหม่ได้ ในกรณีที่ผู้ใช้ได้ทำการบันทึกไว้แล้ว โดยค่าที่ได้มาจะเป็นค่าที่ได้ทำการตั้งค่าไปทั้งหมดก่อนทำการบันทึก เพื่อนำมาแก้ไขเปลี่ยนแปลงการตั้งค่าได้ดังรูป 4.32

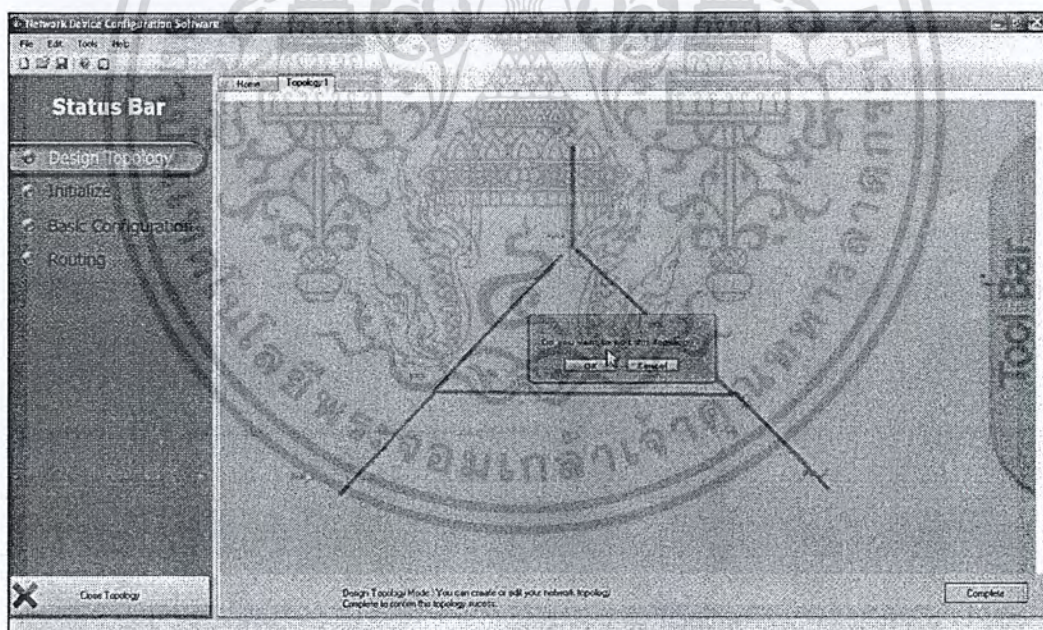


รูป 4.32 การโหลดไฟล์ที่ได้บันทึกการออกแบบและการตั้งค่าเครือข่าย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

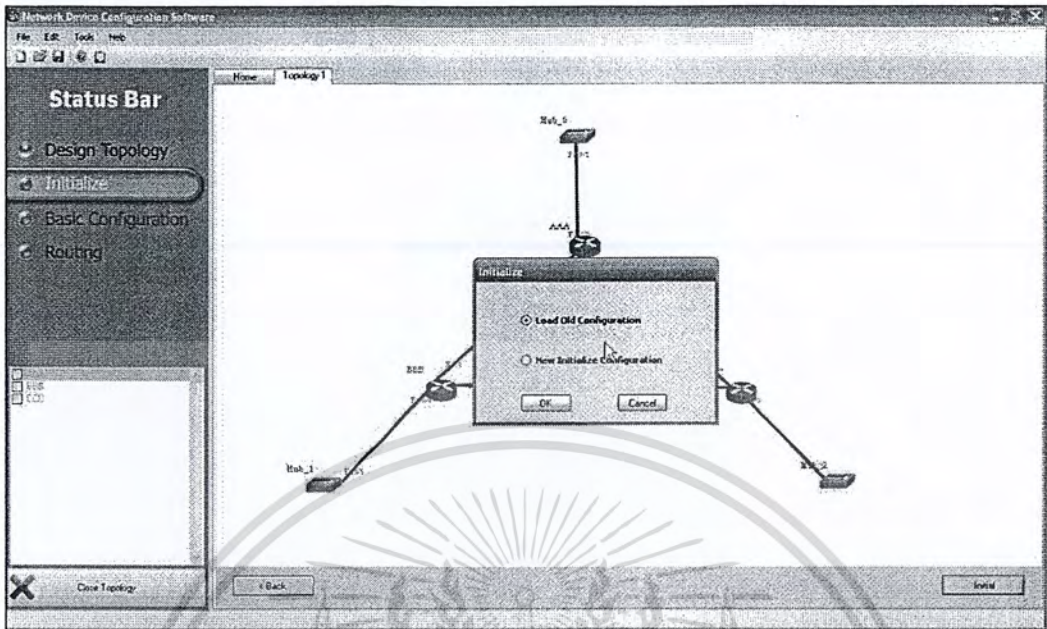


รูป 4.33 โปรแกรมสอบถามว่าต้องการใช้งานกับอุปกรณ์ที่มีการตั้งค่าอยู่ก่อนหรือไม่

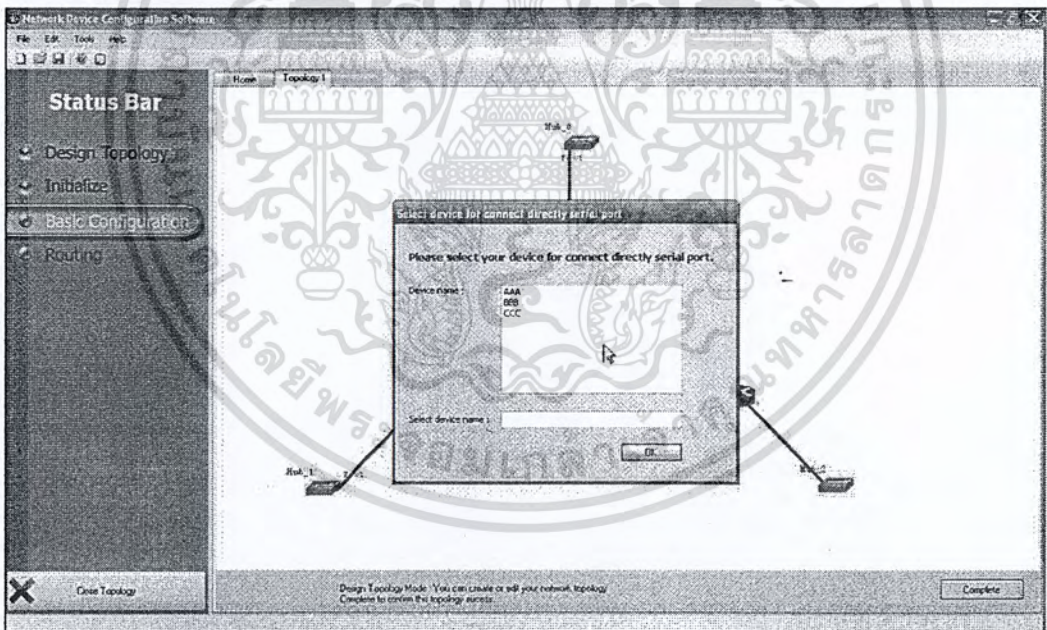


รูป 4.34 โปรแกรมทำการสอบถามผู้ใช้ว่าต้องการแก้ไขการออกแบบหรือไม่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูป 4.35 โปรแกรมสอบถามผู้ใช้ว่าต้องการตั้งค่าการทำงานเริ่มต้นด้วยค่าที่โหลดมาหรือไม่

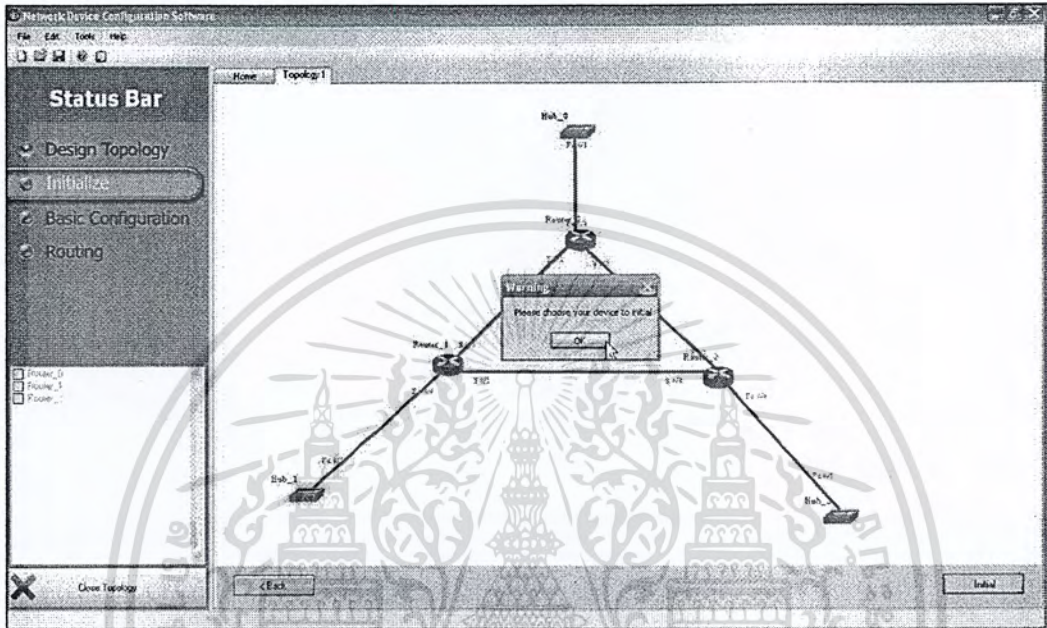


รูป 4.36 ผู้ใช้ทำการโหลดไฟล์เพื่อใช้กับอุปกรณ์เดิมและไม่แก้ไขเครือข่ายเดิม

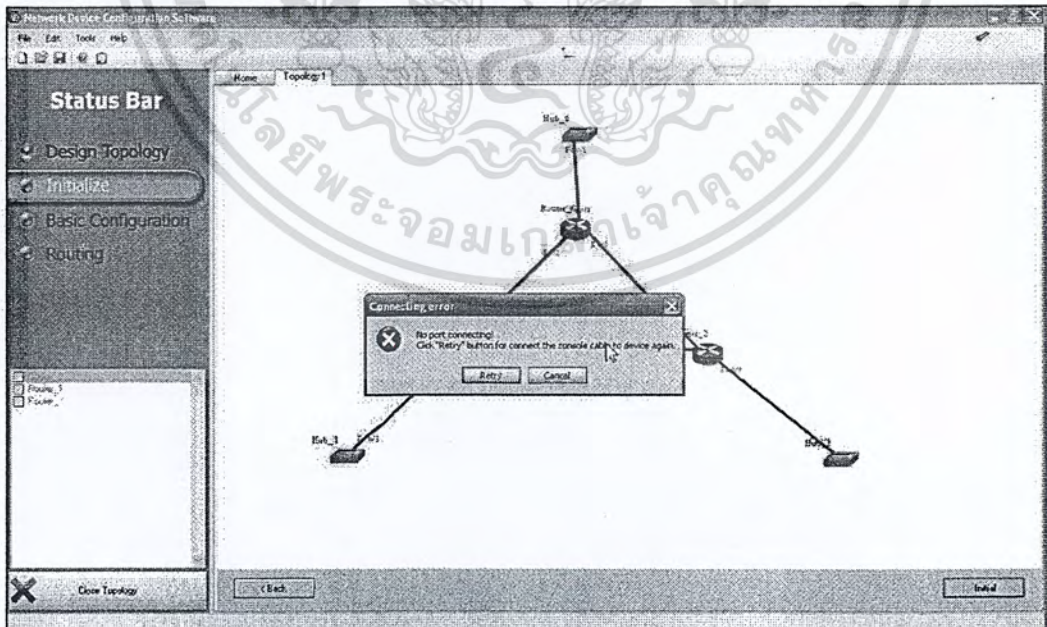
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

#### 4.1.9 รูปแบบการแจ้งเตือนต่างๆในโปรแกรม

ผู้ใช้จะได้รับการแจ้งเตือนเมื่อมีข้อผิดพลาดเกิดขึ้นในโปรแกรม ในกรณีต่างๆ ดังต่อไปนี้

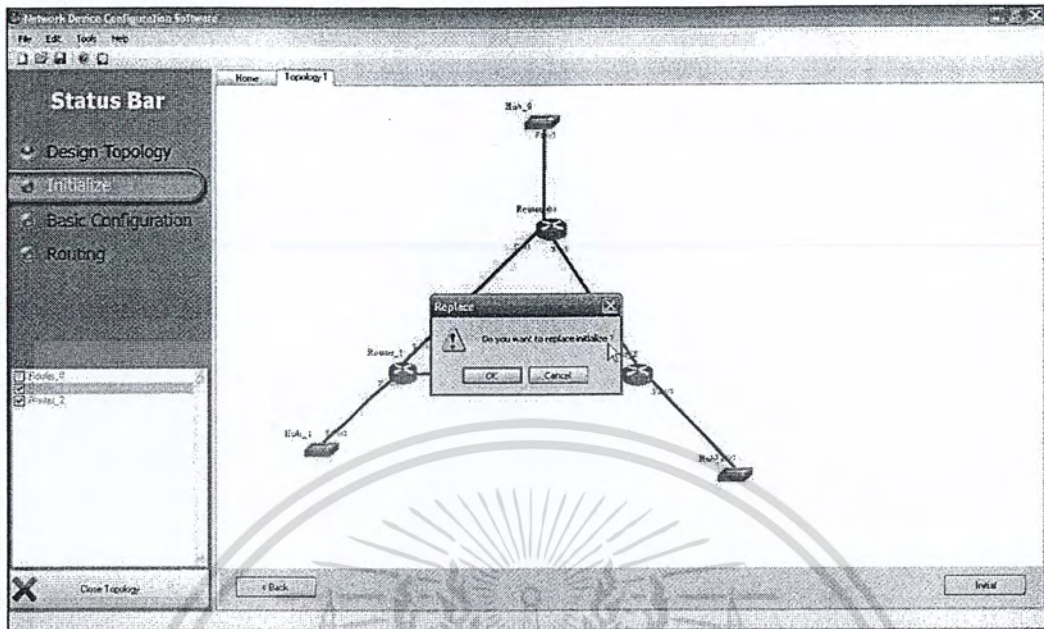


รูป 4.37 การแจ้งเตือนเมื่อผู้ใช้ไม่ได้ทำการเลือกอุปกรณ์ในการตั้งค่าเริ่มต้น

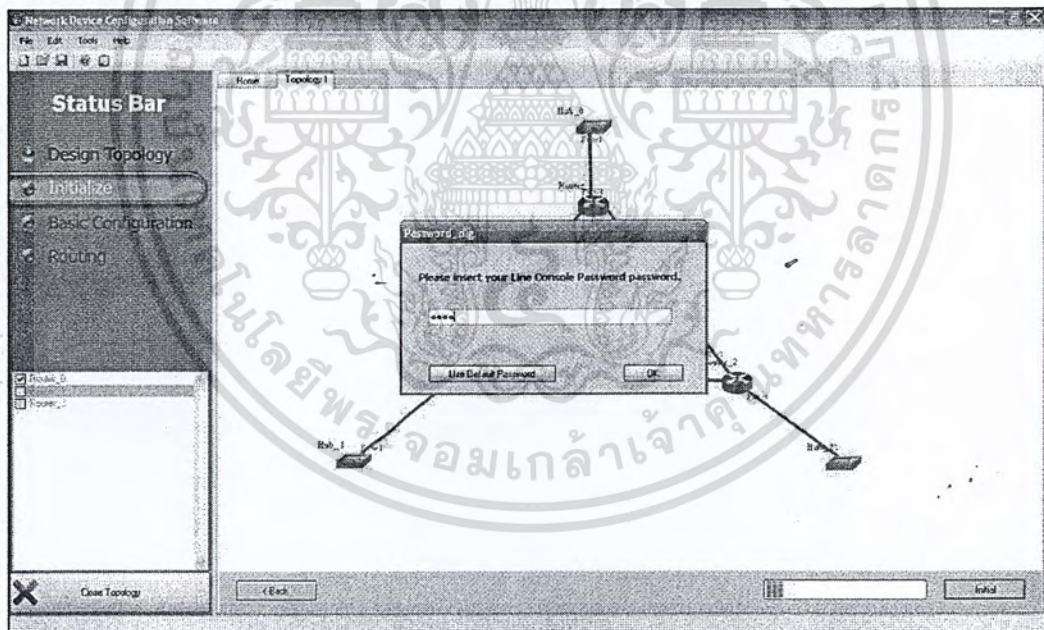


รูป 4.38 การแจ้งเตือนเมื่อขณะการตั้งค่าเริ่มต้นผู้ใช้ไม่ได้เสียบสายคอนโซล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

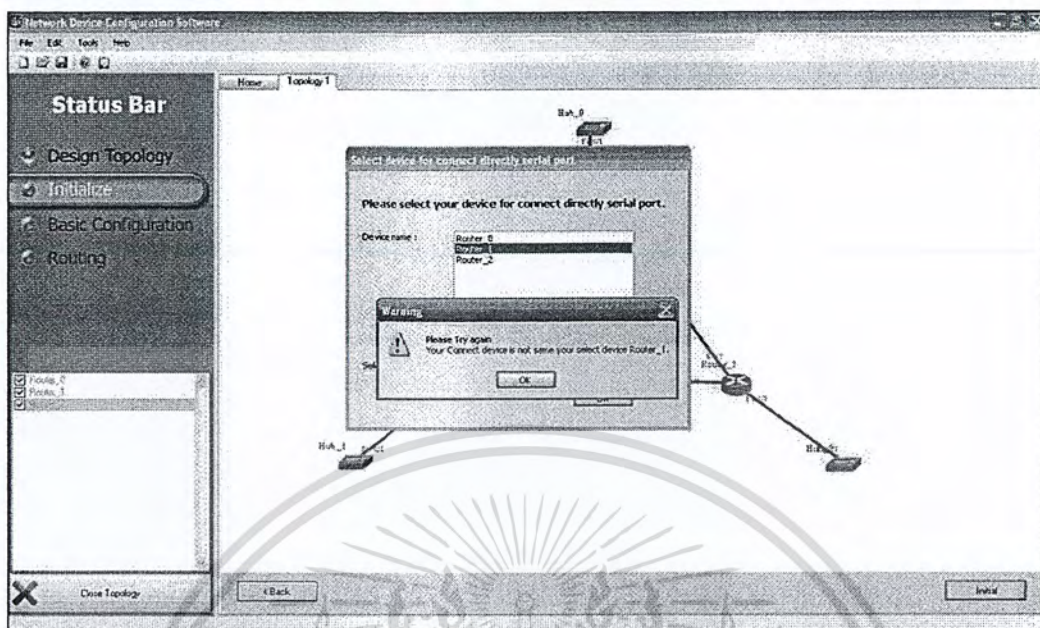


รูป 4.39 การแจ้งเตือนเมื่อขณะการตั้งค่าเริ่มต้นผู้ใช้เลือกตั้งค่าที่อุปกรณ์ตัวเดิมซ้ำ

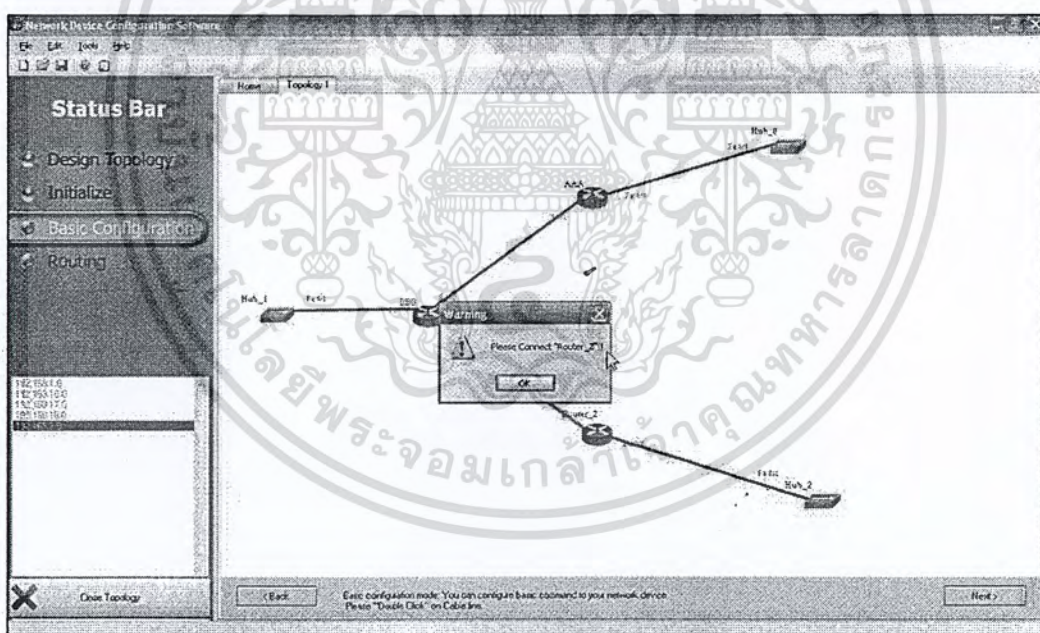


รูป 4.40 การแจ้งเตือนเมื่อขณะการตั้งค่าเริ่มต้นอุปกรณ์มีการตั้งค่าพาสเวิร์ดไว้ก่อนแล้ว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูป 4.41 การแจ้งเตือนเมื่อขณะการเลือกอุปกรณ์ตัวกลางไม่สัมพันธ์กับอุปกรณ์ที่เชื่อมต่อจริง



รูป 4.42 การแจ้งเตือนขณะทำการตั้งค่าพื้นฐานในเครือข่ายแบบกึ่งเพื่อให้การตั้งค่าพื้นฐานสมบูรณ์

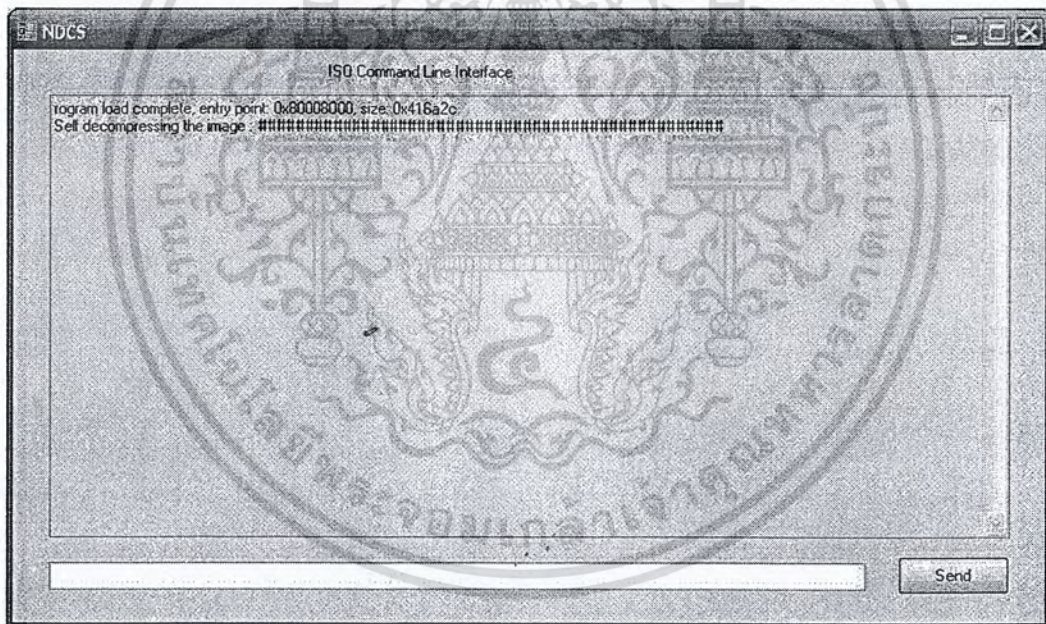
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## 4.2 การรับส่งข้อมูลผ่านพอร์ตอนุกรม

ในส่วนการทดลองนี้ถือได้ว่าเป็นแกนหลักของโครงการงานชิ้นนี้ เนื่องจากการทำงานต่างๆของโครงการงานนี้จะต้องรับส่งข้อมูลผ่านทางพอร์ตอนุกรมทั้งสิ้น ซึ่งถ้าการทดลองนี้ล้มเหลวโครงการงานนี้ จะไม่สามารถดำเนินการต่อไปได้ ในการทดลองนี้ได้ทำการเขียน โปรแกรมแอปพลิเคชันภาษา C# ขึ้นมา เพื่อใช้ในการทดลองรับส่งข้อมูลผ่านทางพอร์ตอนุกรม และทำการเชื่อมต่อพอร์ตอนุกรมกับเราเตอร์จริง ซึ่งได้ผลการทดลอง ดังนี้

### 4.2.1 การรับข้อมูลผ่านพอร์ตอนุกรม

เมื่อทำการเชื่อมต่อ โปรแกรมและเราเตอร์โดยใช้พอร์ตอนุกรม พบว่าตัวโปรแกรมสามารถรับค่าข้อมูลจากเราเตอร์ได้และรับข้อมูลมาแสดงยังตัวโปรแกรมได้ดังรูป 4.43 ซึ่งได้ข้อมูลครบทุกส่วน โดยเปรียบเทียบกับ โปรแกรมไฮเปอร์เทอร์มินอล (Hyper Terminal) ขณะที่เชื่อมต่อกับเราเตอร์ตัวเดียวกัน และรับข้อมูลเดียวกัน

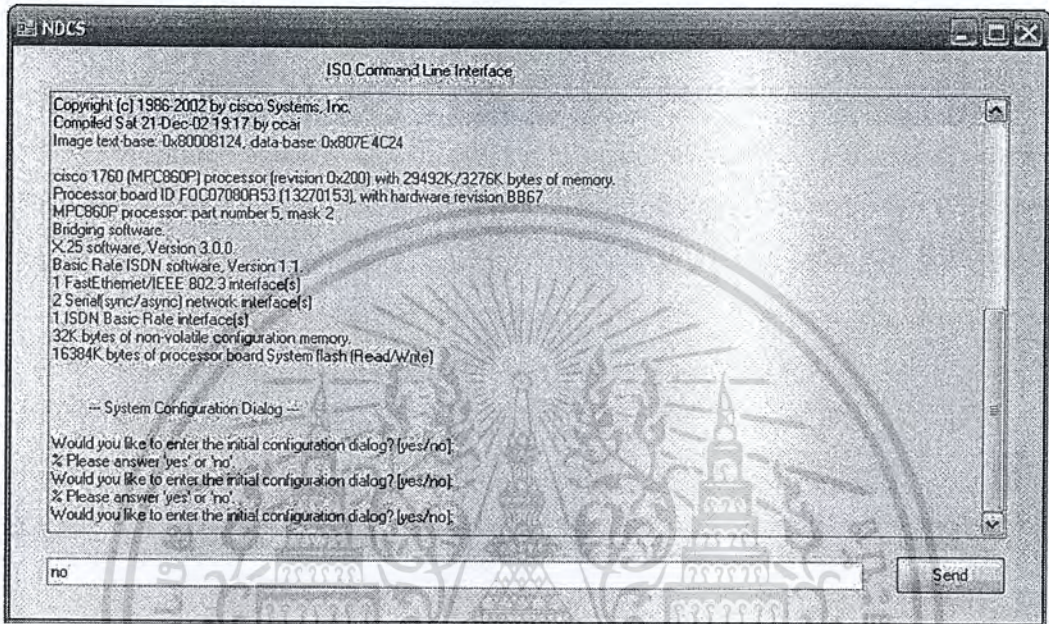


รูป 4.43 การทดลองรับค่าจากพอร์ตอนุกรม

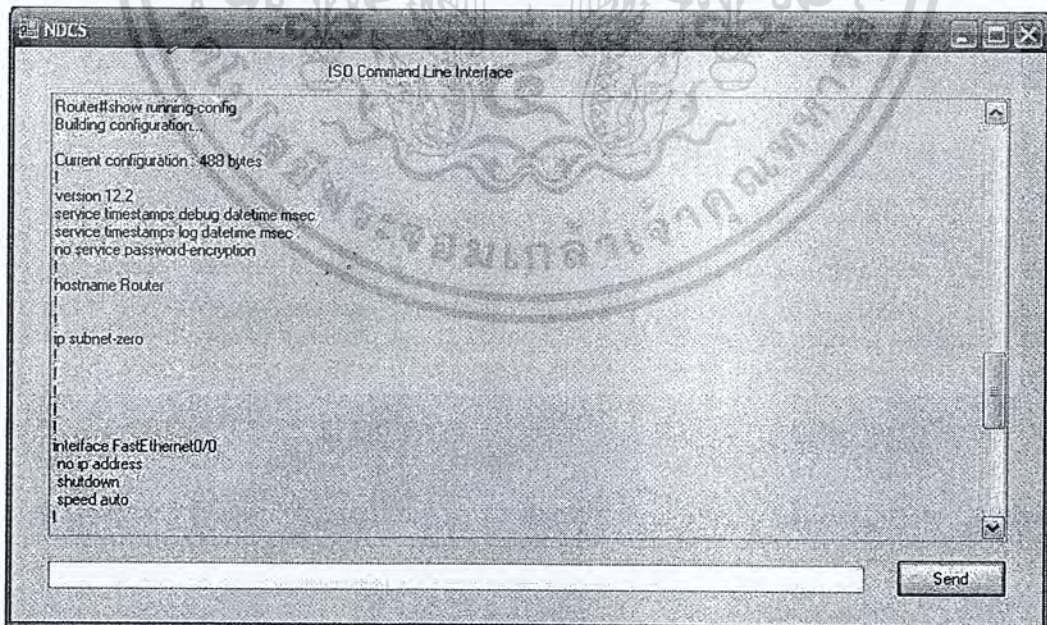
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

#### 4.2.2 การส่งข้อมูลผ่านพอร์ตอนุกรม

โปรแกรมนี้คือสามารถส่งข้อมูลต่างๆ ผ่านทางพอร์ตอนุกรมได้ โดยดูจากการที่เราที่  
เตอร์ส่งข้อมูลได้ตอบการทำงานกลับมายังตัวโปรแกรมได้อย่างครบถ้วนดังรูป 4.44 และ 4.45



รูป 4.44 การทดลองส่งค่าผ่านพอร์ตอนุกรม



รูป 4.45 การทดลองส่งคำสั่งผ่านพอร์ตอนุกรม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 4.3 การนำข้อมูลที่รับส่งผ่านพอร์ตอนุกรมเขียนลงไฟล์

การทดลองนี้เป็นส่วนที่จะนำไปใช้ได้กับตัวของโครงงาน ในด้านการแสดงข้อมูลต่างๆ ที่รับมาจากตัวอุปกรณ์เครือข่าย ซึ่งเราสามารถนำไฟล์นี้ไปแสดงผลให้อยู่ในรูปแบบที่ง่ายและ ผู้ใช้งานโปรแกรมเข้าใจได้มากขึ้น ซึ่งผลลัพธ์ที่ได้คือ สามารถทำการสร้างไฟล์เก็บข้อมูลที่รับและส่งผ่านทางพอร์ตอนุกรมได้ ดังรูป 4.46

```

C:\est1.r - Notepad
File Edit Format View Help
Router#
Router#sh ru
Building configuration...

Current configuration : 488 bytes
!
version 12.2
service timestamps debug datetime msec
service timestamps log datetime msec
no service password-encryption
!
hostname Router
!
!
ip subnet-zero
!
!
!
interface FastEthernet0/0
no ip address
  
```

รูป 4.46 การทดลองนำข้อมูลที่รับส่งผ่านพอร์ตอนุกรมเขียนลงไฟล์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## บทที่ 5

### บทสรุป

#### 5.1 บทสรุป

โปรแกรม Network Device Configuration Software สามารถออกแบบเครือข่ายโดยผู้ใช้ได้จากการวางอุปกรณ์เครือข่ายและเชื่อมต่อเข้าหากัน และทำการ Initial Configuration ในอุปกรณ์เครือข่ายแต่ละตัว ให้มีค่า IP Address, Router Name, Password ตั้งค่า Line vty เพื่อให้สามารถทำ Telnet ได้ โดยค่า IP Address ที่ให้ไปในตอนเริ่มต้นนั้น เป็นเพียงค่า IP Address ที่ทางโปรแกรมได้กำหนดขึ้นมาในตอนต้นเท่านั้น เพื่อให้อุปกรณ์เครือข่ายทุกตัวมีค่า IP Address เริ่มต้น ทำให้สามารถติดต่อถึงกันได้โดยผ่าน โปรโตคอล Telnet เพื่อให้สามารถส่งคำสั่งไปยังอุปกรณ์เครือข่ายทุกตัวได้ โดยผู้ใช้ทำการส่งคำสั่งผ่านอุปกรณ์เครือข่ายเพียงตัวเดียว ก็สามารถทำให้อุปกรณ์เครือข่ายตัวอื่นๆที่ต่ออยู่กับอุปกรณ์เครือข่ายของผู้ใช้ทำงานตามที่ต้องการได้ จากนั้นทำการตั้งค่า Basic Configuration โดยผู้ใช้จะสามารถกำหนดค่า IP Address, Router Name, Password ได้เองในอุปกรณ์เครือข่ายแต่ละตัว จากนั้นผู้ใช้จะทำการเลือกเรทติ้งที่โปรโตคอลเพื่อใช้ในการหาเส้นทางให้กับอุปกรณ์ในเครือข่าย ให้เลือกแบบ Dynamic เป็นต้น เมื่อผู้ใช้กำหนดค่าทุกอย่างตามต้องการแล้ว ก็กดปุ่ม Complete เป็นการเสร็จสิ้นการกำหนดค่าของผู้ใช้ จากนั้นตัวโปรแกรมจะทำการตั้งค่าต่างๆที่ผู้ใช้กำหนดให้กับตัวอุปกรณ์เครือข่ายเอง โดยที่ผู้ใช้แค่ทำการกำหนดค่าผ่านอุปกรณ์เครือข่ายเพียงตัวเดียว และในตัวโปรแกรมยังมีส่วนของการ Monitor เพื่อดูสถานะของการทำงานของอุปกรณ์เครือข่ายโดยสรุปอีกด้วย

#### 5.2 วิจารณ์สิ่งที่ได้จากโรงงาน

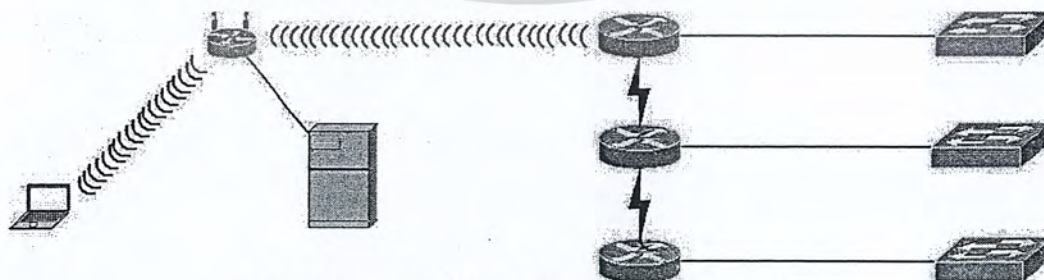
โปรแกรมที่พัฒนาขึ้นมาจะช่วยในการจัดการการตั้งค่าการใช้งานของอุปกรณ์เครือข่ายให้ง่ายขึ้น และทำให้ผู้ใช้ไม่สับสน เพราะหน้าต่างของซอฟต์แวร์เข้าใจได้ง่าย จะช่วยให้ผู้ใช้ที่มีความรู้ทางด้าน การตั้งค่าอุปกรณ์เครือข่ายเบื้องต้น สามารถใช้งานได้อย่างง่ายดาย รวมไปถึงผู้ที่มีความรู้ความเข้าใจอยู่แล้ว สามารถใช้งานซอฟต์แวร์ตัวนี้ เพื่อเพิ่มความสะดวกสบายให้แก่ผู้ใช้งานมากยิ่งขึ้น โดยตัวโปรแกรมจะมีส่วนที่ผู้ใช้สามารถออกแบบรูปร่างโครงข่ายที่ต้องการได้ และทางโปรแกรมจะมีส่วนตั้งค่าเริ่มต้นให้กับอุปกรณ์เครือข่ายด้วย ซึ่งโปรแกรมสามารถส่งต่อคำสั่งไปยังโครงข่ายของอุปกรณ์เครือข่ายที่ได้ติดตั้งเอาไว้ โดยต่อกับอุปกรณ์เครือข่ายเพียงแค่ตัวเดียว จากนั้นอุปกรณ์เครือข่ายจะติดต่อกันเองแล้วส่งคำสั่งไปจัดการให้เป็นไปตามที่ผู้ใช้ต้องการ

### 5.3 ปัญหา อุปสรรค และแนวทางการแก้ไข

ปัญหาเกิดขึ้นเมื่อมีการ Save ข้อมูลทั้งหมดที่ผู้ใช้ได้ตั้งค่าเอาไว้ จะมีค่าข้อมูลหลากหลายประเภทที่ต้องทำการ Save เก็บไว้ ไม่ว่าจะเป็นข้อมูลเครือข่ายจาก Flash หรือ ข้อมูลต่างๆที่ผู้ใช้ตั้งค่าเอาไว้ใน Form ของ C# ก็ตาม ทำให้การ Save ข้อมูลต้องมีการเก็บข้อมูลที่หลากหลายทำให้ต้องสร้างอัลกอริทึมเอาไว้สำหรับเก็บข้อมูลเมื่อผู้ใช้ทำการ Save โดยเฉพาะ และปัญหาที่เกิดจากความเร็วในการรับส่งข้อมูลไม่เท่ากัน แก้ไขได้โดยการตั้งค่า Time Out และขนาด Buffer Size ของการรับส่งข้อมูลผ่านพอร์ตอนุกรม

### 5.4 แนวทางการพัฒนาต่อ

1. พัฒนาด้านความปลอดภัยของข้อมูล ให้มีการเข้ารหัสข้อมูลที่รับและส่งข้อมูลระหว่างโปรแกรมกับตัวอุปกรณ์เครือข่าย
2. พัฒนาด้านความหลากหลายของรุ่นอุปกรณ์ที่ใช้ ให้มีการเพิ่มชุดรูปแบบคำสั่งให้รองรับกับ IOS version อื่นๆ
3. พัฒนาด้านกราฟฟิก ให้มีรูปแบบที่สวยงามและเข้าใจง่ายมากยิ่งขึ้น
4. เพิ่มความสามารถในการตั้งค่าเพิ่มเติมให้มากขึ้น ให้โปรแกรมสามารถทำการตั้งค่าเพิ่มเติมในส่วนของการทำ NAT , DHCP เป็นต้น
5. พัฒนาด้านความสามารถของตัวโปรแกรม โปรแกรมสามารถจำลองเครือข่ายขึ้นมาได้ โดยรับข้อมูลการเชื่อมต่อจากเครือข่ายจริง แล้วนำมาแสดงในรูปแบบที่เข้าใจง่ายยิ่งขึ้น
6. พัฒนาอัลกอริทึมในการตั้งค่าเริ่มต้น ให้มีความเป็นปัญญาประดิษฐ์มากยิ่งขึ้น
7. พัฒนาด้านประสิทธิภาพให้สามารถใช้งานได้สะดวกขึ้น โดยให้มีการสร้าง Server กลางที่มีโปรแกรมตัวนี้ทำงานอยู่ เพื่อให้คอมพิวเตอร์ตัวอื่นๆ สามารถเข้ามาตั้งค่าอุปกรณ์เครือข่ายได้ผ่านทางหน้า web interface



รูป 5.1 ลักษณะการใช้งานโปรแกรมในการพัฒนาต่อ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## บรรณานุกรม

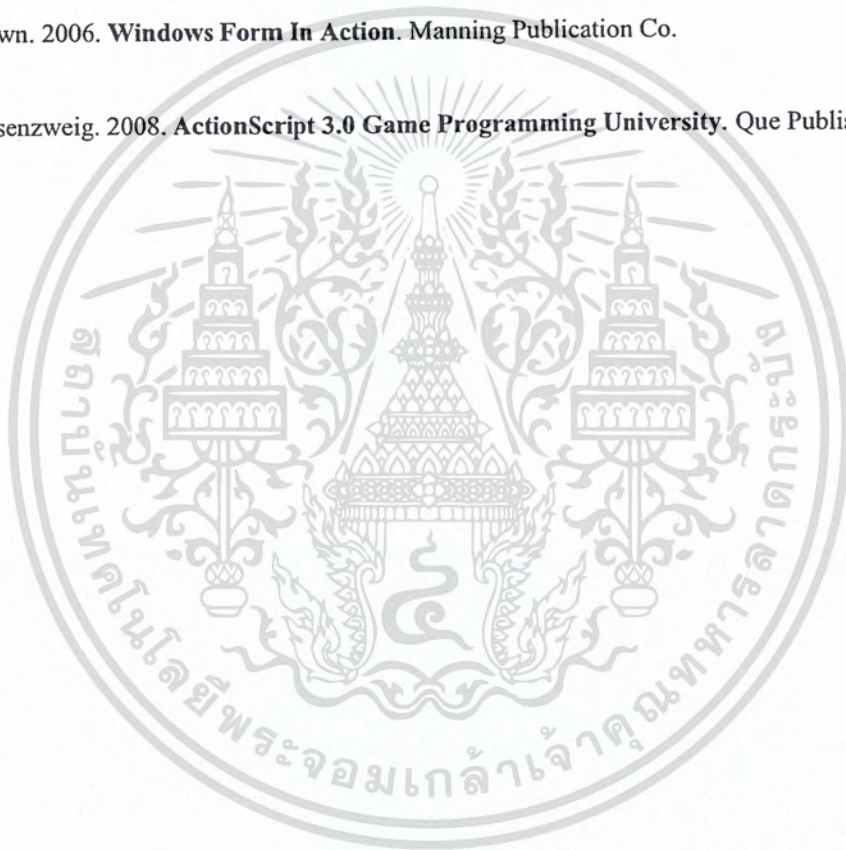
Microsoft Corporation. 2010. **MSDN Library**. [Online].

Available : <http://msdn.microsoft.com/en-gb/library/ms123401.aspx>

Jesse Liberty. 2002. **Programming C#2<sup>nd</sup> Edition**. O'Reilly

Erik Brown. 2006. **Windows Form In Action**. Manning Publication Co.

Gary Rosenzweig. 2008. **ActionScript 3.0 Game Programming University**. Que Publishing



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้