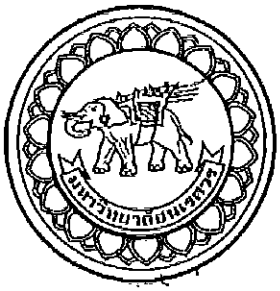


ระบบควบคุมหุ่นยนต์ระยะไกลผ่านเครือข่ายด้วยโปรโตคอลทีซีพี/ไอพี

TCP/IP PROTOCAL BASED TELE-ROBOT CONTROL SYSTEM

นายสุเมธ สมมะลวน รหัส 45360534

นางสาว เศกศิริ เรืองวาณิชยกุล รหัส 45360674

1586923

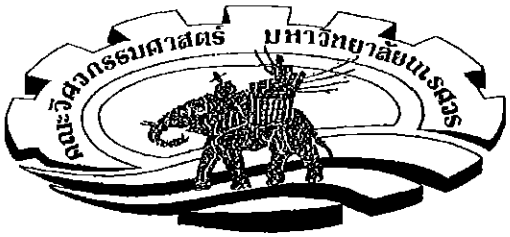
ห้องสมุดคณะวิศวกรรมศาสตร์
วันที่รับ 17 ม.ค. 2549
เลขทะเบียน 4900003
เลขเรียกหนังสือ ปร.
มหาวิทยาลัยนเรศวร ส8 435
2548

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต

สาขาวิชาวิศวกรรมคอมพิวเตอร์ ภาควิชาวิศวกรรมไฟฟ้าและคอมพิวเตอร์

คณะวิศวกรรมศาสตร์ มหาวิทยาลัยนเรศวร

ปีการศึกษา 2548



ใบรับรองโครงงานวิศวกรรม

หัวข้อโครงงาน	ระบบควบคุมหุ่นยนต์ระยะไกลผ่านเครือข่ายด้วยโปรโตคอลทีซีพี/ไอพี		
ผู้ดำเนินโครงงาน	นายสุเมธ	สมมะลวน	รหัส 45360534
	นางสาวเศกศิริ	เรืองวาณิชยกุล	รหัส 45360674
อาจารย์ที่ปรึกษา	ดร. สมยศ	เกียรติวนิชวิไล	
สาขาวิชา	วิศวกรรมคอมพิวเตอร์		
ภาควิชา	วิศวกรรมไฟฟ้าและคอมพิวเตอร์		
ปีการศึกษา	2548		

.....

คณะวิศวกรรมศาสตร์ มหาวิทยาลัยนเรศวร อนุมัติให้โครงงานฉบับนี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรวิศวกรรมศาสตรบัณฑิต สาขาวิชาวิศวกรรมคอมพิวเตอร์

คณะกรรมการการสอบโครงงานวิศวกรรม

.....ประธานกรรมการ
(ดร. สมยศ เกียรติวนิชวิไล)

.....กรรมการ
(ดร. สุรเชษฐ์ กานต์ประชา)

.....กรรมการ
(ดร. พนมขวัญ ริยะมงคล)

หัวข้อโครงงาน	ระบบควบคุมหุ่นยนต์ระยะไกลผ่านเครือข่ายด้วยโปรโตคอลทีซีพี/ไอพี		
ผู้ดำเนินโครงงาน	นายสุเมธ	สมมะลวน	รหัส 45360534
	นางสาวเศกศิริ	เรืองวาณิชยกุล	รหัส 45360674
อาจารย์ที่ปรึกษา	ดร.สมยศ	เกียรติวนิชวิไล	
สาขาวิชา	วิศวกรรมคอมพิวเตอร์		
ภาควิชา	วิศวกรรมไฟฟ้าและคอมพิวเตอร์		
ปีการศึกษา	2548		

บทคัดย่อ

โครงงานนี้ศึกษาและพัฒนาโปรแกรมสำหรับระบบควบคุมหุ่นยนต์ระยะไกลผ่านเครือข่ายด้วยโปรโตคอลทีซีพี/ไอพี จุดประสงค์หลักของโครงงานนี้คือการเพิ่มขีดความสามารถในการควบคุมสั่งการหุ่นยนต์ ให้สามารถสั่งงานระยะไกลได้ โปรแกรมนี้ถูกพัฒนาให้สามารถนำไปเชื่อมต่อใช้กับเครื่องมืออุปกรณ์อื่นๆอาทิ สามารถเห็นการทำงานของหุ่นยนต์ผ่านทางกล้อง (webcam) เป็นต้น ระบบที่ออกแบบขึ้นนี้พัฒนาบนภาษาวิชวลเบสิก 6.0 โดยใช้เทคโนโลยีของ Winsock ซึ่งเป็นเทคโนโลยีพื้นฐานในการติดต่อสื่อสารผ่านทางเครือข่ายด้วยโปรโตคอลทีซีพี/ไอพี ผู้ใช้สามารถความคุมและเห็นการทำงานของอุปกรณ์ที่ควบคุมจากระยะไกลได้โดยผ่านทางทางกล้อง (webcam) และระบบเครือข่ายด้วยโปรโตคอลทีซีพี/ไอพี

Project Title	TCP/ IP Protocol Based Tele-Robot Control System		
Name	Mr. Sumet	Sommaluan	ID. 45360534
	Miss Seksiri	Ruangwanichayakun	ID. 45360674
Project Advisor	Dr. Somyot	Kiattivanichvilai	
Major	Computer Engineering		
Department	Electrical and Computer Engineering		
Academic Year	2005		

Abstract

This project studies and developes a program for TCP/ IP Protocol based Tele-Robot Control System. The main purpose of this project is to enhance the ability of the robot control system to command a robot from the remote area. This developed program is able to connect to any device such as CCD camera, etc. Visual Basic 6.0 is used as the development tools and "Winsock" in this program is used as a component for TCP/IP communication. The remoted user is able to control and view the robot via a CCD camera and TCP/IP protocol.

กิตติกรรมประกาศ

ขอขอบพระคุณทุกๆท่านที่ทำให้โครงงานชิ้นนี้ สำเร็จลุล่วง โดยเฉพาะอย่างยิ่ง อาจารย์ที่ปรึกษาโครงงาน ดร. สมยศ เกียรติวนิชวิไล ที่ได้สละเวลาให้คำปรึกษาและเสนอแนะแนวทางการสำหรับแก้ปัญหาที่เกิดขึ้นขณะที่กำลังดำเนินโครงงาน พร้อมด้วยคณะกรรมกรรมการที่ปรึกษาโครงงาน ดร. สุรเชษฐ์ กานต์ประชา และดร. พนมขวัญ ริยะมงคล ที่ให้คำแนะนำในการดำเนินโครงงานและให้คำปรึกษาในการเขียนรูปเล่มรายงาน และกลุ่มของคุณวุฒิพงษ์ หยองใหญ่ คุณอนรรฆพล แสนทน และคุณสุภาพร ไชยชาญที่ได้ให้ความอนุเคราะห์ หุ่นยนต์เพื่อมาใช้ในการทดสอบการทำงานของโปรแกรม และเพื่อนๆ ที่ร่วมแรงร่วมใจกันทำงานจนบรรลุเป้าหมายในระดับหนึ่ง

สุเมธ สมมะลวน
เศกศิริ เรืองวาณิชยกุล

สารบัญ

หน้า

หน้าบทคัดย่อภาษาไทย ..... ก

บทคัดย่อภาษาอังกฤษ ..... ข

กิตติกรรมประกาศ ..... ค

สารบัญ ..... ง

สารบัญตาราง ..... ฉ

สารบัญรูป ..... ช

บทที่ 1 บทนำ

1.1 ที่มาและความสำคัญของโครงงาน 1

1.2 วัตถุประสงค์โครงงาน 1

1.3 เป้าหมายและขอบเขตของโครงงาน 1

1.4 ขั้นตอนวิธีการดำเนินงาน 1

1.5 แผนการดำเนินงาน 2

1.6 ผลที่คาดว่าจะได้รับ 2

1.7 งบประมาณที่ใช้ 2

บทที่ 2 หลักการและทฤษฎี

2.1 โปรโตคอลที่ใช้ส่งข้อมูลผ่านเครือข่าย (Network) 3

2.2 ความรู้ทั่วไปเกี่ยวกับ Port 22

2.3 ระบบ Client – Server 24

2.4 พื้นฐานในการรับส่งข้อมูล 25

2.5 การเขียนโปรแกรมบนอินเตอร์เน็ตด้วย Winsock (Windows Socket) 28

2.6 ภาพดิจิตอล 32

2.7 พื้นฐานและระบบสีโมเดล RGB 33

2.8 การสื่อสารข้อมูลแบบไร้สาย 35

สารบัญ(ต่อ)

หน้า

บทที่ 3 การดำเนินงาน

3.1 การควบคุมหุ่นยนต์โดยควบคุมและแสดงภาพการทำงานผ่านทางเครือข่ายด้วยโปรโตคอล ทีซีพี/ไอพี (TPC/IP Protocol) 42

3.2 แผนภาพแสดงการควบคุมหุ่นยนต์ผ่านเครือข่ายด้วยโปรโตคอลทีซีพี/ไอพี (TPC/IP Protocol) 43

3.3 การออกแบบโปรแกรมควบคุมหุ่นยนต์ผ่านเครือข่ายด้วยโปรโตคอลทีซีพี/ไอพี (TPC/IP Protocol) 45

3.4 การเขียนโปรแกรมเพื่อควบคุมการทำงานของหุ่นยนต์ผ่านเครือข่ายด้วยโปรโตคอลทีซีพี/ไอพี (TPC/IP Protocol) 48

3.5 การเขียนโปรแกรมเพื่อการแสดงผลภาพการทำงานของหุ่นยนต์ 49

บทที่ 4 ผลการทดลอง

4.1 จุดประสงค์การทดลอง 50

4.2 ขั้นตอนการทดลอง 50

4.3 ผลการทดลอง 51

บทที่ 5 สรุปผลการทดลองและข้อเสนอแนะ

5.1 สรุปผลการทดลอง 63

5.2 ปัญหาและแนวทางแก้ไข 63

5.3 แนวทางการพัฒนาต่อ 64

เอกสารอ้างอิง 65

ภาคผนวก ก 66

ประวัติผู้เขียนโครงงาน 72

สารบัญตาราง

ตารางที่ หน้า

1.1 ขั้นตอนการดำเนินงาน 2

2.1 แสดงจำนวนสีที่เป็นไปได้ซึ่งขึ้นกับจำนวนบิต 34

2.2 ช่วงความถี่ของสัญญาณวิทยุ 35

2.3 ชนิดของการสื่อสารด้วยคลื่นวิทยุ 36

มหาวิทยาลัยนเรศวร

สารบัญรูป

รูปที่ หน้า

2.1 เลเยอร์ของ TCP/IP 3

2.2 โปรโตคอลและเน็ตเวิร์ก ในโมเดล TCP/IP 4

2.3 การส่งข้อมูลใน TCP/IP model 5

2.4 ความสัมพันธ์ของโปรโตคอล TCP/IP กับ Application 6

2.5 TCP Segment Format 7

2.6 การเริ่มต้นการเชื่อมต่อของ TCP 9

2.7 ขั้นตอนการถ่ายโอนข้อมูล 10

2.8 การถ่ายโอนเซกเมนต์ทั้งสองทิศทาง 10

2.9 การปิดการเชื่อมต่อ 11

2.10 การเลื่อนหน้าต่าง (Sliding window) 12

2.11 การเลื่อนหน้าต่างส่ง 12

2.12 การเลื่อนหน้าต่างฝ่ายรับ 13

2.13 การประกาศขนาดหน้าต่าง 14

2.14 แพ็คเก็ต UDP 15

2.15 การทำแฟร็กเมนเตชั่น (Fragmentation) 15

2.16 การรีแอสเซมเบิล (Reassemble) 16

2.17 แพ็คเก็ต IP 16

2.18 ARP Datagram 20

2.19 รูปแบบของ ICMP 21

2.20 การแจกจ่าย Segment แบบ Demultiplexing 23

2.21 header ของ port 23

2.22 การใช้ Service Telnet จากเครื่องปลายทางเครื่องเดียวกัน 24

2.23 การแยกซอฟต์แวร์ส่วน Client และ server 25

2.24 การให้บริการแบบ Client/server 26

2.25 การรับส่งข้อมูลแบบขนาน 26

2.26 การเชื่อมต่อกันระหว่างอุปกรณ์ DTE กับ DCE 27

2.27 การเชื่อมต่อกันระหว่างอุปกรณ์ DTE กับ DTE 27

สารบัญรูป (ต่อ)

รูปที่ หน้า

2.28 การสื่อสารผ่าน Socket 29
2.29 ระดับของ Winsock 29
2.30 การสร้าง Client/Server โดยใช้ฟังก์ชั่นของ Winsock 30
2.31 โมเดล RGB 33
2.32 โหมดสี RGB 34
2.33 บล็อกไดอะแกรมแสดงหลักการทำงานภาคส่งของระบบคลื่นวิทยุ 37
2.34 บล็อกไดอะแกรมแสดงหลักการทำงานภาครับของระบบคลื่นวิทยุ 37
2.35 บล็อกไดอะแกรมแสดงรูปแบบการส่งสัญญาณแบบพัลส์ 38
2.36 แสดงหลักการทำงานของระบบคลื่นวิทยุผ่านทางตัวกลางที่เป็นอากาศ 38
2.37 การใช้สัญญาณพัลส์กระตุ้นให้อุปกรณ์ควบคุมสวิตซ์ทำงาน 39
2.38 วงจรกำหนดความกว้างของพัลส์ 39
2.39 วงจรผลิตความถี่ 40
2.40 วงจรส่งคลื่นวิทยุที่มีทั้งวงจรผลิตความถี่และวงจรกำหนดความกว้างสัญญาณ 40
2.41 วงจรแปลงสัญญาณคลื่นวิทยุ 41
3.1 แสดงการทำงานการควบคุมหุ่นยนต์ 42
3.2 แผนภาพแสดงการทำงานของโปรแกรมส่วนที่เป็นแม่ข่าย 43
3.3 แผนภาพแสดงการทำงานของโปรแกรมส่วนที่เป็นลูกข่าย 44
3.4 แผนภาพแสดงการทำงานของโปรแกรมส่วนที่แสดงผลภาพ 45
3.5 หน้าโปรแกรมส่วนทีทำหน้าที่เป็นแม่ข่าย 46
3.6 หน้าตาของโปรแกรมส่วนทีทำหน้าที่เป็นลูกข่าย 47
3.7 เหตุการณ์ที่เกิดขึ้นเมื่อทำการติดต่อไปที่เครื่องแม่ข่าย 49
4.1 โปรแกรมเครื่องแม่ข่ายเมื่อเริ่มใช้งาน 51
4.2 โปรแกรมเครื่องลูกข่ายเมื่อเริ่มใช้งาน 51
4.3 โปรแกรมเครื่องแม่ข่ายเมื่อรอการร้องขอการติดต่อจากเครื่องลูกข่าย 52
4.4 โปรแกรมเครื่องลูกข่ายเมื่อทำการเชื่อมต่อกับเครื่องแม่ข่ายได้สำเร็จ 52

สารบัญรูป (ต่อ)

รูปที่ หน้า

4.5 โปรแกรมเครื่องแม่ข่ายเมื่อทำการเชื่อมต่อกับเครื่องลูกข่าย 53

4.6 โปรแกรมเครื่องลูกข่ายเมื่อทำการเชื่อมต่อกับเครื่องแม่ข่ายไม่สำเร็จ 53

4.7 โปรแกรมเครื่องลูกข่ายเมื่อทำการรับภาพมาแสดงที่เครื่อง 54

4.8 (ก) โปรแกรมเครื่องลูกข่ายก่อนส่งคำสั่งเดินหน้า 54

4.8 (ข) โปรแกรมเครื่องลูกข่ายเมื่อส่งคำสั่งเดินหน้า 55

4.9 โปรแกรมเครื่องแม่ข่ายเมื่อรับคำสั่งเดินหน้า 55

4.10 (ก) โปรแกรมเครื่องลูกข่ายก่อนส่งคำสั่งถอยหลัง 56

4.10 (ข) โปรแกรมเครื่องลูกข่ายหลังส่งคำสั่งถอยหลัง 56

4.11 โปรแกรมเครื่องแม่ข่ายเมื่อส่งคำสั่งถอยหลัง 57

4.12 (ก) โปรแกรมเครื่องลูกข่ายก่อนส่งคำสั่งเลี้ยวขวา 58

4.12 (ข) โปรแกรมเครื่องลูกข่ายหลังส่งคำสั่งเลี้ยวขวา 58

4.13 โปรแกรมเครื่องแม่ข่ายเมื่อรับคำสั่งเลี้ยวขวา 58

4.14 (ก) โปรแกรมเครื่องลูกข่ายก่อนส่งคำสั่งเลี้ยวซ้าย 59

4.14 (ข) โปรแกรมเครื่องลูกข่ายหลังส่งคำสั่งเลี้ยวซ้าย 59

4.15 โปรแกรมเครื่องแม่ข่ายเมื่อรับคำสั่งเลี้ยวขวา 60

4.16 โปรแกรมเครื่องแม่ข่ายเมื่อรับคำสั่งหยุดการทำงาน 60

4.17 โปรแกรมเครื่องลูกข่ายเมื่อส่งคำสั่งหยุดการทำงาน 61

4.18 โปรแกรมเครื่องลูกข่ายเมื่อยกเลิกการเชื่อมต่อ 61

4.19 สถานะของการเชื่อมต่อที่โปรแกรมเครื่องแม่ข่าย 62

4.20 สถานะของโปรแกรมที่เครื่องลูกข่ายขณะหลุดจากการติดต่อ 62

บทที่1
บทนำ

1.1 ความเป็นมาและความสำคัญของโครงงาน

ในโลกปัจจุบันมีความก้าวหน้าอย่างมากโดยเฉพาะการพัฒนาทางด้านเทคโนโลยีอิเล็กทรอนิกส์ มีการประดิษฐ์หุ่นยนต์มากมายหลายชนิดขึ้นอยู่กับจุดประสงค์ของการใช้งาน เช่น หุ่นยนต์ที่เป็นของเล่น หุ่นยนต์ใช้ในงานการแพทย์ หุ่นยนต์เพื่องานวิจัย หุ่นยนต์สำรวจ และหุ่นยนต์อุตสาหกรรมฯลฯ เมื่อมีการประดิษฐ์หุ่นยนต์หรือเครื่องจักรต่างๆ ขึ้นมามากมายจึงเป็นที่จะต้องมีการควบคุมสิ่งประดิษฐ์เหล่านั้นให้ทำตามคำสั่งที่ต้องการ การที่จะติดต่อสื่อสารเพื่อสั่งงานกับหุ่นยนต์จึงมีความสำคัญ แนวทางหนึ่งในการพัฒนาการควบคุมหุ่นยนต์คือ การสั่งงานหุ่นยนต์จากระยะไกลที่มีความสะดวกรวดเร็วและปลอดภัย ซึ่งในโครงงานนี้จะพัฒนาระบบสั่งงานระยะไกลผ่านเครือข่ายด้วยโปรโตคอล ทีซีพี/ไอพี (TPC/IP Protocol) ทำให้การสั่งงานหุ่นยนต์เกิดความสะดวกต่อการทำงาน สามารถสั่งงานอุปกรณ์อิเล็กทรอนิกส์และหุ่นยนต์จากระยะไกลได้ นอกจากนี้ยังสามารถนำไปเชื่อมต่อใช้ในการควบคุมอุปกรณ์อื่นได้

1.2 วัตถุประสงค์โครงงาน

1. เพื่อศึกษาการติดต่อสื่อสารระหว่างผู้ใช้ไปยังหุ่นยนต์ว่ามีการทำงานอย่างไร โดยผ่านทางเครือข่ายด้วยโปรโตคอลทีซีพี/ไอพี (TPC/IP Protocol)
2. สามารถนำโปรแกรมไปใช้เพื่อติดต่อสั่งการกับหุ่นยนต์เหล่านั้นได้

1.3 เป้าหมายและขอบเขตของโครงงาน

1. สามารถติดต่อสื่อสารกับหุ่นยนต์ได้โดยผ่านทางเครือข่ายด้วยโปรโตคอลทีซีพี/ไอพี (TPC/IP Protocol)
2. สั่งงานหุ่นยนต์ผ่านทางเครือข่ายด้วยโปรโตคอลทีซีพี/ไอพี (TPC/IP Protocol) และแสดงผลภาพการทำงานของหุ่นยนต์ได้

1.4 ขั้นตอนวิธีการดำเนินงาน

1. ค้นคว้าและรวบรวมข้อมูลที่เกี่ยวกับการสื่อสารกับหุ่นยนต์
2. ศึกษาข้อมูลที่รวบรวมไว้เกี่ยวกับการสื่อสารกับหุ่นยนต์ รวมทั้งถึงภาษาที่ใช้เขียนโปรแกรม

3. เขียนโปรแกรม ใช้สำหรับสื่อสารกับหุ่นยนต์
4. ทดสอบโปรแกรมและทดลองใช้งาน
5. สรุปผลการทดสอบโปรแกรม

1.5 แผนการดำเนินงาน

ตารางที่ 1.1 ขั้นตอนการดำเนินโครงงาน

กิจกรรม	เดือนปี				
	มิ.ย. 48	ก.ค. 48	ส.ค. 48	ก.ย. 48	ต.ค. 48
1. ค้นคว้าและรวบรวมข้อมูล	→				
2. ศึกษาการข้อมูลที่รวบรวมไว้		→			
3. เขียนโปรแกรม			→		
4. ทดสอบและใช้งาน				→	
5. สรุปผล					→

1.6 ผลที่คาดว่าจะได้รับ

1. มีความรู้ความเข้าใจในการติดต่อสื่อสารผ่านเครือข่ายด้วยโปรโตคอลทีซีพี/ไอพี (TPC/IP Protocol)
2. มีความรู้เรื่องการเขียนโปรแกรมเกี่ยวกับการสื่อสารผ่านทางเครือข่ายด้วยโปรโตคอลทีซีพี/ไอพี (TPC/IP Protocol)

1.7 งบประมาณที่ใช้

1. ค่าจัดซื้อวัสดุอุปกรณ์	1500	บาท
2. จัดทำเอกสาร	500	บาท
รวม	2000	บาท

บทที่2
หลักการและทฤษฎี

การติดต่อสื่อสารทางเครือข่ายด้วยโปรโตคอลทีซีพี/ไอพี (TPC/IP Protocol) จำเป็นต้องทำความรู้จักกับโปรโตคอลซึ่งในปัจจุบัน IP ถือได้ว่าเป็นโปรโตคอลที่ได้รับความนิยมและรู้จักกันทั่วโลก ใช้กับเทคโนโลยีเครือข่ายที่มีอยู่หลากหลายโปรโตคอลที่ใช้สำหรับสื่อสารข้อมูลผ่านระบบเครือข่ายคอมพิวเตอร์ส่วนใหญ่มักเป็นระบบปิดเฉพาะผู้ผลิตนั้น ๆ การออกแบบบำรุงรักษา และการทำความเข้าใจในแต่ละโปรโตคอลเป็นเรื่องที่ยากมากโดยเฉพาะถ้าในองค์กรหนึ่งมีหลายเครือข่ายหลายโปรโตคอล

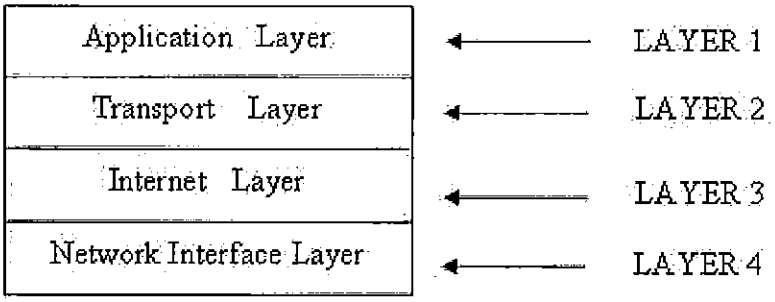
2.1 โปรโตคอลที่ใช้ส่งข้อมูลผ่านเครือข่าย (Network) [1]

2.1.1 ทีซีพี/ไอพี TCP/IP (Transmission Control Protocol/Internet Protocol)

เป็นมาตรฐานของการรับส่งข้อมูลระหว่างเครื่องคอมพิวเตอร์ 2 ระบบ ซึ่งประกอบด้วยส่วนสำคัญอยู่ 2 ส่วนด้วยกัน ก็คือ TCP (Transmission Control Protocol) และอีกส่วนก็คือ IP (Internet Protocol) การแบ่งลักษณะในการทำงานก็จะแบ่งเป็น TCP มีหน้าที่ในการตรวจสอบการรับส่งข้อมูลระหว่างเครื่องคอมพิวเตอร์ผู้รับ และเครื่องคอมพิวเตอร์ผู้ส่ง ให้ได้รับข้อมูลที่ถูกต้องและครบถ้วนหรือว่าหากมีการสูญหายของข้อมูลก็จะมีการแจ้งให้ต้นทางที่ส่งข้อมูลมารับทราบแล้วให้ทำการส่งข้อมูลมาให้ใหม่ ลักษณะการทำงานของ IP นั้น จะทำหน้าที่ในการเลือกเส้นทางที่จะใช้ในการรับส่งข้อมูลในระบบเครือข่าย และทำการตรวจสอบที่อยู่ของผู้รับโดยการใช้ข้อมูลขนาด 4 Byte เป็นตัวกำหนดแอดเดรสหรือที่เราเรียกกันว่า IP Address

1. โครงสร้างโปรโตคอลทีซีพี/ไอพี [1]

โปรโตคอล TCP/IP มีการจัดลักษณะการทำงานเป็นชั้นหรือเลเยอร์ (Layer) เรียงต่อกันดังรูปที่ 2.1



รูปที่ 2.1 เลเยอร์ของ TCP/IP [1]

1. แอพพลิเคชั่นเลเยอร์ (Application Layer) แอพพลิเคชั่นเลเยอร์จะรองรับการทำงานของแอพพลิเคชั่นต่างๆที่ทำงานเป็นโปรเซสอยู่ในเครื่องที่ให้บริการ (server) และเครื่องที่ขอบริการ (client) โดยจัดการเชื่อมต่อระหว่างโปรเซส หรือแอพพลิเคชั่นที่ต่างเครื่องกัน โดยการทำงานของแอพพลิเคชั่นต่างๆ มีการติดต่อกันตามแต่ละโปรโตคอลที่แอพพลิเคชั่นนั้นใช้งาน ซึ่งจะขอบริการจากทรานสปอร์ตเลเยอร์อีกที โปรโตคอลหลักๆในชั้นนี้คือ FTP, Telnet, HTTP และSMIP

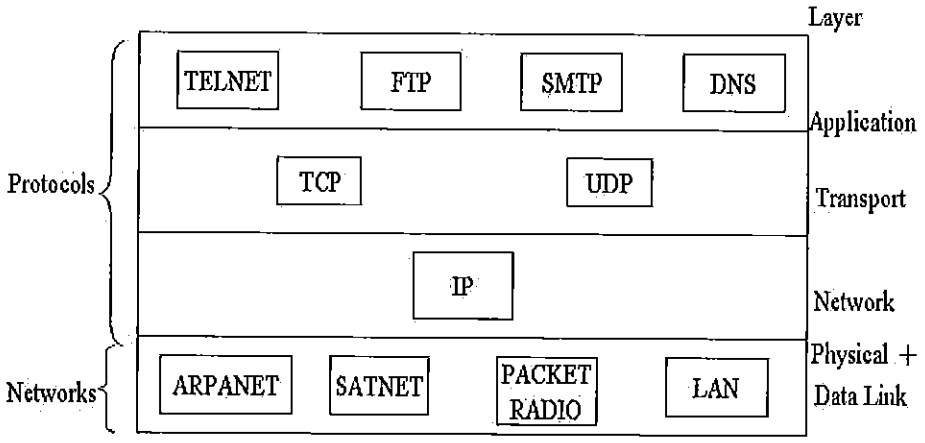
2. ทรานสปอร์ตเลเยอร์ (Transport Layer) มีหน้าที่ในการจัดการต่อจากแอพพลิเคชั่นเลเยอร์ทำหน้าที่สำหรับสร้างการเชื่อมต่อระหว่างแอพพลิเคชั่นแบบ end-to-end โดยจุดที่เชื่อมต่อเพื่อรับ และส่งข้อมูลนี้เรียกว่า พอร์ต (Port) หรือซ็อกเก็ต (Socket) ในเลเยอร์นี้มีบริการหลักอยู่ 2 แบบ คือ Connection Oriented โดยจะทำการเรียกผ่านตัวโปรโตคอล TCP และการเชื่อมต่อเป็นแบบ Connectionless Oriented โดยเรียกผ่านโปรโตคอล UDP

3. อินเตอร์เน็ตเลเยอร์ (Internet Layer) เลเยอร์นี้มีหน้าที่สำหรับส่งผ่านข้อมูลระหว่างเครือข่ายโดยโปรโตคอลที่ทำงานเป็นกลไกสำคัญในการส่งผ่านข้อมูลไปยังเครือข่ายต่างๆ ในอินเตอร์เน็ตคือ IP นอกจากนี้ในเลเยอร์นี้ยังมีโปรโตรคอลที่ทำงานอยู่ด้วยอีก 2 ชนิด คือ ICMP และ ARP

4. เน็ตเวิร์กอินเทอร์เฟสเลเยอร์ (Network Interface Layer) จะทำหน้าที่ในการแปลงข้อมูลให้อยู่ในรูปที่เหมาะสมกับเครือข่ายแต่ละแบบ ซึ่งแตกต่างกันออกไปและแปลงเป็นสัญญาณไฟฟ้าส่งไปยังเครือข่าย ซึ่งระดับชั้นนี้แบ่งการทำงานเป็นสองระดับคือ ฟิสิคัส (Physical) และดาตาลิงค์ (Data Link)

- Physical เป็นเลเยอร์ที่เป็นการกำหนดคุณสมบัติฮาร์ดแวร์ เป็นตัวกำหนดวิธีการถ่ายโอนข้อมูลในระดับบิตเช่น การเชื่อมต่อที่ตรงกับระดับฟิสิคัลเลเยอร์ RS232 และ X.21

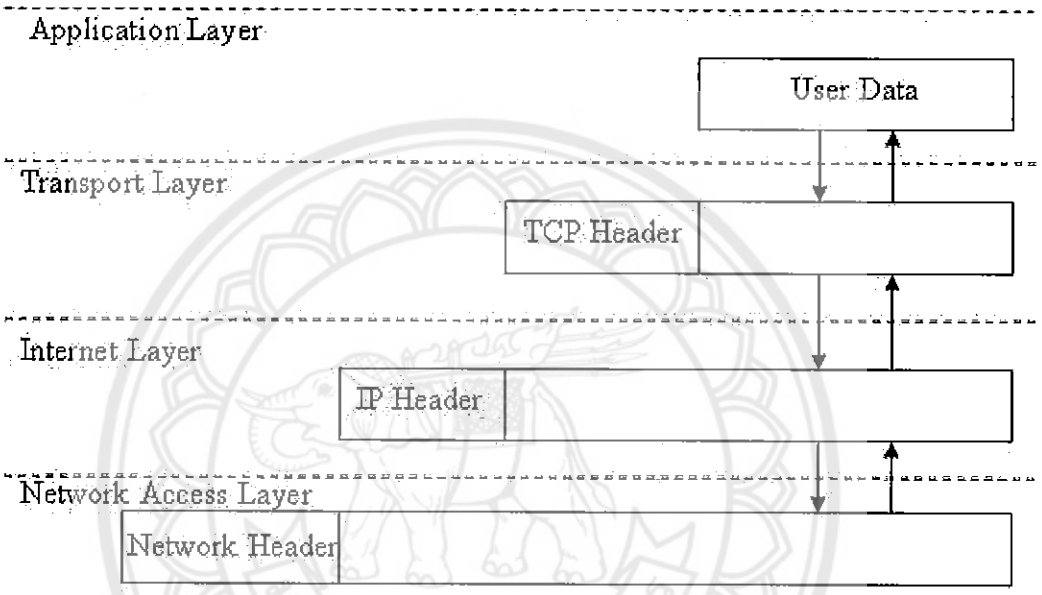
- Data Link เป็นเลเยอร์ของซอฟต์แวร์ (Device Driver) และฮาร์ดแวร์ซึ่งทำงานด้านการเชื่อมโยงเข้ากับสายสื่อสาร ตัวอย่างมาตรฐานในระดับชั้นนี้ ได้แก่ Ethernet Token-Ring เป็นต้น



รูปที่ 2.2 โปรโตคอลและเน็ตเวิร์ก ในโมเดล TCP/IP [1]

2. การสื่อสารข้อมูลระหว่างชั้น [1]

การส่งผ่านข้อมูลจากเครื่องที่ให้บริการไปยังเครื่องที่ขอใช้บริการนั้น ข้อมูลจะส่งผ่านจากโปรโตคอลที่อยู่ระดับชั้นบนสุดของเครื่องที่ให้บริการไปยังระดับล่างจนข้อมูลถูกแปลงให้เป็นสัญญาณแล้วเดินทางผ่านเครือข่ายไปยังเครื่องที่ขอบริการ โปรโตคอลระดับล่างสุดที่เครื่องขอใช้บริการจะแปลงสัญญาณที่รับมาแล้วส่งผ่านโปรโตคอลระดับบนต่อไป



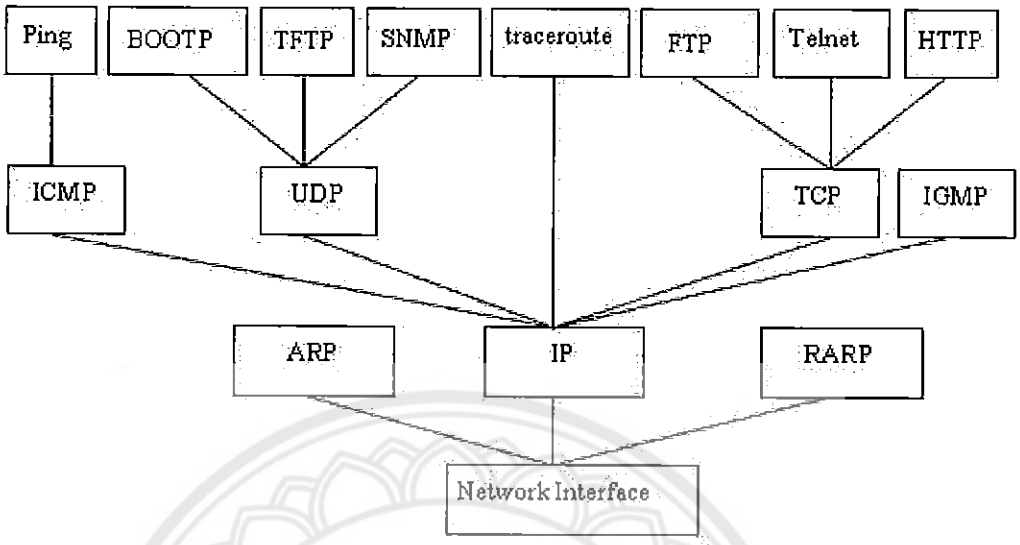
รูปที่ 2.3 การส่งข้อมูลใน TCP/IP model [1]

เมื่อข้อมูลผ่านแต่ละระดับชั้นโปรโตคอลในแต่ละชั้นจะผนวกข่าวสารและการทำงานของโปรโตคอลภายในชั้นนั้นที่เรียกว่า โปรโตคอลเฮดเดอร์ (Protocol Header) เข้ากับข้อมูลโปรโตคอลระดับล่างจะมองเฮดเดอร์และข้อมูลเหมือนเป็นข้อมูลและเพิ่มข้อมูลของชั้นตัวเองเข้าไป ข้อมูลจะถูกหุ้มเป็นชั้นๆ กระบวนการนี้เรียกว่า การเอ็นแคปซูล (Encapsulation) และเมื่อเครื่องที่ขอใช้บริการแพ็คเก็ตนี้ก็จะดำเนินการส่งไปตามลำดับชั้น โปรโตรคอลประจำชั้นจะถูกถอดเฮดเดอร์ออกและส่งส่วนที่เหลือไปยังชั้นถัดไปเฮดเดอร์จะถูกถอดออกจนเหลือเฉพาะข้อมูลเมื่อถึงชั้นบนสุด กระบวนการนี้เรียกว่า การดีแคปซูลเลต (Decapsulation)

3. การทำงานของทีซีพี/ไอพี [1]

การทำงานตามโปรแกรมประยุกต์หนึ่งๆ ไม่ได้ใช้โปรโตคอลพร้อมกันทั้งชุด ใช้แค่เพียงโปรโตคอลที่สัมพันธ์กันไปในแต่ละระดับชั้นของแบบอ้างอิง เช่น Telnet จะอาศัย TCP และ IP

ตามลำดับ การซ้อนทับของโปรโตคอลจากระดับชั้นบนไปชั้นล่างเรียกว่า โปรโตคอลสแต็ก (Stack Protocol)



รูปที่ 2.4 ความสัมพันธ์ของโปรโตคอล TCP/IP กับ Application [1]

IP ซึ่งอยู่ในเน็ตเวิร์กเลเยอร์ เป็นแกนสำคัญของโปรโตคอลสแต็ก เนื่องจากทั้ง TCPและ UDP ต้องใช้ IP เพื่อเลือกเส้นทางส่งแพ็คเก็ต อีกทั้งในเน็ตเวิร์คเลเยอร์ยังมี ICMP สนับสนุนการทำงานของ IP เพื่อรายงานความผิดปกติที่เกิดขึ้นเนื่องจากการส่งแพ็คเก็ต และมี IGMP ดูแลการจัดกลุ่มโอสต์ในเครือข่ายมัลติคาสต์ ส่วนทรานสปอร์ตเลเยอร์มี 2 โปรโตคอลที่สำคัญคือ TCP และ UDP แอพพลิเคชั่นจะเลือกใช้ TCP หรือ UDP ตามความเหมาะสมของลักษณะงานโปรโตคอลระดับล่างถัดจาก IP ได้แก่ โปรโตคอลระดับเน็ตเวิร์กอินเตอร์เฟสซึ่งกำหนดการทำงานตามเทคโนโลยีเครือข่ายที่ใช้งานในระดับนี้มีโปรโตคอลในชุดของ TCP/IP ทำหน้าที่สนับสนุนการทำงานอยู่สองโปรโตคอล คือ ARP และ RARP ทั้งสองโปรโตคอลทำหน้าที่แปลงค่าระหว่าง IP Address กับ Hardware Address

ข้อดีของโปรโตรคอล TCP/IP

1. สามารถส่งข้อมูลไปถึงจุดหมายได้แม้เส้นทางบางแห่งเสียหาย ทำให้ทนต่อความล้มเหลว ในระหว่างการสื่อสารข้อมูลถ้ามีเส้นทางบางเส้นทางใดเสียหายหรือล้มเหลว IP เน็ตเวิร์กจะปรับใช้เส้นทางอื่นทดแทนเพื่อส่งข้อมูลให้ไปถึงปลายทางอย่างอัตโนมัติ ผู้ส่งและผู้รับข้อมูลไม่จำเป็นต้องรับรู้

2. ไม่ขึ้นกับแพลตฟอร์มใดๆ ไม่ว่าเครือข่ายนั้นเป็นเครือข่ายท้องถิ่นหรือเครือข่ายระหว่างภูมิภาค เป็นไฟล์/พรินต์เซิร์ฟเวอร์หรือไคลเอ็นต์/เซิร์ฟเวอร์ เป็นระบบปฏิบัติการใด เน็ตเวิร์ก

อินเทอร์เฟสเป็นแบบใดก็ตาม ในมุมมองของโปรโตคอล TCP/IP ก็จะมองเห็นเป็น IP เน็ตเวิร์กตัวหนึ่ง

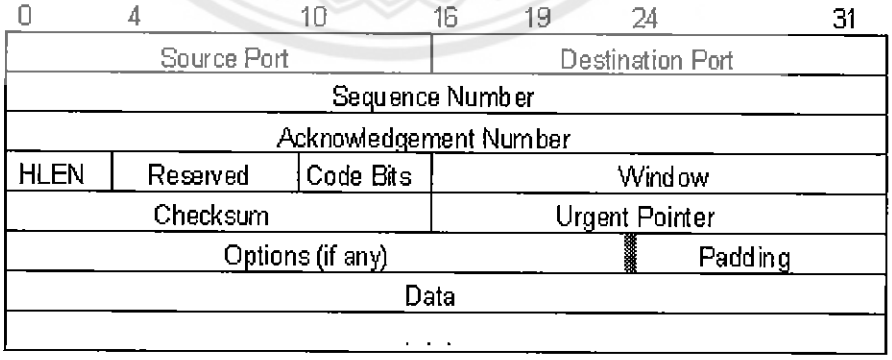
2.1.2 โปรโตคอลทีซีพี (TCP : Transmission Control Protocol) [1]

TCP เป็นโปรโตคอลแบบ Connection-Oriented ที่ให้บริการรับส่งข้อมูลโดยรับประกันความเชื่อถือในการส่งข้อมูล โดยทำหน้าที่ตรวจสอบเซกเมนต์ที่ผิดปกติและจัดส่งเซกเมนต์ใหม่ให้ รวมทั้งการจัดลำดับให้ถูกต้องก่อนส่งต่อไปยังโปรแกรมประยุกต์ระดับบนเฮดเดอร์ และข้อมูลของ TCP รวมเรียกว่า เซกเมนต์ (segment)

การส่งข้อมูลของ TCP นั้นส่งทีละเซกเมนต์โดยเครื่องปลายทางจะส่งสัญญาณตอบรับมายังเครื่องต้นทางทุกเซกเมนต์ที่ได้รับการตรวจสอบแล้วว่าไม่พบข้อผิดพลาด แต่ถ้าเครื่องต้นทางไม่ได้รับสัญญาณตอบกลับมาสำหรับเซกเมนต์ที่ส่งไป สันนิษฐานว่าเซกเมนต์นั้นมีปัญหาเกิดขึ้นและจะส่งเซกเมนต์นั้นไปให้ใหม่อีกครั้งซึ่งเรียกว่า Positive Acknowledgement with Re-transmission (PAR)

หน้าที่หลักของ TCP ในการส่งข้อมูล 6 ข้อคือ

1. ควบคุมการส่งข้อมูล (Basic Data Transfer)
2. ความน่าเชื่อถือในการรับส่งข้อมูล (Reliability)
3. ควบคุมการไหลของข้อมูล (Flow Control)
4. การทำมัลติเพล็กซ์ (Multiplexing)
5. ควบคุมการเชื่อมต่อ (Connection)
6. ความปลอดภัยในการรับส่งข้อมูล (Security)



รูปที่ 2.5 TCP Segment Format [1]

ส่วนประกอบของทีซีพีเฮดเดอร์

- Source Port : เป็นหมายเลขพอร์ตของบริการที่เครื่องต้นทาง

- Destination Port : เป็นหมายเลขพอร์ตของบริการเครื่องปลายทาง

- Sequence Number : เป็นหมายเลขที่บอกลำดับของการรับส่งข้อมูลของเครื่องที่ต้องการขอส่งข้อมูล

- Acknowledgement Number : เป็นหมายเลขที่บอกลำดับของการรับส่งข้อมูลที่ฝั่งรับข้อมูลปกติค่าของ Acknowledgement Number มีค่าเท่ากับของ Sequence Number อีกฝ่ายหนึ่ง +1 เสมอ

- Data Offset : เป็นตัวบอค่าออฟเซตของข้อมูลเพราะ TCP นั้นไม่มีการกำหนดความยาวที่แน่นอนของข้อมูล จึงต้องมีออฟเซตเป็นตัวบอก

- Flag : เป็นบิตที่บอกชนิดของข้อมูล ได้แก่

- URG : Urgent Pointer Field Significant –แสดง Urgent Pointer

- ACK : Acknowledgement Field Significant – แสดงการ Acknowledgement

- PSH : Push Function

- RST : Reset the Connection – แสดงชื่อเมื่อรีเซตการเชื่อมต่อ

- SYN : Synchronize Number – หมายเลขแพ็คเก็ตที่ส่งแบบซิงโครนัส

- FIN : No more data from sender – แสดงว่าไม่มีข้อมูลที่ส่งจากผู้ส่งแล้ว

- Window size : เป็นเลขที่บอกจำนวนของอ็อกเต็ต (Octet) ของข้อมูลจัดการในส่วนของ end-to-end flow control

- Checksum : เป็นส่วนที่ตรวจสอบความถูกต้องของข้อมูล

- Urgent pointer : เป็นตัวชี้ตำแหน่งของ Urgent Data

- Option and Padding : เป็นตัวบอกออฟชั่นของโปรเซสที่ใช้ TCP

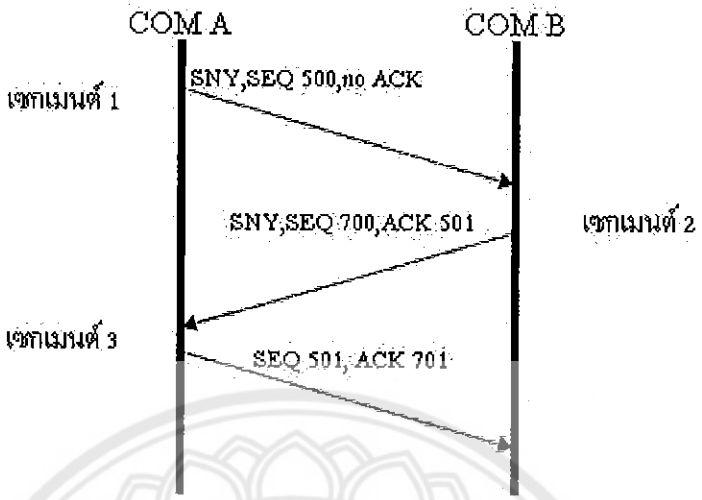
- Data : เนื้อข้อมูลที่ต้องสื่อสารมีขนาดได้ไม่ต่ำกว่า 532 –บิตเวิร์ด (6 บิตแรกสงวนไว้ และกำหนดให้เป็นศูนย์)

1. กลไกการทำงานของทีซีพี [1]

ทีซีพีมีกลไกในการลำเลียงข้อมูลระหว่างต้นทางและปลายทางหลายส่วน และมีกลไกที่สำคัญ 3 ประการคือ กระบวนการเริ่มต้นในการเชื่อมต่อ กระบวนการถ่ายโอนข้อมูลและกระบวนการยกเลิกการเชื่อมต่อ

1. กระบวนการเริ่มต้นในการเชื่อมต่อ เริ่มต้นด้วยกระบวนการทำ 3-way Handshake เป็นขั้นตอนแรกในการเชื่อมต่อในทราน สปอร์ตเลเยอร์ คือ ในการติดต่อกันระหว่างระบบในเครือข่ายต้องมีการสร้างการเชื่อมต่อไปยังระบบที่ให้บริการก่อน โดยผู้ขอใช้บริการจะส่งสัญญาณ SYS มายังผู้ให้บริการ และผู้ให้บริการจะส่งสัญญาณ ACK ตอบรับการเชื่อมต่อที่ร้องขอมา จึงจะสามารถรับส่งข้อมูลกันได้ เช่น การเชื่อมต่อระหว่าง COMA กับ COMB โดยที่ COMA เป็นฝ่ายขอรับบริการจาก COMB โดยโปรโตคอลประยุกต์ของด้าน COMA จะเป็นเครื่องลูกข่าย

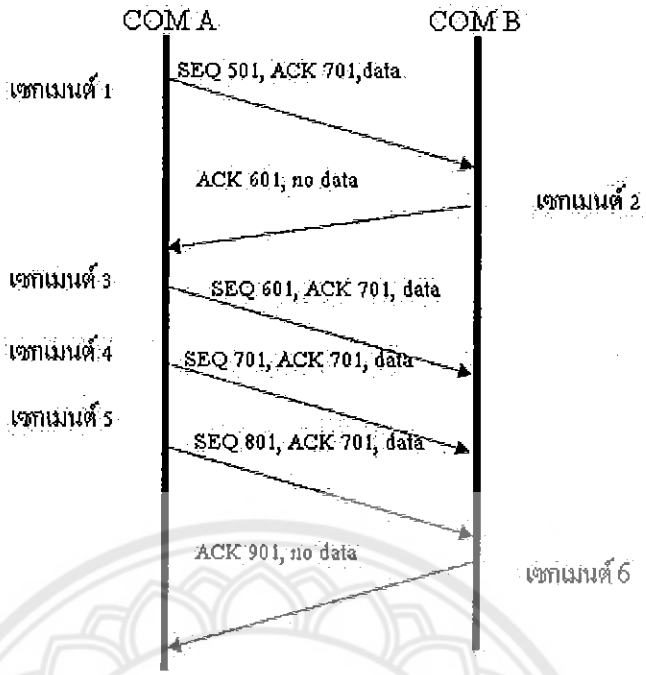
และโปรโตคอลประยุกต์ทางด้าน COMB เป็นเครื่องแม่ข่าย กระบวนการทำ3-way Handshake มีรายละเอียดดังนี้



รูปที่ 2.6 การเริ่มต้นการเชื่อมต่อของ TCP [1]

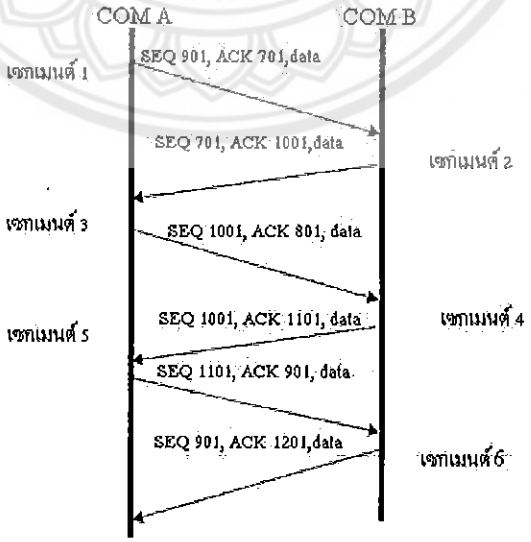
TCP ของ COMA เลือกเลขลำดับเริ่มต้นแล้วส่งเซกเมนต์ที่บรรจุเลขลำดับนี้ไปยัง COMB พร้องทั้งเซตแฟลกของ SYN ให้เป็น "1" สังเกตว่าเซกเมนต์ 1 แฟลก ACK จะมีค่าเป็น "0" TCP ของ COMB ได้รับเซกเมนต์จาก COMA ที่เลือกเลขลำดับเริ่มต้นประจำตัวเช่นเดียวกันแล้วตอบกลับด้วยเซกเมนต์ 2 โดยเซตแฟลกของ SYN และ ACK ให้เป็น "1" ทั้งคู่เพื่อแจ้งว่าได้รับเซกเมนต์1แล้ว และเลขลำดับที่ได้รับจาก COMB บวกด้วยหนึ่ง TCP ของ COMA จะส่งเซกเมนต์ตอบรับกลับไปโดยเซตแฟลก ACK และใช้เลขลำดับที่ได้รับจาก COMB บวกด้วยหนึ่งต่อจากนั้น TCP ของ COMA ได้ทำการแจ้งไปยังโปรโตคอลระดับบนว่าทำการเชื่อมต่อแล้ว ส่วน COMB เมื่อได้รับเซกเมนต์ 3 ทำการแจ้งไปยังโปรโตคอลระดับบนว่าทำการเชื่อมต่อแล้ว เมื่อสิ้นสุดทั้งสองด้าน จะสามารถรับส่งข้อมูลได้

2. กระบวนการถ่ายโอนข้อมูล จะกระทำการถ่ายโอนได้เมื่อมีการเชื่อมต่อเสร็จสิ้นแล้ว โดยเซกเมนต์ที่ 1 ที่ COMA ส่งไปยัง COMB บรรจุข้อมูลไบต์ 501 ถึง 600 และใช้เลขตอบรับ 701 เมื่อ COMB รับข้อมูลเซกเมนต์ 1 แล้วจะตอบกลับด้วยเซกเมนต์ 1และตอบกลับด้วยเซกเมนต์ 2 พร้อมเลขตอบรับกำหนดข้อมูลที่คาดว่าจะได้รับต่อไปคือ 601



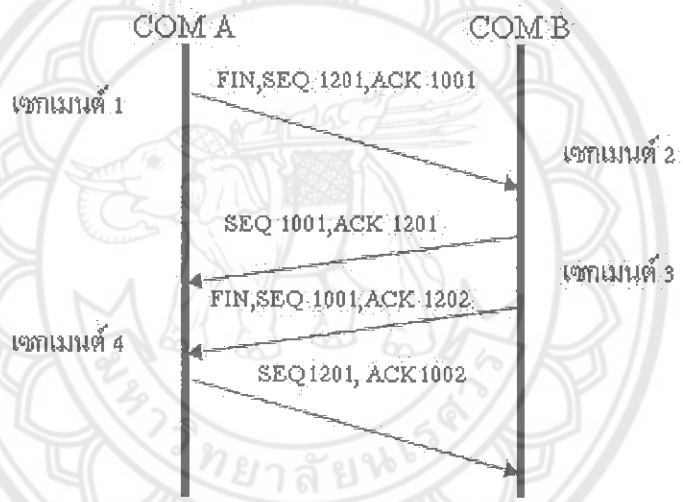
รูปที่ 2.7 ขั้นตอนการถ่ายโอนข้อมูล [1]

COMA สามารถส่งเซกเมนต์ไปอย่างต่อเนื่อง มีการส่งไป 3 เซกเมนต์ คือ 601 701 และ 801 ทาง COMB สมารถตอบกลับมาในคราวเดียวกันได้คือเซกเมนต์ 6 ในกรณีที่ส่งข้อมูลทั้งสองทิศทางระหว่าง COMA และ COMB หลังการทั่วไปยังคงเหมือนเดิมเพียงแต่มีการตอบรับพร้อมกับส่งข้อมูลระหว่างกัน



รูปที่ 2.8 การถ่ายโอนเซกเมนต์ทั้งสองทิศทาง [1]

3. กระบวนการยกเลิกการเชื่อมต่อ เนื่องจากการถ่ายโอน TCP เป็นแบบฟูลดูเพล็กซ์ การยกเลิกการเชื่อมต่อจึงต้องมีขั้นตอนเกิดขึ้นทั้งสองด้าน ถ้าสมมุติให้ COMA ไม่มีข้อมูลที่จะส่งอีกต่อไปจึงต้องเป็นฝ่ายขอปิดบริการเชื่อมต่อโดยโปรโตคอลประยุกต์ของ COMA จะแจ้งไปยัง TCP ว่าส่งข้อมูลหมดแล้วจากนั้นจะมีการแลกเปลี่ยนเซกเมนต์จำนวน 4 เซกเมนต์ดังนี้ TCP ของ COMA ส่งเซกเมนต์ที่มีเลขตอบรับและเลขลำดับตามปกติ แต่เซตแฟล็ก FIN เป็น "1" เมื่อ TCP ของ COMB ได้รับเซกเมนต์ที่มีการเซต FIN จาก COMA จะส่งเซกเมนต์ตอบรับคือเซกเมนต์ 2 และแจ้งไปยังโปรแกรมประยุกต์ COMA ขอปิดการเชื่อมต่อโปรแกรมประยุกต์ COMB แจ้งกลับมายัง TCP ว่าขอปิดการเชื่อมต่อได้ TCP COMB ส่งเซกเมนต์ FIN ไปยัง COMA (เซกเมนต์3) เมื่อ COMA ได้รับเซกเมนต์ FIN จะตอบรับกลับไป (เซกเมนต์ 4) และแจ้งไปยังโปรแกรมประยุกต์ว่าการเชื่อมต่อปิดลงแล้ว

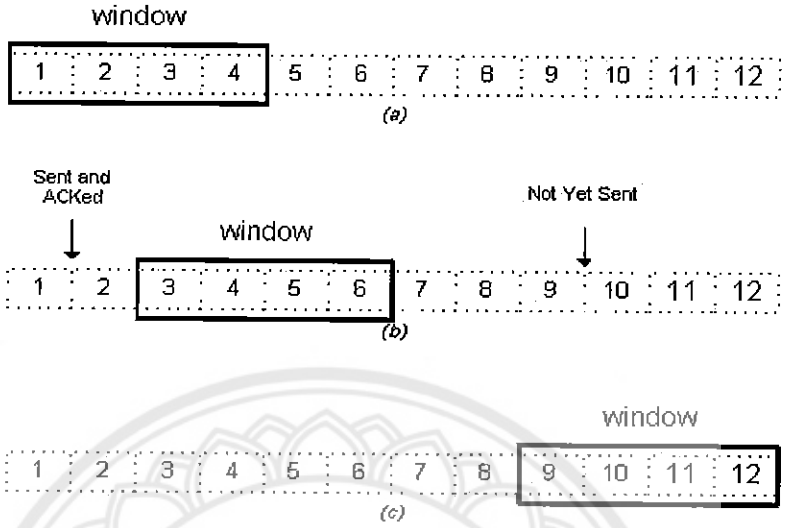


รูปที่ 2.9 การปิดการเชื่อมต่อ [1]

2. การเลื่อนหน้าต่าง (Sliding window) [1]

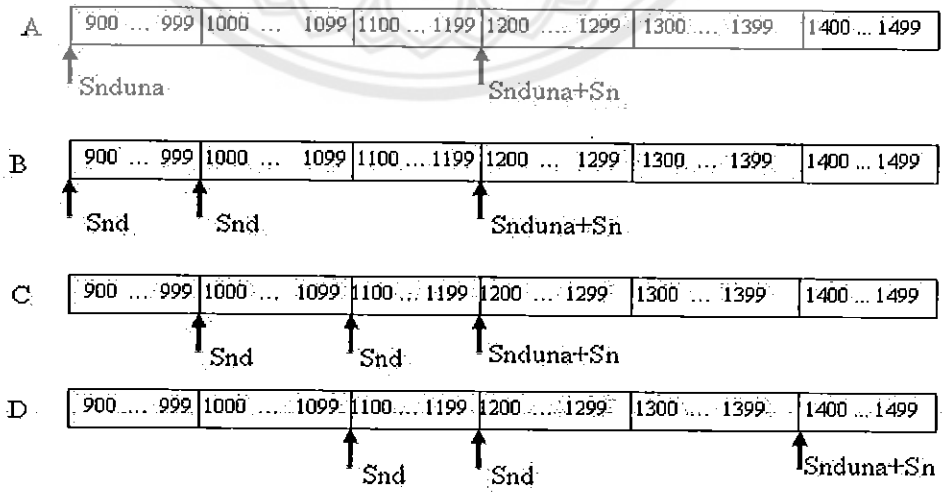
ในสภาวะการถ่ายโอนข้อมูลระหว่างเครื่องลูกข่าย COMA และเครื่องแม่ข่าย COMB ถ้า COMA ใช้หลักการตอบรับเป็นรายเซกเมนต์ คือทุกครั้งที่ส่งเซกเมนต์ต้องรอการตอบจาก COMB ก่อนส่งเซกเมนต์ถัดไปได้ ช่วงที่ COMA รอการตอบรับนั้นจะไม่สามารถส่งข้อมูลได้มันจะเหมือนกับการใช้การสื่อสารแบบฮาล์ฟดูเพล็กซ์ซึ่งทำให้เกิดการสูญเสียช่องสัญญาณไปครึ่งหนึ่ง เนื่องจากต้องรอให้อีกด้านส่งข้อมูลเรียบร้อยเสียก่อน TCP แก้ไขปัญหานี้โดยอาศัยการทำงานแบบฟูลดูเพล็กซ์ ฝ่ายส่งสามารถส่งเซกเมนต์ออกได้หลายเซกเมนต์โดยที่ไม่จำเป็นต้องรอการตอบรับทุกๆเซกเมนต์ที่ส่งออกไป แต่ต้องมีการควบคุมปริมาณการส่งนี้เพราะบัฟเฟอร์ฝ่ายรับย่อมมีขนาด

จำกัด ฝ่ายรับต้องแจ้งขนาดข้อมูลให้กับฝ่ายส่งโดยใช้เลขตอบรับ (Acknowledgment number) และขนาดหน้าต่าง (Window size) สองค่านี้ เป็นตัวกำหนดขนาดการรับส่งข้อมูลแต่ละครั้ง



รูปที่ 2.10 การเลื่อนหน้าต่าง (Sliding window) [1]

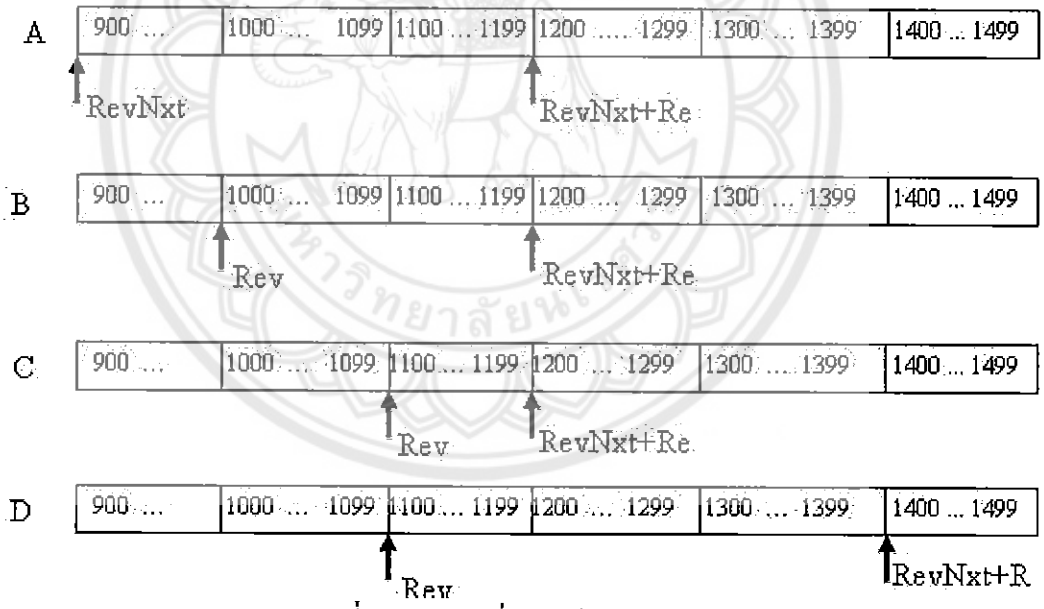
1. หน้าต่างฝ่ายส่ง TCP ควบคุมกระแสข้อมูลที่ฝ่ายส่งโดยใช้ค่า 3 ค่าคือ ตัวแปรชี้ตำแหน่งไบต์ที่ตอบรับ (SudUna) ตัวแปรชี้ตำแหน่งไบต์ที่ต้องส่งครั้งถัดไป (SndNxt) และตัวแปรกำหนดหน้าต่างที่ส่งได้ (Senwnd) ทั้ง 3 ค่าจะแบ่งบัฟเฟอร์ของฝ่ายส่งออกเป็น 4 ส่วนเพื่อให้เห็นการทำงานของหน้าต่างฝ่ายรับ ดังรูป 2.11 โดยกำหนดให้ฝ่ายส่งมีขนาดหน้าต่างเท่ากับ 300 ไบต์ และหมายเลขลำดับเริ่มต้นที่ 900 ดังรูป 2.11 (A)



รูปที่ 2.11 การเลื่อนหน้าต่างส่ง [1]

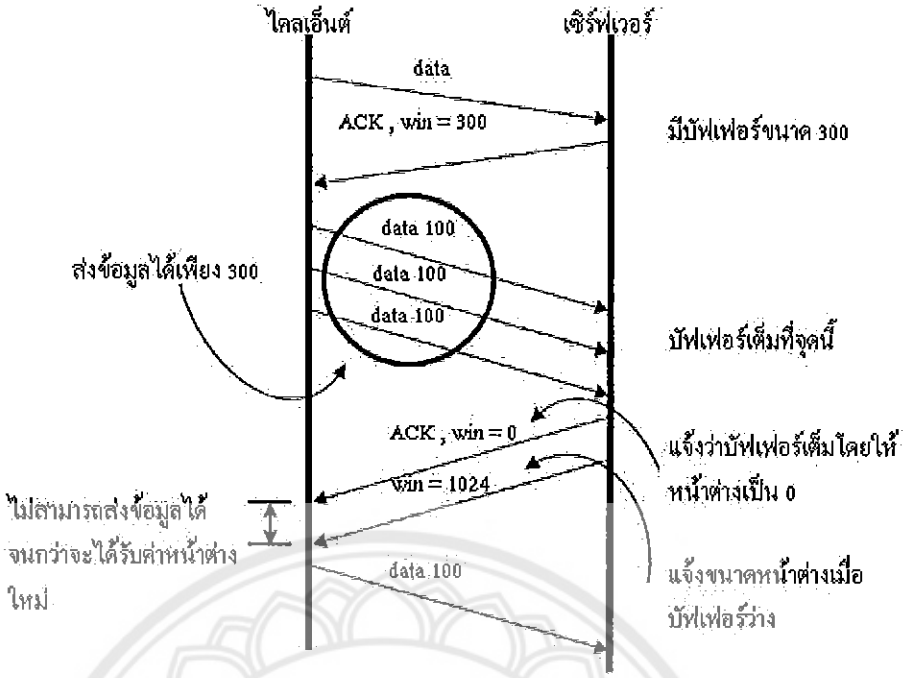
เมื่อส่งข้อมูลออกไป 100 ไบต์แรก SndNext จะเลื่อนไปทางขวา 100 ไบต์ ดังรูป 2.11 (B) ความจำเป็นของตัวแปรชี้ตำแหน่งไบต์ที่ได้รับการตอบรับล่าสุดเพื่อใช้เก็บตำแหน่งของข้อมูลที่ส่งไปแล้วแต่ยังไม่ได้รับการตอบรับเนื่องจากอาจต้องส่งซ้ำใหม่ ขั้นถัดไปหากมีข้อมูลส่งใหม่อีก 100 ไบต์ SndNxt จะขยับไปอยู่ตำแหน่ง 1100 และเมื่อมีการตอบรับข้อมูล 100 ไบต์แรกกลับมา SndUna จะเลื่อนไปที่ตำแหน่ง 1000 ดังรูป 2.11 (C) ในขั้นตอนถัดไปหากมีการตอบรับกลับมาพร้อมกับการกำหนดขนาดหน้าต่าง 300 ไบต์จะเป็นดังรูป 2.11 (D)

2. หน้าต่างฝ่ายรับ TCP ฝ่ายรับจัดการกับบัฟเฟอร์โดยให้มีตัวแปรบอกขนาดหน้าต่าง (RcvWnd) และตัวแปรชี้ตำแหน่งข้อมูลถัดไปที่คาดว่าจะได้รับ (RcvNxt) ตัวแปรทั้งสองนี้จะแบ่งบัฟเฟอร์ออกเป็น 3 ส่วนถ้าพิจารณาการเลื่อนหน้าต่างของฝ่ายรับ โดยกำหนดให้มีขนาดหน้าต่างเริ่มต้นที่ 300 ไบต์ ดังรูป 2.12 (A) เมื่อข้อมูล 100 ไบต์แรกเดินทางมาถึงหน้าต่างข้อมูลที่รับจะเหลือ 200 ไบต์ดังรูป 2.12 (B) ขั้นต่อมาหากได้รับข้อมูลอีก 100 ไบต์ ขนาดหน้าต่างจะลดลงอีกเหลือ 100 ไบต์ ดังรูป 2.11 (C) และเมื่อโปรแกรมประยุกต์นำข้อมูลนั้นไปใช้หน้าต่างก็เลื่อนและขยายขนาดขึ้นดังรูป 2.12 (D)



รูปที่ 2.12 การเลื่อนหน้าต่างฝ่ายรับ [1]

ในกรณีที่บัฟเฟอร์ฝ่ายรับเต็มจนไม่สามารถรับเซกเมนต์เพิ่มได้ TCP จะส่งเซกเมนต์กำหนดขนาดหน้าต่างโดยให้ฟิลด์ window size มีค่าเท่ากับ 0 TCP อีกฝ่ายจะหยุดส่งเซกเมนต์จนกว่าจะได้รับเซกเมนต์ที่กำหนดขนาดหน้าต่างใหม่ดังรูปที่ 2.13



รูปที่ 2.13 การประกาศขนาดหน้าต่าง [1]

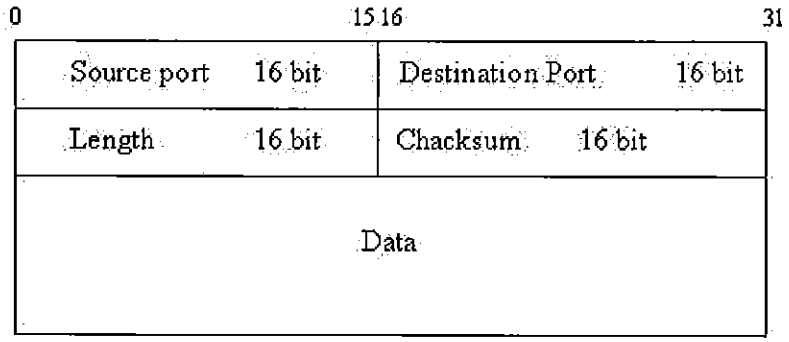
การรอการกำหนดค่าหน้าต่างใหม่เป็นกรณีหนึ่งที่ TCP ต้องดำเนินการเป็นกรณีพิเศษ เพราะเซกเมนต์ประกาศค่าหน้าต่างใหม่เกิดสูญหาย จะทำให้การสื่อสารมาสามารถดำเนินต่อไปได้ TCP แก้ปัญหากรณีนี้โดยใช้ตัวจับเวลาเพื่อรับประกันว่าต้องส่งเซกเมนต์กำหนดค่าหน้าต่างใหม่ภายในเวลาที่กำหนด

2.1.3 โปรโตคอลยูดีพี (UDP: User Datagram Protocol) [1]

โปรโตคอล UDP เป็นโปรโตคอลในการติดต่อสื่อสารในทรานสปอร์ตเลเยอร์ การทำงานคล้ายกับ TCP มากคือจัดการเกี่ยวกับการสื่อสารระหว่างเครื่องแต่เป็นแบบ Connectionless คือทั้งฝ่ายส่งและฝ่ายรับไม่จำเป็นต้องสร้างช่องทางเชื่อมต่อกัน และไม่ต้องมีการแจ้งให้ฝ่ายรับข้อมูลเตรียมรับข้อมูลเหมือนโปรโตคอล TCP และไม่มีการส่งสัญญาณตรวจสอบว่าข้อมูลถึงเครื่องปลายทางอย่างถูกต้องครบถ้วน ในการส่งข้อมูลแต่ละครั้งจึงไม่มีการส่งข้อมูลใหม่อีกในกรณีที่เกิดความผิดพลาดของการส่งข้อมูล

ส่วนประกอบของ UDP Frame

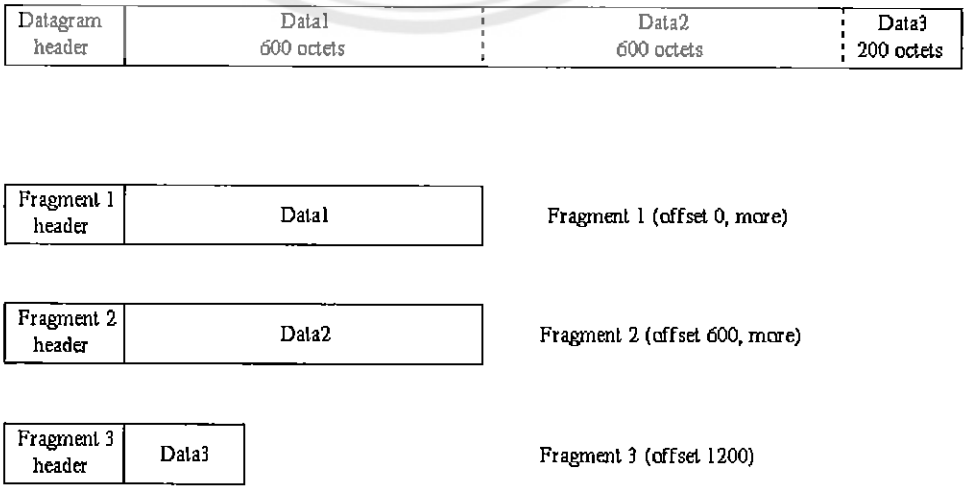
- Source Port : เป็นค่าตัวเลข 16 บิต บอกพอร์ตของบริการที่เครื่องต้นทาง
- Destination Port : เป็นค่าตัวเลข 16 บิต บอกพอร์ตของบริการที่เครื่องปลายทาง
- Length : เป็นค่าตัวเลข 16 บิต บอกความยาวของข้อมูล
- Checksum : เป็นค่าตัวเลข 16 บิต ตรวจสอบความถูกต้องของข้อมูลที่ส่งไป



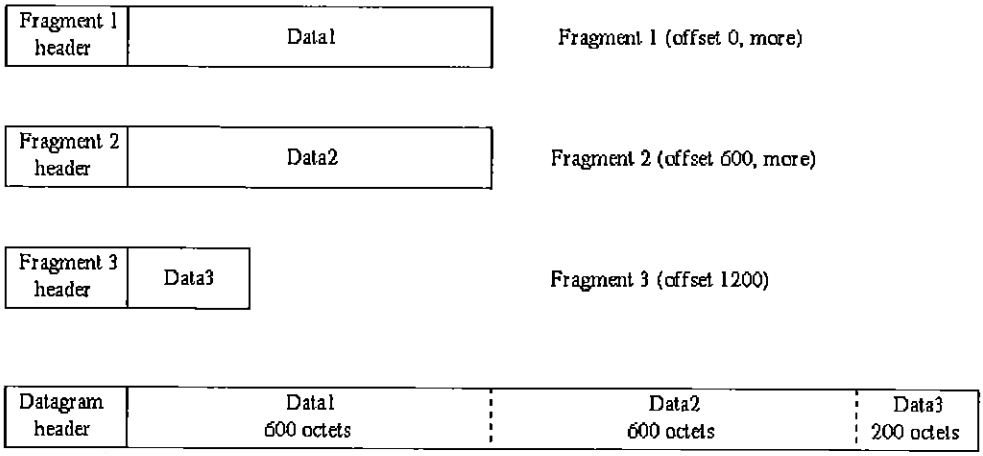
รูปที่ 2.14 แพ็คเก็ต UDP [1]

2.1.4 โปรโตคอลไอพี (IP: Internet Protocol) [1]

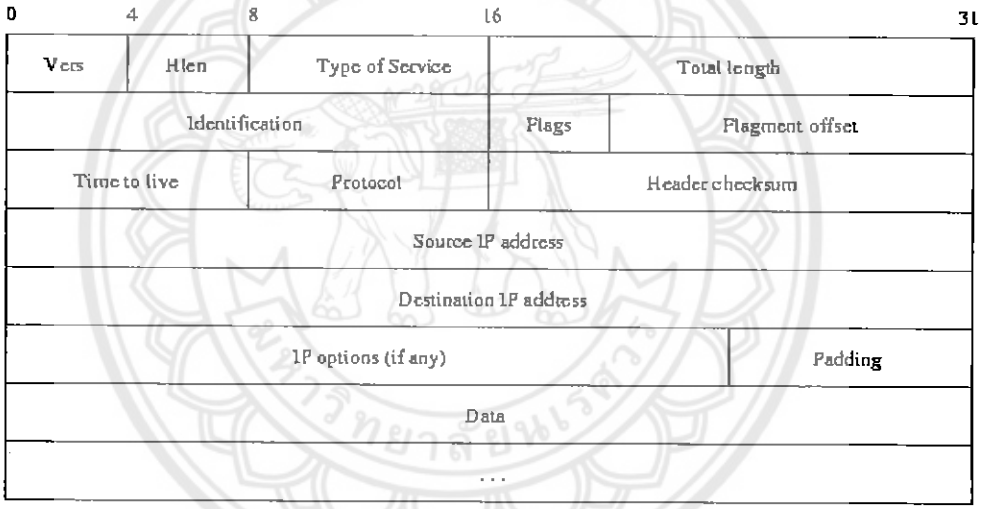
โปรโตคอล IP เป็นโปรโตคอลที่จัดการเกี่ยวกับแอดเดรสของแต่ละแพ็คเก็ต เพื่อให้ส่งแพ็คเก็ตต่างๆไปยังเป้าหมายได้ถูกต้อง การทำงานของ IP เป็นเพียงการส่งข้อมูลไปยังเครื่องเป้าหมายเท่านั้น ไม่มีการส่งสัญญาณขอบริการ หรือสัญญาณให้บริการระหว่างกันเหมือน TCP เรียกว่า การต่อแบบ Connectionless ซึ่งระบบทั้งสองตั้งสมมุติฐานว่าการเชื่อมต่อระหว่างกันไม่มีความผิดพลาดเกิดขึ้นแน่นอน เนื่องจากมาตรฐานในเครือข่ายมีหลากหลาย ขนาดของแพ็คเก็ตในแต่ละมาตรฐานจึงมีความแตกต่างกันออกไปทำให้การส่งข้อมูลระหว่างอุปกรณ์ในเครือข่ายนั้นอาจมีการแบ่งข้อมูลออกเป็นแพ็คเก็ตย่อยๆ ในระหว่างการส่งเรียกว่า การทำแฟร็กเมนเตชั่น (Fragmentation) เช่นแพ็กเก็ตของ FDDI มีขนาด 4500 ไบต์ ถ้าเครื่องปลายอยู่ในเครือข่ายอีเตอร์เน็ต (Ethernet)ซึ่งมีขนาดของแพ็คเก็ตสูงสุดเพียง 1500 ไบต์ ดังนั้นการส่งแพ็คเก็ตไปยังเครื่องปลายทางจึงต้องมีการแบ่งเป็นแพ็คเก็ตย่อย และเมื่อแพ็คเก็ตย่อยมาถึงเครื่องเป้าหมายก็จะมารวมกันเป็นแพ็คเก็ตเดิมที่มีขนาด 4500 ไบต์อีกครั้งเรียกการรวมนี้ว่า การรีแอสเซมเบิล (Reassemble) ซึ่งทำให้ได้ข้อมูลเหมือนที่ส่งมาจากเครื่องต้นทาง



รูปที่ 2.15 การทำแฟร็กเมนเตชั่น (Fragmentation) [1]



รูปที่ 2.16 การรีแอสเซมเบิล (Reassemble) [1]



รูปที่ 2.17 แพ็คเก็ต IP [1]

ส่วนประกอบของแพ็คเก็ตไอพี

1. Version : เป็นค่าตัวเลข 4 บิต บอกเวอร์ชันของมาตรฐาน IP ที่ใช้โดยปกติมีค่าเป็น 4 ซึ่งหมายถึง IPv4

2. Internet Header Length (IHL) : เป็นตัวบอกความยาวเฮดเดอร์ของ IP

3. Type of Service : เป็นส่วนที่จะบอกการทำงานของแพ็คเก็ตที่ส่งไปว่าทำหน้าที่อะไร มีทั้งหมด 8 บิต

Bit 0-2 : บอกรายละเอียดการทำงานของแพ็คเก็ตนั้นๆ

111 – Network Control

110 – Internetwork Control

101 – CRITIC/ECP

100 – Flash override

011 – Flash

010 – Immediate

001 – Priority

000 – Routine

Bit 3 : การบอกถึงลักษณะของดีเลย์ (Delay)

0 – Normal Delay – มีดีเลย์ปกติ

1 – Low Delay – มีดีเลย์ต่ำ

Bit 4 : บอกถึงประเภทของทรูพุต (Throughput)

0 – Normal Throughput – มีทรูพุตปกติ

1 – High Throughput – มีทรูพุตสูง

Bit 5 : บอกถึงประเภทของความน่าเชื่อถือ

0 – Normal Reliability – มีความน่าเชื่อถือพอประมาณ

1 – High Reliability – มีความน่าเชื่อถือสูง

Bit 6-7 : กันไว้ใช้ในอนาคต

4. Total Length : มีขนาด 16 บิต บอกถึงความยาวในดาต้าแกรมของ IP

5. Identification field : เป็นตัวเลข 16 บิต เป็นค่าประจำตัวของ IP นั้นโดยโฮส์ที่ส่งจะเป็นผู้กำหนด และเพิ่มค่าขึ้นหนึ่งเมื่อมีการส่งดาต้าแกรมของ IP ใหม่

6. Flag : เป็นตัวเลข 3 bit บอกลักษณะของแพ็ตเก็ตว่ามีการแฟร็กเมนต์หรือไม่

Bit 0 : สงวนไว้ปกติเป็น 0

Bit 1 : 0 = บอกว่าแพ็คเก็ตมีการแตกแพ็คเก็ตย่อย

1 = บอกว่าแพ็คเก็ตไม่มีการแตกแพ็คเก็ตย่อย

Bit 2 : 0 = บอกว่าแพ็คเก็ตนั้นเป็นแพ็คเก็ตสุดท้ายที่ได้จากการแตกแพ็คเก็ตย่อย

1 = บอกว่าแพ็คเก็ตนั้นยังไม่เป็นแพ็คเก็ตสุดท้ายที่ได้จากการแตกแพ็คเก็ตย่อย

7. Fragment Offset : เป็นค่าตัวเลข 13 บิตบอกออฟเซต (offset) ของแฟร็กเมนต์เมื่อเทียบในดาต้าแกรม

8. Time To Live (TTL) : เป็นตัวเลข 8 บิต บอกช่วงเวลาของแพ็คเก็ตที่ยังอยู่ในเครือข่ายได้ โดยกำหนดค่าเป็นจำนวนเราเตอร์สูงสุดที่ดาต้าแกรมผ่านได้ ซึ่งโดยทั่วไปมีค่า ระหว่าง 32 ถึง 64 และลดค่าเฉลี่ยเรื่อยๆ เมื่อผ่านเราเตอร์ เพื่อเป็นการป้องกันแพ็คเก็ตล้นเครือข่าย

9. Protocol : เป็นตัวเลข 8 บิต บอกถึงโปรโตคอลที่อยู่เหนือขึ้นไปว่าเป็นโปรโตคอลระดับสูงกว่าประเภทใด

10. Header Checksum : เป็นค่าตัวเลข 32 บิต ใช้ตรวจสอบความถูกต้องของเฮดเดอร์

11. Source Address : เป็นตัวเลข 32 บิต บอกถึงไอพีแอดเดรสของเครื่องต้นทาง

12. Destination Address : เป็นค่าตัวเลข 32 บิต บอกถึงไอพีแอดเดรสของเครื่องปลายทาง

1. การใช้ IP เน็ตเวิร์ก [1]

นอกจากแนวความคิดในการนำ IP เน็ตเวิร์กมาผูกติดกับการใช้เป็น VoLP ยังมีแนวความคิดในการนำ IP ไปใช้บนเทคโนโลยีเครือข่ายอื่น ๆ เช่น ทำเป็น VoLP over Frame Relay, IP VPN over Frame Relay, VolP over ATM เป็นต้น ซึ่งเป็นการนำ IP แอดเดรสที่มีอยู่เดิมไปใช้บนเทคโนโลยีเครือข่ายในการส่งข้อมูลที่เป็นทั้งรูปแบบเสียง และข้อมูลบนเน็ตเวิร์กมากกว่าโมเด็มที่มีอยู่ในปัจจุบันมากหลายเท่าตัว Frame Relay, IP หรือ ATM ก็ใช้เทคโนโลยีในการส่งข้อมูลออกไปเป็นแพ็กเก็ต แตกต่างที่ Frame Relay ส่งข้อมูลแบบ Fast Packet หรือ FastFrame ส่วน ATM จะเป็นแบบ Fast Cell ดังนั้น จึงมีการนำความคิดนี้ไปผสานกับเทคโนโลยีของ IP ที่มีการส่งข้อมูลเป็นเฟรมหรือแพ็กเก็ตออกไปยังเครือข่าย แนวความคิดนี้จึงทำให้เทคโนโลยีเครือข่าย Frame Relay และ ATM ผสานกับ IP ได้อย่างลงตัว ทำให้การส่งข้อมูลแบบ IP ไปบนเครือข่ายดังกล่าวมีความปลอดภัย น่าเชื่อถือมากขึ้น จึงทำให้การส่งข้อมูลภายในเน็ตเวิร์กรวดเร็วขึ้น

การส่งข้อมูลแพ็กเก็ตจะมีคุณสมบัติที่ดีกว่าการส่งในรูปแบบของเซอร์กิตสวิตช์ เพราะจะมีเสถียรภาพมากกว่า มีความผิดพลาดในการส่งข้อมูลที่น้อยกว่าเมื่อเกิดความผิดพลาดก็สามารถส่งข้อมูลในช่วงที่มีความผิดพลาดไปใหม่ได้เลย โดยไม่จำเป็นต้องส่งไปใหม่ทั้งหมด ด้วยเหตุนี้จึงทำให้เทคโนโลยี ATM จึงเป็นที่นิยมกัน อย่างที่ทราบกันดีว่า จุดอ่อนของ IP จะอยู่ที่ Qos ที่ไม่สามารถควบคุมการส่งข้อมูลไปยังปลายทาง ซึ่งต้องมีการจัดคิวได้อย่างมีประสิทธิภาพ ซึ่ง IP ยังไม่สามารถทำได้ดีเท่าที่ควร แต่เมื่อมีการนำ IP ไปใช้ผสานกับเทคโนโลยีเครือข่าย ATM และ Frame Relay ทำให้สามารถกำหนดช่องสื่อสารเฉพาะสำหรับการประยุกต์บางอย่าง และสามารถกำหนดขนาดของแบนด์วิดธ์ให้ใช้งานได้ตามที่ผู้ใช้ต้องการส่งข้อมูล

ข้อเสียของ IP มี 2 ประเด็นคือ [2]

1. รับส่งโดยไม่มีการรักษาความปลอดภัยเนื้อข้อมูล การรับส่งข้อมูลด้วย IP แพ็กเก็ตไม่มีทั้งการเข้ารหัสข้อมูลและป้องกันการปลอมแปลงใดๆ การไม่เข้ารหัสข้อมูลอาจทำให้ผู้ไม่ประสงค์ดีระหว่างเส้นทางที่ IP แพ็กเก็ตผ่านดักลอบดูเนื้อข้อมูลอย่างง่ายดาย แม้ว่าเราอาจสามารถบังคับเส้นทางของ IP แพ็กเก็ตได้ก็ไม่อาจมั่นใจได้ว่าระหว่างทางมีการดักลอบดูหรือไม่ ในเรื่องปัญหาการปลอมแปลงแบ่งออกเป็นสองกรณีคือ การปลอมแปลงหรือดัดแปลงเนื้อข้อมูล และการปลอมแปลงส่วนหัวของ IP แพ็กเก็ต ทั้งสองกรณีให้ผลเหมือนกันคือผู้รับได้ข้อมูลที่ผิดจากความเป็นจริง ทว่าจุดประสงค์ต่างกัน หากเป็นกรณีแรกนั้น ผู้ไม่หวังดีต้องการหลอกหรือกลั่นแกล้งให้ได้ข้อมูลผิด ๆ หากเป็นกรณีหลัง ผู้ไม่หวังดีต้องการแอบอ้างว่าข้อมูลนั้นมาจากแหล่งที่ผู้รับไว้ใจหรือแหล่งอื่นที่กลายเป็นเหยื่อของการแอบอ้างโดยไม่รู้ตัว

2. รับส่งโดยไม่คำนึงถึงคุณภาพการให้บริการ การรับส่งต่อ IP แพ็กเก็ตระหว่างเครือข่ายย่อยไปเป็นทอดนั้นใช้หลักการใครมาก่อนได้ก่อน ฉะนั้นจึงคาดเดาไม่ได้ว่าข้อมูลที่นำส่งไปจะไปถึงปลายทางเมื่อใด แม้ว่า IP เน็ตเวิร์กใช้หลักการเลือกเส้นทางที่เหมาะสมที่สุดในขณะนั้นก็ตาม หากแต่ความเหมาะสมนั้นผู้ส่งและผู้รับไม่อาจคาดการณ์หรือมีส่วนร่วมตัดสินใจได้เลยว่าจะช้าเร็วหรือมีโอกาสที่ข้อมูลผิดพลาดมากน้อยเพียงไร

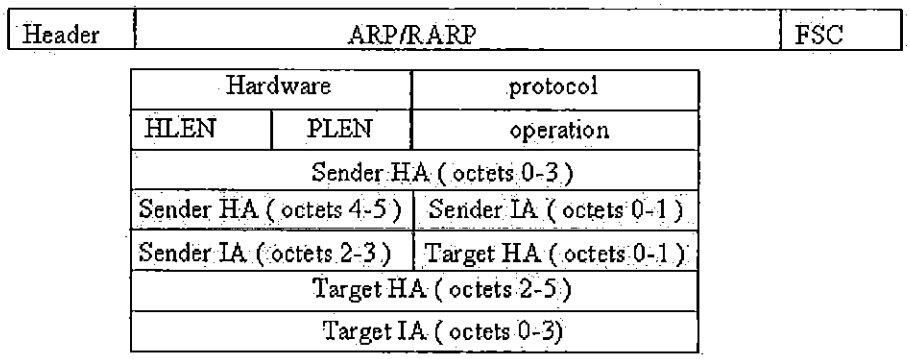
2. แนวทางในการแก้ไข [2]

องค์กรกลางของอินเทอร์เน็ตได้ออกมาตรฐานที่ช่วยแก้ไขปัญหานี้คือ IPSec โดยมีทั้งการเข้าและถอดรหัสเนื้อข้อมูลในระดับ IP แพ็กเก็ตการตรวจสอบความถูกต้องเนื้อข้อมูลและการพิสูจน์ตนของ IP แพ็กเก็ตเพื่อป้องกันการปลอมแปลง

ส่วนเรื่องการรับส่งโดยไม่คำนึงถึงคุณภาพการให้บริการ ตามจริงแล้วมาตรฐานแต่ต้นของ IP เน็ตเวิร์กมีการจัดลำดับความสำคัญของ IP แพ็กเก็ต โดยเป็นแฟร็ก Type of Service สองกลุ่ม กลุ่มแรกเป็นเลขลำดับความสำคัญยิ่งมีค่ามากยิ่งสำคัญมาก กลุ่มที่สองเป็นประเภทของงานที่ใช้ IP แพ็กเก็ตนั้นมีสามประเภทคือ งานที่มีความล่าช้าสูง งานที่ต้องการทรูพุตสูงและงานที่ต้องการความน่าเชื่อถือสูง จึงต้องใส่ใจกับแฟร็กลำดับความสำคัญเพื่อพิจารณาส่งต่อ IP แพ็กเก็ต แนวทางแก้ไขมีสองแนวทางคือใช้ การสื่อสารระดับดาต้าลิงก์ (ระดับล่างของระดับเน็ตเวิร์ก หรือส่งเสริมให้การส่งต่อ IP แพ็กเก็ตโดยพิจารณาลำดับแฟร็ก เหตุที่อินเทอร์เน็ตใช้รูปแบบของโปรโตคอล IP ก็เพราะว่า ณ เวลาที่มีการออกแบบนั้น ระบบเครือข่ายยังมีความเร็วต่ำ และมีสัญญาณรบกวนมาก อีกทั้งการใช้งานในขณะนั้นก็มีเพียงอีเมล์ FTP, Usenet News และการใช้งานออนไลน์แบบเช่น Telnet ซึ่ง ล้วนแล้วแต่อยู่ในรูปแบบของตัวบรเท่านั้น (Text Mode) ลักษณะนี้ก็เพียงพอกับการทำงาน

2.1.5 โปรโตคอลเออาร์พี (ARP: Address Resolution Protocol) [1]

โปรโตคอล ARP เป็นโปรโตคอลที่ออกแบบมาเพื่อใช้ในเครือข่ายที่สนับสนุนการบอร์ดคาสต์ (Broadcast) ถูกเรียกใช้งานโดยโปรโตคอล IP เพื่อช่วยแปลงหมายเลย IP ไปเป็นหมายเลขฮาร์ดแวร์ปลายทาง ตัวอย่างเช่น เว็บเซอร์ฟเวอร์เครื่องหนึ่งเชื่อมต่ออยู่ในเครือข่ายอินเตอร์เน็ต และในการเชื่อมต่อนี้ต้องอาศัยการ์ดแลน (LAN card) ติดตั้งอยู่ ที่แลนการ์ดนี้จะมีหมายเลขประจำฮาร์ดแวร์ ที่ไม่ซ้ำกับใคร เพื่อใช้อ้างอิงการส่งข้อมูลในเครือข่ายแต่เมื่อมาใช้งานในโปรโตคอล TCP/IP ก็ต้องมีการกำหนดหมายเลข IP Address ประจำตัวเพื่อใช้อ้างอิงกันโปรโตคอล ARP จะทำหน้าที่แปลงค่าหมายเลข IP ให้เป็นหมายเลขฮาร์ดแวร์จริงในระดับการทำงานที่อินเทอร์เน็ตเลเยอร์ ซึ่งกลไกการแปลงนี้เรียกว่า แอดเดรสรีโซลูชั่น (address resolution)



รูปที่ 2.18 ARP Datagram [1]

- ส่วนประกอบของเออาร์พีดาต้าแกรม

1. Hardware 16 บิต : กำหนดชนิดของฮาร์ดแวร์เครือข่ายที่ ARP ทำงานอยู่ ค่าใช้งานดังนี้

1 อีเทอร์เน็ต

4 โทเค็นริง

5 เดออส (chaos)

6 เครือข่าย IEEE 802

7 อาร์คเน็ต

12 โลคัลทอล์ค

2. Protocol 16 บิต : ชนิดของโปรโตคอลที่ร้องขอใช้ ARP

3. HLEN 8 บิต : ขนาดของฮาร์ดแวร์แอดเดรสเป็นจำนวนไบต์ ค่าปกติที่ใช้งานคือ 6 ซึ่งเท่ากับ 6 ไบต์ของอีเทอร์เน็ตฮาร์ดแวร์แอดเดรส

4. PLEN 8 บิต : ขนาดของแอดเดรสระดับเน็ตเวิร์กเป็นจำนวนไบต์ค่าปกติที่ใช้คือ 4 ซึ่งเท่ากับ 4 ไบต์ ของ ไอพีแอดเดรส

5. Operation 16 บิต : กำหนดรูปแบบการใช้ดาต้าแกรม ค่าในฟิลด์นี้ใช้กำหนดการทำงานของทั้ง ARPและ RARP ซึ่งมี 4 ค่าคือ

ARP request (ค่าเท่ากับ 1)

ARP reply (ค่าเท่ากับ 2)

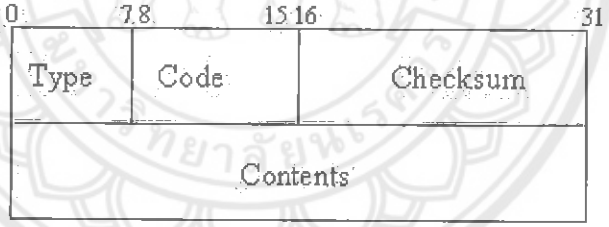
RARP request (ค่าเท่ากับ 3)

RARP reply (ค่าเท่ากับ 4)

6.Address : ฟิลด์แอดเดรสเรียงลำดับจาดฮาร์ดแวร์และเน็ตเวิร์กแอดเดรสของสถานีที่ร้องขอตามด้วยฮาร์ดแวร์และเน็ตเวิร์กแอดเดรสของสถานีที่ร้องขอตามด้วยฮาร์ดแวร์และเน็ตเวิร์กแอดเดรดของสถานีที่ตอบรับ

2.1.6 โปรโตคอลไอซีเอมพี (ICMP : Internet Control Message Protocol) [1]

หน้าที่หลักของโปรโตคอล ICMP คือการแจ้งหรือแสดงข้อความจากระบบ เพื่อบอกให้ผู้ใช้ทราบว่าเกิดอะไรขึ้นในการส่งผ่านข้อมูลนั้นๆ ปัญหาส่วนมากคือ ส่งไปไม่ได้ หรือ ปลายทางไม่สามารถรับข้อมูลได้ เป็นต้น นอกจากนี้โปรโตคอล ICMP ยังถูกเรียกใช้งานจากเครื่องแม่ข่ายและเราเตอร์อีกด้วย เพื่อแลกเปลี่ยนข้อมูลที่ใช้ควบคุม ส่วนรูปแบบการทำงานของโปรโตคอลนี้จะทำงานควบคู่ไปกับโปรโตคอล IP ในระดับเดียวกันและข้อความต่างๆที่แจ้งให้ทราบจะถูกผนึกอยู่ในข้อมูลของ IP (IP Datagram)อีกทีหนึ่ง ข้อความที่โปรโตคอล ICMP ส่งนั้นแบ่งออกเป็น 2 แบบ คือ ICMP error message หรือข้อความแจ้งข้อผิดพลาด และ ICMP query หรือข้อความเรียกข้อมูลเพิ่มเติม ตัวอย่างกลไกการทำงานของโปรโตคอล ICMP เช่น เมื่อมีการส่งผ่านข้อมูลของผู้ใช้ไปยังปลายทางที่ไม่ถูกต้อง หรือขณะนั้นเครื่องปลายทางเกิดปัญหาจนไม่สามารถรับข้อมูลที่เราเตอร์จะส่งข้อความแจ้งเป็น ไอซีเอ็มพีเมสเซจ (ICMP message) ที่ชื่อ destination unreachable ให้กับผู้ส่งข้อมูล นอกจากนี้ตัวข้อมูลที่แจ้งข้อความก็จะมีส่วนของข้อมูล IP Datagram ที่เกิดปัญหาด้วย ดังนั้นเมื่อผู้ส่งข้อมูลได้รับข้อความแจ้งแล้วก็จะทราบว่าจุดที่เกิดปัญหาอยู่ที่ใด ดังนั้นโปรโตคอล ICMP จึงกลายมาเป็นเครื่องมืออย่างหนึ่งในการช่วยทดสอบเครือข่าย เช่น คำสั่ง ping ที่เรามักใช้ทดสอบว่าเครื่องแม่ข่ายที่ให้บริการหรืออุปกรณ์ที่ต่ออยู่ในเครือข่ายอินเตอร์เน็ตนั้นยังทำงานเป็นปกติหรือไม่ แล้วคำสั่ง ping มีการเรียกใช้งานโปรโตคอล ICMP แจ้งเป็นข้อความให้ทราบอีกต่อหนึ่ง



รูปที่ 2.19 รูปแบบของ ICMP [1]

- ส่วนประกอบของไอซีเอ็มพี

Type : ขนาด 8 บิต : กำหนดค่าความผิดพลาดและการรายงานสถานการณ์ใช้งานในปัจจุบันมีทั้งหมด 15 ประเภท

Code ขนาด 8 บิต : รหัสความผิดพลาดย่อย

Checksum ขนาด 16 บิต : ค่าผลรวมตรวจสอบแบบ 1's complement สำหรับใช้ตรวจสอบความผิดพลาด โดยคำนวณผลรวมของ type ,code และ contents

Contents ขนาดไม่คงที่ : ฟิลด์นี้ใช้บรรจุข้อมูลข่าวสารเพิ่มเติมเพื่อแจ้งกลับซึ่งจะขึ้นอยู่กับค่า type และ code

2.2 ความรู้ทั่วไปเกี่ยวกับ Port [3]

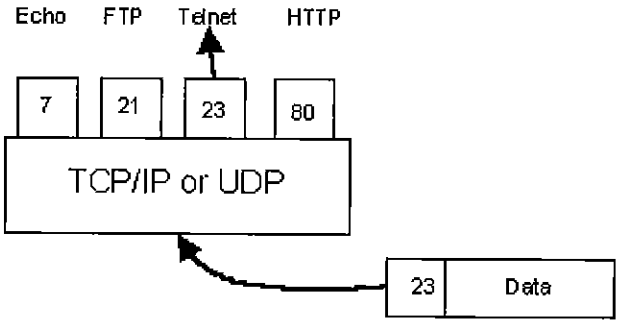
สำหรับพวก Application ในชั้น layer สูงๆ ที่ใช้ TCP (Transmission Control Protocol) หรือ UDP (User Datagram Protocol) จะมีหมายเลข Port หมายเลขของ Port จะเป็นเลข 16 bit เริ่มตั้งแต่ 0 ถึง 65535 หมายเลข Port ใช้สำหรับตัดสินว่า service ใดที่ต้องการเรียกใช้ ในทางทฤษฎีหมายเลข Port แต่ละหมายเลขถูกเลือกสำหรับ service ใดๆ ขึ้นอยู่กับ OS (operating system) ที่ใช้ไม่จำเป็นต้องเหมือนกัน แต่ได้มีกำหนดขึ้นให้ใช้ค่อนข้างเป็นมาตรฐานเพื่อให้มีการติดต่อการส่งข้อมูลที่ดีขึ้น ทาง Internet Assigned Numbers Authority (IANA) เป็นหน่วยงานกลางในการประสานการเลือกใช้ Port ว่า Port หมายเลขใดควรเหมาะสำหรับ Service ใด และได้กำหนดใน Request For Comments (RFC') 1700 ตัวอย่างเช่น เลือกใช้ TCP Port หมายเลข 23 กับ Service Telnet และเลือกใช้ UDP Port หมายเลข 69 สำหรับ Service Trivial File transfer Protocol (TFTP) ตัวอย่างต่อไปนี้เป็นบางส่วนของ File/etc/services แสดงให้เห็นว่า หมายเลข Port แต่ละหมายเลขได้ถูกจับคู่กับ Transport Protocol หนึ่งหรือสอง Protocol ซึ่งหมายความว่า UPP หรือ TCP อาจจะใช้ หมายเลข Port เดียวกันก็ได้ เนื่องจากเป็น Protocol ที่ต่างกัน หมายเลข Port ถูกจัดแบ่งเป็น 2 ประเภท

1. Well Known Ports คือจะเป็น Port ที่ระบบส่วนใหญ่ กำหนดให้ใช้โดย Privileged User (ผู้ใช้ที่มีสิทธิพิเศษ) โดย port เหล่านี้ ใช้สำหรับการติดต่อระหว่างเครื่องที่มีระบบเวลาที่ยาวนานวัตถุประสงค์เพื่อให้ service แก่ผู้ใช้ที่ไม่รู้จักหรือคุ้นเคย จึงจำเป็นต้องกำหนด Port ติดต่อสำหรับ Service นั้นๆ

2. Registered Ports จะเป็น Port หมายเลข 1024 ขึ้นไป ซึ่ง IANA ไม่ได้กำหนดไว้

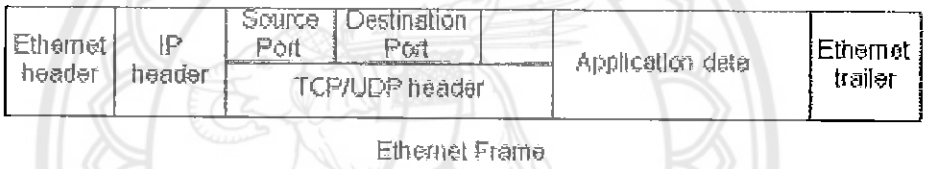
2.2.1 ตัวอย่างการใช้ Port [3]

แต่ละ Transport layer segment จะมีส่วนย่อยที่ประกอบไปด้วยหมายเลข Port ของเครื่องปลายทาง โดยที่เครื่องปลายทาง (Destination host) จะใช้ Port นี้ในการส่งข้อมูลให้ไหลกับ Application ได้ถูกต้อง หน้าที่ในการส่งหรือแจกจ่าย Segment ของข้อมูลให้ตรงกับ Application เรียกว่าการ "Demultiplexing" ในทางกลับกันเครื่องต้นทาง (Source host) หน้าที่ในการรวบรวมข้อมูลจาก Application และเพิ่ม header เพื่อสร้าง segment เรียกว่า "Multiplexing" หรือถ้ายกตัวอย่างเป็นภาษาทั่วๆ ไป คือ ในแต่ละบ้านจะมีคน 1 คนรับผิดชอบเก็บจดหมายจากกล่องจดหมาย ถ้าเป็นการ Demultiplexing คนๆ นั้นจะแจกจ่ายจดหมายที่จ่าหน้าซองให้สอดคล้องกับบุคคลนั้นๆ ในบ้าน ในทางตรงกันข้าม ถ้าเป็นการ Multiplexing คนๆ นั้นก็จะรวบรวมจดหมายจากสมาชิกในบ้านและทำหน้าที่ส่งออกไป Demultiplexing ตามรูปที่ 2.20



รูปที่ 2.20 การแจกจ่าย Segment แบบDemultiplexing [3]

หมายเลข Port จะอยู่ใน 32 bit แรกของ TCP และ UDP header โดยที่ 16 bit แรกเป็นหมายเลข Port ของเครื่องต้นทาง ขณะที่ 16 bit ต่อมาเป็นหมายเลข Port ของ เครื่องปลายทาง ดังแสดงในรูปที่ 2.21



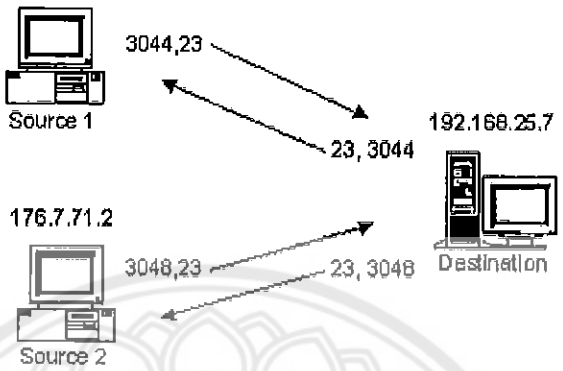
รูปที่ 2.21 header ของ port [3]

TCP หรือ UDP จะดูที่ข้อมูลหมายเลข Port ใน header เพื่อพิจารณาว่า Application ใดที่ต้องการข้อมูลนั้นๆ หมายเลข Port ทั้งต้นทางและปลายทางจำเป็นต้องมีเพื่อให้ เครื่องปลายทางมีความสามารถที่จะรัน process มากกว่า 1 process ในเวลาเดียวกัน

ตามที่ได้กล่าวในข้างต้น "Well know Ports" เป็น Port ที่ค่อนข้างมาตรฐาน ทำให้เครื่องที่อยู่ไกลออกไป (Remote Computer) สามารถรู้ได้ว่าจะติดต่อกับทาง Port หมายเลขอะไรสำหรับ Service เฉพาะนั้นๆ อย่างไรก็ตามยังมี Port อีกประเภทที่เรียกว่า Dynamically Allocated Port ซึ่ง Port ประเภทนี้ไม่ได้ถูก assign ไว้แต่เดิม แต่จะถูก assign เมื่อจำเป็น Port ประเภทนี้ให้ความสะดวกและความคล่องตัวสำหรับระบบที่มีผู้ใช้หลายๆคนพร้อมๆกัน ระบบจะต้องให้ความมั่นใจว่าจะไม่ assign หมายเลข Port ซ้ำกัน

ยกตัวอย่าง สมมติว่ามีผู้ใช้ต้องการใช้ Service Telnet ทางเครื่องต้นทางจะทำการ assign ให้หมายเลข Dynamic Port (เช่น 3044) โดยที่หมายเลข Port ปลายทางคือ 23 เครื่องจะ assign หมายเลข Port ปลายทางจะเป็น 23 เพราะว่า เป็น Well Known Port สำหรับการ Service Telnet จากนั้นเครื่องปลายทางจะทำการตอบรับกลับโดยใช้ Port หมายเลข 23 เป็นหมายเลขต้นทาง และ

หมายเลข Port 3044 เป็นหมายเลข ปลายทาง กลุ่มของหมายเลข Port และ หมายเลข IP เราเรียกว่า Socket ซึ่งจะเป็นตัวบ่งชี้ที่เฉพาะเจาะจงสำหรับ Network process หนึ่งเดียวที่มีอยู่ในทั้งระบบ Internet คู่ของ Socket ที่ประกอบด้วย Socket หนึ่งตัว สำหรับต้นทาง และอีกตัว สำหรับปลายทาง สามารถใช้บรรยายถึงคุณลักษณะของ Connection oriented protocols



รูปที่ 2.22 การใช้ Service Telnet จากเครื่องปลายทางเครื่องเดียวกัน [3]

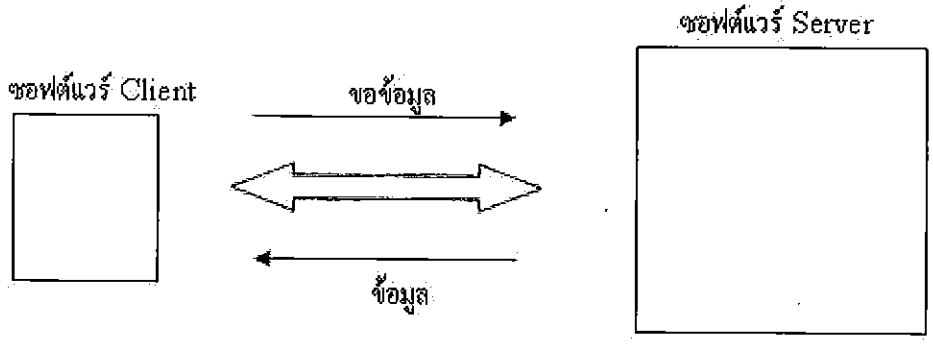
ถ้าผู้ใช้คนที่ 2 ต้องการใช้ Service Telnet จากเครื่องปลายทางเครื่องเดียวกัน ผู้ใช้นั้นก็จะได้รับการ assign หมายเลข Port ต้นทางที่แตกต่างกันออกไป โดยมีหมายเลข Port ปลายทางเหมือนกันกับผู้ใช้คนแรกดังรูปที่ 2.22 จะเห็นได้ว่าการจับคู่ของหมายเลข Port และหมายเลข IP ทั้งต้นทางและปลายทางสามารถทำให้แยกความแตกต่างของ Internet connection ระหว่างเครื่องต้นทางและเครื่องปลายทางได้Active และ Passive Ports การใช้การติดต่อด้วย TCP สามารถกระทำได้สองวิธีคือ Passive และ Active Connection Passive connection คือ การติดต่อที่ Application process สั่งให้ TCP รอหมายเลข Port สำหรับการร้องขอการติดต่อจาก Source Host เมื่อ TCP ได้รับการร้องขอแล้วจึงทำการเลือกหมายเลข Port ให้ แต่ถ้าเป็นแบบ Active TCP ก็จะให้ Application process เป็นฝ่ายเลือกหมายเลข Port ให้เลย

2.3 ระบบ Client – Server [4]

ระบบ Client – Server เป็นสถาปัตยกรรมซอฟต์แวร์ที่ระบบซอฟต์แวร์ได้รับการออกแบบให้แยกออกเป็น 2 ส่วน ส่วนแรกเรียกว่า Client และอีกส่วนเรียกว่าส่วน Server ส่วนที่เรียกว่า Client ต้องสื่อสารติดต่อกับ Server โดยที่ซอฟต์แวร์ Client จะขอใช้ข้อมูลจากซอฟต์แวร์ส่วน Server ซอฟต์แวร์ส่วน Server จะตอบสนองโดยการดึงข้อมูลจากฐานข้อมูล แล้วส่งไปยังส่วน Client เพื่อประมวลผลต่อไป ดังรูปที่ 2.23

ห้องสมุดคณะวิศวกรรมศาสตร์

4900003

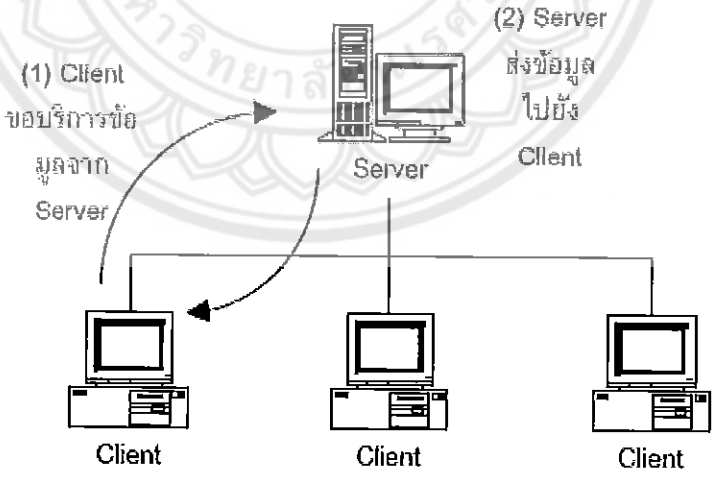


รูปที่ 2.23 การแยกซอฟต์แวร์ส่วน Client และ Server [4]

ส่วน Client กับ Server สามารถอยู่เครื่องเดียวกันได้แต่ถ้าอยู่คนละเครื่องจะต้องเชื่อมผ่านเครือข่าย มีการเชื่อมต่อเครือข่ายทั้งหมด 3 วิธี

1. LAN (Local Area Network)
2. WAN (Wide Area Network)
3. Internet

ปกติแล้วข้อมูลจะอยู่ข้าง Server ในฐานข้อมูล ซึ่งอาจเป็นฐานข้อมูล MS Access , MS SQL , Oracle Informix , DB2, ทางฝั่ง Client จะส่งคำสั่ง SQL ขอใช้ข้อมูลไปยัง Server แล้ว Server จะตีความหมายของคำสั่ง SQL Statement แล้วดึงข้อมูลจากฐานข้อมูลส่งไปยัง Client



รูปที่ 2.24 การให้บริการแบบ Client/Server [7]

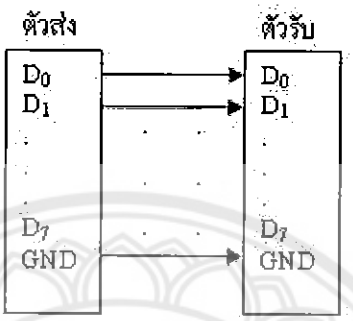
2.4 พื้นฐานในการรับส่งข้อมูล [5]

การรับส่งข้อมูลในระบบคอมพิวเตอร์โดยทั่วไปจะหมายถึง การรับส่งข้อมูลเป็นจำนวนไบต์ๆ ให้กับอุปกรณ์ที่เกี่ยวข้องกับคอมพิวเตอร์ ซึ่งอาจแบ่งประเภทของการรับส่งข้อมูลได้ 2 แบบ

1. การรับส่งข้อมูลแบบขนาน (Parallel)

2. การรับส่งข้อมูลแบบอนุกรม (Serial)

การรับส่งข้อมูลแบบขนานจะเป็นการรับส่งข้อมูล จำนวน 1 ไบต์ ออกไปทางพอร์ตในเวลาเดียวกันในระบบคอมพิวเตอร์ 1 ไบต์ จะมีจำนวน 8 บิตคือ D0-D7 ถ้ามีการส่งข้อมูลส่งข้อมูลแบบขนานจะใช้สายสัญญาณอย่างน้อย 9 เส้นคือ สาย Data 8เส้น และสายกราวด์ 1เส้นดังรูปที่ 2.25



รูปที่ 2.25 การรับส่งข้อมูลแบบขนาน [5]

การรับส่งข้อมูลแบบอนุกรม คือการรับส่งข้อมูลที่ละบิต จนครบ 1 ไบต์ คือ D0-D7 อาจจะส่ง D0 ก่อนแล้วตามด้วย D1 ไปจนถึง D7

ข้อดีของการส่งข้อมูลแบบขนาน

- สามารถส่งข้อมูลได้เร็ว คือส่งเพียงครั้งเดียวแต่ได้ข้อมูลครบ 1 ไบต์

ข้อดีของการส่งแบบอนุกรม

- เมื่อต้องการส่งข้อมูลเป็นระยะไกลๆ จะช่วยประหยัดสายสัญญาณเนื่องจากจะใช้สายสัญญาณอย่างน้อยเพียง 2 เส้น คือ สายสัญญาณกับสายกราวด์

2.4.1 ความรู้เบื้องต้นเกี่ยวกับมาตรฐาน RS-232 [6]

โดยปกติ เครื่องไมโครคอมพิวเตอร์จะมีพอร์ตที่เป็นอนุกรมชื่อว่า RS-232 อยู่ในตัวมันเองอยู่แล้วซึ่งพอร์ต RS-232 จะทำหน้าที่ในการรับ และส่งข้อมูลในแบบอนุกรมที่เรียกว่า Universal Asynchronous Adapter สาเหตุที่มีการเรียกชื่อเรียกว่า RS-232 ก็เนื่องมาจาก สมาคมผู้ผลิตอุปกรณ์อีเล็กทรอนิกส์ ของสหรัฐอเมริกา หรือ EIA (RS-232 : Recommended Standard Number 232,EIA: Electronic Industry Association) ได้กำหนดมาตรฐานของอุปกรณ์สื่อสารแบบอนุกรมเอาไว้ภายใต้ชื่อว่า RS-232 (ความจริงแล้วมาตรฐานของรับส่งข้อมูลแบบอนุกรมมีหลายมาตรฐาน แต่ที่นิยมกันมากที่สุดสำหรับเครื่องไมโครคอมพิวเตอร์

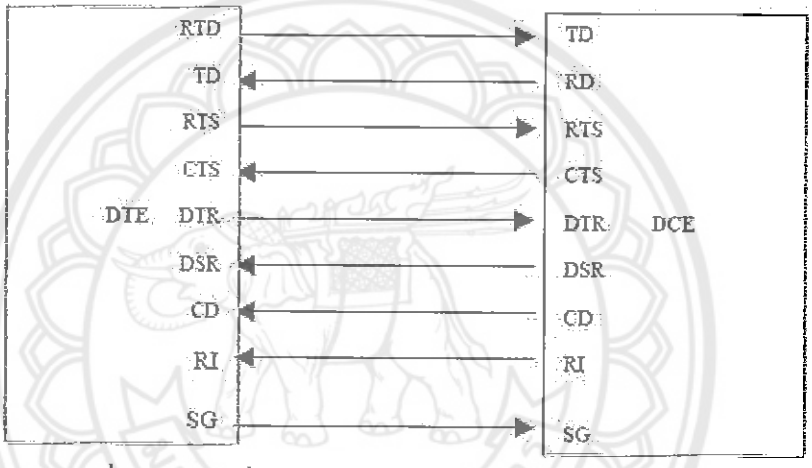
2.4.2 มาตรฐาน RS-232C [6]

มาตรฐาน RS-232 ได้จัดพิมพ์ขึ้นเมื่อ ค.ศ. 1969 RS ย่อมาจาก Recommended Standard ส่วน 232 คือหมายเลขบ่งบอกมาตรฐานตัวนี้ และ C เป็นหมายเลขฉบับสุดท้ายของมาตรฐานตัวนี้

จุดประสงค์ของมาตรฐาน RS-232 ก็เพื่อบรรยายคุณลักษณะของการเชื่อมต่ออุปกรณ์รับส่งข้อมูล (DCE: Data Communication Equipment) กับอุปกรณ์สำหรับสื่อสารข้อมูล (DTE: Data Terminal Equipment) สำหรับผู้ที่ใช้คอมพิวเตอร์ทั่วไป DTE ก็หมายถึงตัวไมโครคอมพิวเตอร์ ส่วน DCE หมายถึง โมเด็ม (modem) และอุปกรณ์อื่นๆ เช่น เครื่องพิมพ์ที่รับสัญญาณแบบอนุกรม อาจจะเป็นไปได้ทั้ง DTE และ DCT

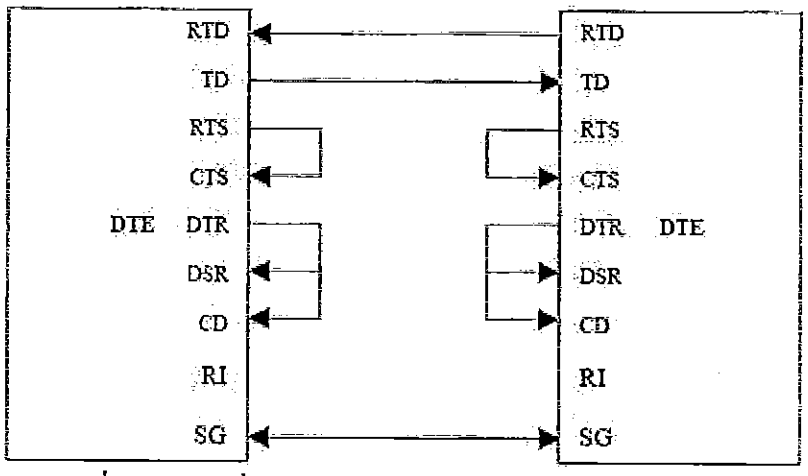
2.4.3 การจัดขาสัญญาณของ RS-232C [6]

มีลักษณะเชื่อมต่อ 2 แบบด้วยกัน การเชื่อมต่อกันระหว่างอุปกรณ์ DTE (DTE : Data terminal Equipment) เช่นเครื่องคอมพิวเตอร์กับอุปกรณ์ DEC (DCE : Data Circuit Terminal) เช่น โมเด็ม แสดงดังรูปต่อไปนี้



รูปที่ 2.26 การเชื่อมต่อกันระหว่างอุปกรณ์ DTE กับ DCE [6]

การเชื่อมต่อกันระหว่างอุปกรณ์ DTE (Data terminal Equipment) กับอุปกรณ์ DTE แสดงดังรูปต่อไปนี้



รูปที่ 2.27 การเชื่อมต่อกันระหว่างอุปกรณ์ DTE กับ DTE [6]

2.4.4 องค์ประกอบการรับส่งข้อมูลแบบอนุกรม [6]

การสื่อสารอนุกรมที่นำมาใช้กับคอมพิวเตอร์นั้นจะเป็นการสื่อสารอนุกรมในรูปแบบของอะซิงโครนัส คือ ต้องการใช้สายสัญญาณเส้นเดียวทำหน้าที่ในการรับส่งทั้งส่วนที่เป็นข้อมูลและส่วนที่ใช้ในการควบคุมการส่งข้อมูล ข้อมูลที่อ่านได้แต่ละบิตจากการส่งแบบอนุกรมต้องถูกแยกว่าใช้สำหรับวัตถุประสงค์ใดโดยจะแบ่งออกเป็น 4 ส่วน

1. บิตเริ่มต้น (Start Bit) จะใช้ใส่ที่บิตเริ่มต้นเสมอเพื่อเตือนว่าอุปกรณ์ฝ่ายรับข้อมูลกำลังจะมาถึงมีขนาด 1 บิต

2. บิตข้อมูล (Data Character) การส่งบิตข้อมูลจะส่งไปเป็นกลุ่มโดยจะส่งเป็น 7 หรือ 8 บิต

3. พาริตี้บิต (Parity Bit) เป็นบิตที่ใช้ตรวจสอบความถูกต้องของข้อมูลที่ส่งไป หลักการกำหนดพาริตี้บิตแบ่งออกเป็น พาริตี้คู่ (Even Parity) พาริตี้คี่ (Odd Parity) ไม่มีพาริตี้บิต (None)

4. บิตจบ (Stop Bit) เป็นบิตที่ถูกส่งมาปิดท้ายข้อมูล ในการเขียนโปรแกรมสำหรับติดต่อกับพอร์ตอนุกรมโดยใช้ MS Comm ที่เป็น API ที่ใช้งานในโปรแกรม วิชวลเบสิก 6.0 ซึ่งเป็นส่วนที่เข้ามาจัดการกับการทำงานในการส่งข้อมูลระหว่างพอร์ตอนุกรมซึ่งการทำงาน จะไม่ทำงานติดต่อกับพอร์ตอนุกรมโดยตรง แต่จะส่งคำสั่งผ่าน API ที่ติดต่อกับพอร์ตอนุกรมอีกครั้ง

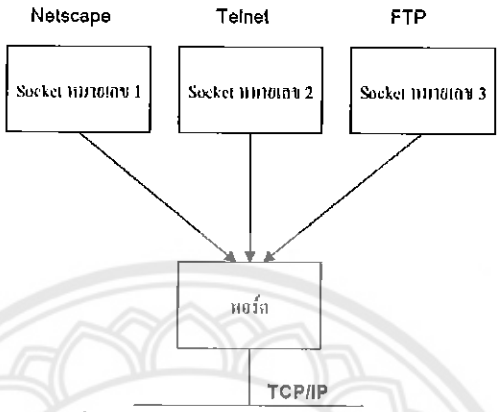
Property ที่สำคัญของ MS Comm

Comport	ใช้ในการกำหนดหมายเลขของพอร์ตอนุกรมที่ต้องการจะเลือกใช้งาน
Setting	ใช้ในการกำหนดอัตราบอดเรด (Baud Rate) หรือความเร็วในการส่งข้อมูลมีหน่วยเป็นบิตต่อวินาที , พาริตี้, จำนวนของบิตข้อมูล และจำนวนบิตปิดท้าย
PortOpen	ใช้สำหรับเปิดและปิดการใช้งานของพอร์ตอนุกรม
InBuffersize	เป็นการกำหนดขนาดของบัพเฟอร์ในการับข้อมูลเข้ามาในพอร์ตอนุกรม
OutBuffersize	เป็นการกำหนดขนาดในการนำของบัพเฟอร์ในการส่งออกไปนอกพอรต์อนุกรม
Inputlen	เป็นการกำหนดค่าของข้อมูลที่อ่านจากบัพเฟอร์ภาครับ
InputMode	เป็นการกำหนดค่าชนิดของข้อมูลที่รับเข้ามา
Input	ใช้ในการอ่านค่าจากพอร์ตอนุกรม
Output	ใช้ในการส่งข้อมูลออกจากพอร์ตอนุกรม
EOFEnable	เป็นการบอกการสิ้นสุดของไฟล์

2.5 การเขียนโปรแกรมบนอินเตอร์เน็ตด้วย Winsock (Windows Socket) [7]

การติดต่อสื่อสารระหว่างเครื่องคอมพิวเตอร์บนระบบอินเตอร์เน็ต เช่นการTelnet เข้าไปสำหรับในระบบปฏิบัติการ Unix ซึ่งระบบต้องสามารถรองรับการทำงานแบบ MultiUser ได้ก็คือระบบจะต้องมีการติดต่อสื่อสารกับ User ได้หลาย User พร้อมๆกัน หรือเมื่อมีการใช้งานโปรแกรม

ที่เกี่ยวกับอินเตอร์เน็ต เช่น Netscape Telnet FTP ซึ่งโปรแกรมต่างๆเหล่านี้ต้องทำงานแยกกันโดยอิสระ ดังนั้นจึงได้เกิด Winsock ขึ้นมา นั่นก็คือ Winsock สามารถที่จะสร้างช่องทางสื่อสารขึ้นมาได้หลายๆ ช่องทางด้วยกัน (ขึ้นอยู่กับเวอร์ชั่นของ Winsock) และแต่ละช่องทางสื่อสารสามารถที่จะส่งข้อมูลได้โดยไม่ขึ้นกับช่องทางสื่อสารอื่นๆ ดังรูปที่ 2.28

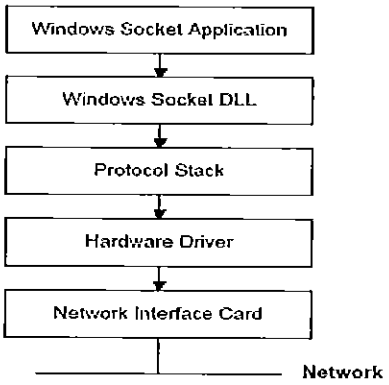


รูปที่ 2.28 การสื่อสารผ่าน Socket [7]

เทคโนโลยี ซ็อกเก็ต (Socket) เป็นเทคโนโลยี ที่มีใช้งานอยู่ในปัจจุบันบนอินเทอร์เน็ตมีความเสถียรภาพ Socket ที่มีการใช้งานอยู่บน MS Windows จะถูกเรียกว่า Windows Socket หรือจะเรียกสั้นๆว่า วินซ็อก (Winsock) ความหมายของซ็อกเก็ต คือ เครื่องมือของโปรแกรมที่จะถูกใช้ในการส่ง และ รับข้อมูลผ่านทางหมายเลขพอร์ตของ TCP/IP ที่ได้กำหนดขึ้นมาโดยโปรแกรมจะสร้างซ็อกเก็ตได้ตามที่ต้องการเพื่อใช้ในการทำงาน แต่ 1 ซ็อกเก็ต จะต้องทำงานกับ 1 พอร์ตของ TCP/IP เท่านั้น

2.5.1 ระดับของ Winsock ในส่วนของ TCP/IP [7]

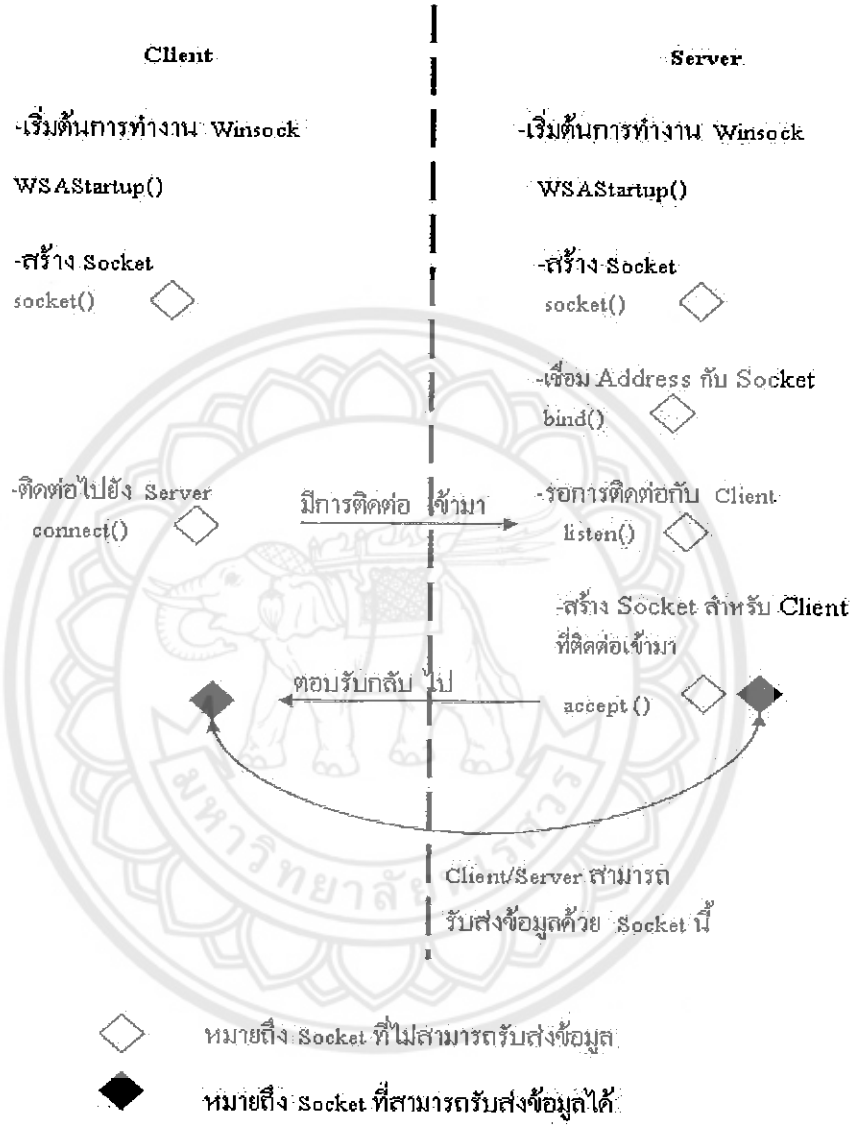
Winsock ทำหน้าที่เป็นตัวกลางในการรับส่งข้อมูลจาก Application กับ TCP/IP จากนั้น TCP/IP จึงส่ง ข้อมูลลงไปบนระบบอินเตอร์เน็ต ดังรูปที่ 2.29



รูปที่ 2.29 ระดับของ Winsock [7]

2.5.2 การสร้าง Client/Server โดยใช้ Winsock [7]

การสร้าง Client/Server โดยใช้ฟังก์ชั่นของ Winsock จะมีขั้นตอนการทำงานซึ่งแสดงได้ดังรูปที่ 2.31



รูปที่ 2.30 การสร้าง Client/Server โดยใช้ฟังก์ชันของ Winsock [7]

2.5.3 การเขียนโปรแกรมติดต่อเครือข่ายโดยใช้ Winsock Control [7]

การใช้งาน Winsock ในการเขียนโปรแกรม ซึ่งจะมีโหมดการทำงานของ Winsock Control ซึ่งทำงานโปรโตคอล TPC/IP แบ่งโหมดการทำงานออกเป็น 2 โหมดด้วยกันคือ

- TCP (Tranmission Control Protocol) เป็นการทำงานในลักษณะ Connection-Based เปรียบเสมือนการทำงานของโทรศัพท์ที่จะมีการตรวจสอบการทำงานทั้ง 2 ฝั่งมีการโต้ตอบระหว่าง

กัน ทำให้เกิดการสื่อสารที่มีประสิทธิภาพ เหมาะสำหรับการส่งข้อมูลที่ต้องการความถูกต้องของข้อมูล เช่น การส่งภาพหรือเสียงผ่านเครือข่าย ซึ่งจะมีการตรวจสอบความถูกต้องของข้อมูลที่มีการส่งถึงกัน ทำให้มีการใช้ทรัพยากรของระบบสูง

- UDP (User Datagram Protocol) มีลักษณะการทำงานตรงกันข้ามกับ TCP นั่นคือ มีการทำงานในลักษณะ Connectionless เปรียบเสมือนการกระจายเสียงวิทยุ เหมาะสำหรับการส่งข้อมูลที่ไม่สำคัญมากนัก เช่น การส่งข้อมูลสถานการณ์ทำงาน ซึ่งมักจะเป็นข้อมูลขนาดเล็ก ทำให้ใช้ทรัพยากรของระบบน้อยกว่า TCP

Property ที่สำคัญของ Winsock Control

Protocol : เป็นการเลือก Protocol สำหรับใช้งาน

LocalPort : เป็นการเลือกหมายเลขพอร์ตที่ของคอมพิวเตอร์ที่จะใช้งานกับ Winsock

RemoteHost : เป็นการกำหนดชื่อของคอมพิวเตอร์ที่เราจะทำการติดต่อด้วยอาจจะเป็น IP Address หรือ เป็นชื่อคอมพิวเตอร์ที่ง่ายต่อการจดจำ

RemotePort : เป็นการกำหนดหมายเลขพอร์ตของคอมพิวเตอร์ที่เราจะทำการติดต่อด้วย

ByteRecive : เป็นจำนวนข้อมูลที่รับเข้ามาในบัพเฟอร์

Method สำคัญของ Winsock Control

Listen : Method ที่ใช้สร้าง Socket ทำให้คอมพิวเตอร์เครื่องอื่นสามารถติดต่อเข้ามาได้

Connect : Method ที่ใช้สร้างการติดต่อแบบ Socket ไปยังเครื่องคอมพิวเตอร์อื่นๆ โดยจะต้องทำการระบุ Socket Address

Accept : Method ที่ใช้รับ Request จากคอมพิวเตอร์ที่ทำการติดต่อเข้ามา

SendData : Method ที่ใช้ส่งข้อมูลไปยังเครื่องอื่น

GetData : Method ที่ใช้รับข้อมูลจากบัพเฟอร์เข้ามาเก็บไว้มนตัวแปรที่กำหนดไว้ โดยสามารถที่จะกำหนดชนิดของตัวแปรและความยาวของข้อมูลได้

Close : Method ที่ใช้ยกเลิกการติดต่อแบบ Socket

Event สำคัญของ Winsock Control

ConnectionRequest : เป็นเหตุการณ์ที่เกิดขึ้นเมื่อ คอมพิวเตอร์เครื่องอื่นมีการ Request เข้ามาซึ่งจะมีการกำหนด ID ให้กับแต่ละ Request ที่เข้ามา

DataArrival : เป็นเหตุการณ์ที่เกิดขึ้นเมื่อมีชุดข้อมูลใหม่เข้ามาเก็บในบัพเฟอร์ ซึ่งเราจะสามารถทำการตรวจสอบขนาดของข้อมูลได้จาก Property ByteRecive

SendProgress : เป็นเหตุการณ์ที่เกิดขึ้นเมื่อมีการส่งข้อมูลระหว่างกันซึ่งจะมีพารามิเตอร์แสดงข้อมูลที่ทำการส่งแล้วข้อมูลที่ยังคงเหลืออยู่

SendCommplete : เป็นเหตุการณ์ที่เกิดขึ้นเมื่อมีการส่งข้อมูลเสร็จสิ้นแล้ว

Error : เป็นเหตุการณ์ที่เมื่อเกิดความผิดพลาดขึ้นจะแสดงหมายเลขของความผิดพลาดนั้นออกมา คำอธิบาย และรายละเอียดอื่นๆสำหรับจัดการกับความผิดพลาดที่เกิดขึ้น

2.6 ภาพดิจิตอล [8]

โดยปกติแล้วสายตาของบุคคลทั่วไปจะมองเห็นภาพทิวทัศน์วิวต่างๆ เป็นลักษณะอนาล็อก ซึ่งสามารถอธิบายได้ด้วยคณิตศาสตร์ที่มีตัวแปรแบบนับได้อย่างต่อเนื่อง แต่เครื่องคอมพิวเตอร์จะใช้เลขฐานสองเป็นหลักในการคำนวณ เมื่อนำภาพนั้นมาแปลงเข้าสู่เครื่องคอมพิวเตอร์ ก็จะกลายเป็น ภาพดิจิตอล (Digital Image)

2.6.1 กระบวนการทำภาพดิจิตอล

มีสองเหตุผลใหญ่ๆ ที่ต้องการทำกระบวนการทำภาพดิจิตอล คือ เพื่อปรับปรุงภาพดิจิตอลให้มองเห็นได้ง่ายขึ้น และเพื่อปรับปรุงภาพให้หุ่นยนต์ตีความหมาย หรือเข้าใจจดจำรูปร่างลักษณะได้อย่างแม่นยำ ตัวอย่างเช่น การจดจำตัวอักษร หรือ Optical Character Recognition (OCR) ที่สามารถจดจำตัวอักขระได้ถึง 99.9% การปรับปรุงภาพให้ใช้พื้นที่เก็บน้อยลง การตรวจสอบลายพิมพ์มือของแต่ละบุคคล เป็นต้น

กระบวนการทำภาพดิจิตอล คือ การที่นำเอาภาพดิจิตอลเข้ามาทำกระบวนการโดยการใส่ฟังก์ชัน อัลกอริทึมต่างๆ เข้าไป ก็จะได้เอาท์พุตเป็นภาพดิจิตอล ที่ตรงตามแนวความคิดของการทำกระบวนการภาพดิจิตอล

2.6.2 เกรย์สเกล

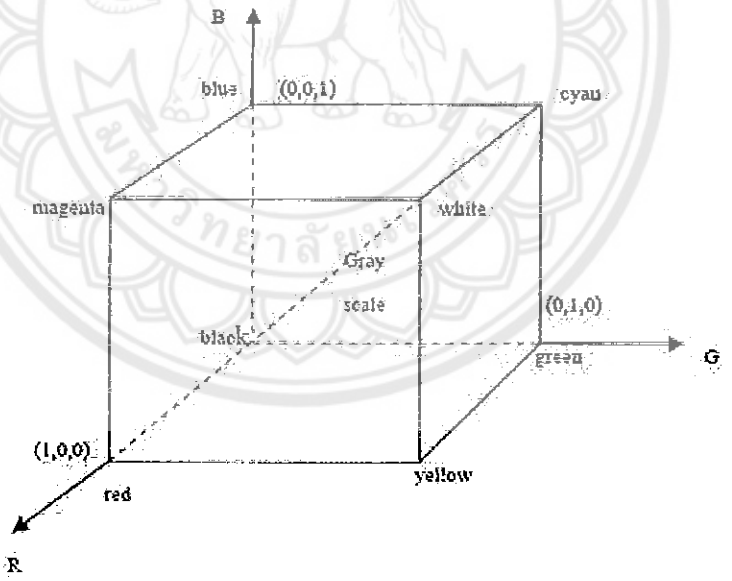
เกรย์สเกล หมายถึง ความแตกต่างของระดับความเข้มแสง โดยเกรย์สเกลหนึ่งๆ อาจจะแบ่งออกเป็น 13, 20 และ 9 ระดับ โดยระดับที่ว่านี้ก็คือ ระดับของสีเทา ในภาพหนึ่งๆ ถ้าต้องการจะแบ่งระดับความเข้มแสงหรือระดับสีเทาให้มีหลายๆ ค่า นั้นมีความจำเป็นอย่างยิ่งจะต้องเพิ่มจำนวนบิตที่แสดงค่าพิกเซล ตัวอย่าง เช่น ภาพที่มีระดับสีเทา 4 ระดับจะต้องถูกแทนด้วยเลขฐานสอง จำนวน 2 บิตด้วย ถ้าต้องการภาพที่มีระดับสีเทา 16 ระดับ จะต้องถูกแทนด้วยเลขฐานสองจำนวน 4 บิต และถ้าต้องการภาพที่มีระดับสีเทา 256 ระดับต้องแทนด้วยเลขฐานสองจำนวน 8 บิต เป็นต้น

จำนวนระดับสีเทาที่ต้องการนี้ก็คือ ค่าเลขยกกำลัง 2 นั่นเอง ซึ่งค่าต่ำสุดหรือ 0 จะแทนสีดำหรือไม่มีความสว่างเลย และค่าที่มากก็คือค่าที่น้อยกว่าจำนวนระดับสีเทาอยู่ 1 เช่นค่า 15 ในระบบที่มีระดับสีเทา 16 ระดับ จะมีสีขาวหรือสว่างมากที่สุด เป็นต้น

2.7 พื้นฐานและระบบสีโมเดล RGB [8]

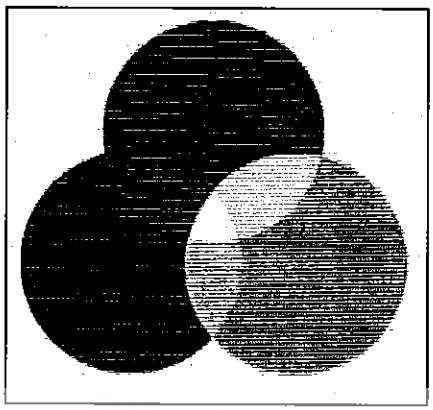
ในโมเดลนี้ สีแต่ละสีจะอยู่ในรูปของสีปฐมภูมิ (แดง, เขียว, น้ำเงิน) โมเดลนี้มีโครงสร้างอยู่ในระบบของพิกัดคาร์ทีเซียน (Cartesian coordinate) มีลักษณะเป็นทรงลูกบาศก์ ดังรูปที่ 2-3 ค่าสี แดง, เขียว, น้ำเงิน จะอยู่ที่มุมทั้งสามบนเส้นทแยงมุมตรงข้ามกัน และค่าสีคราม, มาเจนตา, เหลือง จะอยู่ที่มุมทั้งสามในลักษณะเดียวกัน ส่วนสีดำอยู่ที่จุดกำเนิด สีขาวอยู่ที่มุมที่มีระยะทางไกลที่สุดจากแหล่งกำเนิด ค่าของระดับสีเทา (Gray Scale) จะอยู่บนเส้นที่เชื่อมระหว่างสีดำกับขาว ค่าสี คือ จุดที่อยู่ บนผิวหรือในลูกบาศก์ถูกกำหนดค่าโดยเวกเตอร์ที่ชี้ออกจากจุดกำเนิด เพื่อความสะดวกเราจะสมมุติให้ค่าสีถูกนอมอลไลซ์ (Normalize) ทั้งสามสี โดยให้มีค่าอยู่ในช่วง 0 ถึง 1 ลูกบาศก์ที่แสดงในรูปที่ 2 จึงเป็นลูกบาศก์หนึ่งหน่วย

ภาพในโมเดล RGB ประกอบด้วยภาพสามระนาบที่เป็นอิสระจากกัน สำหรับแต่ละสีปฐมภูมิ เมื่อป้อนเข้าไปในมอนิเตอร์ที่เป็นแบบ RGB ภาพทั้งสามสีจะรวมตัวกันที่จอภาพกลายเป็นภาพสีผสม ดังนั้นการใช้โมเดล RGB ในการประมวลผลภาพนั้นจะสมเหตุสมผลเมื่อภาพถูกแยกออกโดยธรรมชาติให้อยู่ในเทอมของทั้งสามสี กล้องภาพสีส่วนใหญ่ที่ให้ภาพสีดิจิตอลจะอยู่ในรูปแบบของโมเดล RGB ดังนั้นโมเดลนี้จึงเป็นโมเดลที่สำคัญมากในการประมวลผลภาพ



รูปที่ 2.31 โมเดล RGB [8]

2.7.1 โหมดสีแบบ RGB [8]



รูปที่ 2.32 โหมดสี RGB [8]

การผสมสีบนคอมพิวเตอร์นั้นอาศัย primary hues 3 สีคือ สีแดง สีเขียว และ น้ำเงิน หรือที่เรา เรียกว่า RGB colors (Red-Green-Blue) นั้นเอง ความเหมือนจริงของสีคอมพิวเตอร์นั้น ขึ้นอยู่กับในหนึ่งจุด (pixel) ของการแสดงผลนั้นใช้ระดับของสี หรือ color depths ว่ามีค่าเท่าไหร่ เช่นถ้าสี RGB มี color depths เป็น 8 planes นั่นคือ เราใช้ 8 บิตเก็บข้อมูลหนึ่งสี หมายความว่า เฉพาะ primary hues เช่น สีแดง สีเดียวก็จะมีความเป็นแดง อยู่ถึง $2^8 = 256$ ระดับ เมื่อเราผสมสี หนึ่งสี จากแดง-เขียว-น้ำเงิน (RGB) เราต้องใช้สีแดง กี่ส่วนจาก 0 ถึง 255 ส่วน ใช้เขียว กี่ส่วนจาก 0 ถึง 255 และเช่นเดียวกัน สีน้ำเงินกี่ส่วน จาก 0 ถึง 255 สีที่เกิดขึ้นก็จะเกิดจากการผสมของสีทั้งสาม ในอัตราส่วนต่างๆ กัน ตัวอย่าง สีเหลืองธรรมดา เกิดจากการผสมสี แดง 255 ส่วน สีเขียว 255 ส่วน และสีน้ำเงิน 0 ส่วน

ซึ่งระดับของสีแดง มีถึง 256 ระดับ สีเขียว 256 ระดับ สีน้ำเงิน 256 ระดับ ดังนั้น RGB ทั้งหมดใช้ 24 บิต (8+8+8) ในการแสดงสี RGB ของหนึ่งจุด (pixel) ซึ่งสามารถแสดงสีได้มากถึง 256x256x256 เท่ากับ 16.7 ล้านสี ตารางข้างล่างแสดงถึง จำนวนสีที่เป็นไปได้ ซึ่งขึ้นกับจำนวนบิตที่ใช้กำหนดระดับของสี

ตารางที่ 2.1 แสดงจำนวนสีที่เป็นไปได้ซึ่งขึ้นกับจำนวนบิต

จำนวนบิตที่ใช้เก็บสีต่อหนึ่งจุด	Color Mode Name	จำนวนสีที่แสดงได้
1	Black and White	2
4	16-Color (EGA)	16
8	Pseudo Color	256
16	Hi-Color	65,536
24	True Color	16,777,216

2.8 การสื่อสารข้อมูลแบบไร้สาย [10]

การสื่อสารข้อมูลแบบไร้สาย จะใช้อากาศเป็นตัวกลางของการสื่อสาร ลักษณะของการสื่อสารข้อมูลประเภทนี้มีดังนี้

2.8.1 สัญญาณวิทยุ (Radio wave)

มีการส่งข้อมูลเป็นสัญญาณคลื่นวิทยุไปในอากาศไปยังตัวรับสัญญาณจึงทำให้สภาพแวดล้อมสามารถรบกวนได้ในช่วงที่อากาศไม่ดี การส่งสัญญาณวิธีนี้จะช่วยส่งข้อมูลได้ในระยะทางไกล หรือสภาพภูมิประเทศที่ไม่เอื้ออำนวยในการใช้สายส่งข้อมูล สัญญาณวิทยุสามารถแบ่งตามช่วงความถี่ได้ดังนี้

ตารางที่ 2.2 ช่วงความถี่ของสัญญาณวิทยุ [8]

ความถี่		ย่านความถี่	ตัวอย่างการใช้งาน
VLF	Very low frequency	3 KHz – 30 KHz	การสื่อสารใต้น้ำ (submarne communic ation)
LF	Low frequency	30 KHz –300 KHz	สัญญาณนำร่องในการเดินเรือ
MF	Medium frequency	300 KHz –3 MHz	วิทยุ AM
HF	High frequency	3 MHz –30 MHz	วิทยุสื่อสาร
VHF	Very high frequency	30 MHz –300 MHz	สัญญาณโทรทัศน์ช่อง 3,5,7,9,11 วิทยุ FM, วิทยุสายการบิน
UHF	Ultrahigh frequency	300 MHz -3 GHz	สัญญาณโทรทัศน์ช่อง ITV, โทรศัพท์มือถือ
SHF	Superhigh frequency	3 GHz -30 GHz	สัญญาณไมโครเวฟภาคพื้นดิน และภาคดาวเทียม เรดาร์
EHF	Extremely high frequency	30 GHz –300 GHz	สัญญาณไมโครเวฟผ่านดาวเทียม เรดาร์

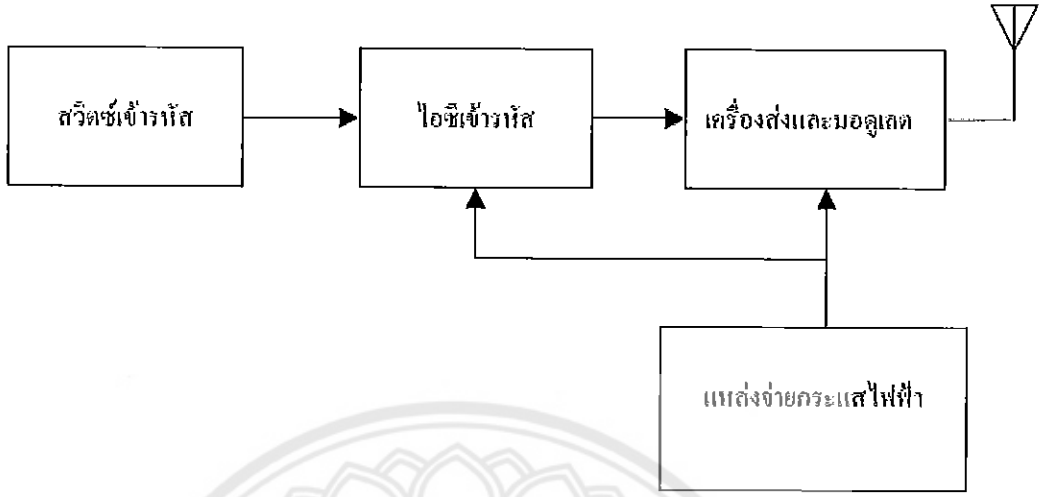
1. การสื่อสารไร้สายด้วยคลื่นวิทยุความถี่สูงระยะไกล

ในส่วนของโครงการนี้ได้ใช้สื่อในการสื่อสารโดยใช้คลื่นวิทยุ ความถี่สูงระยะไกลในการส่งนั้นแบ่งออกเป็น 3 ประเภทใหญ่ๆขึ้นกับกำลังการส่งของสัญญาณที่ใช้ Federal Communications Commission หรือ (FCC) เป็นผู้แยกระดับพลังงานของสัญญาณสำหรับเครื่องรับและเครื่องส่ง ดังตารางที่2.3

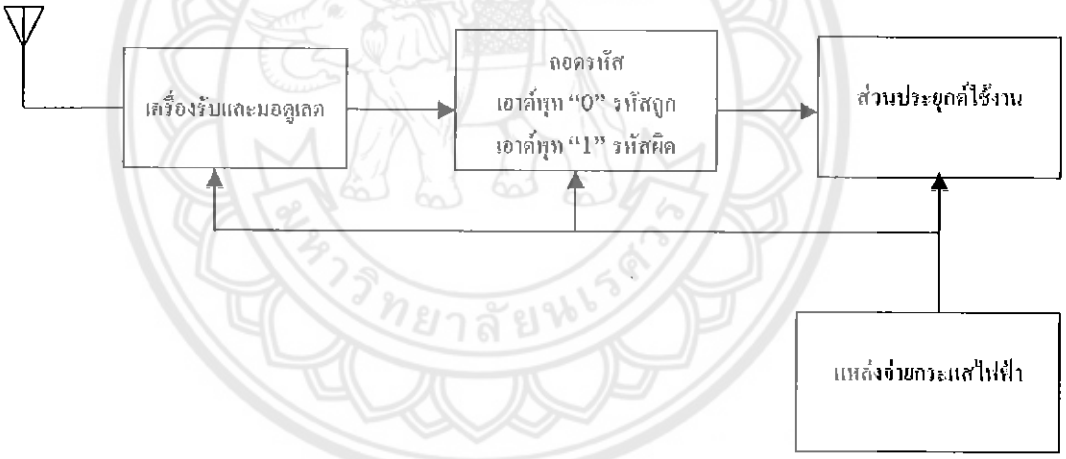
ตารางที่ 2.3 ชนิดของการสื่อสารด้วยคลื่นวิทยุ [10]

Characteristics	Type of Wireless Signal			
	Very Low Power RF	Moderate Power RF	High Power RF	Infrared
Distance	1-10 ft	100 ft to a few mile	A few mile to worldwide	10-50 ft
Data rate (bits per second)	100 bps to 10 kbps	10 kbps to 200 kbps	10 kbps to Mbps	100 bps to 10kbps
License Requirements	None(must comply with FCC Regulations)	None(must comply with FCC Regulations and Operate in selected frequency ranges)	Must be licensed; Design must comply with FCC regulations	None
Operating Power source	Small battery, long life	Large rechargeable battery or power line	Large rechargeable battery or power line	Small battery, long life
Typical operating frequency	Unassigned; usually 100kHz to 350 MHz	Assigned to industrial, scientific, and medical bands,908-928 MHz,2400-2450 MHz	Assigned (often to a specific frequency); depends on the service used, usually VHF or UHF radio	980- μm Wavelength (near infrared)
Modulation	PCM or FM	Spread spectrum	Advanced FM	PCM
Open loop or closed loop	Open (one-way transmission)	Closed	Closed	Both

การสื่อสารโดยใช้คลื่นวิทยุ ความถี่สูงระยะไกล มีหลักการและทฤษฎี ดังรูป 2.33



รูปที่ 2.33 บล็อกไดอะแกรมแสดงหลักการทำงานทางภาคส่งของระบบคลื่นวิทยุ [10]

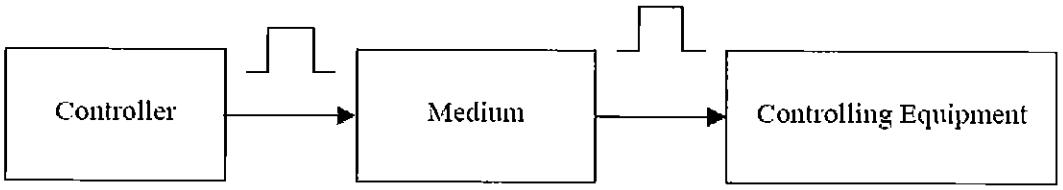


รูปที่ 2.34 บล็อกไดอะแกรมแสดงหลักการทำงานทางภาครับของระบบคลื่นวิทยุ [10]

2. ความรู้พื้นฐานเกี่ยวกับอุปกรณ์ส่งคลื่นวิทยุ (Radio Controller) [8]

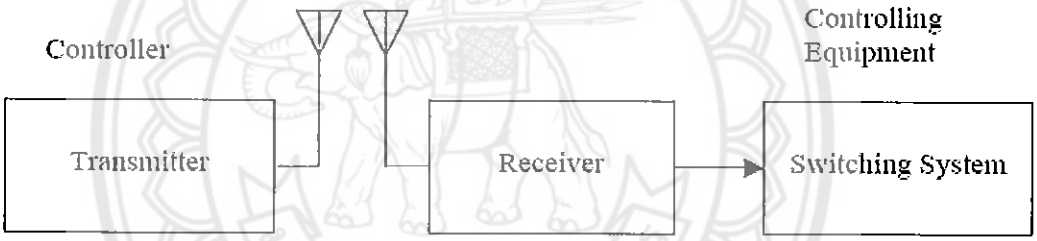
อุปกรณ์ส่งคลื่นวิทยุเป็นตัวควบคุมทางไฟฟ้าหรืออิเล็กทรอนิกส์ ซึ่งมีอุปกรณ์ควบคุมอยู่ระยะไกลจากตัวควบคุม และตัวถูกควบคุมจะต้องมีตัวกลางเป็นตัวส่งผ่านสัญญาณในการควบคุม เช่น สัญญาณจากคลื่นวิทยุ คลื่นเสียงไมโครเวฟ หรือคลื่นอื่นๆ ซึ่งตัวกลางนี้อาจเป็นอากาศ หรือสายส่งสัญญาณ จากรูปที่ 2.34 เป็นการแสดงถึงระบบของอุปกรณ์ส่งสัญญาณวิทยุ ซึ่งเป็นตัวผลิตสัญญาณที่ใช้ในการบังคับตัวถูกควบคุมโดยมีตัวกลางเป็นตัวรับสัญญาณจากอุปกรณ์ส่งสัญญาณวิทยุ ไปยังวงจรที่ถูกควบคุม ในรูปตัวสัญญาณที่ส่งออกมาจากอุปกรณ์ควบคุมจะเป็นในรูปของ

สัญญาณพัลส์ (Pulse) เมื่อสัญญาณพัลส์เข้าไปในตัวอุปกรณ์ที่ถูกควบคุมแล้ว สัญญาณพัลส์ จะเป็นตัวกระตุ้นระบบสวิตซ์ (Switch) ของอุปกรณ์ควบคุม



รูปที่ 2.35 บล็อกไดอะแกรมแสดงรูปแบบการสส่งสัญญาณแบบพัลส์ [10]

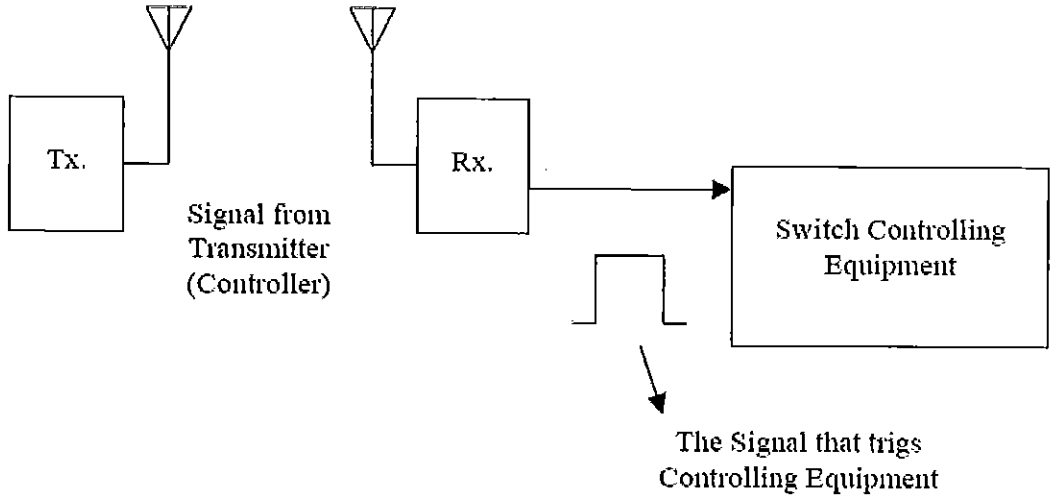
ในทางปฏิบัติ ตัวอุปกรณ์ควบคุมนี้อาจเป็นตัวผลิตสัญญาณพัลส์ ที่ตัวเครื่องส่ง (Transmitter) ตัวกลางอาจจะเป็นอากาศและอุปกรณ์ที่ถูกควบคุมอาจจะเป็นเครื่องรับ (Receiver) และระบบสวิตซ์ดังรูปที่ 2.36



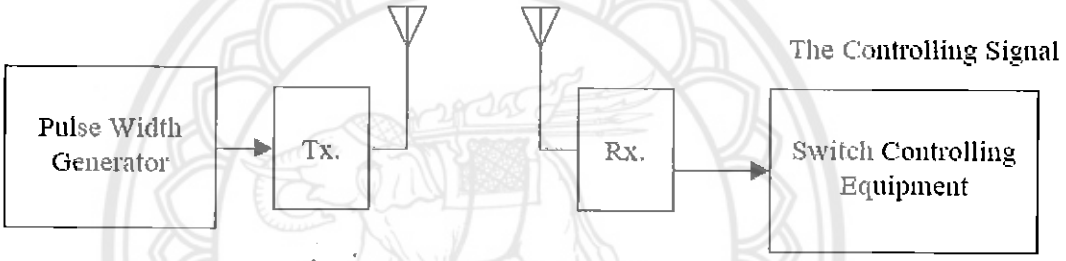
รูปที่ 2.36 แสดงหลักการทำงานของระบบคลื่นวิทยุผ่านทางตัวกลางที่เป็นอากาศ [10]

3. อุปกรณ์ส่งสัญญาณ (Transmitter)

โดยอาศัยหลักการของเครื่องส่งสัญญาณวิทยุ โดยได้ใช้ความถี่บางความถี่ที่สามารถเดินทางในอากาศได้ เช่น MF, RF ไมโครเวฟ (Microwave), อินฟราเรด (Infrared) เป็นต้น เมื่อผลิตความถี่นี้ออกมา แล้วส่งให้เดินทางไปในอากาศ ดังนั้นสำหรับตัวเครื่องส่งแล้ว จะต้องมีตัวผลิตความถี่ที่จะส่งออกอากาศ หรือตัวกำหนดสัญญาณวิทยุและที่ตัวเครื่องรับก็เช่นเดียวกันกับเครื่องรับวิทยุ คือ จะรับคลื่นความถี่ที่ส่งมาทางอากาศนั้นมาแปลความหมาย ซึ่งตัวผลิตความถี่ของเครื่องส่งจะผลิตความถี่เท่าไรนั้น ขึ้นอยู่กับจุดประสงค์ในการใช้งาน เพราะในย่านความถี่ต่างๆ ก็มีทั้งข้อดีและข้อเสียต่างกันออกไป ในการส่งสัญญาณทางอากาศของเครื่องส่งสัญญาณวิทยุนั้น อาศัยความถี่ที่ผลิตได้จากเครื่องส่งเป็นเพียงคลื่นพาหะ (Carrier) เพื่อนำสัญญาณพัลส์ไปเป็นตัวกระตุ้นให้อุปกรณ์ควบคุมสวิตซ์ทำงานดังรูป

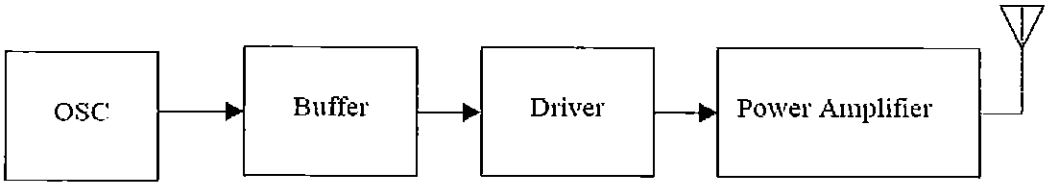


รูปที่ 2.37 การใช้สัญญาณพัลส์กระตุ้นให้ตัวอุปกรณ์ควบคุมสวิตช์ทำงาน [10]



รูปที่ 2.38 วงจรกำหนดความกว้างของพัลส์ [10]

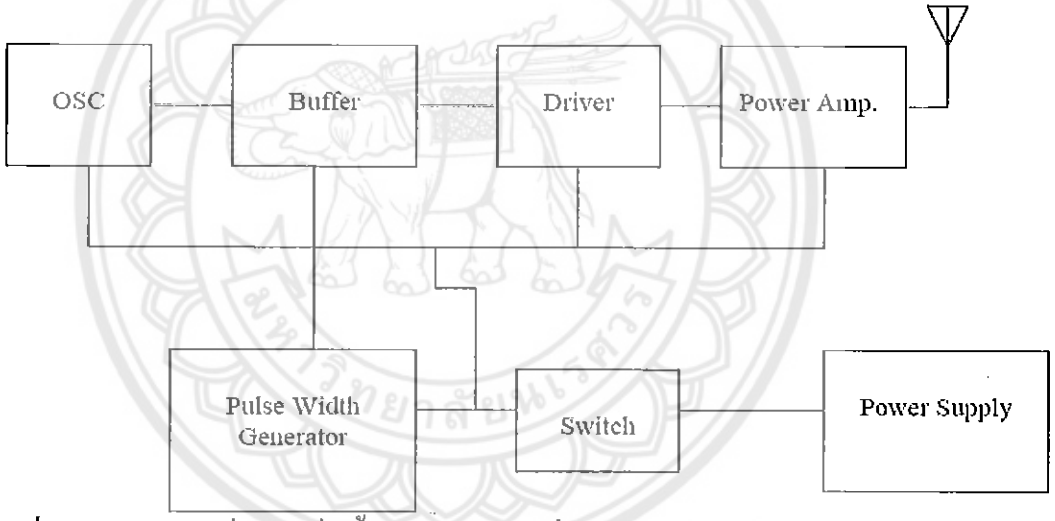
สรุปแล้วในภาคเครื่องส่งก็สามารถแบ่งลักษณะของเครื่องส่งออกได้เป็นสองชนิดคือ ชนิดที่มีเครื่องส่งอย่างเดียว และชนิดที่มีสัญญาณพัลส์กับเครื่องส่งประกอบกัน ดังรูปที่ 2.37 และ 2.38 ซึ่ง Tx คือ เครื่องส่ง และ Rx เครื่องรับและอุปกรณ์กำหนดความกว้างของสัญญาณพัลส์ (Pulse Width Generator) คือ สัญญาณพัลส์นั่นเอง สำหรับบล็อกไดอะแกรมของเครื่องส่งประกอบด้วย ภาคความถี่ (Oscillator) ภาคที่กั้น OSC บัพเฟอร์ (Buffer), ภาคขับสัญญาณ (Driver), และภาคขยายสัญญาณ (Power Amplifier) ดังรูปที่ 2.39 ซึ่งเป็นตัวอย่างของเครื่องส่งอย่างง่ายๆ สำหรับวงจรที่มีการกำหนดความกว้างของสัญญาณ ก็เพิ่มอุปกรณ์กำหนดความกว้างของสัญญาณพัลส์เข้ามาอีก วงจรผลิตความถี่ (OSC) ผลิตความถี่ออกมา เพื่อที่จะผสมกับความกว้างของสัญญาณ เรียกการผสมสัญญาณนั้นว่าตัวปรับสัญญาณ (Modulator) แล้วสัญญาณจะผ่านทางบัพเฟอร์ ภาคขับสัญญาณ จนกระทั่งถึงภาคขยายกำลังเพื่อขยายสัญญาณให้มีขนาดสูงขึ้น จากนั้นส่งออกอากาศดังรูป



รูปที่ 2.39 วงจรผลิตความถี่ [10]

4. สัญญาณจากสัญญาณรหัสกว้าง (Pulse Width)

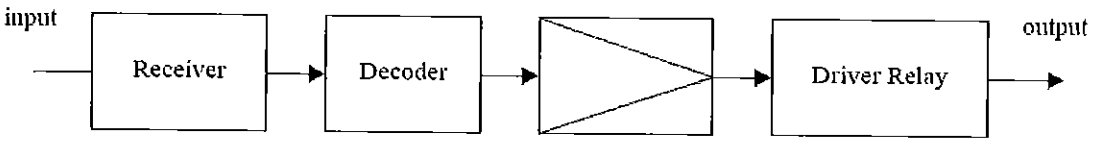
สัญญาณจากวงจรผลิตความถี่ จะเป็นตัวกระตุ้นตัวถูกควบคุม (Controlling Equipment) ทำงานสัญญาณจากสัญญาณรหัสกว้าง จะเป็นตัวถูกกระตุ้นให้ตัวถูกควบคุม (Controlling Equipment) ทำงาน ในทางปฏิบัติสัญญาณจากวงจรผลิตความถี่ และสัญญาณรหัส ก็คือความถี่ทั้งคู่ และอาจจะมีความแตกต่างกันบ้าง โดยทั่วไปแล้วความถี่ของภาคกำหนดความกว้าง ของสัญญาณจะออกมาในรูปของสัญญาณกว้าง



รูปที่ 2.40 วงจรส่งคลื่นวิทยุที่มีทั้งวงจรผลิตความถี่และวงจรกำหนดความกว้างสัญญาณ [10]

5. อุปกรณ์รับสัญญาณ (Receiver)

อุปกรณ์รับสัญญาณของวงจรวิทยุ Radio Control จะเป็นตัวรับสัญญาณจากอาการที่ถูกส่งมาโดย Transmitter ดังนั้นสำหรับภาครับจึงต้องมีมีความสามารถในการรับสัญญาณความถี่ที่ถูกส่งโดยอุปกรณ์ส่งสัญญาณเท่านั้น ทั้งนี้เพื่อป้องกันการรบกวนสัญญาณจากที่อื่น สำหรับอุปกรณ์รับส่งสัญญาณ ส่วนประกอบใหญ่ของวงจรประกอบด้วย ภาครับและวงจรถอดรหัสสัญญาณ ภาครับจะรับสัญญาณที่ส่งมาจากอุปกรณ์ส่งสัญญาณ เข้าไปในเครื่องและตัวถอดรหัสสัญญาณจะทำการถอดสัญญาณที่ได้รับออกมาเป็นสัญญาณพัลส์เพื่อที่จะใช้งานต่อไป ก่อนที่จะนำไปใช้งานสัญญาณที่ได้ ควรจะได้รับการขยายก่อนเพื่อให้สัญญาณมีขนาดที่ใหญ่ขึ้น (สัญญาณแรงมากขึ้น)



รูปที่ 2.41 วงจรแปลงสัญญาณคลื่นวิทยุ [10]

2.8.2 ไมโครเวฟภาคพื้นดิน (Terrestrial microwave) [10]

ไมโครเวฟภาคพื้นดิน เป็นการสื่อสารโดยใช้สื่อนำข้อมูลแบบไร้สายอีกประเภทหนึ่ง การสื่อสารประเภทนี้จะมีการส่งสัญญาณไมโครเวฟที่อยู่ห่างๆ กันทำการส่งข้อมูลไปในอากาศไปยังเสารับข้อมูล ในกรณีระยะเสาห่างกานมากจะต้องใช้สถานีทวนสัญญาณ (repeater station)เพื่อการส่งสัญญาณเป็นช่วงๆ การสื่อสารประเภทนี้สามารถส่งข้อมูลปริมาณมากได้ แต่ในบางครั้งอาจถูกสภาพแวดล้อมรบกวนได้เช่นกัน โดยเฉพาะในช่วงฝนตกหรือมีพายุ จะทำให้การส่งข้อมูลทำได้ไม่ดีนัก

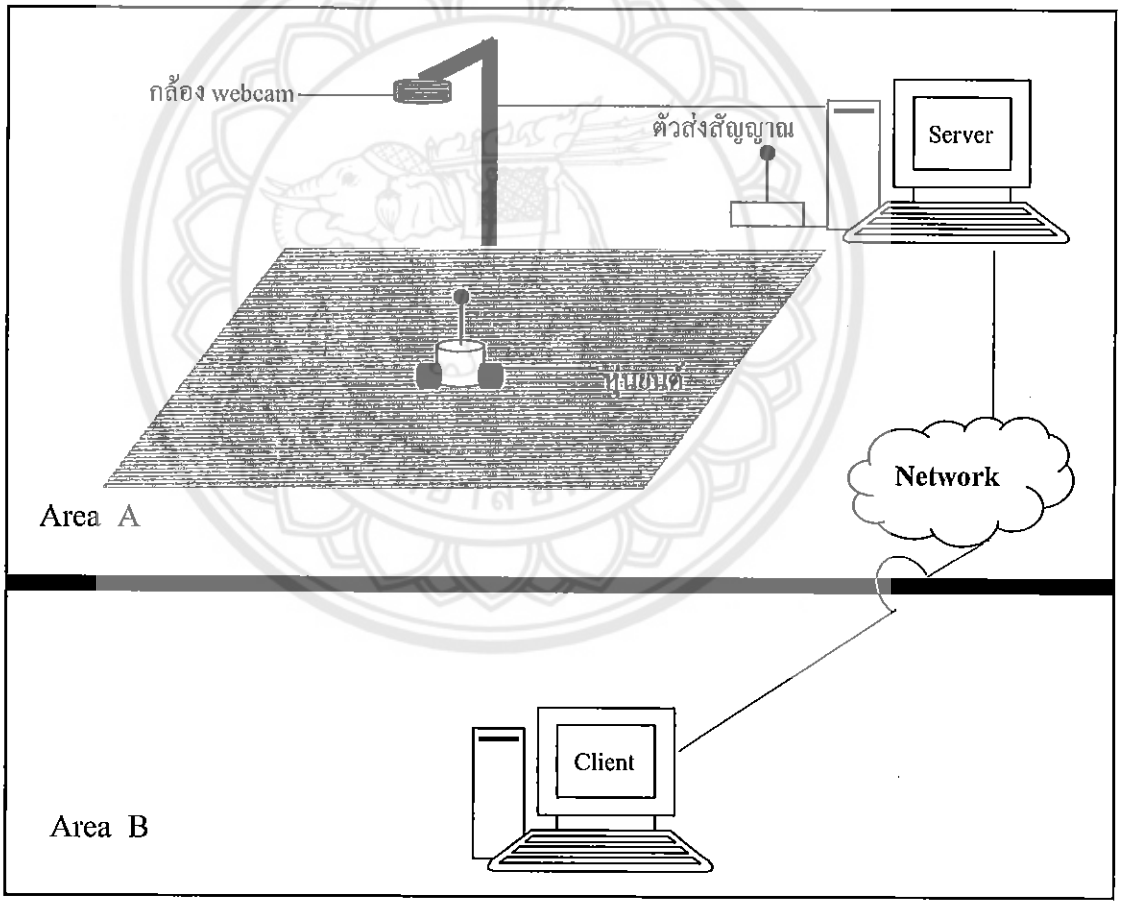
2.8.3 การสื่อสารผ่านดาวเทียม (Satellite communication) [10]

การสื่อสารผ่านดาวเทียม เป็นการสื่อสรจากพื้นโลกที่มีการส่งสัญญาณข้อมูลไปยังดาวเทียม โดยดาวเททียมจะทำหน้าที่เป็นสถานีทวนสัญญาณ เพื่อจัดส่งสัญญาณต่อไปยังสถานีภาคพื้นดินอื่นๆ ระยะทางจากโลกถึงดาวเทียมประมาณ 22,000 ไมล์ ซึ่งเป็นระยะทางที่ไกลมาก ทำให้ข้อมูลที่ส่งไปยังดาวเทียมเกิดความล่าช้าขึ้นได้ โดยเฉลี่ยความล่าช้าที่เกิดขึ้นมีค่าประมาณ 2 วินาที การส่งข้อมูลวิธีนี้จะทำให้ส่งข้อมูล ที่มีระยะทางไกลมากๆ ได้ การสื่อสารผ่านดาวเทียมนิยมใช้สำหรับการสื่อสารระหว่างประเทศ

บทที่ 3
วิธีการดำเนินการ

3.1 การควบคุมหุ่นยนต์โดยควบคุมและแสดงภาพการทำงานผ่านทางเครือข่ายด้วยโปรโตคอล ทีซีพี/ไอพี (TPC/IP Protocol)

การทำงานของหุ่นยนต์จะถูกควบคุมโดยเครื่องที่ทำหน้าที่เป็นแม่ข่ายโดยส่งคำสั่งผ่านพอร์ตอนุกรม RS-232 และส่งคำสั่งผ่านตัวส่งสัญญาณไปยังตัวหุ่นยนต์ ซึ่งรับคำสั่งจากเครื่องลูกข่ายที่ส่งมา ขณะเดียวกันก็รับภาพการแสดงผลการทำงานของหุ่นยนต์จากกล้อง webcam และส่งภาพแสดงการทำงานของหุ่นยนต์กลับไปแสดงผลที่เครื่องลูกข่ายดังแสดงในรูป 3.1

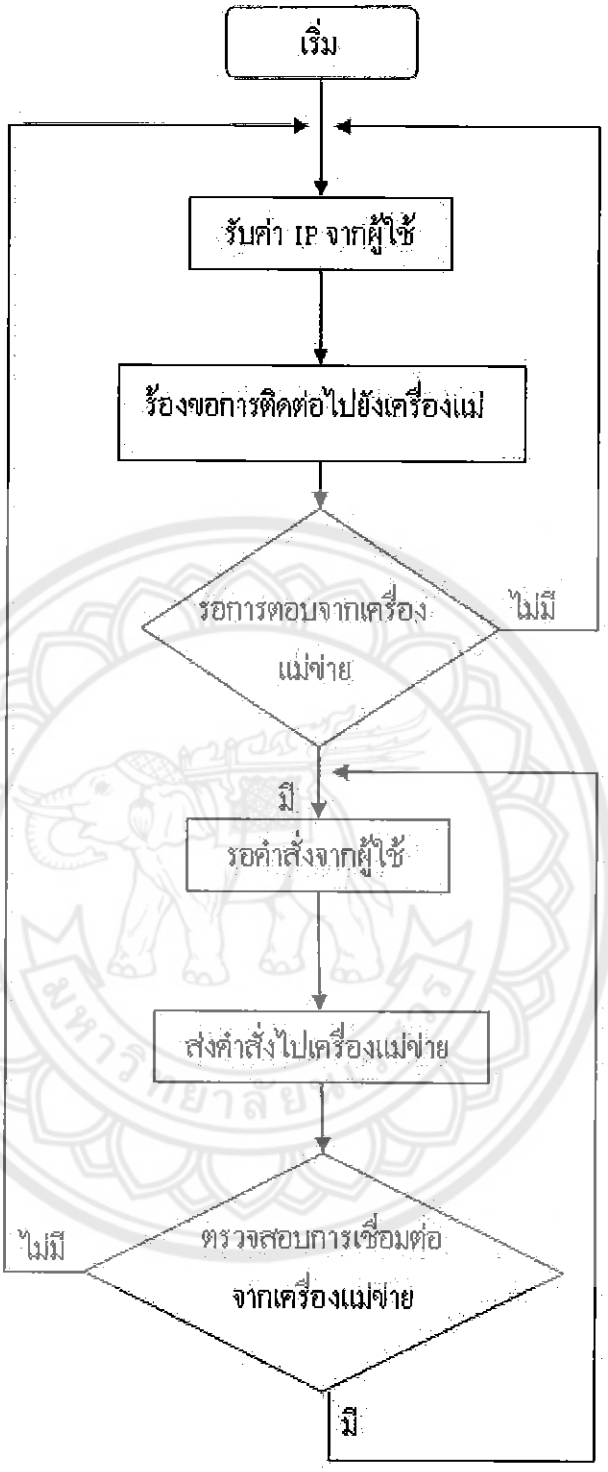


รูป 3.1 แสดงการทำงานการควบคุมหุ่นยนต์

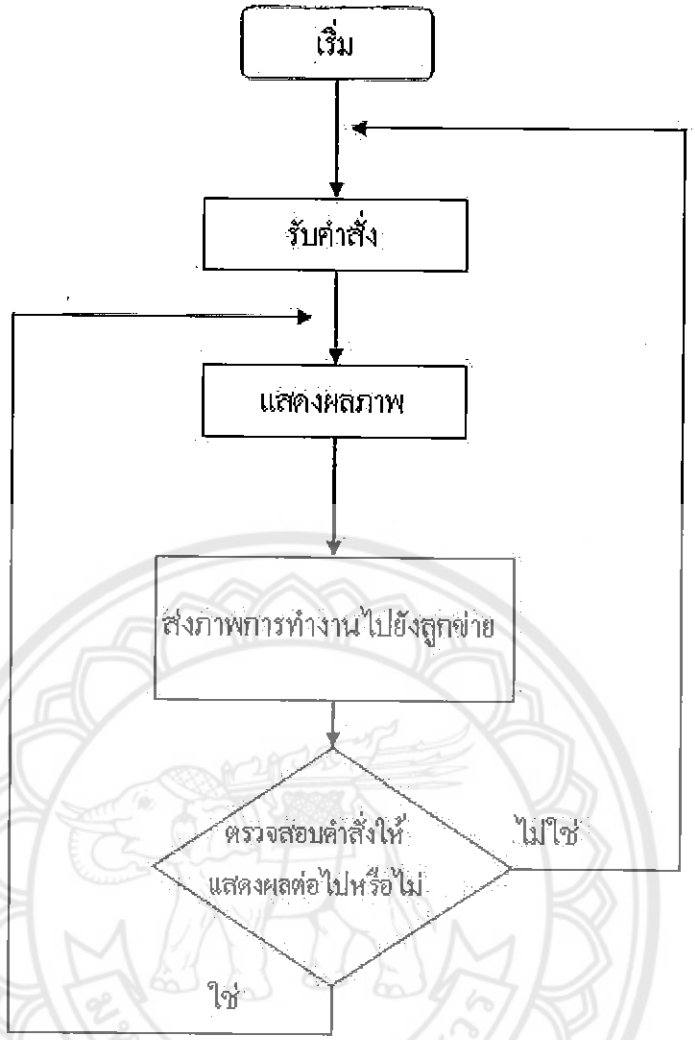
3.2 แผนภาพแสดงการควบคุมหุ่นยนต์ผ่านเครือข่ายด้วยโปรโตคอล ทีซีพี/ไอพี (TPC/IP Protocol)

ในส่วนของโปรแกรมในการทำการควบคุมหุ่นยนต์นั้นจะแบ่งออกเป็น 3 ส่วน คือ โปรแกรมที่ทำหน้าที่เป็นแม่ข่าย เป็นส่วนที่ใช้ในการรับคำสั่งจากเครื่องลูกข่ายและควบคุมการทำงานของหุ่นยนต์ตามที่เครื่องลูกข่ายส่งมา โปรแกรมส่วนที่ทำหน้าที่เป็นลูกข่ายทำหน้าที่ส่งคำสั่งไปยังเครื่องแม่ข่าย และโปรแกรมที่ทำหน้าที่แสดงผล ดังจะได้แสดงการทำงานดังรูปที่ 3.2 รูปที่ 3.3 และรูปที่3.4

รูป 3.2 แผนภาพแสดงการทำงานของโปรแกรมส่วนที่เป็นแม่ข่าย



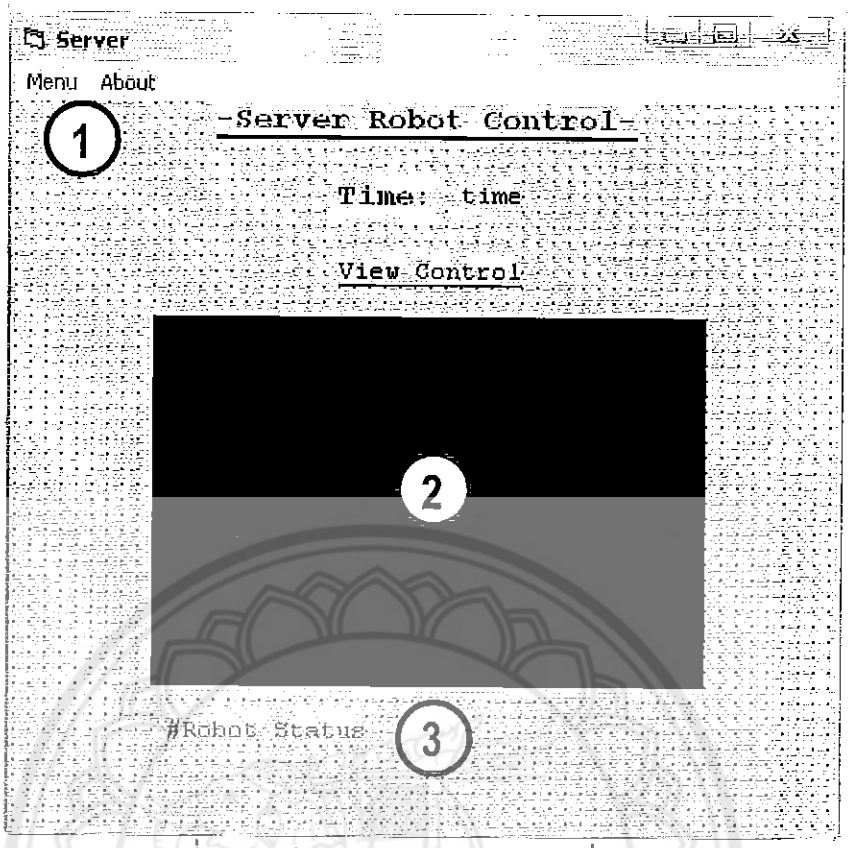
รูปที่ 3.3 แผนภาพแสดงการทำงานของโปรแกรมส่วนที่เป็นลูกข่าย



รูป 3.4 แผนภาพแสดงการทำงานของโปรแกรมส่วนที่แสดงผลภาพ

3.3 การออกแบบโปรแกรมควบคุมหุ่นยนต์ผ่านเครือข่ายด้วยโปรโตคอล ทีซีพี/ไอพี (TPC/IP Protocol)

โปรแกรมควบคุมหุ่นยนต์ผ่านเครือข่าย เขียนด้วยโปรแกรมวิชวลเบสิกเวอร์ชั่น 6.0 เนื่องจากเป็นโปรแกรมที่สามารถใช้งานได้ง่ายและมีเครื่องมือในการสร้างหน้าตาของโปรแกรมให้ง่ายต่อผู้ใช้ ซึ่งมีการนำเอาคอมโพแนนต์ MS Winsock Control ในการเขียนโปรแกรติดต่อกันระหว่างเครือข่ายและ MS Comm ในการเขียนโปรแกรมติดต่อกับพอร์ตอนุกรม โดยโปรแกรมส่วนจะมีอยู่ด้วยกัน 2 โปรแกรม คือ โปรแกรมที่ทำหน้าที่เป็นเครื่องแม่ข่าย และโปรแกรมที่ทำหน้าที่เป็นเครื่องลูกข่าย หน้าตาของโปรแกรมส่วนที่ทำหน้าที่เป็นแม่ข่ายมีหน้าตาดังรูปที่ 3.5



รูปที่ 3.5 หน้าโปรแกรมส่วนทีทำหน้าที่เป็นแม่ข่าย

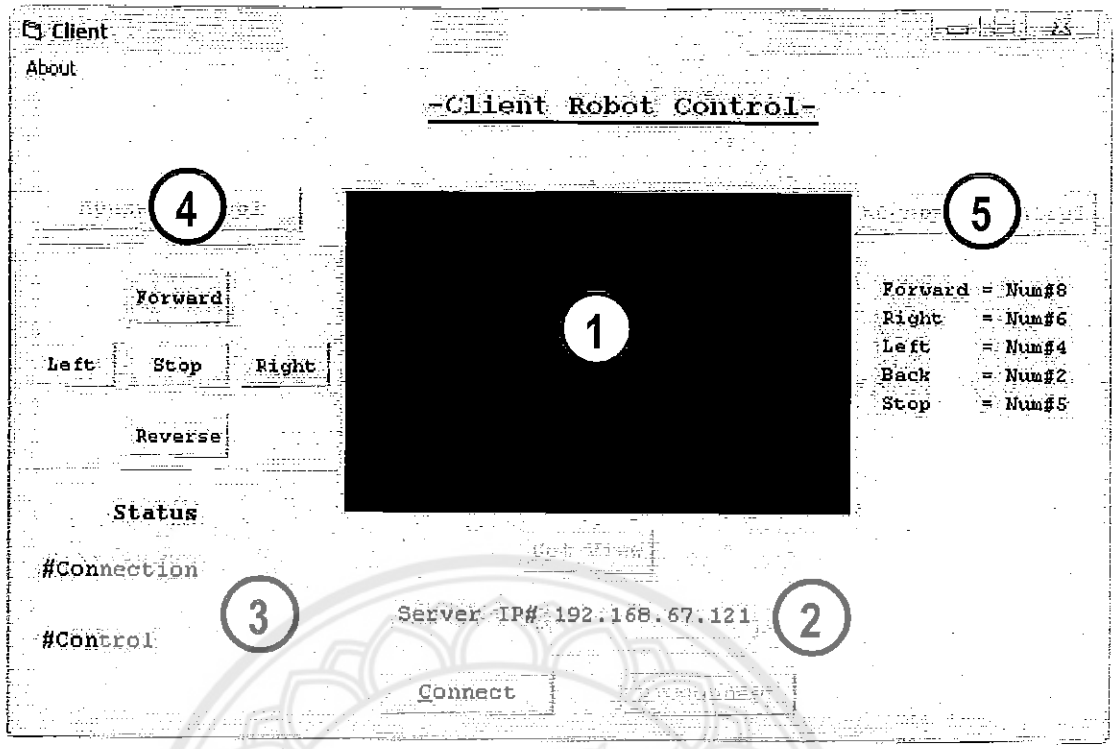
จากรูปการทำงานของโปรแกรมมีส่วนการทำงานได้ดังรูป

ส่วนที่1 Menu เป็นส่วนที่แสดงเมนูการใช้งานโดยจะประกอบไปด้วยเมนูหลัก 2 เมนู และภายในเมนูหลักที่ชื่อว่า Menu จะประกอบด้วยเมนูย่อย 3 เมนูได้แก่

- Listen เป็นเมนูที่สั่งงานให้โปรแกรมส่วนแม่ข่าย ทำการเปิดพอร์ตรอรับการเชื่อมต่อจะโปรแกรมส่วนที่เป็นลูกข่าย
- ShowVideo เป็นเมนูส่วนที่ใช้แสดงผลภาพการทำงานที่เครื่องแม่ข่าย
- Disconnect เป็นเมนูที่ใช้ในการยกเลิกการเชื่อมต่อจากเครื่องลูกข่าย

ส่วนที่ 2 Video Display เป็นส่วนที่ใช้รับภาพจากกล้อง Web Cam ที่แสดงภาพการควบคุมหุ่นยนต์

ส่วนที่ 2 Status เป็นส่วนของโปรแกรมที่ทำหน้าที่รายงานสถานะของหุ่นยนต์ ขณะทำการส่งคำสั่งไปควบคุมการทำงานและโปรแกรมส่วนที่ 2 โปรแกรมส่วนที่ทำหน้าที่เป็นลูกข่ายมีหน้าตาดังรูปที่ 3.6



รูปที่ 3.6 หน้าตาของโปรแกรมส่วนทีทำหน้าที่เป็นลูกข่าย

จากรูปอธิบายการทำงานของของโปรแกรมได้ดังนี้

ส่วนที่ 1 Video Display เป็นส่วนที่แสดงภาพการทำงานของหุ่นยนต์ที่ได้รับจากเครื่องแม่ข่าย โดยจะแสดงผลภาเมื่อกดปุ่ม Get View

ส่วนที่ 2 Connection เป็นส่วนที่ทำหน้าที่ในการสร้างการเชื่อมต่อระหว่างเครื่องลูกข่ายและเครื่องแม่ข่าย โดยจะมีการใช้งานอยู่ คือ

- Server IP จะมีช่องว่างสำหรับเติมที่อยู่ปลายทางซึ่งก็คือหมายเลขเครื่องแม่ข่ายที่จะที่การติดต่อไป
- ปุ่ม Connect เป็นปุ่มที่สั่งให้โปรแกรมส่วนที่เป็นลูกข่ายทำการร้องขอการติดต่อไปยังเครื่องแม่ข่าย
- ปุ่ม Disconnect เป็นปุ่มที่ทำหน้าที่สั่งให้โปรแกรมส่วนที่เป็นลูกข่ายเลิกการติดต่อกับเครื่องแม่ข่าย

ส่วนที่ 3 Status เป็นส่วนที่ทำหน้าที่แสดงสถานะ ของการทำงานของเครื่องลูกข่ายมีการแสดงสถานะดังนี้คือ

- Connection เป็นส่วนที่แสดงสถานการณ์เชื่อมต่อระหว่างเครื่องลูกข่ายกับเครื่องแม่ข่าย
- Control เป็นส่วนที่แสดงการคำสั่งที่ส่งไปยังเครื่องแม่ข่าย

ส่วนที่ 4 Mouse Control Mode เป็นส่วนโปรแกรมที่ใช้เมาส์ในการส่งคำสั่งไปควบคุมหุ่นยนต์สั่งการทำงานเมื่อการทำงานในโหมดนี้ โดยการคลิกที่ปุ่มคำสั่งที่มีอยู่ด้วยกัน 4 ปุ่มคือ

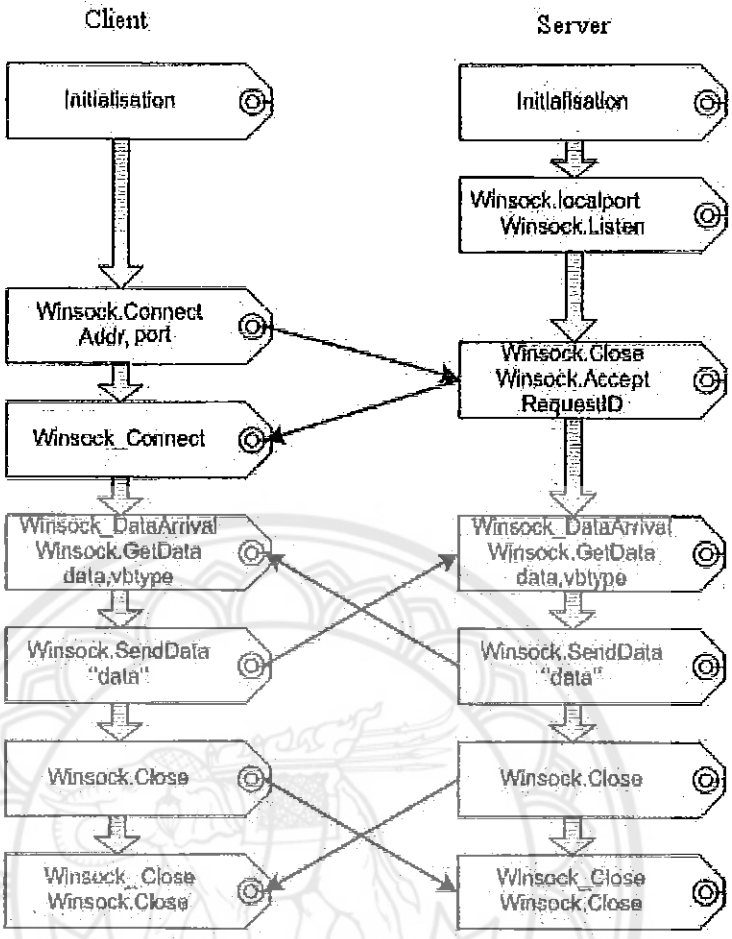
- Forward ปุ่มที่ทำหน้าที่ส่งคำสั่งบังคับให้หุ่นยนต์เดินหน้า
- Reverse ปุ่มที่ทำหน้าที่ส่งคำสั่งบังคับให้หุ่นเดินยนต์ถอยหลัง
- Right ปุ่มที่ทำหน้าที่ส่งคำสั่งบังคับให้หุ่นเดินยนต์เลี้ยวขวา
- Left ปุ่มที่ทำหน้าที่ส่งคำสั่งบังคับให้หุ่นเดินยนต์เลี้ยวซ้าย
- Stop ปุ่มที่ทำหน้าที่ส่งคำสั่งบังคับให้หุ่นเดินยนต์หยุดเดิน

ส่วนที่ 5 Keyboard Control Mode เป็นโปรแกรมส่วนที่ใช้คีย์บอร์ดในการส่งคำสั่งไปควบคุมหุ่นยนต์มีการรับค่าจากคีย์บอร์ดด้วยกัน 4 ค่าเมื่อเลือกใช้หมวดการทำงานนี้ คือ

- Num8 ทำหน้าที่ส่งคำสั่งบังคับให้หุ่นยนต์เดินหน้า
- Num2 ทำหน้าที่ส่งคำสั่งบังคับให้หุ่นเดินยนต์ถอยหลัง
- Num6 ทำหน้าที่ส่งคำสั่งบังคับให้หุ่นเดินยนต์เลี้ยวขวา
- Num4 ทำหน้าที่ส่งคำสั่งบังคับให้หุ่นเดินยนต์เลี้ยวซ้าย
- Num5 ทำหน้าที่ส่งคำสั่งบังคับให้หุ่นเดินยนต์หยุดเดิน

3.4 การเขียนโปรแกรมเพื่อควบคุมการทำงานของหุ่นยนต์ผ่านเครือข่ายด้วยโปรโตคอลทีซีพี/ไอพี (TPC/IP Protocol)

การเขียนโปรแกรมสื่อสารบนเน็ตเวิร์กจะทำการเขียนโปรแกรมสื่อสารโดยใช้โปรโตคอล TCP/IP ในการเขียนโปรแกรมจะทำการเขียนโดยใช้เทคโนโลยีซ็อกเก็ต (socket) ที่มีการใช้งานอยู่บน MS Windows จะถูกเรียกว่า Windows Socket หรือเรียกสั้นๆว่า Winsock ความหมายของซ็อกเก็ต คือ เครื่องมือของโปรแกรมที่จะถูกใช้ในการในการส่ง และ รับ ข้อมูลผ่านทางหมายเลขพอร์ทของ TCP/IP ที่กำหนด โปรแกรมจะสร้างซ็อกเก็ตได้ตามที่ต้องการเพื่อใช้ในการทำงาน แต่ 1 ซ็อกเก็ต จะต้องทำงานกับ 1 พอร์ทของ TCP/IP เท่านั้น โปรแกรมการทำงานจะแบ่งออกเป็น 2 ส่วนคือ ส่วน ส่วนที่เป็นแม่ข่าย และส่วนที่เป็นลูกข่าย ดังที่กล่าวมา โปรแกรมฝั่งลูกข่ายจะสร้างซ็อกเก็ตและทำการกำหนดหมายเลขพอร์ตโดยวิธีการสุ่มหมายเลขขึ้นมา แต่ทางฝั่งแม่ข่ายจะไม่เป็นอย่างนั้น โปรแกรมฝั่งแม่ข่ายจะต้องทำตามข้อกำหนด ที่ได้มีการกำหนดไว้ใน TCP/IP พอร์ต ลูกข่ายจะเริ่มต้นสร้างการติดต่อโดยผ่านเครือข่าย (network sessions) กับเซิร์ฟเวอร์ โดยจะผ่านทาง network protocols ตัวใดตัวหนึ่ง แล้วจะสร้างซ็อกเก็ต และ กำหนดให้มันติดต่อไปยังเซิร์ฟเวอร์ที่ต้องการ เมื่อซ็อกเก็ตได้รับที่อยู่ (address) และ หมายเลขพอร์ต (Port) ของแม่ข่ายแล้วมันก็จะติดต่อไปยังแม่ข่ายนั้นทันที ในส่วนของคอนโทรลที่ชื่อ วินซ็อกแล้ว จะเรียกใช้เมธอด Connect เพื่อทำการติดต่อ เหตุการณ์ต่างจะเกิดขึ้นเมื่อทำการติดต่อดังรูป 3. 7



รูปที่ 3.7 เหตุการณ์ที่เกิดขึ้นเมื่อทำการติดต่อไปที่เครื่องแม่ข่าย

เหตุการณ์ที่เกิดขึ้นหลังจากร้องขอการติดต่อไปยังเครื่องแม่ข่ายเครื่องแม่ข่ายจะทำการตอบรับการติดต่อและทำการเชื่อมต่อกันระหว่างเครื่องแม่ข่ายจากนั้นก็เริ่มทำการสื่อสารกันระหว่างเครื่องแม่ข่ายและเครื่องลูกข่ายจนกว่าจะทำการยกเลิกการติดต่อหรือการติดต่อถูกตัดขาด

3.5 การเขียนโปรแกรมเพื่อการแสดงผลภาพการทำงานของหุ่นยนต์

ในการเขียนโปรแกรมแสดงผลภาพการทำงานของหุ่นยนต์จากกล้อง webcam โดยการใช้งานโมดูล ที่ได้เขียนเอาไว้ซึ่งมีการเรียกการแสดงผลภาพการทำงานของหุ่นยนต์ที่รับจากจากกล้อง webcam และโดยแสดงผลภาพเป็นแบบ RGB ในการส่งการแสดงผลภาพไปยังเครื่องลูกข่าย ภาพจะถูกบีบอัดเป็นรูปแบบ jpeg แล้วส่งไปแสดงผลยังเครื่องลูกข่าย

บทที่ 4
ผลการทดลอง

การทดลองของโครงงานจะทำการทดลองเพื่อทดสอบโปรแกรมที่พัฒนาขึ้นมาว่าสามารถทำเชื่อมต่อจากเครื่องลูกข่ายไปยังเครื่องลูกข่าย และสั่งการทำงานไปควบคุมหุ่นยนต์ผ่านทางเครือข่าย จากนั้นหาข้อผิดพลาดของโปรแกรมเพื่อนำไปแก้ไขต่อไป

4.1 จุดประสงค์ของการทดลองโปรแกรม

1. เพื่อทดสอบโปรแกรมที่พัฒนาขึ้นมาว่าสามารถเชื่อมต่อระหว่างเครื่องข่ายจากเครื่องลูกข่ายไปยังเครื่องแม่ข่ายได้หรือไม่

2. เพื่อทดสอบการสั่งงานจากเครื่องลูกข่ายไปยังเครื่องแม่ข่ายว่าสามารถสั่งงานให้หุ่นยนต์พร้อมกับแสดงผลภาพการทำงานของหุ่นยนต์ว่าผลการทำงานเป็นไปตามที่สั่งคำสั่งไปหรือไม่

3. เพื่อทดสอบหาข้อผิดพลาดที่ได้เกิดขึ้นกับโปรแกรมและนำข้อผิดพลาดมาทำการวิเคราะห์และทางทางดำเนินการแก้ไขข้อผิดพลาดต่อไป

4.2 ขั้นตอนการทดลอง

1. จัดเตรียมการทดลองโดยนำโปรแกรมที่พัฒนาไปติดตั้งที่เครื่องแม่ข่ายและเครื่องลูกข่าย

2. นำอุปกรณ์ที่ใช้ในการทดสอบมาต่อเข้ากับเครื่องแม่ข่ายดังนี้

   2.1 นำกล้องเว็บแคม (webcam) มาต่อเข้ากับเครื่องแม่ข่าย

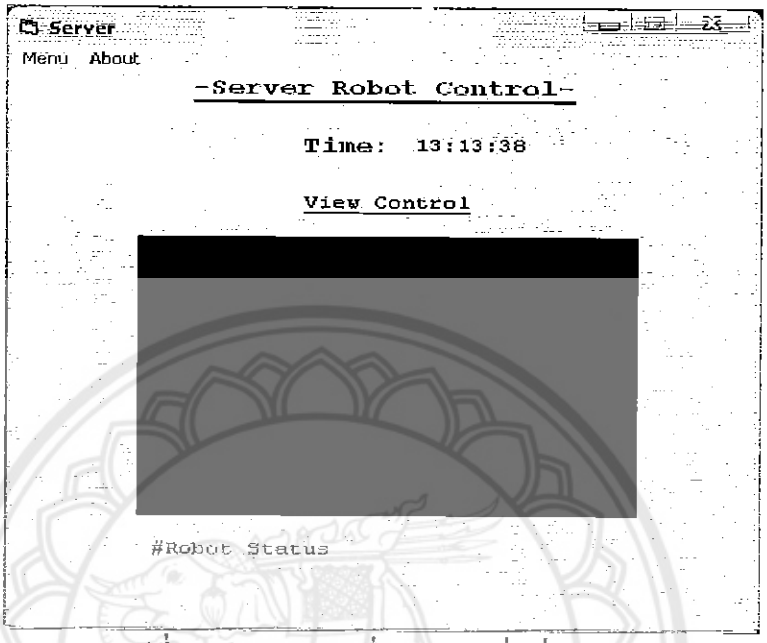
   2.2 นำชุดส่งสัญญาณวิทยุมาต่อเข้ากับเครื่องแม่ข่ายโดยเชื่อมต่อผ่านพอร์ตอนุกรม และชุดรับสัญญาณวิทยุกับหุ่นยนต์

3. ทำการเชื่อมต่อจากเครื่องลูกข่ายไปยังเครื่องแม่ข่าย ทดลองส่งคำสั่งไปควบคุมการทำการทำงานและสังเกตผลการทำงานที่ได้จากภาพที่ส่งมา

4. ตรวจสอบข้อผิดพลาดที่เกิดขึ้นขณะทำการทดลอง จากนั้นนำข้อผิดพลาดมาทำการวิเคราะห์และทำการปรับปรุงแก้ไขข้อผิดพลาดที่เกิดขึ้นต่อไป

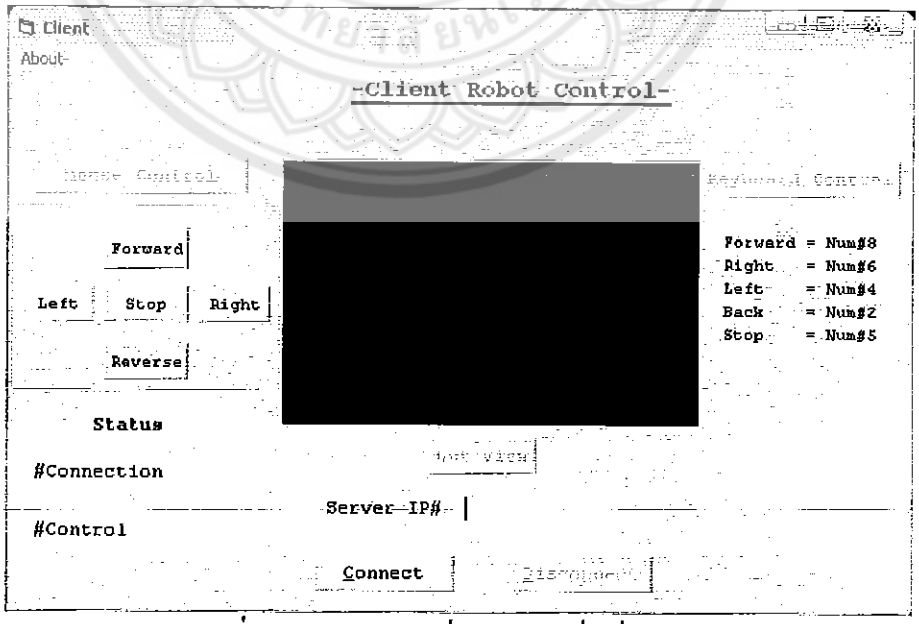
4.3 ผลการทดลอง

เมื่อทำการเปิดโปรแกรมที่ฝั่งเครื่องแม่ข่ายขึ้นมา



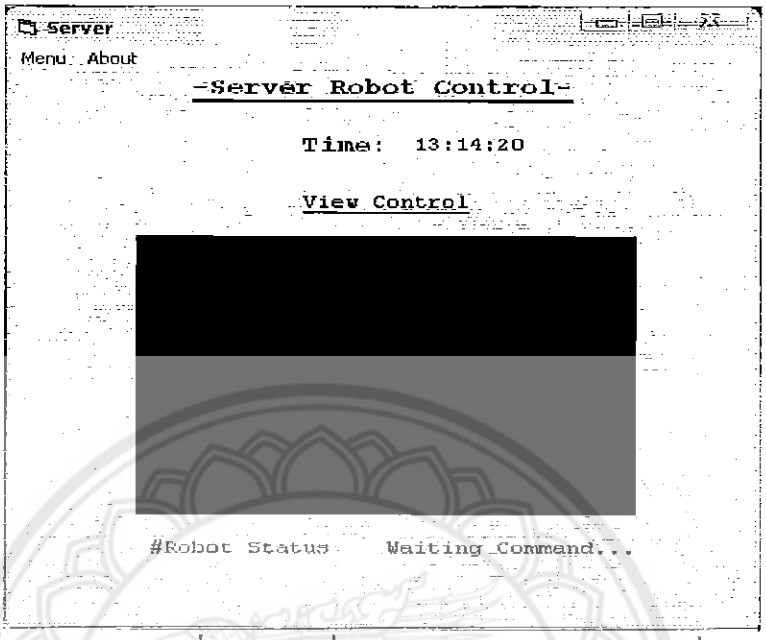
รูปที่ 4.1 โปรแกรมเครื่องแม่ข่ายเมื่อเริ่มใช้งาน

เมื่อทำการเปิดโปรแกรมที่ฝั่งลูกข่ายขึ้นมา



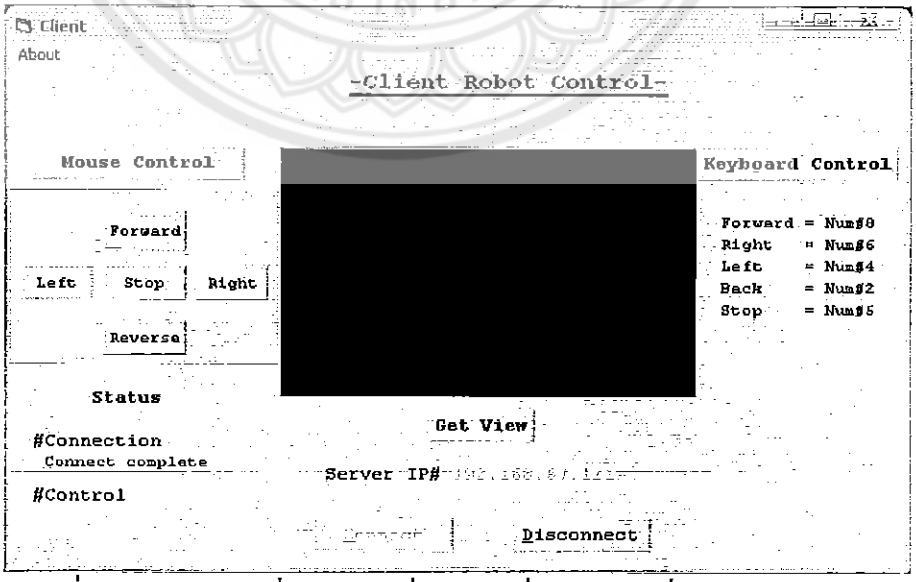
รูปที่ 4.2 โปรแกรมเครื่องลูกข่ายเมื่อเริ่มใช้งาน

เมื่อทำการเปิดทำการเปิดพอร์ตรอรับการร้องขอการติดต่อจากเครื่องลูกข่าย



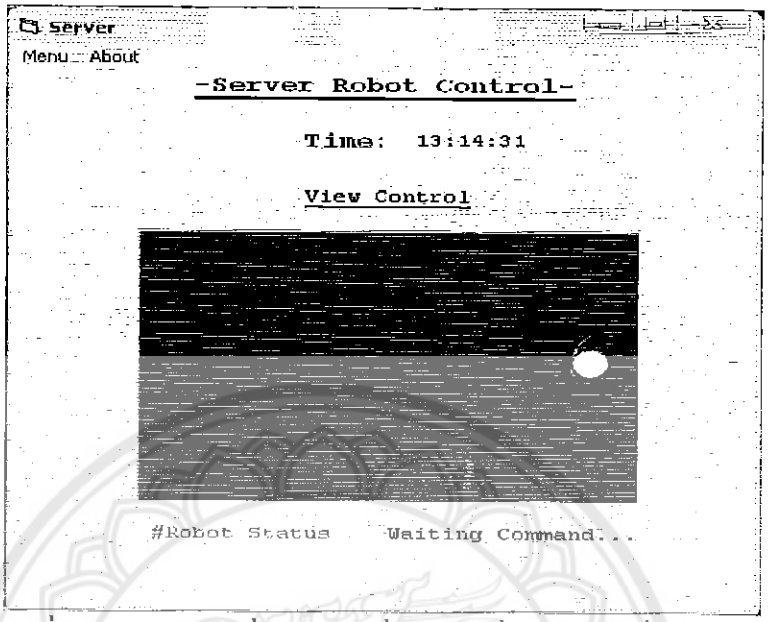
รูปที่ 4.3 โปรแกรมเครื่องแม่ข่ายเมื่อรอการร้องขอการติดต่อจากเครื่องลูกข่าย

เมื่อเครื่องลูกข่ายต้องการติดต่อไปยังเครื่องแม่ข่ายจะใส่ที่อยู่ปลายทางของเครื่องแม่ข่าย และทำการเชื่อมต่อและสามารถเชื่อมต่อกับเครื่องแม่ข่ายได้สำเร็จจะแสดงสถานะ การเชื่อมต่อว่าสามารถเชื่อมต่อกับเครื่องแม่ข่ายได้



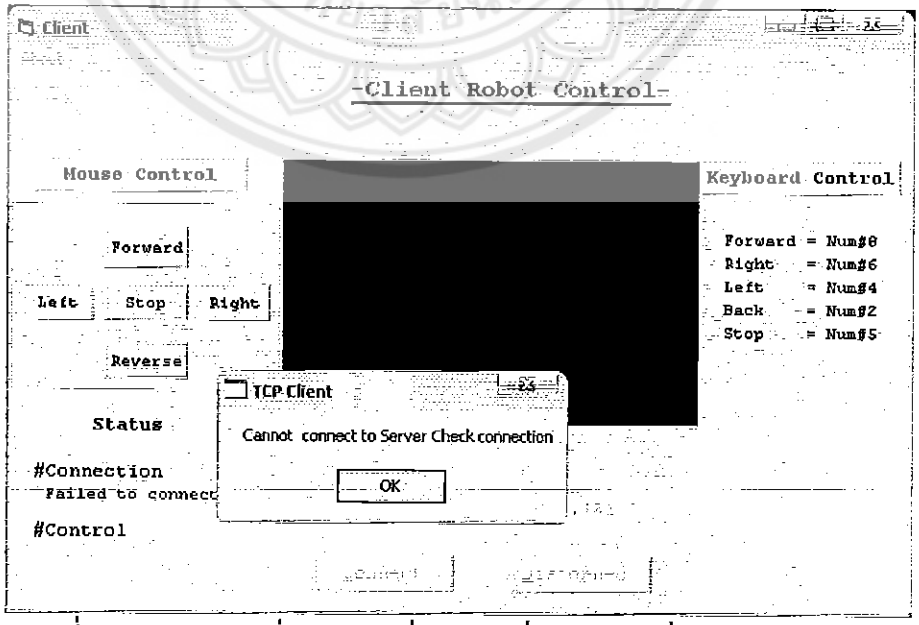
รูปที่ 4.4 โปรแกรมเครื่องลูกข่ายเมื่อทำการเชื่อมต่อกับเครื่องแม่ข่ายได้สำเร็จ

เมื่อเครื่องลูกข่ายติดต่อมายังเครื่องแม่ข่าย ที่หน้าต่างด้านบนจะแสดงที่อยู่ของเครื่องลูกข่ายที่ทำการติดต่อมาและเครื่องแม่ข่ายจะแสดงสถานะ การควบคุมหุ่นยนต์ผลที่ได้ดังรูป 4.5



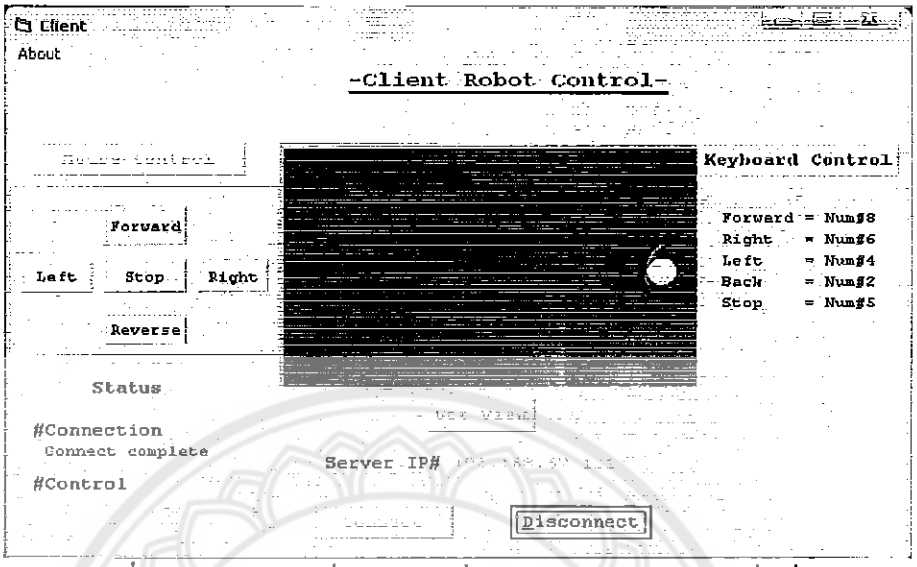
รูปที่ 4.5 โปรแกรมเครื่องแม่ข่ายเมื่อทำการเชื่อมต่อกับเครื่องลูกข่าย

เมื่อเครื่องลูกข่ายไม่สามารถทำการเชื่อมต่อกับเครื่องแม่ข่ายได้สำเร็จจะแสดงหน้าต่างความผิดพลาดออกมาและแสดงข้อความบอกว่าไม่สามารถเชื่อมต่อกับเครื่องแม่ข่ายได้ให้ทำการตรวจสอบการเชื่อมต่อกับเครื่องแม่ข่ายอีกครั้ง



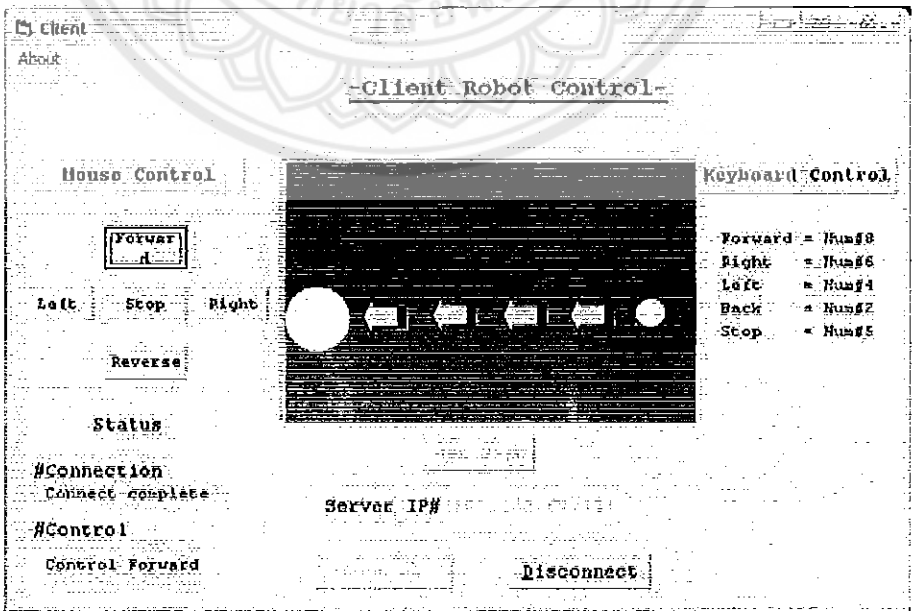
รูปที่ 4.6 โปรแกรมเครื่องลูกข่ายเมื่อทำการเชื่อมต่อกับเครื่องแม่ข่ายไม่สำเร็จ

เมื่อโปรแกรมที่เครื่องลูกข่ายเชื่อมต่อกับเครื่องแม่ข่ายได้สำเร็จกดปุ่มรับภาพมาแสดงที่เครื่องลูกข่ายแสดงได้ดังรูป

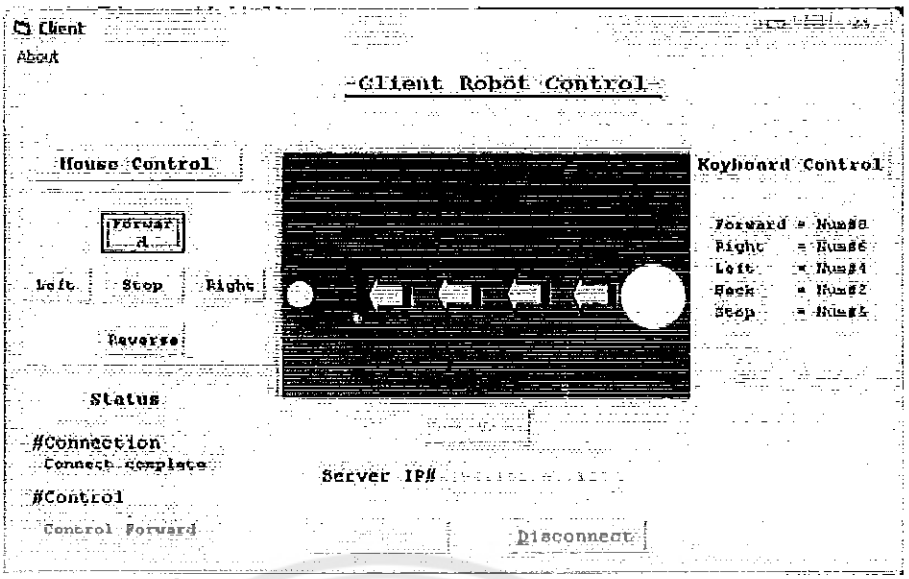


รูปที่ 4.7 โปรแกรมเครื่องลูกข่ายเมื่อทำการรับภาพมาแสดงที่เครื่อง

หลังจากเชื่อมต่อกับเครื่องแม่ข่ายและรับภาพมาแสดงที่เครื่องแม่ข่ายได้แล้วต่อไปจะเป็นการส่งคำสั่งไปยังเครื่องแม่ข่ายเพื่อทำการควบคุมการทำงานของหุ่นยนต์ โดยคำสั่งแรกที่ทำการทดลองส่งไปยังเครื่องแม่ข่าย เลือกใช้โหมด การควบคุมด้วยเมาส์ คือ สั่งให้หุ่นยนต์เดินไปข้างหน้า แสดงผลการทำงานได้ดังรูป4.8

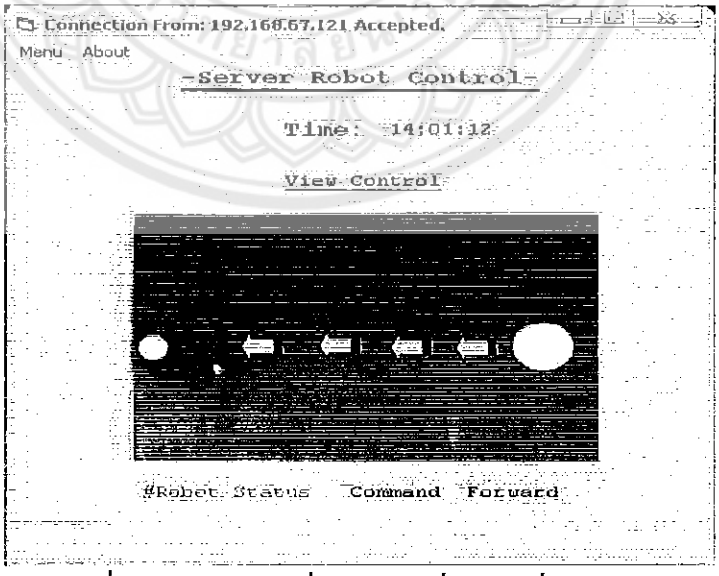


รูปที่ 4.8 (ก) โปรแกรมเครื่องลูกข่ายก่อนส่งคำสั่งเดินหน้า



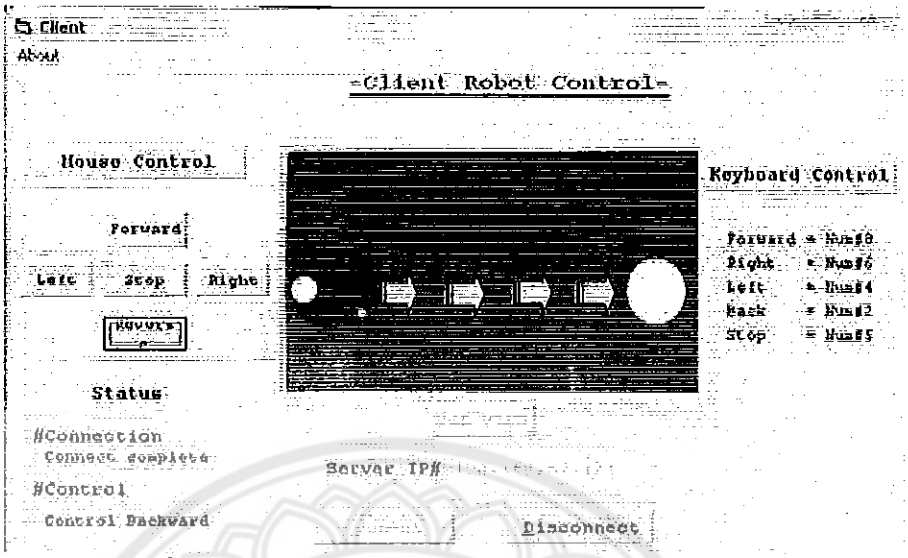
รูปที่ 4.8 (ข) โปรแกรมเครื่องลูกข่ายเมื่อส่งคำสั่งเดินหน้า

โดยเมื่อส่งคำสั่งไปยังเครื่องแม่ข่ายผลการทำงานภาพแสดงผลการทำงานของหุ่นยนต์ที่ได้รับมาแสดงที่เครื่องลูกข่ายจะแสดงผลการทำงานช้ากว่าการทำงานจริงของหุ่นยนต์ โดยหน่วงเวลาไปประมาณ 1 วินาที ผลการแสดงการทำงานเมื่อได้รับหลังจากส่งคำสั่งไปยังเครื่องแม่ข่ายแสดงได้ดังรูป 4.8 และที่เครื่องแม่ข่ายเมื่อรับคำสั่งจากเครื่องลูกข่ายจะส่งคำสั่งผ่านพอร์ตอนุกรม RS – 232 ไปยังอุปกรณ์ส่งสัญญาณวิทยุ หลังจากนั้นตัวส่งสัญญาณวิทยุจะส่งคำสั่งไปยังภาครับสัญญาณที่ตัวหุ่นยนต์โปรแกรมที่เครื่องแม่ข่ายจะแสดงสถานะของการทำงานออกมาดังรูปที่ 4.9

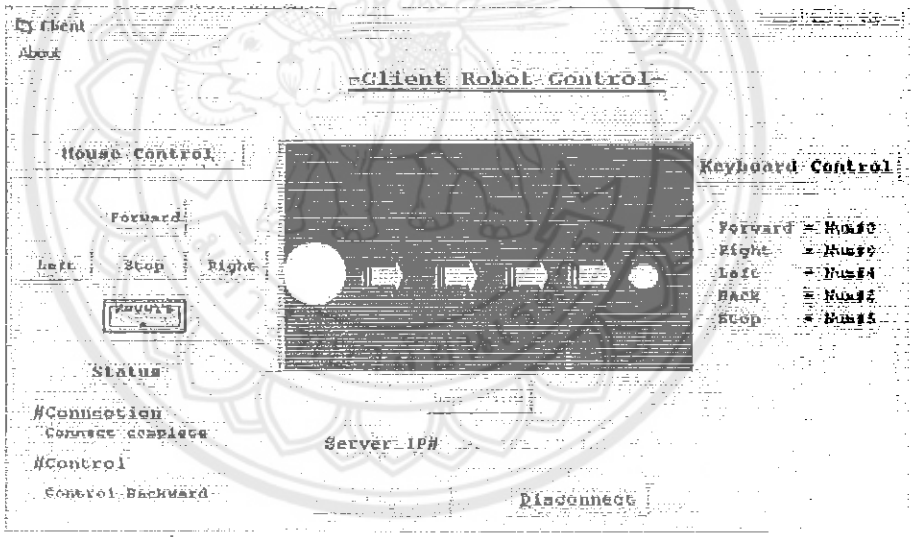


รูปที่4.9 โปรแกรมเครื่องแม่ข่ายเมื่อรับคำสั่งเดินหน้า

เมื่อเครื่องลูกข่ายส่งคำสั่ง ถอยหลัง ไปยังเครื่องแม่ข่ายผลการทำงานแสดงได้ดังรูปที่4.10

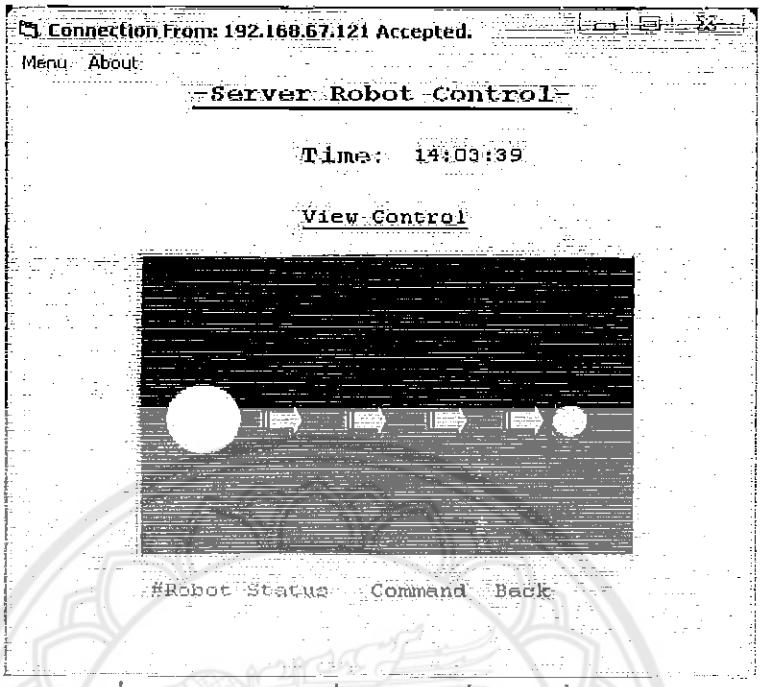


รูปที่4.10 (ก) โปรแกรมเครื่องลูกข่ายก่อนส่งคำสั่งถอยหลัง



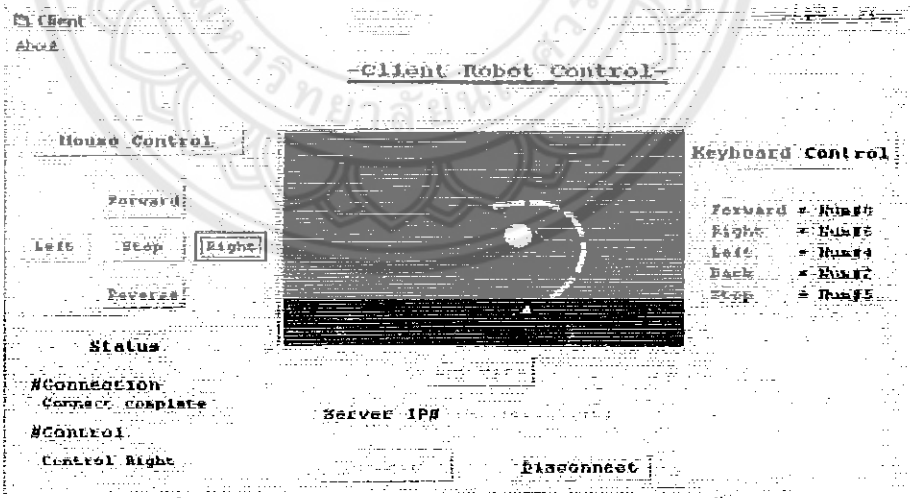
รูปที่4.10 (ข) โปรแกรมเครื่องลูกข่ายหลังส่งคำสั่งถอยหลัง

สถานการณ์ทำงานที่เครื่องแม่ข่ายเมื่อรับคำสั่งให้หุ่นยนต์ถอยหลังแสดงดังรูปที่ 4.11

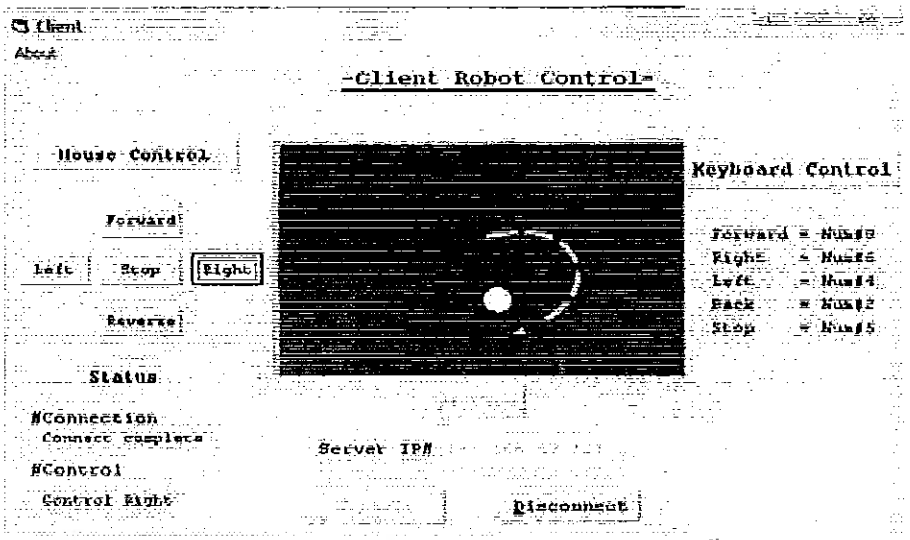


รูปที่4.11 โปรแกรมเครื่องแม่ข่ายเมื่อส่งคำสั่งถอยหลัง

เมื่อเครื่องลูกข่ายส่งคำสั่ง เลี้ยวขวา ไปยังเครื่องแม่ข่ายผลการทำงานแสดงได้ดังรูปที่ 4.12



รูปที่4.12 (ก) โปรแกรมเครื่องลูกข่ายก่อนส่งคำสั่งเลี้ยวขวา

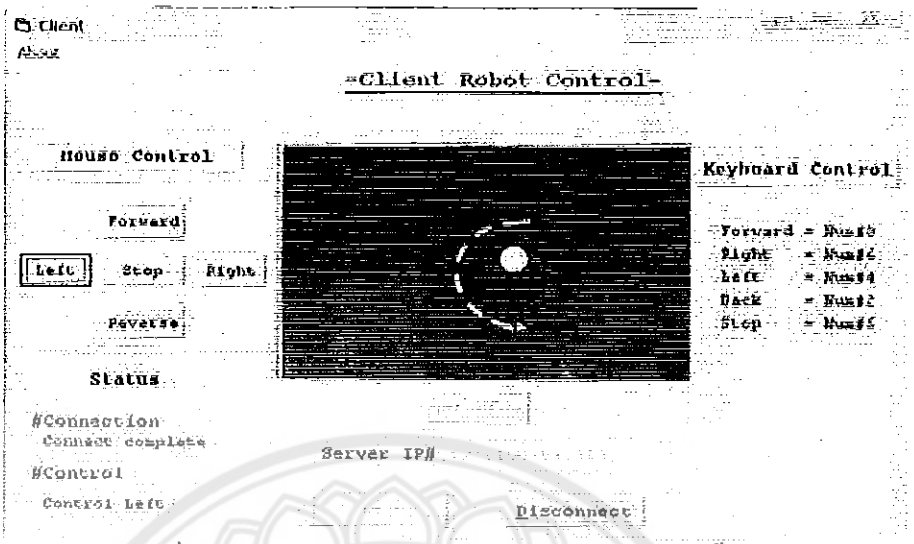


รูปที่4.12 (ข) โปรแกรมเครื่องลูกข่ายหลังส่งคำสั่งเลี้ยวขวา

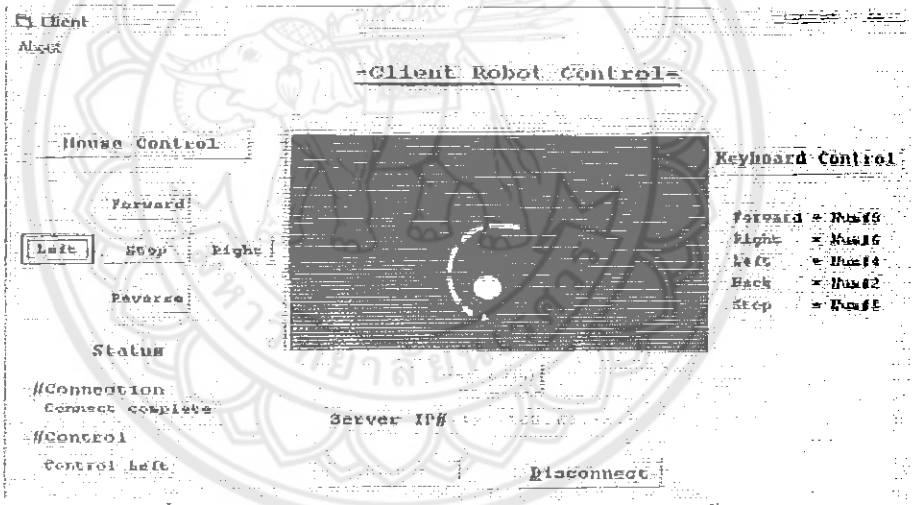
สถานะของการทำงานที่เครื่องแม่ข่ายเมื่อรับคำสั่งให้หุ่นยนต์เลี้ยวขวาแสดงดังรูปที่ 4.13

รูปที่4.13 โปรแกรมเครื่องแม่ข่ายเมื่อรับคำสั่งเลี้ยวขวา

เมื่อเครื่องลูกข่ายส่งคำสั่ง เลี้ยวซ้าย ไปยังเครื่องแม่ข่ายผลการทำงานแสดงได้ดังรูปที่ 4.14

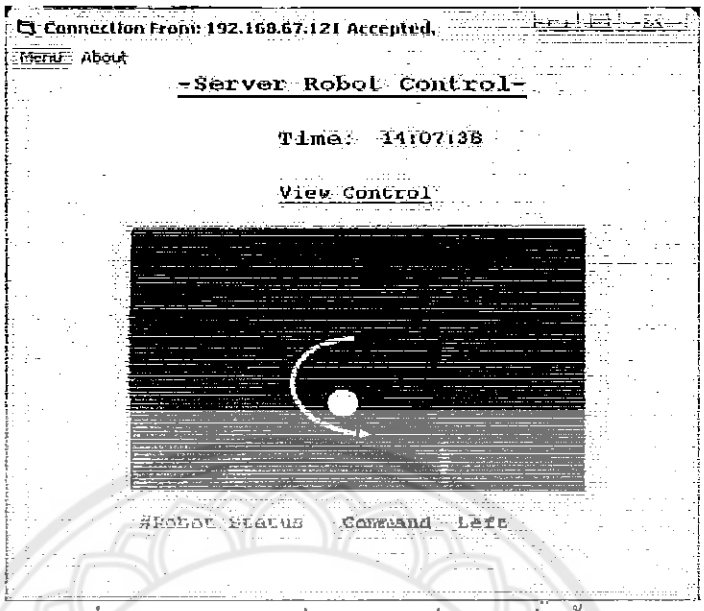


รูปที่4.14 (ก) โปรแกรมเครื่องลูกข่ายก่อนส่งคำสั่งเลี้ยวซ้าย



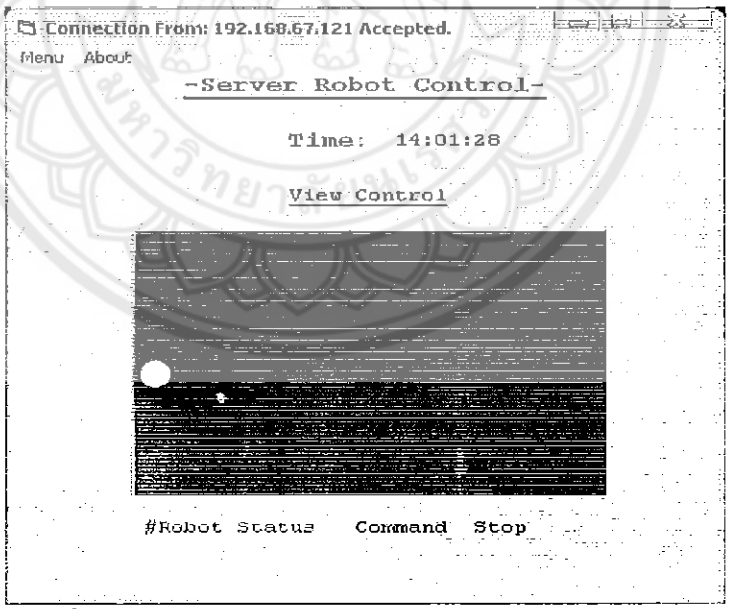
รูปที่4.14 (ข) โปรแกรมเครื่องลูกข่ายหลังส่งคำสั่งเลี้ยวซ้าย

สถานะของการทำงานที่เครื่องแม่ข่ายเมื่อรับคำสั่งให้หุ่นยนต์เลี้ยวซ้ายแสดงดังรูปที่ 4.15



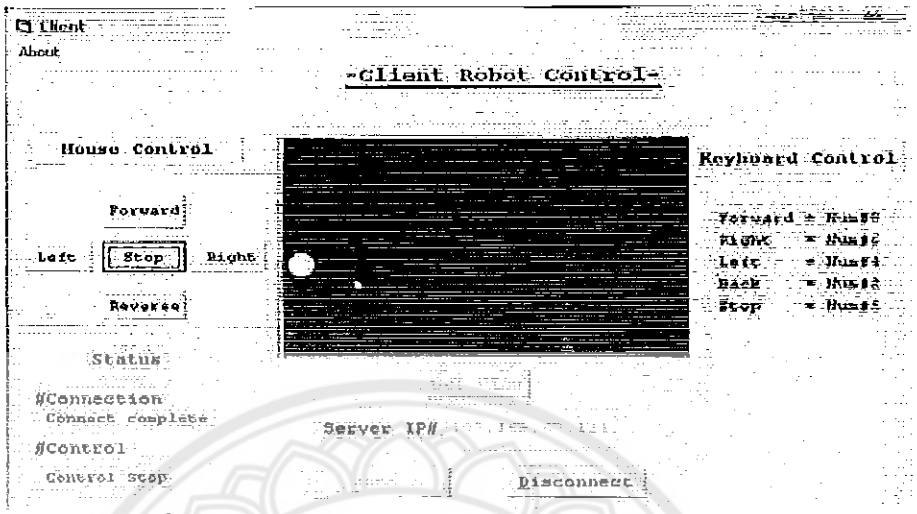
รูปที่4.15 โปรแกรมเครื่องแม่ข่ายเมื่อรับคำสั่งเลี้ยวขวา

เมื่อเครื่องลูกข่ายส่งคำสั่ง หยุด ไปยังเครื่องแม่ข่ายผลการทำงานแสดงได้ดังรูปที่4.16



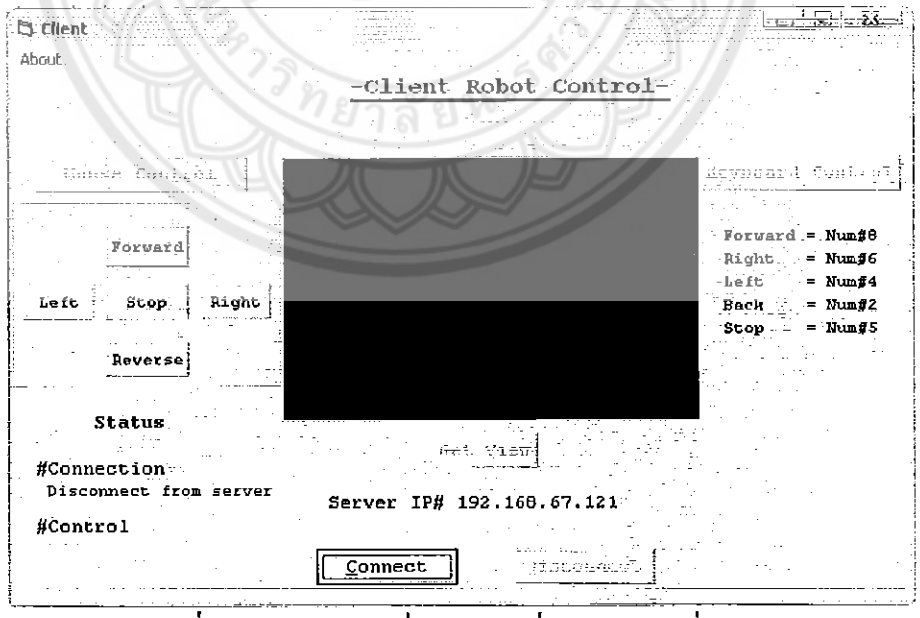
รูปที่4.16 โปรแกรมเครื่องแม่ข่ายเมื่อรับคำสั่งหยุดการทำงาน

สถานะของการทำงานที่เครื่องลูกข่ายเมื่อรับคำสั่งให้หุ่นยนต์ หยุดการทำงาน แสดงดังรูปที่ 4.17



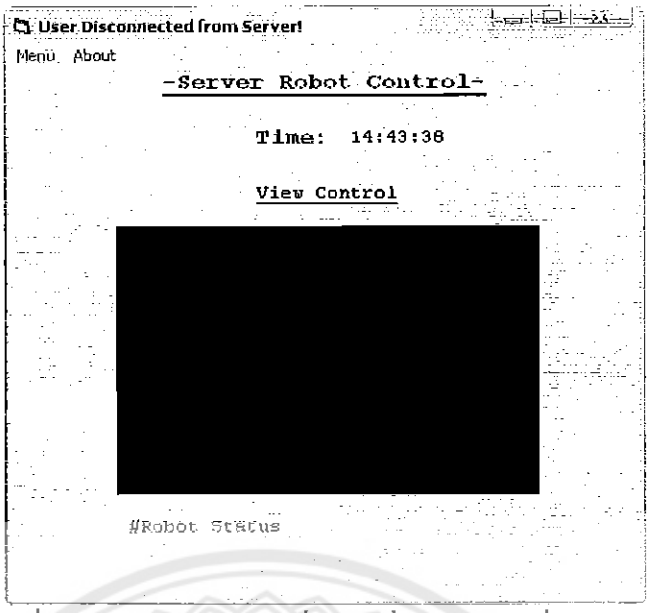
รูปที่4.17 โปรแกรมเครื่องลูกข่ายเมื่อส่งคำสั่งหยุดการทำงาน

เมื่อเครื่องลูกข่ายถ้าต้องการยกเลิกการเชื่อมต่อระหว่างเครื่องแม่ข่ายและให้กดปุ่ม Disconnect ที่โปรแกรมของเครื่องลูกข่ายและผลการทำงานของโปรแกรมเมื่อทำการยกเลิกการติดต่อแสดงได้ดังรูปที่ 4.18



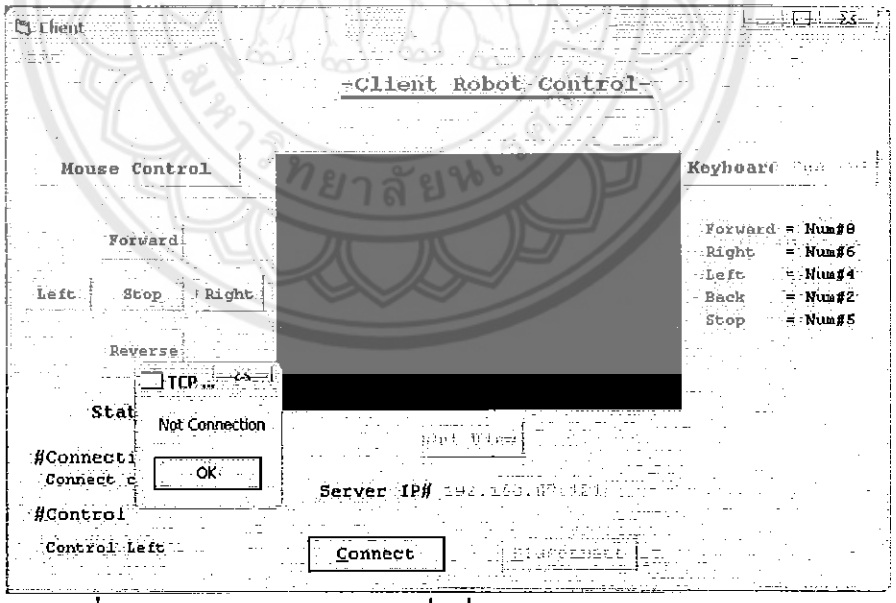
รูปที่4.18 โปรแกรมเครื่องลูกข่ายเมื่อยกเลิกการเชื่อมต่อ

สถานะของการเชื่อมต่อที่โปรแกรมเครื่องแม่ข่ายเมื่อเครื่องลูกยกเลิกการติดต่อดัง รูปที่4.19



รูปที่ 4.19 สถานะของการเชื่อมต่อที่โปรแกรมเครื่องแม่ข่าย

ขณะที่ระหว่างเครื่องแม่ข่ายและเครื่องลูกข่ายทำการติดต่ออยู่นั้นเครือข่ายมีปัญหาทำให้เครื่องแม่ข่ายและเครื่องลูกข่ายไม่สามารถติดต่อกันได้คำสั่งสุดท้ายที่เครื่องลูกข่ายส่งไปยังเครื่องแม่ข่ายจะถูกยกเลิกการทำงานทันทีแสดงได้ดังรูปที่ 4.20



รูปที่ 4.20 สถานะของโปรแกรมที่เครื่องลูกข่ายขณะหลุดจากการติดต่อ

จากการทดลองผลการทำงานของโปรแกรมสามารถทำงานได้ตามคำสั่งที่เครื่องลูกข่ายได้ส่งไปยังเครื่องแม่ข่ายได้ทุกคำสั่งและสามารถแสดงภาพผลการทำงานขณะทำการควบคุมหุ่นยนต์

บทที่ 5
สรุปผล

โครงงานนี้ทำการศึกษาและพัฒนาการเขียนโปรแกรม สั่งงานหุ่นยนต์ผ่านทางเครือข่ายด้วยโปรโตคอล TCP/ IP มาใช้งาน ภาษาคอมพิวเตอร์ที่นำมาใช้พัฒนาการเขียนโปรแกรมในโครงงานนี้ได้เลือกใช้ภาษาวิชวลเบสิก 6.0 มาพัฒนาโปรแกรม เนื่องจากเป็นภาษาที่สามารถทำความเข้าใจได้ง่ายและในการพัฒนาออกแบบโปรแกรมในส่วนหน้าต่างโปรแกรมสามารถทำได้ง่าย

5.1 สรุปผลการทดลอง

จากการจัดทำโครงงานการควบคุมหุ่นยนต์ผ่านเครือข่าย ผลที่ได้รับจากโครงงานนี้ก็ คือ สามารถเขียนโปรแกรมใช้ในการสั่งงานหุ่นยนต์จากระยะไกลได้โปรแกรมที่พัฒนาขึ้นจะสามารถนำมาใช้งานในการสั่งงานหุ่นยนต์ได้โดยมีการทำงาน 2 ส่วน คือโปรแกรมส่วนที่เป็นเครื่องแม่ข่ายและโปรแกรมส่วนที่เป็นเครื่องลูกข่าย ทั้งสองส่วนสามารถเชื่อมต่อกันได้ เมื่อทำการส่งคำสั่งไปควบคุมการทำงานของหุ่นยนต์ที่เครื่องแม่ข่าย โดยแบ่งการทำงานเป็น 2 กรณี คือ เครื่องลูกข่ายส่งคำสั่งไปยังเครื่องแม่ข่ายโดยไม่รับภาพผลการทำงานมาแสดงที่เครื่องลูกข่าย และ รับภาพมาแสดงที่เครื่องลูกข่าย จากผลการทดลองพบว่าระบบจะมีเวลาหน่วง (delay time) เกิดขึ้น คือ ในกรณีที่ไม่รับภาพมาแสดงผลที่เครื่องลูกข่ายจะเกิดเวลาหน่วงประมาณ 0.4 วินาที ส่วนกรณีที่ต้องการรับภาพมาแสดงผลที่เครื่องลูกข่ายจะเกิดเวลาหน่วงประมาณ 1 วินาที ขณะที่กำลังควบคุมการทำงานจากเครื่องลูกข่ายอยู่นั้นการเชื่อมต่อกันระหว่างเครื่องลูกข่ายและเครื่องแม่ข่ายถูกตัดขาดจากกัน คำสั่งสุดท้ายที่ส่งไปจากเครื่องลูกข่ายไปควบคุมการทำงานของหุ่นยนต์จะถูกยกเลิกหรือขณะที่หุ่นยนต์กำลังทำงานอยู่การทำงานนั้นก็จะหยุดลง

จากการทดลองที่ได้สามารถสรุปได้ว่าโปรแกรมที่พัฒนาขึ้นสามารถนำไปใช้สามารถนำไปใช้ควบคุมการทำงานของหุ่นยนต์ระยะไกลได้เป็นอย่างดี

5.2 ปัญหาและแนวทางแก้ไข

ในการแสดงผลภาพที่เครื่องลูกข่ายภาพที่นำมาแสดงผลการทำงานช้ากว่าการทำงานจริงของหุ่นยนต์อาจเป็นเพราะขนาดของภาพที่ส่งมาใช้วิธีการส่งที่ไม่ดีพอหรือการใช้งานเครือข่ายมีสภาพที่แออัด ทำให้ข้อมูลภาพที่ส่งมาใช้งานแสดงผลช้า การแก้ไขโดยการบีบอัดรูปภาพให้เล็กลงหรือใช้วิธีการบีบอัดรูปภาพแบบอื่น ในกรณีที่เกิดจากเครือข่าย แก้ได้โดยการเชื่อมต่อเครื่องแม่ข่ายและเครื่องลูกข่ายโดยใช้รูปแบบการเชื่อมต่อโดยผ่านเครือข่ายความเร็วสูงมาใช้งาน ผู้ออกแบบต้องคำนึงถึงสิ่งเหล่านี้ด้วยหากต้องให้การแสดงผลภาพที่ละเอียดระบบจะช้า

5.3 แนวทางพัฒนาต่อ

ในการส่งคำสั่งไปควบคุมหุ่นยนต์ระยะไกลสิ่งที่คำนึงถึง คือ ผลตอบสนองของเวลา (response time) หลังจากได้ทำการสั่งงานหุ่นยนต์ที่ระบบต้องการเป็นแบบ real time เพื่อให้ได้การทำงานที่ถูกต้องตามที่ต้องการทันทีโดยอาจทำเอาเทคโนโลยีอื่นเข้ามาช่วยเพื่อให้ได้การทำงานที่ตรงกับความต้องการ เช่น

ระบบโทรศัพท์ไร้สาย นำมาประยุกต์ใช้สั่งงานหุ่นยนต์โดยใช้เครือข่ายของระบบโทรศัพท์ไร้สาย อาจจะพัฒนาโปรแกรมลงบนโทรศัพท์ไร้สายในการสั่งงานหุ่นยนต์ ซึ่งจะทำให้สะดวกต่อการสั่งงานหุ่นยนต์ และความเร็วในการสั่งงานอาจจะเร็วขึ้น

การนำการเชื่อมต่อของเครือข่ายในรูปแบบอื่นมาใช้ บางครั้งการเชื่อมต่อเครือข่ายทั่วไปที่ใช้อยู่ จะมีการใช้งานเครือข่ายจากผู้ใช้หลายคนซึ่งจะเป็นผลให้เครือข่ายหนาแน่นทำให้เกิดความล่าช้าในการส่งข้อมูล ซึ่งอาจจะใช้รูปแบบการเชื่อมต่อแบบอื่นมาใช้งาน เช่น การนำการเชื่อมต่อเครือข่ายความเร็วสูงมาใช้งาน

เอกสารอ้างอิง

[1] นายศรายุทธ หอยสังข์ และ นางสาวหัสยา โตมณีพิทักษ์. **"ไฟลล์วอลล์และระบบความปลอดภัยบนเครือข่ายกรณีศึกษาของมหาวิทยาลัยนเรศวร Firewall and Network Security,Case Study of Naresuan University"**. ปริญญานิพนธ์วิศวกรรมศาสตรบัณฑิต สาขาวิศวกรรมศาสตร์ ภาควิชาวิศวกรรมไฟฟ้าและคอมพิวเตอร์ คณะวิศวกรรมศาสตร์ มหาวิทยาลัยนเรศวร.2544

[2] DataNews. "โปรโตรคอล IP เหมาะสมกับเครือข่ายใดบ้าง" [Online].Available : http://www.sut.as.th/ccs.

[3] นพปฎล กุลจรรยาวิวัฒน์. "คำอธิบายเกี่ยวกับเรื่อง Port" [Online]. Available : http://www.sans.org/infosecFAQ/securitybasics/ports.htm .2544

[4] รศ.ดร. สมชาย ชวเสถียร และ นรินทร์ อัครพิเชษฎ์. "Fundamental of Visual Basic Client Server Programming".1998. SUM System Company. Bangkok .Thailand

[5] ผ.ศ.ธีรวัฒน์ ประกอบผล. "การประยุกต์ใช้งานไมโครคอนโทรลเลอร์" .พิมพ์ครั้งที่ 6. กรุงเทพฯ:ส.ส.ท.2546.

[6] ทวิ ว่องชัญกิจ. "การรับส่งข้อมูลแบบอนุกรม." [Online]. Available : http://www.jobpub.com/articles/showarticle.asp?id=151 .2000

[7] ดร.สนั่น ศรีสุข. " Internet Programing Witch Winsock" .หน้า 7-8

[8] นายจักรพงษ์ เสือมี และนายสิทธิชาติ เทิดสิทธิกุล. "การบีบอัดข้อมูลภาพโดยวิธีแปลงเวฟเล็ต". ปริญญานิพนธ์วิศวกรรมศาสตรบัณฑิต สาขาวิศวกรรมศาสตร์ ภาควิชาวิศวกรรมไฟฟ้าและคอมพิวเตอร์ คณะวิศวกรรมศาสตร์ มหาวิทยาลัยนเรศวร.2546

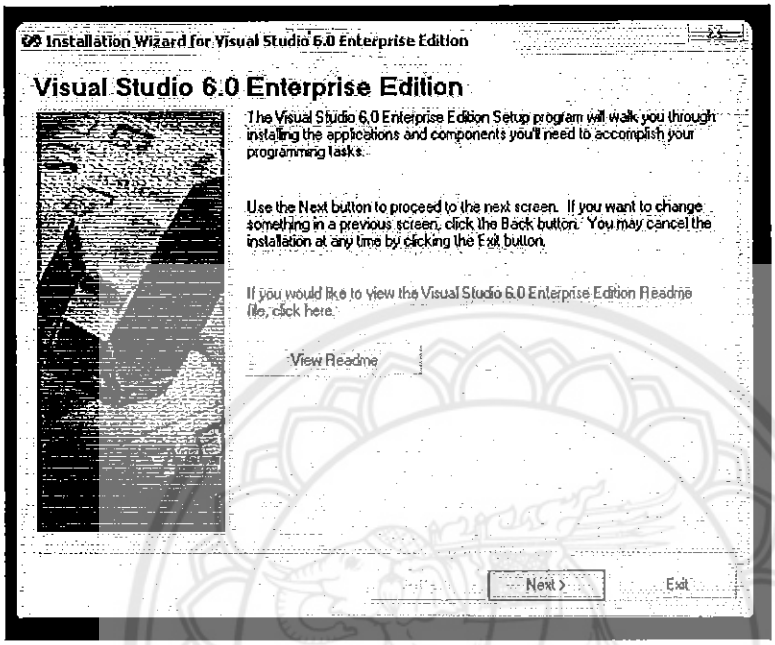
[9] G. Scott Owen. "RGB Color Model." [Online]. Available : http://www.siggraph.org/education/materials/HyperGraph/color/colorgb.htm

[10] เทคโนโลยีคอมพิวเตอร์และสารสนเทศ. " การสื่อสารข้อมูลและเครือข่าย." [Online]. Available : http://www.sa.ac.th/elearning/index68.htm .2000

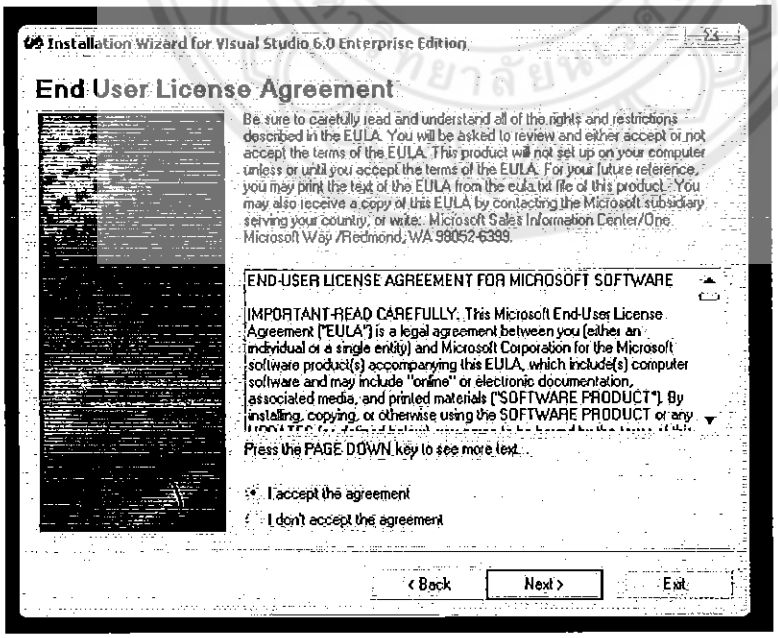
ภาคผนวก ก

ขั้นตอนการลงโปรแกรม Visual Basic 6

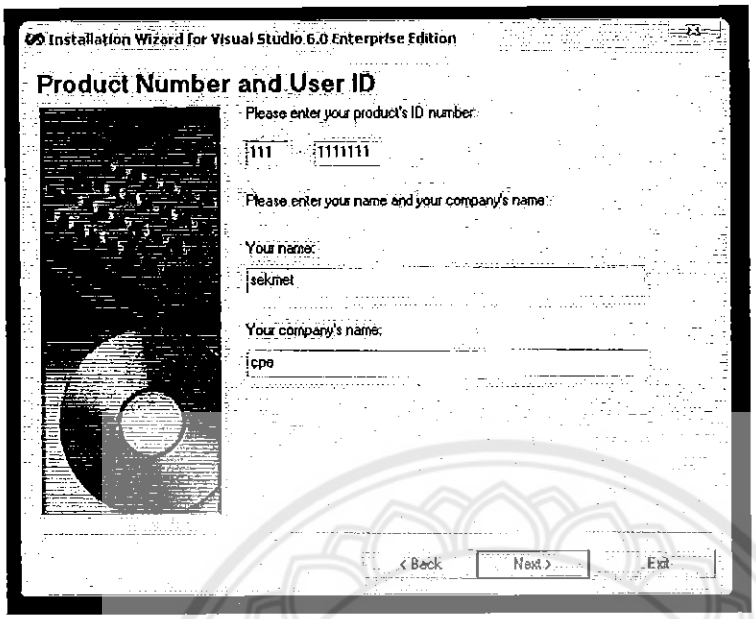
1. นำแผ่นโปรแกรมใส่ใน VCD Rom เครื่องจะทำการ run อัตโนมัติดังรูป



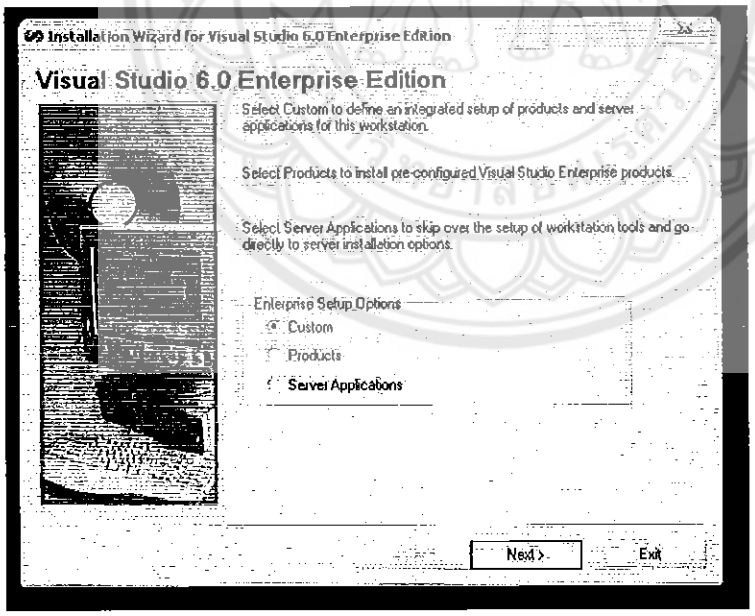
2. คลิกที่ Next >



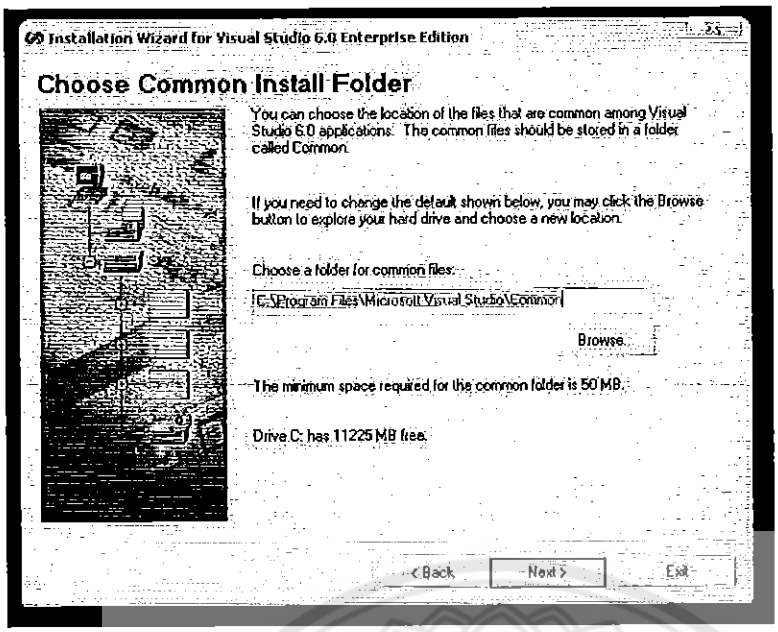
3. เลือก I accept agreement แล้วคลิกที่ Next >



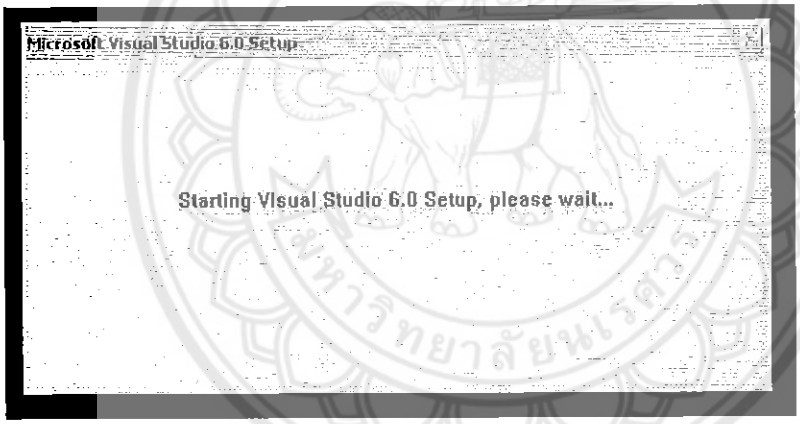
4. ใส่ ID number ใส่ Your name และ Your company's name จากนั้นคลิก Next >



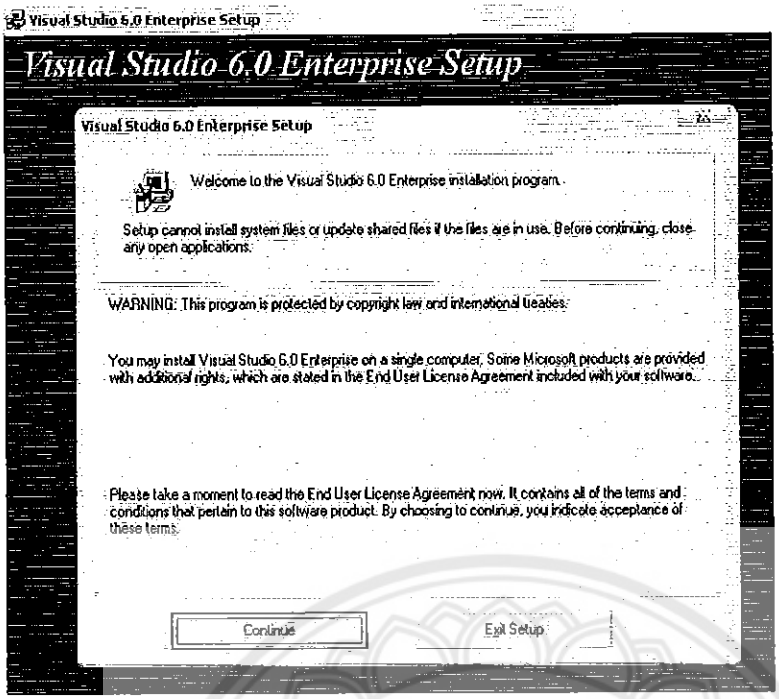
5. เลือก Custom แล้วคลิก Next >



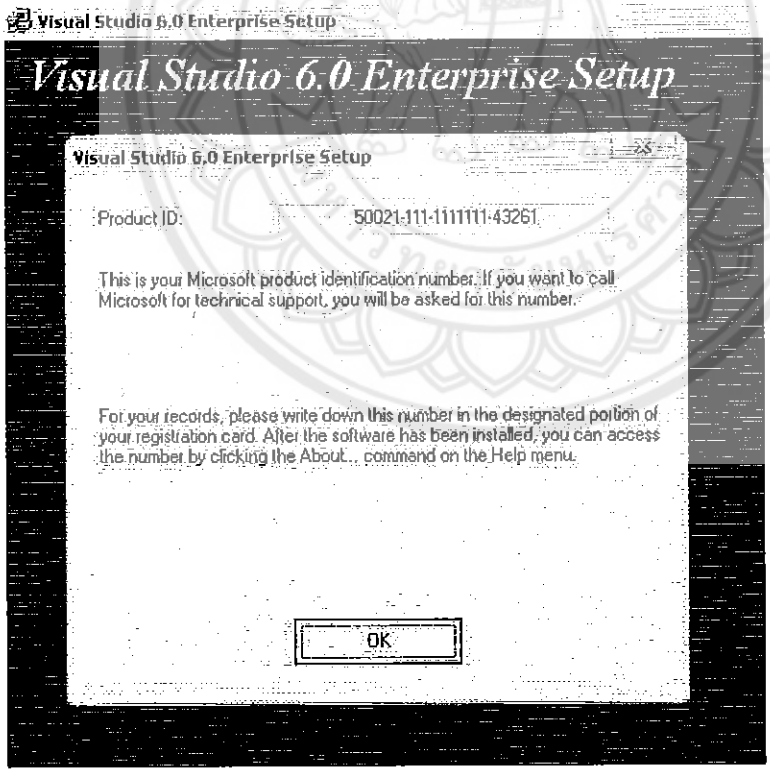
6. ทำการเลือก folder ที่ต้องการลงโปรแกรม แล้วคลิก Next >



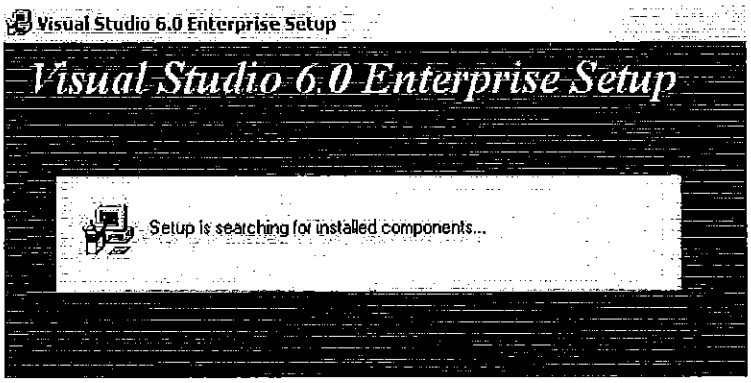
7. รอการ set up ของโปรแกรม



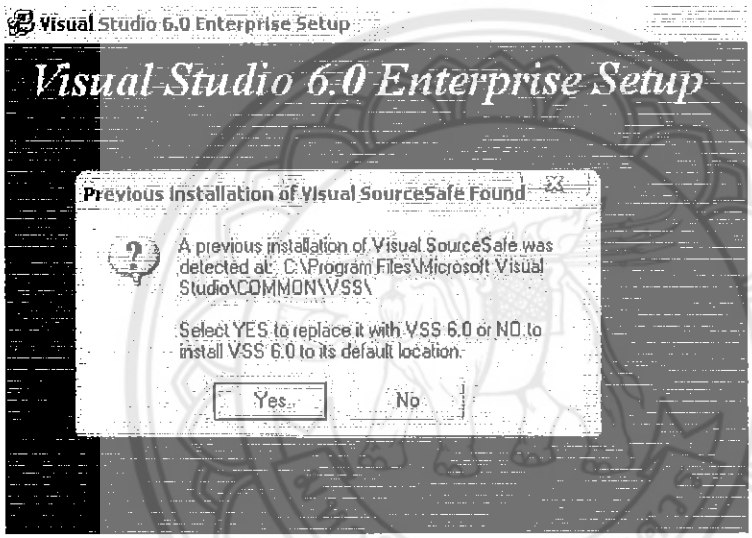
8. ทำการ set up เลือก Continue



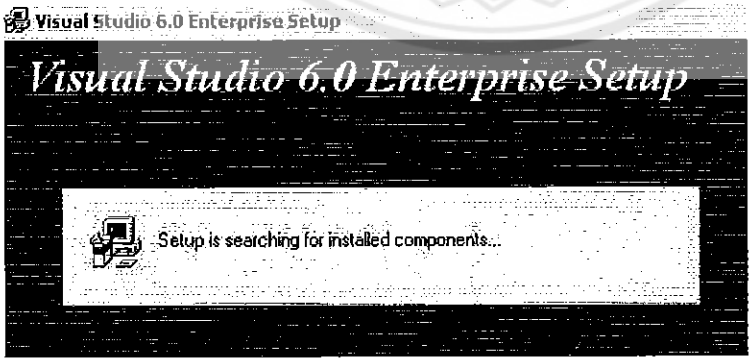
9. คลิก OK



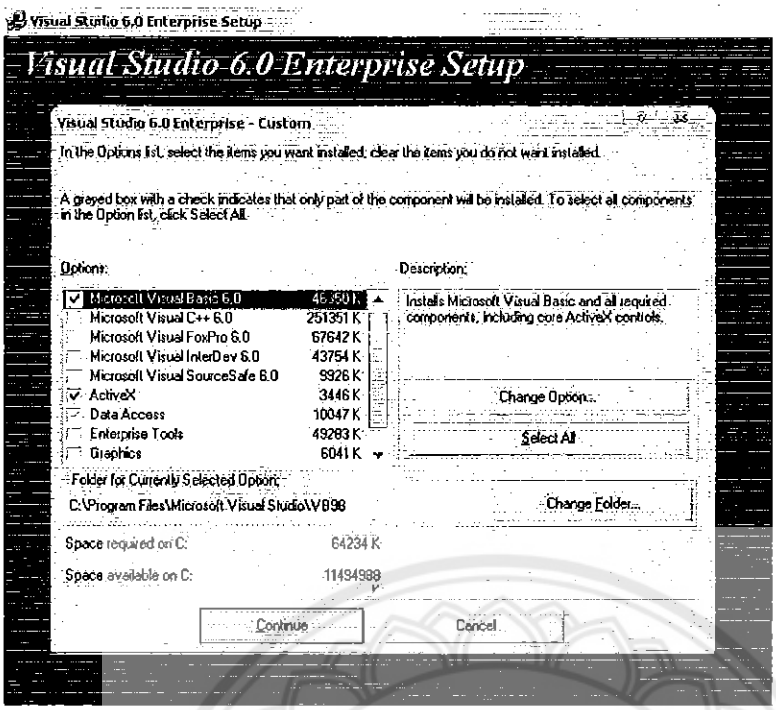
10. เครื่องทำการ set up เพื่อทำการลง components..



11. คลิก Yes

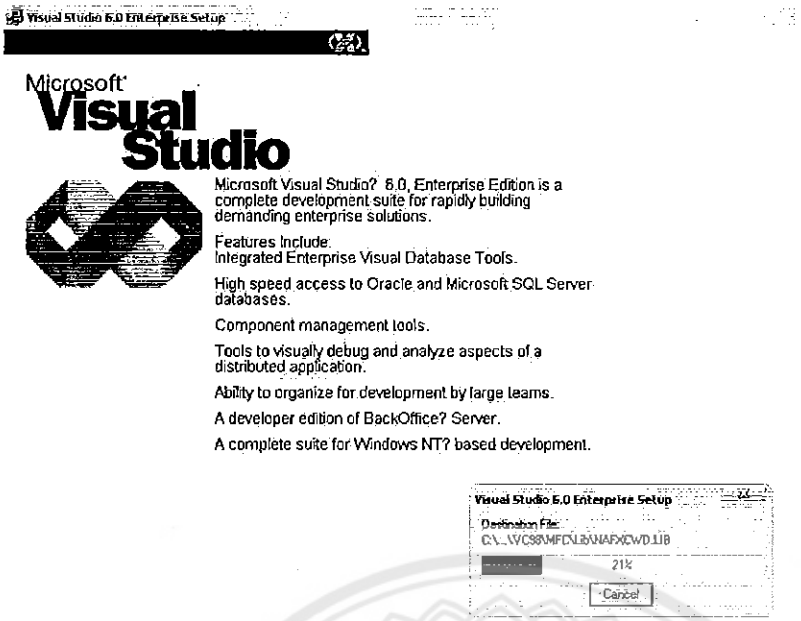


12. เครื่องทำการ set up เพื่อทำการลง components..

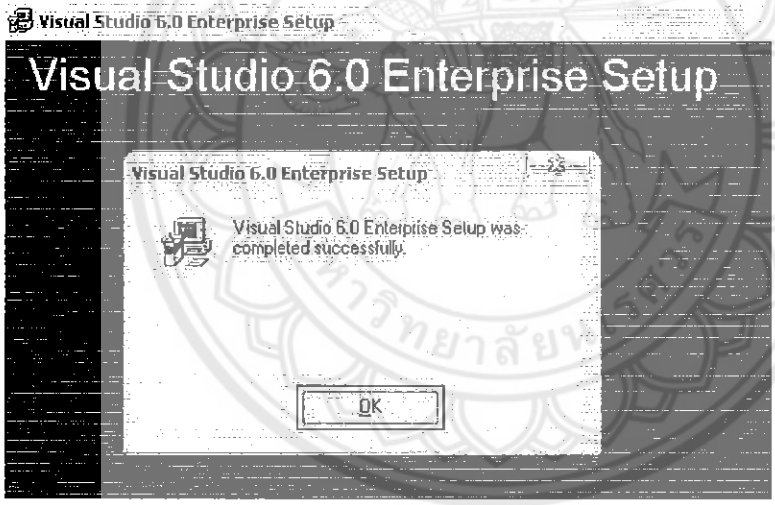


13. เราต้องการลงโปรแกรม Visual Basic 6 อย่าเดียวเพราะฉะนั้นเราก็เลือก Visual Basic 6 , ActiveX , Tool นอกนั้นเลือกออกให้หมด จากนั้น คลิก Continue

14. เครื่องกำลัง check disk

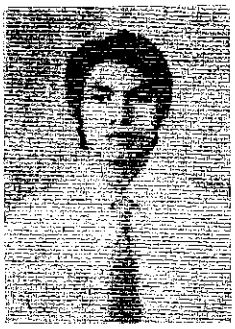


15. เครื่องกำลังลงโปรแกรม



16. ทำการลงโปรแกรมเสร็จสมบูรณ์

ประวัติผู้เขียนโครงงาน



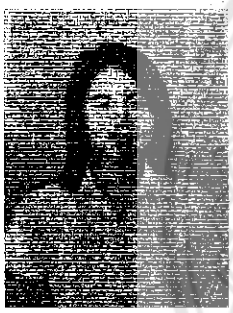
ชื่อ นายสุเมธ สมมะลวน

ภูมิลำเนา 8/4 หมู่6 ต.พยุหะ อ.พยุหะคีรี จ.นครสวรรค์ 60130

ประวัติการศึกษา

- จบระดับมัธยมศึกษาตอนต้นจากโรงเรียน สงขลาวิทยาคม
- จบระดับมัธยมศึกษาตอนปลายจากโรงเรียนพยุหะพิทยา
- ปัจจุบันกำลังศึกษาในระดับปริญญาตรีชั้นปีที่ 4 สาขาวิชาวิศวกรรมศาสตร์ คณะวิศวกรรมศาสตร์ มหาวิทยาลัยนเรศวร

E-mail : **meth_ix@hotmail.com**



ชื่อ นางสาวเศกศิริ เรืองวาณิชยกุล

ภูมิลำเนา 61/1 ถ.ประชาราษฎร์ ต.เมืองสวรรคโลก อ.สวรรคโลก จ.สุโขทัย 64110

ประวัติการศึกษา

- จบมัธยมศึกษาจากโรงเรียนสวรรค์อนันต์วิทยา
- ปัจจุบันกำลังศึกษาในระดับปริญญาตรีชั้นปีที่ 4 สาขาวิชาวิศวกรรมศาสตร์ คณะวิศวกรรมศาสตร์ มหาวิทยาลัยนเรศวร

E-mail : **sa_eikr@hotmail.com**