โปรแกรมจำลองการทำงานสำหรับเน็ตเวิร์คโพรโตคอล TCP/IP เลเยอร์ 1 และ เลเยอร์ 2

TCP/IP NETWORK PROTOCOL (LAYER 1 & 2) SIMULATION

นายชัยภัทร จารุชาต รหัส 48370815

นายนทภพ เริงนภาพล รหัส 48370884

ห้องสมุดคณะวิศวกรรมศาสตร์
วันที่รับ 4 มิ.ย. 2553
เลขทะเบียน 1499697
เลขเรียกหนังสือ วร.
มหาวิทยาลัยนเรศวร 9399 ม
2551

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต

สาขาวิชาวิศวกรรมคอมพิวเตอร์ ภาควิชาวิศวกรรมไฟฟ้าและคอมพิวเตอร์

คณะวิศวกรรมศาสตร์ มหาวิทยาลัยนเรศวร

ปีการศึกษา 2551

ใบรับรองโครงงานวิศวกรรม

หัวข้อโครงงาน โปรแกรมจำลองการทำงานสำหรับเน็ตเวิร์คโพรโตคอล TCP/IP เลเยอร์ 1 และ เลเยอร์ 2

ผู้ดำเนินโครงงาน	นายชัยภัทร	จารุชาต	รหัส 48370815
	นายนทภพ	เริงนภาพล	รหัส 48370884
อาจารย์ที่ปรึกษา	ดร.สุรเดช จิตประไพกุลศาล		
สาขาวิชา	วิศวกรรมคอมพิวเตอร์		
ภาควิชา	วิศวกรรมไฟฟ้าและคอมพิวเตอร์		
ปีการศึกษา	2551		

.....

คณะวิศวกรรมศาสตร์ มหาวิทยาลัยนเรศวร อนุมัติให้โครงงานฉบับนี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรวิศวกรรมศาสตรบัณฑิต สาขาวิชาวิศวกรรมคอมพิวเตอร์

คณะกรรมการสอบโครงงานวิศวกรรม

..... ประธานกรรมการ
(ดร.สุรเดช จิตประไพกุลศาล)

..... กรรมการ
(อาจารย์แสงชัย มังกรทอง)

..... กรรมการ
(อาจารย์ภาณุพงศ์ สอนคม)

หัวข้อโครงงาน	โปรแกรมจำลองการทำงานสำหรับเน็ตเวิร์คโพรโตคอล TCP/IP เลเยอร์ 1 และ เลเยอร์ 2
ผู้ดำเนินโครงงาน	นายชัยภัทร จารุชาต รหัส 48370815
	นายนทภพ เริงนภาพล รหัส 48370884
อาจารย์ที่ปรึกษา	ดร.สุรเดช จิตประไพกุลศาล
สาขาวิชา	วิศวกรรมคอมพิวเตอร์
ภาควิชา	วิศวกรรมไฟฟ้าและคอมพิวเตอร์
ปีการศึกษา	2551

บทคัดย่อ

โครงงานนี้ได้พัฒนาต่อมาจากโครงงานไลบรารีสำหรับจำลองเน็ตเวิร์คโพรโตคอลแสตกแบบ TCP/IP Model (A Library for Simulating Network Protocol Stack) โดยใช้ภาษาจาวา ซึ่งแต่เดิมทำไว้ 2 เลเยอร์ คือ เลเยอร์ 3 และ เลเยอร์ 4 จึงได้ทำการอิมพลีเมนต์ เลเยอร์ 1 และ เลเยอร์ 2 เพิ่มเติมเพื่อให้การจำลองเน็ตเวิร์คโพรโตคอลแสตกมีความสมบูรณ์มากยิ่งขึ้น และในการจำลองสำหรับเลเยอร์ 1 และ เลเยอร์ 2 นั้นได้เลือกโพรโตคอล IEEE 802.3 Ethernet และโพรโตคอล IEEE 802.11 Wireless ซึ่งเป็นที่นิยมในปัจจุบันมาเป็นตัวอย่าง

Project Title	TCP/IP Network Protocol (Layer 1 & 2) Simulation		
Name	Mr.Chaipat	Jaruchart	ID. 48370815
	Mr.Nodthapob	Reangnapapol	ID. 48370884
Project Advisor	Dr.Suradet	Jitprapaikulsarn	
Major	Computer Engineering.		
Department	Electrical and Computer Engineering.		
Academic Year	2008		

ABSTRACT

This project extends from the project 'A Library for Simulating Network Protocol Stack' which created libraries for simulate protocols for network layers 3 and 4. This project focus on creating libraries for protocol in layers 1 and 2 to make the network protocol simulation practically complete. The selected protocols for this project are IEEE 802.3 Ethernet and IEEE 802.11 Wireless.

กิตติกรรมประกาศ

โครงงานวิศวกรรมคอมพิวเตอร์สำเร็จได้ด้วยดี ก็เนื่องจากความอนุเคราะห์จากท่านอาจารย์ที่ปรึกษาโครงงาน ดร.สุรเดช จิตประไพกุลศาล ที่กรุณาสละเวลาให้คำแนะนำในการทำงานตลอดถึงการตรวจสอบการทำงานพร้อมทั้งเสนอแนวทางการแก้ไขปัญหาตลอดระยะเวลาการทำโครงงาน ขอขอบคุณพี่ต้นและพี่โจ้ซึ่งได้ริเริ่มโครงงานนี้ไว้ พร้อมทั้งให้คำแนะนำที่เป็นประโยชน์ทำให้การทำโครงงานเป็นไปอย่างราบรื่น

ขออำนาจคุณพระศรีรัตนตรัย และสิ่งศักดิ์สิทธิ์ทั้งหลายบันดาลให้ อาจารย์และพี่ๆ สุขภาพแข็งแรงและเป็นแรงผลักดันให้คณะวิศวกรรมศาสตร์ มหาวิทยาลัยนเรศวร มีความก้าวหน้าต่อไป

นายชัยภัทร จารุชาต

นายนทภพ เริงนภาพล

สารบัญ

หน้า

บทคัดย่อภาษาไทย ..... ก

บทคัดย่ออังกฤษ ..... ข

กิตติกรรมประกาศ ..... ค

สารบัญ ..... ง

สารบัญตาราง ..... ช

สารบัญรูป ..... ซ

บทที่ 1 บทนำ

1.1 ที่มาและความสำคัญของโครงงาน 1

1.2 วัตถุประสงค์โครงงาน 1

1.3 ขอบเขตของโครงงาน 1

1.4 ขั้นตอนการดำเนินงาน 1

1.5 แผนการดำเนินงาน 2

1.6 ผลที่คาดว่าจะได้รับ 2

1.7 รายละเอียดงบประมาณของโครงงาน 3

บทที่ 2 หลักการและทฤษฎี

2.1 TCP/IP Protocol Suite 4

2.1.1 Application Layer 5

2.1.2 Transport Layer 5

2.1.3 Network Layer 5

2.1.4 Data link Layer 5

2.1.5 Physical Layer 6

2.2 IEEE 802.3 Ethernet 6

2.2.1 Data link Layer ของ IEEE 802.3 Ethernet 6

2.2.2 Physical Layer ของ IEEE 802.3 Ethernet 7

2.3 IEEE 802.11 Wireless 9

2.3.1 Data link Layer ของ IEEE 802.3 Ethernet 10

สารบัญ (ต่อ)

หน้า

2.3.2 Physical Layer ของ IEEE 802.3 Ethernet 12
2.4 การเขียนโปรแกรมภาษาจาวาโดยใช้อินเทอร์เฟซ 14

บทที่ 3 วิธีการดำเนินงาน
3.1 คำอธิบายของระบบ(System Description) 17
3.2 ความต้องการ(Requirement) 17
3.3 สมมติฐานของการออกแบบ(Design Assumption) 18
3.4 การออกแบบโปรแกรม 18
3.4.1 External และ Dynamic View 19
3.4.2 External และ Static View 24
3.4.3 Internal และ Dynamic View 27

บทที่ 4 ผลการทดลอง
4.1 แผนการทดลอง 36
4.2 การทดสอบคลาส 37
4.2.1 ทดสอบการส่งที่ Layer 2 ของ Ethernet 37
4.2.2 ทดสอบการรับที่ Layer 2 ของ Ethernet 38
4.2.3 ทดสอบการส่งที่ Layer 1 ของ Ethernet 39
4.2.4 ทดสอบการรับที่ Layer 1 ของ Ethernet 40
4.2.5 ทดสอบการส่งที่ Layer 2 ของ Wireless 41
4.2.6 ทดสอบการรับที่ Layer 2 ของ Wireless 43
4.2.7 ทดสอบการส่งเฟรมตอบรับที่ Layer 2 ของ Wireless 43
4.2.8 ทดสอบการรับเฟรมตอบรับที่ Layer 2 ของ Wireless 45
4.2.9 ทดสอบการส่งที่ Layer 1 ของ Wireless 46
4.2.10 ทดสอบการรับที่ Layer 1 ของ Wireless 47
4.3 การทดสอบการส่งข้อมูลตั้งแต่ต้นจนจบ 48
4.3.1 ทดสอบการส่งข้อมูลของ Ethernet Suite ที่เครื่องฝั่งส่ง 48
4.3.2 ทดสอบการส่งข้อมูลของ Ethernet Suite ที่เครื่องฝั่งรับ 49

สารบัญ (ต่อ)

หน้า

4.3.3 ทดสอบการส่งข้อมูลของ Wi-Fi Suite ที่เครื่องฝั่งส่ง 50
4.3.4 ทดสอบการส่งข้อมูลของ Wi-Fi Suite ที่เครื่องฝั่งรับ 52
4.3.5 ทดสอบการส่งข้อมูลของ Ethernet on Wi-Fi Suite ที่เครื่องฝั่งส่ง 54
4.3.6 ทดสอบการส่งข้อมูลของ Ethernet on Wi-Fi Suite ที่เครื่องฝั่งรับ 55
4.4 การทดสอบการใช้งานโปรแกรมบนระบบปฏิบัติการ 56
4.5 การทดสอบหน่วยย่อย 59
4.5.1 ทดสอบ CRC-32 59
4.5.2 ทดสอบ CRC-16 60
4.5.3 ทดสอบการเข้ารหัสและถอดรหัสแบบ 4B/5B 61
4.5.4 ทดสอบการเข้ารหัสและถอดรหัสแบบ 4-PPM 63
4.5.5 ทดสอบการแพคข้อมูล 64
4.5.6 ทดสอบการรับค่าแมคแอดเดรส 65
4.5.7 ทดสอบการคำนวณขนาดข้อมูล 66

บทที่ 5 สรุปผล
5.1 ผลการทดลอง 67
5.1.1 ผลการทดลองการทดสอบคลาส 67
5.1.2 ผลการทดลองการทดสอบตั้งต้นจนจบ 68
5.1.3 ผลการทดลองการทดสอบบนระบบปฏิบัติการ 68
5.1.4 ผลการทดลองการทดสอบหน่วยย่อย 68
5.2 ปัญหาและอุปสรรค 68
5.3 ข้อเสนอแนะ 69
5.4 สรุป 69
เอกสารอ้างอิง 70
ประวัติผู้เขียนโครงงาน 71

สารบัญตาราง

ตารางที่		หน้า
2.1	4B/5B Code ที่ถูกใช้ใน 100BaseX	8
2.2	ค่าของส่วน Type และ Sub-type	11
2.3	ส่วนแอดเดรสของ MAC Frame	11
2.4	Infrared data rate และ direct current level adjustment	14
2.5	การเข้ารหัสแบบ 4-PPM	14
3.1	Mapping UML and PSP Views	18
4.1	ตารางแผนการทดลอง	36
4.2	ตารางการทดสอบการใช้งานโปรแกรมบนระบบปฏิบัติการ	56
4.3	Test case ของ CRC-32	59
4.4	Test case ของ CRC-16	60
4.5	Test case ของการเข้ารหัสแบบ 4B/5B	61
4.6	Test case ของการถอดรหัสแบบ 4B/5B	62
4.7	Test case ของการเข้ารหัสแบบ 4-PPM	63
4.8	Test case ของการถอดรหัสแบบ 4-PPM	63
4.9	Test case ของการแพคข้อมูล	64
4.10	Test case ของการรับค่าแมคแอดเดรส	65
4.11	Test case ของการคำนวณขนาดของข้อมูล	66

สารบัญรูป

รูปที่		หน้า
2.1	โครงสร้างของ TCP/IP Model	4
2.2	ความสัมพันธ์ของมาตรฐาน IEEE LAN	5
2.3	รูปแบบเฟรมของ LLC	6
2.4	รูปแบบเฟรมของ IEEE 802.3 Ethernet	6
2.5	Physical Layer แบบต่างๆใน IEEE 802.3 Ethernet	7
2.6	การ Encapsulate ของ MAC Frame ใน 100BaseX	9
2.7	รูปแบบเฟรมของ IEEE 802.11 Wireless	10
2.8	Physical Layer แบบต่างๆใน IEEE 802.11 Wireless	13
2.9	รูปแบบของ Infrared Frame ของ IEEE 802.11	13
2.10	ตัวอย่างคลาสไดอะแกรมของเลเยอร์ 1	15
3.1	Context diagram ของระบบ	19
3.2	Use case diagram ของระบบ	19
3.3	Sequence diagram ของ Ethernet Suite	21
3.4	Sequence diagram ของ Wi-Fi Suite	22
3.5	Sequence diagram ของ Ethernet on Wi-Fi Suite	23
3.6	Class diagram ของระบบ	24
3.7	Class diagram ของเลเยอร์ 2	25
3.8	Class diagram ของเลเยอร์ 1	25
3.9	Class diagram ของส่วนแสดงผล	26
3.10	Activity diagram ของ Configuration	27
3.11	Activity diagram ของเลเยอร์ 2 Ethernet ที่เป็นผู้ส่งข้อความ	28
3.12	Activity diagram ของเลเยอร์ 2 Ethernet ที่เป็นผู้รับข้อความ	29
3.13	Activity diagram ของเลเยอร์ 1 Ethernet ที่เป็นผู้ส่งข้อความ	30
3.14	Activity diagram ของเลเยอร์ 1 Ethernet ที่เป็นผู้รับข้อความ	31
3.15	Activity diagram ของเลเยอร์ 2 Wi-Fi ที่เป็นผู้ส่งข้อความ	32
3.16	Activity diagram ของเลเยอร์ 2 Wi-Fi ที่เป็นผู้รับข้อความ	33
3.17	Activity diagram ของเลเยอร์ 1 Wi-Fi ที่เป็นผู้ส่งข้อความ	34
3.18	Activity diagram ของเลเยอร์ 1 Wi-Fi ที่เป็นผู้รับข้อความ	35
4.1	รายละเอียดของ Ethernet frame ฝั่งส่ง	37

สารบัญรูป (ต่อ)

รูปที่ หน้า

4.2 รายละเอียดของ Ethernet frame ฝั่งรับ 38
4.3 รายละเอียดของ CAT5 frame ฝั่งส่ง 39
4.4 รายละเอียดของ CAT5 frame ฝั่งรับ 40
4.5 รายละเอียดของ Wi-Fi frame ฝั่งส่ง 41
4.6 รายละเอียดของ Wi-Fi frame ฝั่งรับ 43
4.7 รายละเอียดของเฟรมตอบรับของฝั่งผู้รับ 44
4.8 รายละเอียดของเฟรมตอบรับของฝั่งผู้ส่ง 45
4.9 รายละเอียดของ Infrared frame ฝั่งส่ง 46
4.10 รายละเอียดของ Infrared frame ฝั่งรับ 47
4.11 ลำดับการทำงานของ Ethernet Suite ของฝั่งผู้ส่ง 48
4.12 ลำดับการทำงานของ Ethernet Suite ของฝั่งผู้รับ 49
4.13 ลำดับการทำงานของ Wi-Fi Suite ของฝั่งผู้ส่ง 50
4.14 ลำดับการทำงานของ Wi-Fi Suite ของฝั่งผู้รับ 52
4.15 ลำดับการทำงานของ Ethernet on Wi-Fi Suite ของฝั่งผู้ส่ง 54
4.16 ลำดับการทำงานของ Ethernet on Wi-Fi Suite ของฝั่งผู้รับ 55
4.17 ผลการรันบนระบบปฏิบัติการ Linux ของ Ethernet Suite ฝั่งส่ง 57
4.18 ผลการรันบนระบบปฏิบัติการ Linux ของ Ethernet Suite ฝั่งรับ 58
4.19 ผลที่ได้จากการทดสอบ CRC-16 โดย input คือ 1 59
4.20 ค่าที่ได้จากการคำนวณ CRC-16 โดย input คือ 1 59
4.21 ผลที่ได้จากการทดสอบCRC-32 โดย input คือ 111 59
4.22 ค่าที่ได้จากการคำนวณCRC-32 โดย input คือ 111 59
4.23 ผลที่ได้จากการทดสอบCRC-32 โดย input คือ 1010 60
4.24 ค่าที่ได้จากการคำนวณCRC-32 โดย input คือ 1010 60
4.25 ผลที่ได้จากการทดสอบ CRC-16 โดย input คือ 1 60
4.26 ค่าที่ได้จากการคำนวณ CRC-16 โดย input คือ 1 60
4.27 ผลที่ได้จากการทดสอบ CRC-16 โดย input คือ 111 61
4.28 ค่าที่ได้จากการคำนวณ CRC-16 โดย input คือ 111 61
4.29 ผลที่ได้จากการทดสอบ CRC-16 โดย input คือ 1010 61
4.30 ค่าที่ได้จากการคำนวณ CRC-16 โดย input คือ 1010 61

สารบัญรูป (ต่อ)

รูปที่ หน้า

4.31 ผลที่ได้จากการทดสอบการเข้าและถอดรหัสแบบ 4B/5B โดย input คือ 1010100011000000 62

4.32 ผลที่ได้จากการทดสอบการเข้าและถอดรหัสแบบ 4B/5B โดย input คือ 101101001011010 62

4.33 ผลที่ได้จากการทดสอบการเข้าและถอดรหัสแบบ 4B/5B โดย input คือ 101101001011010 62

4.34 ผลที่ได้จากการทดสอบการเข้าและถอดรหัสแบบ 4-PPM โดย input คือ 1000100010000001 63

4.35 ผลที่ได้จากการทดสอบการเข้าและถอดรหัสแบบ 4-PPM โดย input คือ 1000100010000001 10001000 64

4.36 ผลที่ได้จากการทดสอบการเข้าและถอดรหัสแบบ 4-PPM โดย input คือ 01000001 64

4.37 ผลที่ได้จากการทดสอบการแพคข้อมูลโดย input คือ 1010 64

4.38 ผลที่ได้จากการทดสอบการแพคข้อมูลโดย input คือ 10 65

4.39 ผลที่ได้จากการทดสอบการแพคข้อมูลโดย input คือ 10100000 65

4.40 ผลที่ได้จากการทดสอบการแพคข้อมูลโดย input คือ 1010000000 65

4.41 ผลที่ได้จากการทดสอบการรับค่าแมคแอดเดรสโดย input คือ AA-BB-CC-DD-EE-FF 65

4.42 ผลที่ได้จากการทดสอบการรับค่าแมคแอดเดรสโดย input คือ BB-DA-5C-4F-A1-D2 66

4.43 ผลที่ได้จากการทดสอบการคำนวณขนาดข้อมูลโดย input คือ ข้อมูลขนาด 3 ไบท์ 66

4.44 ผลที่ได้จากการทดสอบการคำนวณขนาดข้อมูลโดย input คือ ข้อมูลขนาด 4 ไบท์ 66

4.45 ผลที่ได้จากการทดสอบการคำนวณขนาดข้อมูลโดย input คือ ข้อมูลขนาด 0 ไบท์ 66

บทที่ 1
บทนำ

1.1 ที่มาและความสำคัญของโครงงาน

ในการจำลองเน็ตเวิร์คโพรโตคอลแสตกที่ได้สร้างไลบรารีสำหรับจำลองการทำงานของระบบเครือข่ายขึ้นมาก่อนหน้านี้นั้นได้ทำการอิมพลีเมนต์ที่เลเยอร์ 3 และเลเยอร์ 4 ไว้ เพื่อช่วยให้การจำลองระบบเครือข่ายมีความสมบูรณ์และเพิ่มความเข้าใจในระบบเครือข่ายให้มากขึ้น จึงมีความคิดที่จะอิมพลีเมนต์ในส่วนของเลเยอร์ 1 และเลเยอร์ 2 เพิ่มเติม ซึ่งได้ทำการเลือก Ethernet และ Wireless ที่มีความนิยมมากในปัจจุบันมาเป็นตัวอย่างในการจำลอง ซึ่งไลบรารีที่ทำการสร้างขึ้นนี้ง่ายต่อการพัฒนาสามารถเพิ่มโพรโตคอลใหม่ๆที่ต้องการศึกษาเข้าไปได้

1.2 วัตถุประสงค์โครงงาน

1. เพื่อสร้างตัวอย่างสำหรับจำลองเน็ตเวิร์คโพรโตคอลแสตกสำหรับเลเยอร์ 1 และเลเยอร์ 2
2. เพื่อให้ผู้ใช้สามารถนำไลบรารีไปพัฒนาเพิ่มเติมได้

1.3 ขอบเขตของโครงงาน

1. สามารถสร้างโครงร่างให้ผู้อื่นสามารถใช้งานได้
2. จำลองการทำงานที่เลเยอร์ 1 และเลเยอร์ 2 ของ Ethernet เพื่อเป็นตัวอย่างได้
3. จำลองการทำงานที่เลเยอร์ 1 และเลเยอร์ 2 ของ Wireless เพื่อเป็นตัวอย่างได้
4. สามารถเชื่อมต่อกับเลเยอร์เดิมที่สร้างมาก่อนหน้านี้ได้

1.4 ขั้นตอนการดำเนินงาน

1. ศึกษาความรู้ทั่วไปเกี่ยวกับ TCP model
2. ศึกษาการทำงานของโปรแกรมที่สร้างมาก่อนหน้านี้
3. ศึกษาการทำงานของเฟรม Ethernet
4. ศึกษาการทำงานของเฟรม Wireless
5. สร้างไลบรารี ให้ได้ตามวัตถุประสงค์
6. ทดสอบความถูกต้องของ ไลบรารี
7. จัดทำเอกสารและคู่มือการใช้

1.5 แผนการดำเนินงาน

กิจกรรม	ปี 2551							ปี 2552			
	มิ.ย.	ก.ค.	ส.ค.	ก.ย.	ต.ค.	พ.ย.	ธ.ค.	ม.ค.	ก.พ.	มี.ค.	เม.ย.
1. ศึกษาความรู้ทั่วไปเกี่ยวกับ TCP model											
2. ศึกษาการทำงานของโปรแกรมที่สร้างมาก่อนหน้านี้											
3. ศึกษาการทำงานของเฟรม Ethernet											
4.ศึกษาการทำงานของเฟรม Wireless											
5. สร้างไลบรารีให้ได้ตามวัตถุประสงค์											
6. ทดสอบความถูกต้องของไลบรารี											
7. จัดทำเอกสารและคู่มือการใช้											

1.6 ผลที่คาดว่าจะได้รับ

1. เพื่อช่วยในการศึกษาเกี่ยวกับเน็ตเวิร์คโพรโตคอลแสตกได้ง่ายขึ้น
2. เพื่อใช้ในการจำลองเน็ตเวิร์คโพรโตคอลแสตกได้ง่ายขึ้น
3. ได้ตัวอย่างการรับส่งผ่านเลเยอร์ 1 Ethernet
4. ได้ตัวอย่างการรับส่งผ่านเลเยอร์ 2 Ethernet
5. ได้ตัวอย่างการรับส่งผ่านเลเยอร์ 1 Wireless
6. ได้ตัวอย่างการรับส่งผ่านเลเยอร์ 2 Wireless
7. แสดงให้เห็นลำดับการทำงานของเน็ตเวิร์คโพรโตคอลแสตกของชุดโพรโตคอลต่างๆ

1.7 รายละเอียดงบประมาณของโครงงาน

1. หนังสือประกอบการทำโครงงาน	1,000 บาท	
2. ค่าเอกสาร	500 บาท	
3. อื่นๆ	500 บาท	
รวมเป็นเงินทั้งสิ้น	2,000 บาท	(สองพันบาทถ้วน)

หมายเหตุ ขออนุมัติถัวเฉลี่ยทุกรายการ

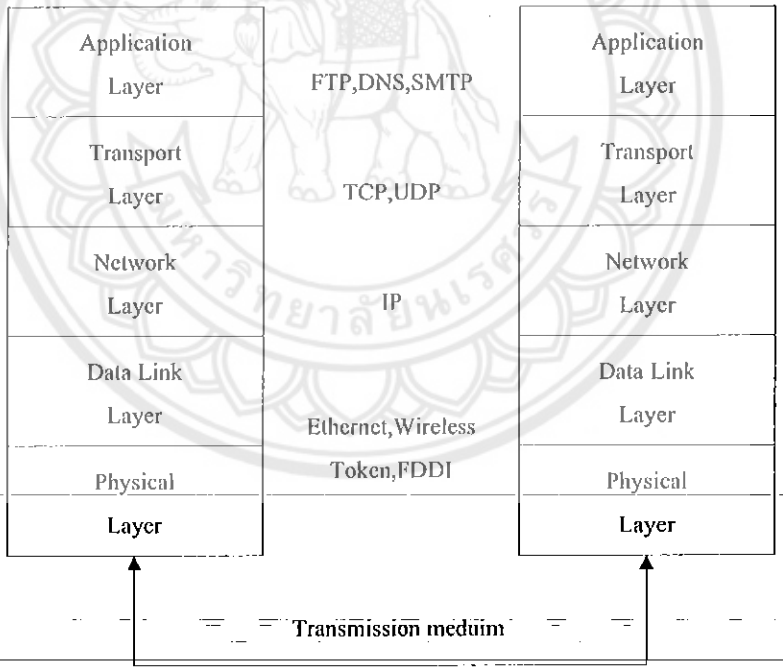
มหาวิทยาลัยนเรศวร

บทที่ 2
หลักการและทฤษฎี

ในบทนี้จะกล่าวถึงทฤษฎี และหลักการในการพัฒนาโปรแกรมเพื่อจำลองการทำงานของเน็ตเวิร์คโพรโตคอลแสตก โดยเริ่มจากภาพรวมของ TCP/IP Protocol Suite จากนั้นจะเน้นไปที่ Data Link Layer และ Physical Layer ซึ่งในสองชั้นนี้ได้เลือกใช้มาตรฐาน IEEE[1] 802.3 Ethernet และ IEEE 802.11 Wireless ซึ่งเป็นโพรโตคอลที่ใช้ในการจำลองของโครงงานนี้ รวมถึงการเขียนโปรแกรมภาษาจาวาโดยใช้อินเทอร์เฟซ

2.1 TCP/IP Protocol Suite

ถูกพัฒนามาจาก OSI[2] Model ประกอบไปด้วย 5 Layer คือ Application Layer, Transport Layer, Network Layer, Data Link Layer, Physical Layer



รูปที่ 2.1 โครงสร้างของ TCP/IP Model

[1] IEEE (The Institute of Electrical and Electronics Engineers) สถาบันที่กำกับ ดูแลมาตรฐานวิจัยและพัฒนาความรู้ เน้นด้านไฟฟ้ากำลัง คอมพิวเตอร์

[2] OSI (Open Systems Interconnection) รูปแบบโครงสร้างมาตรฐานสากลสำหรับการติดต่อสื่อสารระหว่างเครื่องคอมพิวเตอร์

2.1.1 Application Layer

มีหน้าที่สนับสนุนแอพพลิเคชั่นของเครือข่าย ซึ่งมีหลายโพรโตคอลที่เป็นที่รู้จัก เช่น HTTP ซึ่งช่วยในการสร้างรูปแบบหน้าจอ และ FTP ทำการจัดการแฟ้มข้อมูล

2.1.2 Transport Layer

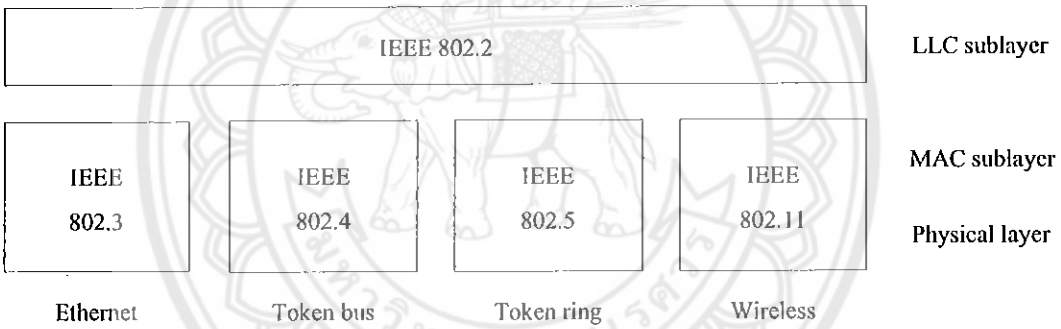
ทำหน้าที่ในการรับข้อมูลมาจากชั้น Application Layer ซึ่งอาจต้องแบ่งข้อมูลออกเป็นแพ็กเก็ตขนาดย่อมหลายแพ็กเก็ต แล้วจึงส่งข้อมูลทั้งชุดต่อไป

2.1.3 Network Layer

ทำหน้าที่รับผิดชอบในการควบคุมการติดต่อรับ-ส่งข้อมูลระหว่างโหนดต่างๆ ในระบบเครือข่ายให้เป็นไปด้วยความเรียบร้อย

2.1.4 Data Link Layer

แบ่งได้เป็น 2 ชั้นสื่อสารย่อย (Sublayer) คือ Media Access Control (MAC) และ Logical Link Control (LLC) ซึ่ง IEEE ได้จัดมาตรฐานดังรูป

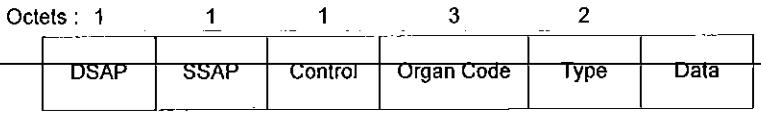


รูปที่ 2.2 ความสัมพันธ์ของมาตรฐาน IEEE LAN

ในส่วนนี้จะกล่าวถึงชั้นสื่อสารย่อย LLC ก่อนเพราะเป็นโพรโตคอลที่ใช้ร่วมกันกับทุกชุดโพรโตคอล (Protocol Suite) โดยปกติ LLC มักจะรวมกับ SNAP[3] เฮดเดอร์ (Sub Network Access Protocol Header) เข้าไปด้วย ซึ่งมีหน้าที่บอกว่า Data Link Layer นั้นเชื่อมต่อกับโพรโตคอลใด

[3] SNAP เป็นส่วนขยายของ IEEE 802.2 LLC ซึ่งถูกใช้กับมาตรฐาน IEEE 802.3, IEEE 802.4, IEEE 802.5, IEEE 802.11

โครงสร้างของ LLC Header



รูปที่ 2.3 รูปแบบเฟรมของ LLC

DSAP (Destination Service Access Point) เป็นจุดให้บริการในการส่งข้อมูลไปยัง Network Layer

SSAP (Source Service Access Point) เป็นจุดให้บริการในการรับข้อมูลจากชั้น Network Layer

Control เป็นส่วนควบคุม

Organ code และ Type เป็นส่วนของ SNAP เฮดเดอร์

สำหรับ TCP/IP จะมีค่าต่างๆ เป็นดังนี้

- DSAP : 170 (decimal) หรือ 0xAA (hex)
- SSAP : 170 (decimal) หรือ 0xAA (hex)
- Control : 3
- Organ Code : 0
- Type : 2048 (decimal) หรือ 0x8000 (hex)
- Data ข้อมูลที่รับมาจากชั้น Network Layer

ชั้นสื่อสารย่อย MAC มีหน้าที่ตรวจจับข้อผิดพลาด กำหนดที่อยู่ในการรับและส่งข้อมูล

2.1.5 Physical Layer

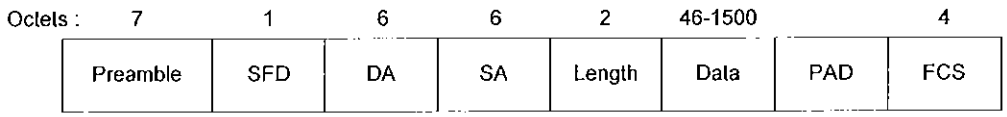
เกี่ยวข้องกับการส่งบิตระหว่างการติดต่อผ่านสื่อต่างๆ โดยมีหน้าที่ดังนี้ แปลงบิตให้อยู่ในรูปของสัญญาณที่เหมาะสมสำหรับการส่งผ่านสื่อต่างๆ การเข้ารหัสต่างๆ

2.2 IEEE 802.3 Ethernet

เป็นที่นิยมในวงการ LAN (Local Area Network) ซึ่งสามารถรันบน coaxial cable, twisted-pair copper wire, fiber optic ก็ได้ และสามารถส่งข้อมูลได้ในอัตราเร็วที่ต่างกัน

2.2.1 Data link Layer ของ IEEE 802.3 Ethernet

MAC Frame Format (IEEE 802.3)



รูปที่ 2.4 รูปแบบเฟรมของ IEEE 802.3 Ethernet

- Preamble มีทั้งหมด 7 ไบท์ โดยแต่ละไบท์เป็น 10101010 มีหน้าที่ทำการเข้าจังหวะสัญญาณนาฬิการะหว่างผู้รับและผู้ส่ง
- Start Frame Delimiter (SFD) มีขนาด 1 ไบท์ บอกให้ทราบว่าเป็นจุดเริ่มของเฟรม
- Destination Address (DA) มีขนาด 6 ไบท์ ทำหน้าที่เก็บ MAC address ของสถานีปลายทางซึ่งเป็นฝ่ายรับเฟรม
- Source Address (SA) มีขนาด 6 ไบท์ ทำหน้าที่เก็บ MAC address ของสถานีต้นทางซึ่งเป็นฝ่ายส่งเฟรม
- Length มีขนาด 2 ไบท์ มีไว้บอกจำนวนไบท์ของข้อมูลใน Data Field
- Data Field มีขนาด 46-1500 ไบท์
- PAD ถ้าข้อมูลใน Data Field มีน้อยกว่า 46 ไบท์ ซึ่งสามารถทราบจากส่วนของ Length Field ส่วนของ PAD จะทำการเพิ่มขนาดเพื่อให้เฟรมมีขนาดที่เล็กที่สุดที่จะส่งไปได้
- Frame Check Sequence (FCS) มีขนาด 4 ไบท์ มีไว้ตรวจจับข้อผิดพลาด โดยใช้ CRC-32 โพลิโนเมียล (Cyclic Redundancy Check-32 Polynomial) ทำการคำนวณบน DA, SA, Length, Data Field, PAD

อัลกอริทึมสำหรับการคำนวณ CRC เป็นดังนี้

กำหนด G(x) เป็น generator polynomial

1. ให้ r เป็นดีกรีของ G(x) ให้เติม '0' จำนวน r บิตเข้าไปที่บิตต่ำของเฟรมนั้น
2. ทำการหารเลขจำนวนนั้นด้วย G(x) โดยใช้การ exclusive or
3. ผลลัพธ์ที่ได้คือค่า CRC นำมาต่อกับข้อมูล

ค่า G(x) ของ CRC-32 คือ $x^{32}+x^{26}+x^{23}+x^{22}+x^{16}+x^{12}+x^{11}+x^{10}+x^{8}+x^{7}+x^{5}+x^{4}+x^{2}+x+1$

2.2.2 Physical Layer ของ IEEE 802.3 Ethernet

มีด้วยกันหลายชนิดซึ่งจำแนกได้จากความเร็วที่ใช้ส่งข้อมูล ชนิดของตัวกลางในการส่ง การเข้ารหัสที่ใช้ชนิดของสัญญาณ เช่น 10 Mbps Ethernet, Fast Ethernet

Data Link	802.2 LLC				
Layer	802.3 Ethernet				
Physical Layer	10Base5 Thick Coax	10Base2 Thin Coax	10Broad36 Broadband	1Base5 StarLAN	10BaseT Twisted Pair

รูปที่ 2.5 Physical Layer แบบต่างๆ ใน IEEE 802.3 Ethernet

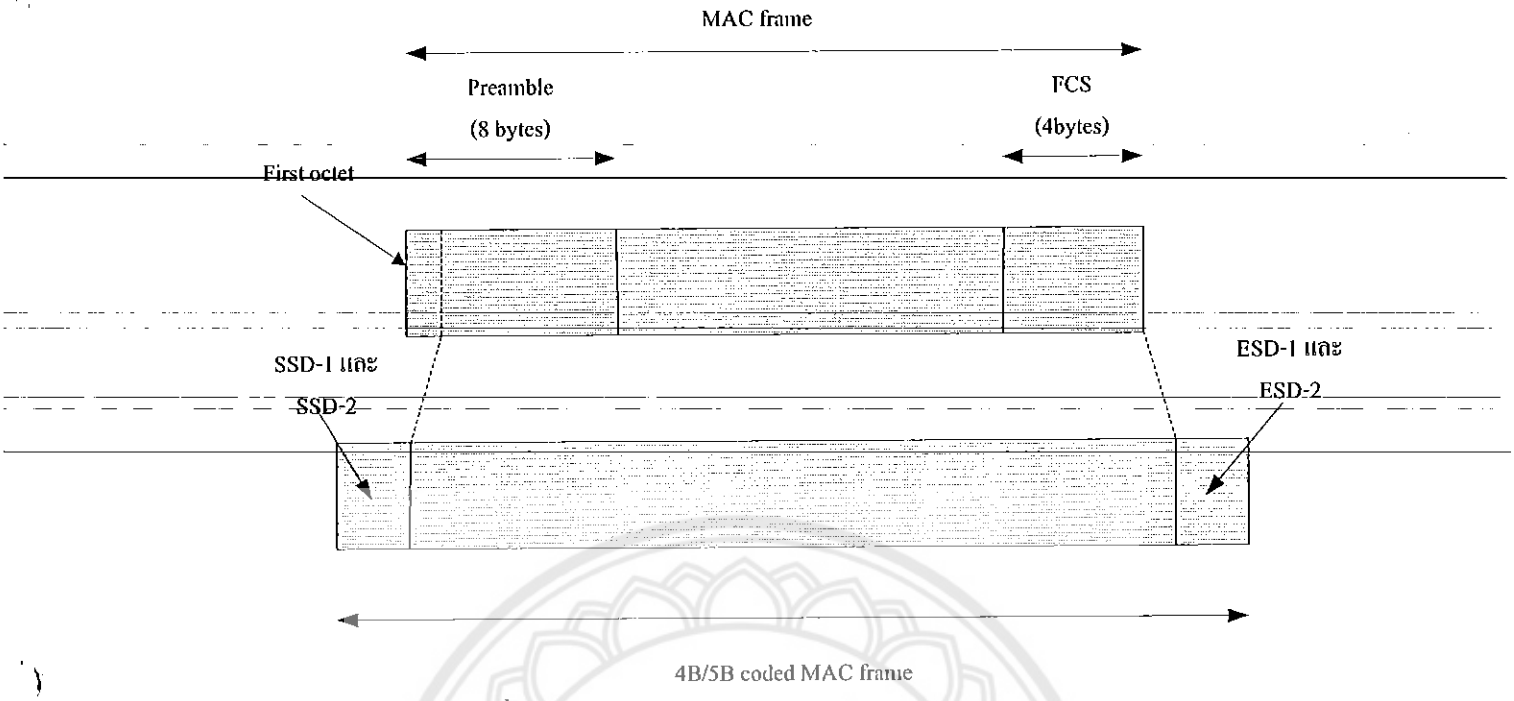
ซึ่งในโครงงานนี้ใช้โพรโตคอลของ 100BaseTX ซึ่งเป็น Fast Ethernet แบบหนึ่งที่ใช้สายแบบ CAT5 UTP ซึ่งมีการเข้ารหัสแบบ 4B/5B ดังตารางที่ 2.1

ตารางที่ 2.1 4B/5B Code ที่ถูกใช้ใน 100BaseX

Hex/Name	4-bit nibble	5-bit code	Hex/Name	4-bit nibble	5-bit code
0	0000	11110	B	1011	10111
1	0001	01001	C	1100	11010
2	0010	10100	D	1101	11011
3	0011	10101	E	1110	11100
4	0100	01010	F	1111	11101
5	0101	01011	I(Idle)	-	11111
6	0110	01110	J(SSD-1)	-	11000
7	0111	01111	K(SSD-2)	-	10001
8	1000	10010	T(ESD-1)	-	01101
9	1001	10011	R(ESD-2)	-	00111
A	1010	10110	H	-	00100

โดยมีการ Encapsulate[4] เฟรมที่มาจากชั้นสื่อสารย่อย MAC ดังรูปที่ 2.6

[4] Encapsulate คือการเพิ่มเฮดเดอร์(ส่วนหัว) และ/หรือ เทรลเลอร์(ส่วนหาง) ให้กับข้อมูล



รูปที่ 2.6 การ Encapsulate ของ MAC เฟรมใน 100BaseX

โดยมี 4 โค้ดที่ใช้เป็น Start-of-Stream Delimiter (SSD) และ End-of-Stream Delimiter (ESD) ซึ่งแต่ละ MAC เฟรมจะถูก Encapsulate ระหว่างโค้ดทั้งสี่นี้ ซึ่งไบท์แรกของส่วนของ Preamble ใน MAC เฟรมจะถูกแทนที่ด้วย SSD-1 และ SSD-2 จากนั้นต่อ ESD-1 และ ESD-2 กับส่วน FCS ของ MAC เฟรม

2.3 IEEE 802.11 Wireless

ระบบเครือข่ายไร้สาย (Wireless LAN: WLAN) เป็นระบบเชื่อมโยงระหว่างคอมพิวเตอร์หรือเครือข่าย คอมพิวเตอร์ที่ใช้สายเข้าด้วยกันหรือเชื่อมต่อกับอินเตอร์เน็ต โดยอาศัยคลื่นวิทยุ ซึ่งใช้มาตรฐาน IEEE 802.11 หรือมีชื่อเล่นว่า Wi-Fi[5] มาตรฐานนี้ทำงานได้สองสภาวะคือ

1. ในสภาวะที่มีสถานีฐาน การสื่อสารทั้งหมดจะถูกส่งผ่านสถานีฐานซึ่งเรียกว่าจุดติดต่อ (Access Point)
2. ในสภาวะที่ไม่มีสถานีฐาน เครื่องคอมพิวเตอร์จะสามารถสื่อสารระหว่างกันได้โดยตรง และไม่มีการติดต่อกับเน็ตเวิร์คภายนอกเรียกว่าเป็นแบบ Ad-Hoc

โดยโครงงานนี้จะเป็นการจำลองแบบ Ad-Hoc

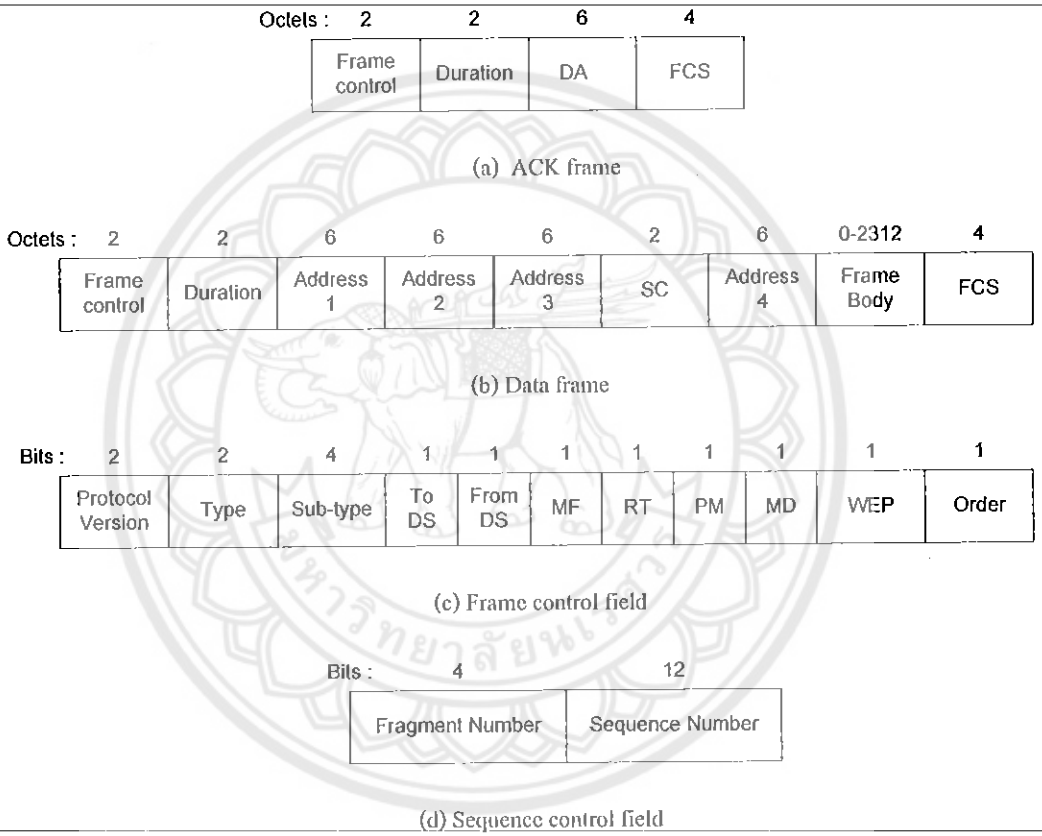
[5] Wi-Fi (Wireless Fidelity) องค์กรที่ทำการทดสอบผลิตภัณฑ์ Wireless LAN หรือเน็ตเวิร์คแบบไร้สายด้วยเทคโนโลยีการสื่อสารภายใต้มาตรฐาน IEEE 802.11

2.3.1 Data link layer ของ IEEE 802.11 Wireless

MAC Frame Format (IEEE802.11)

โดยขอบเขตของโครงงานนี้ประกอบด้วย MAC เฟรม 2 ชนิดด้วยกัน คือ

1. เฟรมข้อมูล (Data Frame) โดยมีรูปแบบตามรูปที่ 2.7(b)
2. เฟรมตอบรับ (Acknowledgement Frame) มีรูปแบบตามรูปที่ 2.7(a) ซึ่งมีความสำคัญมากเพราะไม่เหมือนกับกรณีของ Ethernet ผู้ส่งที่เป็น Wireless ไม่สามารถทราบได้ว่าเฟรมที่ส่งไปนั้นได้ถึงผู้รับหรือไม่



รูปที่ 2.7 รูปแบบของเฟรม IEEE 802.11

Frame control (FC) มีขนาด 2 ไบท์ทำหน้าที่บอกชนิดของเฟรมโดยมีรูปแบบตามรูปที่ 2.7(c) ซึ่งภายในประกอบไปด้วยส่วนต่างๆ ดังนี้

- Protocol version (2บิต) เป็นส่วนที่บอกเวอร์ชันของโพรโตคอล IEEE 802.11 ที่ใช้ปัจจุบันใช้ 00
- Type (2บิต) เป็นส่วนที่บอกว่าเป็นเฟรมชนิดใด (control, data, management) ซึ่งมีค่าตามตารางที่ 2.2
- Sub-type (4บิต) เป็นส่วนที่บอก Sub-type ของเฟรม ซึ่งมีค่าตามตารางที่ 2.2

ตารางที่ 2.2 ค่าของส่วน Type และ Sub-type

• Management Type (00)	
	o Association Request/Response (0000/0001)
	o Reassociation Request/Response (0010/0011)
	o Probe-Request/Response (0100/0101)
	o Beacon (1000)
	o ATIM: Announcement Traffic Indication Map (1001)
	o Dissociation (1010)
• Control Type (01)	
	o Power Save Poll (1010)
	o RTS/CTS (1011/1100)
	o ACK (1101)
	o CF End/CF End with ACK (1110/1111)
• Data Type (10)	
	o Data/Data with CF ACK/No Data (0000/0001)
	o Data Poll with CF/Data Poll with CF and ACK (0010/0011)
	o No Data/CF ACK (0100/0101)
	o CF Poll/CF Poll ACK (0000/0001)
• Unspecified (11)	

- To DS (1บิต) มีค่าตามตารางที่ 2.3
- From DS (1บิต) มีค่าตามตารางที่ 2.3

ตารางที่ 2.3 ส่วนแอดเดรสของ MAC Frame

	To DS	From DS	Address1	Address2	Address3	Address4
Independent BBS	0	0	DA	SA	BSS-Id	-
From AP	0	1	DA	BSS-Id	SA	-
To AP	1	0	BSS-Id	SA	DA	-
Within DS	1	1	RA	TA	DA	SA

- MF (More Fragment 1บิต) เป็น 1 ถ้ายังมี fragment[6] อื่นตามมาอีก
- RT (Retry 1บิต) เป็น 1 ถ้าเฟรมนี้เป็นการส่งเฟรมก่อนหน้านี้อีกครั้ง
- PM (Power Management 1บิต) เป็น 1 ถ้าสถานีส่งอยู่ในสภาวะหลับ(sleep mode)
- MD (More Data 1บิต) เป็น 1 เมื่อชี้ว่าสถานีส่งยังมีข้อมูลที่จะส่งอยู่อีก
- WEP (Wired Equivalent Privacy 1บิต) เป็น 1 เมื่อบิตข้อมูลมีการเข้ารหัส
- Order (1บิต) เป็น 1 เมื่อบริการบนชั้นสื่อสารย่อย MAC เป็นแบบ "Strictly Ordered"

Duration มีขนาด 1 ไบท์ เป็นส่วนที่บอกเวลาเป็น microsecond

Address fields มีขนาดแต่ละแอดเดรสเป็น 6 ไบท์ โดยมีการใช้ตามตารางที่ 2.3

DA : Destination Address

SA : Source Address

BSS-Id : BSS[7] Identifier

RA : Receiver Address

TA : Transmitter Address

ซึ่ง RA และ TA จะอ้างถึงแอดเดรสของจุดติดต่อ (AP) ภายในระบบ

Sequence control (SC) มีขนาด 2 ไบท์ แบ่งเป็น 2 ส่วนมีรูปแบบตามรูปที่ 2.7(d) โดยส่วนแรกเป็นส่วนของ fragment number ซึ่งใช้สำหรับการแบ่งข้อมูล (Fragmentation) และการรวมข้อมูล (Reassembly) ส่วนที่สองเป็นส่วนของ sequence number ซึ่งทำหน้าที่บอกลำดับของเฟรมที่ทำการส่งระหว่างเครื่องส่งและเครื่องรับ

CRC (Cyclic Redundancy Check) ใช้ CRC-32 เพื่อตรวจสอบข้อผิดพลาด

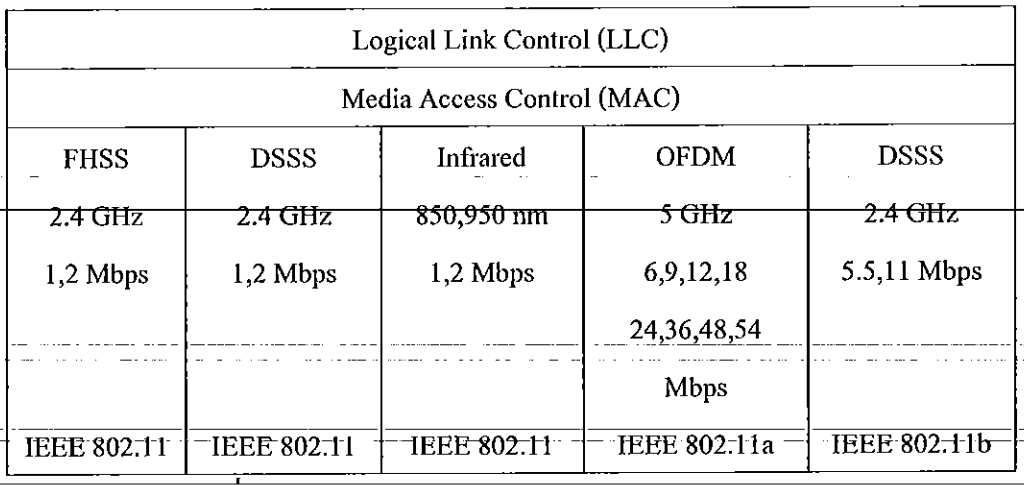
2.3.2 Physical layer ของ IEEE 802.11 Wireless

Physical layer ของ IEEE 802.11 แบ่งออกได้เป็นหลายลักษณะซึ่งจำแนกได้ตามความถี่ของสัญญาณ บิตเรต และการเข้ารหัส ตามรูปที่ 2.8

---

[6] Fragment ข้อมูลที่ถูกแบ่งเป็นส่วนย่อยเพื่อเพิ่มความน่าเชื่อถือของข้อมูลและลดการสูญเสียข้อมูลจากการรบกวน

[7] BSS กลุ่มของอุปกรณ์ไร้สายตั้งแต่ 2 เครื่องขึ้นไปที่อยู่ในขอบเขตพื้นที่บริการเดียวกัน และอยู่ในรัศมีที่สามารถส่งสัญญาณถึงกันได้



รูปที่ 2.8 Physical Layer แบบต่างๆ ใน IEEE 802.11 Wireless

ซึ่งภายในโครงงานนี้ใช้มาตรฐาน IEEE 802.11 Infrared ที่บิตเรต 2 Mbps

PHY[8] Frame Format (IEEE 802.11 Infrared)

รูปที่ 2.9 รูปแบบของอินฟราเรดเฟรมของ IEEE 802.11

- Synchronization (SYNC) มีขนาด 57 บิต เป็นพัลส์ 0, 1 สลับกันมีหน้าที่ทำการเข้าจังหวะสัญญาณนาฬิการะหว่างผู้รับและผู้ส่ง
- Start Frame Delimiter (SFD) มีขนาด 4 บิต มีค่าเป็น 1001 เสมอ บอกให้ทราบว่าเป็นจุดเริ่มของเฟรม
- Data Rate มีขนาด 3 บิต บอกความเร็วในการส่ง มีค่าตามตารางที่ 2.4
- Direct Current Level Adjustment (DCLA) มีขนาด 32 บิต ทำหน้าที่เก็บรูปแบบของอินฟราเรดพัลส์ และช่วยบอกให้อุปกรณ์รับคำนวณหาระดับสัญญาณ มีค่าตามตารางที่ 2.4

[8] PHY หมายถึง Physical Layer

ตารางที่ 2.4 Infrared Data Rate และ Direct Current Level Adjustment

Transmission Speed (Mbps)	Data Rate Value	Direct Current Level Adjustment Value
1.0	000	00000000010000000000000001000000
2.0	001	00100010001000100010001000100010

- Length มีขนาด 16 บิตบอกขนาดของข้อมูล
- Frame Correction Sequence (FCS) มีขนาด 16 บิตใช้ตรวจสอบข้อผิดพลาดโดยใช้ CRC-16
- การหา CRC-16 ให้ดูอัลกอริทึมการหา CRC ด้านบน โดยใช้ค่า G(x) คือ $x^{16}+x^{15}+x^2+1$
- Data ที่ความเร็ว 2 Mbps ใช้การเข้ารหัสแบบ 4-PPM (Pulse Position Modulation) ตามตารางที่ 2.5

ตารางที่ 2.5 การเข้ารหัสแบบ 4-PPM

Data Bit	4-PPM Value
00	0001
01	0010
11	0100
10	1000

2.4 การเขียนโปรแกรมภาษาจาวาโดยใช้อินเทอร์เฟซ

อินเทอร์เฟซ (Interface) ทำงานในลักษณะที่เป็นเทมเพลต หรือทำเป็นต้นแบบเอาไว้ แต่ไม่ได้กำหนดคำสั่งการทำงานใดๆ ให้กับเมธอดในอินเทอร์เฟซ โดยมีลักษณะเด่นดังนี้

1. ไม่ใช่คลาส
2. การเรียกใช้งานใช้คำว่า implements
3. คลาสที่ implement จะต้องระบุเมธอดที่มีอยู่ในอินเทอร์เฟซให้ครบทุกเมธอด

รูปแบบอินเทอร์เฟซมีดังนี้

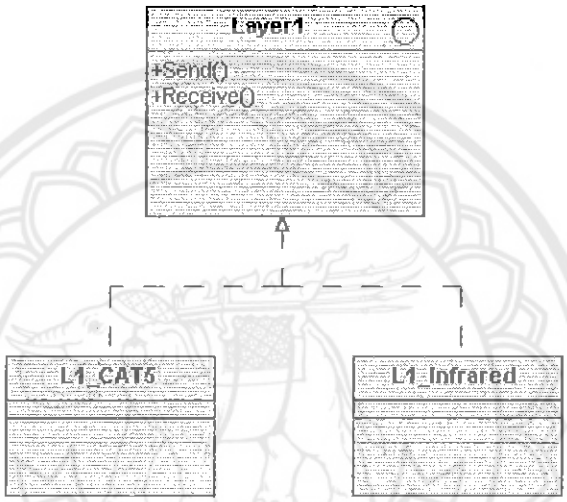
```
public interface ชื่อของ interface {
// รูปแบบของเมธอดภายในอินเทอร์เฟซมีดังนี้
Public  ชนิดของการคืนค่า  ชื่อของเมธอด ;
}
```

การเรียกใช้งานอินเทอร์เฟซ สามารถเรียกใช้งานโดยใช้คำสั่ง implements ซึ่งมีรูปแบบในการใช้งานดังนี้

```
public class ชื่อคลาส implements ชื่ออินเทอร์เฟซ {
}
```

ตัวอย่างการใช้งาน

การสร้างและเรียกใช้งานอินเทอร์เฟซซึ่งเป็นต้นแบบของเลเยอร์ 1 โดยมีทั้งของ Ethernet และ WI-FI



รูปที่ 2.10 ตัวอย่างคลาสไดอะแกรมของเลเยอร์ 1

1. เริ่มจากการสร้างอินเทอร์เฟซ

```
public interface Layer1 {
        public void Send( );
        public void Receive( );
}
```

2. เรียกใช้งานอินเทอร์เฟซของคลาส L1_CAT5

```
Public class L1_CAT5 implements Layer1 {
        Public void Send() {
        // อิมพลีเมนต์โค้ดตามจุดประสงค์
        }
```

```
        Public void Receive() {
        // อิมพลีเมนต์โค้ดตามจุดประสงค์
        }
}
```

3. เรียกใช้งานอินเทอร์เฟซของคลาส L1_Infrared

```
Public class L1_Infrared implements Layer1 {
        Public void Send() {
        // อิมพลีเมนต์โค้ดตามจุดประสงค์
        }
        Public void Receive() {
        // อิมพลีเมนต์โค้ดตามจุดประสงค์
        }
}
```

บทที่ 3
วิธีการดำเนินงาน

3.1 คำอธิบายของระบบ (System Description)

โปรแกรมจำลองการทำงานสำหรับเน็ตเวิร์คโพรโตคอล TCP/IP เลเยอร์ 1 และเลเยอร์2 นั้น ทำการจำลองการเชื่อมต่อระหว่างเลเยอร์ 1 และเลเยอร์ 2 กับเลเยอร์อื่นของเน็ตเวิร์คโพรโตคอล TCP/IP ที่ได้ทำการสร้างขึ้นมาก่อนหน้านี้ ทำให้การจำลองมีความสมบูรณ์มากยิ่งขึ้น และมีการแสดงค่าต่างๆ ภายในโพรโตคอลนั้นๆ

3.2 ความต้องการ (Requirement)

- แสดงให้เห็นว่าเลเยอร์2 (Data Link Layer) ซึ่งในการจำลองทำการเลือกโพรโตคอล 802.3 Ethernet และ โพรโตคอล 802.11 Wireless ว่ามีโครงสร้างเฟรมเป็นอย่างไร ประกอบไปด้วยอะไร
- ภายใน โพรโตคอล Ethernet และ Wireless มีการจัดการเฟรมเป็นอย่างไร และมีบริการอะไร
- แสดงให้เห็นว่าเลเยอร์ 1 (Physical Layer) ซึ่งในการจำลองทำการเลือกโพรโตคอล 802.11 Infrared และ Fast Ethernet แบบ 100BaseX ซึ่งใช้สาย CAT5 ว่ามีโครงสร้างเฟรมเป็นอย่างไร ประกอบไปด้วยอะไร
- ภายในโพรโตคอล Infrared และ 100BaseX มีการจัดการเฟรมอย่างไร และมีบริการอะไร
- แสดงให้เห็นว่าเลเยอร์ทั้งสอง มีการติดต่อกับเลเยอร์อื่นอย่างไรในเน็ตเวิร์คโพรโตคอล TCP/IP ทั้งฝั่งรับ และฝั่งส่ง
- สามารถเลือกดูชุดโพรโตคอลได้ 3 แบบคือ Ethernet Suite ,Wi-Fi Suite ,Ethernet on Wi-Fi[1]
- สามารถเลือกดูการแสดงผลภายในโพรโตคอลได้ 3 แบบ คือ แบบข้อความ แบบ GUI และแบบคอมโบ(แบบข้อความ+แบบGUI)

[1] Ethernet on Wi-Fi หมายถึง เลเยอร์ 2 เป็น Ethernet เลเยอร์ 1 เป็น Wireless

3.3 สมมติฐานของการออกแบบ(Design Assumption)

- การสื่อสารข้อมูลเป็นแบบ Ad-Hoc Network

3.4 การออกแบบโปรแกรม

การออกแบบโปรแกรมใช้มาตรฐาน UML (Unified Modeling Language) ที่เป็นภาษาที่ใช้แสดงแบบการทำงานของระบบ ซึ่งอธิบายพฤติกรรมของซอฟแวร์ด้วยแผนภาพ ตามวิวต่างๆ ดังตาราง

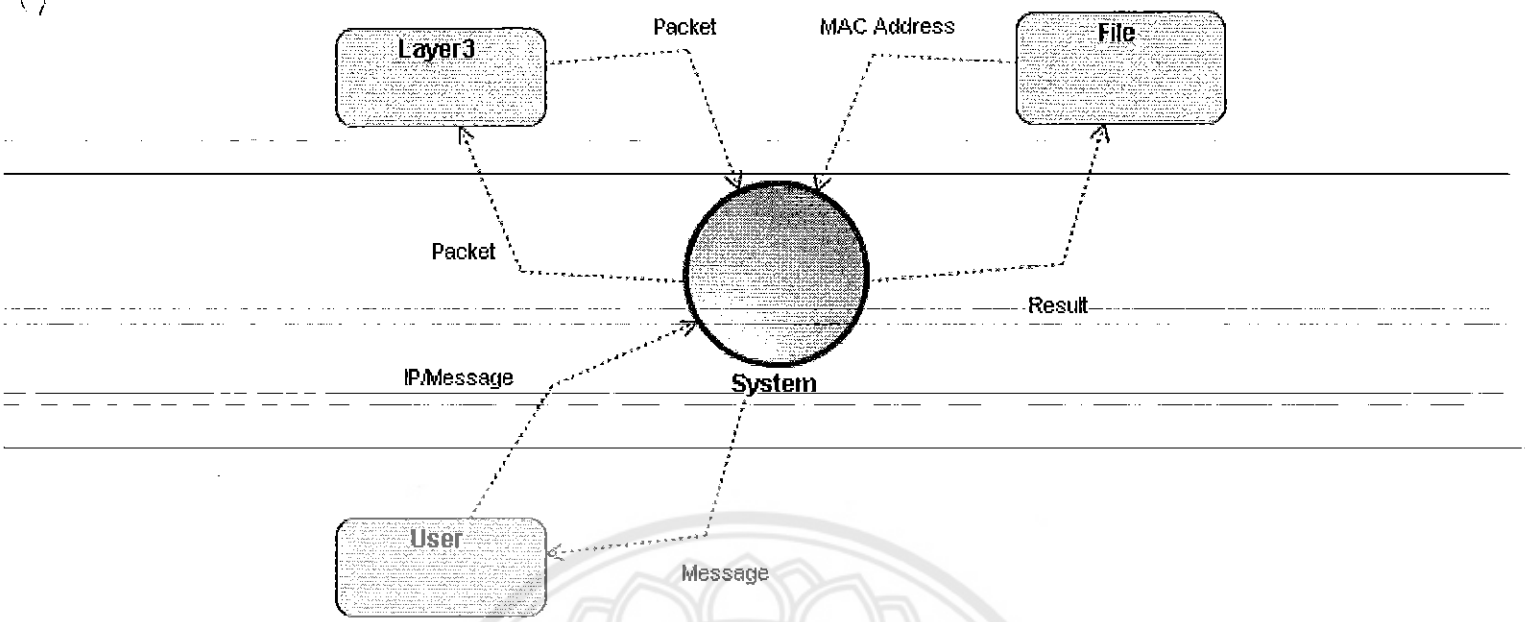
ตารางที่ 3.1 Mapping UML and PSP[2] Views

	Dynamic	Static
External	Use cases Diagrams Sequence Diagrams	Class Diagrams
Internal	Activity Diagrams	

การออกแบบระบบนั้นมีขั้นตอนดังนี้

การออกแบบระบบนั้นเริ่มจากการใช้ Context Diagram เพื่อดูภาพรวมของทั้งระบบก่อน เพื่อดูการเชื่อมต่อของระบบและองค์ประกอบต่างๆ รวมทั้งพฤติกรรมขององค์ประกอบเหล่านั้นโดยเป็นไปตามรูปที่ 3.1 และเพื่อให้ภาพมากขึ้นจะขออธิบายการออกแบบระบบในวิวต่างๆ ของ PSP ตามลำดับจากภายนอกจนถึงภายใน

[2] PSP (The Personal Software Process) กระบวนการที่ช่วยจัดการคุณภาพของโครงการ ช่วยปรับปรุงการประเมินและวางแผน ช่วยลดข้อผิดพลาดของผลิตภัณฑ์

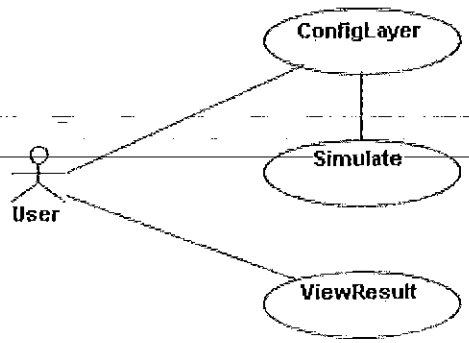


รูปที่ 3.1 Context diagram ของระบบ

3.4.1 External และ Dynamic View

Use case diagram

แสดงให้เห็นถึงวิธีการที่ผู้ใช้มีการติดต่อกับระบบ โดยยูสเซอร์เลือกชุดโพรโตคอล ไอพีที่ต้องการส่ง ข้อความที่ต้องการส่ง เลือกว่าจะแสดงข้อมูลเป็นแบบข้อความ หรือแบบกราฟฟิค เลือกว่าจะเป็นฝั่งรับ หรือฝั่งส่งผ่าน ConfigLayer ส่วนการจำลองการทำงาน (Simulate) ของระบบนั้นทำผ่าน ConfigLayer เช่นเดียวกันซึ่งส่วนของการสร้างคลาส ConfigLayer นั้นสามารถดูได้ในส่วนของ class diagram รูปที่ 3.6 และ ยูสเซอร์สามารถดูข้อมูล (ViewResult) ได้ซึ่งคลาสของการดูข้อมูลนั้นจะอยู่ในส่วนของคลาสของการแสดงผลในรูปที่ 3.9

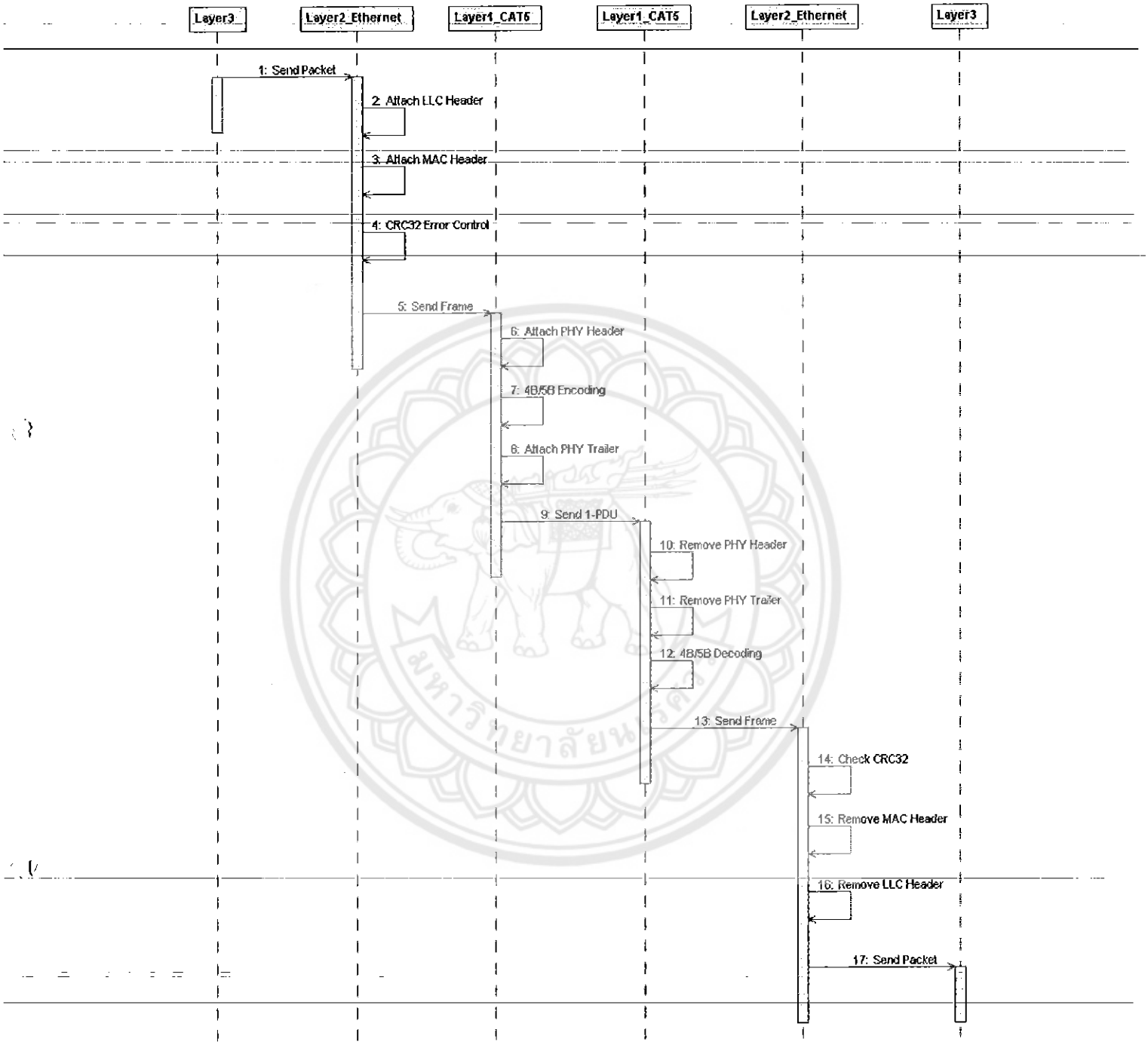


รูปที่ 3.2 Use case diagram ของระบบ

Sequence diagram

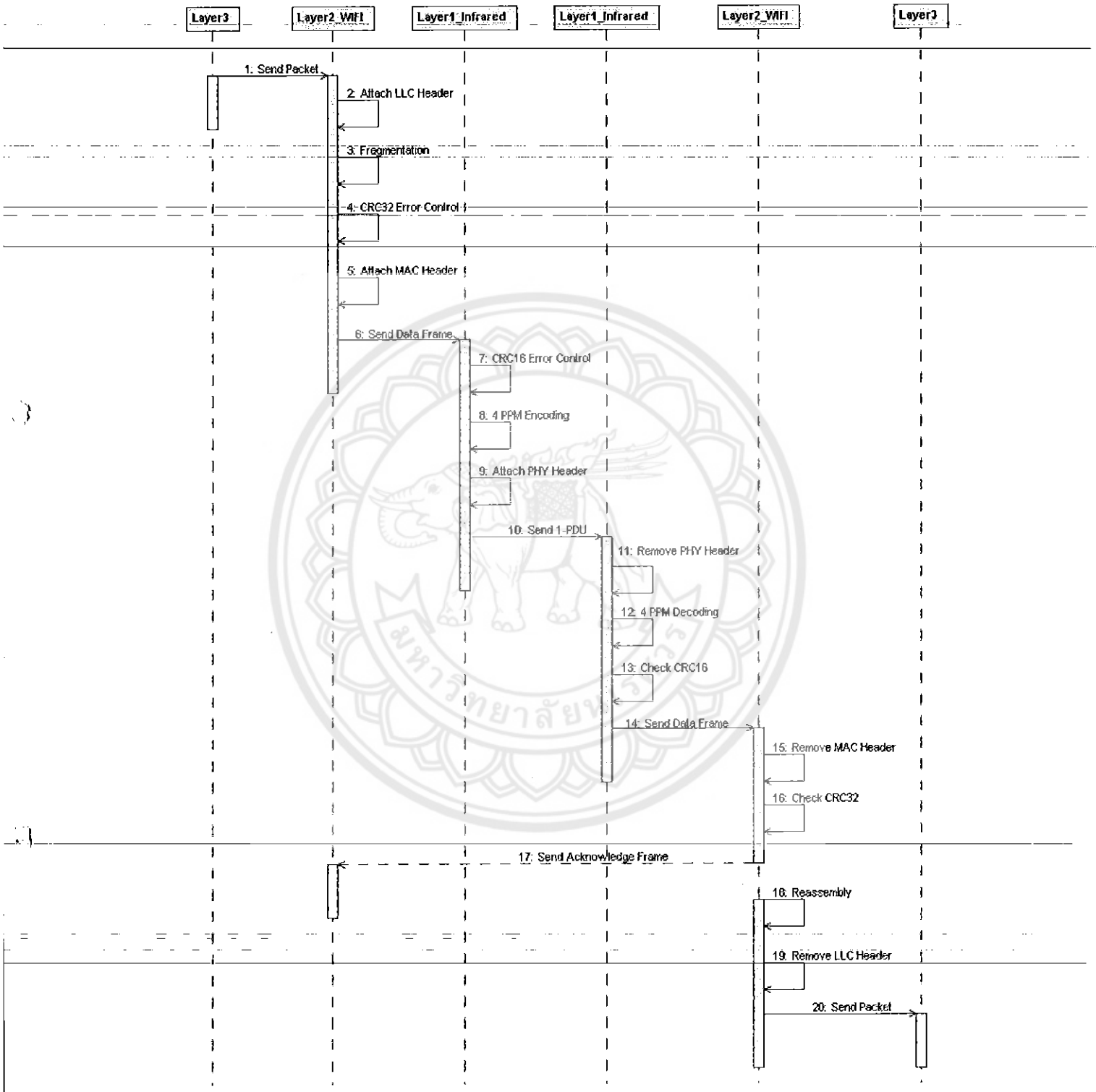
แสดงให้เห็นถึงวิธีการที่ระบบมีการติดต่อกับเลเยอร์อื่นในเน็ตเวิร์คโพรโตคอล TCP/IP ในชุดโพรโตคอลต่างๆ ทั้ง Ethernet suite, Wi-Fi suite, Ethernet on Wi-Fi ซึ่งการกระทำภายในของแต่ละเลเยอร์นั้นสามารถดูได้อย่างละเอียดในหัวข้อ Activity diagram

Sequence diagram ของ Ethernet Suite



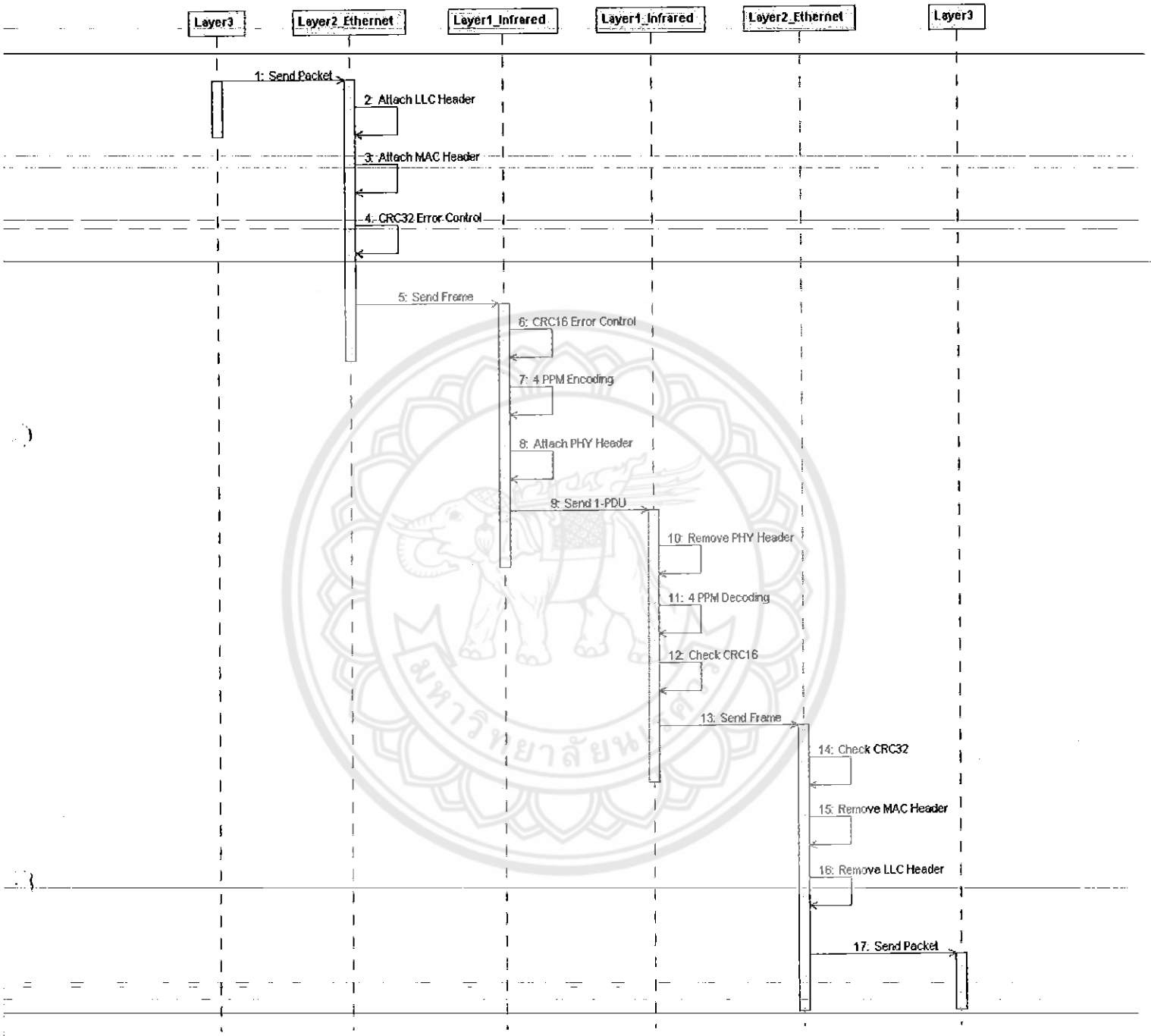
รูปที่ 3.3 Sequence diagram ของ Ethernet Suite

Sequence diagram ของ Wi-Fi Suite



รูปที่ 3.4 Sequence diagram ของ Wi-Fi Suite

Sequence diagram ของ Ethernet on Wi-Fi



รูปที่ 3.5 Sequence diagram ของ Ethernet on Wi-Fi

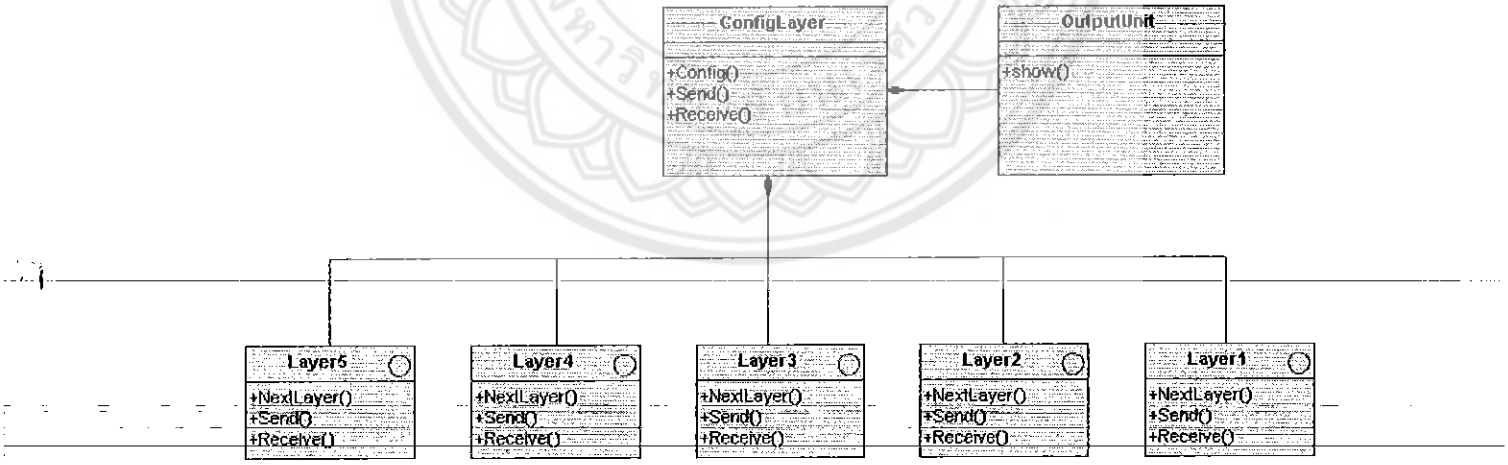
3.4.2 External และ Static View

Class diagram

สร้างคลาสเพื่อรองรับการทำงานของโปรแกรมโดยคลาสที่สร้างทำหน้าที่ดังนี้

- ConfigLayer มีทั้งหมดอยู่ 3 เมธอดคือ
  1. เมธอด Config(..) ทำหน้าที่เลือกว่าจะเป็นฝั่งส่งหรือฝั่งรับ เลือกรูปแบบการแสดงผล เลือกชุดโพรโตคอล
  2. เมธอด Send(..) ทำหน้าที่เป็นฝ่ายส่งข้อมูลให้กับระบบโดยเริ่มส่งจากเลเยอร์ 4
  3. เมธอด Receive(..) ทำหน้าที่เป็นฝ่ายรับข้อมูลกับระบบโดยเริ่มรับจากเลเยอร์ 1

  ซึ่งก็คือส่วนของเมธอด Send(..) และ Receive(..) จะเป็นส่วนของการ Simulate ของโปรแกรมตามรูปที่ 3.2
- OutputUnit เป็นส่วนแสดงผลลัพธ์โดยที่มี 3 แบบ คือ แบบ TextOutput GUIOutput ComboOutput มีเมธอด show ใช้แสดงผล
- Layer ทั้ง 5 เลเยอร์มีทั้งหมด 3 เมธอด คือ
  1. เมธอด NextLayer(..) ทำหน้าที่เชื่อมต่อเลเยอร์ที่ติดกัน
  2. เมธอด Send(..) ทำหน้าที่ส่งข้อมูลไปเลเยอร์ที่ติดกัน
  3. เมธอด Receive(..) ทำหน้าที่รับข้อมูลจากเลเยอร์ที่ติดกัน

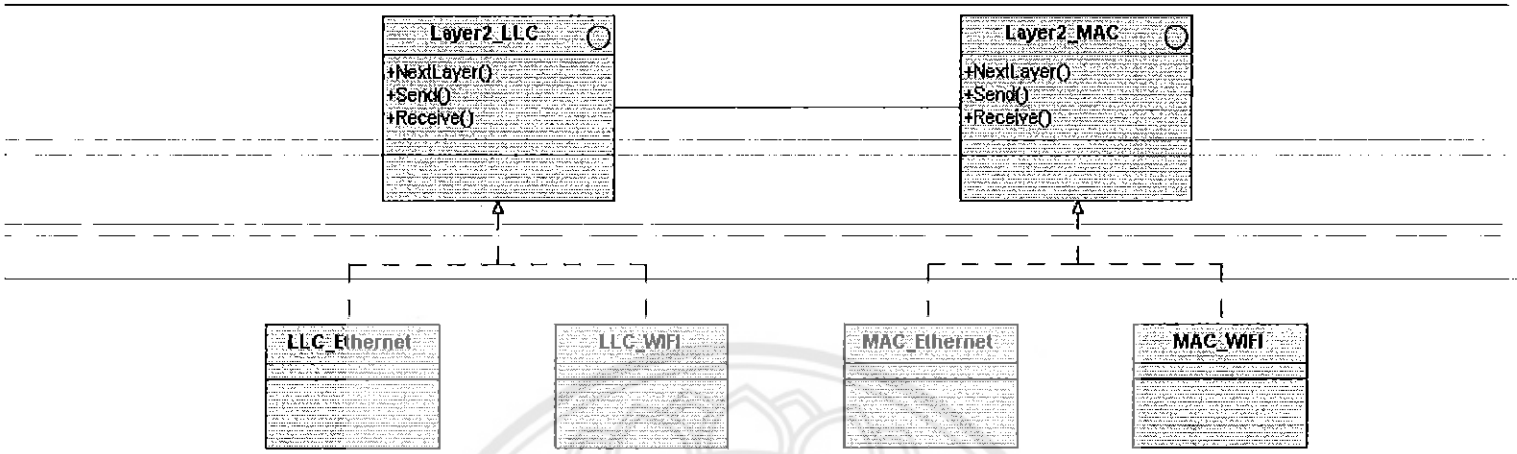


รูปที่ 3.6 Class diagram ของระบบ

ห้องสมุดคณะวิศวกรรมศาสตร์ มหาวิทยาลัยนเรศวร

14995697

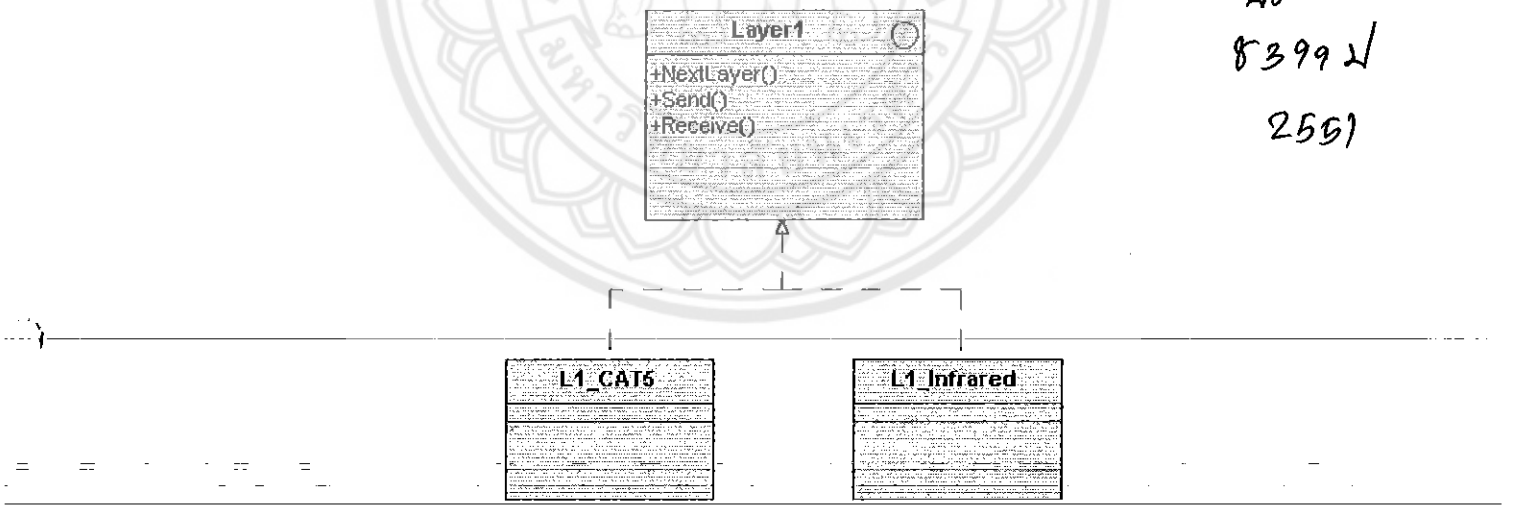
องค์ประกอบของคลาสเลเยอร์2 มีทั้งหมด 2 ส่วนคือ ส่วนของ MAC Layer และส่วนของ LLC Layer และในสองส่วนนั้นยังแบ่งเป็นแบบ Wi-Fi และแบบ Ethernet



รูปที่ 3.7 Class diagram ของเลเยอร์ 2

องค์ประกอบของคลาสเลเยอร์1 แบ่งได้เป็น 2 ส่วน คือ แบบ Wi-Fi และแบบ Ethernet

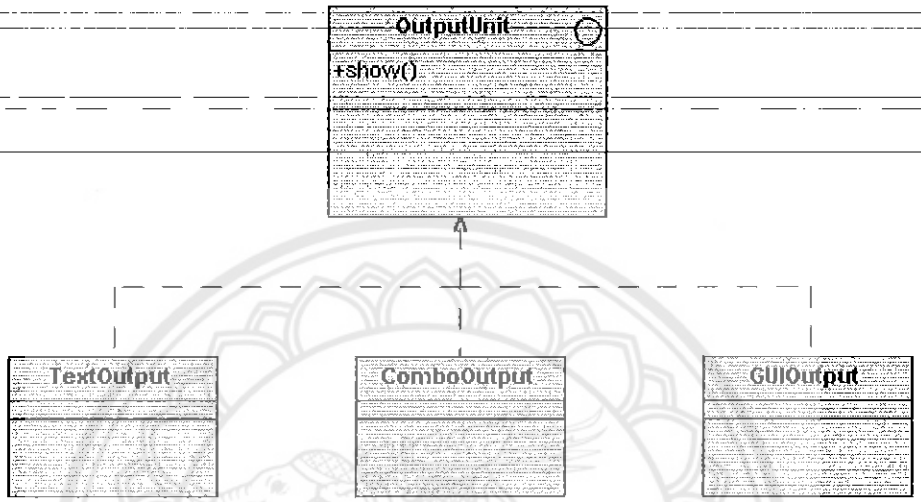
ส5.
8399 ง
2551



รูปที่ 3.8 Class diagram ของเลเยอร์1

องค์ประกอบของคลาสในส่วนแสดงผลมีทั้งหมด 3 ส่วนคือ

1. TextOutput แสดงผลเป็นแบบข้อความ
2. GUIOutput แสดงผลเป็นแบบกราฟฟิค
3. ComboOutput แสดงผลทั้ง 2 แบบพร้อมกัน



รูปที่ 3.9 Class diagram ของส่วนแสดงผล

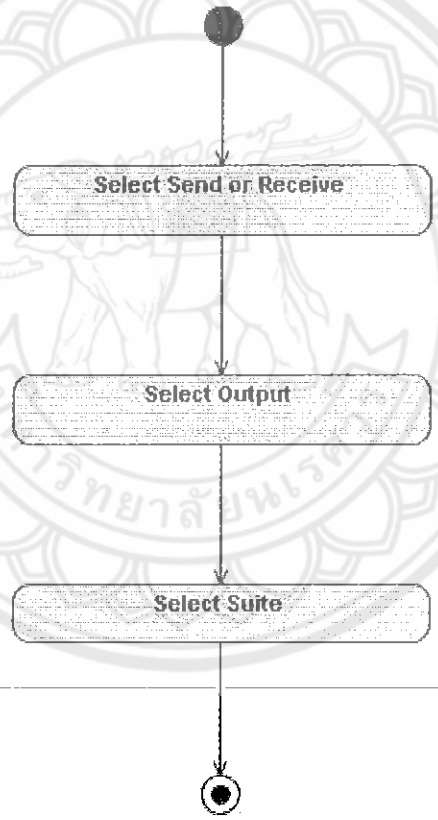
3.4.3 Internal และ Dynamic View

Activity diagram

แสดงให้เห็นถึงวิธีการทำงานภายในคลาสที่สร้างขึ้นมา

การไหลของเหตุการณ์ (Flow of Events) ของ Configuration

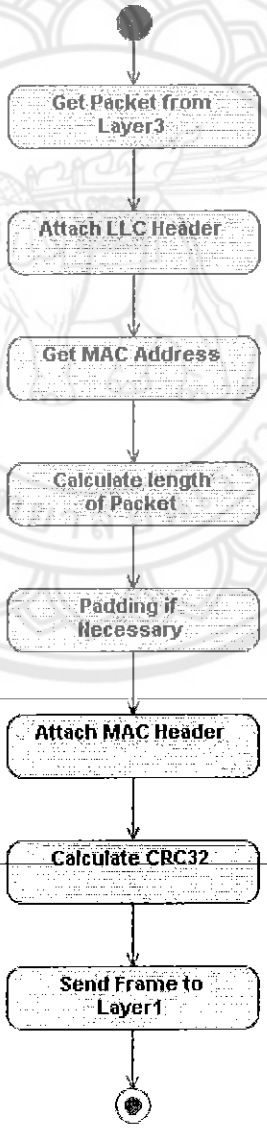
1. เมื่อเข้าสู่โปรแกรมให้ยูสเซอร์ทำการเลือกว่าจะเป็นฝ่ายส่งหรือรับข้อความ จากนั้นถ้าเป็นฝ่ายส่งให้ใส่ข้อความและกรอกไอพีของฝ่ายรับ
2. ยูสเซอร์ทำการเลือกดูผลลัพธ์ที่ต้องการ แบบข้อความ แบบกราฟิก หรือจะเรียกดูพร้อมกัน
3. ยูสเซอร์ทำการเลือกชุดโพรโตคอลที่ต้องการ Ethernet Suite, Wi-Fi Suite, Ethernet on Wi-Fi



รูปที่ 3.10 Activity Diagram ของ Configuration

การไหลของเหตุการณ์ (Flow of Events) ของเลเยอร์ 2 ของ Ethernet ฝ่ายส่ง

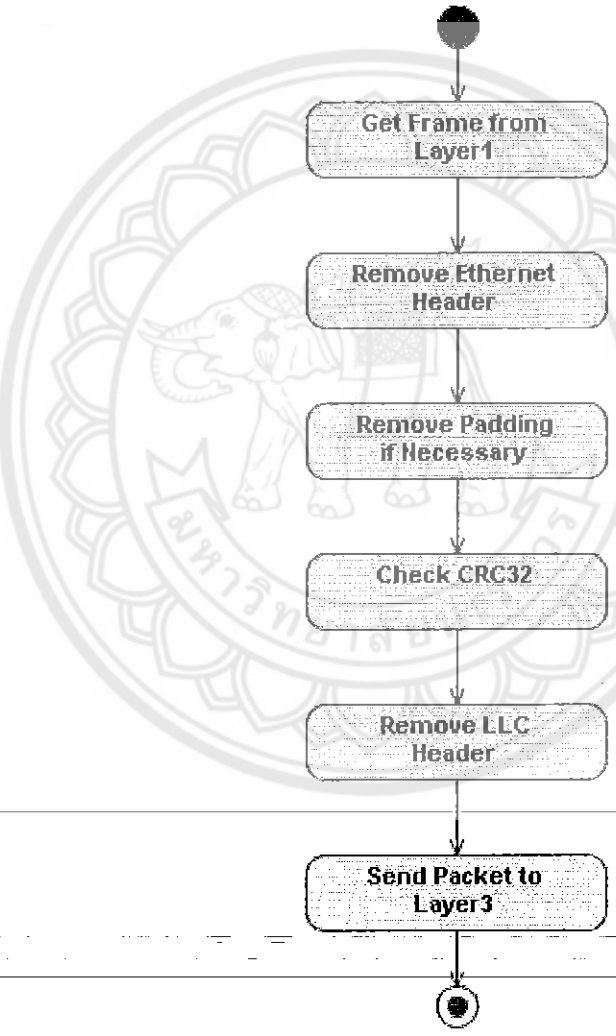
1. เลเยอร์ 2 รับข้อมูลที่ส่งมาจากเลเยอร์ 3
2. เลเยอร์ 2 ทำการใส่ LLC เฮดเดอร์กับข้อมูลที่ส่งมาจากเลเยอร์ 3
3. เลเยอร์ 2 รับค่า MAC แอดเดรสของเครื่องส่งและเครื่องรับจากไฟล์
4. เลเยอร์ 2 ทำการคำนวณขนาดของข้อมูลที่มาจาก LLC ซับเลเยอร์ว่ามีขนาดกี่ไบท์
5. เลเยอร์ 2 ทำการแพดถ้าข้อมูลที่มาจาก LLC ซับเลเยอร์มีขนาดไม่ถึง 46 ไบท์
6. เลเยอร์ 2 ทำการใส่ MAC เฮดเดอร์กับข้อมูลที่มาจาก LLC ซับเลเยอร์
7. เลเยอร์ 2 ทำการคำนวณค่า CRC-32 จากนั้นทำไปติดที่ส่วนท้ายของข้อมูล
8. เลเยอร์ 2 ทำการส่งเฟรมไปที่เลเยอร์ 1



รูปที่ 3.11 Activity Diagram ของเลเยอร์ 2 Ethernet ที่เป็นผู้ส่งข้อความ

การไหลของเหตุการณ์ (Flow of Events) ของเลเยอร์ 2 ของ Ethernet ฝ่ายรับ

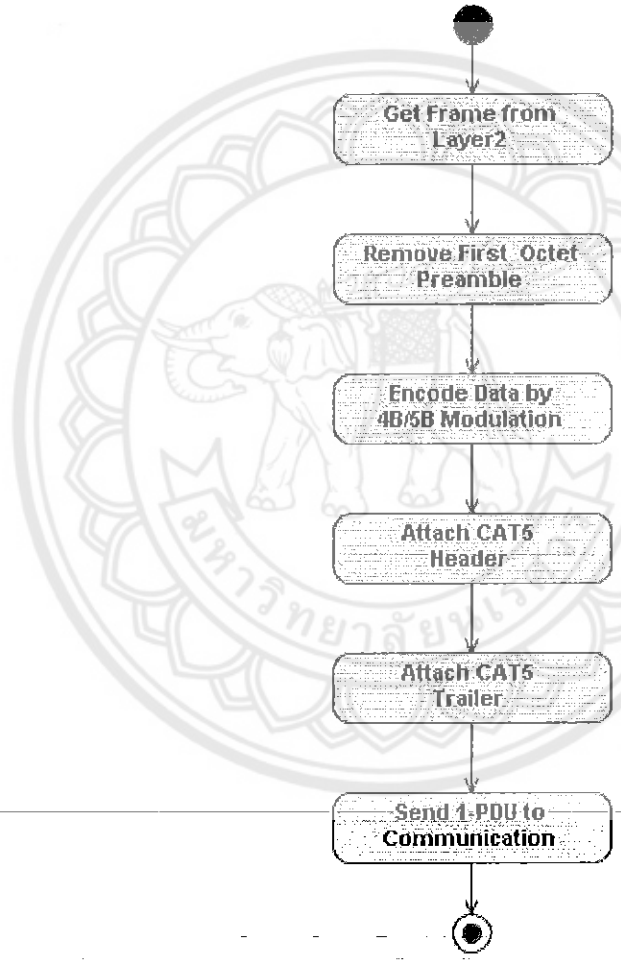
1. เลเยอร์ 2 รับข้อมูลมาจากเลเยอร์ 1
2. เลเยอร์ 2 ถอดส่วน Preamble, Destination address, Source address ออก
3. เลเยอร์ 2 ตรวจสอบขนาดของข้อมูลถ้ามีการแพดให้ทำการถอดออก
4. เลเยอร์ 2 ตรวจสอบ CRC-32 ถ้าตรงกับเทรลเลอร์ที่ส่งมาให้ส่งขึ้น LLC ซับเลเยอร์
5. เลเยอร์ 2 ถอด LLC เฮดเดอร์ออก
6. เลเยอร์ 2 ส่งข้อมูลต่อไปยังเลเยอร์ 3



รูปที่ 3.12 Activity Diagram ของเลเยอร์ 2 Ethernet ที่เป็นผู้รับข้อความ

การไหลของเหตุการณ์ (Flow of Events) ของเลเยอร์ 1 ของ Ethernet ฝ่ายส่ง

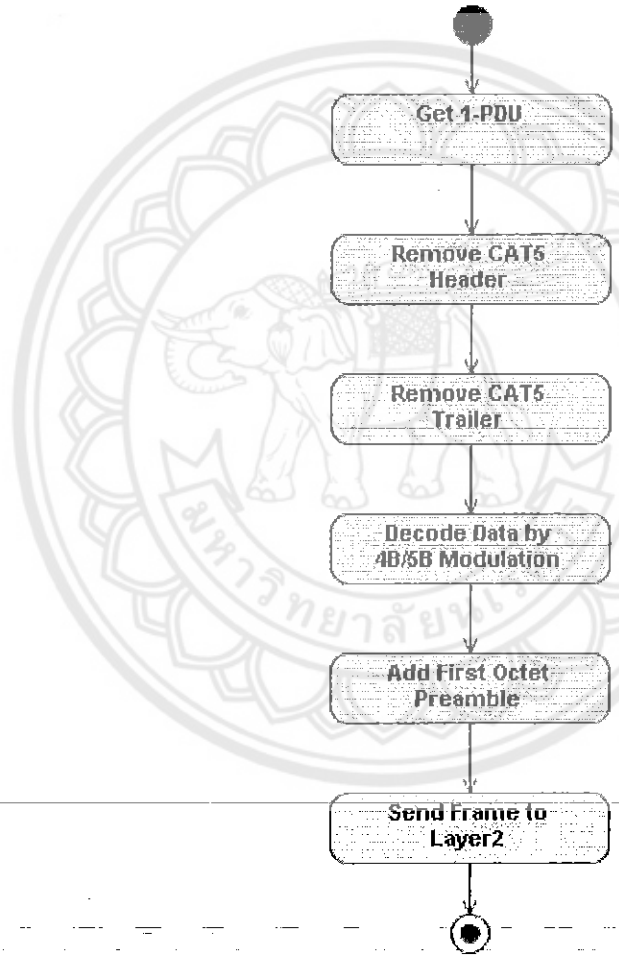
1. เลเยอร์ 1 รับข้อมูลมาจากเลเยอร์ 2
2. เลเยอร์ 1 ทำการถอดไบท์แรกของส่วน Preamble
3. เลเยอร์ 1 ทำการเข้ารหัสข้อมูลที่เหลือทั้งหมดแบบ 4B/5B
4. เลเยอร์ 1 ทำการใส่ SSD-1 และ SSD-2 (ส่วนเฮดเดอร์)
5. เลเยอร์ 1 ทำการใส่ ESD-1 และ ESD-2 (ส่วนเทรลเลอร์)
6. เลเยอร์ 1 ทำการส่งข้อมูลเพื่อการสื่อสารต่อไป



รูปที่ 3.13 Activity Diagram ของเลเยอร์ 1 Ethernet ที่เป็นผู้ส่งข้อความ

การไหลของเหตุการณ์ (Flow of Events) ของเลเยอร์ 1 ของ Ethernet ฝ่ายรับ

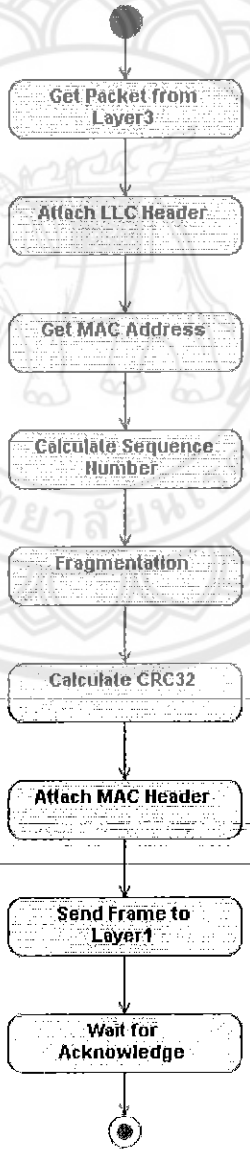
1. เลเยอร์ 1 รับข้อมูลมาจากการสื่อสาร
2. เลเยอร์ 1 ทำการถอดเฮดเดอร์ออก
3. เลเยอร์ 1 ทำการถอดเทรลเลอร์ออก
4. เลเยอร์ 1 ทำการถอดรหัสข้อมูล
5. เลเยอร์ 1 ทำการเพิ่มไบท์แรกของส่วน Preamble
6. เลเยอร์ 1 ทำการส่งข้อมูลไปยังเลเยอร์ 2 ต่อไป



รูปที่ 3.14 Activity Diagram ของเลเยอร์ 1 Ethernet ที่เป็นผู้รับข้อความ

การไหลของเหตุการณ์ (Flow of Events) ของเลเยอร์ 2 ของ Wi-Fi ฝ่ายส่ง

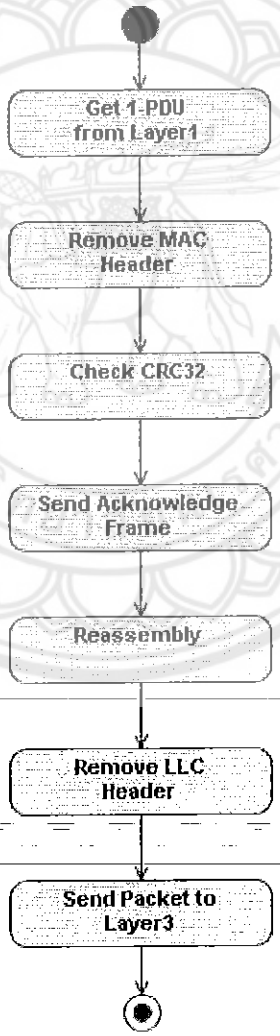
1. เลเยอร์ 2 รับข้อมูลมาจากเลเยอร์ 3
2. เลเยอร์ 2 ทำการใส่ LLC เฮดเดอร์กับข้อมูลที่มาจากเลเยอร์ 3
3. เลเยอร์ 2 รับค่า MAC แอดเดรสจากไฟล์
4. เลเยอร์ 2 คำนวณค่า Sequence number
5. เลเยอร์ 2 คำนวณค่า Fragment number เพื่อใช้ในการแบ่งข้อมูล
6. เลเยอร์ 2 ทำการใส่ MAC เฮดเดอร์
7. เลเยอร์ 2 คำนวณ CRC-32 จากนั้นนำไปติดที่ส่วนท้ายของข้อมูล (เทรลเลอร์)
8. เลเยอร์ 2 ส่งข้อมูลต่อไปยังเลเยอร์ 1
9. เลเยอร์ 2 รอรับเฟรมตอบรับ



รูปที่ 3.15 Activity Diagram ของเลเยอร์ 2 Wi-Fi ที่เป็นผู้ส่งข้อความ

การไหลของเหตุการณ์ (Flow of Events) ของเลเยอร์ 2 ของ Wi-Fi ฝ่ายรับ

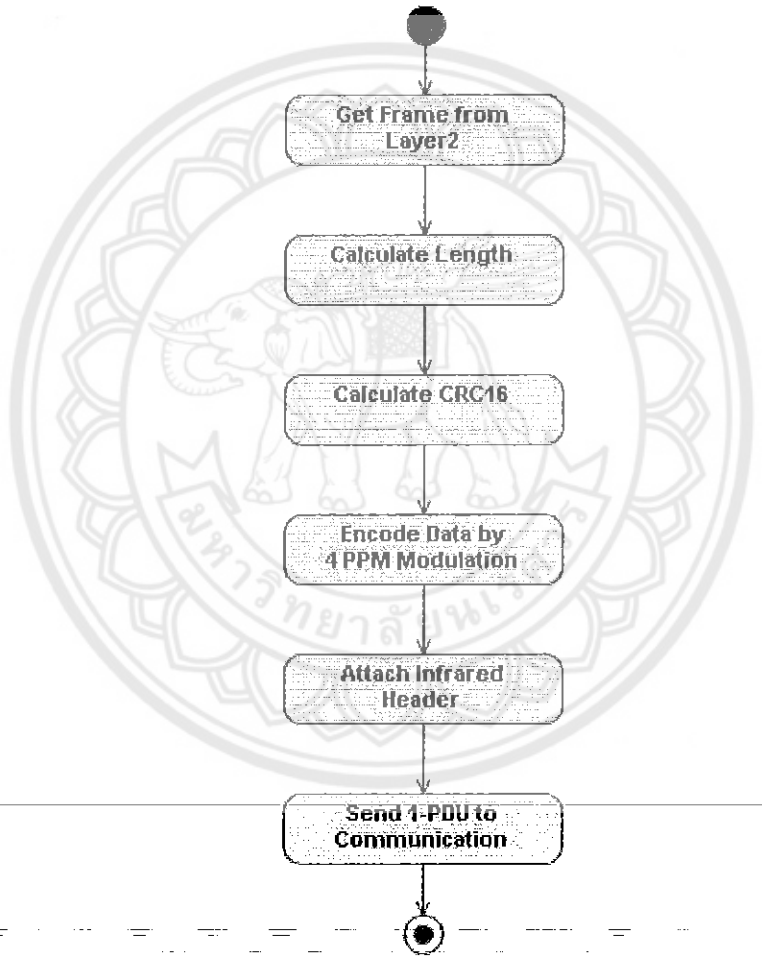
1. เลเยอร์ 2 รับข้อมูลมาจากเลเยอร์ 1
2. เลเยอร์ 2 นำส่วนเฮดเดอร์ต่างๆออก
3. เลเยอร์ 2 ตรวจสอบ CRC-32
4. เลเยอร์ 2 ส่งเฟรมตอบรับกลับไป
5. เลเยอร์ 2 รับข้อมูลที่แบ่งไว้จนครบ
6. เลเยอร์ 2 ทำการรวมข้อมูล แล้วส่งไปยัง LLC ซับเลเยอร์
7. เลเยอร์ 2 ถอด LLC เฮดเดอร์
8. เลเยอร์ 2 ส่งข้อมูลไปยังเลเยอร์ 3 ต่อไป



รูปที่ 3.16 Activity Diagram ของเลเยอร์ 2 Wi-Fi ที่เป็นผู้รับข้อความ

การไหลของเหตุการณ์ (Flow of Events) ของเลเยอร์ 1 ของ Wi-Fi ฝ่ายส่ง

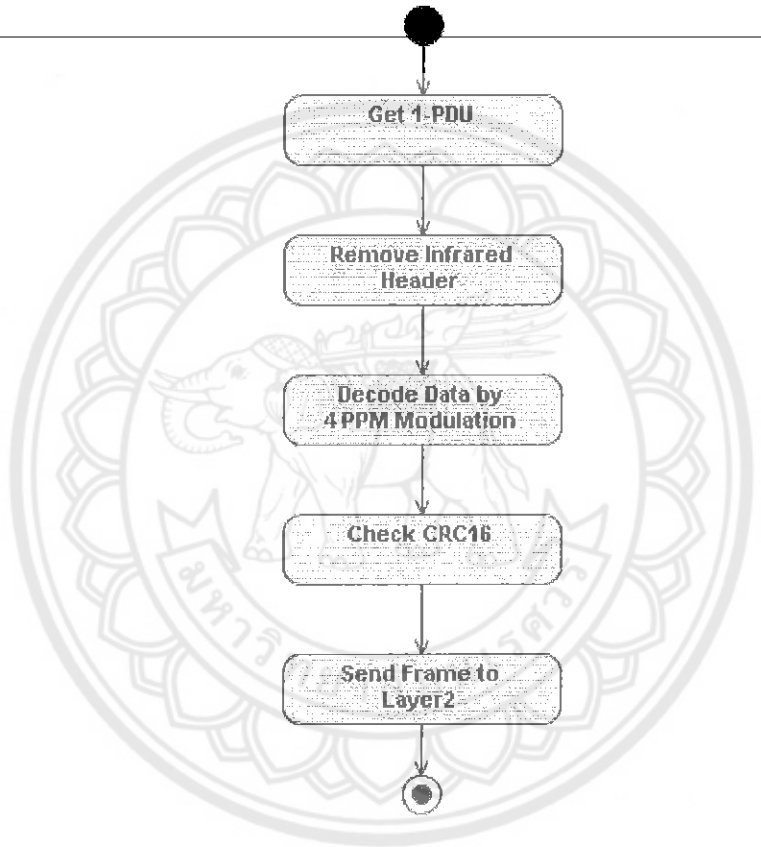
1. เลเยอร์ 1 รับข้อมูลจากเลเยอร์ 2
2. เลเยอร์ 1 คำนวณขนาดของข้อมูล
3. เลเยอร์ 1 คำนวณค่า CRC-16 ของข้อมูล
4. เลเยอร์ 1 เข้ารหัสข้อมูลแบบ 4-PPM
5. เลเยอร์ 1 ใส่เฮดเดอร์
6. เลเยอร์ 1 ส่งข้อมูลเพื่อการสื่อสารต่อไป



รูปที่ 3.17 Activity Diagram ของเลเยอร์ 1 Wi-Fi ที่เป็นผู้ส่งข้อความ

การไหลของเหตุการณ์ (Flow of Events) ของเลเยอร์ 1 ของ Wi-Fi ฝ่ายรับ

1. เลเยอร์ 1 รับข้อมูลมาจากการสื่อสาร
2. เลเยอร์ 1 ถอดเฮดเดอร์ออก
3. เลเยอร์ 1 ถอดรหัสข้อมูล
4. เลเยอร์ 1 ตรวจสอบ CRC-16
5. เลเยอร์ 1 ส่งข้อมูลไปยังเลเยอร์ 2 ต่อไป



รูปที่ 3.18 Activity Diagram ของเลเยอร์ 1 Wi-Fi ที่เป็นผู้รับข้อความ

บทที่ 4

ผลการทดลอง

4.1 แผนการทดลอง

ตารางที่ 4.1 ตารางแผนการทดลอง

ลำดับ	รายการ
1	**การทดสอบคลาส** • ทดสอบการส่งที่ Layer 2 Ethernet • ทดสอบการรับที่ Layer 2 Ethernet • ทดสอบการส่งที่ Layer 1 CAT5 • ทดสอบการรับที่ Layer 1 CAT5 • ทดสอบการส่งที่ Layer 2 Wi-Fi • ทดสอบการรับที่ Layer 2 Wi-Fi • ทดสอบการส่งที่ Layer 1 Infrared • ทดสอบการรับที่ Layer 1 Infrared
2	**การทดสอบการส่งข้อมูลตั้งแต่ต้นจนจบ** • ทดสอบการส่งข้อมูลของ Ethernet Suite • ทดสอบการส่งข้อมูลของ Wi-Fi Suite • ทดสอบการส่งข้อมูลของ Ethernet on Wi-Fi
3	**การทดสอบการใช้งานโปรแกรมบนระบบปฏิบัติการ** • ทดสอบบนระบบปฏิบัติการ Windows XP • ทดสอบบนระบบปฏิบัติการ Linux Ubuntu • ทดสอบระหว่าง Windows XP และ Linux Ubuntu
4	**การทดสอบหน่วยย่อย** • ทดสอบ CRC-32 • ทดสอบ CRC-16 • ทดสอบการเข้ารหัสและถอดรหัสแบบ 4-PPM • ทดสอบการเข้ารหัสและถอดรหัสแบบ 4B/5B

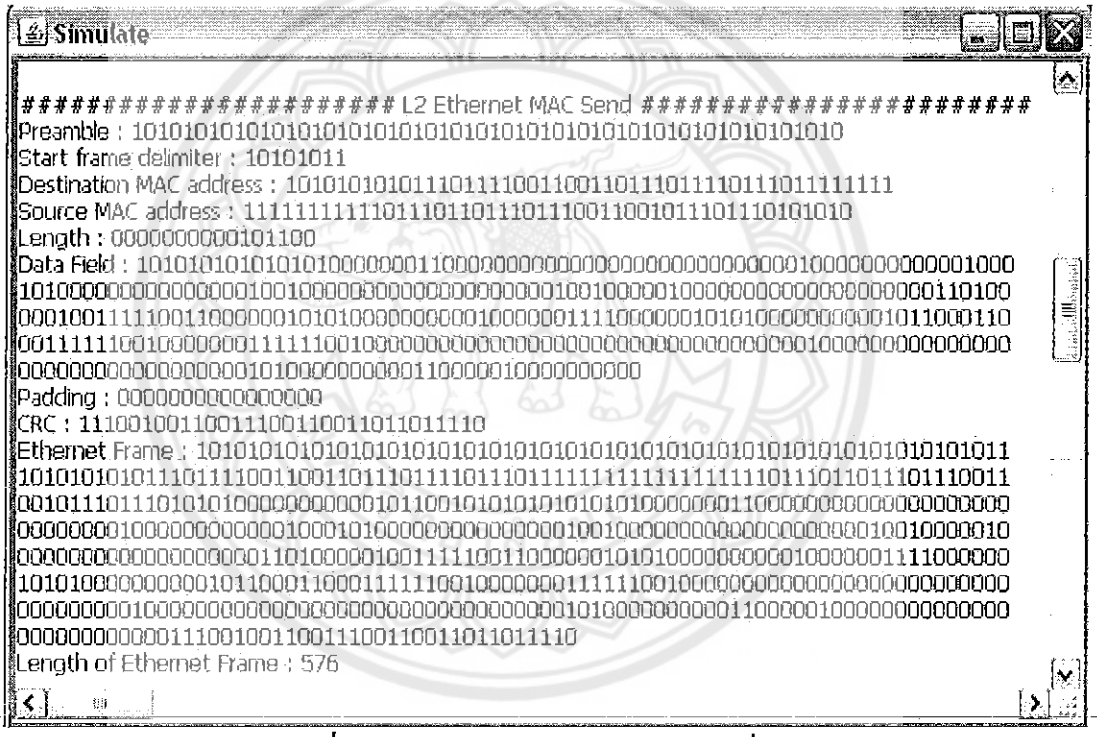
ตารางที่ 4.1(ต่อ) ตารางแผนการทดลอง

	• ทดสอบการแพดข้อมูล • ทดสอบการรับค่าแมคแอดเดรส • ทดสอบการคำนวณขนาดของข้อมูล

4.2 การทดสอบคลาส

4.2.1 การทดสอบการส่งที่เลเยอร์ 2 ของ Ethernet

ทำการทดลองคลาสตัวอย่างการจำลอง Ethernet Protocol ที่ฝั่งผู้ส่งโดยใช้ข้อความว่า "Hello" จะได้รายละเอียดของเฟรมซึ่งแสดงผลแบบ GUI เป็นดังนี้



รูปที่ 4.1 รายละเอียดของ Ethernet frame ฝั่งส่ง

คำอธิบาย

- Preamble มีทั้งหมด 7 ไบท์ โดยแต่ละไบท์เป็น 10101010
- Start frame delimiter คือ 10101011
- Destination MAC address รับค่าจากไฟล์คือ AA-BB-CC-DD-EE-FF
- Source MAC address รับค่าจากไฟล์คือ FF-EE-DD-CC-BB-AA
- Length ขนาดของข้อมูลที่รับมาจากชั้นสื่อสารย่อย LLC คือ 44 ไบท์ แปลงเป็นฐานสองได้ 101100

- Data Field ข้อมูลที่รับมาจากชั้นสื่อสารย่อย LLC ทีทั้งหมด 44 ไบท์
- Padding มีทั้งสิ้น 2 ไบท์เป็น '0' ทั้งหมดเนื่องจากกำหนดขนาดของข้อมูลที่น้อยที่สุดที่ส่งไปเป็น 46 ไบท์
- CRC คำนวณตั้งแต่ Destination MAC address จนถึง Padding โดยใช้อัลกอริทึม CRC-32
- Ethernet Frame รวมทุกส่วนเข้าด้วยกันตั้งแต่ Preamble จนถึง CRC
- Length of Ethernet Frame ขนาดของเฟรม = 7 + 1 + 6 + 6 + 2 + 44 + 2 + 4 = 72 ไบท์ซึ่งเท่ากับ 576 บิตตามรูปที่ 4.1 การคำนวณขนาดของเฟรมดูได้จากรูปที่ 2.4

4.2.2 การทดสอบการรับที่เลเยอร์ 2 ของ Ethernet

ทำการทดลองคลาสตัวอย่างการจำลอง Ethernet Protocol ที่ฝั่งผู้รับโดยใช้ข้อความว่า "Hello" จะได้รายละเอียดของเฟรมซึ่งแสดงผลแบบ GUI เป็นดังนี้

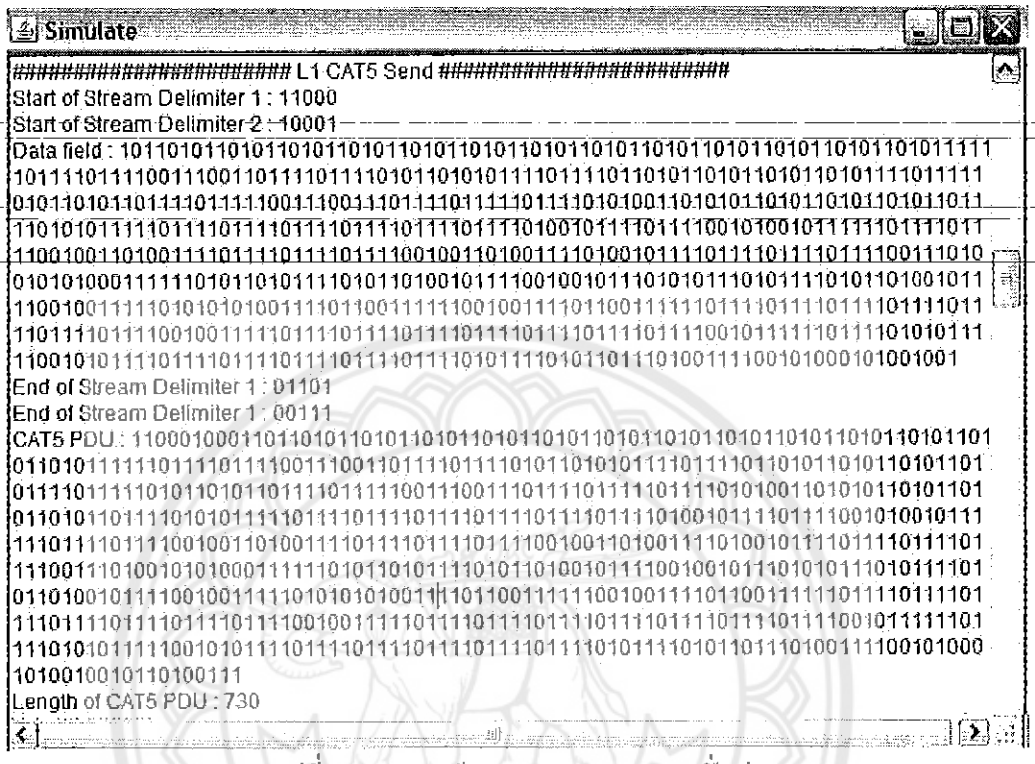
รูปที่ 4.2 รายละเอียดของ Ethernet frame ฝั่งรับ

คำอธิบาย

- เหมือนกับ Ethernet Frame ของฝั่งส่ง ส่วน Data send to L3 คือข้อมูลที่ทำการถอดเฮดเดอร์และเทรลเลอร์ออกหมดแล้วก่อนส่งขึ้นไปที่ชั้นสื่อสารย่อย LLC

4.2.3 การทดสอบการส่งที่เลเยอร์ 1 ของ Ethernet

ทำการทดลองคลาสตัวอย่างการจำลอง CAT5 Protocol ที่ฝั่งผู้ส่งโดยใช้ข้อความว่า “Hello” จะได้รายละเอียดของเฟรมซึ่งแสดงผลแบบ GUI เป็นดังนี้



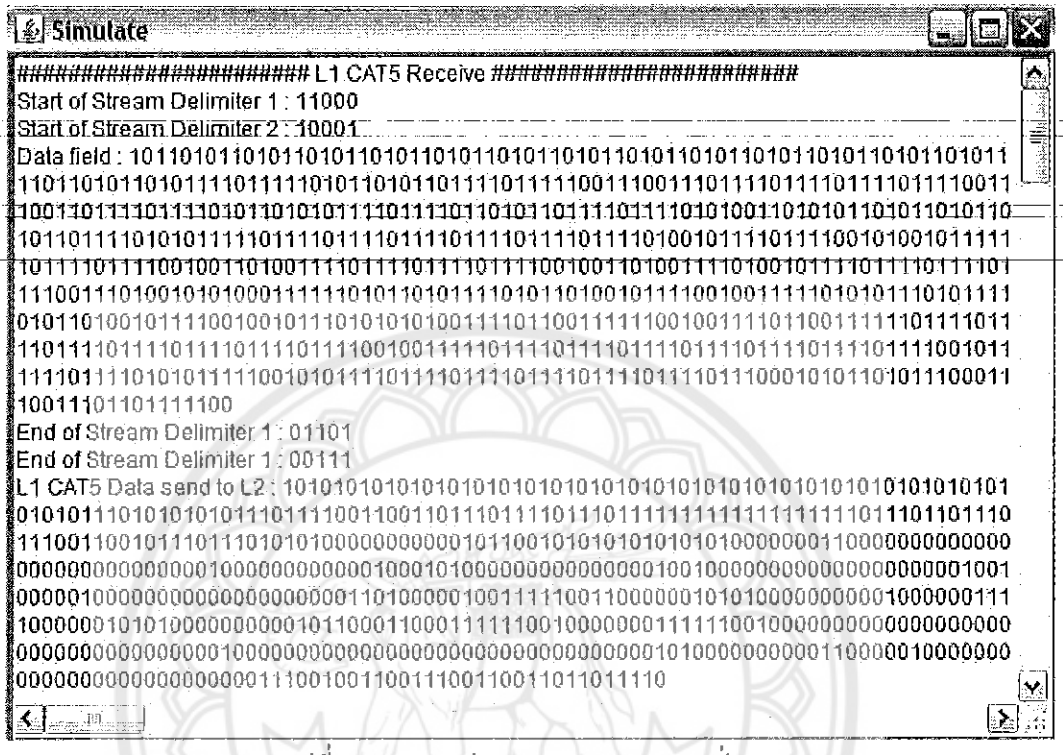
รูปที่ 4.3 รายละเอียดของ CAT5 frame ฝั่งส่ง

คำอธิบาย

- Start of Stream Delimiter 1 คือ 11000
- Start of Stream Delimiter 2 คือ 10001
- Data Field ข้อมูลที่ทำการเข้ารหัสแบบ 4B/5B แล้ว
- End of Stream Delimiter 1 คือ 01101
- End of Stream Delimiter 2 คือ 00111
- CAT5 PDU คือ รวมทุกส่วนเข้าด้วยกันตั้งแต่ Start of Stream Delimiter 1 ถึง End of Stream Delimiter 2
- Length of CAT5 PDU ซึ่งเท่ากับ $5 + 5 + ((5 / 4) * (576 - 8)) + 5 + 5 = 730$ บิต การคำนวณขนาดของข้อมูลดูได้จากรูปที่ 2.6

4.2.4 การทดสอบการรับที่เลเยอร์ 1 ของ Ethernet

ทำการทดลองคลาสตัวอย่างการจำลอง CAT5 Protocol ที่ฝั่งผู้รับโดยใช้ข้อความว่า "Hello" จะได้รายละเอียดของเฟรมซึ่งแสดงผลแบบ GUI เป็นดังนี้



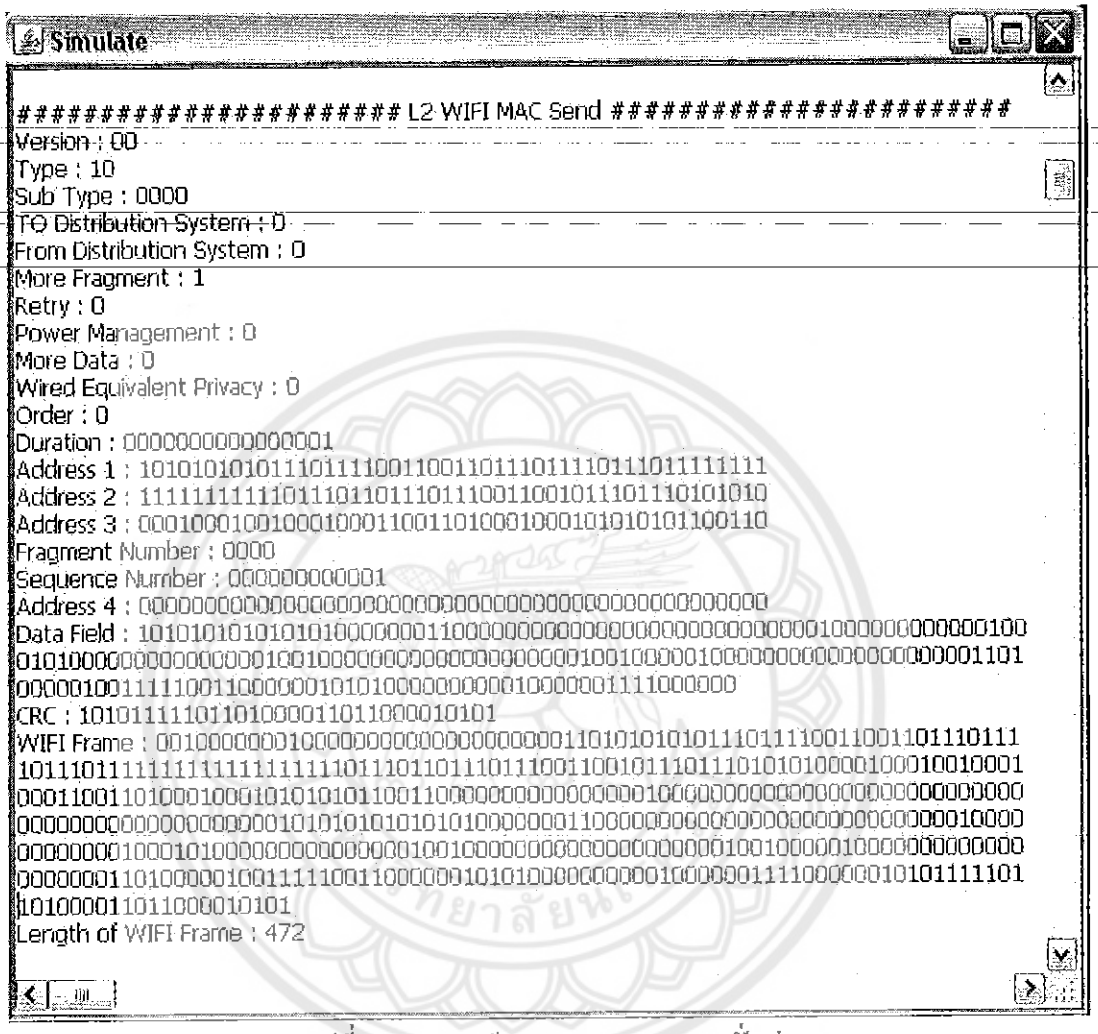
รูปที่ 4.4 รายละเอียดของ CAT5 frame ฝั่งรับ

คำอธิบาย

- เหมือนกับ CAT5 Frame ของฝั่งส่ง ส่วน L1 CAT5 Data send to L2 คือข้อมูลที่ทำการถอดเฮดเดอร์ และเทรลเลอร์ออกหมดแล้วเพิ่มไบท์แรกของ Preamble ก่อนส่งขึ้นไปที่ชั้น Data link Layer

4.2.5 การทดสอบการส่งที่เลเยอร์ 2 ของ Wi-Fi

ทำการทดลองคลาสตัวอย่างการจำลอง Wi-Fi Protocol ที่ฝั่งผู้ส่งโดยใช้ข้อความว่า "Hello" จะได้รายละเอียดของเฟรมซึ่งแสดงผลแบบ GUI เป็นดังนี้



รูปที่ 4.5 รายละเอียดของ Wi-Fi frame ฝั่งส่ง

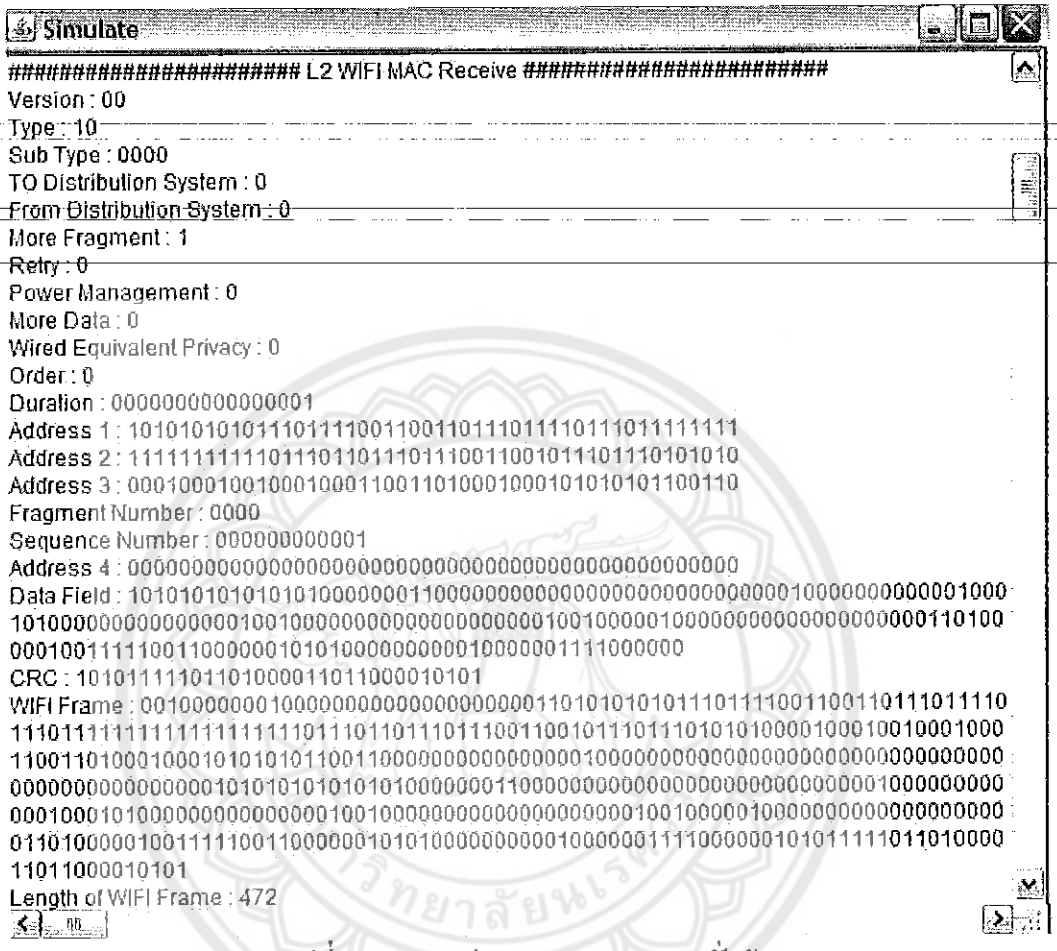
คำอธิบาย

- Version กำหนดให้เป็น 00
- Type ตามตารางที่ 2.2 เป็น 10 เพราะเป็นเฟรมข้อมูล
- Sub-type ตามตารางที่ 2.2 เป็น 0000
- To Distribution System ตามตารางที่ 2.3 เพราะเป็น Ad-Hoc Network
- From Distribution System ตามตารางที่ 2.3 เพราะเป็น Ad-Hoc Network
- More Fragment เป็น 1 เพราะไม่ได้เป็น Fragment สุดท้าย
- Retry ถึง Order เป็น 0 ทั้งหมด

- Duration กำหนดการส่งข้อมูลที่ความเร็ว 1 microsecond
- Address 1 ตามตารางที่ 2.3 เป็น Destination address รับค่าจากไฟล์คือ AA-BB-CC-DD-EE-FF
- Address 2 ตามตารางที่ 2.3 เป็น Source address รับค่าจากไฟล์คือ FF-EE-DD-CC-BB-AA
- Address 3 ตามตารางที่ 2.3 เป็น BSS-Id รับค่าจากไฟล์คือ 11-22-33-44-55-66
- Address 4 ตามตารางที่ 2.3 ไม่มีการใช้งานกำหนดให้เป็น 0 ทั้งหมด 6 ไบท์
- Fragment Number เป็น fragment แรกกำหนดให้เริ่มที่ 0
- Sequence Number เป็น ข้อมูลชุดแรกที่รับมาจากชั้นสื่อสารย่อย LLC
- Data Field ข้อมูลที่ทำการแบ่งแล้ว (Fragmentation) ซึ่งกำหนดค่าเทรสโชลของการแบ่งอยู่ที่ 25 ไบท์ หรือ 200 บิต
- CRC คำนวณตั้งแต่ Version ถึง Data Field โดยใช้อัลกอริทึม CRC-32
- Wi-Fi Frame รวมทุกส่วนเข้าด้วยกันตั้งแต่ Version จนถึง CRC
- Length of Wi-Fi Frame ขนาดของเฟรม 2 + 2 + 6 + 6 + 6 + 2 + 6 + 25 + 4 = 59 ไบท์ ซึ่งเท่ากับ 472 บิตตามรูปที่ 4.5 การคำนวณขนาดของเฟรมดูได้จากรูป 2.7(b)

4.2.6 การทดสอบการรับที่เลเยอร์ 2 ของ Wi-Fi

ทำการทดลองคลาสตัวอย่างการจำลอง Wi-Fi Protocol ที่ฝั่งผู้รับโดยใช้ข้อความว่า "Hello" จะได้รายละเอียดของเฟรมซึ่งแสดงผลแบบ GUI เป็นดังนี้



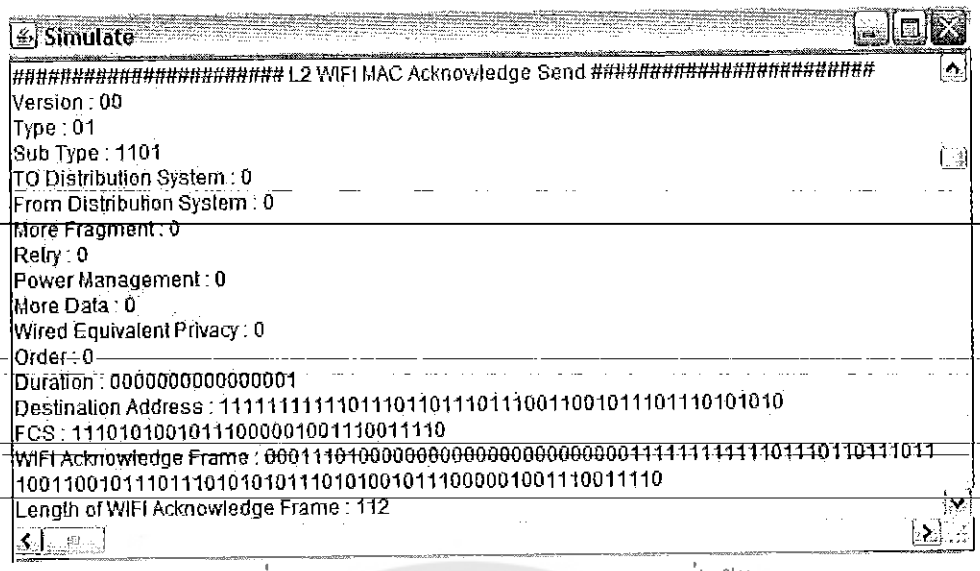
รูปที่ 4.6 รายละเอียดของ Wi-Fi frame ฝั่งรับ

คำอธิบาย

- เหมือนกับ Wi-Fi ของฝั่งส่งทุกประการ

4.2.7 การทดสอบการส่งเฟรมตอบรับที่เลเยอร์ 2 ของ Wi-Fi

ทำการทดลองตัวอย่างการจำลองการส่งเฟรมตอบรับ ที่ฝั่งผู้รับโดยใช้ข้อความว่า "Hello" จะได้รายละเอียดของเฟรมซึ่งแสดงผลแบบ GUI เป็นดังนี้



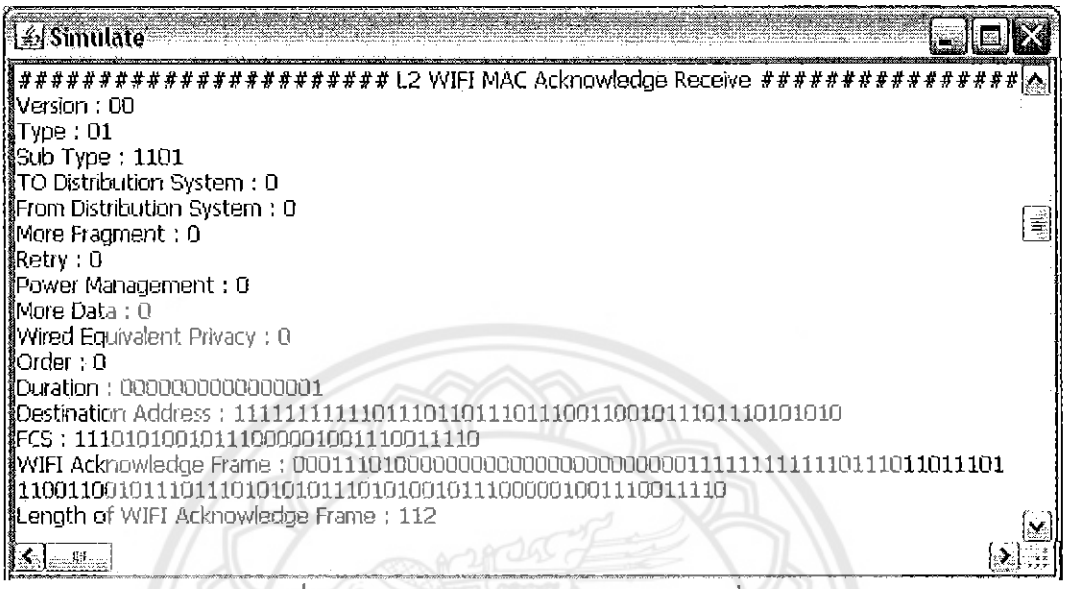
รูปที่ 4.7 รายละเอียดของเฟรมตอบรับของฝั่งผู้รับ

คำอธิบาย

- Version กำหนดให้เป็น 00
- Type ตามตารางที่ 2.2 เป็น 10 เพราะเป็นส่วนคอนโทรล
- Sub-type ตามตารางที่ 2.2 เป็น 1101 เพราะเป็นส่วนเฟรมตอบรับ
- To Distribution System ตามตารางที่ 2.3 เพราะเป็น Ad-Hoc Network
- From Distribution System ตามตารางที่ 2.3 เพราะเป็น Ad-Hoc Network
- More Fragment เป็น 0 เพราะมี Fragment เดียว
- Retry ถึง Order เป็น 0 ทั้งหมด
- Duration กำหนดการส่งข้อมูลที่ความเร็ว 1 microsecond
- Destination address เป็น Source address ของฝั่งส่งคือ FF-EE-DD-CC-BB-AA
- FCS คำนวณตั้งแต่ Version ถึง Destination address โดยใช้อัลกอริทึม CRC-32
- Wi-Fi Acknowledge Frame รวมทุกส่วนเข้าด้วยกันตั้งแต่ Version จนถึง FCS
- Length of Wi-Fi Acknowledge Frame ขนาดของเฟรม 2 + 2 + 6 + 4 = 14 ไบท์ซึ่งเท่ากับ 112 บิต ตามรูปที่ 4.7 การคำนวณขนาดของเฟรมดูได้จากรูปที่ 2.7(a)

4.2.8 การทดสอบการรับเฟรมตอบรับที่เลเยอร์ 2 ของ Wi-Fi

ทำการทดลองตัวอย่างการจำลองการรับเฟรมตอบรับ ที่ฝั่งผู้ส่งโดยใช้ข้อความว่า "Hello" จะได้รายละเอียดของเฟรมซึ่งแสดงผลแบบ GUI เป็นดังนี้



รูปที่ 4.8 รายละเอียดของเฟรมตอบรับของฝั่งผู้ส่ง

คำอธิบาย

- เหมือนกับเฟรมตอบรับของฝั่งผู้รับทุกประการ

4.2.9 การทดสอบการส่งที่เลเยอร์ 1 ของ Wi-Fi

ทำการทดลองคลาสตัวอย่างการจำลอง Infrared Protocol ที่ฝั่งผู้ส่งโดยใช้ข้อความว่า "Hello" จะได้รายละเอียดของเฟรมซึ่งแสดงผลแบบ GUI เป็นดังนี้

รูปที่ 4.9 รายละเอียดของ Infrared frame ของฝั่งผู้ส่ง

คำอธิบาย

- Synchronization บิต 0,1 สลับกัน 57 บิต
- Start Frame Delimiter กำหนดเป็น 1001
- Data rate ที่ความเร็ว 2 Mbps มีค่าเท่ากับ 001
- Direct Current Level Adjustment ที่ความเร็ว 2 Mbps มีค่าเท่ากับ 00100010001000100010001000100010
- Header Error Check ใช้ CRC-16 กับข้อมูลที่รับมา

- Data เข้ารหัสแบบ 4-PPM กับข้อมูลที่รับมา
- Infrared Frame รวมทุกส่วนเข้าด้วยกันตั้งแต่ Synchronization จนถึง Header Error Check
- Infrared Frame Length ขนาดของเฟรม 57 + 4 + 3 + 32 + 16 + 16 + (472 * 2) = 1072 บิต ตามรูปที่ 4.9 การคำนวณขนาดของเฟรมดูได้จากรูปที่ 2.9

4.2.10 การทดสอบการรับที่เลเยอร์ 1 ของ Wi-Fi

ทำการทดลองคลาสตัวอย่างการจำลอง Infrared Protocol ที่ฝั่งผู้รับโดยใช้ข้อความว่า "Hello" จะได้รายละเอียดของเฟรมซึ่งแสดงผลแบบ GUI เป็นดังนี้

รูปที่ 4.10 รายละเอียดของ Infrared frame ของฝั่งผู้รับ

คำอธิบาย

- เหมือนกับ Infrared frame ของฝั่งผู้ส่งทุกประการ ส่วนของ L1 Infrared Msg send to L2 คือข้อมูลที่ทำการตัดเฮดเดอร์ออก แล้วทำการถอดรหัสก่อนส่งขึ้นไปที่ชั้น Data link Layer

4.3 การทดสอบการส่งข้อมูลตั้งแต่ต้นจนจบ

4.3.1 ทดสอบการส่งข้อมูลของ Ethernet Suite ที่เครื่องฝั่งส่ง

การทดสอบการส่งข้อมูลตั้งแต่ต้นจนจบของ Ethernet Suite โดยทำการส่งข้อความ "Hello" จะได้ลำดับการทำงานดังต่อไปนี้

```
OS : Windows XP
Destination : 192.168.1.3
########################## L4 TCP Send                    ##########################
########################## L3 IPv4 Send                   ##########################
L3 Send First Package
########################## L2 LLC Ethernet Send           ##########################
########################## L2 Ethernet Send               ##########################
########################## L1 CAT5 Send                   ##########################
########################## L1 UTP Send                    ##########################
L3 Send Second Package
########################## L2 LLC Ethernet Send           ##########################
########################## L2 Ethernet Send               ##########################
########################## L1 CAT5 Send                   ##########################
########################## L1 UTP Send                    ##########################

------------------------- L4 TCP Wait Acknowledge  -------------------------

########################## L1 UTP Receive                 ##########################
########################## L1 CAT5 Receive                ##########################
########################## L2 Ethernet Receive            ##########################
########################## L2 LLC Ethernet Receive        ##########################
########################## L3 IPv4 Receive                ##########################
L3 Checksum successful!!!
L3 Receive First Package
########################## L1 UTP Receive                 ##########################
########################## L1 CAT5 Receive                ##########################
########################## L2 Ethernet Receive            ##########################
########################## L2 LLC Ethernet Receive        ##########################
########################## L3 IPv4 Receive                ##########################
L3 Checksum successful!!!
L3 Receive Second Package
########################## L4 TCP Receive                 ##########################
L4 Checksum successful !!!
##########################################################################################
########################## L4 Receive Acknowledge         ##########################
##########################################################################################
------------------------------------------------------------------------------
------------------------- Send Message Completed  -------------------------
------------------------------------------------------------------------------
BUILD SUCCESSFUL (total time: 31 seconds)
```

รูปที่ 4.11 ลำดับการทำงานของ Ethernet Suite ของฝั่งผู้ส่ง

คำอธิบาย

เมื่อทำการรับข้อความเข้ามา เลเยอร์ 4 TCP ติดเฮดเดอร์กับข้อมูลแล้วรอรับการตอบรับ ส่งข้อมูล เลเยอร์ 3 IPv4 แบ่งข้อมูลติดเฮดเดอร์ ส่งแพ็กเก็ตแรก เลเยอร์ 2 LLC ติดเฮดเดอร์กับข้อมูลแล้วส่งต่อ เลเยอร์ 2 Ethernet ติดเฮดเดอร์กับข้อมูลแล้วส่งต่อ เลเยอร์ 1 CAT5 ติดเฮดเดอร์กับข้อมูลแล้วส่งต่อ จากนั้นทำขั้นตอนเดิมเพื่อส่งแพ็กเก็ตที่สอง จากนั้นจะทำการรับการตอบรับที่ส่งมาจากเลเยอร์ 4 ของฝั่งรับ เมื่อการตอบรับมาถึงการส่งข้อความนี้ก็จะสมบูรณ์

4.3.2 ทดสอบการส่งข้อมูลของ Ethernet Suite ที่เครื่องฝั่งรับ

การทดสอบการส่งข้อมูลตั้งแต่ต้นจนจบของ Ethernet Suite โดยทำการส่งข้อความ "Hello" จะได้ลำดับการทำงานดังต่อไปนี้

```
run:
########################## L1 UTP Receive                ##########################
########################## L1 CAT5 Receive               ##########################
########################## L2 Ethernet Receive           ##########################
########################## L2 LLC Ethernet Receive       ##########################
########################## L3 IPv4 Receive               ##########################
L3 Checksum successful!!!
L3 Receive First Package
########################## L1 UTP Receive                ##########################
########################## L1 CAT5 Receive               ##########################
########################## L2 Ethernet Receive           ##########################
########################## L2 LLC Ethernet Receive       ##########################
########################## L3 IPv4 Receive               ##########################
L3 Checksum successful!!!
L3 Receive Second Package
########################## L4 TCP Receive                ##########################
L4 Checksum successful !!!
########################## L4 Send Acknowledge           ##########################
########################## L3 IPv4 Send                  ##########################
L3 Send First Package
########################## L2 LLC Ethernet Send          ##########################
########################## L2 Ethernet Send              ##########################
########################## L1 CAT5 Send                  ##########################
########################## L1 UTP Send                   ##########################
L3_Send Second Package
############################ L2 LLC Ethernet Send        ##########################
############################ L2 Ethernet Send            ##########################
############################ L1 CAT5 Send                ##########################
############################ L1 UTP Send                 ##########################
##############################################################################################
########################## L4 Receive finish             ##########################
##############################################################################################

------------------------------------------------------------------------------
------------------------- Receive All Data : Hello ---------------------------
------------------------------------------------------------------------------
```

รูปที่ 4.12 ลำดับการทำงานของ Ethernet Suite ของฝั่งผู้รับ

คำอธิบาย

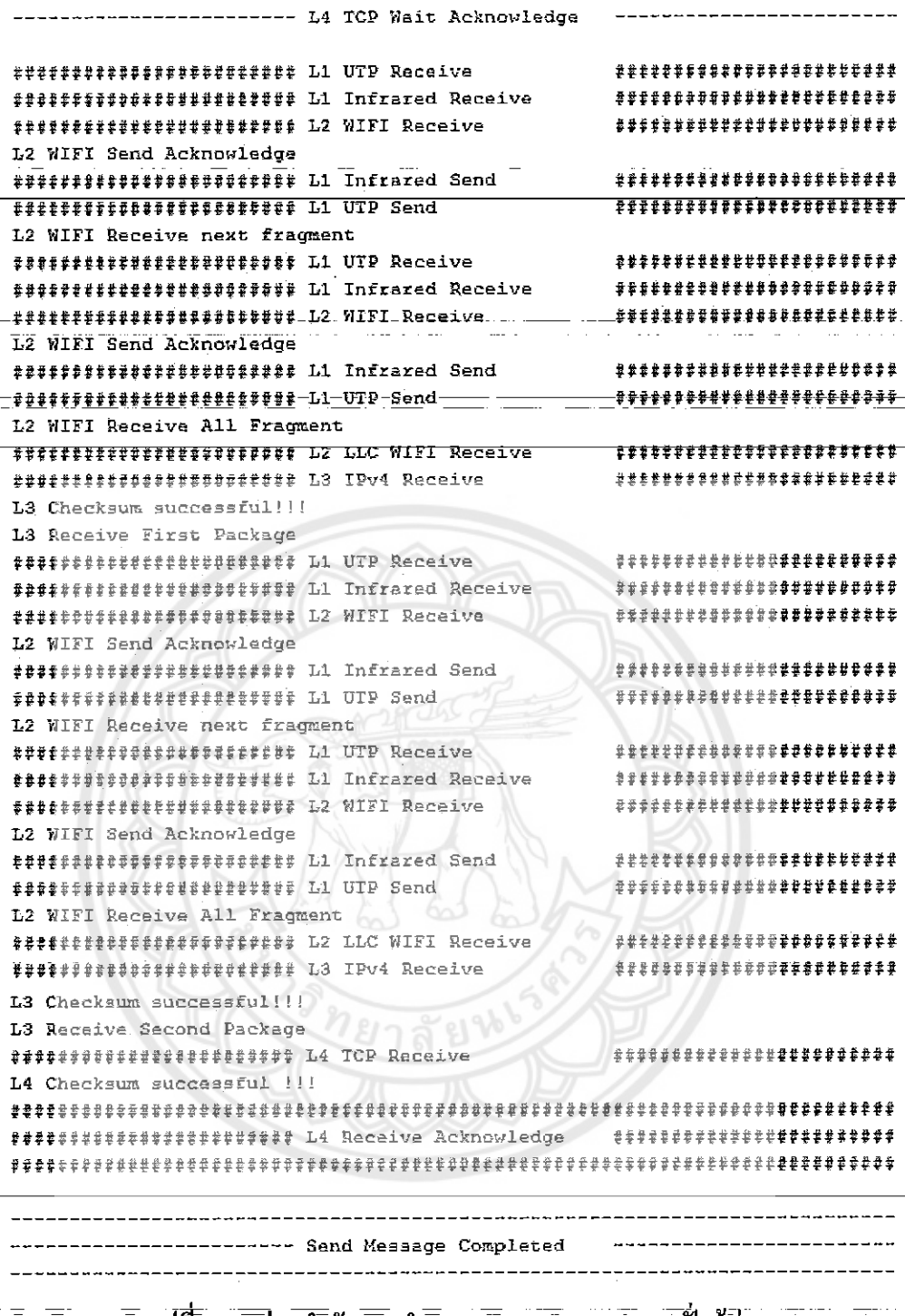
เมื่อฝั่งผู้ส่งได้ส่งข้อความมาถึงฝั่งผู้รับ เลเยอร์ 1 CAT5 ทำการถอดเฮดเดอร์ ส่งข้อมูล เลเยอร์ 2 Ethernet ถอดเฮดเดอร์ ส่งข้อมูล เลเยอร์ 2 LLC ถอดเฮดเดอร์ ส่งข้อมูล เลเยอร์ 3 IPv4 รับแพ็กเก็ตแรก เลเยอร์ 1 CAT5 เริ่มรับแพ็กเก็ตสองทำตามขั้นตอนเดิม ทำการรวมข้อมูล ส่งข้อมูล เลเยอร์ 4 TCP ส่งการตอบรับ ไปยังฝั่งผู้ส่งการรับข้อความนี้ก็จะสมบูรณ์

4.3.3 ทดสอบการส่งข้อมูลของ Wi-Fi Suite ที่เครื่องฝั่งส่ง

การทดสอบการส่งข้อมูลตั้งแต่ต้นจนจบของ Wi-Fi Suite โดยทำการส่งข้อความ "Hello" จะได้ลำดับการทำงานดังต่อไปนี้

```
run:
OS : Windows XP
Destination : 192.168.1.3
########################## L4 TCP Send              ###########################
########################## L3 IPv4 Send             ###########################
L3 Send First Package
########################## L2 LLC WIFI Send         ###########################
########################## L2 WIFI Send             ###########################
L2 WIFI Send fragment 1
########################## L1 Infrared Send         ###########################
########################## L1 UTP Send              ###########################
------------------------- L2 WIFI Wait Acknowledge  -------------------------
########################## L1 UTP Receive           ###########################
########################## L1 Infrared Receive      ###########################
L2 WIFI Receive Acknowledge completed
L2 WIFI Send fragment 2
########################## L1 Infrared Send         ###########################
########################## L1 UTP Send              ###########################
------------------------- L2 WIFI Wait Acknowledge  -------------------------
########################## L1 UTP Receive           ###########################
########################## L1 Infrared Receive      ###########################
L2 WIFI Receive Acknowledge completed
L3 Send Second Package
########################## L2 LLC WIFI Send         ###########################
########################## L2 WIFI Send             ###########################
L2 WIFI Send fragment 1
########################## L1 Infrared Send         ##########################
########################## L1 UTP Send              ##########################
------------------------- L2 WIFI Wait Acknowledge  -------------------------
########################## L1 UTP Receive           ##########################
########################## L1 Infrared Receive      ##########################
L2 WIFI Receive Acknowledge completed
L2 WIFI Send fragment 2
########################## L1 Infrared Send         ##########################
########################## L1 UTP Send              ##########################
------------------------- L2 WIFI Wait Acknowledge  -------------------------
########################## L1 UTP Receive           ##########################
########################## L1 Infrared Receive      ##########################
L2 WIFI Receive Acknowledge completed
```

รูปที่ 4.13 ลำดับการทำงานของ Wi-Fi Suite ของฝั่งผู้ส่ง



รูปที่ 4.13(ต่อ) ลำดับการทำงานของ Wi-Fi Suite ของฝั่งผู้ส่ง

คำอธิบาย

เมื่อทำการรับข้อความเข้ามา เลเยอร์ 4 TCP ติดเฮดเดอร์กับข้อมูลแล้วรอรับการตอบรับ ส่งข้อมูล เลเยอร์ 3 IPv4 แบ่งข้อมูลติดเฮดเดอร์ ส่งแพ็กเก็ตแรก เลเยอร์ 2 LLC ติดเฮดเดอร์กับข้อมูลแล้วส่งต่อ เลเยอร์ 2 Wi-Fi แบ่งข้อมูล ติดเฮดเดอร์กับข้อมูล ส่ง Fragment แรก รอเฟรมตอบรับ เลเยอร์ 1 CAT5 ติดเฮดเดอร์กับข้อมูลแล้วส่งต่อ เมื่อ Fragment แรกไปถึงเลเยอร์ 2 Wi-Fi ของฝั่งผู้รับ

จะทำการส่งเฟรมตอบรับมา เมื่อเฟรมตอบรับมาถึงเลเยอร์ 2 Wi-Fi ของฝั่งผู้ส่งจะทำการส่ง Fragment ต่อไป จากนั้นทำขั้นตอนเดิมเพื่อส่งแพ็กเก็ตที่สอง จากนั้นจะทำการรับการตอบรับที่ส่งมาจากเลเยอร์ 4 ของฝั่งรับ เมื่อการตอบรับมาถึงการส่งข้อความนี้ก็จะสมบูรณ์

4.3.4 ทดสอบการส่งข้อมูลของ Wi-Fi Suite ที่เครื่องฝั่งรับ

การทดสอบการส่งข้อมูลตั้งแต่ต้นจนจบของ Wi-Fi Suite โดยทำการส่งข้อความ "Hello" จะได้ลำดับการทำงานดังต่อไปนี้

```
########################## L1 UTP Receive          ##########################
########################## L1 Infrared Receive     ##########################
########################## L2 WIFI Receive         ##########################
L2 WIFI Send Acknowledge
########################## L1 Infrared Send        ##########################
########################## L1 UTP Send             ##########################
L2 WIFI Receive next fragment
########################## L1 UTP Receive          ##########################
########################## L1 Infrared Receive     ##########################
########################## L2 WIFI Receive         ##########################
L2 WIFI Send Acknowledge
########################## L1 Infrared Send        ##########################
########################## L1 UTP Send             ##########################
L2 WIFI Receive All Fragment
########################## L2 LLC WIFI Receive     ##########################
########################## L3 IPv4 Receive         ##########################
L3 Checksum successful!!!
L3 Receive First Package
########################## L1 UTP Receive          ##########################
########################## L1 Infrared Receive     ##########################
########################## L2 WIFI Receive         ##########################
L2 WIFI Send Acknowledge
########################## L1 Infrared Send        ##########################
########################## L1 UTP Send             ##########################
L2 WIFI Receive next fragment
########################## L1 UTP Receive          ##########################
########################## L1 Infrared Receive     ##########################
########################## L2 WIFI Receive         ##########################
L2 WIFI Send Acknowledge
########################## L1 Infrared Send        ##########################
########################## L1 UTP Send             ##########################
L2 WIFI Receive All Fragment
########################## L2 LLC WIFI Receive     ##########################
########################## L3 IPv4 Receive         ##########################
L3 Checksum successful!!!
L3 Receive Second Package
########################## L4 TCP Receive          ##########################
L4 Checksum successful !!!
```

รูปที่ 4.14 ลำดับการทำงานของ Wi-Fi Suite ของฝั่งผู้รับ

```
########################## L4 Send Acknowledge          ##########################
########################## L3 IPv4 Send                 ##########################
L3 Send First Package
########################## L2 LLC WIFI Send             ##########################
########################## L2 WIFI Send                 ##########################
L2 WIFI Send fragment 1
########################## L1 Infrared Send             ##########################
########################## L1 UTP Send                  ##########################
------------------------- L2 WIFI Wait Acknowledge      -------------------------
########################## L1 UTP Receive               ##########################
########################## L1 Infrared Receive          ##########################
L2 WIFI Receive Acknowledge completed
L2 WIFI Send fragment 2
########################## L1 Infrared Send             ##########################
########################## L1 UTP Send                  ##########################
------------------------- L2 WIFI Wait Acknowledge      -------------------------
########################## L1 UTP Receive               ##########################
########################## L1 Infrared Receive          ##########################
L2 WIFI Receive Acknowledge completed
L3 Send Second Package
########################## L2 LLC WIFI Send             ##########################
########################## L2 WIFI Send                 ##########################
L2 WIFI Send fragment 1
########################## L1 Infrared Send             ##########################
########################## L1 UTP Send                  ##########################
------------------------- L2 WIFI Wait Acknowledge      -------------------------
########################## L1 UTP Receive               ##########################
########################## L1 Infrared Receive          ##########################
L2 WIFI Receive Acknowledge completed
L2 WIFI Send fragment 2
########################## L1 Infrared Send             ##########################
########################## L1 UTP Send                  ##########################
------------------------- L2 WIFI Wait Acknowledge      -------------------------
########################## L1 UTP Receive               ##########################
########################## L1 Infrared Receive          ##########################
L2 WIFI Receive Acknowledge completed
##########################################################################################
########################## L4 Receive finish            ##########################
##########################################################################################
------------------------------------------------------------------------------------------
------------------------- Receive All Data : Hello -------------------------
------------------------------------------------------------------------------------------
```

รูปที่ 4.14(ต่อ) ลำดับการทำงานของ Wi-Fi Suite ของฝั่งผู้รับ

คำอธิบาย

เมื่อฝั่งผู้ส่งได้ส่งข้อความมาถึงฝั่งผู้รับ เลเยอร์ 1 Infrared ทำการถอดเฮดเดอร์ ส่งข้อมูล เลเยอร์ 2 Wi-Fi ถอดเฮดเดอร์ รับ fragment แรก ส่งเฟรมตอบรับ จากนั้นทำตามขั้นตอนเดิม เมื่อได้รับครบทุก Fragment ทำการรวมข้อมูล ส่งข้อมูลที่รวมแล้ว เลเยอร์ 2 LLC ถอดเฮดเดอร์ ส่งข้อมูล เลเยอร์ 3 IPv4 รับแพ็กเก็ตแรก ทำตามขั้นตอนเดิม เมื่อเลเยอร์ 3 IPv4 รับแพ็กเก็ตที่สองแล้ว รวมข้อมูล ส่งข้อมูลที่รวมแล้ว เลเยอร์ 4 TCP ส่งการตอบรับ ไปยังฝั่งผู้ส่งการรับข้อความนี้ก็จะสมบูรณ์

4.3.5 ทดสอบการส่งข้อมูลของ Ethernet on Wi-Fi Suite ที่เครื่องฝั่งส่ง

การทดสอบการส่งข้อมูลตั้งแต่ต้นจนจบของ Ethernet on Wi-Fi Suite โดยทำการส่งข้อความ “Hello” จะได้ลำดับการทำงานดังต่อไปนี้

```
run:
OS : Windows XP
Destination : 192.168.1.3
########################## L4 TCP Send                    ##########################
########################## L3 IPv4 Send                   ##########################
L3 Send First Package
########################## L2 LLC Ethernet Send           ##########################
########################## L2 Ethernet Send               ##########################
########################## L1 Infrared Send               ##########################
#######################  L1 UTP Send                      ##############  ############
L3 Send Second Package
#######################  L2 LLC Ethernet Send             ##############  ############
#######################  L2 Ethernet Send                 ##############  ############
#######################  L1 Infrared Send                 ##############  ############
#######################  L1 UTP Send                      ##############  ############

------------------------ L4 TCP Wait Acknowledge         ------------------------

#######################  L1 UTP Receive                   ##############  ############
#######################  L1 Infrared Receive              ##############  ############
#######################  L2 Ethernet Receive              ##############  ############
#######################  L2 LLC Ethernet Receive          ##############  ############
#######################  L3 IPv4 Receive                  ##############  ############
L3 Checksum successful!!!
L3 Receive First Package
#######################  L1 UTP Receive                   ##############  ############
#######################  L1 Infrared Receive              ##############  ############
#######################  L2 Ethernet Receive              ##############  ############
#######################  L2 LLC Ethernet Receive          ##############  ############
#######################  L3 IPv4 Receive                  ##############  ############
L3 Checksum successful!!!
L3 Receive Second Package
########################## L4 TCP Receive                 ##############  ############
L4 Checksum successful !!!
##################################################################################
########################## L4 Receive Acknowledge         ##########################
##################################################################################

------------------------ Send Message Completed          ------------------------
```

รูปที่ 4.15 ลำดับการทำงานของ Ethernet on Wi-Fi ของฝั่งผู้ส่ง

คำอธิบาย

เหมือน Ethernet Suite ฝั่งผู้ส่งเพียงแต่เลเยอร์ 1 นั้นเป็น Infrared

4.3.6 ทดสอบการส่งข้อมูลของ Ethernet on Wi-Fi Suite ที่เครื่องฝั่งรับ

การทดสอบการส่งข้อมูลตั้งแต่ต้นจนจบของ Ethernet on WI-FI Suite โดยทำการส่งข้อความ "Hello" จะได้ลำดับการทำงานดังต่อไปนี้

```
run:
########################## L1 UTP Receive                ##########################
########################## L1 Infrared Receive           ##########################
########################## L2 Ethernet Receive           ##########################
########################## L2 LLC Ethernet Receive       ##########################
########################## L3 IPv4 Receive               ##########################
L3 Checksum successful!!!
L3 Receive First Package
########################## L1 UTP Receive                ##########################
########################## L1 Infrared Receive           ##########################
########################## L2 Ethernet Receive           ##########################
########################## L2 LLC Ethernet Receive       ##########################
########################## L3 IPv4 Receive               ##########################
L3 Checksum successful!!!
L3 Receive Second Package
########################## L4 TCP Receive                ##########################
L4 Checksum successful !!!
########################## L4 Send Acknowledge           ##########################
########################## L3 IPv4 Send                  ##########################
L3 Send First Package
########################## L2 LLC Ethernet Send          ##########################
########################## L2 Ethernet Send              ##########################
########################## L1 Infrared Send              ##########################
########################## L1 UTP Send                   ##########################
L3 Send Second Package
########################## L2 LLC Ethernet Send          ##########################
########################## L2 Ethernet Send              ##########################
########################## L1 Infrared Send              ##########################
########################## L1 UTP Send                   ##########################
##########################################################################################
########################## L4 Receive finish             ##########################
##########################################################################################

------------------------------------------------------------------------------------------
------------------------- Receive All Data : Hello -------------------------
------------------------------------------------------------------------------------------
```

รูปที่ 4.16 ลำดับการทำงานของ Ethernet on Wi-Fi ของฝั่งผู้รับ

คำอธิบาย

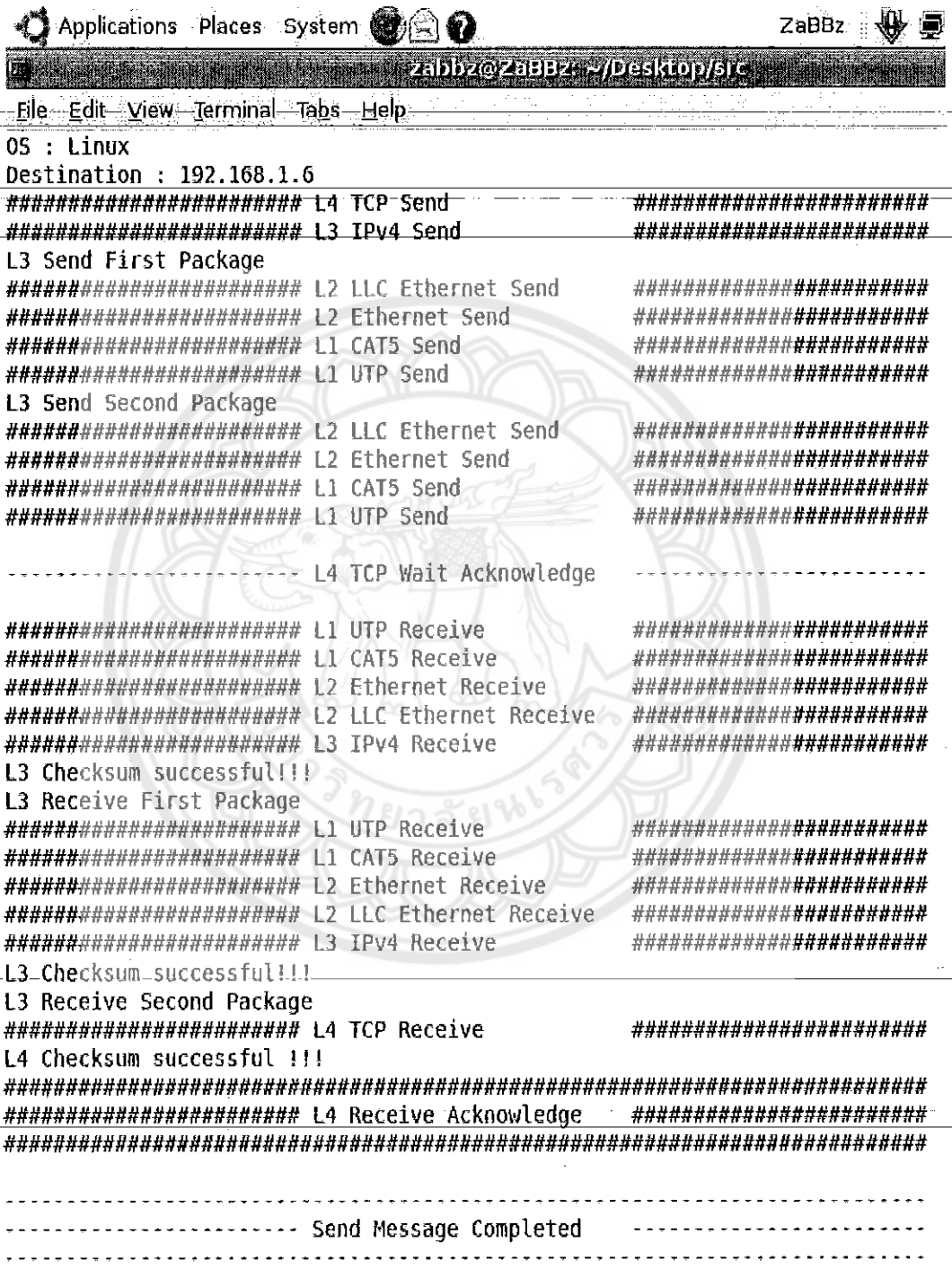
เหมือน Ethernet Suite ฝั่งผู้รับเพียงแต่เลเยอร์ 1 นั้นเป็น Infrared

4.4 การทดสอบการใช้งานโปรแกรมบนระบบปฏิบัติการ

ตารางที่ 4.2 ตารางการทดสอบการใช้งานโปรแกรมบนระบบปฏิบัติการ

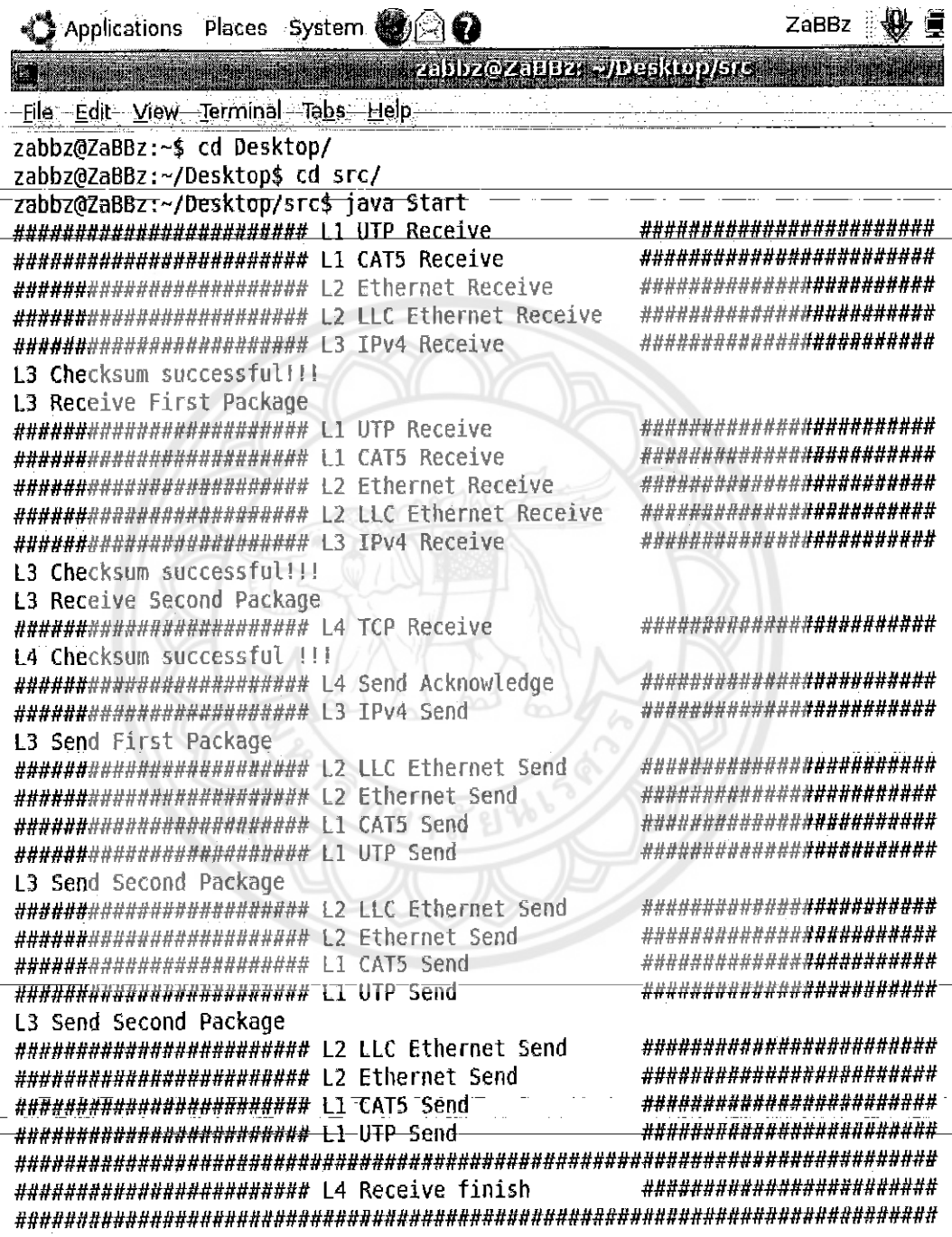
ลำดับ	รายการ	ผลการทดสอบ
1	ทดสอบบนระบบปฏิบัติการ Windows XP ทั้ง 2 เครื่อง	ใช้งานได้
2	ทดสอบบนระบบปฏิบัติการ Linux Ubuntu ทั้ง 2 เครื่อง	ใช้งานได้
3	ทดสอบระหว่างระบบปฏิบัติการ Windows XP และ Linux Ubuntu	ใช้งานได้

ผลการรันบนระบบปฏิบัติการ Linux โดยเลือก Ethernet Suite เป็นตัวอย่าง ทำการทดลองคลาสตัวอย่างการจำลอง Ethernet Protocol ที่ฝั่งผู้ส่งโดยใช้ข้อความว่า "Hello"



รูปที่ 4.17 ผลการรันบนระบบปฏิบัติการ Linux ของ Ethernet ฝั่งส่ง

ผลการรันบนระบบปฏิบัติการ Linux โดยเลือก Ethernet Suite เป็นตัวอย่าง ทำการทดลองคลาสตัวอย่างการจำลอง Ethernet Protocol ที่ฝั่งผู้รับโดยใช้ข้อความว่า "Hello"



รูปที่ 4.18 ผลการรันบนระบบปฏิบัติการ Linux ของ Ethernet ฝั่งรับ

4.5 การทดสอบหน่วยย่อย

4.5.1 ทดสอบ CRC-32

จุดประสงค์ : เพื่อตรวจสอบว่าโปรแกรมสามารถคำนวณค่า CRC-32 ถูกต้องตรงกับค่าที่คำนวณได้และมี 32 บิต

ตารางที่ 4.3 Test case ของ CRC-32

ครั้งที่	Input	Expect output	Actual output	Result
1	1	10011000001000111011011011100000	10011000001000111011011011100000	pass
2	111	11110010001110101000000000101000	11110010001110101000000000101000	pass
3	1010	10111110001010110101101101011000	10111110001010110101101101011000	pass

```
run:
CRC32 of bitstream 1 : 10011000001000111011011011100000
BUILD SUCCESSFUL (total time: 0 seconds)
```

รูปที่ 4.19 ผลที่ได้จากการทดสอบ CRC-32 โดย input คือ 1

```
10000010011000001000111011011011 | 10000 0000 0000 0000 0000 0000 0000 0000
                                  |  10000 0100 1100 0001 0001 1101 1011 0111
                                  |   100 1100 0001 0001 1101 1011 0111
```

รูปที่ 4.20 ค่าที่ได้จากการคำนวณ CRC-32 โดย input คือ 1

```
run:
CRC32 of bitstream 7 : 11110010001110101000000000101000
BUILD SUCCESSFUL (total time: 0 seconds)
```

รูปที่ 4.21 ผลที่ได้จากการทดสอบ CRC-32 โดย input คือ 111

```
10000010011000001000111011011011 | 1010 0000 0000 0000 0000 0000 0000 0000 0000
                                  |  1000 0010 0110 0000 1000 1110 1101 1011 1
                                  |    10 0010 0110 0000 1000 1110 1101 1011 100
                                  |    10 0000 1001 1000 0010 0011 1011 0110 111
                                  |     10 1111 1000 1010 1101 0110 1101 0110
```

รูปที่ 4.22 ค่าที่ได้จากการคำนวณ CRC-32 โดย input คือ 111

```
run:
CRC32 of bitstream 10 : 10111110001010110101101101011000
BUILD SUCCESSFUL (total time: 0 seconds)
```

รูปที่ 4.23 ผลที่ได้จากการทดสอบ CRC-32 โดย input คือ 1010

```
100000100110000010001110110110111 | 1010 0000 0000 0000 0000 0000 0000 0000
                                    1000 0010 0110 0000 1000 1110 1101 1011 1
                                    10 0010 0110 0000 1000 1110 1101 1011 100
                                    10 0000 1001 1000 0010 0011 1011 0110 111
                                    10 1111 1000 1010 1101 0110 1101 0110
```

รูปที่ 4.24 ค่าที่ได้จากการคำนวณ CRC-32 โดย input คือ 1010

4.5.2 ทดสอบ CRC-16

จุดประสงค์ : เพื่อตรวจสอบว่าโปรแกรมสามารถคำนวณค่า CRC-16 ถูกต้องตรงกับค่าที่คำนวณได้และมี 16 บิต

ตารางที่ 4.4 Test case ของ CRC-16

ครั้งที่	Input	Expect output	Actual output	Result
1	1	1000000000000101	1000000000000101	pass
2	111	1000000000010001	1000000000010001	pass
3	1010	1111000000000000	1111000000000000	pass

```
run:
CRC16 of bitstream 1 : 1000000000000101
BUILD SUCCESSFUL (total time: 0 seconds)
```

รูปที่ 4.25 ผลที่ได้จากการทดสอบ CRC-16 โดย input คือ 1

```
1100000000000101 | 1 0000 0000 0000 0000
                   1 1000 0000 0000 0101
                   1000 0000 0000 0101
```

รูปที่ 4.26 ค่าที่ได้จากการคำนวณ CRC-16 โดย input คือ 1

```
run:
CRC16 of bitstream 7 : 1000000000010001
BUILD SUCCESSFUL (total time: 0 seconds)
```

รูปที่ 4.27 ผลที่ได้จากการทดสอบ CRC-16 โดย input คือ 111

```
1100000000000000101 |111 0000 0000 0000 0000
                     110 0000 0000 0001 01
                       1 0000 0000 0001 0100
                       1 1000 0000 0000 0101
                         |1000 0000 0001 0001|
```

รูปที่ 4.28 ค่าที่ได้จากการคำนวณ CRC-16 โดย input คือ 111

```
run:
CRC16 of bitstream 10 : 1111000000000000
BUILD SUCCESSFUL (total time: 0 seconds)
```

รูปที่ 4.29 ผลที่ได้จากการทดสอบ CRC-16 โดย input คือ 1010

```
1100000000000000101 |1010 0000 0000 0000 0000
                     1100 0000 0000 0010 1
                      110 0000 0000 0010 10
                      110 0000 0000 0001 01
                           |11 1100|
```

รูปที่ 4.30 ค่าที่ได้จากการคำนวณ CRC-16 โดย input คือ 1010

4.5.3 ทดสอบการเข้ารหัสและถอดรหัสแบบ 4B/5B

จุดประสงค์ : เพื่อตรวจสอบว่าโปรแกรมสามารถเข้ารหัสได้อย่างถูกต้อง

ตารางที่ 4.5 Test case ของการเข้ารหัสแบบ 4B/5B

ครั้งที่	Input	Expect output	Actual output	Result
1	1010100011000000	10011010010110101111 0	10011010010110101111 0	pass
2	1010100011100	10110100101101 0	10110100101101 0	pass
3	10101000	10110100 10	10110100 10	pass

จุดประสงค์ : เพื่อตรวจสอบว่าโปรแกรมสามารถถอดรหัสได้อย่างถูกต้อง

ตารางที่ 4.6 Test case ของการถอดรหัสแบบ 4B/5B

ครั้งที่	Input	Expect output	Actual output	Result
1	10011010010110101011110	1010100011000000	1010100011000000	pass
2	101101001011010	101010001100	101010001100	pass
3	1011010010	10101000	10101000	pass

```
4B/5B Encoding
Msg : 1010100011000000
Msg after Encoding : 10110100101101011110

4B/5B Decoding
Msg : 10110100101101011110
Msg after Decoding : 1010100011000000
BUILD SUCCESSFUL (total time: 1 second)
```

รูปที่ 4.31 ผลที่ได้จากการทดสอบการเข้าและถอดรหัสแบบ 4B/5B โดย input คือ

1010100011000000

```
4B/5B Encoding
Msg : 101010001100
Msg after Encoding : 101101001011010

4B/5B Decoding
Msg : 101101001011010
Msg after Decoding : 101010001100
BUILD SUCCESSFUL (total time: 1 second)
```

รูปที่ 4.32 ผลที่ได้จากการทดสอบการเข้าและถอดรหัสแบบ 4B/5B โดย input คือ

101101001011010

```
4B/5B Encoding
Msg : 10101000
Msg after Encoding : 1011010010

4B/5B Decoding
Msg : 1011010010
Msg after Decoding : 10101000
BUILD SUCCESSFUL (total time: 1 second)
```

รูปที่ 4.33 ผลที่ได้จากการทดสอบการเข้าและถอดรหัสแบบ 4B/5B โดย input คือ

101101001011010

4.5.4 ทดสอบการเข้ารหัสและถอดรหัสแบบ 4-PPM

จุดประสงค์ : เพื่อตรวจสอบว่าโปรแกรมสามารถเข้ารหัสได้อย่างถูกต้อง

ตารางที่ 4.7 Test case ของการเข้ารหัสแบบ 4-PPM

ครั้งที่	Input	Expect output	Actual output	Result
1	10101000	1000100010000001	1000100010000001	pass
2	101010001010	100010001000000110001000	100010001000000110001000	pass
3	1100	01000001	01000001	pass

จุดประสงค์ : เพื่อตรวจสอบว่าโปรแกรมสามารถถอดรหัสได้อย่างถูกต้อง

ตารางที่ 4.8 Test case ของการถอดรหัสแบบ 4-PPM

ครั้งที่	Input	Expect output	Actual output	Result
1	1000100010000001	10101000	10101000	pass
2	100010001000000110001000	101010001010	101010001010	pass
3	01000001	1100	1100	pass

```
4PPM Encoding
Msg : 10101000
Msg after Encoding : 1000100010000001

4PPM Decoding
Msg : 1000100010000001
Msg after Decoding : 10101000
BUILD SUCCESSFUL (total time: 1 second)
```

รูปที่ 4.34 ผลที่ได้จากการทดสอบการเข้าและถอดรหัสแบบ 4-PPM โดย input คือ 1000100010000001

```
4PPM Encoding
Msg : 101010001010
Msg after Encoding : 100010001000000110001000

4PPM Decoding
Msg : 100010001000000110001000
Msg after Decoding : 101010001010
BUILD SUCCESSFUL (total time: 1 second)
```

รูปที่ 4.35 ผลที่ได้จากการทดสอบการเข้าและถอดรหัสแบบ 4-PPMโดย input คือ 100010001000000110001000

```
4PPM Encoding
Msg : 1100
Msg after Encoding : 01000001

4PPM Decoding
Msg : 01000001
Msg after Decoding : 1100
BUILD SUCCESSFUL (total time: 1 second)
```

รูปที่ 4.36 ผลที่ได้จากการทดสอบการเข้าและถอดรหัสแบบ 4-PPMโดย input คือ 01000001

4.5.5 ทดสอบการแพดข้อมูล

จุดประสงค์ : เพื่อตรวจสอบว่าโปรแกรมสามารถแพดเมื่อข้อมูลน้อยกว่าที่กำหนดไว้ได้อย่างถูกต้อง เพื่อให้ขนาดของเฟรมเป็นไปตามมาตรฐาน

คำอธิบาย : ในการทดสอบกำหนดข้อมูลที่น้อยที่สุดเป็น 1 ไบท์แต่ในโปรแกรมกำหนดไว้ที่ 46 ไบท์ตามมาตรฐานของ IEEE 802.3

ตารางที่ 4.9 Test case ของการแพดข้อมูล

ครั้งที่	Input	Expect output	Actual output	Result
1	1010	0000	0000	pass
2	10	000000	000000	pass
3	10100000	ไม่แพด	ไม่แพด	pass
4	1010000000	ไม่แพด	ไม่แพด	pass

```
If msg less than 8 bits then padding
MSG : 1010
Padding : 0000
```

รูปที่ 4.37 ผลที่ได้จากการทดสอบการแพดข้อมูลโดย input คือ 1010

```
If msg less than 8 bits then padding
MSG : 10
Padding : 000000
```

รูปที่ 4.38 ผลที่ได้จากการทดสอบการแพดข้อมูลโดย input คือ 10

```
If msg less than 8 bits then padding
MSG : 10100000
Padding :
```

รูปที่ 4.39 ผลที่ได้จากการทดสอบการแพดข้อมูลโดย input คือ 10100000

```
If msg less than 8 bits then padding
MSG : 1010000000
Padding :
```

รูปที่ 4.40 ผลที่ได้จากการทดสอบการแพดข้อมูลโดย input คือ 1010000000

4.5.6 ทดสอบการรับค่าแมคแอดเดรส

จุดประสงค์ : เพื่อตรวจสอบว่าโปรแกรมสามารถแปลงค่าแมคแอดเดรสจากฐานสิบหกเป็นฐานสองได้อย่างถูกต้องและมี 48 บิต

ตารางที่ 4.10 Test case ของการรับค่าแมคแอดเดรส

ครั้งที่	Input	Expect output	Actual output	Result
1	AA-BB-CC- DD-EE-FF	101010101011101111001100 110111011110111011111111	101010101011101111001100 110111011110111011111111	pass
2	11-22-33- 44-55-66	000100010010001000110011 010001000101010101100110	000100010010001000110011 010001000101010101100110	pass
3	BB-DA-5C- 4F-A1-D2	101110111101101001011100 010011111010000111010010	101110111101101001011100 010011111010000111010010	pass

```
Source MAC address hex : AA-BB-CC-DD-EE-FF
Des MAC address hex : FF-EE-DD-CC-BB-AA
BSS-Id MAC address hex : 11-22-33-44-55-66
Source MAC address in binary : 101010101011101111001100110111011110111011111111
Des MAC address : 111111111110111011101110111001100101110111010101010
BSS-Id: 000100010010001000110011010001000101010101100110
```

รูปที่ 4.41 ผลที่ได้จากการทดสอบการรับค่าแมคแอดเดรสโดย input คือ AA-BB-CC-DD-EE-FF

```
Source MAC address hex : BB-DA-5C-4F-A1-D2
Des MAC address hex : 12-A3-D2-19-9F-1B
BSS-Id MAC address hex : A3-4D-FE-2C-DA-AC
Source MAC address in binary : 101110111101101001011100010011111010000111010010
Des MAC address : 000100101010001111010010000110011001111100011011
BSS-Id: 101000110100110111111111000101100110110101010101100
```

รูปที่ 4.42 ผลที่ได้จากการทดสอบการรับค่าแมคแอดเดรส โดย input คือ BB-DA-5C-4F-A1-D2

4.5.7 ทดสอบการคำนวณขนาดของข้อมูล

จุดประสงค์ : เพื่อตรวจสอบว่าโปรแกรมสามารถคำนวณขนาดของข้อมูลที่รับมาได้อย่างถูกต้องและมี 16 บิต

คำอธิบาย : ในการคำนวณจะนับขนาดของข้อมูลเป็นจำนวนไบท์ และแปลงให้อยู่ในรูปของเลขฐานสองที่มี 16 บิต

ตารางที่ 4.11 Test case ของการคำนวณขนาดของข้อมูล

ครั้งที่	Input	Expect output	Actual output	Result
1	ข้อมูลขนาด 3 ไบท์	0000000000000011	0000000000000011	pass
2	ข้อมูลขนาด 4 ไบท์	0000000000000100	0000000000000100	pass
3	ข้อมูลขนาด 0 ไบท์	0000000000000000	0000000000000000	pass

```
Data has 3 bytes
Data : 10101100110101100110101100110
Length : 0000000000000011
BUILD SUCCESSFUL (total time: 1 second)
```

รูปที่ 4.43 ผลที่ได้จากการทดสอบการคำนวณขนาดข้อมูลโดย input คือ ข้อมูลขนาด 3 ไบท์

```
Data has 4 bytes
Data : 10101100110101100110101100111010110011
Length : 0000000000000100
BUILD SUCCESSFUL (total time: 1 second)
```

รูปที่ 4.44 ผลที่ได้จากการทดสอบการคำนวณขนาดข้อมูลโดย input คือ ข้อมูลขนาด 4 ไบท์

```
Data has 0 bytes
Data :
Length : 0000000000000000
BUILD SUCCESSFUL (total time: 1 second)
```

รูปที่ 4.45 ผลที่ได้จากการทดสอบการคำนวณขนาดข้อมูลโดย input คือ ข้อมูลขนาด 0 ไบท์

บทที่ 5
สรุปผล

โครงงานนี้ได้ศึกษาและพัฒนาไลบรารีเพิ่มเติมจากโครงงานเดิม โดยใช้ภาษาจาวาในการพัฒนา ซึ่งได้สร้างตัวอย่างสำหรับการจำลองการทำงานของเลเยอร์ 1 และเลเยอร์ 2 ของทั้ง Ethernet และ Wireless ซึ่งสามารถนำไปเชื่อมต่อกับเลเยอร์ 3 และเลเยอร์ 4 ของ เน็ตเวิร์คโพรโตคอลแสตกแบบ TCP/IP ซึ่งทำการอิมพลีเมนต์มาก่อนหน้านี้ เพื่อให้เห็นภาพการทำงานของระบบเครือข่ายเพิ่มมากขึ้น นอกจากนี้ผู้ใช้ยังสามารถนำโครงร่างนี้ไปอิมพลีเมนต์โพรโตคอลอื่นๆ ต่อได้ ไม่ว่าจะเป็นเลเยอร์ไหนก็ตาม ซึ่งสามารถช่วยให้การจำลองเน็ตเวิร์คโพรโตคอลแสตกมีความหลากหลายมากยิ่งขึ้น

ซอฟแวร์ที่ใช้ในการสร้างโปรแกรมในการจำลองของเน็ตเวิร์คโพรโตคอลแสตกแบบ TCP/IP ได้ใช้ภาษาจาวา และใช้โปรแกรม NetBeans 6.5 เป็นเครื่องมือหลักในการพัฒนาโปรแกรมนี้

5.1 ผลการทดลอง

5.1.1 ผลการทดลองการทดสอบคลาส

จากการทดสอบคลาสในเลเยอร์ 2 Ethernet ฝั่งส่ง สามารถรับค่าแมคแอดเดรส คำนวณขนาดของข้อมูลที่รับมา สามารถแพด สามารถคำนวณค่าของ CRC-32 ได้อย่างถูกต้อง และแสดงรายละเอียดของเฮดเดอร์ และข้อมูลให้ผู้ใช้ได้ดูอย่างถูกต้อง ครบถ้วน

จากการทดสอบคลาสในเลเยอร์ 2 Ethernet ฝั่งรับ สามารถตัดส่วนที่แพด ก่อนส่งข้อมูลต่อไปได้ และสามารถแสดงรายละเอียดของเฮดเดอร์และข้อมูลที่รับมาซึ่งตรงกับข้อมูลของเลเยอร์ 2 ในฝั่งส่งให้ผู้ใช้ได้ดูอย่างถูกต้อง ครบถ้วน

จากการทดสอบคลาสในเลเยอร์ 1 Ethernet สามารถเข้ารหัสและถอดรหัสแบบ 4B/5B และแสดงรายละเอียดของเฮดเดอร์และข้อมูลให้ผู้ใช้ได้ดูอย่างถูกต้อง ครบถ้วน

จากการทดสอบคลาสในเลเยอร์ 2 Wireless ฝั่งส่ง สามารถรับค่าแมคแอดเดรส คำนวณลำดับของข้อมูลที่รับมา ทำการแบ่งข้อมูล คำนวณค่า CRC-32 และแสดงรายละเอียดของเฮดเดอร์และข้อมูลให้ผู้ใช้ได้ดูอย่างถูกต้องครบถ้วน

จากการทดสอบคลาสในเลเยอร์ 2 Wireless ฝั่งรับ สามารถทำการรวมข้อมูล ทำการส่งเฟรมตอบรับเมื่อได้รับข้อมูล และสามารถแสดงรายละเอียดของเฮดเดอร์ และข้อมูลที่รับมาซึ่งตรงกับข้อมูลของเลเยอร์ 2 ในฝั่งส่งให้ผู้ใช้ได้ดูอย่างถูกต้องครบถ้วน

จากการทดสอบคลาสในเลเยอร์ 1 Wireless สามารถคำนวณขนาดของเฟรมที่รับมาจากเลเยอร์ 2 สามารถคำนวณ CRC-16 สามารถเข้ารหัสและถอดรหัสแบบ 4-PPM

5.1.2 ผลการทดลองการทดสอบตั้งแต่ต้นจนจบ (End-to-End)

ในการทดสอบตั้งแต่ต้นจนจบ สามารถเรียกใช้คลาสเลเยอร์ 1 และเลเยอร์ 2 ที่สร้างมามาอิมพลีเมนต์ในเลเยอร์ 1 และเลเยอร์ 2 ในไลบรารี ซึ่งสามารถเชื่อมต่อกับเลเยอร์ 3 และเลเยอร์ 4 ที่สร้างมาก่อนหน้านี้ และได้ผลลัพธ์ถูกต้องทั้ง Ethernet Suite, Wi-Fi Suite, Ethernet on Wi-Fi ซึ่งผู้ใช้สามารถจำลองโพรโตคอลอื่นมาอิมพลีเมนต์ในเลเยอร์ 1 และเลเยอร์ 2 หรือเลเยอร์อื่นๆ ได้ตามต้องการ

5.1.3 ผลการทดลองการทดสอบบนระบบปฏิบัติการ

การทดสอบบนระบบปฏิบัติการโดยการนำโปรแกรมมาทดสอบทำการส่งและรับข้อมูลบนระบบการ Window XP และระบบปฏิบัติการ Linux Ubuntu ได้ผลดังนี้

- สามารถทำงานบนระบบปฏิบัติการ Windows XP ทั้ง 2 เครื่อง ได้
- สามารถทำงานบนระบบปฏิบัติการ Linux Ubuntu ทั้ง 2 เครื่อง ได้
- สามารถทำงานระหว่างระบบปฏิบัติการ Windows XP และ Linux Ubuntu ได้

5.1.4 ผลการทดลองการทดสอบหน่วยย่อย

การทดสอบ CRC-32 CRC-16 การเข้ารหัสและถอดรหัสแบบ 4B/5B การเข้ารหัส และถอดรหัสแบบ 4-PPM การแพคข้อมูล การรับค่าแอดเดรส การคำนวณขนาดของข้อมูลถูกต้อง

5.2 ปัญหาและอุปสรรค

1. ขาดความชำนาญในการเขียนโปรแกรมด้วยภาษาจาวาจึงทำให้เกิดความล่าช้าในการทำความเข้าใจในตัวโปรแกรมที่สร้างมาก่อนหน้านี้ และการพัฒนาโปรแกรม
2. การศึกษาเรื่องเน็ตเวิร์ค นั้นเนื่องจากเอกสารที่ใช้ส่วนใหญ่เป็นภาษาต่างประเทศและมีรายละเอียดในแต่ละเอกสารไม่มากนัก จึงต้องทำการรวบรวม และประมวลให้มีความสัมพันธ์กัน ทำให้ต้องใช้เวลามาก
3. มีปัญหาในการที่จะเขียนโปรแกรมให้เชื่อมต่อกับของเดิม ทำให้ต้องมีการเข้าไปแก้ในตัวโปรแกรมเดิมบ้าง
4. เนื่องจากเลเยอร์ 2 มี 2 ชั้นสื่อสารย่อยทำให้การออกแบบมีความยุ่งยากกว่าเดิม

5.3 ข้อเสนอแนะ

1. ผู้ใช้ต้องมีความรู้พื้นฐานในการเขียนโปรแกรมด้วยภาษาจาวาพอสมควรจะทำให้ง่ายต่อการพัฒนาโปรแกรมต่อไป
2. เนื่องจากใช้เวลาในการพัฒนาค่อนข้างนาน จึงควรวางแผนงานให้รอบคอบเพื่อให้เสร็จทันกำหนด
3. เพื่อให้ง่ายต่อการเพิ่มเติมและแก้ไขโปรแกรม Subversion เป็นทางเลือกหนึ่งที่ดี
4. ผู้พัฒนาควรทำการตรวจสอบการรับส่งข้อมูลระหว่างเลเยอร์ว่าเป็นไปอย่างถูกต้อง

5.4 สรุป

จากผลการทดลองโปรแกรมจำลองการทำงานสำหรับเน็ตเวิร์คโพรโตคอล TCP/IP เลเยอร์ 1 และเลเยอร์ 2 (TCP/IP Network Protocol (Layer 1 and Layer 2) Simulation) โดยใช้ภาษาจาวาสามารถใช้คลาสของ Ethernet และ Wireless ที่ทำการสร้างขึ้น มาใช้ในไลบรารีได้ ซึ่งสามารถจำลองการทำงานของเน็ตเวิร์คได้ครบถ้วนสมบูรณ์และหลากหลายมากขึ้น คือได้ทั้ง Ethernet Suite, Wi-Fi Suite, Ethernet on Wi-Fi และโปรแกรมสามารถทำงานได้บนระบบปฏิบัติการ Microsoft Windows XP และระบบปฏิบัติการ Linux Ubuntu ได้ จากการทดลองแสดงให้เห็นว่าโปรแกรมสามารถทำงานได้ตามจุดประสงค์

เอกสารอ้างอิง

[1] Prakash C. Gupta. **Data Communication and Computer Networks.** Second Edition. New Delhi : Prentice-Hall of India Private Limited. Inc. 2006.

[2] Kaveh Pahlavan and Prashant Krishnamurthy. **Principles of Wireless Networks.** New York : Prentice-Hall. Inc. 2002.

[3] Behrouz A. Forouzan. **TCP/IP Protocol Suite.** Third Edition. New York : McGraw-Hill. Inc. 2006.

[4] H. M. Deitel. **Java™ How to Program.** Third Edition. New Jersey : Prentice-Hall. Inc. 2004.

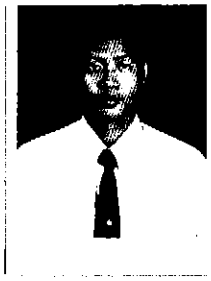
[5] Cay S. Horstmann and Cornell. **Core Java™ Volume II-Advanced Features.** Eighth Edition. California : Prentice-Hall. Inc. 2008.

[6] วรเศรษฐ สุวรรณิก. **Java GUI using NetBeans.** กรุงเทพมหานคร. โรงพิมพ์แห่งจุฬาลงกรณ์มหาวิทยาลัย. พ.ศ. 2551

[7] สัลยุทธ์ สว่างวรรณ. เครือข่ายคอมพิวเตอร์. พิมพ์ครั้งที่ 4. กรุงเทพมหานคร. บริษัท เพียร์สัน เอ็ดดูเคชั่น อินโดไชน่า จำกัด. พ.ศ. 2547

[8] กิตติ ภักดีวัฒนกุล. **JAVA ฉบับโปรแกรมเมอร์.** พิมพ์ครั้งที่ 2. กรุงเทพมหานคร. หจก. ไทยเจริญการพิมพ์. พ.ศ. 2542

ประวัติผู้เขียนโครงงาน



ชื่อ ชัยภัทร จารุชาต

ภูมิลำเนา 30/114 ถ.พระลือ ตำบลพระร่วง อำเภอเมือง จังหวัดพิษณุโลก 65000

ประวัติการศึกษา

- จบมัธยมศึกษาจากโรงเรียนอุตรดิตถ์ จังหวัดอุตรดิตถ์
- ปัจจุบันกำลังศึกษาอยู่ระดับปริญญาตรีชั้นปีที่ 4 สาขาวิชาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์ มหาวิทยาลัยนเรศวร

E-mail: **jaruch_4@hotmail.com**

ชื่อ นายนทภพ เริงนภาพล

ภูมิลำเนา 119/289 ถ.ลำปาง-แม่ทะ ต.พระบาท อ.เมือง จังหวัดลำปาง 52000

ประวัติการศึกษา

- จบมัธยมศึกษาจากโรงเรียนบุญวาทย์วิทยาลัย จังหวัดลำปาง
- ปัจจุบันกำลังศึกษาอยู่ระดับปริญญาตรีชั้นปีที่ 4 สาขาวิชาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์ มหาวิทยาลัยนเรศวร

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