สำนักงานคณะกรรมการวิจัยแห่งชาติ ห้องสมุดงานวิจัย

บทที่ 3
ระเบียบวิธีวิจัย

3.1 แนวทางการวิจัยและพัฒนา

งานวิจัยนี้มีวัตถุประสงค์พัฒนาโปรแกรมตรวจวัดคุณภาพการให้บริการสำหรับการใช้งาน VoIP โดยโปรแกรมที่พัฒนาจะเป็นการสร้างโปรแกรมประยุกต์จากโปรแกรมวิชวลเบสิก ซึ่งจะมี Freeware เป็นเครื่องมือในการช่วยพัฒนาคือ iperf และ ping บนปฏิบัติการ windows มาทำงานร่วมกันโดย iperf สามารถวัดค่าความจุของโครงข่าย แพ็คเก็ตสูญหาย และความแปรผันของความล่าช้าทางเวลา ส่วน ping บนปฏิบัติการ windows ก็จะสามารถวัดค่า ความล่าช้าของแพ็กเก็ต ซึ่งสามารถนำเสนอเพื่อเป็นแนวทางในการปรับปรุงแบนด์วิดท์ของเส้นทางได้ โดยแบ่งขั้นตอนการวิจัยออกเป็น 2 ส่วน ๆ ดังต่อไปนี้

3.1.1 ศึกษาและรวบรวมข้อมูล

1) จัดหา วิเคราะห์และศึกษาโปรแกรมที่เป็น Freeware ที่จะนำไปวัดค่าความจุของเครือข่าย การล่าช้าทางเวลาและความแปรผันของความล่าช้าทางเวลา

2) ศึกษาการส่งสัญญาณเสียงผ่านเครือข่ายไอพี การทำงานของโปรโตคอลและปัจจัยที่มีผลกระทบต่อคุณภาพเสียง เพื่อใช้เป็นแนวทางในการพัฒนาระบบ

3) ศึกษาวิธีการบีบอัดหรือเข้ารหัสสัญญาณเสียง แบบต่างๆ เพื่อใช้เป็นข้อมูลในการพัฒนาระบบ

4) ศึกษาเครื่องมือที่ใช้ในการพัฒนาโปรแกรมโดยจะใช้โปรแกรมวิชวลเบสิก เพื่อใช้สร้างโปรแกรมสำหรับคำนวณค่า MOS และ จำนวนผู้ใช้งานในโครงข่ายไอพีบนปฏิบัติการ Windows โดยอาศัยหลักการออกแบบโปรแกรมลักษณะ Visualize ซึ่งเพียงแต่เลือก Control ที่เหมาะสมแล้ววาดลงบน From ก็สามารถสร้างจอภาพที่ติดต่อกับผู้ใช้ได้

3.1.2 การออกแบบโปรแกรม

การออกแบบลักษณะรูปแบบการทำงานของโปรแกรมจะมี ส่วนที่ติดต่อกับผู้ใช้ ที่สามารถพัฒนาได้อย่างสะดวกเนื่องจากการออกแบบหน้าจอของโปรแกรมวิชวลเบสิกเป็นแบบกราฟฟิกโดยใช้เทคโนโลยีการลากแล้ววาง (Drag and Drop) ก็จะได้ส่วนประกอบที่ต้องการและหน้าจอที่ออกมาในขณะรันแอพพลิเคชั่นจะมีลักษณะเหมือนกับหน้าจอตอนที่ออกแบบ ซึ่งในส่วนนี้จะเป็นส่วนที่ให้ผู้ใช้กำหนดค่าความต้องการของผู้ใช้งาน รวมถึงขนาดความจุของเครือข่าย เพื่อ

ส่งค่าดังกล่าวทั้งหมดเข้าไปในโครงข่ายเพื่อให้โปรแกรมที่พัฒนาขึ้น หาค่าความล่าช้าทางเวลา ความแปรผันของความล่าช้าทางเวลา การสูญหายของแพ็กเก็ต และ ค่าความจุที่เหลือของเครือข่าย เพื่อนำไปคำนวณหาค่า MOS และ จำนวนผู้ใช้งานที่สามารถใช้ VoIP ได้พร้อมกันได้ในเวลาเดียวกัน โดยสามารถแบ่งส่วนการติดต่อกับผู้ใช้งานออกเป็น 2 ส่วนๆ ดังนี้

1) ส่วนกำหนดความต้องการจากผู้ใช้งาน ซึ่งส่วนนี้จะเป็นอินพุท (Input) จากผู้ใช้กำหนดค่าความต้องการเช่นการเลือกใช้งานว่าจะเลือกวัดอะไร เช่น วัด MOS หรือ Concurrent user การเลือก Mode ว่าจะเป็นแบบไหน Server หรือ Client กำหนดข้อมูลทางด้านปริมาณทราฟฟิก เช่น แบนด์วิดท์, IP Address และ CODEC

2) ส่วนแสดงผลคุณภาพของโครงข่าย VoIP โดยจะนำข้อมูลที่รับมาจากส่วนติดต่อกับผู้ใช้งานและ ข้อมูลที่วัดได้จากโครงข่าย มาประมวลผล

3.2 เครื่องมือที่ใช้ในงานวิจัย

3.2.1 ฮาร์ดแวร์

1) คอมพิวเตอร์ 5 เครื่อง ใช้เป็น User Agent Server (UAS) และ User Agent Client (UAC) อย่างละ 2 เครื่อง และใช้เป็น SIP Server (Asterisk) 1 เครื่อง

2) เครื่องมือวัด JDSU รุ่น HST-3000 Handheld Services Tester เพื่อใช้วัดเปรียบเทียบกับโปรแกรมที่พัฒนาขึ้นมา

3) Modem Router 1 ตัว ยี่ห้อ D-Link รุ่น DSL- 2640T (ตาม Spec จะรองรับแบนด์วิดท์สูงสุดที่ 100 Mbps) เพื่อนำมาทำเป็นเครือข่ายจำลองสำหรับใช้ในการทดสอบ

4) Switch 1 ตัว ยี่ห้อ TP-Link รุ่น TL-SF100D (ตาม Spec จะรองรับแบนด์วิดท์สูงสุดที่ 100 Mbps) เพื่อนำมาทำเป็นเครือข่ายจำลองสำหรับใช้ในการทดสอบ

5) Hub 1 ตัว ยี่ห้อ Compex รุ่น TP1008C (ตาม Spec จะรองรับแบนด์วิดท์สูงสุดที่ 10 Mbps) เพื่อนำมาทำเป็นเครือข่ายจำลองสำหรับใช้ในการทดสอบ

3.2.2 ซอฟแวร์

1) โปรแกรม iperf (Freeware) ที่นำมาใช้พัฒนาในงานวิจัยนี้ เพื่อนำมาวัดค่าความจุของเครือข่าย ความล่าช้าทางเวลา ความแปรผันของความล่าช้าทางเวลาและการสูญหายของแพ็กเก็ต

2) Ping บนปฏิบัติการ Windows เพื่อนำมาวัดหาค่าความล่าช้าทางเวลา

3) โปรแกรม Visual Basic เพื่อใช้ในการออกแบบ แอพพลิเคชั่นบนวินโดว์

3.3 แผนการดำเนินงาน

3.3.1 จัดหา วิเคราะห์และศึกษาโปรแกรมที่เป็น Freeware ที่สามารถวัดค่าความจุของเครือข่าย ความล่าช้าทางเวลา ความแปรผันของความล่าช้าทางเวลา การสูญหายของแพ็กเก็ต

3.3.2 ศึกษาโปรแกรมวิชวลเบสิก เพื่อนำมาพัฒนาแอพพลิเคชั่นบนวินโดว์ของโปรแกรมที่จะพัฒนาขึ้น

3.3.3 พัฒนาโปรแกรมเครื่องมือวัดที่เป็น Freeware ให้ง่ายต่อการใช้งาน และให้เป็นไปตามวัตถุประสงค์

3.3.4 ทำการทดสอบโปรแกรมเครื่องมือวัดที่พัฒนาขึ้น

3.3.5 สรุปผลการทดลองของโปรแกรมเครื่องมือวัดที่พัฒนาขึ้น ซึ่งมีการทำงานตามขั้นตอนในตารางที่ 3.1

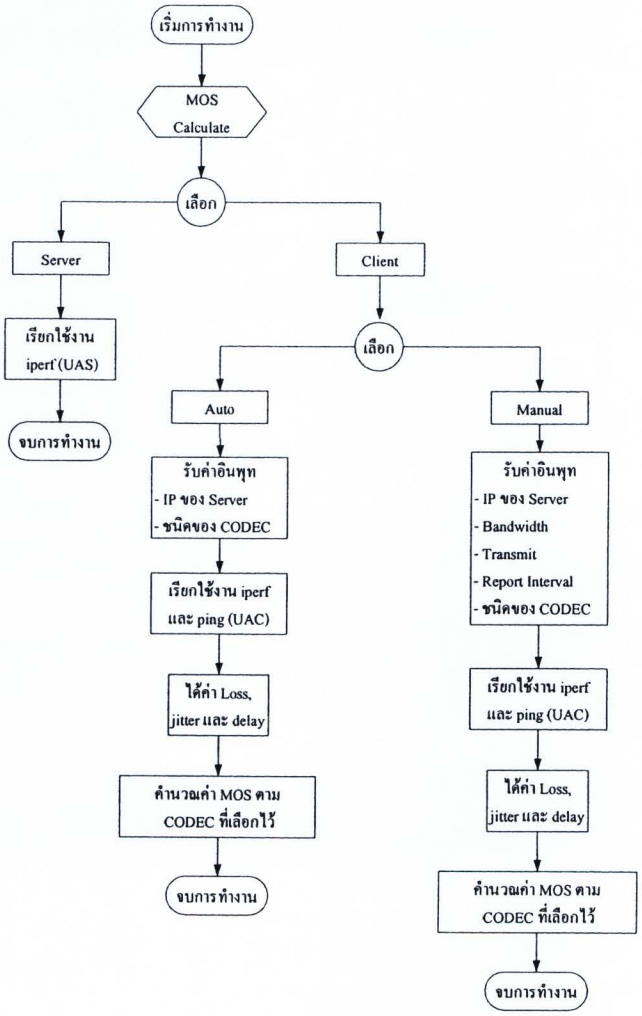
ตารางที่ 3.1 แผนการดำเนินงาน

เดือน / งาน	มค กพ 53	มีค เมย 53	พค มิย 53	กค สค 53	กย ตค 53	พย ธค 53	มค กพ 54	มีค เมย 54
รวบรวมข้อมูลโปรแกรมที่เป็น Freeware	■	■						
ศึกษาโปรแกรม Freeware ที่ใช้ในงานวิจัยนี้		■	■					
ศึกษาโปรแกรม Visual Basic			■	■				
ออกแบบและพัฒนาโปรแกรม				■	■	■		
ทดสอบโปรแกรมที่พัฒนาขึ้น					■	■	■	
สรุปผลการทดสอบ								■

3.4 ขั้นตอนและวิธีดำเนินงาน

3.4.1 แนวคิดการทำงานของโปรแกรม

ก่อนที่จะทำการออกแบบผู้วิจัยมีแนวคิดที่จะนำเอาโปรแกรมที่เป็น Freeware ที่สามารถวัดค่าความจุของเครือข่าย ความล่าช้าทางเวลา ความแปรผันของความล่าช้าทางเวลา และการสูญหายของแพ็กเก็ต ซึ่งจากการที่ได้ศึกษาค้นคว้าก็ได้พบว่าโปรแกรมที่เป็น Freeware ตัวหนึ่งที่ชื่อ iperf ซึ่งสามารถวัดค่าดั่งกล่าวข้างต้นได้ แต่ไม่สามารถวัดได้ค่าความล่าช้าทางเวลา จึงได้นำเอา Command ping ที่ใช้งานบน ปฏิบัติการ Windows มาทำงานร่วมกัน โดยใช้โปรแกรมวิชวลเบสิกเป็นตัวเชื่อมโยง ซึ่งทั้ง 2 โปรแกรมให้ทำงานร่วมกัน ซึ่งมีลักษณะการออกแบบให้ทำงานเป็น 2 ลักษณะคือ การวัดหาค่า MOS และ การวัดหาค่า Concurrent User ซึ่งขั้นตอนทำงานตามแผนภาพ (Flowchart) ดังแสดงดังต่อไปนี้



รูปที่ 3.1 แผนภาพการทำงานของโปรแกรมในการคำนวณหาค่า MOS

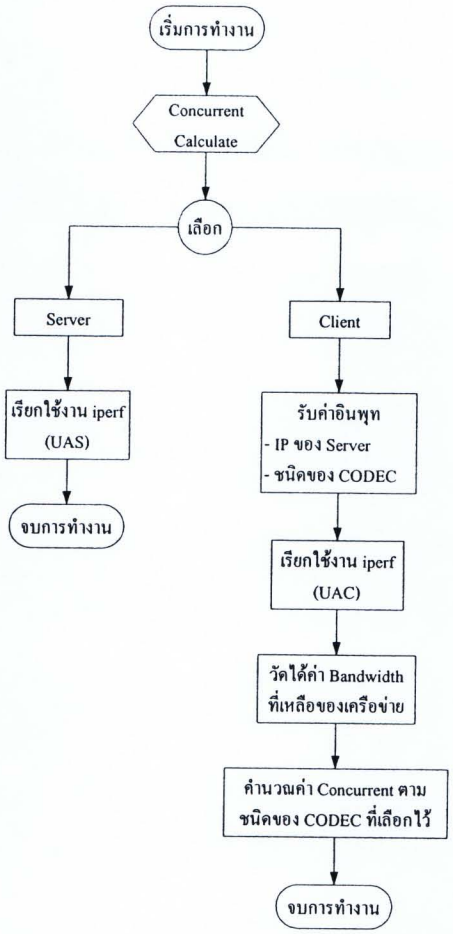
3.4.1.1 จากรูปที่ 3.1 เป็นแผนภาพแสดงการทำงานการของโปรแกรมในการคำนวณหาค่า MOS ซึ่งจะสามารถอธิบายการทำงานได้ดังนี้

เมื่อต้องการให้โปรแกรมที่พัฒนาขึ้นทำการวัดค่า MOS สิ่งที่จะต้องทำเป็นอันดับแรกคือจะต้องให้โปรแกรมที่เครื่องแม่ข่ายทำงานก่อน โดยโปรแกรมที่ทำหน้าที่เป็นเครื่องแม่ข่ายไม่ต้องการข้อมูลอะไรมาก เพียงแต่แค่เลือกว่าต้องการจะวัดค่า MOS กับการเลือกให้โปรแกรมทำงานเป็น Server เท่านั้น ก็สามารถให้โปรแกรมที่เป็นเครื่องแม่ข่ายทำงานได้เลย โดยโปรแกรมที่เป็นเครื่องแม่ข่ายเบื้องหลังจะมีการทำงานของ Iperf เพียงอย่างเดียวเท่านั้น โดยที่อีกฝั่งหนึ่งของเครื่องแม่ข่ายก็ต้องเลือกให้โปรแกรมทำงานเป็นเครื่องลูกข่าย ซึ่งเครื่องลูกข่ายจะมีการทำงานเป็น 2 ลักษณะคือ แบบ Auto และ Manual โดยการทำงานของแต่ละชนิดพอที่จะอธิบายตามขั้นตอนได้ดังต่อนี้

1) การให้โปรแกรมเครื่องลูกข่ายทำงานแบบ Auto โดยผู้ใช้จะต้องป้อนค่าลงไปในหน้าต่างของโปรแกรมตามที่กำหนด ซึ่งก็คือ IP Address ของเครื่องแม่ข่าย และชนิดของ CODEC ที่ต้องการจะทดสอบ แล้วทำการให้โปรแกรมที่เป็นเครื่องลูกข่ายทำงาน โดยการทำงานของโปรแกรมที่พัฒนาขึ้นนี้ เบื้องหลังจะมีการทำงานของโปรแกรม Iperf และ Ping โดยเครื่องลูกข่ายจะทำการส่ง UDP Protocol ซึ่งมีแบนด์วิดท์เท่ากับ CODEC ที่ผู้ใช้งานเลือกไว้ก่อนหน้านี้ ตัวอย่างเช่น ถ้าผู้ใช้งานเลือก CODEC G.711, G.729 และ GSM โปรแกรมที่ทำตัวเป็นเครื่องลูกข่ายก็จะทำการส่งแบนด์วิดท์ที่มีค่าเท่ากับ 88, 32, 37.2 Kbps ตามลำดับ (รวมเฮดเดอร์) ออกไปยังโปรแกรมตัวที่เป็นเครื่องแม่ข่าย และระยะเวลาที่กำหนดไว้ในการส่งไปหาเครื่องแม่ข่ายเท่ากับ 30 วินาที เมื่อครบระยะกำหนดเวลา เครื่องแม่ข่ายก็จะรายงานค่า ความล่าช้าทางเวลา ค่าความแปรผันของความล่าช้าทางเวลา และค่าการสูญหายของแพ็กเก็ต ไปยังเครื่องลูกข่าย ต่อจากนั้นโปรแกรมที่ทำหน้าที่เป็นเครื่องลูกข่ายก็จะทำการคำนวณหาค่า MOS ให้อัตโนมัติ โดยการนำค่าที่วัดได้จากเครือข่ายมาคำนวณค่า MOS ก็คือค่าความล่าช้าทางเวลา ค่าความแปรผันของความล่าช้าทางเวลา ค่าการสูญหายของแพ็กเก็ต และชนิดของ CODEC ที่ผู้ใช้งานเลือก

2) การให้โปรแกรมเครื่องลูกข่ายทำงานแบบ Manual จุดประสงค์ที่ต้องการให้เครื่องลูกข่ายทำงานลักษณะนี้ก็เพราะว่า เมื่อผู้ใช้งานต้องการวัดค่า MOS ในกรณีที่ต้องทำการ Make call จำนวนหลาย ๆ call พร้อม ๆ กัน ก็สามารถทำได้โดยการใส่ค่าลงในช่องของ Bandwidth ตามที่เราต้องการตัวอย่างเช่น เราต้องการหาค่า MOS เมื่อเราใช้งานโดยการโทรพร้อมกัน 3 สาย เมื่อใช้ CODEC G.711 เราก็ใส่ค่าลงในช่องของ Bandwidth เท่ากับ 88*3 = 264 kbps เป็นต้น ส่วนค่าอื่น ๆ ที่จำเป็นจะต้องใส่ เมื่อเลือกการทำงานแบบ Manual ก็คือ ค่า Transmit ซึ่งเป็นค่าระยะเวลาที่จะทำการวัดว่าจะต้องการระยะเวลาในการวัดนานแค่ไหน มีหน่วยเป็นวินาที สุดท้ายเป็นช่อง

ใส่ค่า Report Interval ซึ่งเป็นค่าระยะเวลาที่โปรแกรมจะรายงานออกมาให้ผู้ที่ทำการวัดทราบว่าจะให้รายงานให้ผู้วัดทราบทุก ๆ กี่วินาที



รูปที่ 3.2 แผนภาพการทำงานของโปรแกรมในการคำนวณหาค่า Concurrent Use

3.4.1.2 ส่วนในรูปที่ 3.2 เป็นแผนภาพแสดงการทำงานการของโปรแกรมในการคำนวณหาค่า Concurrent User ซึ่งจะมีการทำงานพอสรุปได้ดังนี้

เมื่อต้องการให้โปรแกรมที่พัฒนาขึ้นทำการวัดหาค่า Concurrent Use โดยตัวโปรแกรมที่เลือกเป็นเครื่องแม่ข่าย ก็ทำเช่นเดียวกับการวัดค่า MOS เพียงต่างกันที่ต้องเลือกวัดค่า Concurrent Use เท่านั้น แล้วให้เครื่องแม่ข่ายทำงานก่อน โดยโปรแกรมที่เป็นเครื่องแม่ข่ายจะมีโปรแกรม Iperf อยู่เบื้องหลังเพียงโปรแกรมเดียว โดยที่อีกฝั่งหนึ่งของเครื่องแม่ข่าย ก็ต้องเลือกให้โปรแกรมทำงานเป็นเครื่องลูกข่าย โดยผู้ใช้จะต้องป้อนค่าลงในโปรแกรมตามที่กำหนด ซึ่งก็คือ IP Address ของเครื่องแม่ข่าย และเลือกชนิดของ CODEC ที่จะใช้งาน แล้วก็ทำการ Run โปรแกรม เมื่อโปรแกรม

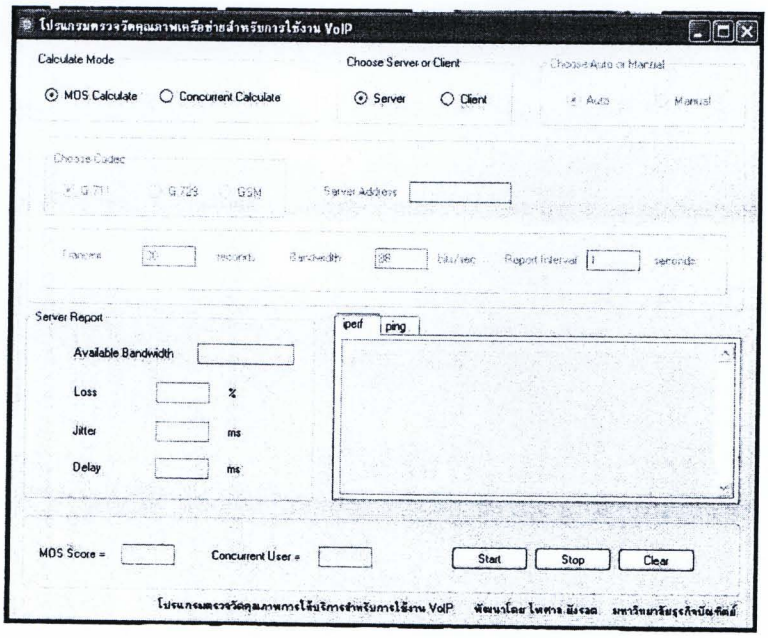
ทำการวัดค่า Bandwidth ที่เหลือของเครือข่ายออกมาได้ ก็จะคำนวณค่า ผู้ใช้งาน VoIP ในเครือข่ายที่สามารถใช้งานพร้อม ๆ กัน โดยค่าของ Concurrent Use ที่ได้ก็คือ ค่าของแบนด์วิดท์ที่เหลือที่วัดได้จากเครือข่าย นำมาหารด้วย แบนด์วิดท์ของ CODEC แต่ละชนิดที่ผู้วัดทำการเลือกไว้ก่อนนี้ โดย CODEC G.711 G.729, GSM จะมีค่าแบนด์วิดท์เท่ากับ 88, 32, 37.2 Kbps ตามลำดับ (รวมเฮดเดอร์) ตัวอย่างเช่น เมื่อวัดแบนด์วิดท์ในเครือข่ายได้เท่ากับ 8 Mbps และได้เลือก CODEC ที่ใช้เป็น G.711 ก็จะได้ค่า Concurrent Use เท่ากับ 39.09 User พร้อมกัน แต่โปรแกรมจะไม่แสดงทศนิยมจะแสดงเฉพาะจำนวนเต็มเท่านั้นคือ 39 User พร้อมกันเป็นต้น

3.4.2 การออกแบบโปรแกรม

ขั้นตอนในการพัฒนาโปรแกรม ประกอบด้วยขั้นตอนหลัก 2 ขั้นตอน ดังนี้

3.4.2.1 การสร้างจอภาพของโปรแกรม

จากที่ได้ออกแบบลักษณะการทำงานของโปรแกรมตามโฟร์ชาร์ต เป็นที่เรียบร้อยแล้ว ถึงตรงนี้ โปรแกรมที่ช่วยในการออกแบบการพัฒนาโปรแกรมตรวจวัดคุณภาพการให้บริการสำหรับการใช้งาน VoIP จะถูกพัฒนาขึ้นด้วยโปรแกรมวิชวล เบสิก เนื่องจากโปรแกรมวิชวล เบสิก เป็นโปรแกรมภาษาที่สามารถพัฒนาแอพพลิเคชั่นบนวินโดว์ได้อย่างสะดวกและรวดเร็ว มีเครื่องมือสนับสนุนมากมายทั้งจากไมโครซอฟท์และบริษัทอื่นๆ และเป็นโปรแกรมแอพพลิเคชั่นที่สนับสนุนการคำนวณและมีใช้งานกันอย่างแพร่หลาย ซึ่งในขั้นต้นจะเป็นการสร้างจอภาพของโปรแกรมเพื่อใช้ในการติดต่อกับผู้ใช้ จะมีลักษณะตามรูปที่ 3.3



รูปที่ 3.3 ลักษณะจอภาพของโปรแกรม

จากรูปแบบจอภาพของโปรแกรม สามารถอธิบายการทำงานของโปรแกรมออกเป็น 2 ส่วนใหญ่ ๆ ส่วนที่ผู้ใช้ต้องป้อนข้อมูล และ ส่วนแสดงผลของโปรแกรม ดังต่อไปนี้

1) ส่วนที่ผู้ใช้ต้องป้อนข้อมูล

(1) Calculate Mode เป็นการเลือกให้โปรแกรมคำนวณหา MOS หรือ คำนวณ Concurrent use จากเครือข่ายที่ต้องการจะวัด

(2) Server or client เป็นส่วนที่จะให้โปรแกรมทำงานเป็น Server หรือ Client

(3) Auto and Manual เป็นการเลือกที่จะให้โปรแกรมทำงานแบบ อัตโนมัติหรือผู้ใช้ต้องการป้อนข้อมูลของส่วนต่าง ๆ เอง

(4) CODEC เป็นการเลือกความต้องการของผู้ใช้ว่าต้องการใช้การเข้ารหัสเสียงแบบใดในเครือข่าย

(5) Server Address เป็นส่วนที่ผู้ใช้ต้องป้อนข้อมูลของ IP Address Server เมื่อผู้ใช้เลือกการทำงานเป็น Client

(6) Bandwidth เป็นส่วนที่ผู้ใช้ต้องป้อนค่า แบนด์วิดท์ ที่จะให้โปรแกรมส่งค่าแบนด์วิทด์นี้เข้าไปในเครือข่ายที่ต้องการวัด เมื่อผู้ใช้เลือกการทำงานแบบ Manual ในการใช้งานเมื่อวัด MOS

(7) Transmit เป็นส่วนที่ผู้ใช้งานต้องป้อนค่าซึ่งมีค่าเป็นวินาที ซึ่งเป็นระยะเวลาที่ผู้ใช้งานต้องการจะวัดว่าต้องการใช้ระยะเวลาในการวัดนานเท่าไร เมื่อผู้ใช้เลือกการทำงานแบบ Manual

(8) Report Interval เป็นส่วนที่ผู้ใช้งานต้องป้อนค่าซึ่งมีค่าเป็นวินาที ซึ่งเป็นระยะเวลาที่ผู้ใช้งานต้องการให้โปรแกรมรายงานทุก ๆ กี่วินาที เมื่อผู้ใช้เลือกการทำงานแบบ Manual

2) ส่วนแสดงผลของโปรแกรม

(1) Available Bandwidth เป็นแบนด์วิดท์ที่เหลือที่โปรแกรมสามารถวัดได้จากเครือข่าย

(2) Loss เป็นการสูญเสียของข้อมูลในเครือข่ายที่สามารถวัดได้ซึ่งจะมีหน่วยเป็นเปอร์เซ็นต์

(3) Jitter เป็นการความผันแปรเฉลี่ยของระยะเวลาดีเลย์ ในเครือข่ายที่สามารถวัดได้โดยจะมีหน่วย ms

(4) Delay เป็นความล่าช้าทางเวลาที่สามารถวัดได้จากเครือข่าย โดยจะมีหน่วยเป็น ms

(5) MOS Score เป็นค่า MOS ที่โปรแกรมสามารถคำนวณได้จากเครือข่ายที่ทำการวัด

(6) Concurrent Use เป็นค่าที่ผู้ใช้งานสามารถใช้งานโทรศัพท์ไอพีได้พร้อมกันในเวลาขณะนั้น

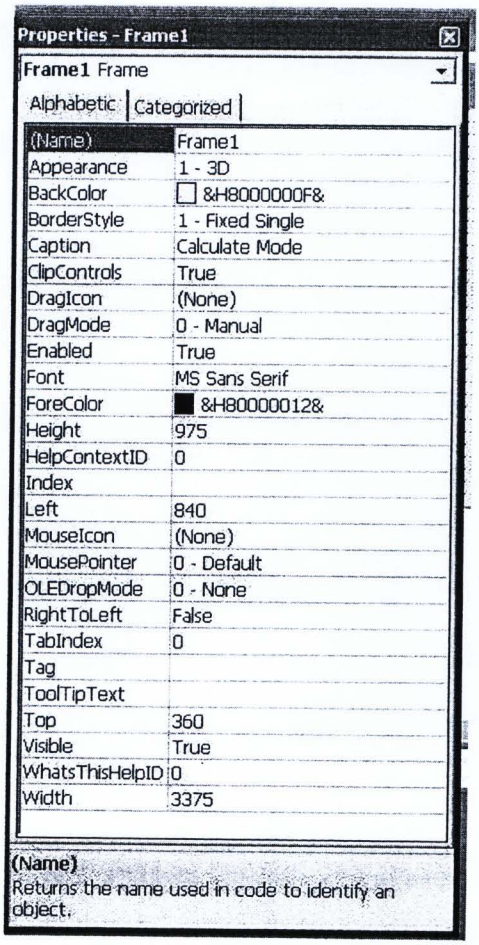
(7) Start เป็นปุ่มให้โปรแกรมเริ่มทำงานเมื่อป้องค่าต่าง ๆ เรียบร้อย

(8) Stop เป็นปุ่มให้โปรแกรมหยุดการทำงานทันที เมื่อต้องการหยุดการทำงานของโปรแกรม

(9) Clear เป็นปุ่มที่ผู้ใช้ต้องการล้างข้อมูลทั้งหมดที่ผู้ใช้งานป้อนเข้าไปในช่องต่าง ๆ

3.4.2.2 การเขียนคำสั่ง

เมื่อขั้นตอนการออกแบบสร้างจอภาพของโปรแกรมเพื่อใช้ในการติดต่อกับผู้ใช้เสร็จสิ้น ขั้นตอนต่อมาก็จะเป็นการกำหนดคุณสมบัติ Property ให้กับแต่ละ Object ที่ปรากฏอยู่บน Form ซึ่งรูปที่จะแสดงให้เห็นเป็นการกำหนดค่าให้กับ Property ของ Calculate Mode ดังรูปที่ 3.4



รูปที่ 3.4 ตัวอย่างรูป Property ของ Calculate Mode

การเขียนรูปแบบของคำสั่งเพื่อกำหนดการทำงานให้แต่ละ Object ภายใต้เหตุการณ์ต่างๆ ที่จะเกิดขึ้นกับจอภาพนั้นๆ โดยจะแสดงรายละเอียดของคำสั่งเป็นตัวอย่าง ซึ่งเป็นคำสั่งบน Object บนปุ่มที่ชื่อว่า Server ดังแสดงในรูปที่ 3.5

```
Private Sub OptServer_Click()
If StartStatus Then
cmdStartServer.Enabled = False
Else
cmdStartServer.Enabled = True
End If
If StartStatus Then
cmdGetData.Enabled = True
Else
cmdGetData.Enabled = False
End If

cmdGetData.Visible = True
cmdProcess.Visible = False
txtListenPort.Enabled = True
txtListenPort.BackColor = &H80000005
txtServerAddress.Text = ""
txtServerAddress.Enabled = False
txtServerAddress.BackColor = &H8000000F
txtBandWidth.Enabled = False
txtPort.Enabled = False
txtReInterval.Enabled = False
txtTransmit.Enabled = False
txtBandWidth.BackColor = &H8000000F
txtPort.BackColor = &H8000000F
txtReInterval.BackColor = &H8000000F
txtTransmit.BackColor = &H8000000F
End Sub
```

รูปที่ 3.5 ลักษณะของรูปแบบคำสั่งบน Object ที่ชื่อว่า Server

ในรูปที่ 3.5 จะแสดงตัวอย่างรูปแบบของคำสั่งเพื่อให้คำนวณการหา Concurrent User ที่เกิดขึ้นบน Object ที่ชื่อว่า Start เมื่อทำการเลือกให้โปรแกรมทำงานเป็นตัว Client

```
Private Sub calConcurrentUser()
  If txtServerAddress.Text = "" Or txtBandWidth.Text = "" Or txtPort.Text = "" Or txtReInterval.Text = "" Or txtTransmit.Text = "" Then
    Exit Sub
  End If
  Dim BandwidthValue As Single
  Dim ConcurrentUserValue As Integer

  If optPPP.Value = True Then
    If optG711.Value = True Then
      BandwidthValue = 84.8
    End If
    If optG729.Value = True Then
      BandwidthValue = 28.8
    End If
    If optGSM.Value = True Then
      BandwidthValue = 34
    End If
  End If

  If optEthenet.Value = True Then
    If optG711.Value = True Then
      BandwidthValue = 88
    End If
    If optG729.Value = True Then
      BandwidthValue = 32
    End If
    If optGSM.Value = True Then
      BandwidthValue = 37.2
    End If
  End If

  If optFrameRelay.Value = True Then
    If optG711.Value = True Then
      BandwidthValue = 84
    End If
    If optG729.Value = True Then
      BandwidthValue = 28
    End If
    If optGSM.Value = True Then
      BandwidthValue = 33.2
    End If
  End If

  ConcurrentUserValue = Fix((lblBanRemain.Caption * 1024) / BandwidthValue)
  lblConcurrentUser.Caption = ConcurrentUserValue
End Sub
```

รูปที่ 3.6 ตัวอย่างของคำสั่งในการคำนวณหา Concurrent User

จากการศึกษา วิธีการคำนวณหาคุณภาพของเครือข่ายจากโปรแกรมที่พัฒนาขึ้น ซึ่งสามารถวัดค่าความล่าช้าทางเวลา, การสูญหายของแพ็กเก็ต และ ความแปรผันของความล่าช้าทางเวลาแล้ว จากนั้นนำมาคำนวณตามสมการที่ได้ศึกษาค้นคว้ามา ซึ่งใช้ในเครื่องวัดที่มีชื่อว่า PingPlotter (Nessoft, LLC, 2005) ค่า R-factor ที่นำมาใช้ในสมการ มาจากตารางที่ 3.2 ซึ่งแสดงการเปรียบค่า MOS ต่างๆ กับค่า R-factor

ตารางที่ 3.2 การเปรียบค่า MOS Score กับค่า R-factor

User Opinion	R Factor	MOS Score
Very satisfied	90 - 100	4.3 – 5.0
Satisfied	80 - 90	4.0 – 4.3
Some users satisfied	70 - 80	3.6 – 4.0
Many users dissatisfied	60 - 70	3.1 – 3.6
Nearly all users dissatisfied	50 -60	2.1 – 3.1
Not recommended	0 - 50	1.0 – 2.6

อ้างอิงจากบทที่ 2 จะเห็นว่าค่า MOS Score ของแต่ละ CODEC มีค่าดังนี้

CODEC	G.711	จะมี MOS สูงสุด	4.20
CODEC	G.729	จะมี MOS สูงสุด	3.91
CODEC	GSM	จะมี MOS สูงสุด	3.57

ตัวอย่างวิธีการคำนวณหาค่า MOS ของโปรแกรมที่พัฒนาโดยเริ่มจาก CODEC G.711 (ค่า MOS ที่ดีที่สุดจะมีค่าเท่ากับ 4.2 ซึ่งอ้างอิงจากเครื่องมือวัด JDSU) ซึ่งจะได้ค่าคงที่เท่ากับ 85.3 ซึ่งค่าคงที่นี้ได้จากการแทนค่าจากสมการที่ (3) ถึงสมการที่ (7) ในบทที่ 2 เพื่อให้ได้ผลลัพธ์ค่า MOS เท่ากับ 4.2 โดยการคำนวณนี้ค่าคงที่นี้ยังไม่รวมกับค่าผลกระทบที่เกิดจากค่าความล่าช้าทางเวลา ความแปรผันของความล่าช้าทางเวลา และการสูญหายของแพ็กเก็ตในเครือข่ายรวมเข้ามา (คือให้ค่าเหล่านั้นเป็น 0) แล้วแทนค่าแล้วให้ได้ผลลัพธ์ของค่า MOS เท่ากับ 4.20

จากนั้นเมื่อใช้โปรแกรมที่เราได้พัฒนาขึ้นมาตรวจวัดในเครือข่ายก็จะมีค่า ความล่าช้าทางเวลา ความแปรผันของความล่าช้าทางเวลา และการสูญหายของแพ็กเก็ต แสดงออกมาให้เห็นก็จะนำค่าเหล่านั้นมาคำนวณตามสมการที่ (3) ถึง (7) ในบทที่ 2 ดังจะแสดงให้เห็นต่อไปนี้

<u>ตัวอย่าง</u> สมมติค่าที่วัดได้จากโปรแกรมที่พัฒนาขึ้นมาสามารถวัดค่าได้ค่าดังนี้

Loss = 2%

Jitter = 3 ms

Delay = 8 ms

จากสมการ Effective Latency = [Average Latency + (Jitter * 2) + 10]

= [8 + (3*2) + 10]

= 24

จากสมการหาค่า R_I = 85.3 - (Effective Latency / 40)

= 85.3 – (24 / 40)

= 84.7

จากสมการหาค่า R_f = R_I - (Packet Loss * 2.5)

= 84.7 – (2 * 2.5)

= 79.7

จากสมการหาค่า MOS $= 1 + (0.035) * R_f + (.000007) * R_f * (R_f - 60) * (100 - R_f)$

$= 1 + (0.035) * 79.7 + (.000007) * 79.7 * (79.7 - 60) *$ (100- 79.7)

$= 1 + 2.79 + 0.22$

ฉะนั้นจะได้ค่า MOS = 4.01 (เป็นค่า MOS ที่วัดได้จากโปรแกรมที่พัฒนาขึ้น)

เช่นเดียวกันที่ CODEC G.729 (ค่า MOS ที่ดีที่สุดจะมีค่าเท่ากับ 3.91 ซึ่งอ้างอิงจากเครื่องมือวัด JDSU) ซึ่งจะได้ค่าคงที่เท่ากับ 77.3 ซึ่งค่าคงที่นี้ได้จากการแทนค่าจากสมการที่ (3) ถึงสมการที่ (7) ในบทที่ 2 เพื่อให้ได้ผลลัพธ์ค่า MOS เท่ากับ 3.91 โดยการคำนวณนี้ค่าคงที่นี้ยังไม่รวมกับค่าผลกระทบที่เกิดจากค่าความล่าช้าทางเวลา ความแปรผันของความล่าช้าทางเวลา และการสูญหายของแพ็กเก็ตในเครือข่ายรวมเข้ามา (คือให้ค่าเหล่านั้นเป็น 0) แล้วแทนค่าแล้วให้ได้ผลลัพธ์ของค่า MOS เท่ากับ 3.91

จากนั้นเมื่อใช้โปรแกรมที่เราได้พัฒนาขึ้นมาตรวจวัดในเครือข่ายก็จะมีค่า ความล่าช้าทางเวลา ความแปรผันของความล่าช้าทางเวลา และการสูญหายของแพ็กเก็ต แสดงออกมาให้เห็นก็จะนำค่าเหล่านั้นมาคำนวณตามสมการที่ (3) ถึง (7) ในบทที่ 2 ดังจะแสดงให้เห็นต่อไปนี้

<u>ตัวอย่าง</u> สมมติค่าที่วัดได้จากโปรแกรมที่พัฒนาขึ้นมาสามารถวัดค่าได้ค่าดังนี้

Loss = 1 %

Jitter = 2 ms

Delay = 8 ms

จากสมการ Effective Latency = [Average Latency + (Jitter * 2) + 10]

= [8 + (2*2) + 10]

= 22

จากสมการหาค่า R_I = 77.3 - (Effective Latency / 40)

= 77.3 – (22 / 40)

= 76.75

จากสมการหาค่า R_f = R_I - (Packet Loss * 2.5)

= 76.75 – (1 * 2.5)

= 74.25

จากสมการหาค่า MOS $= 1 + (0.035) * R_f + (.000007) * R_f * (R_f - 60) * (100 - R_f)$

$= 1 + (0.035) * 74.25 + (.000007) * 74.25 * (74.25 - 60) *$ (100- 74.25)

$= 1 + 2.59 + 0.19$

ฉะนั้นจะได้ค่า MOS = 3.78 (เป็นค่า MOS ที่วัดได้จากโปรแกรมที่พัฒนาขึ้น)

ส่วนที่ CODEC GSM (ค่า MOS ที่ดีที่สุดจะมีค่าเท่ากับ 3.57 ซึ่งอ้างอิงจากเครื่องมือวัด JDSU) ซึ่งจะได้ค่าคงที่เท่ากับ 69.7 ซึ่งค่าคงที่นี้ได้จากการแทนค่าจากสมการที่ (3) ถึงสมการที่ (7) ในบทที่ 2 เพื่อให้ได้ผลลัพธ์ค่า MOS เท่ากับ 3.57 โดยการคำนวณนี้ค่าคงที่นี้ยังไม่รวมกับค่าผลกระทบที่เกิดจากค่าความล่าช้าทางเวลา ความแปรผันของความล่าช้าทางเวลา และการสูญหายของแพ็กเก็ตในเครือข่ายรวมเข้ามา (คือให้ค่าเหล่านั้นเป็น 0) แล้วแทนค่าแล้วให้ได้ผลลัพธ์ของค่า MOS เท่ากับ 3.57

จากนั้นเมื่อใช้โปรแกรมที่เราได้พัฒนาขึ้นมาตรวจวัดในเครือข่ายก็จะมีค่า ความล่าช้าทางเวลา ความแปรผันของความล่าช้าทางเวลา และการสูญหายของแพ็กเก็ต แสดงออกมาให้เห็นก็จะนำค่าเหล่านั้นมาคำนวณตามสมการที่ (3) ถึง (7) ในบทที่ 2 ดังจะแสดงให้เห็นต่อไปนี้

<u>ตัวอย่าง</u> สมมติค่าที่วัดได้จากโปรแกรมที่พัฒนาขึ้นมาสามารถวัดค่าได้ค่าดังนี้

Loss = 0 %

Jitter = 0 ms

Delay = 4 ms

จากสมการ Effective Latency = [Average Latency + (Jitter * 2) + 10]

= [4 + (0*2) + 10]

= 14

จากสมการหาค่า R_I = 69.7 - (Effective Latency / 40)

= 69.7 – (14 / 40)

= 69.35

จากสมการหาค่า R_f = R_I - (Packet Loss * 2.5)

= 69.35 – (0 * 2.5)

= 69.35

จากสมการหาค่า MOS = $1 + (0.035) * R_f + (.000007) * R_f * (R_f - 60) * (100 - R_f)$

$= 1 + (0.035) * 69.35 + (.000007) * 69.35 * (69.35 - 60) * (100 - 69.35)$

$= 1 + 2.43 + 0.14$

ฉะนั้นจะได้ค่า MOS = 3.57 (เป็นค่า MOS ที่วัดได้จากโปรแกรมที่พัฒนาขึ้น)

ส่วนในการคำนวณหา Concurrent Use นั้น ก็ได้ใช้โปรแกรม Iperf ในการหาแบนด์วิดท์ที่เหลือของเครือข่าย โดยการป้อนแบนด์วิดท์โดยคร่าวๆ ของเครือข่าย ควรจะป้อนค่าแบนด์วิดท์ที่สูงกว่าแบนด์วิดท์จริงในโครงข่ายเล็กน้อยแล้วคอยสังเกตค่าดีเลย์ และจิสเตอร์ แล้วค่อยๆ ลดค่าแบนด์วิดท์ลงมาจนค่าดีเลย์ และจิสเตอร์ ไม่มีในเครือข่าย นั้นก็คือค่าแบนด์วิดท์ที่เหลือในเครือข่ายที่สามารถใช้งานได้จริง จากนั้นโปรแกรมก็จะทำการคำนวณ ตาม CODEC และ ลักษณะการเชื่อมต่อ ที่ผู้ใช้งานเลือกไว้ ก็จะสามารถคำนวณออกมาเป็นจำนวนผู้ใช้งานจริง ที่สามารถใช้งาน VoIP ได้ในเครือข่าย ณ ขณะนั้น ซึ่ง CODEC ต่างๆ ที่มาคำนวณ จะมีค่าตามตารางที่ 3.3

ตารางที่ 3.3 ค่าแบนด์วิดท์โดยรวมของ CODEC ต่างๆ ในการเชื่อมต่อแบบ Ethernet

TYPE OF CODEC	แบนด์วิดท์รวม (Kbps)
G.711	88
G.729	32
GSM	37.2

ตัวอย่างการคำนวณของโปรแกรมในการหาจำนวนผู้ใช้งานพร้อมกัน อย่างเช่นเมื่อผู้ใช้งานต้องการใช้ CODEC แบบ G.711 ซึ่งมีแบนด์วิทด์รวม เท่ากับ 88 Kbps แล้วทำการ Start โปรแกรม ให้ทำการวัด โปรแกรมวัดค่าแบนด์วิดท์ออกมาได้เท่ากับ 2 Mbps ก็จะมีวิธีการคำนวณค่า Concurrent User ดังนี้

ค่า Concurrent Use = 2000 Kbps / 88 Kbps

= 22.72 Concurrent User

แต่โปรแกรมที่พัฒนานี้จะตัดเศษหลังจุดทศนิยมทิ้งไป โดยจะแสดงค่าที่เป็นจำนวนเต็มเท่านั้นคือจะแสดงแค่ 22 Concurrent Use สาเหตุที่ให้แสดงเฉพาะจำนวนเต็มโดยการปัดทศนิยมทิ้งก็เพราะว่า ถ้าเศษปัดทศนิยมขึ้นจะมีผลกับคุณภาพของเสียงเมื่อใช้งานเต็มจำนวน

เมื่อผู้ใช้งานต้องการใช้ CODEC แบบ G.729 ซึ่งมีแบนด์วิทด์รวม เท่ากับ 32 Kbps แล้วทำการ Start โปรแกรม ให้ทำการวัด โปรแกรมวัดค่าแบนด์วิดท์ออกมาได้เท่ากับ 2 Mbps ก็จะมีวิธีการคำนวณค่า Concurrent User ดังนี้

ค่า Concurrent Use = 2000 Kbps / 32 Kbps

= 62.5 Concurrent User

โปรแกรมจะแสดงค่าที่เป็นจำนวนเต็มเท่านั้นคือจะแสดงแค่ 62 Concurrent Use เท่านั้น

ตัวอย่างสุดท้ายเมื่อผู้ใช้งานต้องการใช้ CODEC แบบ GSM ซึ่งมีแบนด์วิทด์รวม เท่ากับ 37.2 Kbps แล้วทำการ Start โปรแกรม ให้ทำการวัด โปรแกรมวัดค่าแบนด์วิดท์ออกมาได้เท่ากับ 2 Mbps ก็จะมีวิธีการคำนวณค่า Concurrent User ดังนี้

ค่า Concurrent Use = 2000 Kbps / 37.2 Kbps

= 53.76 Concurrent User

โปรแกรมจะแสดงค่าที่เป็นจำนวนเต็มเท่านั้นคือจะแสดงแค่ 53 Concurrent Use เท่านั้น

เมื่อได้ทำการออกแบบทุกอย่างเรียบร้อย ไม่ว่าจะเป็นหน้าตาลักษณะการทำงานของโปรแกรมและการเขียนคำสั่ง ของในแต่ละ Object รวมถึงวิธีการคำนวณหาค่าต่าง ๆ ของโปรแกรมที่พัฒนาขึ้น ขั้นตอนต่อไปก็จะเป็นวัดหาความถูกต้องและวิธีทดสอบโปรแกรมที่ได้พัฒนาขึ้น โดยจะทำสร้างเครือข่ายจำรองอย่างง่าย ๆ เพื่อจะนำมาทดสอบความถูกต้องของโปรแกรมที่พัฒนาขึ้น ซึ่งเบื้องต้นซึ่งจะใช้ Router , Switch อย่างละ 1 ตัว และจะมี PC อีก 2 ตัว เป็นอย่างน้อง โดยจะมีลักษณะดังแสดงในรูปที่ 3.7

รูปที่ 3.7 ลักษณะของเครือข่ายที่ใช้ในการทดสอบ

จากรูปที่ 3.7 สามารถอธิบายวิธีการทำงานของโปรแกรมที่พัฒนาขึ้นโดยการจำลองเครือข่ายขึ้นมาเพื่อการทดสอบโดยจะให้ PC 1 เป็นคอมพิวเตอร์ที่จะให้โปรแกรมที่พัฒนาขึ้นทำงานเป็นเครื่องแม่ข่าย ส่วน PC 2 เป็นคอมพิวเตอร์ที่จะให้โปรแกรมที่พัฒนาขึ้นทำงานเป็นเครื่องลูกข่าย โดยจะต้องให้ เครื่องแม่ข่ายให้ทำงานก่อนแล้วจากนั้นก็ทำการใส่ข้อมูลที่จำเป็นบนเครื่องลูกข่าย แล้วทำการ Run โปรแกรมที่พัฒนาขึ้นก็จะทำการวัดค่า MOS หรือ Concurrent Use ระหว่างเครื่องที่เป็นแม่ข่าย กับเครื่องที่เป็นลูกข่าย ตามจุดประสงค์ของผู้ใช้งานต่อไป