

การพัฒนาระบบฝากข้อความเสียงผ่านเครือข่ายอินเทอร์เน็ต

DEVELOPMENT OF INTERNET VOICEMESSAGE SYSTEM



สมภพ ตริยะประเสริฐพร
สุกิจการ เรืองศิษย์
เอกอนันต์ แสงวัฒนาโรจน์



ปัญหาพิเศษนี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรวิทยาศาสตรบัณฑิต

ภาควิชาคณิตศาสตร์และวิทยาการคอมพิวเตอร์

คณะวิทยาศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2543

ปจ
ค. 7/ก
8543

เลขหม.....

เลขทะเบียน 39681

วัน, เดือน, ปี 19 มิ.ย. 2544

b.....
i.....

สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

DEVELOPMENT OF INTERNET VOICEMESSAGE SYSTEM



**A SPECIAL PROJECT SUBMITTED IN PARTIAL FULFILLMENT
OF THE REQUIRMENT FOR THE DEGREE OF BACHELOR OF SCIENCE
DEPARTMENT OF MATHEMATICS AND COMPUTER SCIENCES
FACULTY OF SCIENCE
KING MONGKUT'S INSTITUTE OF TECHNOLOGY LADKRABANG
ACADEMIC YEAR 2000**


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หัวข้อปัญหาพิเศษ การพัฒนาระบบฝากข้อความเสียงผ่านเครือข่ายอินเทอร์เน็ต
 DEVELOPMENT OF INTERNET VOICEMESSAGE SYSTEM

ชื่อนักศึกษา นายสมภพ ตรียะประเสริฐพร 40056090
 นายสุกิจการ เรืองศิษุ 40056095
 นายเอกอนันต์ แสงวัฒนาโรจน์ 40056123

ภาควิชา คณิตศาสตร์และวิทยาการคอมพิวเตอร์
 สาขาวิชา วิทยาการคอมพิวเตอร์
 อาจารย์ที่ปรึกษา ผู้ช่วยศาสตราจารย์ไพโรบลย์ พันธรักษ์พงษ์
 ผู้ช่วยศาสตราจารย์ธีรวัฒน์ ประกอบผล

ภาควิชาคณิตศาสตร์และวิทยาการคอมพิวเตอร์ คณะวิทยาศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง อนุมัติให้นับปัญหาพิเศษนี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรวิทยาศาสตรบัณฑิต สาขาวิทยาการคอมพิวเตอร์ ประจำปีการศึกษา 2543

	คณะกรรมการสอบ	ลายมือชื่อ
ประธานกรรมการ	อาจารย์วิสันต์ ตั้งวงษ์เจริญ	
กรรมการ	ผู้ช่วยศาสตราจารย์กฤษณา ไตรสุรัตน์	
กรรมการและ อาจารย์ที่ปรึกษา	ผู้ช่วยศาสตราจารย์ไพโรบลย์ พันธรักษ์พงษ์	
กรรมการและ อาจารย์ที่ปรึกษา	ผู้ช่วยศาสตราจารย์ธีรวัฒน์ ประกอบผล	



(ผู้ช่วยศาสตราจารย์ไพโรบลย์ พันธรักษ์พงษ์)
 หัวหน้าภาควิชาคณิตศาสตร์และวิทยาการคอมพิวเตอร์

กิตติ์ธิ์ของภาควิชาคณิตศาสตร์และวิทยาการคอมพิวเตอร์ คณะวิทยาศาสตร์
 สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
 ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หัวข้อปัญหาพิเศษ	การพัฒนาระบบฝากข้อความเสียงผ่านเครือข่ายอินเทอร์เน็ต	
ชื่อนักศึกษา	นายสมภพ ศรียะประเสริฐพร	40056090
	นายสุกิจการ เรืองศิษย์	40056095
	นายเอกอนันต์ แสงวัฒนาโรจน์	40056123
ปริญญา	วิทยาศาสตรบัณฑิต	
ภาควิชา	คณิตศาสตร์และวิทยาการคอมพิวเตอร์ คณะวิทยาศาสตร์	
สาขาวิชา	วิทยาการคอมพิวเตอร์	
ปีการศึกษา	2543	
อาจารย์ที่ปรึกษา	ผู้ช่วยศาสตราจารย์ไพโรบลย์ พันธรักษ์พงษ์	
	ผู้ช่วยศาสตราจารย์ธีรวัฒน์ ประกอบผล	

บทคัดย่อ

ปัจจุบันเครือข่ายอินเทอร์เน็ตได้เข้ามามีบทบาทสำคัญต่อการสื่อสารข้อมูล เพราะความสะดวก รวดเร็วและค่าใช้จ่ายที่น้อยกว่า ทำให้เครือข่ายอินเทอร์เน็ตมีความแพร่หลาย และเติบโตขยายโครงข่ายขึ้นอย่างรวดเร็ว แต่ด้วยข้อจำกัดของเครือข่ายอินเทอร์เน็ตที่เป็นระบบปิด ที่ผู้ใช้ต้องเชื่อมต่อกับเครือข่ายเท่านั้น จึงจะสามารถสื่อสารกันได้ ทำให้ข่าวสารข้อมูลที่จะได้รับ ต้องสูญเสียเวลาไปกับการรอคอย หากผู้รับไม่สามารถต่อเข้ามายังเครือข่ายได้

การพัฒนาระบบฝากข้อความเสียงผ่านเครือข่ายอินเทอร์เน็ต เป็นการนำเอาระบบสื่อสารพื้นฐาน คือ โทรศัพท์ มาเชื่อมโยงกับเครือข่ายอินเทอร์เน็ต ในรูปแบบของข้อความเสียง โดยผู้รับข้อความเสียง สามารถโทรศัพท์เข้ามาฟังข้อความที่ฝากไว้ได้ โดยไม่จำเป็นต้องต่อกับเครือข่ายอินเทอร์เน็ต อีกทั้งเป็นการพัฒนาระบบสื่อสารทางไกลจากตัวกลางที่เป็นระบบโทรศัพท์ มาเป็นเครือข่ายอินเทอร์เน็ต ซึ่งเสียค่าใช้จ่ายน้อยกว่ามาก การพัฒนาระบบฝากข้อความเสียงผ่านเครือข่ายอินเทอร์เน็ต ถือเป็นการพัฒนาเพื่อเพิ่มขีดความสามารถของเครือข่ายอินเทอร์เน็ตให้มากขึ้น และก่อให้เกิดประโยชน์ต่อผู้ใช้สูงสุด

Special Project Title	DEVELOPMENT OF INTERNET VOICEMESSAGE SYSTEM	
Students	Mr. Sompob Treeyaprasertporn	40056090
	Mr. Sukijkarn Ruangdit	40056095
	Mr. Eakanun Sangwatanaroj	40056123
Degree	Bachelor 's Degree of Science	
Department	Mathematics and Computer Sciences, Faculty of Science	
Programme	Computer Sciences	
Academic Year	2000	
Special Project Advisor	Assistant Professor Praiboon Pantaragphong	
	Assistant Professor Teerawat Prakobphon	

ABSTRACT

Nowadays, Internet plays an important role in communication. Due to convenience, high-speed and lower cost make popular on internet and are growing on the fast showing as increased network. However, Internet is the closing network which users are necessary to connect while communicate. As a result, incoming information will be waited and make disadvantage if receiver will not be on Internet.

Development of internet voicemail system takes the basic telephone-based system to combine with internet via voice messaging system. The receiver will call a phone to receive voice message without using internet network. Moreover, distant communication system is developed in this project from all telephone lines to Internet lines which has much advantages in lower cost. Totally, development of internet voicemail system stands as the development of internet capabilities and the most user benefits.

กิตติกรรมประกาศ

ในการทำปัญหาพิเศษเรื่อง การพัฒนาระบบฝากข้อความเสียงผ่านเครือข่ายอินเทอร์เน็ต ให้สามารถสำเร็จลุล่วงไปได้ด้วยดีนั้น คณะผู้จัดทำขอขอบพระคุณท่านอาจารย์ที่ปรึกษาทั้งสอง คือ ผศ.ไพโรบลย์ พันธรักษ์พงษ์ และ ผศ.ธีรวัฒน์ ประกอบผล เป็นอย่างสูงที่กรุณาให้คำแนะนำและเป็นที่ปรึกษาในการแก้ปัญหาต่างๆ ตลอดจนความช่วยเหลือในเรื่องอุปกรณ์ที่ใช้กับปัญหาพิเศษนี้ รวมทั้งเป็นผู้ตรวจสอบความถูกต้องของเอกสารปัญหาพิเศษฉบับนี้

นอกจากนี้คณะผู้จัดทำต้องขอขอบพระคุณ บิดา มารดา ที่ได้ให้ความสนับสนุนทางด้าน กำลังใจและทุนทรัพย์ จนการทำปัญหาพิเศษนี้สำเร็จลุล่วงไปด้วยดี ขอขอบคุณ คุณกัมปนาท เมฆอำมพุก ที่ให้ความช่วยเหลือในด้าน โปรแกรมและตัวอย่างข้อมูล และทุกท่านที่ให้ความช่วยเหลือในการทำปัญหาพิเศษไว้ ณ โอกาสนี้ด้วย

คณะผู้จัดทำ

มีนาคม 2544

สารบัญ

	หน้า
บทคัดย่อภาษาไทย.....	I
บทคัดย่อภาษาอังกฤษ.....	II
กิตติกรรมประกาศ.....	III
สารบัญ.....	IV
สารบัญตาราง.....	VII
สารบัญรูป.....	VIII
บทที่ 1 บทนำ.....	1
1.1 ความเป็นมาและความสำคัญของปัญหา.....	1
1.2 ความมุ่งหมายและวัตถุประสงค์ของการศึกษา.....	1
1.3 สมมติฐานของการศึกษา.....	1
1.4 ขอบเขตของการศึกษา.....	2
1.5 ขั้นตอนการศึกษา.....	2
1.6 ประโยชน์ที่คาดว่าจะได้รับ.....	2
1.7 ข้อตกลงเบื้องต้น.....	2
บทที่ 2 การทำงานของระบบและทฤษฎีที่เกี่ยวข้อง.....	3
2.1 โครงสร้างของระบบ.....	3
2.2 การทำงานของระบบ.....	4
2.2.1 การทำงานของระบบฝากข้อความเสียงในระบบเครือข่ายอินเทอร์เน็ต.....	4
2.2.2 การทำงานของระบบโทรศัพท์ภายในระบบ.....	9
2.3 ทฤษฎีที่เกี่ยวข้องในการทำงานของระบบ.....	12
2.3.1 โพรโทคอลส่งผ่านข้อมูล(File Transfer Protocol).....	12
2.3.2 โครงสร้างของ FTP.....	12
2.3.3 ประวัติความเป็นมาของโพรโทคอล TCP/IP.....	15
2.3.4 การติดต่อสื่อสารระหว่างคอมพิวเตอร์ที่ใช้โพรโทคอล TCP/IP.....	15
2.3.5 ความหมายของ Windows Sockets.....	17
2.3.6 เครื่องมือที่ใช้ในการพัฒนาโปรแกรม FTP (File Transfer Protocol).....	19

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

	หน้า
2.3.7 โทรศัพท์.....	24
2.3.8 การทำงานของ Modem.....	26
2.3.9 การ Modulate และ Demodulate.....	27
2.3.10 การทำงานของ Voice Modem.....	28
2.3.11 การสุ่มสัญญาณ.....	29
2.3.12 การเทียบระดับแรงดัน.....	32
2.3.13 วงจรถอดรหัสหมายเลข DTMF.....	34
2.3.14 วงจรกรองความถี่และวงจรตรวจจับ (filtering and detector).....	35
2.3.15 โครงสร้างการติดต่อกับ Modem.....	36
2.3.16 เครื่องมือที่ใช้พัฒนาด้านการติดต่อสื่อสารกับระบบโทรศัพท์.....	38
2.3.17 การเชื่อมต่อ Application กับ TAPI.....	38
2.3.18 การทำงานของ Multimedia API.....	42
2.3.19 ดักฟังการกดปุ่ม โทรศัพท์.....	43
2.3.20 การใช้งาน Microsoft Agent.....	44
2.3.21 ความหมายของ Data Access Object (DAO).....	45
2.3.22 การแปลงไฟล์ Wav เป็นไฟล์ MP3.....	46
2.3.23 การจัดระบบไฟล์ด้วย File System Object.....	47
บทที่ 3 การศึกษาและพัฒนา.....	50
3.1 โครงสร้างระบบ.....	50
3.1.1 โครงสร้างส่วนไคลเอนท์ (DevIVM Client).....	50
3.1.2 โครงสร้างส่วนเซิร์ฟเวอร์ (DevIVM Server).....	51
3.1.3 โครงสร้างการทำงาน.....	51
3.1.4 โครงสร้างระบบสื่อสารทางไกล.....	54
3.2 การศึกษาเพื่อการพัฒนา.....	55
3.2.1 การศึกษาเพื่อการพัฒนาส่วนไคลเอนท์.....	55
3.2.2 การศึกษาเพื่อการพัฒนาส่วนเซิร์ฟเวอร์.....	56
3.3 ข้อมูลและตัวอย่างที่ใช้ในการพัฒนา.....	57
3.4 เครื่องมือและวิธีการในการพัฒนา.....	57
3.5 การทำงานของโปรแกรมที่พัฒนา.....	57

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 4 ประเมินและอภิปรายผล.....	64
4.1 ลักษณะของโปรแกรมระบบฝากข้อความเสี่ยงผ่านเครือข่ายอินเทอร์เน็ต.....	
(DevIVM).....	64
4.2 ข้อกำหนดของการประเมิน.....	65
4.3 การประเมิน.....	65
4.4 การอภิปรายผล.....	65
บทที่ 5 สรุปผลการดำเนินงานและข้อเสนอแนะ.....	66
ภาคผนวก ก คู่มือการติดตั้ง โปรแกรมระบบฝากข้อความเสี่ยงผ่านเครือข่ายอินเทอร์เน็ต.....	
ส่วนไคลเอนท์.....	68
ภาคผนวก ข คู่มือการติดตั้ง โปรแกรมระบบฝากข้อความเสี่ยงผ่านเครือข่ายอินเทอร์เน็ต.....	
ส่วนเซิร์ฟเวอร์.....	74
ภาคผนวก ค คู่มือการใช้งาน โปรแกรมระบบฝากข้อความเสี่ยงผ่านเครือข่ายอินเทอร์เน็ต.....	
ส่วนไคลเอนท์.....	81
ภาคผนวก ง คู่มือการใช้งาน โปรแกรมระบบฝากข้อความเสี่ยงผ่านเครือข่ายอินเทอร์เน็ต.....	
ส่วนเซิร์ฟเวอร์.....	88
บรรณานุกรม.....	93

สารบัญตาราง

ตารางที่	หน้า
2.1 มาตรฐานของ Port Number ของ TCP.....	16
2.2 มาตรฐานของ Port Number ของ UDP.....	16
2.3 Property ของ Internet Transfer Protocol Control.....	19
2.4 Method ของ Internet Transfer Protocol Control.....	20
2.5 คำสั่งที่ใช้กับ Method Execute.....	21
2.6 ตัวอย่างการใช้งานของคำสั่งที่ใช้กับ Method Execute.....	22
2.7 ค่าพารามิเตอร์ของ Event StateChanged.....	24
2.8 แสดงขั้นตอนในการเรียกใช้งาน TAPI จาก Application.....	38
2.9 แสดง Handle 3 ชนิดของ TAPI.....	39
2.10 แสดง TAPI Message ที่เกี่ยวข้องกับระบบ.....	40
2.11 แสดงหน้าที่ของฟิลด์ข้อมูลพื้นฐานทั้ง 3 ฟิลด์.....	41
2.12 แสดง Device Class มาตรฐานของ Multimedia API.....	43
2.13 ตารางออบเจกต์ของ File System Object.....	47
2.14 คุณสมบัติของออบเจกต์ Drive.....	48
2.15 คุณสมบัติของออบเจกต์ Folder.....	48
2.16 เมตดอดของออบเจกต์ File.....	48

สารบัญรูป

รูปที่	หน้า
2.1 แสดงให้เห็นถึงการทำงานโดยรวมของระบบ.....	3
2.2 แสดงแผนผังการทำงานในส่วนการฝากข้อความเสียงผ่านเครือข่ายอินเทอร์เน็ต.....	4
2.3 แสดงรายละเอียดย่อยของสเตทเริ่ม.....	6
2.4 แสดงรายละเอียดย่อยของสเตท ระบุ ID Number ปลายทางและบันทึกเสียง.....	7
2.5 แสดงรายละเอียดย่อยของสเตทค้นหา Remote Server.....	8
2.6 แสดงแผนผังการทำงานในส่วนของระบบโทรศัพท์.....	10
2.7 แสดงการทำงานของ FTP.....	12
2.8 ใ้ไดอะแกรมการรับส่งคำสั่งโปรโตคอลส่งผ่านข้อมูลและการตอบกลับ.....	14
2.9 แสดงถึงการติดต่อสื่อสารระหว่างคอมพิวเตอร์ที่ใช้โปรโตคอล TCP/IP.....	15
2.10 แสดงหลักการการทำงานแบบไคลเอนท์/เซิร์ฟเวอร์โดยใช้ Socket.....	18
2.11 แสดงระบบโทรศัพท์แบบหมุน.....	25
2.12 แสดงแป้นกดหมายเลขและค่าความถี่ในแนวนอนและแนวตั้งของหมายเลขนั้นๆ.....	26
2.13 แสดงขั้นตอนการเปลี่ยนรูปแบบการส่งข้อมูลผ่าน โมเด็ม.....	27
2.14 แสดงพื้นฐานกระบวนการทำงานของการส่งข้อมูลแบบ PCM.....	29
2.15 แสดงการสุ่มสัญญาณ.....	30
2.16 แสดงสเปกตรัมของสัญญาณ.....	31
2.17 แสดงการกำหนดระดับตัดสินของระบบทวนสัญญาณดิจิทัลซึ่งกำหนดย่าน แรงดันตัดสินไว้ 2 ย่าน.....	33
2.18 ความถี่ของระบบ DTMF.....	34
2.19 ผลตอบสนองความถี่ของวงจรกรองความถี่.....	35
2.20 บล็อกไดอะแกรมของวงจรถอดรหัสหมายเลขแบบ DTMF.....	36
2.21 แสดงโครงสร้างการติดต่อ โมเด็ม.....	37
3.1 โครงสร้างระบบฝากข้อความเสียงผ่านเครือข่ายอินเทอร์เน็ต.....	50
3.2 โครงสร้างส่วนไคลเอนท์ (DevIVM Client).....	50
3.3 โครงสร้างส่วนเซิร์ฟเวอร์ (DevIVM Server).....	51
3.4 โครงสร้างการทำงานของส่วนไคลเอนท์.....	51
3.5 โครงสร้างการทำงานของส่วนไคลเอนท์เชื่อมโยงผ่านเครือข่ายอินเทอร์เน็ต..... ไปยังส่วนเซิร์ฟเวอร์.....	52

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่	หน้า
3.6 โครงสร้างการทำงานของส่วนเซิร์ฟเวอร์ที่เชื่อมต่อกับระบบโทรศัพท์ภายนอก.....	52
3.7 Trace Diagram แสดงการทำงานของระบบฝากข้อความเสียงผ่านเครือข่ายอินเทอร์เน็ต.....	53
3.8 โครงสร้างการพัฒนาระบบเป็นแบบสื่อสารทางไกลผ่านเครือข่ายอินเทอร์เน็ต.....	54
3.9 Context Diagram ของระบบฝากข้อความเสียงผ่านเครือข่ายอินเทอร์เน็ต.....	58
3.10 (1) Flow Chart ส่วนที่ 1 ของโปรแกรมระบบฝากข้อความเสียงผ่านเครือข่าย..... อินเทอร์เน็ตส่วนไคลเอนท์ (DevIVM Client).....	60
3.10 (2) Flow Chart ส่วนที่ 2 ของโปรแกรมระบบฝากข้อความเสียงผ่านเครือข่าย..... อินเทอร์เน็ตส่วนไคลเอนท์ (DevIVM Client).....	61
3.11 (1) Flow Chart ส่วนที่ 1 ของโปรแกรมระบบฝากข้อความเสียงผ่านเครือข่าย..... อินเทอร์เน็ตส่วนเซิร์ฟเวอร์ (DevIVM Server)	62
3.11 (2) Flow Chart ส่วนที่ 1 ของโปรแกรมระบบฝากข้อความเสียงผ่านเครือข่าย..... อินเทอร์เน็ตส่วนเซิร์ฟเวอร์ (DevIVM Server).....	63
ก(1) ไอคอน Setup.....	69
ก(2) แสดงหน้าจอติดตั้งโปรแกรมส่วนไคลเอนท์ 1.....	69
ก(3) แสดงหน้าจอติดตั้งโปรแกรมส่วนไคลเอนท์ 2.....	70
ก(4) แสดงหน้าจอติดตั้งโปรแกรมส่วนไคลเอนท์ 3.....	71
ก(5) แสดงหน้าจอติดตั้งโปรแกรมส่วนไคลเอนท์ 4.....	71
ก(6) แสดงหน้าจอติดตั้งโปรแกรมส่วนไคลเอนท์ 5.....	72
ก(7) แสดงหน้าจอติดตั้งโปรแกรมส่วนไคลเอนท์ 6.....	72
ก(8) แสดงหน้าจอติดตั้งโปรแกรมส่วนไคลเอนท์ 7.....	73
ก(9) แสดงหน้าจอเรียกใช้โปรแกรมส่วนไคลเอนท์ผ่าน Windows.....	73
ข(1) แสดงไฟล์ติดตั้งโปรแกรมส่วนเซิร์ฟเวอร์.....	75
ข(2) เริ่มการทำงานติดตั้งโปรแกรมส่วนเซิร์ฟเวอร์.....	76
ข(3) การทำงานติดตั้งโปรแกรมส่วนเซิร์ฟเวอร์.....	77
ข(4) ข้อตกลงในการติดตั้ง (License Agreement)	78
ข(5) หน้าจอ User Information.....	78
ข(6) หน้าจอแสดงการติดตั้งบน Storage.....	79
ข(7) โปรแกรมส่วนเซิร์ฟเวอร์เสร็จสิ้นการติดตั้ง.....	79
ข(8) โปรแกรมส่วนเซิร์ฟเวอร์เรียกจาก Start Program ใน Windows.....	80
ค(1) ไอคอน Insert Password.....	82

รูปที่	หน้า
ค(2) ไดอะล็อก Main Program.....	82
ค(3) ไดอะล็อก Quick Connect.....	83
ค(4) ไดอะล็อกเมนู Option > Setting.....	84
ค(5) แสดงการ Setting Timeout.....	84
ค(6) แสดงการเลือกเมนูเปลี่ยน Password.....	85
ค(7) หน้าจอ Set Password.....	86
ค(8) หน้าจอเมนู Help.....	86
ค(9) Mr. Genie ตัวการ์ตูนแสดงความช่วยเหลือ.....	87
ค(10) หน้าจอ Tip of the Day.....	87
ง(1) หน้าจอแรกโปรแกรมระบบฝากข้อความเสียงผ่านเครือข่ายอินเทอร์เน็ตส่วน เซิร์ฟเวอร์ (DevIVM Telephony Server).....	89
ง(2) หน้าจอเช็ค Password เพื่อเข้าสู่ระบบ Configuration.....	89
ง(3) หน้าจอเมนูหลักและส่วนแสดงสถานะระบบ.....	90
ง(4) หน้าจอแสดงการแก้ไขจำนวนครั้งที่ตั้ง.....	90
ง(5) หน้าจอแสดงการแก้ไขความยาวของเบอร์โทรศัพท์.....	91
ง(6) แสดงหน้าจอการเปลี่ยนระดับความดัง-เบาของเสียง โมเด็ม.....	91
ง(7) แสดงหน้าจอข้อมูลเกี่ยวกับโปรแกรมส่วนเซิร์ฟเวอร์.....	92

บทที่ 1

บทนำ

1.1 ความเป็นมาและความสำคัญของปัญหา

ปัจจุบันการสื่อสารผ่านเครือข่ายอินเทอร์เน็ตมีบทบาทสำคัญในชีวิตประจำวัน เนื่องจากประโยชน์ที่มีอยู่อย่างมากมาย ทั้งสามารถส่งผ่านข้อมูลเสียงและภาพไปมาได้สะดวก แต่อินเทอร์เน็ตก็ยังถือว่าเป็นระบบปิด เนื่องจากผู้ใช้ต้องต่อเข้ามาในระบบเครือข่ายถึงจะใช้งานได้ ดังนั้น การนำระบบสื่อสารพื้นฐานที่มีอยู่เดิม คือ โทรศัพท์ ให้สามารถเชื่อมโยงกับเครือข่ายอินเทอร์เน็ตได้ โดยไม่จำเป็นต้องต่อเข้ามาในระบบเครือข่ายตลอดเวลา จึงเป็นแนวความคิดหนึ่งที่จะทำให้ระบบเครือข่ายอินเทอร์เน็ตนั้น ถดช่องว่างระหว่างการสื่อสารลงและเป็นระบบที่เปิดกว้างมากกว่าเดิม

การพัฒนาระบบฝากข้อความทางเสียงบนเครือข่ายอินเทอร์เน็ต เป็นการเพิ่มขีดความสามารถให้ผู้ที่เราติดต่อด้วย รับรู้ข่าวสารทางเสียงจากผู้ส่ง โดยไม่จำเป็นต้องอยู่หรือติดต่อกับเครือข่ายอินเทอร์เน็ตได้ ซึ่งจากจุดนี้ทำให้เครือข่ายอินเทอร์เน็ตเป็นระบบที่เปิดสู่ระบบสื่อสารอื่นๆ ได้อย่างแท้จริง

1.2 ความมุ่งหมายและวัตถุประสงค์ของการศึกษา

1. เพื่อพัฒนาระบบเครือข่ายอินเทอร์เน็ตให้สามารถเชื่อมโยงกับระบบสื่อสารอื่นๆ ได้ โดยสามารถเชื่อมโยงกับระบบโทรศัพท์พื้นฐานที่มีอยู่เดิม ให้สามารถฝากข้อความเสียงผ่านเครือข่ายอินเทอร์เน็ตได้
2. เพื่อเพิ่มประสิทธิภาพ และขีดความสามารถในการใช้งานบนเครือข่ายอินเทอร์เน็ตได้มากยิ่งขึ้น

1.3 สมมุติฐานของการศึกษา

สร้างระบบฝากข้อความเสียงผ่านเครือข่ายอินเทอร์เน็ต และสามารถเชื่อมโยงให้ผู้รับข้อความ รับรู้ข่าวสารที่ส่งมาโดยใช้โทรศัพท์พื้นฐานได้

1.4 ขอบเขตของการศึกษา

1. พัฒนาระบบฝากข้อความทางเสียงจากเครื่องคอมพิวเตอร์ไคลเอนท์
2. พัฒนาระบบกระจายข้อมูล คือ ข้อความเสียงที่ฝากไว้จากไคลเอนท์ ไปยังเครื่องคอมพิวเตอร์ (เทเลโฟนนี่เซิร์ฟเวอร์) ที่อยู่ใกล้ผู้รับข้อความที่สุด
3. พัฒนาระบบเชื่อมโยงกับโทรศัพท์พื้นฐาน ให้สามารถรับข้อความเสียงที่ฝากไว้ได้

1.5 ขั้นตอนการศึกษา

1. ศึกษาถึงระบบส่วนต่างๆ ได้แก่ การฝากข้อความเสียงทางอินเทอร์เน็ต การกระจายข้อมูลจากคอมพิวเตอร์ไปยังคอมพิวเตอร์ผ่านเครือข่ายอินเทอร์เน็ต และการเชื่อมโยงกับระบบสื่อสารโทรศัพท์พื้นฐาน
2. วิเคราะห์ข้อมูลที่ได้ศึกษาและวางแผน
3. ออกแบบระบบงาน
4. การเขียนโปรแกรมคอมพิวเตอร์ เพื่อพัฒนาระบบให้สามารถใช้งาน
5. การทดสอบระบบและปรับปรุงแก้ไข
6. การจัดทำเอกสารปัญหาพิเศษ

1.6 ประโยชน์ที่คาดว่าจะได้รับ

1. เพิ่มประสิทธิภาพในการใช้งานบนเครือข่ายอินเทอร์เน็ตได้มากขึ้น
2. ลดช่องว่างในการติดต่อระหว่างระบบ โทรศัพท์พื้นฐานกับเครือข่ายอินเทอร์เน็ต
3. นำระบบสื่อสารที่มีอยู่แล้วมาพัฒนาให้เกิดประโยชน์สูงสุด
4. สามารถนำระบบไปใช้งานในด้านต่างๆ ให้เกิดประโยชน์ เช่น การสื่อสารทางไกล เป็นต้น

1.7 ข้อตกลงเบื้องต้น

1. ทำงานภายใต้ระบบปฏิบัติการ Windows 95/98 หรือ Windows NT
2. ใช้ภาษาคอมพิวเตอร์ คือ Visual Basic 6.0 และ Visual C++ เป็นเครื่องมือในการพัฒนาระบบ
3. ทำงานบนเครื่องคอมพิวเตอร์ส่วนบุคคล (PC)
4. มีวอร์ชโมเด็ม อย่างน้อย 1 เครื่อง
5. สายโทรศัพท์สำหรับติดต่อสื่อสารอย่างน้อย 1 สาย
6. อุปกรณ์มัลติมีเดีย สำหรับอัดเสียงและฟังเสียง เช่น ไมค์, ลำโพง

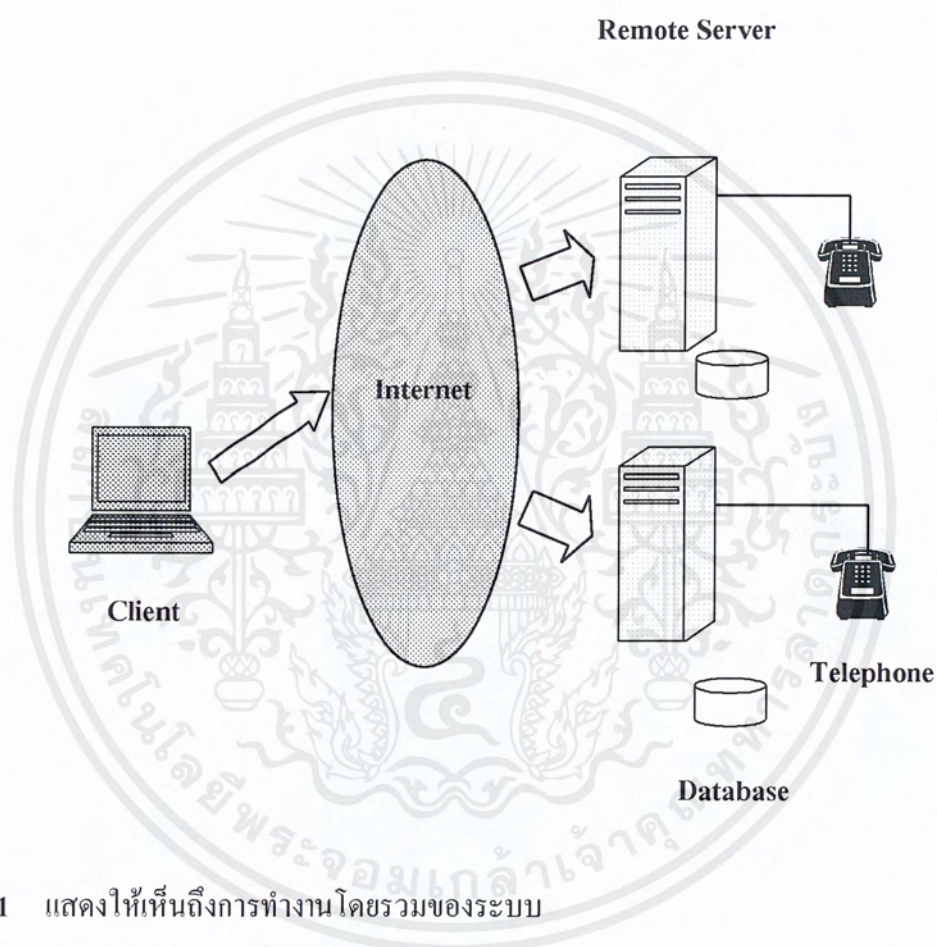
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 2

การทำงานของระบบและทฤษฎีที่เกี่ยวข้อง

2.1 โครงสร้างของระบบ

ระบบฝากข้อความเสียงผ่านเครือข่ายอินเทอร์เน็ตมีโครงสร้างการทำงานโดยรวมทั้งระบบตามแผนภาพ ดังรูปต่อไปนี้



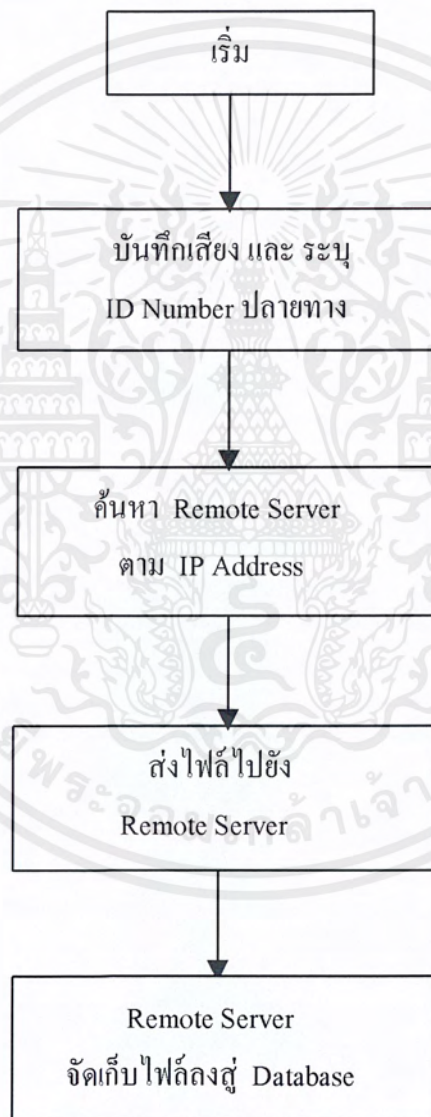
รูปที่ 2.1 แสดงให้เห็นถึงการทำงานโดยรวมของระบบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.2 การทำงานของระบบ

การทำงานของระบบพัฒนาการฝากข้อความเสียงผ่านเครือข่ายอินเทอร์เน็ต แบ่งได้เป็นสองส่วนคือ การทำงานในส่วนของระบบเครือข่ายอินเทอร์เน็ต และ การทำงานเชื่อมโยงระหว่างเครือข่ายอินเทอร์เน็ตมายังโทรศัพท์พื้นฐาน

2.2.1 การทำงานของระบบฝากข้อความเสียงในระบบเครือข่ายอินเทอร์เน็ต มีการทำงานดังแผนภาพดังนี้



รูปที่ 2.2 แสดงแผนผังการทำงานในส่วนการฝากข้อความเสียงผ่านเครือข่ายอินเทอร์เน็ต

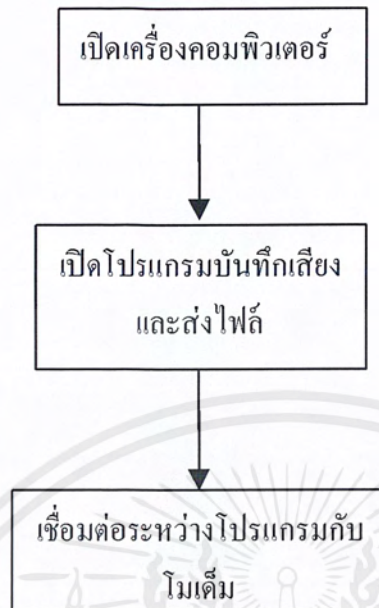
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากแผนภาพสามารถอธิบายได้ดังนี้

เริ่ม	เริ่มการทำงานของโปรแกรม
บันทึกเสียง และ ระบุ ID Number ปลายทาง	ผู้ใช้จะทำการบันทึกเสียงที่ต้องการจะส่ง ไปให้ ผู้รับปลายทาง และระบุ ID Number ของผู้รับ ปลายทาง
ส่งไฟล์และข้อมูล ID Number ไปยัง Remote Server	ทำการ upload ข้อมูลทั้งหมดไปยัง Remote Server ซึ่งข้อมูลทั้งหมดประกอบไปด้วยไฟล์ เสียงชนิดเวฟ (Wave File) และ ID Number
ค้นหา Remote Server ตาม IP Address	ทำการค้นหา Remote Server โดยใช้ IP address ซึ่งแปลงได้มาจาก ID Number เป็นตัวระบุ Remote Server
ส่งไฟล์ไปยัง Remote Server	ทำการส่งข้อมูลทั้งหมดไปยัง Remote Server ซึ่งข้อมูลทั้งหมดประกอบไปด้วยไฟล์ เสียงชนิดเวฟ (Wave File) และ ID Number
Remote Server จัดเก็บไฟล์ลงสู่ Database	ทำการจัดเก็บข้อมูลที่ได้มาทั้งหมดลงสู่ ฐานข้อมูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รายละเอียดย่อของสตาเป็นดังนี้
 สตา เริ่ม :



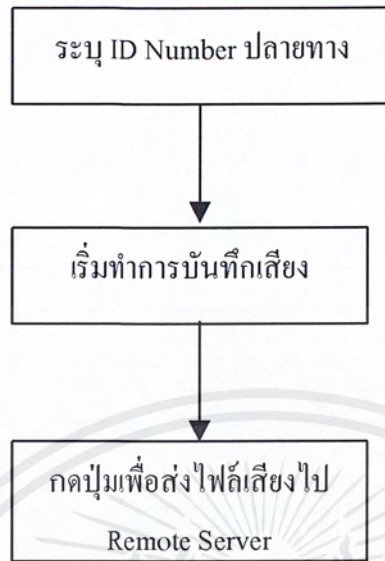
รูปที่ 2.3 แสดงรายละเอียดย่อของสตาเริ่ม

จากแผนภาพสามารถอธิบายได้ดังนี้

เปิดเครื่องคอมพิวเตอร์	ทำการเปิดเครื่องคอมพิวเตอร์ซึ่งติดตั้งโมเด็มเข้ากับเครื่องไว้เรียบร้อยแล้วพร้อมใช้งาน
เปิดโปรแกรมบันทึกเสียงและส่งไฟล์	ทำการเปิดโปรแกรมบันทึกเสียงและส่งไฟล์ให้พร้อมใช้งาน
เชื่อมต่อระหว่างโปรแกรมกับโมเด็ม	โปรแกรมจะทำการตรวจเช็คโมเด็มให้พร้อมใช้งานกับโปรแกรม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สเตท ระบุ ID Number ปลายทางและบันทึกเสียง :



รูปที่ 2.4 แสดงรายละเอียดย่อยของสเตท ระบุ ID Number ปลายทางและบันทึกเสียง

แผนภาพสามารถอธิบายได้ดังนี้

ระบุ ID Number ปลายทาง

ระบุ ID Number สำหรับผู้รับปลายทางไฟล์เสียงให้
แก่โปรแกรม

เริ่มทำการบันทึกเสียง

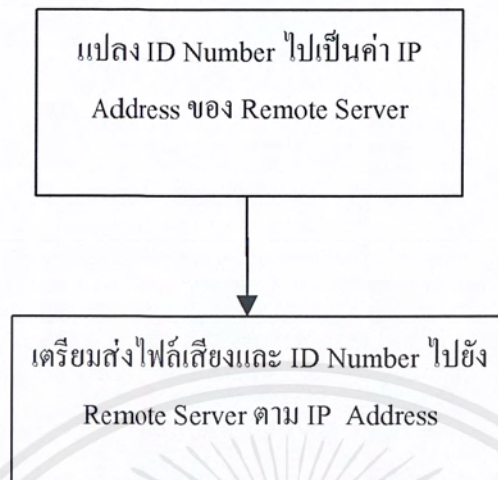
ผู้ส่งกดปุ่มเริ่มการบันทึกเสียงจากอุปกรณ์รับเสียง

กดปุ่มเพื่อส่งไฟล์เสียงไป Remote Server

เมื่อสิ้นสุดการบันทึกเสียง และผู้ส่งกดปุ่มส่งไฟล์เสียงออกไป ไฟล์เสียงจะถูกส่งไปยัง Remote Server โดยใช้ IP Address ที่ Set ไว้สำหรับโปรแกรมกับ Server ตัวนั้นๆ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สเตท ค้นหา Remote Server



รูปที่ 2.5 แสดงรายละเอียดย่อยของสเตทค้นหา Remote Server

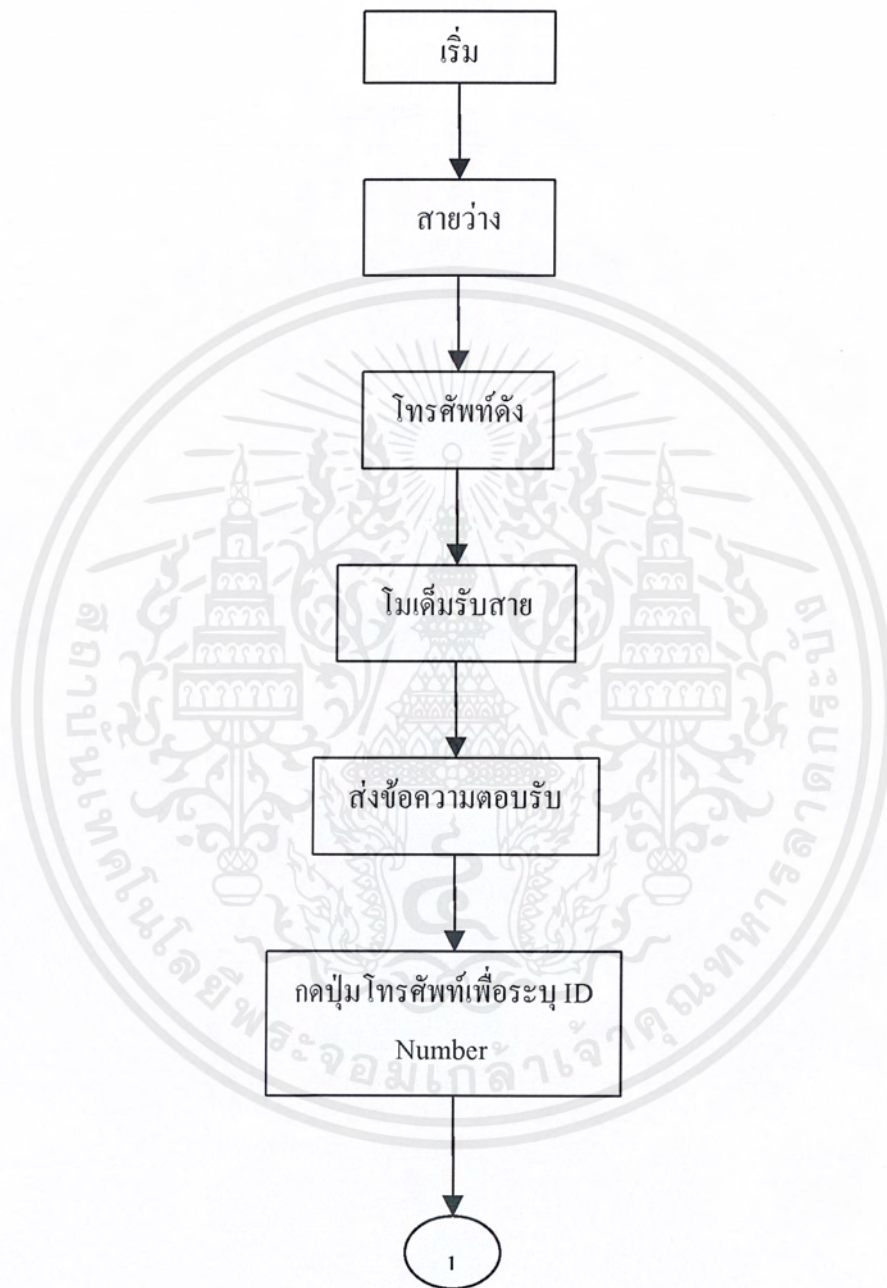
แผนภาพสามารถอธิบายได้ดังนี้

<p>แปลง ID Number ไปเป็นค่า IP Address ของ Remote Server</p>	<p>ID Number ของผู้รับที่ผู้ส่งได้ระบุจะเป็นข้อมูลสำหรับการติดต่อกับ Remote Server เพื่อกำหนด IP Address สำหรับ Remote Server แต่ละตัว</p>
<p>เตรียมส่งไฟล์เสียงและ ID Number ไปยัง Remote Server ตาม IP Address</p>	<p>หลังจากกำหนด IP Address ของ Remote Server แล้ว ก็จะเตรียมการส่งไฟล์เสียงตามที่ Locate ไว้</p>

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.2.2 การทำงานของระบบโทรศัพท์ภายในระบบ

ระบบฝากข้อความเสียงผ่านเครือข่ายอินเทอร์เน็ตมีโครงสร้างการทำงานในส่วนของระบบโทรศัพท์ ดังแผนภาพดังนี้



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.6 แสดงแผนผังการทำงานในส่วนของบริษัท

แผนภาพสามารถอธิบายได้ดังนี้

เริ่ม

เริ่มการทำงานของโปรแกรม และ Voice modem ต่อเข้ากับ Server แล้ว อย่างเรียบร้อย

สายว่าง

เป็นสแตทที่ Voice Modem อยู่ในสถานะว่างและรอให้ผู้สอบถามโทรศัพท์เข้ามาในระบบ

โทรศัพท์ดั่ง	มีสัญญาณ โทรศัพท์เข้าดั่งขึ้นที่ Voice Modem
โมเด็มรับสาย	Voice Modem รับสายโทรศัพท์หลังจาก สัญญาณการ โทรศัพท์เข้าดั่งขึ้น
ส่งข้อความตอบรับ	Voice Modem ส่งข้อความตอบรับออกไปตามสายโทรศัพท์ และรอการกด ID
กดปุ่มโทรศัพท์เพื่อระบุค่า ID Number	เป็นสเตรอร์รับสัญญาณปุ่มกดจากผู้สอบถาม
ค้นหาไฟล์เสียงตาม ID Number	Server จะค้นหาไฟล์เสียงตาม ID Number ที่กดปุ่มไว้ ถ้าไม่มีจะเล่นข้อความเสียงบอกตอบรับ และ จบการติดต่อ
เล่นไฟล์เสียง	เล่นไฟล์เสียง(Wav Files) ผ่าน Voice Modem ไปยังสายโทรศัพท์
จบการติดต่อ	โปรแกรมวางสายจากผู้สอบถาม

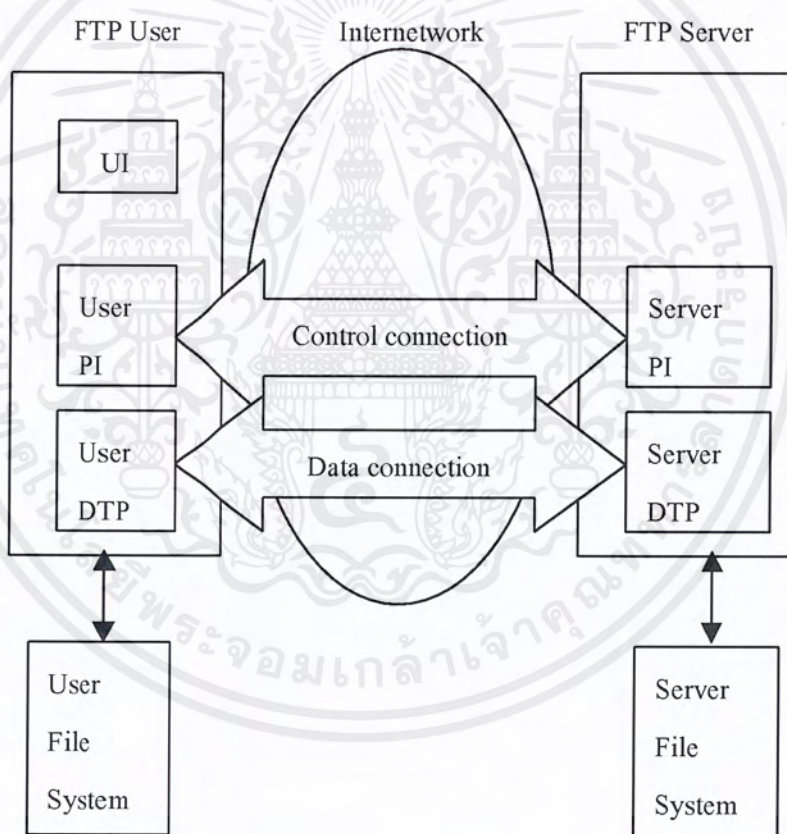
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.3 ทฤษฎีที่เกี่ยวข้องในการทำงานของระบบ

2.3.1 โพรโทคอลส่งผ่านข้อมูล(File Transfer Protocol)

เป็นโพรโทคอลที่ใช้ในการรับส่งไฟล์ระหว่างคอมพิวเตอร์สองเครื่องไคลเอนท์กับเซิร์ฟเวอร์ โดยอาศัยโพรโทคอล TCP/IP โพรโทคอลส่งผ่านข้อมูลแตกต่างจากโพรโทคอลแอปพลิเคชันส่วนใหญ่ตรงที่โพรโทคอลส่งผ่านข้อมูลจะใช้โพรโทคอล TCP สองตัวแยกจากกันในการเชื่อมต่อระหว่างโพรโทคอลส่งผ่านข้อมูลฝั่งไคลเอนท์ และโพรโทคอลส่งผ่านข้อมูลฝั่งเซิร์ฟเวอร์ โดยโพรโทคอล TCP แรกที่ใช้เมื่อเกิดการเชื่อมต่อครั้งแรกเป็นการเชื่อมต่อการควบคุม (Control connection) ใช้สำหรับเซิร์ฟเวอร์ในการส่งการตอบรับ ส่วนโพรโทคอล TCP ตัวที่สองเป็นการเชื่อมต่อข้อมูล (Data connection) ใช้เพื่อการรับส่งข้อมูล

2.3.2 โครงสร้างของ FTP



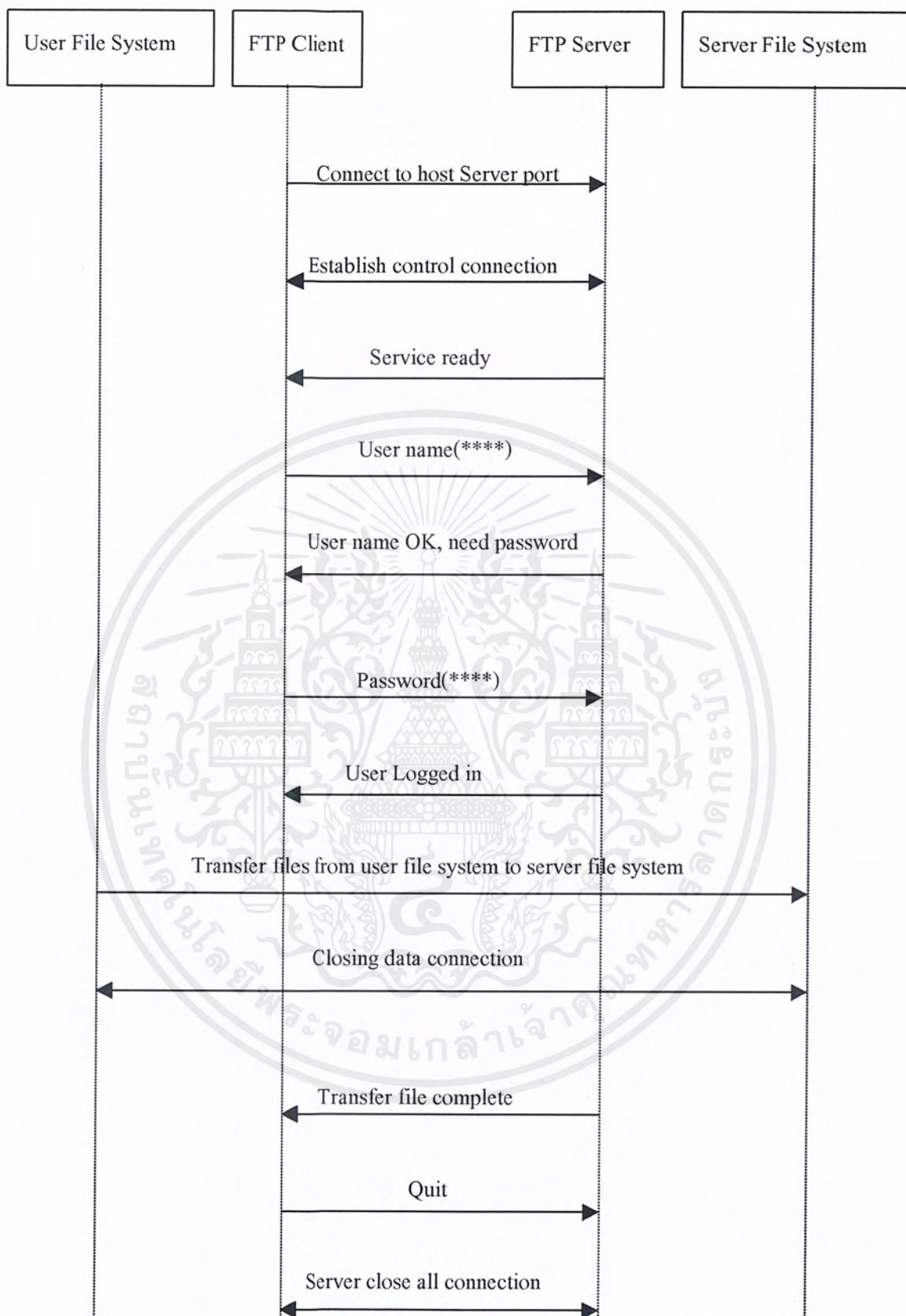
รูปที่ 2.7 แสดงการทำงานของ FTP

ส่วนของโพรโทคอลส่งผ่านข้อมูลฝั่งไคลเอนท์ ประกอบไปด้วย ส่วนติดต่อกับผู้ใช้(User Interface) ส่วนที่สองคือ User Protocol Interpreter (PI) ส่วนที่สามคือ User Data Transfer Process (DTP) ส่วนของโพรโทคอลฝั่งเซิร์ฟเวอร์ประกอบด้วยสองส่วนคือ Server Protocol Interpreter และ User Data Transfer Process

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เมื่อเริ่มทำงานโพรโตคอลส่งผ่านข้อมูลฝั่งไคลเอนท์จะรับชื่อ และตำแหน่ง IP ของโพรโตคอลส่งผ่านข้อมูลฝั่งเซิร์ฟเวอร์จากผู้ใช้ แล้ว PI ฝั่งไคลเอนท์จะสร้างการเชื่อมต่อโดยใช้โพรโตคอล TCP ไปยังพอร์ตของโพรโตคอลส่งผ่านข้อมูลฝั่งเซิร์ฟเวอร์ ซึ่งเชื่อมต่อกับ PI ฝั่งเซิร์ฟเวอร์เป็นชนิดการเชื่อมต่อการควบคุม ในส่วนของโพรโตคอลส่งผ่านข้อมูลฝั่งเซิร์ฟเวอร์ เมื่อเริ่มทำงานเปิดพอร์ต TCP พอร์ต 21 ไว้เพื่อรอการเชื่อมต่อควบคุมจาก PI ผู้ใช้ ซึ่งจะส่งคำสั่งมา และ PI เซิร์ฟเวอร์จะส่งคำตอบรับกลับมาให้ผู้ใช้งาน สถานะต่าง ๆ จะถูกแสดงออกมาทาง UI

ในการส่งผ่านข้อมูลจะทำโดยโปรแกรมรับส่งไฟล์ข้อมูลฝั่งไคลเอนท์ จะเปิดพอร์ตชั่วคราว (Local port) และทำการผ่านค่า IP address และหมายเลขพอร์ต TCP ไปให้โปรแกรมรับส่งข้อมูลฝั่งเซิร์ฟเวอร์ และใช้พอร์ตนี้ในการรับส่งผ่านข้อมูล ดังนั้น ก่อนที่ข้อมูลจะถูกส่งผ่านระหว่างโปรแกรมรับส่งข้อมูลฝั่งไคลเอนท์/เซิร์ฟเวอร์ จำเป็นต้องมีการสร้างการเชื่อมต่อ TCP ตัวที่สองระหว่าง DTP ฝั่งไคลเอนท์กับ DTP ฝั่งเซิร์ฟเวอร์โดยเมื่อ PI ไคลเอนท์ส่งหมายเลขพอร์ตมายัง PI เซิร์ฟเวอร์ ผ่านทางการเชื่อมต่อควบคุมแล้ว DTP ฝั่งเซิร์ฟเวอร์ จะแอดที่พอร์ต 20 บนเซิร์ฟเวอร์ สำหรับการเชื่อมต่อข้อมูล เมื่อการเชื่อมต่อถูกสร้างขึ้น PI ไคลเอนท์จะส่งคำสั่งคอมมานด์ส่งผ่านข้อมูลมา แล้วข้อมูลจะถูกส่งผ่านส่วนการเชื่อมต่อข้อมูล เมื่อทำการส่งข้อมูลเรียบร้อยแล้ว เซิร์ฟเวอร์จะทำการปิดส่วนการเชื่อมต่อข้อมูลลง เป็นผลให้การเชื่อมต่อในด้านไคลเอนท์ถูกปิดลงด้วย



รูปที่ 2.8 โปรแกรมการรับส่งคำสั่งโปรโตคอลส่งผ่านข้อมูลและการตอบกลับ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

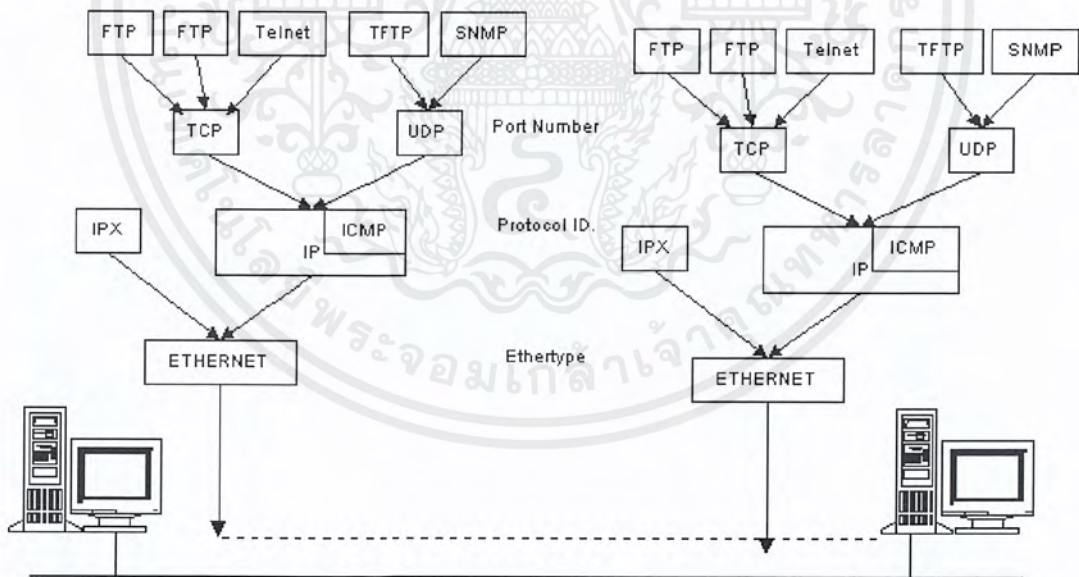
2.3.3 ประวัติความเป็นมาของโพรโทคอล TCP/IP

สืบเนื่องมาจากแนวความคิดของกระทรวงกลาโหมของประเทศสหรัฐอเมริกาที่ต้องการเชื่อมต่อเครื่องคอมพิวเตอร์ที่มีอยู่มากมายหลายผู้จัดจำหน่ายเข้าด้วยกันเป็นระบบเครือข่ายเดียวกัน นับเป็นจุดเริ่มต้นของชุดโพรโทคอล TCP/IP เพราะมีการแสดงให้เห็นถึงการเชื่อมต่อเครื่องคอมพิวเตอร์จาก 4 แห่งเข้าด้วยกัน โดยเรียกเครือข่ายนี้ว่า ARPANET รูปแบบการสื่อสารข้อมูลใช้เทคนิคของการสวิตช์กลุ่มข้อมูล (Packet switching)

ในปี ค.ศ. 1972 เป็นการสาธิตความสามารถของระบบเครือข่ายเป็นครั้งแรก โดยมีเครื่องคอมพิวเตอร์จำนวน 50 เครื่องต่อเข้ากับระบบที่มีอุปกรณ์สวิตช์กลุ่มข้อมูลทำหน้าที่ ส่งผ่านข้อมูลระหว่างเครื่องคอมพิวเตอร์

ในช่วงต้นทศวรรษที่ 80 ชุดโพรโทคอล TCP/IP ได้ถูกรวมเข้าเป็นความสามารถหนึ่งของระบบปฏิบัติการยูนิกซ์ภายใต้ชื่อ UNIX BSD 4.2 เพื่อความปลอดภัยของข้อมูลทางทหารจึงได้ทำการแบ่งเครือข่าย ARPANET ออกเป็น 2 เครือข่าย คือ เครือข่ายทางด้านพลเรือนและงานวิจัย ซึ่งยังคงใช้ชื่อ ARPANET ตามเดิม และเครือข่ายทางทหารใช้ชื่อ MILNET นับแต่นั้นเป็นต้นมาจนถึงระบบปัจจุบันระบบสื่อสารข้อมูลโดยใช้ชุดโพรโทคอล TCP/IP ก็เป็นที่ยอมรับแก่คนทั่วไป

2.3.4 การติดต่อสื่อสารระหว่างคอมพิวเตอร์ที่ใช้โพรโทคอล TCP/IP



รูปที่ 2.9 แสดงถึงการติดต่อสื่อสารระหว่างคอมพิวเตอร์ที่ใช้โพรโทคอล TCP/IP

จากรูปโพรโทคอล TCP/IP ประกอบด้วยโพรโทคอลย่อยหลายโพรโทคอลซึ่งทำงานในระดับต่าง ๆ กัน โดยที่โพรโทคอลย่อยที่ทำงานในระดับ Application Layer ได้แก่ FTP (File Transfer Protocol), TELNET (Terminal Emulator), SNMP (Simple Network Management Protocol)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โพรโทคอลเหล่านี้จะเรียกใช้โพรโทคอลย่อยในระดับ Transport ด้านล่างแบบใดแบบหนึ่งระหว่าง TCP (Transmission Control Protocol) หรือ UDP (User Datagram Protocol) และโพรโทคอลย่อยทั้งสองจะเรียกใช้โพรโทคอลระดับ Network ต่อซึ่งได้แก่ IP หรือ ICMP (Internet Communication Message Protocol)

การทำงานในระดับ Transport ที่มีโพรโทคอลย่อย TCP หรือ UDP นั้นโปรแกรมที่อยู่ด้านบนจะเรียกใช้ผ่านช่องทางที่เป็นตัวเลขที่เรียกว่า Port Number โดยหมายเลขนี้เป็นเลขที่มาตรฐานในโพรโทคอลแบบ TCP/IP มาตรฐานของ Port Number ทั้ง TCP และ UDP ได้แสดงไว้ในตารางต่อไปนี้

ตารางที่ 2.1 มาตรฐานของ Port Number ของ TCP

TCP Port Number	Application Layer Services
5	Remote Job Entry
7	Echo
20	FTP Data
21	FTP Control
23	TELNET
25	SMTP
37	Time
53	Domain Name Server(DNS)
66	Oracle SQL*NET
80	World Wide Web HTTP
110	Post Office Protocol(POP3)

ตารางที่ 2.2 มาตรฐานของ Port Number ของ UDP

UDP Port Number	Application Layer Interface
7	Echo
13	DayTime
37	Time
69	Trival File Transfer Protocol(TFTP)
70	Gopher
107	Remote Telnet
161	SNMP(Simple Network Management Protocol)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ในระบบปฏิบัติการแบบ Windows โพรโตคอล TCP/IP จะถูกสร้างอยู่ในรูปของ DLL (Dynamic Link Library) เพื่อให้โปรแกรมในวินโดวส์สามารถเรียกใช้งานได้ โดยมีมาตรฐานเรียกว่า WinSock (ย่อมาจาก Window Socket) ซึ่งมีบริษัทต่างๆที่ได้คิดค้นโพรโตคอลแบบ TCP/IP ที่ทำงานในลักษณะนี้หลายบริษัท เช่น บริษัทไมโครซอฟต์, บริษัทโนเวล, FTP Software Corp, NetManage Corp รวมทั้งโปรแกรมที่เป็นประเภท Shareware ได้แก่ Trumpet Winsock

2.3.5 ความหมายของ Windows Sockets

Windows Sockets คือ ส่วนติดต่อกับเน็ตเวิร์ค เสมือนเป็นเส้นทางผ่านระหว่างไมโครซอฟท์ วินโดวส์เน็ตเวิร์ค และช่วยให้แอปพลิเคชันสามารถติดต่อกันผ่านเน็ตเวิร์คได้ ซึ่งมีเน็ตเวิร์คแอปพลิเคชันมากมายที่สนับสนุน Windows Sockets ภายใต้อินเทอร์เน็ตโปรโตคอลต่างๆ เช่น Transmission Control Protocol / Internet Protocol (TCP/IP), Xerox Network System (XNS), Digital Equipment Corporation's DECNET Protocol, Internet Packet Exchange/Sequence Packet Exchange (IPX/SPX) ฯลฯ แต่โดยพื้นฐานแล้วจะต้องสนับสนุน TCP/IP หรือ โพรโตคอลใดๆที่สามารถทำงานร่วมกับ Windows Sockets ได้โดยการ implement ไว้ใน DLL (Dynamic Link Library) ซึ่ง Sockets จะช่วยให้โปรแกรมเมอร์ไม่จำเป็นต้องทราบว่าเน็ตเวิร์คทำงานอย่างไร เพราะ Sockets ได้จัดการส่วนนั้นให้หมดแล้ว

Microsoft Foundation Class Library (MFC) สนับสนุนการโปรแกรมกับ Windows Sockets API โดยมี 2 class ให้ใช้ หนึ่งในนั้นคือ class Csocket มันคอยจัดการเกี่ยวกับการเขียนโปรแกรมการสื่อสารข้อมูลบนเน็ตเวิร์คให้

Sockets คือ ปลายทางของการสื่อสารข้อมูล Sockets ต้องมี type และต้องรวมอยู่ใน process ที่กำลังทำงานอยู่ โดยทั่วไป Socket จะสื่อสารข้อมูลกับ Socket ตัวอื่นๆ (ที่กำลังทำงานอยู่) ใน Communication Domain เดียวกัน โดยใช้ IP ช่วย Socket สามารถแบ่งได้เป็น 2 ชนิด คือ

1. Sockets แบบ Stream

ทำงานกับข้อมูลที่ส่งไปโดยไม่กำหนดขอบเขตเรคคอร์ดหรือไม่เป็นเรคคอร์ด ข้อมูลที่ส่งเรียกว่า a stream of bytes และ stream จะรับประกันว่าข้อมูลถูกส่งถึงปลายทางอย่างถูกต้องโดยข้อมูลไม่สลับกันและไม่มีการส่งซ้ำซ้อน

Stream Sockets ใช้ในการส่งข้อมูลในลักษณะซึ่งไม่เป็นเรคคอร์ดซึ่ง stream of bytes นี้ สามารถส่งได้ทั้งสองทาง (ส่งและรับใน Socket เดียวกัน) Stream นี้สามารถเชื่อถือได้ว่า จะถูกส่งอย่างถูกต้อง ไม่สลับกันและไม่ซ้ำกันและส่งไปถึงอย่างแน่นอน โดย Stream ยังสามารถจัดการส่ง class Csocket จะจัดการแตกและรวมให้เอง

การสื่อสารข้อมูลแบบ Stream นั้นเป็นการสื่อสารข้อมูลแบบ Connection-Oriented. Stream นั้นจะมีการเชื่อมต่อกันอย่างชัดเจน ถ้า Socket A ต้องการเชื่อมต่อกับ Socket B แล้ว

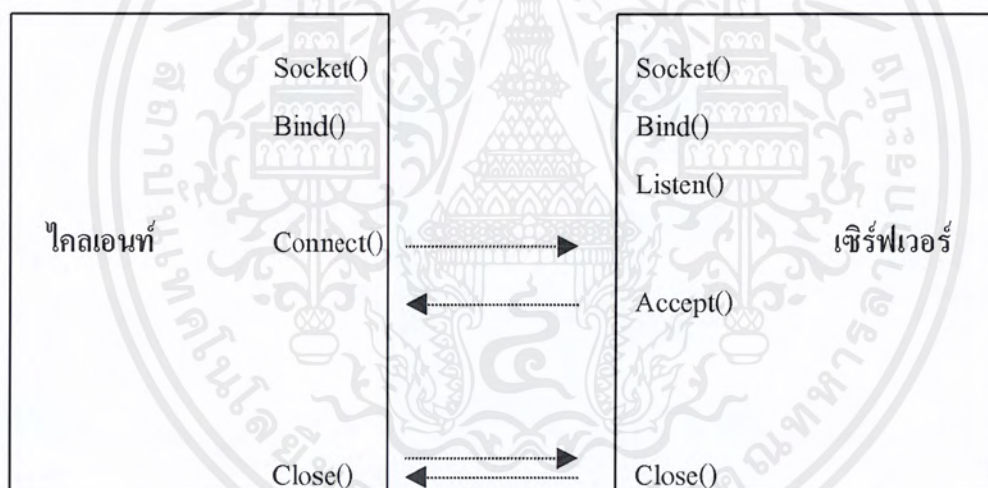
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Socket A จะต้องมีการร้องขอการเชื่อมต่อไปยัง Socket B แล้ว Socket B จะต้องยอมรับหรือปฏิเสธการร้องขอนั้นๆเหมือนกับการโทรศัพท์ ซึ่งจะต้องมีผู้เรียกแล้วผู้รับจะต้องรับสายจึงจะคุยกันได้ และการสนทนาจะไม่มีเสียงซ้ำกัน และสิ่งที่สำคัญที่สุดของ Stream Sockets เป็นที่นิยมกว่า Datagram Sockets เพราะว่าข้อมูลที่ถูกส่งไปนั้นจะถึงมือผู้รับอย่างแน่นอน

2. Sockets แบบ Datagram

สนับสนุนการส่งข้อมูลเป็นเรคอร์ด แต่ไม่ประกันว่าได้ส่งไปถึงและข้อมูลจะมีการสลับกัน Datagram Sockets จะสนับสนุนการส่งข้อมูลแบบสองทาง โดยไม่รับประกันว่าข้อมูลจะถูกส่งไปถึงอย่างถูกต้องหรือไม่และข้อมูลอาจมีการสลับกันหรือซ้ำกัน แต่ขนาดของเรคอร์ดจะได้รับการปกป้องรักษา แม้ว่าขนาดของเรคอร์ดจะใหญ่กว่าขนาดที่ผู้รับกำหนด

การสื่อสารข้อมูลแบบ Datagram เป็นการสื่อสารข้อมูลแบบ connectionless ไม่จำเป็นต้องสร้างการเชื่อมต่อโดยชัดเจน เพียงส่ง Datagram Message ไปยัง Socket ที่กำหนดก็จะสามารถสื่อสารข้อมูลกันได้แล้ว และ Datagram Sockets ยังมีความสามารถส่งข้อมูลแบบกระจาย (Broadcasting Message)



รูปที่ 2.10 แสดงหลักการทำงานแบบไคลเอนท์/เซิร์ฟเวอร์โดยใช้ Socket

2.3.6 เครื่องมือที่ใช้ในการพัฒนาโปรแกรม FTP (File Transfer Protocol)

เนื่องจากปัจจุบัน ได้มีเครื่องมือหลายประเภทที่มีความสามารถในการสร้างโปรแกรม FTP ซึ่งแต่ละประเภทก็มีลักษณะการใช้งานที่แตกต่างกันไป โปรแกรม Visual Basic เป็นโปรแกรมหนึ่งที่มี Active X Control สำหรับติดต่อกับไฟล์ข้อมูลบนเครือข่ายอินเทอร์เน็ต ซึ่ง Control ดังกล่าวคือ Microsoft Internet Transfer Control 6.0 ซึ่ง Control ดังกล่าวสามารถที่จะส่งและรับไฟล์บนเครือข่ายอินเทอร์เน็ตผ่าน โพรโตคอลได้ทั้ง File Transfer Protocol (FTP) และ Hypertext Transfer Protocol (HTTP) และยังสนับสนุนการถ่ายโอนไฟล์แบบ Synchronous ซึ่งหมายถึงจะต้องรอการถ่ายโอนไฟล์จนเสร็จเรียบร้อยก่อน จึงจะสามารถกระทำคำสั่งอื่นได้ และแบบ Asynchronous ซึ่งก็หมายถึงสามารถทำคำสั่งอื่นได้ขณะมีการถ่ายโอนไฟล์ ซึ่งก่อนที่จะใช้ Control Microsoft Internet Transfer Control 6.0 จะต้องทราบ Property Method และ Event ของ Control ซึ่งมีดังนี้

ตารางที่ 2.3 Property ของ Internet Transfer Protocol Control

Property	ประเภทข้อมูล	คำอธิบาย
AccessType	Enum	กำหนดการส่งแบบ Internet directly หรือ ผ่านทาง Proxy server ถ้าส่งผ่าน Proxy server ก็ควรกำหนด Property เท่ากับ icNameProxy ถ้าส่งแบบ Internet directly ควรกำหนดให้เท่ากับ icDirectly หรืออาจจะกำหนดเป็นค่า Default ก็กำหนดค่าเท่ากับ icUseDefault
Document	String	กำหนดไฟล์หรือเอกสารที่ใช้กับ Method Execute
Hinternet	Long	ใช้สำหรับรับค่า Internet handle จาก WININET.DLL เพื่อเรียกใช้ Application Programming Interface (API) ไม่สามารถใช้ใน Visual Basic
Password	String	กำหนดค่า Password เพื่อ logon เข้าสู่เซิร์ฟเวอร์
Protocol	Enum	กำหนดประเภทของโพรโตคอล เป็น FTP หรือ HTTP ถ้ากำหนดเป็น FTP ก็ให้ Property เท่ากับ icFTP ถ้าเป็น HTTP ก็เท่ากับ icHTTP หรืออาจกำหนดให้เท่ากับ icDefault หรือ icUnknown ในกรณีไม่ทราบโพรโตคอล
Proxy	String	กำหนดชื่อ Proxy server ที่ใช้ในการเชื่อมต่ออินเทอร์เน็ต
RemoteHost	String	กำหนดค่าแอดเดรสหรือชื่อโฮสต์ของเซิร์ฟเวอร์ที่จะทำการติดต่อ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Property	ประเภทข้อมูล	คำอธิบาย
RemotePort	Integer	กำหนดค่าพอร์ตที่ใช้ในการเชื่อมต่อเครือข่ายอินเทอร์เน็ต ซึ่ง FTP จะใช้พอร์ต 21 World Wide Web จะใช้พอร์ต 80
RequestTimeout	Long	กำหนดเวลาเป็นวินาที จะทำการกรายกเลิกการเชื่อมต่อในกรณีไม่สามารถเชื่อมต่อได้ภายในเวลาที่กำหนด
ResponseCode	Long	คืนค่าโค้ด Error จากระบบที่อยู่ต่างเครือข่าย
ResponseInfo	String	คืนค่าข้อความ Error จากระบบที่อยู่ต่างเครือข่าย
StillExecuting	Boolean	ใช้สำหรับตรวจสอบว่า Control กำลัง Execute คำสั่งอยู่หรือไม่
URL	String	กำหนดค่า URL ซึ่งจะใช้กับ Method Execute หรือ OpenURL ซึ่งจะประกอบด้วยโปรโตคอลและชื่อโฮสต์ เช่น http://ftp.microsoft.com
Username	String	กำหนดค่า Username เพื่อ Logon เข้าสู่เซิร์ฟเวอร์

ตารางที่ 2.4 Method ของ Internet Transfer Protocol Control

Method	คำอธิบาย
Cancel	ยกเลิกการเชื่อมต่อกับเซิร์ฟเวอร์
Execute	ประมวลผลคำสั่งเพื่อส่งไปยังเซิร์ฟเวอร์
GetChunk	รับข้อมูลจากเซิร์ฟเวอร์ที่เชื่อมต่อ
GetHeader	รับ Header text จากไฟล์ HTTP
OpenURL	รับเอกสารจาก URL ที่กำหนด

Cancel Method เป็น Method ที่ใช้สำหรับยกเลิกการติดต่อกับเซิร์ฟเวอร์ที่กำลังติดต่อซึ่งมีรูปแบบการใช้งานคือ

Object.Cancel

Execute Method จะส่งคำสั่งไปยัง FTP หรือ HTTP เซิร์ฟเวอร์ เพื่อทำงานอย่างใดอย่างหนึ่ง ซึ่งมีรูปแบบการใช้งานที่ประกอบไปด้วยค่า Argument 4 ตัว ซึ่งมีรูปแบบคือ

Object.Execute url, operation, data, requestheaders

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

url argument ก็คือกำหนดค่า URL ที่จะส่งคำสั่งไปทำงานกับ URL นั้น ๆ

operation argument ก็คือคำสั่งที่ส่งไปยังเซิร์ฟเวอร์เพื่อทำงานบางอย่างตามแต่ละคำสั่งซึ่งบางคำสั่งจะสามารถทำงานได้กับเฉพาะโปรโตคอล FTP หรือ HTTP ใดอย่างหนึ่งดังตารางดังนี้

ตาราง 2.5 คำสั่งที่ใช้กับ Method Execute

Protocol	Operation/Syntax	คำอธิบาย
HTTP	GET	รับข้อมูลจาก URL ที่กำหนด
HTTP	HEAD	ส่งค่า Requestheaders argument
HTTP	POST	ส่งข้อมูลไปสู่เซิร์ฟเวอร์
HTTP	PUT	ส่งหน้าข้อมูลไปสู่เซิร์ฟเวอร์
FTP	CD dir	เปลี่ยนไดเรกทอรี
FTP	CDUP	เปลี่ยนไดเรกทอรีเป็นไดเรกทอรีที่สูงขึ้น
FTP	CLOSE	ปิดการเชื่อมต่อ
FTP	DELETE file	ลบไฟล์
FTP	DIR dir	ค้นหาไดเรกทอรี
FTP	LS dir	ค้นหาไดเรกทอรี
FTP	MKDIR dir	สร้างไดเรกทอรี
FTP	PUT file1 file2	ส่งค่าจาก path file1 ไปยังค่า path file2 ของเซิร์ฟเวอร์
FTP	PWD	กำหนดค่าไดเรกทอรี โดย GetChunk method ต้องใช้ในการรับข้อมูล
FTP	QUIT	ปิดการเชื่อมต่อ
FTP	RECV file1 file2	รับข้อมูลโดยกำหนด path file1 ซึ่งส่งมาจาก path file2
FTP	RENAME file1 file2	เปลี่ยนชื่อไฟล์จาก file1 เป็น file2
FTP	RMDIR dir	ลบไดเรกทอรีทิ้ง
FTP	SEND file1 file2	ส่งข้อมูลจาก path file1 ไปยัง path file2 ที่เซิร์ฟเวอร์
FTP	SIZE dir	กำหนดค่าขนาดของไดเรกทอรี

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตาราง 2.6 ตัวอย่างการใช้งานของคำสั่งที่ใช้กับ Method Execute

Operation/Syntax	ตัวอย่างการใช้งาน
GET	Execute <code>ftp://www.microsoft.com/default.htm</code> , “GET”
HEAD	Execute , “HEAD”
POST	Execute , “POST” , strFormData
PUT	Execute , “PUT” , “replace.html”
CD dir	Execute , “CD docs\mydocs”
CDUP	Execute , “CDUP”
CLOSE	Execute , “CLOSE”
DELETE file	Execute , “DELETE discard.txt”
DIR dir	Execute , “DIR /mydocs”
GET file1 file2	Execute , “GET C:\putme.txt /putme.txt”
LS dir	Execute , “LS c:docs”
MKDIR dir	Execute , “MKDIR /my.Dir”
PUT file1 file2	Execute , “PUT C:\putme.txt /putme.txt”
PWD	Execute , “PWD”
QUIT	Execute , “QUIT”
RECV file1 file2	Execute , “RECV old.txt new.txt”
RENAME file1 file2	Execute , “RENAME old.txt new.txt”
RMDIR dir	Execute , “RMDIR oldDir”
SEND file1 file2	Execute , “SEND C:\putme.txt /putme.txt”
SIZE dir	Execute , “SIZE /largefile.txt”

Data argument บางคำสั่งต้องมีการกระทำกับข้อมูลจึงต้องกำหนดข้อมูลเพื่อรองรับคำสั่ง

Requestheader argument มีไว้เพื่อกำหนดหรือบอกข้อมูลของไฟล์ซึ่งจะไว้ที่ Header file

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

GetChunk Method บางคำสั่งจะมีการรับข้อมูลจากเซิร์ฟเวอร์และใส่ลงในบัฟเฟอร์ของ Internet Transfer Control บนเครื่องไคลเอนต์ การใช้ Method GetChunk ก็คือเป็นอ่านข้อมูลจากบัฟเฟอร์มาอีกทีซึ่งมีรูปแบบการใช้คือ

Object.GetChunk (size,[, datatype])

ค่าพารามิเตอร์ Size ก็คือขนาดของไบต์ที่รับมาจากบัฟเฟอร์

ค่าพารามิเตอร์ datatype ก็คือสามารถที่จะเลือกว่าจะรับข้อมูลจากบัฟเฟอร์แบบสตริง (String) ก็จะกำหนดเป็น icString หรือแบบเป็นชุดไบต์ (Byte Array) ก็จะกำหนดเป็น icByteArray

GetHeader Method เป็นการใส่ Header file เพื่อบอกเกี่ยวกับคุณสมบัติหรือลักษณะของไฟล์อาทิ ขนาดไฟล์ วันที่เวลาที่ส่งไฟล์ ตำแหน่งต้นไฟล์ ท้ายไฟล์ เซิร์ฟเวอร์ที่ส่งไฟล์ เป็นต้น ซึ่งมีรูปแบบการใช้งานดังนี้

Object.GetHeader ([headername])

OpenURL Method เป็น Method ที่อนุญาตให้ทำการรับส่งไฟล์ตาม URL ที่กำหนดได้ ซึ่งสามารถใช้ได้ทั้งโพรโทคอล FTP และ HTTP มีรูปแบบการใช้งานดังนี้คือ

Object.OpenURL url [, datatype]

พารามิเตอร์ url ก็คือ URL ที่ต้องการรับหรือส่งไฟล์

พารามิเตอร์ datatype ก็คือสามารถที่จะเลือกว่าจะรับข้อมูลจากบัฟเฟอร์แบบสตริง (String) ก็จะกำหนดเป็น icString หรือแบบเป็นชุดไบต์ (Byte Array) ก็จะกำหนดเป็น icByteArray

StateChanged ใน Internet Transfer Control มี Event StateChanged เดียวเท่านั้นในการทำงาน โดยมีรูปแบบการทำงานคือ

Object_StateChanged (ByVal State As Integer)

Event StateChanged จะทำงานตลอดเวลาในขณะที่ติดต่อกับเซิร์ฟเวอร์ โดยจะคืนค่าออกมาเป็นค่าต่าง ๆ ตามสถานะในขณะที่มีการติดต่อกับเซิร์ฟเวอร์ ซึ่งมีสถานะต่าง ๆ กันดังต่อไปนี้

ตาราง 2.7 ค่าพารามิเตอร์ของ Event StateChanged

VB Constant	ค่า	Control State
icNone	0	State ว่าง
icHostResolvingHost	1	ค้นหาเซิร์ฟเวอร์ตาม IP Address
icHostResolved	2	ค้นพบเซิร์ฟเวอร์ตาม IP Address
icConnecting	3	ติดต่อกับเซิร์ฟเวอร์
IcConnected	4	ติดต่อกับเซิร์ฟเวอร์ได้สำเร็จ
icRequesting	5	ส่งคำร้องขอไปยังเซิร์ฟเวอร์
icRequestSent	6	ส่งคำร้องขอไปยังเซิร์ฟเวอร์เรียบร้อยแล้ว
icReceivingResponse	7	รับตอบสนองจากเซิร์ฟเวอร์
icResponseReceived	8	รับตอบสนองจากเซิร์ฟเวอร์เรียบร้อยแล้ว
icDisconnecting	9	ยกเลิกการติดต่อ
icDisconnected	10	ยกเลิกการติดต่อสำเร็จ
IcError	11	เกิดข้อผิดพลาดในการติดต่อกับเซิร์ฟเวอร์
IcResponseCompleted	12	ส่งคำร้องขอ ได้สำเร็จและได้รับข้อมูลจากเซิร์ฟเวอร์

2.3.7 โทรศัพท์

เนื่องจากการทำงานของระบบนั้น ต้องทำการติดต่อกับโทรศัพท์ธรรมดาที่ใช้กันอยู่ทั่วไป จึงจำเป็นที่จะต้องรู้ว่าข้อมูลจากโทรศัพท์ที่ส่งเข้ามาสู่ระบบมีลักษณะเป็นอย่างไร ซึ่งจะดูได้จากการทำงานภายในตัวโทรศัพท์นี้

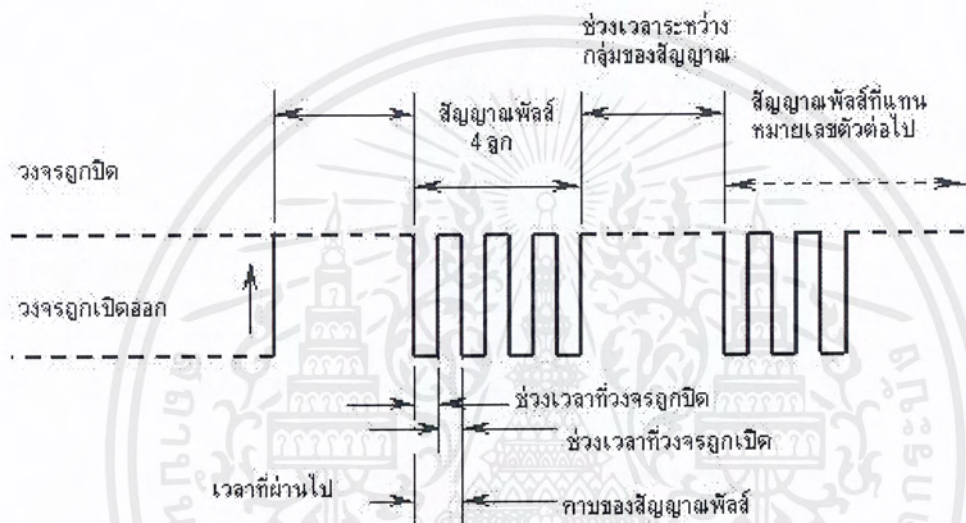
ภายในโทรศัพท์จะมีวงจรกำเนิดเสียงเรียก “Ringer” ซึ่งจะส่งสัญญาณเรียกเมื่อมีการติดต่อมาจากผู้อื่น วงจรนี้จะติดต่อกับชุมสายโดยตรง และถัดมาจะเป็นวงจร Switch Hook เมื่อวางหูโทรศัพท์ไว้กับที่วางปกติ วงจรใน Switch Hook จะถูกเปิดออก ทำให้ไม่มีแรงดันจากชุมสายผ่านไปยังวงจรที่อยู่หลัง Switch Hook ทำให้ไม่สามารถติดต่อไปยังชุมสายได้ เมื่อวางหูโทรศัพท์นั้นไว้

เมื่อมีการยกหูโทรศัพท์ขึ้น วงจร Switch Hook ก็จะปิดลง ทำให้กระแสไหลครบวงจรเข้าเครื่องโทรศัพท์ได้ และกระแสยังไหลไปยังชุมสาย ทำให้ที่ชุมสายพร้อมที่จะทำการติดต่อกับเครื่องโทรศัพท์ได้ จากนั้นชุมสายจะส่งสัญญาณหมุน(Dial Tone) ไปยังผู้ที่ยกโทรศัพท์ เพื่อให้ผู้นั้นส่งหมายเลขโทรศัพท์ที่ต้องการจะติดต่อด้วยมายังชุมสาย หลังจากชุมสายได้รับหมายเลขแรกแล้ว ทางชุมสายก็จะเลิกส่งสัญญาณหมุนไปอย่างรวดเร็ว

การส่งหมายเลข โทรศัพท์ไปยังชุมสายนั้น ทำได้ 2 วิธี คือ การส่งสัญญาณพัลส์ที่แสดงถึงค่าของหมายเลขต่างๆ และการส่งสัญญาณเป็นความถี่ต่างๆกัน โดยค่าของตัวเลขจะถูกแทนด้วยความถี่ 2 ความถี่ที่ Modulate กัน

1) การส่งสัญญาณพัลส์ในระบบโทรศัพท์แบบหมุน

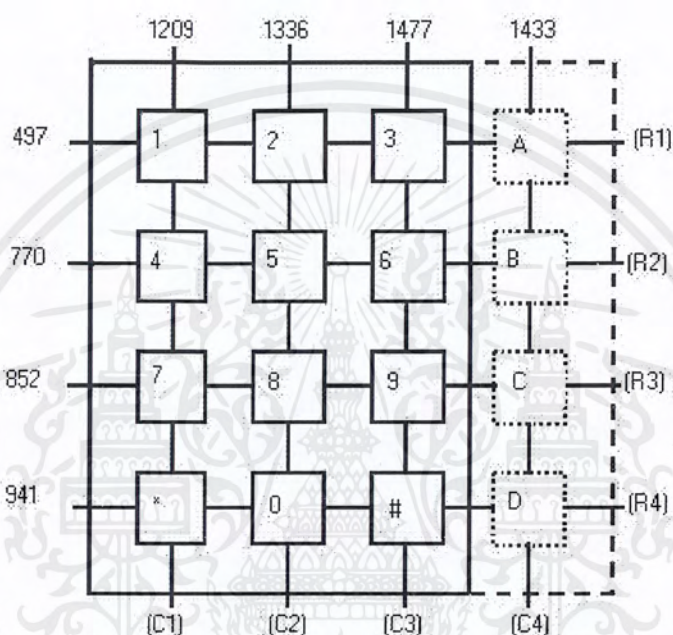
โทรศัพท์แบบหมุนจะใช้การขจัดจังหวะการไหลของกระแส โดยการใช้การเปิด-ปิด Switch ตามหมายเลขที่หมุน เช่น ถ้าหมุนหมายเลข 4 Switch จะถูกทำให้เปิดออก 4 ครั้ง ทำให้เกิดพัลส์ 4 ลูก ดังรูป



รูปที่ 2.11 แสดงระบบโทรศัพท์แบบหมุน

2) การส่งสัญญาณเป็นความถี่ต่างกัน ในระบบโทรศัพท์แบบกดปุ่ม

เป็นระบบการส่งสัญญาณความถี่อีกแบบหนึ่ง ซึ่งโดยทั่วไปจะพบได้มากกว่าในระบบการส่งเป็นสัญญาณพัลส์ โดยระบบนี้จะเรียกว่า Dual Tone Multifrequency Type (DTMF) ในการส่งหมายเลขของผู้ที่ต้องการติดต่อด้วย จะใช้การส่งสัญญาณความถี่ 2 ค่าออกไป Modulate กัน ได้ ผลลัพธ์เป็นตัวหมายเลขที่กด เช่น หมายเลข 5 จะมีความถี่ 770 Hz. และ 1336 Hz. Modulate กันออกมาแสดงถึงหมายเลข 5



รูปที่ 2.12 แสดงเป็นกหนดหมายเลขและค่าความถี่ในแนวนอนและแนวตั้งของหมายเลขนั้นๆ

2.3.8 การทำงานของ Modem

โดยทั่วไปการสื่อสารข้อมูลระหว่างอุปกรณ์คอมพิวเตอร์โดยปกติ โดยจะเรียกอุปกรณ์ที่ใช้เชื่อมต่อรับส่งข้อมูลในคอมพิวเตอร์เหล่านี้ว่า อุปกรณ์ Data Terminal Equipment (DTE) อุปกรณ์ DTE ด้านหนึ่งจะทำการรับส่งข้อมูลกับอุปกรณ์ DTE ปลายทางโดยผ่านเครือข่ายโทรศัพท์ ทิศทางการรับส่งข้อมูลเป็นไปได้ทั้งจากผู้ส่งไปผู้รับ และผู้รับไปยังผู้ส่ง

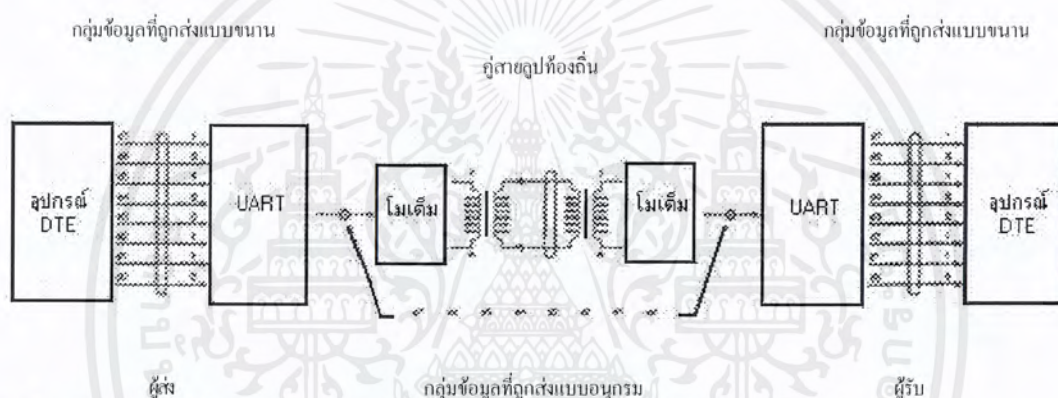
การส่งข้อมูลจากผู้ส่งไปยังผู้รับจะเริ่มจากการที่อุปกรณ์ DTE ด้านผู้ส่ง ส่งข้อมูลซึ่งถูกส่งแบบขนานไปให้ Universal Asynchronous Receive Transmitter (UART) ทำหน้าที่แปลงข้อมูลที่เข้ามาแบบขนานให้ออกไปเป็นอนุกรมส่งไปยัง Modem โดย UART จะจัดให้ส่งบิตที่มีนัยสำคัญต่ำสุดออกไปก่อน จากนั้นข้อมูลแต่ละบิตจะต้องถูกแปลงให้เป็นสัญญาณเสียงที่มีความถี่อยู่ในย่านที่เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สามารถส่งผ่านสายโทรศัพท์ออกไปได้ก่อนจึงส่งออกไป หน้าที่ในการแปลงบิตข้อมูลเป็นเสียงนี้ จะทำโดย Modem ซึ่งคำว่า Modem มาจากการรวมของคำสองคำ คือ Modulation และ Demodulation เข้าด้วยกัน

สำหรับด้านผู้รับ สัญญาณเสียงที่ถูกส่งผ่านสายโทรศัพท์จะถูกแปลงให้อยู่ในรูปของข้อมูลแบบอนุกรมโดย Modem จากนั้น UART จะจัดรูปแบบข้อมูลให้เป็นแบบขนานเพื่อส่งให้กับอุปกรณ์ DTE ด้านรับต่อไป

การแปลงรูปแบบของข้อมูลดิจิทัลที่ต้องการจะส่ง ให้อยู่ในรูปของสัญญาณซึ่งมีความถี่อยู่ในย่าน Bandwidth ของคู่สาย โทรศัพท์ให้ได้ การแปลงสัญญาณดังกล่าว คือ Modulation

ส่วนการแปลงรูปสัญญาณดิจิทัลที่ต้องการรับจากสัญญาณพาหะที่ถูก Modulate ซึ่งถูกส่งจาก Modem ต้นทาง คือ Demodulation



รูปที่ 2.13 แสดงขั้นตอนการเปลี่ยนรูปแบบการส่งข้อมูลผ่าน โมเด็ม

2.3.9 การ Modulate และ Demodulate

เนื่องจากแบนด์วิธของสายโทรศัพท์มีค่าค่อนข้างแคบ ความถี่ที่ยอมให้ผ่านได้จะอยู่ในช่วง 300 เฮิรตซ์ ถึง 3400 เฮิรตซ์ สัญญาณที่มีความถี่สูงมากๆ จะไม่สามารถส่งผ่านคู่สายไปได้ และเมื่อพิจารณาถึงรูปแบบของสัญญาณที่ใช้ในระบบคอมพิวเตอร์จะเห็นว่าเป็นสัญญาณดิจิทัล ซึ่งมีระดับแรงดันที่แน่นอนสองระดับ การเปลี่ยนแปลงระดับแรงดันในกรณีที่ติดต่อเนื่องกันมีค่าไม่เท่ากันจะเป็นไปอย่างรวดเร็วมาก ซึ่งการเปลี่ยนแปลงระดับสัญญาณอย่างรวดเร็วก่อให้เกิดความถี่ที่สูงมาก แต่เพียงแค่ว่ารูปแบบของรูปคลื่นที่ใช้แทนระดับลอจิกอันเกือบจะเป็นรูปพัลส์ก็ไม่สามารถที่จะถูกส่งผ่านไปบนคู่สายโทรศัพท์ได้แล้วดังนั้นจึงจำเป็นต้องทำการแปลงรูปแบบของข้อมูลดิจิทัลที่ต้องการจะส่งให้อยู่ในรูปของสัญญาณซึ่งมีความถี่อยู่ในย่านแบนด์วิธของคู่สายโทรศัพท์ให้ได้ การแปลงสัญญาณดังกล่าวมีชื่อเรียกว่า การมอดูเลต (Modulation)

การ Modulate เป็นกระบวนการเปลี่ยนรูปแบบบางอย่างของคลื่นสัญญาณไฟฟ้า นิยมเรียกว่า สัญญาณพาหะ (Carrier) ให้เปลี่ยนไปตามรูปแบบของอีกสัญญาณหนึ่ง ซึ่งเรียกว่า Modulating signal

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สำหรับในกรณีของการรับส่งข้อมูลดิจิทัลผ่านคู่สายโทรศัพท์ กระบวนการ Modulate จะนำคลื่น สัญญาณพาหะซึ่งเป็นสัญญาณที่อยู่ในช่วงความถี่ 300 เฮิรตซ์ ถึง 3400 เฮิรตซ์ มาทำการเปลี่ยนแปลง รูปแบบบางอย่างตามข้อมูลไบนารีที่ถูกส่งจากอุปกรณ์ DTE รูปแบบของสัญญาณพาหะที่สามารถ เปลี่ยนแปลงได้ก็ได้แก่แอมพลิจูดของสัญญาณ, ความถี่ของสัญญาณ และ เฟสของสัญญาณ ซึ่งการ มอดูเลตแต่ละแบบต่างก็มีการใช้งานใน Modem แต่ละรุ่น

ในส่วนของ Modem ที่ทำหน้าที่เป็นตัวรับสัญญาณทางด้านผู้รับ จะทำหน้าที่ Demodulate หรือแปลงรูปสัญญาณดิจิทัลที่ต้องการรับจากสัญญาณพาหะที่ถูก Modulate ซึ่งถูกส่งจาก Modem ด้านทาง ในส่วนของภาค Demodulate จะประกอบไปด้วยวงจรกรองความถี่สัญญาณ (filter) และวงจร ตรวจจับสัญญาณ (detector) ซึ่งใช้ตรวจจับสัญญาณซึ่งถูกส่งมาโดยถูกต้องตรงกับความถี่ส่ง

นอกจากนี้ยังมีวงจรสัญญาณทางแอมพลิจูด,เฟส และความถี่ วงจรปรับระดับสัญญาณ เพื่อแปลงรูปสัญญาณที่ได้รับไม่ว่าจะเป็นการ Modulate แบบใดให้กลับมามีอยู่ในรูปของสัญญาณ ไบนารีพร้อมกับปรับระดับสัญญาณ ไบนารีเหล่านั้นให้มีระดับความแรงของสัญญาณตามมาตรฐาน ก่อนที่จะส่งไปยังอุปกรณ์ UART และ DTE ต่อไป

2.3.10 การทำงานของ Voice Modem

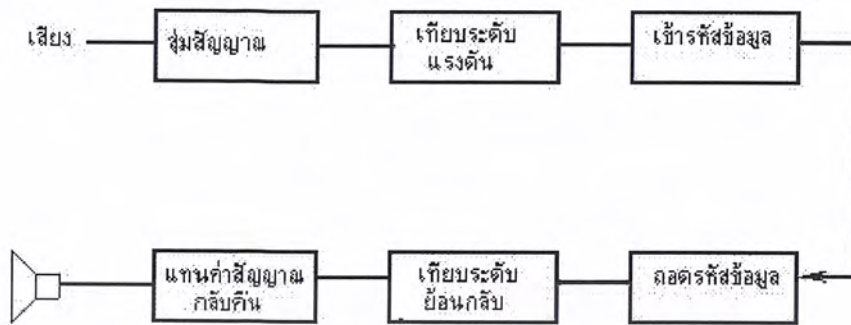
Voice Modem จะมีความสามารถในการทำ Modulate สัญญาณดิจิทัลเหมือนกับ Modem ปกติทุกประการ แต่ได้เพิ่มความสามารถในการทำ Modulate สัญญาณในลักษณะที่เป็นข้อมูลเสียง เอาไว้ด้วย ดังนั้นภายใน Modem ชนิดนี้จะเพิ่มวงจรการแปลงข้อมูลเสียงทั้งจากสัญญาณ Analog เป็นสัญญาณ Digital ด้วย โดยการ Modulate จะใช้หลักการของ Pulse Code Modulation (PCM)

พีซีเอ็ม (PCM) เป็นชื่อย่อมาจาก Pulse Code Modulation ซึ่งเป็นการส่งสัญญาณอะนาลอก โดยเปลี่ยนให้อยู่ในรูปของข้อมูลดิจิทัลที่ถูกเข้ารหัสอยู่ผ่านเครือข่ายสื่อสารระบบดิจิทัล ข้อดีของ การส่งข้อมูลในรูปดิจิทัล คือ การป้องกันการรบกวนจากสภาพแวดล้อม ทั้งนี้เนื่องจากสัญญาณ ดิจิทัลเป็นสัญญาณที่มีระดับของแรงดันเพียง 2 สถานะ คือ สูง (high) และต่ำ (low) เท่านั้น และยัง สามารถลดปัญหาการผิดเพี้ยนของรูปร่างสัญญาณที่เกิดขึ้นระหว่างการส่งได้ ทั้งที่เกิดจากการลด ทอนสัญญาณตามระยะทางและที่เกิดจากสัญญาณรบกวน ซึ่งจะถูกแก้ไขเมื่อถึงเครื่องรับปลายทาง โดยใช้หลักการเปรียบเทียบค่าระดับแรงดันของสัญญาณที่มาถึงปลายทางกับระดับแรงดันเทรชโฮลด์ (Threshold Voltage) หากแรงดันที่ได้รับมีค่าสูงกว่าแรงดันเทรชโฮลด์ เครื่องรับจะกำหนดให้เป็นลอ- จิกสูง และในทางตรงข้าม เครื่องรับก็จะกำหนดให้สัญญาณที่ได้รับเป็นลอจิกต่ำ หากมีระดับแรงดัน ต่ำกว่าแรงดันเทรชโฮลด์

การเปลี่ยนสัญญาณอะนาลอกให้เป็นข้อมูลพีซีเอ็มประกอบด้วย 3 กระบวนการหลัก ๆ คือ การสุ่มสัญญาณ (sampling), การเทียบระดับแรงดัน, และการเข้ารหัสข้อมูล (encoding) หลังจากนั้น จึงทำการส่งข้อมูลผ่านเครือข่าย และเมื่อข้อมูลดังกล่าวถึงปลายทางก็จะผ่านกระบวนการย้อนกลับ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

คือ การถอดรหัสข้อมูล (decoding), การเทียบระดับแรงดันย้อนกลับและการแทนค่าสัญญาณกลับคืน (recovering) ซึ่งกระบวนการทั้งหมดแสดงดังในรูป

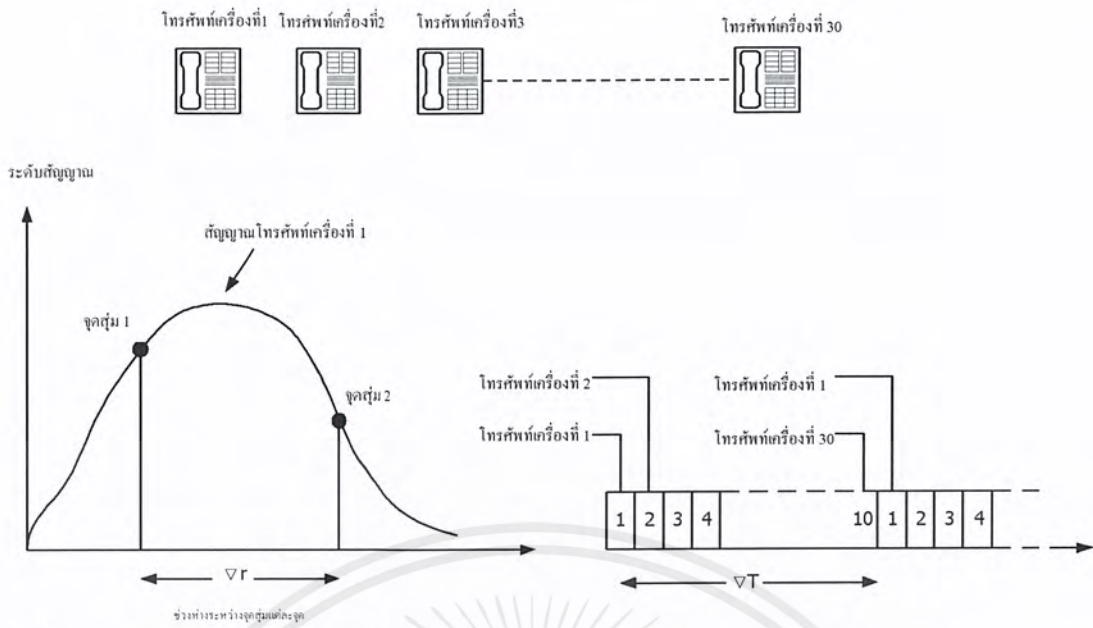


รูปที่ 2.14 แสดงพื้นฐานกระบวนการทำงานของการส่งข้อมูลแบบ PCM

2.3.11 การสุ่มสัญญาณ

การสุ่มสัญญาณเป็นขั้นตอนแรกของการแปลงสัญญาณอะนาลอกเป็นสัญญาณดิจิทัล โดยวงจรสุ่มสัญญาณจะตรวจจับขนาดของสัญญาณอะนาลอกที่ถูกส่งตามช่วงเวลาที่กำหนด โดยทั่วไปมักจะวัดขนาดในรูปของแรงดันไฟฟ้า ซึ่งแท้จริงแล้วกระบวนการสุ่มสัญญาณเป็นกระบวนการตรวจวัดค่าแรงดันของสัญญาณในช่วงเวลาต่าง ๆ ซึ่งมีคาบการตรวจจับคงที่

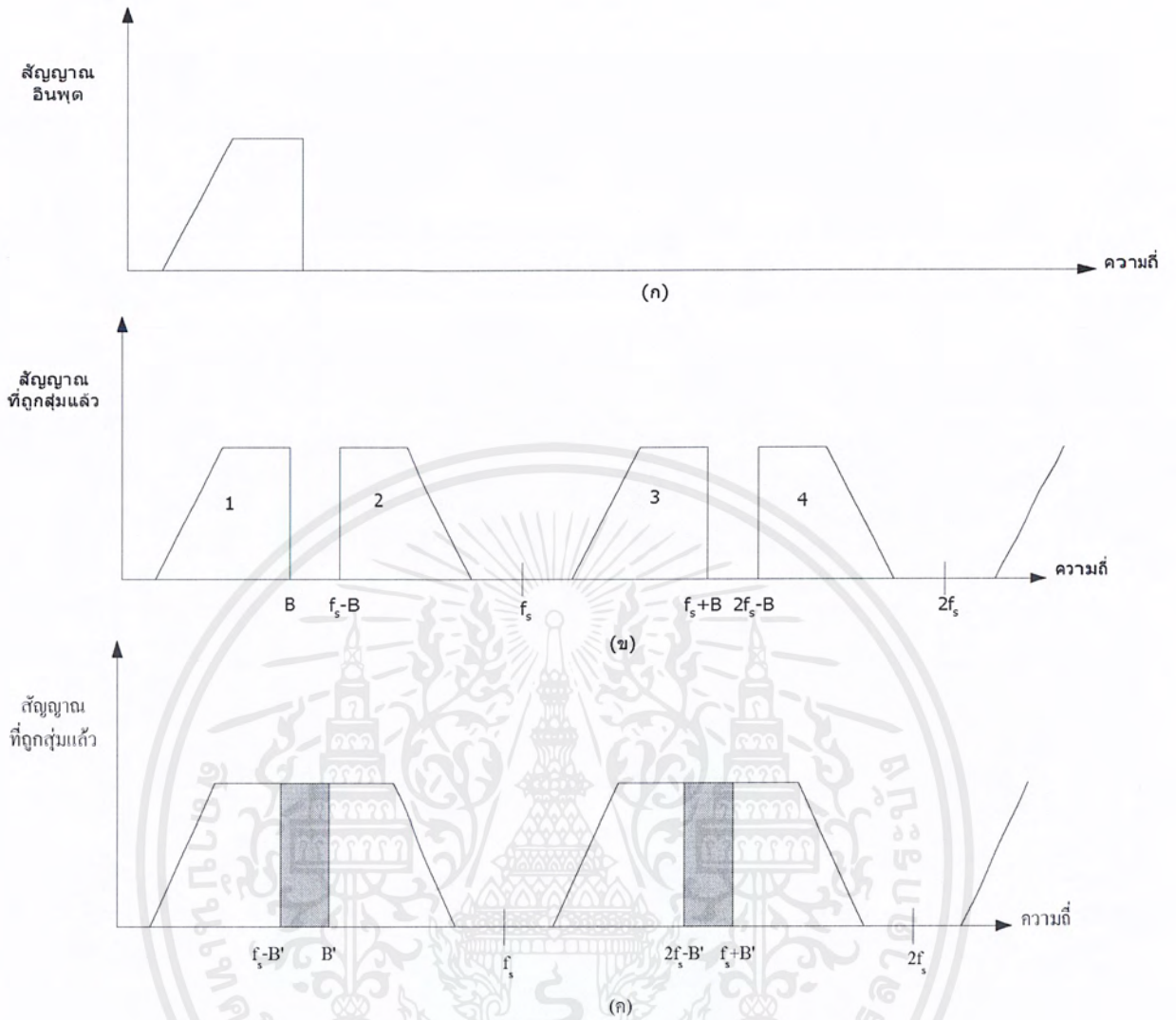
อัตราหรือความถี่ของการสุ่มสัญญาณเป็นคาบนี้จะกำหนดให้อยู่ในหน่วยของจำนวนจุดสุ่มต่อหนึ่งหน่วยเวลา ยกตัวอย่างเช่น ในระบบโทรศัพท์อัตราการสุ่มดังกล่าวจะมีค่าเป็น 8,000 ครั้งต่อวินาทีหรือ 8 กิโลเฮิร์ตซ์ หรืออาจกล่าวได้ว่าคาบของการสุ่ม (ช่วงเวลาระหว่างการสุ่มแต่ละครั้ง) มีค่าเป็น $1/8,000$ หรือ 125 ไมโครวินาที สำหรับวิธีในการคำนวณหาอัตราสุ่มระบบโทรศัพท์จะเป็นไปตามข้อกำหนดในทฤษฎีของการสุ่มสัญญาณ ซึ่งถูกกำหนดขึ้นโดยแชนนอน (Shannon) นักคณิตศาสตร์ชาวสหรัฐอเมริกา กล่าวว่า อัตราการสุ่มจะต้องมีความถี่ไม่น้อยไปกว่า 2 เท่าของความถี่สูงสุดของสัญญาณอะนาลอกที่จะทำการส่งนั้น จึงจะสามารถสร้างสัญญาณต้นฉบับกลับคืนจากสัญญาณสุ่มได้



รูปที่ 2.15 แสดงการสุ่มสัญญาณ

ในช่วงคาบเวลาระหว่างการสุ่มแต่ละจุดนั้น เราสามารถนำค่าแรงดันสุ่มของสัญญาณจากหลาย ๆ ช่องสัญญาณ (หลายคู่สาย) มาวางเรียงต่อกันได้ดังในรูปด้านบน โดยเรียกกระบวนการนี้ว่าการมัลติเพล็กซ์ทางเวลา (Time Division Multiplex : TDM) ซึ่งเป็นวิธีการที่ใช้ในระบบพีซีเอ็มทุกระบบ ตัวอย่างเช่น ระบบพีซีเอ็ม 30 ช่องสัญญาณซึ่งจะกล่าวถึงต่อไปนั้น แรงดันสุ่มจากคู่สายจำนวน 30 คู่สายจะถูกนำมาเตรียมพร้อมเพื่อทำการส่งภายในคาบเวลาของการสุ่มเพียง 1 คาบ

สัญญาณที่ผ่านกระบวนการสุ่มแล้ว จะเก็บรูปแบบของสัญญาณต้นฉบับก่อนที่จะถูกสุ่มได้อย่างสมบูรณ์เพียงใด ขึ้นอยู่กับความถี่ที่ใช้ในการสุ่ม ซึ่งจะต้องมีค่าไม่น้อยไปกว่า 2 เท่าของความถี่สูงสุดของสัญญาณต้นฉบับ สมมติว่าความถี่ที่ใช้ในการสุ่มมีค่าเป็น f_s เฮิรตซ์ และความถี่สูงสุดของสัญญาณต้นฉบับมีค่าเป็น B ซึ่งมีสเปกตรัมของสัญญาณดังในรูป 2.16(ก) เพราะตามหลักการทางคณิตศาสตร์ หากนำสัญญาณซึ่งมีแบนด์วิดท์ B มาสุ่มด้วยสัญญาณซึ่งมีความถี่ f_s จะได้สัญญาณซึ่งมีสเปกตรัมดังในรูป 2.16(ข) สัญญาณนี้เองที่จะถูกส่งผ่านเครือข่ายสื่อสารเพื่อไปยังเครื่องรับปลายทาง หากพิจารณาสเปกตรัมของสัญญาณในรูป 2.16(ข) จะพบว่ามีส่วนประกอบของสเปกตรัมย่อย ๆ จำนวนมาก สเปกตรัมย่อยชุดแรก (หมายเลข 1) เป็นสเปกตรัมของสัญญาณอยู่ในช่วงความถี่ 0 ถึง B อันเป็นสเปกตรัมของสัญญาณก่อนที่จะถูกสุ่มนั่นเอง สเปกตรัมย่อยหมายเลข 3 มีรูปร่างคล้ายกับสเปกตรัมย่อยหมายเลข 1 แต่ถูกเลื่อนไปด้วยความถี่ f_s สเปกตรัมย่อยหมายเลข 2 มีรูปร่างเหมือนภาพกลับกับสเปกตรัมย่อยหมายเลข 3 โดยมีความถี่ f_s เป็นคล้ายกระจกเงาในการสะท้อน และนอกจากนี้ จะพบคู่สเปกตรัมย่อย 4-5, 6-7,... เรื่อยไปจนถึงอนันต์ โดยมีระยะห่างระหว่างกันเท่ากับ f_s



- (ก) สเปกตรัมของสัญญาณอะนาลอกก่อนผ่านการสุ่ม
- (ข) สเปกตรัมของสัญญาณที่ถูกสุ่มแล้วโดย $f_s > 2B$
- (ค) สเปกตรัมของสัญญาณที่ถูกสุ่มแล้วโดย $f_s < 2B$

รูปที่ 2.16 แสดงสเปกตรัมของสัญญาณ

เนื่องจากสเปกตรัมของสัญญาณที่ถูกสุ่มแล้วประกอบด้วยส่วนประกอบของสัญญาณต้นฉบับในช่วงความถี่ทุก ๆ ค่า f_s ซึ่งไม่มีสเปกตรัมย่อยใดเกิดการทับกันตลอดย่านความถี่ จึงกล่าวได้ว่าสัญญาณที่ถูกสุ่มแล้วจะยังคงเก็บรายละเอียดของสัญญาณต้นฉบับไว้ได้ครบถ้วน ทรานสคริปต์ที่อัตราการสุ่ม f_s มีค่ามากกว่า $2B$ ซึ่งสามารถใช้วงจรกรองความถี่ต่ำผ่านดึงสัญญาณต้นฉบับกลับคืนมาได้ที่เครื่องรับปลายทาง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ในทางกลับกัน หากอัตราการสุ่ม f_s มีค่าน้อยกว่า $2B$ จะเกิดเหตุการณ์ แสดงในรูปที่ 2.16(ค) จะพบว่าสเปกตรัมย่อยแต่ละตัวเกิดการซ้อนทับกันเป็นบางส่วน พื้นที่ของการซ้อนทับจะมากหรือน้อยขึ้นอยู่กับค่าความถี่ f_s ผลที่เกิดขึ้นทำให้เกิดการสูญเสียข้อมูลของสัญญาณต้นฉบับในส่วนที่เกิดการทับกัน ทั้งนี้ เนื่องจากวงจรกรองความถี่ต่ำผ่านที่เครื่องรับปลายทางไม่สามารถแยกสัญญาณที่เกิดการซ้อนทับออกจากกันได้ ปรากฏการณ์ดังกล่าวมีชื่อเรียกว่า Aliasing distorting

2.3.12 การเทียบระดับแรงดัน

การเทียบระดับแรงดันเป็นกระบวนการในการนำสัญญาณที่ถูกสุ่มแล้วมาจัดกลุ่มภายในระดับซึ่งแบ่งออกเป็นช่วง ๆ เรียกว่าช่วงควอนไทซ์ (quantizing interval) แต่ละช่วงจะถูกแทนค่าด้วยค่าคงที่มีชื่อเรียกว่า ค่าควอนไทซ์ ซึ่งระดับค่าควอนไทซ์ในสถานีต้นทางและสถานีปลายทางจะเหมือนกันทุกประการ

โดยทั่วไป ขั้นตอนการเทียบระดับแรงดันจะเกี่ยวข้องกับการเข้ารหัสสัญญาณอย่างมาก การจัดระดับควอนไทซ์มีจุดประสงค์หลักเพื่อปรับค่าแรงดันของสัญญาณที่ผ่านกระบวนการสุ่มสัญญาณมาแล้วให้ลงตัวกับค่าแรงดันมาตรฐานจำนวนหนึ่งที่ถูกกำหนดเตรียมไว้โดยต้องตรงกับระดับควอนไทซ์แต่ละค่า เพื่อทำการแปลงเป็นรหัสไบนารีนั่นเอง จากรูปจะแสดงรายละเอียดและศัพท์ที่ใช้ในกระบวนการเทียบระดับแรงดัน ซึ่งผู้อ่านควรจะทราบความหมายของศัพท์ต่าง ๆ ดังนี้

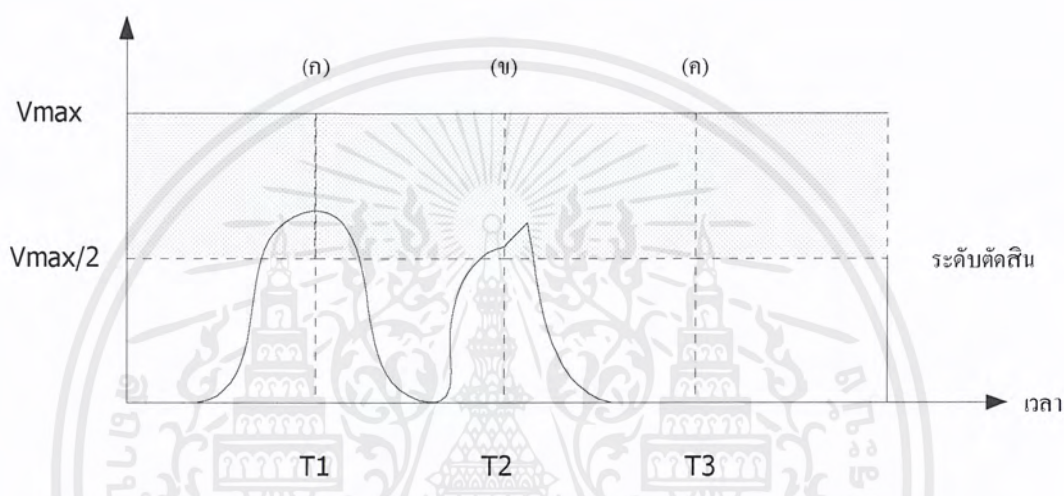
ย่านดำเนินการ เป็นย่านแรงดันซึ่งยอมให้มีการเทียบระดับแรงดันได้ สัญญาณที่ถูกสุ่มซึ่งมีระดับแรงดันอยู่ในย่านดังกล่าวจะสามารถผ่านกระบวนการนี้ไปได้โดยเกิดความคลาดเคลื่อนน้อยที่สุด ซึ่งจะกล่าวต่อไป ส่วนสัญญาณที่มีขนาดแรงดันสูงหรือต่ำกว่าย่านดำเนินการจะถูกแทนค่าด้วยระดับตัดสินใจ (จะกล่าวถึงต่อไป) สูงสุดหรือต่ำสุดที่มีได้ ภายในย่านดังกล่าวจะถูกแบ่งออกเป็นช่วงควอนไทซ์จำนวนมาก ในระบบโทรศัพท์ทั่วไป ย่านดำเนินการจะถูกแบ่งออกเป็น 256 ช่วง

ระดับตัดสิน เป็นค่าแรงดันอ้างอิงซึ่งถูกกำหนดโดยขอบเขตรอยต่อของช่วงควอนไทซ์คู่หนึ่งๆ หรือถ้าเป็นในระบบทวนสัญญาณดิจิทัล ค่าดังกล่าวจะเป็นค่าเทรซโฮลด์ระหว่างย่านแรงดัน 2 ย่านดังในรูปที่ 2.17 โดยรูปที่ 2.17(ก) แสดงถึงพัลส์ที่ไม่ถูกสัญญาณรบกวน แต่เกิดการลดทอนของแรงดันอันเนื่องมาจากการลดทอนตามระยะทางของสายส่ง รูปที่ 2.17(ข) แสดงถึงพัลส์ซึ่งถูกสัญญาณรบกวนในแง่ของแรงดัน ซึ่งแรงดันในขณะที่ถูกสุ่มสัญญาณมีค่ามากกว่า $V_{max}/2$ สัญญาณทั้งสองกรณีจะถูกป้อนเข้าวงจรทวนสัญญาณ ซึ่งจะทำการสุ่มสัญญาณทุก ๆ ช่วงเวลา T_s โดยกำหนดให้ตรงกับจุดกึ่งกลางของพัลส์ ถ้าระดับแรงดันของสัญญาณ ณ ช่วงการสุ่มมีค่ามากกว่าแรงดันระดับตัดสิน ($V_{max}/2$) วงจรทวนสัญญาณจะถือว่าตรวจพบลอจิก “1” ในทางกลับกัน ถ้าแรงดัน ณ จุดสุ่มมีค่าน้อยกว่าแรงดันระดับตัดสิน วงจรทวนสัญญาณจะถือว่าตรวจพบลอจิก “0” จากตัวอย่างสัญญาณทั้งสองกรณีเมื่อผ่านเครื่องทวนสัญญาณแล้วจะถูกปรับรูปให้เป็นพัลส์สมบูรณ์แบบดังในรูปที่ 2.17(ค) ในกรณีดังกล่าวค่าเทรซโฮลด์ซึ่งมีค่า $V_{max}/2$ ถิ่นับได้ว่าเป็นระดับตัดสินเช่นกัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ระดับตัดสินใจเสมือน เป็นระดับตัดสินใจที่อยู่ ณ ตำแหน่งปลายสุดของผ่านดำเนินการ ค่าดังกล่าวกำหนดขอบเขตขนาดของสัญญาณซึ่งสามารถผ่านการเทียบระดับไปได้โดยไม่เกิดการขลิบขนาดของสัญญาณ โดยจะเกิดขึ้นเมื่อสัญญาณที่ถูกสุ่มมีระดับเกินกว่าระดับตัดสินใจเสมือน

ระดับสุ่มกลับคืน เป็นระดับควอนไทล์ซึ่งถูกสร้างขึ้นที่เอาต์พุตของวงจรถอดรหัสของเครื่องปลายทางค่าแรงดันดังกล่าวเกิดจากการป้อนรหัสไบนารีที่ส่งจากสถานีต้นทางไปยังเครื่องรับปลายทาง วงจรถอดรหัสจะทำการแทนค่ารหัสไบนารีซึ่งถูกป้อนเข้ามาด้วยระดับสุ่มกลับคืน แล้วทำการส่งไปเข้าวงจรกรองความถี่ต่ำผ่านเพื่อสร้างสัญญาณต้นฉบับกลับคืนมา



- (ก) แสดงพัลส์ที่ไม่ถูกสัญญาณรบกวน แต่เกิดการลดทอนสัญญาณอันเกิดจากระยะทางของสายส่ง
- (ข) แสดงพัลส์ที่ถูกสัญญาณรบกวนในแง่ของแรงดัน
- (ค) แสดงพัลส์ที่ถูกปรับรูปแล้วเป็นพัลส์ที่สมบูรณ์

รูปที่ 2.17 แสดงการกำหนดระดับตัดสินของระบบทวนสัญญาณดิจิทัลซึ่งกำหนดย่านแรงดันตัดสินไว้ 2 ย่าน

เมื่อสัญญาณที่ผ่านกระบวนการสุ่มถูกป้อนเข้าวงจรควอนไทล์ ระดับแรงดัน ณ จุดสุ่มนั้นจะถูกนำมาจัดเข้าในช่วงควอนไทล์ที่เหมาะสม สมมติว่าระดับดังกล่าวมีค่าแรงดันในช่วง V_n ถึง V_{n+1} วงจรจะถือว่า ค่าควอนไทล์ของสัญญาณ ณ จุดนั้นมีค่าเป็นค่าควอนไทล์ภายในช่วงแรงดันนั้น ไม่ว่าจะแรงดัน ณ จุดสุ่มใด ๆ ก็ตามที่อยู่ในช่วง V_n ถึง V_{n+1} ก็จะถูกแทนค่าด้วยค่าควอนไทล์เดียวกัน

การจัดระดับควอนไทล์ตามที่กล่าวมาเป็นรูปแบบที่ช่วงควอนไทล์แต่ละช่วงมีค่าเท่ากัน รูปที่ 2.17 นั้นแสดงถึงลักษณะการแปลงสัญญาณที่ผ่านกระบวนการสุ่มมาแล้วให้อยู่ในระดับควอนไทล์

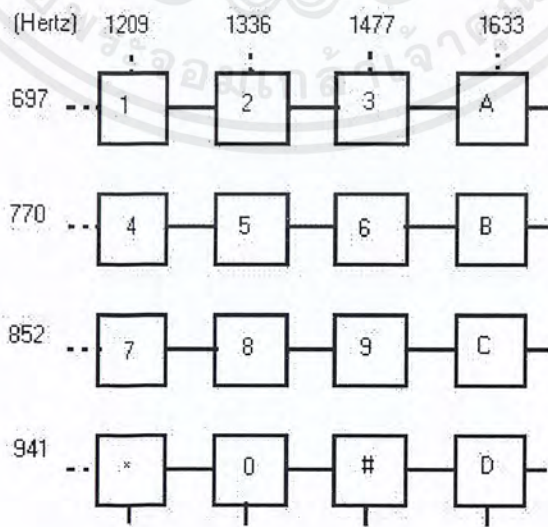
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ซึ่งแทนค่าด้วยข้อมูลไบนารีขนาด 4 บิต ตามข้อกำหนดพีซีเอ็มของ CCITT และทวีปอเมริกาเหนือ กำหนดให้ขนาดของข้อมูลที่ให้แทนในแต่ละระดับควอนไทซ์มีขนาด 8 บิต โดยที่จำนวนขั้นของระดับควอนไทซ์ให้มากขึ้น อย่างไรก็ตาม การเพิ่มระดับควอนไทซ์ส่งผลให้มีการเพิ่มจำนวนของบิตที่แทนค่าระดับควอนไทซ์ ซึ่งจะทำให้ความกว้างของพัลส์ของข้อมูลไบนารีที่จะถูกส่งผ่านเครือข่ายแคบลงในกรณีที่ต้องการส่งข้อมูลด้วยอัตราเร็วเท่าเดิม อันเป็นการลดจำนวนช่องสัญญาณที่จะสามารถส่งแบบมัลติเพล็กซ์ไปพร้อมกันได้

2.3.13 วงจรถอดรหัสหมายเลข DTMF

ภายใน Voice Modem จะมีวงจรที่ทำหน้าที่ถอดรหัสหมายเลขที่ส่งมาแบบ DTMF (DTMF receiver) เพื่อตรวจสอบการกดปุ่มโทรศัพท์จากผู้ที่โทรศัพท์เข้ามา จะกล่าวถึงที่มาของวงจรดังนี้

ในช่วงแรก ๆ จะใช้วงจรถอดรหัส 1 วงจรต่อ 1 คู่สาย เมื่อมีการขยายการใช้งานโทรศัพท์กันมากขึ้น ขุมสายก็มีขนาดใหญ่ขึ้น ภายในขุมสายจะมีคู่สายภายใต้การควบคุมเป็นจำนวนมาก การใช้งานของวงจรถอดรหัสหมายเลขแบบ DTMF ซึ่งจะประกอบไปด้วยสัญญาณที่มีความถี่ต่างกัน 2 สัญญาณตามตำแหน่งคอลัมน์และแถวของปุ่มกดหมายเลข และทำการมอดูเลตเข้าด้วยกันก่อนที่จะทำการส่งออกไป อย่างไรก็ตาม มาพบทวนกันอีกครั้งเมื่อกดปุ่มหมายเลขใด ๆ ไปแล้วจะมีสัญญาณความถี่ค่าเท่าไรบ้างที่ถูกผลิตออกมา ในรูป 2.18 จะเป็นค่าความถี่ต่าง ๆ ในคอลัมน์และโรว์ ซึ่งจะเป็นค่าที่กำหนดไว้เป็นค่ามาตรฐานของระบบการเข้ารหัสแบบ DTMF อยู่แล้ว ซึ่งที่ขุมสายหลังจากที่รับสัญญาณ DTMF มาแล้วก็จะนำไปผ่านวงจรกรองความถี่ที่มีผลตอบสนองตามรูปนี้ เมื่อสัญญาณ DTMF ผ่านวงจรกรองความถี่มาแล้ว ก็จะได้สัญญาณความถี่ 2 ค่า ซึ่งก็เป็นความถี่เดียวกับความถี่มาตรฐานก่อนที่จะทำการมอดูเลตนั่นเอง

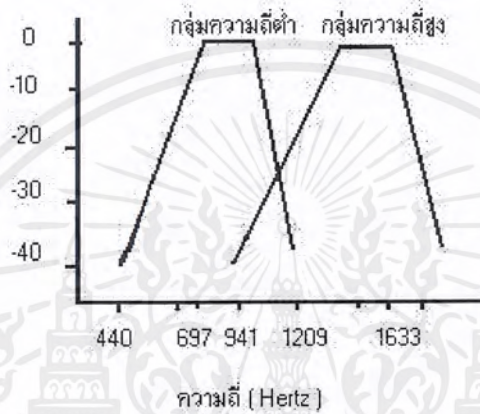


รูปที่ 2.18 ความถี่ของระบบ DTMF

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.3.14 วงจรกรองความถี่และวงจรตรวจจับ (filtering and detector)

ในรูปที่ 2.19 จะเห็นว่าวงจรกรองความถี่เป็นส่วนสำคัญของวงจรถอดรหัสจากรูปกราฟที่แสดงผลตอบสนองความถี่ วงจรกรองความถี่ชนิดที่ใช้สำหรับการทำงานให้ได้ผลตอบสนองตามรูป 2.19 จะต้องใช้เวลากกรองความถี่ชนิดแยกย่านความถี่ (bandsplit filter) ดังนั้น สัญญาณ DTMF ที่ผ่านวงจรกรองความถี่ออกมาแล้วก็จะแยกได้เป็นกลุ่มความถี่ที่สูง (high group) กับกลุ่มความถี่ที่ต่ำ (low group) ส่วนสัญญาณความถี่ที่อยู่นอกเหนือย่านนี้ ซึ่งไม่ตรงกับค่าความถี่มาตรฐานหรือมีค่าเบี่ยงเบนเกิน 2 % ก็จะไม่สามารถผ่านวงจรกรองความถี่นี้ไปได้

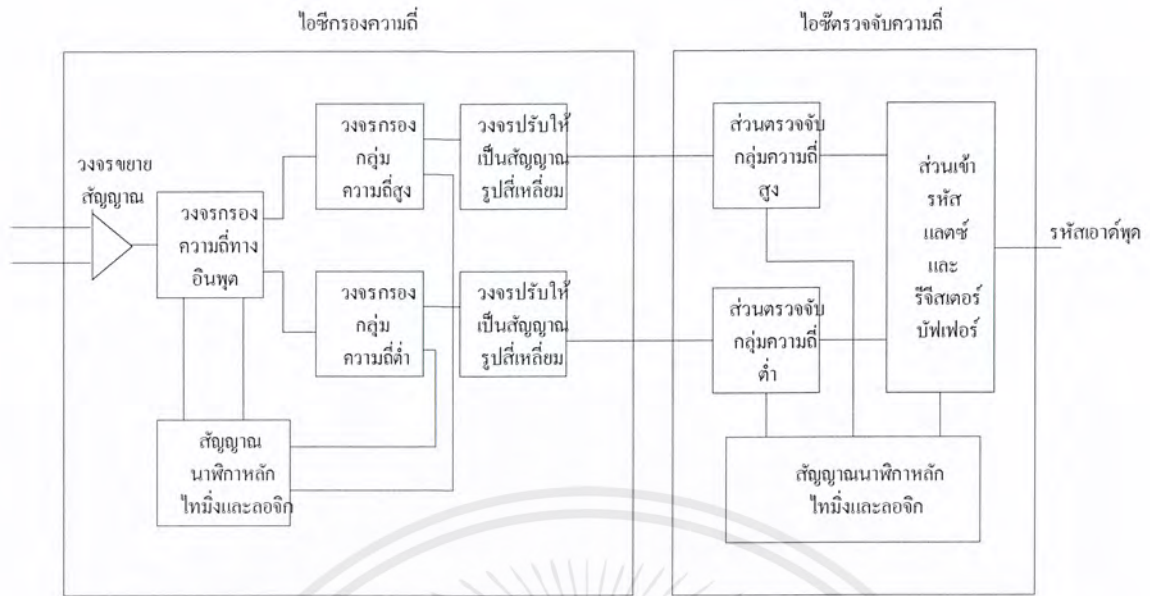


รูปที่ 2.19 ผลตอบสนองความถี่ของวงจรกรองความถี่

จากนั้น สัญญาณความถี่ที่ผ่านออกมาก็จะถูกนำไปผ่านวงจรตรวจจับ (detector) เพื่อที่จะทำการประมวลผลต่อไปในรูปด้านล่าง เป็นบล็อกไดอะแกรมของวงจรกรองความถี่และวงจรตรวจจับ

สำหรับการทำงานของวงจรตรวจจับความถี่ทั้ง 2 ย่าน ที่ผ่านมาจากวงจรกรองความถี่แล้ว ก็จะถูกนำไปผ่านวงจรสร้างสัญญาณรูปสี่เหลี่ยม (squaring circuits) เพื่อทำให้เป็นสัญญาณในระบบดิจิทัลเสียก่อน จากนั้นวงจรตรวจจับก็จะทำการประมวลผลสัญญาณ ซึ่งก็จะใช้วิธีการนับจำนวนพัลส์ โดยจะทำการนับจำนวนพัลส์ภายใน 1 คาบสัญญาณรูปสี่เหลี่ยมนั่นเอง

วิธีการเช่นนี้จะทำให้วงจรตรวจจับความถี่สามารถหาค่าความถี่ของสัญญาณที่เข้ามาได้ ซึ่งก็จะทำให้ทราบถึงค่าของความถี่ที่ประกอบกันขึ้นเป็นสัญญาณ DTMF ได้ และทำการถอดรหัสออกมาเป็นหมายเลขโทรศัพท์ได้ในที่สุด



รูปที่ 2.20 บล็อกไอซีแอมป์ของวงจรถอดรหัสหมายเลขแบบ DTMF

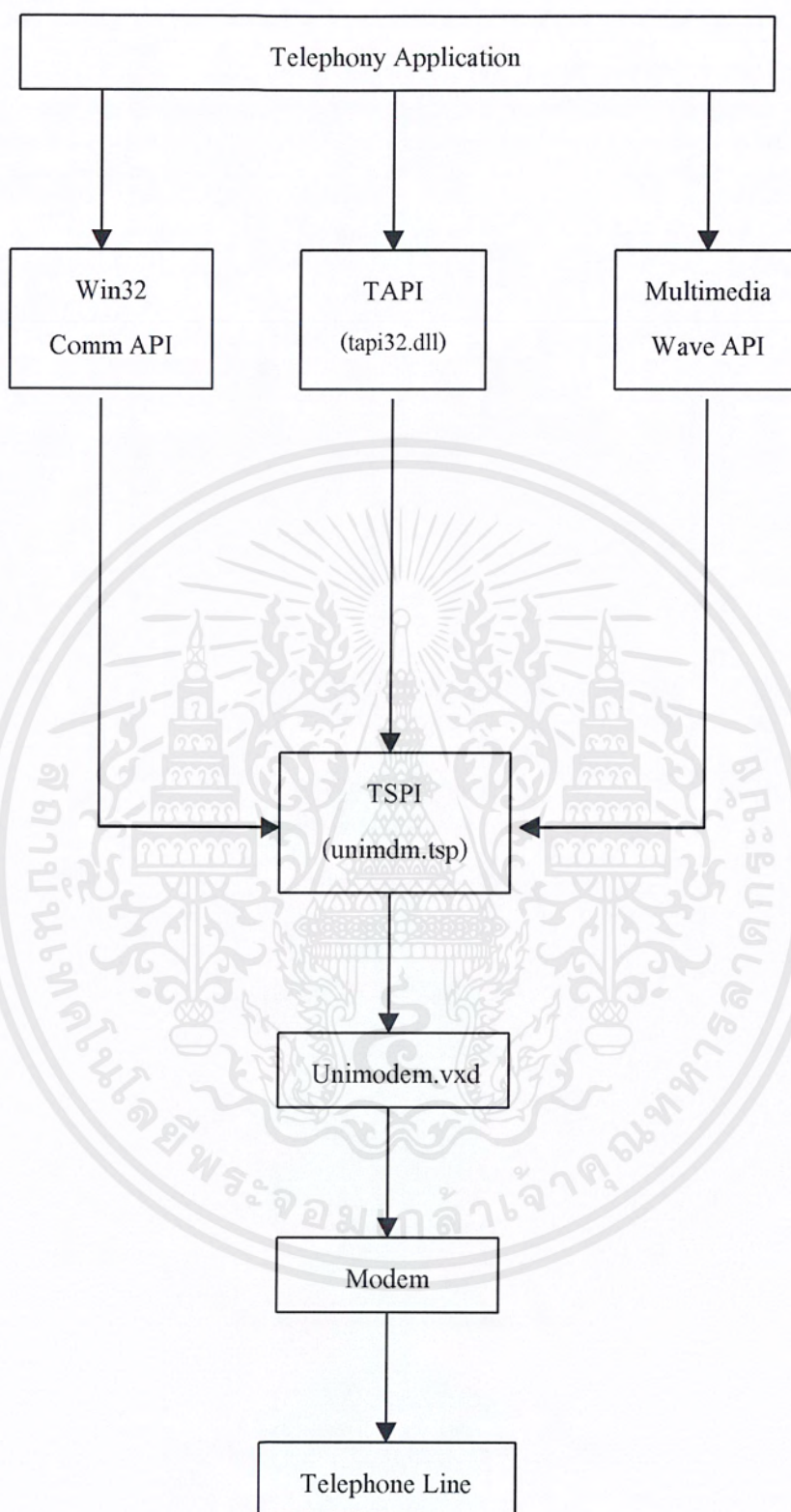
แต่ปัญหาประการสำคัญของผู้ออกแบบวงจรถอดรหัสก็คือ การที่มีเสียงพูดเข้ามาในวงจร ซึ่งไม่ใช่ความถี่ DTMF ซึ่งอาจทำให้เกิดความผิดพลาดในการถอดรหัสหมายเลขได้ ดังนั้นจึงต้องมีการกำหนดระยะเวลาในการประมวลผลแต่ละครั้งไว้ประมาณ 10 มิลลิวินาที ซึ่งถ้าเวลาในการประมวลผลน้อยกว่านี้จะทำให้เกิดความผิดพลาดในการถอดรหัสได้

2.3.15 โครงสร้างการติดต่อกับ Modem

Voice Modem ที่ทำงานอยู่บนระบบปฏิบัติการ Windows จะมี Software Driver ควบคุมการทำงานอยู่ โดย Driver นั้นจะสนับสนุน Function มาตรฐานบน Windows 3 กลุ่มด้วย คือ Win32 Comm API, Telephony Application Programming Interface (TAPI) และ Multimedia Wave API ในการทำงานของโปรแกรมจะไม่มีคำสั่ง AT Command ไปยัง Driver ของ Modem โดยตรง แต่จะใช้การเรียก Function จาก TAPI แทน ซึ่ง TAPI คือ กลุ่มของ Function ใน Win32 API ที่ทำหน้าที่จัดการเกี่ยวกับอุปกรณ์ด้านโทรศัพท์ Function เหล่านี้จะอยู่ใน Library ของ TAPI ซึ่งจะมีการเก็บไว้ทั้งในลักษณะที่เป็น Static Library (tapi32.lib) และ Dynamic Link Library (tapi32.dll)

จากที่กล่าวมาจะเป็นการทำงานกับส่วนที่เป็นโทรศัพท์เท่านั้น แต่ถ้าต้องการส่งข้อมูลออกทางสายโทรศัพท์แล้ว จะไม่สามารถใช้ TAPI ในการทำงานส่วนนี้ได้เนื่องจาก TAPI จะจัดการกับอุปกรณ์ทางด้านโทรศัพท์เท่านั้น จึงมีความจำเป็นที่จะต้องดึงความสามารถของ API อื่นๆเข้ามาเกี่ยวข้องด้วย อย่างไรก็ตามโครงการนี้ต้องการส่งข้อมูลออกทางสายโทรศัพท์ในรูปแบบของเสียง ดังนั้นจึงนำ Function ใน Multimedia Wave API มาช่วยทำงานในส่วนที่เป็นการเล่นเสียงไปยังโทรศัพท์ได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.21 แสดงโครงสร้างการติดต่อโมเด็ม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.3.16 เครื่องมือที่ใช้พัฒนาด้านการติดต่อสื่อสารกับระบบโทรศัพท์

จากที่ได้ศึกษาวิธีการติดต่อกับ Modem ที่ทำงานบนระบบปฏิบัติการ Windows พบว่าการจะควบคุมให้ Modem ทำงานจัดการเกี่ยวกับสัญญาณโทรศัพท์จำเป็นต้องใช้ TAPI เป็น Function หลักในการควบคุมการทำงานกับโทรศัพท์ และนำความสามารถในการเล่นไฟล์เสียงของ Multimedia Wave API เข้าช่วยในการเล่นเสียงตอบรับ เมื่อได้ศึกษาถึงความสามารถของ Compiler นั้น Visual C++ จึงมีความเหมาะสมที่จะรองรับการทำงานในสองส่วนนี้ได้เป็นอย่างดี

Telephony Application Programming Interface (TAPI) คือหนึ่งในชุด API ที่สำคัญที่สุดที่ถูกนำออกเผยแพร่ โดย Microsoft ซึ่งตัว Telephony API ก็คือชุดฟังก์ชัน call ชุดหนึ่งที่ยอมให้โปรแกรมเมอร์จัดการ และประสานการติดต่อสื่อสารชนิดต่างๆ ระหว่าง PC และสายโทรศัพท์ (telephony line) TAPI ตั้งชุดของ call ที่เป็นแบบเดียว(uniform) สามารถนำไปดัดแปลงกับฮาร์ดแวร์ชนิดต่างๆ ที่มี TAPI-compliant service provider interface (SPI) เอาไว้ให้บริการกับ Application ที่เรียกใช้งานฮาร์ดแวร์ผ่านทาง TAPI

2.3.17 การเชื่อมต่อ Application กับ TAPI

การเรียกใช้งาน TAPI จาก Application จะต้องทำตามขั้นตอนดังนี้
ตารางที่ 2.8 แสดงขั้นตอนในการเรียกใช้งาน TAPI จาก Application

ขั้นที่	TAPI Function	จุดประสงค์
1.	<i>LineInitialize/Ex</i>	เพื่อทำการเชื่อมต่อกันระหว่าง Application และ TAPI Server
2.	<i>LineNegotiateAPIVersion</i>	ทำการตรวจสอบ version numbers ของโมเด็มทุกตัวที่ติดตั้งอยู่ในระบบ เนื่องจากการทำงานในระบบตอบรับต้องใช้ TAPI Version 1.4 ขึ้นไป
3.	<i>LineGetDevCaps</i>	ทำการตรวจสอบความสามารถของอุปกรณ์โมเด็ม เพื่อตรวจสอบอุปกรณ์โมเด็มว่าสนับสนุนการทำงานแบบเสียงตอบรับหรือไม่
4.	<i>LineGetAddressCaps.</i>	ตรวจสอบ Address ของอุปกรณ์โมเด็ม เนื่องจากโมเด็มบางชนิดมีจุดเชื่อมต่อกับโทรศัพท์ได้มากกว่า 1 ที่(Address) ในอุปกรณ์ 1 ชิ้น (ไม่มีขายตามท้องตลาด ต้องสั่งซื้อจากบริษัทขายระบบตอบรับเท่านั้น)
5.	<i>LineOpen</i>	เป็นการเปิดใช้อุปกรณ์โมเด็ม โดยเลือกกำหนดอุปกรณ์โมเด็มที่จะเปิดใช้ และกำหนดลักษณะการทำงาน เช่น เป็นการตอบรับโดยใช้เสียง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ชั้นที่	TAPI Function	จุดประสงค์
6.	<i>LineAnswer</i>	เปรียบได้กับการยกหูโทรศัพท์ขึ้นมาเตรียมจะตอบรับ
7.	<i>LineGetID</i>	ค้นหาหมายเลขของอุปกรณ์(Device ID)ในโมเด็มที่ใช้เล่นเสียงไฟล์เสียง
8.	<i>LineShutdown</i>	ปิดการใช้งานอุปกรณ์โมเด็ม การเรียกใช้ <i>lineShutdonw</i> จะทำให้Application หยุดการเชื่อมต่อกับ TAPI Server

ทั้ง 8 ชั้นตอนนี้ จะเป็นการเริ่มต้นทำงานกับ TAPI และหยุดการทำงานกับ TAPI แต่ยังมีการทำงานใน TAPI ส่วนอื่นๆ ที่เกิดขึ้นตามการทำงานของอุปกรณ์โมเด็ม เช่น การโทรเข้า การวางหู การกดปุ่มบนเป็นโทรศัพท์ เป็นต้น สิ่งต่างๆ เหล่านี้จะมีการทำงานแยกออกไปซึ่งจะกล่าวไว้ในส่วนของ TAPI Handle และ TAPI Message

TAPI Handle เปรียบเสมือน Record ที่เก็บข้อมูลของ Call อีกนัยหนึ่งก็คือเก็บข้อมูลการติดต่อกับระบบโทรศัพท์ ณ จุดหนึ่งๆ เมื่อต้องทำการติดต่อผ่านทางโมเด็ม ใน TAPI จะมี Handle อยู่ 3 ชนิดดังนี้

ตารางที่ 2.9 แสดง Handle 3 ชนิดของ TAPI

	ประเภท	คำอธิบาย
1.	HXxxx handle	เป็น handle ที่สร้างขึ้นจากส่วนของ TAPI แล้วส่งให้กับ Application
2.	HtXxxx handle	เป็น handle ที่สร้างขึ้นจาก TAPI แล้วส่งให้กับ TSP เมื่อ TSP ต้องการส่งข้อมูลที่เกิดขึ้นจากอุปกรณ์โมเด็มจะใช้ handle นี้เป็นตัวบอกว่าสิ่งที่เกิดขึ้นนั้นเกิดขึ้นกับส่วนไหนใน TAPI โดยการระบุ handle ลงไป
3.	HdXxxx handle	เป็น handle ที่สร้างขึ้นจาก TSP แล้วส่งให้กับ TAPI ใช้อธิบายของค์ประกอบต่างๆ ใน TSP และ TAPI จะอ้างอิง handle นี้เมื่อมีการติดต่อกับ TSP

TAPI Message เปรียบเสมือนข้อความที่ส่งมาจาก TAPI ให้กับ Application เพื่อบอกให้รู้ว่าขณะนี้เกิดอะไรขึ้นกับอุปกรณ์โมเด็ม เช่น มีโทรศัพท์เข้า ผู้โทรเข้าวางหูแล้ว หรือการกดปุ่มบนเป็นโทรศัพท์ เป็นต้น Message นี้มักส่งออกมาคู่กับ handle ของ Call หรือ handle ของอุปกรณ์โมเด็มเพื่อบอกให้รู้ถึงที่มาของ Message ด้วย เมื่อ Application รู้ที่มาและสิ่งที่เกิดขึ้นกับอุปกรณ์โมเด็มแล้ว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ทำให้สามารถตัดสินใจได้ว่าทำงานอะไรกับอุปกรณ์โมเด็มเป็นขั้นต่อไป เช่น หลักจากมีการโทรศัพท์เข้ามาเมื่อมีสัญญาณดัง 2 ครั้งให้รับโทรศัพท์ เป็นต้น

ตารางที่ 2.10 แสดง TAPI Message ที่เกี่ยวข้องกับระบบ

Main Message	คำอธิบาย
LINE_APPNEWCALL	TAPI จะส่ง Message นี้มายัง Application เพื่อแจ้งว่ามีสัญญาณโทรเข้าดังเข้ามายังโมเด็ม
LINE_DEVICESTATE	TAPI จะส่ง Message นี้เข้ามายัง Application เพื่อบอกให้รู้ว่ามี การเปลี่ยนแปลงเกิดขึ้นกับอุปกรณ์โมเด็ม เช่น เสียงกริ่งของโทรศัพท์ Message นี้จะส่งมาพร้อมกับ Message ย่อยที่แสดงถึงว่าการเปลี่ยนแปลงนั้นคืออะไร
LINE_CALLSTATE	หลังจากที่ Application รับโทรศัพท์แล้วจะถือว่า Call ที่เข้ามานั้นทำงานและเมื่อมีการเปลี่ยนแปลงเกิดขึ้นกับ Call ทาง TAPI Server จะแจ้งกับทาง Application โดยส่ง Message LINE_CALLSTATE ให้กับ Application พร้อมกับ Message ย่อยที่บอกถึงการเปลี่ยนแปลง
LINE_MONITORDIGITS	TAPI Server จะส่ง Message นี้ให้กับ Application เมื่อมีการกดปุ่มบนเป็นโทรศัพท์ แต่จะต้องเปิดใช้ความสามารถส่วนที่ใส่ฝ้าตัดฟังการกดปุ่มของอุปกรณ์โมเด็มก่อนจึงจะมีการส่ง Message นี้ และ Message นี้จะส่งมาพร้อมกับค่าของปุ่มที่กดเป็นตัวอักษร

ข้อมูลชนิดวาเรียนท์เป็นข้อมูลที่มีลักษณะพิเศษคือ สามารถเปลี่ยนแปลงขนาดของหน่วยความจำที่ใช้เก็บข้อมูลได้โครงสร้างข้อมูลของข้อมูลชนิดวาเรียนท์ในภาษา C/C++ เป็นดังนี้

```
Struct VARDATA
{
    DWORD dwTotalSize;
    DWORD dwNeededSize;
    DWORD dwUsedSize;
};
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โครงสร้างของข้อมูลชนิดการเรียนรู้ที่ประกอบด้วยฟิลด์ข้อมูลพื้นฐานอยู่ 3 ฟิลด์ด้วยกัน สามารถอธิบายหน้าที่ของทั้ง 3 ฟิลด์ได้ดังตาราง

ตารางที่ 2.11 แสดงหน้าที่ของฟิลด์ข้อมูลพื้นฐานทั้ง 3 ฟิลด์

ชื่อฟิลด์	คำอธิบาย
DwTotalSize	ใช้เก็บจำนวน Bytes ทั้งหมดที่ใช้ตัวแปรนี้คือรวมเนื้อที่ที่ใช้เก็บค่าของทั้ง 3 ฟิลด์นี้ด้วย
DwNeededSize	ใช้เก็บจำนวน Bytes ที่ต้องการใช้ในการเก็บค่าของข้อมูลที่ได้รับกลับมาจากการเรียกใช้ฟังก์ชัน เป็นเหมือนค่าประมาณว่าจะใช้เก็บข้อมูลจริงกี่ Bytes
DwUsedSize	เป็นจำนวน Bytes ที่ใช้จริงในการเก็บข้อมูลที่เป็นประโยชน์ต่อการทำงานเมื่อทำการปรับเปลี่ยนค่าเรียบร้อยแล้วขนาดของหน่วยความจำที่ใช้เก็บข้อมูลจะเท่ากับค่านี้

ทั้งสามฟิลด์นี้เป็นเพียงฟิลด์พื้นฐาน ที่ใช้ควบคุมขนาดหน่วยความจำที่ใช้ในการเก็บข้อมูล เมื่อทำการส่งข้อมูลชนิดการเรียนรู้แบบอ้างอิงไปให้กับ TAPI Server จากนั้นทางด้าน TAPI Server จะทำการเติมข้อมูลที่ Application ต้องการลงในตัวแปรการเรียนรู้ซึ่งจะมีโครงสร้างที่แตกต่างกันไปตามฟังก์ชันที่ Application เรียกใช้ ตัวอย่างเช่น ถ้าใน Application เรียกใช้ฟังก์ชัน lineGetDevCaps TAPI Function ข้อมูลที่คืนกลับทางตัวแปรการเรียนรู้จะมีโครงสร้างแบบหนึ่งซึ่งเรียกว่า LINEDEVCAPS ถ้าใน Application เรียกใช้ lineGetID TAPI Function ข้อมูลที่คืนกลับมาในตัวแปรการเรียนรู้จะมีโครงสร้างอีกแบบหนึ่งเรียกว่า VARSTRING เป็นต้น

ตัวอย่างโครงสร้างข้อมูลชนิด VARSTRING โครงสร้าง VARSTRING ใช้ในการคืนค่าสตริงที่มีขนาดไม่แน่นอน โครงสร้างข้อมูลแบบนี้ถูกใช้ในการรับค่าจากการตรวจสอบหมายเลขอุปกรณ์ด้วยฟังก์ชัน lineGetID และสามารถอธิบายลักษณะของโครงสร้างได้ดังนี้

```
typedef struct varstring_tag {
    DWORD dwTotalSize;
    DWORD dwNeededSize;
    DWORD dwUsedSize;
    DWORD dwStringFormat;
    DWORD dwStringSize;
    DWORD dwStringOffset;
}
VARSTRING, FAR *LPVARSTRING;
```

2.3.18 การทำงานของ Multimedia API

ความต้องการของระบบทดสอบคือต้องการที่จะทดสอบให้กับผู้สอบถามในรูปแบบของเสียง แต่ทว่าการทำงานของ TAPI จะเป็นการควบคุมในส่วนของการติดต่อกับอุปกรณ์โมเด็มในลักษณะของการเชื่อมต่อโทรศัพท์เท่านั้น การที่จะเล่นไฟล์เสียงผ่านอุปกรณ์โมเด็มออกไปยังสายโทรศัพท์ได้จะต้องใช้ API ที่ควบคุมด้านเสียงเข้าช่วยในส่วนนี้ ซึ่ง API ที่ใช้ควบคุมด้านเสียงนี้เป็นส่วนหนึ่งของ Multimedia API

การนำ Multimedia API มาใช้งานร่วมกับ TAPI จะมีจุดเชื่อมต่อคือฟังก์ชัน lineGetID ดังที่กล่าวไปแล้วในตารางขั้นตอนการติดต่อกับ TAPI Server คือจะทำการค้นหาหมายเลขอุปกรณ์ที่ใช้สำหรับเล่นเสียงที่ติดตั้งอยู่ในโมเด็ม แต่ทว่าการทำงานของฟังก์ชันนี้ไม่ได้ค้นหาได้เฉพาะหมายเลขอุปกรณ์ที่ใช้เล่นเสียงเท่านั้นแต่ยังสามารถค้นหาหมายเลขอุปกรณ์ชนิดอื่นๆ ได้อีกด้วย อุปกรณ์แต่ละประเภทนั้นจะเรียกว่า Device class ซึ่งจะแบ่งเป็น Device class มาตรฐานได้ดังนี้

ตารางที่ 2.12 แสดง Device Class มาตรฐานของ Multimedia API

Device Class Name	Description
Comm	การสื่อสารผ่าน Communicationport
Comm/datamodem	การสื่อสารผ่าน Communicationport ไปยัง Modem
Comm/datamodem/portname	ชื่อของอุปกรณ์ที่โมเด็มเชื่อมต่ออยู่
Wave/in	รับข้อมูลในลักษณะของเสียง
Wave/out	ส่งข้อมูลออกในลักษณะของเสียง
Midi/in	รับข้อมูลเข้าในลักษณะของเสียงเครื่องดนตรี
midi/out	ส่งข้อมูลออกในลักษณะของเสียงเครื่องดนตรี
Tapi/line	อุปกรณ์ตามมาตรฐานของ Line device ใน TAPI
Tapi/phone	อุปกรณ์ตามมาตรฐานของ Phone device ใน TAPI
ndis	อุปกรณ์ทางด้าน Network
Tapi/terminal	อุปกรณ์ Terminal

สำหรับอุปกรณ์ที่สนับสนุน TAPI อื่นๆ อาจมี Device class ต่างจากนี้ได้ซึ่งจะขึ้นอยู่กับอุปกรณ์นั้นๆ ว่าได้ถูกออกแบบมาสำหรับทำงานด้านใด การใช้เสียงตอบผ่านอุปกรณ์โมเด็มต้องใช้ Device class ประเภท wave/out ในขณะที่เรียก lineGetID ทำให้ได้หมายเลขอุปกรณ์ที่ใช้เล่นเสียงผ่านทางโมเด็มกลับคืนมาเพื่อนำหมายเลขอุปกรณ์นี้ไปใช้ใน Multimedia API อีกครั้งหนึ่ง

2.3.19 ดักฟังการกดปุ่มโทรศัพท์

การกดปุ่มโทรศัพท์แต่ละครั้งจะเกิดคลื่นเสียงที่มีความถี่แตกต่างกันส่งเข้ามายังอุปกรณ์โมเด็ม Service Provider หรือซอฟต์แวร์ควบคุมโมเด็มที่มีความสามารถในการดักจับสัญญาณการกดปุ่มโทรศัพท์ สามารถเปิดใช้ความสามารถนี้ได้ด้วย MonitorDigit() TAPI Function

เมื่อมีการกดปุ่มโทรศัพท์เกิดขึ้นจะเป็นหน้าที่ของ Service Provider ในการแจ้งเข้ามายัง TAPI Server ถึงการเปลี่ยนแปลงที่เกิดขึ้น และเป็นหน้าที่ของ TAPI Server ในการกระจายข่าวสารไปยัง Application ต่างๆ ให้รู้ถึงการกดปุ่มและนำค่าของปุ่มนั้นไปประมวลผลซึ่งจะอยู่ในลักษณะของตัวอักษร

2.3.20 การใช้งาน Microsoft Agent

Microsoft Agent เป็นเครื่องมืออย่างหนึ่ง ที่ช่วยเพิ่มความสามารถในการอ่านข้อความให้กับแอปพลิเคชัน โดยจะต้องมีการเพิ่มคอนโทรลที่มีชื่อว่า Microsoft Agent Control 2.0 โดยการเลือกคำสั่ง References จากเมนู Project นอกจากนั้นจะต้องติดตั้งโปรแกรม Microsoft Agent Core Component ซึ่งเป็นโปรแกรมที่ใช้ในการอ่านตัวอักษรเท่านั้น จะต้องมีการติดตั้งแอนิเมชันเพิ่มเติม ซึ่งปัจจุบัน Microsoft Agent ได้เตรียมไว้ให้ใช้ 4 ตัว ได้แก่ Genie Merlin Robby และ Peedy ซึ่งเวลาติดตั้งแอนิเมชันแต่ละตัวจะอยู่ใน Windows\MsAgent\Chars โดยแต่ละตัวก็จะมีกิริยาท่าทางการแสดงออกที่แตกต่างกัน

ตัวอย่างขั้นตอนการเขียน โปรแกรม Microsoft Agent

1. สร้างตัวแปรพบกลไกในส่วน General Declarations ดังนี้

```
Dim WithEvents CoreAgent As Agent
Dim Assistant As IAgentCtlCharacterEx
```

2. เปิด Agent โดยการสร้างออปเจกต์ขึ้นมาใหม่ แล้วโหลดตัวแอนิเมชัน จากนั้นกำหนดออปเจกต์ Assistant ให้กับแอนิเมชันที่สร้างขึ้น

```
Set CoreAgent = New Agent
CoreAgent.Characters.Load "Assistant", "ตัวแอนิเมชัน.acs"
Set Assistant = CoreAgent.Characters ("Assistant")
```

3. กำหนดลักษณะกิริยาการแสดงผลทางคำพูดให้กับแอนิเมชัน ซึ่งถ้าต้องการให้แอนิเมชันเปล่งเสียงได้ต้องมีการติดตั้งโปรแกรม Text-to-Speech ด้วย

With Assistant

```
.show
.speak ("Hello! Welcome to Internet Voice Message")
.think ("I Love You")
.play ("Read")
.hide
```

End With

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4. สามารถที่จะเปลี่ยนฟอนต์ใน speech balloon ได้เขียนโค้ดดังนี้

```
Assistant.Balloon.FontName = "MS Sans serif"
```

5. ตัวแอนิเมชันแต่ละตัวมีความสามารถในการแสดงกิริยาท่าทางที่แตกต่างกัน แต่สามารถเขียนโปรแกรมเพื่อแสดงรายการกิริยาท่าทางที่แอนิเมชันแต่ละตัวสามารถที่จะแสดงได้ โดยสามารถเขียนโค้ดได้ดังต่อไปนี้

```
Dim Animation As Variant
```

```
For Each Animation In Assistant.AnimationNames
```

```
MsgBox Animation
```

```
Next
```

2.3.21 ความหมายของ Data Access Object (DAO)

DAO เป็นตัวแปรชนิดพิเศษที่ Visual Basic 6.0 มีมาเพื่อให้สามารถติดต่อกับฐานข้อมูล DAO ถ้าจะมองให้เข้าใจจริง ๆ ก็เปรียบเสมือนกับเป็นชนิดของตัวแปรแบบหนึ่ง เหมือนอย่าง Integer หรือ String แต่มันสามารถที่จะทำหน้าที่เสมือนเป็นตัวแทนของฐานข้อมูล คิวรี ที่อยู่ในดิสก์ หรือในหน่วยความจำก็ได้ โดยที่สามารถทำการค้นหาข้อมูลได้โดยไม่ต้องแสดงข้อมูลบนหน้าจอ

นอกจากนี้ก็ยังมีย่อปลั๊กอินอีกประเภทหนึ่งที่จะใช้คือ RDO (Remote Data Object) ซึ่งเอาไว้ติดต่อกับฐานข้อมูลที่อยู่บน Database Server

สำหรับส่วนประกอบต่าง ๆ พื้นฐานของ DAO ก็จะมี DbEngine, Workspace, Database, Recordset

สิ่งที่ DAO จะมีความแตกต่างทางด้านการทำงานเมื่อเปรียบเทียบกับการใช้ Data Control ทำงาน กับฐานข้อมูลคือ DAO จะยังไม่เปิดฐานข้อมูลขึ้นมาตรวจใดที่ยังไม่มีการสั่งให้เปิดขึ้นมา ฉะนั้นเวลาเริ่มทำงานเราก็ต้องสั่งให้เปิดฐานข้อมูลขึ้นมาก่อนจะเริ่มทำงานใด ๆ ต่อไป

โดยปรกติการที่เราจะทำการควบคุมหรือจัดการใดๆ กับฐานข้อมูลก็จะต้องมี DBMS แต่ว่าเราสามารถจะใช้ Jet Engine เข้ามาทำงานแทนได้ โดยเราจะทำการติดต่อกับ Jet Engine ผ่านทาง Object DbEngine ซึ่งตัว DbEngine นี้ ก็จะมีการแบ่งพื้นที่การทำงานออกเป็น Workspace โดยเราสามารถเลือกได้ว่าจะให้ Database ของเราทำงานกับ Workspace อันไหนก็ได้ เมื่อประกาศอ็อบเจกต์ที่จะใช้กับ Workspace นั้นเรียบร้อยแล้ว เราก็จะทำการเปิดฐานข้อมูลโดยใช้ Object ตัวนั้นในการเปิดฐานข้อมูลต่อไป

เมื่อเราต้องการที่จะใช้งานข้อมูลใดๆในตารางเราจะต้องเปิด Recordset ขึ้นมาทำงานเสียก่อน เพื่อให้อ้างอิงกับ Table, Field, Record ใดๆที่อยู่ในฐานข้อมูลนั้นๆ เช่นถ้าต้องการจะนับจำนวน Record ที่มีอยู่ในตารางนั้นหรือจะตั้งให้เก็บ Record ใดๆที่ตรงตามเงื่อนไขที่เกิดจากการทำด้วย Query เป็นต้น

สำหรับการใช้งานของ Recordset นี้เรายังแบ่งการทำงานออกเป็น 3 ประเภทคือ

- 1) Recordset แบบ Table Recordset แบบนี้จะทำการจัดการข้อมูลในตารางได้เพียงทีละตารางเท่านั้นและใช้วิธีการเข้าถึงข้อมูลในตาราง โดยใช้การอ้างอิงจาก Index ซึ่งทำให้สามารถเข้าถึงข้อมูลได้เร็ว แต่จะมีข้อเสียตารางที่ใช้จัดการข้อมูลเป็นบางส่วนของตารางไม่ได้
- 2) Recordset แบบ Dynaset การใช้ Recordset แบบนี้จะทำให้สามารถเข้าถึงข้อมูลได้ช้ากว่า แต่ก็มีความยืดหยุ่นคือ สามารถจัดการกับข้อมูลได้ทีละหลายๆตาราง โดยเราสามารถเลือกดึงข้อมูลเฉพาะบาง Record หรือบาง Field ที่ต้องการจากหลายๆตารางได้โดยใช้คำสั่ง SQL
- 3) Recordset แบบ Snapshot จะจัดการข้อมูลทีละหลายๆตารางได้เหมือนกับแบบ Dynaset แต่จะทำงานได้เร็วกว่าแต่เราไม่สามารถทำการแก้ไขข้อมูลได้

2.3.22 การแปลงไฟล์ Wav เป็นไฟล์ MP3

การ Upload ไฟล์ไปยังเครื่องเซิร์ฟเวอร์บางครั้งไฟล์ Wav ที่ส่งไปมีขนาดใหญ่มาก การ Upload จึงใช้เวลานานและเสี่ยงต่อความผิดพลาดในขณะการ Upload ไฟล์ ดังนั้นการแปลงไฟล์เป็นไฟล์ MP3 จึงเป็นทางเลือกที่น่าสนใจ เพราะคุณภาพเสียงมีคุณภาพที่ใกล้เคียงกับไฟล์ Wav แต่มีขนาดที่ต่างกันมากคือไฟล์ MP3 มีขนาดเล็กกว่าไฟล์ Wav ประมาณถึง 10 เท่า Visual Basic 6.0 มีคอมโพเนนต์ที่มีความสามารถในการแปลงไฟล์ Wav เป็นไฟล์ MP3 คือ Dialog-Median MP3Enc Active X Control Module (MP3Enc) ซึ่งสามารถเรียกใช้ได้โดยการเลือก Project เมนู เลือก Component และเลือก Dialog-Median MP3Enc Active X Control Module

ตัวอย่างขั้นตอนการเขียนโค้ดกับคอมโพเนนต์ MP3Enc

```
Public Sub EncodeToMP3()
```

```
    Mp3.AllowDownSample = True
```

```
    Mp3.DownMix = True
```

```
    Mp3.BitRate = 32000
```

```
    inputwav = App.Path & "\Voice\" & Form1.tel.Text & ".wav"
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
outputmp3 = App.Path & "\Voice\" & Form1.tel.Text & ".mp3"
```

```
Mp3.Open inputwav, outputmp3
```

```
Mp3.Encode
```

```
End With
```

```
End Sub
```

ส่วนที่สำคัญๆก็คือ BitRate โดยปรกติจะกำหนดไว้ที่ 32000 bits/s และที่ Open ตรง inputwav ก็จะใส่เป็นค่า path ของไฟล์ Wav ที่ต้องการแปลง ส่วน outputmp3 จะเป็น path พร้อมกำหนดชื่อไฟล์เป็นตำแหน่งที่บันทึกเป็นไฟล์ MP3 โดยหลักๆ โค้ดเพียงเท่านี้ก็เพียงพอสำหรับการแปลงไฟล์ Wav เป็นไฟล์ MP3

2.3.23 การจัดการระบบไฟล์ด้วย File System Object

ใน Visual Basic 6.0 มีระบบออบเจกต์แบบใหม่ที่เรียกว่า File System Object Model ที่ช่วยจัดการเกี่ยวกับระบบไฟล์ โดยการใช้งานจะต้องอ้างอิงถึง Microsoft Script Runtime ตรงคำสั่ง Project>Reference โดยโครงสร้างของ File System Object จะประกอบด้วยออบเจกต์ต่างๆดังตารางต่อไปนี้

การเข้าถึงออบเจกต์ Drive Folder และ File

GetDrive ใช้เข้าออบเจกต์ไดร์ฟที่ส่งเป็นพารามิเตอร์ไป

GetFolder ใช้เข้าออบเจกต์โฟลเดอร์ที่เราได้ส่งเป็นพารามิเตอร์ไป

GetFile ใช้เข้าถึงออบเจกต์ไฟล์ที่เราได้ส่งเป็นพารามิเตอร์ไป

ตาราง 2.13 ตารางออบเจกต์ของ File System Object

ชื่อออบเจกต์	คำอธิบาย
Drive	เป็นออบเจกต์ที่จะให้เราเข้าถึงรายละเอียดต่างๆ ของไดร์ฟที่มีในระบบทั้งหมด โดยไดร์ฟที่ใช้ได้นั้นเป็นได้ตั้งแต่ฮาร์ดดิสก์จนถึงไดร์ฟของระบบเครือข่าย
Folder	เป็นออบเจกต์ที่ใช้ทำงานกับโฟลเดอร์
File	เป็นออบเจกต์ที่ใช้ทำงานกับไฟล์
FileSystemObject	เป็นออบเจกต์หลักของกลุ่มจะมีคุณสมบัติ Drives ที่เป็นคอลเลกชันที่อ้างอิงถึงไดร์ฟทั้งหมดในระบบ
TextStream	เป็นออบเจกต์ที่ทำงานเกี่ยวกับเท็กซ์ไฟล์โดยเฉพาะ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตาราง 2.14 คุณสมบัติของออบเจ็กต์ Drive

คุณสมบัติ	คำอธิบาย
TotalSize	เป็นขนาดของไดรฟ์ในหน่วยไบต์
FreeSpace	เป็นขนาดของไดรฟ์ที่เหลือในหน่วยไบต์
DriveLetter	เป็นอักขระที่เรากำหนดให้กับไดรฟ์ เช่น C , D เป็นต้น
DriveType	เป็นชนิดของไดรฟ์ เช่น Fixed Disk, CD-Rom เป็นต้น
SerialNumber	เป็น Serial Number ของไดรฟ์
FileSystem	เป็นชนิดของระบบไฟล์ที่ใช้เช่น FAT32 เป็นต้น
IsReady	จะบอกว่าไดรฟ์นั้นใช้ได้หรือไม่
VolumeName	เป็นชื่อ Volume ของไดรฟ์
RootFolder	เป็น Path ของโฟลเดอร์ราก

ตาราง 2.15 คุณสมบัติของออบเจ็กต์ Folder

คุณสมบัติ	คำอธิบาย
Files	เป็นคอลเลกชันที่เก็บไฟล์ในโฟลเดอร์นี้ทั้งหมด
SubFolders	เป็นคอลเลกชันที่เก็บโฟลเดอร์ย่อยของโฟลเดอร์นี้ทั้งหมด
IsRootFolder	ใช้ตรวจสอบว่าเป็นโฟลเดอร์รากหรือไม่

การอ้างอิงถึงไฟล์ที่ต้องการได้โดยใช้เมธอด GetFile ซึ่งจะทำให้เราใช้ออบเจ็กต์ File ทำงานกับไฟล์นั้นได้ ซึ่งออบเจ็กต์ File มีเมธอดที่สำคัญสำหรับการก๊อปปี้ ย้าย และลบไฟล์ดังตารางต่อไปนี้

ตาราง 2.16 เมธอดของออบเจ็กต์ File

การทำงาน	เมธอดที่ใช้
การก๊อปปี้ไฟล์	FileSystemObject.CopyFile
การลบไฟล์	FileSystemObject.DeleteFile
การย้ายไฟล์	FileSystemObject.MoveFile

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตัวอย่างการใช้งาน File System Object

```
Dim fso New FileSystemObject
Dim objFolder As Folder
Dim objFol1 As Folder
Dim objFile As File
Set objFolder = fso.GetFolder ("C:\ Windows")
Fso.DeleteFile "E:\config.sys"
```

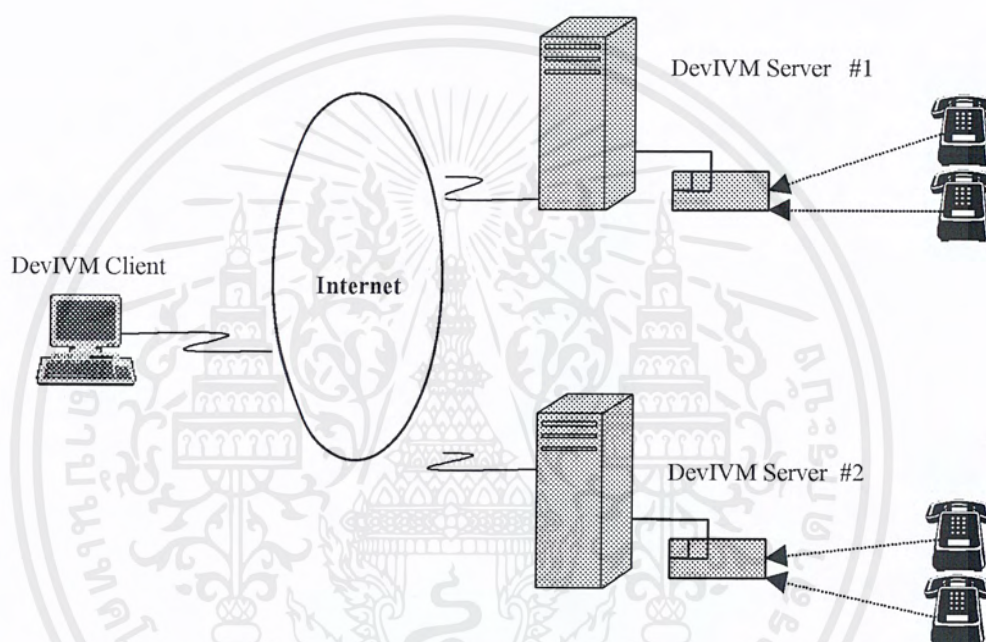


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 3 การศึกษาและพัฒนา

3.1 โครงสร้างระบบ

โครงสร้างของระบบฝากข้อความเสียงผ่านเครือข่ายอินเทอร์เน็ตแบ่งตามลักษณะโปรแกรมได้ออกเป็น 2 ส่วน คือ โครงสร้างส่วนไคลเอนท์ (DevIVM Client) และ โครงสร้างส่วนเซิร์ฟเวอร์ (DevIVM Server)



รูปที่ 3.1 โครงสร้างระบบฝากข้อความเสียงผ่านเครือข่ายอินเทอร์เน็ต

3.1.1 โครงสร้างส่วนไคลเอนท์ (DevIVM Client)

โครงสร้างส่วนนี้เป็นส่วนที่จะติดตั้งกับเครื่องคอมพิวเตอร์ลูกข่ายที่ต้องการให้ผู้ใช้งานสามารถฝากข้อความเสียงผ่านระบบอินเทอร์เน็ตได้ โดยโครงสร้างส่วนนี้มีหน้าที่อยู่สองส่วนสำคัญ คือ จัดการด้านข้อความเสียง และ จัดการส่งข้อความเสียงไปยังโครงสร้างส่วนเซิร์ฟเวอร์

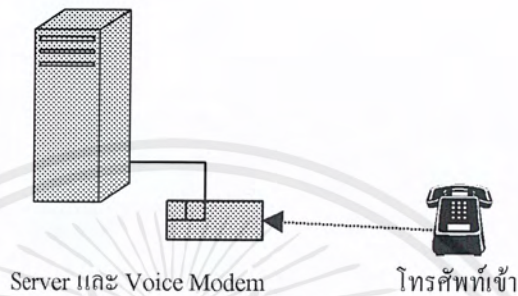


รูปที่ 3.2 โครงสร้างส่วนไคลเอนท์ (DevIVM Client)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.1.2 โครงสร้างส่วนเซิร์ฟเวอร์ (DevIVM Server)

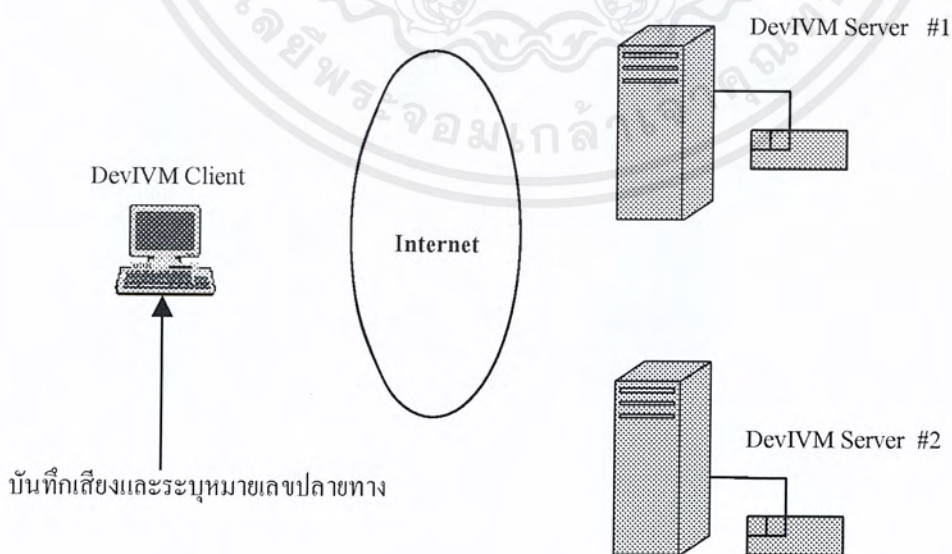
โครงสร้างส่วนนี้จะเป็นส่วนที่ติดตั้งกับเครื่องคอมพิวเตอร์แม่ข่าย ซึ่งจะใช้เป็นเครื่องที่เก็บข้อความเสียงจากผู้ส่งส่วนโคลอนท์ โดยผู้รับจะโทรศัพท์เข้ามารับฟังข้อความเสียงจากเครื่องแม่ข่ายที่อยู่ในพื้นที่เดียวกันกับโทรศัพท์ โครงสร้างส่วนเซิร์ฟเวอร์จะมีอุปกรณ์ที่เกี่ยวข้องกับระบบโทรศัพท์ คือ Voice Modem



รูปที่ 3.3 โครงสร้างส่วนเซิร์ฟเวอร์ (DevIVM Server)

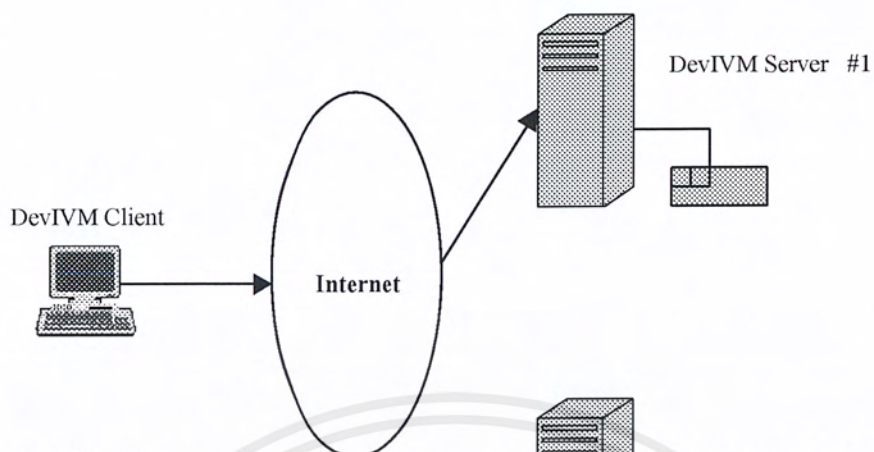
3.1.3 โครงสร้างการทำงาน

โครงสร้างทั้ง 2 ส่วนของระบบฝากข้อความเสียงผ่านเครือข่ายอินเทอร์เน็ตมีทั้งการทำงานในแต่ละส่วน และมีการทำงานที่เชื่อมโยงระหว่างกัน โดยผ่านเครือข่ายอินเทอร์เน็ต และส่วนระบบโทรศัพท์ภายนอกที่จะต้องเชื่อมโยงกับโครงสร้างส่วนเซิร์ฟเวอร์โดยรูปที่ 3.4 ถึง 3.6 จะแสดงถึงโครงสร้างการทำงานของระบบ



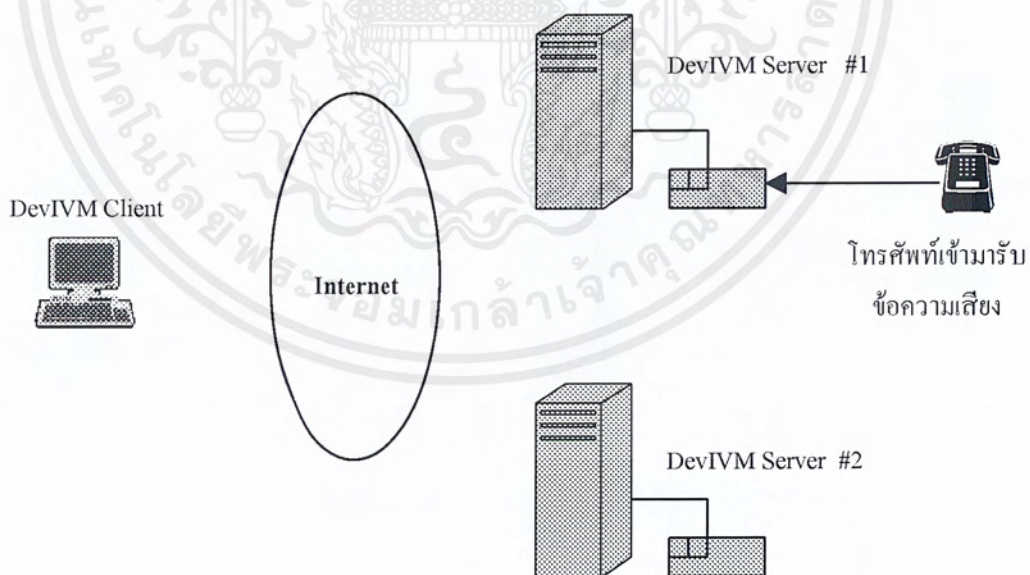
รูปที่ 3.4 โครงสร้างการทำงานของส่วนโคลอนท์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



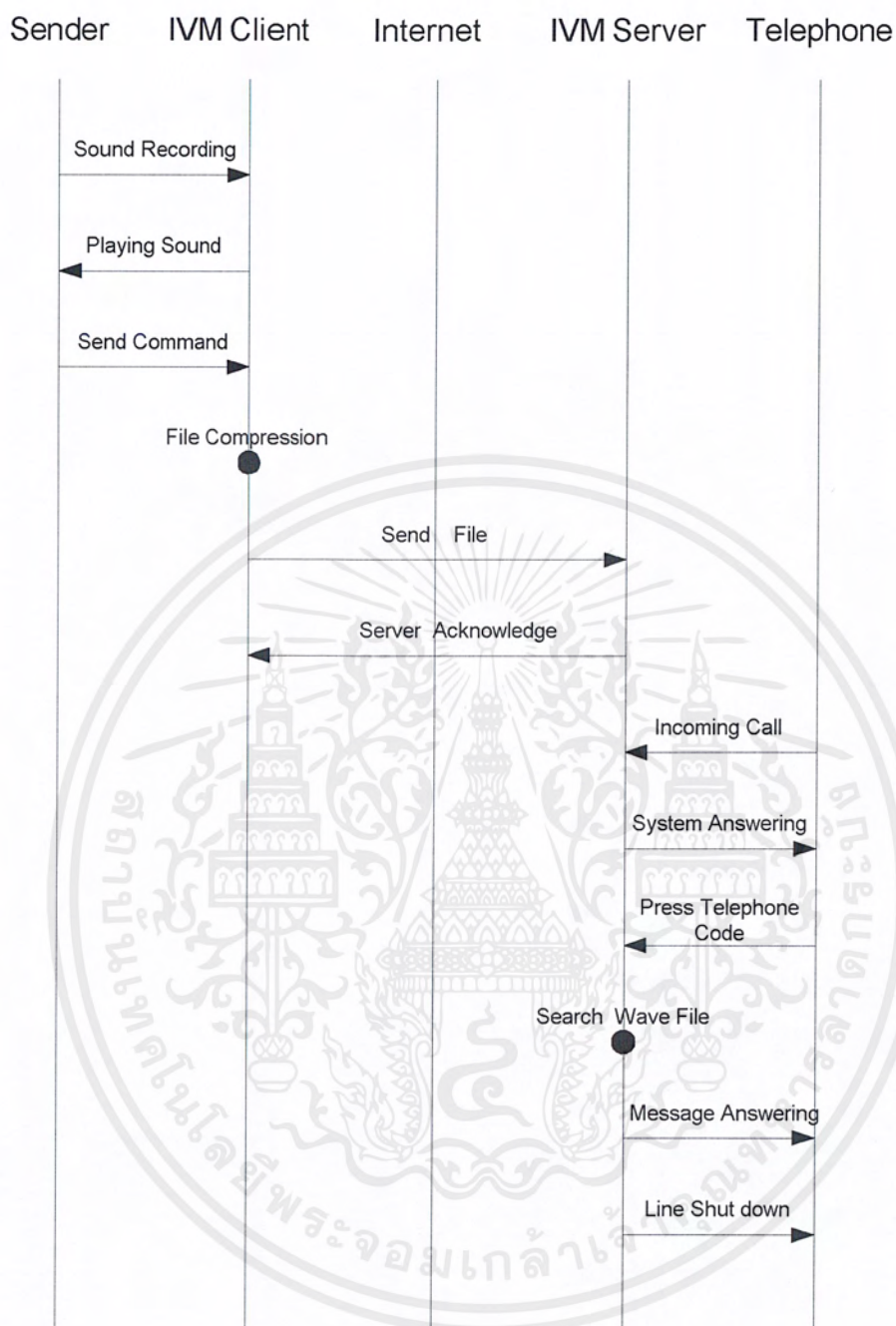
สร้างการติดต่อไปยัง Server
และส่งไฟล์เสียงผ่านอินเทอร์เน็ต

รูปที่ 3.5 โครงสร้างการทำงานของส่วนโคลนที่เชื่อมโยงผ่านเครือข่ายอินเทอร์เน็ตไปยังส่วนเซิร์ฟเวอร์



รูปที่ 3.6 โครงสร้างการทำงานของส่วนเซิร์ฟเวอร์ที่เชื่อมต่อกับระบบโทรศัพท์ภายนอก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

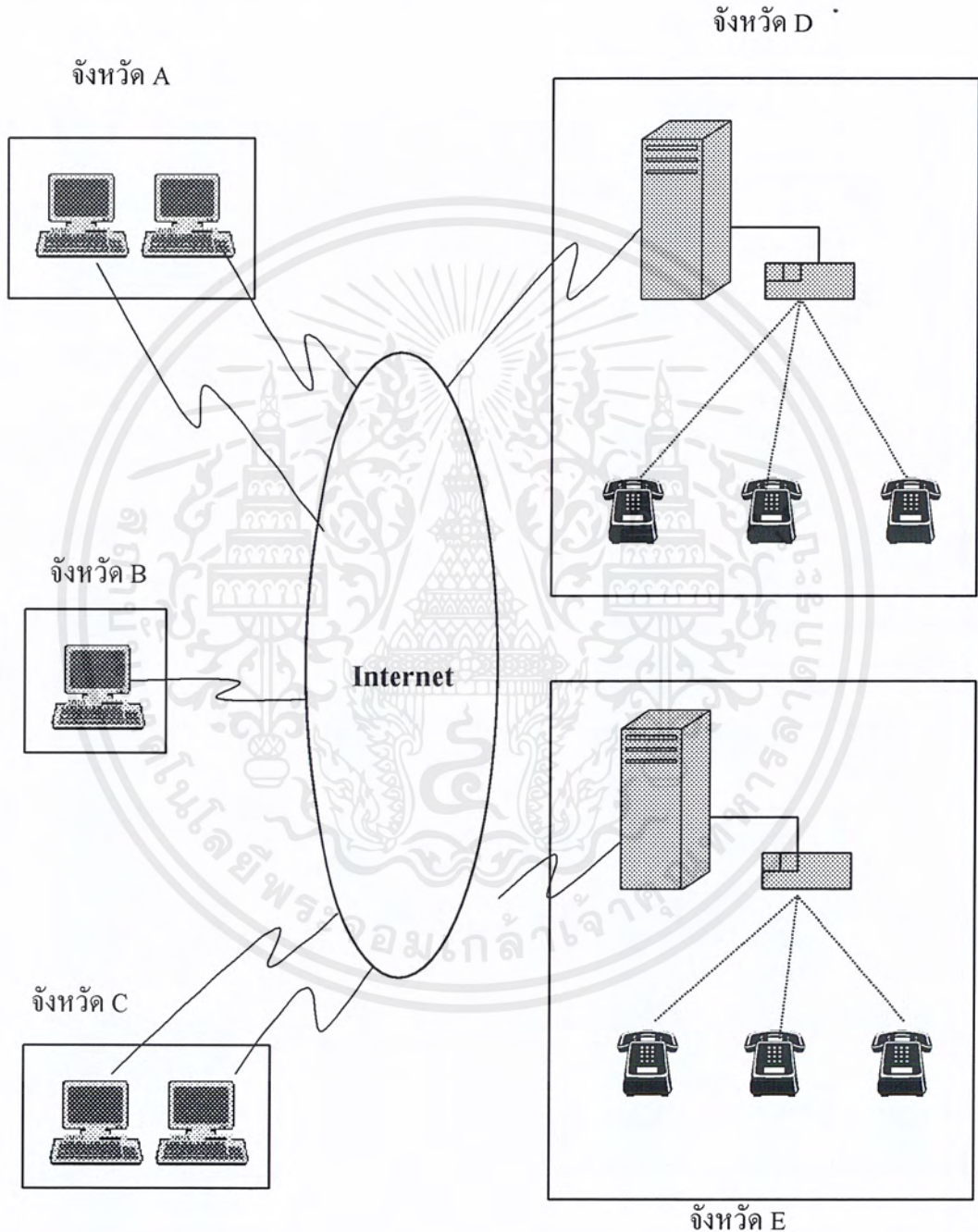


รูปที่ 3.7 Trace Diagram แสดงการทำงานของระบบฝากข้อความเสียงผ่านเครือข่ายอินเทอร์เน็ต

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.1.4 โครงสร้างระบบสื่อสารทางไกล

การพัฒนาระบบฝากข้อความเสียงผ่านเครือข่ายอินเทอร์เน็ตนั้น เป็นการพัฒนาระบบสื่อสารทางไกลผ่านเครือข่ายอินเทอร์เน็ตแบบหนึ่ง ซึ่งนำโครงสร้างส่วนไคลเอนท์และส่วนเซิร์ฟเวอร์ไปติดตั้งไว้ตามจังหวัดหรือพื้นที่ต่างๆกัน โดยมีลักษณะโครงสร้างดังรูปที่ 3.8



รูปที่ 3.8 โครงสร้างการพัฒนาเป็นแบบสื่อสารทางไกลผ่านเครือข่ายอินเทอร์เน็ต

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.2 การศึกษาเพื่อการพัฒนา

การพัฒนาระบบฝากข้อความเสียงผ่านเครือข่ายอินเทอร์เน็ต ได้แบ่งการดำเนินงานออกเป็น 2 ระบบคือ ส่วนไคลเอนท์ และ ส่วนเซิร์ฟเวอร์ ดังนี้

3.2.1 การศึกษาเพื่อการพัฒนาส่วนไคลเอนท์

การศึกษาในส่วนไคลเอนท์นี้จะเกี่ยวข้องกับด้านต่างๆ คือ ระบบเกี่ยวกับข้อความเสียง การส่งไฟล์ไปยังเครื่องเซิร์ฟเวอร์ ระบบฐานข้อมูล และ อินเทอร์เน็ตที่เกี่ยวข้องกับผู้ใช้งานส่วนไคลเอนท์

ระบบที่เกี่ยวข้องกับข้อความเสียง ได้แก่ ฟังก์ชันต่างๆดังนี้

- การบันทึกเสียง
- การบีบอัดเสียง
- การจัดเก็บไฟล์เสียง

ระบบที่เกี่ยวข้องกับการส่งไฟล์เป็นการส่งไฟล์เสียงที่ทำการบันทึกเสียงจากผู้ส่งเรียบร้อยแล้ว ไปยังเครื่องเซิร์ฟเวอร์ ซึ่งมีฟังก์ชันการทำงานดังนี้

- การสร้างคอนเน็คชันระหว่างเครื่องไคลเอนท์กับเซิร์ฟเวอร์
- การส่งไฟล์แบบ File Transfer Protocol ผ่านเครือข่ายอินเทอร์เน็ต

ระบบฐานข้อมูลเป็นส่วนที่เกี่ยวข้องกับข้อมูลผู้ใช้และการเชื่อมโยงไปยังเซิร์ฟเวอร์ตัวต่างๆ ซึ่งมีฟังก์ชันการทำงานดังนี้

- ระบบฐานข้อมูลผู้ใช้งาน
- ระบบฐานข้อมูลเครื่องเซิร์ฟเวอร์

อินเทอร์เน็ตที่เกี่ยวข้องกับผู้ใช้งานส่วนไคลเอนท์ เป็นการทำให้ผู้ใช้งานมีความสะดวกและเข้าใจการทำงานได้มากขึ้น ซึ่งศึกษาถึงการทำงานดังนี้

- รูปแบบอินเทอร์เน็ตผู้ใช้
- รูปแบบความช่วยเหลือแบบต่างๆ
- Microsoft Agent

3.2.2 การศึกษาเพื่อการพัฒนาส่วนเซิร์ฟเวอร์

การศึกษาในส่วนเซิร์ฟเวอร์นี้จะเกี่ยวข้องกับด้านต่างๆ คือ การติดต่อกับระบบโทรศัพท์ การเล่นเกมทางโทรศัพท์ การรับและจัดเก็บไฟล์เสียง และ อินเทอร์เน็ตที่เกี่ยวข้องกับผู้ใช้งานส่วนเซิร์ฟเวอร์

การติดต่อกับระบบโทรศัพท์เป็นส่วนที่เกี่ยวข้องกับขั้นตอนตั้งแต่ผู้ใช้โทรศัพท์เข้ามาสู่ระบบจนกระทั่งวางสาย ซึ่งจำเป็นต้องศึกษาถึงฟังก์ชันต่างๆดังนี้

- การทำงานของ Voice Modem ที่เกี่ยวข้องกับระบบโทรศัพท์(Telephony)
- การตั้งสภาวะของระบบให้อยู่ในสภาวะเตรียมพร้อมกับการรับสาย
- การจับสัญญาณ โทรเข้าและจำนวนครั้งที่รับสาย
- การจับสัญญาณปุ่มที่กด
- การตัดสายและกลับสู่สภาวะเตรียมพร้อม

การเล่นเกมของระบบ เป็นส่วนที่แสดงให้เห็นให้ผู้ใช้งานสามารถรับฟังเสียงที่ต้องการได้เปรียบได้กับ Output ของระบบ ซึ่งจำเป็นต้องศึกษาถึงฟังก์ชันต่างๆดังนี้

- การทำงานของ Voice Modem ที่เกี่ยวข้องกับการเล่นเกมเสียงออกทางสายโทรศัพท์
- การเล่นเกมตามฟังก์ชันของระบบ โทรศัพท์ที่สั่งมา
- การเล่นเกมตามขั้นตอนต่างๆของ โปรแกรม
- การบันทึกเสียงและจัดเก็บ ไฟล์เสียงขั้นตอนระบบ

การรับและจัดเก็บไฟล์เสียงเป็นส่วนที่ดำเนินการจากส่วนไคลเอนท์ที่ส่งไฟล์เสียงมาให้แก่เซิร์ฟเวอร์ จะศึกษาถึงฟังก์ชันการทำงานดังนี้

- การสร้างคอนเน็คชันระหว่างเครื่องไคลเอนท์กับเครื่องเซิร์ฟเวอร์
- การรับ ไฟล์เสียงแบบ File Transfer Protocol
- การจัดเก็บไฟล์เสียงให้เชื่อมโยงกับระบบ
- การเรียกใช้ไฟล์เสียงจากระบบ

อินเทอร์เน็ตที่เกี่ยวข้องกับผู้ควบคุมส่วนเซิร์ฟเวอร์จะเป็นการเซ็คข้อมูลและแสดงผลของสถานะระบบ ซึ่งจะศึกษาถึงฟังก์ชันการทำงานดังนี้

- ระบบการเซ็คข้อมูลของระบบ
- ระบบการตรวจสอบสถานะของระบบ
- รูปแบบการใช้งานของผู้ควบคุมส่วนเซิร์ฟเวอร์

เมื่อศึกษาถึงส่วนต่างๆทั้งหมด จะต้องนำส่วนเหล่านี้มาเชื่อมโยงกันและสร้างเป็นระบบที่ใช้งานได้

3.3 ข้อมูลและตัวอย่างที่ใช้ในการพัฒนา

ในการดำเนินการพัฒนา ระบบฝากข้อความเสียงผ่านเครือข่ายอินเทอร์เน็ต ข้อมูลและตัวอย่างที่พบส่วนใหญ่จะเป็นระบบตอบรับโทรศัพท์อัตโนมัติ (Automatic Answering Machine) ซึ่งช่วยให้การใช้ฟังก์ชันเกี่ยวกับโทรศัพท์และฟังก์ชันเล่นเสียงทางโทรศัพท์ทำได้ง่ายขึ้น

3.4 เครื่องมือและวิธีการในการพัฒนา

การพัฒนาระบบฝากข้อความเสียงผ่านเครือข่ายอินเทอร์เน็ตแบ่งเป็น 2 ส่วน ดังที่กล่าวไปแล้วในเบื้องต้น ในส่วนของไคลเอนท์จะใช้ภาษา Visual Basic ในการพัฒนา และ ในส่วนของเซิร์ฟเวอร์เทคโนโลยีนี้จะใช้ Visual C++ ในการพัฒนา

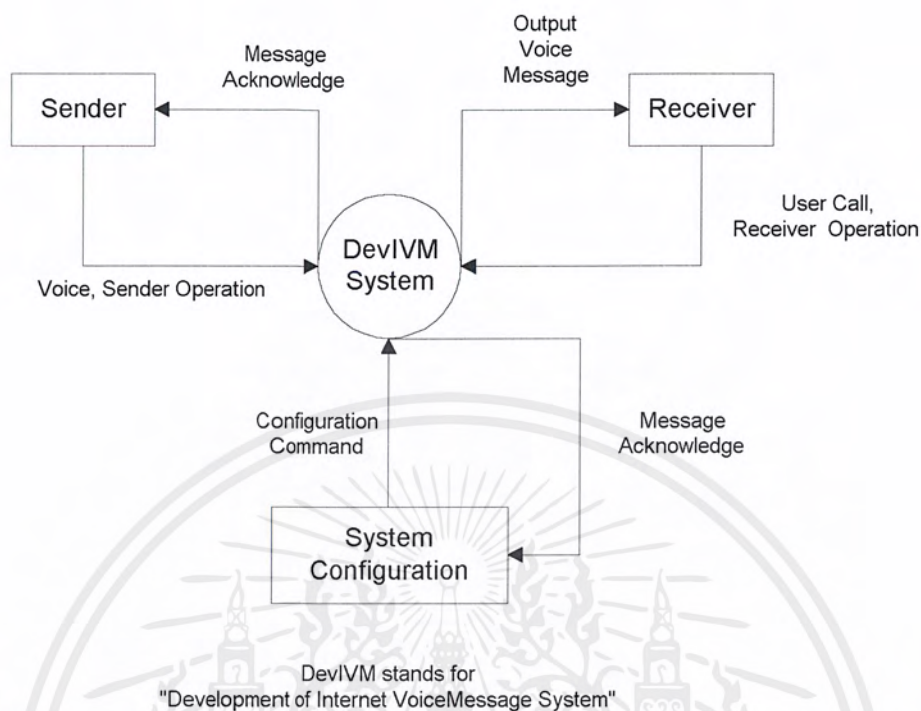
วิธีการพัฒนาจะแยกเป็น 3 ส่วนคือ

1. โปรแกรมส่วนไคลเอนท์ ได้แก่ บันทึกเสียง ส่งไฟล์ และ ระบบฐานข้อมูล
2. โปรแกรมส่วนเซิร์ฟเวอร์ ได้แก่ ระบบตอบรับโทรศัพท์ และ ระบบฐานข้อมูล
3. การเชื่อมต่อระหว่างโปรแกรม ได้แก่ การสร้างคอนเน็คชัน และการรับส่งไฟล์แบบ File Transfer Protocol

3.5 การทำงานของโปรแกรมที่พัฒนา

การพัฒนาระบบฝากข้อความเสียงผ่านเครือข่ายอินเทอร์เน็ต มีขั้นตอนในการทำงานของโปรแกรมที่พัฒนา แสดงเป็นแผนภาพต่างๆ โดยมีรายละเอียดดังนี้

1. Context Diagram เป็นแผนภาพแสดงถึงระบบโดยรวมและหน่วยงานนอกระบบที่เกี่ยวข้อง
2. Flow Chart เป็นแผนภาพแสดงถึงการทำงานต่างๆที่เกิดขึ้นภายในระบบซึ่งมีอยู่ 2 ส่วน คือ
 - 2.1 Flow Chart ส่วนไคลเอนท์ เป็นขั้นตอนการทำงานของโปรแกรมที่พัฒนาส่วนไคลเอนท์ (DevIVM Client)
 - 2.2 Flow Chart ส่วนเซิร์ฟเวอร์เป็นขั้นตอนการทำงานของโปรแกรมที่พัฒนาส่วนเซิร์ฟเวอร์ (DevIVM Server)



รูปที่ 3.9 Context Diagram ของระบบฝากข้อความเสียงผ่านเครือข่ายอินเทอร์เน็ต

Context Diagram เป็นระบบโดยรวมของระบบฝากข้อความเสียงผ่านเครือข่ายอินเทอร์เน็ต ซึ่งมีหน่วยเกี่ยวข้องกับระบบภายนอก ได้แก่

- Sender หมายความว่า ผู้ส่งข้อความเสียง
 - Input : ได้แก่ เสียงและคำสั่งทำงานของผู้ส่ง
 - Output : ได้แก่ การตอบรับจากระบบ เช่น ยืนยันการส่ง เป็นต้น
- Receiver หมายความว่า ผู้รับข้อความเสียง
 - Input : ได้แก่ การโทรศัพท์เข้าและคำสั่งทำงานของผู้รับจากปุ่มกดโทรศัพท์
 - Output : ได้แก่ ข้อความเสียง
- System Configuration หมายความว่า ผู้แก้ไขระบบและดูแลระบบ
 - Input : ได้แก่ คำสั่งการแก้ไขระบบ
 - Output : ได้แก่ การตอบรับการแก้ไขและการแสดงสถานะระบบ

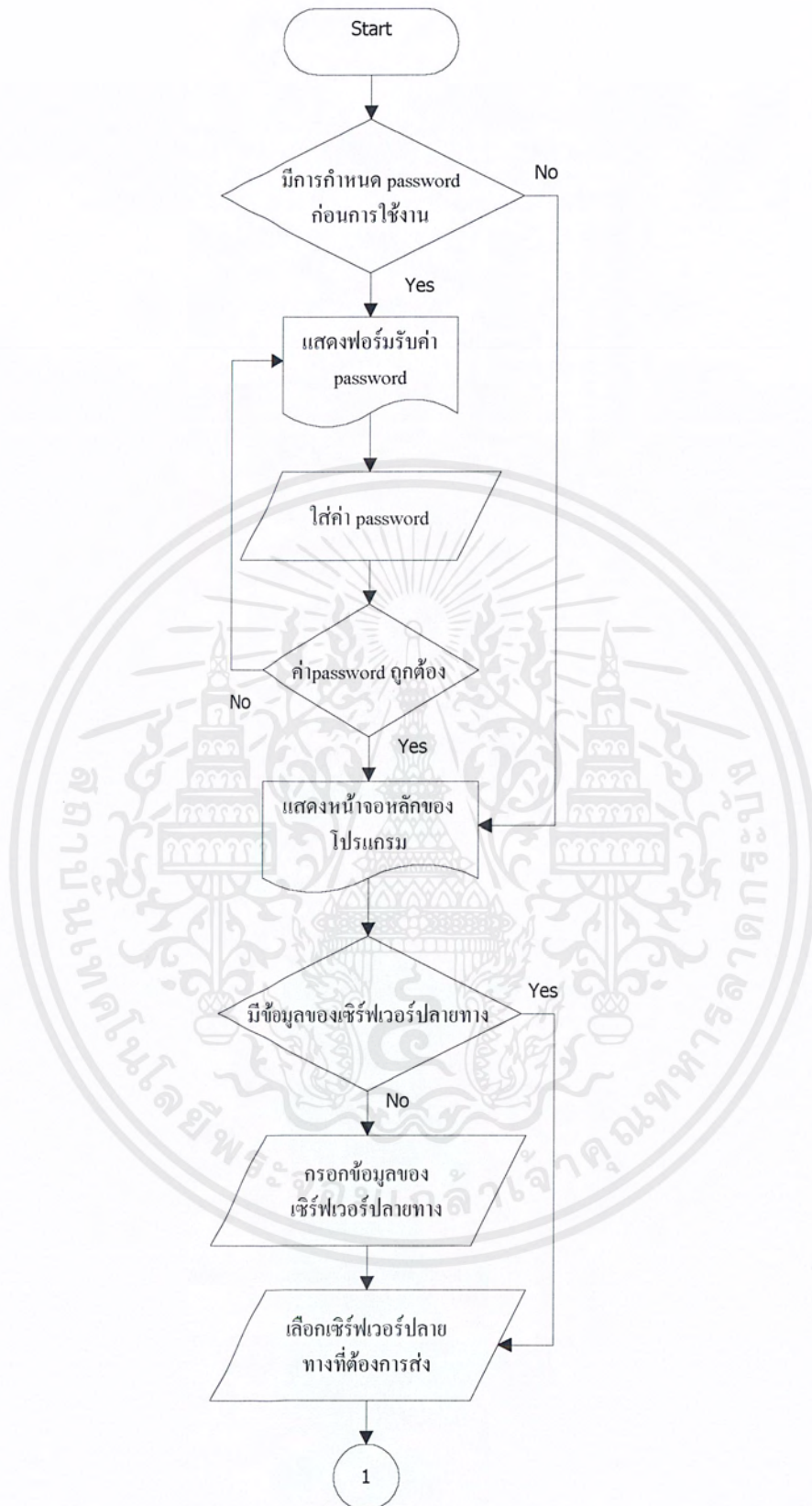
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Flow Chart ของระบบฝากข้อความเสียงผ่านเครือข่ายอินเทอร์เน็ตซึ่งแบ่งออกเป็น 2 ส่วน คือ ส่วนไคลเอนท์ และ ส่วนเซิร์ฟเวอร์ ซึ่งแสดงได้ดังนี้

Flow Chart รูปที่ 3.10(1) และ 3.10(2) แสดงการทำงานของโปรแกรมระบบฝากข้อความเสียงผ่านเครือข่ายอินเทอร์เน็ตส่วนไคลเอนท์ (DevIVM Client) ซึ่งเป็นการทำงานตั้งแต่ การดำเนินการเกี่ยวกับข้อมูลผู้ใช้ บันทึกเสียงจากผู้ส่ง ระบุหมายเลขปลายทาง ส่งไฟล์ไปยังเซิร์ฟเวอร์ปลายทาง และการตอบรับการส่งเป็นที่เรียบร้อย

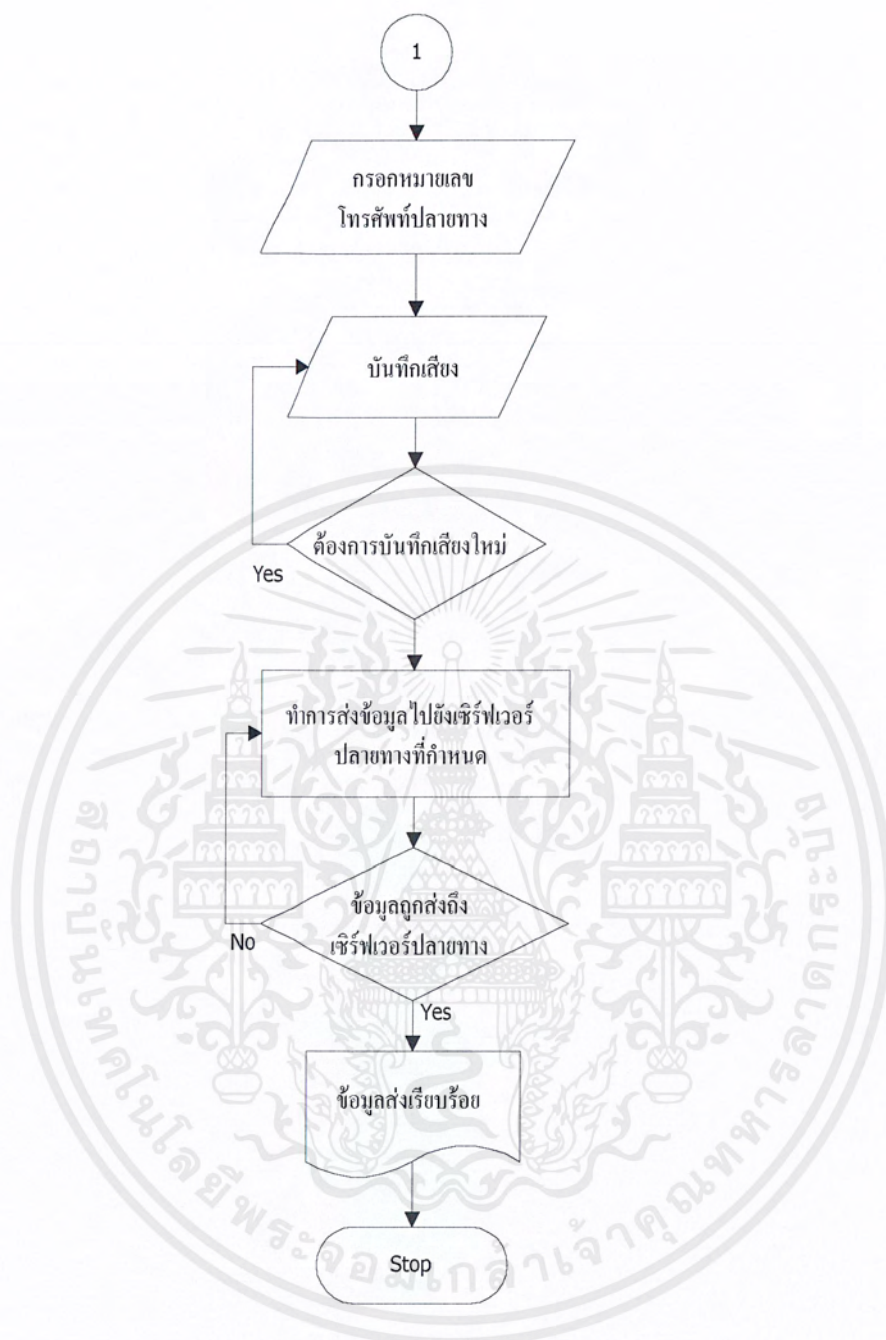
Flow Chart รูปที่ 3.11(1) และ 3.11(2) แสดงการทำงานของโปรแกรมระบบฝากข้อความเสียงผ่านเครือข่ายอินเทอร์เน็ตส่วนเซิร์ฟเวอร์ (DevIVM Server) เป็นการทำงานตั้งแต่ เริ่มมีโทรศัพท์เข้ามายังระบบ การตอบรับจากระบบ การรับข้อมูลจากการกดปุ่มโทรศัพท์ การค้นหาไฟล์ข้อความเสียงที่ฝากไว้ การเล่นไฟล์ข้อความเสียง และการจบการทำงาน





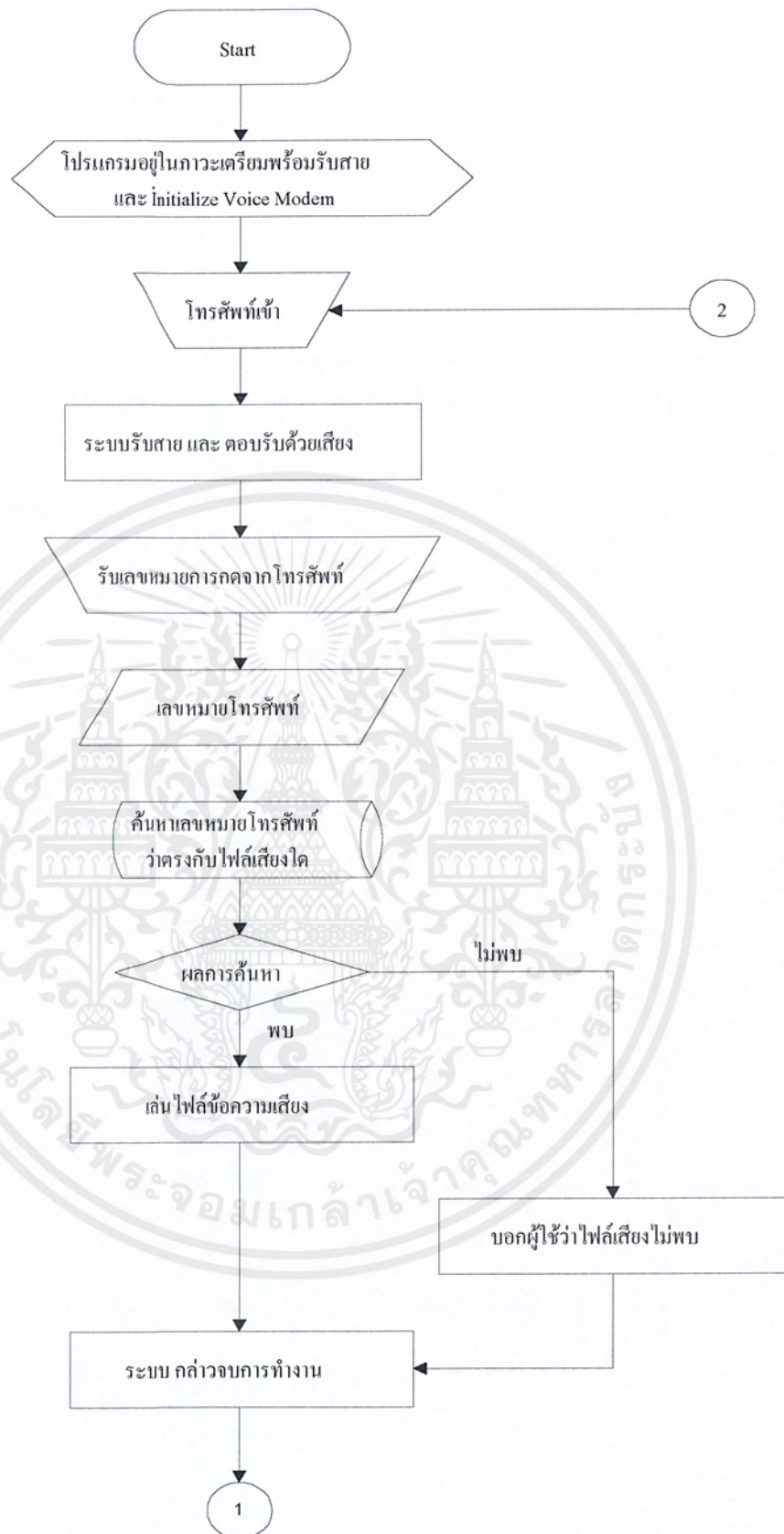
รูปที่ 3.10(1) Flow Chart ส่วนที่ 1 ของโปรแกรมระบบฝากข้อความเสียงผ่านเครือข่าย อินเทอร์เน็ตส่วน ไคลเอนท์ (DevIVM Client)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



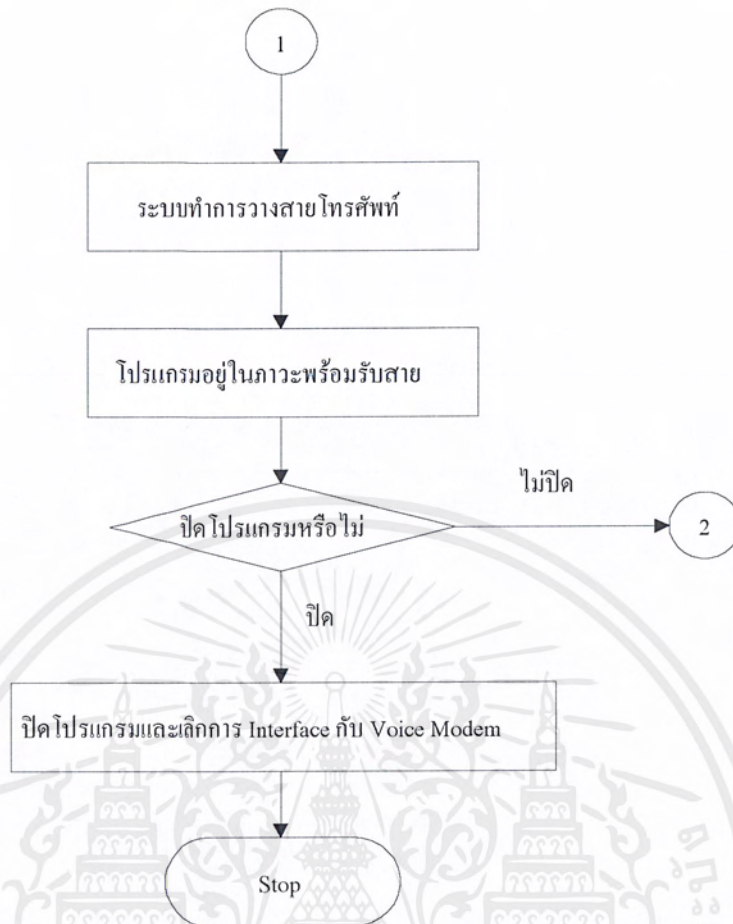
รูปที่ 3.10(2) Flow Chart ส่วนที่ 2 ของ โปรแกรมระบบฝากข้อความเสียงผ่านเครือข่าย
อินเทอร์เน็ตส่วนไคลเอนท์ (DevIVM Client)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.11(1) Flow Chart ส่วนที่ 1 ของโปรแกรมระบบฝากข้อความเสียงผ่านเครือข่าย
อินเทอร์เน็ตส่วนเซิร์ฟเวอร์ (DevIVM Server)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.11(2) Flow Chart ส่วนที่ 2 ของโปรแกรมระบบฝากข้อความเสียงผ่านเครือข่ายอินเทอร์เน็ตส่วนเซิร์ฟเวอร์ (DevIVM Server)

บทที่ 4

ประเมินและอภิปรายผล

4.1 ลักษณะของโปรแกรมระบบฝากข้อความเสียงผ่านเครือข่ายอินเทอร์เน็ต (DevIVM)

การพัฒนาโปรแกรมระบบฝากข้อความเสียงผ่านเครือข่ายอินเทอร์เน็ต ได้แบ่งลักษณะการทำงานของโปรแกรมออกเป็น 2 ส่วนคือ โปรแกรมส่วนไคลเอนท์ และ โปรแกรมส่วนเซิร์ฟเวอร์

1. โปรแกรมส่วนไคลเอนท์ (DevIVM Client) เป็นส่วนโปรแกรมที่เกี่ยวข้องกับผู้ส่งข้อความเสียง ซึ่งจะติดตั้งไว้กับเครื่องคอมพิวเตอร์ของผู้ที่ต้องการส่งข้อความเสียง ทางกลุ่มจัดทำได้พัฒนาโปรแกรมส่วนไคลเอนท์ ด้วยภาษา Visual Basic โดยมีลักษณะดังนี้
 - มีระบบข้อมูลผู้ใช้งาน เก็บข้อมูลต่างๆ เช่น Username, Password
 - บันทึกเสียง และ จัดเก็บไฟล์เสียงได้
 - สามารถบีบอัดไฟล์เสียงแบบ MP3 ได้
 - สามารถส่งไฟล์เสียงไปยังเครื่องเซิร์ฟเวอร์ผ่านอินเทอร์เน็ตโดยใช้ File Transfer Protocol (FTP)
 - มีระบบ Locate Telephony Server สำหรับระบุเครื่องเซิร์ฟเวอร์ที่ส่งไฟล์ไป
 - มีฟังก์ชันการให้ความช่วยเหลือผู้ใช้งานแบบ Interactive
 - รูปแบบอินเตอร์เฟซ คือ ระบบต่างๆกระทำในหน้าจอเดียว และฟังก์ชันการทำงานใช้แบบเมนู
2. โปรแกรมส่วนเซิร์ฟเวอร์ (DevIVM Telephony Server) เป็นส่วนโปรแกรมที่เกี่ยวข้องกับผู้รับข้อความเสียง ซึ่งจะติดตั้งไว้กับเครื่องที่ทำหน้าที่เป็นเซิร์ฟเวอร์ให้ผู้ใช้โทรศัพท์เข้ามารับฟังข้อความเสียง ในกรณีของการนำระบบไปสื่อสารแบบทางไกลข้ามจังหวัด หรือข้ามประเทศนั้น โปรแกรมนี้จะถูกติดตั้งไว้ที่จังหวัด หรือ ประเทศที่มีผู้รับข้อความเสียง ทางกลุ่มจัดทำได้พัฒนาโปรแกรมส่วนเซิร์ฟเวอร์ ด้วยภาษา Visual C++ โดยมีลักษณะดังนี้
 - สามารถรับไฟล์ข้อความเสียงที่ถูกส่งมาจากโปรแกรมส่วนไคลเอนท์ แบบ File Transfer Protocol โดยใช้โปรแกรม Server U
 - เชื่อมต่อกับ Voice Modem ได้ทันที
 - มีระบบข้อความเสียงตอบรับ และ ข้อความเสียงแสดงขั้นตอนการทำงาน
 - สามารถแก้ไขข้อกำหนดระบบได้ เช่น ปรับจำนวนครั้งโทรศัพท์ ปรับระดับเสียง
 - สามารถปรับการใช้งานตามจำนวนเลขเบอร์โทรศัพท์สำหรับท้องถิ่นต่างๆได้
 - มีหน้าจอแสดงสถานะของระบบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.2 ข้อกำหนดของการประเมิน

ติดตั้งโปรแกรมแต่ละส่วนไว้ในเครื่องคอมพิวเตอร์แต่ละเครื่อง โดยเครื่องทั้งสองต่อเข้ากับเครือข่ายอินเทอร์เน็ต และเครื่องเซิร์ฟเวอร์ต่อเข้ากับเครื่อง Voice Modem 1 เครื่องที่มีสายโทรศัพท์ต่ออยู่ด้วย เครื่องไคลเอนต์ใช้ IP : 161.246.58.25 และเครื่องเซิร์ฟเวอร์ใช้ IP : 161.246.58.26

4.3 การประเมิน

การทดสอบการใช้งานของระบบ ทำการทดสอบเมื่อวันที่ 6 มีนาคม พ.ศ. 2544

1. เมื่อเริ่มการใช้งานโปรแกรมส่วนไคลเอนต์ ระบุหมายเลขปลายทางเป็นรหัสทางไกล 038 – 9999999 ซึ่งจะเชื่อมโยงไปยังเครื่องเซิร์ฟเวอร์ที่มี IP : 161.246.58.26
2. ทำการบันทึกเสียงข้อความเสียงที่จะส่งเป็นเวลา 10 วินาที เก็บเป็นไฟล์เสียงแบบ Wave
3. ส่งไฟล์ข้อความเสียงไปยังเครื่องเซิร์ฟเวอร์ ใช้เวลาประมาณ 5 วินาที
4. เริ่มการใช้งานโปรแกรมส่วนเซิร์ฟเวอร์ ใช้โทรศัพท์อีกเครื่องหนึ่งโทรเข้ามายังเบอร์โทรศัพท์ที่ต่อเข้ากับ Voice Modem ของเครื่องเซิร์ฟเวอร์
5. โปรแกรมส่วนเซิร์ฟเวอร์ รับโทรศัพท์หลังจากเสียงกริ่งดัง 2 ครั้ง
6. มีเสียงตอบรับเข้าสู่ระบบ
7. กดปุ่มบนโทรศัพท์ จำนวน 7 หลัก ที่มีเลขหมายเดียวกับหมายเลขปลายทาง ที่ระบุในโปรแกรมส่วนไคลเอนต์
8. มีเสียงทวนเลขหมายที่กดปุ่มจำนวน 7 หลัก
9. มีเสียงของข้อความเสียงที่ฝากไว้ตอบรับกลับมา
10. มีเสียงข้อความจบการทำงาน
11. โปรแกรมส่วนเซิร์ฟเวอร์กลับสู่สภาวะเตรียมพร้อม

4.4 การอภิปรายผล

หลังจากการทดสอบการใช้งานของระบบ พบว่า สามารถใช้งานได้ดี ระบบสามารถเชื่อมโยงการทำงานระหว่าง 2 ส่วนได้อย่างเรียบร้อย ไฟล์เสียงที่ส่งผ่านโดยใช้เครือข่ายอินเทอร์เน็ตแบบ File Transfer Protocol (FTP) มีความรวดเร็วอยู่ในระดับดี ในส่วนของระบบตอบรับโทรศัพท์ของโปรแกรมส่วนเซิร์ฟเวอร์ ดำเนินไปอย่างเรียบร้อย แต่มีข้อเสียในเรื่องของเสียงตอบรับที่เบา ซึ่งเกิดขึ้นเนื่องจากผ่านตู้สาขา (PABX) ซึ่งคณะจัดทำได้เคยทดลองก่อนหน้านี้โดยใช้โทรศัพท์สายตรง พบว่า ไม่เกิดปัญหานี้

การทดสอบครั้งนี้จะเป็นแนวทางการนำระบบฝากข้อความเสียงผ่านเครือข่ายอินเทอร์เน็ตไปใช้เป็นระบบสื่อสารทางไกลได้อย่างสมบูรณ์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 5

สรุปผลการดำเนินงานและข้อเสนอแนะ

การพัฒนาระบบฝากข้อความเสียงผ่านเครือข่ายอินเทอร์เน็ต ได้ดำเนินการตามวัตถุประสงค์ และประโยชน์ที่ได้วางไว้ คือ สามารถเชื่อมโยงข้อมูลจากเครือข่ายอินเทอร์เน็ตมาสู่ระบบ โทรศัพท์พื้นฐานได้ ในรูปแบบของข้อความเสียง เป็นการเพิ่มประสิทธิภาพของเครือข่ายอินเทอร์เน็ตให้มากขึ้น และลดช่องว่างระหว่างระบบสื่อสารข้อมูลให้น้อยลง ตลอดจนเป็นการพัฒนาระบบสื่อสารทางไกลด้วยเสียงผ่านเครือข่ายอินเทอร์เน็ตที่มีค่าใช้จ่ายน้อยกว่าระบบสื่อสารแบบโทรศัพท์โดยตรงมาก ซึ่งการพัฒนาในครั้งนี้จะเป็นแนวทางนำไปสู่การพัฒนาเป็นระบบโทรศัพท์อินเทอร์เน็ตต่อไปในอนาคต

อนึ่ง การพัฒนาระบบฝากข้อความเสียงผ่านเครือข่ายอินเทอร์เน็ตยังมีข้อเสนอแนะอีกส่วนหนึ่งซึ่งจะเป็นแนวทางในการพัฒนาต่อไป กล่าวคือ

1. โปรแกรมส่วนเซิร์ฟเวอร์ ใช้ภาษา Visual C++ ในการพัฒนา ดังนั้น ควรศึกษาภาษา Visual C++ วิทยะหนึ่งก่อนเนื่องจากฟังก์ชันการทำงานเกี่ยวกับโทรศัพท์ ต้องมีความเข้าใจในระดับหนึ่ง
2. ระบบทำงานเองอัตโนมัติ (Automatic Alert) เป็นส่วนที่เป็นประโยชน์อย่างยิ่งสำหรับผู้พัฒนาต่อไป กล่าวคือ หากสามารถทำระบบที่สามารถทำงานด้วยตัวเอง หรือ ตั้งเวลาให้ทำงานตามกำหนดได้ ระบบก็จะมีประสิทธิภาพสูงสุด ในขั้นตอนที่มีความจำเป็นในระบบนี้ คือ
 - ระบบแปลงไฟล์เสียงที่เครื่องเซิร์ฟเวอร์กลับเป็น Wave จาก Mp3
 - ระบบโทรศัพท์เตือนรับข้อความ ไปยังผู้รับอัตโนมัติ
3. ระบบโทรศัพท์เตือนรับข้อความ เป็นส่วนที่คณะจัดทำมีแนวคิดที่จะให้มีฟังก์ชันนี้ในระบบ แต่ต้องใช้ประสิทธิภาพของ Hardware ที่สามารถตรวจสอบสัญญาณขงโทรศัพท์ของผู้รับ จึงยังเป็นที่ต้องให้ผู้ที่จะพัฒนาศึกษาต่อไป
4. การพัฒนาขั้นตอนระบบฝากข้อความให้มีประสิทธิภาพมากขึ้น ซึ่งคณะจัดทำมีแนวคิดที่ให้พัฒนาต่อไป คือ
 - พัฒนาระบบข้อความให้สามารถฟังข้อความแบบข้าม, วนซ้ำข้อความ โดยควบคุมจากปุ่มกดโทรศัพท์
 - พัฒนาระบบคุณภาพเสียงข้อความให้ดียิ่งขึ้น
 - พัฒนาระบบให้สามารถบอกวัน เวลาที่ข้อความถูกส่งมา
 - พัฒนาให้ผู้ที่ใช้ระบบสามารถลบข้อมูลเสียงออกได้ด้วยตนเอง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5. การพัฒนาระบบสื่อสารสองทาง ซึ่งคณะจัดทำมีแนวคิดให้ในการพัฒนาต่อไป คือ
 - พัฒนาให้ผู้รับข้อความสามารถส่งข้อความกลับไปยังผู้ส่งได้ โดยส่งกลับในรูปแบบไฟล์เสียงเหมือนเดิม
 - พัฒนาระบบตอบรับยืนยันการรับข้อความแล้ว กลับไปยังผู้ส่งข้อความได้
6. การพัฒนาระบบกระจายข้อมูลระหว่างเซิร์ฟเวอร์ เพื่อพัฒนาเป็นระบบสื่อสารสองทาง คณะจัดทำมีแนวคิดในการพัฒนาต่อไป คือ
 - ระบบกระจายข้อมูล ให้เชื่อมโยงระหว่างเครื่องเซิร์ฟเวอร์ จากจุดนี้ทำให้สามารถพัฒนาเป็นระบบสื่อสารสองทางได้
7. การพัฒนาระบบฝากข้อความเสียงส่วนไคลเอนต์ผ่านเว็บไซค์ เป็นส่วนหนึ่งที่จะทำให้ระบบใช้งานได้อย่างแพร่หลายและสะดวกมากยิ่งขึ้น
8. การพัฒนาระบบโทรศัพท์อินเทอร์เน็ต (Internet Phone) เป็นเป้าหมายสูงสุดของการพัฒนา ซึ่งแนวทางในการพัฒนา คือ
 - รูปแบบระบบโทรศัพท์อินเทอร์เน็ต เป็นการ Save File ที่ต้นทาง, Transfer ไฟล์ไปปลายทางและ Open File ที่ปลายทาง เช่นเดียวกับระบบฝากข้อความเสียงผ่านเครือข่ายอินเทอร์เน็ต แต่เป็นระบบแบบเวลาจริง (Real-Time) ซึ่งต้องศึกษาถึงลักษณะของ Packet ไฟล์เสียงเป็นสำคัญ
 - เป็นการแปลงจาก สัญญาณ Analog เป็น Digital และ Digital กลับเป็น Analog ดังนั้น หากมีอุปกรณ์แปลงสัญญาณ (Converter) จะช่วยในการพัฒนาสะดวกขึ้น

ภาคผนวก ก
คู่มือการติดตั้ง

โปรแกรมระบบฝากข้อความเสียงผ่านเครือข่ายอินเทอร์เน็ตส่วนไอคลเอนท์



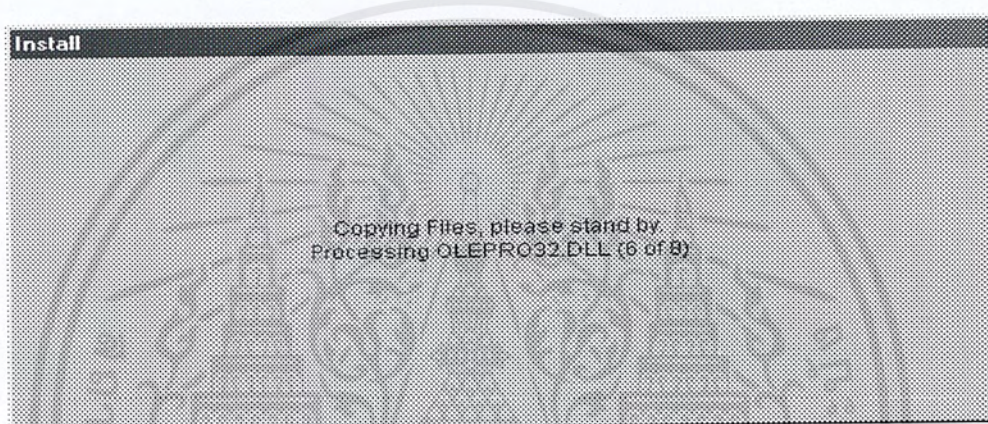
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1. คลิกไอคอน setup.exe เพื่อเริ่มต้นการติดตั้งโปรแกรม



รูปที่ ก(1) ไอคอน Setup

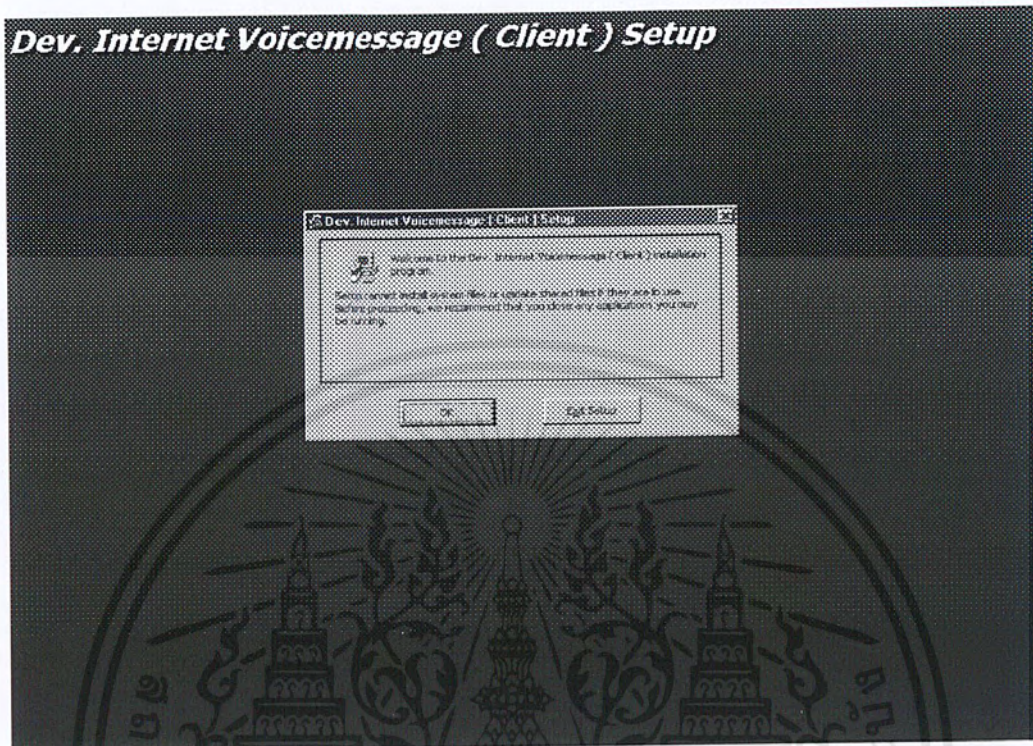
2. เริ่มต้นขั้นตอนการติดตั้งโปรแกรม



รูปที่ ก(2) แสดงหน้าจอติดตั้งโปรแกรมส่วนไคลเอนท์ 1

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

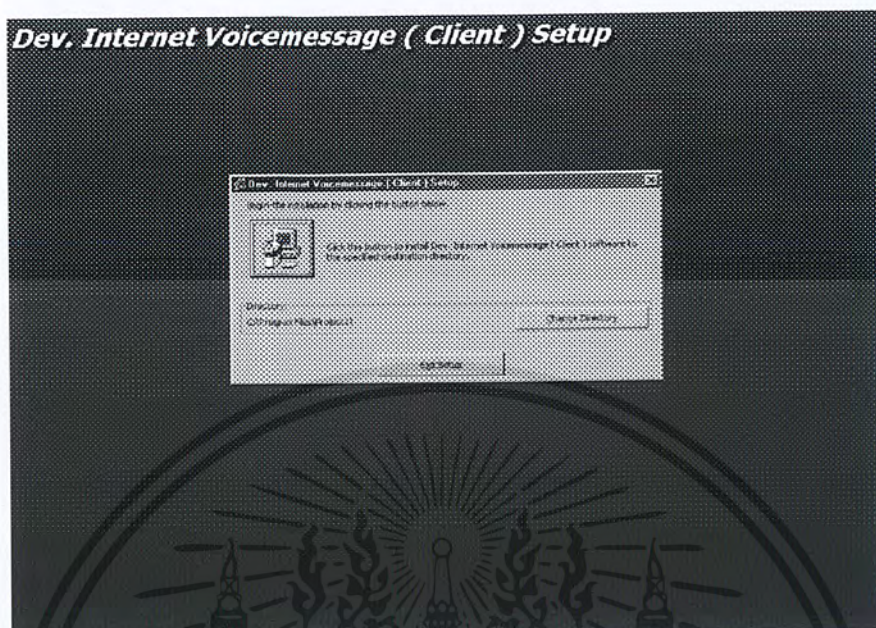
3. คลิกปุ่ม O.K. เพื่อทำงานติดตั้งโปรแกรม



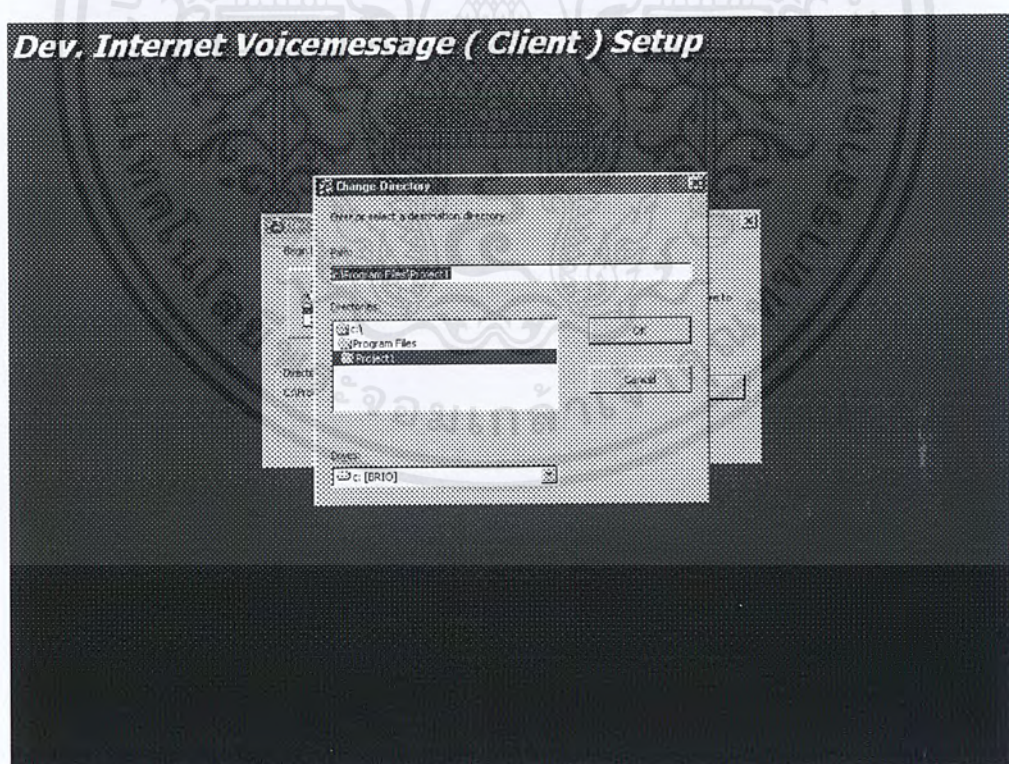
รูปที่ ก(3) แสดงหน้าจอติดตั้งโปรแกรมส่วนไคลเอนท์ 2

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4. คลิกปุ่ม Change Directory ในกรณีที่ต้องการเปลี่ยนไดเรกทอรีในการติดตั้งโปรแกรม



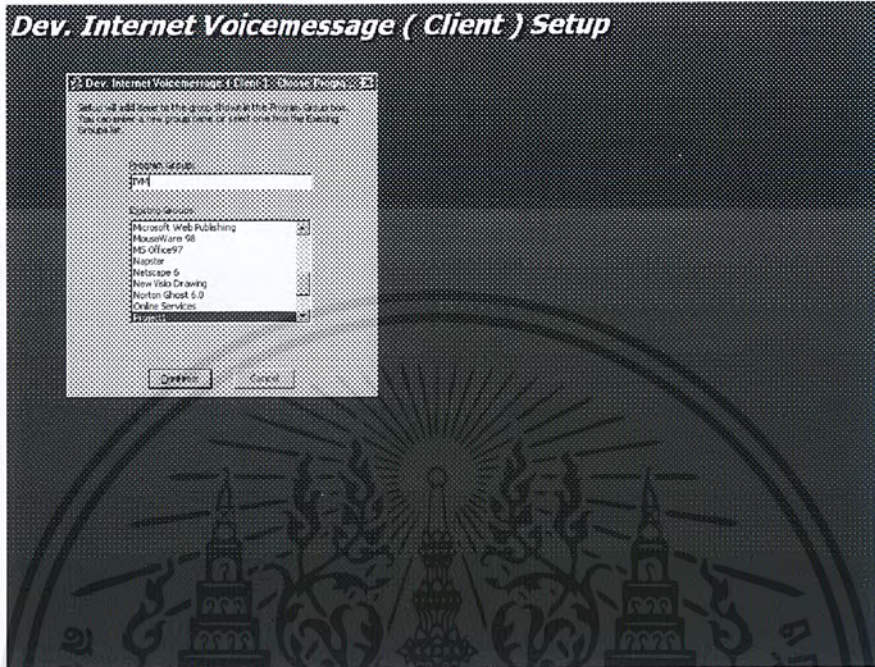
รูปที่ ก(4) แสดงหน้าจอติดตั้งโปรแกรมส่วนไคลเอนท์ 3



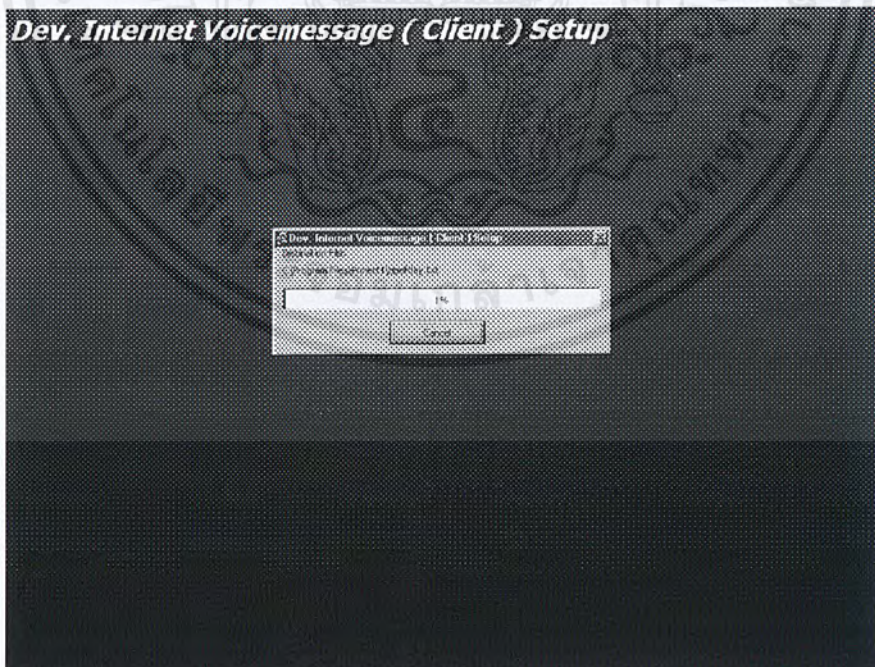
รูปที่ ก(5) แสดงหน้าจอติดตั้งโปรแกรมส่วนไคลเอนท์ 4

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5. ตั้งชื่อโปรแกรมหลังจากติดตั้งเสร็จเรียบร้อยแล้ว จากนั้นคลิกปุ่ม Continue เพื่อเริ่มต้นการติดตั้งโปรแกรมสู่ระบบ



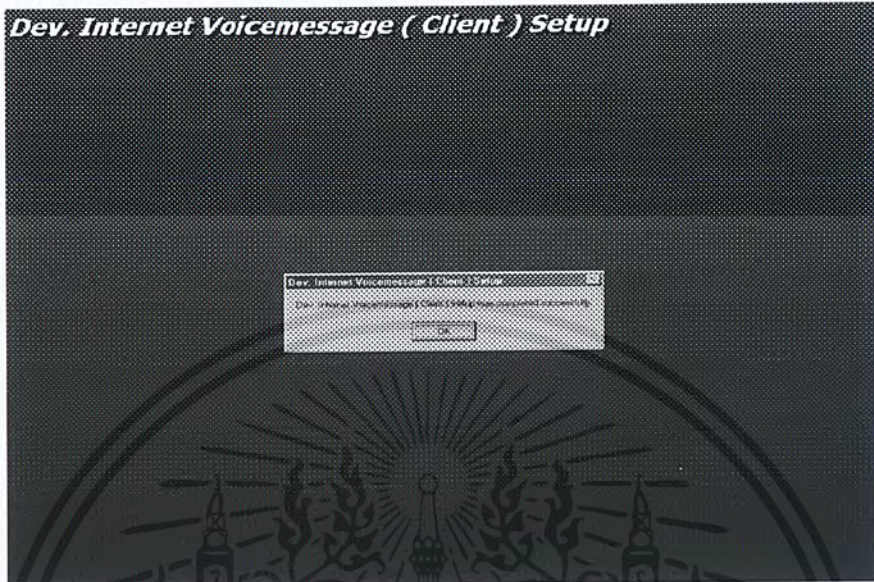
รูปที่ ก(6) แสดงหน้าจอติดตั้ง โปรแกรมส่วนไคลเอนท์ 5



รูปที่ ก(7) แสดงหน้าจอติดตั้ง โปรแกรมส่วนไคลเอนท์ 6

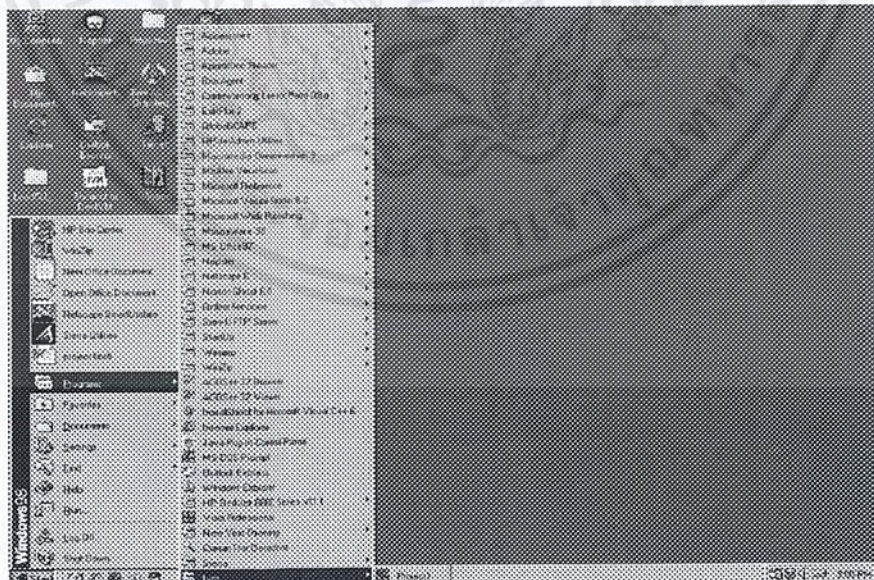
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

6. เมื่อติดตั้งโปรแกรมเสร็จเรียบร้อย ก็จะปรากฏหน้าจอตั้งภาพ คลิกปุ่ม O.K. ก็เป็นอันเสร็จสิ้นการติดตั้งโปรแกรม Development Internet Voicemail



รูปที่ ก(8) แสดงหน้าจอติดตั้งโปรแกรมส่วนไคลเอนท์ 7

7. ทดสอบโดยการเรียกใช้โปรแกรม Development Internet Voicemail โดยเรียก Start>IVM>Project1 ซึ่งชื่อที่ปรากฏจะเป็นไปตามที่กำหนดไว้ในตอนติดตั้งโปรแกรม



รูปที่ ก(9) แสดงหน้าจอเรียกใช้โปรแกรมส่วนไคลเอนท์ผ่าน Windows

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ข
คู่มือการติดตั้ง
โปรแกรมระบบฝากข้อความเสียงผ่านเครือข่ายอินเทอร์เน็ตส่วนเซิร์ฟเวอร์



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

คู่มือการติดตั้ง

โปรแกรมระบบฝากข้อความเสียงผ่านเครือข่ายอินเทอร์เน็ตส่วนเซิร์ฟเวอร์

การติดตั้งโปรแกรมระบบฝากข้อความเสียงผ่านเครือข่ายอินเทอร์เน็ตส่วนเซิร์ฟเวอร์จะเป็นการติดตั้งบนเครื่องคอมพิวเตอร์ส่วนบุคคลทั่วไป โดยเครื่องคอมพิวเตอร์จะต้องมี Voice Modem ต่ออยู่และทำการ Install Voice Modem เป็นที่เรียบร้อยแล้ว

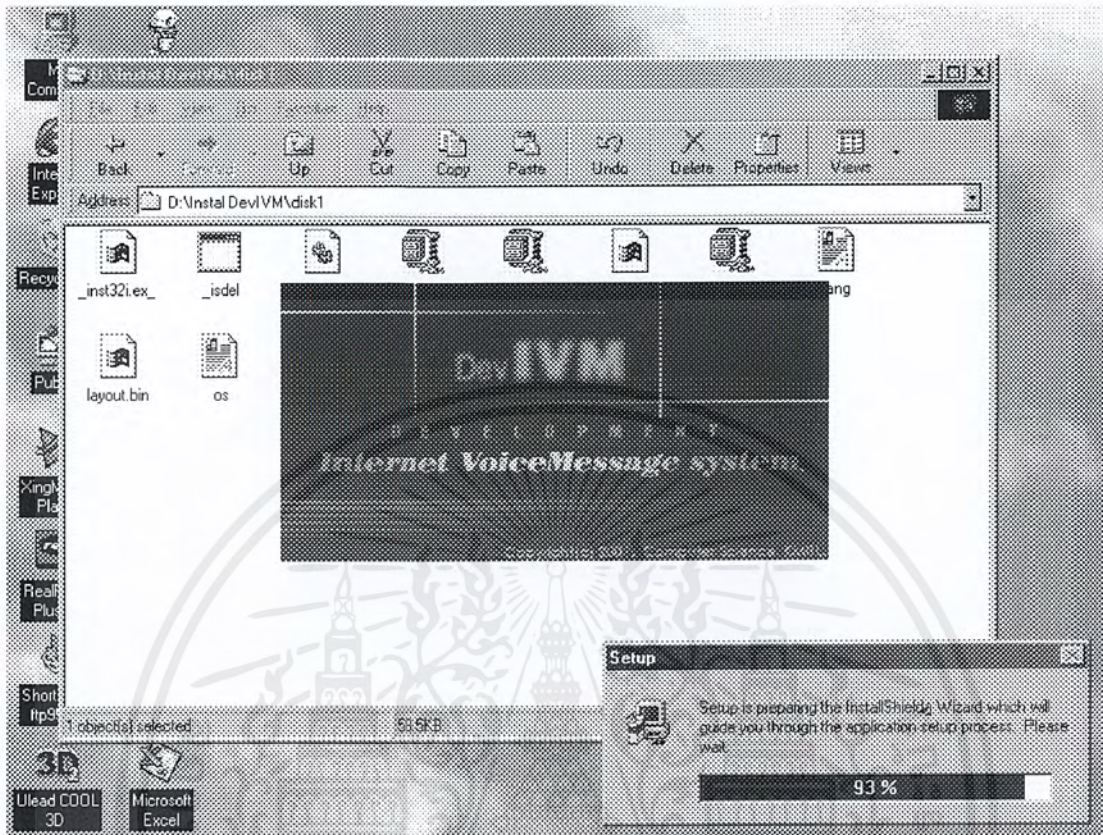
ขั้นตอนในการติดตั้งมีดังนี้

1. คลิก Setup.exe ที่แผ่นติดตั้ง



รูปที่ ข(1) แสดงไฟล์ติดตั้งโปรแกรมส่วนเซิร์ฟเวอร์

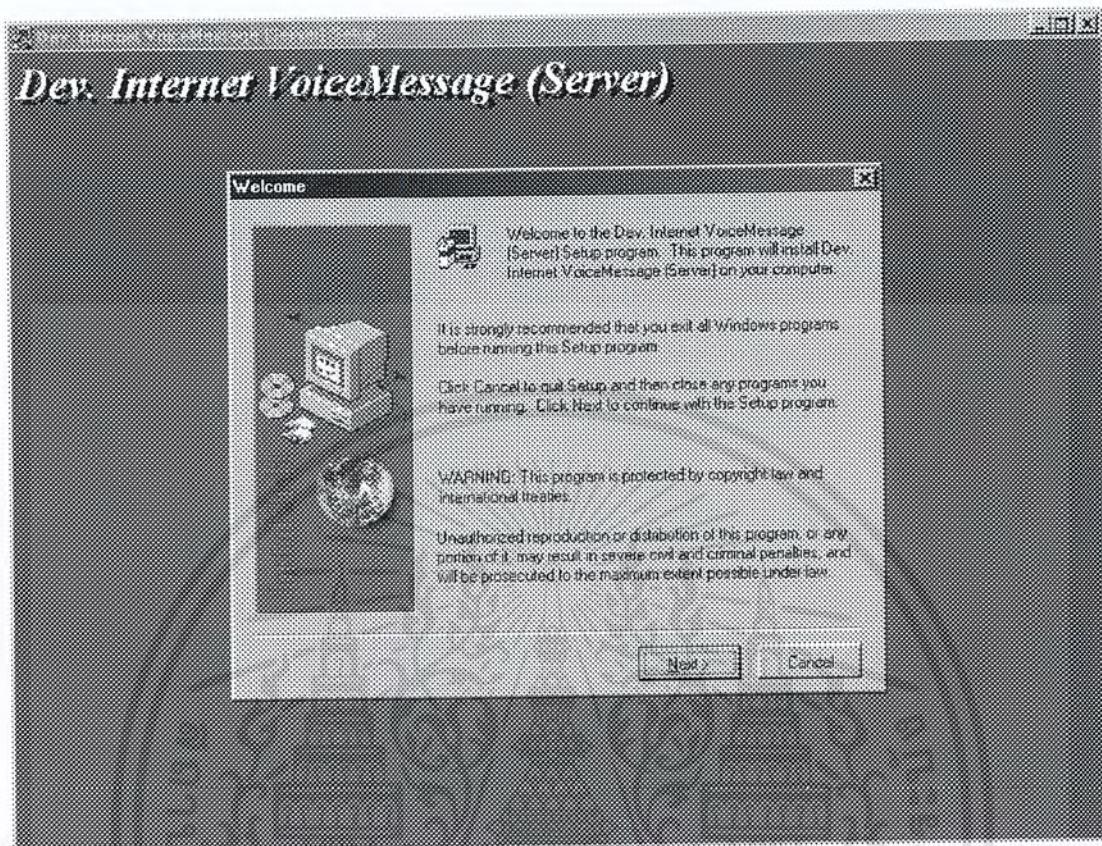
2. โปรแกรมติดตั้งจะทำงานและเข้าสู่การติดตั้ง



รูปที่ ข(2) เริ่มการทำงานติดตั้งโปรแกรมส่วนเซิร์ฟเวอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

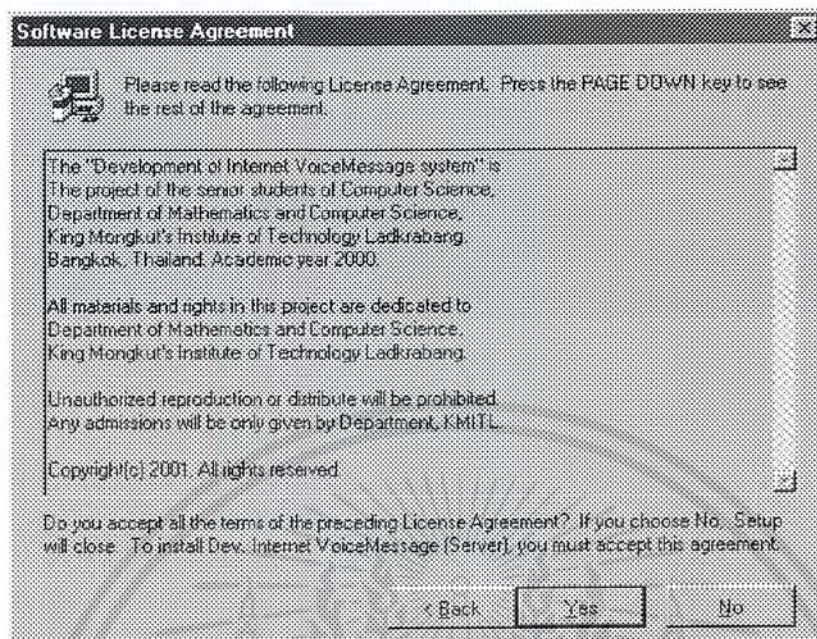
3. จะมีข้อความแสดงให้อ่าน และ กดปุ่ม Next ไป



รูปที่ ข(3) การทำงานติดตั้งโปรแกรมส่วนเซิร์ฟเวอร์ 1

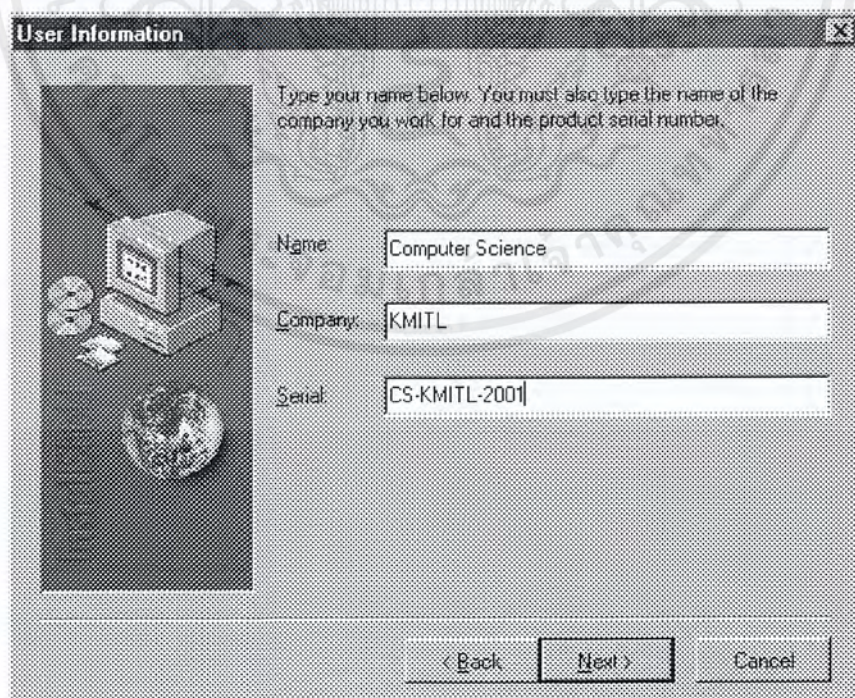
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4. จะมีหน้าจอแสดงข้อตกลงในการติดตั้งโปรแกรม



รูปที่ ข(4) ข้อตกลงในการติดตั้ง (License Agreement)

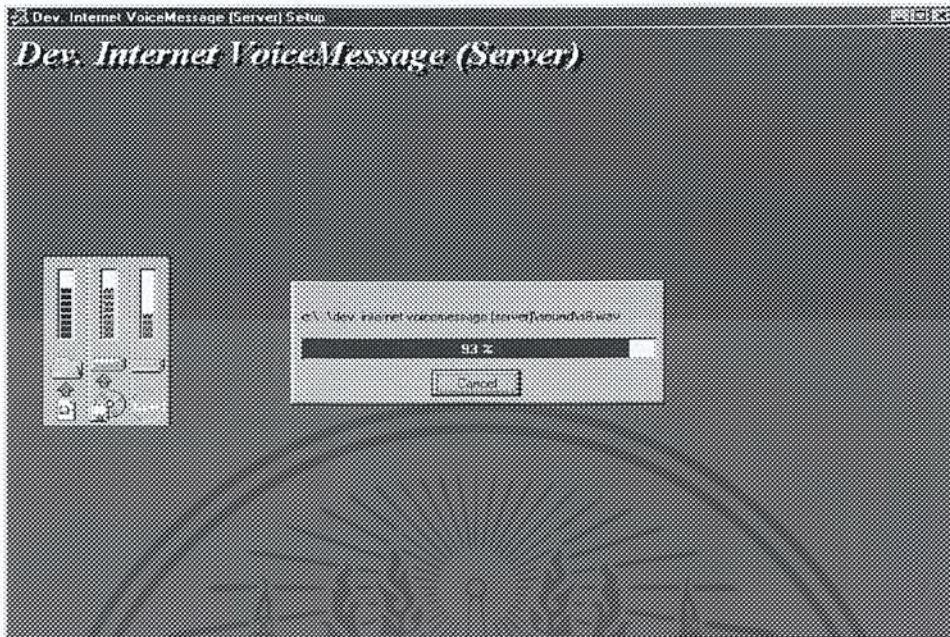
5. จากนั้นจะมีหน้าจอ User Information ให้ใช้ Serial Number เป็น CS-KMITL-2001



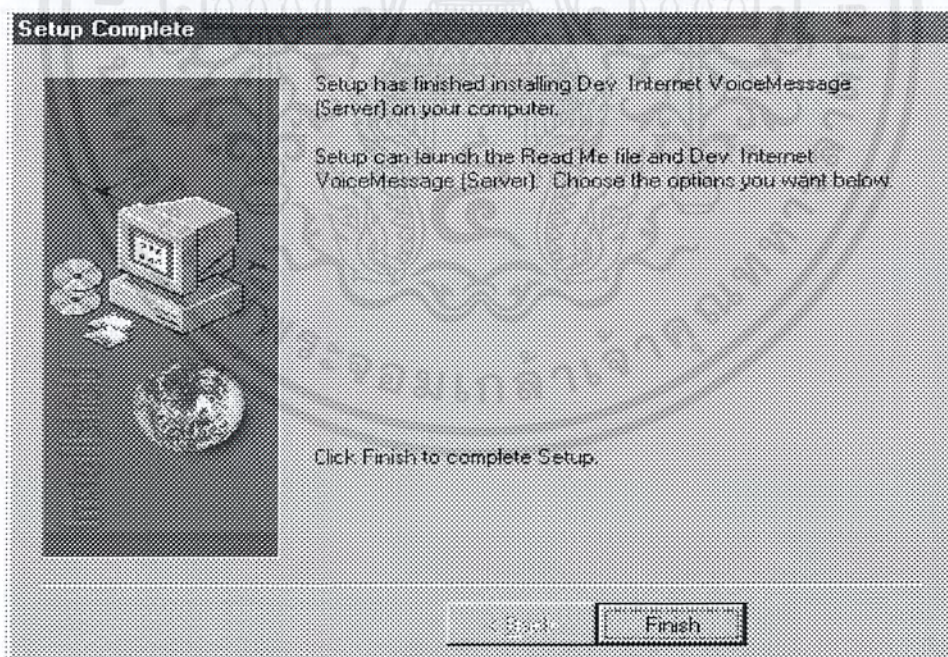
รูปที่ ข(5) หน้าจอ User Information

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

6. จากนั้นโปรแกรมจะทำการติดตั้งบน Storage จนเสร็จเรียบร้อย



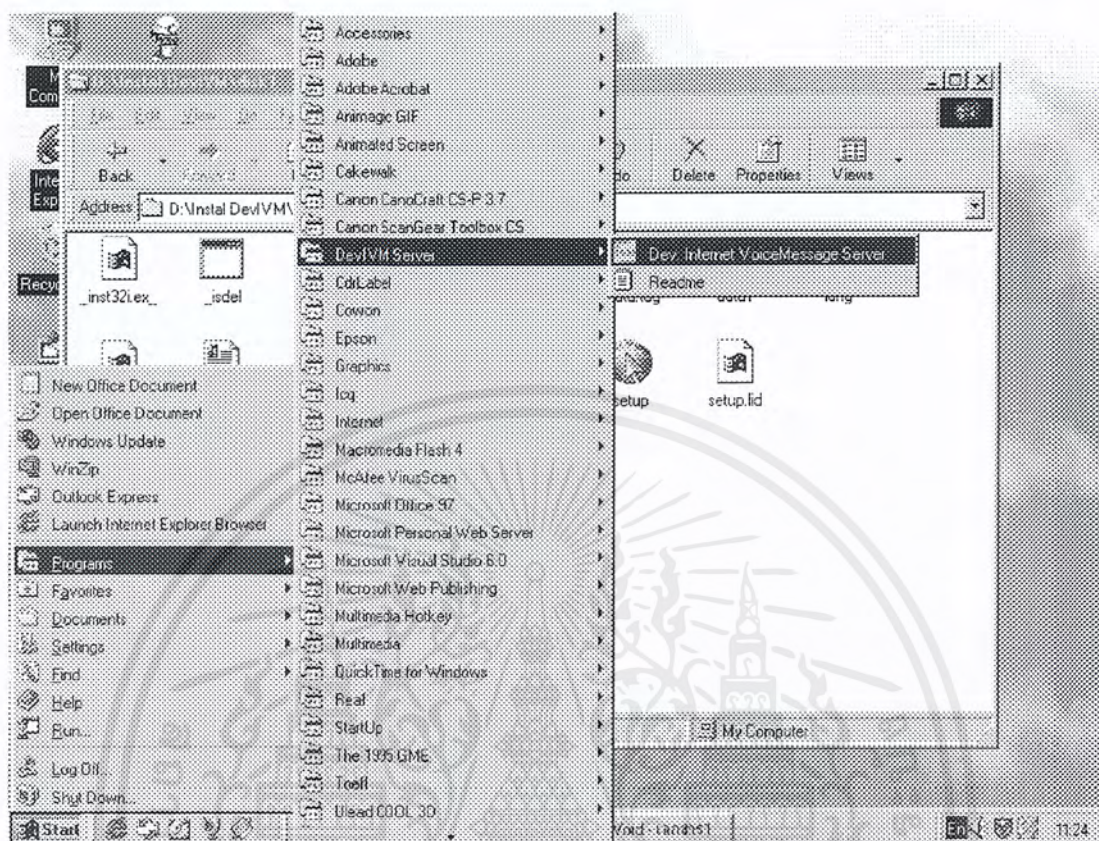
รูปที่ ข(6) หน้าจอแสดงการติดตั้งบน Storage



รูปที่ ข(7) โปรแกรมส่วนเซิร์ฟเวอร์เสร็จสิ้นการติดตั้ง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โปรแกรมส่วนเซิร์ฟเวอร์ถูกติดตั้งและเรียกใช้งานจาก Start Program Menu ใน Windows



รูปที่ ข(8) โปรแกรมส่วนเซิร์ฟเวอร์เรียกจาก Start Program ใน Windows

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ค
คู่มือการใช้งาน
โปรแกรมระบบฝากข้อความเสียงผ่านเครือข่ายอินเทอร์เน็ตส่วนไกลเอนท์

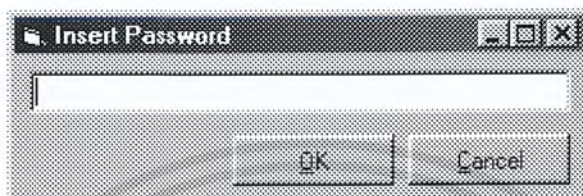


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

คู่มือการใช้งาน

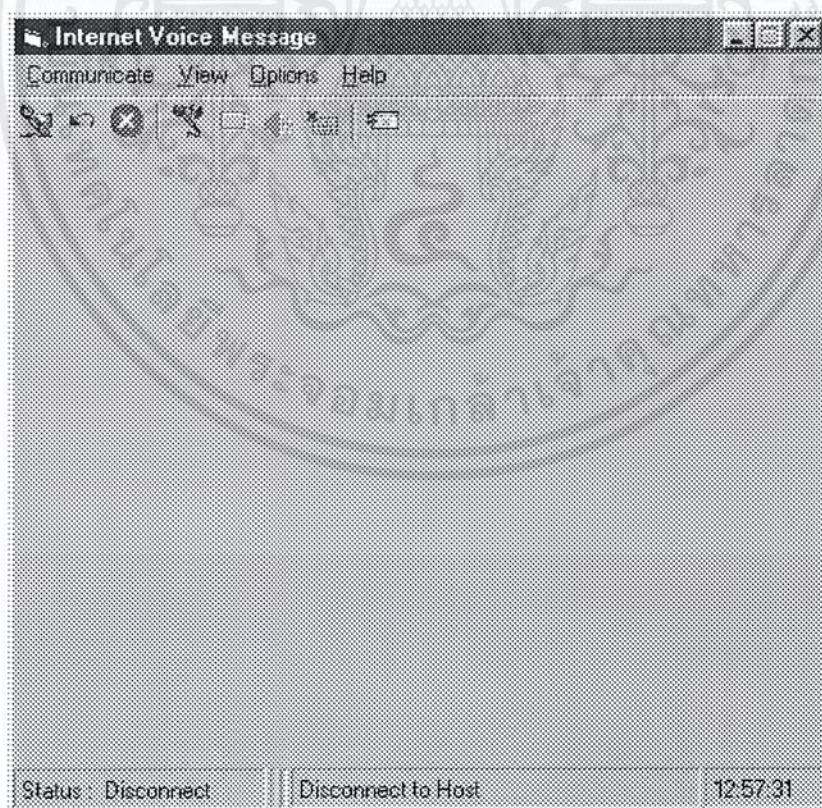
โปรแกรมระบบฝากข้อความเสียงผ่านเครือข่ายอินเทอร์เน็ตส่วนไคลเอนท์

เมื่อเริ่มใช้โปรแกรมในกรณีที่มีการตั้ง Password ไว้ในส่วนของ Security จะต้องใส่ Password ก่อนการใช้โปรแกรม ถ้าไม่มีการตั้ง Password จะข้ามไคอะด็อก Insert Password ไป



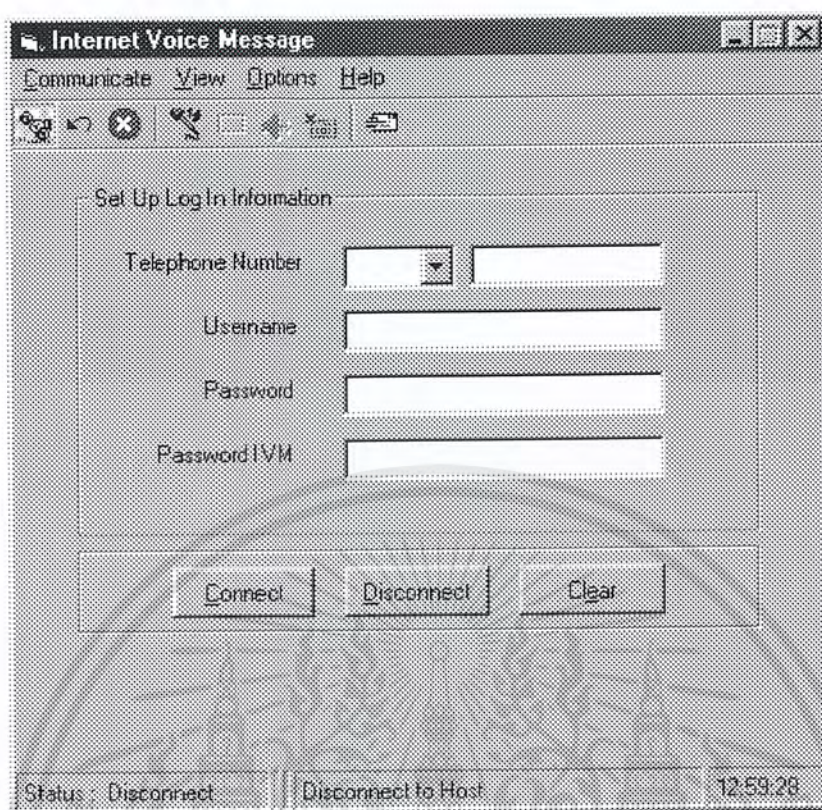
รูปที่ ค(1) ไคอะด็อก Insert Password

เมื่อข้ามส่วนไคอะด็อก Insert Password ก็จะปรากฏในส่วนหน้าจอโปรแกรมหลักซึ่งจะมีหน้าที่หลักอยู่ 2 ส่วนด้วยกันคือ ส่วนของการส่งไฟล์ และส่วนของการบันทึกเสียง



รูปที่ ค(2) ไคอะด็อก Main Program

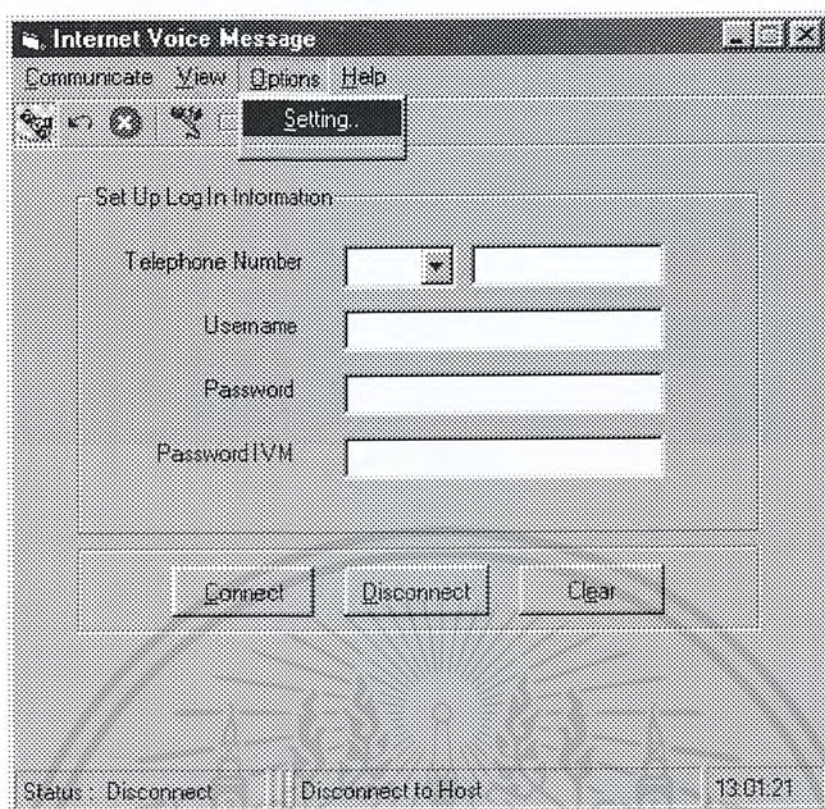
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ ค(3) ไอโอดีอก Quick Connect

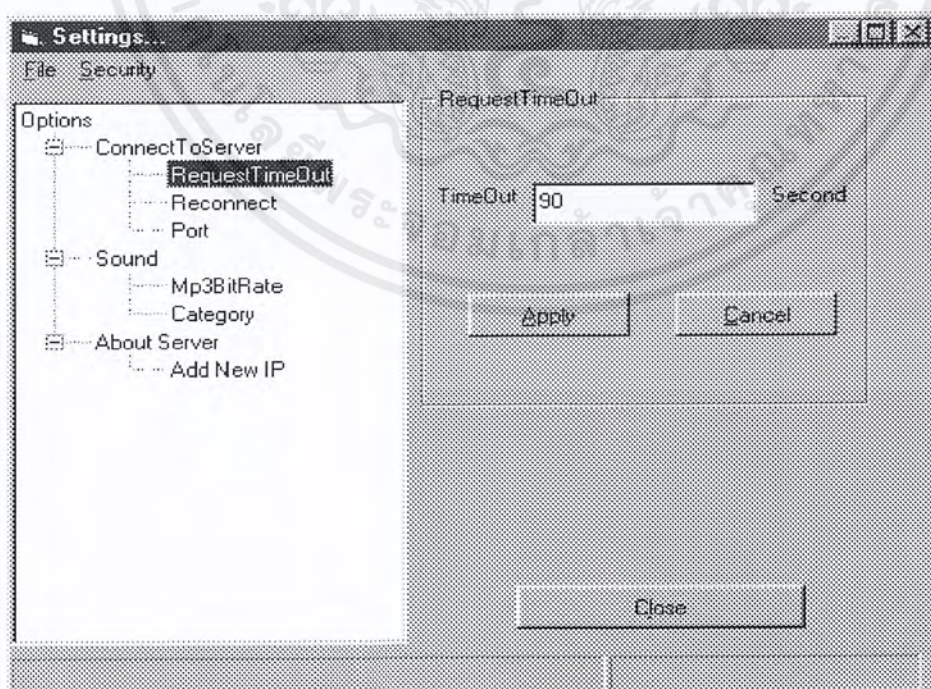
- ปุ่มที่ 1 เมื่อคลิกแล้วจะปรากฏช่องเพื่อกรอกข้อมูลส่วนที่ใช้ติดต่อกับเซิร์ฟเวอร์ (Quick Connect)
- ปุ่มที่ 2 ทำหน้าที่ในการติดต่อซ้ำอีกครั้งในกรณีที่ไม่สามารถติดต่อได้ในครั้งแรก (Reconnect)
- ปุ่มที่ 3 ทำหน้าที่ในการยกเลิกการติดต่อกับเซิร์ฟเวอร์ (Disconnect)
- ปุ่มที่ 4 ทำหน้าที่ในการเริ่มบันทึกเสียงเพื่อส่งไปยังเซิร์ฟเวอร์ (Record)
- ปุ่มที่ 5 ทำหน้าที่หยุดการบันทึกเสียง (Stop)
- ปุ่มที่ 6 ทำหน้าที่เล่นเสียงที่บันทึกก่อนจะทำการส่งไปยังเซิร์ฟเวอร์ (Play)
- ปุ่มที่ 7 ทำหน้าที่ลบไฟล์เสียงที่บันทึก (Delete)
- ปุ่มที่ 8 ทำหน้าที่ในการส่งไฟล์เสียงไปยังเซิร์ฟเวอร์ที่กำหนด (Connect)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ ค(4) ไปคลิกเมนู Option > Setting..

เมื่อทำการเลือกเมนู Option และเลือก Setting.. จะปรากฏไปคลิก Setting.. ดังนี้



รูปที่ ค(5) แสดงการ Setting Timeout

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หมวด ConnectToServer จะประกอบไปด้วย

RequestTimeOut เป็นส่วนที่กำหนดระยะเวลาในการ Connect มีหน่วยเป็นวินาที

Reconnect เป็นส่วนที่กำหนดจำนวนครั้งในการ Reconnect

Port เป็นการกำหนดหมายเลขพอร์ตในการติดต่อ

หมวด Sound จะประกอบไปด้วย

Mp3BitRate เป็นส่วนที่กำหนด BitRate ค่า Default จะกำหนดไว้ที่ 32000

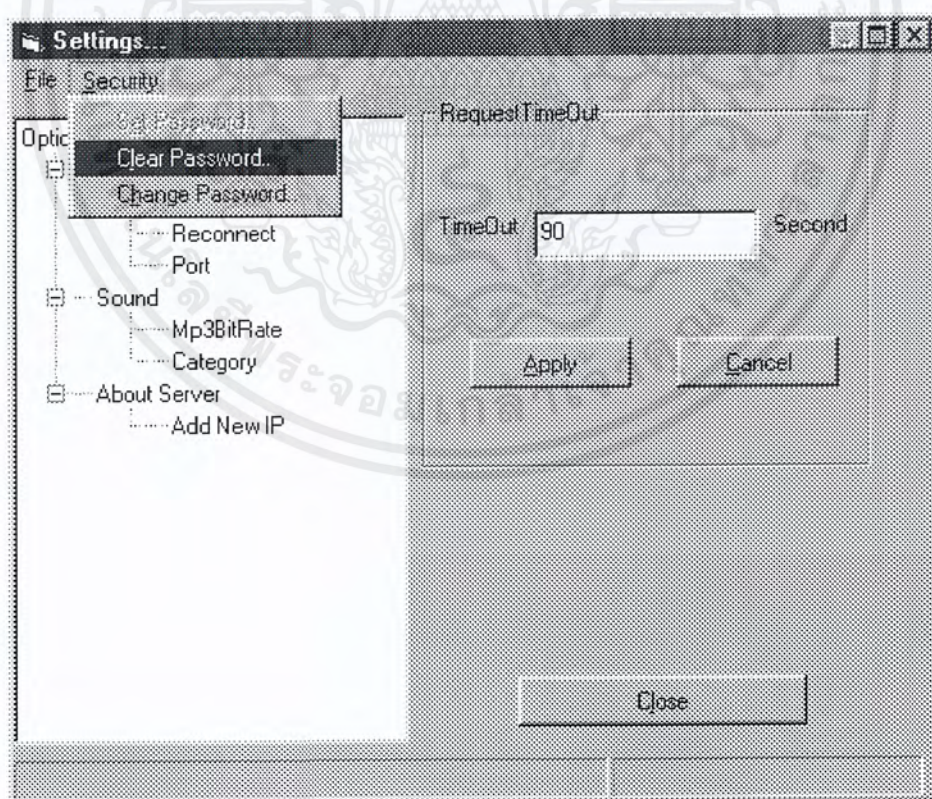
Category เป็นส่วนที่กำหนดว่าจะส่งไฟล์แบบ Wav หรือ MP3

หมวด About Server ประกอบไปด้วย

AddNewIP เป็นส่วนที่ทำหน้าที่บันทึกข้อมูลในการติดต่อกับเซิร์ฟเวอร์

กำหนดหมายเลข Password ก่อนใช้โปรแกรม

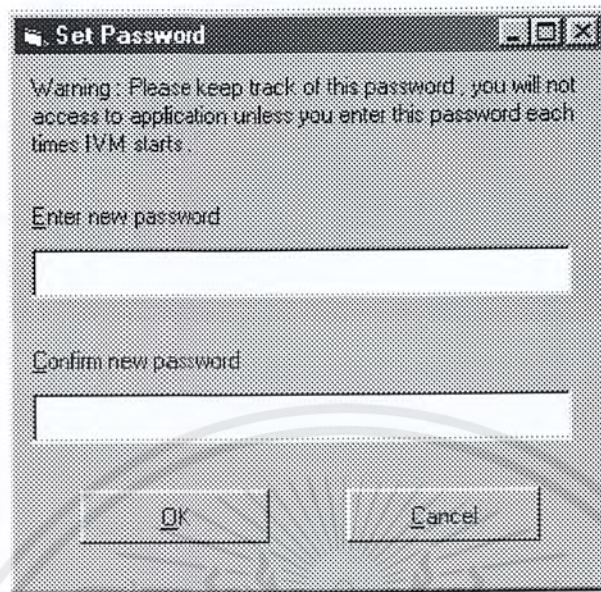
สามารถที่จะตั้ง Password ก่อนมีการเข้าใช้โปรแกรม เพื่อป้องกันมิให้ผู้อื่นสามารถเข้าใช้โปรแกรมนี้ได้



รูปที่ ค(6) แสดงการเลือกเมนูเปลี่ยน Password

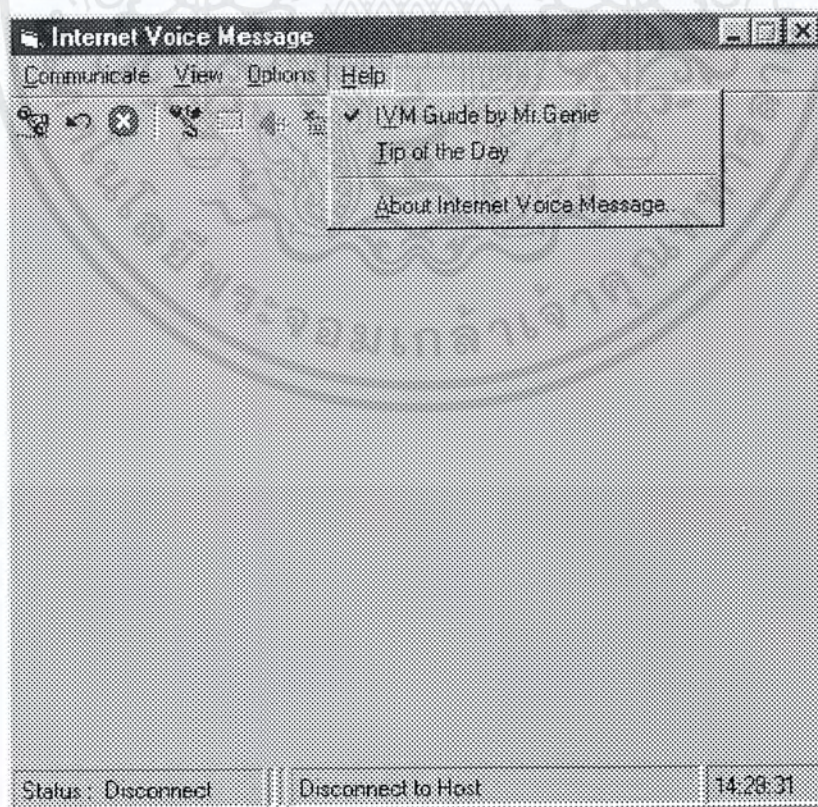
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เมื่อเลือก Set Password จะปรากฏหน้าจอ Set Password ดังรูปดังนี้



รูปที่ ค(7) หน้าจอ Set Password

เมื่อเลือกเมนู Help จะมีเมนูปรากฏดังรูป



รูปที่ ค(8) หน้าจอเมนู Help

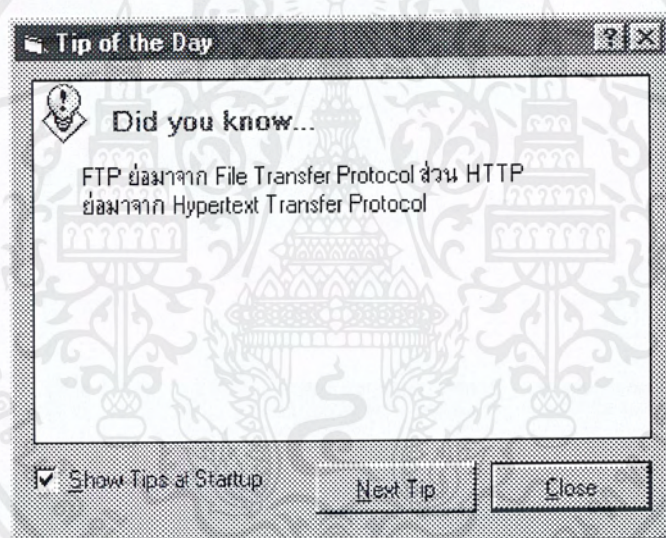
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เมื่อเลือก IVM Guide by Mr.Genie จะปรากฏตัวแอนิเมชันดังรูป



รูปที่ ๑(๙) Mr. Genie ตัวการ์ตูนแสดงความช่วยเหลือ

เมื่อเลือก Tip of the Day จะปรากฏดังรูปดังนี้



รูปที่ ๑(๑๐) หน้าจอ Tip of the Day

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ง
คู่มือการใช้งาน
โปรแกรมระบบฝากข้อความเสียงผ่านเครือข่ายอินเทอร์เน็ตส่วนเซิร์ฟเวอร์

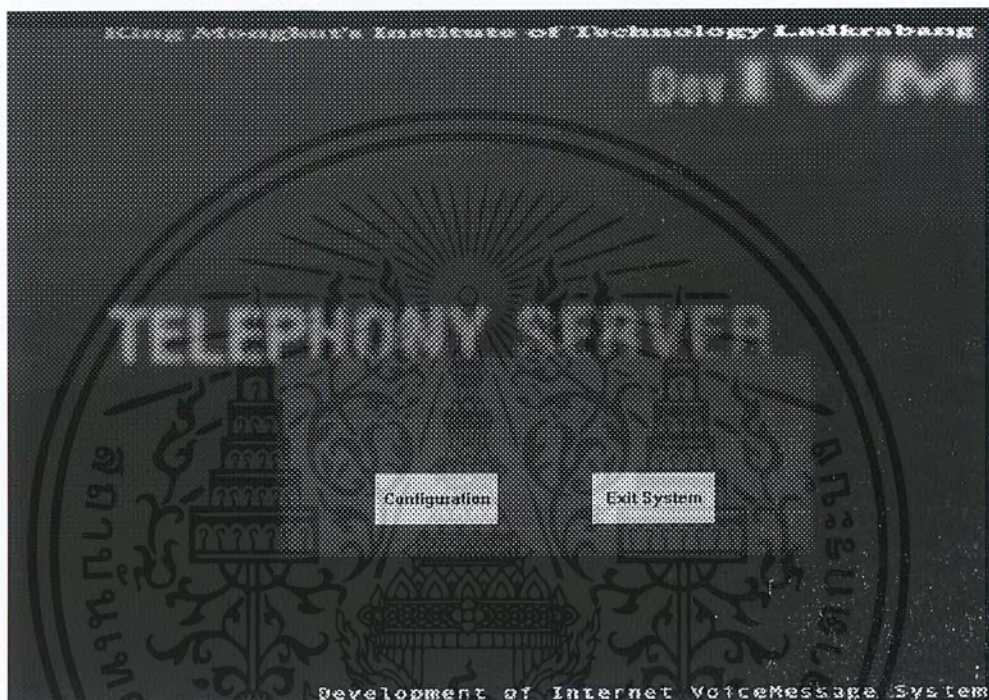


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

คู่มือการใช้งาน

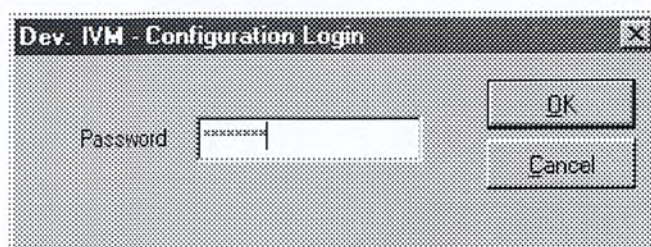
โปรแกรมระบบฝากข้อความเสียงผ่านเครือข่ายอินเทอร์เน็ตส่วนเซิร์ฟเวอร์

โปรแกรมระบบฝากข้อความเสียงผ่านเครือข่ายอินเทอร์เน็ตส่วนเซิร์ฟเวอร์ (DevIVM Telephony Server) เมื่อเปิดโปรแกรมส่วนนี้จะปรากฏหน้าจอ ดังรูป



รูปที่ ง(1) หน้าจอแรกโปรแกรมระบบฝากข้อความเสียงผ่านเครือข่ายอินเทอร์เน็ตส่วนเซิร์ฟเวอร์ (DevIVM Telephony Server)

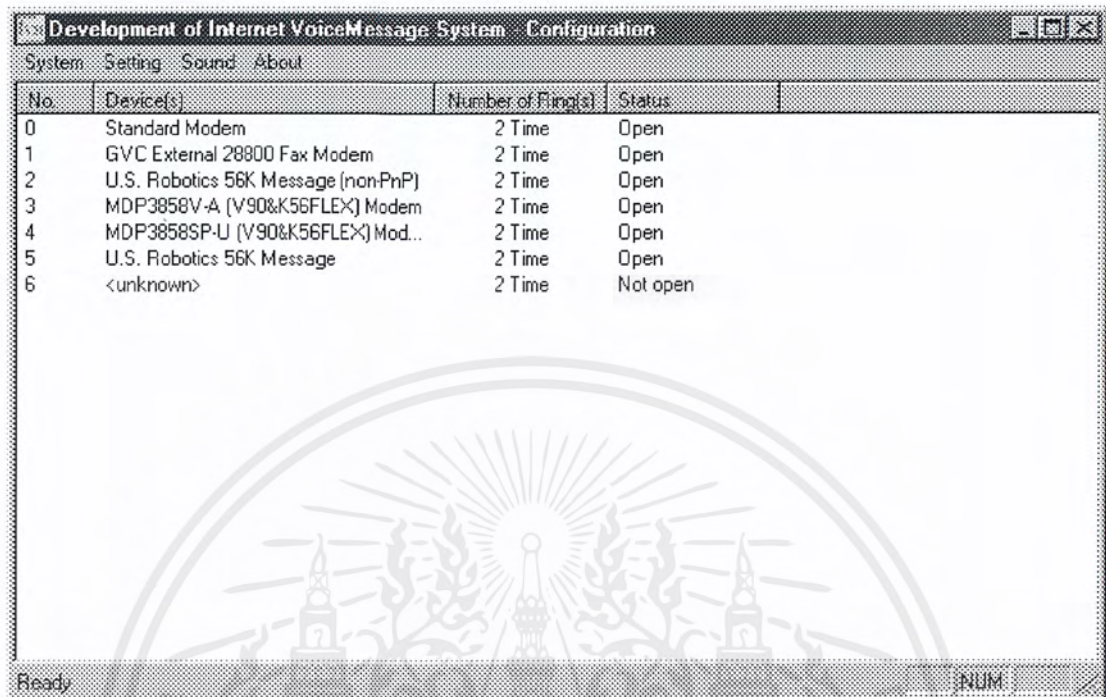
เมื่อกดปุ่ม Configuration จะปรากฏหน้าจอให้พิมพ์ Password ดังรูป



รูปที่ ง(2) หน้าจอเช็ค Password เพื่อเข้าสู่ระบบ Configuration

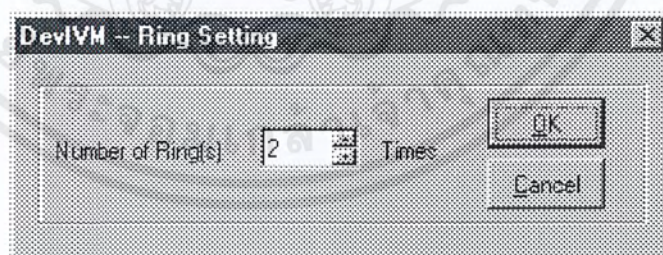
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เมื่อเข้าสู่ระบบ จะพบหน้าจอเมนูหลักและแสดงสถานะของระบบได้แก่ อุปกรณ์ จำนวนครั้งที่ตั้ง และสถานะของ Voice Modem ในขณะนั้น



รูปที่ ง(3) หน้าจอเมนูหลักและส่วนแสดงสถานะระบบ

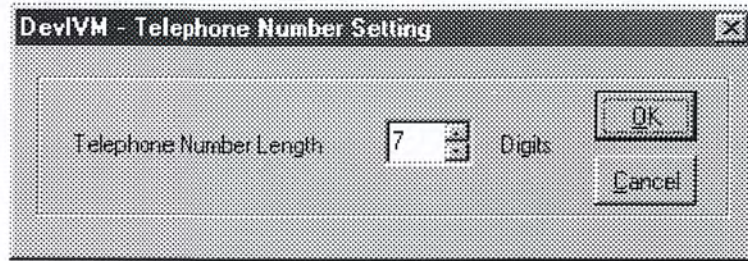
ในการแก้ไขระบบ ส่วนที่สามารถแก้ไขได้ คือ จำนวนครั้งที่ตั้ง ที่เมนู Ring Setting



รูปที่ ง(4) หน้าจอแสดงการแก้ไขจำนวนครั้งที่ตั้ง

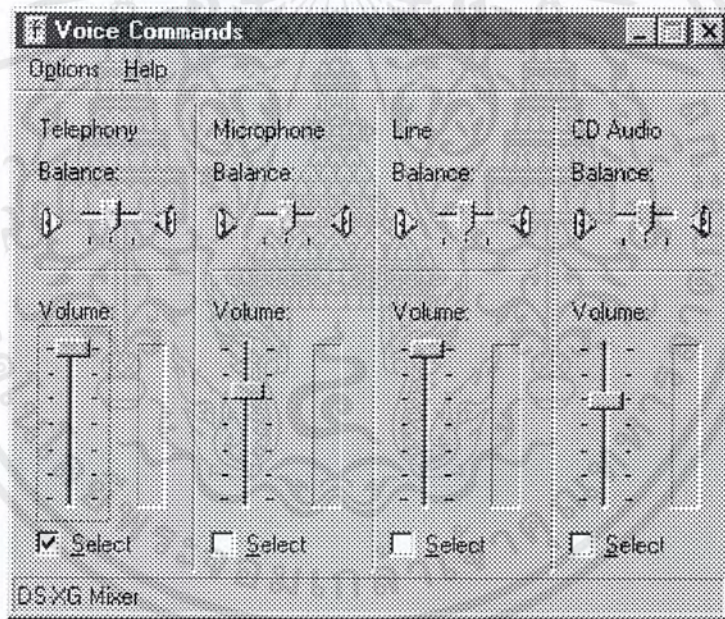
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

อีกส่วนหนึ่งที่แก้ไขได้ คือ จำนวนความยาวของเบอร์โทรศัพท์ที่ปรับไปตามท้องถิ่นต่างๆ



รูปที่ ง(5) หน้าจอแสดงการแก้ไขความยาวของเบอร์โทรศัพท์

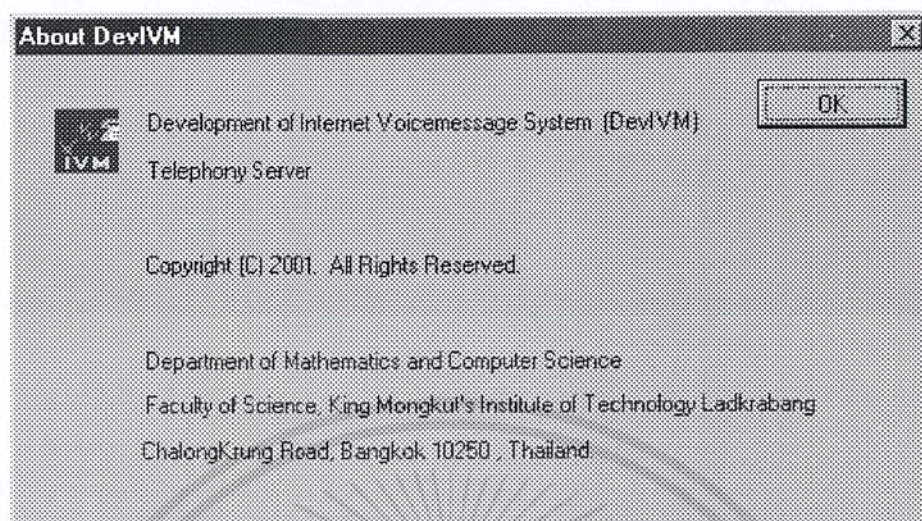
อีกส่วนหนึ่งที่แก้ไขได้ คือ ระดับความดัง-เบาของเสียงโมเด็ม



รูปที่ ง(6) แสดงหน้าจอการเปลี่ยนระดับความดัง-เบาของเสียงโมเด็ม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ในส่วนของคุณข้อมูลเกี่ยวกับโปรแกรมสามารถเรียกดูได้ที่เมนู About



รูปที่ ๗(๗) แสดงหน้าจอข้อมูลเกี่ยวกับโปรแกรมส่วนเซิร์ฟเวอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บรรณานุกรม

ธาริน ทิทธิธรรมชารี. 2543. คู่มือการเขียนโปรแกรม Microsoft Visual Basic Version 6.0.

กรุงเทพฯ : ชัคเชส มีเดีย.

นิรุช อำนวยศิลป์. 2542. คู่มือการเขียนโปรแกรม Microsoft Visual C++ Version 6.0. กรุงเทพฯ

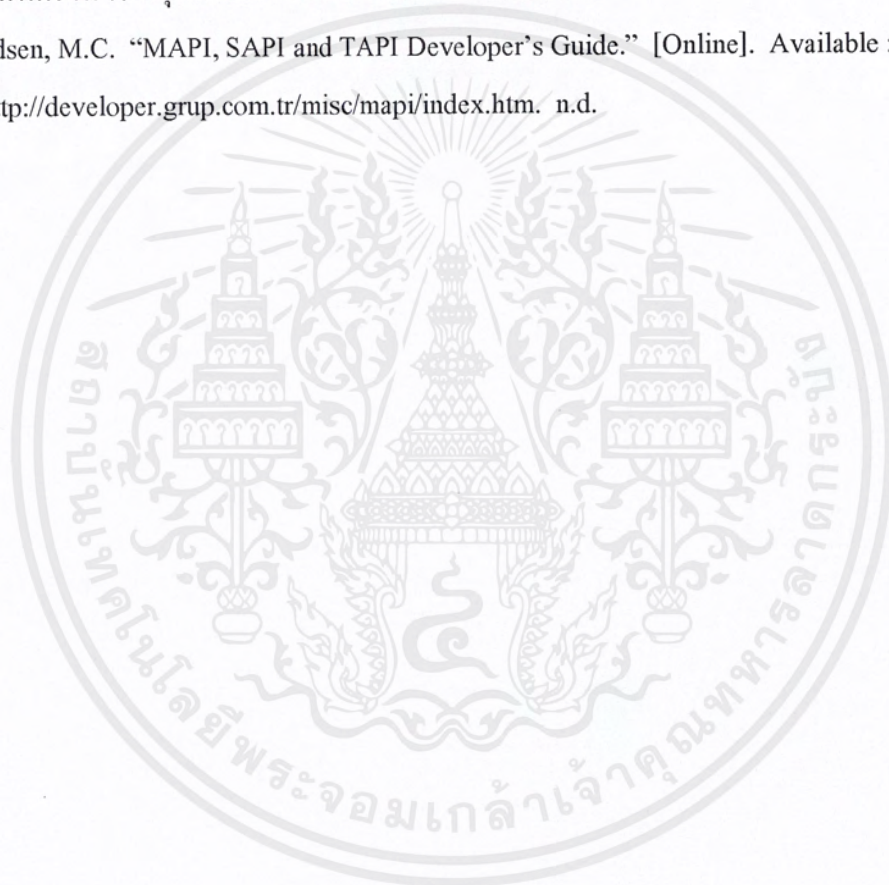
: ชัคเชส มีเดีย.

กิตติ ภัคดีวิวัฒน์กุล และจำลอง ครูอุตสาหะ. 2543. Visual Basic 6.0 ฉบับโปรแกรมเมอร์.

พิมพ์ครั้งที่ 7. กรุงเทพฯ : เคทีพี คอมพ์แอนด์คอนซัลท์.

Amundsen, M.C. "MAPI, SAPI and TAPI Developer's Guide." [Online]. Available :

<http://developer.gup.com.tr/misc/mapi/index.htm>. n.d.



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้