ซอฟท์แวร์จำลองการทำงานของรหัสลับเชิงควอนตัม

QUANTUM CRYPTOGRAPHY DEMONSTRATION SOFTWARE

โดย

นายขวัญชัย ทองลอย

นายณัฐพล นารอด

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต

ภาควิชาวิศวกรรมสารสนเทศ

คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2548

QUANTUM CRYPTOGRAPHY DEMONSTRATION SOFTWARE

BY

MR. KHWANCHAI THONGLOY

MR. NATTHAPHON NAROT

A THESIS SUBMITTED IN PARTIAL FULFILLMENT OF

THE REQUIREMENT FOR THE DEGREE OF

BACHELOR IN DEPARTMENT OF INFORMATION ENGINEERING

FACULTY OF ENGINEERING

KING MONGKUT'S INSTITUTE OF TECHNOLOGY LADKRABANG

2005

หัวข้อปริญญานิพนธ์	**ซอฟท์แวร์จำลองการทำงานของรหัสลับเชิงควอนตัม**		
ชื่อนักศึกษา	**นายขวัญชัย**	**ทองลอย**	**รหัสประจำตัว 45010075**
	นายณัฐพล	**นารอด**	**รหัสประจำตัว 45010232**
อาจารย์ที่ปรึกษา	**รศ. นภพินทุ์**	**อนันตรศิริชัย**	
ที่ปรึกษาร่วม	**ดร.เกียรติศักดิ์**	**ศรีพิมานวัฒน์**	
ระดับการศึกษา	ปริญญาตรี วิศวกรรมศาสตรบัณฑิต		
	สาขาวิศวกรรมสารสนเทศ		
ภาควิชา	วิศวกรรมสารสนเทศ		
ปีการศึกษา	2548		

ปริญญานิพนธ์ฉบับนี้ได้รับการอนุมัติเป็นส่วนหนึ่งของการศึกษาตามหลักสูตรวิศวกรรมศาสตร์บัณฑิต คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

.....

(รองศาสตราจารย์ นภพินทุ์ อนันตรศิริชัย)

อาจารย์ที่ปรึกษา

หัวข้อปริญญานิพนธ์	ซอฟท์แวร์จำลองการใช้รหัสลับเชิงควอนตัม		
ชื่อนักศึกษา	นายขวัญชัย	ทองลอย	รหัสประจำตัว 45010075
	นายณัฐพล	นารอด	รหัสประจำตัว 45010232
อาจารย์ที่ปรึกษา	รศ. นภพินทุ์	อนันตรศิริชัย	
ที่ปรึกษาร่วม	ดร.เกียรติศักดิ์	ศรีพิมานวัฒน์	
ระดับการศึกษา	ปริญญาตรี วิศวกรรมศาสตรบัณฑิต		
	สาขาวิศวกรรมสารสนเทศ		
ภาควิชา	วิศวกรรมสารสนเทศ		
ปีการศึกษา	2548		

บทคัดย่อ

ปริญญานิพนธ์นี้กล่าวถึงการสร้างซอฟท์แวร์จำลองการทำงานของรหัสลับเชิงควอนตัม (Quantum Cryptography Demonstration Software) เพื่อใช้ในการป้องกันการโจรกรรมข้อมูลดิจิตอลจากบุคคลที่สาม โดยทำการสร้างและพัฒนาซอฟท์แวร์จำลองการรับและส่งข้อมูลดิจิตอลระหว่างเครื่องคอมพิวเตอร์ 2 เครื่อง ประยุกต์ใช้เทคโนโลยีของ Quantum Cryptography เพื่อส่งข้อมูลที่เข้ารหัสลับผ่าน Public Network ซึ่งในการเข้ารหัสลับจะใช้คีย์ (Key) ที่ได้จากการสุ่มโดยโปรแกรมคอมพิวเตอร์แทนคีย์ที่ได้จาก Quantum Random Generation และจำลองการส่งคีย์ (Key) ผ่านทาง Serial Port แทนการส่งผ่านช่องสัญญาณควอนตัม รวมทั้งมีโปรโตคอลในการตรวจสอบความปลอดภัยของข้อมูลด้วยโปรโตคอล BB84 ทั้งหมดนี้เป็นการเริ่มต้นพัฒนาซอฟท์แวร์กับระบบวิทยาการรหัสลับเชิงควอนตัมซึ่งเป็นเทคโนโลยีสมัยใหม่ที่ศูนย์ศูนย์เทคโนโลยีอิเล็กทรอนิคส์และคอมพิวเตอร์แห่งชาติกำลังพัฒนา

Thesis Title	*Quantum Cryptography Demonstration Software*	
Student	Mr. Khwanchai Thongloy	ID. 45010075
	Mr. Natthaphon Narot	ID. 45010232
Advisor	Assoc. Prof. Noppin Anantrasirichai	
	Dr. Keatisak Sripimanwat	
Graduate Level	Bachelor Degree of Information Engineering	
Department	Information Engineering	
Academic Year	2005	

ABSTRACT

The Topic of this thesis is "Quantum Cryptography Demonstration software" for protect hacking information form hacker. In this project we create software for demonstration of communicated between 2 computers. Before being sent through public network, the data are encrypted. In addition to processing the encryption, this software automatically uses the key from computer's random value instead of Quantum Cryptography Circuit and also used RS-232 (COM port) connection instead of Quantum Channel. This software have a process for correct a key that send via RS-232 by use BB84 Protocol. This Project is a beginning point for software development of Quantum Cryptography System that develop by NECTEC

กิตติกรรมประกาศ

ปริญญานิพนธ์ฉบับนี้สำเร็จลุล่วงได้ด้วยดี ต้องขอกราบขอบพระคุณบิดา มารดาที่ให้กำลังใจเสมอมา ขอขอบพระคุณ รศ. นภพินทุ์ อนันตรศิริชัย อาจารย์ที่ปรึกษาที่ได้ช่วยเหลือชี้แนะสิ่งต่าง ๆตลอดมา และขอขอบพระคุณ ดร.เกียรติศักดิ์ ศรีพิมานวัฒน์ ที่ได้ช่วยเหลือในด้านความรู้ใหม่ ๆ สนับสนุนเรื่องอุปกรณ์และเครื่องมือในการทำโครงงานและปริญญานิพนธ์ฉบับนี้เป็นอย่างดี

สุดท้ายนี้ทางคณะผู้จัดทำ ขอขอบคุณ อาจารย์ทุกท่านที่กรุณาประสิทธิ์ประสาทวิชาความรู้ รวมทั้งแนวทางความคิด แนวทางปฏิบัติ และแนวความคิดใหม่ ๆที่ทันเหตุการณ์ในปัจจุบันแก่คณะผู้จัดทำ จนทำให้ปริญญานิพนธ์ฉบับนี้สำเร็จลุล่วงไปด้วยดี

นายขวัญชัย ทองลอย

นายณัฐพล นารอด

สารบัญ

สารบัญ (ต่อ)

สารบัญรูป

สารบัญรูป(ต่อ)

สารบัญตาราง

บทที่ 1
บทนำ

ปัจจุบันพัฒนาการด้านเทคโนโลยีและวิทยาการด้านต่าง ๆ โดยเฉพาะวิทยาการเกี่ยวกับการสื่อสารและสารสนเทศ ได้มีการคิดค้นและพัฒนาสร้างสรรค์อย่างต่อเนื่อง เพื่อตอบสนองความต้องการของมนุษย์รวมทั้งความต้องการขององค์กรต่าง ๆ ทั่วโลก และหลายๆเทคโนโลยีได้มีการพัฒนาออกมาเป็นผลิตภัณฑ์เข้าสู่ตลาดเทคโนโลยี โดย 1 ใน 20 เทคโนโลยีที่น่าสนใจในปัจจุบันก็คือ ระบบวิทยาการรหัสลับเชิงควอนตัม (Quantum Cryptography System) เป็นเทคโนโลยีด้านการรักษาความปลอดภัยในการส่งข้อมูลข่าวสารที่ต้องการความปลอดภัยและเป็นความลับสูงสุด ซึ่งในหลาย ๆ ประเทศกำลังทำการวิจัยและพัฒนา และในหลาย ๆ ประเทศได้มีการพัฒนาออกมาเป็นผลิตภัณฑ์และขายในตลาดเทคโนโลยีแล้วเช่นกัน รวมทั้งประเทศไทยซึ่งได้เล็งเห็นถึงความสำคัญของวิทยาการรหัสลับเชิงควอนตัมต่อการใช้งานในอนาคต กล่าวคือหากความสามารถของคอมพิวเตอร์สูงขึ้นมากเท่าใด ความปลอดภัยในการสื่อสารในปัจจุบันก็มีข้อเสียในเรื่องของความปลอดภัยมากเท่านั้น ซึ่งในอนาคตวิทยาการรหัสลับเชิงควอนตัมจะต้องเข้ามามีบทบาทในการสื่อสารทั่วไปอย่างแน่นอน ซึ่งปัจจุบันศูนย์เทคโนโลยีอิเล็กทรอนิคส์และคอมพิวเตอร์แห่งชาติ (NECTEC) ก็กำลังดำเนินการวิจัยและพัฒนาอย่างต่อเนื่อง ดังนั้นจึงได้มีการวางแผนจัดทำโครงงานในส่วนของ ซอฟท์แวร์จำลองการใช้งานรหัสลับเชิงควอนตัม เพื่อเป็นองค์ประกอบและทำการศึกษาร่วมกับโครงงานหลักของศูนย์เทคโนโลยีอิเล็กทรอนิคส์และคอมพิวเตอร์แห่งชาติ (NECTEC) โดยจะพัฒนาในส่วนของซอฟท์แวร์โดยพัฒนาคู่ขนานกับโครงงานของ NECTEC ซึ่งกำลังพัฒนาในส่วนของฮาร์ดแวร์ อุปกรณ์ทางแสง และอื่น ๆ แต่เนื่องจากการพัฒนาซอฟท์แวร์จำเป็นต้องมีการทดสอบการใช้งาน จึงได้มีการสร้างระบบจำลองเพื่อใช้ในการทดสอบ รวมทั้งหาอุปกรณ์ทางฮาร์ดแวร์มาใช้สำหรับทดสอบซอฟท์แวร์อีกด้วย (อย่างเช่นการใช้ Serial Port แทนช่องสัญญาณควอนตัม ซึ่งเป็นอุปกรณ์ทางแสง) ซึ่งอุปกรณ์ที่จะใช้ทดสอบจะเป็นอุปกรณ์ที่หาง่ายสะดวกในการใช้งาน และนำไปใช้ร่วมกับโครงงานของศูนย์เทคโนโลยีอิเล็กทรอนิคส์และคอมพิวเตอร์แห่งชาติ (NECTEC) ต่อไป

1.1 แนวคิดและที่มาของปัญหา

ปัจจุบันโลกของเรามีพัฒนาการด้านเทคโนโลยีอย่างต่อเนื่องและรวดเร็วมาก ประเทศใดที่สามารถคิดค้นและนำเทคโนโลยีสมัยใหม่มาใช้ให้เกิดประโยชน์ได้ ก็จะสามารถพัฒนาประเทศได้อย่างรวดเร็ว สามารถประหยัดงบประมาณในการซื้อเทคโนโลยีต่าง ๆ จากต่างประเทศซึ่งมักมีราคาที่สูงมากได้อีกด้วย

องค์กรต่าง ๆ ทั่วโลกมีความจำเป็นในการสื่อสาร และมีความจำเป็นต้องมีการรับ-ส่งข้อมูลอยู่ตลอดเวลา ซึ่งบางครั้งข้อมูลที่ต้องการส่งนั้นมีความสำคัญ และต้องการความปลอดภัยในการส่งข้อมูลสูงสุด เพราะหากความลับหรือข้อมูลข่าวสารนั้นถูกลักลอบดักฟังขณะทำการรับ-ส่งข้อมูล ก็จะทำให้เกิดความเสียหายต่อองค์กร หรือประเทศได้ เช่น องค์กรทางการทหาร ธนาคาร บริษัทประมูล เป็นต้น แต่ในเทคโนโลยีที่ใช้ในการรับส่งข้อมูลในปัจจุบันก็คือการใช้รหัสลับ (Cryptography) ซึ่งให้ความสำคัญกับความยากในการถอดรหัสลับ แต่เมื่อความสามารถของคอมพิวเตอร์สูงขึ้นเรื่อย ๆ จะสามารถถอดรหัสข้อมูลโดยปราศจากกุญแจไขความลับได้ไม่ยาก สักวันวิธีการในปัจจุบันก็จะไม่สามารถใช้งานได้อีกต่อไป

ปัจจุบันมีการใช้เทคโนโลยีรหัสลับกันอย่างแพร่หลาย แต่มีข้อสังเกตหลายประการดังนี้

1.1.1 การเข้ารหัสนั้นมุ่งเน้นที่ความยากในการที่จะถอดรหัส ต้องใช้คอมพิวเตอร์ที่มีความสามารถสูง ร่วมกับระยะเวลาที่นาน สำหรับใช้ในการถอดรหัส ซึ่งในอนาคตเทคโนโลยีพัฒนาไปมากขึ้น ความสามารถของคอมพิวเตอร์ก็ย่อมสูงมากขึ้นเรื่อยๆ แน่นอนว่าการเข้ารหัสที่เน้นความยากในการถอดรหัส ก็ไม่ใช่เรื่องยากอีกต่อไป

1.1.2. การเข้ารหัสในปัจจุบัน ใช้วิธีเข้ารหัสด้วยคีย์เดิม หรือใช้ซ้ำๆ นานๆถึงจะทำการเปลี่ยนแปลง ทำให้ Hacker ที่สามารถถอดรหัสข้อมูลได้แล้วหรือค้นพบคีย์ที่ใช้ในการถอดรหัสได้แล้ว ก็สามารถนำไปถอดรหัสข้อมูลอื่นๆที่เข้ารหัสด้วยคีย์ตัวเดิมได้

1.1.3. การเปลี่ยนคีย์บ่อยๆทำได้ยาก เพราะต้องจดจำว่ามีคีย์ตัวใดใช้กับข้อมูลใด และหากเราสร้างคีย์โดยใช้ฟังชันก์ Random ในคอมพิวเตอร์ ก็จะไม่ใช่การสุ่มที่ 100% อีกด้วย

เทคโนโลยีหนึ่งที่น่าจับตามอง และจะสามารถมาแทนที่การวิทยาการรหัสลับในปัจจุบันได้ก็คือ ระบบวิทยาการรหัสลับเชิงควอนตัม (Quantum Cryptography System) เป็นระบบวิทยาการรหัสลับที่ความปลอดภัย "กุญแจไขความลับ" สามารถรับประกันด้วยกฎพื้นฐานทางควอนตัมฟิสิกส์ รวมทั้งระบบนี้ยังสามารถตรวจจับผู้ลักลอบดักฟังข้อมูลได้เสมอ

ในระบบวิทยาการรหัสลับเชิงควอนตัมนี้จะถูกใช้ในการรับส่งข้อมูลที่ต้องการความปลอดภัยสูงสุด โดยในระบบจะมีการสร้างกุญแจไขความลับจากอุปกรณ์ทาง Hardware ที่เรียกว่า Quantum Random Generation ซึ่งมีความสามารถในการสุ่ม 100% ระบบวิทยาการรหัสลับเชิงควอนตัมนั้นจะให้ความสำคัญกับความปลอดภัยของกุญแจไขความลับมาก โดยจะต้องทำการส่งกุญแจไขความลับไปในช่องทางที่มีความปลอดภัยสูงสุด ซึ่งในระบบนี้ช่องทางที่ใช้ส่งกุญแจไขความลับเรียกว่า ช่องสัญญาณควอนตัม (Quantum Channel) ซึ่งก็คือการส่งผ่านสาย Fiber Optic โดยทำการยิงโฟตอนเดี่ยว ผ่านช่องสัญญาณ ในการส่งกุญแจไขความลับ ตามหลักความไม่แน่นอนของไฮเซนเบิร์ก ได้ระบุไว้ว่าแสงอนุภาคเดี่ยว หรือโฟตอน เมื่อมีการรบกวนหรือมีการตรวจจับจะทำให้คุณสมบัติของโฟตอนเปลี่ยนไป ซึ่งตามหลักการนี้นี่เองจะเป็นส่วนหนึ่งที่ระบบนี้จะใช้ในการวัดความปลอดภัยของกุญแจไขความลับได้ โดยระบบจะมีรูปแบบการตรวจสอบ (Protocol) ที่ใช้ในการตรวจสอบความถูกต้องของคีย์ที่จะทำให้สามารถยืนยันว่ากุญแจไขความลับนั้นปลอดภัย หากพบว่าข้อมูลของกุญแจไขความลับนั้นมีความผิดพลาดเกินที่กำหนดไว้ จะถือว่ามีการดักจับข้อมูล ผู้ส่งจะยกเลิกกุญแจไขความลับนั้น และทำการสุ่มกุญแจไขความลับและส่งใหม่ หากพบว่าปลอดภัยภาครับจะทำการยืนยันความปลอดภัยไปที่ภาคส่งผ่านช่องสัญญาณสาธารณะ และก็ใช้กุญแจไขความลับนั้นเข้ารหัสข้อมูลด้วยวิธีการที่ได้ออกแบบไว้และส่งตามช่องสัญญาณสาธารณะทั่วไป

เนื่องจากการพัฒนาซอฟท์แวร์นั้นจำเป็นต้องมีการทดสอบและทดลอง จึงมีความจำเป็นที่ต้องสร้างระบบจำลองขึ้นเพื่อที่จะใช้ในการทดสอบ ซึ่งอุปกรณ์ที่เลือกมาใช้จำลองในการทดสอบนั้นจะเลือกอุปกรณ์ที่หาง่าย ราคาถูก เพื่อการพัฒนาสู่ซอฟท์แวร์ที่จะสามารถใช้งานได้ในระบบ Quantum Cryptography System ในอนาคตต่อไป

ภาพรวมของโครงงานเป็นดังนี้

- ทำซอฟท์แวร์เพื่อไปใช้จริงในโครงการร่วมของศูนย์เทคโนโลยีอิเล็กทรอนิคส์และคอมพิวเตอร์แห่งชาติ (NECTEC) โดยนำความรู้ต่าง ๆ มาประยุกต์และพัฒนาซอฟท์แวร์ที่สามารถจำลองการทำงานของวิทยาการรหัสลับเชิงควอนตัม โดยจะพัฒนาคู่ขนานไปกับส่วนของฮาร์ดแวร์ ซึ่งพัฒนาโดย NECTEC แต่การพัฒนาซอฟท์แวร์จำเป็นต้องทดสอบและทดลองจึงได้มีการออกแบบฮาร์ดแวร์เพื่อใช้ทดสอบเบื้องต้นก่อน ดังที่ได้กล่าวไว้ข้างต้น รวมทั้งการรวมเนื้องานด้านรหัสลับมีการรวมรวมวิธีการเข้ารหัสแบบต่าง ๆ มาประยุกต์ใช้ในระบบเพื่อเพิ่มความปลอดภัยให้ระบบการส่งข้อมูลมากขึ้น

- รวมเนื้องานด้านชุดรับส่งทางแสง ซึ่งในโครงงานนี้จะใช้ Serial Port แทน เนื่องจากโครงงานนี้เป็นซอฟท์แวร์จำลอง จึงทำการเลือกอุปกรณ์ที่สามารถนำมาประยุกต์ใช้ได้ง่ายและราคาถูก มาแทนอุปกรณ์ชุดรับส่งทางแสงซึ่งมีราคาที่สูงและยากต่อการติดตั้งในชุดจำลอง

1.2 วัตถุประสงค์

1.2.1 เพื่อสร้างซอฟท์แวร์เพื่อไปใช้จริงในโครงการร่วมของศูนย์ NECTEC โดยเป็นการเริ่มต้นการพัฒนาซอฟท์แวร์โดยให้ใช้ความคิดสร้างสรรค์ในการออกแบบและพัฒนาซอฟท์แวร์โดยจำลองใช้งานกับอุปกรณ์ทางฮาร์ดแวร์ที่หาง่ายและติดตั้งใช้ง่ายง่าย เพื่อเป็นการทดสอบซอฟท์แวร์ที่พัฒนาขึ้น และจำลองภาพรวมของโครงการในระบบวิทยาการรหัสลับเชิงควอนตัมได้

1.2.2 เพื่อทำการเรียนรู้และศึกษาวิธีการในการใช้รหัสลับแบบต่าง ๆ เพื่อนำไปใช้เพิ่มความปลอดภัยของระบบวิทยาการรหัสลับเชิงควอนตัม รวมทั้งศึกษาวิธีการต่าง ๆ เพื่อใช้ในการตรวจสอบความถูกต้องของข้อมูลที่ส่งผ่านช่องสัญญาณควอนตัม

1.2.3 เพื่อทำการเรียนรู้และทำความเข้าใจในระบบวิทยาการรหัสลับเชิงควอนตัมโดยศึกษาจากซอฟต์แวร์จำลอง รวมทั้งสามารถใช้ซอฟต์แวร์จำลองนี้ไปใช้ประโยชน์ในด้านการศึกษาและวิจัยเกี่ยวกับระบบวิทยาการรหัสลับเชิงควอนตัม (Quantum Cryptography System)

1.3 ขอบเขตของโครงงาน

1.3.1 สร้างโปรแกรมจำลองการรับและส่งข้อมูลดิจิตอลระหว่างคอมพิวเตอร์ 2 เครื่องผ่าน Public Network

1.3.2 สร้างโปรแกรมที่ทำหน้าที่ในการ Generate Key แทนการสุ่มคีย์จาก Quantum Random Generation และสร้างโปรแกรมที่มีหน้าที่ส่งผ่านข้อมูลคีย์ ผ่านทาง Serial Port (RS-232) แทนการส่งผ่านช่องสัญญาณควอนตัม เนื่องจากการสุ่มคีย์จาก Quantum Random Generation และช่องสัญญาณควอนตัม เป็นวิธีการที่มีราคาแพงและใช้วิทยาการขั้นสูงซึ่งปัจจุบันยังอยู่ในขั้นตอนการวิจัย โครงงานนี้จึงจำลองโดยการใช้อุปการณ์ที่หาได้และราคาถูกในการสร้างและพัฒนา

1.3.3 ศึกษาทำความเข้าใจในวิธีการเข้ารหัสลับแบบสมมาตร (Symmetric key algorithms) และทำการสร้างโปรแกรมเข้ารหัสข้อมูลแบบต่าง ๆ เพื่อความหลากหลายในการเลือกใช้ของผู้ใช้

1.3.4 ศึกษาและทำความเข้าใจในวิธีการในการตรวจสอบความถูกต้องของข้อมูลที่ส่งผ่านช่องสัญญาณควอนตัม รวมทั้งทำการออกแบบและเขียนโปรแกรมที่สามารถตรวจเช็คความถูกต้องของข้อมูลที่ทำการส่งได้

1.4 ผลที่คาดว่าจะได้รับ

1.4.1 ซอฟท์แวร์สามารถนำไปใช้งานร่วมกับระบบวิทยาการรหัสลับเชิงควอนตัมซึ่งเป็นโครงการร่วมของทาง NECTEC ได้

1.4.2 ซอฟต์แวร์สามารถจำลองการทำงานของ Quantum Cryptography System และสามารถนำไปใช้ประโยชน์ในด้านการศึกษา และการวิจัย ใช้เป็นตัวแบบจำลองในการอธิบายและสาธิตการทำงานของระบบจริงได้

1.4.3 สามารถทำความเข้าใจและวิเคราะห์ระบบ Quantum Cryptography System จากซอฟต์แวร์จำลองที่สร้างขึ้นได้1.4.4 สามารถนำความรู้ทางด้านรหัสลับแบบเก่ามาประยุกต์ใช้กับระบบวิทยาการรหัสลับเชิงควอนตัมได้

1.5 ขั้นตอนการดำเนินงาน

ขั้นตอนการดำเนินการของโครงงาน มีการวางแผนในการดำเนินการโครงงาน โดยจะแบ่งเป็น 5 ส่วนดังนี้

- มิถุนายน 2548 – กรกฎาคม 2548 เป็นช่วงแรกของการดำเนินงาน เป็นการวิเคราะห์ปัญหา ความต้องการต่าง ๆ และทางแก้ไขภายในโครงงาน
- กรกฎาคม 2548– ตุลาคม 2548 เป็นช่วงของการวิเคราะห์ระบบและออกแบบระบบ ซึ่งจะประกอบไปด้วยการออกแบบซอฟท์แวร์และการออกแบบหน้าจอของโปรแกรมที่ใช้ในโครงงาน
- สิงหาคม 2548 – มกราคม 2549 เป็นช่วงของการดำเนินการสร้างผลงานตามโครงงานที่ได้ออกแบบไว้
- สิงหาคม 2549 – กุมภาพันธ์ 2549 เป็นช่วงของการทดสอบและแก้ไขผลงานหรือซอฟท์แวร์ช่วงนี้จะดำเนินงานไปพร้อมๆกับช่วงของการสร้างผลงาน
- กรกฎาคม 2548 – กุมภาพันธ์ 2549 เป็นช่วงของการจัดทำเอกสารปริญญานิพนธ์ ซึ่งส่วนนี้เป็นช่วงเวลานาน เนื่องจากการทำเอกสารนี้ได้ทำไปพร้อมกับการดำเนินงานของโครงงาน และจะมีการแก้ไขและปรับปรุงให้เรียบร้อยสมบูรณ์ในช่วงท้าย

ตารางที่ 1.1 แสดงแผนการดำเนินงาน

ID	Task Name	2005							2006		
		Jun	Jul	Aug	Sep	Oct	Nov	Dec	Jan	Feb	Mar
1	Problem Definition & Get Requirment										
2	Analysis & Design										
3	Software Design										
4	Graphics User Interface Design										
5	Implementation										
6	Generate & Authorization Key										
7	Encryption & Decryption										
8	Transmission on Network										
9	Test & Debug										
10	Documentation										

บทที่ 2
ทฤษฎีที่เกี่ยวข้อง

ระบบวิทยาการรหัสลับเชิงควอนตัม เป็นระบบวิทยาการในเรื่องของการรักษาความปลอดภัยในการส่งข้อมูล ซึ่งมีหลักการสำคัญอยู่ความปลอดภัยของกุญแจไขความลับที่จะใช้ในการเข้ารหัสข้อมูลก่อนที่จะทำการส่ง โดยระบบจะต้องทำการส่งกุญแจไขความลับไปในช่องทางพิเศษ ซึ่งเรียกว่าช่องสัญญาณควอนตัม ซึ่งเป็นช่องทางที่สามารถตรวจสอบความปลอดภัยในการส่งได้แน่นอนด้วยคุณสมบัติทางแสงของโฟตอน และโปรโตคอลที่ใช้ในการตรวจสอบความปลอดภัย ซึ่งสามารถเชื่อถือได้ เมื่อกุญแจปลอดภัยก็จะใช้ในการเข้ารหัสข้อมูล และทำการส่งข้อมูลที่เข้ารหัสแล้วไปยังผู้รับผ่านช่องสัญญาณสาธารณะทั่วไป

ในการสร้างซอฟท์แวร์จำลองการทำงานของรหัสลับเชิงควอนตัม เพื่อที่จะให้สามารถจำลองการทำงานของระบบวิทยาการรหัสลับเชิงควอนตัมได้นั้น จะต้องมีความรู้ความเข้าใจในระบบวิทยาการรหัสลับเชิงควอนตัมเป็นอย่างดี และต้องใช้ความรู้ในเรื่องของการเข้ารหัสข้อมูลเพื่อใช้ในการเขียนโปรแกรมให้ระบบสามารถที่จะเข้ารหัสข้อมูลได้ นอกจากนี้จะต้องมีความรู้ในด้านโปรแกรมที่ใช้ในการติดต่อสื่อสารผ่านช่องสัญญาณสาธารณะ (Internet) รวมทั้งต้องใช้ความรู้ในด้านการเขียนโปรแกรมเพื่อติดต่ออุปกรณ์ฮาร์ดแวร์ ซึ่งเป็นส่วนของการส่งข้อมูลกุญแจไขความลับไปในช่องทางพิเศษ ซึ่งในโครงงานนี้ได้มีการนำความรู้ในด้านการเขียนโปรแกรมด้วยภาษา Visual Basic 6 ทั้งในส่วนของโปรแกรมการเข้ารหัสลับ การติดต่อกับอุปกรณ์ฮาร์ดแวร์ การสื่อสารผ่านช่องสัญญาณสาธารณะหรือว่า Internet ซึ่งการที่จะสามารถที่จะเขียนโปรแกรมต่าง ๆ ได้นั้นต้องมีความรู้ในทฤษฎีหรือหลักการในส่วนต่าง ๆ ของซอฟท์แวร์ก่อน ซึ่งมีรายละเอียดดังนี้

2.1 ความรู้เบื้องต้นเกี่ยวกับวิทยาการรหัสลับเชิงควอนตัม

วิทยาการรหัสลับเชิงควอนตัม ถูกวิจัยและพัฒนามาตั้งแต่ยุค 1980 ซึ่งนักพัฒนาทั่วโลกต่างให้ความสนใจและคิดค้นวิธีการและระบบตัวอย่างของการใช้วิทยาการรหัสลับเชิงควอนตัม โดยใช้หลักการทางฟิสิกส์ควอนตัม ทฤษฎีแสงอนุภาคเดี่ยวมาใช้

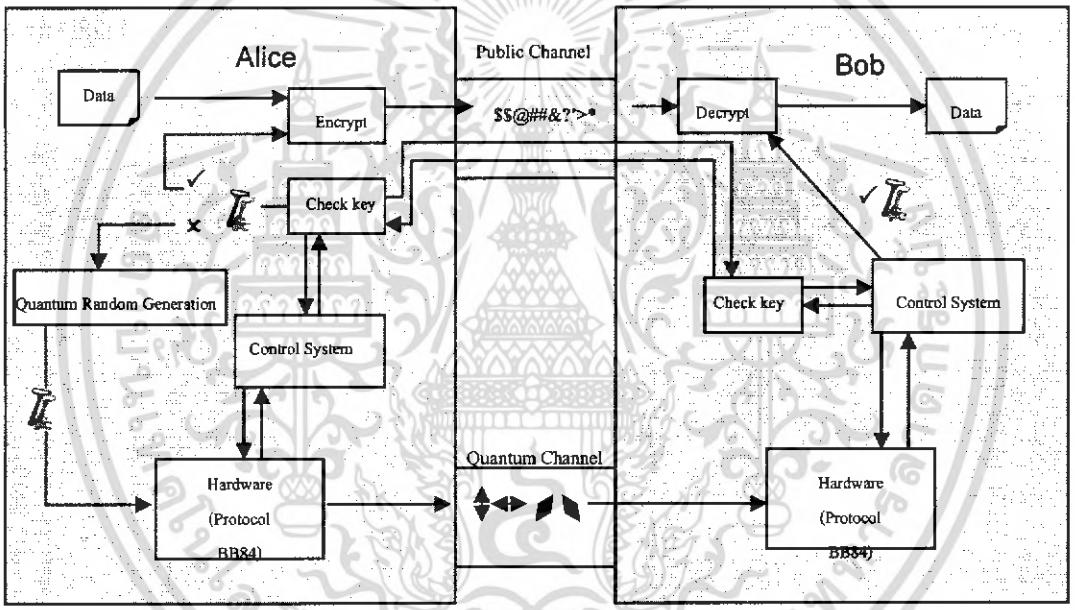
2.1.1 ในระบบวิทยาการรหัสลับเชิงควอนตัมจะมีวิธีการสร้างกุญแจไขความลับแบบสุ่ม 100% ด้วยอุปกรณ์ที่เรียกว่า Quantum Random Generation ด้วยวิธีการทางแสงและหลักความไม่แน่นอนของไฮเซนเบิร์ก

2.1.2 ใช้ no-cloning theorem ในการตรวจสอบการดักจับข้อมูลจากบุคคลที่สาม เป็นโปรโตคอลในการควบคุมและตรวจสอบการถูกดักจับ ซึ่งสามารถตรวจสอบได้แน่นอน

2.1.3 ส่งผ่านกุญแจไขความลับผ่านช่องสัญญาณควอนตัม ซึ่งใช้แสงเป็นตัวส่งผ่านสาย Fiber Optic ซึ่งคุณสมบัติของช่องสัญญาณนี้ยากที่จะดักสัญญาณ และสามารถตรวจสอบความผิดพลาดจนกระทั่งล่วงรู้ถึงการถูกดักจับได้

2.1.4 มีมาตรฐานการตรวจสอบหลังจากได้รับกุญแจไขความลับ โดยทั้งภาครับและภาคส่งจะมีการตรวจสอบคีย์ด้วยวิธีการที่ได้ออกแบบไว้ และมีความน่าเชื่อถือได้

จุดสำคัญของ Quantum Cryptography คือการสร้างคีย์และการส่งคีย์ที่ปลอดภัยจากการถูกดักจับโดยบุคคลที่สาม ซึ่งมีรูปแบบ Protocol ต่าง ๆ ในปัจจุบันที่ใช้อยู่ได้แก่ BB84, B92[5,7], etc.



รูปที่ 2.1 แสดงการทำงานของระบบ Quantum Cryptography System

2.2 ความรู้เบื้องต้นเกี่ยวกับรหัสลับ (*เรียบเรียงจากบทความเรื่อง ความรู้เบื้องต้นของการเข้ารหัสข้อมูล โดย ดร. บรรจง หะรังษี)

ในระบบวิทยาการรหัสลับเชิงควอนตัม เป็นระบบวิทยาการที่เกี่ยวข้องกับการใช้รหัสลับเพื่อรักษาความปลอดภัยของข้อมูล ในการสร้างซอฟท์แวร์จำลองการทำงานของรหัสลับเชิงควอนตัม จึงต้องใช้ความรู้และเข้าใจในเรื่องของการเข้ารหัสเป็นความรู้เบื้องต้นในการพัฒนาซอฟท์แวร์ นอกจากนี้ยังทำให้ผู้ที่สนใจในระบบสามารถเข้าใจระบบได้ง่ายยิ่งขึ้นอีกด้วย

2.2.1 วัตถุประสงค์ที่สำคัญของการเข้ารหัสข้อมูล

2.2.1.1 การทำให้ข้อมูลเป็นความลับ เพื่อป้องกันไม่ให้ผู้ที่ไม่มีสิทธิ์ในการเข้าถึงข้อมูลสามารถเข้าถึงข้อมูลได้

2.2.1.2 ป้องกันการแก้ไขหรือเปลี่ยนข้อมูลกลางทาง ซึ่งจะทำให้สามารถเชื่อมั่นในความถูกต้องของข้อมูลได้

2.2.1.3 ป้องกันการแอบอ้างการส่งข้อมูลโดยบุคคลอื่น

2.2.2 การเข้ารหัสข้อมูล

การเข้ารหัสข้อมูลเป็นวิธีการที่จะทำให้ข้อมูลเปลี่ยนเป็นข้อมูลที่ไม่สามารถอ่านได้โดยปราศจากกุญแจไขความลับ โดยใช้พื้นฐานทางด้านคณิตศาสตร์มาใช้ในการแปลงข้อมูล ซึ่งการแปลงข้อมูลนี้เรียกว่า Encryption และในแปลงข้อมูลกลับคืนมาเป็นข้อมูลที่สามารถอ่านได้นั้นเรียกว่า Decryption ซึ่งต้องใช้กุญแจไขความลับ หรือข้อมูลอีกชุดหนึ่งช่วยในการถอดรหัสข้อมูลออกมาเป็นข้อมูลที่สามารถอ่านได้

2.2.3 ประเภทของรหัสลับ

อัลกอริธึมในการเข้ารหัสข้อมูลมี สอง ประเภทหลัก ๆ ดังนี้

2.2.3.1 อัลกอริธึมแบบสมมาตร (Symmetric key algorithms) เป็นรูปแบบวิธีการเข้ารหัสลับโดยใช้ "กุญแจไขความลับ" ตัวเดียวกันในการเข้ารหัส และถอดรหัสข้อมูล ซึ่งในระบบวิทยาการรหัสลับเชิงควอนตัม รวมทั้งในซอฟท์แวร์จำลองการทำงานของรหัสลับเชิงควอนตัม จะใช้วิธีการเข้ารหัสลับแบบนี้ เนื่องจากเป็นระบบที่จะให้ความสำคัญกับความปลอดภัยของกุญแจไขความลับนั่นเอง

2.2.3.2 อัลกอริทึมแบบอสมมาตร (Asymmetric key algorithms) เป็นรูปแบบวิธีการเข้ารหัสลับโดยใช้ "กุญแจไขความลับ" เช่นกัน หากเพียงแต่ในการถอดรหัสข้อมูลจะไม่ได้ใช้ "กุญแจไขความลับ" ตัวเก่าที่ใช้ในการเข้ารหัส โดยทั้งสองฝ่าย ผู้ส่งและผู้รับต้องทำการตกลงกันว่าผู้ส่งจะทำการเข้ารหัสด้วยกุญแจใด และภาครับจะใช้กุญแจใดในการถอดรหัสข้อมูล โดยทั้งสองฝ่ายต้องรักษาความลับในเรื่องของกุญแจไขความลับของตัวเองไว้เพื่อป้องกันไม่ให้ผู้อื่นสามารถถอดรหัสข้อมูลที่เป็นความลับของเราได้

2.2.4 ปัจจัยที่มีผลต่อความยากในการถอดรหัสลับของผู้อื่น

ปัจจัยที่ส่งผลต่อความยากง่ายในการถอดรหัสข้อมูลมีดังนี้

2.2.4.1 กุญแจไขความลับนั้น ผู้ดูแลต้องเก็บไว้อย่างเป็นความลับไม่ให้มีการสูญหายหรือนำมาเปิดเผย

2.2.4.2 ความยาวของกุญแจไขความลับ ควรมีความยาวที่เหมาะสมเพื่อไม่ให้ผู้อื่นสามารถเดาสุ่ม กุญแจไขความลับ และทำการทดลองถอดรหัสได้โดยง่าย

2.2.4.3 การออกแบบและคิดค้นวิธีการหรืออัลกอริทึม ต้องไม่มีช่องโหว่หรือสามารถใช้กลวิธีในการถอดรหัสลับโดยปราศจากกุญแจไขความลับได้

2.2.5 ความยาวของกุญแจที่ใช้ในการเข้ารหัส

ความยาวของกุญแจเข้ารหัสมีหน่วยนับเป็นบิด หนึ่งบิตในคอมพิวเตอร์เป็นตัวเลขฐานสองที่ประกอบด้วยค่า "0" และ "1" กุญแจที่มีความยาว 1 บิต ตัวเลขที่เป็นไปได้เพื่อแทนกุญแจนั้น จึงอาจมีค่าเป็น "0" หรือ "1" กุญแจที่มีความยาว 2 บิต ตัวเลขที่เป็นไปได้จึงเป็น 0, 1, 2 และ 3 ตามลำดับ กุญแจที่มีความยาว 3 บิต ตัวเลขที่เป็นไปได้จะอยู่ระหว่าง 0 ถึง 7 ดังนั้นเมื่อเพิ่มความยาวของกุญแจทุกๆ 1 บิต ค่าที่เป็นไปได้ของกุญแจจะเพิ่มขึ้นเป็นสองเท่าตัว หรือจำนวนกุญแจที่เป็นไปได้จะเพิ่มขึ้นเป็น 2 เท่าตัวนั่นเอง

ฉะนั้นจะเห็นได้ว่ากุญแจยิ่งมีความยาวมาก โอกาสที่ผู้บุกรุกจะสามารถคาดเดากุญแจที่ตรงกับหมายเลขที่ถูกต้องของกุญแจจะยิ่งยากมากขึ้นตามลำดับ ในการที่ผู้บุกรุกลองผิดลองถูกกับกุญแจโดยใช้กุญแจที่มีหมายเลขต่างๆ กัน เพื่อหวังที่จะพบกุญแจที่ถูกต้องและสามารถใช้ถอดรหัสข้อมูลได้ การลองผิดลองถูกนี้เราเรียกกันว่า Key search หรือการค้นหากุญแจนั่นเอง ทฤษฎีได้กล่าวไว้ว่าการลองผิดลองถูกนี้โดยเฉลี่ยจะต้องทดลองกับกุญแจเป็นจำนวนครึ่งหนึ่งของกุญแจทั้งหมดก่อนที่จะพบกุญแจที่ถูกต้อง

ความยาวของกุญแจที่มีขนาดเหมาะสมจึงขึ้นอยู่กับความเร็วในการค้นหากุญแจของผู้บุกรุกและระยะเวลาที่ต้องการให้ข้อมูลมีความปลอดภัย ตัวอย่างเช่น ถ้าผู้บุกรุกสามารถลองผิดลองถูกกับกุญแจเป็นจำนวน 10 กุญแจภายในหนึ่งวินาทีแล้ว กุญแจที่มีความยาว 40 บิต จะสามารถป้องกันข้อมูลไว้ได้ 3,484 ปี ถ้าผู้บุกรุกสามารถลองได้เป็นจำนวน 1 ล้านกุญแจในหนึ่งวินาที (เทคโนโลยีปัจจุบันสามารถทำได้) กุญแจที่มีความยาว 40 บิตจะสามารถป้องกันข้อมูลไว้ได้เพียง 13 วันเท่านั้น (ซึ่งอาจไม่เพียงพอสำหรับในบางลักษณะงาน) ด้วยเทคโนโลยีในปัจจุบันหากผู้บุกรุกสามารถทดลองได้เป็นจำนวน 1,000

ล้านกุญแจในหนึ่งวินาที กุญแจขนาด 128 บิตจะสามารถป้องกันข้อมูลไว้ได้ 1022 ปี ดังนั้นด้วยลักษณะงานทั่วไปกุญแจขนาด 128 บิตจะพอเพียงต่อการรักษาความลับของข้อมูลเอาไว้ได้

2.3 การเข้ารหัสข้อมูลแบบ One-time pads (Vernam cipher)

ในระบบวิทยาการรหัสลับเชิงควอนตัมได้มีการนำการเข้ารหัสข้อมูลแบบ One-time pads มาประยุกต์ใช้ในระบบซึ่งวิธีการเข้ารหัสลับแบบนี้จะเป็นการใช้กุญแจไขความลับ 1 กุญแจ ต่อข้อมูล 1 ชุด ไม่มีการใช้กุญแจไขความลับซ้ำกันในการเข้ารหัสข้อมูลแต่ละครั้ง ซึ่งส่งผลให้รหัสลับมีความปลอดภัยสูงขึ้นอยากมาก ในซอฟท์แวร์จำลองการทำงานของรหัสลับเชิงควอนตัม ได้มีการนำความรู้ในส่วนของหลักการในการเข้ารหัสแบบ One-time pads มาประยุกต์ในการพัฒนาซอฟท์แวร์จำลองด้วยเช่นกัน

One-time pads เป็นวิธีการเข้ารหัสแบบง่าย ๆ แต่มีความปลอดภัยสูง ถูกคิดค้นขึ้นในปี 1917 โดยนักวิทยาศาสตร์ชาวอเมริกา Gilbert Vernam เป็นการเข้ารหัสแบบสมมาตรโดยมีหลักการพื้นฐานที่ว่า ในการสร้างกุญแจไขความลับที่ใช้ในการเข้ารหัสข้อมูล จะต้องมีขนาดของกุญแจไขความลับเท่ากับขนาดของข้อมูล มีการเข้ารหัสแบบ โดยจะสร้างกุญแจไขความลับแบบสุ่ม (ถ้าสุ่มได้ใกล้เคียงกับสุ่มที่แท้จริงแล้วยิ่งช่วยให้เพิ่มความปลอดภัยมากขึ้น) และในการใช้กุญแจไขความลับในการเข้ารหัสข้อมูลนั้นจะใช้เพียง 1 ครั้งในการเข้ารหัสข้อมูล 1 ชุด และจะต้องทำการสุ่มในการเข้ารหัสข้อมูลชุดต่อไป

การเข้ารหัสข้อมูลรูปแบบนี้มีความปลอดภัยสูงจริง แต่มีข้อจำกัดในเรื่องของการสุ่มกุญแจไขความลับ (key) เพราะยังต้องหาวิธีการสุ่มที่ใกล้เคียงกับการสุ่มที่แท้จริงมากที่สุด นอกจากนี้ยังมีปัญหาในเรื่องของขนาดของกุญแจไขความลับ เพราะหากมีขนาดใหญ่มากจะทำให้ข้อมูลรหัสลับมีขนาดใหญ่ และเป็นปัญหาในการจัดส่งข้อมูลอีกด้วย แต่ในปัจจุบันเราสามารถส่งข้อมูลได้ในความเร็วที่สูง รวมทั้งความเร็วของช่องทางการส่งข้อมูลมีความเร็วสูงขึ้นจึงลดปัญหาข้อนี้ลงได้ และในระบบวิทยาการรหัสลับเชิงควอนตัมได้มีการใช้ช่องสัญญาณควอนตัม ซึ่งเป็นช่องสัญญาณที่ใช้แสงเป็นตัวนำผ่านสาย Fiber Optic ซึ่งมีความเร็วที่สูงมากปัญหาดังกล่าวจึงหายไป รวมทั้งในระบบยังมีการพัฒนาอุปกรณ์ทาง Hardware มาช่วยในการสร้าง กุญแจไขความลับ ที่เกิดจากการสุ่มที่ใกล้เคียงกับการสุ่มที่แท้จริงอีกด้วย

2.3.1 หลักการในการเข้ารหัสข้อมูลแบบ One-time pads

หลักการในการเข้ารหัสข้อมูลแบบ One-time pads มีดังนี้

$C_i = E(P_i, K_i)$

โดยที่ C_i คือผลลัพธ์ของการเข้ารหัส ตัวที่ i

E คือฟังก์ชันการเข้ารหัส

P_i คือข้อมูลตัวที่ I

K_i คือคีย์ที่ใช้เข้ารหัส ตัวที่ i

จากลักษณะของการเข้ารหัสแบบ One-time pads ฟังก์ชั่นที่นิยมใช้ในการเข้ารหัสและถอดรหัสข้อมูลคือการ XOR ระหว่างข้อมูลต้นฉบับกับคีย์แบบ บิตต่อบิต วิธีการนี้ทั้งการเข้าและถอดรหัสจะใช้ฟังก์ชันเดียวกัน

ข้อมูลเข้า		Output
ข้อมูล	คีย์	
0	0	0
0	1	1
1	0	1
1	1	0

ตารางที่ 2.1 ตารางแสดงการ XOR ของเลขฐานสอง

2.3.2 ตัวอย่างการเข้ารหัสแบบ One-time pads โดยวิธี XOR

จากสมการ $C_i = E(P_i, K_i)$

โดยที่ C_i คือผลลัพธ์ของการเข้ารหัส ตัวที่ i

E คือฟังก์ชันการเข้ารหัส

P_i คือข้อมูลตัวที่ I

K_i คือคีย์ที่ใช้เข้ารหัส ตัวที่ i

กระบวนการเข้ารหัส

ข้อมูล	A	1000001
คีย์	#	0100011
ผลการ XOR	b	1100010

กระบวนการถอดรหัส

ข้อมูล	b	1100010
คีย์	#	0100011
ผลการ XOR	A	1000001

ถอดรหัสด้วยคีย์ที่สุ่มขึ้นมา

ข้อมูล	b	1100010
คีย์	$	0100100
ผลการ XOR	F	1000110

ตารางที่ 2.2 แสดงข้อมูลในกระบวนการเข้ารหัส ถอดรหัสด้วย Vernam

2.3.3 ความปลอดภัยของการเข้ารหัสแบบ One-time pads

การเข้ารหัสแบบ One-time pads เป็นการเข้ารหัสที่ไม่สามารถถูกถอดรหัสได้ ถ้าไม่มีกุญแจไขความลับที่แท้จริง (แต่การสุ่มสร้างกุญแจไขความลับนั้นต้องใกล้เคียงกับการสุ่มที่แท้จริงมากที่สุด) ถึงแม้ว่าปัจจุบันนี้จะมีการใช้เทคนิคการถอดรหัสแบบ brute force (เป็นการคำนวณหากุญแจไขความลับที่เป็นไปได้ทุกกรณี) ซึ่งอาจจะถอดรหัสออกมาได้ แต่ก็ยังไม่รู้ว่าข้อมูลชุดใดคือข้อความต้นฉบับที่ถูกต้อง กล่าวคือ แม้จะถอดรหัสข้อมูลได้ แต่ก็ไม่มีมีทางรู้ได้ว่าข้อมูลที่ได้นั้นสมบูรณ์ หรือเป็นข้อมูลที่ถูกต้องได้หรอไม่

2.3.4 การจัดการกับคีย์ (Key Management)

การเข้ารหัสแบบ One-time pads เป็นที่รู้กันแล้วว่าจะต้องใช้กุญแจไขความลับที่มีขนาดเท่ากับข้อมูล ซึ่งหากข้อมูลมีขนาดใหญ่จะทำให้กุญแจนั้นมีขนาดใหญ่ไปด้วย จึงจำเป็นต้องมีวิธีการที่จะจัดการกุญแจเหล่านี้ ทั้งในเรื่องของการจัดส่งคีย์ไปยังฝั่งผู้รับเพื่อใช้ในการถอดรหัส ซึ่งโดยทั่วไปจะส่งไปในช่องทางที่มีความปลอดภัยสูง นอกจากนี้จะต้องดูแลในเรื่องของการจัดเก็บละทำลายกุญแจ ให้ดีอีกด้วยเพื่อป้องกันการล่วงละเมิดกุญแจเหล่านั้น จึงจะได้รหัสลับที่มีความปลอดภัยสูงสุด

จะเห็นได้ว่าระบบวิทยาการรหัสลับเชิงควอนตัม สามารถนำวิธีการเข้ารหัสแบบนี้ไปใช้ได้อย่างสมบูรณ์แบบ เนื่องจากมีวิธีการและรูปแบบการจัดการต่าง ๆ ที่สามารถทำให้ข้อมูลมีความปลอดภัยสูงสุด

2.4 การเข้ารหัสข้อมูลแบบ Data Encryption Standard (DES) [1]

DES (Data Encryption Standard) ในปัจจุบันเป็นชื่อย่อของมาตรฐานการเข้ารหัสลับข้อมูลที่อธิบายถึงกรรมวิธีหรืออัลกอริทึมสำหรับการเข้ารหัสลับข้อมูลประเภทกุญแจลับ (Secret key encryption) มาตรฐาน DES ได้รับการร่างขึ้นครั้งแรกเมื่อปี ค.ศ. 1977 โดย NIST (National Institute of Standards and Technology) ซึ่งเป็นหน่วยงานราชการภาพใต้การดูแลของกระทรวงพาณิชย์ ประเทศสหรัฐอเมริกา โดยมีวัตถุประสงค์เพื่อใช้ในการปกป้องข้อมูลทางราชการจากการดักฟังจากผู้ที่ไม่ได้รับอนุญาตหรือป้องกันข้อมูลเหล่านี้มิให้มีการเปลี่ยนแปลงในระหว่างที่ส่งผ่านช่องสัญญาณหรือที่บรรจุอยู่ในฐานข้อมูล จากนั้นได้มีการเปลี่ยนแปลงและพัฒนาเพิ่มเติมเพื่อเพิ่มสมรรถนะในการรักษาความความปลอดภัยของข้อมูล เพิ่มความแข็งแกร่งและปลอดภัยให้ยิ่งขึ้น

2.4.1 มาตรฐานการเข้ารหัสลับข้อมูล DES

ในขั้นตอนการเข้ารหัสลับตามมาตรฐาน DES นั้นจะพิจารณาข้อความต้นฉบับ (Plaintext) ทีละ 64 บิต เพื่อนำไปป้อนเข้าสู่กระบวนการเข้ารหัสลับ (Enciphered) ที่มีการทำงานทั้งหมด 16 รอบ ตามที่แสดงไว้ในรูปที่ 2.2 และผลลัพธ์ที่ได้จากการเข้าหัสคือข้อความไซเฟอร์ (Ciphertext) ที่มีขนาดเท่าเดิมคือ 64 บิต ส่วนขั้นตอนการถอดรหัสลับ (Deciphering) ก็มีกรรมวิธีที่คล้ายคลึงกับการเข้ารหัสลับและการใช้กุญแจลับ (Secret Key) ขนาด 64 บิตชุดเดียวกัน หากแต่รายละเอียดการนำกุญแจลับมาใช้งานจะแตกต่างไป กล่าวคือ มีการใช้งานด้วยลำดับที่กลับกัน แผนภาพโครงสร้างโดยรวมของการเข้ารหัสลับสามารถแบ่งออกได้เป็นสองส่วนหลัก คือ การจัดเตรียมกุญแจ (Key Schedules) และการเข้ารหัสลับ (Encipherment) โดยที่กรรมวิธีการทำงานล้วนแล้วแต่อาศัยกล่องอยู่สองลักษณะ คือ กล่องสลับลำดับ (Permutation) หรือเรียกโดยย่อว่า P-box และกล่องแทนค่า (Substitutions) หรือ S-boxes

[1] (เรียบเรียงจาก วิทยาการรหัสลับเบื้องต้น สำนักพิมพ์แห่งจุฬาลงกรณ์มหาวิทยาลัย 2548)

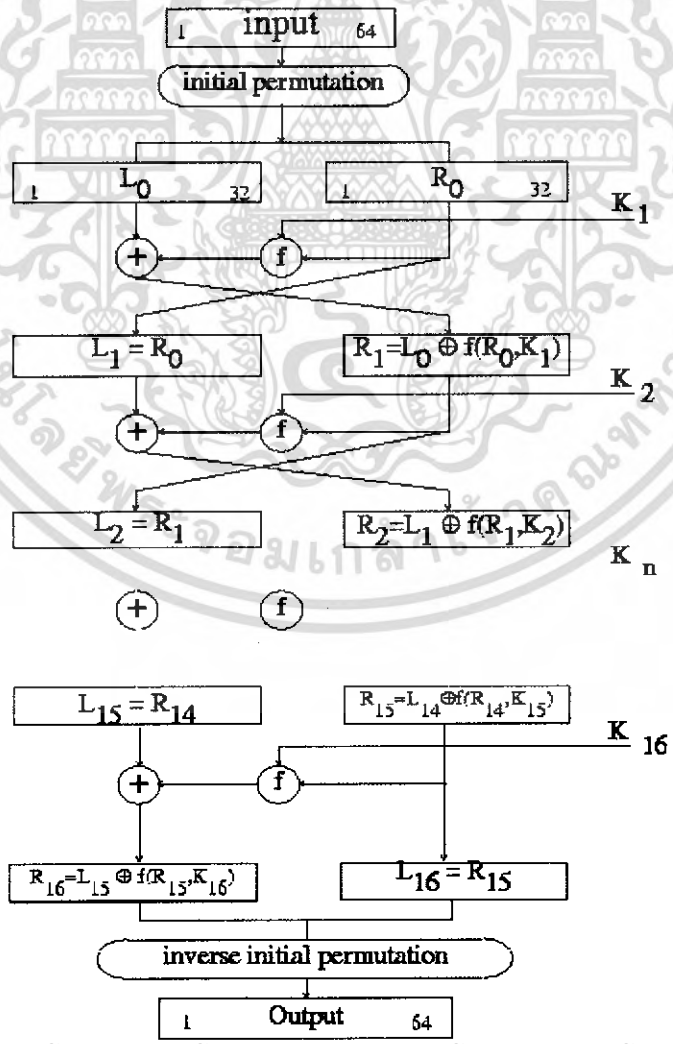
ในการอธิบายถึงรายละเอียดของแต่ละขั้นตอนจะอาศัยการนิยามตัวแปรดังนี้ ข้อความต้นฉบับ Plaintext $X = (x_1, x_2,..., x_{64})$ ข้อมูลที่ได้จากการเข้ารหัสลับจะแสดงในรูปแบบของข้อความไซเฟอร์ Ciphertext $Y = (y_1, y_2,..., y_{64})$ ส่วนกุญแจลับที่ใช้ Key $K = (k_1, k_2,..., k_{64})$ ทั้งนี้จำนวนบิตของกุญแจลับที่ใช้งานจริงมีเพียง 56 บิตเท่านั้น อีก 8 บิตที่เหลือทำหน้าที่เป็นบิตตรวจสอบพาริตี (Parity Check) เท่านั้น ซึ่งหมายความว่าปริภูมิของกุญแจ (Key Space) มีทั้งหมด 2^{56} รูปแบบ ในการอธิบายขั้นตอนการเข้ารหัสลับจะอาศัยตัวอย่างประกอบ โดยที่กำหนดให้ข้อความต้นฉบับเป็น

X = (F 9 5 E C 5 A 6 B D 1 F E 5 2 C)

และมีกุญแจลับเป็น

K = (B 9 A 8 4 F 0 8 F A D D 0 B B 7)

ซึ่งทั้งสองค่าได้เขียนแสดงในรูปของตัวเลขฐาน 16 เพื่อความกระชับ



รูปที่ 2.2 แสดงแผนภาพการเข้ารหัสลับตามมาตรฐาน DES

2.4.2 การจัดเตรียมกุญแจ

ในลำดับแรกจะขออธิบายถึงขั้นตอนการจัดกุญแจ ขั้นแรกบิตของกุญแจลับ K ในการเข้ารหัสลับจะถูกนำมาเข้ากระบวนการเริ่มต้นที่เรียกว่า Initial Permutation (PC-1) ซึ่งจะเลือกเฉพาะบิต 56 บิตที่เกี่ยวข้องมาใช้งานจากข้อมูลของกุญแจซึ่งมีทั้งหมด 64 บิต โดยบิตไพริตีที่ดึงออกมาทั้ง 8 บิต ได้แก่ (k_8, k_{16}, k_{24}, k_{32}, k_{40}, k_{48}, k_{56}, k_{64}) หลังจากที่ผ่านการสลับลำดับตามที่กำหนดไว้ในกล่องสลับลำดับ PC-1 ซึ่งมีรายละเอียดตามในตารางที่ 2.3 แล้ว บิตของกุญแจลับ 56 บิตจะถูกนำไปแยกออกเป็น 2 ส่วนเท่ากัน คือส่วนละ 28 บิตเพื่อป้อนเข้าสู่วงจรชิฟต์เรจิสเตอร์ (Shift Register) ที่มีชื่อเรียกว่า C_0 และ D_0 โดยจะทำการเลื่อนตำแหน่งบิตไปทางซ้ายแบบวนกลับ ซึ่งอาจเป็น 1 หรือ 2 ตำแหน่งก็ได้ ขึ้นอยู่กับรอบของการเข้ารหัสตามที่กำหนดในตาราง 2.4 ผลที่ได้บรรจุอยู่ในตัวแปรที่เรียกว่า C_1 และ D_1 จากนั้นนำค่าของตัวแปรทั้งสองที่มีทั้งสิ้น 56 บิต ไปผ่านกล่องสลับลำดับ PC-2 เพื่อสร้างเป็นตัวแปร K_1 ที่มีขนาด 48 บิต สำหรับไปใช้ในการเข้ารหัสลับรอบที่ 1 ต่อไป สำหรับรายละเอียดของกล่องสลับลำดับ PC-2 ได้แสดงไว้ในตารางที่ 2.5 กระบวนการทั้งหมดที่ได้กล่าวมานี้เป็นขั้นตอนการจัดกุญแจสำหรับการใช้ในการเข้ารหัสรอบที่ 1

สำหรับการเข้ารหัสที่รอบอื่นที่เหลือตั้งแต่รอบที่ 2-16 ก็จะต้องมีการเตรียมกุญแจสำหรับใช้งานในแต่ละรอบด้วยเช่นกัน และกุญแจที่จัดเตรียมจะมีชื่อเรียกว่า K_2-K_{16} ตามลำดับ การเตรียมกุญแจที่เหลือก็มีขั้นตอนเหมือนกับการเตรียมกุญแจ K_1 ยกตัวอย่างเช่น การเตรียมค่า K_2 ให้ทำโดยการนำค่าทั้ง 56 บิตในตัวแปร C_1 และ D_1 ไปผ่านวงจรชิพต์เรจิสเตอร์ LS ซึ่งจะได้ผลเป็นตัวแปร C_2 และ D_2 ที่เมื่อป้อนเข้าสู่กล่องสลับลำดับ PC-2 แล้วจะให้ผลเป็นกุญแจ K_2

C	57	49	41	33	25	17	9	1	58	50	42	34	26	18
	10	2	59	51	43	35	27	19	11	3	60	52	44	36
D	63	55	47	39	31	23	15	7	62	54	46	38	30	32
	14	6	61	53	45	37	29	21	13	5	28	20	12	4

ตารางที่ 2.3 กล่องสลับลำดับ Permuted Choice 1 (PC-1)

รอบที่	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
จำนวนบิต	1	1	2	2	2	2	2	2	1	2	2	2	2	2	2	1

ตารางที่ 2.4 จำนวนการเลื่อนบิตไปทางซ้ายมือแบบวนกลับสำหรับการเข้ารหัสแต่ละรอบ

C	14	17	11	24	1	5	3	28	15	6	21	10
	23	19	12	4	26	8	16	7	27	20	13	2
D	41	52	31	37	47	55	30	40	51	45	33	48
	44	49	39	56	34	53	46	42	50	36	29	32

ตารางที่ 2.5 กล่องสลับลำดับ Permutated Choice 2 (PC-2)

เพื่อให้เห็นภาพได้ชัดเจนยิ่งขึ้นจะขอแสดงรายละเอียดการจัดเตรียมกุญแจให้กับตัวอย่างที่เกริ่นไว้ก่อนหน้าคือ

K = (B 9 A 8 4 F A D D 0 B B 7)

หรือหากแสดงในรูปของตัวเลขฐานสองจะได้เป็น

K = (1011 1001 1010 1000 0100 1111 0000 1000
1111 1010 1101 1101 0000 1011 1011 0111)

ขั้นตอนแรกคือการหาค่าของ C_0 และ D_0 โดยได้จากการเลือกบิตจากกุญแจลับ K และตำแหน่งที่เลือกขึ้นอยู่กับกล่องสลับลำดับ PC-1 ในตารางที่ 2.3 ผลที่ได้เป็นดังนี้

C_0 = (1 0 1 1 0 0 1 1 0 0 1 1 1
0 0 1 0 0 1 0 0 1 1 1 1 1)

D_0 = (1 1 0 1 0 1 0 0 1 0 1 0 1
0 0 0 1 1 1 1 1 1 1 0 0 1)

จากนั้นเราจะใช้ตารางที่ 2.4 เพื่อพิจารณาหา C_1 และ D_1 โดยในรอบแรกจะทำการเลื่อนตำแหน่งบิตไปทางซ้ายจำนวน 1 ตำแหน่งดังที่ระบุในตาราง

C_1 = (0110 0110 0110 1001 0010 0111 0111)

D_1 = (1010 1001 0100 1000 1111 1110 0011)

ในการหาค่า C_2 และ D_2 ก็คำนวณได้ไม่ยากโดยการนำค่าใน C_1 และ D_1 ไปผ่านวงจร LS ซึ่งจะเลื่อนบิตไปทางซ้ายแบบวนกลับไปเป็นจำนวน 1 ตำแหน่ง ตามที่ระบุไว้ในตารางที่ 2.4 จะได้

C_2 = (1100 1100 1101 0010 0100 1110 1110)

D_2 = (0101 0010 1001 0001 1111 1100 0111)

สำหรับค่า $(C_2, D_2), (C_3, D_3), \ldots, (C_{16}, D_{16})$ นั้นก็มีลักษณะการคำนวณเช่นเดียวกัน

หลังจากที่ได้คำนวณค่าของ C และ D ครบทั้ง 16 ชุดแล้วให้นำไปใช้ในการหาค่า K_1 ถึง K_{16} ได้ทันที โดยการป้อนค่า C และ D แต่ละชุดเข้าไปในวงจรสลับลำดับ Permutated Choice 2

(PC-2) ตามที่แสดงไว้ในตารางที่ 2.5 และผลที่ได้คือชุดกุญแจ K_1 ถึง K_{16} ที่ใช้ประกอบในการเข้ารหัสแต่ละรอบ ยกตัวอย่างเช่น K_1 หาได้จาก C_1 ถึง D_1 ซึ่งเมื่อพิจารณาจากตารางที่ 2.5 ประกอบจะได้

K_1 = (001100 110101 110010 111011
101011 001111 010100 101110)

เช่นเดียวกัน K_2 หาได้จาก C_2 และ D_2 โดยอาศัย PC-2 จากตารางที่ 2.5 ประกอบ

K_2 = (000011 001111 101010 001001
000111 110101 110100 101001)

เมื่อเราได้ค่า K_1 ถึง K_{16} ครบทุกค่าแล้วจะได้นำค่าเหล่านี้ไปใช้ประกอบในการเข้ารหัสลับให้แก่ข้อมูลต้นฉบับในแต่ละรอบ ซึ่งผลที่ได้หลังจากทำครบ 16 รอบแล้วจะได้เป็นข้อความไซเฟอร์ขนาด 64 บิตตามต้องการ

2.4.3 การเข้ารหัสแต่ละรอบ

ในหัวข้อนี้จะเป็นกระบวนการในส่วนของการนำข้อความต้นฉบับ (Plaintext) ขนาด 64 บิต ไปผ่านการเข้ารหัส โดยอาศัยชุดกุญแจ K_1 ถึง K_{16} ที่ได้เตรียมไว้แล้ว สำหรับกรอธิบายในที่นี้จะอาศัยข้อมูลต้นฉบับเท่ากับ

X = (F 9 5 E C 5 A 6 B D 1 F E 5 2 C)

หรือถ้าเขียนในรูปของตัวเลขฐานสองจะได้เป็น

$X = (x_1, x_2, ..., x_{64})$
= (1111 1001 0101 1110 1100 0101 1010 0110
1011 1101 0001 1111 1110 0101 0010 1100)

พิจารณาจากแผนภาพที่ 2.2 แสดงให้เห็นว่าข้อความต้นฉบับ Plaintext X จะได้รับการป้อนเข้าสู่กล่องสลับลำดับ Initial Permutation (IP) เพื่อสลับลำดับของบิต พร้อมกันนั้นเองก็ยังได้มีการแยกบิตออกเป็น 2 บล็อก คือ L_0 และ R_0 โดยแต่ละบล็อกประกอบไปด้วย 32 บิต ตามที่กำหนดไว้ในตารางที่ 2.6 สำหรับในกรณีตัวอย่างข้อความต้นฉบับ X ในที่นี้ จะได้ว่า

L_0 = (0100 0111 0011 0011 1111 1110 0111 0101)
4 7 3 3 F E 7 5

R_0 = (0101 1101 1101 1001 1011 0011 0010 1010)
5 D D 9 B 3 2 A

L_0	=	(50	50	42	34	26	18	10	2
		60	52	44	36	28	20	12	4
		62	54	46	38	30	22	14	6
		64	56	48	40	32	24	16	8)
R_0	=	(57	49	41	33	25	17	9	1
		59	51	43	35	27	19	11	3
		61	53	45	37	29	21	13	5
		63	55	47	39	31	23	15	7)

ตารางที่ 2.6 กล่องสลับลำดับ Initial Permutation (IP)

32	1	2	3	4	5
4	5	6	7	8	9
8	9	10	11	12	13
12	13	14	15	16	17
16	17	18	19	20	21
20	21	22	23	24	25
24	25	26	27	28	29
28	29	30	31	32	1

ตารางที่ 2.7 E Bit-Selection Table

เมื่อพิจารณาดูการเข้ารหัสในรอบแรกจะเห็นว่า บล็อก L_0 จะไม่มีการประมวลผลแต่อย่างใด ในขณะที่ R_0 ถูกนำไปผ่านขั้นตอนการประมวลผลหลายขั้นตอน และมีการนำชุดกุญแจ K_1 มาประกอบการเข้ารหัสด้วย ลำดับแรกบิตในบล็อก R_0 จำนวน 32 บิตจะถูกขยายขึ้นมาเป็น 48 บิต โดยฟังก์ชัน $E(R_0)$ ที่มีลักณะตามที่แสดงไว้ในตารางที่ 2.7 สำหรับในกรณีตัวอย่างที่ใช้ประกอบอธิบายจะได้ว่า

$E(R_0)$ = (001011 111011 111011 110011
110110 100110 100101 010100)

จากนั้นจึงนำ $E(R_0)$ ที่ได้ไปบวกแบบมอดุโล 2 กับชุดกุญแจ K_1 ที่มีขนาด 48 บิตเท่ากัน ผลลัพธ์ที่ได้บรรจุลงในตัวแปร Γ_1 ที่มีชื่อเรียกว่า The key-dependent function กล่าวคือ

$$\Gamma_1 = E(R_0) \oplus K_1$$

ในตัวอย่างที่ใช้ Γ_1 มีค่าเท่ากับ

$$\Gamma_1 = E(R_0) \oplus K_1$$

$$= (000111\ 001110\ 001001\ 001000$$

$$011101\ 101001\ 110001\ 111010)$$

สำหรับการคำนวณค่า Γ_i ของการเข้ารหัสในรอบอื่น ๆ สามารถหาได้ดังนี้

$$\Gamma_j = E(R_i) \oplus K_j, \qquad 0 \le i \le 15, \qquad 1 \le j \le 16$$

S_1

14	4	12	1	2	15	11	8	3	10	6	12	5	9	0	7
0	15	7	4	14	2	12	1	10	6	12	11	9	5	3	8
4	1	14	8	13	6	2	11	15	12	9	7	3	10	5	0
15	12	8	2	4	9	1	7	5	11	3	14	10	0	6	13

S_2

15	1	8	14	6	11	3	4	9	7	2	13	12	0	5	10
3	13	4	7	15	2	8	14	12	0	1	10	6	9	11	5
0	14	7	11	10	4	13	1	5	8	12	6	9	3	2	15
13	8	10	1	3	15	4	2	11	6	7	12	0	5	14	9

S_3

10	0	9	14	6	3	15	5	1	13	12	7	11	4	2	8
13	7	0	9	3	4	6	10	2	8	5	14	12	11	15	1
13	6	4	9	8	15	3	0	11	1	2	12	5	10	14	7
1	10	13	0	6	9	8	7	4	15	14	3	11	5	2	12

S_4

7	13	14	3	0	6	9	10	1	2	8	5	11	12	4	15
13	8	11	5	6	15	0	3	4	7	2	12	1	10	14	9
10	6	9	0	12	11	7	13	15	1	3	14	5	2	8	4

3	15	0	6	10	1	13	8	9	4	5	11	12	7	2	14

S_5

2	12	4	1	7	10	11	6	8	5	3	15	13	0	14	9
14	11	2	12	4	7	13	1	5	0	15	10	3	9	8	6
4	2	1	11	10	13	7	8	15	9	12	5	6	3	0	14
11	8	12	7	1	14	2	13	6	15	0	9	10	4	5	3

S_6

12	1	10	15	9	2	6	8	0	13	3	4	14	7	5	11
10	15	4	2	7	12	9	5	6	1	13	14	0	11	3	8
9	14	15	5	2	8	12	3	7	0	4	10	1	13	11	6
4	3	2	12	9	5	15	10	11	14	1	7	6	0	8	13

S_7

4	11	2	14	15	0	8	13	3	12	9	7	5	10	6	1
13	0	11	7	4	9	1	10	14	3	5	12	2	15	8	6
1	4	11	13	12	3	7	14	10	15	6	8	0	5	9	2
6	11	13	8	1	4	10	7	9	5	0	15	14	2	3	12

S_8

13	2	8	4	6	15	11	1	10	9	3	14	5	0	12	7
1	15	13	8	10	3	7	4	12	5	6	11	0	14	9	2
7	11	4	1	9	12	14	2	0	6	10	13	15	3	5	8
2	1	14	7	4	10	8	13	15	12	9	0	3	5	6	11

ตารางที่ 2.8 แสดง S_1-S_8 Primitive S-Box Function

หลังจากได้ Γ_1 ขนาด 48 บิตเรียบร้อยแล้ว ให้แบ่งออกเป็น 8 กลุ่ม แต่ละกลุ่มมีขนาด 6 บิต เพื่อนำชุดบิตทั้ง 8 กลุ่มไปป้อนให้กับกล่องแทนค่า S-Box ที่มีทั้งหมด 8 ชุด กล่องแทนค่าแต่ละชุดจะให้ผลลัพธ์เป็นบิตข้อมูลที่มีขนาดลดลงเหลือ 4 บิต ฉะนั้นโดยรวมแล้วหลังผ่านขั้นตอนนี้จะได้จำนวนบิตลดลงเหลือ 32 บิต ลักษณะการทำงานของ S-Box เหล่านี้คือ นำบิตแรกและบิตสุดท้ายมา

ใช้ระบุหมายเลขแถวของตาราง S-Box ที่แสดงในตาราง 2.8 และใช้ค่าของบิตที่ 2 ถึงบิตที่ 6 เพื่อระบุหมายเลขคอลัมน์ของตาราง S-Box สำหรับผลลัพธ์ที่ได้คือค่าที่บรรจุอยู่ในตาราง ณ ตำแหน่งที่ระบุ สังเกตว่าตัวเลขที่ได้มีขนาดอยู่ระหว่าง 0-15 ซึ่งเมื่อเขียนเป็นเลขฐานสองจะมีขนาด 4 บิตพอดี

พิจารณาตัวอย่างค่า Γ_1 จากตัวอย่างที่ทำมา เมื่อนำมาเข้ากระบวนการคำนวณสำหรับ S_1-S_8 จะได้ผลดังนี้

$S_1(01, 0011) = S_1(1, 3) = 4 = 0100$

$S_2(00, 0111) = S_2(0, 7) = 4 = 0100$

$S_3(01, 0100) = S_3(1, 4) = 3 = 0011$

$S_4(00, 0100) = S_4(0, 4) = 0 = 0000$

$S_5(01, 1110) = S_5(1, 14) = 8 = 1000$

$S_6(11,0100) = S_6(3, 4) = 9 = 1001$

$S_7(11, 1000) = S_7(3, 8) = 9 = 1001$

$S_8(10, 1101) = S_8(2, 13) = 3 = 0011$

ค่าแต่ละค่าขนาด 4 บิตข้างต้น เป็นบิตที่ออกจากกล่องแทนค่า S-Box แต่ละชุด เมื่อนำมารวมกันจะได้ผลลัพธ์ที่ด้านออกในรูปแบบของ B_1 เท่ากับ

$B_1 = (0100\ 0100\ 0011\ 0000\ 1000\ 1001\ 1001\ 0011)$

จากนั้นค่าของ B_1 ที่ได้นี้ นำไปผ่านกระบวนการ $P(B_1)$ ที่มีหน้าที่เป็นกล่องสลับตำแหน่ง (Permutation Function) โดยจะรับค่า B_1 ที่มีขนาด 32 บิตและให้ผลลัพธ์เป็นชุดบิตขนาดเท่าเดิมแต่มีการสลับตำแหน่งไปจากเดิม โดยรูปแบบการสลับตำแหน่งมีลักษณะตามตารางที่ 2.9 เมื่อนำค่าตัวอย่าง B_1 ไปป้อนเข้ากล่องสลับตำแหน่งจะได้ผลเป็น

$P(B_1) = (0001\ 0111\ 0000\ 0010\ 1010\ 1000\ 0001\ 0101)$

ค่าของ $P(B_1)$ ที่ได้จะนำไปใช้ในการหาค่า R_1 โดยอาศัยความสัมพันธ์ดังต่อไปนี้

$R_1 = P(B_1) \oplus L_0$

สำหรับตัวอย่างที่ใช้ในการอธิบายจะให้ค่า R_1 เท่ากับ

$R_1 = (0101\ 0000\ 0011\ 0001\ 0101\ 0110\ 0110\ 0000)$

5 0 3 1 5 6 6 0

ในการหาค่า L_1 ให้นำค่า R_0 มาแทนค่า L_1 ได้เลย สำหรับตัวอย่างที่ทำมาจะได้เป็น

$L_1 = R_0 = (0101\ 1101\ 1101\ 1001\ 1011\ 0011\ 0010\ 1010)$

5 D D 9 B 3 2 A

16	7	20	21
29	12	28	17
1	15	23	26
5	18	31	10
2	8	24	14
32	27	3	9
19	13	30	6
22	11	4	25

ตารางที่ 2.9 Permutation Function P

รายละเอียดขั้นตอนการทำงานได้อธิบายมาทั้งหมดนี้ครอบคลุมการเข้ารหัสลับในรอบที่ 1 ครบสมบูรณ์ทุกประการ สำหรับการทำงานอีก 15 รอบที่เหลือมีรูปแบบการทำงานเหมือนเดิม เพียงแต่ชุดกุญแจที่ใช้ในแต่ละรอบแตกต่างกันไป ผลที่ได้จากทั้ง 16 ขั้นตอนคือ L_{16} และ R_{16} ซึ่งจะนำไปป้อนเข้าสู่กล่องสลับลำดับผกผัน IP^{-1} ตามที่แสดงในตาราง 2.10 เพื่อให้ได้เป็นข้อความไซเฟอร์ Y ตามต้องการ สำหรับในกรณีตัวอย่างที่ใช้ในการอธิบายจะได้ค่า Y เท่ากับ

Y = 1100 1100 1110 1100 0010 0010 0011 1011

0100 1010 1111 0101 1001 0110 1110 1101

Y = (C C E C 2 2 3 B 4 A F 5 9 6 E D)

R	40	8	48	16	56	24	64	32
	39	7	47	15	55	23	63	31
	38	6	46	14	54	22	62	30
	37	5	45	13	53	21	61	29
L	36	4	44	12	52	20	60	28
	35	3	43	11	51	19	59	27
	34	2	42	10	50	18	58	26
	33	1	41	9	49	17	57	25

ตารางที่ 2.10 กล่องสลับลำดับผกผัน (Inverse of initial Permutation, IP^{-1})

ส่วนการถอดรหัสลับทำเช่นเดียวกับการเข้ารหัสข้อมูลแต่ทำสลับขั้นตอนย้อนหลังก็จะได้ข้อมูลต้นฉบับที่ครบถ้วนสมบูรณ์

2.5 การเข้ารหัสข้อมูลแบบ Blowfish (Blowfish Algorithm)

Blowfish เป็นอัลกอริทึมในการเข้ารหัสแบบหนึ่ง เป็นอัลกอริทึมแบบใช้กุญแจลับ หรือเป็นการเข้ารหัสลับแบบสมมาตร คือใช้คีย์ตัวเดียวในการเข้ารหัสและถอดรหัสข้อมูล การเข้ารหัสจะเข้ารหัสทีละบล็อก โดยแต่ละบล็อกจะมีขนาด 64 บิต และขนาดของคีย์ที่ใช้จะมีขนาดสูงสุดที่ 448 บิต

ในการทำงานของ Blowfish จะแบ่งออกเป็นสองส่วนคือส่วนของการเตรียมคีย์ และส่วนของการเข้ารหัสลับซึ่งจะทำงานทั้งสิ้น 16 รอบ

2.5.1 ขั้นตอนการเตรียมคีย์

Blowfish Algorithm จะมีคีย์ย่อยขนาดใหญ่เพื่อใช้ในการเข้ารหัสลับ โดยคีย์ดังกล่าวจะต้องมีการเตรียมและคำนวณไว้ล่วงหน้าก่อนที่จะนำข้อมูลมาทำการเข้ารหัสลับ หรือถอดรหัสลับเสมอ

P-array เป็นอาเรย์ของค่าคงที่ 18 อาเรย์ โดยในแต่ละอาเรย์หรือแต่ละชุดข้อมูลนั้นจะมี 32 บิต เช่น P1, P2, P3, ..., P18 เป็นต้น

S-boxes เป็นชุดของข้อมูลจำนวน 4 ชุด ซึ่งในแต่ละชุดจะประกอบด้วย 32 บิต กับ 256 ข้อมูลที่เพิ่มเข้าไป เช่น

S1,0, S1,1,..., S1,255;

S2,0, S2,1,..., S2,255;

S3,0, S3,1,..., S3,255;

S4,0, S4,1,..., S4,255.

ในการสร้างคีย์ย่อยนี้จะทำการสร้างโดยใช้วิธีใน Blowfish Algorithm โดยขั้นตอนการสร้างคีย์ย่อยมีขั้นตอนดังต่อไปนี้

2.5.1.1 เริ่มต้นจากการเตรียมค่า P-array และ S-boxes ซึ่งค่าเหล่านี้เป็นค่าคงที่ที่ได้ทำการเฉพาะเจาะจงไว้แล้ว โดยจะเป็นค่าคงที่ของ hexadecimal ดังตัวอย่างเช่น

P1 = 0x243f6a88

P2 = 0x85a308d3

P3 = 0x13198a2e

P4 = 0x03707344 เป็นต้น

2.5.1.2 จากนั้นนำค่า P1 XOR กับ 32 บิตแรกของคีย์ และนำค่า P2 XOR กับ 32 บิตต่อมาของคีย์ จากนั้นทำเช่นนี้ไปเรื่อย ๆ ตลอดจนครบบิตคีย์ ทำการทำซ้ำจนหว่าจะได้ P-array ทั้ง 18 ชุด

2.5.1.3 ทำการเข้ารหัสข้อมูล 0 ด้วยวิธีการเฉพาะใน Blowfish Algorithm โดยวิธีการสร้างคีย์ย่อยที่กล่าวมาข้างต้นใน 2.5.1.1 และ 2.5.1.2

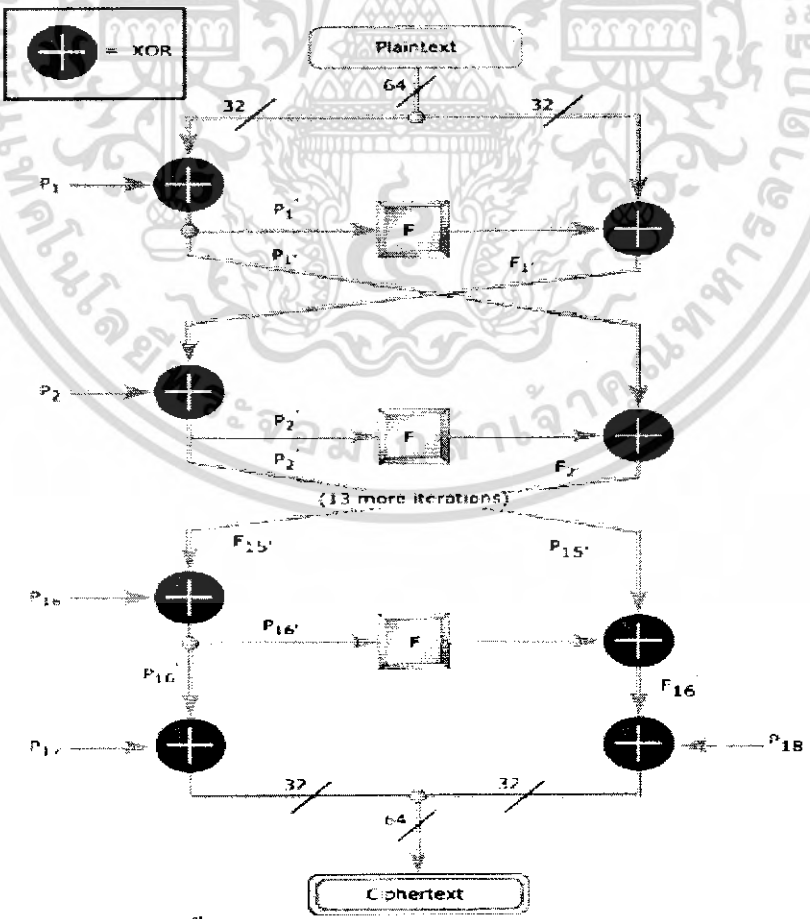
2.5.1.4 หลังจากทำครบแล้วให้ทำการแทนค่า P1 และ P2 ด้วยเอาท์พุตใน 2.5.1.3

2.5.1.5 ทำการเข้ารหัสข้อมูลเอาท์พุตในข้อ 2.5.1.3 อีกครั้งด้วยวิธีการเฉพาะในการสร้างคีย์ย่อยของ Blowfish Algorithm

2.5.1.6 แทนค่า P3 และ P4 ด้วยเอาท์พุตในข้อ 2.5.1.5

2.5.1.7 ทำขั้นตอนข้างต้นต่อไปเรื่อย ๆ จนสามารถแทนค่าข้อมูลใน P-array จนครบ จากนั้น ข้อมูล S-boxes จำนวน 4 ชุด และ P-array ในการเตรียมของ Blowfish ก็จะสามารถนำไปใช้เข้ารหัสลับแบบ Blowfish Algorithm ได้

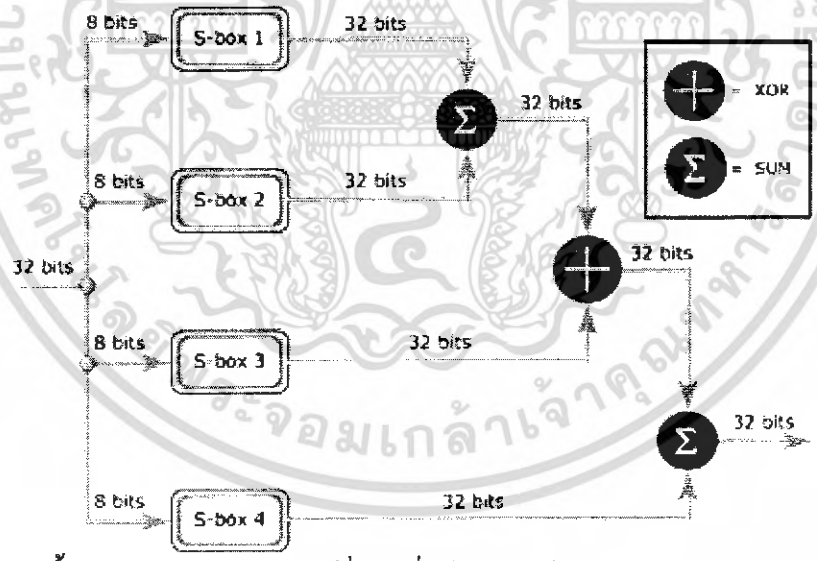
2.5.2 ขั้นตอนการเข้ารหัสลับด้วย Blowfish Algorithm



รูปที่ 2.3 แสดงขั้นตอนการเข้ารหัสลับข้อมูลด้วย Blowfish Algorithm

ในการเข้ารหัสลับข้อมูลด้วยวิธีการของ Blowfish Algorithm นั้นจะทำทั้งหมด 16 รอบดังรูปที่ 2.3 ด้วยข้อมูลอินพุต 64 บิต โดยทำการแบ่งข้อมูล Plaintext ออกเป็น 32 บิตจำนวน 2 ชุด จากนั้น 32 บิตทางด้านซ้าย (ดูจากรูปที่ 3.2) จะทำการ XOR กับข้อมูลใน P-array เพื่อสร้างข้อมูลที่เรียกว่า P' นำค่า P' เข้าสู่ฟังก์ชันการคำนวณ F และนำค่าที่ออกจากฟังก์ชั่นไปทำการ XOR กับข้อมูล 32 บิตทางด้านขวา (ดูจากรูปที่ 3.3) ผลลัพธ์ที่ได้จะเรียกว่า F' จากนั้นนำค่า F' ไปแทนที่ข้อมูล 32 บิตทางซ้ายมือ และนำค่า P' ไปแทนที่ 32 บิตทางขวามือจากนั้นดำเนินขั้นตอนเช่นนี้อีก 15 รอบหรือมากกว่าจนครบถึง P16 เมื่อทำครบจนเสร็จสิ้นถึง P16 นำค่าผลสุดท้ายของ P' และ F' มาทำการ XOR กับ 2 ชุดข้อมูล P-array สุดท้าย (P17 และ P18) จากนั้นนำข้อมูลผลลัพธ์ทั้งหมดที่ออกมาจากการประมวลผลเข้ารหัสลับ เป็นข้อมูล Cipher ที่เข้ารหัสลับเรียบร้อยแล้ว จำนวน 64 บิต โดยสามารถดูรายละเอียดในรูปที่ 2.3 ได้

ฟังก์ชั่น F ที่ใช้ในกระบวนการประมวลผลเข้ารหัสลับข้อมูลด้วย Blowfish Algorithm นั้น โดยฟังก์ชั่น F จะรับอินพุต 32 บิต หรือ 4 ไบต์ เป็นดัชนีชี้ไปยังค่าในข้อมูล S-array ดูผลที่ได้แล้วทำการบวก และ XOR ตามรูปที่ 2.4 จนได้เอาท์พุตออกมาดังรูป



รูปที่ 2.4 แสดงขั้นตอนการทำงานในฟังก์ชั่น F ที่ใช้ในการเข้ารหัสลับแบบ Blowfish Algorithm

2.6 โปรโตคอล BB84

BB84 เป็นโปรโตคอลในการรักษาความปลอดภัยของข้อมูลที่ส่งผ่านช่องสัญญาณควอนตัมจากการถูกดักจับจากบุคคลที่สาม ถูกสร้างขึ้นในปี 1984 โดย Bennett and Brassard ซึ่งมีหลักสำคัญคือ จะใช้ 4 สถานะสำหรับข้อมูล Binary (0 , 1) ที่จะใช้ส่งใน Quantum Channel โดยอาศัยคุณสมบัติเฉพาะของโฟตอน ซึ่งมีขั้นตอนดังต่อไปนี้

2.6.1 ผู้ส่งทำการสร้างคีย์แบบสุ่มขึ้นมา

2.6.2 ทำการสุ่มเวคเตอร์ฐานแล้วใช้การเข้ารหัสคีย์ที่สร้างขึ้นก่อนจะส่งไปทาง ควอนตัมแชนแนล

2.6.3 ผู้รับจะสุ่มเวคเตอร์ฐานเพื่อถอดรหัสที่ผู้ส่งได้ส่งมาโดยเมื่อรับคีย์ครบแล้วจะติดต่อไปหาผู้ส่งผ่าน public network เพื่อทำการยืนยัน-ตรวจสอบคีย์ซึ่งมีขั้นตอนย่อย ๆ ดังต่อไปนี้

- ผู้รับส่งเวคเตอร์ฐานที่ตนเองสุ่มขึ้นมา ให้ผู้ส่ง โดยทั้งสองฝั่งจะลบบิตที่มีเวคเตอร์ฐานไม่ตรงกันออก
- ทั้งสองฝั่งจะได้ raw key ที่ได้จากการตัดบิตที่สุ่มตรงกัน

2.6.4 ทำการตรวจสอบการดักจับซึ่งมีขั้นตอนดังต่อไปนี้นำข้อมูล 10 % จาก raw key มาตรวจสอบกันระหว่างผู้รับ-ผู้ส่งโดย

- หากตรวจสอบแล้วคีย์มีความผิดพลาดเกิน 25 % จะถือว่ามีการดักจับเกิดขึ้นจะยกเลิกขั้นตอนทั้งหมด และเริ่มต้นสร้างคีย์และทำการส่งกันใหม่
- หากตรวจสอบแล้วไม่ถูกดักจับจะทำการตรวจสอบขั้นต่อไปนี้

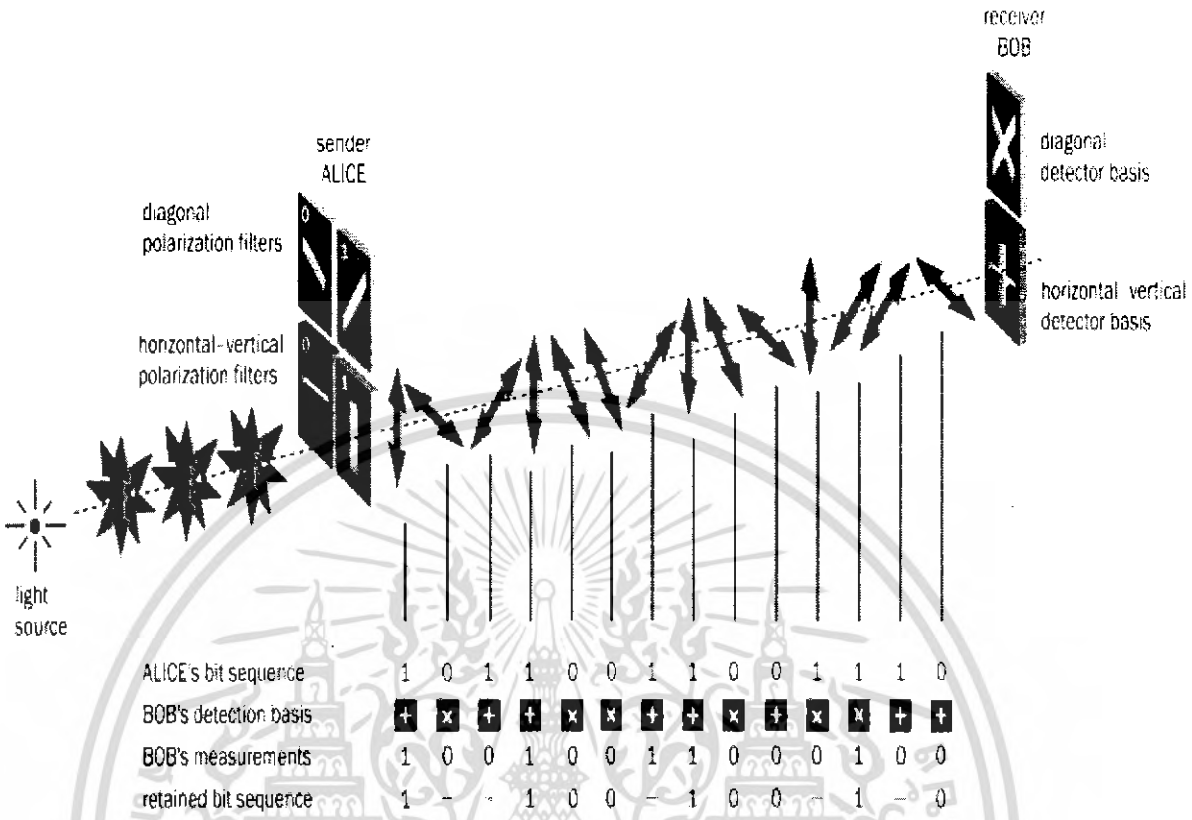
2.6.5 ทำให้คีย์ทั้งสองฝั่งเหมือนกัน Error Correction โดยใช้วิธีการเช็คพาริตี้บิต

2.6.6 จะได้คีย์ที่เหมือนกันทั้งสองฝั่งเพื่อนำไปใช้เข้ารหัส และ ผู้รับก็เก็บคีย์นี้ไว้ถอดรหัส

จะเห็นได้ว่าโปรโตคอลนี้มีจุดแข็งที่ทำให้ยากต่อการคาดเดา หรือ ดักจับรหัสได้ในหลายประเด็นดังต่อไปนี้

- ผู้ส่งสุ่มคีย์ที่จะใช้ส่ง และ สุ่มเวคเตอร์ฐานที่จะรับ
- ผู้รับสุ่มเวคเตอร์ฐานที่จะรับ

หากมีคนดักจับข้อมูลไปจะทำให้คุณสมบัติโฟตอนเปลี่ยนไป ปริมาณความผิดพลาดของข้อมูลจะมากขึ้นทำให้ตรวจสอบได้ว่าถูกดักจับ และยกเลิกคีย์นั้น ทำให้คีย์ที่ดักจับไปไม่สามารถนำไปใช้ประโยชน์ได้เลย นอกจากนี้ระบบจัดการคีย์จะเป็นไปโดยอัตโนมัติผู้ใช้ไม่จำเป็นต้องจดจำคีย์ที่ใช้ จึงทำให้ใช้คีย์ 1 ครั้ง ต่อการเข้ารหัส 1 ครั้งเช่นกัน ดังนั้น แม้ว่าผู้ดักจับจะใช้วิธีสุ่มรหัสผ่าน จนสามารถรู้คีย์ที่จะถอดรหัสบางไฟล์ได้ แต่ก็ไม่สามารถใช้คีย์นี้กับทุกไฟล์ได้



รูปที่ 2.5 แสดงตัวอย่างการทำงานของ BB84 Protocol

2.7 การสื่อสารแบบอนุกรม (Serial Port)

หัวข้อนี้จะกล่าวถึงทฤษฎีในส่วนของการติดต่อสื่อสารแบบอนุกรม หรือการสื่อสารผ่าน Serial Port ซึ่งมีส่วนเกี่ยวข้องในการประยุกต์สร้างซอฟท์แวร์จำลองการทำงานของรหัสลับเชิงควอนตัม ซึ่งจำเป็นต้องใช้การสื่อสารผ่าน Serial Port แทนช่องสัญญาณควอนตัมเพื่อให้เห็นถึงภาพลักษณ์ของการส่งข้อมูลในระบบวิทยาการรหัสลับเชิงควอนตัม แต่ในโครงงานไม่สามารถทำการส่งผ่านทางช่องสัญญาณควอนตัมได้ เนื่องจากอุปกรณ์ทางแสงนั้นเป็นเรื่องยากและยากต่อการนำมาใช้งาน จึงใช้ วิธีการส่งผ่าน Serial Port แทน โดยข้อมูลเหล่านี้จะช่วยเป็นประโยชน์ต่อการศึกษาและพัฒนาซอฟท์แวร์เป็นอย่างยิ่ง

2.7.1 พื้นฐานการสื่อสารแบบอนุกรม

การสื่อสารแบบอนุกรมสามารถแบ่งออกเป็น 3 รูปแบบ ดังนี้

2.7.1.1 Simplex สามารถส่งข้อมูลได้อย่างเดียว เป็นการสื่อสารทางเดียว

2.7.1.2 Half-Duplex สามารถส่งข้อมูลไปยังปลายทางและสามารถรับข้อมูลจากปลายทางได้ แต่ไม่สามารถทำการส่งและรับข้อมูลได้ในเวลาเดียวกัน

2.7.1.3 Full-Duplex สามารถรับและส่งข้อมูลได้ในเวลาเดียวกัน

นอกจากนี้แล้วยังสามารถแบ่งประเภทของการสื่อสารแบบอนุกรมตามลักษณะสัญญาณในการส่งได้อีก 2 แบบคือ การสื่อสารแบบซิงโครนัส เป็นการสื่อสารที่ใช้สัญญาณนาฬิกาควบคุมการรับส่งสัญญาณ โดยจะมีสายสัญญาณเส้นหนึ่งเป็นสายของข้อมูล อีกเส้นหนึ่งเป็นสัญญาณนาฬิกา ส่วนการสื่อสารแบบ อะซิงโครนัสเป็นการสื่อสารที่ใช้สายสัญญาณข้อมูลอย่างเดียวแต่จะใช้รูปแบบการส่งข้อมูลเป็นตัวกำหนดว่าส่วนไหนเป็นส่วนเริ่มของข้อมูล ส่วนไหนเป็นตัวข้อมูล โดยจะกำหนดให้สัญญาณนาฬิกาเท่ากันทั้งภาครับและภาคส่ง ในการส่งข้อมูลแบบนี้จะมีตัวควบคุมที่ชื่อว่า UART ควบคุมการรับและส่งข้อมูล

2.7.2 องค์ประกอบของการรับส่งข้อมูลแบบอนุกรม

การสื่อสารแบบอนุกรมที่นิยมใช้กับคอมพิวเตอร์นั้น เป็นการสื่อสารข้อมูลแบบอะซิงโครนัส นั่นคือ ต้องใช้สายสัญญาณเส้นเดียวทำหน้าที่ทั้งส่งส่วนที่เป็นข้อมูลและส่วนที่ใช้ควบคุมการส่งข้อมูล ดังนั้นข้อมูลที่อ่านได้แต่ละบิตจากการส่งแบบอนุกรมจึงต้องถูกแยกแยะว่าใช้สำหรับวัตถุประสงค์ใด โดยเราสามารถแบ่งได้เป็น 4 ส่วนคือ

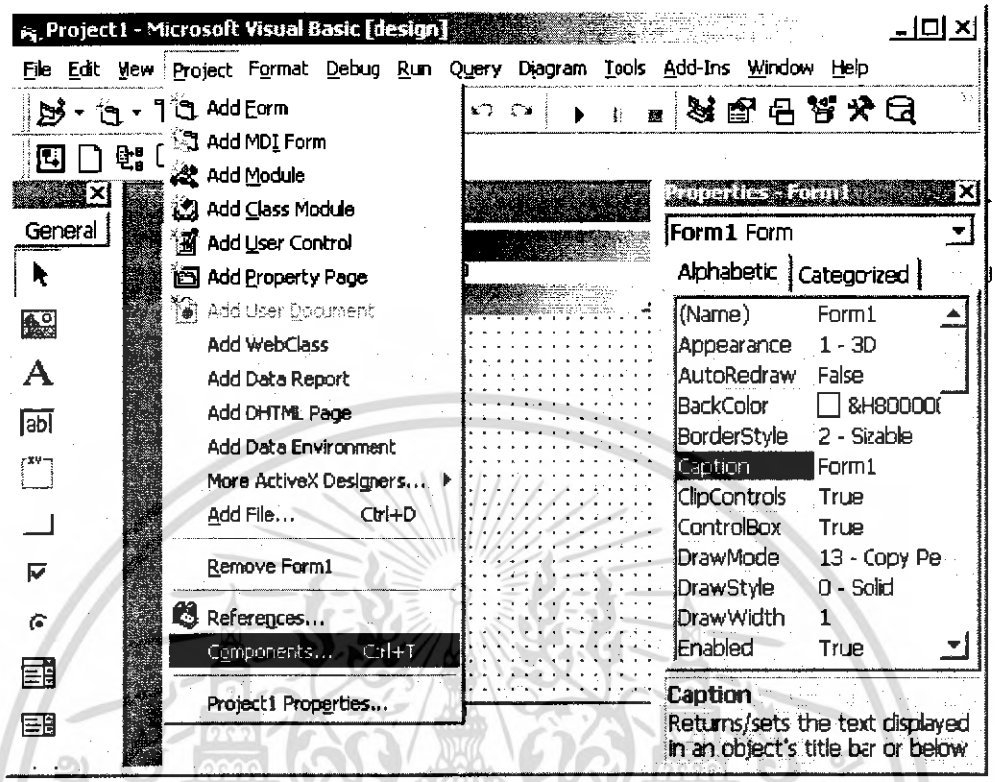
- Start Bit ขนาด 1 บิต
- บิตข้อมูล (Data Character) ขนาด 7 หรือ 8 บิต
- Parity Bit ขนาด 1 บิต
- Stop Bit ขนาด 1 หรือ 2 บิต

2.7.3 อัตราเร็วในการรับส่งข้อมูลแบบอนุกรม

การที่อุปกรณ์ 2 อย่างจะติดต่อสื่อสารกันได้นั้น จะต้องทำงานด้วยอัตราเร็วเท่ากัน ซึ่งอัตราเร็วในการสื่อสารแบบอะซิงโครนัสคือ ค่าบอดเรต (Baud Rate) มีหน่วยเป็นบิตต่อวินาที ซึ่งค่าอัตราเร็วในการสื่อสารแบบอนุกรมสำหรับมาตรฐาน RS-232C นั้นมีใช้ดังนี้ 110, 150, 300,600, 1200, 2400, 4800, 9600 และ 19200 บิตต่อวินาที

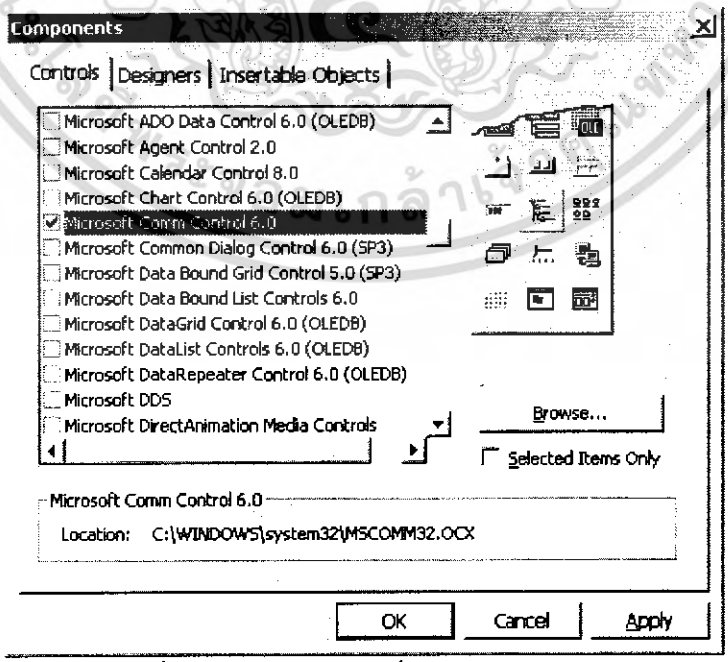
2.7.4 การเขียนโปรแกรมติดต่อและควบคุม Serial Port กับ Visual Basic

คอนโทรลที่สำคัญในการทำให้ Visual Basic สามารถสื่อสารผ่านพอร์ตอนุกรมได้นั่นก็คือคอนโทรล MSComm ซึ่งไม่ใช่คอนโทรลมาตรฐาน ดังนั้นถ้าเราต้องการใช้งาน MSComm เราจะต้องทำการเพิ่มคอนโทรลนี้เข้าไปใน Toolbox ซึ่งสามารถกระทำได้โดยคลิกขวาที่ Toolbox แล้วเลือกเมนู Components ดังรูป 2.10-2.12



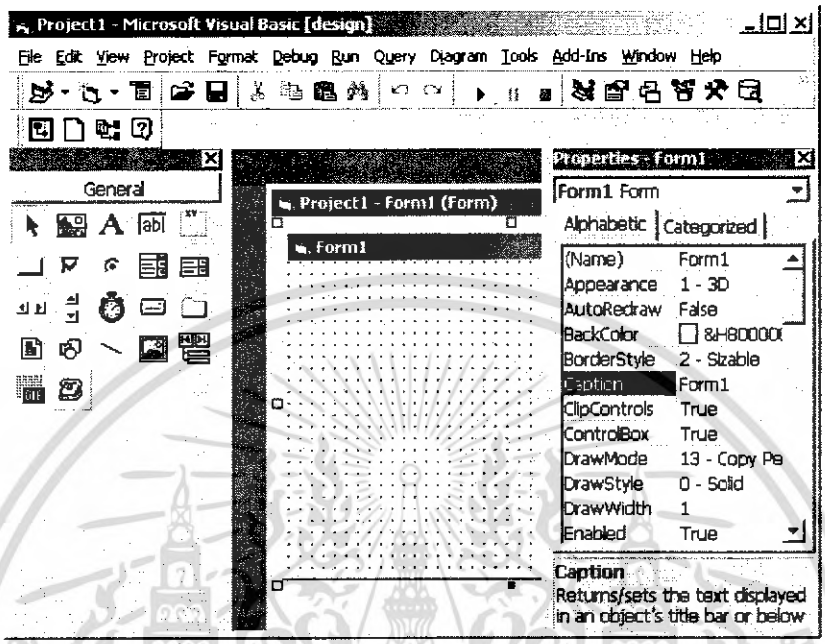
รูปที่ 2.6 แสดงการเพิ่มคอมโพเนนต์ MSComm

จากนั้นจะปรากฏไดอะล็อก Components ขึ้นมา จากนั้นให้คลิกเลือกที่ Microsoft Comm Control 6.0 แล้วคลิกปุ่ม OK ดังรูป



รูปที่ 2.7 แสดงการเลือกที่รายการ MSComm

จากนั้นก็จะปรากฏภายใน Toolbox จะมีไอคอนรูปโทรศัพท์ ซึ่งเป็นไอคอนของคอนโทรล MSComm ปรากฏขึ้นมาให้เราเลือกใช้งาน



รูปที่ 2.8 แสดงคอนโทรล MSComm พร้อมทำงาน

2.7.5 พรอพเพอร์ตี้ที่สำคัญในการใช้งาน MSComm [2]

- **CommPort** ใช้ในการกำหนดหมายเลขของพอร์ตอนุกรมที่เราต้องการจะติดต่อ

  รูปแบบ => object.CommPort [= value]

  ตัวอย่าง => MSComm1.CommPort=1

- **Settings** ใช้ในการกำหนดอัตราบอด (Baud Rate) หรือ ความเร็วในการส่งข้อมูลมีหน่วยเป็นบิตต่อวินาที พาริตี้ จำนวนของบิตข้อมูล จำนวนของบิตปิดท้าย โดยมีรูปแบบของการใช้งานดังนี้

  รูปแบบ => object.Settings [=value]

  ตัวอย่าง => MSComm1.Settings="1200,N,8,1"

- **PortOpen** ใช้สำหรับเปิดและปิดการใช้งานพอร์ตอนุกรม โดยมีรูปแบบของการทำงานดังนี้

  รูปแบบ => object.PortOpen [=value]

  รูปแบบ => MSComm1.PortOpen=True

- **InBuffersize** เป็นการกำหนดขนาดของ Buffer ในการรับข้อมูลเข้ามาโดยมีรูปแบบการกำหนดค่าดังนี้

  object.InBuffersize [= value]

- **OutBufferize** เป็นการกำหนดขนาดของ Buffer ในการส่งข้อมูลออกไปโดยมีรูปแบบการกำหนดค่าดังนี้

  object.OutBuffersize [=value]

- **Inputlen** เป็นการกำหนดค่าของข้อมูลที่อ่านจาก Buffer ภาครับ โดยมีแบบการกำหนดค่าดังนี้

  object.Inputlen [=value]

- **InputMode** เป็นการกำหนดค่าชนิดของข้อมูลที่รับเข้ามา โดยมีรูปแบบการกำหนดค่าดังนี้

  object.InputMode [=value]

  โดยที่เราสามารถเลือกชนิดของข้อมูลได้ 2 ประเภท คือ

  - comInputModeText ข้อมูลที่รับเข้ามาเป็นข้อความปกติ เราสามารถตั้งค่าให้อยู่ในโหมดนี้ได้โดยการกำหนด value ให้เป็น "0"
  - comInputModeBinary ข้อมูลที่รับเข้ามาเป็นข้อมูลไบนารี่ เราสามารถตั้งค่าให้อยู่ในโหมดนี้ได้โดยการกำหนดค่า Value ให้เป็น "1"

- **Input** ใช้ในการอ่านค่าข้อมูลจากพอร์ตอนุกรม โดยมีรูปแบบการอ่านค่าดังนี้

  object.Input

- **Output** ใช้ในการส่งข้อมูลออกไปจากพอร์ตอนุกรม โดยมีรูปแบบของการเขียนดังนี้

  object.Output [=value]

- **EOFEnable** เป็นการบอกว่าสิ้นสุดของไฟล์ End of File [EOF] โดยมีรูปแบบการใช้งานดังนี้

  object.EOFEnable [=value]

---

[2] Properties ต่าง ๆ ทำการเรียบเรียงเนื้อหาจาก www.thaiio.com

2.8 การเขียนโปรแกรม Visual Basic ควบคุมการสื่อสารผ่าน Network

ปัจจุบันการสื่อสารผ่านระบบเครือข่าย (Network) รวมทั้งระบบ Internet เป็นที่แพร่หลายมากในแง่ของการใช้งานในชีวิตประจำวันต่าง ๆ เนื่องจากระบบสื่อสารที่รวดเร็วและสามารถเชื่อมโยง เข้ากับหลาย ๆ ระบบได้ ทำให้สามารถนำมาประยุกต์ใช้งานผ่านระบบเครือข่ายได้หลากหลายรูปแบบ

ในการพัฒนาซอฟท์แวร์จำลองการทำงานของรหัสลับเชิงควอนตัมก็มีการติดต่อสื่อผ่านผ่านทางช่องสัญญาณสาธารณะ หรือ Internet ด้วย เนื่องจากในการส่งข้อมูลระหว่างผู้รับ-ผู้ส่งของระบบ จำเป็นต้องมีการติดต่อสื่อสารกัน มีการส่งข้อมูลที่ผ่านการเข้ารหัสแล้วผ่านช่องสัญญาณสาธารณะ มีการตรวจสอบความถูกต้องของกุญแจไขความลับที่ได้จากช่องทางอื่น (ช่องสัญญาณควอนตัม) ผ่านทางช่องสัญญาณสาธารณะอีกด้วย ดังนั้นผู้พัฒนาและผู้ศึกษาจะต้องมีความรู้พื้นฐานในด้านการติดต่อสื่อสารผ่านทาง Internet ด้วยเช่นกัน

2.8.1 TCP/IP

TCP/IP เป็นโปรโตคอลมาตรฐานในการติดต่อสื่อสารผ่านระบบ Internet กับเครื่องคอมพิวเตอร์เครื่องอื่น ๆ โดยโปรโตคอล TCP/IP จะมีส่วนประกอบที่สำคัญ 2 ส่วน คือ TCP (Transmission Control Protocol) และ IP (Internet Protocol) ในความเป็นจริงแล้วเราไม่สามารถเห็นขั้นตอนการทำงานของระบบได้เพราะเป็นการทำงานของ Software กับ Hardware แต่จะอธิบายเพื่อความเข้าใจของโปโตคอล TCP/IP ว่ามีส่วนประกอบดังนี้

2.8.1.1 IP Address สำหรับการรับส่งข้อมูลในระบบ Internet จะถูกกำหนดและอ้างอิงด้วยหมายเลขประจำเครื่องนั้นก็คือ IP Address

2.8.1.2 Routing Configuration ข้อดีของโปโตคอล TCP/IP ก็คือในการกำหนดเส้นทางสำหรับการรับส่ง ที่สามารถเลือกเส้นทางในการรับส่งข้อมูลได้อย่างอัตโนมัติ หากถ้าเกิดเส้นทาง บ้างเส้นทางเสียหาย ระบบกลไกในการกำหนดเส้นทางสำหรับรับส่งข้อมูลของโปรโตคอล TCP/IP ก็จะเลือกเส้นทางให้เหมาะสมถูกต้องให้สามารถรับส่งข้อมูลได้

2.8.1.3 Protocol, Ports, Socket เป็นช่องทางสำหรับกำหนดทิศทางของการรับส่งข้อมูลนอกเหนือจากที่จะต้องกำหนดหลังจาก IP Address

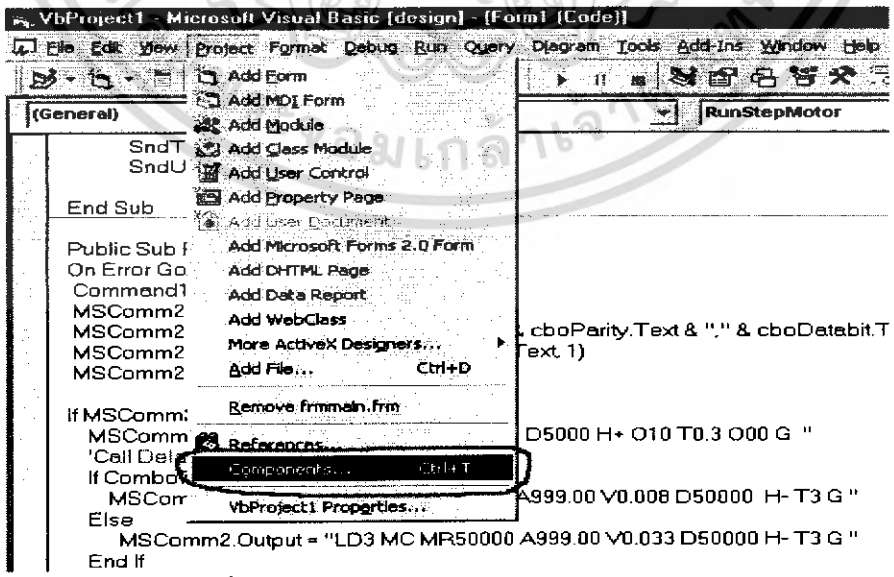
2.8.2 Server and Client

เมื่อพูดถึงการเขียนโปรแกรมเพื่อใช้ติดต่อสื่อสารในระบบเครือข่าย (Network) จะมีการแบ่งฝ่ายผู้ติดต่อกันโดยจะแบ่งเป็น 2 ส่วนคือ Server กับ Client ซึ่ง Server จะทำหน้าที่เหมือนเป็นศูนย์กลางการทำงานของระบบ ส่วน Client จะทำหน้าที่เป็นผู้ใช้หรือผู้ติดต่อเพื่อทำงานกับ Server นั้นๆ ทั้ง Server และ Client จะต้องมี IP Address ของตัวเอง และมีช่องทางการติดต่อ (Port) โดยจะติดต่อกันได้นั้น ต้องมี Port ที่ตรงกันซึ่งเราสามารถกำหนดหมายเลขของพอร์ตได้ และในการเขียนโปรแกรมนั้นจะต้องมีการอ้างอิงหมายเลขของ Port ทุกครั้งเช่นกัน

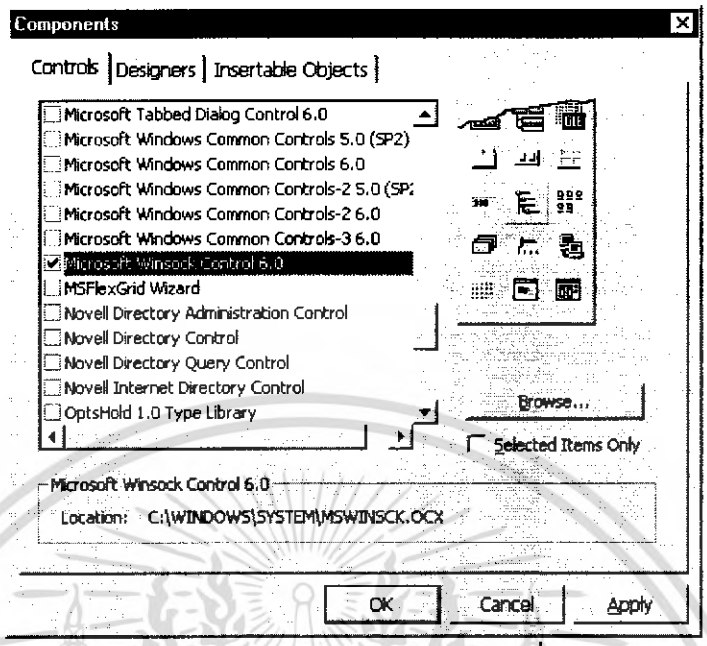
2.8.3 MS Winsock Control 6

Winsock Control 6 เป็นเครื่องมือที่ช่วยให้เราสามารถเขียนโปรแกรมติดต่อผ่านระบบเครือข่ายได้ โดยประยุกต์ใช้กับในการเขียนโปรแกรมโดยใช้ Visual Basic เนื่องจากการพัฒนาซอฟท์แวร์ของโครงงานซอฟท์แวร์จำลองการทำงานของรหัสลับเชิงควอนตัมนั้นได้พัฒนาโดย Visual Basic เป็นหลัก ซึ่งต่อไปจะอธิบายถึงการใช้งาน Winsock Control รวมทั้งอธิบายถึง Properties ที่จำเป็นซึ่งเป็นประโยชน์ต่อเขียนโปรแกรม และส่วนอื่น ๆ ที่จำเป็นต่อการเขียนโปรแกรมดังนี้

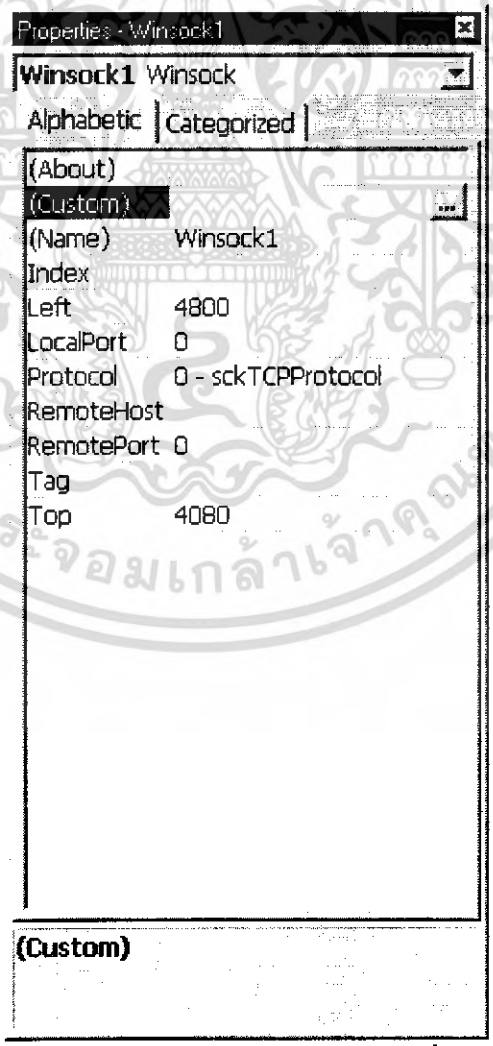
2.7.3.1 Add Winsock Control ในการการอธิบายในส่วนของการ Add Winsock Control นั้นถ้าอธิบายโดยประโยคอาจจะเกิดความสับสนได้ จึงขออธิบายด้วยรูปภาพประกอบแทนซึ่งจะเป็นขั้นตอนเรียงลำดับตามภาพประกอบ



รูปที่ 2.9 แสดงการเลือกเข้าสู่การ Add Winsock Control



รูปที่ 2.10 แสดงการเลือก Microsoft Winsock Control 6.0 เพื่อใช้ในการเขียนโปรแกรม



รูปที่ 2.11 แสดงตัวอย่าง Properties Winsock Dialog และหน้าที่ต่างๆของ Properties

2.8.3.2 Winsock Procedure ในส่วนของ Windows Code Object -->Winsock มี Procedure สำหรับทำการติดต่อสื่อสารทั้งทางฝ่าย Server และ Client มีดังนี้

- Close คือ เหตุการณ์เมื่อมีหยุดหรือยกเลิกการติดต่อสื่อสารของฝ่าย Server หรือ Client โดย Function Winsock.Close ซึ่งจะเราสามารถจะใช้ตรวจสอบ ฝ่ายตรงข้ามว่ามีการติดต่ออยู่หรือไม่ โดยอาจจะใส่ Massage เตือนเป็นต้น

- Connect เป็นเหตุการณ์ที่ฝ่าย Client มีการส่งสัญญาณติดต่อกับมายัง Sever ส่งผลให้ Procedure นี้ของฝ่าย Server ก็เลยทำงานขึ้นมาเหมือนเดิมเราสามารถนำ Code Massage ไปใส่เพื่อตวจสอบได้เช่นกัน

- ConnectionRequest เป็นเหตุการณ์เมื่อฝ่าย Client ส่งสัญญาณติดต่อกลับมายัง Server Procedure ส่วนนี้ก็จะทำงานพร้อมกับค่า requestID As Long ซึ่งเป็นหมายที่ Gen ขึ้นมาในระบบค่านั้นจะไม่เหมือนเดิม โดยจะให้ฝ่าย Server รับรู้ว่าใช้ ID จากคอนโทรลตัวใดเพื่อจะได้สื่อสารถูกต้อง

- DataArrival เหตุการณ์นี้เกิดขึ้นเมื่อมีการส่งข้อมูลระหว่าง Server และ Client Procedure นี้ก็จะทำงานขึ้นมาพร้อมกับค่าจำนวน bytesTotal As Long ที่รับเข้ามา

- Error เหตุการณ์ที่เกิดความผิดพลาดระหว่างการติดต่อสื่อสารระหว่าง Server และ Client โดยจะส่งค่า Number As Integer มาให้ว่าเป็นหมายเลขใดพร้อมทั้งรายละเอียดของการผิดพลาดในเหตุการณ์นั้นๆ คือ Description As String

- SendProgress จะเกิดขึ้นในขณะที่มาการส่งข้อมูลอยู่เหตุการณ์นี้ก็จะทำงานเมื่อส่งข้อมูล หมดแล้วก็จะส่งผลทำให้เกิด Event SendComplete

- SendComplete เหตุการณ์เมื่อมีการส่งข้อมูลออกไปยังฝ่ายตรงข้ามเสร็จเรียบร้อยแล้ว

2.8.3.3 Winsock Properties and Events

Accept (requestID) คือการดกลงกันระหว่าง Server และ Client ในการเลือกหมายเลข ID Control ให้ตรงกันเพื่อสามารถสื่อสารได้ถูกต้อง

Close เป็นการส่งสัญญาณยกเลิกการติดต่อระหว่างกัน จะเป็นฝ่าย Server หรือ Client ก็ได้ ที่จะใช้ Function นี้ จากนี้จะทำให้ Procedure close ในฝ่ายตรงข้ามทำงาน

Connect เป็นการส่งสัญญาณว่าตอนนี้ทำการติดต่อเรียบร้อยแล้ว ซึ่งจะส่งผลให้ Procedure ฝ่ายตรงข้ามทำงาน

Getdata เป็นการรับข้อมูลเมื่อฝ่ายตรงข้ามส่งมาโดยประโยคคำสั่งนี้จะอยู่ในส่วนของ Procedure DataArrival เนื่องจากเป็นเหตุการณ์ที่การกระทำขณะเมื่อฝ่ายตรงข้ามส่งข้อมูลเข้ามา

Listen การกระทำที่จะคอยตรวจสอบสัญญาณที่ส่งไปว่าฝ่ายตรงข้ามตอบรับการร้องขอการติดต่อ

LocalHostName คำสั่งนี้จะส่งชื่อของ Computer name ของเครื่องนั้นๆ

```
Debug.Print  Winsock1. LocalHostName
```

LocalIP คำสั่งนี้จะทำการส่งหมายเลข IP Address

```
Debug.Print  Winsock1.LocalIP
```

LocalPort คำสั่งที่จะส่งค่าของหมายเลขในการดิดต่อ TCP/IP ของเครื่องนั้นๆ

```
Debug.Print  Winsock1.LocalPort
```

RemoteHost กำหนดหรือคืนค่าชื่อ Computer name ของเครื่องที่จะทำการติดต่อ

```
Winsock1.RemoteHost =MyServer
```

RemoteHostIP กำหนดหมายเลข IP Address ของเครื่องที่จะทำการติดต่อ

Winsock1. RemoteHostIP =10.10.0.0

RemoteHostPort กำหนดหมายเลข Port ที่จะใช้ในการติดต่อระหว่างกัน

Winsock1. RemoteHostIP =5000

SocketHandle จะคืนค่าของช่องทางที่ใช้ในการติดต่อระหว่างกันซึ่งสามารถเรียกดูได้ดังนี้

Debug.Print Winsock1.SocketHandle

State จะคืนค่าของสถานะของ Socket ขณะที่ใช้ติดต่อระหว่างอยู่โดยอาจจะใช้ตรวจสถานะโดยค่าคงที่เหล่านี้เช่น sckClosed (มีค่า=0) Socket ปิดการใช้งาน, sckOpen (มีค่า= 1) Socket เปิดใช้งาน หรือ sckError(มีค่า = 9) Socket มีความผิดผลาดเกิดขึ้น เป็นต้น

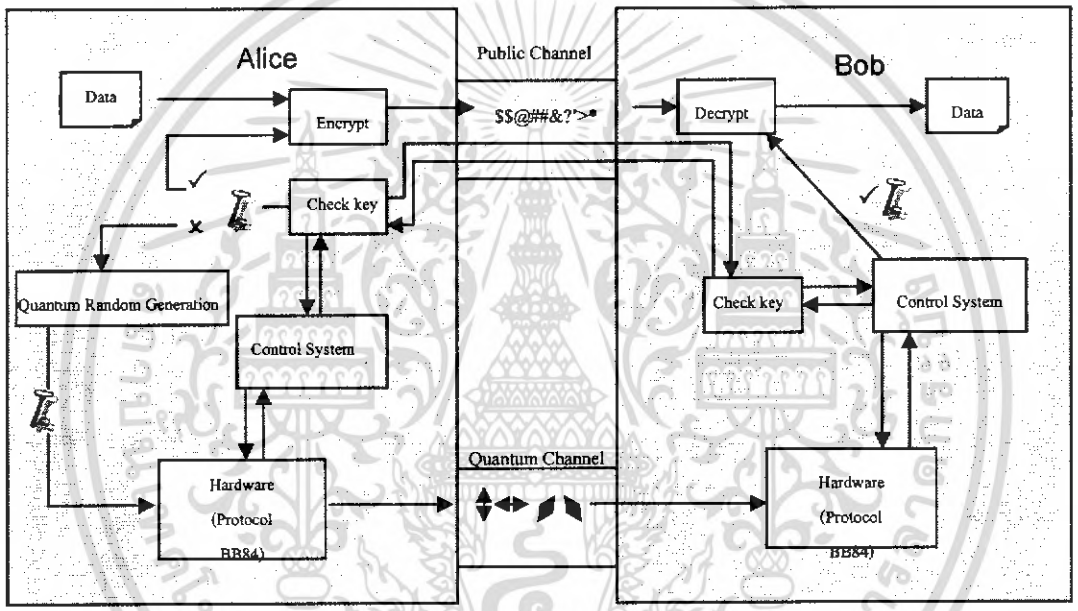
ในส่วนของข้อมูลของการเขียนโปรแกรมการติดต่อ Network โดย Winsock Control[3] มีรายละเอียดมากจึงขออธิบายไว้พอสังเขป

ทั้งหมดนี้เป็นทฤษฎีประกอบในการพัฒนาซอฟท์แวร์ที่จำเป็นต่อผู้พัฒนาและผู้สนใจ เพื่อช่วยให้ง่ายต่อการเข้าในการพัฒนาและรูปแบบของโครงงานที่แท้จริง

[3] เรียบเรียงจาก www.thaiio.com

บทที่ 3
การวิเคราะห์และออกแบบระบบ

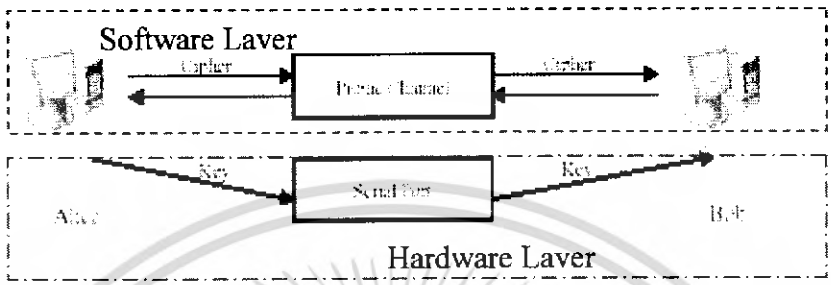
โครงงานซอฟท์แวร์จำลองการทำงานของรหัสลับเชิงควอนตัม เป็นโครงงานที่เน้นไปทางด้านซอฟท์แวร์เป็นหลัก ซึ่งการพัฒนานี้จะต้องมีการวิเคราะห์ระบบและการออกแบบ เพื่อให้ตรงกับความต้องการของผู้ใช้ และตรงตามวัตถุประสงค์ที่ตั้งไว้ ซึ่งระบบที่จะทำการจำลองจะมีขั้นตอนการทำงานดังรูปที่ 3.1



รูปที่ 3.1 แสดงระบบวิทยาการรหัสลับเชิงควอนตัม

จากรูปที่ 3.1 เป็นการแสดงการทำงานของระบบวิทยาการรหัสลับเชิงควอนตัม ที่เราจะใช้เป็นต้นแบบในการพัฒนาซอฟท์แวร์จำลองการทำงานของรหัสลับเชิงควอนตัม ซึ่งเป็นระบบที่จะแสดงให้เห็นถึงการทำงานหลัก ๆ ของระบบวิทยาการรหัสลับเชิงควอนตัม ซึ่งในซอฟท์แวร์จำลองจะไม่สามารถจำลองคุณสมบัติบางประการได้ ดังเช่น คุณสมบัติความปลอดภัยในช่องสัญญาณควอนตัมในเรื่องของคุณสมบัติทางโฟตอน เนื่องจากซอฟท์แวร์จำลองจะใช้ Serial Port แทนช่องสัญญาณควอนตัม แต่โดยภาพรวมก็จะสามารถจำลองการทำงานหลัก ๆ ได้ ทำให้ผู้ใช้ซอฟท์แวร์จำลองนี้ สามารถเข้าใจในระบบวิทยาการรหัสลับเชิงควอนตัมได้มากขึ้น

จากแผนภาพหลักของระบบวิทยาการรหัสลับเชิงควอนตัม จะประกอบด้วยฮาร์ดแวร์ ซึ่งได้แก่ คอมพิวเตอร์ สายไฟเบอร์ออปติก แต่ในโครงงานนี้ได้จำลองการทำงานของระบบจริงในส่วนหลัก ๆ ดังรูปที่ 3.2



รูปที่ 3.2 แสดงการแบ่งส่วนประกอบของระบบ

แต่เนื่องจากโครงงานนี้เป็นโครงงานที่มีจุดประสงค์หลักคือ เป็นส่วนเริ่มในการพัฒนาซอฟท์แวร์ที่ใช้ในระบบวิทยาการรหัสลับเชิงควอนตัม ซึ่งได้พัฒนาคู่ขนานไปกับทางด้านฮาร์ดแวร์ ซึ่ง NECTEC กำลังพัฒนาอยู่ แต่ในการพัฒนาส่วนของซอฟท์แวร์จำเป็นต้องมีการทดสอบความถูกต้องของการทำงาน จึงได้ประยุกต์ใช้ฮาร์ดแวร์ที่จัดหาได้ง่ายในการทดลองโปรแกรมต่าง ๆ ดังที่ปรากฏในโครงงานนี้นั่นเอง

อุปกรณ์และโปรแกรมที่ใช้ในการพัฒนาโครงงานนี้ส่วนใหญ่จะเป็นการเขียนโปรแกรมซึ่งเลือกใช้ Visual Basic 6.0 เนื่องจากมีความสามารถในการติดต่อกับฮาร์ดแวร์ได้ตรงตามวัตถุประสงค์ของโครงงาน รวมทั้งเป็นโปรแกรมที่ผู้จัดทำโครงงานมีความถนัดในการใช้งานอีกด้วย นอกจากนี้ยังมีอุปกรณ์อื่น ๆ ที่ใช้ในการพัฒนาอีกเช่น สาย Serial Port ซึ่งจะใช้เป็นช่องทางการติดต่ออีกช่องทางหนึ่ง (แทนช่องสัญญาณควอนตัม) รวมทั้งต้องใช้คอมพิวเตอร์ในการทดสอบซอฟท์แวร์จำนวนสองเครื่อง และใช้วิธีการออกแบบซอฟท์แวร์ด้วยวิธีการ UML

โดยในการออกแบบจะมีการแบ่งการออกแบบโครงงานเป็น 3 ส่วนหลัก ๆ ด้วยกัน ได้แก่ การออกแบบซอฟท์แวร์ การออกแบบฮาร์ดแวร์ และการออกแบบทางด้าน User Interface ซึ่งจะได้กล่าวต่อไป

3.1 การออกแบบซอฟท์แวร์ (Software Design)

ในการออกแบบในส่วนของซอฟท์แวร์ จะทำการออกแบบให้ซอฟท์แวร์มีคุณสมบัติและความสามารถในการทำงานตามการทำงานหลัก ๆ ของระบบวิทยาการรหัสลับเชิงควอนตัม ซึ่งระบบมีความสามารถดังต่อไปนี้คือ

- สามารถจัดการการเข้ารหัสไฟล์ได้
- ถอดรหัสไฟล์ได้ถูกต้อง
- ส่งข้อความสนทนาระหว่างกันได้
- แลกเปลี่ยนไฟล์ผ่านระบบเครือข่ายคอมพิวเตอร์
- จัดการและส่งข้อมูลผ่าน Serial Port
- จัดการการตรวจสอบคีย์โดยใช้มาตรฐาน BB84 Protocol

โดยซอฟท์แวร์ที่พัฒนานั้นใช้ Visual Basic 6.0 เป็นเครื่องมือในการพัฒนา

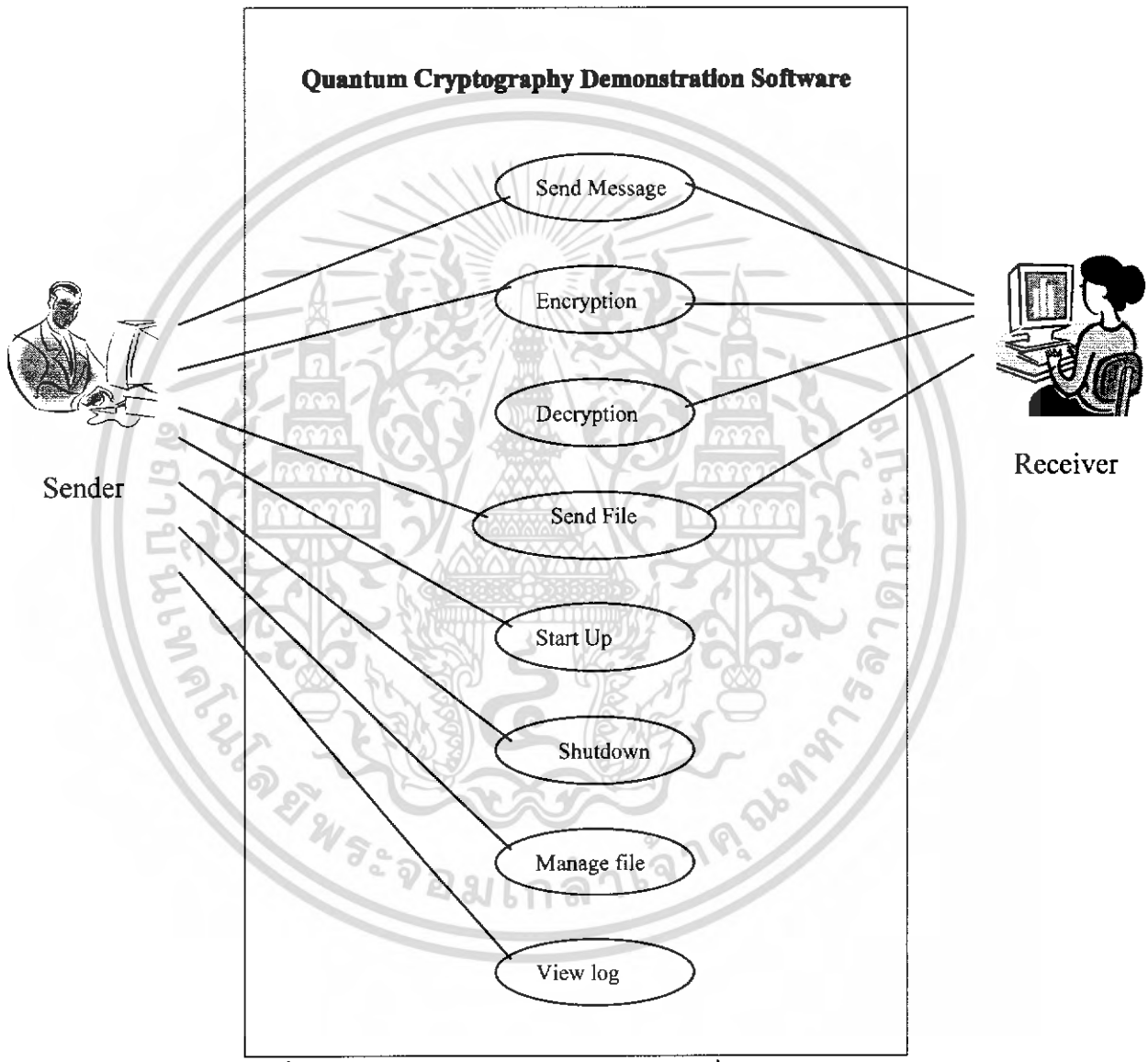
สำหรับการออกแบบนั้นได้ใช้วิธีการ UML โดยเป็นการออกแบบระบบจากแนวคิดแบบ object oriented ซึ่งมีรายละเอียดในแต่ละขั้นตอนโดยเริ่มจากความต้องการพื้นฐานของระบบไปจนถึงการออกแบบระดับซอฟท์แวร์ โดยในรายละเอียดการออกแบบด้วยวิธี UML สามารถดูรายละเอียดในภาคผนวกได้ แต่ ณ ที่นี้จะทำการยกตัวอย่างการออกแบบซอฟท์แวร์ที่มีความสำคัญในการออกแบบและพัฒนา เพื่อทำให้เห็นภาพของการออกแบบมากขึ้นดังนี้

Actor	Goal	UC (Use Case)
System admin (Sender and Receiver)	- Encrypt Data	-Encrypt Data
	- Decrypt Cipher	-Decrypt Cipher
	- Send Cipher and data to public channel	-Send File
	- Send MSG	-Send Message
	- Start up program	- Start server
	- Shutdown program	- Shutdown server
	-View log	- View log
	- manage file and key	- Manage file
	- Connect Server	- Connect Server
	- Receive File	- Receive File

ตารางที่ 3.1 แสดงการกำหนด Actor-Goal และกำหนด Use Caseของระบบ

3.1.1 Use Case Diagram

ในส่วนนี้เป็นการวิเคราะห์การใช้งานของระบบที่ได้ทำการออกแบบ ว่ามีผู้ใช้งานในลักษณะใดบ้าง และมีการใช้งานในส่วนไหนของซอฟท์แวร์บ้าง มีระบบที่คอยตอบรับและตอบสนองคำสั่งอัตโนมัติ ซึ่งอธิบายไว้ในรูปของ Diagram ดังรูป

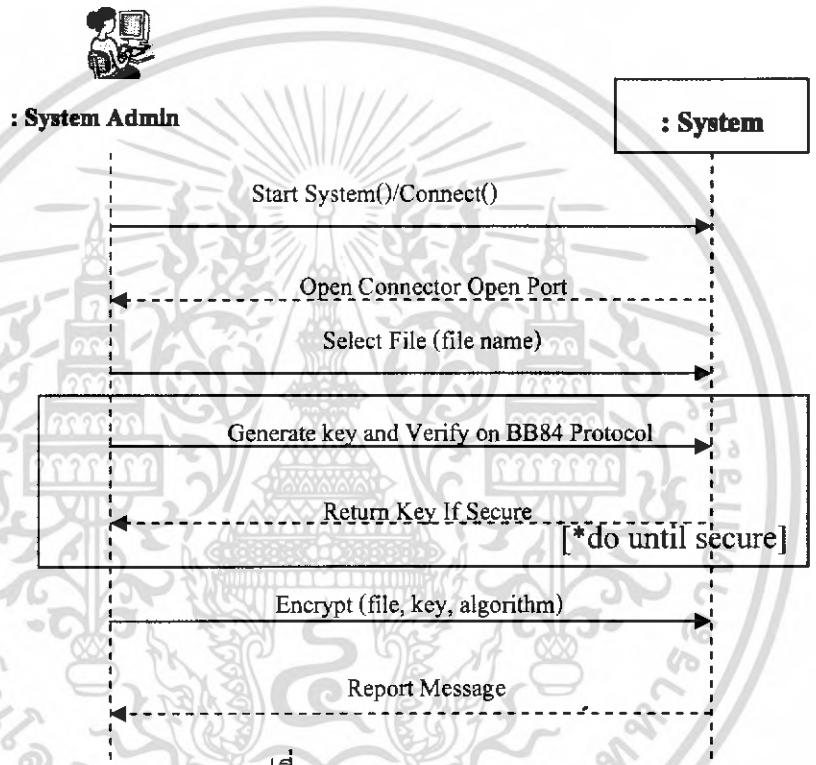


รูปที่ 3.3 แสดงUse Case Diagram ของระบบที่ออกแบบ

จากรูปที่ 3.3 แสดงให้เห็นถึงลักษณะความต้องการใช้งานของผู้ใช้ระบบ แสดงให้เห็นภาพรวมถึงความต้องการของผู้ใช้สู่การออกแบบและพัฒนาซอฟท์แวร์ ดังรูปข้างต้น ก็แสดงให้เห็นถึงความต้องการของผู้ใช้ซอฟท์แวร์ เช่น ผู้ส่งต้องการ เปิดระบบ ส่งไฟล์ สนทนา เข้ารหัสข้อมูล ดูบันทึกการใช้งาน และอื่น ๆ รวมทั้งผู้รับยังมีความต้องการที่จะถอดรหัส ส่งไฟล์ สนทนา และอื่น ๆ ซึ่งผู้พัฒนาซอฟท์แวร์ต้องนำข้อมูลส่วนนี้ไปใช้เป็นแนวทางในการพัฒนาต่อไป

3.1.2 System Sequence Diagram

เป็นการวิเคราะห์ระบบในแต่ละองค์ประกอบของระบบ ว่าแต่ละส่วนมีการทำงานอย่างไรบ้าง มีการรับคำสั่งจากผู้ใช้ และมีการตอบสนองจากระบบอย่างไรบ้าง ทำให้เห็นการทำงานระหว่างผู้ใช้และระบบ ได้อย่างชัดเจนมากขึ้น ซึ่งได้มีการวิเคราะห์ไว้อย่างละเอียดในส่วนของภาคผนวก ณ ที่นี้จะยกตัวอย่างการออกแบบการทำงานของระบบใน Process ที่มีความสำคัญต่อซอฟท์แวร์ที่ได้ทำการวิเคราะห์และออกแบบซึ่งได้แก่ Process Encrypt Data

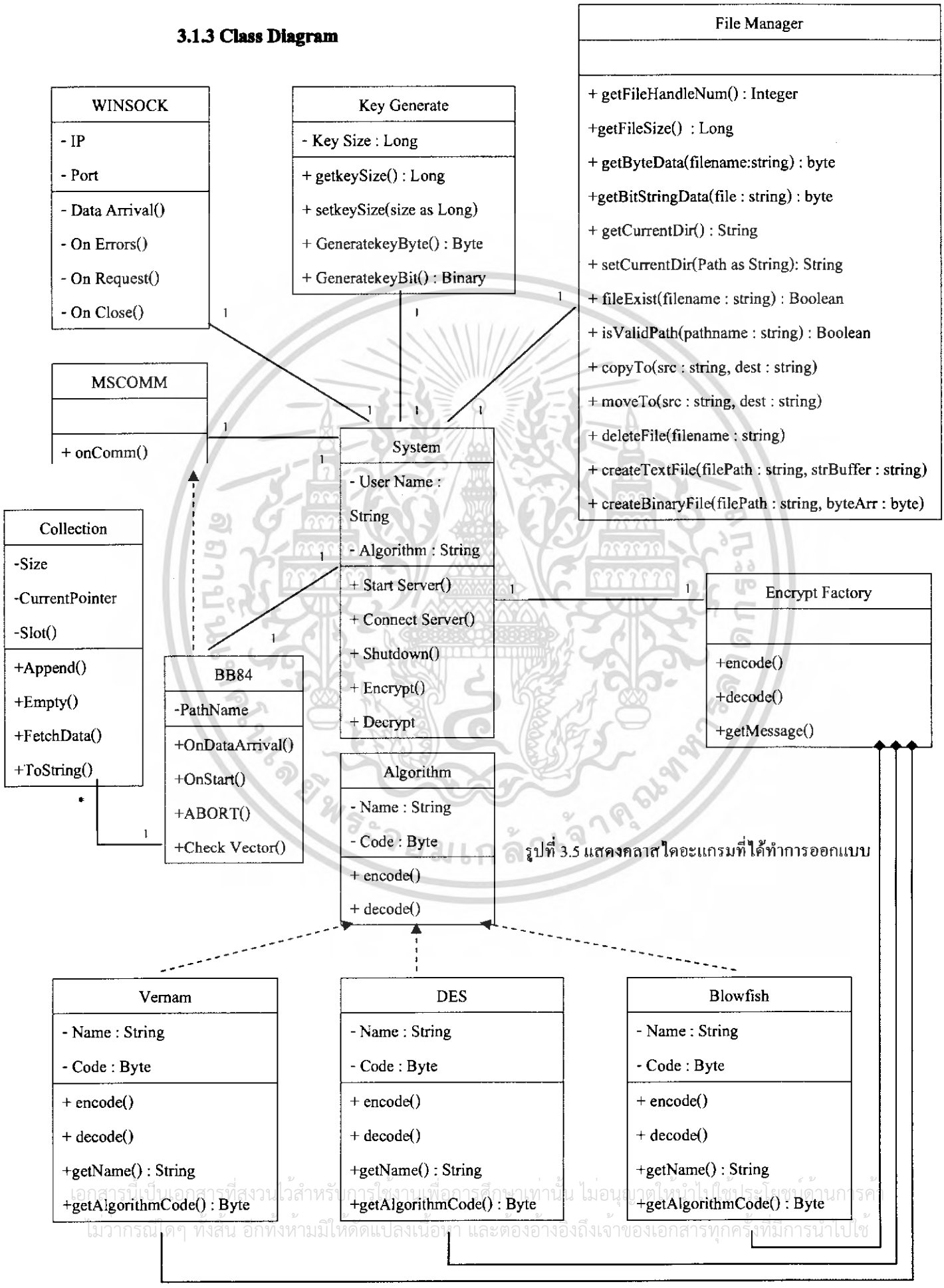


รูปที่ 3.4 SSD: Process Encrypt Data

จากรูปที่ 3.4 แสดงให้เห็นรายละเอียดของการทำงานในส่วนของ Process Encrypt Data โดยจากไดอะแกรม จะมีการทำงานของระบบดังนี้

- จะเริ่มจากผู้ดูแลระบบทำการเปิดระบบ และทำการติดต่อไปยังผู้รับ
- ระบบจะรับคำสั่ง และทำการเปิดพอร์ต เพื่อทำการเชื่อมต่อระบบ
- ผู้ดูแลระบบทำการเลือกไฟล์ที่ต้องการเข้ารหัสข้อมูล
- ผู้ดูแลระบบทำการเริ่มคำสั่งสุ่มคีย์ และตรวจสอบความปลอดภัยผ่าน BB84 โปรโตคอล
- ระบบจะคืนค่าคีย์มาให้ หากพบว่ามีความปลอดภัย
- ผู้ดูแลระบบทำการเลือกไฟล์ คีย์ อัลกอริทึม แล้วสั่งให้เริ่มทำการเข้ารหัสข้อมูล จากนั้นระบบจะทำการรายงานการทำงานให้ทราบ

3.1.3 Class Diagram



รูปที่ 3.5 แสดงคลาสไดอะแกรมที่ได้ทำการออกแบบ

จากรูปที่ 3.5 ได้แสดงคลาสต่าง ๆ ที่ได้ทำการออกแบบไว้เพื่อใช้ประโยชน์ในการเขียนโปรแกรมจริง ช่วยเป็นแนวทางในการเขียนโปรแกรมรวมทั้งเมื่อมีปัญหาการทำงานของโปรแกรมก็สามารถมาวิเคราะห์หาสาเหตุจากคลาสไดอะแกรมนี้ได้ เนื่องจากเป็นการจำลองโปรแกรมมาไว้ให้อยู่ในรูปที่เข้าใจได้ง่ายขึ้น จากคลาสไดอะแกรมของซอฟท์แวร์ที่เราทำการออกแบบจะแบ่งออกเป็นคลาสหลัก ๆ เช่น Class System (main class), Class File Manager, Class Encrypt Factory และคลาสอื่น ๆ อีก ซึ่งจากคลาสไดอะแกรมจะเห็นว่าทุกคลาส จะทำงานผ่านคลาสเมนทุกคลาสไป ในแต่ละคลาสจะแสดงให้เห็นถึงค่าตัวแปรต่าง ๆ ในคลาสนั้น ๆ รวมทั้งยังแสดงให้เห็นถึงเมดธอดในการทำงานในคลาสนั้น ๆ อีกด้วย ตัวอย่างเช่น Class System จะมีตัวแปรที่ใช้ภายในคลาสคือ User Name, Algorithm และมีเมธอดในการทำงานของคลาสคือ Start Server(), Connect Server(), Shutdown(), Encrypt() เป็นต้น

3.2 การออกแบบฮาร์ดแวร์ (Hardware Design)

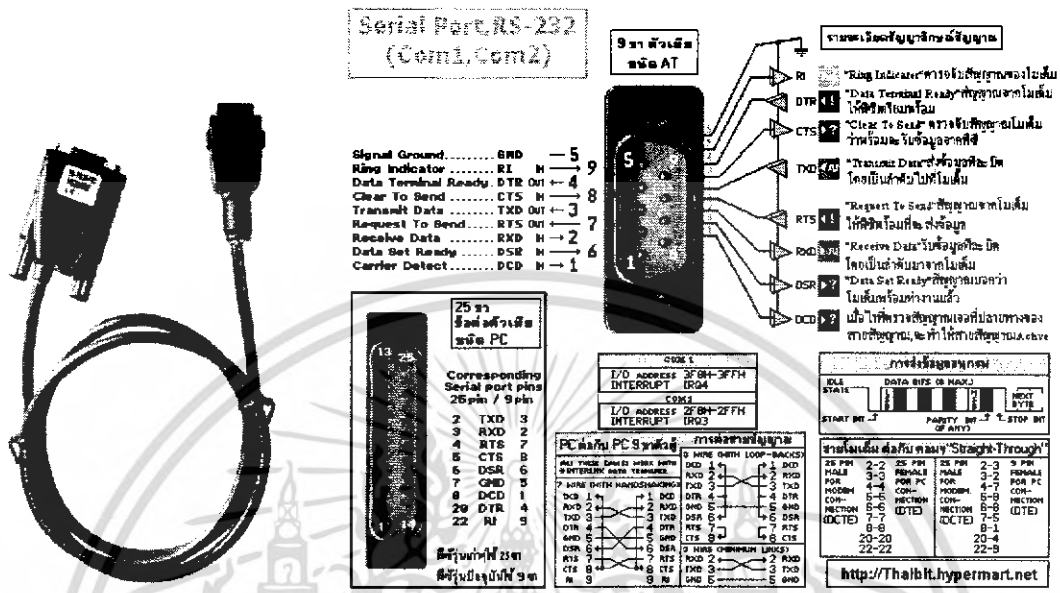
การออกแบบในส่วนของฮาร์ดแวร์ที่ใช้ในการพัฒนาซอฟท์แวร์จำลองการทำงานของรหัสลับเชิงควอนตัมมีดังนี้

3.2.1 Quantum Random Generation เป็นอุปกรณ์ทางฮาร์ดแวร์ที่ทำหน้าที่สำหรับการสุ่มคีย์ที่เป็นการสุ่ม 100 เปอร์เซ็นต์ ซึ่งอยู่ในขั้นตอนการพัฒนาโดยกลุ่มพัฒนาโครงงานของ NECTEC ซึ่งในส่วนของการพัฒนาด้านซอฟท์แวร์จะเลือกใช้การสุ่มจากคอมพิวเตอร์ โดยการเขียนโปรแกรมสร้าง Class Random Generator มาแทนก่อนเพื่อใช้ในการทดสอบซอฟท์แวร์

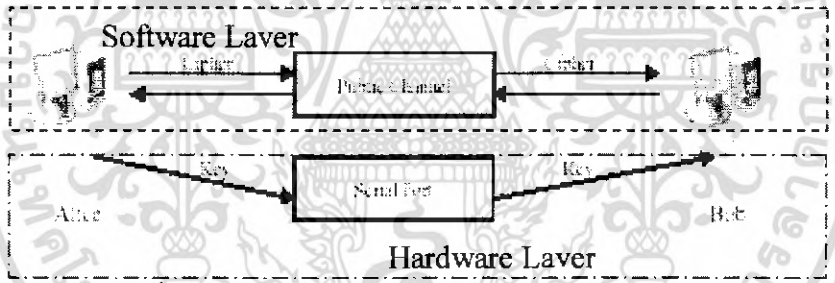
3.2.2 Quantum Channel เป็นช่องสัญญาณพิเศษที่ใช้ในการส่งข้อมูลคีย์ ซึ่งแยกออกจากช่องสัญญาณที่ทำการติดต่อสื่อสารโดยทั่วไป โดยในระบบวิทยาการรหัสลับเชิงควอนตัม จะใช้สายไฟเบอร์ออปติก ซึ่งใช้วิธีการส่งโดยใช้แสงเป็นตัวกลางซึ่งอาศัยหลักการทางควอนตัมของแสงมาช่วยเสริมสร้างความปลอดภัย แต่ในการพัฒนาซอฟท์แวร์เราจำเป็นต้องทดสอบการทำงานของซอฟท์แวร์จึงจำเป็นต้องหาอุปกรณ์ที่หาได้ง่ายและใช้งานได้ง่ายกว่ามาใช้ในการทดสอบ จึงทำการแทนช่องสัญญาณควอนตัมโดยการใช้สาย Serial Port เป็นช่องสัญญาณที่ใช้ส่งคีย์แทนในการทดสอบซอฟท์แวร์

ในส่วนของช่องสัญญาณควอนตัมซึ่งในโครงงานนี้จะเลือกใช้การส่งข้อมูลผ่านสาย Serial Port แทนการส่งผ่านทางแสง เนื่องจาก Serial Port เป็นอุปกรณ์ที่หาง่ายและสามารถนำมาประยุกต์ใช้ได้ง่ายกว่าอุปกรณ์ส่งทางแสง ซึ่งการจำลองนี้จะจำลองให้เห็นถึงการส่งข้อมูลที่มีการส่งข้อมูลในต่างช่องทางของระบบ แต่สาย Serial Port นี้จะไม่สามารถจำลองคุณสมบัติของช่องสัญญาณควอนตัมได้ แต่ระบบจะมีโปรโตคอลในการตรวจสอบความปลอดภัย ที่ช่วยให้ระบบ

การส่งข้อมูลมีความปลอดภัยเช่นเดิม (จำลองการเชื่อมต่อ Quantum Channel ซึ่งปรกติใช้ Fiber Optic มาใช้ RS-232 สำหรับการแลกเปลี่ยน คีย์ระหว่างคอมพิวเตอร์ทั้ง 2 ฝั่ง)



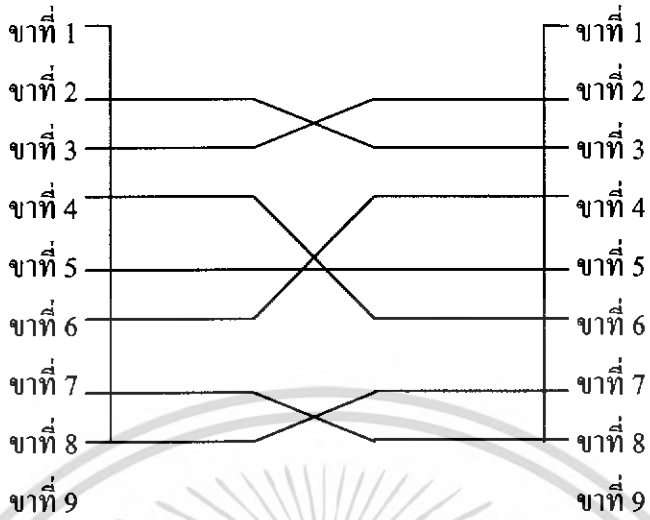
รูปที่ 3.6 แสดงรายละเอียดของสาย Serial Port



รูปที่ 3.7 แสดงการใช้ RS-232 ประยุกต์ใช้ในโครงงาน

หน้าที่ของ RS-232 ในซอฟท์แวร์จำลองการทำงานของรหัสลับเชิงควอนตัมนี้ คือเป็นตัวกลางในการรับส่งข้อมูล กุญแจไขความลับ ซึ่งจะทำก่อนที่จะทำการเข้ารหัส ซึ่งก่อนเข้ารหัสจะต้องผ่านกระบวนการตรวจสอบความปลอดภัยด้วยโปรโตคอล BB84 ก่อน หากปลอดภัยจึงจะได้ใช้ กุญแจไขความลับนี้ในการเข้ารหัสข้อมูล

ในการนำสาย Serial Port (Serial Port 9 Pin) มาใช้งานในระบบเพื่อใช้ในการส่งผ่าน “กุญแจไขความลับ” นั้นเป็นการติดต่อระหว่าง Serial Port ระหว่างคอมพิวเตอร์ 2 เครื่อง จึงจำเป็นต้องมีการตัดแปลงสาย Serial Port ทำการสลับขาการเชื่อมต่อบางขา เพื่อให้สามารถส่งข้อมูลระหว่างกันได้ หรือที่เรียกว่าการคลอสสาย ซึ่งในการสลับขาเพื่อการสื่อสารข้อมูลนั้นจะทำการสลับขาดังรูป 3.6



รูปที่ 3.6 แสดงการปรับปรับแต่งสาย Serial Port เพื่อใช้ในการสื่อสาร

ในการเขียนโปรแกรมติดต่อกันระหว่าง Computer ผ่าน Serial Port นั้นต้องใช้ Microsoft Communication Control ของ VB6.0 ซึ่งได้มีการออกแบบ Message ที่สามารถติดต่อผ่าน RS-232 ได้ดังนี้

รูปที่ 3.7 แสดงการติดต่อระหว่างคอมพิวเตอร์ผ่าน Serial Port

ในส่วนของช่องสัญญาณควอนตัม หรือการออกแบบการส่งข้อมูลผ่าน Serial Port นอกเหนือจากการส่งข้อมูลผ่านสาย RS-232 แล้วยังทำการทำการจำลองของลักษณะของโฟตอนที่ใช้ส่งผ่านช่องสัญญาณควอนตัมซึ่งมีลักษณะและรูปแบบไปตามบิตข้อมูล โดยการใช้การเปลี่ยนแปลงของบิตข้อมูล บิต 0 หรือ 1 ให้ตรงกับหลักทฤษฎีของการโพลาไรซ์ของแสงตามโปรโตคอล BB 84 โดยทำการเขียนโปรแกรมแสดงกราฟฟิกเพื่อใช้แสดงการโพลาไรซ์ของแสงไปตามบิตข้อมูล

ในการแสดงลักษณะของบิตข้อมูลในรูปของการโพลาไรซ์ของแสงนั้น จะต้องมีข้อกำหนดในการใช้งาน ซึ่งจะต้องกำหนดลักษณะไปตามเวคเตอร์ฐานที่ใช้ ซึ่งมีเวคเตอร์ฐานอยู่ 2 ลักษณะคือ

The rectilinear basis ใช้สัญลักษณ์ + ↗ ↔↕↖

The diagonal basis ใช้สัญลักษณ์ ×

เวคเตอร์ฐานสองชนิดนี้จะเป็นตัวกำหนดทิศทางของการโพลาไรซ์ของแสง ในแต่ละบิตข้อมูล โดยมีการออกแบบและกำหนดลักษณะไว้ดังตารางต่อไปนี้

บิตข้อมูล (Input)	เวคเตอร์ฐาน	สถานะของบิตข้อมูลที่ใช้ในการส่ง
0	+	→
1	+	↑
0	×	↖
1	×	↗

ตารางที่ 3.2 แสดงการออกแบบสถานะของบิตข้อมูล

จากตารางข้างต้นแสดงให้เห็นว่าในการส่งข้อมูล ทั้งฝั่งผู้รับและฝั่งผู้ส่งจำเป็นต้องใช้ตารางข้อมูลนี้ในการเข้ารหัสและถอดรหัสข้อมูล ซึ่งสถานะที่ได้ทำการจำลองขึ้นนี้ ได้ทำการจำลองตามทฤษฎีที่ใช้จริงในระบบวิทยาการรหัสลับเชิงควอนตัม

การออกแบบแพคเกจ bb84

Header	Data	Trailer

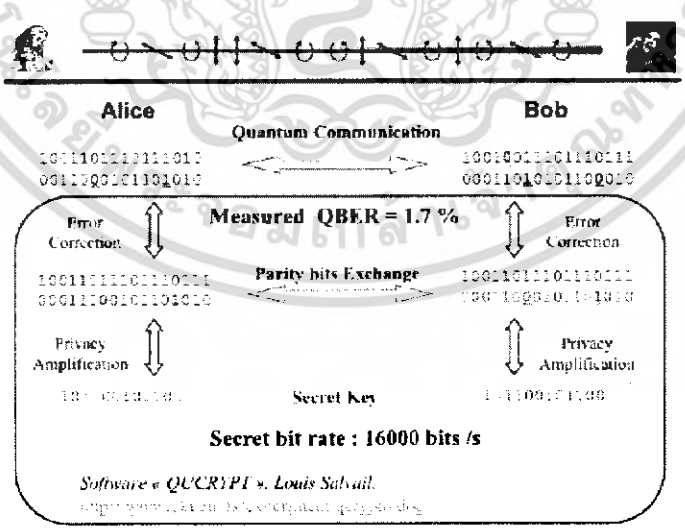
Header: รหัสคำสั่งในการทำงานของ โปรโตคอล

Data : ข้อมูล

Trailer : รหัสปิดท้ายคำสั่ง

ชื่อคำสั่ง	คำอธิบาย
quantum_ECHO	ตรวจสอบการเชื่อมต่อในควอนตัมแชนแนล
quantum_PRESET	ส่งข้อมูลให้ผู้รับเตรียมพร้อมรับข้อมูลคีย์
quantum_READY	ส่งข้อมูลให้ผู้ส่งว่าพร้อมรับคีย์แล้ว
quantum_KEYSTREAM	ส่งข้อมูลคีย์ทีละ 8 บิต
quantum_KEYRECEIVE	ส่งข้อมูลให้ผู้ส่งว่าได้รับคีย์แล้ว
quantum_TERMINATE	ส่งข้อมูลว่ายกเลิกการส่งข้อมูลคีย์
packet_BB84VectorVerifyRequest	ผู้ส่งร้องขอการตรวจสอบข้อมูลเวคเตอร์
packet_BB84VectorVerifyReply	ผู้รับส่งเวคเตอร์ตอบกลับ
packet_BB84VectorVerifyResult	ผู้ส่งตรวจสอบเวคเตอร์และแจ้งผลไปยังผู้รับ
packet_BB84TrapVerifyRequest	ผู้ส่งร้องขอการตรวจสอบความปลอดภัย
packet_BB84TrapVerifyStream	ผู้รับส่งข้อมูลคีย์บางส่วนให้ตรวจสอบ
packet_BB84TrapVerifyReply	ผู้ส่งตรวจสอบและส่งข้อมูลการดักจับให้ผู้รับ
packet_BB84FinalKeyRequest	ผู้รับขอตรวจสอบความถูกต้องของบิต
packet_BB84FinalKeyReply	ผู้ส่งสร้างโค้ดตรวจสอบพาริตี้บิต
packet_BB84FinalKeyCommit	ผู้รับทำการแก้บิตผิดและยอมรับคีย์

ตารางที่ 3.3 ตารางแสดงการออกแบบPacket



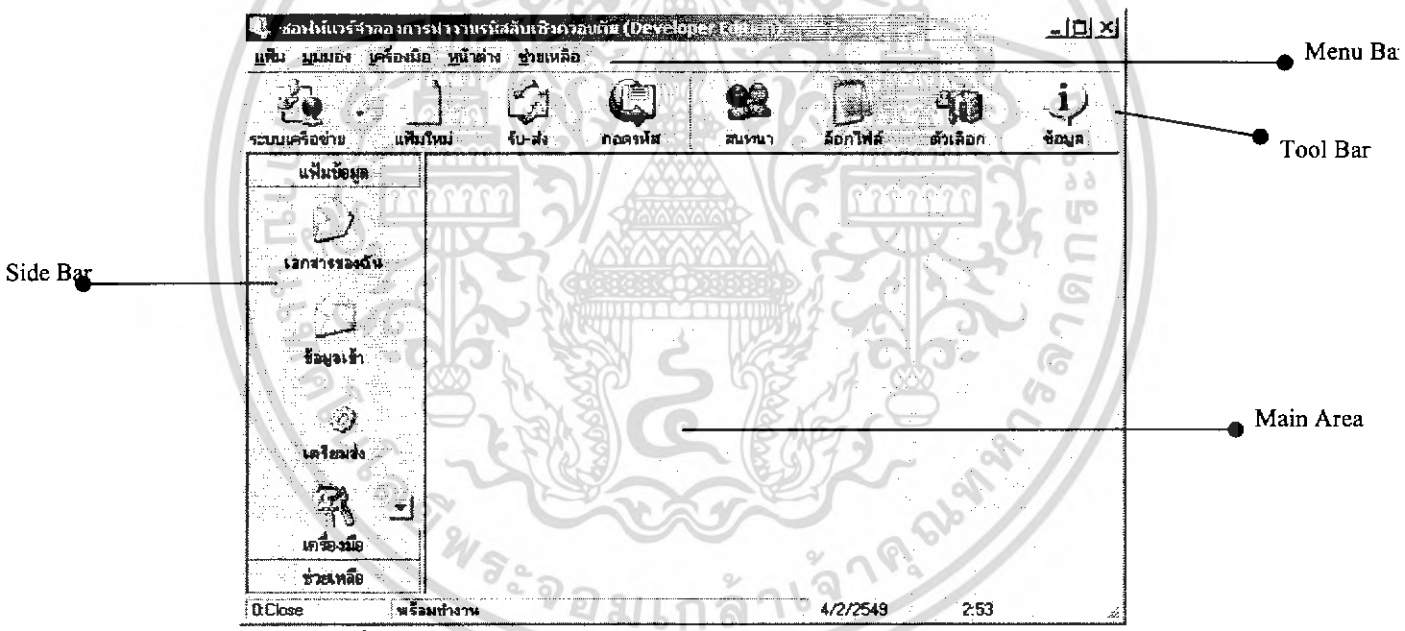
รูปที่ 3.8 แสดงการทำงานของโปรโตคอล BB84

3.2.3 Public Channel เป็นการออกแบบการติดต่อสื่อสารผ่านช่องสัญญาณสาธารณะ ซึ่งใช้สำหรับการติดต่อสื่อสารทั่วไป ส่ง Cipher และอื่น ๆ ซึ่งจะใช้ระบบ Computer Network ในการติดต่อสื่อสาร ซึ่งสามารถใช้ได้ทั้งในระบบ LAN หรือ Internet ก็ได้ ซึ่งใน VB 6.0 นี้ เราจะใช้ Winsock Control ในการติดต่อ

3.3 การออกแบบ GUI (Graphic User Interface Design)

การออกแบบ Graphic User Interface นั้นก็เพื่อให้ได้โปรแกรมที่มีความน่าใช้งาน มีความง่าย และสามารถทำความเข้าใจในโปรแกรมได้ไม่ยากนัก ในส่วนนี้ได้มีการออกแบบให้มีรูปร่างของโปรแกรมที่มี Interface ที่น่าใช้และสามารถเข้าใจได้ไม่ยากนัก เน้นความง่ายในการใช้งาน

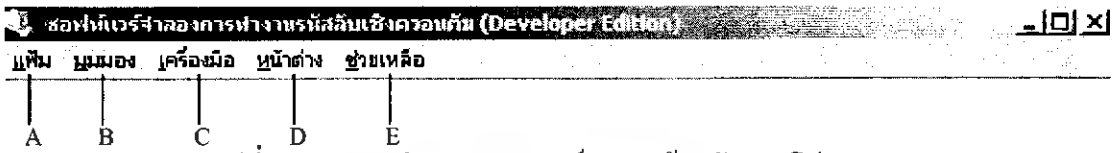
3.3.1 การออกแบบหน้าจอการทำงานหลักของโปรแกรม



รูปที่ 3.9 แสดงการออกแบบหน้าจอการทำงานหลักของโปรแกรม

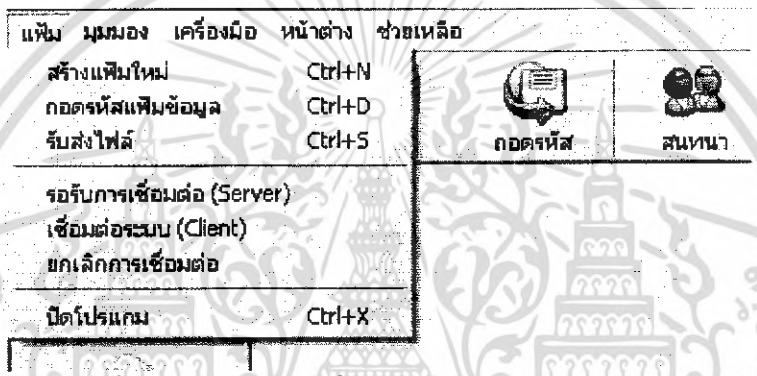
จากรูปที่ 3.8 ได้แสดงให้เห็นหน้าตาของหน้าจอหลักของโปรแกรมที่ได้ทำการออกแบบ ซึ่งจะแบ่งส่วนประกอบของหน้าจอหลักเป็น 4 ส่วนดังนี้

3.3.1.1 เมนูบาร์ (Menu Bar) เป็นเมนูที่ใช้ในการควบคุมการทำงานทั้งหมดของโปรแกรม



รูปที่ 3.10 แสดงส่วนของเมนูบาร์ ของหน้าหลักของโปรแกรม

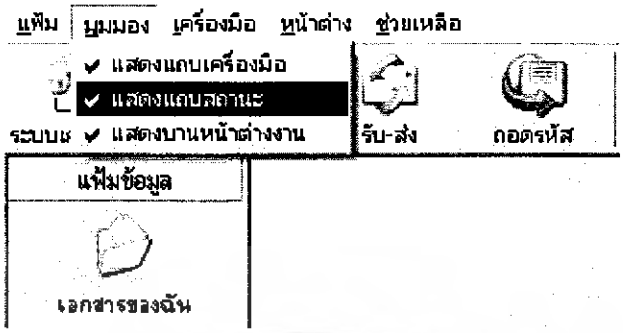
A: แฟ้ม เป็นเมนูควบคุมการทำงานหลัก ๆ ของโปรแกรม



รูปที่ 3.11 แสดงเมนูคำสั่งในส่วนของ แฟ้ม ในเมนูบาร์

- สร้างแฟ้มใหม่ เป็นการสร้างข้อมูลที่จะทำการเข้ารหัสใหม่
- ถอดรหัสแฟ้มข้อมูล เป็นการเลือกไฟล์เพื่อทำการถอดรหัส
- รับส่งไฟล์ เป็นคำสั่งเข้าสู่การรับส่งไฟล์ ใน Public Channel
- ยกเลิกการเชื่อมต่อ เป็นคำสั่งยกเลิกการเชื่อมต่อระบบ ทั้งใน Serial Port และ Public Channel
- ปิดโปรแกรม เป็นคำสั่งปิดโปรแกรมทันที

B: มุมมอง เป็นเมนูควบคุม View แสดงหน้าจอ



รูปที่ 3.12 แสดงเมนูคำสั่งในส่วนของ มุมมอง ในเมนูบาร์

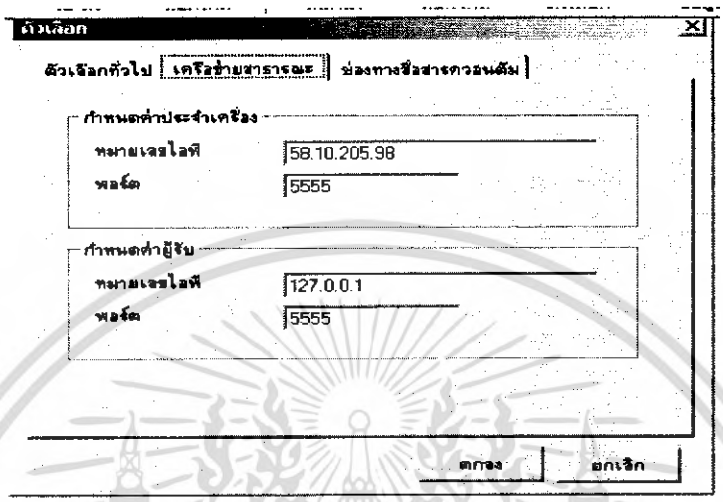
- แสดงแถบเครื่องมือ เป็นคำสั่งเพื่อเปิด-ปิดแถบเครื่องมือของโปรแกรม
- แสดงแถบสถานะ เป็นคำสั่งเพื่อเปิด-ปิดแถบสถานะของโปรแกรม
- แสดงบานหน้าต่างงาน เป็นคำสั่งเพื่อเปิด-ปิด Side Bar

C: เครื่องมือ เป็นเมนูที่ใช้ในการเลือกที่จะเปิดปิดหน้าต่างการทำงานต่าง ๆ ของโปรแกรม

รูปที่ 3.13 แสดงเมนูคำสั่งในส่วนของ เครื่องมือ ในเมนูบาร์

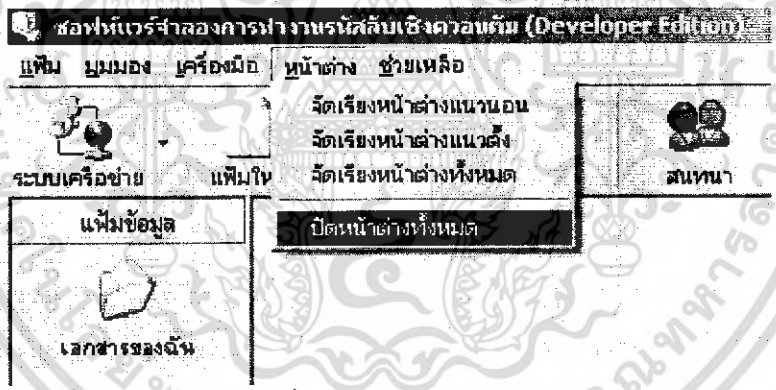
- หน้าต่างสนทนา เป็นเมนูคำสั่งเพื่อเปิดหน้าจอสนทนา เพื่อนสนทนากับผู้ที่ทำการติดต่ออยู่
- หน้าต่างจัดการไฟล์ จะเป็นเมนูคำสั่งเพื่อเข้าสู่ส่วนของการจัดเก็บข้อมูลซึ่งมีทั้งกล่องข้อมูลขาเข้า ขาออก กุญแจ ถังขยะ เป็นต้น
- หน้าต่างแสดงบันทึกการทำงาน เป็นคำสั่งเปิดหน้าจอแสดงบันทึกการทำงานทั้งหมดของโปรแกรม
- ตรวจสอบระบบ เป็นเมนูคำสั่งเพื่อเปิดหน้าจอการตรวจสอบการเชื่อมต่อของโปรแกรม ทั้งใน Public Channel และ RS-232

- ตัวเลือก เป็นคำสั่งเพื่อเปิดหน้าจอการกำหนดค่าที่สำคัญให้แก่ระบบ ทั้งในส่วนของ ข้อมูลตัวเลือกทั่วไป ข้อมูลเครือข่ายสาธารณะ และช่องสัญญาณควอนตัม



รูปที่ 3.14 แสดงหน้าจอคำสั่ง ตัวเลือก ในส่วนของเมนูบาร์ (เครื่องมือ)

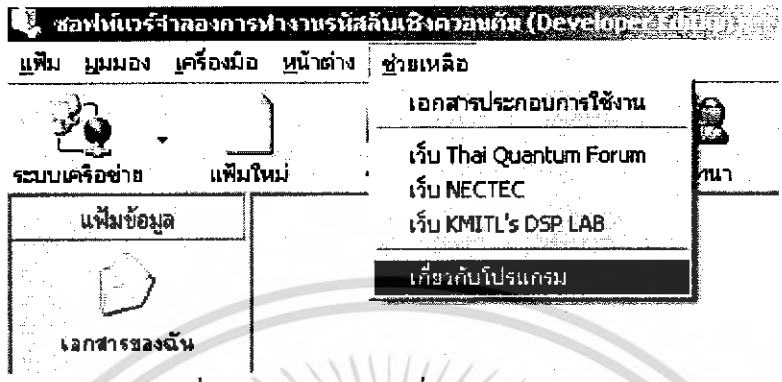
D: หน้าต่าง เป็นเมนูที่ใช้ควบคุมการจัดเรียงหน้าต่าง



รูปที่ 3.15 แสดงเมนูคำสั่งในการจัดเรียงหน้าต่างการแสดงข้อมูล

- จัดเรียงหน้าต่างแนวนอน เป็นคำสั่งในการจัดเรียงหน้าต่างในแนวนอน
- จัดเรียงหน้าด่างแนวตั้ง เป็นคำสั่งในการจัดเรียงหน้าด่างในแนวตั้ง
- จัดเรียงหน้าต่างทั้งหมด เป็นคำสั่งในการจัดเรียงหน้าต่างในแนวซ้อนทับกัน
- ปิดหน้าต่างทั้งหมด เป็นคำสั่งในการปิดหน้าต่างการทำงานของโปรแกรมทั้งหมด

E: ช่วยเหลือ เป็นเมนูที่ให้ความช่วยเหลือในการใช้งานโปรแกรม รวมทั้งลิงค์ไปยังแหล่งข้อมูลที่เกี่ยวข้องกับระบบ



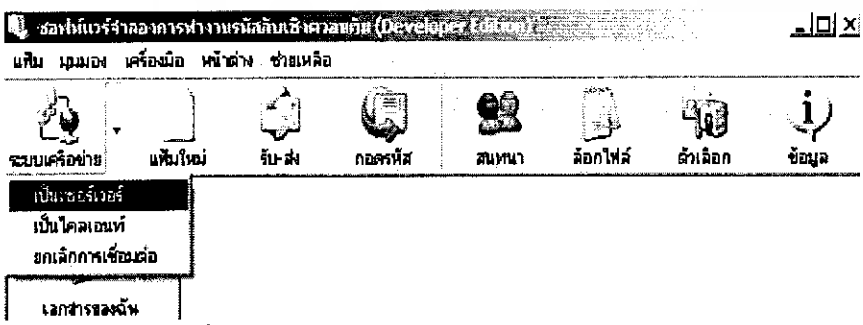
รูปที่ 3.16 แสดงเมนูคำสั่งในเมนู ช่วยเหลือ

- เอกสารประกอบการใช้งาน เป็นชุดคำสั่งเปิดเอกสารคู่มือการใช้งานโปรแกรม
- เวป Thai Quantum Forum, NECTEC, KMITL's DSP LAB เป็นคำสั่งในการเปิดเวปอ้างอิงถึงแหล่งข้อมูลสนับสนุนโครงงาน
- เกี่ยวกับโปรแกรม จะแสดงรายละเอียดในการพัฒนาโปรแกรม

3.3.1.2 ทูลบาร์ (Tool Bar) เป็นปุ่มที่ใช้ควบคุมการทำงานของโปรแกรม ซึ่งช่วยให้สามารถเรียกใช้คำสั่งต่าง ๆ ได้อย่างสะดวกและรวดเร็ว โดยเพียงแค่คลิกเมาส์ที่ปุ่มเท่านั้น โดยปุ่มต่าง ๆ จะมีหน้าที่ดังต่อไปนี้

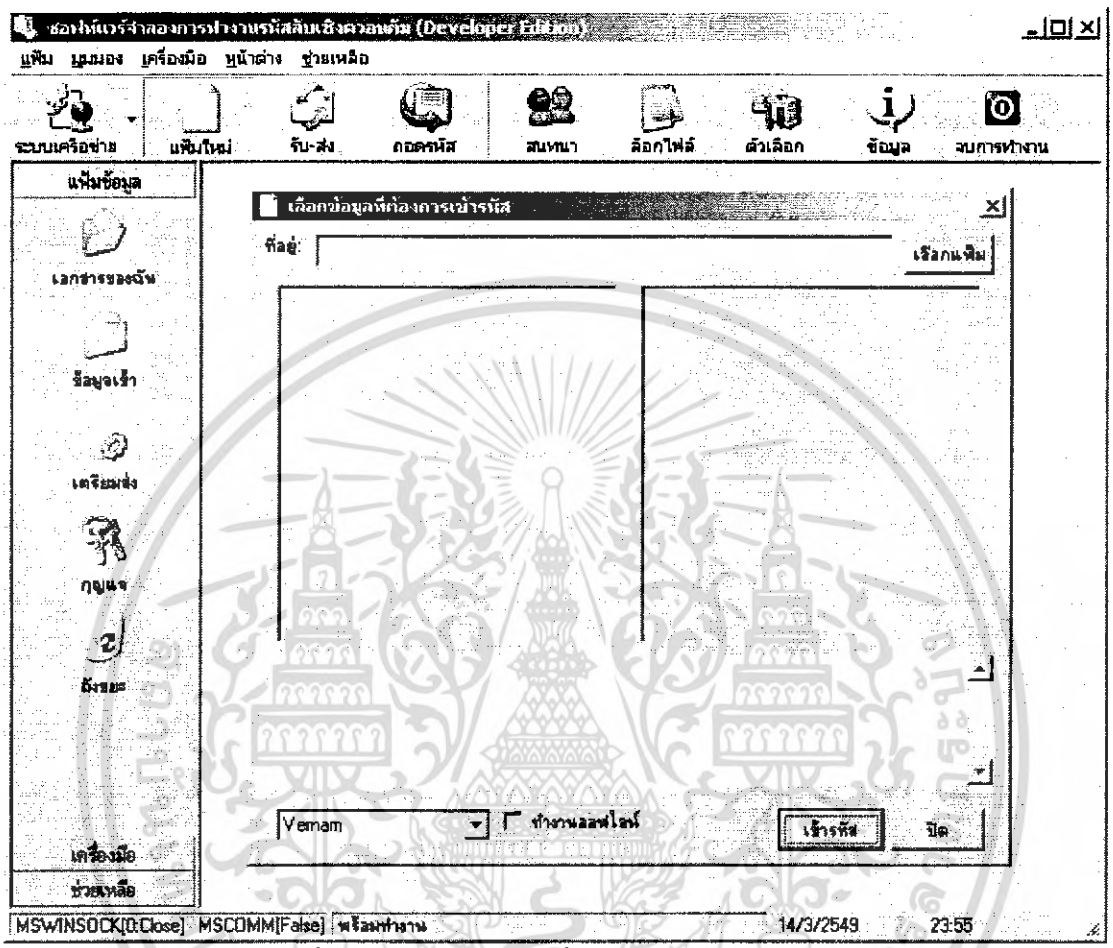
รูปที่ 3.17 แสดงส่วนของทูลบาร์

A: ระบบเครือข่าย เป็นส่วนของการเข้าสู่การเชื่อมต่อระบบและกำหนดว่าจะเชื่อมต่อโดยเป็น Client หรือ Server และมีคำสั่งยกเลิกการเชื่อมต่อได้



รูปที่ 3.18 แสดงส่วนของทูลบาร์ ระบบเครือข่าย

B: แฟ้มใหม่ เป็นปุ่มที่เข้าถึงคำสั่งการเริ่มต้นเข้ารหัสไฟล์ โดนจะมีหน้าจอให้เลือกไฟล์เพื่อเข้ารหัส



รูปที่ 3.19 แสดงหน้าต่างเมื่อเลือกทูลบาร์ แฟ้มใหม่

C: รับ-ส่ง เป็นปุ่มที่เข้าถึงคำสั่งส่งไฟล์ เป็นเมนูลัดเข้าสู่การรับส่งไฟล์ โดยจะมีหน้าต่างเลือกไฟล์ที่จะส่งขึ้นมา

D: ถอดรหัส เป็นปุ่มคำสั่ง ที่เข้าถึงคำสั่งเรียกหน้าต่างการถอดรหัส โดยจะปรากฏหน้าจอที่ให้เลือกไฟล์ เพื่อที่จะถอดรหัสขึ้นมา

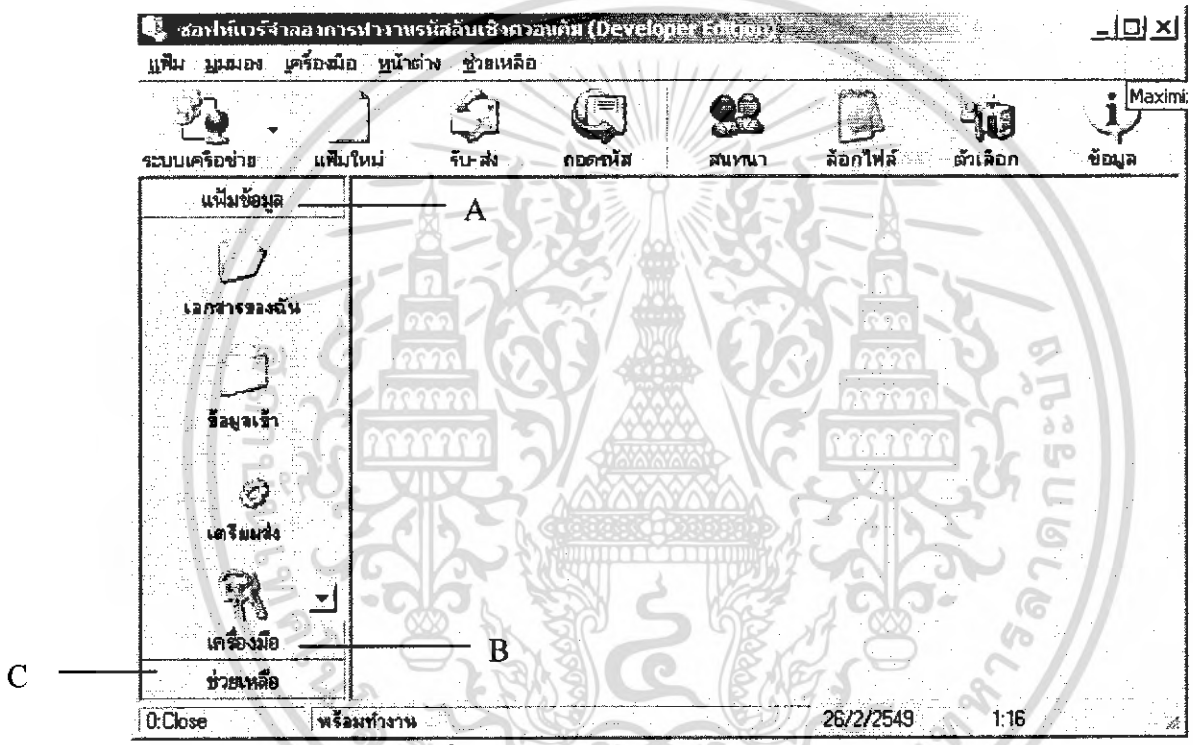
E: สนทนา เป็นปุ่มคำสั่งที่เรียกหน้าจอสนทนาระหว่าง Client กับ Server ซึ่งจะปรากฏหน้าจอการสนทนาดังรูป

F: ล็อกไฟล์ เป็นปุ่มคำสั่งที่เรียกหน้าจอที่จะแสดงบันทึกการทำงานออกมา

G: ตัวเลือก เป็นปุ่มคำสั่งเพื่อเข้าสู่การกำหนดค่าพื้นฐานในการติดต่อสื่อสารระหว่างเครื่องแม่ข่ายและเครื่องลูกข่ายที่ทำการติดต่อกัน ซึ่งค่านี้จะกำหนดก่อนการเชื่อมต่อ หากกำหนดไม่ถูกต้องการเชื่อมต่อก็จะไม่สำเร็จได้

H: ข้อมูล เป็นส่วนที่จะแสดงข้อมูลเกี่ยวกับโปรแกรม เช่น ผู้พัฒนา เวอร์ชั่นของโปรแกรม เป็นต้น

3.3.1.3 Side Bar เป็นส่วนของโปรแกรมที่เป็นเมนูอำนวยความสะดวกในการใช้งานแก่ผู้ใช้ ซึ่งได้ออกแบบให้มีความแปลกใหม่ และน่าใช้งานมากขึ้น เมนูที่อยู่ในส่วนของ Side Bar ยกตัวอย่างเช่น แฟ้มข้อมูลที่ใช้ในการเก็บไฟล์ต่าง ๆ ที่ใช้ในระบบ ข้อมูลช่วยเหลือ และเครื่องมือในการทำงานอื่น ๆ ของโปรแกรมอีกด้วย



รูปที่ 3.20 แสดงรายละเอียดของส่วน Side Bar

A: Side Bar แฟ้มข้อมูล เป็นส่วนของเมนูลัดเข้าสู่แฟ้มข้อมูลประเภทต่าง ๆ ซึ่งสามารถเข้าถึงแฟ้มได้เพียงกดปุ่มที่กำหนดไว้

รูปที่ 3.21 แสดงรายละเอียดในส่วนของ Side Bar แทปแฟ้มข้อมูล

- เอกสารของฉัน เป็นแฟ้มข้อมูลเก็บไฟล์ต้นฉบับที่ใช้ในการทำงานของระบบ

- ข้อมูลเข้า เป็นแฟ้มข้อมูลที่เก็บไฟล์ที่ได้รับจากผู้ที่กำลังเชื่อมต่อระบบด้วย ซึ่งจะเป็นไฟล์ที่ทำการเข้ารหัสอยู่ใช้งานไม่ได้

- เตรียมส่ง เป็นแฟ้มข้อมูลที่เก็บไฟล์ที่ได้ทำการเข้ารหัสไว้แล้ว พร้อมที่จะทำการส่งให้ผู้ที่กำลังเชื่อมต่อระบบด้วย

- กุญแจ เป็นแฟ้มเก็บข้อมูลไฟล์ กุญแจไขความลับ (Key) ที่จะใช้ในการถอดรหัสไฟล์ เพื่อนำไฟล์ข้อมูลไปใช้งานต่อไป

- ถังขยะ เป็นแฟ้มข้อมูลที่เก็บไฟล์ต่าง ๆ ที่ได้ทำการลบทิ้งจากโปรแกรมไปแล้ว

B: Side Bar เครื่องมือ เป็นแถบเมนูลัดเพื่อเข้าสู่หน้าจอการทำงานที่ใช้เป็นประจำ เพื่อความสะดวกในการเข้าถึงการทำงาน

รูปที่ 3.22 แสดงรายละเอียดในส่วนของ Side Bar แทบเครื่องมือ

- สนทนา เป็นเมนูคำสั่งเข้าสู่หน้าจอการสนทนากับผู้ที่กำลังเชื่อมต่อระบบกันอยู่ ซึ่งต้องทำการเชื่อมต่อก่อนถึงจะสามารถใช้งานการสนทนาได้

- บันทึกการทำงาน เป็นเมนูคำสั่งเข้าสู่หน้าจอบันทึกการทำงานของโปรแกรม

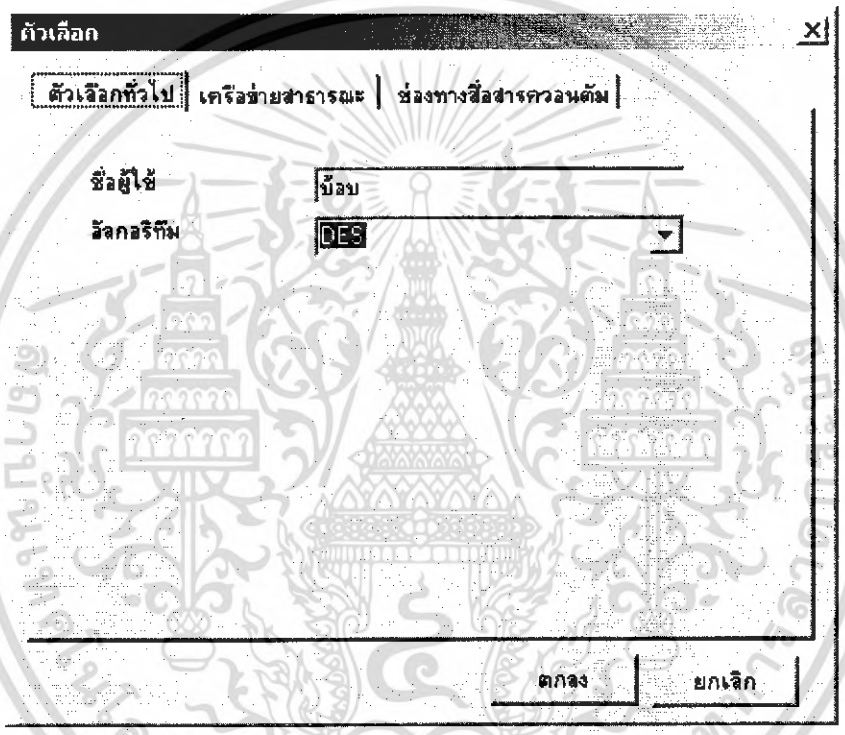
- ตรวจสอบระบบ เป็นเมนูคำสั่งเพื่อสั่งให้โปรแกรมทำการตรวจสอบการเชื่อมต่อระบบ ทั้งช่องสัญญาณสาธารณะ และช่องสัญญาณพิเศษ (RS-232)

C: Side Bar ช่วยเหลือ เป็นแถบเมนูลัดเพื่อลิงค์เข้าสู่แหล่งข้อมูลต่าง ๆ ที่เกี่ยวข้องกับวิทยาการรหัสลับเชิงควอนตัมของประเทศไทย ซึ่งสามารถหาข้อมูลเพิ่มเติม และศึกษาทำความเข้าใจในระบบ อีกทั้งความคืบหน้าในการพัฒนาระบบที่กำลังพัฒนาอยู่

3.3.2 การออกแบบหน้าจอการกำหนดค่าการเชื่อมต่อให้ระบบ (ตัวเลือก)

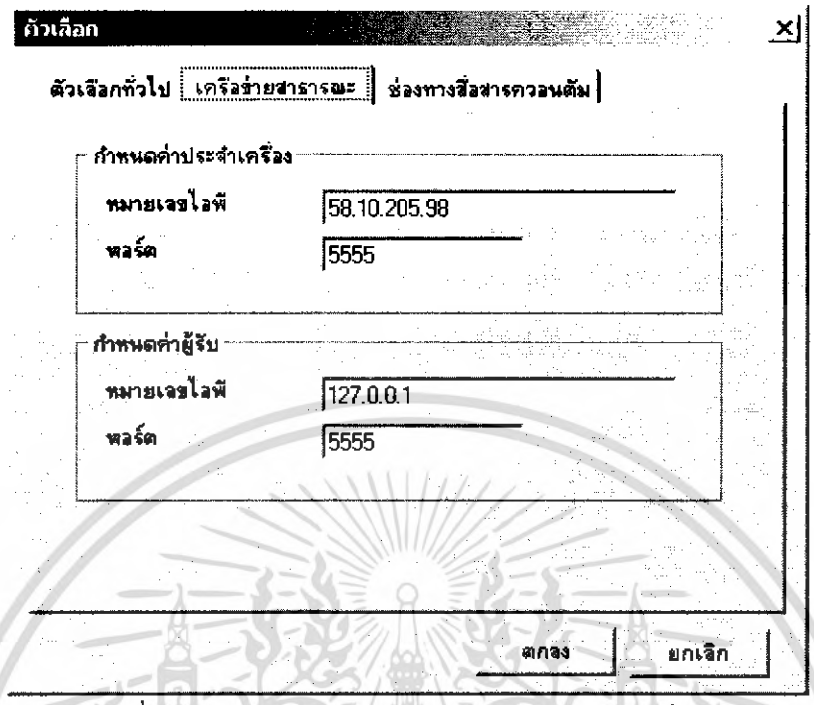
ในส่วนของการกำหนดค่าต่าง ๆ ให้กับระบบ ได้มีการออกแบบการเซตค่าต่าง ๆ ไว้สามหน้าจอ ซึ่งผู้ใช้สามารถเลือกที่จะกำหนดค่าได้ตามการใช้งานดังนี้

3.3.2.1 ตัวเลือกทั่วไป เป็นการกำหนดค่าพื้นฐานในการเชื่อมต่อเกี่ยวกับผู้ใช้งาน ได้แก่ การกำหนดชื่อที่ใช้งาน และกำหลดอัลกอริทึมพื้นฐานที่เลือกใช้งานในระบบ ดังรูปที่ 3.23



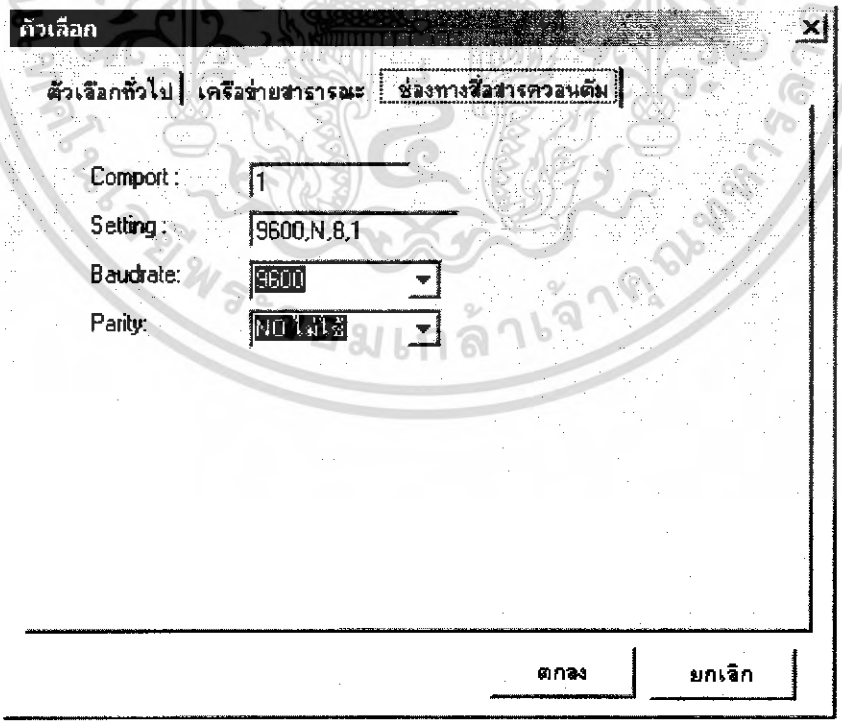
รูปที่ 3.23 แสดงหน้าจอการกำหนดค่าตัวเลือกในแท็บ ตัวเลือกทั่วไป

3.3.2.2 เครือข่ายสาธารณะ เป็นการกำหนดค่าเพื่อกำหนดเส้นทางการเชื่อมต่อทางช่องสัญญาณสาธารณะ ได้แก่การกำหนด หมายเลขไอพี และหมายเลขพอร์ตของการเชื่อมต่อทั้งเครื่องแม่ขาย และลูกข่าย ดังรูป 3.24



รูปที่ 3.24 แสดงหน้าจอการกำหนดค่าตัวเลือกในแท็บ เครือข่ายสาธารณะ

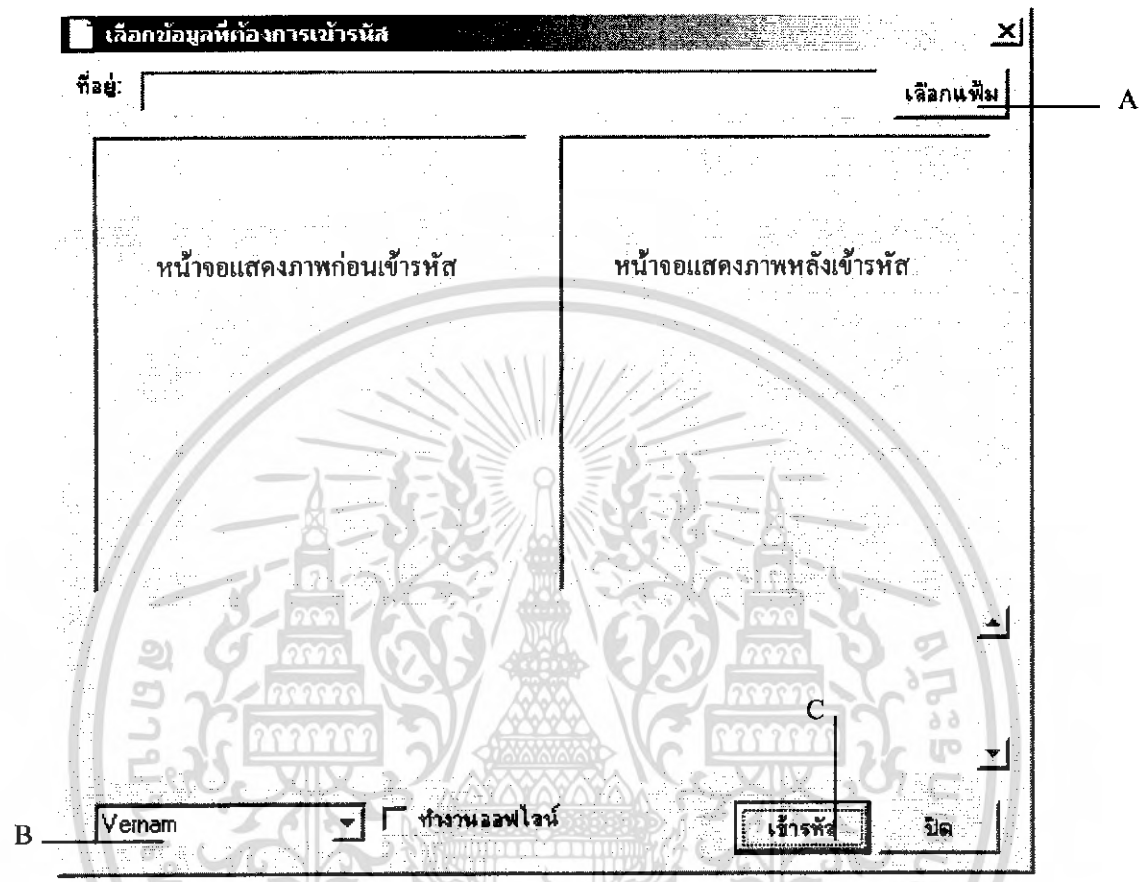
3.3.2.3 ทางสื่อสารควอนตัม เป็นการกำหนดค่าเพื่อใช้ในการเชื่อมต่อในช่องสัญญาณพิเศษ ซึ่งในโปรแกรมนี้เป็นการกำหนดค่าการเชื่อมต่อผ่าน Serial Port ซึ่งมีรายละเอียดดังรูป 3.25



รูปที่ 3.25 แสดงหน้าจอการกำหนดค่าตัวเลือกในแท็บ ช่องทางสื่อสารควอนตัม

3.3.3 การออกแบบหน้าจอการทำงานอื่น ๆ ที่ใช้งานประจำของโปรแกรม

3.3.3.1 หน้าต่างการทำงานของการเข้ารหัสไฟล์ใหม่



รูปที่ 3.26 แสดงหน้าต่างการเข้ารหัสข้อมูล

- เริ่มต้นด้วยการเลือกที่ เลือกแฟ้ม (A) เพื่อเลือกไฟล์ที่ต้องการเข้ารหัส
- ทำการเลือก Algorithm (B)ที่ใช้ในการเข้ารหัส ซึ่งในโปรแกรมได้มีตัวเลือกในส่วนของอัลกอริทึม 3 ตัว ได้แก่ Vernam, Blowfish และ DES
- ทำการเลือก เข้ารหัส (C) เพื่อทำการเข้ารหัสข้อมูล ไฟล์ที่เข้ารหัสแล้วจะเข้าไปเก็บไว้ในแฟ้มข้อมูล เตรียมส่ง

3.3.3.2 หน้าต่างการถอดรหัสข้อมูล

รูปที่ 3.27 แสดงหน้าต่างการถอดรหัสข้อมูล

- เริ่มต้นด้วยการเลือกแฟ้ม (A) เพื่อทำการเลือกไฟล์รหัสลับที่ต้องการถอดรหัส
- เลือกกุญแจ (C) เพื่อทำการเลือกกุญแจที่ใช้ในการถอดรหัส
- เลือก อัลกอริทึม (B) เพื่อทำการเลือกอัลกอริทึมที่ถูกต้องในการถอดรหัส
- ถอดรหัส (D) เพื่อเริ่มการถอดรหัสข้อมูล

3.3.3.3 หน้าต่างการสนทนา

รูปที่ 3.28 แสดงหน้าจอการสนทนา

ในส่วนของการออกแบบหน้าต่างในการทำงานต่าง ๆ ที่ได้มาแสดงนั้นเป็นเพียงส่วนหนึ่งของโปรแกรมเท่านั้น ซึ่งหากต้องการเห็นภาพที่ละเอียดและครบถ้วนควรเปิดโปรแกรมแล้วลองทำการศึกษา จะเข้าใจมากขึ้น

ทั้งหมดนี้เป็นการออกแบบทั้งในส่วนของซอฟท์แวร์ ฮาร์ดแวร์ และ GUI ที่ได้แสดงให้เห็นมาข้างต้น ซึ่งได้ทำการเรียบเรียงจากหลักทางเทคนิคในการออกแบบมาเป็นรูปลักษณ์ที่สามารถเข้าใจได้ง่ายขึ้น

บทที่ 4
ผลการทดลอง

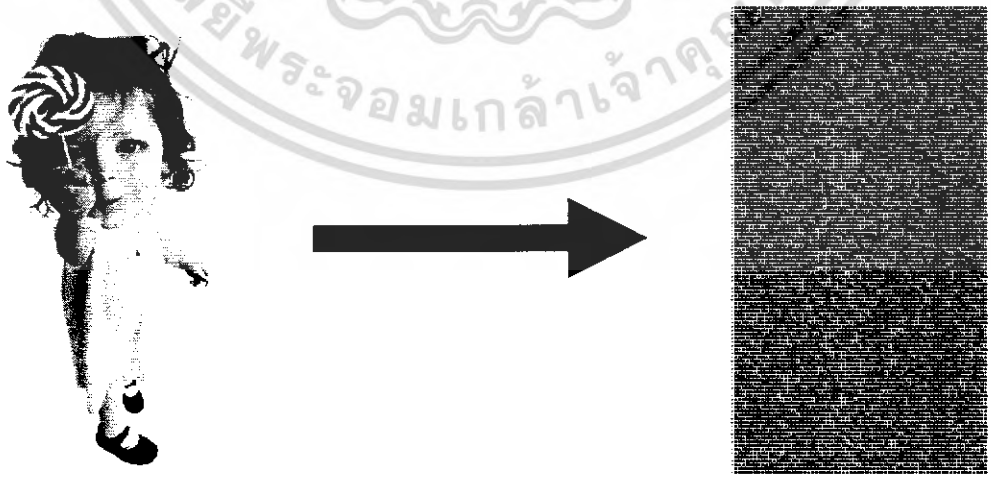
โครงงานพัฒนาซอฟท์แวร์จำลองการใช้รหัสลับเชิงควอนตัม เป็นโครงงานที่เป็นจุดเริ่มต้นของการพัฒนาด้านซอฟท์แวร์ในการพัฒนาระบบวิทยาการรหัสลับเชิงควอนตัม ซึ่งได้อธิบายรายละเอียดต่าง ๆ ไว้ก่อนหน้าเรียบร้อยแล้ว ในบทนี้จะกล่าวถึงเพียงผลการทดลองในการนำซอฟท์แวร์มาทดลองการทำงานด้านต่าง ๆ ซึ่งต้องตรงตามวัตถุประสงค์และการวางแผนการพัฒนาไว้ แต่รายละเอียดการใช้งานซอฟท์แวร์ จะมีในบทที่สามแล้ว

4.1 การเข้ารหัสและถอดรหัสแฟ้มข้อมูล

การเข้ารหัสแฟ้มข้อมูลและการถอดรหัสแฟ้มข้อมูลเป็นความสามารถหลักในการทำงานของซอฟท์แวร์ที่ได้พัฒนาขึ้น ซึ่งได้มีการออกแบบให้มีการใช้งานได้ 3 อัลกอริทึม ได้แก่ DES, Blowfish และ Vernam ในการใช้งานผู้ใช้สามารถเลือกอัลกอริทึมที่ต้องการใช้งานได้ และในการเข้ารหัสและถอดรหัสนั้นจะต้องทำการเลือกใช้อัลกอริทึมที่ถูกต้องตรงกันทั้งผู้ส่งและผู้รับเสมอ

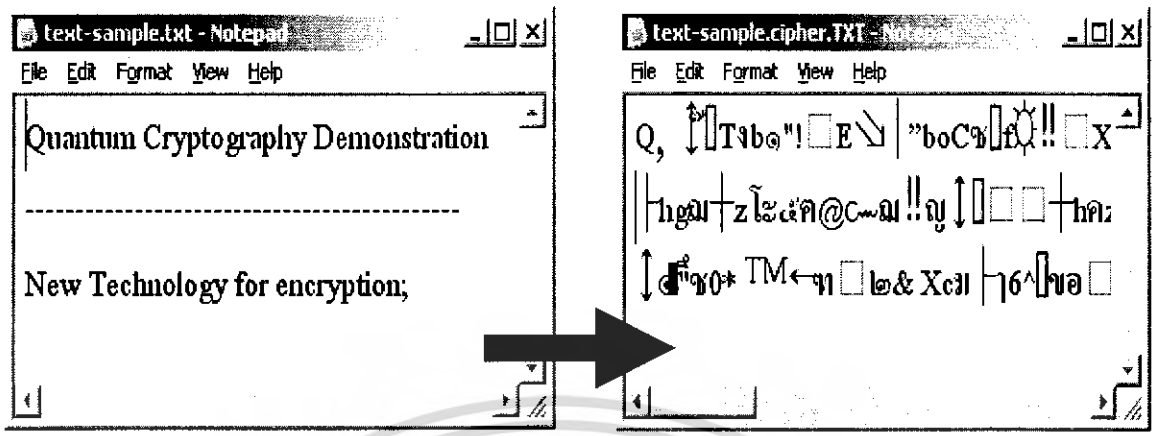
4.1.1 ผลการทดลองการเข้ารหัสแฟ้มข้อมูล

ในการแสดงผลการทดลองในส่วนของการเข้ารหัสแฟ้มข้อมูล เราจะแสดงการเข้ารหัสทั้งไฟล์รูป และไฟล์ข้อความ เพื่อให้เห็นประสิทธิภาพในการทำงานของการเข้ารหัสของซอฟท์แวร์ที่ได้ทำการพัฒนาขึ้น ดังนี้



รูปภาพก่อนเข้ารหัส รูปภาพหลังทำการเข้ารหัสแล้ว

รูปที่ 4.1 แสดงผลการทดลองการเข้ารหัสรูปภาพ



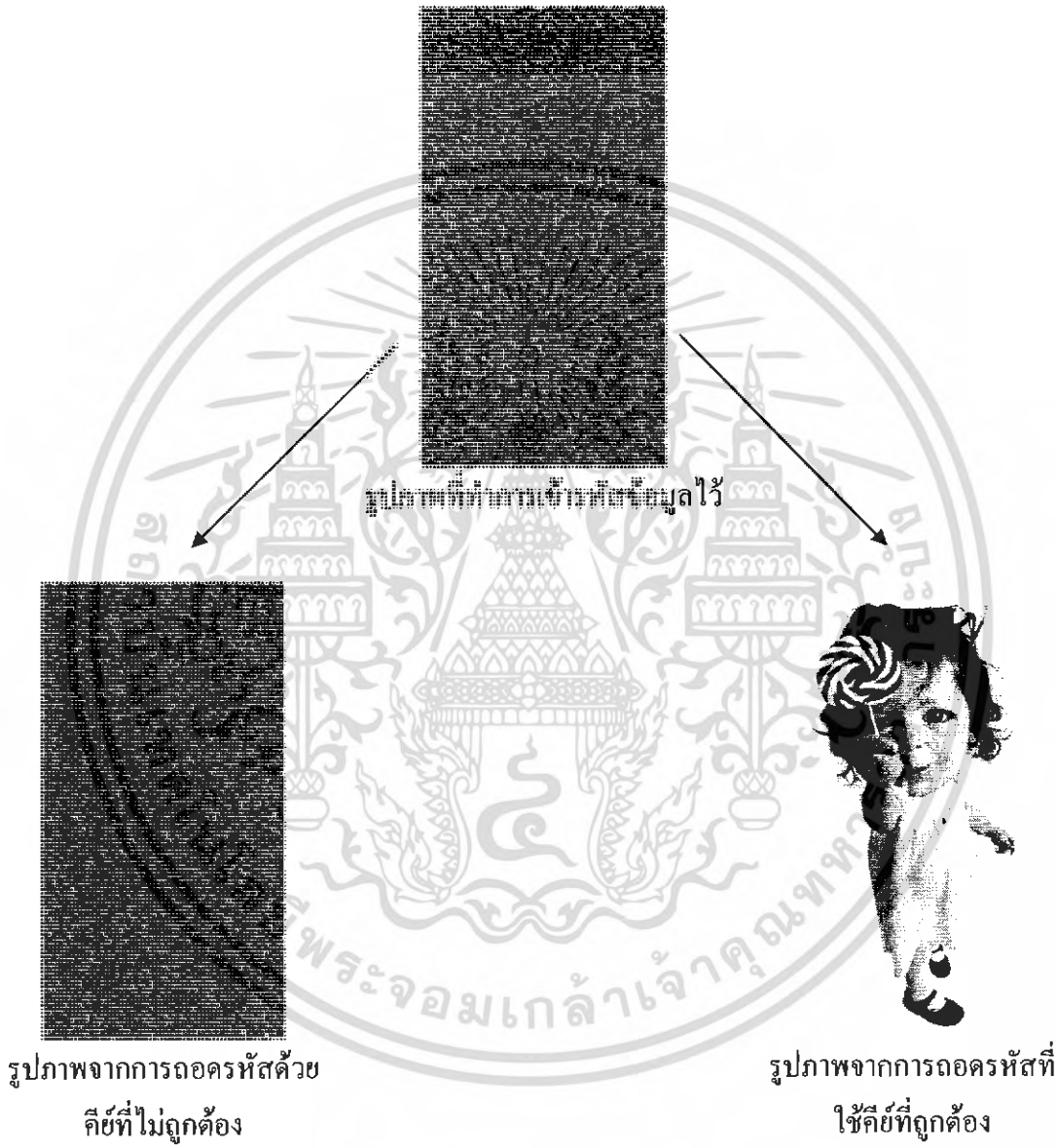
ข้อความก่อนทำการเข้ารหัส ข้อความหลังการเข้ารหัส

รูปที่ 4.2 แสดงผลการทดลองการเข้ารหัสข้อความ

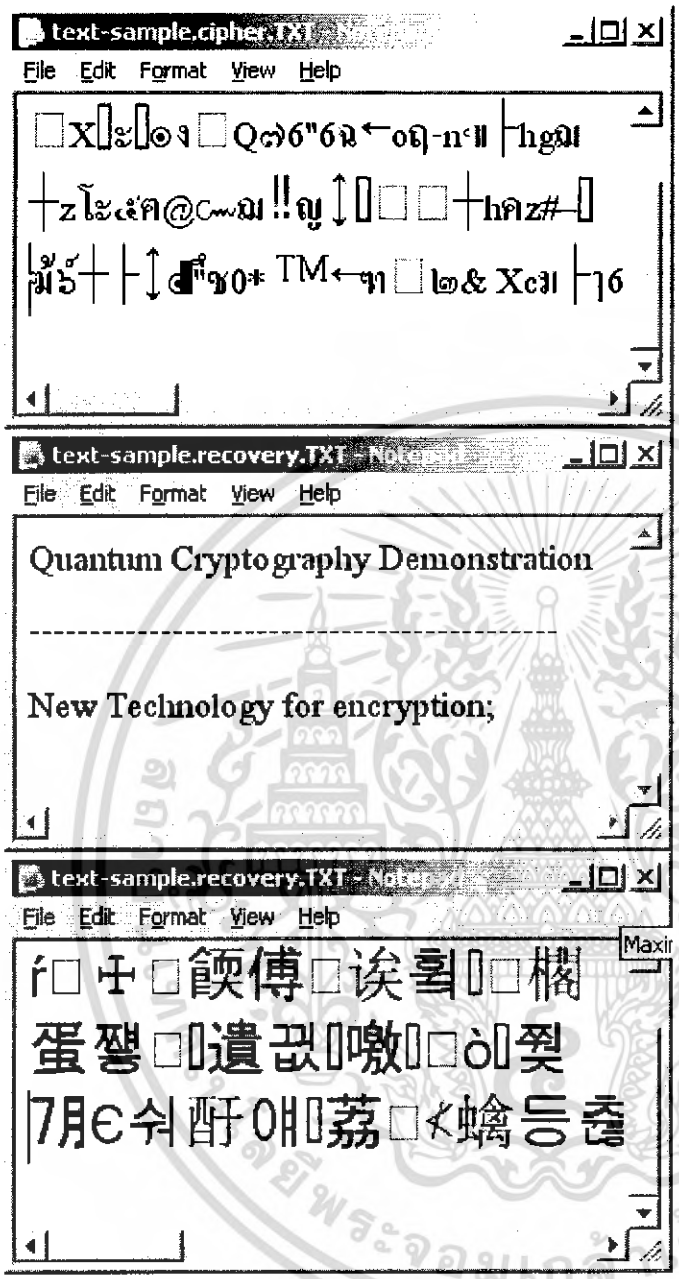
จากผลการทดลองทำให้เห็นว่ารูปแบบและข้อความที่ได้ทำการเข้ารหัสด้วยอัลกอริทึมต่าง ๆ นั้น เป็นข้อมูลที่ไม่สามารถนำไปใช้งานได้หากปราศจากการถอดรหัสเสียก่อน และต้องได้รับการถอดรหัสด้วยคีย์ที่ถูกต้องเท่านั้นจึงจะได้ข้อมูลที่ถูกต้องและสามารถนำไปใช้งานได้ การเข้ารหัสข้อมูลจึงเป็นกลไกหนึ่งเพื่อที่จะเพิ่มความปลอดภัยให้กับข้อมูลของเรา

4.1.1 ผลการทดลองการถอดรหัสแฟ้มข้อมูล

ในการแสดงผลการทดลองในส่วนของการถอดรหัสแฟ้มข้อมูล เราจะแสดงการเข้ารหัสทั้งไฟล์รูป และไฟล์ข้อความ เพื่อให้เห็นประสิทธิภาพในการทำงานของการเข้ารหัสของซอฟท์แวร์ที่ได้ทำการพัฒนาขึ้น ดังนี้



รูปที่ 4.3 แสดงผลการทดลองการถอดรหัสไฟล์รูปภาพด้วยคีย์ที่ถูกต้องและคีย์ที่ไม่ถูกต้อง



ข้อความที่เข้ารหัสลับไว้

ข้อความหลังจากได้ทำการ
ถอดรหัสข้อมูลด้วยคีย์ที่ถูกต้อง

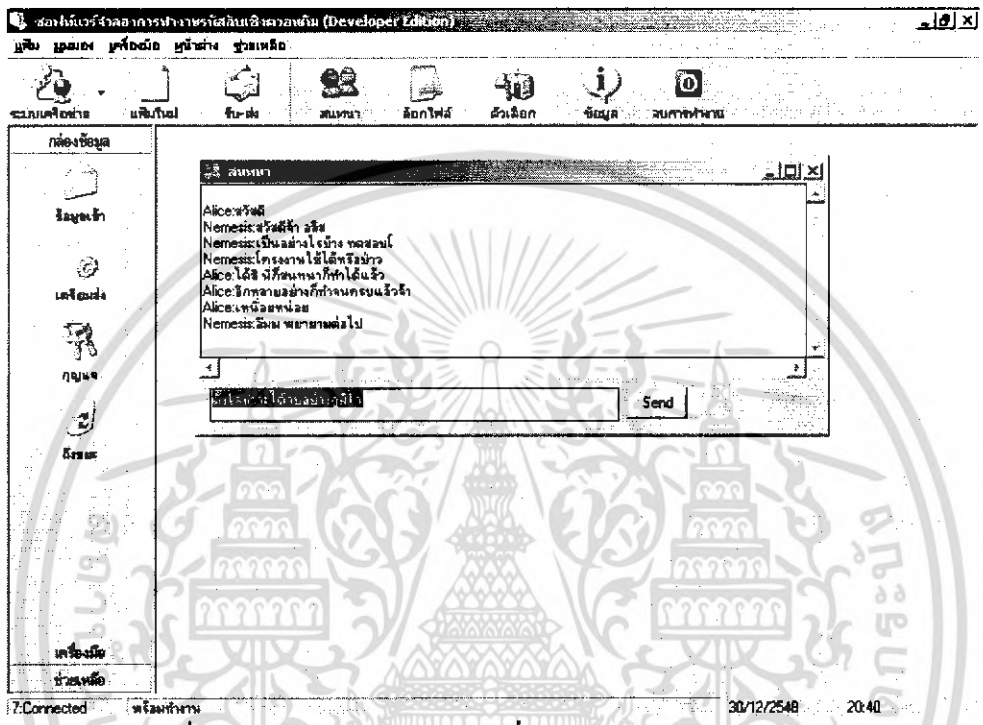
ข้อความหลังจากได้ทำการ
ถอดรหัสข้อมูลด้วยคีย์ที่ไม่ถูกต้อง

รูปที่ 4.4 แสดงผลการทดลองการถอดรหัสข้อความด้วยคีย์ที่ถูกต้องและคีย์ที่ไม่ถูกต้อง

จากการทดลองในส่วนของการเข้ารหัสนั้นจะทำให้ทราบได้ว่าซอฟท์แวร์เข้ารหัสที่ได้พัฒนาขึ้นนั้น หลังจากการเข้ารหัสแล้วข้อมูลรหัสลับนั้นจะเป็นความลับ หากต้องการใช้งานข้อมูลที่เข้ารหัสแล้วจะต้องถอดรหัสด้วยคีย์ที่เป็นคีย์ตัวเดียวกับตอนเข้ารหัสเท่านั้นถึงจะได้ข้อมูลที่ถูกต้องและสามารถนำไปใช้งานได้

4.2 การติดต่อสื่อสารด้วยข้อความ (Instant Messaging)

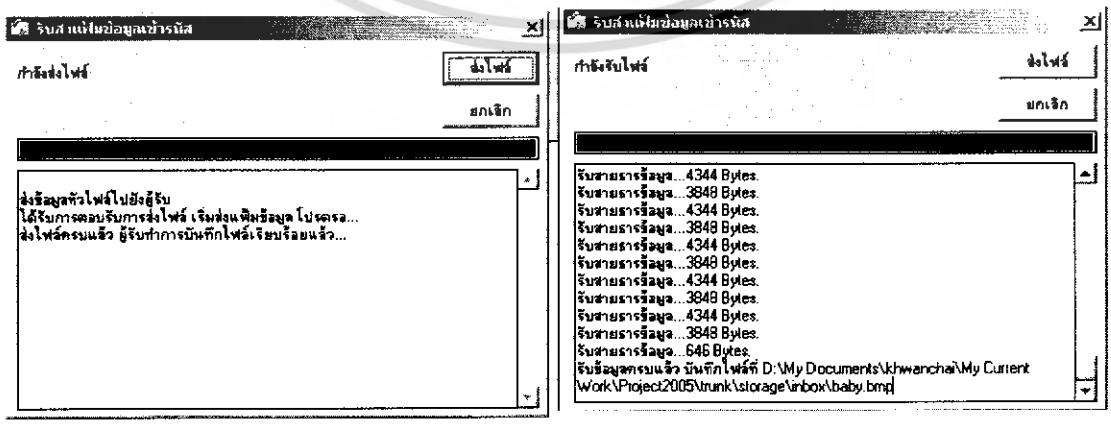
ขณะที่มีการเชื่อมต่อระบบอยู่นั้น ผู้ใช้งานสามารถส่งข้อความสนทนา ถึงกันได้ โดยสามารถเลือกเมนูสนทนา แต่ระบบต้องทำการเชื่อมต่อระบบกันอยู่ถึงจะสามารถใช้งานได้ โดยใช้สำหรับการสนทนาระหว่างผู้ใช้งานสองฝ่าย



รูปที่ 4.5 แสดงหน้าจอการติดต่อสื่อสารด้วยข้อความของระบบ

4.3 การรับส่งข้อมูลผ่านระบบเครือข่ายคอมพิวเตอร์ Public Network

ซอฟท์แวร์สามารถส่งแฟ้มข้อมูลหรือไฟล์ข้อมูลชนิดต่าง ๆ ผ่านทางระบบเครือข่ายคอมพิวเตอร์ได้รวมไปถึงการส่งการตรวจสอบคีย์ด้วยเช่นกัน

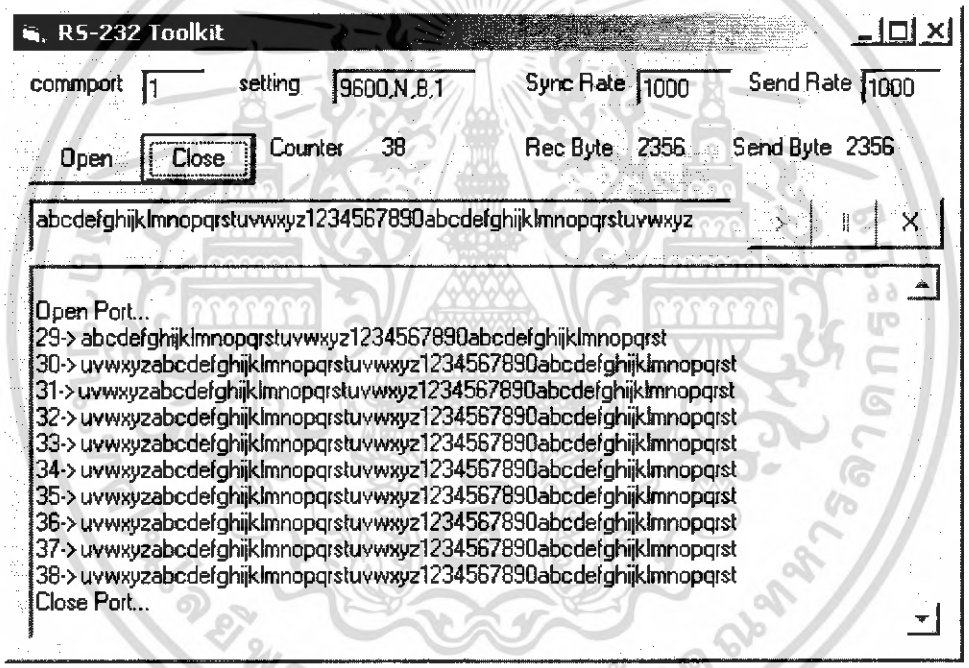


รูปที่ 4.6 แสดงการรับส่งข้อมูลผ่านระบบเครือข่าย Public Network

จากรูปที่ 4.6 จะแสดงให้เห็นการทำงานของซอฟท์แวร์ในส่วนของการส่งข้อมูลผ่าน Public Channel โดยจะใช้ในการส่งข้อมูล Cipher หรือไฟล์อื่น ๆ ได้ตามความต้องการ โดยโปรแกรมจะบอกความสำเร็จและปริมาณข้อมูลในการส่งเพื่อให้ผู้ใช้งานได้ตรวจสอบว่าถูกต้องและตรวจสอบความครบถ้วนของข้อมูลได้

4.4 การรับส่งข้อมูลผ่านพอร์ตอนุกรม RS-232

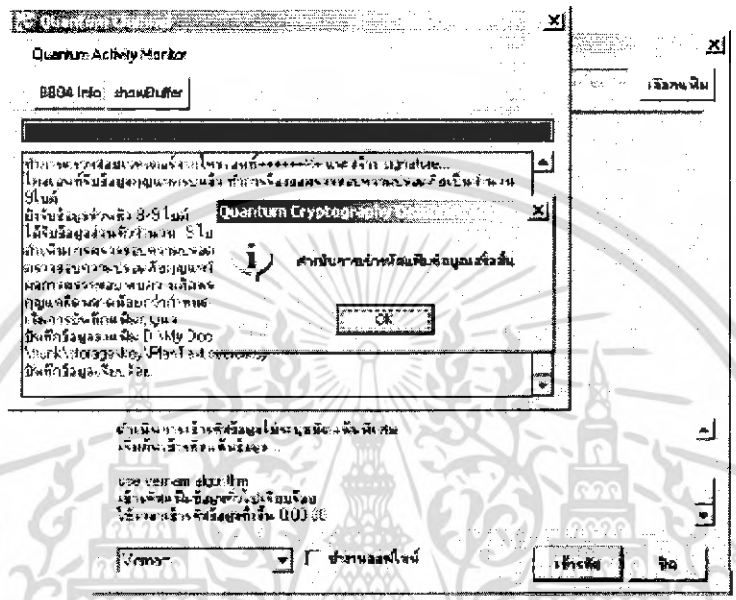
โปรแกรมสามารถรับส่งข้อมูลคีย์ผ่านสาย RS-232 ได้ ซึ่งเป็นส่วนสำคัญของการแลกเปลี่ยนคีย์ที่ใช้ในการเข้ารหัส ซึ่งรูปที่ 4.7 จะแสดงหน้าจอการทำงานของการส่งข้อมูลผ่านพอร์ตอนุกรม (RS-232)



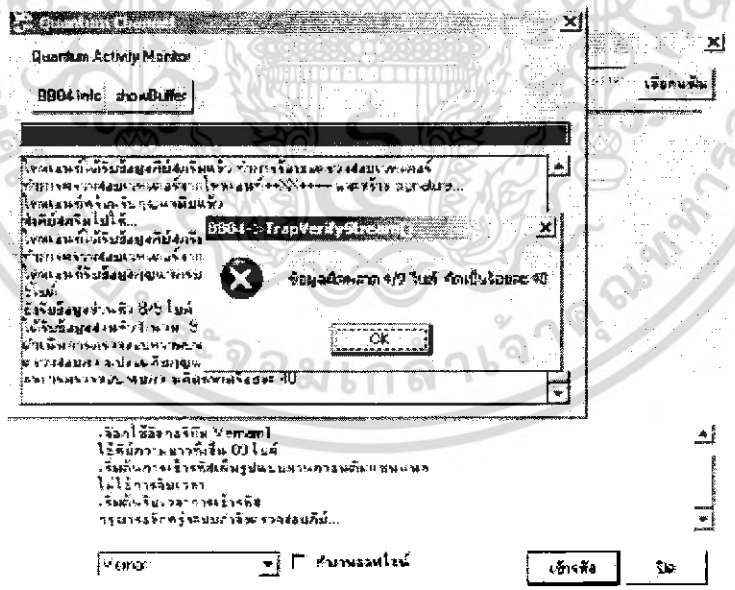
รูปที่ 4.7 แสดงการรับส่งข้อมูลผ่านพอร์ตอนุกรม RS-232

4.5 การตรวจสอบความปลอดภัยของโปรแกรมด้วยโปรโตคอล BB84

ในการทำงานของซอฟท์แวร์จะมีโปรโตคอลที่ไว้สำหรับตรวจสอบข้อมูลในการส่งข้อมูลผ่านช่องสัญญาณ ซึ่งใช้หลักการของโปรโตคอล BB84 ซึ่งได้มีการทำงานของโปรแกรมดังรูปที่ 4.8



รูปที่ 4.8 แสดงการทำงานของการส่งคีย์ที่ปลอดภัย และทำการเข้ารหัสไฟล์ข้อมูลได้อย่างถูกต้อง



รูปที่ 4.9 แสดงการตรวจสอบความผิดพลาดของข้อมูลมากกว่ากำหนด (มากกว่า 25 %) จะยกเลิกการดักจับคีย์

รูปที่ 4.8 และ 4.9 จะแสดงการทำงานที่ปลอดภัยและการทำงานที่มีความผิดพลาดหรือแสดงถึงความไม่ปลอดภัยของข้อมูล โดยระบบจะทำการเตือนและยกเลกคีย์หากการส่งไม่ปลอดภัย หากการส่งคีย์ปลอดภัยระบบจะทำการเข้ารหัสข้อมูลทันที

บทที่ 5
สรุปผลโครงงาน

โครงงานการพัฒนาซอฟท์แวร์จำลองการใช้งานรหัสลับเชิงควอนตัม ได้มีการออกแบบและพัฒนาซอฟท์แวร์เพื่อใช้งานจริงในโครงงานระบบวิทยาการรหัสลับเชิงควอนตัม ซึ่งเป็นโครงการหลักของศูนย์เทคโนโลยีอิเล็กทรอนิคส์และคอมพิวเตอร์แห่งชาติ (NECTEC) โดยในส่วนของโครงงานพัฒนาซอฟท์แวร์จำลองจะเป็นจุดเริ่มต้นของการพัฒนาซอฟท์แวร์ ควบคู่ไปกับการพัฒนาทางด้านฮาร์ดแวร์ ซึ่งการพัฒนาโครงงานพัฒนาซอฟท์แวร์จำลองการทำงานของรหัสลับเชิงควอนตัมก็ได้มีการพัฒนาจนได้ตรงตามเป้าหมายที่กำหนดไว้ซึ่งจะกล่าวในรายละเอียดของการสรุปผลโครงงานต่อไป

5.1 สรุปผลการพัฒนาโครงงาน

การสรุปผลการพัฒนาโครงงาน จะเป็นการสรุปผลการทำงานของซอฟท์แวร์โดยทำการเปรียบเทียบกับวัตถุประสงค์ เป้าหมาย และการออกแบบซอฟท์แวร์ที่ได้มีการออกแบบไว้ก่อนหน้า โดยมีรายละเอียดดังนี้

5.1.1 ซอฟท์แวร์ที่พัฒนาสามารถทำการติดต่อสื่อสารระหว่างคอมพิวเตอร์สองเครื่อง ในการเชื่อมต่อระบบระหว่างคอมพิวเตอร์สองเครื่องจะมีเครื่องหนึ่งเป็นเซิร์ฟเวอร์ และอีกเครื่องหนึ่งเป็นไคลเอนต์ และสามรถส่งข้อความหรือข้อมูลไฟล์ผ่านช่องสัญญาณสาธารณะได้อย่างครบถ้วนและสมบูรณ์ โดยโปรแกรมที่พัฒนาขึ้นนั้นสามารถกำหนดให้ทำงานเป็นเซิร์ฟเวอร์หรือไคลเอนต์ก็ได้ แล้วแต่การกำหนดค่าก่อนเชื่อมต่อระบบ

5.1.2 ซอฟท์แวร์ที่พัฒนาสามารถทำการสุ่มคีย์และทำการส่งข้อมูลคีย์ ผ่านทางสาย Serial Port (Quantum Channel Media) และสามารถทำการตรวจเช็คความถูกต้องของการส่งข้อมูลได้อย่างถูกต้องด้วยวิธีการตามหลักการของโปรโตคอล BB84

5.1.3 ซอฟท์แวร์ที่พัฒนามีโมดูลเกี่ยวกับการเข้ารหัสและถอดรหัสลับ ซึ่งมีอัลกอริทึมที่ให้ผู้ใช้ได้เลือกในการใช้งาน โดยในการทดสอบการใช้งาน ซอฟท์แวร์สามารถทำงานได้อย่างถูกต้อง โดยผู้ใช้งานต้องทำการเลือกคีย์ที่ตรงกับผู้ส่งในการถอดรหัสข้อมูล ซึ่งคีย์ได้ถูกส่งผ่านสาย RS-232 และทำการตรวจเช็คความถูกต้องก่อนหน้าแล้ว

5.1.4 มีการออกแบบ Message ที่ใช้ในการติดต่อทั้งในช่องสัญญาณสาธารณะและสาย Serial Port เพื่อใช้ในการตรวจสอบความถูกต้องและความปลอดภัยของข้อมูล

5.1.5 ผู้พัฒนาได้ทำการสร้างตัวติดตั้งโปรแกรมเพื่อสะดวกในการนำไปใช้งาน ทำให้ผู้ใช้สามารถติดตั้งโปรแกรมได้ง่ายและรวดเร็ว

5.2 ปัญหาด้านการพัฒนา

ปัญหาที่ผู้พัฒนาซอฟท์แวร์ได้พบในระหว่างการพัฒนาโครงงานซอฟท์แวร์จำลองการทำงานของรหัสลับเชิงควอนตัม ซึ่งปัญหาที่พบมีดังนี้

5.2.1 เนื่องจากระบบวิทยาการรหัสลับเชิงควอนตัม เป็นวิทยาการสมัยใหม่ที่ยังไม่แพร่หลายในวงการเทคโนโลยีมากเท่าไร ทำให้การสืบค้นข้อมูลเพื่อใช้ในการศึกษาและอ้างอิงจำนวนน้อย และใช้เวลานานในการทำความเข้าใจที่ถูกต้องเกี่ยวกับระบบวิทยาการรหัสลับเชิงควอนตัม

5.2.2 เนื่องจากการพัฒนาซอฟท์แวร์จำลองการทำงานของรหัสลับเชิงควอนตัมนั้น ได้ใช้ Visual Basic 6 ในการพัฒนา ซึ่งจะพบปัญหาเกี่ยวกับการทำงานที่เข้าถึงไฟล์ขนาดใหญ่ เพราะเมื่อเข้าถึงไฟล์ขนาดใหญ่โปรแกรมจะทำงานช้า เนื่องจากมีลูปการทำงานและการตรวจเช็คความถูกต้องของการส่งข้อมูลหลายขั้นตอน แต่จะทำงานได้รวดเร็วกับไฟล์ขนาดเล็ก

5.2.3 การพัฒนาซอฟท์แวร์จำลองการทำงานของรหัสลับเชิงควอนตัมสามารถจำลองการทำงานโดยภาพรวมของระบบวิทยาการรหัสลับเชิงควอนตัม มีการแยกช่องสัญญาณในการส่งข้อมูลอย่างชัดเจน แต่หากจะนำไปใช้งานในระบบวิทยาการรหัสลับเชิงควอนตัมจริง ๆ นั้นต้องทำการแก้ไขและปรับปรุงในเรื่องต่าง ๆ เพื่อความเหมาะสมกับการใช้งานจริงให้มากยิ่งขึ้น

5.3 แนวทางในการพัฒนาต่อ

โครงงานการพัฒนาซอฟท์แวร์จำลองการทำงานของรหัสลับเชิงควอนตัม เป็นจุดเริ่มต้นของการพัฒนาซอฟท์แวร์ที่ใช้งานในโครงงานวิทยาการรหัสลับเชิงควอนตัมซึ่งเป็นโครงงานหลัก ซึ่งซอฟท์แวร์จำลองเป็นการพัฒนาซอฟท์แวร์ใช้งานกับอุปกรณ์ทางฮาร์ดแวร์ที่จำลองขึ้นมาเพื่อทดสอบการทำงานของซอฟท์แวร์ การที่จะนำซอฟท์แวร์ที่พัฒนาขึ้นไปใช้งานจริงนั้น ต้องมีการพัฒนาต่อเพิ่มเติม เพื่อแก้ไขให้ตรงตามการใช้งาน ดังนี้

5.3.1 ในการนำไปประยุกต์ใช้งานจริงนั้นจะต้องแก้ไขในส่วนของ Quantum Channel Media ที่ใช้อยู่ในโครงงานซอฟท์แวร์จำลองการทำงานของรหัสลับเชิงควอนตัมนั้น ไปใช้อุปกรณ์อื่น ๆ ได้เช่น สายไฟเบอร์ออฟติค เป็นต้น ซึ่งตามหลักการของระบบวิทยาการรหัสลับเชิงควอนตัมนั้นจะต้องใช้สายไฟเบอร์เป็นช่องสัญญาณควอนตัม

5.3.2 ในส่วนของการนำซอฟท์แวร์ที่พัฒนาขึ้นในโครงงานนี้ไปใช้ร่วมกับอุปกรณ์ทางฮาร์ดแวร์ต่าง ๆ อาจจะต้องเปลี่ยนจากการใช้ Visual Basic 6 ในการพัฒนาซอฟท์แวร์มาใช้ภาษาอื่นในการพัฒนาซอฟท์แวร์เพราะจะช่วยให้การติดต่อกับฮาร์ดแวร์ได้สะดวกและรวดเร็ว

5.3.3 ใช้ข้อมูลในการออกแบบซอฟท์แวร์ที่ได้พัฒนาในการพัฒนาระบบจำลองการทำงานของรหัสลับเชิงควอนตัม ที่สามารถจำลองการทำงานและแสดงให้เห็นภาพการทำงานของระบบวิทยาการรหัสลับเชิงควอนตัมได้ใกล้เคียงความจริงมากที่สุด

การพัฒนาต่อเพิ่มเติมในด้านซอฟท์แวร์นั้นอาจจะเปลี่ยนการใช้ภาษาในการพัฒนาเพราะเนื่องจากมีการออกแบบซอฟท์แวร์ที่ยืดหยุ่น เพื่อรองรับต่อการแก้ไขไว้แล้ว จึงง่ายต่อการพัฒนาต่อและง่ายต่อการแก้ไขเพิ่มเติมส่วนต่าง ๆ ของซอฟท์แวร์

บรรณานุกรม

- Bruce Schneier, Applied Cryptography, Wiley, 1994, ISBN 0-471-59756-2.
- Kitt Tientanopajai, and Prathan Srimanchandra, The Internet Security, Special Study Report, Asian Institute of Technology, March 1998.
- National Institute of Standard and Technologies, NIST FIPS PUB 46, January 1977.
- J. Daemen and V. Rijmen, *AES Proposal: Rijndael*, AES Algorithm Submission, September 3, 1999, available at [1].
- J. Daemen and V. Rijmen, The block cipher Rijndael, Smart Card research and Applications, LNCS 1820, Springer-Verlag, pp. 288-296.
- Problems and Solutions in Quantum Computing and Quantum information, Willi-Hans Steeb Yoriek Hardy, Copyright 2004 by World Scientific Publishing Co. Ptc. Ltd.
- Printciples of Quantum Computation and information Volume I, Giuliano Benenti, Casami, Giuliano Strini, Copyright 2004 by World Scientific Publishing Co. Ptc. Ltd.
- Simplified Visual Basic, Willie J. Wilborn, QA 76.73.B3W453 2004, Pearson Prentice Hall
- Visual Basic 6.0 Advanced Topics Michael V. Ekedanl, Course Technology 2000
- วิทยาการรหัสลับเบื้องต้น ลัญฉกร วุฒิสิทธิกุลกิจ ISBN 974-13-3322-6 สำนักพิมพ์จุฬาลงกรณ์ มหาวิทยาลัย 2548

ภาคผนวก

การออกแบบ Software ด้วยวิธี UML

1. Vision

เป็นขั้นตอนการออกแบบ Software โดยเริ่มต้นด้วยการวิเคราะห์ถึงปัญหา สาเหตุของความจำเป็นในการพัฒนา Software การนำไปใช้งาน เพื่อใช้เป็นข้อมูลในการออกแบบและพัฒนาต่อไป

1.1 Introduction

ปัจจุบันระบบวิทยาการสร้างรหัสลับเชิงควอนตัม (Quantum Cryptography System) ซึ่งเป็นระบบวิทยาการรหัสลับที่ความปลอดภัยของคีย์จะรับประกันด้วยกฎพื้นฐานทางฟิสิกส์นั้น จัดเป็นเทคโนโลยี 1 ใน 20 เทคโนโลยีที่น่าจับตามอง ถึงแม้ว่าจะมีบริษัทผู้ผลิตระบบวิทยาการสร้างรหัสลับเชิงควอนตัม (Quantum Cryptography System) ออกมาจำหน่ายบ้างแล้วแต่ก็ยังคงมีราคาสูงมาก รวมทั้งงานวิจัยก็ยังไม่ถึงจุดอิ่มตัว ยังไม่มีการกำหนดมาตรฐานการใช้งานที่แน่นอน ดังนั้น NECTEC จึงได้ทำการวิจัยและสร้างระบบวิทยาการสร้างรหัสลับเชิงควอนตัม (Quantum Cryptography System) ขึ้นมาใช้เอง เพื่อสามารถนำมาใช้ให้เกิดประโยชน์และลดค่าใช้จ่ายในอนาคต ซึ่งในปัจจุบันกำลังดำเนินงานอยู่ในขั้นตอนการวิจัยและพัฒนา

การสร้างรหัสลับเชิงควอนตัมมีความแตกต่างจากการเข้ารหัสทั่วไปโดยที่จะมีการเข้ารหัสที่ใช้คีย์ที่ได้จากการสุ่มร้อยเปอร์เซ็นต์ และทำการส่งคีย์ไปยังผู้รับเพื่อใช้ในการถอดรหัสผ่านทาง ช่องสัญญาณควอนตัม (Quantum Channel) แทนการส่งคีย์ด้วยวิธีอื่น ๆ ที่พบได้ทั่วไปในปัจจุบันเช่น ส่งผ่าน Internet, จดหมาย, ใช้คนเป็นผู้ส่งโดยตรง ซึ่งวิธีการเหล่านี้ล้วนแล้วแต่ใช้หลักแห่งความเชื่อใจ ไม่ปลอดภัยเหมือนกับการส่งผ่านช่องสัญญาณควอนตัม (Quantum Channel) เนื่องจากช่องสัญญาณควอนตัม (Quantum Channel) เมื่อมีใครลักลอบดักจับสัญญาณผู้รับสามารถตรวจสอบได้แน่นอน

ซอฟท์แวร์จำลองการใช้รหัสลับเชิงควอนตัม (Quantum Cryptography Demonstration Software) นี้เป็นการจำลองการทำงานของระบบวิทยาการสร้างรหัสลับเชิงควอนตัม (Quantum Cryptography System) เพื่อให้เห็นการทำงานของระบบ โดยใช้หลักการและวิธีการที่คล้ายคลึงกับระบบจริง

1.2 Business opportunity

Quantum Cryptography System เราสามารถนำมาประยุกต์ใช้กับงานสื่อสารข้อมูลดิจิตอลที่ต้องการความปลอดภัยสูง และต้องการความน่าเชื่อถือสูง เช่น สถาบันการเงิน, องค์กรของรัฐ, บริษัทต่างๆ ที่มีการส่งข้อมูลผ่าน Public Channel ที่ต้องการความปลอดภัยของข้อมูล เป็นต้น

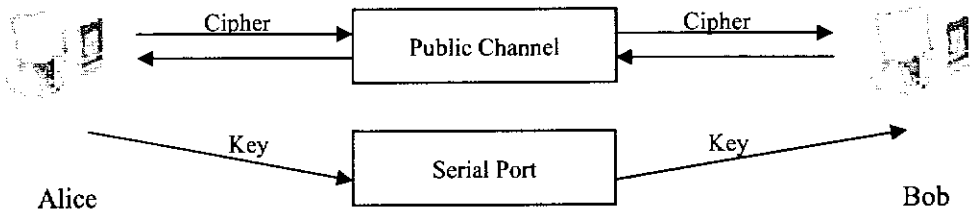
1.3 Stakeholder Descriptions

High-Level Goal	Priority	Problems and Concerns	Current Solutions
Quantum Channel	High	Protect key from Hacking	-
Encrypt Data	High	Protect data into cipher	General Encryption
Generate key	High	Prevent key prediction	Make from simple word or random
Authorize key	High	Make sure key is secure	Receiver check with transmitter
Send cipher to public channel	Low	Direct send cipher	FTP, Web download, E-mail

1.4 Product overview

การทำงานของระบบ (Alice ต้องการส่งข้อมูลไปยัง Bob)

- Alice ทำการสุ่มคีย์เพื่อใช้ในการเข้ารหัสข้อมูล และส่งคีย์ไปยังBob ผ่าน Quantum Channel (ในรูปแบบจำลองจะทำการส่งผ่าน Serial Port)
- Bob นำคีย์ที่ได้รับมาทำการ Check ความปลอดภัยกับ Alice ใน Public Channel เมื่อพบว่าปลอดภัย Alice จะทำการเข้ารหัสข้อมูลด้วยคีย์ที่ปลอดภัย ข้างต้น และทำการส่ง Cipher ผ่าน Public Channel ไปยัง Bob
- เมื่อ Bob ได้รับ Cipher ก็จะนำมาถอดรหัสกับคีย์ที่ได้รับมาก่อนหน้า และจะได้มาซึ่งข้อมูลที่ถูกต้องและปลอดภัย



รูปที่ 1(ภาคผนวก) แสดงการทำงานของซอฟท์แวร์จำลองการใช้รหัสลับเชิงควอนตัม

1.5 Summary of Benefits

Supporting Feature	Benefit
100% Key Generate	User ได้คีย์ที่ยากต่อการคาดเดา
One time-pad (วิธีการเข้ารหัส)	1 คีย์ : 1 ข้อมูล จะทำให้ข้อมูลปลอดภัยมากขึ้น
Quantum Channel	Key จะมีการเปลี่ยนแปลงหากมีการดักจับ
Authorize key	ป้องกันการถูกดักจับได้ค่อนข้างแม่นยำ

1.6 Summary of system feature

- Generate key from Quantum Random Generation (ระบบจำลองการสุ่มด้วยโปรแกรมคอมพิวเตอร์)
- Encode-Decode file correctly
- Authorize key for security
- Change key for each file

2 Use Case

2.1 System Boundary

ขอบเขตของโครงงานจะอยู่ในส่วนของการทำซอฟท์แวร์จำลองการทำงานของระบบวิทยาการสร้างรหัสลับเชิงควอนตัม (Quantum Cryptography System) ซึ่งมีขอบเขตดังนี้

1. สร้างโปรแกรมจำลองการรับและส่งข้อมูลดิจิตอลระหว่างคอมพิวเตอร์สองเครื่องผ่าน Public Network

2. สร้างโปรแกรมที่ทำหน้าที่ในการ Generate Key แทนการสุ่มคีย์จาก Quantum Random Generation

3. สร้างโปรแกรมที่มีหน้าที่ควบคุมและส่งผ่านข้อมูลคีย์ ผ่านทาง Serial Port (RS-232) แทนการส่งผ่านช่องสัญญาณควอนตัม เนื่องจากการสุ่มคีย์จาก Quantum Random Generation และ ช่องสัญญาณควอนตัม เป็นวิธีการที่มีราคาแพงและใช้วิทยาการขั้นสูงซึ่งปัจจุบันยังอยู่ในขั้นตอนการวิจัย โครงงานนี้จึงจำลองโดยการใช้อุปการณ์ที่หาได้และราคาถูกในการสร้างและพัฒนา

4. สร้างโปรแกรมในส่วนของการ Authorize Key ซึ่งทำผ่านระบบ Public Channel

2.2 The Actor-Goal List

Actor	Goal	UC (Use Case)
System admin (Sender and Receiver)	- Encrypt Data	-Encrypt Data
	- Decrypt Cipher	-Decrypt Cipher
	- Send Cipher and data to public channel	-Send File
	- Send MSG	-Send Message
	- Start up program	- Start server
	- Shutdown program	- Shutdown server
	-View log	- View log
	- manage file and key	- Manage file
	- Connect Server	- Connect Server
	- Receive File	- Receive File

ตารางที่ 1(ภาคผนวก) แสดงการกำหนด Actor-Goal และกำหนด Use Caseของระบบ

1) Use Case: Start Server

Primary actor: System admin

Stakeholders & Interests: System admin ต้องการเปิดระบบ software โดยจะตรวจสอบความพร้อมของ Public Channel & Quantum Channel (serial port) ก่อนระบบจึงจะเริ่มทำงาน

Main Success Scenario:

Actor Action	System Response
1. System Admin try to start server	2. Open connection 3. record log file

Extensions:

Actor Action	System Response
1a. Start server without public network connected 1. start server	2. Display Error MSG 3. Cancel operation
1b. Stat Server without Quantum Channel Connected (RS-232) 1. start server	2. Display Error MSG 3. Cancel Operation

Special Requirements

- Public Network Connected (LAN, Internet)
- Quantum Channel (RS-232) COM PORT Connected

2) Use Case: Connect Server

Primary actor: System admin

Stakeholders & Interests: System admin ทำการเชื่อมต่อไปยังคู่เครือข่าย

Main Success Scenario:

Actor Action	System Response
1. System Admin try to connect server	2. connection 3. record log file 4. display/set status "CONNECTED"

Extensions:

Actor Action	System Response
1a. Connect without public network connected 1. Connect	2. Display Error MSG 3. Cancel operation
1b. Stat Server without Quantum Channel Connected (RS-232) 1. Connect	2. Display Error MSG 3. Cancel Operation

Special Requirements

1) Public Network Connected (LAN, Internet)

2) Quantum Channel (RS-232) COM PORT Connected

3) Use Case: View log

Primary actor: System admin

Stakeholders & Interests: System admin ต้องสามารถดูบันทึกการทำงาน และสามารถลบ log file ทิ้งในส่วนของบันทึกการทำงานสำเร็จและบันทึกการทำงานล้มเหลวของระบบได้

Main Success Scenario:

Actor Action	System Response
1. System Admin สั่งดู log file	2. Load log file และแสดง Log file

4) Use Case: Send Message

Primary actor: System admin

Stakeholders & Interests: System admin สามารถทำการส่งข้อความพูดคุยกับฝ่ายตรงข้ามได้ผ่านทาง Public Channel

Main Success Scenario:

Actor Action	System Response
1. System Admin type message 2. Send message	 3. ส่งข้อความไปยังผู้รับ 4. Update list ที่แสดงผลข้อความ

Extensions:

Actor Action	System Response
2a. Send message 1. ส่งข้อความแล้วผู้รับไม่สามารถรับได้เนื่องจาก Connection fail	 2. System ทำการแจ้งเตือนหากส่งไม่ได้

5) Use Case: Manage file

Primary actor: System admin

Stakeholders & Interests: System admin สามารถเลือกไฟล์มาใส่ไว้ใน folderเพื่อรอการเข้ารหัสได้ โดยสามารถทำการจัดกานกับข้อมูล (Data), ข้อมูลลับ (Cipher) และกุญแจไขรหัสลับ (Key)

Main Success Scenario:

Actor Action	System Response
1. System Admin กด Browse เลือกไฟล์	 2. system ทำการคัดลอกไฟล์ลงแฟ้มข้อมูล

6) Use Case: Send File

Primary actor: System admin

Stakeholders & Interests: System admin ทำการส่งไฟล์

Main Success Scenario:

Actor Action	System Response
1. System Admin เลือกไฟล์ที่ต้องการส่ง	2. เตรียมความพร้อม ตัดแบ่งข้อมูล สร้างHeader ก่อนส่ง 3. เริ่มส่งข้อมูลจนครบ 4. รายงานการส่งข้อมูล

Extensions:

Actor Action	System Response
1a. ทำการเลือกไฟล์แล้วส่ง 1. Connection fail	2. System ทำการแจ้งเตือนหากส่งไม่ได้ 3. ยืนยันการส่งใหม่อีกรอบ

7) Use Case: Encrypt Data

Primary actor: System admin

Stakeholders & Interests: System admin ต้องการเข้ารหัสไฟล์ข้อมูล ซึ่งตรวจสอบแล้วว่า key มีความปลอดภัย โดยใช้ Vernam หรือ Algorithm ที่ระบุ

Main Success Scenario:

Actor Action	System Response
1. System Admin เลือกไฟล์ที่ต้องการทำการเข้ารหัส	
2. สั่งเข้ารหัส	3. อ่านขนาดไฟล์
	4. Generate key ตามจำนวน Bit ที่อ่านมาหรือสามารถกำหนดเองได้
	5. ส่งkeyผ่าน Quantum Channel (RS-232)
	6. ตรวจสอบคีย์ ผ่าน public channel
	7. เข้ารหัสข้อมูลด้วยคีย์ที่ถูกต้องและปลอดภัย

Extensions:

Actor Action	System Response
2a.	
1. สั่งเข้ารหัสไฟล์ แล้วคีย์ถูกดักจับ	
	2. สั่ง Generate key ใหม่
	3. ส่ง key ผ่าน Quantum Channel(RS-232)
	4. ตรวจสอบ
	5. เข้ารหัส

8) Use Case: Decrypt Cipher

Primary actor: System admin

Stakeholders & Interests:

System admin ทำการถอดรหัสCipher

Main Success Scenario:

Actor Action	System Response
1. System Admin เลือกไฟล์ที่ต้องการทำการถอดรหัส	2. เลือกคีย์ที่ตรงกับไฟล์แล้วทำการถอดรหัสไปเก็บไว้ที่folderที่จัดไว้ 3. แจ้งเตือนว่าถอดรหัสสำเร็จ/preview ข้อมูลได้ 4. ลบ Cipher และ คีย์ ที่ใช้แล้วทิ้งไป

Extensions:

Actor Action	System Response
1a. เลือก Cipher แล้วหาคีย์ที่จะใช้ถอดรหัสไม่เจอ 1. ส่งถอดรหัสไฟล์	2. ระบบแจ้งเตือน 3. ลบ cipher ทิ้งไป
1b. คีย์ผิดพลาด 1. ส่งถอดรหัสไฟล์	2. ระบบแจ้งเตือน 3. ลบ Cipher ทิ้ง 4. ลบคีย์ทิ้ง

9) Use Case: Shutdown Server

Primary actor: System admin

Stakeholders & Interests: System admin ต้องการปิดระบบ อย่างปลอดภัย โดยการปิดจาก Quantum Channel (serial port) -->Public Channel

Main Success Scenario:

Actor Action	System Response
1. System Admin สั่งปิดระบบ	2. ปิด Quantum Channel(RS-232) 3. ปิด Public Channel (LAN, Internet)

3 Supplementary Specification

3.1 Functionality

- Software สามารถรองรับและทำงานกับไฟล์ข้อมูลหลายๆชนิดได้

3.2 Usability

- เป็น Software ที่มีรูปแบบ Interface ที่ง่ายต่อการใช้งานและสามารถศึกษาวิธีการใช้งานได้ง่าย

3.3 Reliability

- ผู้ใช้สามารถมั่นใจในความปลอดภัยของคีย์ที่ทำการส่งว่าปลอดภัยได้ 100% (หรือมี Error ได้น้อยกว่า 11%)

3.4 Supportability

- Adaptability: ซอร์ฟแวร์ support การใช้งานร่วมกับ Comport, USB, IRDA

- Configurability: สามารถใช้งานได้กับระบบ Network เดิมได้ (Public Channel)

3.5 Implementation Constraints

ใช้ Visual Basic ในการพัฒนา Software และเลือกวิธีการส่งคีย์ผ่าน RS-232 ในการImplementในProjectนี้แทนการส่งผ่าน Quantum Channel

3.6 Interface

- Hardware Interfaces: RS-232 (Other cable for direct connection), Network connection (LAN, Internet)

- Software Interface: Software สามารถทำงาน P2P เนื่องจากสามารถประพฤติตัวเป็นได้ทั้ง Client และ Server นอกจากนั้นSoftwareยังสามารถรองรับและทำงานกับไฟล์ได้หลายรูปแบบ ทั้งไฟล์ข้อมูล รูปภาพ มัลติมีเดีย เป็นต้น

3.7 Domain (Business) Rules

ID	Rule	Changeability	Source
RULE 1	Generate key เพื่อใช้ในการเข้ารหัสบิตต่อบิต (Vernam)	Low. อาจใช้ Algorithm อื่นในการเข้ารหัสได้	Vernam Theory
RULE 2	Key ที่ใช้จะใช้เพียงครั้งเดียว	-	One time-pad
RULE 3	Key ที่ทำการตรวจสอบแล้ว Error มากว่า 11% จะทำการยกเลิกkey นั้น เนื่องจากไม่ปลอดภัย	Low. อาจใช้เกณฑ์อื่นในการกำหนดระดับความปลอดภัยได้	Standard

ตารางที่ 2(ภาคผนวก) ตารางแสดง Domain Rules

3.8 Information in Domain of Interest

- Algorithm ที่ใช้ในการเข้ารหัส ในที่นี้จะใช้วิธีการแบบ one time-pad (Vernam) เข้ารหัสบิตต่อบิต และสามารถเลือกใช้ Symmetric Key Algorithms อื่น ๆ ที่มีในโปรแกรมได้

- Key Authorize การตรวจสอบคีย์ต้องมีความแม่นยำและแน่นอนในการใช้งาน โดยระดับที่จะยอมรับได้คือ สามารถมี Error ได้อย่างมาก 11%

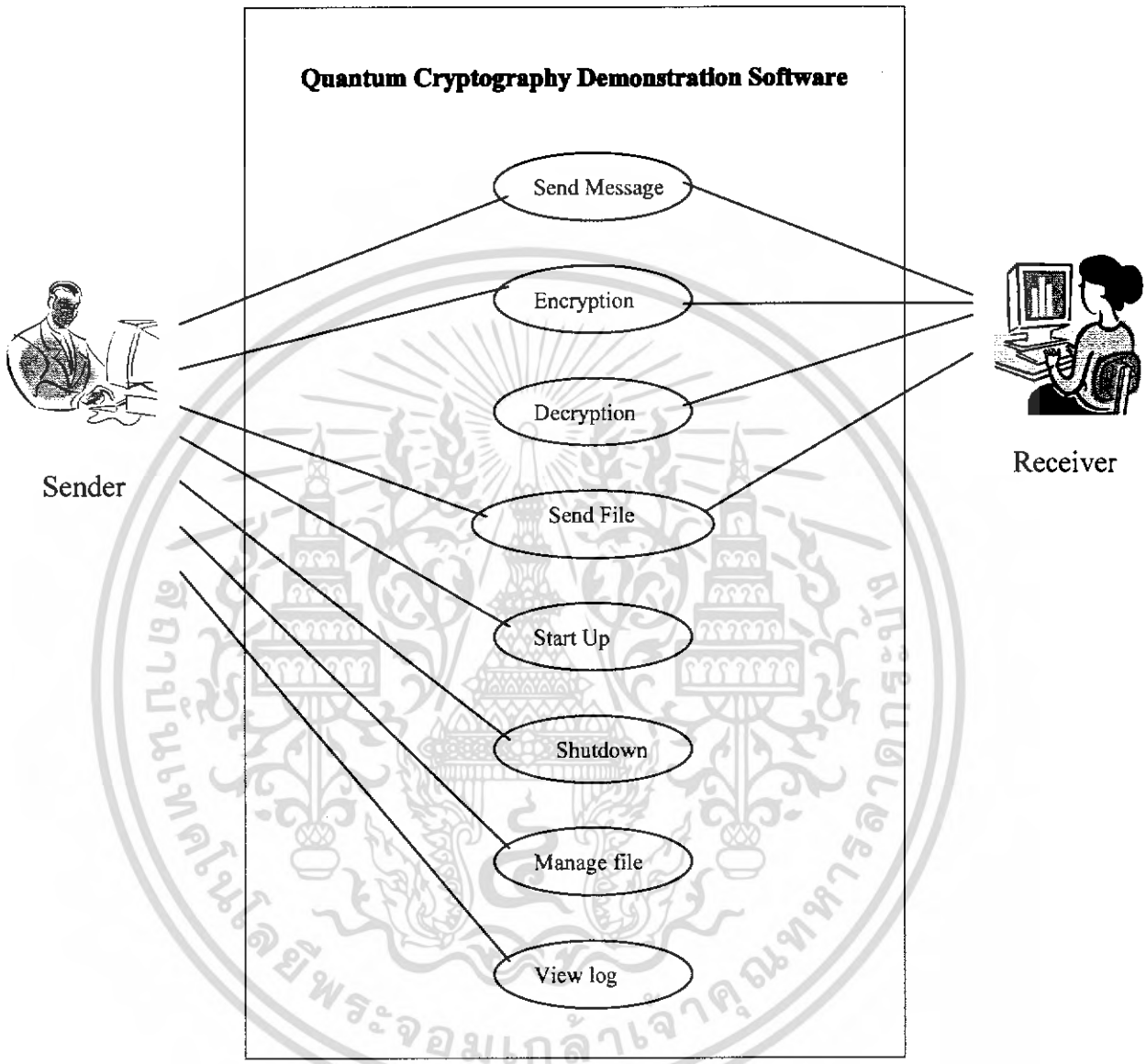
- Key Pattern รูปแบบการ Generate key จะใช้ Random function จาก Computer แทนการสุ่มจาก Hardware และเป็นคีย์ที่ใช้ครั้งเดียวแบบ one-time pad

4. Glossary

Term	Definition and Information	Aliases
Encryption	การเข้ารหัสข้อมูลด้วยkeyเพื่อเพิ่มความปลอด	
Decryption	การถอดรหัสด้วยข้อมูลที่เข้ารหัสไว้ด้วย key จะได้ข้อมูลที่แท้จริงออกมา	
Key	เป็นชุดข้อมูลที่ใช้ในการเข้ารหัส เพื่อสร้างข้อมูลที่ไม่สามารถใช้ได้ หากไม่มีคีย์ตัวที่ใช้ในการเข้ารหัส	
Public Network	ระบบเครือข่ายที่เป็นสาธารณะผู้ใดก็สามารถเข้าใช้งานได้	
Cipher	ข้อมูลที่ทำการเข้ารหัสลับแล้ว	
Quantum Channel	ช่องทางการส่งข้อมูล ด้วยแสงผ่านสาย Fiber ซึ่งสามารถตรวจสอบผู้ลักลอบดักข้อมูลได้	

ตาราง 3.3 แสดงการคำศัพท์ต่างๆ ที่เกียวข้องกับระบบที่ออกแบบ

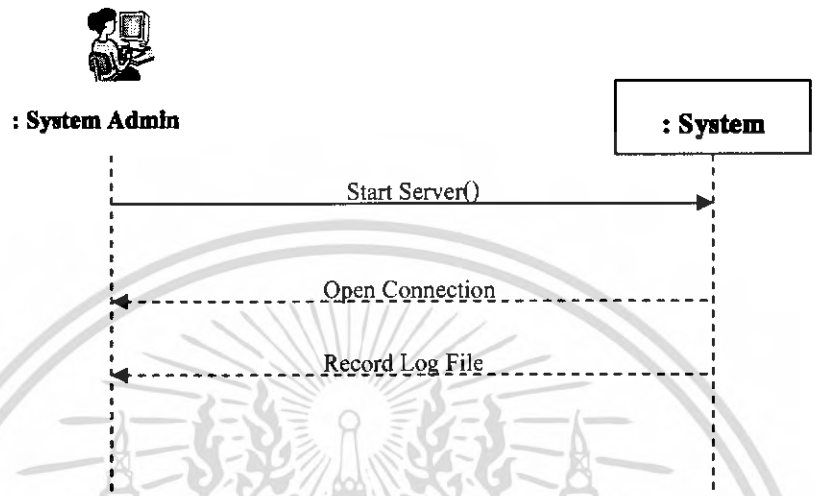
5. Use Case Diagram



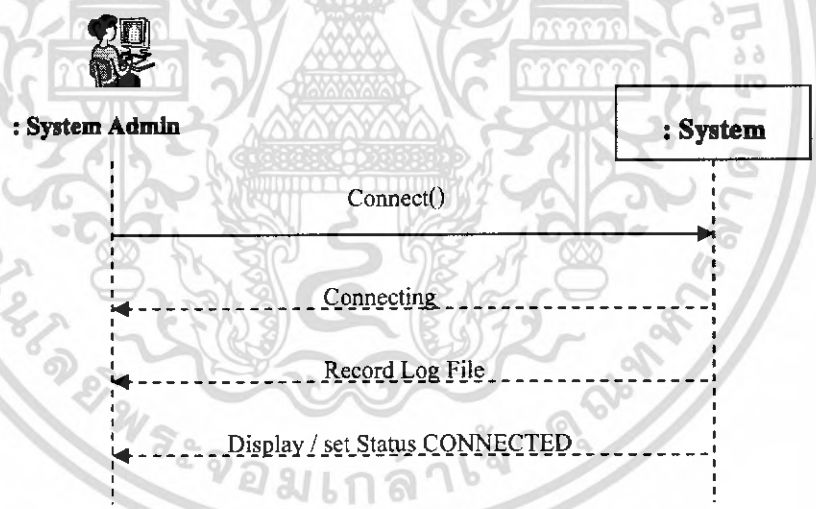
รูปที่ 2(ภาคผนวก) แสดงUse Case Diagram ของระบบที่ออกแบบ

6. System Sequence Diagrams

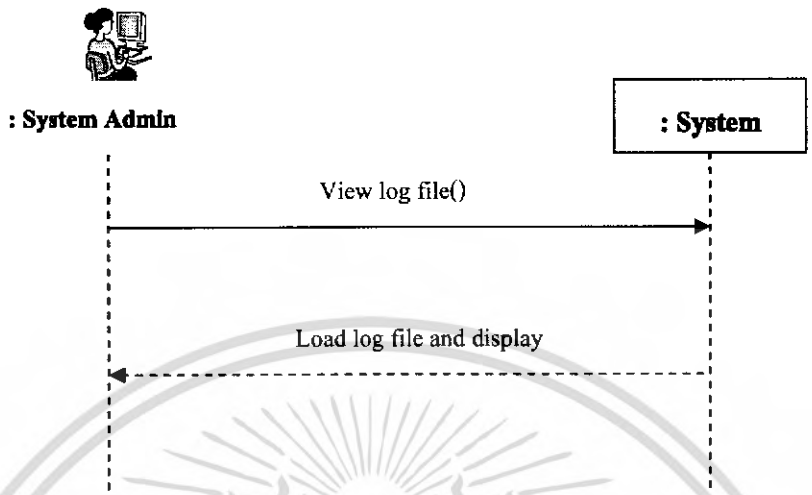
- **SSD: Process Start Server**



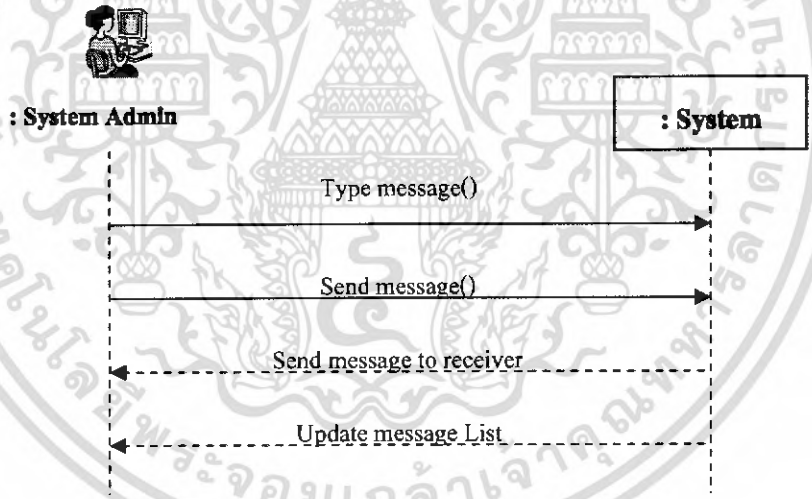
- **SSD: Process Connect Server**



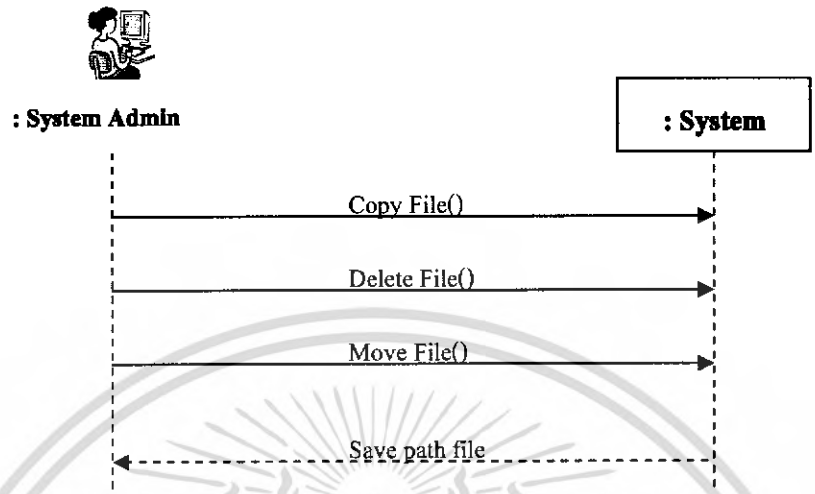
- **SSD: Process View Log**



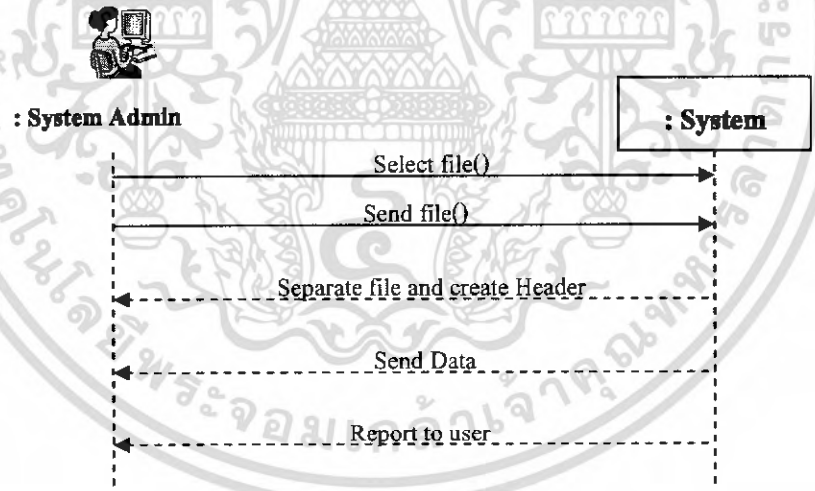
- **SSD: Process Send Message**



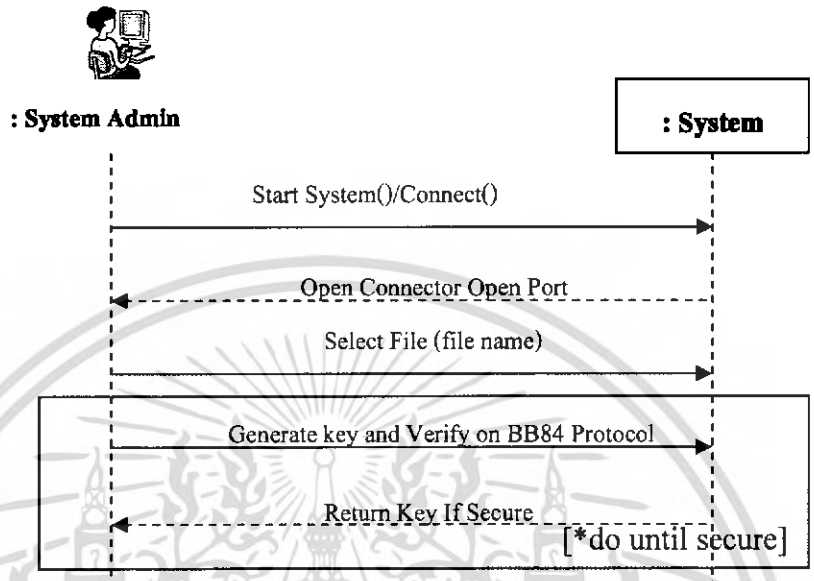
- **SSD: Process Manage file**



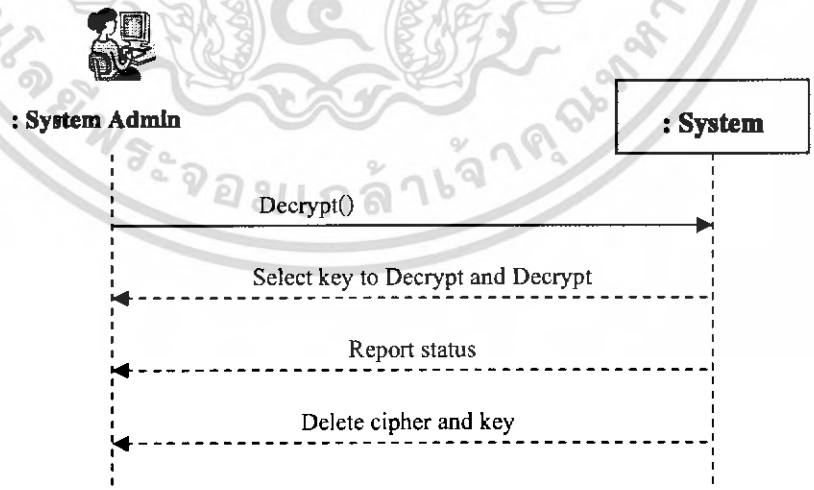
- **SSD: Process Send File**



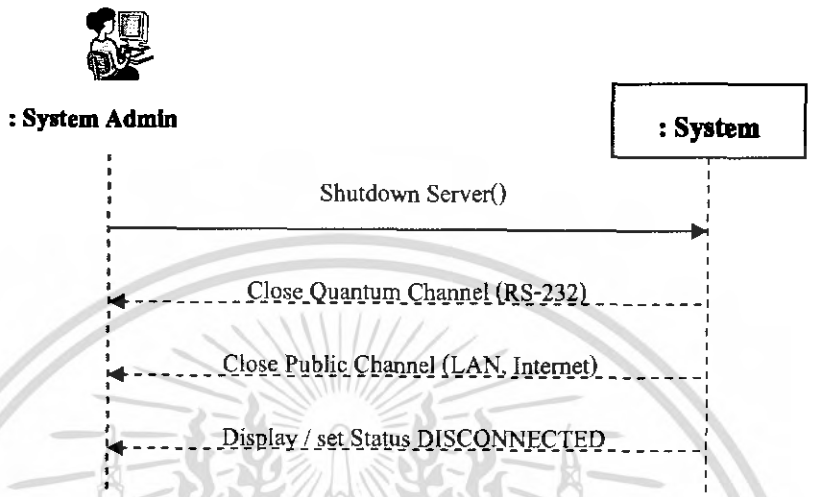
- **SSD: Process Encrypt Data**



- **SSD: Process Decrypt Cipher**



- **SSD: Process Shutdown Server**



7. Contracts and Use Case Realizations

- **Contract CO: Start Server**

Operation: Cross	Start Server()
References:	Use Case: Process Start Server
Preconditions:	none
Post conditions:	- ระบบจะทำการเปิด Connection พร้อมทำงาน - ทำการ Record Log file - ระบบจะทำการแสดงผลstatusของ server บนหน้าแสดงผล

- **Contract CO: Connect**

Operation: Cross	Connect()
References:	Use Case: Connect Server
Preconditions:	Server Started
Post conditions:	- ระบบจะทำการ Connecting ทั้ง Public Channel และ RS- 232 - ทำการ Record Log file และแสดงสถานะ CONNECTED

- **Contract CO: View log file**

Operation: Cross	*View log file()*
References:	Use Case: View log
Preconditions:	System Started
Post conditions:	- ระบบจะทำการ Load log file ที่ต้องการ - Display log file ที่หน้าจอแสดงผล

- **Contract CO: Send message**

Operation: Cross	Send message()
References:	Use Case: Send Message
Preconditions:	System Connected
Post conditions:	- ระบบจะทำการส่งข้อมูลไปยังผู้รับผ่าน Public Channel - Update Display message บนหน้าจอประมวลผล

- **Contract CO: Browse file**

Operation: Cross	Browse file()
References:	Use Case: Manage file
Preconditions:	System Started
Post conditions:	- ระบบทำการบันทึก path file ที่จะเข้าทำการเข้ารหัส

- **Contract CO: Send file**

Operation: Cross	Send file()
References:	Use Case: Send Public Network
Preconditions:	- System Started - Select file()
Post conditions:	- ทำการแบ่งไฟล์เป็นส่วนๆ และสร้างHeader - ทำการส่งข้อมูลให้ครบทุกส่วน - รายงานการส่งข้อมูล โดยแสดงผลบนหน้าจอแสดงผล

- **Contract CO: Encrypt**

Operation: Cross	Encrypt()
References:	Use Case: Encrypt Data
Preconditions:	- System Connected - ระบบทำการอ่านข้อมูลหาขนาดไฟล์ - ทำการสุ่มคีย์ตามจำนวนbitของข้อมูลและทำการส่งไปยังผู้รับผ่าน RS-232 - ตรวจเช็คความถูกต้องของคีย์ระหว่างผู้รับและผู้ส่งผ่าน Public Channel และได้รับการยืนยันความถูกต้อง
Post conditions:	- เมื่อพบว่าคีย์ที่ส่งไปปลอดภัยระบบจะทำการเข้ารหัสไฟล์ โดยใช้Algorithm: One-time Pad

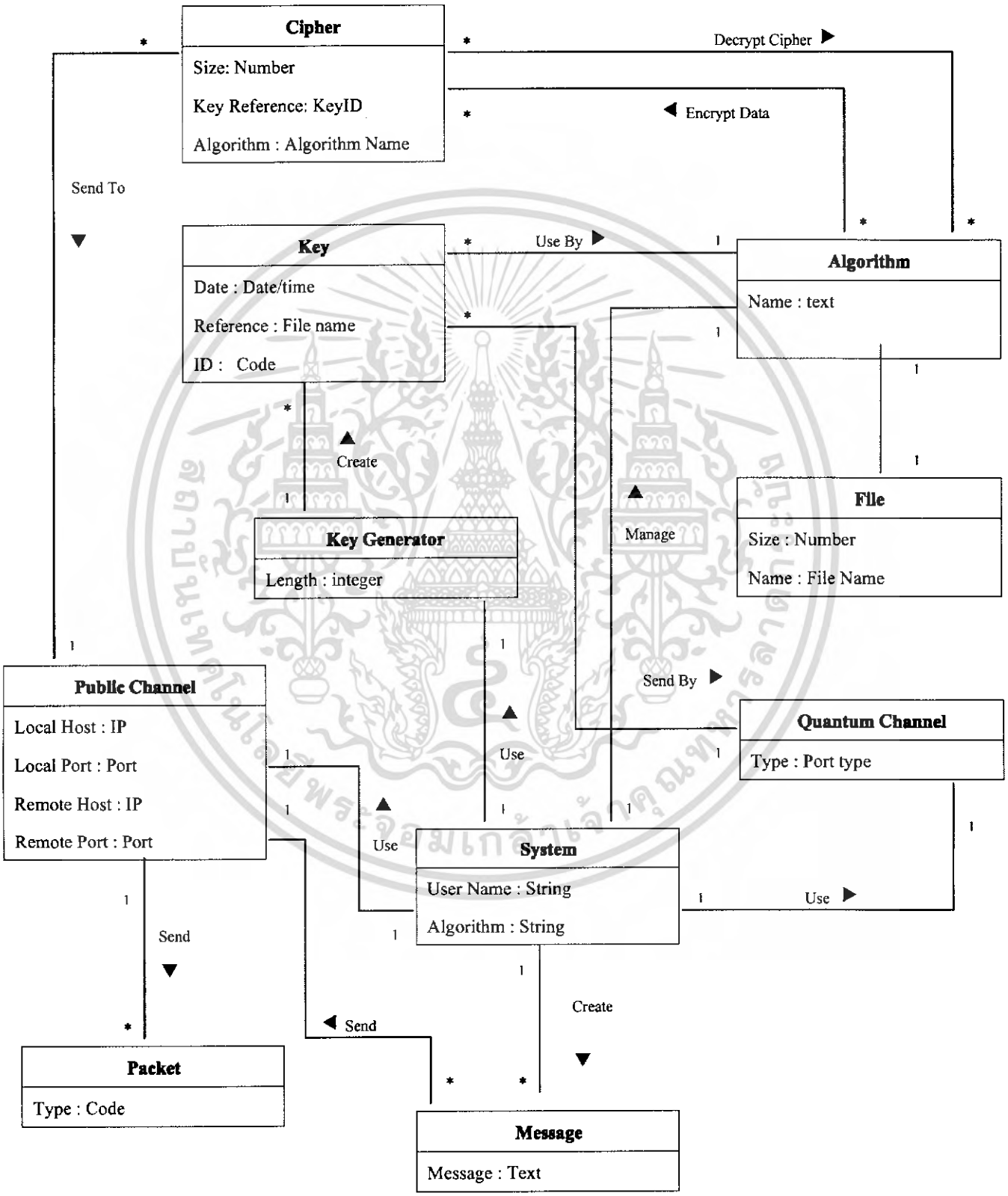
- **Contract CO: Decrypt**

Operation: Cross	Decrypt()
References:	Use Case: Decrypt Cipher
Preconditions:	- ได้รับข้อมูล Cipher ครบถ้วน - ได้รับคีย์ที่ตรวจเช็คความถูกต้อง และมีความผิดพลาดน้อยกว่า 25% - Select file() ที่ต้องการถอดรหัส - ตรวจสอบคีย์เพื่อใช้ในการถอดรหัส
Post conditions:	- ระบบจะทำการถอดรหัส Cipher - ระบบจะแจ้งเตือนเมื่อถอดรหัสสำเร็จหรือFailed และสามารถ Preview ดูข้อมูลได้ - ระบบจะทำการลบ Cipher และ key ที่ใช้ในการถอดรหัสทิ้ง

- **Contract CO: Shutdown Server**

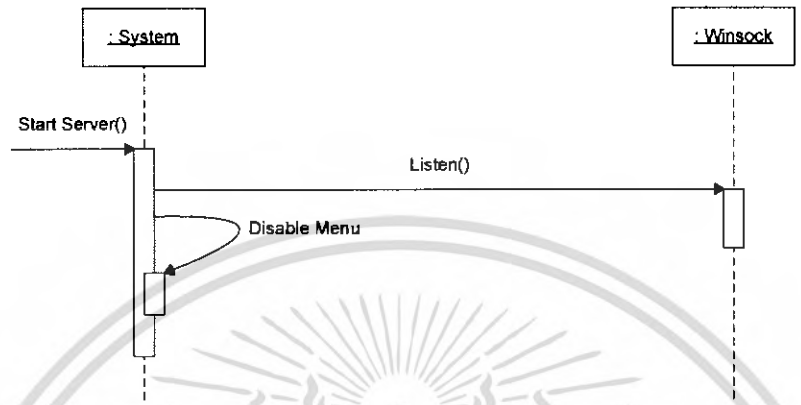
Operation: Cross	Shutdown Server()
References:	Use Case: Shutdown Server
Preconditions:	System Connected
Post conditions:	- ระบบจะทำการปิด Connection ทั้ง Public Channel และ RS- 232 - Exit Program

8. Domain Model

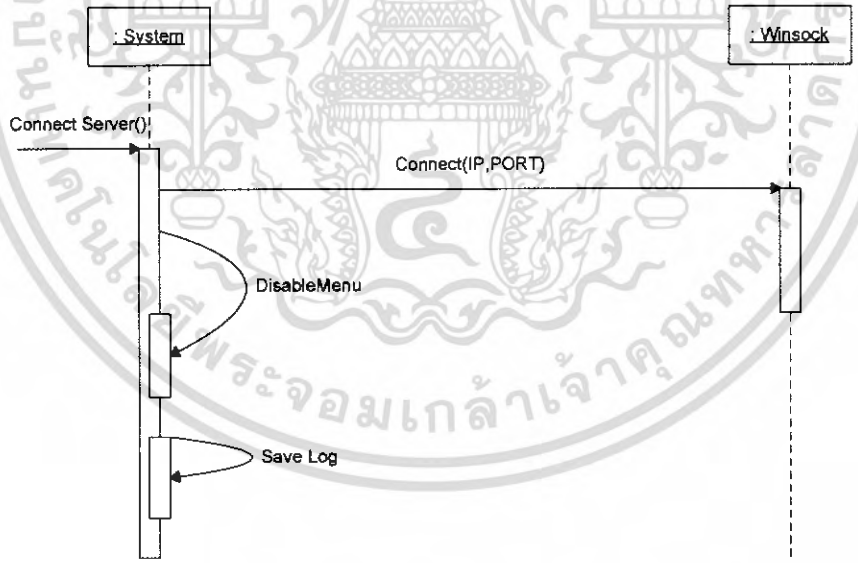


9. Interaction Diagram and Use Case Realizations

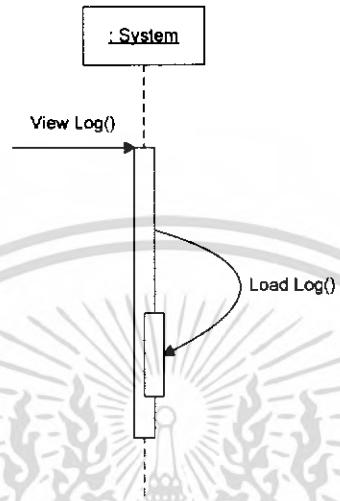
- **Interaction Diagram for Use Case Start Server**



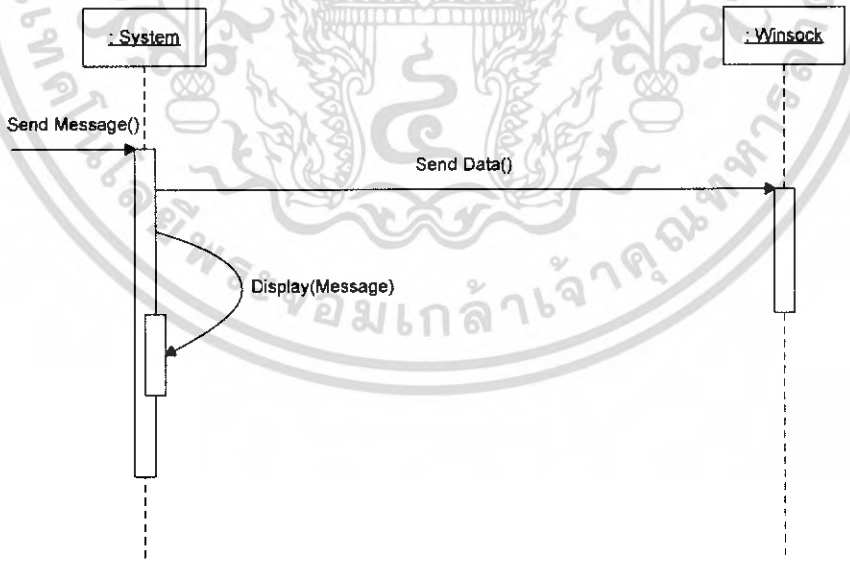
- **Interaction Diagram for Use Case Connect Server**



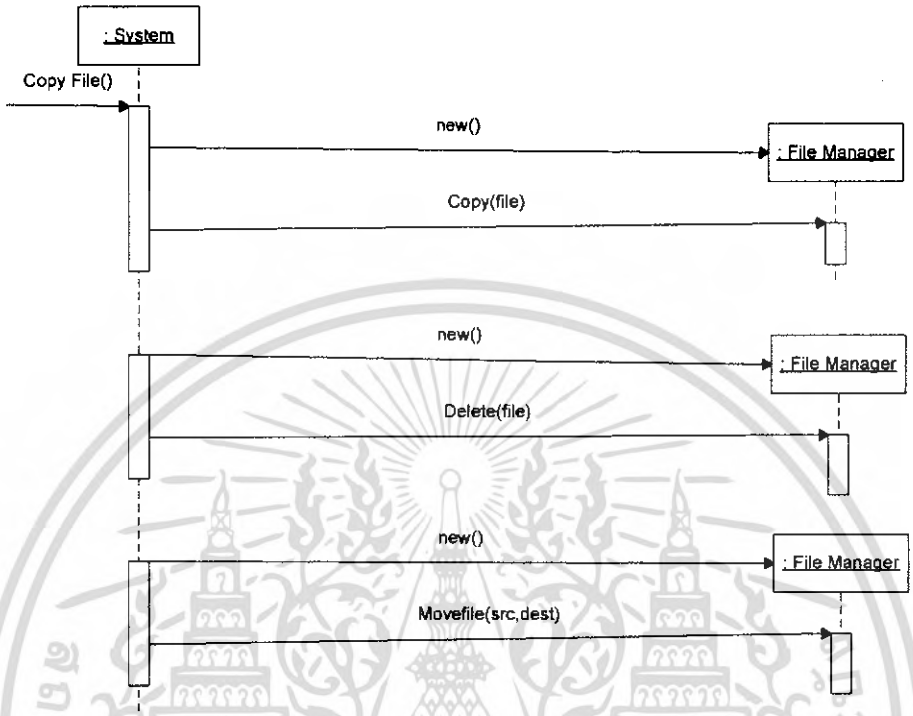
- **Interaction Diagram for Use Case View Log**



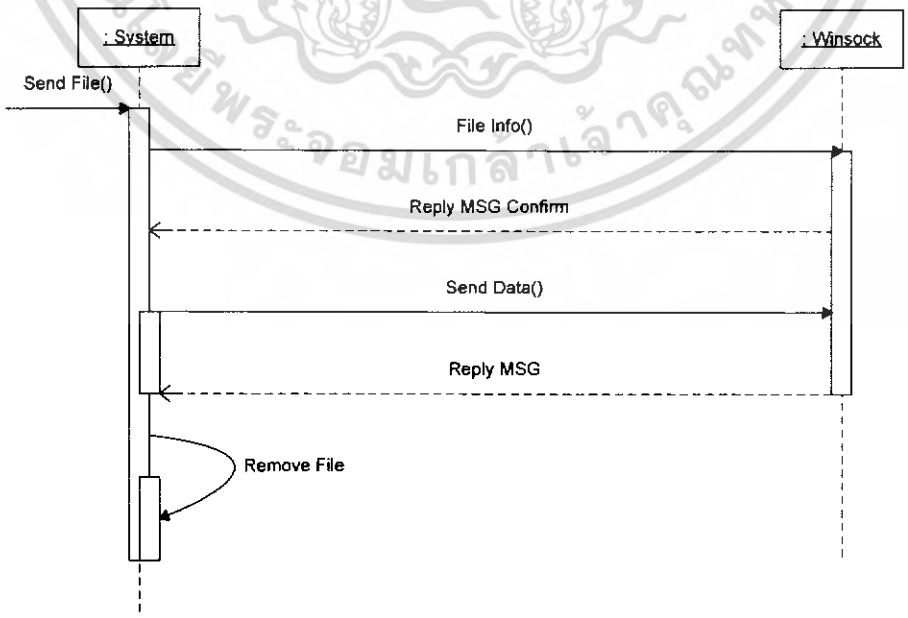
- **Interaction Diagram for Use Case Send Message**



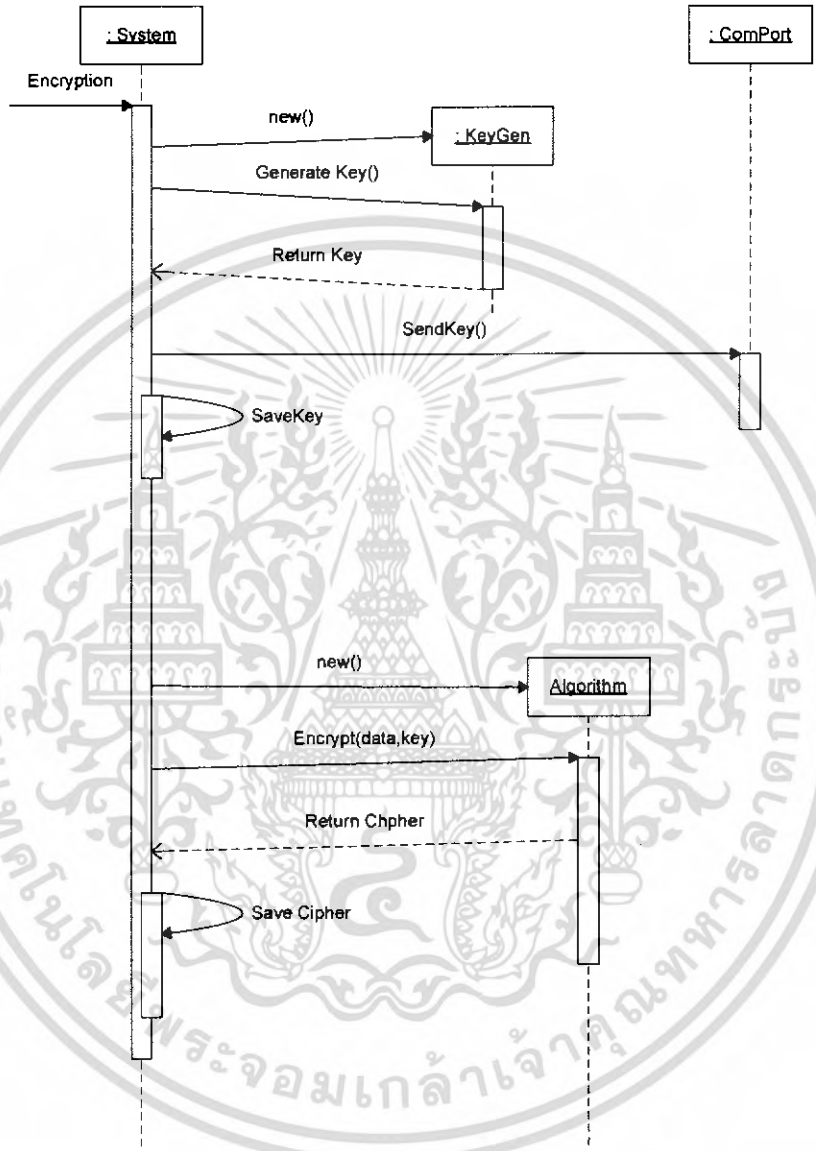
- **Interaction Diagram for Use Case Manage file**



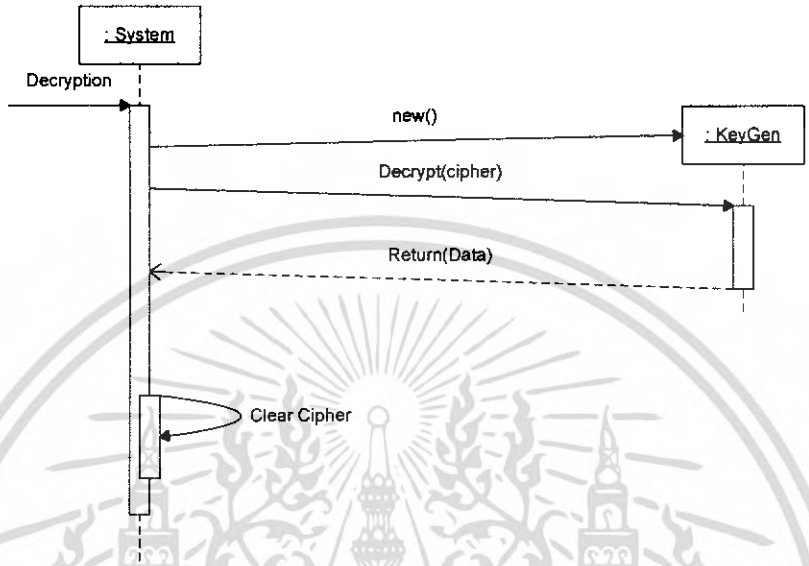
- **Interaction Diagram for Use Case Send file**



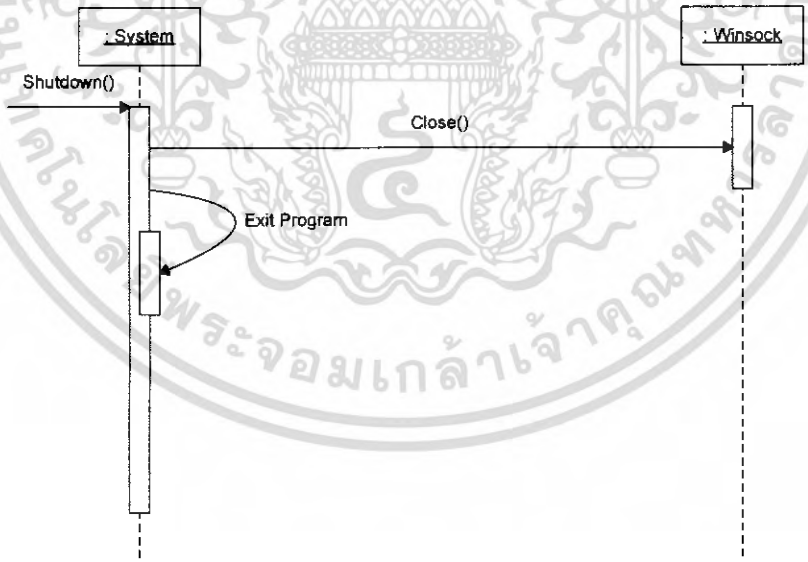
- **Interaction Diagram for Use Case Encryption**



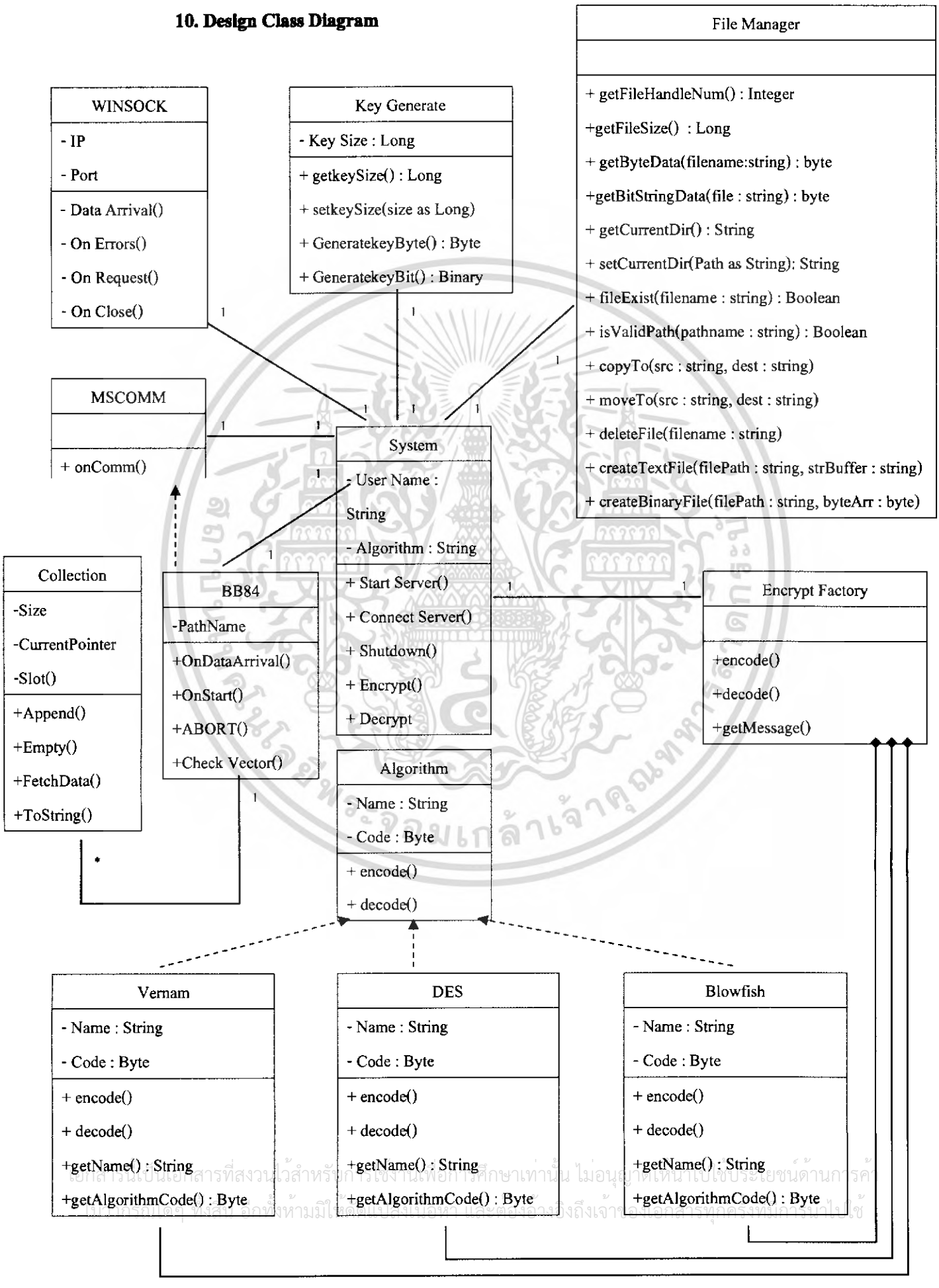
- **Interaction Diagram for Use Case Decryption**



- **Interaction Diagram for Use Case Shutdown Server**



10. Design Class Diagram



ข่าวสารความเคลื่อนไหวของการพัฒนาระบบวิทยาการรหัสลับเชิงควอนตัม

นอกเหนือจากโครงงานการพัฒนาซอฟท์แวร์จำลองการใช้รหัสลับเชิงควอนตัมแล้ว ศูนย์เทคโนโลยีอิเล็กทรอนิคส์และคอมพิวเตอร์แห่งชาติ (NECTEC) ซึ่งเป็นผู้พัฒนาโครงงานวิทยาการรหัสลับเชิงควอนตัมซึ่งเป็นโครงงานหลัก ก็ได้ทำการพัฒนาอุปกรณ์ทางด้านฮาร์ดแวร์ควบคู่กันไป ซึ่งมีรายละเอียดของอุปกรณ์ต่าง ๆ ดังนี้

1. Laser Driver

เลเซอร์ไดรเวอร์เป็นอุปกรณ์ที่ใช้สำหรับขับ เลเซอร์ไดโอด เนื่องความสว่างของเลเซอร์ไดโอด จะแปลตามกระแสไฟฟ้าที่จ่าย ดังนั้น เมื่อต้องการให้ความสว่างของเลเซอร์ไดโอดมีความถูกต้อง จึงต้องจ่ายกระแสไฟฟ้าที่มีความเสถียรให้กับเลเซอร์ไดโอด ซึ่งอุปกรณ์นี้จะพัฒนาไปสู่อุปกรณ์การยิงแสงเพื่อเข้าสู่ช่องสัญญาณควอนตัม เพื่อทำการส่งข้อมูลกุญแจไขความลับต่อไป

2. Quantum Random Generator

เครื่องกำเนิดจำนวนสุ่มเชิงควอนตัม เป็นเครื่องมือที่ใช้กำเนิดจำนวนสุ่ม โดยอาศัยวิธีทางควอนตัม ซึ่งในปัจจุบัน เป็นวิธีการเพียงอย่างเดียวที่สามารถยืนยันได้ว่าจำนวนที่ได้มาเป็นจำนวนสุ่มอย่างแท้จริง ซึ่งเป็นอุปกรณ์ที่จะนำไปใช้ในการสุ่มข้อมูลคีย์เพื่อใช้ในการเข้ารหัสข้อมูลของระบบวิทยาการรหัสลับเชิงควอนตัมต่อไป

3. Quantum Cryptography System

วิทยาการรหัสลับเชิงควอนตัม เป็นการใช้วิธีการกระจายกุญแจ (key) สำหรับเข้ารหัส และถอดรหัสข้อมูล ซึ่งยืนยันความปลอดภัยของกุญแจโดยอาศัยคุณสมบัติทางควอนตัม โดยขณะนี้ทางศูนย์เทคโนโลยีอิเล็กทรอนิคส์และคอมพิวเตอร์แห่งชาติ (NECTEC) กำลังพัฒนาระบบโดยวางแผนและพัฒนาโดยแยกเป็นส่วนย่อย ๆ แล้วทำการรวบรวมและประกอบชิ้นงานเพื่อเป็นโครงงานหลักขึ้นมา